

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

INFORMATION MANAGEMENT SYSTEM FOR ELECTRONIC
VOTING IN SUPPORT OF THE SCHIEFFELIN AWARD
FOR EXCELLENCE IN TEACHING

by

Samuel A. Magliano

September 2001

Thesis Advisor:
Co-Advisor:

Daniel R. Dolk
Dennis R. Mar

Approved for public release; distribution is unlimited.

Report Documentation Page		
Report Date 30 Sep 2001	Report Type N/A	Dates Covered (from... to) -
Title and Subtitle Information Management System for Electronic Voting in Support of the Schieffelin Award for Excellence in Teaching	Contract Number	
	Grant Number	
	Program Element Number	
Author(s) Magliano, Samuel A.	Project Number	
	Task Number	
	Work Unit Number	
Performing Organization Name(s) and Address(es) Research Office Naval Postgraduate School Monterey, Ca 93943-5138	Performing Organization Report Number	
Sponsoring/Monitoring Agency Name(s) and Address(es)	Sponsor/Monitor's Acronym(s)	
	Sponsor/Monitor's Report Number(s)	
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes		
Abstract		
Subject Terms		
Report Classification unclassified	Classification of this page unclassified	
Classification of Abstract unclassified	Limitation of Abstract UU	
Number of Pages 400		

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2001		3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE Information Management System for Electronic Voting in Support of the Schieffelin Award for Excellence in Teaching			5. FUNDING NUMBERS	
6. AUTHOR (S) Magliano, Samuel A.				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the U.S. Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The purpose of this research was to evaluate, automate, refine and develop a management information system that facilitated data collection, organization, query, analysis, and counting of ballots submitted over the Internet/Intranet with regard to the Rear Admiral John J. Schieffelin Award for Excellence in Teaching. Research included conducting a detailed analysis of the current system (reengineering), preparing hardware and software requirements for an automated system, determining security requirements for an Intranet based voting system and implementing a prototype to demonstrate feasibility. The "Rear Admiral John J. Schieffelin Award for Excellence in Teaching", has traditionally been a manual system interwoven with several legacy systems that make analysis, voter response and data collection difficult. Development of an Internet/Intranet based Information System coupled with a Decision Support System for statistical analysis, streamlines the flow of information, thus allowing for more robust analysis/querying as well as possibly increasing voter response by providing a friendly user interface that allows quick and easy ballot submissions.				
14. SUBJECT TERMS Database, Schieffelin Award, Data Model			15. NUMBER OF PAGES 400	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution unlimited

INFORMATION MANAGEMENT SYSTEM FOR ELECTRONIC VOTING IN
SUPPORT OF THE SCHIEFFELIN AWARD FOR EXCELLENCE IN TEACHING

Samuel A. Magliano
Major, United States Marine Corps
B.S., Oregon State University, 1989

Submitted in partial fulfillment of the
requirements for the degree of

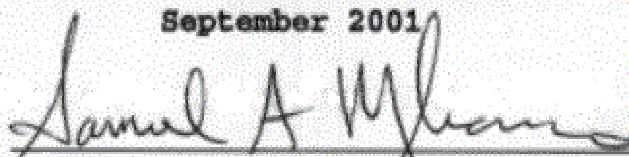
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL

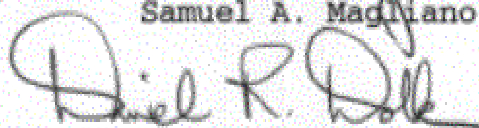
September 2001

Author:



Samuel A. Magliano

Approved by:



Daniel R. Dolk, Thesis Advisor



Dennis R. Mar, Associate Advisor



Dan C. Boger, Chairman
Information Systems Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The purpose of this research was to evaluate, automate, refine and develop a management information system that facilitated data collection, organization, query, analysis, and counting of ballots submitted over the Internet/Intranet with regard to the Rear Admiral John J. Schieffelin Award for Excellence in Teaching. Research included conducting a detailed analysis of the current system (reengineering), preparing hardware and software requirements for an automated system, determining security requirements for an Intranet based voting system and implementing a prototype to demonstrate feasibility. The "Rear Admiral John J. Schieffelin Award for Excellence in Teaching", has traditionally been a manual system interwoven with several legacy systems that make analysis, voter response and data collection difficult. Development of an Internet/Intranet based Information System coupled with a Decision Support System for statistical analysis, streamlines the flow of information, thus allowing for more robust analysis/querying as well as possibly increasing voter response by providing a friendly user interface that allows quick and easy ballot submissions.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	BACKGROUND	1
B.	OBJECTIVES	4
C.	METHODOLOGY AND SCOPE	4
D.	STRUCTURE	5
II.	IMPROVING THE TEACHING AWARD PROCESS (DEVELOPING A PLAN TO IMPROVE)	7
A.	IDENTIFYING A NEED FOR IMPROVEMENT	7
B.	DOCUMENTING THE CURRENT PROCESS	9
C.	CREATING A VISION OF THE NEW PROCESS	23
D.	DEFINING THE SCOPE OF THE IMPROVEMENT PROCESS	24
III.	DATABASE DESIGN	29
A.	DATA MODELING	29
B.	LOGICAL DATA MODEL	30
C.	PHYSICAL DATA MODEL	31
D.	THE RELATIONAL DATABASE	32
E.	WEB BASED DATABASE ARCHITECTURE	37
	1. Two-Tiered Client/Server Database	37
	2. Three-Tiered Client/Server Database	39
IV.	ADMINISTRATION AND IMPLEMENTATION CONSIDERATIONS	41
A.	SECURITY	41
	1. Server Security	42
	2. User Authentication Security	44
	3. Session Security	46
	4. Software/Hardware Security	46
B.	DATA INTEGRITY	49
	1. Constraints	50
	a. <i>Stored Procedures</i>	51
	b. <i>Triggers</i>	51
	c. <i>Default</i>	51
	d. <i>Primary Key</i>	52
	e. <i>Foreign Key</i>	52
	2. Entity Integrity	53
	3. Domain Integrity	53
	4. Referential Integrity	53
	5. User-defined Integrity	55
C.	PRIVACY	55
D.	USER INTERFACE (UI)	56
V.	PROTOTYPING	59
A.	APPROACHES TO PROTOTYPING	59

B.	PROTOTYPING STEPS	60
C.	THE ADVANTAGES AND DISADVANTAGES OF PROTOTYPING ...	61
D.	THE SADMS EVOLUTIONARY PROTOTYPE	62
1.	Database Views	63
2.	The Ballot Pages	64
3.	The Administrative Pages	72
a.	Chairman and Committee Scores Reports	74
b.	Chairman and Committee History Reports ...	74
c.	Searching for Faculty	75
d.	Computing Eligibility	75
e.	Editing Eligibility Reports	78
f.	Updating Professor History	81
g.	Updating the Weights/Viewing the Ballot ..	81
4.	The Scoring Function	82
VI.	CONCLUSIONS AND RECOMMENDATIONS	85
A.	FEASIBILITY	85
B.	MILESTONES	85
C.	EVOLUTIONARY PROTOTYPE	87
1.	User Interface	87
2.	Validation and Error Correction	88
3.	Internet Access	89
D.	STANDALONE SYSTEM	89
E.	CONCLUSION	90
APPENDIX A.	PROCESS FLOW CHART	91
APPENDIX B.	SAMPLE BALLOT	93
APPENDIX C.	DATA ENTRY INSTRUCTIONS	101
APPENDIX D.	LOGICAL MODEL	109
APPENDIX E.	DATA DICTIONARY	111
APPENDIX F.	WEB PAGE DEPENDENCIES	123
APPENDIX G.	APPLICATION CODE	127
A.	INDEX.ASP	127
B.	BALLOT.ASP	130
C.	BALLOT25.ASP	134
D.	BALLOT3.ASP	143
F.	FAIL.HTML	153
G.	FAIL1.HTML	154
H.	LOGIN.ASP	155
I.	INDEX.HTML	157
K.	INPUT_FORM2.HTML	161
L.	INPUT_FORM3.HTML	162
M.	INPUT_FORM4.HTML	164
N.	INPUT_FORM5.HTML	165
O.	INPUT_FORM6.HTML	191

P.	INPUT_FORM7.HTML	193
Q.	INPUT_FORM8.HTML	194
R.	INPUT_FORM10.HTML	195
S.	INPUT_FORM11.HTML	196
T.	COMPUTE_ELIGIBILITY.ASP	198
U.	COMPUTE_SCORES.ASP	198
V.	DATA_ENTRY_ELIGIBILITY_ALL.ASP	199
W.	DATA_ENTRY_ELIGIBILITY_DEPT.ASP	207
X.	DATA_ENTRY_ELIGIBILITY_LNAME.ASP	216
Y.	DATA_ENTRY_HISTORY.ASP	225
Z.	ELIGIBLE_DETAILED.ASP	233
AA.	ELIGIBLE_SIMPLE.ASP	239
BB.	EMPID.ASP	241
CC.	HISTORY.ASP	248
DD.	HISTORY1.ASP	251
EE.	LNAME.ASP	253
FF.	RANK1.ASP	261
GG.	RANK2.ASP	267
HH.	UPDATE.ASP	272
II.	UPDATE_WEIGHTS.ASP	276
JJ.	WEIGHTS.ASP	281
APPENDIX H.	SCORING REPORTS	287
APPENDIX I.	HISTORY REPORTS	291
APPENDIX J.	QUERY FOR PROFESSOR BY EMPLOYEE ID	293
APPENDIX K.	QUERY FOR PROFESSOR BY LAST NAME	295
APPENDIX L.	STORED PROCEDURES	297
A.	CALL SCORING	297
B.	COUNT VOTES	298
C.	DETERMINE SCHIEFFELIN ELIGIBILITY	327
D.	FINAL SCORE	329
E.	GET LOGIN	331
F.	K VALUES	331
G.	SCHIEFFELIN CREDIT HOURS	336
H.	SCHIEFFEIN INELIGIBLE	337
I.	SCHIEFFELIN RANKS	338
J.	SCHIEFFELIN SCORING	339
K.	SCHIEFFELIN SEGMENTS	356
L.	VIEW ELIGIBLE FACULTY	358
APPENDIX M.	VIEWS	359
A.	ELIGIBLE FACULTY	359
B.	FACULTY COUNT	359
C.	FIRST, SECOND & THIRD CHOICE	360
D.	NUMBER OF SUPPORT STATEMENTS	361
E.	SCHIEFFELIN CURRIC ID	362

F.	SCHIEFFELIN RANKING	362
G.	SCHIEFFELIN FIRST,SECOND & THIRD CHOICE	362
H.	SCHIEFFELIN HISTORY	364
I.	SELECT 1-25	364
APPENDIX N.	ELIGIBILITY REPORTS	369
APPENDIX O.	EDIT PROFESSOR ELIGIBILITY	371
APPENDIX P.	INPUT PROFESSOR HISTORY	375
APPENDIX Q.	VIEWING AND EDITING SCORING WEIGHTS	377
LIST OF REFERENCES		379
INITIAL DISTRIBUTION LIST		383

LIST OF FIGURES

Figure 1.	Waterfall Method of Project Development.	30
Figure 2.	Entity Relationship (ER) Diagram for the SADBMS. .	32
Figure 3.	Two-Tiered Client/Server Architecture.	38
Figure 4.	Three-Tiered Web Database Application.	39
Figure 5.	An Example of Referential Integrity.	54
Figure 6.	The SADBMS Main Web Page.	65
Figure 7.	User Already Voted.	66
Figure 8.	User Not Student.	67
Figure 9.	Failed Logon.	68
Figure 10.	Top of Ballot.	69
Figure 11.	Bottom of Ballot.	69
Figure 12.	Second Page of Ballot.	70
Figure 13.	Third Page of Ballot.	71
Figure 14.	Ballot Summary Page.	72
Figure 15.	Schieffelin Administrator Main Web Page.	73
Figure 16.	Process Flow Chart.	91
Figure 17.	Logical Model	109
Figure 18.	Input_Form1 & Input_From2	287
Figure 19.	Rank1.asp (Chairman Copy)	288
Figure 20.	Rank2.asp (Committee Copy)	289
Figure 21.	History.asp (Chairman History Report).	291
Figure 22.	History1.asp (Committee History Report).	292
Figure 23.	Input_form8.	293
Figure 24.	Empid.asp	293
Figure 25.	Input_form7.	295
Figure 26.	Lname.asp	295
Figure 27.	Eligible_simple.asp (Eligibility Report)	369
Figure 28.	Eligible_detailed.asp (Eligibility Report)	370
Figure 29.	Input_form4.	371
Figure 30.	Data_entry_eligibility_all.asp	371
Figure 31.	Input_form5.	372
Figure 32.	Data_entry_eligibility_dept.	372
Figure 33.	Input_form11.	373
Figure 34.	Lname.asp.	373
Figure 35.	Input_form3.	375
Figure 36.	Data_entry_history.asp	375
Figure 37.	Weights.asp (View Weights)	377
Figure 38.	Weights.asp (Edit Weights).	377
Figure 39.	Update.asp.	378

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Partial Sorting of Professors who Taught by Calendar Year.	10
Table 2.	A Professor and all Courses s/he Taught in a Calendar Year.	13
Table 3.	Eligibility list of professors by department.	13
Table 4.	Professors Sorted by Overall Score/Rank.	17
Table 5.	A Summary of Voter Responses for Fifteen (15) Sample Ballots.	19
Table 6.	Summary of Total Number of Rankings Received by a Professor from 15 Sample Ballots.	19
Table 7.	Summary of Scores Generated from Sample Ballots. ..	21
Table 8.	Paired Comparison Output from TAWARD4.	21
Table 9.	Previous Performance Data from TAWARD5 (Schieffelin History).	23
Table 10.	Options for Enforcing Data Integrity.	50
Table 11.	Data Dictionary	111
Table 12.	Web Page Dependencies	124

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

I would like to thank the PYTHON Group for their gracious support in providing me with a lab and server to host my prototype Web application.

Specifically, I would like to thank James Caroland for his mastery of SQL server and all the coding he did in producing the stored procedures that make up the bulk of the SADBMS.

I would like to thank Vince Garcia for his help with Active Server Pages. His knowledge was instrumental in developing the dynamic Web application portion of this thesis.

Additionally, I would like to thank Dr. Daniel Dolk and Mr. Dennis Mar for their knowledge and guidance as my advisor and associate advisor.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

With increased reliance on the Internet, there is a marked increase in opportunities for moving Information Systems within the Department of Defense (DOD) and throughout the world to the World Wide Web (WWW). The WWW provides DOD with a level of interconnectivity unrealized in the past. The "Rear Admiral John J. Schieffelin Award for Excellence in Teaching", has traditionally been a manual system interwoven with several legacy systems that make analysis, voter response and data collection difficult. Developing an Internet based Information System coupled with a Decision Support System for statistical analysis, will streamline the flow of information, thus allowing for more robust analysis/querying as well as possibly increasing voter response by providing a friendly user interface that allows quick and easy ballot submissions.

Electronic Voting over the Internet is a popular topic in the aftermath of the 2000 presidential elections. There are many issues that must be resolved in order to provide a secure environment. While the Schieffelin Award voting system won't require the same degree of security as a presidential election, there are still several aspects of security that pertain to it, including:

- o Voter Authentication
- o Ballot Secrecy
- o Ballot Integrity

- o Reliable Vote Transport and Storage
- o Prevention of Multiple Voting

The degree to which the system meets these security issues should be at least equal to the level of security that is provided by the current balloting system. The system must be able to determine that Internet voters are who they say they are (authentication) and then once the vote is cast, the system must be able to forget whom the voter was in order to allow for secrecy. Current digital identification and data encryption technologies may provide the secrecy and integrity required to meet system requirements, but in order to increase voter response, the burdens placed on the voter to ensure a required security level must not outweigh the benefits of flexibility and ease of use provided by a Web-based voting system.

There are many excellent teachers at the Naval Post Graduate School, but the Award for Teaching Excellence only recognizes the "best-of-the-best". Created in 1969 by Rear Admiral Robert W. McNitt, the Naval Postgraduate School's Superintendent, the award is given each summer during the Spring Quarter Graduation Commencement Ceremonies to the school's most outstanding instructor. The first award was given in June of 1970. In March of 1972, the award was designated as the Rear Admiral John J. Schieffelin Award. (Scango, 1972, p. 7)

Rear Admiral John J. Schieffelin established a grant to the Naval Postgraduate School allowing for a significant cash award to the recipient. In addition to receiving the

cash award, the recipient's name is engraved on a plaque in the Dudley Knox Library.

Guidelines established by a selection committee are used to determine eligibility for the award. Selection is determined by ballot submitted by the student body and alumni that have graduated from the school within the last three years. Ballots are distributed to the student body through their curricular officers and to alumni via the US Postal system.

The ballot requires voters to:

Identify all candidates on the ballot with whom they are familiar regarding their teaching ability

Identify at least five candidates, but no more than twenty-five for their ballots to be accepted in the voting process.

Identify their top three choices; additionally, there is an opportunity for the voters to make a short statement in support of their first choice.

Supply demographic information concerning their curricular area and whether they are a student or alumnus.

Once returned, data from the ballots are processed by a series of computer programs that attempt to eliminate bias and rank eligible candidates based on data gathered from the ballots. An award committee consisting of appointed faculty members and a committee chairman evaluates the processed balloting data and selects the best recipient.

B. OBJECTIVES

The purpose of this research is to evaluate, automate, refine and develop an information management system that will facilitate data collection, organization, query, analysis, and counting of ballots submitted over the Internet related to the Rear Admiral John J. Schieffelin Award for Excellence in Teaching. The goal is to evaluate the existing system and through reengineering, technical, functional and operational analysis develop a replacement of the current manual system with an Internet based system. It will address issues related to Internet security, privacy and data integrity. Furthermore, it will address the software and hardware (system architecture) requirements and develop a prototype to demonstrate the feasibility of voting for the "Rear Admiral John J. Schieffelin Award for Excellence in Teaching" over the Internet.

C. METHODOLOGY AND SCOPE

In accomplishing the above objectives, a detailed analysis of the existing system is required. This analysis consists of interviewing all stakeholders involved in the past and current award process, a search of all subject matter literature, and review of actual documentation pertaining to the award process itself. This analysis will enable us to construct a detailed diagram of the process, gain an understanding of the information flow, identify necessary requirements for developing a web-based balloting system, and prototype a model that fully integrates with the Naval Post Graduate School's Education Management

System. Lastly, we must determine the degree of security required in implementing a web-based system.

D. STRUCTURE

The remainder of this thesis is structured as follows: Chapter II is an overview of process improvement. It describes the current award process in detail and identifies the need for improvement. It provides the reader with a vision of our web-based balloting proposal and identifies requirements necessary in achieving it.

Chapter III is an overview of the relational database system design. It discusses modeling concepts and provides a high-level conceptual data model for the Schieffelin Award database application.

Chapter IV discusses issues dealing with implementing the web-based balloting system for next year's election process. Security requirements, data integrity and user interface topics are covered in this chapter issues.

Chapter V deals with prototyping and discusses the details of the first prototype balloting model.

Chapter VI consists of final recommendations and conclusions for school-wide implementation.

THIS PAGE INTENTIONALLY LEFT BLANK

II. IMPROVING THE TEACHING AWARD PROCESS (DEVELOPING A PLAN TO IMPROVE)

As a leading advocate of continuous process improvement, W. Edwards Deming believed that "the process of tomorrow's world is continual improvement". This principle is essential in establishing the groundwork for this thesis. This chapter will build upon part one of the "Deming Cycle", developing a plan for improvement. Part one of the Deming Cycle is broken down into four steps:

- Step 1 - Identifying a need for improvement,
- Step 2 - Documenting the current process,
- Step 3 - Creating a vision of the new process, and
- Step 4 - Defining the scope of the improvement process. (Scherkenbach, 1991)

A. IDENTIFYING A NEED FOR IMPROVEMENT

The current Schieffelin Award process has been in place for thirty years and has served the Award Committee well in their task of identifying the most deserving professor. There have been minor changes in how eligible faculty are scored and ranked, but for the most part, the process has remained unchanged (Read, 1995). As with most organizational processes, the Schieffelin Award process is being overcome by technology and the costs associated with maintaining expertise to operate legacy programs on an expensive mainframe computer.

Process Improvement focuses on ways in which process outputs can be produced more rapidly, with higher quality

and at lower cost. It's concerned with developing new and better ways to do things. Certainly an overriding principle to improving the award process is to develop a more efficient, higher quality process with lower costs. As the current Award process is described, the reader may find it useful to follow along with the process flow chart found in Appendix A.

Similar to other organizations that exist in today's rapidly changing business environment, the Naval Postgraduate School has recognized the need for information support with minimal expenditures on maintenance, as well as efficient use of systems and people. One example of the schools movement in this direction is the use of the "PYTHON" group to incorporate modern technology and student knowledge in developing a reliable, easy-to-maintain education management system. This management system will provide universal data with dedicated relational databases applying automated business rules that increase data reliability and offer future scalability. All of this functionality will utilize Commercial-Off-the-Shelf software (COTS). (Caroland, 2001)

The PYTHON management system is advantageous for the Schieffelin Award, because much of what is required to transform the archaic award process into a system that takes advantage of current information technology is already included in the school's strategy. This means that enterprise-wide planning is already in motion, the full spectrum of the institution's needs are being considered, and representatives from across the institution are involved in the decision making process.

B. DOCUMENTING THE CURRENT PROCESS

Process improvement offers a structured approach for documenting the current process. Documentation is a major step on the road to improving. When done well, documenting a process enables the current picture to be communicated and analyzed efficiently.

As described earlier, the current award process is interwoven with legacy systems that require interaction with several members of the Naval Postgraduate School staff and outside agencies. The process begins with identifying all eligible faculty members. Past winners of the award, current committee members, the committee chairman and contracted faculty are all ineligible. Eligible candidates must have taught three or more courses and eleven or more quarter hours during the calendar year of the award. The following criteria are used in determining the number of instructional hours a candidate is given credit for:

- o One-for-one credit is given for a normal course of instruction.
- o Proportional credit is given for joint/team teaching.
- o One hour of credit is given for every two hours of laboratory instruction.
- o The same professor must teach laboratory courses associated with a lecture.
- o Reading courses, directed study, lab only and short courses are not counted.

Producing the list of eligible faculty currently requires the effort of several members of the staff. The Department of Instruction manually inputs the header information from the Student Opinion Forms (SOFs) submitted during the calendar year of the award. Once the SOFs are scanned into the system, a list of professors who taught in the calendar year can be produced (Table 1).

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
98203	POLLARD	S	03	AO2020	01	20	1	46
98203	BANK	MH	03	AO2030	01	10	1	46
98203	FIGLOCK	R	03	AO3000	01	21	1	46

Table 1. Partial Sorting of Professors who Taught by Calendar Year.

The first column indicates Academic Year (98), quarter (2nd) and the professor's primary department (03). The second and third columns are the professor's last name and initials. The fourth column is the department offering the course. The fifth column is the course identification number. The sixth column is the course section number. The seventh column is a two-digit number with course hours as the first digit and lab hours as the second digit. The eighth column is the number of professors who taught the course. The ninth and last column is the number of students in the class. Running a sorting job called "TAWARD0" on the mainframe computer produces this list.

Once TAWARD0 has been run, its output is verified. A member of the staff in the Graduate School of Business and Public Policy (GSBPP) is currently responsible for verifying this data. At this point, the data is

reviewed to resolve issues with synonym courses, variable hours, incorrect names, joint/team teaching, professors who changed departments and professors who changed their last names.

- o Synonym courses occur when the same course is listed more than once in one or more departments under different course identifiers. Students meet in the same classroom at the same time. Both course identifiers will appear on the SOF. Data must be adjusted so that an instructor only receives credit for one of the courses.
- o Variable hours arise in special cases where classes are listed in the course catalog with an unspecified number of class hours. In these cases, the professor must be contacted to verify the number of course hours.
- o Incorrect names occur when names used on the SOF forms differ from names used in the teaching award historical database. In these cases the names in the historical record that differ must be verified for accuracy.
- o Joint/team teaching requires that the credit for the class be proportionally divided amongst the team of professors.
- o An instructor who changed departments or changed his/her last name requires a change to that particular record in the professor's history file.

After verifying the report, the GSBPP staff member receives an updated faculty list from the committee chairman. The flat database files on the mainframe and in the chairman's Schieffelin database only contain records of professors who were eligible for the award in previous years. This means that the committee chairman must request an updated faculty list from the Department of Instruction that includes professors that have arrived or departed in the last year. Professors who are eligible for the first time must be added to the Schieffelin Database. Those professors added are assigned an identification number that is unique only to the Schieffelin Award. Professors who have departed are deleted from the Schieffelin Award Database. Currently, identification numbers for departed professors are deleted as well. This means that it is possible for a new professor to receive an identification number that has been used before, resulting in erroneous data.

The Schieffelin professor ID consists of four (4) digits. The last digit is a check digit to catch input errors and is determined by the vector "351". If the number 243 signified the first three digits of the ID the last digit would be determined by multiplying the vector 243 by the vector 351 as follows:

$$\begin{array}{r}
 3 \\
 243 \times 5 = (2)(3) + (4)(5) + (3)(1) = 29 \\
 1
 \end{array}$$

Resulting in a four digit ID 2439.

Once the chairman and GSBPP staff member are confident that all eligible professors have been added to the database, the following two reports are given to the committee.

```

*****DEPARTMENT CODE 67*****
Professor - AGRAWAL      B.
      (1)      (2)      (3)      (4)      (5)
      AA3101-00      (2-2)      Q-2      AY-97      10 STUDENTS
      AA3251-00      (1-2)      Q-3      AY-97      14 STUDENTS
      AA3251-00      (4-0)      Q-3      AY-97      11 STUDENTS
      AA3251-00      (2-2)      Q-4      AY-97      17 STUDENTS
*****
Professor - BIBLARZ      O.
      AA3804-00      (3-0)      Q-2      AY-97      14 STUDENTS
      AA3851-00      (3-2)      Q-4      AY-97      13 STUDENTS
      AA4505-00      (3-2)      Q-1      AY-98      07 STUDENTS

```

Table 2. A Professor and all Courses s/he Taught in a Calendar Year.

```

*****DEPARTMENT CODE 67*****
(1)      (2)      (3)      (4)      (5)      (6)      (7)
AGRAWAL      B      4.0      12.0      0 LABS      0 READING
BALDWIN      C      2.0      7.0      0 LABS      0 READING      XXXX
BALL      RE      4.0      17.5      0 LABS      0 READING
BIBLARZ      O      3.0      11.0      0 LABS      0 READING
CHANDRASEKHAM      .5      2.0      0 LABS      0 READING      XXXX

```

Table 3. Eligibility list of professors by department.

Both reports are broken down by department code and distributed to the appropriate committee members to validate. Table 2 summarizes the courses and labs the professor taught during the calendar year, the course hours, the academic quarter, the academic year and the number of students in the class. Table 3 summarizes the teaching load for each instructor. Columns 3 and 4 are the number of segments and the number of credits taught during the calendar year, where one lecture hour is worth one credit and one lab hour is worth half a credit. Column 7 of Figure 3 is the eligibility column. "XXXX" means that the instructor is not eligible. If the column is "blank" the instructor is eligible. "???" is used when the professor's eligibility is close and care should be taken to examine his/her record. The chairman gives the committee 7-10 days to validate the lists and return them. Once all the corrections are made. The list is sent to the Naval Postgraduate School's Department of Instruction and the ballots are constructed.

Ballots are distributed to the students through their curricular officers. Students return the ballots by dropping them off in ballot boxes located in the curriculum offices. Alumni ballots are distributed through the mail along with return envelopes. The Department of Instruction queries the FOCUS database for alumni that have graduated within the last three (3) years. The school currently does not maintain an accurate mailing address for graduates. A disk file with all of the alumni on it is sent via guard mail to the Defense Manpower Data Center (DMDC) at the Presidio of Monterey (POM) Annex. The DMDC queries the Defense Enrollment Eligibility Reporting System (DEERS) for

alumni addresses. DEERS is a worldwide database of military sponsors, families and others who are covered by TRICARE. Addresses not found in DEERS are searched for in the Active Duty Master File for their current Unit Identification Code (UIC). DMDC receives monthly updates to their databases from the Service Personnel Centers, so their information is relatively current. Ballots not received by the announced deadline are not counted.

The Department of Instruction collects the ballots, assigns a ballot ID and verifies that the ballots are valid. The ballot ID is a sequential unique number ending with either an "A" for alumni or "S" for student as indicated by the voters response on the ballot. An example ballot is included as Appendix B. The ballot must have at least five (5) professors identified (circled) and no more than twenty-five (25). If a voter put the same professor for his first, second and third choice, the second and third choice votes are ignored. An image of each ballot is created on the mainframe using an interactive program (The process of coding the ballot information into a computer file is done with an interactive program running on the mainframe time sharing system.). The complete instructions for entering ballot information utilizing this interactive program are included as Appendix C.

After the ballot images have been created, there are several programs on the mainframe that score the data based on variables and weights that have been adjusted over the years to eliminate bias. For the purposes of this thesis, we will only address four of those programs in detail. We have selected them on the basis that, if a new award

process can duplicate the outputs of these four programs, it will provide all the functionality of the current system. The programs omitted, TAWARD2 and TAWARD6, are merely queries (sorting) of data that can be determined from the other programs.

Program TAWARD1 checks for transposition errors. If a professor's ID is entered incorrectly, the program utilizes the check digit in the professor's ID to validate proper entry. The program generates an error report listing all ID's that are incorrect. These ID's must be compared to the original ballots to determine whether the voter entered an erroneous ID or whether the operator entering the ballot image entered it incorrectly. If the voter listed an erroneous ID, it is discarded. For example, if an erroneous ID number was written in as the voter's first preference, the first preference would be discarded. However, if the voter's second and third preferences were valid ID's, they would still be counted. If it was merely a transposition error by the operator, the error is corrected.

Program TAWARD3 scores all the ballots and creates a report sorted by score (highest to lowest). Table 4 is a partial report generated by TAWARD3. There are actually two copies of this report: One copy for the chairman that shows the professor's name and another copy that only shows the professor's identification number. This prevents the committee from knowing who the candidates are and allows them to make their decision based solely on performance and not by who the professor is. Only after the final decision is made does the committee learn the name of the winner.

RANK	SCORE	N	D	X1	X2	X3	Z1	Z2	Z3	Z4
1	6.50	28	10	17	6	2	15.21	4.87	1.3	.699
2	5.97	20	5	12	3	2	10.53	2.42	1.44	.823
3	5.39	74	11	34	17	6	28.53	11.53	3.23	3.98
4	5.13	77	11	27	19	12	23.94	14.89	8.07	4.98
5	4.99	17	4	9	2	1	7.844	1.417	.667	1.33

Table 4. Professors Sorted by Overall Score/Rank.

S is the professors overall score. It is determined by Equation (1):

$$S = \left(\sum_i^4 w_i z_i \right) \div N^p \quad (1)$$

Where w_i are the assigned weights (7,4,3,1). A professor receives a 7 for a first place vote, a 4 for a second, a 3 for a third and a 1 for being identified on a ballot. These weights have been stable over the last several years. The constant p is set at .9 and is used to eliminate bias that can be created because professors can be observed by a wide range of students. The weights and "p" value were discovered while reading the code used to generate scores for the current balloting system. Past debates considered that high exposure to students could detract from a professor's ability to generate a competitive score. "p" is used to compensate for this. N is the number of ballots on which the professor was identified (circled). D is the number of comments submitted on the professor when she/he was identified as a voter's first choice. x_1 , x_2 and x_3 are the number of times

the professor was identified as a first, second or third choice. Although not shown in Table 4, x_4 is the number of times a professor was circled but not marked as a first, second or third choice. It is determined by subtracting x_1 , x_2 and x_3 from the total number of ballots the professor was identified on (N). z_1 , z_2 and z_3 are the sum of scores for all first, second and third place choices. z_4 is the sum of scores for all times the professor was identified (circled) on a ballot but not marked as one of the top three choices. The z_i variables are calculated with regard to how many professors were identified on a ballot (k). (Read, 1995)

$$z_1 = k/(k+1) \quad (2)$$

$$z_2 = (k-1)/(k+1) \quad (3)$$

$$z_3 = (k-2)/(k+1) \quad (4)$$

$$z_4 = (k-3)/2(k+1) \quad (5)$$

For a single ballot submission, with 15 faculty identified (k), a professor who ranked first would receive a score of $15/(15+1) = 15/16$. If this were the only ballot for which that professor received a first place ranking, then that value would be his z_1 value. However, if that professor received more first place rankings, his z_1 score would be the sum of all of them. This process is repeated for all z_i .

Table 5 summarizes professor rankings for fifteen (15) sample ballots. The professors are denoted as A, B and C. Under normal balloting conditions, there would be many professors and a few hundred ballots. The reason some columns/rows are left blank is because neither professors A, B or C were ranked nor identified on that ballot. On

ballot 4 for example, instructor C was ranked first, instructor B was ranked second and instructor A was identified but not ranked.

BALLOT Number	Ranked First	Ranked Second	Ranked Third	Identified	k
1	A		B	C	20
2	B			A, C	5
3	A	C	B		15
4	C	B		A	25
5					10
6					15
7	A	C	B		12
8	B	C	A		9
9	A		C		15
10	B			C	7
11		B			18
12	C		B	A	22
13					20
14		A		B, C	17
15					24

Table 5. A Summary of Voter Responses for Fifteen (15) Sample Ballots.

Table 6 depicts the number of ballots on which a professor received a 1st, 2nd and 3rd place rankings (x_i) on as well as the number of ballots the professor was identified on but not ranked (also referred to as 4th place ranking). N is the total number of ballots the instructor was both ranked and identified on.

Professor	x_1	x_2	x_3	x_4	N
A	4	1	1	2	8
B	3	2	4	1	10
C	2	3	1	2	8

Table 6. Summary of Total Number of Rankings Received by a Professor from 15 Sample Ballots.

From the information presented above, the scores for professors A, B and C can be computed. First we determine the professor's score for each 1st, 2nd, 3rd and 4th place ranking using equations 2 thru 5 respectively.

(From Equation 2)

$$Z_{1A} = (K_1 / (K_1 + 1)) + (K_3 / (K_3 + 1)) + (K_7 / (K_7 + 1)) + (K_9 / (K_9 + 1))$$

$$Z_{1A} = (20 / (20 + 1)) + (15 / (15 + 1)) + (12 / (12 + 1)) + (15 / (15 + 1))$$

$$Z_{1A} = 3.75$$

(From Equation 3)

$$Z_{2A} = ((K_{14} - 1) / (K_{14} + 1))$$

$$Z_{2A} = ((17 - 1) / (17 + 1))$$

$$Z_{2A} = .888$$

(From Equation 4)

$$Z_{3A} = ((K_8 - 2) / (K_8 + 1))$$

$$Z_{3A} = ((9 - 2) / (9 + 1))$$

$$Z_{3A} = .700$$

(From Equation 5)

$$Z_{4A} = (K_2 - 3) / (2 (K_2 + 1)) + ((K_4 - 3) / (2 (K_4 + 1))) + ((K_{12} - 3) / (2 (K_{12} + 1)))$$

$$Z_{4A} = ((5 - 3) / (2 (5 + 1))) + ((25 - 3) / (2 (25 + 1))) + ((22 - 3) / (2 (22 + 1)))$$

$$Z_{4A} = 1.003$$

Where z_{1A} , z_{2A} , z_{3A} and z_{4A} are sum of all the 1st, 2nd, 3rd and 4th place ranking scores for instructor A and k_i is the score for the ballot being considered (i.e. k_{14} is the k score [17] for ballot number 14 from Table 5). Since instructor A was ranked first on four ballots (Table 6), there are four values of k used in determining z_{1A} .

By using the professor's Z_i scores and applying the appropriate weights (W_i), the professor's total score can be determined using equation (1).

(From Equation 1)

$$S_A = ((7 \times 3.75) + (4 \times 0.89) + (3 \times 0.70) + (1 \times 1.003)) / 8^9$$

$$S_A = 3.72$$

The remaining professors scores are calculated in the same manner and are summarized in Table 7.

INSTRUCTOR	S	Z_1	Z_2	Z_3	Z_4
A	5.06	3.75	0.89	0.70	1.003
B	5.52	2.61	1.82	3.31	0.39
C	4.19	1.92	2.52	0.81	1.28

Table 7. Summary of Scores Generated from Sample Ballots.

Program TAWARD4 computes a paired comparison score (PC score). Table 8 is a partial report generated by this program and is sorted by P.C. Score. The table is in order of P.C. score. The "Rank" column is based on the

P.C. RANK	RANK	TOTAL RANK	N	P.C. SCORE	% STRESS
1	1	2	28	.729	6.453
2	16	18	9	.529	1.696
3	6	9	18	.475	1.837
4	2	6	20	.472	3.912
5	4	9	77	.348	7.289

Table 8. Paired Comparison Output from TAWARD4.

professor's total score as computed in the sample ballot example. The "Total Rank" column is the sum of the "P.C. Rank" and "Rank" columns. The P.C. Score is the calculated paired comparison score. To determine the PC score, a 60 x 60 matrix is used. The committee was interested in how one professor ranked against another professor only on ballots that identified both of them. When two professors appear on the same ballot, they are scored based on what professor is preferred over the other. The preferred instructor (ranked higher on the ballot) receives a value of one (1). Values of $\frac{1}{2}$ are assigned for ties (both instructors identified but not ranked). A program written in "S-PLUS" based on the Bradley-Terry model then determines the scores. The paired comparison test has been helpful in past election years, but in most cases there were not enough ballots that identified the same pairs professors (Read, 1995). The last column, % Stress, is no longer used.

In examining Table 8, the professor who had the highest Score (S) also had the highest paired comparison score. One noticeable problem with the P.C. ranking is the mixing up of the original rankings, making it difficult to interpret. For this reason, the committee relies more heavily on the basic scoring information (S, N, and x's) and the historical data in making their selection (Read 1995). The paired comparison scores are used in determining the professors who fall out in the top five (5) and fifteen (15) percent.

Program TAWARD5 prints the Schieffelin voting history. Table 9 is a partial report of this program. This report

shows the scores of the current year and seven years of history (1994, 1993 and 1992 data omitted from Table 6

1999						1998		1997		1996		1995	
Rank	N	D	Score	P.C. Rank	STR No	N	Code	N	Code	N	Code	N	Code
1	28	10	6.50	1	1	32	A	38	B	42	E	36	E
2	20	5	5.97	4	1	14	B	18	A	26	B	43	A
3	74	11	5.39	17	1	30	A	26	E	79	E		I
4	77	11	5.13	5	1	58	A	25	B				
5	17	4	4.99	10	1	15	A		I		E		E
6	18	2	4.88	3	1	30	A	32	A	32	A	30	E

Table 9. Previous Performance Data from TAWARD5 (Schieffelin History).

A = Professor finished in top 5% during that year.
 B = Professor finished in top 15% during that year.
 I = Professor was not eligible for the award during that year.
 E = Professor was eligible for the award during that year.

to save space). The Committee uses the previous performance data to see if a professor is demonstrating a consistent approach to teaching excellence.

Program TAWARD7 is used to update the Schieffelin history file. Once the winner is determined and the professor rankings are final, the top five (5) and fifteen (15) percentile are identified and updated in the history file. Only the past seven (7) years of data is kept.

C. CREATING A VISION OF THE NEW PROCESS

Now that the current process has been outlined, let's return to the fundamental issue: Which is the appropriate design to transform this into an automated web-based system? Web based decision support systems have become a powerful tool for organizations to manage their data. The current Schieffelin Award balloting process has not taken

advantage of this technology. Human intervention and manual processes intertwined with legacy systems create an environment where errors can be easily introduced to the system.

Moving the balloting process to the web creates a dynamic environment that is intuitive, easy to use and easy to maintain. When the Award process is based on a relational database model vice the current "flat file" management system, a significant improvement in data storage, management, access and control are gained. A simple query of eligible professors will create a dynamic ballot that will collect voter responses and store them in the relational database. Built-in business rules based on the current model will be used to calculate the results and print required reports that will facilitate the Schieffelin committee's decision making. This will also simplify the process of applying different voting criteria for selecting the winner should the faculty decide this is appropriate.

D. DEFINING THE SCOPE OF THE IMPROVEMENT PROCESS

When examining an existing information system or analyzing the information that is going to be designed, it is important to:

- o Recognize what the data is,
- o Where the data comes from,
- o How it passes from one point to another within the information system; and finally
- o How the intended audience or user will use it.

Starting with an existing process certainly makes it easier than starting from scratch when trying to examine data requirements. Determining these requirements was accomplished by interviewing past and present committee members, interviewing faculty members who participated in data entry, collection and management, and by examining how current reports were being generated. By accomplishing this, the following general requirements were determined:

- o The system will be web based and allow for both student and alumni voting.
- o The system must provide privacy, security and authenticity equivalent to the current balloting system.
- o Development process must be able to be repeated (provide good documentation).
- o The system must utilize hardware and software technologies currently available on campus.
- o Database requirements must be developed in conjunction with the PYTHON development team to allow for complete integration and longevity.
- o The balloting system must provide interfaces that are easily used and understood to include:
 - A Windows GUI and browser-based interfaces.
 - Productive functionality (e.g., icons toolbars, default values entered automatically unless overridden and automation of common tasks).
 - Administrative tasks must be easily accessible for the Schieffelin Administrator.

- Interface should be uniform and consistent.
 - Response times must support processing requirements.
 - Maximum use of error control mechanisms to prevent erroneous data input.
 - Make use of built in tool tips for user help.
- o Maximum use must be made of dynamic code in Web Design so that functionality is the same from year to year. (For example, we don't want to have to recode the web-based system each year as the ballot changes).
 - Dynamically create ballot based on relational model.
 - *Professor, Department, Status, Curriculum* are all dynamically created on the web based ballot.
 - o Utilization of Windows NT authentication for student access.
 - o Utilization of Alumni user ID and password for alumni access.
 - o Web Design must meet current NPS and DOD standards.
 - o Must create a unique ballot that captures all the data captured by the current award ballot.
 - o Ballot must not be traceable to the voter who cast the ballot.
 - o System must prevent "ballot stuffing": allowing a voter to vote twice.
 - o System must be able to compute a ranking based on the same business rules used by the current system.

- o System must be able to generate all of the reports that the current system generates:
 - Total score.
 - Paired Comparison Score
 - History Report
 - Breakout of various statistics based on departments, curriculums and voter status.
 - General capability to retrieve ad hoc information based administrator queries/views.
- o Existing history data must be retained in new system.
- o System must provide protection of variables used to produce scores, yet be flexible enough to allow the administrator "web access" when an update is required.
- o System must allow the administrator to generate required reports via the web.
- o The relational database management system must be able to track professor eligibility based on:
 - The number of course hours and segments taught.
 - Whether the professor was a past winner of the award.
 - Whether the professor is on the Schieffelin Committee (including the Chairman).
 - If the professor is contracted faculty.
- o The relational database management system must be able to track Schieffelin historical statistics for up to seven years.

These requirements are the basis for determining the system architecture, hardware requirements and software requirements needed to build a successful web-based balloting system in support of the Schieffelin Award.

Now that the process has been fully documented and its requirements have been identified, we can begin developing the conceptual model for the Schieffelin Award database.

III. DATABASE DESIGN

An important part of database design is conceptual data modeling. There are many data modeling methods. Extended Relational Analysis (ERA), Object Oriented (OO), Object Role Modeling (ORM) and Semantic Object Modeling (SOM) all use the same fundamental Entity Relationship (ER) concepts. This chapter focuses on a traditional approach based on the ER model. Additionally, we consider what client/server architecture is best suited for implementing our model.

A. DATA MODELING

"Data modeling is the activity of discovering and documenting information requirements." (DeAngelis, 2000) The data model can be used to illustrate detailed needs of an organization or the single needs of a software application. While it might not be apparent to the reader at this point, the basic project development methodology being used for this thesis is often referred to as the *waterfall method*. It begins with a Problem Definition, followed by Requirements Analysis, Conceptual Design, Detail Design, Implementation and finally, Testing. The chapters in this thesis have been paralleling this methodology as depicted in Figure 1.

The data model provides a visual representation of the data structures, data and business rules for the database. There are two components to the data model, a logical model and a physical model.

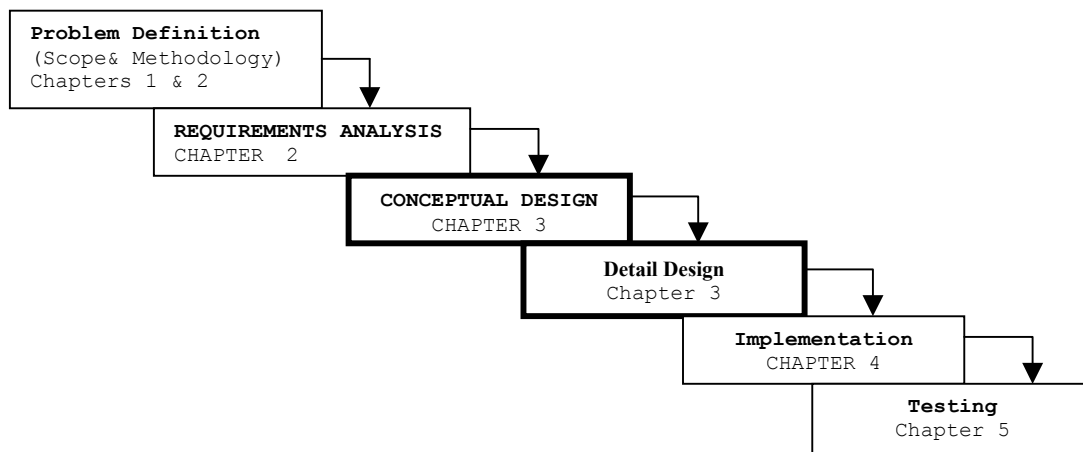


Figure 1. Waterfall Method of Project Development.

B. LOGICAL DATA MODEL

An accurate data model requires detailed analysis of system requirements and business rules. This analysis should result in definitions for all entities, attributes and relationships. During analysis, all metadata (data about the data) should be documented. This analysis was conducted with user involvement through interviews and practical application with the current award process. Additionally, during the analysis, data that was created by combining data elements (derived data) was identified. The derived data deals primarily with combining different ballot data required by the scoring algorithm.

The logical data model represents data structures, data attributes and business rules. The goal of the logical model is to represent the data in a way that is easily understood by both users and database developers. The logical design is not based on any particular platform or design language.

A logical model uses entities, attributes and relationships to represent data and business rules. Entities represent the entities of the Schieffelin Award about which we need to store information. Attributes are the data that describe the entities that we want to model, and relationships define associations between the entities. Appendix D is the logical ER model.

C. PHYSICAL DATA MODEL

The entities, attributes and relationships of the logical model become the tables, columns and integrity constraints of the physical model. After completing the logical model, a decision regarding an appropriate platform can be made. One of the requirements for the Schieffelin Award Database Management System (SADBMS) is that it must make use of existing software/hardware technologies available at the Naval Postgraduate School and that it must integrate with the school's EMS. Since the EMS is based on SQL Server, the SADBMS will utilize the same platform.

The physical model is a graphical representation of the database that will actually be implemented. It contains the tables, columns, keys, and relationships of the database. The physical model will vary depending on the platform selected. The physical model can be used to create views that support user requirements. The physical model consists of the relational tables themselves. The ER model is very similar to the physical model. We will be utilizing the ER model.

D. THE RELATIONAL DATABASE

We based our relational model upon the entities depicted in Figure 2. "Entities are the principal data objects about which information is to be collected; they usually denote a person, place, thing, or event of informational interest." (Teorey, 1994)

The entities `tnpFaculty`, `tnpEmployee`, `tnpDepartment`, `tnpFacultyClass` and `tnpDepartment_Employee` were already modeled for use by the EMS. The SADBMS makes use of these entities by accessing required attributes necessary for managing and processing balloting information. The entities `tnpSchieffelinHistory`, `tnpSchiefVotingHistory`, `tnpSchieffelinBallot` and `tnpSchieffelinWeights` were modeled specifically for the SADBMS. The "tnp" naming convention stands for "table naval postgraduate". The Metadata, Primary keys and entity/attribute definitions are included in Appendix E.

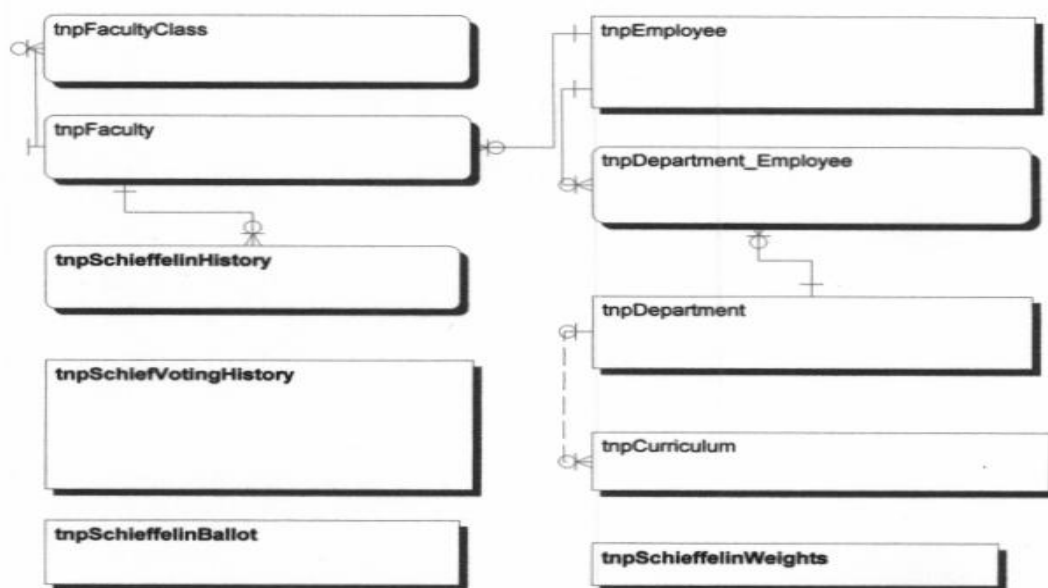


Figure 2. Entity Relationship (ER) Diagram for the SADBMS.

Portions of the SADBMS were reversed engineered from existing historical data. The historical data for the current award process were stored in "flat files" on the FOCUS mainframe computer. While we only encountered minor violations of data integrity, this older structure for managing the data could result in redundant and erroneous data that would leave the balloting process unreliable. Because of this, we have modeled a relational database that is more reliable and provides an acceptable level of data integrity. The relational database represents each object in the database, related or not, as its own database table. The relational model eliminates the need to introduce duplicate data into the database. We use relational database management systems (RDBMS) because of their power and their simplicity (Barman, 1998):

- o Data is presented as a collection of relations,
- o Each relation is represented by a table,
- o Columns of the table are attributes and
- o A "key" uniquely identifies each table and its sets of attributes.

The concept for building a relational model for the Schieffelin Award is centered on "total" integration with the school's movement toward a consolidated Education Management System (EMS). By integrating in this manner, we have assured support from all levels of management, and enhanced longevity and scalability by taking advantage of the leading hardware and software technologies utilized by PYTHON.

Initially, we thought the most significant challenge in designing the new system would be converting data from the old flat file database into the relational database system. There were approximately 450 records, each with several years of historical data to be converted. We were able to get column delineated text files of these records, which allowed us to import the data seamlessly into the new database structure.

The entities identified in Figure 2 are the basic entities required for a web-based balloting system. These entities combined with built in views and stored procedures, provide the foundation for querying, analyzing and computing all functions required to replicate the current balloting process with a web based one. Additional explanation of the views and stored procedures will be provided in the next chapter. We provide a simple description of these entities (see Appendix E for a detailed description of these entities).

The `tnpSchieffelinBallot` entity models the web-based ballot. It provides the same functionality as the current manual ballot. It collects voter demographics as well as all data necessary to complete the scoring function. A unique ballot is created every time a voter submits a ballot via the web based balloting system. A voter can be either an alumnus or a student. The *BallotID* field uniquely identifies the `tnpSchieffelinBallot` entity and is an auto numbered field.

The `tnpSchieffelinHistory` entity holds historical and current year voting history. *CalendarYear* and *EmployeeID* are used to uniquely identify records in this entity. All

professors have history even if it is simply an "I" meaning that the professor was ineligible for the award that year.

The `tnpSchieffelinWeights` entity is used to hold the variables for the scoring function. `CalYear` is the unique identification for this entity. `FirstChoiceWeight`, `SecondChoiceWeight`, `ThirdChoiceWeight` and `SelectedWeight` are equivalent to W_1 , W_2 , W_3 and W_4 under the old balloting system and currently are assigned the values of 7,4,3 and 1 respectively. `Pvalue` is synonymous with "p" under the old balloting system and is currently assigned the value .9. This entity was created to allow the Schieffelin Administrator a more convenient way to monitor the weights, modify them if necessary and track their changes.

The `tnpSchiefVotingHistory` entity is used to track voters during a particular calendar year of the balloting process to prevent users from voting more than once. `VotingYear` and `UserID` are its unique identifiers. `UserId` is the voter's NPGS Domain user ID. `HasVoted` is a Boolean field set to "Yes" after a user submits a vote.

So far, the entities described are unique to the Schieffelin Award and were created to support the balloting process. The following entities were created as a requirement for the Naval Postgraduate School's Education Management System, and the Schieffelin Award Database Management System draws upon these entities to provide the necessary robustness to carry out the balloting process. Some of these entities are complex and require a great deal of explanation to be fully understood. For the purpose of this research, we will only address those attributes relevant to the Schieffelin Award.

The tnpEmployee entity is used by SADBMS to collect attributes about employees (candidates for the award as well as students and alumni). *EmployeeID* is its unique identifier. The attributes: *IsCivilian*, *IsStaff*, *IsStudent*, *IsContractor* and *IsAlumnus* are Boolean fields used for validating user logon into the balloting area of the web-based ballot. The NPGS attribute is used to hold the user ID for the Windows NT user Domain. Other attributes in this entity utilized for the balloting process include: *LastName*, *FirstName* and *MiddleName*.

The entity tnpFaculty is used by the balloting system to determine faculty eligibility. *EmployeeId* is its unique key. *HasWonScheffelin* is a Boolean field used to eliminate previous winners of the award from the current eligible faculty list. *IsContracted* is also used to eliminate contracted faculty from the eligible faculty list. This entity also contains the attribute *YearWonScheffelin* which tracks the year a particular faculty member won the Schieffelin Award.

The entity tnpFacultyClass holds information about what professors have taught what classes. The SADMS uses this entity when producing eligibility reports to show the professor and all the classes he/she has taught during the voting period being considered. *Segment*, *CourseID*, *Quarter*, *EmployeeID* and *Year* are all keys used to uniquely identify records in this entity. A more detailed description of this entity and its keys can be viewed in Appendix E.

The entity tnpDepartment contains all the attributes about a department. The SADBMS uses this entity to create

the ballot. The ballot is organized by department with the eligible faculty for that department listed beneath each department. This appears to be the easiest way for voters to be able to search and find faculty on the ballot with whom they are familiar.

The entity `tnpDepartment_Employee` holds all the departments and what employees are in them. The SADBMS uses this entity to identify eligible faculty by department. Additionally, the attribute `IsPrimaryDepartment` is used to prevent faculty from showing up twice as being eligible when they are listed in more than one department.

The final entity required to complete the basic web-based balloting system is the `tnpCurriculum` entity. This entity is used by the balloting system to allow the voter to select which curriculum he/she was/is a member of while a student at NPS. This information is used as part of the demographics collected about each voter.

E. WEB BASED DATABASE ARCHITECTURE

A web based database application can be implemented with several different technologies, all of which are subject to the preferences of the designer. We considered two different architectures for implementing the SADBMS: the two-tiered client/server architecture and the three-tiered client/server architecture.

1. Two-Tiered Client/Server Database

There are actually three parts of a two-tiered architecture: a client, a server and a protocol. The

protocol connects the client and server. Two-tiered client/server architecture is depicted in Figure 3. The client talks directly to the server. This type of architecture is appropriate for network programming and GUI programs where functionality can be allocated to the host. GUI code is resident (normally) on the host and the business logic is resident on the server. This means that the clients manage the user interface (UI), validate user-entered data, submit requests, execute database transactions and manage data integrity. What this provides is the ability to apply validation and feedback on the client and reduce server overhead and turnaround time. The results are more available network and server resources.

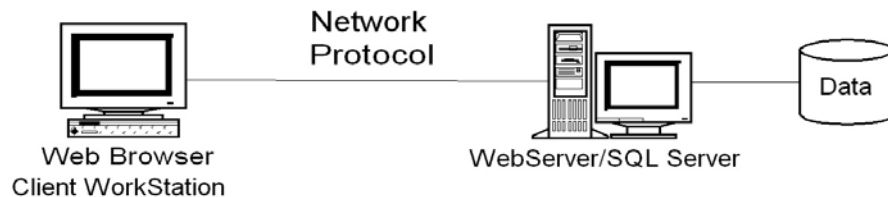


Figure 3. Two-Tiered Client/Server Architecture.

Additionally, the business logic is resident on the server where it is secure and can utilize server resources. A typical two-tiered application is a client running a web browser or GUI written in a high-level language (Java, C++ or Visual Basic) and a web or database server.

There is a distinct separation between the tiers in a two-tiered architecture. The client tier doesn't have to handle data storage or multiple processing requests, and the server doesn't have to worry about any user input validation or interface. The big advantage of this architecture is that with the bulk of the database

processing done on the back-end, the Database Management System is free to work at its own speed because it isn't tied to the processing speed of the client. In a two-tiered environment, the client will still need to store data on the server. This data is usually stored in the file system and can lead to data integrity issues when multiple clients simultaneously request information from the server. This arises because most file systems lack concurrency controls that can lock files while they are being accessed. While the two-tiered architecture works well in a static environment with a fixed set of rules, the three-tiered architecture works well in more dynamic environments.

2. Three-Tiered Client/Server Database

The Schieffelin Database Management System was developed and designed as a three tiered system. A web browser resides on the client tier, the middle tier is a web server and databases specializing in storing, retrieving, and indexing data are added as a third tier. Figure 4 depicts the basic architecture of a three-tiered database application.

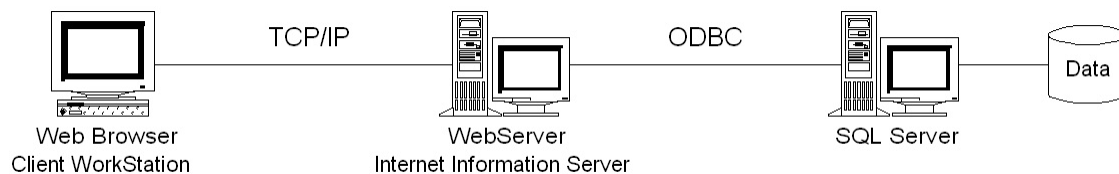


Figure 4. Three-Tiered Web Database Application.

The client browser requests a web page or data from the web server. Within the web server, the request is

converted to a form that the database server can interpret and sent to the database server. The database server performs the transaction (a query, update or insert) and returns the result to the web server where it is sent in a form that the web browser can interpret (an HTML web page for example).

The main advantage of the three-tiered system is that it spreads processing requirements out thus providing greater modularity. Additionally, it reduces network traffic and increases security. Clients and servers in a three-tiered system are often said to be "thin" clients and servers.

Rather than just separating the GUI and business logic, as in the two-tiered architecture, a three-tiered architecture allows for the separation of the business logic and the data access. This allows for optimized data indexing and retrieval. It also provides a means for replications, backup and redundancy.

Another major benefit to a three-tiered architecture is scalability. As the organization grows, the modularity of the three-tiered architecture allows you to add more web servers or database servers.

In the next chapter, we will see how our model and architecture are implemented based on user and functional requirements identified in the previous chapters. We will address the software and hardware requirements for implementation as well as security and data integrity considerations. And finally, we will bring it all together in a prototype web based balloting system.

IV. ADMINISTRATION AND IMPLEMENTATION CONSIDERATIONS

There are many issues that must be considered in administering and implementing a Web-based voting system. In particular, we must consider how data security, data integrity, and privacy are handled, and then determine what hardware/software best meets these security requirements and what type of user interface is required.

A. SECURITY

A major concern for any Intranet/Internet application is security. Security is not one-dimensional. It encompasses both the security of the data within the servers themselves and the security of the data as it travels over the network. The World Wide Web (WWW) is a convenient, cheap and fast way of publishing information. While disseminating this information is relatively easy, it is important to ensure that the information is only accessible to users who we want to allow access.

The Schieffelin Award balloting process is accomplished through the use of dynamically created Web pages that retrieve their data from the Education Management System (EMS). The creation of dynamic Web pages from a database makes information security even more vital. In the past, direct access or access through specialized client software was required to view data within the database. Today, anyone with a Web browser can view data that is not properly protected in the database. The movement from mainframe computers to client/server

networking to the Internet has opened up a plethora of possible penetration points. New security vulnerabilities arise daily. Thus, security procedures and technology are rapidly changing.

When implementing a Web based balloting system, we must address security in three primary areas:

- o *Server Security:* the security of data/HTML files on the server.
- o *User Authentication Security:* ensuring only those authorized to access the system are allowed in.
- o *Session Security:* ensuring that unauthorized users do not view data as it travels over the Intranet/Internet. (Rahmel, 1997)

Each of these areas is separate and should be considered as a separate layer of protection. If there is a weak spot in any one area, the entire system is at risk.

1. Server Security

Server security deals with the protection of data stored on the server. Securing the server should be the responsibility of the network administrator. However, when publishing data to the Web, as in the case of the Schieffelin Award, the Schieffelin Award administrator needs to play an active role in implementing security policy.

Information from a database can be published to a Web site in two primary ways. The first is through the use of

static Web pages and the second is through the use of dynamic Web pages. Since the Schieffelin Award utilizes dynamic Web page publishing, we will focus on securing these types of Web pages.

Dynamic Web page publishing is favored because Web pages are generated instantly as changes to the database are made or as database queries become more detailed. Hypertext Markup Language (HTML) code and a query are stored on the server. There is no actual data stored on the Web server. When a user accesses the page a query automatically executes. The Web server then retrieves the desired data from a database server. The data is displayed in the HTML template for the user to view. The connection between the Web server and the database server is one of the first security issues that a database administrator needs to confront.

In most dynamic connections to databases, full access privileges must be given to the Web server because different queries require access to different tables or views in order to construct the HTML for the Web page. This issue only needs to be addressed by the Schieffelin Award administrator because he is the only user who will be executing different queries. The Schieffelin administrator (SA) is a trusted agent of the process and care need only be taken to ensure that he/she just has access to data within the EMS necessary to carry out tasks associated with the Schieffelin Award. Specifically, the database administrator (DBA) must give permissions to only those tables that require modification by the SA. The only table that the SA requires modification permissions on is the

tnpSchiefflenHistory table. Within this table, the SA must be able to update a candidate's eligibility and eligibility code. This pertains to the IsEligible and EligibilityCode attributes within that table. All other administrative functions require only the execution of queries based on views already built into the database's structure.

Users who access the Web-based balloting system to vote will not be executing any queries. Data for these users, while dynamically generated, is predetermined based on candidate eligibility criteria. They will merely be filling in a form and submitting that information to the Web server, which will in turn send that information to the database server. Users will not be able to upload any scripts or programs that could be utilized to gain access to the database.

2. User Authentication Security

User authentication is the primary security method for ensuring only those users authorized to access the information are allowed into the system. The user requires some form of identification before he/she is granted access. For the Schieffelin Award, we will be able to utilize the Windows NT built in authentication method. Each student already has a userID and password that will be used to grant access. Alumni will be handled somewhat differently.

The table tnpEmployee contains an attribute called NPGS. This attribute is the userID for everyone at NPS. In fact, anyone who ever had a NT account while at NPS will have a unique user ID regardless of whether or not they are

still at the school. When an attempt is made to logon to the Schieffelin Award main Web page, the individual is challenged for his/her user ID and password. If he/she doesn't possess a valid user ID and password for the NPGS domain, access is denied. The user ID is captured as a session variable and used to validate access to other pages within the balloting system. From the main Web page, only students are authorized access to the actual balloting pages. This is accomplished by comparing the captured user ID to another attribute called IsStudent in tnpEmployee. If the user requesting access is not a student, they are redirected to a Web page informing them that they do not have access. The code for this login is contained within a page called "login.asp" and can be viewed in Appendix F.

Alumni will be accessing the system from outside the NPS firewall. This means that a port in the firewall must be opened so that alumni can gain access using their Internet Service Providers (ISP). All alumni user ID's are retained in the EMS as mentioned above.

When Alumni are notified via the US Postal system that voting for the award has commenced, they may be issued a password and their old user ID. By providing that information to the Web-server, the Web-server will be able to authenticate Alumni and allow access to the electronic ballot. Currently, the NPS firewall will not permit this type of access. While the first prototype will be ready to accept alumni voting (with some modification to login.asp), the feature will not be able to be tested until further analysis is completed on the technical requirements for providing a secure balloting procedure for Alumni from off

campus, and the firewall is configured to allow such access.

3. Session Security

Once the user has supplied his/her identification to access the system, care must be taken to ensure that private data is not intercepted as it travels over the network. Basic Internet protocols are not point-to-point. Packets of data travel across a network through various points. "Packet Sniffers" can be used to intercept these packets of data. There is no single easy way for preventing the interception of this data. Some typical methods that organizations use to prevent private information from being viewed is through Public and Private Key Security, Secure Socket Layers (SSL), Digital Signatures and Passwords and Certificate Servers. The first prototype will not utilize any of these methods. The information being processed for the SADBMS is probably not critical enough to warrant implementation of any of these methods. However, Microsoft's Internet Information Server has the capability for SSL and could be implemented if required.

4. Software/Hardware Security

Maintaining Internet security is a tedious process. The key is to set up the site without any obvious security holes and then to manage the site and discover weaknesses in software or administrative practices. Based on our security requirements and what is already in place at NPS, we will be utilizing Microsoft's NT based Operating

Systems, Internet Information Server (IIS) and SQL Server for the Schieffelin Award Web-based balloting process.

Windows 2000 Server is an NT based operating system. Its Windows NT file security (NTFS) offers a robust method of securing files and directories on the server. NTFS permissions are used to protect resources from users who can access the computer locally (sitting at a computer) or remotely (connecting to a shared folder). NTFS permissions allow setting file and directory permissions to a fine degree of granularity. They provide the ability to set individual permissions for each file within a folder. They can also allow different users different permissions and even deny access to individual users or groups of users.

IIS and Windows 2000 Server allow for flexible Web site design while still providing an adequate level of security. IIS uses the users and groups implemented within 2000 Server to secure Web pages, files and directories. Microsoft merely made the NT security system available to IIS, rather than design a complete new security system (Strebe, 2000).

Another unique feature of IIS is its ability to log transactions. It can store these logs in a file or a database. This gives an administrator a powerful tool for reviewing transactions with his/her Web pages. Additionally, IIS can use secure connections with Secure Socket Layers (SSL) for connections between clients and IIS, it can require passwords for access to Web pages, and it can allow or disallow access from individual computers or subnets.

When securing Web pages, you must first make sure that IIS and NTFS settings for the Web pages are correctly set. When configuring user and group permissions, we must be sure not to give people access to things we don't want them to have access to. If we want to block anonymous users from access to a particular Web page, we must ensure that the `IUSER_computername` (a group that allows anonymous internet access) is not a member of a group to which we assigned access. Additionally, the `EVERYONE` group is, by default, given full control to directories and files on your server. If we want to prevent everyone from having access to a particular directory or file, we must remove this group from its permissions.

An easy way to ruin the security of a site is to allow directory browsing. While there may be times when you want to allow directory browsing, it is, by and large, a significant security risk. Directory browsing allows users to see the structure of a Web site; a malicious user may be able to determine some functionality that you may not want users to know. Directory browsing can be disabled from IIS in the properties of the WWW Service.

Disabling Script and DLL execution is another way of increasing the security of a site. This prevents a user with write permissions from being able to place a malicious script or DLL on your Web site where it could be run. Read access to directories that have script or execute access should be disabled, especially in directories that contain Active Server Pages. This provides limited protection because it prevents users from being able to read your scripts and figure out how they work.

Many of the techniques above are oriented toward security through obscurity. New threats arise daily, and administrators need to utilize every means available to make it as difficult as possible for malicious users to break into their Web sites.

The Schieffelin Award default Web directory in Microsoft's Internet Information Server is configured to allow anonymous access. This allows Internet and Intranet users the ability to visit the award home page and read about the award. The administration directory will not allow anonymous access and is set up to only allow the Schieffelin administrator access to those pages that are used for administering the Schieffelin Award. The balloting Web pages will not allow anonymous access either. Users are challenged for their Windows NT NPGS Domain user ID and password. Additionally, an active server page (ASP) that validates whether or not a user is a student is used to determine access to the balloting pages. Faculty are not allowed access to the balloting pages. When alumni are added to the balloting process via the Internet, the code on this ASP page will have to be updated.

Security alone will not protect our data. As described in the following section, we must concern ourselves with how the data is input, updated and deleted from the database as well.

B. DATA INTEGRITY

The primary goal of SQL Server is to enable data storage. It is essential that designers ensure that when data is entered, it is valid. Reliability and usefulness of

a database is only as good as its data. As we will see, constraints in SQL Server can be used to ensure accurate and reliable data is entered.

1. Constraints

Integrity constraints (ICs), are features of the database that allow for data validation. This data validation takes place as the data is being entered and before it is actually written in tables. Data integrity falls into four categories. These categories and the constraints used to enforce them are illustrated in Table 10.

We can configure the following types of constraints: default, primary key, foreign key, unique, Not Null, stored procedures, triggers, identity, column and table and check. We will examine configuring the stored procedures, triggers, default, primary key, and foreign key constraints.

Integrity type	Recommended Constraints
Entity	PRIMARY KEY constraint UNIQUE constraint IDENTITY property
Domain	DEFAULT definition FOREIGN KEY constraint CHECK constraint NOT NULL
Referential	FOREIGN KEY constraint CHECK constraint
User-defined	Column and table level constraints in CREATE TABLE Stored Procedures Triggers

Table 10. Options for Enforcing Data Integrity.

a. Stored Procedures

"A stored procedure is a set of logical SQL code that selects columns and rows of data from one or more tables in your database. Stored procedures also allow you to write SQL statements that insert, update and delete data." (Kauffman 1999) Anytime we are modifying tables, we must concern ourselves with the integrity of our data and ensure it is being entered correctly. Multiple programs can use stored procedures. If multiple programs are utilizing the same stored procedure to update, insert or delete data, we ensure that the data is modified in the same manner every time.

b. Triggers

Care must be taken to maintain the integrity of the database so that it contains only validated, auditable information. Most client/server databases, like SQL Server, provide built in mechanisms called "triggers" to maintain database integrity and perform other operations. If a rule is violated, the trigger will not execute the command; instead it sends an error message notifying the user of a violation.

c. Default

Default values or constraints can be configured for every column in a table. If information is not entered into the column during data entry, SQL Server will assign that column its default value. In the SADMS, the default value for voter status is "Student". When a voter goes to the form for entering this information, it is already set

to "Student". If the voter doesn't specify another value for voter status (i.e. Alumnus), SQL Server will set the voter status column's value to 'Student'.

d. Primary Key

Specifying primary keys when creating tables creates an index for the table. Indexes come in two types: clustered or nonclustered. Clustered indexes force SQL Server to store rows in order of the primary key. Nonclustered indexes are stored as separate database objects.

A primary key is normally used when searching for records. The primary key must be unique. It can't be duplicated throughout the entire table. Using a primary key helps protect the integrity of the data. Additionally, since the primary key is used for indexing, the primary key is not allowed to contain a null value and SQL Server will not allow duplicate entries in that column within the table.

e. Foreign Key

Foreign key constraints are defined when a column is created in one table that's identical to the primary key for another table. When a foreign key has been defined, if primary key values are changed, SQL Server will automatically update the values in the foreign key column in the other table.

Now that we've defined several of these constraints, let's examine some of the integrity types they can be used to enforce (Table 10).

2. Entity Integrity

Entity integrity defines a row as a unique entity for a particular table. Entity integrity enforces the integrity of the identifier column(s) or the primary key of a table through indexes, unique constraints, primary key constraints, or identity properties.

3. Domain Integrity

Domain integrity refers to the validity of data entered in a given column. Domain integrity can be enforced by restricting data types, using check constraints and rules for format, or by using foreign key constraints, check constraints, default values, null definitions and other rules to validate the possible range of values.

4. Referential Integrity

Referential integrity is used to maintain relationships between entities when records are entered, updated or deleted. In SQL Server, referential integrity is determined by the relationships between foreign keys and primary keys or between foreign keys and unique keys. Referential integrity is used to ensure key values are consistent across all entities in the database. What this means is that references are not made to values that don't

exist and that when a key changes, the values to any references to that key change throughout the database.

Enforcing referential integrity in SQL Server prevents users from:

- o Adding records to a related table if there is no associated record in the primary table.
- o Changing values in a primary table that result in orphaned records in a related table.
- o Deleting records from a primary table if there are matching related records.

In the following example (Figure 5), with the tnpSchieffelinHistory and tnpEmployee tables in the Award database, referential integrity is based on the relationship between the foreign key (EmployeeID) in the tnpSchieffelinHistory table and the primary key (EmployeeID) in the tnpEmployee table. Any changes to a record will be cascaded throughout the database to other records that share the same key.

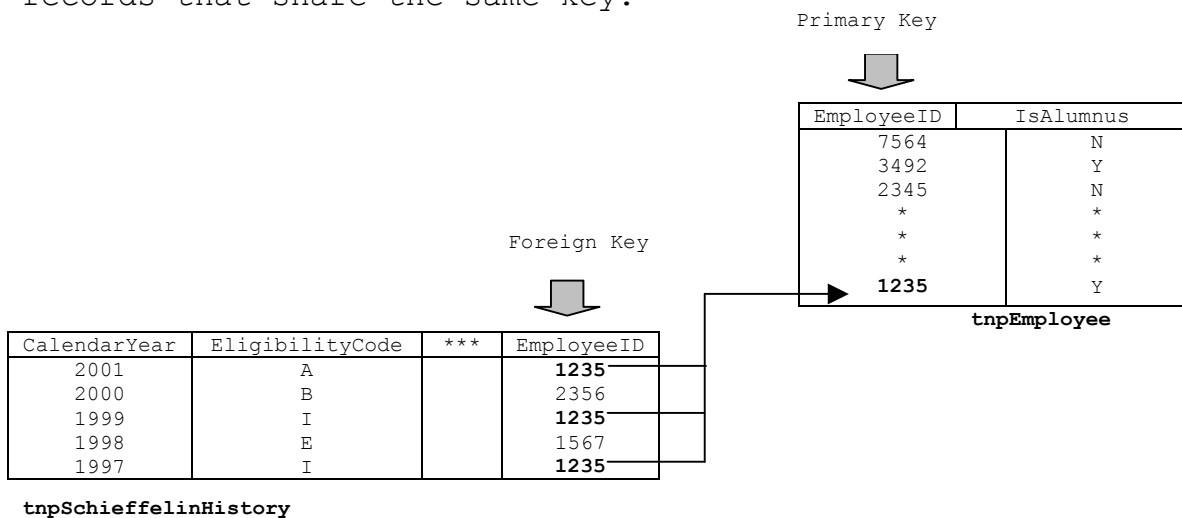


Figure 5. An Example of Referential Integrity.

5. User-defined Integrity

Each of the mentioned integrity types supports user-defined integrity. User-defined integrity allows you to define specific business rules not defined in one of the other integrity categories.

C. PRIVACY

The concern with privacy when dealing with electronic balloting systems is that the submitted ballot cannot be traced back to the user who submitted it. There are several ways to deal with this concern. One way is by using cryptography. By using an algorithm to encrypt the users identity as the data packets travel across the Internet we can protect that user's identity from any malicious attempts to view it by some third party. However, any technique that requires authentication and identification essentially has the underlying roots to trace a ballot back to its source.

The best practice for ensuring that private information is not revealed is not to send any private information. The current requirement calls for displaying the faculty member's name on the ballot. The current login for the ballot provides reasonable protection in this regard. While it is not displayed for the user to view, the employee ID is used because it is a unique identifier and is not related to a social security number. The employee ID is linked to a checkbox on the ballot so that we can ensure we have captured the candidate the user voted for.

Within the SADBMS, the table that contains the ballot (tnpSchieffelinBallot) and the table that tracks whether voters have voted or not (tnpSchiefVotingHistory), are not joined. It is possible that a malicious database administrator or a malicious user who gains administrator privileges could in fact join the tables and trace a ballot back to its originator. We're not sure why somebody would want to do this, but it is possible. There are other areas within the EMS that a malicious user would be more attracted to than the SADBMS.

D. USER INTERFACE (UI)

Usability and user interface are directly proportional. If a software product is poorly designed, its usability will also be poor. The user interface is probably the most important factor in the success or failure of most software systems.

To maximize the usability of an interface, Shneiderman (1997) proposed eight golden rules of graphical user interface design:

- o Strive for consistency,
- o Enable frequent users to use shortcuts,
- o Offer informative feedback,
- o Design dialogs to yield closure,
- o Offer error prevention and simple error handling,
- o Permit easy reversal of actions,
- o Support internal locus of control, and
- o Reduced short-term memory load.

While not all of these rules are applicable to the SADBMS, several of them provide the basic foundation for developing a user interface that will promote ease of use for the balloting process. The underlying goal for the balloting process is to allow voters to easily select their preferred candidates, verify their choices and submit their ballot in a manner that doesn't make them feel like they are wasting their time.

Consistency allows users to perform the same functions in the same manner in similar situations. This rule is frequently abused and can lead to user frustration and confusion. This means designers should use identical terminology, menus, and help screens throughout the design.

Feedback should be provided for every action a user performs. Feedback allows the user to understand the status of the system at any point in time. Each process should provide a level of feedback at its start and ending points. Feedback at the end of a process gives the user a sense of closure and that he/she was successful (Shneiderman, 1997).

A user must not be able to input a value or start a process that could generate an error. The system must detect errors and provide the user notification and simple instructions on how to correct the error.

The user must be provided a reasonable means to reverse his/her actions. If the user entered some information that he/she later determines unsatisfactory, the user should be provided an option to reverse that

action. This also means that notification should be given to the user that he/she is about to submit information that can't be reversed.

The SADBMS utilizes a Web-based interface. Users are presented with a series of screens that make up the ballot. Upon submission of their ballot, they are shown a summary of the ballot they submitted. The Schieffelin Award administrator also interacts with the SADBMS with a Web interface. He/she is able to query, view and print reports. Screen shots of the user interface and their descriptions are presented in more detail in the next chapter on prototyping.

V. PROTOTYPING

Laudon et al (Laudon and others, 1999) define prototyping as the process of building an experimental system quickly and inexpensively for demonstration and evaluation so users can better determine information requirements. When reviewed by users, the prototype can be used as the format for creating the final system.

Prototyping is used to alleviate problems found in the traditional software development approach. Prototyping addresses the problem that frequently occurs of users not really understanding their needs. It reduces the time-consuming effort required for a technical documentation of the system and it increases communication among the development team.

A. APPROACHES TO PROTOTYPING

There are two types of prototypes: throwaway and evolutionary. With a throwaway prototype, a model is developed, then discarded and replaced by the final system. The throwaway method tends to increase the time and costs for the requirements definition phase of development, but it decreases costs over the entire cycle (Gavurin, 1991). Gavurin (1991) refers to the evolutionary prototype model as the "keep it" approach. The evolutionary model is refined and enhanced until the final system is produced. Evolutionary prototyping creates a repetitive cycle of learning until the final product is complete.

B. PROTOTYPING STEPS

Prototypes are most appropriate for complete development of an application when requirements are not well understood, or when a proof of concept technology is required. The prototype in these cases becomes a working model of an information system and is meant to only be preliminary. As refinement of the prototype continues, developers become more familiar with user requirements. Prototyping can help build systems that are better tailored to user requirements and are therefore more readily acceptable when deployed. This process of building a preliminary design, testing it, refining it and testing it is an iterative one. The iterative nature of prototyping actively promotes design changes and often leads to an application that more accurately reflects users' requirements (Laudon and others, 1999).

Laudon et al also identify four steps to prototyping:

- o Identify the user's basic requirements.
- o Develop an initial prototype.
- o Use the prototype.
- o Enhance and revise the prototype.

When *identifying basic requirements*, the system designer and system user work together to capture the user's needs. In *developing the initial prototype*, a working prototype is created quickly through the use of several different design tools including: fourth-generation software, interactive multimedia, or computer-aided software engineering (CASE) tools. While *using the*

prototype, the end user is encouraged to use the system as s/he would under normal working conditions, looking for bugs in the system and making suggestions on ways to improve the prototype. *Enhancing and revising* the prototype means the designer applies changes the user requested and refines the prototype to meet those needs. After revision, the cycle repeats itself by returning to steps 3 and 4 until a system emerges which satisfies the user.

C. THE ADVANTAGES AND DISADVANTAGES OF PROTOTYPING

There are many advantages claimed for prototyping, which we list below:

- o Resolves much of the uncertainty about requirements or design solutions;
- o Provides assistance in the design of the user interface, such as on-line display, data-entry screens, reports or web pages;
- o Enables users to provide immediate input about the part of the system with which they will be working;
- o Encourages user involvement throughout the system's development;
- o Reduces project costs;
- o Decreases communication problems;
- o Increases productivity and business responsiveness;
- o Reduces risk and training time;

- o Eases installation.

Prototyping can also create problems during application development. Designers sometimes misuse prototyping to circumvent proper analysis and design which, in turn, leads to poor requirements identification and a poorly designed application. Problems may also be encountered when pressure to develop the prototype creates compromises with implementation. Implementation compromises include failing to take into consideration plans for deployment, user support/training and documentation.

Prototyping has become a popular tool for helping define requirements and capabilities. Because of the inherent problems of the traditional life cycle approach to building software, information systems managers are increasing turning to prototyping as a software development process.

D. THE SADMS EVOLUTIONARY PROTOTYPE

The prototype we've developed for the SADBMS is not intended to be a "throwaway" prototype, but rather an evolutionary model. The idea behind the first prototype is to deliver a web-based system that exhibits the functionality found in the current award process. We want to be able to demonstrate that the ballot information can be captured via the Web and processed in exactly the same way as the current system.

We've subdivided the discussion of the SADBMS prototype into four sections:

1. The database views used by the SADBMS,

2. The ballot interface and processing,
3. The administrator functions, and
4. The scoring and ranking function.

Screen shots are used where necessary to provide emphasis. Appendix F depicts all the Web pages of the SADBMS, their relationships/dependencies, and all views, stored procedures and tables used in the SADBMS. Appendix G contains the Web page code for the prototype.

1. Database Views

Database views are a valuable feature of relational technology that SQL Server supports. They are simple to create and implement. A view basically takes an SQL SELECT statement and creates a virtual table that is accessible using SQL in the same way as an actual physical table. A database view is simply a representation of data gathered from one or more physical tables stored in the database. The data in the view are not stored physically as a table in the database, but exist only in the source tables used to create the view.

The view *vnpschiefCurricID* for example, is created from the base tables *tnpCurriculum* and *tnpSchieffelinBallot*. The view contains the attributes *CurricID*, *CurricNumber*, *CurricShortName* and *DepartmentID* from the *tnpCurriculum* table and it contains the attribute *CurriculumID* from the *tnpSchieffelinBallot* table. These attributes are only a subset of all the attributes contained in the base tables. The view is used to populate a drop down box on page three of the ballot with a list of curriculums (*CurricShortName*). When users select their

curriculum, it is written to the table tnpSchieffelinBallot table as part of the user's demographics. The following SQL code creates the view:

```
CREATE VIEW dbo.vnpSchiefCurricID
AS

SELECT DISTINCT
dbo.tnpSchieffelinBallot.CurriculumID,

dbo.tnpCurriculum.CurricID,

dbo.tnpCurriculum.CurricNumber,

dbo.tnpCurriculum.CurricShortName,

dbo.tnpCurriculum.DepartmentID

FROM  dbo.tnpCurriculum INNER JOIN
      dbo.tnpSchieffelinBallot ON
      dbo.tnpCurriculum.DepartmentID =
      dbo.tnpSchieffelinBallot.DeptID
```

The "CREATE VIEW" statement creates the view as a logical table in the database. The "SELECT DISTINCT" statement selects the attributes from the base tables as DISTINCT (no duplicate entries). The "FROM" statement identifies the base tables and their relationship. In this particular view, the relationship is based on where *DepartmentID* in the tnpCurriculum table is equal to *DeptID* in the tnpSchieffelinBallot table. Multiple views are utilized by the SADBMS, and documented in Appendix M.

2. The Ballot Pages

To access the ballot, all users are required to enter the Naval Postgraduate School's main Web page. Currently, the main Web page is accessible by anyone with access to

the Naval Postgraduate School's Intranet. Eventually, when access is provided for alumni, this page will be accessible via the WWW. The intent of the main Web page is to provide users with some background information about the Schieffelin Award. Users can view past award winners and view instructions about the balloting process. Eventually, a letter from the Superintendent may also be included like the one provided with the current ballot (Appendix B). Figure 6 is a screen shot of the main Web page.

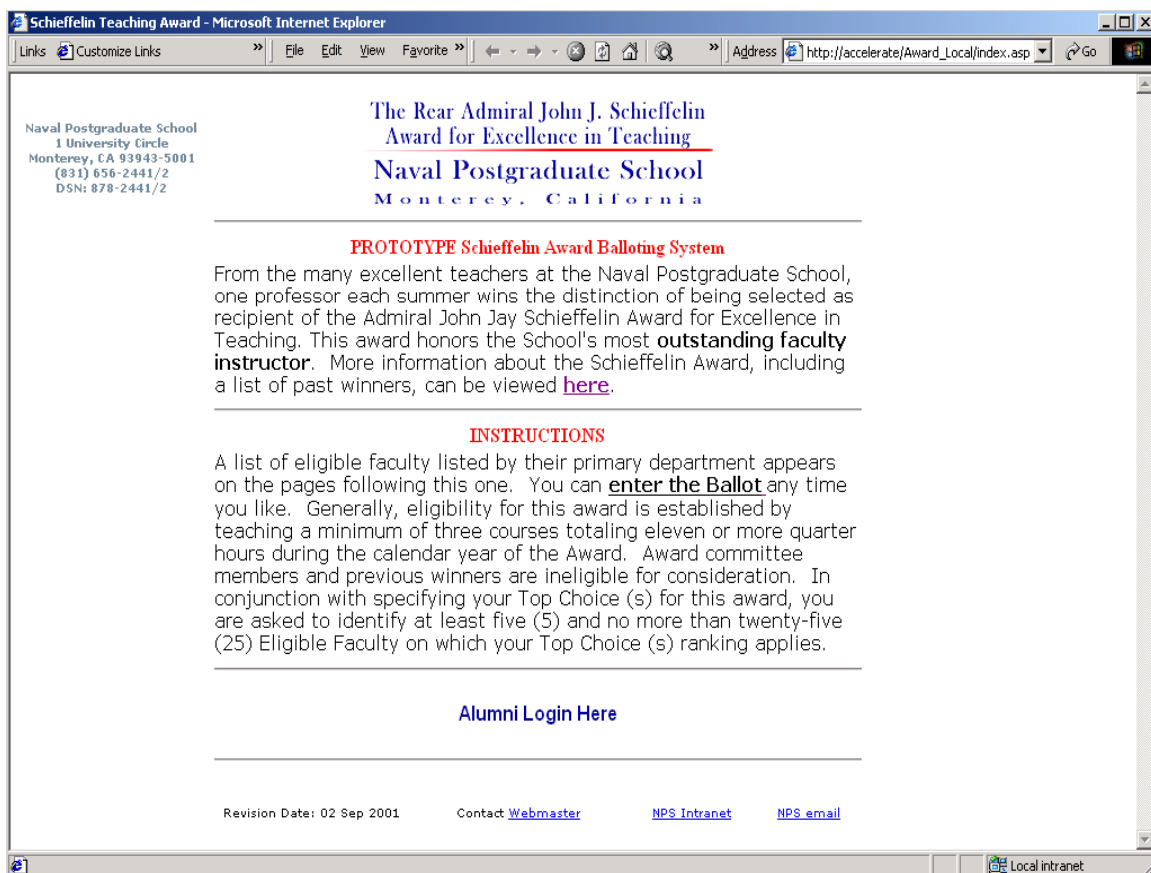


Figure 6. The SADBMS Main Web Page.

From the main Web page, students and alumni will be able to enter the balloting area. Currently there is no login capability for alumni because of issues with the NPS firewall. To enter the ballot, the SADBMS uses the

integrated Windows NT logon method. If the user is currently logged-on to the NPGS Domain, the SADMS is able to authenticate the user and allow access. Concurrently, and transparently to the user, s/he is actually sent to a login page that checks to see if the user has voted already and that the user is a student (The code for this appears in Appendix G.). If the user has already voted, they are redirected to fail1.html (Figure 7), where they are informed, "You have already voted. If this is an error, please contact your Administrator." Administrator is a hyperlink to the Schieffelin Award Administrator, who will most likely be the Chairman of the Schieffelin Award Committee.

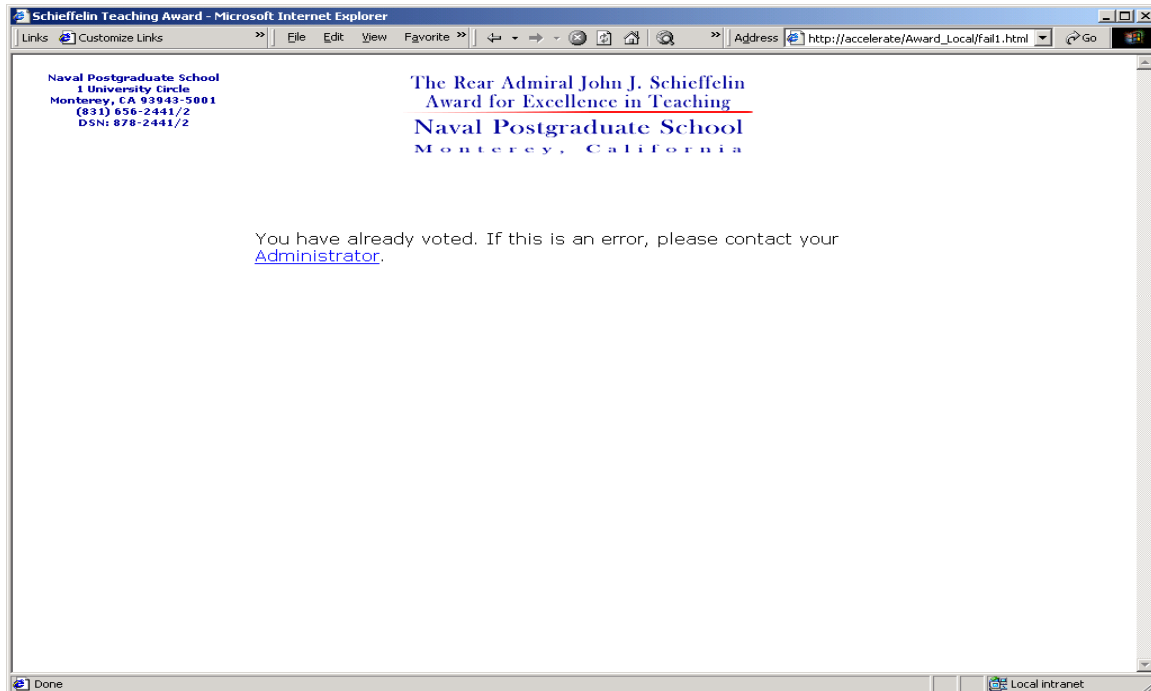


Figure 7. User Already Voted.

If the user is not a student, s/he is redirected to fail.html where s/he is informed, "Only students are authorized to enter the Ballot Area. If you

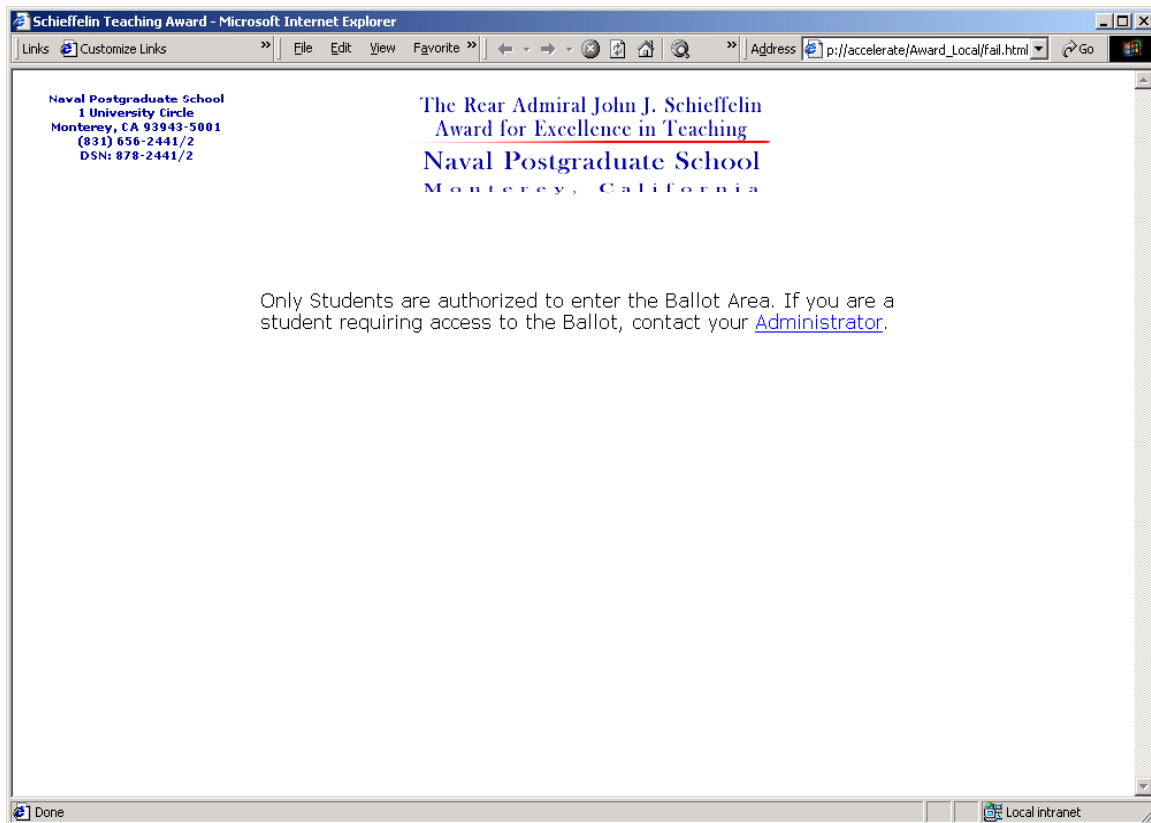


Figure 8. User Not Student.

are a student requiring access to the ballot, contact your Administrator" (Figure 5-3). As above, the Administrator is a link to the Schieffelin Award administrator.

There is one other case that could result in a failed logon. If a user is logging in from outside the NPGS Domain and does not have an NT account or has more than one account, s/he will be redirected to "login.asp" (Figure 9).

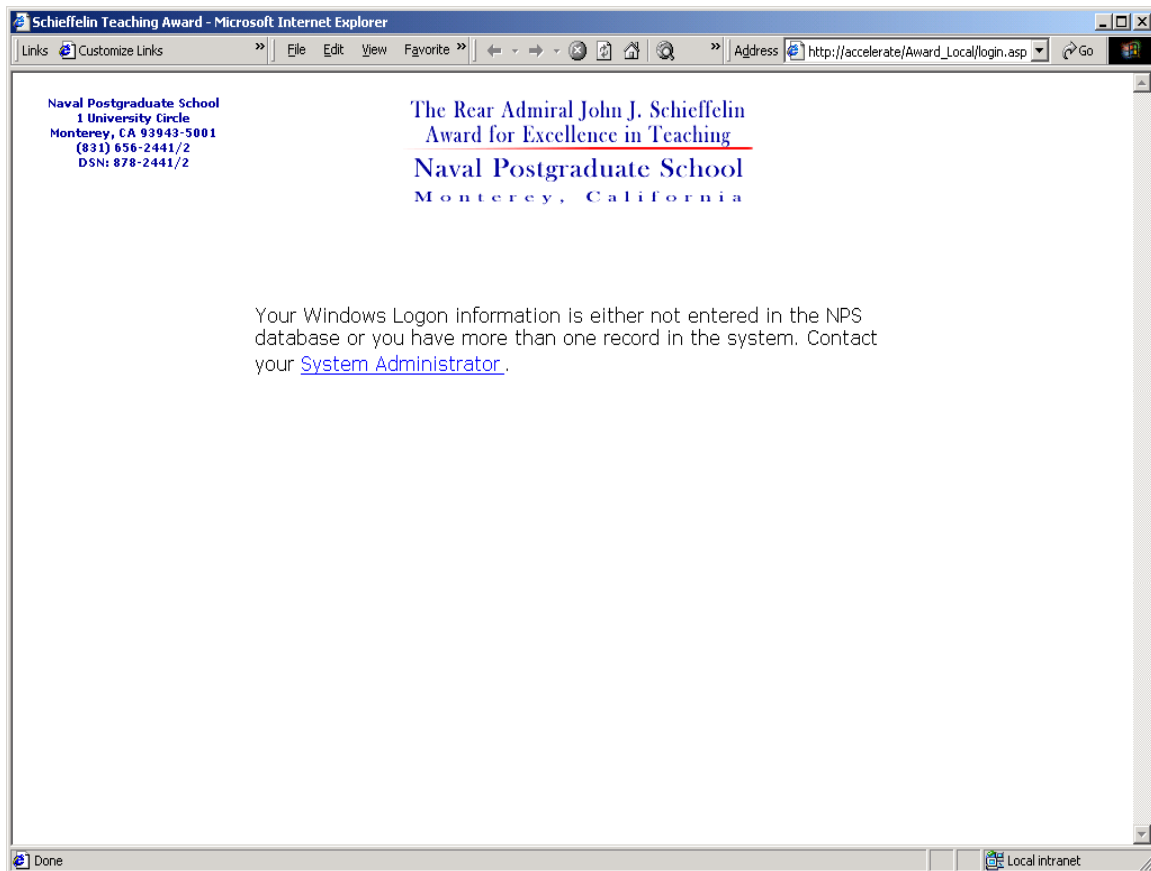


Figure 9. Failed Logon.

If correctly authenticated, voters are allowed to enter the ballot (Figure 10 and Figure 11). The first page of the ballot (ballot.asp) lists all eligible faculty by department. Voters read final instructions about filling out the ballot and select from 5 to 25 candidates with whom they are familiar by placing checkmarks in the checkboxes next to the professor's name. When they've completed checking their candidates they submit the form by pressing the "submit" button and are taken to the second page of the ballot (Figure 12).

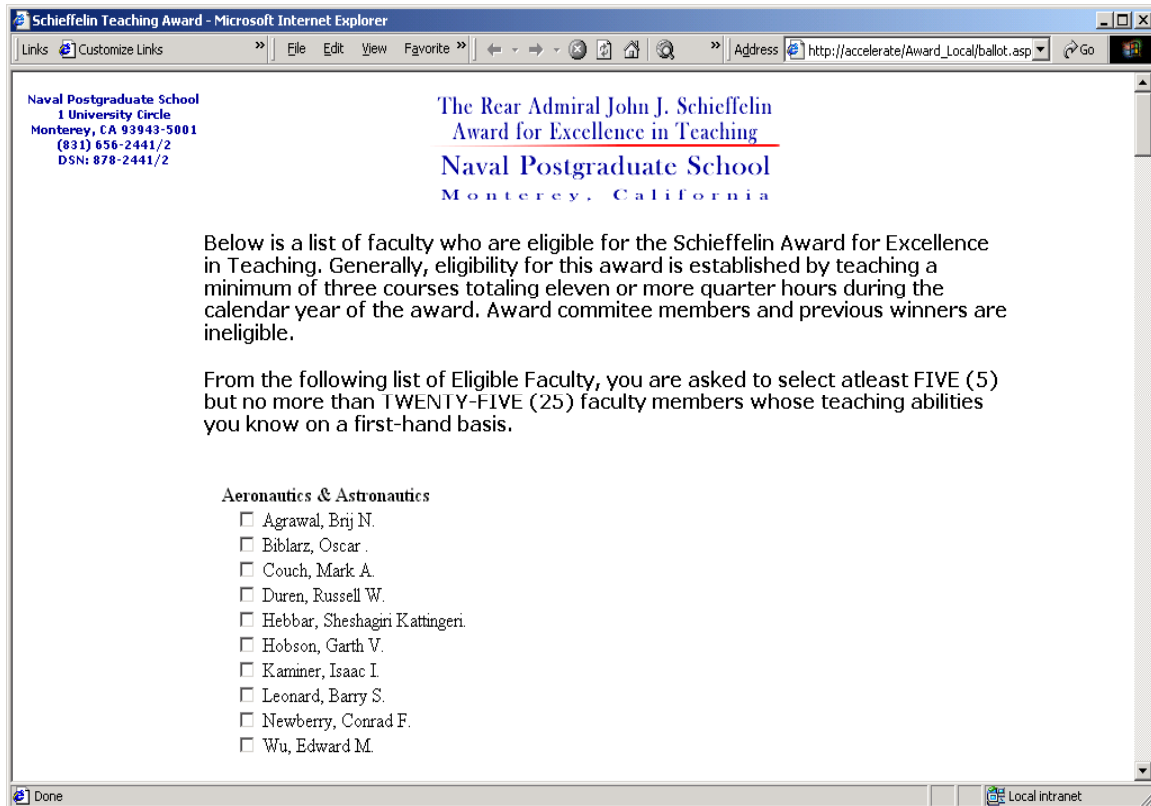


Figure 10. Top of Ballot.

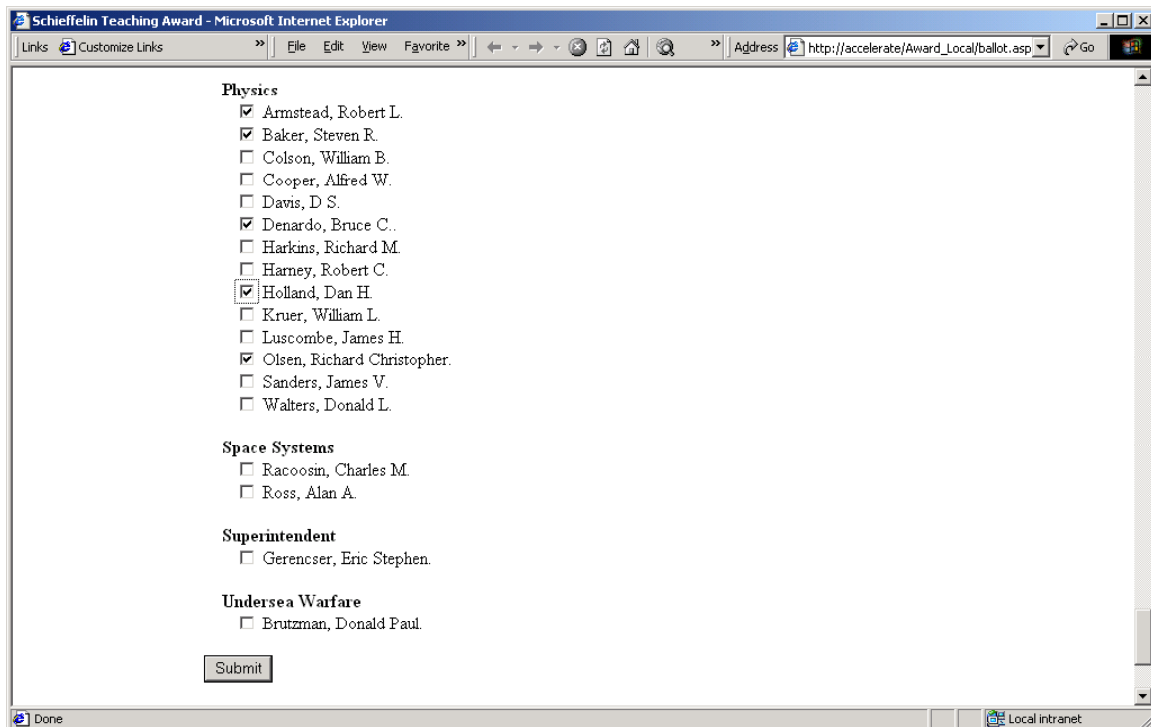


Figure 11. Bottom of Ballot.

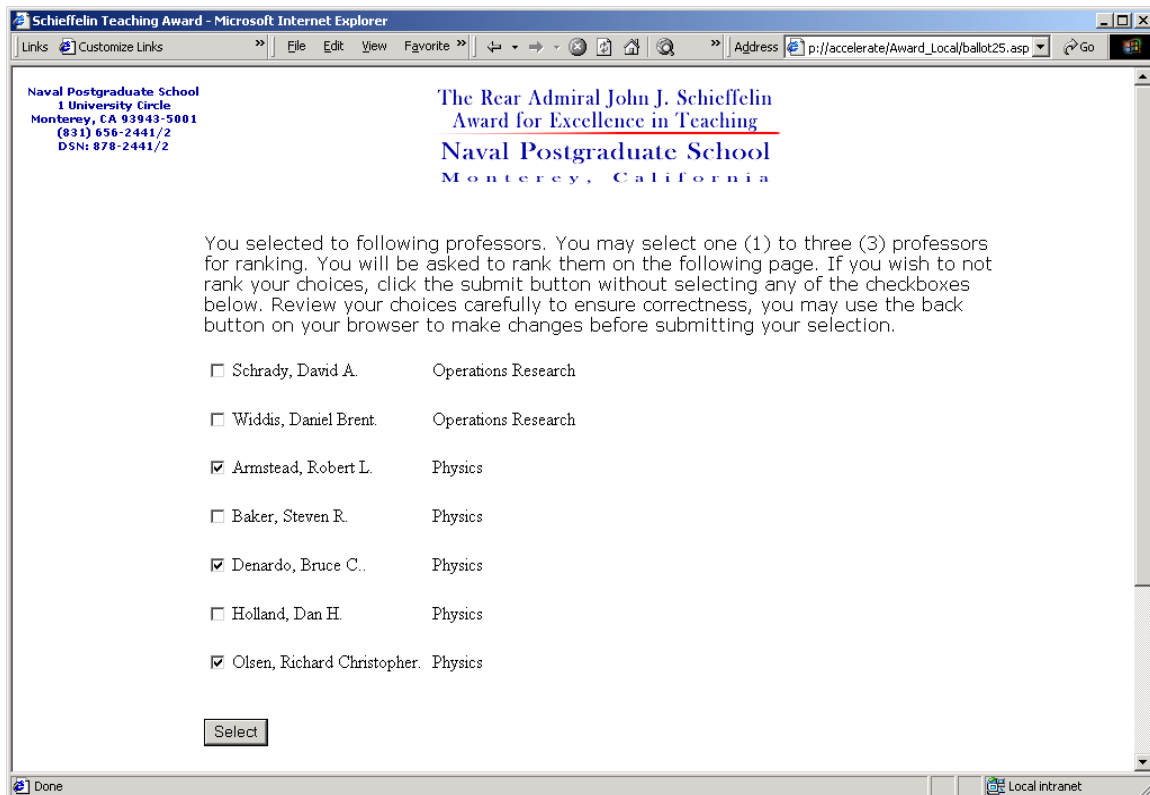


Figure 12. Second Page of Ballot.

The second page of the ballot lists all those candidates the voter selected on the first page. On the second ballot page, the user is asked to identify his/her top one, two or three choices. At this point the user can also use the back button on his/her browser to make changes to those candidates s/he selected. The user may also choose not to select anyone as his or her top choices and continue with the balloting process. When the user is satisfied with his/her choices s/he presses the "submit" button. At this point, all candidates the user has identified are written to the ballot as choices one (1) through twenty-five (25) and the user is directed to the third balloting page (Figure 13).

On the third balloting page the user is asked to select whether s/he is a "Student" or "Alumnus" and what curriculum s/he belongs, or belonged, to from the corresponding drop down boxes. This information is used as demographic data about the voter. If the user selected any top choices from the previous balloting page, s/he is asked to rank them from 1st to 3rd using the provided radio

Schiefelin Teaching Award - Microsoft Internet Explorer

Links Customize Links File Edit View Favorite Address http://accelerate/Award_Local/ballot3.asp Go

Monterey, California

Please select your voter Category and Curriculum Area.

Student 7 Information Technology Management

Rank your top choice(s) with 1st being the most preferred and 3rd being the least. If no professors are listed, it is because you opted to not rank your choices. Ensuring you selected your voter category and curriculum above, press the submit button to continue.

1st	2nd	3rd	NAME	DEPT
☒	☐	☐	Armstead, Robert L.	Physics
☐	☒	☐	Denardo, Bruce C.	Physics
☐	☐	☒	Olsen, Richard Christopher.	Physics

Support Statement for Top Choice (s). Your are invited to furnish a short statement in Support of your First Choice candidate. List those qualities which you feel make this person an outstanding teacher. If you don't not wish to make a statement, or you have not ranked your choices, you may leave the field blank.

Professor Armstead is....

Submit

Revision Date: 02 Sep 2001 Contact [Webmaster](#) [NPS Intranet](#) [NPS_email](#)

Done Local intranet

Figure 13. Third Page of Ballot.

buttons. The user is also allowed the opportunity to make any comments about his/her top choice.

When the user presses the "submit" button, the demographic information, the first, second and third choices and the comments are all written to the ballot table in the database and the user is directed to a ballot

summary page (Figure 14) where s/he may view the results of her/his submission. At this point, there is no ability to make changes to the submitted ballot.

The ballot summary displays to the user all instructors s/he identified on the ballot and his/her top choices ranked as s/he ranked them. If the user wants to print a copy of the ballot, s/he can use the "print menu" of the browser. This page should have some added closing comments thanking the user for his/her participation in the award or some other appropriate remarks.

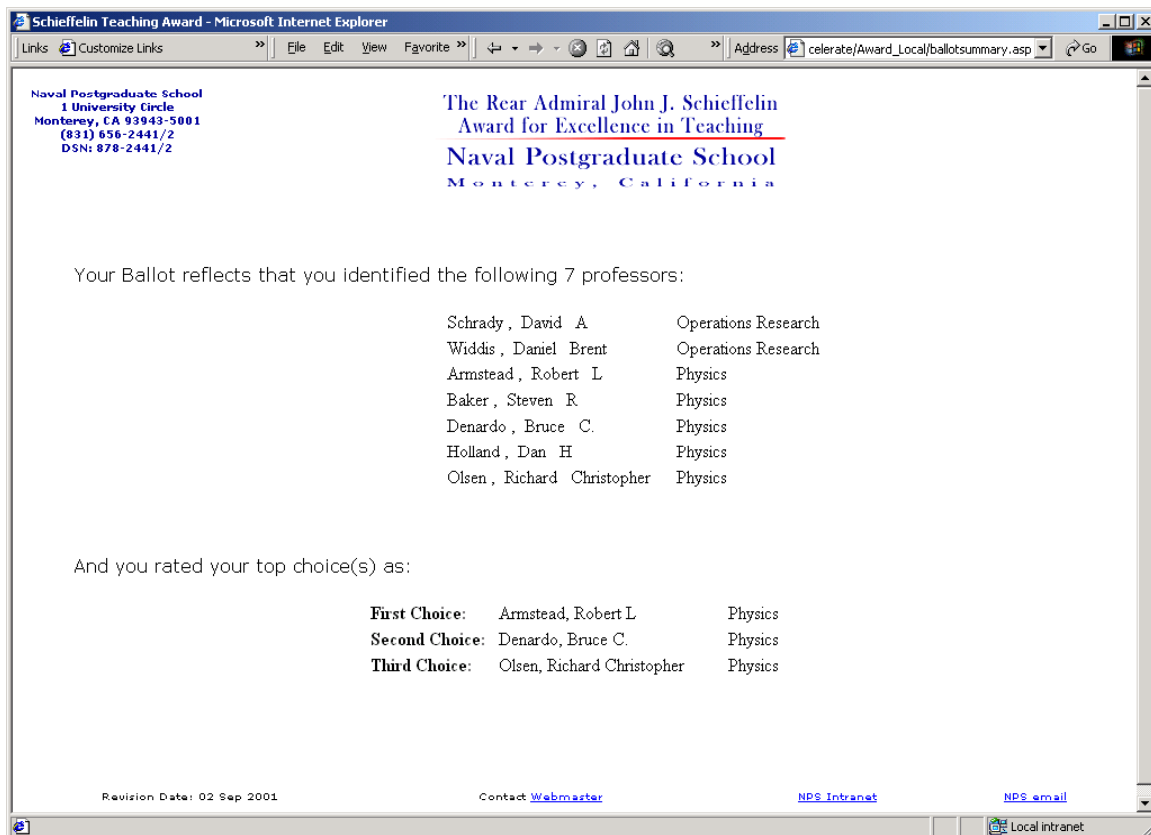


Figure 14. Ballot Summary Page.

3. The Administrative Pages

The administrative pages are designed to allow the Schieffelin Award administrator, or suitable

representatives, to manage and maintain the SADBMS. Access to the administrative pages is controlled using Windows NT authentication. Only the Schieffelin Award administrator (SA) or his representatives are given access permissions. The code for these pages is included in Appendix G.

The SA is given a main Web page where s/he can perform required reporting and administering tasks. The main Web page is illustrated in Figure 15. Specific tasks that the SA can accomplish include: Viewing and printing reports, editing professor eligibility, inputting professor

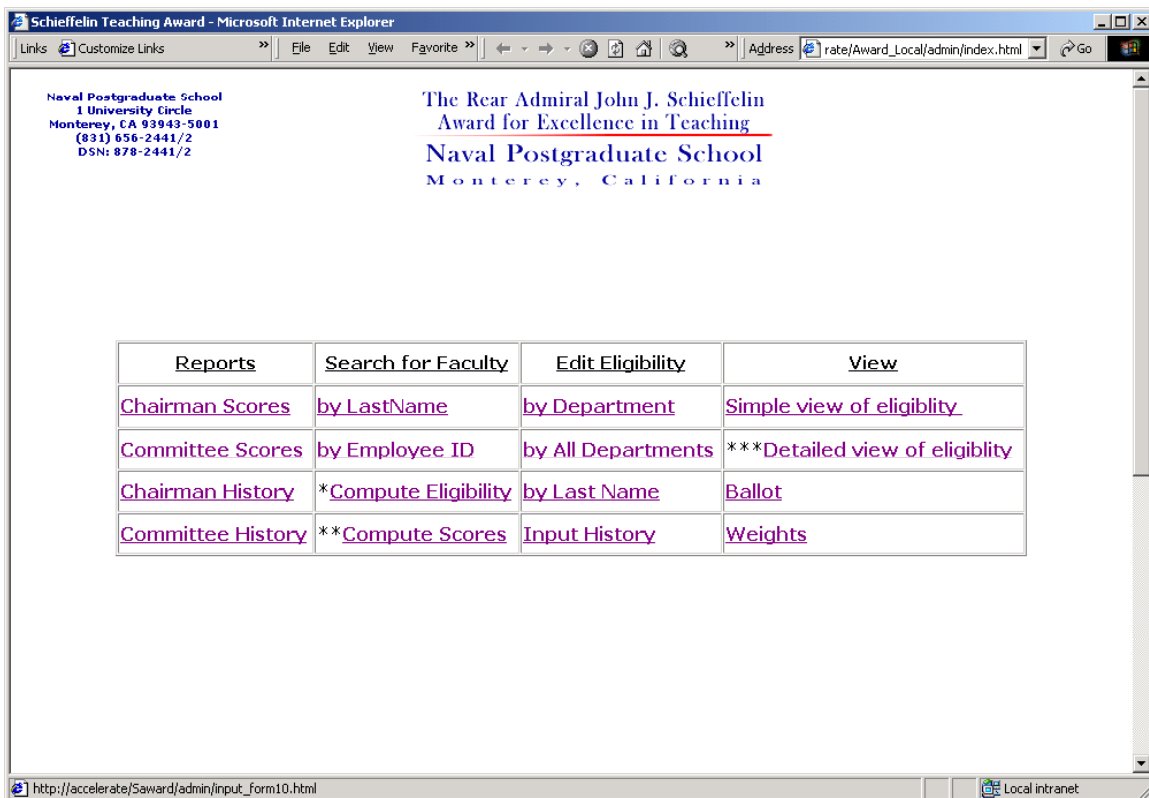


Figure 15. Schieffelin Administrator Main Web Page.

history, searching for professors by last name or employee ID, viewing teaching history by year, viewing history,

computing eligibility and computing scores. Each of these tasks will be discussed in more detail.

a. Chairman and Committee Scores Reports

The Chairman and Committee scores are the reports given to the Schieffelin Committee Chairman and Schieffelin Committee members respectively. The difference between the two reports is that the chairman's copy has the names of the professors on his report and the committee member's copy does not. This keeps the selection process anonymous. When the SA clicks on either hyperlink, s/he is prompted for the teaching year for which s/he wants to view the scores. After typing in the year, the page is displayed with the corresponding scores. The reports for teaching year 2000 scores and the input forms are included in Appendix H.

b. Chairman and Committee History Reports

These reports show the current teaching year scores as well as the previous six years of teaching history. These reports contain the same information that the original reports generated by the mainframe computer produced; they are just formatted slightly differently. Again, the difference between the two reports is that the chairman's copy has the professors' names on them. There are no input parameters for these reports. The page dynamically takes into consideration the current year and prints the teaching year and the six previous years information to the Web page. For example, on the 31st of December 2000, if you clicked on the link to view either of

these reports you would get a report that showed the scores for teaching year 1999 and the historical data for years 1993-1998. After the first of the year, if you clicked on the link, you would get a report that showed the scores for teaching year 2000 (if the ballots had been received) and the historical data for years 1994-1999. The reports for teaching year 2000 are included in Appendix I.

c. Searching for Faculty

We have included the ability for the SA to query the EMS for faculty. We discovered that when producing eligible faculty, the list is not always 100% accurate. This sometimes requires the SA to search the database for a professor in order to change his eligibility, to find out what department s/he is in, or to find out his/her employee ID. We added the ability to query by either last name or employee ID. The search forms and result forms for searching by employee ID are depicted in Appendix J and for searching by last name in Appendix K.

d. Computing Eligibility

Eligibility is based on the requirements identified in chapter 2. We've created a stored procedure in the SQL database that determines eligible professors based on those requirements. The code for the stored procedure is included in Appendix L.

The stored procedure requires an input parameter for the current teaching year. The hyperlink "Determine Eligibility" on the SA's main Web page goes to an active server page (compute_eligibility.asp) that executes the

stored procedure and then returns to the administration page once the stored procedure is complete. The code for this active server page is in Appendix G.

The input parameter (year) is dynamically determined based on the current year. Since we are determining eligibility for the previous year, we subtract one (1) from the current year and use that year as the input parameter. For example, balloting for the teaching year 2000 took place in calendar year 2001. Balloting will always take place in the calendar year after the teaching year (i.e. calendar year is always one year greater than teaching year). So the code in the page "compute_eligibility.asp", subtracts one (1) from the calendar year to get the input parameter (2000) required by the stored procedure, and eligible faculty are determined based on teaching year 2000. This means that when determining eligible faculty using the "Compute Eligibility" hyperlink, it must be in the calendar year of the award or the code will not determine eligibility for the correct year.

The stored procedure initially moves all professors into the Schieffelin History table and sets their eligibility to "Yes". The stored procedure then steps through the table setting the eligibility flag to "No" for all employees that don't meet the stated requirements.

This particular stored procedure is called "snpDetermineScheffelinEligibility." When run, it calls three other stored procedures: one that checks for previous award winners, one that eliminates faculty that have taught

less than three course segments, and one that eliminates faculty that have taught less than eleven credit hours. The code for these stored procedures is also included in Appendix L.

This stored procedure produces a fairly accurate list of eligible faculty. However, it is not 100% accurate, which is why we've provided the SA the ability to edit professor eligibility in order to produce a correct ballot. The Aviation Safety School and the Defense Resource Management Institute (DRMI) both have to submit eligible faculty lists because their courses are not based on the same credit hour scale as the other departments. The SA must enter the eligible faculty from those departments manually. There will undoubtedly be other manual entries required for professors. An example might be a professor who was scheduled to teach a course in the EMS, but for some reason (e.g., illness or other personal emergency) a different professor taught the course. If the EMS wasn't appropriately updated, the stored procedure will not calculate the professors' hours or segments correctly. To correct for these types of errors, the initial eligibility list produced by the SADBMS will still need to be circulated to the departments for validation. Once the chairman approves that list, the SA can make the necessary corrections by editing eligible faculty. The result will be a corrected ballot ready for the balloting process to begin. Of particular importance is the fact that once the eligible list is approved, and the corresponding changes have been made in the SADBMS, rerunning the stored procedure to determine eligibility will destroy all corrections, and they will have to be reentered. The

following sub-section describes the methods for editing the eligibility reports.

e. Editing Eligibility Reports

There are two types of eligibility reports in the SADBMS. One is a simple eligibility report and the other is a detailed eligibility report. Both reports are included in Appendix N.

The simple eligibility report is a basic listing of eligible professors by department. When the SA clicks on the link to view this report, s/he is prompted for the year in which s/he wants to view eligible faculty. This report allows you to view all eligible faculty including faculty that have been manually changed to eligible by the SA.

The detailed eligible faculty list is a list of eligible faculty by department. It is accessed in the same manner as the simple eligibility page. It depicts all the courses a professor was scheduled to teach during the designated award period. The award period consists of the 2nd, 3rd and 4th quarters of the previous calendar year and the 1st quarter of the current year. For example, for teaching year 2000 (award presented during June graduation of 2001), the award period includes the 2nd, 3rd and 4th quarters of calendar year 2000 and the 1st quarter of calendar year 2001. The code for the page automatically calculates the award period based on the year entered by the SA when s/he accesses the page.

The detailed eligibility page includes professors and the courses s/he taught, the course segment numbers,

the quarter in which the course was taught, its lab hours (if any), its lecture hours, the number of instructors who taught it (for determining whether the course was team taught), the year the course was taught and the type of course it was. The type of course means whether it was a resident course, a distance-learning course, a LEADS course, etc. The report will not accurately reflect the Aviation Safety School professors or DRMI professors unless they were scheduled in the EMS to teach courses recognized under the same credit hour scheme as the rest of the Naval Postgraduate School. However, this report is useful for circulating to the rest of the departments for validating their eligible faculty.

Once it has been determined that professors need to be added or deleted from the eligibility reports, the SA can use three different methods for editing professor eligibility. The first method begins when the SA clicks on the hyperlink to edit eligible faculty "By all Departments". The SA is prompted to enter the teaching year that he wants to edit eligibility for and then is presented with a Web page that displays text fields of the department, employee ID, last name, first name, middle name, year and eligible. The only field on this page that the SA can change is the "eligible" field". S/he can change the field value to either "Y" for yes or "N" for no. When the navigation bar at the bottom of the page is clicked on, the record is updated and the fields are changed to the next professor in the department. As the navigation bar is clicked, the fields change to the next professor, until all professors in that department have been viewed and then it switches to the next department and

continues with all professors in that department. The utility of the interface for this process has not been determined, so it remains in the first prototype.

The second method seems to be of more use. When the SA clicks on the hyperlink to edit eligible faculty "By Department", s/he is prompted for the teaching year and a department. For convenience, a list of all the departments and their descriptions are displayed so that the SA knows the exact format of the text to enter as the value for the department. Once the required information is entered, the SA submits the form and is presented with a Web page that has the same text boxes and navigation bar as discussed above. This time however, the SA has some flexibility about which department s/he wishes to search through to edit professor eligibility. Once the SA has scrolled through the department, there is a hyperlink for switching to another department. Records are updated in the same manner as above (by clicking the navigation bar).

The third method of updating professor eligibility is even more granular. It allows the SA to update a professor's eligibility by searching for his/her last name. When the SA clicks on the hyperlink edit eligibility "By Last Name", s/he is prompted for the year and the professor's last name. When that information is entered, the SA presses the submit button and is directed to a Web page that displays the same text fields as the previous two methods. Once the SA edits the eligibility field and updates the record by clicking the navigation bar, there is a hyperlink that allows the SA to search for another professor by his/her last name. In cases where

there is more than one professor with the same last name, the navigation bar can be used to scroll through the professors until the correct one is identified. The professor's department and employee ID can be used for unique identification. The pages used to edit eligibility can be viewed in Appendix O.

f. Updating Professor History

After the Schieffelin Chairman and committee have determined a winner, they identify professors in the top five and fifteen percent. This information has to be input into each professor's history in the EMS. The SA accomplishes this by clicking on the hyperlink "Input History". The Web pages for updating history are included as Appendix P. The SA is prompted for the teaching year and the last name of the professor s/he wishes to update. Once submitted the SA is directed to a Web page where s/he can change the professor's eligibility code to reflect the top five percent (an "A") or the top fifteen percent (a "B"). The record is updated by pressing the navigation bar, and a link is provided to add history for another professor.

g. Updating the Weights/Viewing the Ballot

There are three pages associated with updating the weights used in calculating the final scores for the award. The Web pages for updating the weights are included as Appendix Q. When the SA clicks on the hyperlink update weights, s/he is directed to a Web page where the current assigned weights can be viewed. If the SA wants to update

the weights, there is a hyperlink ("Edit Weights") that directs the SA to a Web Page where changes can be made. Once the desired changes are made, the SA clicks on the "update" button and the SA is notified that the weights have been successfully updated. If the SA wishes to confirm that the update was successful, s/he can click on the hyperlink "View Weights" to see the changes first hand. The capability to edit the weights under the old balloting system was not as convenient. However, if the weights are accidentally changed, they can easily be changed back using the same procedure. The weights must be changed prior to running the scoring function or the old weights will be used.

The final hyperlink on the administration page is the "Ballot" hyperlink. This link is used to view the current ballot. Its primary function is just a sanity check to ensure the ballot is formatted correctly and displaying all the eligible professors that we expect it to display.

4. The Scoring Function

A stored procedure is used to calculate the scores for the award. The code for the stored procedure that calculates scores is included in Appendix L. The stored procedure utilizes the same formula discussed in chapter 2 of this thesis. The stored procedure is executed by clicking on the hyperlink "Compute Scores". This hyperlink executes the code in an active server page called "compute_scores.asp". The code for this active server page is included in Appendix G. This stored procedure requires

the teaching year as an input parameter and it is given the parameter in the same manner as the stored procedure that determines eligibility as discussed in sub-section d above. When the stored procedure is finished, the code on the page returns to the administration page.

This stored procedure is executed after the balloting period is over. Running this stored procedure at anytime other than after balloting is complete will not harm the system; it just won't provide accurate scores until all the ballots are in the system.

What we've discussed above is the first prototype of the Schieffelin Award Database Management System. The prototype provides the basic functionality, reporting requirements and scoring requirements necessary for proof of concept. In the next chapter we will make recommendations on what performance steps and enhancements need to take place in order to produce the final system.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND RECOMMENDATIONS

The time it takes to build and implement software systems is always underestimated. In terms of the initial tasks we set out to accomplish, there are several which still remain to be completed.

A. FEASIBILITY

The Web-based system, SADBMS, we've constructed demonstrates that electronic submission of ballots via the NPS Intranet is feasible. Additionally, the model demonstrates that the scoring function is able to produce the exact same scores as the mainframe system (see below for more details). While we were not able to test the balloting process from the Internet for alumni, analysis indicates that it is also feasible with proper configuration of the NPS firewall.

B. MILESTONES

When we began development of the SADBMS, we defined several goals:

- o Modeling the current award process.
- o Building the relational tables and integrating them with the EMS.
- o Populating the EMS with the existing teaching award history.
- o Documenting the new balloting process.

- o Creating administrative tools necessary for the SA to be able to carry out tasks associated with completing the balloting process.
- o Testing the new balloting process.
- o Prototyping the online ballot.

The first three goals have been achieved in full. The last four goals, documentation, administrative tools, testing and prototyping are partially accomplished.

Documentation is complete as it stands, and is mentioned only because the current prototype is evolutionary and will require many modifications, and these modifications must also be documented.

The *administrative tools* consisted of building the Web interface necessary for the Schieffelin Award Administrator to be able to carry out the reporting, querying and analyzing functions that s/he currently performs while working with the existing balloting system. These basic functions are included in the first prototype. However, we believe that the reporting function could be much more robust. The ability to provide this function fully through a Web interface is limited. We also believe that providing Microsoft Access as a "front-end" would provide a powerful reporting tool that could take full advantage of the information within the SADBMS. Microsoft Access has a powerful report generator that is much more capable than trying to create and format reports with a Web interface.

Testing should continue as well. It is a major milestone for validating the system's readiness for deployment. The scoring function was tested on year 2000

ballot data that was imported into the SADBMS. The function produced the exact scores that the mainframe balloting system produced. However, we have not tested the system from start to finish. We've tested individual functions that work when tested alone, including the system's ability to allow students access to the ballot, its ability to capture user votes, its ability to determine eligible professors, its ability to allow editing of eligible instructors to produce an accurate ballot, and various other report generating functions. Testing still needs to take place where ballots are collected for a simulated voting year and the scoring function executed to determine if accurate scores are produced when implemented from start to finish.

It is recommended that a scenario be developed with a known outcome. For example, having several voters cast their predetermined votes from designated places across campus. The systems output could then be compared to a known sample to determine system accuracy.

C. EVOLUTIONARY PROTOTYPE

The devolved prototype provides the basic functions for proof of concept. However, there are a few areas that must be given greater attention.

1. User Interface

In building the first prototype, our efforts were oriented toward developing a system that proved the balloting process could take place via the Web. Much improvement/analysis is left to make the prototype "user

friendly". The first page of the ballot is very long. An average ballot contains about two hundred eligible professors. One proposal is to make the ballot even more dynamic than it already is by having it configure to the user. One major weak point is that fair scoring depends on the assumption that the voters will actually identify all faculty candidates with whom they have had experience. One way to partially account for this is to configure the ballot so that it dynamically shows only the professors that the user actually had as an instructor. This is feasible and could be considered as an enhancement to the prototype.

2. Validation and Error Correction

The current system provides little support in terms of user validation. This prototype works well when data is input exactly as it is supposed to be input. However, consideration needs to be given for when users enter data that is not acceptable. For instance, one stipulation is that the user must identify at least five but no more than twenty-five candidates with whom s/he is familiar. Programming needs to be included that actually checks these parameters and informs the users when he/she makes an input error. User input validation should be implemented in the next prototype using some type of client side validation.

Another proposal is to validate the ballots. Every time a ballot is submitted, some form of paper copy or log file generated outside of the SADBMS to track these ballots in case of a catastrophic failure or in the event that some type of manual intervention is required.

3. Internet Access

Because of the current configuration of the NPS firewall, implementation and testing of access for eligible alumni voters was not conducted. For the SADBMS to become an enterprise wide tool, the ability to access it from the Internet is critical. We recommend that access to a Web server for alumni be permitted during the balloting period. This will allow for complete implementation of the SADBMS. However, before this can be fully implemented, testing must occur.

The prototype must be configured to distinguish between alumni in terms of how long it has been since they graduated from NPS. The current prototype recognizes alumni, but only alumni who have graduated within the last three years are allowed to vote.

D. STANDALONE SYSTEM

Some thought must be given to implementing the SADBMS as a standalone system in the advent that the EMS is not implemented by NPS. The significant shortcoming of a standalone system would be the loss of SADBMS's ability to determine eligible faculty. In this situation, we would have to resort to the old method of determining eligible faculty. Beyond that, the SADBMS could still be based on the software/hardware architecture discussed in this thesis. Some minor modifications would be required to the physical model of the database, but the balloting process and scoring function would remain unchanged.

E. CONCLUSION

The prototype designed and implemented based on user requirements and business rules identified during the functional description of the current balloting system, demonstrates the feasibility of building an enterprise Internet/Intranet system in support of balloting for the Schieffelin Excellence in Teaching Award. The EMS, with its powerful data management capabilities, provides the same functionality found in the current balloting system. This system not only provides an integrated environment in support of balloting, it provides tools for administration and analysis as well. The integrity of a fully operational, Web-based system for the Schieffelin Award must be beyond reproach. The SADBMS prototype executes faithfully the processes and computations of the current system. What remains to be implemented are more stringent, and in some cases, manual auditing procedures and safeguards which ensure that the credibility of this Web-based system cannot be compromised by voters or faculty.

APPENDIX A. PROCESS FLOW CHART

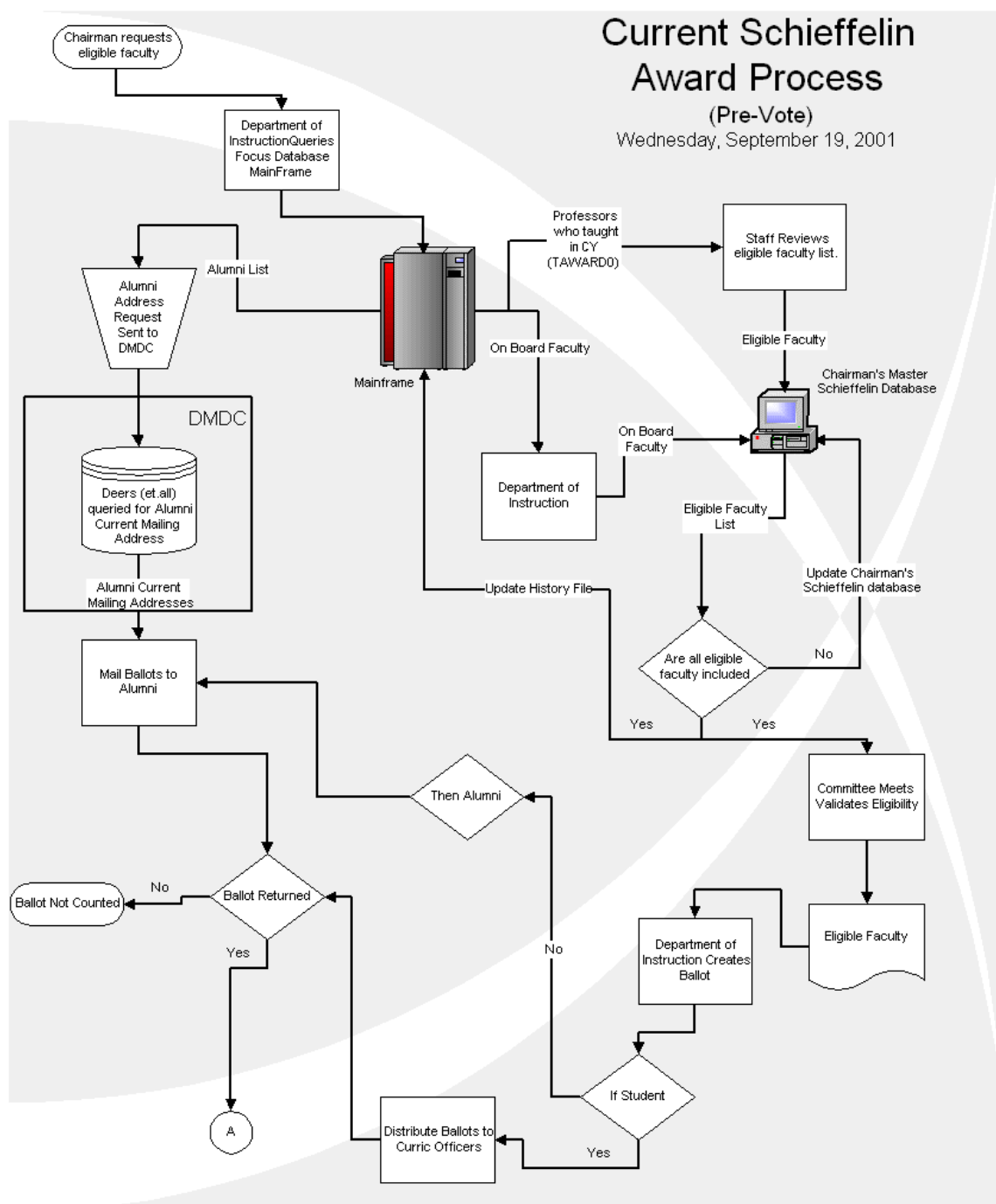


Figure 16. Process Flow Chart.

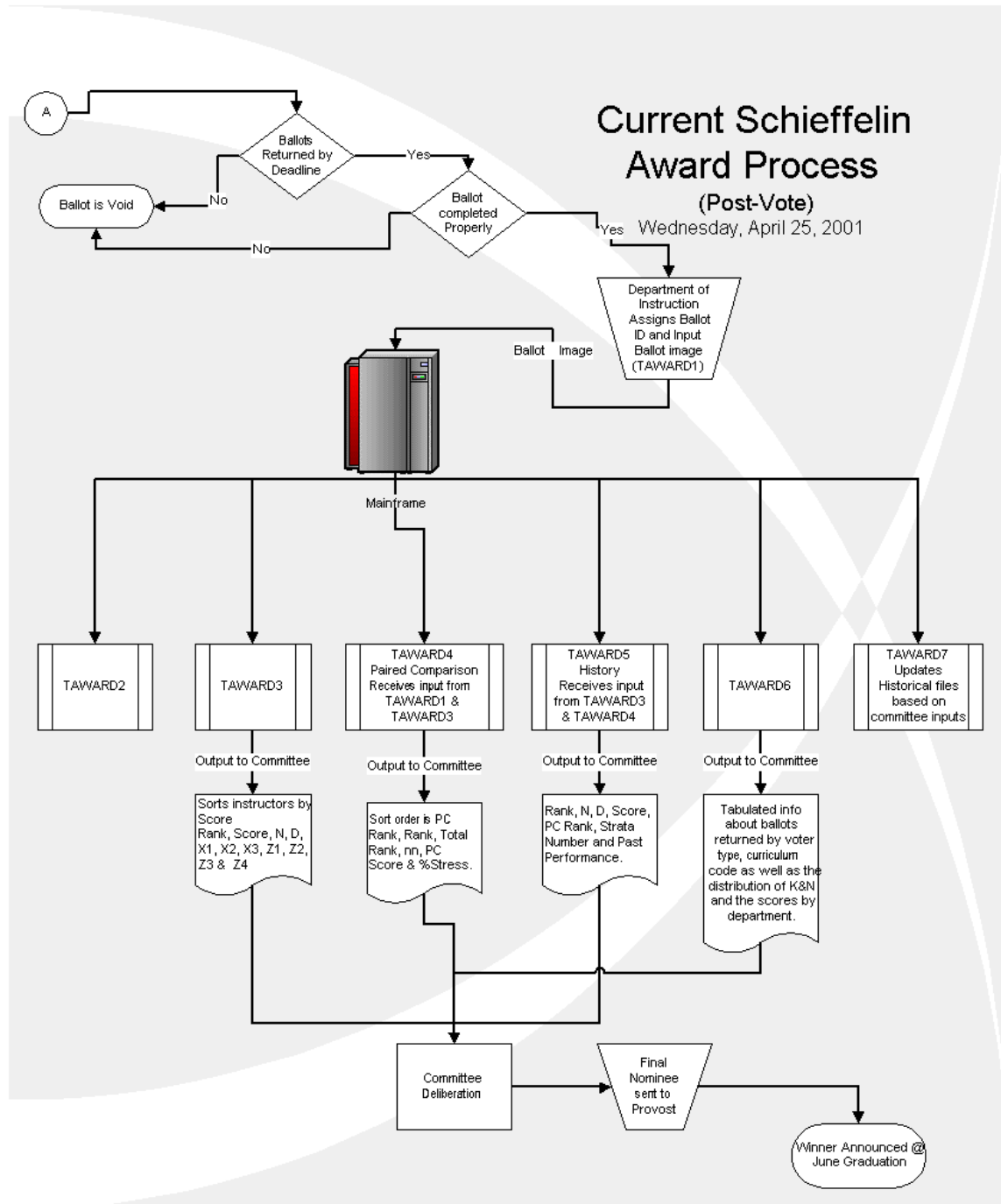


Figure 16. Process Flow Chart (cont)

APPENDIX B. SAMPLE BALLOT

12000
Ser 01B3/058

MEMORANDUM

From: Superintendent
To: NPS Students and Alumni

**Subj: RADM JOHN J. SCHIEFFELIN AWARD FOR EXCELLENCE IN
TEACHING**

Encl: (1) Ballot for Excellence in Teaching

1. Our institution places high value on providing quality education and we especially emphasize the role of teaching. The Rear Admiral John J. Schieffelin Award for Excellence in Teaching is one means we have to recognize superior teaching. In addition to the honor, the Schieffelin Award provides a substantial monetary grant for the recipient. This has been made possible through a gift from the endowment of the Naval Postgraduate School Foundation. The award is presented during graduation ceremonies each June, and the recipient is determined by the objective processing of ballot information supplied by current students and recent graduates. The Shieffelin Award for 1999 went to Associate Professor Ashok Gopinath of the Mechanical Engineering Department. All previous Shieffelin Awardees may be viewed at <http://www.nps.navy.mil/SAward/JJSAward.htm> .

2. We encourage your participation in the selection process. The ballot at enclosure (1) requests you identify and rate the top three instructors from a group of five to twenty-five instructors with whom you are familiar. Your views of good teaching can be reflected in a voluntary statement regarding those characteristics that support one or more of your top three choices. Previous winners, award committee members and those professors who did not meet the general eligibility criteria of teaching a minimum of three or more courses totaling eleven or more quarter hours during calendar year 2000 have not been included in the list of eligibles.

3. Please complete the enclosed ballot and **return it within ten days of receipt but, in any case, no later than 30 April 2001.** Alumni ballot packages include a preaddressed, postage paid envelope for return of the ballot. Students on campus are asked to place their completed ballot in the ballot box located in their curricular office. Returned ballots will be compiled by an impartial handling source.

4. Your participation in the selection process for this prestigious award is greatly appreciated.

DAVID R. ELLISON

BALLOT

RADM JOHN J. SCHIEFFELIN AWARD FOR EXCELLENCE IN TEACHING FOR CALENDAR YEAR 2000

GENERAL INFORMATION: A List of Eligible Faculty appears on the pages following this instruction section. Generally, eligibility for this award is established by teaching a minimum of three courses totaling eleven or more quarter hours during calendar year 2000. Award committee members and previous winners are ineligible for consideration. In conjunction with specifying your Top Choice(s) for this award, you are asked to identify five to twenty five Eligible Faculty on which your Top Choice(s) ranking applies. Please follow the instructions carefully.

PLEASE DESTROY YOUR BALLOT IF YOU DO NOT WISH TO PARTICIPATE

Step 1 **VOTER CATEGORY:** Circle your Voter Category of Alumnus or Student.

Voter Category: A Alumnus S Student

Step 2 **CURRICULUM AREA:** From the following list of Curriculum Areas, circle the area most closely related to your curriculum.

Curriculum Area

(Circle One)

30 Operations Research	37 Electronic/Information Warfare & Special Operations
31 Aerospace Engineering & Space Systems	38 National Security Intelligence
32 Computer & Information Programs	39 Joint C4I
34 Combat Systems, Electronics & Naval/Mechanical Engineering	40 Aviation Safety School
35 Meteorology, Oceanography & Undersea Warfare	64 Defense Resources Management Institute
36 Systems Management	

Step 3 **ELIGIBLE FACULTY:** From the "List of Eligible Faculty" provided on the attached pages, select at least five but no more than twenty-five faculty members whose teaching abilities you know on a first-hand basis.

In order for your ballot to be counted, you must identify at least five but no more than twenty-five Eligible Faculty.

In the space provided for 1st Choice, write in the four-digit identification number of your nominee. Repeat this procedure for your 2nd Choice and 3rd Choice.

1st Choice		2nd Choice		3rd Choice	

Step 5 **SUPPORT STATEMENT FOR TOP CHOICE(S):** You are invited to furnish a short statement in support of the Top Choice candidate(s) you selected in Step 4. List those qualities which you feel make this person an outstanding teacher. Circle support statement option of "0" for no statement or "1" for support statement.

96

Ballots must be returned to the Naval Postgraduate School no later than **April 30, 2001** in order to be processed.

If you received a ballot through your curricular office:
Place your ballot in the envelope, seal it, and place the envelope in the ballot box in your curricular office.

COMPILATION: Ballot information will be compiled by an impartial handling source at the Naval Postgraduate School. The award winner will be announced during the June 2001 graduation ceremonies.

REMINDER: For your ballot to be counted, it must be returned in its entirety (omit the cover letter) and adhere to the following criteria:

- (a) you circled at least **five** but no more than **twenty-five** identification numbers assigned to Eligible Faculty.
- (b) those you selected in the Top 3 Choices section must have been among the five to twenty-five you circled in the Eligible Faculty section of the ballot.
- (c) you completed and returned your ballot by **April 30, 2001**.

Thank you for your participation.

LIST OF ELIGIBLE FACULTY

This list includes faculty who have met the general eligibility criteria of teaching three or more courses totaling eleven or more quarter hours during calendar year 2000. Members of the selection committee and past award winners are not eligible and are therefore not listed.

DEPARTMENT OF MATHEMATICS

2206	Borges, C.	2473	Fredricksen, H.M.	8259	Neta, B.
0972	Canright, D.	2251	Frenzen, C.	2440	Owen, G.
2217	Fahroo, F.	8226	Gragg, W.	2451	Rasmussen, C.
2240	Franke, R.H.	2295	Jayachandran, T.	8260	Scandrett, C.

DEPARTMENT OF MECHANICAL ENGINEERING

4657	Calvano, C.	8154	Gopinath, A.	2673	Kwon, Y.
8587	Dutta, I.	2534	Gordis, J.	2695	Millsaps, K.

					T.
8598	Fox, A.	2662	Healey, A. J.	0866	Papoulias, F.
1331	Gardner, S.	2606	Kelleher, M. D.		

DEPARTMENT OF METEOROLOGY

4585	Durkee, P.A.	0772	Nuss, W. A.	7001	Williams, R.T.
5061	Humiston, R.				

DEPARTMENT OF NATIONAL SECURITY AFFAIRS

8859	Brown, R. M.	1986	Looney, R.	1964	Robinson, G.
6781	Bruneau, T.	7223	Magnus, R. H.	7256	Roessler, T.
8293	Callahan, M.	8871	Minott, R.	5177	Roy, D.
0538	Ghoreishi, A.	6736	Moran, D.	4341	Trinkunas, H.A.
3959	Giraldo, J.	2062	Olsen, E. A.	8343	Tsyppkin, M.
6608	Lawson, L.	2639	Patenaude, B. M.	8882	Wirtz, J.
				7067	Yost, D. S.

DEPARTMENT OF OCEANOGRAPHY

6992	Collins, C.A.	0383	Jones, H. C.	0811	Semtner, A.
8132	Herbers, T.	0411	Poulain, P.		

DEPARTMENT OF OPERATIONS RESEARCH

8271	Anderson, T.	6275	Hoivik, T. H.	5955	Maher, K.
5722	Bradley, G.	3209	Jacobs, P. A.	6008	Olwell, D. H.
5300	Brown, R.	9124	Kerman, M.	3382	Taylor, J. G.
8648	Buss, A.	1364	Koyak, R.	3410	Washburn, A. R.
1025	Buttrey, S.	6970	Krebs, W.	8510	Whitaker, L. D.
3043	Dell, R.	2701	Lucas, T.	4596	Wood, K.
8787	Eagle, J.N.	5322	MacKinnon, D.		

DEPARTMENT OF PHYSICS

3454	Armstead, R.L.	7612	Hofler, T.	6947	Olsen, R.C.
6914	Baker, S.	3560	Holland, D.	4318	Smith, K.
3487	Cooper, A. W.	5977	Larrazza, A.	4507	Walters, D.L.
7278	Denardo, B.C.	8093	Luscombe, J.		
4802	Harney, R.C.	8387	Maruyama, X.K.		

DEPARTMENT OF SYSTEMS MANAGEMENT

6392	Boudreau, MW.	1270	Fremgen, J.M.	6208	McCaffery, J.L.
4291	Brinkley, D.	0350	Gue, K.	6220	Mehay, S.L.
3504	Cook, M. B.	5294	Haga, W.J.	9418	Naegle, B.
8404	Crawford, A.	1308	Hildebrandt, G.	7045	Nissen, M.
3137	Cuskey, J.	1236	Hleba, T.	4424	Owen, W.
0033	Edwards, L.	1014	Jansen, E.	4374	Roberts, N.C.
4396	Eitelberg, M.J.	6770	Kang, K.	8004	San Miguel, M.J.
1436	Euske, K.J.	3548	Kendry, A.	0572	Simon, C.
1447	Evered, R.D.	7645	Liao, S.S.	4963	Smith, D.
5399	Feitler, J.	0594	Matthews, D.		

ACADEMIC GROUPS

5144	Arquilla, J.	5449	Jansen, L.B.	3298	Porter, G.
0288	Brady, T.L.	3065	Kemple, W.	1386	Schleher, C.
3032	Brutzman, D.	7090	Kendal, T.	5472	Simons, A.
0022	Buddenberg, R.	7690	Lober, G.	2323	Tucker, D.
2189	Feely, M.	8037	McCormick, G.	3193	Zolla, G.
8798	Gibson, J.	7689	Marvel, O.		

AVIATION SAFETY SCHOOL

7573	Cooper, G.	8710	Figlock, R.	2017	Schmidt, J.
1342	Keane, C.	7267	Robey, S.	8071	Wood, B.

DEFENSE RESOURCES MANAGEMENT INSTITUTE

4307	Angelis, D.	0466	Felli, J	8626	Rose, D
3171	Blandin, J	7151	Frederiksen, P	5450	Stage, R
4257	Bonsper, D	8693	Hurst, S	1997	Standrowicz, S
3760	Boynton, R	6981	LaCivita, C	3876	Vaughan, L
4163	Costain, P	8282	Melese, F	3926	Webb, N
5249	Czarnecki, J	4280	Morris, J		
3793	Dawson, J	3904	Polley, A		

DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS

1070	Biblarz, O.	1175	Kaminer, I.	4530	Wood, E. R.
5283	Duren, R.	9546	Newberry, C. F.	6464	Wu, E. M.
0183	Hebbar, S.K.	1164	Platzer, M.F.		

DEPARTMENT OF COMPUTER SCIENCE

4941	Bachmann, E.	1720	Hensgen, D.	6664	McGhee, R. B.
5827	Baer, W.	5311	Holden, M.	2006	McGregor, D.
6620	Berzins, V.	0644	Irvine, C.	7245	Michael, B.
1353	Clark, P.	0905	Kern, D.	3676	Rowe, N.
2112	Eagle, C.	4079	Lewis, T.	0927	Shing, M.
6031	Falby, J.	0622	Lundy, G. M.	4479	Shultes, B.

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

1258	Adamiak, D.	0716	Hippenstiel, R.	1575	Pieper, R.
1531	Ashton, R.	6242	Hutchins, R.	1681	Robertson, R.C.
5100	Barsanti, R.	5777	Ives, R.	6569	Terman, F. W.
0677	Butler, J.	1564	Jenn, D.	8554	Wadsworth, D.
8932	Ciezki, J.	6969	Loomis, H. H.	5366	Weatherford, T.
6958	Cristi, R.	9335	McEachen, J.	1892	Wight, R. C.
1586	Fouts, D.	4446	Michael, S.	5733	Yun, X.
2039	Garcia, V.	7812	Pace, P.	9052	Ziomek, L. J.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. DATA ENTRY INSTRUCTIONS

These are the instructions for entering a ballot image into FOCUS.

INTRODUCTION

The Schieffelin ballot information is entered into the Schieffelin Teaching Award System that is located in the FOCUS mainframe.

ACCESSING THE MAINFRAME

To access the mainframe, double click on the **mainframe icon** on the computer desktop. Enter mainframe account number and password.

EXITING THE MAINFRAME

To exit the mainframe, **press F3**, a blank screen will appear. At the cursor prompt (located at the lower left-hand corner of the screen) type **FIN** and **press ENTER**. Then type **LOG** and **press ENTER**. You will return to the computer desktop.

SCHIEFFELING TEACHING AWARD SYSTEM'S MAIN MENU SCREEN

NAVAL POSTGRADUATE SCHOOL	
SCHIEFFELIN TEACHING AWARD SYSTEM	

TA DATA ADMINSTRATOR MENU	
1. ADD A BALLOT	
2. CHANGE BALLOT DATA	
3. DELETE A BALLOT	
4. CREATE BALLOT INPUT FILE	
5. PRINT RESULTS (2 REPORTS)	
Selection:	
Press enter - continue	F3 - Exit

FILE: c:\schief\bal_inst.doc
Modified: 1 April 99 (step 3: enter 4 digit plus alpha character; step 4 enter "0" for no comments or "1" for comments
STEP 1: Access mainframe

Double click on mainframe icon.

Enter userid number: **6295p**

Enter password and press ENTER. The following on screen message will appear.

**"NO PASSWORD NEEDED YOU ARE IN THE INTEGRATED DATABASE
PRESS ENTER TO CONTINUE"**

STEP 2: Add a Ballot

The following **Main Menu Screen** will appear.

NAVAL POSTGRADUATE SCHOOL	
SCHIEFFELIN TEACHING AWARD SYSTEM	

TA DATA ADMINSTRATOR MENU	
1. ADD A BALLOT	
2. CHANGE BALLOT DATA	
3. DELETE A BALLOT	
4. CREATE BALLOT INPUT FILE	
5. PRINT RESULTS (2 REPORTS)	
Selection:	
Press enter - continue	F3 - Exit

Select **Option 1** (Add a Ballot) and **press ENTER** to continue.

STEP 3: Enter Ballot ID Number

The **Ballot Add Screen** should be shown as below.

```
*****
NAVAL POSTGRADUATE SCHOOL
SCHIEFFELIN TEACHING AWARD SYSTEM
*****

      BALLOT ADD SCREEN

      BALLOT ID:

      PRESS ENTER TO CONTINUE ... PF03 TO CANCEL
```

Enter the Ballot ID number (four digit plus alpha character - i.e. 0001A, 0002S). Ballot ID number is located in the upper right hand corner of each ballot.

Press ENTER to continue.

STEP 4: Enter Ballot Information

The **Add a Ballot Record** screen should be shown as below.

NAVAL POSTGRADUATE SCHOOL			
SCHIEFFELIN TEACHING AWARD SYSTEM			

- ADD A BALLOT RECORD -			
CURM:			
COMMENT:	VOTE1:	VOTE2:	
VOTE3:	VOTE4:	VOTE5:	
VOTE6:	VOTE7:	VOTE8:	VOTE9:
VOTE10:	VOTE11:	VOTE12:	
VOTE13:	VOTE14:	VOTE15:	
VOTE16:			

CURM FIELD: Enter the two digit curriculum number circled (i.e. 30 for Operations Research). This number will be circled under the Curriculum Area.

COMMENT FIELD: Enter "0" for no comments or "1" for comments.

VOTE1 thru VOTE3 FIELDS: Enter the four- digit identification numbers selected for the **first, second and third choice nominees**. If any of the top three choices are left blank, tab over to the appropriate VOTE field and enter in the next four-digit identification number that corresponds to the top three choices. If all the top three choices are left blank, tab over to VOTE4 and begin entering the four-digit identification numbers circled from the "List of Eligible Faculty".

VOTE4 thru VOTE16 FIELDS: Enter the four-digit identification numbers circled from the "List of Eligible Faculty" and press enter to continue.

Note: If there are more than 13 four -digit identification numbers circled then the remaining four -digit identifications numbers need to be entered in the **Ballot Continuation Card Add Screen**.

The following screen will appear and ask, "Do you want to add a continuation card?"

This screen automatically defaults to "Y" for yes, unless it is overridden with a "N" for no.

```
DO YOU WANT TO ADD A CONTINUATION CARD? ( Y OR N )   - Y -  
PRESS ENTER TO CONTINUE OR PF03 TO QUIT
```

If response is no, enter a "N" or if response is yes, enter "Y".

Press ENTER to continue or press F3 to Quit.

If response entered is a "N", you will return to the **Ballot Add Screen**. You may now add a new ballot. Follow steps 3 and 4 again.

If response is a "Y", the **Ballot Continuation Card Add Screen** should appear as shown below.

```
BALLOT CONTINUATION CARD ADD SCREEN
```

```
BALLOT ID:      0001A
```

```
BALLOT CONTINUATION CARD:
```

```
PRESS ENTER TO CONTINUE OR PF3 TO QUIT
```

Enter the Ballot ID number after the **BALLOT CONTINUATION CARD PROMPT** and **press ENTER** to continue.
Enter remaining four-digit ballot identification numbers in the Ballot Continuation Card.

The **Ballot Continuation Card Screen** should appear as shown below.

BALLOT ID: 0001A			
CONTINUATION CARD: 0001A			
CVOTE1:	CVOTE2:	CVOTE3:	
COTE4:	CVOTE5:	CVOTE6:	CVOTE7:
CVOTE8:	CVOTE9:	CVOTE10:	CVOTE11:
CVOTE12:			
ANOTHER BALLOT ID (Y OR N) ? - Y -			
PRESS ENTER TO CONTINUE		PF3 TO QUIT	

CVOTE1 thru CVOTE12: Enter the remaining four- digit identification numbers from the "List of Eligible Faculty".

Follow remaining on screen instructions.

ANOTHER BALLOT ID (Y OR N)? To add a new ballot, enter "Y" and press ENTER. You will return to the **Ballot Add Screen**. Follow Steps 3 and 4 again.

To quit. Enter "N" and press F3.

EDIT BALLOT DATA:

To edit ballot data, select **Option 2** from the **Main Menu Screen**. The **Edit A Ballot Record Screen** will appear as shown below.

EDIT A BALLOT RECORD

CURM: 64

COMMENT: 0 VOTE1: 1234 VOTE2: 5678

VOTE3: 9134 VOTE4: 0987 VOTE5: 6543 VOTE6: 3210

VOTE7: 2345 VOTE8: 6789 VOTE9: 7890 VOTE10: 3456

VOTE11: 4455 VOTE12: 1122 VOTE13: 3344

VOTE14: 5566 VOTE15: 7788 VOTE16: 9900

EDIT A CONTINUATION CARD (Y OR N) ? - Y

PRESS ENTER TO CONTINUE OR PF3 TO QUIT

Make your changes in the appropriate fields. Follow remaining on screen instructions.

DELETING A BALLOT:

To delete a ballot, select **Option 3** from the **Main Menu Screen**. The **Ballot Delete Screen** will appear as shown below. Follow on screen instructions.

TA BALLOT DELETE SCREEN

PLEASE ENTER BALLOT ID YOU WISH TO DELETE -

PRESS ENTER TO DELETE OR F3 TO QUIT

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. LOGICAL MODEL

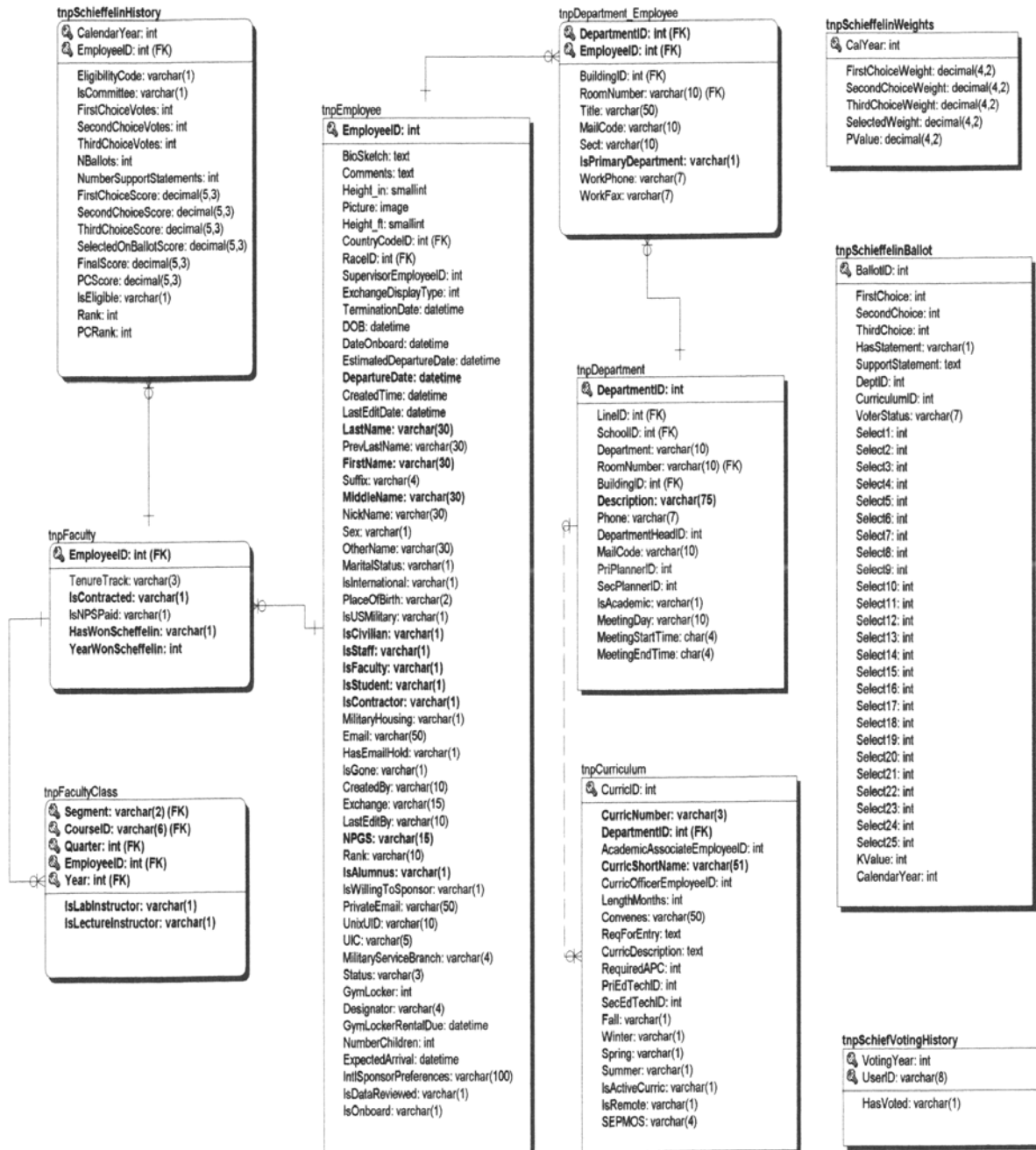


Figure 17. Logical Model

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. DATA DICTIONARY

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
tnpCurriculum	All Curriculums at NPS	CurricID	integer	4	Auto Number Field (PK)
		CurricNumber	varchar	3	Assigned Curriculum Number i.e. "360" for Operations Analysis
		DepartmentID	integer	4	The ID of the Department the curriculum belongs to
		CurricShortName	varchar	51	Assigned Curriculum Name ie "Operations Analysis"
tnpDepartment	All Departments at NPS	DepartmentID	integer	4	Auto Number Field (PK)
		Department	varchar	10	Assigned Department Designator i.e. "001 for Superintendents Office or NS for National Security Affairs
		Description	varchar	75	Title of Department: "Director of Student Affairs"
tnpDepartment_Employee	Employees that are in the Department	DepartmentID	integer	4	
		EmployeeID	integer	4	
		IsPrimaryDepartment	varchar	1	Assigns an employee to the principle department he/she belongs to when the employee works within more than one department.
		EmployeeID	integer	4	Auto Number Field (PK)
tnpEmployee	All personnel assigned to NPS	DepartureDate	datetime	8	Date employee left NPS and was no longer employed there
		LastName	varchar	30	Employees Last Name
		FirstName	varchar	30	Employees First Name
		MiddleName	varchar	30	Employees Middle Name or Initial
		IsCivilian	varchar	1	Checked when the employee is a civilian

Table 11. Data Dictionary

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
		IsStaff	varchar	1	Checked when the employee is staff
		IsFaculty	varchar	1	Checked when the employee is faculty
		IsStudent	varchar	1	Checked when the employee is a student
		IsContractor	varchar	1	Checked when the employee is a contractor
		NPGS	varchar	15	NPGS Domain UserID
		IsAlumnus	varchar	1	Checked if the employee is a graduate of NPS
tnpFaculty	An employee that teaches at NPS	EmployeeID	integer	4	
		IsContracted	varchar	1	Checked if the faculty is a contracted faculty
		HasWonScheffelin	varchar	1	Previous winner of the Schieffelin Award
		YearWonScheffelin	integer	4	Year the faculty member won the Schieffelin Award
tnpFacultyClass	Records the courses a faculty member taught during a particular quarter.	Segment	varchar	2	Identifies a particular course when the course is taught more than once during a quarter
		CourseID	varchar	6	Identifies a course i.e. "OS3101"
		Quarter	integer	4	The academic quarter (1,2,3 or 4)
		EmployeeID	integer	4	
		Year	integer	4	The year the course was taught
		IsLabInstructor	varchar	1	Checked if the course requires a lab

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
		IsLectureInstructor	varchar	1	Checked if the course requires lecture time
tnpSchieffelinBallot	Captures the demographics about voters and collects voter preferences about eligible faculty so that a ranked listing of professors can be calculated based on those preferences.	BallotID	integer	4	Auto Number Field (PK)
		FirstChoice	integer	4	Contains the employeeID of the candidate selected as the #1 candidate on this Ballot
		SecondChoice	integer	4	Contains the employeeID of the candidate selected as the #2 candidate on this Ballot
		ThirdChoice	integer	4	Contains the employeeID of the candidate selected as the #3 candidate on this Ballot
		HasStatement	varchar	1	Indicates whether there is a supporting statement for the FirstChoice on this Ballot
		SupportStatement	text	16	The voters supporting statement for his/her first choice on the ballot
		DeptID	integer	4	The ID of the Department the voter was or is a member of
		CurriculumID	integer	4	The Curriculum the voter was or is a member of
		VoterStatus	varchar	7	Indicates whether the voter is a Student or Alumnus

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
		Select1	integer	4	Select 1- Select 25 are the EmployeeIDs of all candidates identified by the voter
		Select2	integer	4	
		Select3	integer	4	
		Select4	integer	4	
		Select5	integer	4	
		Select6	integer	4	
		Select7	integer	4	
		Select8	integer	4	
		Select9	integer	4	
		Select10	integer	4	
		Select11	integer	4	
		Select12	integer	4	
		Select13	integer	4	
		Select14	integer	4	
		Select15	integer	4	
		Select16	integer	4	
		Select17	integer	4	
		Select18	integer	4	
		Select19	integer	4	
		Select20	integer	4	
		Select21	integer	4	
		Select22	integer	4	
		Select23	integer	4	
		Select24	integer	4	
		Select25	integer	4	

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
		KValue	integer	4	The number of candidates identified on the ballot
tnpSchieffelinHistory	Maintains the results of past balloting years.	CalendarYear	integer	4	Links an employees Schieffelin History to a particular calendar year
		EmployeeID	integer	4	
		EligibilityCode	varchar	1	Designates an employee's eligibility during a calendar year i.e. "I" for Ineligible, "E" for Eligible, "A" for top 5% or "B" for top 15%
		IsCommittee	varchar	1	
		FirstChoiceVotes	integer	4	The number of times the employee was identified as a first choice during that years balloting
		SecondChoiceVotes	integer	4	The number of times the employee was identified as a second choice during that years balloting
		ThirdChoiceVotes	integer	4	The number of times the employee was identified as a third choice during that years balloting
		NBallots	integer	4	The number of ballots the employee was identified on during that years balloting
		NumberSupportStatements	integer	4	The number of support statements the employee received for first choice votes during that balloting year

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
		FirstChoiceScore	decimal		The score (Z1) the employee received during that balloting year for all ballots he/she was selected as a first choice.
		SecondChoiceScore	decimal	5	The score (Z2) the employee received during that balloting year for all ballots he/she was selected as a second choice.
		ThirdChoiceScore	decimal	5	The score (Z3) the employee received during that balloting year for all ballots he/she was selected as a third choice.
		SelectedOnBallotScore	decimal	5	The score (Z4) the employee received during that balloting year for all ballots he/she was identified on (appeared in Select 1-Select 25).
		FinalScore	decimal	5	The total score for the employee for that balloting year
		PCScore	decimal	5	The Paired Comparison score for that balloting year
		IsEligible	varchar	1	Indicates Yes or No if the employee is eligible or not eligible for the award during that year
		Rank	integer	4	The ordering of the employees based on highest to lowest Final Score
		PCRank	integer	4	The Paired Comparison Rank based on highest to lowest PCScore

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Entity Definition</u>	<u>Entity Attribute Name</u>	<u>Data Type</u>	<u>Length</u>	<u>Entity Attribute Definition</u>
tnpSchieffelinWeights	Holds the weights and "p" value for the scoring function.	CalYear	integer	4	The Year the Schieffelin Award Weights were updated
		FirstChoiceWeight	decimal	5	The value assigned to calculate the first choice score (w1)
		SecondChoiceWeight	decimal	5	The value assigned to calculate the second choice score (w2)
		ThirdChoiceWeight	decimal	5	The value assigned to calculate the third choice score (w3)
		SelectedWeight	decimal	5	The value assigned to calculate the selected on ballot score (w4)
		PValue	decimal	5	The value used to normalize the population of voters (p)
tnpSchiefVotingHistory	Tracks who has voted during each balloting year	VotingYear	integer	4	The year voting took place
		HasVoted	varchar	1	Indicates whether a voter has voted or not
		UserID	varchar	8	The NPGS Domain UserID

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Attribute Name</u>	<u>Attribute Is PK</u>	<u>Attribute Is FK</u>	<u>Attribute Column Is PK</u>	<u>Attribute Column Is FK</u>
tnpCurriculum	CurricID	Yes		Yes	
	CurricNumber	No		No	
	DepartmentID		Yes		Yes
	CurricShortName				
	CurricDescription				
tnpDepartment	DepartmentID	Yes		Yes	
	Department		No		No
	Description		No		No
	DepartmentID	Yes	Yes	Yes	Yes
tnpDepartment_Employee	EmployeeID				
	IsPrimaryDepartment				
	EmployeeID	Yes		Yes	
	DepartureDate				
	LastName				
	FirstName				
	MiddleName				
	IsCivilian				
	IsStaff				
	IsFaculty				
tnpEmployee	IsStudent				
	IsContractor				
	NPGS				
	IsAlumnus				
	EmployeeID	Yes	Yes	Yes	Yes
	IsContracted				
	HasWonScheffelin				

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Attribute Name</u>	<u>Attribute Is PK</u>	<u>Attribute Is FK</u>	<u>Attribute Column Is PK</u>	<u>Attribute Column Is FK</u>
tnpFacultyClass	YearWonScheffelin				
	Segment				
	CourseID				
	Quarter				
	EmployeeID				
	Year				
	IsLabInstructor	No	No	No	No
tnpSchieffelinBallot	IsLectureInstructor				
	BallotID	Yes		Yes	
	FirstChoice	No		No	
	SecondChoice				
	ThirdChoice				
	HasStatement				
	SupportStatement				
	DeptID				
	CurriculumID				
	VoterStatus				
	Select1				
	Select2				
	Select3				
	Select4				
	Select5				
	Select6				
	Select7				
	Select8				
	Select9				
	Select10				
	Select11				

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Attribute Name</u>	<u>Attribute Is PK</u>	<u>Attribute Is FK</u>	<u>Attribute Column Is PK</u>	<u>Attribute Column Is FK</u>
	Select12				
	Select13				
	Select14				
	Select15				
	Select16				
	Select17				
	Select18				
	Select19				
	Select20				
	Select21				
	Select22				
	Select23				
	Select24				
	Select25				
	KValue				
tnpSchieffelinHistory	CalendarYear	Yes		Yes	
	EmployeeID		Yes		Yes
	EligibilityCode	No	No	No	No
	IsCommittee				
	FirstChoiceVotes				
	SecondChoiceVotes				
	ThirdChoiceVotes				
	NBallots				
	NumberSupportStatements				
	FirstChoiceScore				
	SecondChoiceScore				

Table 11. Data Dictionary (cont)

<u>Entity Name</u>	<u>Attribute Name</u>	<u>Attribute Is PK</u>	<u>Attribute Is FK</u>	<u>Attribute Column Is PK</u>	<u>Attribute Column Is FK</u>
	ThirdChoiceScore				
	SelectedOnBallotScore				
	FinalScore				
	PCScore				
	IsEligible				
	Rank				
	PCRank				
tnpSchieffelinWeights	CalYear	Yes		Yes	
	FirstChoiceWeight	No		No	
	SecondChoiceWeight				
	ThirdChoiceWeight				
	SelectedWeight				
	PValue				
tnpSchiefVotingHistory	VotingYear	Yes		Yes	
	HasVoted				
	UserID				

Table 11. Data Dictionary (cont)

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F. WEB PAGE DEPENDENCIES

The tables in this appendix illustrate Web page dependencies. For example, the first Web page in the left column of the table is "compute_eligibility.asp". It depends on the stored procedure "snpDetermineSchieffelinEligibility" (2nd column) which depends on tnpSchieffeinHistory, snpSchieffelinIneligible, snpSchiefSegments and snpSchieffelinCreditHours. Each of those stored procedures have their own dependencies as illustrated. The idea is to be able to trace each of those dependencies to the table(s) in the SADBMS the Web page relies on to get data.

WEB PAGE DEPENDENCIES

<u>Web Page</u>	<u>Dependent on:</u>	<u>Dependent on:</u>	<u>Dependent on:</u>
Compute_eligibility.asp	snpDetermineSchieffelinEligibility	tmpSchieffelinHistory snpSchieffelinIneligible snpSchiefSegments snpSchieffelinCreditHours	tmpFaculty tmpSchieffelinHistory tmpFacultyClass vnpFacultyCount tmpFacultyClass tmpSchieffelinHistory vnpFacultyCount tmpFacultyClass tmpSchieffelinHistory vnpFacultyCount vnpSelect1-vnpSelect25 vnpFirstChoice vnpSecondChoice vnpThirdChoice vnpNumberSupportStatements tmpSchieffelinBallot tmpSchieffelinBallot tmpSchieffelinBallot tmpSchieffelinBallot tmpSchieffelinBallot tmpSchieffelinHistory vnpSchieffelinRanking tmpSchieffelinHistory
Compute_scores.asp	snpCallScoring	snpCountVotes snpKVvalues tmpSchieffelinBallot snpSchieffelinScoring snpSchieffelinRanks snpFinalScore	tmpSchieffelinBallot tmpSchieffelinHistory vnpSchieffelinRanking tmpSchieffelinHistory
Data_entry_eligibility_all.asp	tmpSchieffelinHistory tmpEmployee tmpDepartment_Employee tmpDepartment		
Data_entry_eligibility_dept.asp	tmpSchieffelinHistory tmpEmployee tmpDepartment_Employee tmpDepartment		
Data_entry_eligibility_lname.asp	tmpSchieffelinHistory tmpEmployee tmpDepartment_Employee tmpDepartment		
Data_entry_history.asp	tmpEmployee tmpSchieffelinHistory		

Table 12. Web Page Dependencies

Eligible_detailed.asp	vnpEligibleFaculty	<i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i> <i>tnpFacultyClass</i> <i>tnpClass</i> <i>tnpClass Type</i> vnpFacultycount vnpFacultycount vnpEligibleFaculty	<i>tnpFacultyClass</i> <i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
Eligible_simple.asp	snpViewEligibleFaculty	vnpEligibleFaculty	<i>tnpFacultyClass</i> <i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
Empid.asp	<i>tnpEmployee</i> <i>tnpDepartment</i> <i>tnpDepartment_Employee</i> vnpSchieffHistory		
History.asp	vnpSchieffHistory	<i>tnpSchieffelinHistory</i> vnpSchieffelinRanking <i>tnpEmployee</i>	<i>tnpSchieffelinHistory</i>
History1.asp	vnpSchieffHistory	<i>tnpSchieffelinHistory</i> vnpSchieffelinRanking <i>tnpEmployee</i>	<i>tnpSchieffelinHistory</i>
Lname.asp	<i>tnpEmployee</i> <i>tnpDepartment</i> <i>tnpDepartment_Employee</i> vnpSchiefflinRanking <i>tnpEmployee</i>		
Rank1.asp	vnpSchiefflinRanking	<i>tnpSchieffelinHistory</i>	
Rank2.asp	vnpSchiefflinRanking	<i>tnpSchieffelinHistory</i>	
Update.asp	<i>tnpSchieffelinWeights</i>		
Update_weights.asp	<i>tnpSchieffelinWeights</i>		
Weights.asp	<i>tnpSchieffelinWeights</i>		
Ballot.asp	<i>tnpSchieffVotingHistory</i> snpViewEligibleFaculty	vnpEligibleFaculty	<i>tnpDepartment</i> <i>tnpEmployee</i>

Table 12. Web Page Dependencies

Ballot25.asp	<i>tnpSelectVotingHistory</i> <i>tnpSchieffelinBallot</i>	<i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
Ballot3.asp	<i>tnpSchieffVotingHistory</i> <i>tnpSchieffelinBallot</i> vnpSchieffCurricID	<i>tnpCurriculum</i> <i>tnpSchieffelinBallot</i>
BallotSummary.asp	<i>tnpSchieffelinBallot</i> vnpSchieffFirstChoice	<i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
	vnpSchieffSecondChoice	<i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
	vnpSchieffThirdChoice	<i>tnpDepartment</i> <i>tnpEmployee</i> <i>tnpSchieffelinHistory</i> <i>tnpDepartment_Employee</i> <i>tnpFaculty</i>
Login.asp	snpGetLogin	<i>tnpEmployee</i> <i>tnpPythonProfile</i>

Table 12. Web Page Dependencies

APPENDIX G. APPLICATION CODE

This appendix contains the HTML and ASP code for the Web pages of the Schieffelin Award Database Management System. Each sub-title below is the file name of the page, followed by the code that makes up the page.

A. INDEX.ASP

```
<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="760"
height="1">
  <tr>
    <td width="160" height="3"></td>
    <td width="1175" height="3"></td>
    <td width="79" height="3"></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1">
      <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 0px"
align="center"><font face="Verdana, Arial, Helvetica, sans-serif"
color="lightslategray" size="1"><strong></strong> </font>&nbsp;</p>
      <p style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 0px"
align="center"><font face="Verdana, Arial, Helvetica, sans-serif"
color="lightslategray" size="1"><strong>Naval Postgraduate School<br>1
University Circle<br>Monterey, CA
93943-5001<br>(831) 656-2441/2<br>DSN:
878-2441/2</strong></font></p><a
href="http://www.nps.navy.mil"></a>
      <p><font color="lightslategray"></font>&nbsp;</p>
      <p><br></p>
    </td>
    <td rowspan="2" valign="top" height="1">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">
            <div align="center"></div>
          </td>
        </tr>
      </table>
    </td>
  </tr>
</table>
```

```

    </tr>
    <tr>
        <td width="575" valign="top">
            <center>
                <hr>
                <p style="MARGIN-TOP: 3px; MARGIN-BOTTOM: 3px"
align="center"><font color="#ff0000"><b>
    PROTOTYPE Schieffelin Award Balloting System</b></font></p>
                <p style="MARGIN-TOP: 3px; MARGIN-BOTTOM: 3px"
align="left"><font face="Verdana"><font color="black"><span style="mso-
bidi-font-size: 11.0pt"><font face="Verdana, Arial, Helvetica, sans-
serif">From
                    the many excellent teachers at the Naval Postgraduate School,
one
                    professor each summer wins the distinction of being selected
as
                    recipient of the Admiral John Jay Schieffelin Award for
Excellence in
                    Teaching. This award honors the School's most <b>outstanding
faculty
                    instructor</b>.</font>&nbsp;</span><span style="mso-bidi-
font-size: 12.0pt">
                    More information about the Schieffelin Award, including a list
of past
                    winners, can be viewed <a
href="http://www.nps.navy.mil/SAward/JJSAward.htm"><b>here</b></a>.
                    </o:p>
                </span></font></font></p>
                <hr>

                <p style="MARGIN-TOP: 3px; MARGIN-BOTTOM: 3px"
align="center"><font color="#ff0000"><b>INSTRUCTIONS</b></font></p>
                <p style="MARGIN-TOP: 3px; MARGIN-BOTTOM: 3px"
align="left"><font face="Verdana" color="black">A list of
                    eligible faculty listed by their primary department appears
on the
                    pages following this one.&nbsp;</font>

                    <a href="/SAward/login.asp"><strong><font face="Verdana"
color="black">enter the Ballot</font>
                    </strong></a><font face="Verdana" color="black"> any time
you like.&nbsp;</font> Generally,
                    eligibility for this award is established by teaching a
minimum of
                    three courses totaling eleven or more quarter hours during
the
                    calendar year of the Award.&nbsp;</font> Award committee members
and
                    previous winners are ineligible for consideration.&nbsp;</font> In
conjunction with specifying your Top Choice (s) for this
award, you
                    are asked to identify at least five (5)&nbsp;</font>and no more
than
                    twenty-five (25)&nbsp;</font>Eligible Faculty on which your Top
Choice (s)

```

```

        ranking applies.</font>
    </p>
    <p style="MARGIN-TOP: 3px; MARGIN-BOTTOM: 3px" align="left">
<font>
    <hr>

</font></p>
    <p></p>
    <p align="center"><b><font color="#000099"
face="Arial">Alumni Login
Here</font></b></p>
    <p align="center">
    <hr align="center">
    </p>

    </td>
</tr>
<tr>
    <td width="575">
    </td>
</tr>
<tr>
    <td width="575">
        <table width="100%" border="0">
            <tr>
                <td width="30%">
                    <div align="center"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">Revision Date: <!--webbot
bot="Timestamp" S-Type="EDITED" S-Format="%d %b %Y" startspan -->02 Sep
2001<!--webbot bot="Timestamp" endspan i-checksum="14928" -->
                    </font></div>
                </td>
                <td width="34%">
                    <div align="center"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">Contact
                    <a
href="mailto:samaglia@nps.navy.mil">Webmaster</a></font></div>
                </td>
                <td width="20%">
                    <div align="center"><a
href="http://intranet.nps.navy.mil"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">NPS
                    Intranet</font></a></div>
                </td>
                <td width="16%">
                    <div align="center"><a
href="https://itwarrior.nps.navy.mil/exchange/logon.asp"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">NPS
                    email</font></a></div>
                </td>
            </tr>
        </table>
    </td>
</tr>
</table>
    <td width="79" height="1"></td>

```

```

</tr>
<tr>
  <td valign="top" align="left" bordercolor="#ff0000" height="1">
    <p align="center">&nbsp;</p>
    <p align="center">&nbsp;</p>
    <p align="center">&nbsp;</p>
  </td>
</tr>
</table>
</body>
</html>
-->

```

B. BALLOT.ASP

```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<!--This page reads from the view (vnpEligibleFaculty) and writes all
eligiblefaculty by department to the page with an associated checkbox
that is linked to the candidates employeeID. The employeeID is then
used to track which candidatesare being selected thourghout the rest of
the balloting pages (ballot25.asp, ballot3.aspand the ballotsummary.asp
pages-->
<script id="DebugDirectives" runat="server" language="javascript">
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</script>
<%
Dim intDate
'Subtracts 1 year from the current year to produce Eligible Faculty for
the correct year
intDate = Year(Date)-1

Set RsHasVoted = server.CreateObject ("ADODB.recordset")
strHasVoted = "SELECT VotingYear, UserID, HasVoted FROM
tnpSchiefVotingHistory;"
RsHasVoted.Open strHasVoted, Application("saward_ConnectionString")
'variable to store user id
Dim strUserID

'sets strNPGS = to the current user's userID
strUserID = Request.ServerVariables ("AUTH_USER")

'Checks the voting history table (tnpSchiefVotingHistory)
to see if the
'current voter has already voted for this years ballot. If
yes, he is redirected
'to fail1.html and notified that he has already voted.
If not RsHasVoted.EOF then
do while not RsHasVoted.EOF
If RsHasVoted ("UserID") = strUserID and RsHasVoted
("VotingYear") = intDate then
Response.Redirect ("fail1.html")

```



```

        end if
        RsHasVoted.MoveNext
    loop
end if

RsHasVoted.close ()
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Dim strSQL
    strSQL = "Exec snpViewEligibleFaculty "& intDate
%>
<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="880"
height="1">
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="162">
            <p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1"><b>
                &nbsp;Naval Postgraduate School<br>
                &nbsp;1 University Circle<br>
                &nbsp;Monterey, CA 93943-5001<br>
                &nbsp;(831) 656-2441/2<br>
                &nbsp;DSN: 878-2441/2</b></font></p>
            <p align="left">&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="720" border="0" align="center">
                <tr>
                    <td width="720">
                        <div align="center"></div>
                    </td>
                </tr>
                <tr>
                    <td width="720">

```

<form action="ballot25.asp" id="frmVote" method="post" name="Form1">

<p>Below is a list of faculty who are eligible for the Schieffelin Award for Excellence in Teaching. Generally, eligibility for this award is established by teaching a minimum of three courses totaling eleven or more quarter hours during the calendar year of the award. Award committee members and previous winners are ineligible.

</p>

<p>From the following list of Eligible Faculty, you are asked to select at least FIVE (5) but no more than TWENTY-FIVE (25) faculty members whose teaching abilities you know on a first-hand basis.</p>


```

        <div align="center"><a
href="http://intranet.nps.navy.mil"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">NPS
        Intranet</font></a></div>
    </td>
    <td width="16%">
        <div align="center"><a
href="https://itwarrior.nps.navy.mil/exchange/logon.asp"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">NPS
        email</font></a></div>
    </td>
</tr>
</table>
</td>
</tr>
</table>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

C. BALLOT25.ASP

```

<!--
<%@ Language=VBScript %>
<!--The purpose of this page is to collect all the professors who were
identifiedby the voter on ballot.asp and write them to this page to
allow the voter toselect his top 3 choices. The final portion of this
page then opens a recordsetthat captures all the candidates identified
by the voter and adds them to a uniquerecord in the ballot
(tnpSchieffelinBallot) table as choices 1-25 ("select1- select25).-->
<%
Dim intDate
'Subtracts 1 year from the current year to produce Eligible Faculty for
the correct year
intDate = Year(Date)-1

Set RsHasVoted = server.CreateObject ("ADODB.recordset")
strHasVoted = "SELECT VotingYear, UserId, HasVoted FROM
tnpSchiefVotingHistory;"
RsHasVoted.Open strHasVoted, Application("saward_ConnectionString")

Dim strUserID 'variable to store user id
strUserID = Request.ServerVariables ("AUTH_USER") 'sets
strNPGS = to

        'current user's userID
        'Checks the voting history table (tnpSchiefVotingHistory)
to see if the
        'current voter has already voted for this years ballot. If
yes, he is redirected
        'to fail1.html and notified that he has already voted.

```

```

If not RsHasVoted.EOF then
    do while not RsHasVoted.EOF
        If RsHasVoted ("UserID") = strUserID and RsHasVoted
("VotingYear") = intDate then
            Response.Redirect ("fail1.html")
        end if
        RsHasVoted.MoveNext
    loop
end if
RsHasVoted.close ()

Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Dim iCount, i                'iCount is the total number of checkboxes
(eligible faculty) displayed on view.asp
Dim jCount                  'tracks who was selected as top 3
candidates

Dim RsBallot
Dim strSql
Dim adOpenDynamic,adlockPessimistic
adOpenDynamic=2
adLockPessimistic = 2
Set RsBallot = Server.CreateObject ("ADODB.Recordset")
strSql = "SELECT * FROM tnpSchieffelinBallot;"
RsBallot.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
%>

<html>
    <head>
        <title>Schieffelin Teaching Award
        </title>
    </head>
    <body bgcolor="#ffffff">
        <table border="0" cellpadding="0" cellspacing="0" width="880"
height="1">
            <tr>
                <td valign="top" align="left" height="1"
bordercolor="#e3e4cd" width="162">
                    <p align="center"><font color="#000099"
face="Verdana, Arial, Helvetica, sans-serif" size="1"><b>
                        &nbsp;  Naval Postgraduate School<br>
                        &nbsp;  1 University Circle<br>
                        &nbsp;  Monterey, CA 93943-5001<br>
                        &nbsp;  (831) 656-2441/2<br>
                        &nbsp;  DSN: 878-2441/2</b></font></p>
                    <p align="left">&nbsp;  </p>
                </td>
                <td valign="top" height="1" width="602">
                    <table width="720" border="0" align="center">
                        <tr>
                            <td width="720">

```

```


136


```

```

        <input type="HIDDEN" name="txtFirstName<%=jCount%>"
value="<%=Request("txtFirstName" & i)%>">
        <input type="HIDDEN" name="txtMI<%=jCount%>"
value="<%=Request("txtMI" & i)%>">
        <input type="HIDDEN" name="txtID<%=jCount%>"
value="<%=Request("chkVote" & i)%>">
        <input type="HIDDEN" name="txtDepartmentName<%=jCount%>"
value="<%=Request("txtDepartmentName" & i)%>">
<%
    'This is probably the hard way of doing this but the following
group of if statements
    'add all of the employee ids of the selected checkboxes from
view.asp to the Ballot table
    'as votes 1 to 25.
    if jCount = 1 then

        RsBallot.AddNew
        RsBallot("Select1") = Request("chkVote" & i)
        RsBallot.Update

        'Gets the current record being updated from
tnpSchieffelinBallot
        strRsID = "Select Max(BallotID) As maxnum FROM
tnpSchieffelinBallot"
        Set RsID = objConn.Execute (strRsID)
        maxnum= RsID("maxnum").Value

    elseif jCount = 2 then
        Set RsBallotID2 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID2.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID2("Select2") = Request("chkVote" & i)
        RsBallotID2.Update
        RsBallotID2.Close ()

    elseif jCount = 3 then
        Set RsBallotID3 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID3.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID3("Select3") = Request("chkVote" & i)
        RsBallotID3.Update
        RsBallotID3.Close ()

    elseif jCount = 4 then
        Set RsBallotID4 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID4.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID4("Select4") = Request("chkVote" & i)
        RsBallotID4.Update
        RsBallotID4.Close ()

```

```

elseif jCount = 5 then
    Set RsBallotID5 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID5.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID5("Select5") = Request("chkVote" & i)
    RsBallotID5.Update
    RsBallotID5.Close ()

elseif jCount = 6 then
    Set RsBallotID6 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID6.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID6("Select6") = Request("chkVote" & i)
    RsBallotID6.Update
    RsBallotID6.Close ()

elseif jCount = 7 then
    Set RsBallotID7 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID7.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID7("Select7") = Request("chkVote" & i)
    RsBallotID7.Update
    RsBallotID7.Close ()

elseif jCount = 8 then
    Set RsBallotID8 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID8.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID8("Select8") = Request("chkVote" & i)
    RsBallotID8.Update
    RsBallotID8.Close ()

elseif jCount = 9 then
    Set RsBallotID9 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID9.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID9("Select9") = Request("chkVote" & i)
    RsBallotID9.Update
    RsBallotID9.Close ()

elseif jCount = 10 then
    Set RsBallotID10 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)

```



```

        RsBallotID10.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID10("Select10") = Request("chkVote" & i)
        RsBallotID10.Update
        RsBallotID10.Close ()

    elseif jCount = 11 then
        Set RsBallotID11 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID11.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID11("Select11") = Request("chkVote" & i)
        RsBallotID11.Update
        RsBallotID11.Close ()

    elseif jCount = 12 then
        Set RsBallotID12 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID12.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID12("Select12") = Request("chkVote" & i)
        RsBallotID12.Update
        RsBallotID12.Close ()

    elseif jCount = 13 then
        Set RsBallotID13 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID13.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID13("Select13") = Request("chkVote" & i)
        RsBallotID13.Update
        RsBallotID13.Close ()

    elseif jCount = 14 then
        Set RsBallotID14 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID14.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID14("Select14") = Request("chkVote" & i)
        RsBallotID14.Update
        RsBallotID14.Close ()

    elseif jCount = 15 then
        Set RsBallotID15 = Server.CreateObject ("ADODB.Recordset")
        strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
        RsBallotID15.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
        RsBallotID15("Select15") = Request("chkVote" & i)
        RsBallotID15.Update
        RsBallotID15.Close ()

```

```

elseif jCount = 16 then
    Set RsBallotID16 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID16.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID16("Select16") = Request("chkVote" & i)
    RsBallotID16.Update
    RsBallotID16.Close ()

elseif jCount = 17 then
    Set RsBallotID17 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID17.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID17("Select17") = Request("chkVote" & i)
    RsBallotID17.Update
    RsBallotID17.Close ()

elseif jCount = 18 then
    Set RsBallotID18 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID18.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID18("Select18") = Request("chkVote" & i)
    RsBallotID18.Update
    RsBallotID18.Close ()

elseif jCount = 19 then
    Set RsBallotID19 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID19.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID19("Select19") = Request("chkVote" & i)
    RsBallotID19.Update
    RsBallotID19.Close ()

elseif jCount = 20 then
    Set RsBallotID20 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID20.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallotID20("Select20") = Request("chkVote" & i)
    RsBallotID20.Update
    RsBallotID20.Close ()

elseif jCount = 21 then
    Set RsBallotID21 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID21.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic

```

```

RsBallotID21("Select21") = Request("chkVote" & i)
RsBallotID21.Update
RsBallotID21.Close ()

elseif jCount = 22 then
    Set RsBallotID22 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID22.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
RsBallotID22("Select22") = Request("chkVote" & i)
RsBallotID22.Update
RsBallotID22.Close ()

elseif jCount = 23 then
    Set RsBallotID23 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID23.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
RsBallotID23("Select23") = Request("chkVote" & i)
RsBallotID23.Update
RsBallotID23.Close ()

elseif jCount = 24 then
    Set RsBallotID24 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID24.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
RsBallotID24("Select24") = Request("chkVote" & i)
RsBallotID24.Update
RsBallotID24.Close ()

elseif jCount = 25 then
    Set RsBallotID25 = Server.CreateObject ("ADODB.Recordset")
    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID
= " & cint(maxnum)
    RsBallotID25.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
RsBallotID25("Select25") = Request("chkVote" & i)
RsBallotID25.Update
RsBallotID25.Close ()

end if
end if
next

'Since actual voting takes place one calendar year after the
actual period covered by the
'award we need to subtract 1 year from the correct calendar year
to put in the ballot table
'intDate = Year(Date)-1
Set RsCalendarYear = Server.CreateObject ("ADODB.Recordset")
strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID = " &
cint(maxnum)

```

```

RsCalendarYear.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
RsCalendarYear("CalendarYear") = intDate
RsCalendarYear.Update
RsCalendarYear.Close ()
%>
<%

RsBallot.Close ()

' This sets maxnum (BallotID in ballot table) to a variable to
pass to top3.asp
Dim BallotID
BallotID = cint(maxnum)
%>
<!--Passes the values of jCount and ICount to top3.asp-->



  
<input type="Submit" value="Select" id="SUBMIT1">
<p></p>
</form>

<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<hr>

</td>
</tr>
<tr>
  |
```

```


<a
href="https://itwarrior.nps.navy.mil/exchange/logon.asp"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">NPS
email</font></a></div>
</td>
</tr>
</table>
</td>
</tr>
</table>
</td>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
</body>
</html>
-->


```

D. BALLOT3.ASP

```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<%
    'The intDate is used to update the "VotingYear" in
tnpSchiefVotingHistory
    'it is part of the primary key and along with the voters
UserId,
    'prevents the voter from being able to access the ballot
pages a second time
    'i.e. prevents him/her from voting twice.
Dim intDate
    'Subtracts 1 year from the current year to produce Eligible
Faculty for the correct year
intDate = Year(Date)-1

Set RsHasVoted = server.CreateObject ("ADODB.recordset")
strHasVoted = "SELECT VotingYear, UserId, HasVoted FROM
tnpSchiefVotingHistory;"
RsHasVoted.Open strHasVoted, Application("saward_ConnectionString")

    'variable to store user id
Dim strID
    'sets strNPGS = to current user's userID (NPGS\UserID)
strID = Request.ServerVariables ("AUTH_USER")

    'If current voter has an entry in tnpSchiefVotingHistory,
this block
    'of code will redirect the user to fail1.html and not allow
him/her to vote.

if not RsHasVoted.EOF then
    do while not RsHasVoted.EOF

```

```

        If RsHasVoted ("UserID") = strUserID and RsHasVoted
("VotingYear") = intDate then
            Response.Redirect ("faill.html")
        end if
        RsHasVoted.MoveNext
    loop
end if
RsHasVoted.close ()

'Opens database connection
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Dim j, jCount, kCount, iCount                'Counters for number of
checkboxes
Dim maxnum                                    'Variable holding
last entered ballotID
Dim BallotID                                'Variable used to
capture ballotID from select25.asp

'Opens a recordset of all columns in the tnpSchieffelinBallot table
'and gets the current (last record in the table) record being updated
by the voter
Dim strRsID, RsID
strRsID = "Select Max(BallotID) As maxnum FROM tnpSchieffelinBallot"
Set RsID = objConn.Execute (strRsID)
maxnum= RsID("maxnum").Value

Dim strSql
Dim RsBallot
Set RsBallot = Server.CreateObject ("ADODB.Recordset")
strSql = "SELECT * FROM tnpSchieffelinBallot;"
RsBallot.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic

%>
<html>
    <head>
        <title>Schieffelin Teaching Award
        </title>
    </head>
    <body bgcolor="#ffffff">
        <table border="0" cellpadding="1" cellspacing="0" width="880"
height="1">
            <tr>
                <td valign="top" align="left" height="1"
bordercolor="#e3e4cd" width="162">
                    <p align="center"><font color="#000099"
face="Verdana, Arial, Helvetica, sans-serif" size="1"><b>
                        &nbsp;&nbsp;&nbsp;Naval Postgraduate School<br>
                        &nbsp;&nbsp;&nbsp;1 University Circle<br>
                        &nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
                        &nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>

```



```

</select>
    <%RsCurric.Close ()%>

<br>
<br><font face="Verdana">Rank your top choice(s) with 1st being the
most preferred and 3rd being the
least.</strong> <br> If no professors are listed, it is because you
opted
to not rank your choices. Ensuring you selected your voter category
and curriculum above,
press the submit button to continue.</font>

<table align="center" cellSpacing="1" cellPadding="1" border="0">
    <br>
    <tr>
        <th> 1st </th>
        <th> 2nd </th>
        <th> 3rd </th>
        <th ALIGN="center"> NAME </th>
        <th ALIGN="left"> DEPT </th>
    </tr>
<%
    'gets the candidates who were checked on ballot25.asp
for j = 1 to jCount
    kCount = kCount + 1
    if Request("chkVote" & j) = "" then%> <!--if the checkbox was
unchecked, get those candidates and put them in a HIDDEN          text
field-->
        <input type="HIDDEN" name="txtLastName<%=kCount%>"
value="<%=Request("txtLastName" & j)%>">
        <input type="HIDDEN" name="txtFirstName<%=kCount%>"
value="<%=Request("txtFirstName" & j)%>">
        <input type="HIDDEN" name="txtMI<%=kCount%>"
value="<%=Request("txtMI" & j)%>">
        <input type="HIDDEN" name="txtDepartmentName<%=kCount%>"
value="<%=Request("txtDepartmentName" & j)%>">
    <%
    else
%>
    <tr> <!--assigns the employee's ID to the radio button and
writes those canidates who were selected                on
ballot25.asp-->
        <td><input TYPE="RADIO" NAME="chkFirstVote"
VALUE="<%=Request("chkVote" & j)%>"></td>
        <td><input TYPE="RADIO" NAME="chkSecondVote"
VALUE="<%=Request("chkVote" & j)%>"></td>
        <td><input TYPE="RADIO" NAME="chkThirdVote"
VALUE="<%=Request("chkVote" & j)%>"></td>
        <td> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;& <%Response.Write
Request("txtLastName" & j) & ", "
                Response.Write Request("txtFirstName" & j) & "
"
                Response.Write Request("txtMI" & j) & ". "%></td>
        <td> &nbsp;&nbsp;&nbsp;&nbsp;& <%Response.Write
Request("txtDepartmentName" & j)%></td>
    </tr>

```



```

<!--puts candidate information in a hidden text area for use on
ballotsummary.asp-->


```

Support of your First Choice candidate. List those qualities which you feel make this person an outstanding teacher. If you don't not wish to make a statement, or you have not ranked your choices, you may leave the field blank.

```

        <input type="HIDDEN" name="txtCount2" value="<%=kCount%>">
<!--textarea to capture voters comments for his/her first choice-->
<textarea rows="6" cols="80" name="txtcomments" value>
</textarea>
        <br><input type="submit" value="Submit" id="SUBMIT1">
</form></td>
<td>

                &nbsp;   </td>

</tr>
<tr>
        <td>

                &nbsp;   </td>

        <td>

                <table width="100%" border="0">
<tr>
                        <td width="30%">
                                <div align="center"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">Revision Date:
<!--webbot bot="Timestamp" S-Type="EDITED" S-Format="%d %b %Y"
startspan -->02 Sep 2001<!--webbot bot="Timestamp" endspan i-
checksum="14928" -->

                                </font></div>
                        </td>
                        <td width="34%">
                                <div align="center"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">Contact
                                <a
href="mailto:samaglia@nps.navy.mil">Webmaster</a></font></div>
                        </td>
                        <td width="20%">
                                <div align="center"><a
href="http://intranet.nps.navy.mil"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">NPS
                                Intranet</font></a></div>
                        </td>
                        <td width="16%">
                                <div align="center"><a
href="https://itwarrior.nps.navy.mil/exchange/logon.asp"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">NPS
                                email</font></a></div>
                        </td>
                </tr>
        </table></td>
</tr>
</table>
<%RsBallot.Close ()%>
</body>
</html>-->

```

E. BALLOT_SUMMARY.ASP

```
<!--
<%@ Language=VBScript %>
<% Option Explicit%>
<!--#include virtual="/adovbs.inc"-->
<%
Dim objConn
Dim strRsID                         'Variable that gets the last record from
tnpSchieffelinBallot
Dim RsCurric, strCur             'Recordset and SQL statement
Dim RsID                         'Open Recordset to hold last record
Dim maxnum                        'This is the last BallotID entered into
tnpSchieffelinBallot
Dim RsBallot                       'Recordset for adding the First, Second
and Third choice votes to
                                   'tnpSchieffelinBallot
Dim strSQL                         'SQL statement
Dim strRsID                       'Variable that gets the last record from
vnpSchieffFirstChoice
Dim RsID                         'Open Recordset to hold last record
Dim kCount, k                     'counter to count number of check boxes
selected
Dim RsFirstChoice, strSQL1, RsSecondChoice, strSQL2, RsThirdChoice,
strSQL3
                                   'Recordsets and SQL statements to return the voters first, second
and third choices

Set objConn = Server.CreateObject("ADODB.Connection")
objConn.ConnectionString =Application("saward_ConnectionString")
objConn.Open

strRsID = "Select Max(BallotID) As maxnum FROM tnpSchieffelinBallot"
Set RsID = objConn.Execute (strRsID)
maxnum= RsID("maxnum").Value
Set RsBallot = Server.CreateObject ("ADODB.Recordset")

                                   'variable to capture voter category from combo boxes
Dim strVoterStatus
strVoterStatus = Request.Form ("cboCategory")

                                   'capture voter comments for first choice
Dim strComments
strComments = cstr(Request.Form ("txtComments"))
                                   'Removes the curric shortname from the cbocurric value
leaving just the CurricID
Dim Remove, strCurric, intCurricID
strCurric = Request.Form ("cboCurric")
Remove = Left(strCurric,2)

                                   'convert Remove to an integer
intCurricID= cint(Remove)
'open the ballot and get the last record and update it with the voters
'top 3 choices, comments, curriculum and voter status
```

```

    strSql = "SELECT * FROM tnpSchieffelinBallot WHERE BallotID = " &
cint(maxnum)
    RsBallot.Open strSql,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    RsBallot ("FirstChoice") = Request ("chkFirstVote")
    RsBallot ("SecondChoice") = Request ("chkSecondVote")
    RsBallot ("ThirdChoice") = Request ("chkThirdVote")
    RsBallot ("SupportStatement") = Request ("txtComments")
    RsBallot ("CurriculumID") = intCurricID
    If strComments = "" then
    RsBallot ("HasStatement") = "N"
    Else
    RsBallot ("HasStatement") = "Y"
    end if
    if strVoterStatus = 1 then
    RsBallot ("VoterStatus") = "Alumni"
    end if
    if strVoterStatus = 2 then
    RsBallot ("VoterStatus") = "Student"
    end if
    RsBallot.Update
    RsBallot.Close ()

```

```

strRs1ID = "Select Max(BallotID) As maxnum FROM vnpSchieffFirstChoice"
Set Rs1ID = objConn.Execute (strRs1ID)
    maxnum= Rs1ID("maxnum").Value

```

```

%>
<html>
    <head>
        <title>Schieffelin Teaching Award
        </title>
    </head>
    <body bgcolor="#ffffff">
        <table border="0" cellpadding="1" cellspacing="0" width="880"
height="1">
            <tr>
                <td valign="top" align="left" height="1"
bordercolor="#e3e4cd" width="162">
                    <p align="center"><font color="#000099"
face="Verdana, Arial, Helvetica, sans-serif" size="1"><b>
                        &nbsp;   Naval Postgraduate School<br>
                        &nbsp;   1 University Circle<br>
                        &nbsp;   Monterey, CA 93943-5001<br>
                        &nbsp;   (831) 656-2441/2<br>
                        &nbsp;   DSN: 878-2441/2</b></font></p>
                    <p align="left">&nbsp;   </p>
                </td>
                <td width="720">
                    <div align="center"></div>
                </td>
            <tr>
                <td>
                    &nbsp;   </td></tr></table>

```

```

<!--This block writes all of the candidates the voter identified so he
can see a summary of who he selected-->

```

```

<%
kCount = Request("txtCount2")%><blockquote dir="ltr" style="MARGIN-
RIGHT: 0px">
    <p><br>
    &nbsp;<font face="Verdana">Your Ballot
    reflects that you identified the following <%Response.Write
(kCount)%> professors:</font><br><br></p></blockquote>
<table align="center" border="0">
<%
for k = 1 to kCount
    %>
    <tr><td width="100">&nbsp;</td>
    <td width="200"><%Response.Write Request ("txtLastName" & k)%>
, &nbsp;<br>
    <%Response.Write Request ("txtFirstName" & k)%> &nbsp;<br>
    <%Response.Write Request ("txtMI" & k)%></td>
    <td><%Response.Write Request ("txtDepartmentName" & k)%></td><%
next
%></tr>
</table>
<blockquote dir="ltr" style="MARGIN-RIGHT: 0px">
    <p>
<br><br> &nbsp;<font face="Verdana">And you rated your top choice(s)
as:</font><br><br></p></blockquote>
<!--the following code writes the voters first, second and third choice
that was identified on ballot3.asp-->
<%
Set RsFirstChoice = Server.CreateObject ("ADODB.Recordset")
    strSQL1 = "SELECT * FROM vnpSchiefFirstChoice WHERE BallotID = "
& cint(maxnum)
    RsFirstChoice.Open strSQL1,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    %>
    <table align="center" border="0">
    <tr><td width="110"><strong>First Choice:</strong></td>
    <td width="200"><%Response.Write RsFirstChoice ("LastName")
& ", " & " &nbsp;<br>
    Response.Write RsFirstChoice ("FirstName")%>&nbsp;<br><%
    Response.Write RsFirstChoice ("MiddleName")%>&nbsp;</td>
    <td><%Response.Write RsFirstChoice
("Description")%></td></tr>
    <%
    RsFirstChoice.Close ()
%>
<%
Set RsSecondChoice = Server.CreateObject ("ADODB.Recordset")
    strSQL2 = "SELECT * FROM vnpSchiefSecondChoice WHERE BallotID = "
& cint(maxnum)
    RsSecondChoice.Open strSQL2,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
    %>

    <tr><td width="110"><strong>Second Choice:</strong></td>
    <td><%Response.Write RsSecondChoice ("LastName") & ", " &
" &nbsp;<br>
    Response.Write RsSecondChoice ("FirstName")%>&nbsp;<br><%

```

```

                Response.Write RsSecondChoice ("MiddleName")%>&nbsp;  </td>
                <td><%Response.Write RsSecondChoice
("Description")%></td></tr>
                <%
                RsSecondChoice.Close ()
%>
<%
Set RsThirdChoice = Server.CreateObject ("ADODB.Recordset")
    strSQL3 = "SELECT * FROM vnpSchiefThirdChoice WHERE BallotID = "
& cint(maxnum)
    RsThirdChoice.Open strSQL3,
Application("saward_ConnectionString"),adOpenDynamic ,adLockPessimistic
%>

    <tr><td width="110"><strong>Third Choice:</strong></td>
    <td><%Response.Write RsThirdChoice ("LastName") & ", " & "   "
Response.Write RsThirdChoice ("FirstName")%>&nbsp;  <%
Response.Write RsThirdChoice ("MiddleName")%>&nbsp;  </td>
    <td><%Response.Write RsThirdChoice ("Description")%></td></tr>
    <%
    RsThirdChoice.Close ()
%>
</table><br><br>
<td>

                &nbsp;  </td><br><br></tr>

    <tr>
        <td>
            &nbsp;  </td>
        <td>
            <table width="100%" border="0">
                <tr>
                    <td width="30%">
                        <div align="center"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">Revision Date:
<!--webbot bot="Timestamp" S-Type="EDITED" S-Format="%d %b %Y"
startspan -->02 Sep 2001<!--webbot bot="Timestamp" endspan i-
checksum="14928" -->

                        </font></div>
                    </td>
                    <td width="34%">
                        <div align="center"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">Contact
                        <a
href="mailto:samaglia@nps.navy.mil">Webmaster</a></font></div>
                    </td>
                    <td width="20%">
                        <div align="center"><a
href="http://intranet.nps.navy.mil"><font face="Verdana, Arial,
Helvetica, sans-serif" size="1">NPS
                        Intranet</font></a></div>
                    </td>
                    <td width="16%">
                        <div align="center"><a
href="https://itwarrior.nps.navy.mil/exchange/logon.asp"><font
face="Verdana, Arial, Helvetica, sans-serif" size="1">NPS
                        email</font></a></div>

```

```

                                </td>
                                </tr>
                        </table></td>
                </tr></table>
</body>
</html>
-->

```

F. FAIL.HTML

```

<!--
<%@ Language=VBScript %>
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;&nbsp;&nbsp;
                        <p><FONT face=Verdana></FONT>&nbsp;&nbsp;&nbsp;</p><FONT

```

face=Verdana>Only Students are authorized to enter the
Ballot Area.

If you are a student requiring access to the Ballot,
contact your

<FONT
face=Verdana>Administrator<FONT
face=Verdana>.
<p> </p></td></tr></table></td></tr></table>

</body>

</html>

-->

G. FAIL1.HTML

FAIL1

<!--

<%@ Language=VBScript %>

<html>

<head>

<title>Schieffelin Teaching Award</title>

<meta http-equiv="Content-Type" content="text/html;">

<meta http-equiv="Pragma" content="no-cache">

</head>

<body bgcolor="#ffffff">

<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">

<tr>

<td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>

<td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>

<td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>

</tr>

<tr>

<td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">

<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1"> Naval

Postgraduate School

 1 University Circle

 Monterey, CA 93943-5001

 (831) 656-2441/2

 DSN: 878-2441/2</p>

<p align="left"> </p>

<p align="left"> </p>

<p align="left"> </p>

</td>

<td valign="top" height="1" width="602">

<table width="583" border="0" align="center">

<tr>

<td width="575">


```


## H. LOGIN.ASP



```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
				---	---				

```



155


```

```

        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;
                        <p><FONT></FONT>&nbsp;</p>
<%
Dim strNPGS
    strNPGS = Request.ServerVariables ("AUTH_USER")
IF LEFT (UCASE (strNPGS),5) = "NPGS\" Then
    strNPGS = RIGHT (strNPGS,LEN(strNPGS)-5)
END IF

set objConn = server.CreateObject ("ADODB.Connection")
objConn.open Application("saward_ConnectionString")

set rs_login = server.CreateObject ("ADODB.recordset")
strsql="Exec snpGetLogin '"&strNPGS&"'"
rs_login.Open strsql,objConn,adOpenDynamic,adLockReadOnly

If rs_login.EOF or rs_login.RecordCount > 1 then
    %>
        <FONT face=Verdana>Your Windows Logon
information is either not
entered in the NPS database or you have more than one
record in the
system. Contact your </FONT>

                <A
href="mailto:samaglia@nps.navy.mil"><FONT
face=Verdana>System Administrator</FONT> </a><FONT
face=Verdana
size=4><FONT size=3>.</FONT>
<%
Else

    Session ("EmpID") = rs_login ("EmployeeID")
    Session ("Student") = rs_login ("IsStudent")
    Session ("Faculty") = rs_login ("IsFaculty")

        if rs_login ("IsFaculty") = "Y" Then
            Response.Redirect ("fail.html")
        else
        end if

        if rs_login ("IsStaff") = "Y" Then
            Response.Redirect ("fail.html")
        else
        end if

```

```

        if rs_login ("IsStudent") = "Y" Then
            Response.Redirect ("ballot.asp")
        else
        end if

        if rs_login ("IsCivilian") = "Y" Then
            Response.Redirect ("fail.html")
        else
        end if

        if rs_login ("IsContractor") = "Y" Then
            Response.redirect ("fail.html")
        else
        end if

```

```

end if
%></FONT></td></tr></table></td></tr></table>

```

```

</body>
</html>
-->

```

I. INDEX.HTML

```

<!--
<%@ Language=VBScript %>
<%
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Set RsHistory = Server.CreateObject ("ADODB.Recordset")
sqltext= "SELECT Rank, EmployeeID, FirstName, LastName, MiddleName, "
&_
"CalendarYear, EligibilityCode, NBallots, NumberSupportStatements,
FinalScore," &_
"PCScore, PCRank FROM vnpSchiefHistory;"
RsHistory.Open sqltext, objconn
%>

<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="../images/spacer.gif"
width=160 border=0></td>
        <td width="602" height="3"><IMG height=1 src="../images/spacer.gif"
width=600 border=0></td>
        <td width="3" height="3"><IMG height=1 src="../images/spacer.gif"
width=1 border=0></td>
    </tr>

```



```

        <TD><A
            href="/Saward/admin/input_form2.html"><STRONG><FONT
face=Verdana>Committee Scores
            </FONT>
            </STRONG></A> </TD>
        <TD><STRONG><A href="/Saward/admin/input_form8.html"><FONT
face=Verdana>by
            Employee ID </FONT>
            </A>
            </STRONG> </TD>
        <TD><STRONG><A
            href="/Saward/admin/input_form4.html"><FONT face=Verdana>by All
            Departments </FONT>
            </A></STRONG> </TD>
        <TD><STRONG><FONT
            face=Verdana>***</FONT><A
href="/Saward/admin/input_form10.html"><FONT
            face=Verdana>Detailed view of eligiblity </FONT>
            </A></STRONG></TD></TR>
    <TR>
        <TD><STRONG><A
            href="/Saward/admin/history.asp"><FONT face=Verdana>Chairman
            History </FONT> </A></STRONG> </TD>
        <TD><STRONG><FONT face=Verdana
            >*</FONT><A
            href="/Saward/admin/compute_eligibility.asp"><FONT
            face=Verdana>Compute Eligibility </FONT>
            </A></STRONG></TD>
        <TD><STRONG><A href="/Saward/admin/input_form11.html"><FONT
            face=Verdana>by&nbsp;Last Name </FONT>
            </A>
            </STRONG></TD>
        <TD><STRONG><A
            href="/Saward/ballot.asp"><FONT
            face=Verdana>Ballot</FONT></A></STRONG></TD></TR>
    <TR>
        <TD>
            <P><STRONG><A
                href="/Saward/admin/history1.asp"><FONT face=Verdana>Committee
                History </FONT>
                </A></STRONG> </P></TD>
        <TD><STRONG><FONT
            face=Verdana>*</FONT><A
href="http://pm01.nps.navy.mil/Saward/admin/compute_scores.asp"><FONT
face=Verdana>Compute Scores
            </FONT>
            </A></STRONG></TD>
        <TD><A
            href="/Saward/admin/input_form3.html"><STRONG><FONT
face=Verdana>Input
            History</FONT></STRONG></A><STRONG><FONT face=Verdana>
            </FONT></STRONG></TD>
        <TD><STRONG><A
            href="/Saward/admin/weights.asp"><FONT
            face=Verdana>Weights</FONT></A><FONT size=2>

```

```

</FONT>
    </STRONG> </TD></TR></TABLE></P><FONT
size=2><FONT size=2>
<P dir=ltr style="MARGIN-RIGHT: 0px" align=left>
<P dir=ltr style="MARGIN-RIGHT: 0px" align=left><FONT size=2><FONT
size=3>
</FONT></FONT></P>
<P></P>
<P><FONT face=Verdana size=3></FONT>&nbsp;</P>
<P><FONT face=Verdana size=3>*Running
the "Compute Eligibility" function will overwrite any changes you have
made to
the Eligibility List for the selected Year.&nbsp;&nbsp;&nbsp;Run the Compute
Eligibility function after January 1st of the Award Year.&nbsp;&nbsp;&nbsp;Then
view the "Simple" or "Detailed" eligibility views.&nbsp;&nbsp;&nbsp;Once
corrections to these views are determined, use the "Edit Eligibility by
<U>Department</U> or <U>All Departments</U>" to update the eligibility
list.&nbsp;&nbsp;&nbsp;The ballot is dynamically created based off of the
Eligibility Views.&nbsp;&nbsp;&nbsp;Once those views are correct, the ballot is
correct.&nbsp;&nbsp;&nbsp;</FONT></P>
<P><FONT size=3 face=Verdana>**The "Compute Scores" function my take up
to 30 seconds to run.&nbsp;&nbsp;&nbsp;</FONT><FONT size=2></P>
<P dir=ltr style="MARGIN-RIGHT: 0px" align=left><FONT size=3
face=Verdana>***Because of the way DRMI and the Aviation Safety School
track classes, the majority of their professors
do not show in this view.</FONT></P></FONT></FONT></FONT>
</body>
</html>
-->

```

J. INPUT_FORM1.HTML

```

<!--
<%@ Language=VBScript %>
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">

```

```

<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT></b></a></p>
<p align="left"><FONT face=Verdana></FONT>&nbsp;&nbsp;&nbsp;</p>
</td>
<td valign="top" height="1" width="602">
<table width="583" border="0" align="center">
<tr>
<td width="575">
<div align="center"><IMG height=86
src="../../../images/AwardHdr.gif" width=311></div>
</td>
</tr>
<tr>
<td width="575">
&nbsp;&nbsp;&nbsp;
<p>&nbsp;&nbsp;&nbsp;</p>

<form action="rank1.asp" id="FORM1" method="post" name="FORM1">
<P style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 0px"
align=left><B>&nbsp;&nbsp;&nbsp;</B></P>
<P style="MARGIN-TOP: 0px; MARGIN-BOTTOM: 0px"
align=left><B></B>&nbsp;&nbsp;&nbsp;</P>
<P><FONT face=Verdana>Enter
Year:</FONT> &nbsp;&nbsp;&nbsp;<INPUT id=txtYear name=txtYear>
<input type="submit" value="Submit" id="SUBMIT1"></P></tr>
</table>
<P></P></FORM></tr></table>
</body>
</HTML>
-->

```

K. INPUT_FORM2.HTML

```

<!--
<%@ Language=VBScript %>

<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
<tr>

```

```

        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
        <p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;<b>Naval
        Postgraduate School<br>
        &nbsp;&nbsp;1 University Circle<br>
        &nbsp;&nbsp;Monterey, CA 93943-5001<br>
        &nbsp;&nbsp;(831) 656-2441/2<br>
        &nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
        <p align="left">&nbsp;&nbsp;</p>
        <p align="left">&nbsp;&nbsp;&nbsp;&nbsp;</p>
        <p align="left">&nbsp;&nbsp;&nbsp;&nbsp;< A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></a></p>
        <p align="left"><FONT face=Verdana></FONT>&nbsp;&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;&nbsp;&nbsp;&nbsp;
                        <p>&nbsp;&nbsp;</p>
                        <form action="rank2.asp" id="FORM1" method="post" name="FORM1">
                        <P>&nbsp;&nbsp;</P>
                        <P><FONT face=Verdana>Enter
                            Year:</FONT> &nbsp;&nbsp;<INPUT id=txt1 name=txtYear>
                        <input type="submit" value="Submit" id="SUBMIT1">
                        </P></form></tr></table></tr></table>
            </body>
        </table>
    </HTML>
-->

```

L. INPUT_FORM3.HTML

```

<!--
<%@ Language=VBScript %>
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">

```



```

        </P>
</form></td></tr></table></td></tr></table>
</body>
</table>
</html>
-->

```

M. INPUT_FORM4.HTML

```

<!--
<%@ Language=VBScript %>

<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT></b></a></p>
<p align="left"><FONT face=Verdana></FONT>&nbsp;&nbsp;&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="../images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;&nbsp;&nbsp;
                        <p>&nbsp;&nbsp;&nbsp;</p>

```

```

<form action="data_entry_eligibility_all.asp" id="FORM1" method="post"
name="FORM1">
<p>&nbsp;</p>
<p><FONT face=Verdana>Enter
        Year:</FONT> &nbsp;<input id="text1" name="txtCalYear">

<input type="submit" value="Submit" id="SUBMIT1">
</p></form></tr></table></tr></table>

</body>
</table>
</html>
-->

```

N. INPUT_FORM5.HTML

```

<!--
<%@ Language=VBScript %>

<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;<1 University Circle<br>
&nbsp;<Monterey, CA 93943-5001<br>
&nbsp;<(831) 656-2441/2<br>
&nbsp;<DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;< A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></a></p>
<p align="left"><FONT face=Verdana></FONT>&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">

```


[illegible]

Aeronautics & Astronautics

Analysis

Assistant Director of Programs

Aviation Safety

Base Communications Office

234F

[illegible]

Center

Dean of Students

DAPS

Defence Automated Printing Service

Defense Health Management Institute

DMDC

Defense Manpower Data Center

Dental

23A

Electrical and Computer Engineering

Engineering Division

Environmental Division

Equal Opportunity Officer

Facilities and Redevelopment

Facilities Management

Family Housing Support

Family Services Center

FISC-SD Contracting

FISC-SD Monterey Detachment

Flag Secretary

FNMOC

Fleet Numerical and Oceanography

Food & Beverage

GSBPP

Graduate School of Business and Public Policy

032B

[illegible]

[illegible]

Leisure Services

Library

Material Division

MA

Mathematics

ME

MR

Meteorology

35

METOC

Reserves

Military Reserve Units

MOVES

Modeling, Virtual Environments & Simulation Group

2251

MWR Admin

225

MWR Department

NS

National Security Affairs

National Security Affairs/Intelligence Curricular Office

Naval Engineering Curricular Office

NMAU

Naval Medical Admin Unit

NRL

NSGD

[illegible]

Operations Research

214

Payroll & Labor

PERSONNEL SECURITY

232

PERSONNEL SUPPORT

PSD

PHYSICS

2313

Seabee Division

2311

Seabees

005

Security Manager

Shops Division

SP

SPEAR

SOLIC

Special Operations/Low Intensity Conflict

52

Special Projects

005

Special Security Officer (SSO)

006

Staff JAG

00

Superintendent

001

Superintendent's Office

SI

System Engineering and Integration

Systems Management Curricular Office

Systems Support

Training Command

Undersea Warfare

37


```

<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><font
face="Verdana">Admin Home</font> </b></a></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">
            <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
          </td>
        </tr>
        <tr>
          <td width="575">
            &nbsp;&nbsp;&nbsp;
            <p>&nbsp;&nbsp;&nbsp;</p>
          </td>
        </tr>
      </table>

<form action="eligible_Simple.asp" id="frmVote" method="post"
name="Form1">
<label><font face="Verdana">Enter&nbsp;&nbsp;&nbsp;Year:</font>

</label>
<input id="text1" name="cyear">
<input type="submit" value="Submit" id="SUBMIT">
  <p></p>
</form>

<p>&nbsp;&nbsp;&nbsp;</p>
  </td>

```

```

        </tr>
    </table>
</td>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

P. INPUT_FORM7.HTML

```

<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></A></p>
<p align="left"><FONT face=Verdana></FONT>&nbsp;&nbsp;&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
            </table>
        </td>
    </tr>

```

```

        &nbsp;
        <p>&nbsp;</p>

<form action="lname.asp" id="frmVote" method="post" name="Form1">
<label><FONT
        face=Verdana>Enter Last Name:</FONT>

</label>
<input id="text1" name="txtLName">
<input type="submit" value="Submit" id="SUBMIT">
        <p></p>
</form>
        </td>
    </tr>

    </table>
</td>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

Q. INPUT_FORM8.HTML

```

<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
    Postgraduate School<br>
    &nbsp;<b>1 University Circle<br>
    &nbsp;<b>Monterey, CA 93943-5001<br>
    &nbsp;<b>(831) 656-2441/2<br>
    &nbsp;<b>DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;&nbsp;</p>

```



```

<p align="left">&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></A></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
  </td>
  <td valign="top" height="1" width="602">
    <table width="583" border="0" align="center">
      <tr>
        <td width="575">
          <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
        </td>
      </tr>
      <tr>
        <td width="575">
          &nbsp;&nbsp;&
          <p>&nbsp;&nbsp;&</p>

<form action="empid.asp" id="frmVote" method="post" name="Form1">
<label><FONT
          face=Verdana>Enter Employee ID:</FONT>

</label>
<input id="text1" name="txtEmpID">
<input type="submit" value="Submit" id="SUBMIT">
      <p></p>
</form>
    </td>
  </tr>

  </table>
</td>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

R. INPUT_FORM10.HTML

```

<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>

```

```

        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
        <p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
        Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
        <p align="left">&nbsp;&nbsp;&nbsp;</p>
        <p align="left">&nbsp;&nbsp;&nbsp;</p>
        <p align="left">&nbsp;&nbsp;&nbsp;<A href="index.html"><b><font
face="Verdana">Admin Home</font> </b></a></p>
        <p align="left">&nbsp;&nbsp;&nbsp;</p>
        </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="../../../images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;&nbsp;&nbsp;
                        <p>&nbsp;&nbsp;&nbsp;</p>
                    </td>
                </tr>
            </table>
            <form action="eligible_detailed.asp" id="frmVote" method="post"
name="Form1">
            <label><font face="Verdana">Enter Year:</font>

            </label>
            <input id="text1" name="cYear">
            <input type="submit" value="Submit" id="SUBMIT">
            <p></p>
            </form>
        </td>
    </tr>
</table></tr></table>
</body>
</html>
-->

```

S. INPUT_FORM11.HTML

```

<!--
<%@ Language=VBScript %>

<html>

```

```

<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT></b></A></p>
<p align="left">&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">
            <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
          </td>
        </tr>
        <tr>
          <td width="575">
            &nbsp;&nbsp;&nbsp;&nbsp;<p>&nbsp;&nbsp;</p>
<form action="data_entry_eligibility_lname.asp" id="FORM1"
method="post" name="FORM1">
<p>&nbsp;&nbsp;</p>
<p><FONT face=Verdana>Enter
      Year:</FONT> &nbsp;&nbsp;<input id="text1" name="txtCalYear"></p>
<p><FONT face=Verdana>Enter Last
      Name:</FONT> &nbsp;&nbsp;&nbsp;&nbsp;<input id="text2"
name="txtLastName" style="WIDTH: 91px; HEIGHT: 22px" size="13"></p>

<input type="submit" value="Submit" id="SUBMIT1">
</form></td></tr>
</table>

```

```
<p>&nbsp;</p></td></tr></table>
```

```
</body>
```

```
</html>
```

```
-->
```

T. COMPUTE_ELIGIBILITY.ASP

```
<!--
```

```
<%@ Language=VBScript %>
```

```
<%
```

```
Dim objConn
```

```
Set objConn = Server.CreateObject("ADODB.Connection")
```

```
objConn.Open Application("saward_ConnectionString")
```

```
Dim intDate
```

```
'Subtracts 1 year from the current year to produce Eligible Faculty for  
the correct year
```

```
intDate = Year(Date)-1
```

```
Dim strSQL
```

```
strSQL = "Exec snpDetermineScheffelinEligibility"& intDate
```

```
Dim RsScores
```

```
Set RsScores = Server.CreateObject ("ADODB.Recordset")
```

```
RsScores.Open strSQL, objconn
```

```
%>
```

```
<%Response.Redirect ("index.html")%>
```

```
<HTML>
```

```
<HEAD>
```

```
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
```

```
</HEAD>
```

```
<BODY>
```

```
<P>&nbsp;</P>
```

```
</BODY>
```

```
</HTML>
```

```
-->
```

U. COMPUTE_SCORES.ASP

```
<!--
```

```
<%@ Language=VBScript %>
```

```
<%
```

```
Dim objConn
```

```
Set objConn = Server.CreateObject("ADODB.Connection")
```

```
objConn.Open Application("saward_ConnectionString")
```

```
Dim intDate
```

```
'Subtracts 1 year from the current year to produce Eligible Faculty for  
the correct year
```

```
intDate = Year(Date)-1
```

```
Dim strSQL
```

```
strSQL = "Exec snpCallScoring "& intDate
```

```

Dim RsScores
Set RsScores = Server.CreateObject ("ADODB.Recordset")
RsScores.Open strSQL, objconn
%>
<%Response.Redirect ("index.html")%>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<P>&nbsp;</P>
</BODY>
</HTML>
-->

```

V. DATA_ENTRY_ELIGIBILITY_ALL.ASP

```

<!--
<%@ Language=VBScript %>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<title>Schieffelin Teaching Award</title>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
Postgraduate School<br>
&nbsp;<b>1 University Circle<br>

```

```

&nbsp;Monterey, CA 93943-5001<br>
&nbsp;(831) 656-2441/2<br>
&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></a></p>
<p align="left">&nbsp;</p>
</td>
<td valign="top" height="1" width="602">
<table width="583" border="0" align="center">
<tr>
<td width="575">
<div align="center"><IMG height=86
src="../../../images/AwardHdr.gif" width=311></div>
</td>
</tr>
<tr>
<td width="575">
&nbsp;
<p>&nbsp;</p>

<%
Dim intCalYear
intCalYear = Request.Form ("txtCalYear")

%>
<P>
<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsEligibility
classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q, TCDBObject_Unmatched=\qSQL\sStatement\q, TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpEmployee.LastName, \sdbo.tnpEmployee
e.FirstName, \sdbo.tnpEmployee.MiddleName, \sdbo.tnpSchieffelinHistory.Ca
lendarYear, \sdbo.tnpSchieffelinHistory.EmployeeID, \sdbo.tnpSchieffelinH
istory.IsEligible, \sdbo.tnpDepartment.Description\sFROM\sdbo.tnpSchieff
elinHistory\sINNER\sJOIN\sdbo.tnpEmployee\SON\sdbo.tnpSchieffelinHistor
y.EmployeeID\s=\sdbo.tnpEmployee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepart
ment_Employee\SON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Emp
loyee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartment\SON\sdbo.tnpDepartment
_Employee.DepartmentID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s (dbo.
tnpSchieffelinHistory.CalendarYear\s=\s?)\sORDER\sBY\sdbo.tnpDepartment
.Description, \sdbo.tnpEmployee.LastName\q, TCControlID_Unmatched=\qRsEli
gibility\q, TCPPConn=\qsaward\q, TCPPDBObject=\qTables\q, TCPPDBObjectName
=\qtnpSchieffelinHistory\s (dbo) \q, RCDBObject=\qRCSQLStatement\q, TCSQLSt
atement_Unmatched=\qSELECT\sdbo.tnpEmployee.LastName, \sdbo.tnpEmployee.
FirstName, \sdbo.tnpEmployee.MiddleName, \sdbo.tnpSchieffelinHistory.Cale
ndarYear, \sdbo.tnpSchieffelinHistory.EmployeeID, \sdbo.tnpSchieffelinHis
tory.IsEligible, \sdbo.tnpDepartment.Description\sFROM\sdbo.tnpSchieff
elinHistory\sINNER\sJOIN\sdbo.tnpEmployee\SON\sdbo.tnpSchieffelinHistory.
EmployeeID\s=\sdbo.tnpEmployee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartme
nt_Employee\SON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Emplo
yee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartment\SON\sdbo.tnpDepartment_E
mployee.DepartmentID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s (dbo.tn

```

```
pSchieffelinHistory.CalendarYear\s=s?)\sORDER\sBY\sdbo.tnpDepartment.D
escription,\sdbo.tnpEmployee.LastName\q,TCCursorType=\q3\s-
\sStatic\q,TCCursorLocation=\q3\s-\sUse\sclient-
side\sursors\q,TCLockType=\q3\s-
\sOptimistic\q,TCCacheSize_Unmatched=\q10\q,TCCommTimeout_Unmatched=\q1
0\q,CCPrepared=0,CCallRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSynta
x_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q,TCTargetPlatform=\qInherit\sfrom\spage\q,RCCache=\qRCBook
Page\q,CCOpen=0,GCPParameters=(Rows=1,Row1=(CType_Unmatched=\q?\q,CParNa
me_Unmatched=\qParam1\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched
=\q4\q,CReq=1,CValue_Unmatched=\qintCalYear\q))"><PARAM
NAME="LocalPath" VALUE="../">
```

```
</OBJECT>
```

```
-->
```

```
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
```

```
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
```

```
function _setParametersRsEligibility()
```

```
{
```

```
    RsEligibility.setParameter(0,intCalYear);
```

```
}
```

```
function _initRsEligibility()
```

```
{
```

```
    RsEligibility.advise(RS_ONBEFOREOPEN,
```

```
_setParametersRsEligibility);
```

```
    var DBConn = Server.CreateObject('ADODB.Connection');
```

```
    DBConn.ConnectionTimeout =
```

```
Application('saward_ConnectionTimeout');
```

```
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
```

```
    DBConn.CursorLocation = Application('saward_CursorLocation');
```

```
    DBConn.Open(Application('saward_ConnectionString'),
```

```
Application('saward_RuntimeUserName'),
```

```
Application('saward_RuntimePassword'));
```

```
    var cmdTmp = Server.CreateObject('ADODB.Command');
```

```
    var rsTmp = Server.CreateObject('ADODB.Recordset');
```

```
    cmdTmp.ActiveConnection = DBConn;
```

```
    rsTmp.Source = cmdTmp;
```

```
    cmdTmp.CommandType = 1;
```

```
    cmdTmp.CommandTimeout = 10;
```

```
    cmdTmp.CommandText = 'SELECT dbo.tnpEmployee.LastName,
```

```
dbo.tnpEmployee.FirstName, dbo.tnpEmployee.MiddleName,
```

```
dbo.tnpSchieffelinHistory.CalendarYear,
```

```
dbo.tnpSchieffelinHistory.EmployeeID,
```

```
dbo.tnpSchieffelinHistory.IsEligible, dbo.tnpDepartment.Description
```

```
FROM dbo.tnpSchieffelinHistory INNER JOIN dbo.tnpEmployee ON
```

```
dbo.tnpSchieffelinHistory.EmployeeID = dbo.tnpEmployee.EmployeeID INNER
```

```
JOIN dbo.tnpDepartment_Employee ON dbo.tnpEmployee.EmployeeID =
```

```
dbo.tnpDepartment_Employee.EmployeeID INNER JOIN dbo.tnpDepartment ON
```

```
dbo.tnpDepartment_Employee.DepartmentID =
```

```
dbo.tnpDepartment.DepartmentID WHERE
```

```
(dbo.tnpSchieffelinHistory.CalendarYear = ?) ORDER BY
```

```
dbo.tnpDepartment.Description, dbo.tnpEmployee.LastName';
```

```
    rsTmp.CacheSize = 10;
```

```
    rsTmp.CursorType = 3;
```

```
    rsTmp.CursorLocation = 3;
```

```
    rsTmp.LockType = 3;
```

```
    RsEligibility.setRecordSource(rsTmp);
```



```

        txtLLName.setDataField('LastName');
        txtLLName.disabled = true;
        txtLLName.setMaxLength(20);
        txtLLName.setColumnCount(20);
    }
    function _txtLLName_ctor()
    {
        CreateTextbox('txtLLName', _inittxtLLName, null);
    }
</script>
<% txtLLName.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
<P><FONT face=Verdana>First

Name:</FONT>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtFName
        classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0><PARAM
NAME="_ExtentX" VALUE="3175"><PARAM NAME="_ExtentY" VALUE="503"><PARAM
NAME="id" VALUE="txtFName"><PARAM NAME="ControlType" VALUE="0"><PARAM
NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsEligibility"><PARAM NAME="DataField" VALUE="FirstName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="20"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="..">
        </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtFName()
{
    txtFName.setStyle(TXT_TEXTBOX);
    txtFName.setDataSource(RsEligibility);
    txtFName.setDataField('FirstName');
    txtFName.disabled = true;
    txtFName.setMaxLength(20);
    txtFName.setColumnCount(20);
}
function _txtFName_ctor()
{
    CreateTextbox('txtFName', _inittxtFName, null);
}
</script>
<% txtFName.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
<P><FONT face=Verdana>Middle

Name:</FONT>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtMName
        classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0><PARAM
NAME="_ExtentX" VALUE="3175"><PARAM NAME="_ExtentY" VALUE="503"><PARAM

```



```

NAME="MovePrev" VALUE="-1"><PARAM NAME="PrevImage" VALUE="0"><PARAM
NAME="NextCaption" VALUE=" > "><PARAM NAME="MoveNext" VALUE="-
1"><PARAM NAME="NextImage" VALUE="0"><PARAM NAME="LastCaption" VALUE="
>| "><PARAM NAME="MoveLast" VALUE="-1"><PARAM NAME="LastImage"
VALUE="0"><PARAM NAME="Alignment" VALUE="1"><PARAM NAME="LocalPath"
VALUE="../">

</OBJECT>
-->
<!--#INCLUDE FILE="../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/RNavBar.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initRecordsetNavbar1()
{
    RecordsetNavbar1.setAlignment(1);
    RecordsetNavbar1.setButtonStyles(170);
    RecordsetNavbar1.setDataSource(RsEligibility);
    RecordsetNavbar1.getButton(0).value = ' |< ';
    RecordsetNavbar1.getButton(1).value = ' < ';
    RecordsetNavbar1.getButton(2).value = ' > ';
    RecordsetNavbar1.getButton(3).value = ' >| ';
}
function _RecordsetNavbar1_ctor()
{
    CreateRecordsetNavbar('RecordsetNavbar1', _initRecordsetNavbar1,
null);
}
</script>
<% RecordsetNavbar1.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P></tr></table></tr></table>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

W. DATA_ENTRY_ELIGIBILITY_DEPT.ASP

```

<!--
<%@ Language=VBScript %>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>

```

```

<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></a></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">
            <div align="center"><IMG height=86
src="../images/AwardHdr.gif" width=311></div>
          </td>
        </tr>
        <tr>
          <td width="575">
            &nbsp;&nbsp;&nbsp;<p>&nbsp;&nbsp;&nbsp;</p>
          </td>
        </tr>
      </table>
    </td>
  </tr>
</table>
Dim intCalYear
intCalYear = Request.Form ("txtCalYear")
Dim strDept
strDept = Request.Form ("txtDept")
%>
<P>
      <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsEligibility style="LEFT: 0px; TOP: 0px"
      classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM

```

```

NAME="State"
VALUE=" (TConn=\qsaward\q, TCDBObject_Unmatched=\qSQL\sStatement\q, TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpEmployee.LastName, \sdbo.tnpEmpleyee
.FirstName, \sdbo.tnpEmployee.MiddleName, \sdbo.tnpSchieffelinHistory.Ca
lendarYear, \sdbo.tnpSchieffelinHistory.EmployeeID, \sdbo.tnpSchieffelinH
istory.IsEligible, \sdbo.tnpDepartment.Description\sFROM\sdbo.tnpSchieff
elinHistory\sINNER\sJOIN\sdbo.tnpEmployee\sON\sdbo.tnpSchieffelinHistor
y.EmployeeID\s=\sdbo.tnpEmployee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepart
ment_Employee\sON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Emp
loyee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartment\sON\sdbo.tnpDepartment
_Employee.DepartmentID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s (dbo.
tnpSchieffelinHistory.CalendarYear\s=\s?) \sAND\s (dbo.tnpDepartment_Empl
oyee.IsPrimaryDepartment\s=\s'Y') \sAND\s (dbo.tnpDepartment.Department\s
=\s?) \sORDER\sBY\sdbo.tnpDepartment.Description, \sdbo.tnpEmployee.LastN
ame\q, TCControlID_Unmatched=\qRsEligibility\q, TCPPConn=\qsaward\q, TCPPD
BObject=\qTables\q, TCPPDBObjectName=\qtnpSchieffelinHistory\s (dbo) \q, RC
DBObject=\qRCSQLStatement\q, TCSQLStatement_Unmatched=\qSELECT\sdbo.tnpE
mployee.LastName, \sdbo.tnpEmployee.FirstName, \sdbo.tnpEmployee.MiddleNa
me, \sdbo.tnpSchieffelinHistory.CalendarYear, \sdbo.tnpSchieffelinHistory
.EmployeeID, \sdbo.tnpSchieffelinHistory.IsEligible, \sdbo.tnpDepartment.
Description\sFROM\sdbo.tnpSchieffelinHistory\sINNER\sJOIN\sdbo.tnpEmple
yee\sON\sdbo.tnpSchieffelinHistory.EmployeeID\s=\sdbo.tnpEmployee.Emple
yeeID\sINNER\sJOIN\sdbo.tnpDepartment_Employee\sON\sdbo.tnpEmployee.Empl
oyeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\sINNER\sJOIN\sdbo.tnp
Department\sON\sdbo.tnpDepartment_Employee.DepartmentID\s=\sdbo.tnpDepa
rtment.DepartmentID\sWHERE\s (dbo.tnpSchieffelinHistory.CalendarYear\s=\
s?) \sAND\s (dbo.tnpDepartment_Employee.IsPrimaryDepartment\s=\s'Y') \sAND
\s (dbo.tnpDepartment.Department\s=\s?) \sORDER\sBY\sdbo.tnpDepartment.De
scription, \sdbo.tnpEmployee.LastName\q, TCCursorType=\q3\s-
\sStatic\q, TCCursorLocation=\q3\s-\sUse\sclient-
side\sursors\q, TCLockType=\q3\s-
\sOptimistic\q, TCCacheSize_Unmatched=\q10\q, TCCommTimeout_Unmatched=\q1
0\q, CCPPrepared=0, CCallRecords=1, TCNRecords_Unmatched=\q10\q, TCODBCSynta
x_Unmatched=\q\q, TCHTargetPlatform=\q\q, TCHTargetBrowser_Unmatched=\qSe
rver\s (ASP) \q, TCTargetPlatform=\qInherit\sfrom\spage\q, RCCache=\qRCBook
Page\q, CCOpen=0, GCPParameters= (Rows=2, Row1= (CType_Unmatched=\q?\q, CParNa
me_Unmatched=\qParam1\q, CDataType_Unmatched=\qInteger\q, CSize_Unmatched
=\q4\q, CReq=1, CValue_Unmatched=\qintCalYear\q), Row2= (CType_Unmatched=\q
?\q, CParName_Unmatched=\qParam2\q, CDataType_Unmatched=\qVarChar\q, CSize
_Unmatched=\q10\q, CReq=1, CValue_Unmatched=\qstrDept\q)) ) "><PARAM
NAME="LocalPath" VALUE="../">

```

</OBJECT>

```

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsEligibility()
{
    RsEligibility.setParameter(0,intCalYear);
    RsEligibility.setParameter(1,strDept);
}
function _initRsEligibility()
{
    RsEligibility.advise(RS_ONBEFOREOPEN,
_setParametersRsEligibility);
    var DBConn = Server.CreateObject('ADODB.Connection');

```

```

        DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
        DBConn.CommandTimeout = Application('saward_CommandTimeout');
        DBConn.CursorLocation = Application('saward_CursorLocation');
        DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
        var cmdTmp = Server.CreateObject('ADODB.Command');
        var rsTmp = Server.CreateObject('ADODB.Recordset');
        cmdTmp.ActiveConnection = DBConn;
        rsTmp.Source = cmdTmp;
        cmdTmp.CommandType = 1;
        cmdTmp.CommandTimeout = 10;
        cmdTmp.CommandText = 'SELECT dbo.tnpEmployee.LastName,
dbo.tnpEmployee.FirstName, dbo.tnpEmployee.MiddleName,
dbo.tnpSchieffelinHistory.CalendarYear,
dbo.tnpSchieffelinHistory.EmployeeID,
dbo.tnpSchieffelinHistory.IsEligible, dbo.tnpDepartment.Description
FROM dbo.tnpSchieffelinHistory INNER JOIN dbo.tnpEmployee ON
dbo.tnpSchieffelinHistory.EmployeeID = dbo.tnpEmployee.EmployeeID INNER
JOIN dbo.tnpDepartment_Employee ON dbo.tnpEmployee.EmployeeID =
dbo.tnpDepartment_Employee.EmployeeID INNER JOIN dbo.tnpDepartment ON
dbo.tnpDepartment_Employee.DepartmentID =
dbo.tnpDepartment.DepartmentID WHERE
(dbo.tnpSchieffelinHistory.CalendarYear = ?) AND
(dbo.tnpDepartment_Employee.IsPrimaryDepartment = \'Y\') AND
(dbo.tnpDepartment.Department = ?) ORDER BY
dbo.tnpDepartment.Description, dbo.tnpEmployee.LastName';
        rsTmp.CacheSize = 10;
        rsTmp.CursorType = 3;
        rsTmp.CursorLocation = 3;
        rsTmp.LockType = 3;
        RsEligibility.setRecordSource(rsTmp);
        if (thisPage.getState('pb_RsEligibility') != null)

            RsEligibility.setBookmark(thisPage.getState('pb_RsEligibility'));
    }
function _RsEligibility_ctor()
{
    CreateRecordset('RsEligibility', _initRsEligibility, null);
}
function _RsEligibility_dtor()
{
    RsEligibility._preserveState();
    thisPage.setState('pb_RsEligibility',
RsEligibility.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" ends-->
<%RsEligibility.open %>
</P>
<P>&nbsp;</P>
<P><FONT

```



```

        classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0 width=18
        height=19><PARAM NAME="_ExtentX" VALUE="476"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="txtIsElig"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsEligibility"><PARAM NAME="DataField"
VALUE="IsEligible"><PARAM NAME="Enabled" VALUE="-1"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="6"><PARAM
NAME="DisplayWidth" VALUE="3"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="../">
        </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtIsElig()
{
    txtIsElig.setStyle(TXT_TEXTBOX);
    txtIsElig.setDataSource(RsEligibility);
    txtIsElig.setDataField('IsEligible');
    txtIsElig.setMaxLength(6);
    txtIsElig.setColumnCount(3);
}
function _txtIsElig_ctor()
{
    CreateTextbox('txtIsElig', _inittxtIsElig, null);
}
</script>
<% txtIsElig.display %>

<!--METADATA TYPE="DesignerControl" endspan-->
</FONT>
</P>
<P>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RecordsetNavbar1
        classid=clsid:58F3D268-FEDF-11D0-9C7F-0060081840F3><PARAM
NAME="_ExtentX" VALUE="4075"><PARAM NAME="_ExtentY" VALUE="714"><PARAM
NAME="id" VALUE="RecordsetNavbar1"><PARAM NAME="Platform"
VALUE="0"><PARAM NAME="DataSource" VALUE="RsEligibility"><PARAM
NAME="UpdateOnMove" VALUE="-1"><PARAM NAME="FirstCaption" VALUE=" |<
"><PARAM NAME="MoveFirst" VALUE="-1"><PARAM NAME="FirstImage"
VALUE="0"><PARAM NAME="PrevCaption" VALUE=" < "><PARAM
NAME="MovePrev" VALUE="-1"><PARAM NAME="PrevImage" VALUE="0"><PARAM
NAME="NextCaption" VALUE=" > "><PARAM NAME="MoveNext" VALUE="-
1"><PARAM NAME="NextImage" VALUE="0"><PARAM NAME="LastCaption" VALUE="
>| "><PARAM NAME="MoveLast" VALUE="-1"><PARAM NAME="LastImage"
VALUE="0"><PARAM NAME="Alignment" VALUE="1"><PARAM NAME="LocalPath"
VALUE="../">
        </OBJECT>
-->
<!--#INCLUDE FILE="../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/RNavBar.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initRecordsetNavbar1()
{
    RecordsetNavbar1.setAlignment(1);
    RecordsetNavbar1.setButtonStyles(170);
    RecordsetNavbar1.setDataSource(RsEligibility);

```

```

        RecordsetNavbar1.getButton(0).value = ' |< ';
        RecordsetNavbar1.getButton(1).value = ' < ';
        RecordsetNavbar1.getButton(2).value = ' > ';
        RecordsetNavbar1.getButton(3).value = ' >| ';
    }
    function _RecordsetNavbar1_ctor()
    {
        CreateRecordsetNavbar('RecordsetNavbar1', _initRecordsetNavbar1,
null);
    }
</script>
<% RecordsetNavbar1.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
<P><A href="/Saward/admin/input_form5.html" ><FONT face=Verdana>Switch
        Departments</FONT>
</A>
</P>
<P>
</P></tr></table></tr></table>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

X. DATA_ENTRY_ELIGIBILITY_LNAME.ASP

```

<!--
<%@ Language=VBScript %>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body>
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>

```

```

        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#ffffff"
width="192">
        <p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
        Postgraduate School<br>
        &nbsp;<1 University Circle<br>
        &nbsp;<Monterey, CA 93943-5001<br>
        &nbsp;<(831) 656-2441/2<br>
        &nbsp;<DSN: 878-2441/2</b></font></p>
        <p align="left">&nbsp;</p>
        <p align="left">&nbsp;</p>
        <p align="left">&nbsp;< A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></a></p>
        <p align="left">&nbsp;</p>
    </td>
        <td valign="top" height="1" width="602">
            <table width="583" border="0" align="center">
                <tr>
                    <td width="575">
                        <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
                    </td>
                </tr>
                <tr>
                    <td width="575">
                        &nbsp;<
                        <p>&nbsp;</p>
    <%
Dim intCalYear
intCalYear = Request.Form ("txtCalYear")
Dim strLastName
strLastName = Request.Form ("txtLastName")
%>
<P>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsEligibility style="LEFT: 0px; TOP: 0px"
        classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q,TCDBObject_Unmatched=\qSQL\sStatement\q,TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpEmployee.LastName,\sdbo.tnpEmployee
e.FirstName,\sdbo.tnpEmployee.MiddleName,\sdbo.tnpSchieffelinHistory.Ca
lendarYear,\sdbo.tnpSchieffelinHistory.EmployeeID,\sdbo.tnpSchieffelinH
istory.IsEligible,\sdbo.tnpDepartment.Description,\sdbo.tnpDepartment_E
mployee.IsPrimaryDepartment\sFROM\sdbo.tnpSchieffelinHistory\sINNER\sJO
IN\sdbo.tnpEmployee\sON\sdbo.tnpSchieffelinHistory.EmployeeID\s=\sdbo.t
npEmployee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartment_Employee\sON\sdbo
.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\sINNE

```

```

R\sJOIN\sdbo.tnpDepartment\sON\sdbo.tnpDepartment_Employee.DepartmentID
\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s(dbo.tnpSchieffelinHistory.
CalendarYear\s=\s?)\sAND\s(dbo.tnpEmployee.LastName\s=\s?)\sAND\s(dbo.t
npDepartment_Employee.IsPrimaryDepartment\s=\s'y')\sORDER\sBY\sdbo.tnpD
epartment.Description,\sdbo.tnpEmployee.LastName\q,TCControlID_Unmatche
d=\qRsEligibility\q,TCPConn=\qsaward\q,TCPDBObject=\qTables\q,TCPDBO
bjectName=\qtnpSchieffelinHistory\s(dbo)\q,RCDBObject=\qRCSQLStatement\
q,TCSQLStatement_Unmatched=\qSELECT\sdbo.tnpEmployee.LastName,\sdbo.tnp
Employee.FirstName,\sdbo.tnpEmployee.MiddleName,\sdbo.tnpSchieffelinHis
tory.CalendarYear,\sdbo.tnpSchieffelinHistory.EmployeeID,\sdbo.tnpSchie
ffelinHistory.IsEligible,\sdbo.tnpDepartment.Description,\sdbo.tnpDepar
tment_Employee.IsPrimaryDepartment\sFROM\sdbo.tnpSchieffelinHistory\sIN
NER\sJOIN\sdbo.tnpEmployee\sON\sdbo.tnpSchieffelinHistory.EmployeeID\s=
\sdbo.tnpEmployee.EmployeeID\sINNER\sJOIN\sdbo.tnpDepartment_Employee\s
ON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeI
D\sINNER\sJOIN\sdbo.tnpDepartment\sON\sdbo.tnpDepartment_Employee.Depar
tmentID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s(dbo.tnpSchieffelinH
istory.CalendarYear\s=\s?)\sAND\s(dbo.tnpEmployee.LastName\s=\s?)\sAND\
s(dbo.tnpDepartment_Employee.IsPrimaryDepartment\s=\s'y')\sORDER\sBY\sdb
o.tnpDepartment.Description,\sdbo.tnpEmployee.LastName\q,TCCursorType=
\q3\s-\sStatic\q,TCCursorLocation=\q3\s-\sUse\sclient-
side\sursors\q,TCLockType=\q3\s-
\sOptimistic\q,TCCacheSize_Unmatched=\q10\q,TCCommTimeout_Unmatched=\q1
0\q,CCPrepared=0,CCallRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSynta
x_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q,TCTargetPlatform=\qInherit\sfrom\spage\q,RCCache=\qRCBook
Page\q,CCOpen=0,GCPParameters=(Rows=2,Row1=(CType_Unmatched=\q?\q,CParNa
me_Unmatched=\qParam1\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched
=\q4\q,CReq=1,CValue_Unmatched=\qintCalYear\q),Row2=(CType_Unmatched=\q
?\q,CParName_Unmatched=\qParam2\q,CDataType_Unmatched=\qVarChar\q,CSize
_Unmatched=\q30\q,CReq=1,CValue_Unmatched=\qstrLastName\q))"><PARAM
NAME="LocalPath" VALUE="../">

```

</OBJECT>

```

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsEligibility()
{
    RsEligibility.setParameter(0,intCalYear);
    RsEligibility.setParameter(1,strLastName);
}
function _initRsEligibility()
{
    RsEligibility.advise(RS_ONBEFOREOPEN,
_setParametersRsEligibility);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;

```


[illegible]


```

        height=19><PARAM NAME="_ExtentX" VALUE="476"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="txtIsElig"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsEligibility"><PARAM NAME="DataField"
VALUE="IsEligible"><PARAM NAME="Enabled" VALUE="-1"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="6"><PARAM
NAME="DisplayWidth" VALUE="3"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="..">

        </OBJECT>

-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtIsElig()
{
    txtIsElig.setStyle(TXT_TEXTBOX);
    txtIsElig.setDataSource(RsEligibility);
    txtIsElig.setDataField('IsEligible');
    txtIsElig.setMaxLength(6);
    txtIsElig.setColumnCount(3);
}
function _txtIsElig_ctor()
{
    CreateTextbox('txtIsElig', _inittxtIsElig, null);
}
</script>
<% txtIsElig.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</P>
<P><FONT face=Verdana>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RecordsetNavbar1 style="LEFT: 0px; TOP: 0px"
        classid=clsid:58F3D268-FEDF-11D0-9C7F-0060081840F3><PARAM
NAME="_ExtentX" VALUE="4075"><PARAM NAME="_ExtentY" VALUE="714"><PARAM
NAME="id" VALUE="RecordsetNavbar1"><PARAM NAME="Platform"
VALUE="0"><PARAM NAME="DataSource" VALUE="RsEligibility"><PARAM
NAME="UpdateOnMove" VALUE="-1"><PARAM NAME="FirstCaption" VALUE=" |<
"><PARAM NAME="MoveFirst" VALUE="-1"><PARAM NAME="FirstImage"
VALUE="0"><PARAM NAME="PrevCaption" VALUE=" < "><PARAM
NAME="MovePrev" VALUE="-1"><PARAM NAME="PrevImage" VALUE="0"><PARAM
NAME="NextCaption" VALUE=" > "><PARAM NAME="MoveNext" VALUE="-
1"><PARAM NAME="NextImage" VALUE="0"><PARAM NAME="LastCaption" VALUE="
>| "><PARAM NAME="MoveLast" VALUE="-1"><PARAM NAME="LastImage"
VALUE="0"><PARAM NAME="Alignment" VALUE="1"><PARAM NAME="LocalPath"
VALUE="..">

        </OBJECT>

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/RNavBar.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initRecordsetNavbar1()
{
    RecordsetNavbar1.setAlignment(1);
    RecordsetNavbar1.setButtonStyles(170);

```

```

        RecordsetNavbar1.setDataSource(RsEligibility);
        RecordsetNavbar1.getButton(0).value = ' |< ';
        RecordsetNavbar1.getButton(1).value = ' < ';
        RecordsetNavbar1.getButton(2).value = ' > ';
        RecordsetNavbar1.getButton(3).value = ' >| ';
    }
function _RecordsetNavbar1_ctor()
{
    CreateRecordsetNavbar('RecordsetNavbar1', _initRecordsetNavbar1,
null);
}
</script>
<% RecordsetNavbar1.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</P>
<P><A href="/Saward/admin/input_form11.html" ><FONT face=Verdana>Search
another Last
        Name</FONT>    </A><FONT face=Verdana> </FONT>
</P>
<P>
</P></tr></table></tr></table>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

Y. DATA_ENTRY_HISTORY.ASP

```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>
<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
</head>
<body bgcolor="#ffffff" >

```

```

<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;<1 University Circle<br>
&nbsp;<Monterey, CA 93943-5001<br>
&nbsp;<(831) 656-2441/2<br>
&nbsp;<DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;<&nbsp;</p>
<p align="left">&nbsp;<&nbsp;< A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></A></p>
<p align="left">&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">
        <tr>
          <td width="575">
            <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311></div>
          </td>
        </tr>
        <tr>
          <td width="575">
            &nbsp;<
            <p>&nbsp;</p>
          </td>
        </tr>
      </table>
    </td>
  </tr>
<%
Dim intCalYear
intCalYear = Request.Form ("txtYear")
Dim strLastName
strLastName = Request.Form ("txtLastName")

%>
<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsHistory style="LEFT: 0px; TOP: 0px"
  classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q,TCDBObject_Unmatched=\qSQL\sStatement\q,TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpSchieffelinHistory.CalendarYear,\s
dbo.tnpSchieffelinHistory.EmployeeID,\sdbo.tnpSchieffelinHistory.Eligib
ilityCode,\sdbo.tnpEmployee.LastName,\sdbo.tnpEmployee.FirstName,\sdbo.
tnpEmployee.MiddleName\sFROM\sdbo.tnpEmployee\sINNER\sJOIN\sdbo.tnpSchi
effelinHistory\sON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpSchieffelinHi

```



```

story.EmployeeID\sWHERE\s(dbo.tnpSchieffelinHistory.CalendarYear\s=\s?)
\sAND\s(dbo.tnpSchieffelinHistory.IsEligible\s=\s'y')\sAND\s(dbo.tnpEmp
loyee.LastName\s=\s?)\sORDER\sBY\sdbo.tnpEmployee.LastName\q,TCControlI
D_Unmatched=\qRsHistory\q,TCPConn=\qsaward\q,TCPDBObject=\qTables\q,T
CPDBObjectName=\qtnpSchieffelinHistory\s(dbo)\q,RCDBObject=\qRCSQLStat
ement\q,TCSQLStatement_Unmatched=\qSELECT\sdbo.tnpSchieffelinHistory.Ca
lendarYear,\sdbo.tnpSchieffelinHistory.EmployeeID,\sdbo.tnpSchieffelinH
istory.EligibilityCode,\sdbo.tnpEmployee.LastName,\sdbo.tnpEmployee.Fir
stName,\sdbo.tnpEmployee.MiddleName\sFROM\sdbo.tnpEmployee\sINNER\sJOIN
\sdbo.tnpSchieffelinHistory\sON\sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnp
SchieffelinHistory.EmployeeID\sWHERE\s(dbo.tnpSchieffelinHistory.Calend
arYear\s=\s?)\sAND\s(dbo.tnpSchieffelinHistory.IsEligible\s=\s'y')\sAND
\s(dbo.tnpEmployee.LastName\s=\s?)\sORDER\sBY\sdbo.tnpEmployee.LastName
\q,TCCursorType=\q3\s-\sStatic\q,TCCursorLocation=\q3\s-\sUse\sclient-
side\sursors\q,TCLockType=\q3\s-
\sOptimistic\q,TCCacheSize_Unmatched=\q10\q,TCCommTimeout_Unmatched=\q1
0\q,CCPrepared=0,CCallRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSynta
x_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q,TCTargetPlatform=\qInherit\sfrom\spage\q,RCCache=\qRCBook
Page\q,CCOpen=0,GCPParameters=(Rows=2,Row1=(CType_Unmatched=\q?\q,CParNa
me_Unmatched=\qParam1\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched
=\q4\q,CReq=1,CValue_Unmatched=\qintCalYear\q),Row2=(CType_Unmatched=\q
?\q,CParName_Unmatched=\qParam2\q,CDataType_Unmatched=\qVarChar\q,CSize
_Unmatched=\q30\q,CReq=1,CValue_Unmatched=\qstrLastName\q))"><PARAM
NAME="LocalPath" VALUE="../">

```

</OBJECT>

-->

<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->

<SCRIPT LANGUAGE="JavaScript" RUNAT="server">

function _setParametersRsHistory()

```

{
    RsHistory.setParameter(0,intCalYear);
    RsHistory.setParameter(1,strLastName);
}

```

function _initRsHistory()

```

{
    RsHistory.advise(RS_ONBEFOREOPEN, _setParametersRsHistory);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;
    cmdTmp.CommandText = 'SELECT
dbo.tnpSchieffelinHistory.CalendarYear,
dbo.tnpSchieffelinHistory.EmployeeID,

```



```

</OBJECT>

-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/RNavBar.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initRecordsetNavbar1()
{
    RecordsetNavbar1.setAlignment(1);
    RecordsetNavbar1.setButtonStyles(170);
    RecordsetNavbar1.setDataSource(RsHistory);
    RecordsetNavbar1.getButton(0).value = ' |< ';
    RecordsetNavbar1.getButton(1).value = ' < ';
    RecordsetNavbar1.getButton(2).value = ' > ';
    RecordsetNavbar1.getButton(3).value = ' >| ';
}
function _RecordsetNavbar1_ctor()
{
    CreateRecordsetNavbar('RecordsetNavbar1', _initRecordsetNavbar1,
null);
}
</script>
<% RecordsetNavbar1.display %>

<!--METADATA TYPE="DesignerControl" ends-->
&nbsp; Press Navbar button to update
        record. </FONT>

</P>
<P><A href="/Saward/admin/input_form3.html"><FONT
        face=Verdana>Add History for another Professor</FONT>

</A><FONT
        face=Verdana>
</FONT>
</P></td></tr></table></td></tr></table>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

Z. ELIGIBLE_DETAILED.ASP

```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>
<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../../../_ScriptLibrary/pm.asp"-->

```

```

<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>

<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="../../../images/spacer.gif"
width=160 border=0></td>
    <td width="602" height="3"><IMG height=1 src="../../../images/spacer.gif"
width=600 border=0></td>
    <td width="3" height="3"><IMG height=1 src="../../../images/spacer.gif"
width=1 border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p></td>
    <td><div align="center"><IMG height=86 src="../../../images/AwardHdr.gif"
width=311 border=0></div></td></tr>
</table>
<P>&nbsp;&nbsp;&nbsp;</P>
<P><A href="/Saward/admin/index.html"><STRONG><FONT
face=Verdana>Admin Home</FONT> </STRONG></A><br><br>
<%
Dim intYear
intYear = Request.Form ("cYear")
Dim strSQL
strSQL = "Exec snpViewEligibleFaculty "& intYear
%>
<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsEligible style="LEFT: 0px; TOP: 0px"
classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE=" (TCConn=\qsaward\q,TCDBObject_Unmatched=\qSQL\sStatement\q,TCDBO
bjectName_Unmatched=\qSELECT\sDISTINCT\sdbo.vnpEligibleFaculty.Descript
ion,\sdbo.vnpEligibleFaculty.FirstName,\sdbo.vnpEligibleFaculty.LastNam
e,\sdbo.vnpEligibleFaculty.MiddleName,\sdbo.vnpEligibleFaculty.Employee
ID,\sdbo.vnpFacultyHours.Year,\sdbo.vnpFacultyHours.AssignedLabHours,\s
dbo.vnpFacultyHours.AssignedLectureHours,\sdbo.vnpFacultyHours.ClassTyp
e,\sdbo.vnpFacultyHours.Segment,\sdbo.vnpFacultyHours.CourseID,\sdbo.vn
pFacultyHours.Quarter,\sdbo.vnpFacultyHours.Instructors,\sdbo.vnpEligib
leFaculty.CalendarYear\sFROM\sdbo.vnpEligibleFaculty\sINNER\sJOIN\sdbo.
vnpFacultyHours\sON\sdbo.vnpEligibleFaculty.EmployeeID\s=\sdbo.vnpFacul
tyHours.EmployeeID\sWHERE\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?)\

```



```

sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s2) \sAND\s (dbo.vnpEligibleFaculty.IsEligible\s=\s'y') \sAND\s (dbo.vnpEligibleFaculty.IsPrimaryDepartment\s=\s'y') \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s3) \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s4) \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?\s+\s1) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s1) \sORDER\sBY\sdbo.vnpEligibleFaculty.Description,\sdbo.vnpEligibleFaculty.LastName,\sdbo.vnpFacultyHours.Year\sDESC,\sdbo.vnpFacultyHours.Quarter\q,TCCControlID_Unmatched=\qRsEligible\q,TCPConn=\qaward\q,TCPDBObject=\qTables\q,TCPDBObjectName=\qBALLOT2000tmp\s (dbo)\q,RCDObject=\qRCSQLStatement\q,TCSQLStatement_Unmatched=\qSELECT\sDISTINCT\sdbo.vnpEligibleFaculty.Description,\sdbo.vnpEligibleFaculty.FirstName,\sdbo.vnpEligibleFaculty.LastName,\sdbo.vnpEligibleFaculty.MiddleName,\sdbo.vnpEligibleFaculty.EmployeeID,\sdbo.vnpFacultyHours.Year,\sdbo.vnpFacultyHours.AssignedLabHours,\sdbo.vnpFacultyHours.AssignedLectureHours,\sdbo.vnpFacultyHours.ClassType,\sdbo.vnpFacultyHours.Segment,\sdbo.vnpFacultyHours.CourseID,\sdbo.vnpFacultyHours.Quarter,\sdbo.vnpFacultyHours.Instructors,\sdbo.vnpEligibleFaculty.CalendarYear\sFROM\sdbo.vnpEligibleFaculty\sINNER\sJOIN\sdbo.vnpFacultyHours\sON\sdbo.vnpEligibleFaculty.EmployeeID\s=\sdbo.vnpFacultyHours.EmployeeID\sWHERE\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s2) \sAND\s (dbo.vnpEligibleFaculty.IsEligible\s=\s'y') \sAND\s (dbo.vnpEligibleFaculty.IsPrimaryDepartment\s=\s'y') \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s3) \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s4) \sOR\s (dbo.vnpEligibleFaculty.CalendarYear\s=\s?) \sAND\s (dbo.vnpFacultyHours.Year\s=\s?\s+\s1) \sAND\s (dbo.vnpFacultyHours.Quarter\s=\s1) \sORDER\sBY\sdbo.vnpEligibleFaculty.Description,\sdbo.vnpEligibleFaculty.LastName,\sdbo.vnpFacultyHours.Year\sDESC,\sdbo.vnpFacultyHours.Quarter\q,TCCursorType=\q3\s-\sStatic\q,TCCursorLocation=\q3\s-\sUse\sclient-side\sursors\q,TCLockType=\q1\s-\sRead\sOnly\q,TCCacheSize_Unmatched=\q10\q,TCCommTimeout_Unmatched=\q10\q,CCPrepared=0,CCallRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSyntax_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qServer\s (ASP) \q,TCTargetPlatform=\qInherit\sfrom\spage\q,RCCache=\qRCBookPage\q,CCOpen=0,GCPParameters=(Rows=8,Row1=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam1\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row2=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam2\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row3=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam3\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row4=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam4\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row5=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam5\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row6=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam6\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row7=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam7\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q),Row8=(CType_Unmatched=\q?\q,CParName_Unmatched=\qParam8\

```

```

q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched=\q4\q,CReq=1,CValue_Un
nmatched=\qintYear\q)))"><PARAM NAME="LocalPath" VALUE="..">
</OBJECT>

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsEligible()
{
    RsEligible.setParameter(0,intYear);
    RsEligible.setParameter(1,intYear);
    RsEligible.setParameter(2,intYear);
    RsEligible.setParameter(3,intYear);
    RsEligible.setParameter(4,intYear);
    RsEligible.setParameter(5,intYear);
    RsEligible.setParameter(6,intYear);
    RsEligible.setParameter(7,intYear);
}
function _initRsEligible()
{
    RsEligible.advise(RS_ONBEFOREOPEN, _setParametersRsEligible);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;
    cmdTmp.CommandText = 'SELECT DISTINCT
dbo.vnpEligibleFaculty.Description, dbo.vnpEligibleFaculty.FirstName,
dbo.vnpEligibleFaculty.LastName, dbo.vnpEligibleFaculty.MiddleName,
dbo.vnpEligibleFaculty.EmployeeID, dbo.vnpFacultyHours.Year,
dbo.vnpFacultyHours.AssignedLabHours,
dbo.vnpFacultyHours.AssignedLectureHours,
dbo.vnpFacultyHours.ClassType, dbo.vnpFacultyHours.Segment,
dbo.vnpFacultyHours.CourseID, dbo.vnpFacultyHours.Quarter,
dbo.vnpFacultyHours.Instructors, dbo.vnpEligibleFaculty.CalendarYear
FROM dbo.vnpEligibleFaculty INNER JOIN dbo.vnpFacultyHours ON
dbo.vnpEligibleFaculty.EmployeeID = dbo.vnpFacultyHours.EmployeeID
WHERE (dbo.vnpEligibleFaculty.CalendarYear = ?) AND
(dbo.vnpFacultyHours.Year = ?) AND (dbo.vnpFacultyHours.Quarter = 2)
AND (dbo.vnpEligibleFaculty.IsEligible = \'y\') AND
(dbo.vnpEligibleFaculty.IsPrimaryDepartment = \'y\') OR
(dbo.vnpEligibleFaculty.CalendarYear = ?) AND (dbo.vnpFacultyHours.Year
= ?) AND (dbo.vnpFacultyHours.Quarter = 3) OR
(dbo.vnpEligibleFaculty.CalendarYear = ?) AND (dbo.vnpFacultyHours.Year
= ?) AND (dbo.vnpFacultyHours.Quarter = 4) OR
(dbo.vnpEligibleFaculty.CalendarYear = ?) AND (dbo.vnpFacultyHours.Year
= ? + 1) AND (dbo.vnpFacultyHours.Quarter = 1) ORDER BY

```

```

dbo.vnpEligibleFaculty.Description, dbo.vnpEligibleFaculty.LastName,
dbo.vnpFacultyHours.Year DESC, dbo.vnpFacultyHours.Quarter';
    rsTmp.CacheSize = 10;
    rsTmp.CursorType = 3;
    rsTmp.CursorLocation = 3;
    rsTmp.LockType = 1;
    RsEligible.setRecordSource(rsTmp);
    if (thisPage.getState('pb_RsEligible') != null)
        RsEligible.setBookmark(thisPage.getState('pb_RsEligible'));
}
function _RsEligible_ctor()
{
    CreateRecordset('RsEligible', _initRsEligible, null);
}
function _RsEligible_dtor()
{
    RsEligible._preserveState();
    thisPage.setState('pb_RsEligible', RsEligible.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" ends-->
<%RsEligible.Open strSQL%>
<%
Dim strDescription
Dim strEmpID
%>
</P>
<table border = "0" align = "center">
<%
do while not RsEligible.EOF
    if RsEligible.fields.getValue ("Description")<>
strDescription then%>
        <tr><td><strong><%Response.Write
RsEligible.fields.getValue
("Description")%></strong></td><th>Segment</th><th>Qtr</th><th>LecHrs</
th><th>LabHrs</th><th>#Instr</th><th>Year</th><th>Type</th></tr>

        <tr><td><strong><EM>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<%Respons
e.Write RsEligible.fields.getValue ("LastName") & ", "
        Response.Write RsEligible.fields.getValue
("FirstName") & " "
        Response.Write RsEligible.fields.getValue
("MiddleName") & "." %></strong></EM></td>

        <tr><td>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<%Respons
e.Write RsEligible.fields.getValue ("CourseID")%></td>
        <td align = "middle"><%Response.Write
RsEligible.fields.getValue ("Segment")%></td>
        <td align = "middle"><%Response.Write
RsEligible.fields.getValue ("Quarter")%></td>
        <td align = "middle"><%Response.Write
RsEligible.fields.getValue ("AssignedLectureHours")%></td>
        <td align = "middle"><%Response.Write
RsEligible.fields.getValue ("AssignedLabHours")%></td>

```



```

        strDescription = RsEligible.fields.getValue
("Description")
        strEmpID = RsEligible.fields.getValue ("EmployeeID")
        RsEligible.MoveNext
Loop
RsEligible.Close ()
%>

</table>
</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

AA. ELIGIBLE_SIMPLE.ASP

```

<!--
<%@ Language=VBScript %>
<!--#include virtual="/adovbs.inc"-->
<%
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Dim intDate
intDate = Request.Form ("cYear")
Dim strSQL
strSQL = "Exec snpViewEligibleFaculty "& intDate
%>
<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="../images/spacer.gif"
width=160 border=0></td>
        <td width="602" height="3"><IMG height=1 src="../images/spacer.gif"
width=600 border=0></td>
        <td width="3" height="3"><IMG height=1 src="../images/spacer.gif"
width=1 border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>

```

```

        &nbsp;DSN: 878-2441/2</b></font></p></td>
    <td><div align="center"><IMG height=86 src="../../images/AwardHdr.gif"
width=311 border=0></div></td></tr>
</table>
<br>
<p>
<%
        'Create Recordset object
Dim RsEligibleFaculty
Set RsEligibleFaculty = Server.CreateObject ("ADODB.Recordset")
RsEligibleFaculty.CursorLocation = adUseClient
RsEligibleFaculty.Open strSQL, objconn
RsEligibleFaculty.Sort = "Description, LastName"
%><br><A href="/Saward/admin/index.html"><strong><FONT
face=Verdana>Admin Home</FONT>
</strong></A><br>
<table border="0" align="center" width="556" style="WIDTH: 556px;
HEIGHT: 77px">
<%
        'Get all eligible faculty from vnpEligibleFaculty using
snpViewEligibleFaculty
        Do while not RsEligibleFaculty.EOF
            iCount = iCount + 1

            'strDescription is used to format the page with all of the faculty
            'written to the page underneath thier department Description
            Dim strDescription
            if RsEligibleFaculty("Description") <> strDescription then
%><!--Writes the Department description and faculty until
strDescription is not equal to "Description" then writes the next
Department and so on-->

                <tr><td align="middle"><FONT face=Verdana><strong><br><%Response.Write
RsEligibleFaculty ("Description")%></A></strong></td></tr>
                <tr><td><br><%
                    Response.Write RsEligibleFaculty("LastName") & ", "
                    Response.Write RsEligibleFaculty("FirstName") & " "
                    Response.Write RsEligibleFaculty("MiddleName") & "."
%></td></tr><%
                    else
%>
<tr><td>
<%
                    Response.Write RsEligibleFaculty("LastName") & ", "
                    Response.Write RsEligibleFaculty("FirstName") & " "
                    Response.Write RsEligibleFaculty("MiddleName") & "."
%></font></td></tr>
<%
            end if

            strDescription= RsEligibleFaculty ("Description")
            RsEligibleFaculty.MoveNext
        Loop
RsEligibleFaculty.Close ()
%>

```

```

</table>

        <p> </p>
        <p>&nbsp;</p>
        <p>&nbsp;</p>
        <p>&nbsp;</p></TD></TR></TABLE></TD>
    <td width="3" height="1"></td></TR></TABLE>
</body>
</html>
-->

```

BB. EMPID.ASP

```

<!--
<%@ Language=VBScript %>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;<b>1 University Circle<br>
&nbsp;<b>Monterey, CA 93943-5001<br>
&nbsp;<b>(831) 656-2441/2<br>
&nbsp;<b>DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;&nbsp;</p>

```

```

<p align="left">&nbsp;&nbsp;&nbsp;<A href="index.html"><b><FONT
face=Verdana>Admin Home</FONT> </b></A></p>
    <P align=left>&nbsp;&nbsp;&nbsp;<A
href="/Saward/admin/input_form8.html"><STRONG><FONT
face=Verdana>Search
Again</FONT>
</STRONG></A></P>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
        <table width="583" border="0" align="center">
            <tr>
                <td width="575">
                    <div align="center"><IMG height=86
src="../../../images/AwardHdr.gif" width=311></div>
                </td>
            </tr>
            <tr>
                <td width="575">
                    &nbsp;&nbsp;&nbsp;
                    <p>&nbsp;&nbsp;&nbsp;</p>

<%Dim intEmpID
intEmpID= Request.Form ("txtEmpID")

%>
<P>
    <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsName style="LEFT: 0px; TOP: 0px"
    classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q, TCDBObject_Unmatched=\qSQL\sStatement\q, TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpEmployee.EmployeeID, \sdbo.tnpEmplo
yee.LastName, \sdbo.tnpEmployee.FirstName, \sdbo.tnpEmployee.MiddleName, \
sdbo.tnpDepartment.Department, \sdbo.tnpDepartment.Description\sFROM\sdb
o.tnpEmployee\sINNER\sJOIN\sdbo.tnpDepartment_Employee\sON\sdbo.tnpEmpl
oyee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\sINNER\sJOIN\
sdbo.tnpDepartment\sON\sdbo.tnpDepartment_Employee.DepartmentID\s=\sdbo
.tnpDepartment.DepartmentID\sWHERE\s (dbo.tnpEmployee.EmployeeID\s=\s?) \
q, TCControlID_Unmatched=\qRsName\q, TCPPConn=\qsaward\q, TCPPDBObject=\qT
ables\q, TCPPDBObjectName=\qtntpEmployee\s (dbo) \q, RCDBObject=\qRCSQLState
ment\q, TCSQLStatement_Unmatched=\qSELECT\sdbo.tnpEmployee.EmployeeID, \s
dbo.tnpEmployee.LastName, \sdbo.tnpEmployee.FirstName, \sdbo.tnpEmployee.
MiddleName, \sdbo.tnpDepartment.Department, \sdbo.tnpDepartment.Descripti
on\sFROM\sdbo.tnpEmployee\sINNER\sJOIN\sdbo.tnpDepartment_Employee\sON\
sdbo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\s
INNER\sJOIN\sdbo.tnpDepartment\sON\sdbo.tnpDepartment_Employee.Departme
ntID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s (dbo.tnpEmployee.Empl
oyeeID\s=\s?) \q, TCCursorType=\q3\s-\sStatic\q, TCCursorLocation=\q3\s-
\sUse\sclient-side\sursors\q, TCLockType=\q3\s-
\sOptimistic\q, TCCacheSize_Unmatched=\q10\q, TCCommTimeout_Unmatched=\q1
0\q, CCPrepared=0, CCallRecords=1, TCNRecords_Unmatched=\q10\q, TCODBCSynta
x_Unmatched=\q\q, TCHTargetPlatform=\q\q, TCHTargetBrowser_Unmatched=\qSe
rver\s (ASP) \q, TCTargetPlatform=\qInherit\sfrom\spage\q, RCCache=\qRCBook
Page\q, CCOpen=0, GCPParameters=(Rows=1, Row1=(CType_Unmatched=\q?\q, CParNa
me_Unmatched=\qParam1\q, CDataType_Unmatched=\qInteger\q, CSize_Unmatched

```



```

=\q4\q,CReq=1,CValue_Unmatched=\qintEmpID\q))"><PARAM NAME="LocalPath"
VALUE="..">

</OBJECT>

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsName()
{
    RsName.setParameter(0,intEmpID);
}
function _initRsName()
{
    RsName.advise(RS_ONBEFOREOPEN, _setParametersRsName);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;
    cmdTmp.CommandText = 'SELECT dbo.tnpEmployee.EmployeeID,
dbo.tnpEmployee.LastName, dbo.tnpEmployee.FirstName,
dbo.tnpEmployee.MiddleName, dbo.tnpDepartment.Department,
dbo.tnpDepartment.Description FROM dbo.tnpEmployee INNER JOIN
dbo.tnpDepartment_Employee ON dbo.tnpEmployee.EmployeeID =
dbo.tnpDepartment_Employee.EmployeeID INNER JOIN dbo.tnpDepartment ON
dbo.tnpDepartment_Employee.DepartmentID =
dbo.tnpDepartment.DepartmentID WHERE (dbo.tnpEmployee.EmployeeID = ?)';
    rsTmp.CacheSize = 10;
    rsTmp.CursorType = 3;
    rsTmp.CursorLocation = 3;
    rsTmp.LockType = 3;
    RsName.setRecordSource(rsTmp);
    if (thisPage.getState('pb_RsName') != null)
        RsName.setBookmark(thisPage.getState('pb_RsName'));
}
function _RsName_ctor()
{
    CreateRecordset('RsName', _initRsName, null);
}
function _RsName_dtor()
{
    RsName._preserveState();
    thisPage.setState('pb_RsName', RsName.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" ends-->
</P>

```

```

<%RsName.open%>
<P>&nbsp;</P>
<P>
<TABLE cellSpacing=1 cellPadding=1 border=0 style="WIDTH: 642px;
HEIGHT: 62px">

    <TR>
        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>EmpID</STRONG></FONT></P></TD>

        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>Last
Name</STRONG></FONT></P></TD>

        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>First
Name</STRONG></FONT></P></TD>

        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>Middle
Name</STRONG></FONT></P></TD>

        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>DeptID</STRONG></FONT></P></TD>

        <TD>

            <P align=center><FONT
            face=Verdana><STRONG>Department</STRONG></FONT></P></TD></TR>
    <TR>
        <TD><FONT face=Verdana>
            <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=EmpID style="WIDTH: 42px; HEIGHT: 19px"
            classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0
width=42
            height=19><PARAM NAME="_ExtentX" VALUE="1111"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="EmpID"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsName"><PARAM NAME="DataField"
VALUE="EmployeeID"><PARAM NAME="Enabled" VALUE="0"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="10"><PARAM
NAME="DisplayWidth" VALUE="7"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="../">

            </OBJECT>

-->
<!--#INCLUDE FILE="../_ScriptLibrary/TextBox.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initEmpID()
{
    EmpID.setStyle(TXT_TEXTBOX);
    EmpID.setDataSource(RsName);
    EmpID.setDataField('EmployeeID');
    EmpID.disabled = true;
    EmpID.setMaxLength(10);

```

```

        EmpID.setColumnCount(7);
    }
    function _EmpID_ctor()
    {
        CreateTextbox('EmpID', _initEmpID, null);
    }
</script>
<% EmpID.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</TD>
    <TD>
        <P align=center><FONT face=Verdana>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtLName
        style="LEFT: 0px; WIDTH: 104px; TOP: 1px; HEIGHT:
19px"
        classid=clsid:B5F0E469-DC5F-11D0-9846-
0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2752"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtLName"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="LastName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="17"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

        </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtLName()
{
    txtLName.setStyle(TXT_TEXTBOX);
    txtLName.setDataSource(RsName);
    txtLName.setDataField('LastName');
    txtLName.disabled = true;
    txtLName.setMaxLength(20);
    txtLName.setColumnCount(17);
}
function _txtLName_ctor()
{
    CreateTextbox('txtLName', _inittxtLName, null);
}
</script>
<% txtLName.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</P>
</TD>
    <TD>
        <P align=center><FONT face=Verdana>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtFname style="WIDTH: 104px; HEIGHT: 19px"

```

```

        classid=clsid:B5F0E469-DC5F-11D0-9846-
0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2752"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtFname"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="FirstName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="17"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

```

```

</OBJECT>

```

```

-->

```

```

<SCRIPT LANGUAGE=JavaScript RUNAT=Server>

```

```

function _inittxtFname()

```

```

{
    txtFname.setStyle(TXT_TEXTBOX);
    txtFname.setDataSource(RsName);
    txtFname.setDataField('FirstName');
    txtFname.disabled = true;
    txtFname.setMaxLength(20);
    txtFname.setColumnCount(17);
}

```

```

function _txtFname_ctor()

```

```

{
    CreateTextbox('txtFname', _inittxtFname, null);
}

```

```

</script>

```

```

<% txtFname.display %>

```

```

<!--METADATA TYPE="DesignerControl" ends-->

```

```

</FONT>

```

```

</P>

```

```

</TD>

```

```

    <TD>

```

```

        <P align=center><FONT face=Verdana>

```

```

        <!--METADATA TYPE="DesignerControl" startspan

```

```

<OBJECT id=txtMName style="WIDTH: 108px; HEIGHT: 19px"

```

```

        classid=clsid:B5F0E469-DC5F-11D0-9846-

```

```

0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2858"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtMName"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="MiddleName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="18"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

```

```

</OBJECT>

```

```

-->

```

```

<SCRIPT LANGUAGE=JavaScript RUNAT=Server>

```

```

function _inittxtMName()

```

```

{
    txtMName.setStyle(TXT_TEXTBOX);
    txtMName.setDataSource(RsName);
    txtMName.setDataField('MiddleName');
    txtMName.disabled = true;
}

```

```

        txtMName.setMaxLength(20);
        txtMName.setColumnCount(18);
    }
    function _txtMName_ctor()
    {
        CreateTextbox('txtMName', _inittxtMName, null);
    }
</script>
<% txtMName.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</P>
</TD>
    <TD>
        <P align=center><FONT face=Verdana>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtDept style="WIDTH: 66px; HEIGHT: 19px"
        classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0
width=66
        height=19><PARAM NAME="_ExtentX" VALUE="1746"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="txtDept"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsName"><PARAM NAME="DataField"
VALUE="Department"><PARAM NAME="Enabled" VALUE="0"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="20"><PARAM
NAME="DisplayWidth" VALUE="11"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="../">

        </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtDept()
{
    txtDept.setStyle(TXT_TEXTBOX);
    txtDept.setDataSource(RsName);
    txtDept.setDataField('Department');
    txtDept.disabled = true;
    txtDept.setMaxLength(20);
    txtDept.setColumnCount(11);
}
function _txtDept_ctor()
{
    CreateTextbox('txtDept', _inittxtDept, null);
}
</script>
<% txtDept.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</P>
</TD>
    <TD><FONT face=Verdana>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtDescription style="WIDTH: 210px; HEIGHT: 19px"

```

```

classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0
width=210
height=19><PARAM NAME="_ExtentX" VALUE="5556"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id"
VALUE="txtDescription"><PARAM NAME="ControlType" VALUE="0"><PARAM
NAME="Lines" VALUE="3"><PARAM NAME="DataSource" VALUE="RsName"><PARAM
NAME="DataField" VALUE="Description"><PARAM NAME="Enabled"
VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars"
VALUE="50"><PARAM NAME="DisplayWidth" VALUE="35"><PARAM NAME="Platform"
VALUE="0"><PARAM NAME="LocalPath" VALUE="..">

```

```

</OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtDescription()
{
    txtDescription.setStyle(TXT_TEXTBOX);
    txtDescription.setDataSource(RsName);
    txtDescription.setDataField('Description');
    txtDescription.disabled = true;
    txtDescription.setMaxLength(50);
    txtDescription.setColumnCount(35);
}
function _txtDescription_ctor()
{
    CreateTextbox('txtDescription', _inittxtDescription, null);
}
</script>
<% txtDescription.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</FONT>
</TD></TR></TABLE></P></td></tr></table></td></tr></table>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

CC. HISTORY.ASP

```

<!--
<%@ Language=VBScript %>
<%
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Set RsHistory = Server.CreateObject ("ADODB.Recordset")
sqltext= "SELECT Rank, EmployeeID, FirstName, LastName, MiddleName, "
&_

```

```

"CalendarYear, EligibilityCode, NBallots, NumberSupportStatements,
FinalScore," &_
"PCScore, PCRank FROM vnpSchiefHistory;"
RsHistory.Open sqltext, objconn
%>

<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="../images/spacer.gif"
width=160 border=0></td>
    <td width="602" height="3"><IMG height=1 src="../images/spacer.gif"
width=600 border=0></td>
    <td width="3" height="3"><IMG height=1 src="../images/spacer.gif"
width=1 border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;&nbsp;&1 University Circle<br>
&nbsp;&nbsp;&Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&(831) 656-2441/2<br>
&nbsp;&nbsp;&DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&</p>
<p align="left">&nbsp;&nbsp;&&nbsp;&nbsp;&</p>
<p align="left">&nbsp;&nbsp;&&nbsp;&nbsp;& <b><A href="index.html"><FONT
  face=Verdana>Admin Home</FONT> </A></b></p>
<p align="left">&nbsp;&nbsp;&&nbsp;&nbsp;& <b><A
href="/Saward/admin/history1.asp"><FONT face=Verdana>Committee
Copy</FONT> </A></b></p>
<p align="left"><b> &nbsp;&nbsp;&&nbsp;&nbsp;&</b></p>
    <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&</p>
    <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&</p>
  </td>
  <td valign="top" height="1" width="602">
    <table width="583" border="0" align="center">
      <tr>
        <td width="575">
          <div align="center"><IMG height=86
src="../images/AwardHdr.gif" width=311 border=0></div>
        </td>
      </tr>
      <tr>
        <td nowrap>
          <%Dim cYear
cYear = Year(Date) - 1%>

```

```

<p align="center"></p>
<p align="right"><strong>For
    teaching year <% Response.write cYear%></strong></p>
<p align="left">
<strong><FONT face=Verdana>PREVIOUS PERFORMANCE
    DATA</FONT> </strong><br><br>
<TABLE width = 975>
    <TBODY>
<tr><th>Rank</th><th>Name</th><th>Year</th><th>Score</th><th>N</th><th>
D</th><th>PcRank</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th> Code</th>
<%
Dim ID
Dim iCount
RsHistory.MoveFirst
Do while not RsHistory.EOF

    if ID = RsHistory ("EmployeeID") and iCount <= 5 then 'Number of
years to write to page
        iCount = iCount + 1
        %>
        <td><%Response.Write RsHistory ("CalendarYear")%></td>
<td><%Response.Write RsHistory ("NBallots")%></td>
        </td><td align="middle"><%Response.Write RsHistory
("EligibilityCode")%></td>
        <% RsHistory.MoveNext

    elseif ID = RsHistory ("EmployeeID") and iCount > 5 then
%><td><%Response.write "&nbsp;"%></td><%
RsHistory.MoveNext

    else
%>
        <tr><td align="middle"><%Response.Write RsHistory ("Rank")%></td>

        <td nowrap><%Response.Write RsHistory ("LastName")%>

        ,
            <%Response.Write RsHistory ("FirstName")%>
            <%Response.Write "&nbsp;" & RsHistory ("MiddleName")%></td>
        <td><%Response.Write "&nbsp;" & RsHistory ("CalendarYear")%></td>
        <td><%Response.Write RsHistory ("FinalScore") %></td>
        <td><%Response.write RsHistory ("NBallots")%></td>
        <td><%Response.write RsHistory ("NumberSupportStatements")%></td>
        <td align="middle"><%Response.write "&nbsp;" & RsHistory
("PCRank")%></td>
        <% ID = RsHistory ("EmployeeID")
RsHistory.MoveNext
iCount = 0
end if
loop
%></tr>
</table></P></td></tr></table></TD></TR></TBODY></TABLE>

```



```
</body>
</html>
```

```
-->
```

DD. HISTORY1.ASP

```
<!--
<%@ Language=VBScript %>
<%
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.Open Application("saward_ConnectionString")

Set RsHistory = Server.CreateObject ("ADODB.Recordset")
sqltext= "SELECT Rank, EmployeeID, FirstName, LastName, MiddleName, "
&_
"CalendarYear, EligibilityCode, NBallots, NumberSupportStatements,
FinalScore," &_
"PCScore, PCRank FROM vnpSchiefHistory;"
RsHistory.Open sqltext, objconn
%>

<html>
<head>
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
  <tr>
    <td width="192" height="3"><IMG height=1 src="../images/spacer.gif"
width=160 border=0></td>
    <td width="602" height="3"><IMG height=1 src="../images/spacer.gif"
width=600 border=0></td>
    <td width="3" height="3"><IMG height=1 src="../images/spacer.gif"
width=1 border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;<b>1 University Circle<br>
&nbsp;<b>Monterey, CA 93943-5001<br>
&nbsp;<b>(831) 656-2441/2<br>
&nbsp;<b>DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;</p>
<p align="left">&nbsp;<b><A href="index.html"><FONT
  face=Verdana>Admin Home</FONT> </A></b></p>
<p align="left"><FONT face=Verdana>&nbsp;<b><A
href="/Saward/admin/history.asp">Chairman Copy</A></b></FONT></p>
<p align="left"><b> &nbsp;<b></b></p>
```

```

        <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;</p>
        <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
        <table width="583" border="0" align="center">
            <tr>
                <td width="575">
                    <div align="center"><IMG height=86
src="../../images/AwardHdr.gif" width=311 border=0></div>
                </td>
            </tr>
            <tr>
                <td nowrap></td></tr>
        </table>
        <%Dim cYear
cYear = Year(Date) - 1%>
        <p align="center"></p>
        <p align="right"><strong><FONT
face=Verdana>For teaching year <% Response.write
cYear%></FONT></strong></p>
        <p align="left">
        <strong><FONT
face=Verdana>PREVIOUS PERFORMANCE DATA</FONT> </strong><br><br>

<%
Dim ID
Dim iCount
RsHistory.MoveFirst
%>
<table border=0 width =875>
    <TBODY>
    <tr><th>Rank</th><th>Year</th><th>Score</th><th>N</th><th>D</th><th>PcR
ank</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th>
Code</th><th>Year</th><th>N</th><th> Code</th>
    <%
Do while not RsHistory.EOF

    if ID = RsHistory ("EmployeeID") and iCount <= 5 then 'Number of
years to write to page
        iCount = iCount + 1
        %>
        <td><%Response.Write RsHistory ("CalendarYear")%></td>
    <td><%Response.Write RsHistory ("NBallots")%></td>
        </td><td align = middle><%Response.Write RsHistory
("EligibilityCode")%></td>
        <% RsHistory.MoveNext

    elseif ID = RsHistory ("EmployeeID") and iCount > 5 then
    %><td><%Response.write "&nbsp;"%></td><%
RsHistory.MoveNext

```

```

else
    %>

    <tr><td align = middle><%Response.Write RsHistory
("Rank")%></td>
    <td><%Response.Write "&nbsp;" & RsHistory ("CalendarYear")%></td>
    <td><%Response.Write RsHistory ("FinalScore") %></td>
    <td><%Response.write RsHistory ("NBallots")%></td>
    <td><%Response.write RsHistory ("NumberSupportStatements")%></td>
    <td align=middle><%Response.write "&nbsp;" & RsHistory
("PCRank")%></td>
    <% ID = RsHistory ("EmployeeID")
    RsHistory.MoveNext
    iCount = 0
    end if
    loop
    %></tr>
</table></P></TD></TR></TBODY></TABLE>

</body>
</html>
-->

```

EE. LNAME.ASP

```

<!--
<%@ Language=VBScript %>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../../../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
<title>Schieffelin Teaching Award</title>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
        <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
        <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
    </tr>

```

```
|  |  |  |
| --- | --- | --- |
| | | |
|  |
|  |

```

<%Dim strLName
strLName= Request.Form ("txtLName")
%>
<P>

```

<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsName style="LEFT: 0px; TOP: 0px"
classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q, TCDBObject_Unmatched=\qSQL\sStatement\q, TCDBO
bjectName_Unmatched=\qSELECT\sdbo.tnpEmployee.EmployeeID, \sdbo.tnpEmpl
oyee.LastName, \sdbo.tnpEmployee.FirstName, \sdbo.tnpEmployee.MiddleName, \
sdbo.tnpDepartment.Department, \sdbo.tnpDepartment.Description\sFROM\sdb
o.tnpEmployee\sINNER\sJOIN\sdbo.tnpDepartment_Employee\son\sdbo.tnpEmpl
oyee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\sINNER\sJOIN\
sdbo.tnpDepartment\son\sdbo.tnpDepartment_Employee.DepartmentID\s=\sdbo
.tnpDepartment.DepartmentID\sWHERE\s (dbo.tnpEmployee.LastName\s=\s?) \q,
TCControlID_Unmatched=\qRsName\q, TCPPConn=\qsaward\q, TCPPDBObject=\qTab
les\q, TCPPDBObjectName=\qtnpEmployee\s (dbo) \q, RCDBObject=\qRCSQLStateme
nt\q, TCSQLStatement_Unmatched=\qSELECT\sdbo.tnpEmployee.EmployeeID, \sdb
o.tnpEmployee.LastName, \sdbo.tnpEmployee.FirstName, \sdbo.tnpEmployee.Mi
ddleName, \sdbo.tnpDepartment.Department, \sdbo.tnpDepartment.Description
\sFROM\sdbo.tnpEmployee\sINNER\sJOIN\sdbo.tnpDepartment_Employee\son\sdb

```

```

bo.tnpEmployee.EmployeeID\s=\sdbo.tnpDepartment_Employee.EmployeeID\sIN
NER\sJOIN\sdbo.tnpDepartment\sON\sdbo.tnpDepartment_Employee.Department
ID\s=\sdbo.tnpDepartment.DepartmentID\sWHERE\s (dbo.tnpEmployee.LastName
\s=\s?)\q,TCCursorType=\q3\s-\sStatic\q,TCCursorLocation=\q3\s-
\sUse\sclient-side\sursors\q,TCLockType=\q3\s-
\sOptimistic\q,TCCacheSize_Unmatched=\q10\q,TCCommTimeout_Unmatched=\q1
0\q,CCPrepared=0,CCallRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSynta
x_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q,TCTargetPlatform=\qInherit\sfrom\spage\q,RCCache=\qRCBook
Page\q,CCOpen=0,GCPParameters=(Rows=1,Row1=(CType_Unmatched=\q?\q,CParNa
me_Unmatched=\qParam1\q,CDataType_Unmatched=\qVarChar\q,CSize_Unmatched
=\q30\q,CReq=1,CValue_Unmatched=\qstrLName\q)) "><PARAM
NAME="LocalPath" VALUE="../">

```

</OBJECT>

```

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsName()
{
    RsName.setParameter(0,strLName);
}
function _initRsName()
{
    RsName.advise(RS_ONBEFOREOPEN, _setParametersRsName);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;
    cmdTmp.CommandText = 'SELECT dbo.tnpEmployee.EmployeeID,
dbo.tnpEmployee.LastName, dbo.tnpEmployee.FirstName,
dbo.tnpEmployee.MiddleName, dbo.tnpDepartment.Department,
dbo.tnpDepartment.Description FROM dbo.tnpEmployee INNER JOIN
dbo.tnpDepartment_Employee ON dbo.tnpEmployee.EmployeeID =
dbo.tnpDepartment_Employee.EmployeeID INNER JOIN dbo.tnpDepartment ON
dbo.tnpDepartment_Employee.DepartmentID =
dbo.tnpDepartment.DepartmentID WHERE (dbo.tnpEmployee.LastName = ?)';
    rsTmp.CacheSize = 10;
    rsTmp.CursorType = 3;
    rsTmp.CursorLocation = 3;
    rsTmp.LockType = 3;
    RsName.setRecordSource(rsTmp);
    if (thisPage.getState('pb_RsName') != null)
        RsName.setBookmark(thisPage.getState('pb_RsName'));
}

```

```

function _RsName_ctor()
{
    CreateRecordset('RsName', _initRsName, null);
}
function _RsName_dtor()
{
    RsName._preserveState();
    thisPage.setState('pb_RsName', RsName.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
<%RsName.open%>
<P>&nbsp;</P>
<P>
<TABLE cellSpacing=1 cellPadding=1 border=0>

    <TR>
        <TD>

            <P align=center><STRONG><FONT
            face=Verdana>EmpID</FONT></STRONG></P></TD>

        <TD>

            <P align=center><STRONG><FONT face=Verdana>Last
            Name</FONT> </STRONG></P></TD>

        <TD>

            <P align=center><STRONG><FONT face=Verdana>First
            Name</FONT> </STRONG></P></TD>

        <TD>

            <P align=center><STRONG><FONT face=Verdana>Middle
            Name</FONT> </STRONG></P></TD>

        <TD>

            <P align=center><STRONG><FONT
            face=Verdana>DeptID</FONT></STRONG></P></TD>

        <TD>

            <P align=center><STRONG><FONT
            face=Verdana>Department</FONT></STRONG></P></TD></TR>

    <TR>
        <TD>

            <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=EmpID style="WIDTH: 42px; HEIGHT: 19px"
            classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0
width=42
            height=19><PARAM NAME="_ExtentX" VALUE="1111"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="EmpID"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsName"><PARAM NAME="DataField"
VALUE="EmployeeID"><PARAM NAME="Enabled" VALUE="0"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="10"><PARAM
NAME="DisplayWidth" VALUE="7"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="../">

            </OBJECT>

            -->
<!--#INCLUDE FILE="../_ScriptLibrary/TextBox.ASP"-->

```

```

<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initEmpID()
{
    EmpID.setStyle(TXT_TEXTBOX);
    EmpID.setDataSource(RsName);
    EmpID.setDataField('EmployeeID');
    EmpID.disabled = true;
    EmpID.setMaxLength(10);
    EmpID.setColumnCount(7);
}
function _EmpID_ctor()
{
    CreateTextbox('EmpID', _initEmpID, null);
}
</script>
<% EmpID.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</TD>
        <P align=center>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtLName
        style="LEFT: 0px; WIDTH: 104px; TOP: 1px; HEIGHT:
19px"
        classid=clsid:B5F0E469-DC5F-11D0-9846-
0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2752"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtLName"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="LastName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="17"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

        </OBJECT>

-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtLName()
{
    txtLName.setStyle(TXT_TEXTBOX);
    txtLName.setDataSource(RsName);
    txtLName.setDataField('LastName');
    txtLName.disabled = true;
    txtLName.setMaxLength(20);
    txtLName.setColumnCount(17);
}
function _txtLName_ctor()
{
    CreateTextbox('txtLName', _inittxtLName, null);
}
</script>
<% txtLName.display %>

<!--METADATA TYPE="DesignerControl" ends-->

```

```

</P>
</TD>
    <P align=center>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtFname style="WIDTH: 104px; HEIGHT: 19px"
        classid=clsid:B5F0E469-DC5F-11D0-9846-
0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2752"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtFname"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="FirstName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="17"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

                                </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtFname()
{
    txtFname.setStyle(TXT_TEXTBOX);
    txtFname.setDataSource(RsName);
    txtFname.setDataField('FirstName');
    txtFname.disabled = true;
    txtFname.setMaxLength(20);
    txtFname.setColumnCount(17);
}
function _txtFname_ctor()
{
    CreateTextbox('txtFname', _inittxtFname, null);
}
</script>
<% txtFname.display %>

<!--METADATA TYPE="DesignerControl" endspan-->
</P>
</TD>
    <P align=center>
        <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtMName style="WIDTH: 108px; HEIGHT: 19px"
        classid=clsid:B5F0E469-DC5F-11D0-9846-
0000F8027CA0><PARAM NAME="_ExtentX" VALUE="2858"><PARAM NAME="_ExtentY"
VALUE="503"><PARAM NAME="id" VALUE="txtMName"><PARAM NAME="ControlType"
VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM NAME="DataSource"
VALUE="RsName"><PARAM NAME="DataField" VALUE="MiddleName"><PARAM
NAME="Enabled" VALUE="0"><PARAM NAME="Visible" VALUE="-1"><PARAM
NAME="MaxChars" VALUE="20"><PARAM NAME="DisplayWidth" VALUE="18"><PARAM
NAME="Platform" VALUE="0"><PARAM NAME="LocalPath" VALUE="../">

                                </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtMName()

```



```

{
    txtMName.setStyle(TXT_TEXTBOX);
    txtMName.setDataSource(RsName);
    txtMName.setDataField('MiddleName');
    txtMName.disabled = true;
    txtMName.setMaxLength(20);
    txtMName.setColumnCount(18);
}
function _txtMName_ctor()
{
    CreateTextbox('txtMName', _inittxtMName, null);
}
</script>
<% txtMName.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
</TD>
    <TD>
        <P align=center>
            <!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=txtDept style="WIDTH: 66px; HEIGHT: 19px"
            classid=clsid:B5F0E469-DC5F-11D0-9846-0000F8027CA0
width=66
            height=19><PARAM NAME="_ExtentX" VALUE="1746"><PARAM
NAME="_ExtentY" VALUE="503"><PARAM NAME="id" VALUE="txtDept"><PARAM
NAME="ControlType" VALUE="0"><PARAM NAME="Lines" VALUE="3"><PARAM
NAME="DataSource" VALUE="RsName"><PARAM NAME="DataField"
VALUE="Department"><PARAM NAME="Enabled" VALUE="0"><PARAM
NAME="Visible" VALUE="-1"><PARAM NAME="MaxChars" VALUE="20"><PARAM
NAME="DisplayWidth" VALUE="11"><PARAM NAME="Platform" VALUE="0"><PARAM
NAME="LocalPath" VALUE="..">

            </OBJECT>
-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _inittxtDept()
{
    txtDept.setStyle(TXT_TEXTBOX);
    txtDept.setDataSource(RsName);
    txtDept.setDataField('Department');
    txtDept.disabled = true;
    txtDept.setMaxLength(20);
    txtDept.setColumnCount(11);
}
function _txtDept_ctor()
{
    CreateTextbox('txtDept', _inittxtDept, null);
}
</script>
<% txtDept.display %>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
</TD>

```



```

1"><PARAM NAME="LastImage" VALUE="0"><PARAM NAME="Alignment"
VALUE="1"><PARAM NAME="LocalPath" VALUE="..">
  </OBJECT>
-->
<!--#INCLUDE FILE="../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/RNavBar.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initRecordsetNavbar1()
{
    RecordsetNavbar1.setAlignment(1);
    RecordsetNavbar1.setButtonStyles(170);
    RecordsetNavbar1.updateOnMove = false;
    RecordsetNavbar1.setDataSource(RsName);
    RecordsetNavbar1.getButton(0).value = ' |< ';
    RecordsetNavbar1.getButton(1).value = ' < ';
    RecordsetNavbar1.getButton(2).value = ' > ';
    RecordsetNavbar1.getButton(3).value = ' >| ';
}
function _RecordsetNavbar1_ctor()
{
    CreateRecordsetNavbar('RecordsetNavbar1', _initRecordsetNavbar1,
null);
}
</script>
<% RecordsetNavbar1.display %>

<!--METADATA TYPE="DesignerControl" ends-->

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

FF. RANK1.ASP

```

<!--
<%@ Language=VBScript %>
<%Option Explicit%>
<script id="DebugDirectives" runat="server" language="javascript">
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</script>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<form name="thisForm" METHOD="post">
<%
Dim RsScore
Dim objItem
Dim intYear
%>

```

```

<%
'Request.Form ("intYear")
intYear= Request.Form ("txtYear")
%>

<html>
<head>
<meta NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">

</head>
<body bgcolor="#ffffff">

<p align="center"><A href="index.html"><strong><FONT face=Verdana>Admin
Home</FONT>
</strong></A></p>
<p>

<b><FONT face=Verdana><font size="5">Chairman Copy for Teaching
Year <%Response.Write Request ("txtYear") %>&nbsp;&nbsp;&nbsp;</font> </FONT>
</b>
<p><FONT face=Verdana></FONT></p>

<p>
<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsScore style="LEFT: 0px; TOP: 0px"
classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TConn=\qsaward\q,TCDDBObject_Unmatched=\qSQL\Statement\q,TCDBo
bjectName_Unmatched=\qSELECT\sdbo.vnpSchieffelinRanking.*, \sdbo.vnpSchi
effelinRanking.Rank\sAS\sExpr1, \sdbo.tnpEmployee.LastName\sAS\sExpr2, \s
dbo.tnpEmployee.FirstName\sAS\sExpr3, \sdbo.tnpEmployee.MiddleName\sAS\s
Expr4\sFROM\sdbo.vnpSchieffelinRanking\sINNER\sJOIN\sdbo.tnpEmployee\sO
N\sdbo.vnpSchieffelinRanking.EmployeeID\s=\sdbo.tnpEmployee.EmployeeID\
sWHERE\s(dbo.vnpSchieffelinRanking.CalendarYear\s=\s?)\sORDER\sBY\sdbo.
vnpSchieffelinRanking.Rank\q,TCControlID_Unmatched=\qRsScore\q,TCPConn
=\qsaward\q,TCPDBObject=\qViews\q,TCPDBObjectName=\qvnpSchieffelinRan
king\s(dbo)\q,RCDDBObject=\qRCSQLStatement\q,TCSQLStatement_Unmatched=\q
SELECT\sdbo.vnpSchieffelinRanking.*, \sdbo.vnpSchieffelinRanking.Rank\sA
S\sExpr1, \sdbo.tnpEmployee.LastName\sAS\sExpr2, \sdbo.tnpEmployee.FirstN
ame\sAS\sExpr3, \sdbo.tnpEmployee.MiddleName\sAS\sExpr4\sFROM\sdbo.vnpSc
hieffelinRanking\sINNER\sJOIN\sdbo.tnpEmployee\sON\sdbo.vnpSchieffelinR
anking.EmployeeID\s=\sdbo.tnpEmployee.EmployeeID\sWHERE\s(dbo.vnpSchief
felinRanking.CalendarYear\s=\s?)\sORDER\sBY\sdbo.vnpSchieffelinRanking.
Rank\q,TCCursorType=\q3\s-\sStatic\q,TCCursorLocation=\q3\s-
\sUse\sclient-side\sursors\q,TCLockType=\q3\s-
\sOptimistic\q,TCCacheSize_Unmatched=\q10\q,TCommTimeout_Unmatched=\q1
0\q,CCPrepared=0,CCAllRecords=1,TCNRecords_Unmatched=\q10\q,TCODBCSynta
x_Unmatched=\q\q,TCHTargetPlatform=\q\q,TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q,TCTargetPlatform=\qInherit\sfrom\sPage\q,RCCache=\qRCBook
Page\q,CCOpen=0,GCPParameters=(Rows=1,Row1=(CType_Unmatched=\q?\q,CParNa
me_Unmatched=\qParam1\q,CDataType_Unmatched=\qInteger\q,CSize_Unmatched
=\q4\q,CReq=1,CValue_Unmatched=\qintYear\q))) "><PARAM NAME="LocalPath"
VALUE="..">

```

</OBJECT>

```

-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsScore()
{
    RsScore.setParameter(0,intYear);
}
function _initRsScore()
{
    RsScore.advise(RS_ONBEFOREOPEN, _setParametersRsScore);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;
    cmdTmp.CommandText = 'SELECT dbo.vnpSchieffelinRanking.*,
dbo.vnpSchieffelinRanking.Rank AS Expr1, dbo.tnpEmployee.LastName AS
Expr2, dbo.tnpEmployee.FirstName AS Expr3, dbo.tnpEmployee.MiddleName
AS Expr4 FROM dbo.vnpSchieffelinRanking INNER JOIN dbo.tnpEmployee ON
dbo.vnpSchieffelinRanking.EmployeeID = dbo.tnpEmployee.EmployeeID WHERE
(dbo.vnpSchieffelinRanking.CalendarYear = ?) ORDER BY
dbo.vnpSchieffelinRanking.Rank';
    rsTmp.CacheSize = 10;
    rsTmp.CursorType = 3;
    rsTmp.CursorLocation = 3;
    rsTmp.LockType = 3;
    RsScore.setRecordSource(rsTmp);
    if (thisPage.getState('pb_RsScore') != null)
        RsScore.setBookmark(thisPage.getState('pb_RsScore'));
}
function _RsScore_ctor()
{
    CreateRecordset('RsScore', _initRsScore, null);
}
function _RsScore_dtor()
{
    RsScore._preserveState();
    thisPage.setState('pb_RsScore', RsScore.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" endspan-->
</p>
<%RsScore.open%>

<!--METADATA TYPE="DesignerControl" startspan

```

```

<OBJECT id=Grid1 style="LEFT: 0px; WIDTH: 748px; TOP: 0px; HEIGHT:
147px"
classid=clsid:277FC3F2-E90F-11D0-B767-0000F81E081D width=748
height=147><PARAM NAME="_ExtentX" VALUE="19791"><PARAM NAME="_ExtentY"
VALUE="3889"><PARAM NAME="DataConnection" VALUE=""><PARAM
NAME="SourceType" VALUE=""><PARAM NAME="Recordset"
VALUE="RsScore"><PARAM NAME="CtrlName" VALUE="Grid1"><PARAM
NAME="UseAdvancedOnly" VALUE="0"><PARAM NAME="AdvAddToStyles" VALUE="-
1"><PARAM NAME="AdvTableTag" VALUE=""><PARAM NAME="AdvHeaderRowTag"
VALUE=""><PARAM NAME="AdvHeaderCellTag" VALUE=""><PARAM
NAME="AdvDetailRowTag" VALUE=""><PARAM NAME="AdvDetailCellTag"
VALUE=""><PARAM NAME="ScriptLanguage" VALUE="1"><PARAM
NAME="ScriptingPlatform" VALUE="0"><PARAM NAME="EnableRowNav"
VALUE="0"><PARAM NAME="HiliteColor" VALUE=""><PARAM
NAME="RecNavBarHasNextButton" VALUE="-1"><PARAM
NAME="RecNavBarHasPrevButton" VALUE="-1"><PARAM
NAME="RecNavBarNextText" VALUE="    >    "><PARAM
NAME="RecNavBarPrevText" VALUE="    <    "><PARAM NAME="ColumnsNames"
VALUE='"Rank","Expr2","Expr3","Expr4","FinalScore","NBallots","NumberSu
pportStatements","FirstChoiceVotes","SecondChoiceVotes","ThirdChoiceVot
es","FirstChoiceScore","SecondChoiceScore","ThirdChoiceScore","Selected
OnBallotScore"'><PARAM NAME="columnIndex"
VALUE="0,1,2,3,4,5,6,7,8,9,10,11,12,13"><PARAM NAME="displayWidth"
VALUE="68,68,68,68,68,68,68,68,68,68,68,68,68,68"><PARAM NAME="Coltype"
VALUE="1,1,1,1,1,1,1,1,1,1,1,1,1,1"><PARAM NAME="formatted"
VALUE="1,0,0,0,0,0,0,0,0,0,0,0,0,0"><PARAM NAME="DisplayName"
VALUE='"Rank","LName","FName","MName","Score","N","D","X1","X2","X3","Z
1","Z2","Z3","Z4"'><PARAM NAME="DetailAlignment"
VALUE="0,,,,,,,,,,,,,"><PARAM NAME="HeaderAlignment"
VALUE="0,,,,,,,,,,,,,"><PARAM NAME="DetailBackColor"
VALUE="White,,,,,,,,,,,,,"><PARAM NAME="HeaderBackColor"
VALUE="White,,,,,,,,,,,,,"><PARAM NAME="HeaderFont"
VALUE="Arial,,,,,,,,,,,,,"><PARAM NAME="HeaderFontColor"
VALUE="Black,,,,,,,,,,,,,"><PARAM NAME="HeaderFontSize"
VALUE="3,,,,,,,,,,,,,"><PARAM NAME="HeaderFontStyle"
VALUE="1,,,,,,,,,,,,,"><PARAM NAME="DetailFont"
VALUE="Arial,,,,,,,,,,,,,"><PARAM NAME="DetailFontColor"
VALUE="Black,,,,,,,,,,,,,"><PARAM NAME="DetailFontSize"
VALUE="2,,,,,,,,,,,,,"><PARAM NAME="DetailFontStyle"
VALUE="0,,,,,,,,,,,,,"><PARAM NAME="ColumnCount" VALUE="14"><PARAM
NAME="CurStyle" VALUE="Simple List"><PARAM NAME="TitleFont"
VALUE="Arial"><PARAM NAME="titleFontSize" VALUE="3"><PARAM
NAME="TitleFontColor" VALUE="0"><PARAM NAME="TitleBackColor"
VALUE="16777215"><PARAM NAME="TitleFontStyle" VALUE="1"><PARAM
NAME="TitleAlignment" VALUE="0"><PARAM NAME="RowFont"
VALUE="Arial"><PARAM NAME="RowFontColor" VALUE="0"><PARAM
NAME="RowFontStyle" VALUE="0"><PARAM NAME="RowFontSize"
VALUE="2"><PARAM NAME="RowBackColor" VALUE="16777215"><PARAM
NAME="RowAlignment" VALUE="0"><PARAM NAME="HighlightColor3D"
VALUE="268435455"><PARAM NAME="ShadowColor3D" VALUE="268435455"><PARAM
NAME="PageSize" VALUE="60"><PARAM NAME="MoveFirstCaption" VALUE="    |<
"><PARAM NAME="MoveLastCaption" VALUE="    >|    "><PARAM
NAME="MovePrevCaption" VALUE="    <<    "><PARAM NAME="MoveNextCaption"
VALUE="    >>    "><PARAM NAME="BorderSize" VALUE="2"><PARAM
NAME="BorderColor" VALUE="8421504"><PARAM NAME="GridBackColor"
VALUE="16777215"><PARAM NAME="AltRowBckgnd" VALUE="268435455"><PARAM

```

```

NAME="CellSpacing" VALUE="1"><PARAM NAME="WidthSelectionMode"
VALUE="1"><PARAM NAME="GridWidth" VALUE="748"><PARAM
NAME="EnablePaging" VALUE="-1"><PARAM NAME="ShowStatus" VALUE="-
1"><PARAM NAME="StyleValue" VALUE="453577"><PARAM NAME="LocalPath"
VALUE="..">

```

```

</OBJECT>
-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/RNavBar.ASP"-->
<!--#INCLUDE FILE="../../../_ScriptLibrary/DataGrid.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initGrid1()
{
Grid1.pageSize = 60;
Grid1.setDataSource(RsScore);
Grid1.tableAttributes = ' cellpadding=2 cellspacing=1 bordercolor=Gray
bgcolor=White border=2 cols=14 rules=ROWS WIDTH=748';
Grid1.headerAttributes = ' bgcolor=White align=Left';
Grid1.headerWidth[0] = ' WIDTH=68';
Grid1.headerWidth[1] = ' WIDTH=68';
Grid1.headerWidth[2] = ' WIDTH=68';
Grid1.headerWidth[3] = ' WIDTH=68';
Grid1.headerWidth[4] = ' WIDTH=68';
Grid1.headerWidth[5] = ' WIDTH=68';
Grid1.headerWidth[6] = ' WIDTH=68';
Grid1.headerWidth[7] = ' WIDTH=68';
Grid1.headerWidth[8] = ' WIDTH=68';
Grid1.headerWidth[9] = ' WIDTH=68';
Grid1.headerWidth[10] = ' WIDTH=68';
Grid1.headerWidth[11] = ' WIDTH=68';
Grid1.headerWidth[12] = ' WIDTH=68';
Grid1.headerWidth[13] = ' WIDTH=68';
Grid1.headerFormat = '<Font face="Arial" size=3 color=Black> <b>';
Grid1.colHeader[0] = '\Rank\';
Grid1.colHeader[1] = '\LName\';
Grid1.colHeader[2] = '\FName\';
Grid1.colHeader[3] = '\MName\';
Grid1.colHeader[4] = '\Score\';
Grid1.colHeader[5] = '\N\';
Grid1.colHeader[6] = '\D\';
Grid1.colHeader[7] = '\X1\';
Grid1.colHeader[8] = '\X2\';
Grid1.colHeader[9] = '\X3\';
Grid1.colHeader[10] = '\Z1\';
Grid1.colHeader[11] = '\Z2\';
Grid1.colHeader[12] = '\Z3\';
Grid1.colHeader[13] = '\Z4\';
Grid1.rowAttributes[0] = ' bgcolor = White align=Left
bordercolor=Gray';
Grid1.rowFormat[0] = ' <Font face="Arial" size=2 color=Black >';

```

```

Grid1.colAttributes[0] = ' WIDTH=68 bgcolor=White align=Left';
Grid1.colFormat[0] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[0] = 'RsScore.fields.getValue(\'Rank\')';
Grid1.colAttributes[1] = ' WIDTH=68';
Grid1.colFormat[1] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[1] = 'RsScore.fields.getValue(\'Expr2\')';
Grid1.colAttributes[2] = ' WIDTH=68';
Grid1.colFormat[2] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[2] = 'RsScore.fields.getValue(\'Expr3\')';
Grid1.colAttributes[3] = ' WIDTH=68';
Grid1.colFormat[3] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[3] = 'RsScore.fields.getValue(\'Expr4\')';
Grid1.colAttributes[4] = ' WIDTH=68';
Grid1.colFormat[4] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[4] = 'RsScore.fields.getValue(\'FinalScore\')';
Grid1.colAttributes[5] = ' WIDTH=68';
Grid1.colFormat[5] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[5] = 'RsScore.fields.getValue(\'NBallots\')';
Grid1.colAttributes[6] = ' WIDTH=68';
Grid1.colFormat[6] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[6] =
'RsScore.fields.getValue(\'NumberSupportStatements\')';
Grid1.colAttributes[7] = ' WIDTH=68';
Grid1.colFormat[7] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[7] = 'RsScore.fields.getValue(\'FirstChoiceVotes\')';
Grid1.colAttributes[8] = ' WIDTH=68';
Grid1.colFormat[8] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[8] = 'RsScore.fields.getValue(\'SecondChoiceVotes\')';
Grid1.colAttributes[9] = ' WIDTH=68';
Grid1.colFormat[9] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[9] = 'RsScore.fields.getValue(\'ThirdChoiceVotes\')';
Grid1.colAttributes[10] = ' WIDTH=68';
Grid1.colFormat[10] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[10] = 'RsScore.fields.getValue(\'FirstChoiceScore\')';
Grid1.colAttributes[11] = ' WIDTH=68';
Grid1.colFormat[11] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[11] = 'RsScore.fields.getValue(\'SecondChoiceScore\')';
Grid1.colAttributes[12] = ' WIDTH=68';
Grid1.colFormat[12] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[12] = 'RsScore.fields.getValue(\'ThirdChoiceScore\')';
Grid1.colAttributes[13] = ' WIDTH=68';
Grid1.colFormat[13] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[13] =
'RsScore.fields.getValue(\'SelectedOnBallotScore\')';
Grid1.navbarAlignment = 'Right';
var objPageNavbar = Grid1.showPageNavbar(170,1);
objPageNavbar.getButton(0).value = ' |< ';
objPageNavbar.getButton(1).value = ' << ';
objPageNavbar.getButton(2).value = ' >> ';
objPageNavbar.getButton(3).value = ' >| ';
Grid1.hasPageNumber = true;
}
function _Grid1_ctor()
{
    CreateDataGrid('Grid1',_initGrid1);
}

```



```

</SCRIPT>

<%      Grid1.display %>

<!--METADATA TYPE="DesignerControl" endspan-->
<P></P>
<p>&nbsp;</p>
<p>
<input type= hidden name=txtYear value="intYear">
</p>
<p>&nbsp;</p>
<p>&nbsp;</p>

</body>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</form>
</html>
-->

```

GG. RANK2.ASP

```

<!--
<%@ Language=VBScript %>
<%Option Explicit%>
<SCRIPT id=DebugDirectives runat=server language=javascript>
// Set these to true to enable debugging or tracing
@set @debug=false
@set @trace=false
</SCRIPT>

<% ' VI 6.0 Scripting Object Model Enabled %>
<!--#include file="../_ScriptLibrary/pm.asp"-->
<% if StartPageProcessing() Then Response.End() %>
<FORM name=thisForm METHOD=post>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY bgcolor="#ffffff">
<P align=center><A href="index.html"><STRONG><FONT face=Verdana>Admin
Home</FONT> </STRONG></A></P>
<b><font size="5"><FONT face=Verdana>Commitee Copy for Teaching Year
<%Response.Write Request("txtYear")%>&nbsp;&nbsp;</font>
</b>
<%
Dim RsScore
Dim objItem
Dim intYear
'for each objItem in Request.Form
'Response.Write objItem & "=" & Request.Form (objItem)& "<BR>"
'next
%>

<%

```

```

'Request.Form ("intYear")
intYear= Request.Form ("txtYear")
%>

<P>
<P>
<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=RsScore style="LEFT: 0px; TOP: 0px"
classid=clsid:9CF5D7C2-EC10-11D0-9862-0000F8027CA0><PARAM
NAME="ExtentX" VALUE="12197"><PARAM NAME="ExtentY" VALUE="2090"><PARAM
NAME="State"
VALUE="(TCCConn=\qsaward\q, TCDBObject_Unmatched=\qSQL\sStatement\q, TCDBO
bjectName_Unmatched=\qSELECT\sdbo.vnpSchieffelinRanking.*, \sRank\sAS\sE
xpr1\sFROM\sdbo.vnpSchieffelinRanking\sWHERE\s(CalendarYear\s=\s?)\sORD
ER\sBY\sRank\q, TCControlID_Unmatched=\qRsScore\q, TCPPConn=\qsaward\q, TC
PPDBObject=\qViews\q, TCPPDBObjectName=\qvnpSchieffelinRanking\s(dbo)\q,
RCDBObject=\qRCSQLStatement\q, TCSQLStatement_Unmatched=\qSELECT\sdbo.vn
pSchieffelinRanking.*, \sRank\sAS\sExpr1\sFROM\sdbo.vnpSchieffelinRankin
g\sWHERE\s(CalendarYear\s=\s?)\sORDER\sBY\sRank\q, TCCursorType=\q3\s-
\sStatic\q, TCCursorLocation=\q3\s-\sUse\sclient-
side\sursors\q, TCLockType=\q3\s-
\sOptimistic\q, TCCacheSize_Unmatched=\q10\q, TCCommTimeout_Unmatched=\q1
0\q, CCPrepared=0, CCAAllRecords=1, TCNRecords_Unmatched=\q10\q, TCODBCSynta
x_Unmatched=\q\q, TCHTargetPlatform=\q\q, TCHTargetBrowser_Unmatched=\qSe
rver\s(ASP)\q, TCTargetPlatform=\qInherit\sfrom\sPage\q, RCCache=\qRCBook
Page\q, CCOpen=0, GCParameters=(Rows=1, Row1=(CType_Unmatched=\q?\q, CParNa
me_Unmatched=\qParam1\q, CDataType_Unmatched=\qInteger\q, CSize_Unmatched
=\q4\q, CReq=1, CValue_Unmatched=\qintYear\q)) "><PARAM NAME="LocalPath"
VALUE="../">

</OBJECT>

-->
<!--#INCLUDE FILE="../_ScriptLibrary/Recordset.ASP"-->
<SCRIPT LANGUAGE="JavaScript" RUNAT="server">
function _setParametersRsScore()
{
    RsScore.setParameter(0,intYear);
}
function _initRsScore()
{
    RsScore.advise(RS_ONBEFOREOPEN, _setParametersRsScore);
    var DBConn = Server.CreateObject('ADODB.Connection');
    DBConn.ConnectionTimeout =
Application('saward_ConnectionTimeout');
    DBConn.CommandTimeout = Application('saward_CommandTimeout');
    DBConn.CursorLocation = Application('saward_CursorLocation');
    DBConn.Open(Application('saward_ConnectionString'),
Application('saward_RuntimeUserName'),
Application('saward_RuntimePassword'));
    var cmdTmp = Server.CreateObject('ADODB.Command');
    var rsTmp = Server.CreateObject('ADODB.Recordset');
    cmdTmp.ActiveConnection = DBConn;
    rsTmp.Source = cmdTmp;
    cmdTmp.CommandType = 1;
    cmdTmp.CommandTimeout = 10;

```

```

        cmdTmp.CommandText = 'SELECT dbo.vnpSchieffelinRanking.*, Rank AS
Expr1 FROM dbo.vnpSchieffelinRanking WHERE (CalendarYear = ?) ORDER BY
Rank';
        rsTmp.CacheSize = 10;
        rsTmp.CursorType = 3;
        rsTmp.CursorLocation = 3;
        rsTmp.LockType = 3;
        RsScore.setRecordSource(rsTmp);
        if (thisPage.getState('pb_RsScore') != null)
            RsScore.setBookmark(thisPage.getState('pb_RsScore'));
    }
function _RsScore_ctor()
{
    CreateRecordset('RsScore', _initRsScore, null);
}
function _RsScore_dtor()
{
    RsScore._preserveState();
    thisPage.setState('pb_RsScore', RsScore.getBookmark());
}
</SCRIPT>

<!--METADATA TYPE="DesignerControl" ends-->
</P>
<%RsScore.open%>

<!--METADATA TYPE="DesignerControl" startspan
<OBJECT id=Grid1 style="LEFT: 0px; WIDTH: 544px; TOP: 0px; HEIGHT:
147px" classid="clsid:277FC3F2-E90F-11D0-B767-0000F81E081D"
width=544 height=147>
    <PARAM NAME="_ExtentX" VALUE="14393">
    <PARAM NAME="_ExtentY" VALUE="3889">
    <PARAM NAME="DataConnection" VALUE="">
    <PARAM NAME="SourceType" VALUE="">
    <PARAM NAME="Recordset" VALUE="RsScore">
    <PARAM NAME="CtrlName" VALUE="Grid1">
    <PARAM NAME="UseAdvancedOnly" VALUE="0">
    <PARAM NAME="AdvAddToStyles" VALUE="-1">
    <PARAM NAME="AdvTableTag" VALUE="">
    <PARAM NAME="AdvHeaderRowTag" VALUE="">
    <PARAM NAME="AdvHeaderCellTag" VALUE="">
    <PARAM NAME="AdvDetailRowTag" VALUE="">
    <PARAM NAME="AdvDetailCellTag" VALUE="">
    <PARAM NAME="ScriptLanguage" VALUE="1">
    <PARAM NAME="ScriptingPlatform" VALUE="0">
    <PARAM NAME="EnableRowNav" VALUE="0">
    <PARAM NAME="HiliteColor" VALUE="">
    <PARAM NAME="RecNavBarHasNextButton" VALUE="-1">
    <PARAM NAME="RecNavBarHasPrevButton" VALUE="-1">
    <PARAM NAME="RecNavBarNextText" VALUE="    >    ">
    <PARAM NAME="RecNavBarPrevText" VALUE="    <    ">
    <PARAM NAME="ColumnsNames"
VALUE=' "Rank", "FinalScore", "NBallots", "NumberSupportStatements", "FirstC
hoiceVotes", "SecondChoiceVotes", "ThirdChoiceVotes", "FirstChoiceScore", "
SecondChoiceScore", "ThirdChoiceScore", "SelectedOnBallotScore"'>
    <PARAM NAME="columnIndex" VALUE="0,1,2,3,4,5,6,7,8,9,10">

```

```

        <PARAM NAME="displayWidth"
VALUE="68,68,68,68,68,68,68,68,68,68,68">
        <PARAM NAME="Coltype" VALUE="1,1,1,1,1,1,1,1,1,1,1">
        <PARAM NAME="formatted" VALUE="1,0,0,0,0,0,0,0,0,0,0">
        <PARAM NAME="DisplayName"
VALUE="'Rank","Score","N","D","X1","X2","X3","Z1","Z2","Z3","Z4" '>
        <PARAM NAME="DetailAlignment" VALUE="2,,,,,,,,,">
        <PARAM NAME="HeaderAlignment" VALUE="0,,,,,,,,,">
        <PARAM NAME="DetailBackColor" VALUE="White,,,,,,,,,">
        <PARAM NAME="HeaderBackColor" VALUE="#e3e4cd,,,,,,,,,">
        <PARAM NAME="HeaderFont" VALUE="Arial,,,,,,,,,">
        <PARAM NAME="HeaderFontColor" VALUE="Black,,,,,,,,,">
        <PARAM NAME="HeaderFontSize" VALUE="3,,,,,,,,,">
        <PARAM NAME="HeaderFontStyle" VALUE="1,,,,,,,,,">
        <PARAM NAME="DetailFont" VALUE="Arial,,,,,,,,,">
        <PARAM NAME="DetailFontColor" VALUE="Black,,,,,,,,,">
        <PARAM NAME="DetailFontSize" VALUE="2,,,,,,,,,">
        <PARAM NAME="DetailFontStyle" VALUE="0,,,,,,,,,">
        <PARAM NAME="ColumnCount" VALUE="11">
        <PARAM NAME="CurStyle" VALUE="Simple List">
        <PARAM NAME="TitleFont" VALUE="Arial">
        <PARAM NAME="titleFontSize" VALUE="3">
        <PARAM NAME="TitleFontColor" VALUE="0">
        <PARAM NAME="TitleBackColor" VALUE="16777215">
        <PARAM NAME="TitleFontStyle" VALUE="1">
        <PARAM NAME="TitleAlignment" VALUE="0">
        <PARAM NAME="RowFont" VALUE="Arial">
        <PARAM NAME="RowFontColor" VALUE="0">
        <PARAM NAME="RowFontStyle" VALUE="0">
        <PARAM NAME="RowFontSize" VALUE="2">
        <PARAM NAME="RowBackColor" VALUE="16777215">
        <PARAM NAME="RowAlignment" VALUE="0">
        <PARAM NAME="HighlightColor3D" VALUE="268435455">
        <PARAM NAME="ShadowColor3D" VALUE="268435455">
        <PARAM NAME="PageSize" VALUE="60">
        <PARAM NAME="MoveFirstCaption" VALUE="      |<      ">
        <PARAM NAME="MoveLastCaption" VALUE="      >|      ">
        <PARAM NAME="MovePrevCaption" VALUE="      <<      ">
        <PARAM NAME="MoveNextCaption" VALUE="      >>      ">
        <PARAM NAME="BorderSize" VALUE="2">
        <PARAM NAME="BorderColor" VALUE="8421504">
        <PARAM NAME="GridBackColor" VALUE="16777215">
        <PARAM NAME="AltRowBckgnd" VALUE="268435455">
        <PARAM NAME="CellSpacing" VALUE="1">
        <PARAM NAME="WidthSelectionMode" VALUE="1">
        <PARAM NAME="GridWidth" VALUE="544">
        <PARAM NAME="EnablePaging" VALUE="-1">
        <PARAM NAME="ShowStatus" VALUE="-1">
        <PARAM NAME="StyleValue" VALUE="453577">
        <PARAM NAME="LocalPath" VALUE=".."></OBJECT>
-->
<!--#INCLUDE FILE="../_ScriptLibrary/Button.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/RNavBar.ASP"-->
<!--#INCLUDE FILE="../_ScriptLibrary/DataGrid.ASP"-->
<SCRIPT LANGUAGE=JavaScript RUNAT=Server>
function _initGrid1()

```

```

{
Grid1.pageSize = 60;
Grid1.setDataSource(RsScore);
Grid1.tableAttributes = ' cellpadding=2 cellspacing=1 bordercolor=Gray
bgcolor=White border=2 cols=11 rules=ROWS WIDTH=544';
Grid1.headerAttributes = ' bgcolor=White align=Left';
Grid1.headerWidth[0] = ' WIDTH=68';
Grid1.headerWidth[1] = ' WIDTH=68';
Grid1.headerWidth[2] = ' WIDTH=68';
Grid1.headerWidth[3] = ' WIDTH=68';
Grid1.headerWidth[4] = ' WIDTH=68';
Grid1.headerWidth[5] = ' WIDTH=68';
Grid1.headerWidth[6] = ' WIDTH=68';
Grid1.headerWidth[7] = ' WIDTH=68';
Grid1.headerWidth[8] = ' WIDTH=68';
Grid1.headerWidth[9] = ' WIDTH=68';
Grid1.headerWidth[10] = ' WIDTH=68';
Grid1.headerFormat = '<Font face="Arial" size=3 color=Black> <b>';
Grid1.colHeader[0] = '\Rank\';
Grid1.colHeader[1] = '\Score\';
Grid1.colHeader[2] = '\N\';
Grid1.colHeader[3] = '\D\';
Grid1.colHeader[4] = '\X1\';
Grid1.colHeader[5] = '\X2\';
Grid1.colHeader[6] = '\X3\';
Grid1.colHeader[7] = '\Z1\';
Grid1.colHeader[8] = '\Z2\';
Grid1.colHeader[9] = '\Z3\';
Grid1.colHeader[10] = '\Z4\';
Grid1.rowAttributes[0] = ' bgcolor = White align=Left
bordercolor=Gray';
Grid1.rowFormat[0] = ' <Font face="Arial" size=2 color=Black >';
Grid1.colAttributes[0] = ' WIDTH=68 bgcolor=White align=Center';
Grid1.colFormat[0] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[0] = 'RsScore.fields.getValue('\Rank\')';
Grid1.colAttributes[1] = ' WIDTH=68';
Grid1.colFormat[1] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[1] = 'RsScore.fields.getValue('\FinalScore\')';
Grid1.colAttributes[2] = ' WIDTH=68';
Grid1.colFormat[2] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[2] = 'RsScore.fields.getValue('\NBallots\')';
Grid1.colAttributes[3] = ' WIDTH=68';
Grid1.colFormat[3] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[3] =
'RsScore.fields.getValue('\NumberSupportStatements\')';
Grid1.colAttributes[4] = ' WIDTH=68';
Grid1.colFormat[4] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[4] = 'RsScore.fields.getValue('\FirstChoiceVotes\')';
Grid1.colAttributes[5] = ' WIDTH=68';
Grid1.colFormat[5] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[5] = 'RsScore.fields.getValue('\SecondChoiceVotes\')';
Grid1.colAttributes[6] = ' WIDTH=68';
Grid1.colFormat[6] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[6] = 'RsScore.fields.getValue('\ThirdChoiceVotes\')';
Grid1.colAttributes[7] = ' WIDTH=68';
Grid1.colFormat[7] = '<Font Size=2 Face="Arial" Color=Black >';

```

```

Grid1.colData[7] = 'RsScore.fields.getValue(\'FirstChoiceScore\')';
Grid1.colAttributes[8] = ' WIDTH=68';
Grid1.colFormat[8] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[8] = 'RsScore.fields.getValue(\'SecondChoiceScore\')';
Grid1.colAttributes[9] = ' WIDTH=68';
Grid1.colFormat[9] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[9] = 'RsScore.fields.getValue(\'ThirdChoiceScore\')';
Grid1.colAttributes[10] = ' WIDTH=68';
Grid1.colFormat[10] = '<Font Size=2 Face="Arial" Color=Black >';
Grid1.colData[10] =
'RsScore.fields.getValue(\'SelectedOnBallotScore\')';
Grid1.navbarAlignment = 'Right';
var objPageNavbar = Grid1.showPageNavbar(170,1);
objPageNavbar.getButton(0).value = ' |< ';
objPageNavbar.getButton(1).value = ' << ';
objPageNavbar.getButton(2).value = ' >> ';
objPageNavbar.getButton(3).value = ' >| ';
Grid1.hasPageNumber = true;
}
function _Grid1_ctor()
{
    CreateDataGrid('Grid1',_initGrid1);
}
</SCRIPT>

<% Grid1.display %>

<!--METADATA TYPE="DesignerControl" ends-->
<P></P>
</BODY>
<% ' VI 6.0 Scripting Object Model Enabled %>
<% EndPageProcessing() %>
</FORM>
</HTML>
-->

```

HH. UPDATE.ASP

```

<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
<!-- Fireworks 4.0 Dreamweaver 4.0 target. Created Fri Dec 08
09:16:10 GMT-0800 (Pacific Standard Time) 2000-->
<script language="JavaScript">
<!--
function MM_displayStatusMsg(msgStr) { //v3.0
    status=msgStr; document.MM_returnValue = true;
}

function MM_findObj(n, d) { //v3.0

```

```

    var p,i,x;  if(!d) d=document;
    if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++)
    x=d.forms[i][n];
    for(i=0;!x&&d.layers&&i<d.layers.length;i++)
    x=MM_findObj(n,d.layers[i].document); return x;
}
function MM_nbGroup(event, grpName) { //v3.0
    var i,img,nbArr,args=MM_nbGroup.arguments;
    if (event == "init" && args.length > 2) {
        if ((img = MM_findObj(args[2])) != null && !img.MM_init) {
            img.MM_init = true; img.MM_up = args[3]; img.MM_dn = img.src;
            if ((nbArr = document[grpName]) == null) nbArr =
document[grpName] = new Array();
            nbArr[nbArr.length] = img;
            for (i=4; i < args.length-1; i+=2) if ((img =
MM_findObj(args[i])) != null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = img.MM_dn = args[i+1];
                nbArr[nbArr.length] = img;
            } }
        } else if (event == "over") {
            document.MM_nbOver = nbArr = new Array();
            for (i=1; i < args.length-1; i+=3) if ((img = MM_findObj(args[i]))
!= null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = (img.MM_dn && args[i+2]) ? args[i+2] : args[i+1];
                nbArr[nbArr.length] = img;
            }
        } else if (event == "out" ) {
            for (i=0; i < document.MM_nbOver.length; i++) {
                img = document.MM_nbOver[i]; img.src = (img.MM_dn) ? img.MM_dn :
img.MM_up; }
        } else if (event == "down") {
            if ((nbArr = document[grpName]) != null)
                for (i=0; i < nbArr.length; i++) { img=nbArr[i]; img.src =
img.MM_up; img.MM_dn = 0; }
            document[grpName] = nbArr = new Array();
            for (i=2; i < args.length-1; i+=2) if ((img = MM_findObj(args[i]))
!= null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = img.MM_dn = args[i+1];
                nbArr[nbArr.length] = img;
            } }
        }
    }
}

function MM_preloadImages() { //v3.0
    var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
        var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0;
i<a.length; i++)
            if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image;
d.MM_p[j++].src=a[i];}}
}

//-->

```

```

</script>
</head>
<body bgcolor="#ffffff"
onLoad="MM_preloadImages('images/Academics_f2.gif','images/Academics_f4
.gif','images/Academics_f3.gif','images/About_f2.gif','images/About_f4.
.gif','images/About_f3.gif','images/Research_f3.gif','images/Research_f2
.gif','images/Research_f4.gif','images/ExecutiveEducation_f3.gif','imag
es/ExecutiveEducation_f2.gif','images/ExecutiveEducation_f4.gif','image
s/Students_f3.gif','images/Students_f2.gif','images/Students_f4.gif','i
images/Faculty_f3.gif','images/Faculty_f2.gif','images/Faculty_f4.gif','i
images/Administration_f3.gif','images/Administration_f2.gif','images/Ad
ministration_f4.gif','images/Alumni_f3.gif','images/Alumni_f2.gif','ima
ges/Alumni_f4.gif','images/Library_f3.gif','images/Library_f2.gif','ima
ges/Library_f4.gif','images/Search_f3.gif','images/Search_f2.gif','imag
es/Search_f4.gif','images/About_f3.gif','images/About_f2.gif','images/A
bout_f4.gif','images/Disclaimers_f3.gif','images/Disclaimers_f2.gif','i
images/Disclaimers_f4.gif','images/ExecutiveEducation_f3.gif','images/Ex
ecutiveEducation_f2.gif','images/ExecutiveEducation_f4.gif')"
leftmargin="0" topmargin="0" marginwidth="0" marginheight="0">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1"><!-- fwtable fwsrc="T01Academics.png"
fwbase="01Academics.gif" fwstyle="Dreamweaver" fwdocid = "742308039"
fwnested="0" -->
  <tr>
    <td width="192" height="3"><IMG height=1 src="spacer.gif" width=160
border=0></td>
    <td width="602" height="3"><IMG height=1 src="spacer.gif" width=600
border=0></td>
    <td width="3" height="3"><IMG height=1 src="spacer.gif" width=1
border=0></td>
  </tr>
  <tr>
    <td valign="top" align="left" height="1" bordercolor="#e3e4cd"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
  Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;<b><A href="index.html"><FONT
  face=Verdana>Admin Home</FONT> </a></b></p>
<p align="left"><b><FONT
  face=Verdana>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</FONT> <A href="weights.asp"><FONT
face=Verdana>View
  Weights</FONT> </a></b></p>
  <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&nbsp;</p>
  <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
      <table width="583" border="0" align="center">

```



```

fp_fTableFormat=True
fp_fMenuFormat=False
fp_sMenuChoice=""
fp_sMenuValue=""
fp_iDisplayCols=16
fp_fCustomQuery=True
BOTID=0
fp_iRegion=BOTID
%>
<!--#include file="../../../_fpclass/fpdbrgn1.inc"--><!--webbot
bot="DatabaseRegionStart" i-checksum="36288" endspace --><!--webbot
bot="DatabaseRegionEnd" startspan b-tableformat="TRUE" b-
menuformat="FALSE" u-dbrgn2="../../../_fpclass/fpdbrgn2.inc" i-groupsize="0"
clientside tag="TBODY" local_preview="&lt;tr&gt;&lt;td colspan=64
bgcolor=&quot;#FFFF00&quot;; align=&quot;left&quot;;
width=&quot;100%&quot;;&gt;&lt;font color=&quot;#000000&quot;;&gt;This is
the end of a Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" preview="&lt;tr&gt;&lt;td
colspan=64 bgcolor=&quot;#FFFF00&quot;; align=&quot;left&quot;;
width=&quot;100%&quot;;&gt;&lt;font color=&quot;#000000&quot;;&gt;This is
the end of a Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" --><!--#include
file="../../../_fpclass/fpdbrgn2.inc"--><!--webbot bot="DatabaseRegionEnd" i-
checksum="56926" endspace -->
        </tbody>
    </table>

    <p>&nbsp;</p>
    <p>&nbsp;</p>
    <p>&nbsp;</p>
    <p>&nbsp;</p>

    </td>
</tr>
<tr>
    <td width="575">
    </td>
</tr>
</table>
</td>
    <td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

II. UPDATE_WEIGHTS.ASP

```

<!--

<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">

```

```

<!-- Fireworks 4.0 Dreamweaver 4.0 target. Created Fri Dec 08
09:16:10 GMT-0800 (Pacific Standard Time) 2000-->
<script language="JavaScript">
<!--
function MM_displayStatusMsg(msgStr) { //v3.0
    status=msgStr; document.MM_returnValue = true;
}

function MM_findObj(n, d) { //v3.0
    var p,i,x;  if(!d) d=document;
    if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++)
x=d.forms[i][n];
    for(i=0;!x&&d.layers&&i<d.layers.length;i++)
x=MM_findObj(n,d.layers[i].document); return x;
}

function MM_nbGroup(event, grpName) { //v3.0
    var i,img,nbArr,args=MM_nbGroup.arguments;
    if (event == "init" && args.length > 2) {
        if ((img = MM_findObj(args[2])) != null && !img.MM_init) {
            img.MM_init = true; img.MM_up = args[3]; img.MM_dn = img.src;
            if ((nbArr = document[grpName]) == null) nbArr =
document[grpName] = new Array();
            nbArr[nbArr.length] = img;
            for (i=4; i < args.length-1; i+=2) if ((img =
MM_findObj(args[i])) != null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = img.MM_dn = args[i+1];
                nbArr[nbArr.length] = img;
            }
        }
        else if (event == "over") {
            document.MM_nbOver = nbArr = new Array();
            for (i=1; i < args.length-1; i+=3) if ((img = MM_findObj(args[i]))
!= null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = (img.MM_dn && args[i+2]) ? args[i+2] : args[i+1];
                nbArr[nbArr.length] = img;
            }
        }
        else if (event == "out" ) {
            for (i=0; i < document.MM_nbOver.length; i++) {
                img = document.MM_nbOver[i]; img.src = (img.MM_dn) ? img.MM_dn :
img.MM_up; }
        }
        else if (event == "down") {
            if ((nbArr = document[grpName]) != null)
                for (i=0; i < nbArr.length; i++) { img=nbArr[i]; img.src =
img.MM_up; img.MM_dn = 0; }
            document[grpName] = nbArr = new Array();
            for (i=2; i < args.length-1; i+=2) if ((img = MM_findObj(args[i]))
!= null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = img.MM_dn = args[i+1];
                nbArr[nbArr.length] = img;
            }
        }
    }
}

```

```

function MM_preloadImages() { //v3.0
    var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
        var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0;
i<a.length; i++)
            if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image;
d.MM_p[j++].src=a[i];}}
    }

    //-->
</script>
</head>
<body bgcolor="#ffffff">
<table border="0" cellpadding="0" cellspacing="0" width="803"
height="1">
    <tr>
        <td width="192" height="3"></td>
        <td width="602" height="3"></td>
        <td width="3" height="3"></td>
    </tr>
    <tr>
        <td valign="top" align="left" height="1" bordercolor="#E3E4CD"
width="192">
<p align="center"><font color="#000099" face="Verdana, Arial,
Helvetica, sans-serif" size="1">&nbsp;&nbsp;&nbsp;<b>Naval
    Postgraduate School<br>
&nbsp;&nbsp;&nbsp;1 University Circle<br>
&nbsp;&nbsp;&nbsp;Monterey, CA 93943-5001<br>
&nbsp;&nbsp;&nbsp;(831) 656-2441/2<br>
&nbsp;&nbsp;&nbsp;DSN: 878-2441/2</b></font></p>
<p align="left">&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</p>
<p align="left">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<b><font face="Verdana"><a
href="index.html">Admin Home</a></b></font></p>
<p align="left"><b>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<font face="Verdana"><a
href="weights.asp">View Weights</a></b></font></p>
        <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&nbsp;</p>
        <p align="center" style="MARGIN-TOP: 0px; MARGIN-BOTTOM:
0px">&nbsp;&nbsp;&nbsp;</p>
    </td>
    <td valign="top" height="1" width="602">
        <table width="583" border="0" align="center">
            <tr>
                <td width="575">
                    <div align="center"></div>
                </td>
            </tr>
            <tr>
                <td width="575">
                    &nbsp;&nbsp;&nbsp;<!--webbot bot="DatabaseRegionStart" startspan
s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe

```

```

ight,SelectedWeight,PValue" s-columntypes="3,131,131,131,131,131" s-
dataconnection="saward" b-tableformat="FALSE" b-menuformat="FALSE" s-
menuchoice="FirstChoiceWeight" s-menuvalue="FirstChoiceWeight" b-
tableborder="TRUE" b-tableexpand="TRUE" b-tableheader="TRUE" b-
listlabels="TRUE" b-listseparator="TRUE" i-ListFormat="7" b-
makeform="TRUE" s-recordsource="tnpSchieffelinWeights" s-
displaycolumns="FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWeight,
SelectedWeight,PValue,CalYear" s-criteria s-order s-sql="SELECT * FROM
tnpSchieffelinWeights" b-procedure="FALSE" clientside
SuggestedExt="asp" s-DefaultFields s-NoRecordsFound="No records
returned." i-MaxRecords="256" i-GroupSize="0" BOTID="0" u-
dblib="../_fpclass/fpdblib.inc" u-dbrgn1="../_fpclass/fpdbrgn1.inc" u-
dbrgn2="../_fpclass/fpdbrgn2.inc" tag="BODY" local_preview="<table
border=0 width="100%">>><tr>>><td
bgcolor="#FFFFF0" align="left">>>Database Results regions will not preview
unless this page is fetched from a Web server using a web browser. The
section of the page from here to the end of the Database Results region
will repeat once for each record returned by the
query.</font></td></tr></table>"
preview="<table border=0 width="100%">>><tr>>><td
bgcolor="#FFFFF0" align="left">>>This is the start of a Database Results
region. The page must be fetched from a web server with a web browser
to display correctly; the current web is stored on your local disk or
network.</font></td></tr></table>" b-
WasTableFormat="FALSE" --><!--#include file="../_fpclass/fpdblib.inc"--
>
<%
fp_sQry="SELECT * FROM tnpSchieffelinWeights"
fp_sDefault=""
fp_sNoRecords="No records returned."
fp_sDataConn="saward"
fp_iMaxRecords=256
fp_iCommandType=1
fp_iPageSize=0
fp_fTableFormat=False
fp_fMenuFormat=False
fp_sMenuChoice="FirstChoiceWeight"
fp_sMenuValue="FirstChoiceWeight"
fp_iDisplayCols=6
fp_fCustomQuery=False
BOTID=0
fp_iRegion=BOTID
%>
<!--#include file="../_fpclass/fpdbrgn1.inc"-->
<!--webbot bot="DatabaseRegionStart" i-checksum="53612" endspan -->
<form METHOD="POST" action="update.asp">
  <!--webbot bot="PurpleText" PREVIEW="Set this
form's properties so it submits user input to the appropriate page." --
>

  <table BORDER="0">
    <tr>
      <td><b><font
face="Verdana">FirstChoiceWeight:</b></font></td>

```

```

        <td><input TYPE="TEXT" NAME="FirstChoiceWeight"
SIZE="40" VALUE="<%=FP_FieldHTML(fp_rs,"FirstChoiceWeight")%>"></td>
    </tr>
    <tr>
        <td><b><font
face="Verdana">SecondChoiceWeight:</b></font></td>
        <td><input TYPE="TEXT"
NAME="SecondChoiceWeight" SIZE="40"
VALUE="<%=FP_FieldHTML(fp_rs,"SecondChoiceWeight")%>"></td>
    </tr>
    <tr>
        <td><b><font
face="Verdana">ThirdChoiceWeight:</b></font></td>
        <td><input TYPE="TEXT" NAME="ThirdChoiceWeight"
SIZE="40" VALUE="<%=FP_FieldHTML(fp_rs,"ThirdChoiceWeight")%>"></td>
    </tr>
    <tr>
        <td><b><font
face="Verdana">SelectedWeight:</b></font></td>
        <td><input TYPE="TEXT" NAME="SelectedWeight"
SIZE="40" VALUE="<%=FP_FieldHTML(fp_rs,"SelectedWeight")%>"></td>
    </tr>
    <tr>
        <td><b><font
face="Verdana">PValue:</b></font></td>
        <td><input TYPE="TEXT" NAME="PValue" SIZE="40"
VALUE="<%=FP_FieldHTML(fp_rs,"PValue")%>"></td>
    </tr>
    <tr>
        <td><b><font
face="Verdana">CalYear:</b></font></td>
        <td><input TYPE="TEXT" NAME="CalYear" SIZE="40"
VALUE="<%=FP_FieldHTML(fp_rs,"CalYear")%>"></td>
    </tr>
    <tr>
        <td COLSPAN="2"><br>
        <input TYPE="Submit" NAME="fp_submit"
value="Update"><input TYPE="Reset" NAME="fp_reset"></td>
    </tr>
</table>
</form>
<hr>
<!--webbot bot="DatabaseRegionEnd" startspan b-
tableformat="FALSE" b-menuformat="FALSE" u-
dbrgn2="../_fpclass/fpdbrgn2.inc" i-groupsize="0" clientside tag="BODY"
local_preview="&lt;table border=0
width=&quot;100%&quot;&gt;&lt;tr&gt;&lt;td bgcolor=&quot;#FFFF00&quot;
align=&quot;center&quot;&gt;&lt;font color=&quot;#000000&quot;&gt;End
of Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;&lt;/table&gt;"
preview="&lt;table border=0 width=&quot;100%&quot;&gt;&lt;tr&gt;&lt;td
bgcolor=&quot;#FFFF00&quot; align=&quot;left&quot;&gt;&lt;font
color=&quot;#000000&quot;&gt;This is the end of a Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;&lt;/table&gt;" --><!--
#include file="../_fpclass/fpdbrgn2.inc"-->
<!--webbot bot="DatabaseRegionEnd" i-checksum="56926" endspan -->

```

```

                <hr>
            </td>
        </tr>

    </table>
</td>

</tr>
</table>
</body>
</html>
-->

```

JJ. WEIGHTS.ASP

```

<!--
<html>
<head>
<title>Schieffelin Teaching Award</title>
<meta http-equiv="Content-Type" content="text/html;">
<meta http-equiv="Pragma" content="no-cache">
<!-- Fireworks 4.0 Dreamweaver 4.0 target. Created Fri Dec 08
09:16:10 GMT-0800 (Pacific Standard Time) 2000-->
<script language="JavaScript">
<!--
function MM_displayStatusMsg(msgStr) { //v3.0
    status=msgStr; document.MM_returnValue = true;
}

function MM_findObj(n, d) { //v3.0
    var p,i,x;  if(!d) d=document;
    if((p=n.indexOf("?"))>0&&parent.frames.length) {
        d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
    if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++)
x=d.forms[i][n];
    for(i=0;!x&&d.layers&&i<d.layers.length;i++)
x=MM_findObj(n,d.layers[i].document); return x;
}

function MM_nbGroup(event, grpName) { //v3.0
    var i,img,nbArr,args=MM_nbGroup.arguments;
    if (event == "init" && args.length > 2) {
        if ((img = MM_findObj(args[2])) != null && !img.MM_init) {
            img.MM_init = true; img.MM_up = args[3]; img.MM_dn = img.src;
            if ((nbArr = document[grpName]) == null) nbArr =
document[grpName] = new Array();
            nbArr[nbArr.length] = img;
            for (i=4; i < args.length-1; i+=2) if ((img =
MM_findObj(args[i])) != null) {
                if (!img.MM_up) img.MM_up = img.src;
                img.src = img.MM_dn = args[i+1];
                nbArr[nbArr.length] = img;
            } }
    } else if (event == "over") {
        document.MM_nbOver = nbArr = new Array();
    }
}

```

```

        for (i=1; i < args.length-1; i+=3) if ((img = MM_findObj(args[i]))
!= null) {
            if (!img.MM_up) img.MM_up = img.src;
            img.src = (img.MM_dn && args[i+2]) ? args[i+2] : args[i+1];
            nbArr[nbArr.length] = img;
        }
    } else if (event == "out" ) {
        for (i=0; i < document.MM_nbOver.length; i++) {
            img = document.MM_nbOver[i]; img.src = (img.MM_dn) ? img.MM_dn :
img.MM_up; }
    } else if (event == "down") {
        if ((nbArr = document[grpName]) != null)
            for (i=0; i < nbArr.length; i++) { img=nbArr[i]; img.src =
img.MM_up; img.MM_dn = 0; }
        document[grpName] = nbArr = new Array();
        for (i=2; i < args.length-1; i+=2) if ((img = MM_findObj(args[i]))
!= null) {
            if (!img.MM_up) img.MM_up = img.src;
            img.src = img.MM_dn = args[i+1];
            nbArr[nbArr.length] = img;
        } }
    }
}

function MM_preloadImages() { //v3.0
    var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
    var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0;
i<a.length; i++)
        if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image;
d.MM_p[j++].src=a[i];}}
}

//-->
</script>
</head>
<body bgcolor="#ffffff"
onLoad="MM_preloadImages('images/Academics_f2.gif','images/Academics_f4
.gif','images/Academics_f3.gif','images/About_f2.gif','images/About_f4.
gif','images/About_f3.gif','images/Research_f3.gif','images/Research_f2
.gif','images/Research_f4.gif','images/ExecutiveEducation_f3.gif','imag
es/ExecutiveEducation_f2.gif','images/ExecutiveEducation_f4.gif','image
s/Students_f3.gif','images/Students_f2.gif','images/Students_f4.gif','i
images/Faculty_f3.gif','images/Faculty_f2.gif','images/Faculty_f4.gif','
images/Administration_f3.gif','images/Administration_f2.gif','images/Ad
ministration_f4.gif','images/Alumni_f3.gif','images/Alumni_f2.gif','ima
ges/Alumni_f4.gif','images/Library_f3.gif','images/Library_f2.gif','ima
ges/Library_f4.gif','images/Search_f3.gif','images/Search_f2.gif','imag
es/Search_f4.gif','images/About_f3.gif','images/About_f2.gif','images/A
bout_f4.gif','images/Disclaimers_f3.gif','images/Disclaimers_f2.gif','i
images/Disclaimers_f4.gif','images/ExecutiveEducation_f3.gif','images/Ex
ecutiveEducation_f2.gif','images/ExecutiveEducation_f4.gif')"
leftmargin="0" topmargin="0" marginwidth="0" marginheight="0">
<table border="0" cellpadding="0" cellspacing="0" width="880"
height="1"><!-- fwtable fwsrc="T01Academics.png"
fwbase="01Academics.gif" fwstyle="Dreamweaver" fwdocid = "742308039"
fwnested="0" -->
<tr>

```



```
  |
```

```

        </thead>
        <tbody><!--webbot bot="DatabaseRegionStart"
startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-columntypes="3,131,131,131,131,131" s-
dataconnection="saward" b-tableformat="TRUE" b-menuformat="FALSE" s-
menuchoice s-menuvalue b-tableborder="TRUE" b-tableexpand="TRUE" b-
tableheader="TRUE" b-listlabels="TRUE" b-listseparator="TRUE" i-
ListFormat="0" b-makeform="TRUE" s-recordsource="tnpSchieffelinWeights"
s-
displaycolumns="FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWeight,
SelectedWeight,PValue" s-criteria s-order s-sql="SELECT * FROM
tnpSchieffelinWeights" b-procedure="FALSE" clientside
SuggestedExt="asp" s-DefaultFields s-NoRecordsFound="No records
returned." i-MaxRecords="256" i-GroupSize="0" BOTID="0" u-
dblib="../_fpclass/fpdblib.inc" u-dbrgn1="../_fpclass/fpdbrgn1.inc" u-
dbrgn2="../_fpclass/fpdbrgn2.inc" tag="TBODY"
local_preview="&lt;tr&gt;&lt;td colspan=64 bgcolor=&quot;#FFFF00&quot;
align=&quot;left&quot; width=&quot;100%&quot;&gt;&lt;font
color=&quot;#000000&quot;&gt;Database Results regions will not preview
unless this page is fetched from a Web server with a web browser. The
following table row will repeat once for every record returned by the
query.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" preview="&lt;tr&gt;&lt;td
colspan=64 bgcolor=&quot;#FFFF00&quot; align=&quot;left&quot;
width=&quot;100%&quot;&gt;&lt;font color=&quot;#000000&quot;&gt;This is
the start of a Database Results region. The page must be fetched from a
web server with a web browser to display correctly; the current web is
stored on your local disk or
network.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" --><!--#include
file="../_fpclass/fpdblib.inc"-->
<%
fp_sQry="SELECT * FROM tnpSchieffelinWeights"
fp_sDefault=""
fp_sNoRecords="<tr><td colspan=5 align=left width=""100%"">No records
returned.</td></tr>"
fp_sDataConn="saward"
fp_iMaxRecords=256
fp_iCommandType=1
fp_iPageSize=0
fp_fTableFormat=True
fp_fMenuFormat=False
fp_sMenuChoice=""
fp_sMenuValue=""
fp_iDisplayCols=5
fp_fCustomQuery=False
BOTID=0
fp_iRegion=BOTID
%>
<!--#include file="../_fpclass/fpdbrgn1.inc"--><!--webbot
bot="DatabaseRegionStart" i-checksum="27989" endspan -->
        <tr>
        <td><!--webbot bot="DatabaseResultColumn"
startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-column="FirstChoiceWeight" b-
tableformat="TRUE" b-hasHTML="FALSE" clientside local_preview="&lt;font

```

```

size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;FirstChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;FirstChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" --
><%=FP_FieldVal(fp_rs,"FirstChoiceWeight")%><!--webbot
bot="DatabaseResultColumn" i-checksum="45647" endspan -->
</td>
<td><!--webbot bot="DatabaseResultColumn"

startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-column="SecondChoiceWeight" b-
tableformat="TRUE" b-hasHTML="FALSE" clientside local_preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;SecondChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;SecondChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" --
><%=FP_FieldVal(fp_rs,"SecondChoiceWeight")%><!--webbot
bot="DatabaseResultColumn" i-checksum="56583" endspan -->
</td>
<td><!--webbot bot="DatabaseResultColumn"

startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-column="ThirdChoiceWeight" b-
tableformat="TRUE" b-hasHTML="FALSE" clientside local_preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;ThirdChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;ThirdChoiceWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" --
><%=FP_FieldVal(fp_rs,"ThirdChoiceWeight")%><!--webbot
bot="DatabaseResultColumn" i-checksum="46340" endspan -->
</td>
<td><!--webbot bot="DatabaseResultColumn"

startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-column="SelectedWeight" b-
tableformat="TRUE" b-hasHTML="FALSE" clientside local_preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;SelectedWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" preview="&lt;font
size="&quot;-
1&quot;&gt;&lt;&lt;&lt;/font&gt;SelectedWeight&lt;font
size="&quot;-1&quot;&gt;&gt;&gt;&lt;/font&gt;" --
><%=FP_FieldVal(fp_rs,"SelectedWeight")%><!--webbot
bot="DatabaseResultColumn" i-checksum="40142" endspan -->
</td>
<td><!--webbot bot="DatabaseResultColumn"

startspan s-
columnnames="CalYear,FirstChoiceWeight,SecondChoiceWeight,ThirdChoiceWe
ight,SelectedWeight,PValue" s-column="PValue" b-tableformat="TRUE" b-
hasHTML="FALSE" clientside local_preview="&lt;font size="&quot;-

```

```

1&quot;&gt;&amp;lt;&amp;lt;&lt;/font&gt;PValue&lt;font size=&quot;-
1&quot;&gt;&amp;gt;&amp;gt;&lt;/font&gt;" preview="&lt;font
size=&quot;-1&quot;&gt;&amp;lt;&amp;lt;&lt;/font&gt;PValue&lt;font
size=&quot;-1&quot;&gt;&amp;gt;&amp;gt;&lt;/font&gt;" --
><%=FP_FieldVal(fp_rs,"PValue")%><!--webbot bot="DatabaseResultColumn"
i-checksum="13018" endspan -->
</td>
</tr><!--webbot bot="DatabaseRegionEnd" startspan
b-tableformat="TRUE" b-menuformat="FALSE" u-
dbrgn2="../_fpclass/fpdbrgn2.inc" i-groupsize="0" clientside
tag="TBODY" local_preview="&lt;tr&gt;&lt;td colspan=64
bgcolor=&quot;#FFFF00&quot; align=&quot;left&quot;
width=&quot;100%&quot;&gt;&lt;font color=&quot;#000000&quot;&gt;This is
the end of a Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" preview="&lt;tr&gt;&lt;td
colspan=64 bgcolor=&quot;#FFFF00&quot; align=&quot;left&quot;
width=&quot;100%&quot;&gt;&lt;font color=&quot;#000000&quot;&gt;This is
the end of a Database Results
region.&lt;/font&gt;&lt;/td&gt;&lt;/tr&gt;" --><!--#include
file="../_fpclass/fpdbrgn2.inc"--><!--webbot bot="DatabaseRegionEnd" i-
checksum="56926" endspan -->
</tbody>
</table>
<p>&nbsp;</p>
</td>
</tr>

</table>
</td>
<td width="3" height="1"></td>
</tr>
</table>
</body>
</html>
-->

```

APPENDIX H. SCORING REPORTS

These are the screen shots of Web pages used for viewing the score reports.

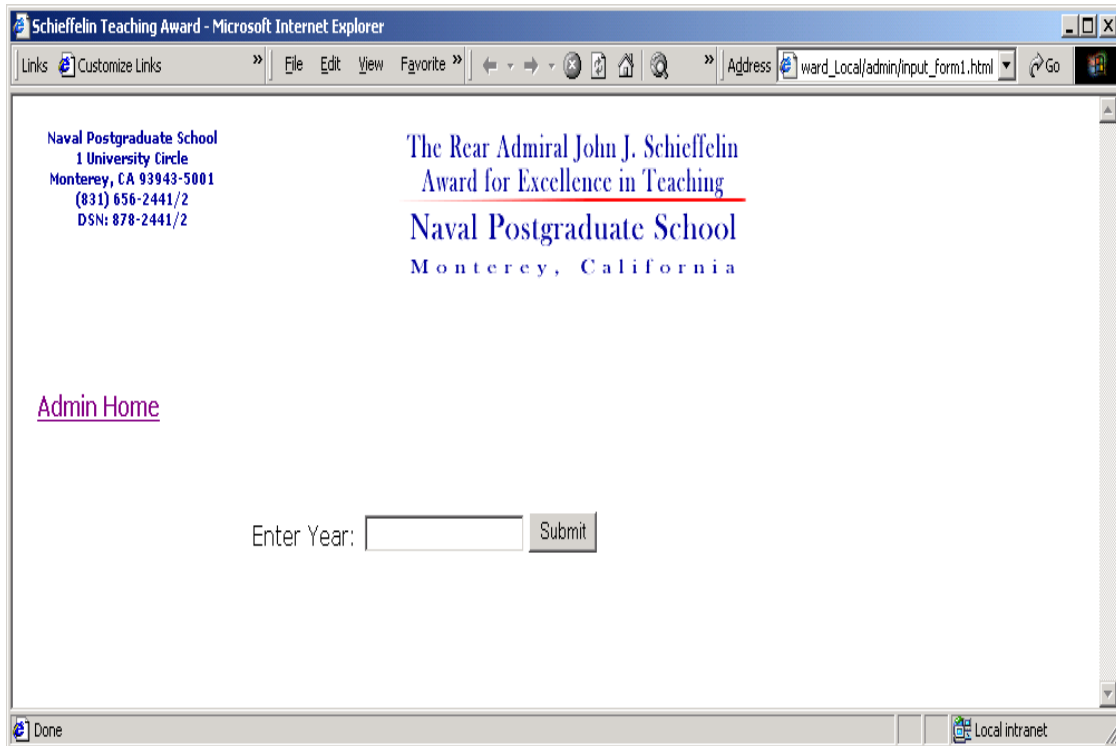


Figure 18. Input_Form1 & Input_Form2

Input_form1.html and Input_form2.html look identical. One posts to the chairman's copy of the scores, the other posts to the committee's copy.

http://accelerate/Award_Local/admin/rank1.asp - Microsoft Internet Explorer

Links Customize Links » File Edit View Favorite » Address http://accelerate/Award_Local/admin/ran Go

[Admin Home](#)

Chairman Copy for Teaching Year 2000

Rank	LName	FName	MName	Score	N	D	X1	X2	X3	Z1	Z2	Z3	Z4
1	Eagle	Christopher	Steven	5.454	64	15	28	12	6	24.752	9.73	4.138	5.688
2	Bonsper	Donald	E	5.104	51	11	21	8	11	18.109	5.669	7.59	3.462
3	Armstead	James	H	5.086	15	5	7	4	1	6.136	3.045	0.667	1.053
4	Lober	George	William	5.002	7	1	3	2	2	2.693	1.5	1.324	0
5	Hiles	John	E	4.837	11	4	6	0	1	5.42	0	0.813	1.485
6	Barrett	Frank	J	4.788	6	3	3	1	1	2.631	0.8	0.7	0.3
7	Lucas	Thomas	W	4.706	26	6	10	4	4	9.142	3.22	2.868	2.85
8	Simons	Anna		4.356	5	2	3	0	0	2.583	0	0	0.464
9	Morris	James	H	4.351	37	7	12	9	4	10.429	7.017	2.325	4.143
10	Robinson	Glenn	E	4.206	7	2	3	2	0	2.565	1.445	0	0.5
11	Leonard	Barry	S	4.063	12	1	5	1	0	4.608	0.895	0	2.192
12	Buttrey	Samuel	E	4.055	51	3	14	8	10	12.448	6.229	6.979	6.6
13	Myers	Jan	P	3.984	73	8	20	20	7	16.876	13.506	3.821	5.761
14	Durkee	Philip	A	3.979	20	4	7	2	4	6.214	1.485	2.542	1.92
15	Gue	Kevin	R	3.864	17	1	5	4	2	4.429	3.207	1.25	1.905
16	Shavit	Arthur		3.857	4	0	2	0	0	1.855	0	0	0.445
17	Looney	Robert	E	3.741	20	2	7	3	2	6.049	2.095	1	1.724
18	Ha	Tri	T	3.563	6	1	2	1	1	1.795	0.667	0.769	0.334
19	Miller	Harold	Lyman	3.553	6	2	2	2	0	1.666	1.445	0	0.381
20	Canright	David	R	3.529	49	6	11	10	5	9.799	7.779	3.363	7.371
21	Colson	William	B	3.494	12	2	4	1	1	3.534	0.875	0.8	2.068
22	Cooper	Glenn	C	3.488	46	7	13	7	5	10.961	4.835	2.821	4.886
23	Hobson	Garth	V	3.335	15	2	5	1	1	4.401	0.75	0.571	2.638
24				3.22	44	2	2	5	4	3.544	2.588	0.737	1.242

Done Local intranet

Figure 19. Rank1.asp (Chairman Copy)

Chairmain's copy of year 2000 scores and ranking.

http://accelerate/Award_Local/admin/rank2.asp - Microsoft Internet Explorer

Links Customize Links File Edit View Favorite Address http://accelerate/Award_Local/admin/rank2.asp Go

[Admin Home](#)

Committee Copy for Teaching Year 2000

Rank	Score	N	D	X1	X2	X3	Z1	Z2	Z3	Z4
1	5.454	64	15	28	12	6	24.752	9.73	4.138	5.688
2	5.104	51	11	21	8	11	18.109	5.669	7.59	3.462
3	5.086	15	5	7	4	1	6.136	3.045	0.667	1.053
4	5.002	7	1	3	2	2	2.693	1.5	1.324	0
5	4.837	11	4	6	0	1	5.42	0	0.813	1.485
6	4.788	6	3	3	1	1	2.631	0.8	0.7	0.3
7	4.706	26	6	10	4	4	9.142	3.22	2.868	2.85
8	4.356	5	2	3	0	0	2.583	0	0	0.464
9	4.351	37	7	12	9	4	10.429	7.017	2.325	4.143
10	4.206	7	2	3	2	0	2.565	1.445	0	0.5
11	4.063	12	1	5	1	0	4.608	0.895	0	2.192
12	4.055	51	3	14	8	10	12.448	6.229	6.979	6.6
13	3.984	73	8	20	20	7	16.876	13.506	3.821	5.761
14	3.979	20	4	7	2	4	6.214	1.485	2.542	1.92
15	3.864	17	1	5	4	2	4.429	3.207	1.25	1.905
16	3.857	4	0	2	0	0	1.855	0	0	0.445
17	3.741	20	2	7	3	2	6.049	2.095	1	1.724
18	3.563	6	1	2	1	1	1.795	0.667	0.769	0.334
19	3.553	6	2	2	2	0	1.666	1.445	0	0.381
20	3.529	49	6	11	10	5	9.799	7.779	3.363	7.371
21	3.494	12	2	4	1	1	3.534	0.875	0.8	2.068
22	3.488	46	7	13	7	5	10.961	4.835	2.821	4.886
23	3.335	15	2	5	1	1	4.401	0.75	0.571	2.638
24	3.32	14	2	3	5	1	2.541	3.598	0.727	1.342

Done Local intranet

Figure 20. Rank2.asp (Committee Copy)

Committee's copy of year 2000 scores and ranking.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX I. HISTORY REPORTS

Screen shots of the Schieffelin history reports.

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

For teaching year 20

PREVIOUS PERFORMANCE DATA

Rank	Name	Year	Score	N	D	PcRank	Year	N	Code	Year	N	Code	Year	N	Code	Year	N	Code	Year	N	Code
1	Eagle, Christopher Steven	2000	5.454	64	15		1999	77	A	1998	58	A	1997	25	B						
2	Bonsper, Donald E	2000	5.104	51	11		1999	54	A	1998	57	A	1997	0	I	1996	51	B	1995	43	B
3	Armstead, James H	2000	5.086	15	5		1999	11	E	1998	30	B	1997	29	E	1996	50	E	1995	58	E
4	Lober, George William	2000	5.002	7	1		1999	9	B	1998	0	I									
5	Hiles, John E	2000	4.837	11	4																
6	Barrett, Frank J	2000	4.788	6	3		1999	0	I	1998	28	E	1997	36	E	1996	41	B	1995	42	E
7	Lucas, Thomas W	2000	4.706	26	6		1999	28	B												
8	Simons, Anna	2000	4.356	5	2		1999	9	E	1998	0	I									
9	Morris, James H	2000	4.351	37	7		1999	36	E	1998	21	E	1997	63	A	1996	41	B	1995	44	B
10	Robinson, Glenn E	2000	4.206	7	2		1999	20	B	1998	15	B	1997	15	A	1996	20	E	1995	26	E
11	Leonard, Barry S	2000	4.063	12	1		1999	0	I	1998	0	I	1997	11	E	1996	0	I	1995	0	I
12	Buttrey, Samuel E	2000	4.055	51	3		1999	74	A	1998	43	A	1997	59	A	1996	0	I			
13	Myers, Jan P	2000	3.984	73	8																
14	Durkee, Philip A	2000	3.979	20	4		1999	20	B	1998	30	B	1997	20	E	1996	24	E	1995	0	I
15	Gue, Kevin R	2000	3.864	17	1		1999	13	E	1998	14	E	1997	12	E	1996	0	I			
16	Shavit, Arthur	2000	3.857	4	0		1999	0	I	1998	0	I	1997	21	E	1996	0	I	1995	0	I
17	Looney, Robert E	2000	3.741	20	2		1999	40	E	1998	28	E	1997	36	E	1996	38	E	1995	45	E
18	Ha, Tri T	2000	3.563	6	1		1999	0	I	1998	0	I	1997	14	B	1996	13	B	1995	15	E
19	Miller, Harold Lyman	2000	3.553	6	2																

Figure 21. History.asp (Chairman History Report).

Report shows the current year scores and ranking as well as the professor's Schieffelin history for the past six years.

Schieffelin Teaching Award - Microsoft Internet Explorer

Links Customize Links File Edit View Favorite Address elearning/Award_Local/admin/history1.asp Go

Postgraduate School
University Circle
Cincinnati, OH 45221-0001
Tel: 616-2441/2
Fax: 616-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

For teaching year 2000

[in Home](#) PREVIOUS PERFORMANCE DATA

[erman Copy](#)

Rank	Year	Score	N	D	PcRank	Year	N	Code	Year	N	Code	Year	N	Code	Year	N	Code	Year	N	Code
1	2000	5.454	64	15		1999	77	A	1998	58	A	1997	25	B						
2	2000	5.104	51	11		1999	54	A	1998	57	A	1997	0	I	1996	51	B	1995	43	B
3	2000	5.086	15	5		1999	11	E	1998	30	B	1997	29	E	1996	50	E	1995	58	E
4	2000	5.002	7	1		1999	9	B	1998	0	I									
5	2000	4.837	11	4																
6	2000	4.788	6	3		1999	0	I	1998	28	E	1997	36	E	1996	41	B	1995	42	E
7	2000	4.706	26	6		1999	28	B												
8	2000	4.356	5	2		1999	9	E	1998	0	I									
9	2000	4.351	37	7		1999	36	E	1998	21	E	1997	63	A	1996	41	B	1995	44	B
10	2000	4.206	7	2		1999	20	B	1998	15	B	1997	15	A	1996	20	E	1995	26	E
11	2000	4.063	12	1		1999	0	I	1998	0	I	1997	11	E	1996	0	I	1995	0	I
12	2000	4.055	51	3		1999	74	A	1998	43	A	1997	59	A	1996	0	I			
13	2000	3.984	73	8																
14	2000	3.979	20	4		1999	20	B	1998	30	B	1997	20	E	1996	24	E	1995	0	I
15	2000	3.864	17	1		1999	13	E	1998	14	E	1997	12	E	1996	0	I			
16	2000	3.857	4	0		1999	0	I	1998	0	I	1997	21	E	1996	0	I	1995	0	I
17	2000	3.741	20	2		1999	40	E	1998	28	E	1997	36	E	1996	38	E	1995	45	E
18	2000	3.563	6	1		1999	0	I	1998	0	I	1997	14	B	1996	13	B	1995	15	E

Done Local intranet

Figure 22. History1.asp (Committee History Report).

Report shows the current year scores and ranking as well as the professor's Schieffelin history for the past six years.

APPENDIX J. QUERY FOR PROFESSOR BY EMPLOYEE ID

These are screen shots of the Web pages used in querying the EMS for a particular professor by his/her employee ID.

The screenshot shows a Microsoft Internet Explorer window titled "Schieffelin Teaching Award - Microsoft Internet Explorer". The address bar displays "ward_Local/admin/input_form8.html". The page content includes the Naval Postgraduate School logo and contact information on the left, and the award name in the center. At the bottom, there is a link for "Admin Home" and a search form labeled "Enter Employee ID:" with a text input field and a "Submit" button.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

Enter Employee ID:

Figure 23. Input_form8.

Enter the employee's ID in the search form.

The screenshot shows the same Microsoft Internet Explorer window, but the address bar now displays "erate/Award_Local/admin/empid.asp". The page content is identical to Figure 23, but the search form has been replaced by a table of results. The table has columns for EmpID, Last Name, First Name, Middle Name, DeptID, and Department. The first row of data shows EmpID 135, Last Name Biblarz, First Name Oscar, and Department Aeronautics & Astronautics. There are also links for "Admin Home" and "Search Again".

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)
[Search Again](#)

EmpID	Last Name	First Name	Middle Name	DeptID	Department
135	Biblarz	Oscar		AA	Aeronautics & Astronautics

Figure 24. Empid.asp

The results based on the employee ID entered on the search form.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX K. QUERY FOR PROFESSOR BY LAST NAME

These are screen shots of the Web pages used in querying the EMS for a particular professor by his/her last name.

The screenshot shows a Microsoft Internet Explorer window titled "Schieffelin Teaching Award - Microsoft Internet Explorer". The address bar shows "ward_Local/admin/input_form7.html". The page content includes the Naval Postgraduate School logo and contact information on the left, and the title "The Rear Admiral John J. Schieffelin Award for Excellence in Teaching" in the center. Below the title is a link "Admin Home" and a search form with the label "Enter Last Name:" followed by a text input field and a "Submit" button.

Figure 25. Input_form7.

Enter the employee's last name in the search form.

The screenshot shows the same Microsoft Internet Explorer window, but the address bar now shows "erate/Award_Local/admin/lname.asp". The page content is similar to Figure 25, but it includes a "Search Again" link and a table of search results. The table has columns for EmpID, Last Name, First Name, Middle Name, DeptID, and Department. The results show a single entry for EmpID 398, Last Name Dolk, First Name Daniel, Middle Name R, DeptID IS, and Department Information Systems. Below the table are navigation buttons: "<|", "<", ">", and ">|".

EmpID	Last Name	First Name	Middle Name	DeptID	Department
398	Dolk	Daniel	R	IS	Information Systems

Figure 26. Lname.asp

The results based on last name entered on the search form.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX L. STORED PROCEDURES

This appendix contains the application code for all the stored procedures used by the SADBMS.

A. CALL SCORING

This stored procedure initializes the Schieffelin History table for the current year first by setting all scores to zero. It then executes a stored procedure (snpCountVotes) that counts all the votes for that year. Next it determines the "K" value for each ballot with snpKValues. It then calculates the score for each individual ballot. Then sums them and determines the final score and ranking.

```
CREATE PROCEDURE snpCallScoring
    @year int
AS
BEGIN
    UPDATE tnpSchieffelinHistory
    SET FirstChoiceScore = 0,
        SecondChoiceScore = 0,
        ThirdChoiceScore = 0,
        SelectedOnBallotScore = 0,
        FinalScore = 0
    --,PCScore = 0
    WHERE CalendarYear = @year
END

BEGIN
    EXEC snpCountVotes @year
END

BEGIN
    EXEC snpKValues @year
END

BEGIN
    declare scoring cursor for
```

```

SELECT BallotID
FROM tnpSchieffelinBallot
WHERE CalendarYear = @year

for read only
declare @ballotid int
open scoring
fetch scoring into @ballotid
while @@fetch_status = 0
BEGIN

EXEC snpSchieffelinScoring @ballotid, @year

fetch scoring into @ballotid
END
close scoring
deallocate scoring
END

BEGIN
EXEC snpFinalScore @year
END

BEGIN
EXEC snpSchieffelinRanks @year
END
GO

```

B. COUNT VOTES

This stored procedure counts the votes for each employee by ballot using the employee's ID and storing the results in a temporary table. The results are used by the stored procedure snpCallScoring.

```

CREATE PROCEDURE snpCountVotes
    @year int

AS

CREATE TABLE #t (
    EmployeeID int,
    FirstChoiceVotes int,

```



```
SecondChoiceVotes int,  
ThirdChoiceVotes int,  
Statements int,  
vote1 int,  
vote2 int,  
vote3 int,  
vote4 int,  
vote5 int,  
vote6 int,  
vote7 int,  
vote8 int,  
vote9 int,  
vote10 int,  
vote11 int,  
vote12 int,  
vote13 int,  
vote14 int,  
vote15 int,  
vote16 int,  
vote17 int,  
vote18 int,  
vote19 int,  
vote20 int,  
vote21 int,  
vote22 int,  
vote23 int,  
vote24 int,  
vote25 int)
```

```
BEGIN
```

```
declare lead cursor for
```

```
SELECT select1, vote1 FROM vnpSelect1  
WHERE CalendarYear = @year
```

```
for read only
```

```
declare @id1 int, @votes1 int
```

```
open lead
```

```
fetch lead into @id1, @votes1
```

```

while @@fetch_status = 0
    BEGIN
        INSERT INTO #t (EmployeeID, vote1)
        VALUES (@id1, @votes1)

        fetch lead into @id1, @votes1
    END
close lead
deallocate lead
END

```

```

BEGIN

declare lead cursor for

SELECT select2, vote2 FROM vnpSelect2
WHERE CalendarYear = @year

for read only

declare @id2 int, @votes2 int

open lead

fetch lead into @id2, @votes2

while @@fetch_status = 0
BEGIN
if @id2 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote2 = @votes2
        WHERE EmployeeID = @id2
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote2)
        VALUES (@id2, @votes2)
    END

fetch lead into @id2, @votes2
END
close lead
deallocate lead

```

END

BEGIN

declare lead cursor for

SELECT select3, vote3 FROM vnpSelect3
WHERE CalendarYear = @year

for read only

declare @id3 int, @votes3 int

open lead

fetch lead into @id3, @votes3

while @@fetch_status = 0

BEGIN

if @id3 IN(SELECT EmployeeID FROM #t)

BEGIN

UPDATE #t

SET vote3 = @votes3

WHERE EmployeeID = @id3

END

else

BEGIN

INSERT INTO #t (EmployeeID, vote3)

VALUES (@id3, @votes3)

END

fetch lead into @id3, @votes3

END

close lead

deallocate lead

END

BEGIN

declare lead cursor for

SELECT select4, vote4 FROM vnpSelect4

```

WHERE CalendarYear = @year

for read only

declare @id4 int, @votes4 int

open lead

fetch lead into @id4, @votes4

while @@fetch_status = 0
BEGIN
if @id4 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote4 = @votes4
        WHERE EmployeeID = @id4
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote4)
        VALUES (@id4, @votes4)
    END

fetch lead into @id4, @votes4
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select5, vote5 FROM vnpSelect5
WHERE CalendarYear = @year

for read only

declare @id5 int, @votes5 int

open lead

fetch lead into @id5, @votes5

```

```

while @@fetch_status = 0
BEGIN
if @id5 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote5 = @votes5
        WHERE EmployeeID = @id5
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote5)
        VALUES (@id5, @votes5)
    END

fetch lead into @id5, @votes5
END
close lead
deallocate lead
END

```

```

-----

BEGIN

declare lead cursor for

SELECT select6, vote6 FROM vnpSelect6
WHERE CalendarYear = @year

for read only

declare @id6 int, @votes6 int

open lead

fetch lead into @id6, @votes6

while @@fetch_status = 0
BEGIN
if @id6 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote6 = @votes6
        WHERE EmployeeID = @id6
    END

```

```

else
    BEGIN
        INSERT INTO #t (EmployeeID, vote6)
        VALUES (@id6, @votes6)
    END

    fetch lead into @id6, @votes6
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select7, vote7 FROM vnpSelect7
WHERE CalendarYear = @year

for read only

declare @id7 int, @votes7 int

open lead

fetch lead into @id7, @votes7

while @@fetch_status = 0
BEGIN
    if @id7 IN(SELECT EmployeeID FROM #t)
        BEGIN
            UPDATE #t
            SET vote7 = @votes7
            WHERE EmployeeID = @id7
        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, vote7)
            VALUES (@id7, @votes7)
        END

    fetch lead into @id7, @votes7
END
close lead

```

```
deallocate lead
END
```

```
BEGIN

declare lead cursor for

SELECT select8, vote8 FROM vnpSelect8
WHERE CalendarYear = @year

for read only

declare @id8 int, @votes8 int

open lead

fetch lead into @id8, @votes8

while @@fetch_status = 0
BEGIN
if @id8 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote8 = @votes8
        WHERE EmployeeID = @id8
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote8)
        VALUES (@id8, @votes8)
    END

fetch lead into @id8, @votes8
END
close lead
deallocate lead
END
```

```
BEGIN

declare lead cursor for
```

```

SELECT select9, vote9 FROM vnpSelect9
WHERE CalendarYear = @year

for read only

declare @id9 int, @votes9 int

open lead

fetch lead into @id9, @votes9

while @@fetch_status = 0
BEGIN
if @id9 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote9 = @votes9
        WHERE EmployeeID = @id9
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote9)
        VALUES (@id9, @votes9)
    END

fetch lead into @id9, @votes9
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select10, vote10 FROM vnpSelect10
WHERE CalendarYear = @year

for read only

declare @id10 int, @votes10 int

open lead

```



```

fetch lead into @id10, @votes10

while @@fetch_status = 0
BEGIN
if @id10 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote10 = @votes10
        WHERE EmployeeID = @id10
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote10)
        VALUES (@id10, @votes10)
    END

fetch lead into @id10, @votes10
END
close lead
deallocate lead
END

```

```

-----

BEGIN

declare lead cursor for

SELECT select11, vote11 FROM vnpSelect11
WHERE CalendarYear = @year

for read only

declare @id11 int, @votes11 int

open lead

fetch lead into @id11, @votes11

while @@fetch_status = 0
BEGIN
if @id11 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote11 = @votes11
        WHERE EmployeeID = @id11
    END

```

```

        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, vote11)
            VALUES (@id11, @votes11)
        END

        fetch lead into @id11, @votes11
    END
    close lead
    deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select12, vote12 FROM vnpSelect12
WHERE CalendarYear = @year

for read only

declare @id12 int, @votes12 int

open lead

fetch lead into @id12, @votes12

while @@fetch_status = 0
BEGIN
    if @id12 IN(SELECT EmployeeID FROM #t)
        BEGIN
            UPDATE #t
            SET vote12 = @votes12
            WHERE EmployeeID = @id12
        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, vote12)
            VALUES (@id12, @votes12)
        END

        fetch lead into @id12, @votes12
    END

```

```
close lead
deallocate lead
END
```

```
BEGIN

declare lead cursor for

SELECT select13, vote13 FROM vnpSelect13
WHERE CalendarYear = @year

for read only

declare @id13 int, @votes13 int

open lead

fetch lead into @id13, @votes13

while @@fetch_status = 0
BEGIN
if @id13 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote13 = @votes13
        WHERE EmployeeID = @id13
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote13)
        VALUES (@id13, @votes13)
    END

fetch lead into @id13, @votes13
END
close lead
deallocate lead
END
```

```
BEGIN

declare lead cursor for
```

```
SELECT select14, vote14 FROM vnpSelect14
WHERE CalendarYear = @year
```

```
for read only
```

```
declare @id14 int, @votes14 int
```

```
open lead
```

```
fetch lead into @id14, @votes14
```

```
while @@fetch_status = 0
```

```
BEGIN
```

```
if @id14 IN(SELECT EmployeeID FROM #t)
```

```
    BEGIN
```

```
        UPDATE #t
```

```
        SET vote14 = @votes14
```

```
        WHERE EmployeeID = @id14
```

```
    END
```

```
else
```

```
    BEGIN
```

```
        INSERT INTO #t (EmployeeID, vote14)
```

```
        VALUES (@id14, @votes14)
```

```
    END
```

```
fetch lead into @id14, @votes14
```

```
END
```

```
close lead
```

```
deallocate lead
```

```
END
```

```
-----
```

```
BEGIN
```

```
declare lead cursor for
```

```
SELECT select15, vote15 FROM vnpSelect15
```

```
WHERE CalendarYear = @year
```

```
for read only
```

```
declare @id15 int, @votes15 int
```

```
open lead
```

```

fetch lead into @id15, @votes15

while @@fetch_status = 0
BEGIN
if @id15 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote15 = @votes15
        WHERE EmployeeID = @id15
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote15)
        VALUES (@id15, @votes15)
    END

fetch lead into @id15, @votes15
END
close lead
deallocate lead
END

```

```

-----

BEGIN

declare lead cursor for

SELECT select16, vote16 FROM vnpSelect16
WHERE CalendarYear = @year

for read only

declare @id16 int, @votes16 int

open lead

fetch lead into @id16, @votes16

while @@fetch_status = 0
BEGIN
if @id16 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote16 = @votes16
    END

```

```

        WHERE EmployeeID = @id16
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote16)
        VALUES (@id16, @votes16)
    END

    fetch lead into @id16, @votes16
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select17, vote17 FROM vnpSelect17
WHERE CalendarYear = @year

for read only

declare @id17 int, @votes17 int

open lead

fetch lead into @id17, @votes17

while @@fetch_status = 0
BEGIN
    if @id17 IN(SELECT EmployeeID FROM #t)
        BEGIN
            UPDATE #t
            SET vote17 = @votes17
            WHERE EmployeeID = @id17
        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, vote17)
            VALUES (@id17, @votes17)
        END

    fetch lead into @id17, @votes17

```

```

END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select18, vote18 FROM vnpSelect18
WHERE CalendarYear = @year

for read only

declare @id18 int, @votes18 int

open lead

fetch lead into @id18, @votes18

while @@fetch_status = 0
BEGIN
if @id18 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote18 = @votes18
        WHERE EmployeeID = @id18
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote18)
        VALUES (@id18, @votes18)
    END

fetch lead into @id18, @votes18
END
close lead
deallocate lead
END

-----

BEGIN

```

```

declare lead cursor for

SELECT select19, vote19 FROM vnpSelect19
WHERE CalendarYear = @year

for read only

declare @id19 int, @votes19 int

open lead

fetch lead into @id19, @votes19

while @@fetch_status = 0
BEGIN
if @id19 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote19 = @votes19
        WHERE EmployeeID = @id19
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote19)
        VALUES (@id19, @votes19)
    END

fetch lead into @id19, @votes19
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select20, vote20 FROM vnpSelect20
WHERE CalendarYear = @year

for read only

declare @id20 int, @votes20 int

```



```

open lead

fetch lead into @id20, @votes20

while @@fetch_status = 0
BEGIN
if @id20 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote20 = @votes20
        WHERE EmployeeID = @id20
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote20)
        VALUES (@id20, @votes20)
    END

fetch lead into @id20, @votes20
END
close lead
deallocate lead
END

```

```

BEGIN

declare lead cursor for

SELECT select21, vote21 FROM vnpSelect21
WHERE CalendarYear = @year

for read only

declare @id21 int, @votes21 int

open lead

fetch lead into @id21, @votes21

while @@fetch_status = 0
BEGIN
if @id21 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t

```

```

        SET vote21 = @votes21
        WHERE EmployeeID = @id21
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote21)
        VALUES (@id21, @votes21)
    END

    fetch lead into @id21, @votes21
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select22, vote22 FROM vnpSelect22
WHERE CalendarYear = @year

for read only

declare @id22 int, @votes22 int

open lead

fetch lead into @id22, @votes22

while @@fetch_status = 0
BEGIN
    if @id22 IN(SELECT EmployeeID FROM #t)
        BEGIN
            UPDATE #t
            SET vote22 = @votes22
            WHERE EmployeeID = @id22
        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, vote22)
            VALUES (@id22, @votes22)
        END

```

```

fetch lead into @id22, @votes22
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT select23, vote23 FROM vnpSelect23
WHERE CalendarYear = @year

for read only

declare @id23 int, @votes23 int

open lead

fetch lead into @id23, @votes23

while @@fetch_status = 0
BEGIN
if @id23 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote23 = @votes23
        WHERE EmployeeID = @id23
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote23)
        VALUES (@id23, @votes23)
    END

fetch lead into @id23, @votes23
END
close lead
deallocate lead
END

-----

BEGIN

```

```

declare lead cursor for

SELECT select24, vote24 FROM vnpSelect24
WHERE CalendarYear = @year

for read only

declare @id24 int, @votes24 int

open lead

fetch lead into @id24, @votes24

while @@fetch_status = 0
BEGIN
if @id24 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote24 = @votes24
        WHERE EmployeeID = @id24
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote24)
        VALUES (@id24, @votes24)
    END

fetch lead into @id24, @votes24
END
close lead
deallocate lead
END

```

```

BEGIN

declare lead cursor for

SELECT select25, vote25 FROM vnpSelect25
WHERE CalendarYear = @year

for read only

declare @id25 int, @votes25 int

```

```

open lead

fetch lead into @id25, @votes25

while @@fetch_status = 0
BEGIN
if @id25 IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET vote25 = @votes25
        WHERE EmployeeID = @id25
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, vote25)
        VALUES (@id25, @votes25)
    END

```

```

fetch lead into @id25, @votes25
END
close lead
deallocate lead
END

```

```

BEGIN

declare lead cursor for

SELECT FirstChoice, FirstChoiceVotes FROM vnpFirstChoice
WHERE CalendarYear = @year

for read only

declare @idfirst int, @votesfirst int

open lead

fetch lead into @idfirst, @votesfirst

while @@fetch_status = 0
BEGIN
if @idfirst IN(SELECT EmployeeID FROM #t)
    BEGIN

```

```

        UPDATE #t
        SET FirstChoiceVotes = @votesfirst
        WHERE EmployeeID = @idfirst
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, FirstChoiceVotes)
        VALUES (@idfirst, @votesfirst)
    END

    fetch lead into @idfirst, @votesfirst
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT SecondChoice, SecondChoiceVotes FROM vnpSecondChoice
WHERE CalendarYear = @year

for read only

declare @idsecond int, @votessecond int

open lead

fetch lead into @idsecond, @votessecond

while @@fetch_status = 0
BEGIN
    if @idsecond IN(SELECT EmployeeID FROM #t)
        BEGIN
            UPDATE #t
            SET SecondChoiceVotes = @votessecond
            WHERE EmployeeID = @idsecond
        END
    else
        BEGIN
            INSERT INTO #t (EmployeeID, SecondChoiceVotes)
            VALUES (@idsecond, @votessecond)
        END

```

```

fetch lead into @idsecond, @votessecond
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT ThirdChoice, ThirdChoiceVotes FROM vnpThirdChoice
WHERE CalendarYear = @year

for read only

declare @idthird int, @votesthird int

open lead

fetch lead into @idthird, @votesthird

while @@fetch_status = 0
BEGIN
if @idthird IN(SELECT EmployeeID FROM #t)
BEGIN
UPDATE #t
SET ThirdChoiceVotes = @votesthird
WHERE EmployeeID = @idthird
END
else
BEGIN
INSERT INTO #t (EmployeeID, ThirdChoiceVotes)
VALUES (@idthird, @votesthird)
END

fetch lead into @idthird, @votesthird
END
close lead
deallocate lead
END

-----

```

```

BEGIN

declare lead cursor for

SELECT FirstChoice, NumberStatements FROM
vnpNumberSupportStatements
WHERE CalendarYear = @year

for read only

declare @idstatements int, @statements int

open lead

fetch lead into @idstatements, @statements

while @@fetch_status = 0
BEGIN
if @idstatements IN(SELECT EmployeeID FROM #t)
    BEGIN
        UPDATE #t
        SET Statements = @statements
        WHERE EmployeeID = @idstatements
    END
else
    BEGIN
        INSERT INTO #t (EmployeeID, Statements)
        VALUES (@idstatements, @statements)
    END

fetch lead into @idstatements, @statements
END
close lead
deallocate lead
END

-----
BEGIN

UPDATE #t
SET FirstChoiceVotes = 0
WHERE FirstChoiceVotes is null

UPDATE #t
SET SecondChoiceVotes = 0
WHERE SecondChoiceVotes is null

```



```
UPDATE #t
SET ThirdChoiceVotes = 0
WHERE ThirdChoiceVotes is null
```

```
UPDATE #t
SET Statements = 0
WHERE Statements is null
```

```
UPDATE #t
SET vote1 = 0
WHERE vote1 is null
```

```
UPDATE #t
SET vote2 = 0
WHERE vote2 is null
```

```
UPDATE #t
SET vote3 = 0
WHERE vote3 is null
```

```
UPDATE #t
SET vote4 = 0
WHERE vote4 is null
```

```
UPDATE #t
SET vote5 = 0
WHERE vote5 is null
```

```
UPDATE #t
SET vote6 = 0
WHERE vote6 is null
```

```
UPDATE #t
SET vote7 = 0
WHERE vote7 is null
```

```
UPDATE #t
SET vote8 = 0
WHERE vote8 is null
```

```
UPDATE #t
SET vote9 = 0
WHERE vote9 is null
```

```
UPDATE #t
```

```
SET vote10 = 0
WHERE vote10 is null
```

```
UPDATE #t
SET vote11 = 0
WHERE vote11 is null
```

```
UPDATE #t
SET vote12 = 0
WHERE vote12 is null
```

```
UPDATE #t
SET vote13 = 0
WHERE vote13 is null
```

```
UPDATE #t
SET vote14 = 0
WHERE vote14 is null
```

```
UPDATE #t
SET vote15 = 0
WHERE vote15 is null
```

```
UPDATE #t
SET vote16 = 0
WHERE vote16 is null
```

```
UPDATE #t
SET vote17 = 0
WHERE vote17 is null
```

```
UPDATE #t
SET vote18 = 0
WHERE vote18 is null
```

```
UPDATE #t
SET vote19 = 0
WHERE vote19 is null
```

```
UPDATE #t
SET vote20 = 0
WHERE vote20 is null
```

```
UPDATE #t
SET vote21 = 0
WHERE vote21 is null
```

```
UPDATE #t
SET vote22 = 0
WHERE vote22 is null
```

```
UPDATE #t
SET vote23 = 0
WHERE vote23 is null
```

```
UPDATE #t
SET vote24 = 0
WHERE vote24 is null
```

```
UPDATE #t
SET vote25 = 0
WHERE vote25 is null
```

```
END
-----
```

```
BEGIN
```

```
declare lead cursor for
```

```
SELECT EmployeeID, FirstChoiceVotes, SecondChoiceVotes,
ThirdChoiceVotes, Statements
FROM #t
```

```
for read only
```

```
declare @empid int, @fcvotes int, @scvotes int, @tcvotes
int, @nstatements int
```

```
open lead
```

```
fetch lead into @empid, @fcvotes, @scvotes, @tcvotes,
@nstatements
```

```
while @@fetch_status = 0
BEGIN
```

```
UPDATE dbo.tnpSchieffelinHistory
SET FirstChoiceVotes = @fcvotes,
SecondChoiceVotes = @scvotes,
ThirdChoiceVotes = @tcvotes,
NumberSupportStatements = @nstatements
```

```

WHERE (EmployeeID = @empid and CalendarYear = @year)

fetch lead into @empid, @fcvotes, @scvotes, @tcvotes,
@nstatements
END
close lead
deallocate lead
END

-----

BEGIN

declare lead cursor for

SELECT EmployeeID, vote1,vote2, vote3 , vote4 , vote5 ,
vote6 , vote7 , vote8 ,
    vote9 , vote10 , vote11 , vote12 , vote13 , vote14 ,
vote15 , vote16 , vote17 , vote18,
    vote19 , vote20 , vote21 , vote22 , vote23 , vote24 ,
vote25
FROM #t

for read only

declare @emplid int,
@v1 int,
@v2 int,
@v3 int,
@v4 int,
@v5 int,
@v6 int,
@v7 int,
@v8 int,
@v9 int,
@v10 int,
@v11 int,
@v12 int,
@v13 int,
@v14 int,
@v15 int,
@v16 int,
@v17 int,
@v18 int,
@v19 int,
@v20 int,

```

```

@v21 int,
@v22 int,
@v23 int,
@v24 int,
@v25 int

open lead

fetch lead into @emplid,
@v1,@v2,@v3,@v4,@v5,@v6,@v7,@v8,@v9,@v10,
@v11,@v12,@v13,@v14,@v15,@v16,@v17,@v18,@v19,@v20,@v21,@v22
,@v23,@v24,@v25

while @@fetch_status = 0
BEGIN

UPDATE dbo.tnpSchieffelinHistory
SET NBallots =
(@v1+@v2+@v3+@v4+@v5+@v6+@v7+@v8+@v9+@v10+@v11+@v12+
@v13+@v14+@v15+@v16+@v17+@v18+@v19+@v20+@v21+@v22+@v23+@v24
+@v25)
WHERE (EmployeeID = @emplid and CalendarYear = @year)

fetch lead into @emplid,
@v1,@v2,@v3,@v4,@v5,@v6,@v7,@v8,@v9,@v10,
@v11,@v12,@v13,@v14,@v15,@v16,@v17,@v18,@v19,@v20,@v21,@v22
,@v23,@v24,@v25
END
close lead
deallocate lead
END
GO

```

C. DETERMINE SCHIEFFELIN ELIGIBILITY

This stored procedure determines Schieffelin Award eligibility for faculty for the inputted calendar year. Eligibility is initially set to yes for all faculty.

The following criteria make a faculty member ineligible:

- o Is a contracted faculty

- o Has already won the Schieffelin Award
- o Is on the Award Committee
- o Has taught less than 3 course segments during the year
- o Lab and lecture must be taught by the same faculty member
- o Must have greater than 5 students in the class to qualify
 - Has taught less than 11 credit hours
 - Directed study courses do not count
 - Lecture hours = 1 credit. Lab Hours = 0.5 credit
 - Team teaching = proportional credit.

```

CREATE PROCEDURE snpDetermineScheffelinEligibility
    @year int
AS
declare @year1 int
SET @year1 = @year + 1

--Put all faculty in the Schieffelin History table for the
inputted year.

declare lead cursor for
SELECT EmployeeID FROM tnpFaculty
WHERE NOT EXISTS (SELECT EmployeeID FROM
tnpSchieffelinHistory
WHERE CalendarYear = @year)

for read only

declare @id int

open lead

fetch lead into @id

while @@fetch_status = 0
BEGIN
INSERT INTO tnpSchieffelinHistory (EmployeeID,
CalendarYear)
VALUES (@id, @year)

```

```

fetch lead into @id
END
close lead
deallocate lead

--Set all faculty for inputted year as eligible, initially.
BEGIN
UPDATE tnpSchieffelinHistory
SET IsEligible = 'Y'
WHERE CalendarYear = @year
END

--Make faculty ineligible that are contracted, on the Award
--Committee, or past winners.

EXEC snpSchieffelinIneligible @year

--Make faculty ineligible that have taught <3 segments

EXEC snpSchiefSegments @year

--Make faculty ineligible that have taught <11 credit hours
--Account for team teaching
--Lecture = 1.0 credit hour.  Lab = 0.5 credit hour.

EXEC snpSchieffelinCreditHours @year

GO

```

D. FINAL SCORE

This stored procedure calculates the final score of a faculty member's Schieffelin Award votes for the input calendar year.

```

CREATE PROCEDURE snpFinalScore
    @year int
AS
declare
@w1 decimal(5,3),
@w2 decimal(5,3),
@w3 decimal(5,3),
@w4 decimal(5,3),
@pvalue decimal(5,3),

```

```

@finalscore decimal(5,3)

SELECT @w1 = FirstChoiceWeight,
@w2 = SecondChoiceWeight,
@w3 = ThirdChoiceWeight,
@w4 = SelectedWeight,
@pvalue = PValue
FROM tnpSchieffelinWeights
WHERE CalYear = @year

BEGIN

declare lead cursor for

SELECT EmployeeID, NBallots,
FirstChoiceScore, SecondChoiceScore, ThirdChoiceScore, SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE CalendarYear = @year
for read only
declare @id int,
@nballots decimal(5,3),
@z1 decimal(5,3),
@z2 decimal(5,3),
@z3 decimal(5,3),
@z4 decimal(5,3)

open lead
fetch lead into @id, @nballots, @z1, @z2, @z3, @z4
while @@fetch_status = 0
BEGIN
SET @finalscore = ((@w1 * @z1) + (@w2 * @z2) + (@w3 * @z3)
+ (@w4 * @z4)) / (POWER(@Nballots, @pvalue))

UPDATE tnpSchieffelinHistory
SET FinalScore = @finalscore
WHERE EmployeeID = @id and CalendarYear = @year
fetch lead into @id, @nballots, @z1, @z2, @z3, @z4

END

close lead
deallocate lead
END
GO

```


E. GET LOGIN

This procedure is used to get the users login ID (userID for the NPGS Domain.

```
CREATE PROCEDURE dbo.snpGetLogin
(
@strlogin varchar(20)
)
AS
SELECT dbo.tnpEmployee.*, dbo.tnpPythonProfile.*
FROM   dbo.tnpEmployee INNER JOIN
        dbo.tnpPythonProfile ON
        dbo.tnpEmployee.EmployeeID =
        dbo.tnpPythonProfile.EmployeeID
WHERE  dbo.tnpEmployee.NPGS = @strlogin

GO
```

F. K VALUES

This stored procedure calculates the number of candidates identified on each ballot (K) for the given year.

```
CREATE PROCEDURE snpKValues
    @year int
AS

declare lead cursor for

SELECT ballotid,
select1,
select2,
select3,
select4,
select5,
select6,
select7,
select8,
select9,
select10,
```

```

select11,
select12,
select13,
select14,
select15,
select16,
select17,
select18,
select19,
select20,
select21,
select22,
select23,
select24,
select25
FROM tnpSchieffelinBallot
WHERE CalendarYear = @year

```

for read only

```

declare
@ballotid int,
@s1 int,
@s2 int,
@s3 int,
@s4 int,
@s5 int,
@s6 int,
@s7 int,
@s8 int,
@s9 int,
@s10 int,
@s11 int,
@s12 int,
@s13 int,
@s14 int,
@s15 int,
@s16 int,
@s17 int,
@s18 int,
@s19 int,
@s20 int,
@s21 int,
@s22 int,
@s23 int,
@s24 int,

```

```

@s25 int

open lead

fetch lead into @ballotid,
@s1,@s2,@s3,@s4,@s5,@s6,@s7,@s8,@s9,@s10,@s11,@s12,@s13,@s1
4,@s15,@s16,@s17,@s18,@s19,@s20,
@s21,@s22,@s23,@s24,@s25

while @@fetch_Status = 0
begin
declare @kvalue int

SET @kvalue = 25

if @s25 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s24 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s23 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s22 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s21 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s20 is null
    BEGIN
        SET @kvalue = @kvalue - 1
    END

if @s19 is null

```

```

        BEGIN
        SET @kvalue = @kvalue - 1
        END

if @s18 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s17 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s16 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s15 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s14 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s13 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s12 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s11 is null
    BEGIN
    SET @kvalue = @kvalue - 1
    END

if @s10 is null
    BEGIN

```

```

        SET @kvalue = @kvalue - 1
    END

    if @s9 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s8 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s7 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s6 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s5 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s4 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s3 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    if @s2 is null
        BEGIN
            SET @kvalue = @kvalue - 1
        END

    UPDATE tnpSchieffelinBallot
    SET KValue = @kvalue
    WHERE BallotID = @ballotid and CalendarYear = @year

```

```

fetch lead into @ballotid,
@s1,@s2,@s3,@s4,@s5,@s6,@s7,@s8,@s9,@s10,@s11,@s12,@s13,@s1
4,@s15,@s16,@s17,@s18,@s19,@s20,
@s21,@s22,@s23,@s24,@s25
end
close lead
deallocate lead
GO

```

G. SCHIEFFELIN CREDIT HOURS

This stored procedure sums up the total number of credit hours the professor taught during the award period.

```

CREATE PROCEDURE snpSchieffelinCreditHours
    @year int

AS

declare @year1 int
SET @year1 = @year + 1

/*SELECT Quarter, CourseID, Segment, [Year], COUNT(*) AS
Instructors
FROM dbo.tnpFacultyClass
GROUP BY Quarter, CourseID, Segment, [Year],
IsLabInstructor, IsLectureInstructor
HAVING (IsLabInstructor = 'Y') AND (IsLectureInstructor =
'Y')*/

UPDATE tnpSchieffelinHistory
SET IsEligible = 'N'
WHERE tnpSchieffelinHistory.CalendarYear = @year AND
tnpSchieffelinHistory.EmployeeID IN
(SELECT dbo.tnpFacultyClass.EmployeeID
FROM dbo.tnpFacultyClass INNER JOIN
dbo.tnpClass ON dbo.tnpFacultyClass.Segment =
dbo.tnpClass.Segment AND dbo.tnpFacultyClass.CourseID =
dbo.tnpClass.CourseID AND
dbo.tnpFacultyClass.Quarter = dbo.tnpClass.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.tnpClass.[Year] INNER JOIN
dbo.tnpClassType ON dbo.tnpClass.ClassTypeID =
dbo.tnpClassType.ClassTypeID INNER JOIN

```

```

dbo.vnpFacultyCount ON dbo.tnpFacultyClass.Segment =
dbo.vnpFacultyCount.Segment AND
dbo.tnpFacultyClass.CourseID = dbo.vnpFacultyCount.CourseID
AND dbo.tnpFacultyClass.Quarter =
dbo.vnpFacultyCount.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.vnpFacultyCount.[Year]
WHERE (NOT (dbo.tnpClassType.ClassType = 'Directed Study'
OR
dbo.tnpClassType.ClassType = 'Thesis' OR
dbo.tnpClassType.ClassType = 'Seminar')) AND
(dbo.tnpFacultyClass.IsLabInstructor = 'Y') AND
(dbo.tnpFacultyClass.IsLectureInstructor = 'Y')
AND ((tnpFacultyClass.[Year] = @year AND
((tnpFacultyClass.[Quarter] = 2) OR
(tnpFacultyClass.[Quarter] = 3) OR
(tnpFacultyClass.[Quarter] = 4)))
OR ((tnpFacultyClass.[Year] = @year1) AND
(tnpFacultyClass.[Quarter] = 1)))
GROUP BY EmployeeID
HAVING (((SUM((AssignedLectureHours/Instructors) +
((AssignedLabHours / 2)/Instructors))) < 11.0)))

GO

```

H. SCHIEFFELIN INELIGIBLE

This stored procedure updates the Schieffelin history table by setting the "IsEligible" flag to "N" for the given year for all faculty that can be determined as ineligible based on if they are contracted, are previous winners or are committee members.

```

CREATE PROCEDURE snpSchieffelinIneligible
    @year int
AS

UPDATE tnpSchieffelinHistory
SET IsEligible = 'N'
WHERE tnpSchieffelinHistory.EmployeeID IN (
SELECT dbo.tnpFaculty.EmployeeID
FROM dbo.tnpFaculty INNER JOIN
dbo.tnpSchieffelinHistory ON dbo.tnpFaculty.EmployeeID =
dbo.tnpSchieffelinHistory.EmployeeID
WHERE (dbo.tnpFaculty.IsContracted = 'Y') OR

```

```

(dbo.tnpFaculty.HasWonScheffelin = 'Y'))

UPDATE tnpSchieffelinHistory
SET IsEligible = 'N'
WHERE (dbo.tnpSchieffelinHistory.IsCommittee = 'Y') AND
(CalendarYear = @year)

GO

```

I. SCHIEFFELIN RANKS

This stored procedure ranks the candidates based on their final scores in descending order.

```

CREATE PROCEDURE snpSchieffelinRanks
    @year int

AS
declare rank cursor for
SELECT EmployeeID
FROM dbo.vnpSchieffelinRanking
WHERE CalendarYear = @year
ORDER BY FinalScore DESC
for read only

declare @id int
open rank
fetch rank into @id
declare @rank int
SET @rank = 0
while @@fetch_status = 0
BEGIN
SET @rank = @rank + 1

UPDATE vnpSchieffelinRanking
SET Rank = @rank
WHERE EmployeeID = @id and CalendarYear = @year

fetch rank into @id
END
close rank
deallocate rank

GO

```


J. SCHIEFFELIN SCORING

This stored procedure calculates FirstChoiceScore, SecondChoiceScore, ThirdChoiceScore & SelectedOnBallotScore for all faculty members who were identified on ballots for the given year.

```
CREATE PROCEDURE snpSchieffelinScoring
    @ballotid int,
    @year int
AS
-----

BEGIN

declare schieffscore cursor for

SELECT FirstChoice,SecondChoice,ThirdChoice,
select1,
select2,
select3,
select4,
select5,
select6,
select7,
select8,
select9,
select10,
select11,
select12,
select13,
select14,
select15,
select16,
select17,
select18,
select19,
select20,
select21,
select22,
select23,
select24,
select25,
```

```

KValue
FROM tnpSchieffelinBallot
WHERE BallotID = @ballotid and CalendarYear = @year

for read only

declare
@firstchoice int,
@secondchoice int,
@thirdchoice int,
@s1 int,
@s2 int,
@s3 int,
@s4 int,
@s5 int,
@s6 int,
@s7 int,
@s8 int,
@s9 int,
@s10 int,
@s11 int,
@s12 int,
@s13 int,
@s14 int,
@s15 int,
@s16 int,
@s17 int,
@s18 int,
@s19 int,
@s20 int,
@s21 int,
@s22 int,
@s23 int,
@s24 int,
@s25 int,
@kvalue decimal(5,3)

open schieffscore

fetch schieffscore into
@firstchoice,@secondchoice,@thirdchoice,@s1,@s2,@s3,@s4,@s5
,@s6,@s7,@s8,@s9,@s10,@s11,@s12,@s13,@s14,@s15,@s16,@s17,@s
18,@s19,@s20,
@s21,@s22,@s23,@s24,@s25,@kvalue

while @@fetch_status = 0

```

BEGIN

```
declare @z1 decimal(5,3),@z2 decimal(5,3),@z3 decimal(5,3),
@oldz1 decimal(5,3),@oldz2 decimal(5,3),@oldz3
decimal(5,3),
@sel1 decimal(5,3),
@sel2 decimal(5,3),
@sel3 decimal(5,3),
@sel4 decimal(5,3),
@sel5 decimal(5,3),
@sel6 decimal(5,3),
@sel7 decimal(5,3),
@sel8 decimal(5,3),
@sel9 decimal(5,3),
@sel10 decimal(5,3),
@sel11 decimal(5,3),
@sel12 decimal(5,3),
@sel13 decimal(5,3),
@sel14 decimal(5,3),
@sel15 decimal(5,3),
@sel16 decimal(5,3),
@sel17 decimal(5,3),
@sel18 decimal(5,3),
@sel19 decimal(5,3),
@sel20 decimal(5,3),
@sel21 decimal(5,3),
@sel22 decimal(5,3),
@sel23 decimal(5,3),
@sel24 decimal(5,3),
@sel25 decimal(5,3),
@oldsel1 decimal(5,3),
@oldsel2 decimal(5,3),
@oldsel3 decimal(5,3),
@oldsel4 decimal(5,3),
@oldsel5 decimal(5,3),
@oldsel6 decimal(5,3),
@oldsel7 decimal(5,3),
@oldsel8 decimal(5,3),
@oldsel9 decimal(5,3),
@oldsel10 decimal(5,3),
@oldsel11 decimal(5,3),
@oldsel12 decimal(5,3),
@oldsel13 decimal(5,3),
@oldsel14 decimal(5,3),
@oldsel15 decimal(5,3),
@oldsel16 decimal(5,3),
```

```

@oldsel17 decimal(5,3),
@oldsel18 decimal(5,3),
@oldsel19 decimal(5,3),
@oldsel20 decimal(5,3),
@oldsel21 decimal(5,3),
@oldsel22 decimal(5,3),
@oldsel23 decimal(5,3),
@oldsel24 decimal(5,3),
@oldsel25 decimal(5,3),
@z1formula decimal(5,3),
@z2formula decimal(5,3),
@z3formula decimal(5,3),
@z4formula decimal(5,3)

--*****
--Scoring Formulas
--IF SCORING FORMULAS CHANGE, THEY ONLY NEED TO BE CHANGED
HERE.
--*****

SET @z1formula = (@kvalue)/(@kvalue+1.000)
SET @z2formula = (@kvalue-1.000)/(@kvalue+1.000)
SET @z3formula = (@kvalue-2.000)/(@kvalue+1.000)
SET @z4formula = (@kvalue-3.000)/((2.000)*(@kvalue+1.000))

--
*****
*****
--First Choice Score
--
*****
*****

if not(@firstchoice is null)
BEGIN

SELECT @oldz1 = FirstChoiceScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @firstchoice and CalendarYear = @year

SET @z1 = @z1formula + @oldz1

UPDATE tnpSchieffelinHistory
SET FirstChoiceScore = @z1
WHERE EmployeeID = @firstchoice and CalendarYear = @year

```

```

END

--
*****
*****
--Second Choice Score
--
*****
*****

if not(@secondchoice is null)
BEGIN

SELECT @oldz2 = SecondChoiceScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @secondchoice and CalendarYear = @year

SET @z2 = @z2formula + @oldz2

UPDATE tnpSchieffelinHistory
SET SecondChoiceScore = @z2
WHERE EmployeeID = @secondchoice and CalendarYear = @year

END

--
*****
*****
--Third Choice Score
--
*****
*****

if not(@thirdchoice is null)
BEGIN

SELECT @oldz3 = ThirdChoiceScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @thirdchoice and CalendarYear = @year

SET @z3 = @z3formula + @oldz3

UPDATE tnpSchieffelinHistory
SET ThirdChoiceScore = @z3
WHERE EmployeeID = @thirdchoice and CalendarYear = @year

```

```

END

--
*****
*****
--Selected on Ballot Score
--
*****
*****

if ((not(@s1 is null) and ((not(@s1 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s1 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s1 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsell1 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s1 and CalendarYear = @year

SET @sell1 = @z4formula + @oldsell1

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sell1
WHERE EmployeeID = @s1 and CalendarYear = @year

END

--
*****
*****

if ((not(@s2 is null) and ((not(@s2 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s2 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s2 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel2 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s2 and CalendarYear = @year

SET @sel2 = @z4formula + @oldsel2

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel2

```

```

WHERE EmployeeID = @s2 and CalendarYear = @year

END

--
*****

if ((not(@s3 is null) and ((not(@s3 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s3 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s3 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel3 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s3 and CalendarYear = @year

SET @sel3 = @z4formula + @oldsel3

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel3
WHERE EmployeeID = @s3 and CalendarYear = @year

END

--
*****

if ((not(@s4 is null) and ((not(@s4 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s4 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s4 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel4 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s4 and CalendarYear = @year

SET @sel4 = @z4formula + @oldsel4

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel4
WHERE EmployeeID = @s4 and CalendarYear = @year

```

```

END

--
*****
*****

if ((not(@s5 is null) and ((not(@s5 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s5 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s5 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel5 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s5 and CalendarYear = @year

SET @sel5 = @z4formula + @oldsel5

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel5
WHERE EmployeeID = @s5 and CalendarYear = @year

END

--
*****
*****

if ((not(@s6 is null) and ((not(@s6 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s6 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s6 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel6 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s6 and CalendarYear = @year

SET @sel6 = @z4formula + @oldsel6

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel6
WHERE EmployeeID = @s6 and CalendarYear = @year

END

```



```

--
*****
*****

if ((not(@s7 is null) and ((not(@s7 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s7 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s7 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel7 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s7 and CalendarYear = @year

SET @sel7 = @z4formula + @oldsel7

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel7
WHERE EmployeeID = @s7 and CalendarYear = @year

END

--
*****
*****

if ((not(@s8 is null) and ((not(@s8 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s8 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s8 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel8 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s8 and CalendarYear = @year

SET @sel8 = @z4formula + @oldsel8

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel8
WHERE EmployeeID = @s8 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s9 is null) and ((not(@s9 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s9 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s9 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel9 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s9 and CalendarYear = @year

SET @sel9 = @z4formula + @oldsel9

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel9
WHERE EmployeeID = @s9 and CalendarYear = @year

END

--
*****
*****

if ((not(@s10 is null) and ((not(@s10 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s10 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s10 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel10 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s10 and CalendarYear = @year

SET @sel10 = @z4formula + @oldsel10

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel10
WHERE EmployeeID = @s10 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s11 is null) and ((not(@s11 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s11 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s11 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel11 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s11 and CalendarYear = @year

SET @sel11 = @z4formula + @oldsel11

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel11
WHERE EmployeeID = @s11 and CalendarYear = @year

END

--
*****
*****

if ((not(@s12 is null) and ((not(@s12 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s12 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s12 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel12 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s12 and CalendarYear = @year

SET @sel12 = @z4formula + @oldsel12

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel12
WHERE EmployeeID = @s12 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s13 is null) and ((not(@s13 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s13 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s13 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel13 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s13 and CalendarYear = @year

SET @sel13 = @z4formula + @oldsel13

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel13
WHERE EmployeeID = @s13 and CalendarYear = @year

END

--
*****
*****

if ((not(@s14 is null) and ((not(@s14 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s14 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s14 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel14 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s14 and CalendarYear = @year

SET @sel14 = @z4formula + @oldsel14

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel14
WHERE EmployeeID = @s14 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s15 is null) and ((not(@s15 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s15 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s15 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel15 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s15 and CalendarYear = @year

SET @sel15 = @z4formula + @oldsel15

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel15
WHERE EmployeeID = @s15 and CalendarYear = @year

END

--
*****
*****

if ((not(@s16 is null) and ((not(@s16 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s16 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s16 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel16 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s16 and CalendarYear = @year

SET @sel16 = @z4formula + @oldsel16

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel16
WHERE EmployeeID = @s16 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s17 is null) and ((not(@s17 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s17 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s17 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel17 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s17 and CalendarYear = @year

SET @sel17 = @z4formula + @oldsel17

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel17
WHERE EmployeeID = @s17 and CalendarYear = @year

END

--
*****
*****

if ((not(@s18 is null) and ((not(@s18 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s18 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s18 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel18 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s18 and CalendarYear = @year

SET @sel18 = @z4formula + @oldsel18

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel18
WHERE EmployeeID = @s18 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s19 is null) and ((not(@s19 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s19 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s19 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel19 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s19 and CalendarYear = @year

SET @sel19 = @z4formula + @oldsel19

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel19
WHERE EmployeeID = @s19 and CalendarYear = @year

END

--
*****
*****

if ((not(@s20 is null) and ((not(@s20 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s20 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s20 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel20 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s20 and CalendarYear = @year

SET @sel20 = @z4formula + @oldsel20

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel20
WHERE EmployeeID = @s20 and CalendarYear = @year

END

```

```

--
*****
*****

if ((not(@s21 is null) and ((not(@s21 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s21 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s21 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel21 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s21 and CalendarYear = @year

SET @sel21 = @z4formula + @oldsel21

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel21
WHERE EmployeeID = @s21 and CalendarYear = @year

END

--
*****
*****

if ((not(@s22 is null) and ((not(@s22 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s22 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s22 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel22 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s22 and CalendarYear = @year

SET @sel22 = @z4formula + @oldsel22

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel22
WHERE EmployeeID = @s22 and CalendarYear = @year

END

```



```

--
*****
*****

if ((not(@s23 is null) and ((not(@s23 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s23 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s23 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel23 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s23 and CalendarYear = @year

SET @sel23 = @z4formula + @oldsel23

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel23
WHERE EmployeeID = @s23 and CalendarYear = @year

END

--
*****
*****

if ((not(@s24 is null) and ((not(@s24 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s24 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s24 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel24 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s24 and CalendarYear = @year

SET @sel24 = @z4formula + @oldsel24

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel24
WHERE EmployeeID = @s24 and CalendarYear = @year

END

```

```
--
*****
*****

if ((not(@s25 is null) and ((not(@s25 = @firstchoice)) or
(@firstchoice is null)) and ((not(@s25 = @secondchoice)) or
(@secondchoice is null)) and ((not(@s25 = @thirdchoice)) or
(@thirdchoice is null))))
BEGIN

SELECT @oldsel25 = SelectedOnBallotScore
FROM tnpSchieffelinHistory
WHERE EmployeeID = @s25 and CalendarYear = @year

SET @sel25 = @z4formula + @oldsel25

UPDATE tnpSchieffelinHistory
SET SelectedOnBallotScore = @sel25
WHERE EmployeeID = @s25 and CalendarYear = @year

END

fetch schieffscore into
@firstchoice,@secondchoice,@thirdchoice,@s1,@s2,@s3,@s4,@s5
,@s6,@s7,@s8,@s9,@s10,@s11,@s12,@s13,@s14,@s15,@s16,@s17,@s
18,@s19,@s20,
@s21,@s22,@s23,@s24,@s25,@kvalue
END
close schieffscore
deallocate schieffscore
END

GO
```

K. SCHIEFFELIN SEGMENTS

This stored procedure determines the number of course segments professors taught during the award period.

```
CREATE PROCEDURE snpSchieffSegments
    @year int

AS
declare @year1 int
SET @year1 = @year + 1
```

--Make faculty ineligible that have taught <3 segments

```
UPDATE tnpSchieffelinHistory
SET IsEligible = 'N'
WHERE (tnpSchieffelinHistory.CalendarYear = @year) AND
(tnpSchieffelinHistory.EmployeeID IN (
SELECT dbo.tnpFacultyClass.EmployeeID
FROM dbo.tnpFacultyClass INNER JOIN
dbo.tnpClass ON dbo.tnpFacultyClass.Segment =
dbo.tnpClass.Segment AND dbo.tnpFacultyClass.CourseID =
dbo.tnpClass.CourseID AND
dbo.tnpFacultyClass.Quarter = dbo.tnpClass.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.tnpClass.[Year] INNER JOIN
dbo.tnpClassType ON dbo.tnpClass.ClassTypeID =
dbo.tnpClassType.ClassTypeID INNER JOIN
dbo.vnpFacultyCount ON dbo.tnpFacultyClass.Segment =
dbo.vnpFacultyCount.Segment AND
dbo.tnpFacultyClass.CourseID = dbo.vnpFacultyCount.CourseID
AND dbo.tnpFacultyClass.Quarter =
dbo.vnpFacultyCount.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.vnpFacultyCount.[Year]
WHERE (NOT (dbo.tnpClassType.ClassType = 'Directed Study'
OR
dbo.tnpClassType.ClassType = 'Thesis' OR
dbo.tnpClassType.ClassType = 'Seminar')) AND
(dbo.tnpFacultyClass.IsLabInstructor = 'Y') AND
(dbo.tnpFacultyClass.IsLectureInstructor = 'Y')
AND ((tnpFacultyClass.[Year] = @year AND
((tnpFacultyClass.[Quarter] = 2) OR
(tnpFacultyClass.[Quarter] = 3) OR
(tnpFacultyClass.[Quarter] = 4)))
OR ((tnpFacultyClass.[Year] = @year1) AND
(tnpFacultyClass.[Quarter] = 1)))
GROUP BY EmployeeID
HAVING (COUNT(*) < 3)))
```

--This accounts for if still onboard, but no classes taught that year.

```
UPDATE tnpSchieffelinHistory
SET IsEligible = 'N'
WHERE (tnpSchieffelinHistory.EmployeeID NOT IN
(SELECT EmployeeID FROM tnpFacultyClass
WHERE ((tnpFacultyClass.[Year] = @year AND
((tnpFacultyClass.[Quarter] = 2) OR
```

```
(tnpFacultyClass.[Quarter] = 3) OR  
(tnpFacultyClass.[Quarter] = 4)))  
OR ((tnpFacultyClass.[Year] = @year1) AND  
(tnpFacultyClass.[Quarter] = 1))))  
GO
```

L. VIEW ELIGIBLE FACULTY

This stored procedure is used to select eligible faculty from the view "vnpEligibleFaculty" for displaying on the dynamic ballot (ballot.asp).

```
CREATE PROCEDURE snpViewEligibleFaculty  
    @year int  
  
AS  
  
SELECT * FROM vnpEligibleFaculty  
WHERE CalendarYear = @year  
GO
```

APPENDIX M. VIEWS

This appendix contains the code for all the views used in the SADBMS.

A. ELIGIBLE FACULTY

This view builds a logical table consisting of all eligible faculty. It is used for building the ballot (ballot.asp).

```
CREATE VIEW dbo.vnpEligibleFaculty
AS
SELECT dbo.tnpDepartment.Department,
       dbo.tnpDepartment.Description, dbo.tnpEmployee.FirstName,
       dbo.tnpEmployee.LastName,
           dbo.tnpSchieffelinHistory.CalendarYear,
       dbo.tnpEmployee.EmployeeID, dbo.tnpEmployee.MiddleName,

       dbo.tnpDepartment_Employee.IsPrimaryDepartment,
       dbo.tnpDepartment.DepartmentID,
       dbo.tnpSchieffelinHistory.IsEligible
FROM   dbo.tnpEmployee INNER JOIN
           dbo.tnpFaculty ON dbo.tnpEmployee.EmployeeID
= dbo.tnpFaculty.EmployeeID INNER JOIN
           dbo.tnpDepartment_Employee ON
       dbo.tnpEmployee.EmployeeID =
       dbo.tnpDepartment_Employee.EmployeeID INNER JOIN
           dbo.tnpDepartment ON
       dbo.tnpDepartment_Employee.DepartmentID =
       dbo.tnpDepartment.DepartmentID INNER JOIN
           dbo.tnpSchieffelinHistory ON
       dbo.tnpFaculty.EmployeeID =
       dbo.tnpSchieffelinHistory.EmployeeID
WHERE  (dbo.tnpSchieffelinHistory.IsEligible = 'Y') AND
       (dbo.tnpDepartment_Employee.IsPrimaryDepartment = 'Y')
```

B. FACULTY COUNT

This view creates a logical table of all courses that have lab and lecture instruction. It is used in computing professor's eligibility.

```

CREATE VIEW vnpFacultyCount
AS
SELECT Quarter, CourseID, Segment, [Year], COUNT(*) AS
Instructors
FROM dbo.tnpFacultyClass
GROUP BY Quarter, CourseID, Segment, [Year],
IsLabInstructor, IsLectureInstructor
HAVING (IsLabInstructor = 'Y') AND (IsLectureInstructor =
'Y')

CREATE VIEW vnpFacultyHours
AS
SELECT dbo.tnpFacultyClass.EmployeeID,
dbo.tnpFacultyClass.[Year], dbo.tnpClass.AssignedLabHours,
dbo.tnpClass.AssignedLectureHours,
dbo.tnpClassType.ClassType, dbo.tnpFacultyClass.Segment,
dbo.tnpFacultyClass.CourseID, dbo.tnpFacultyClass.Quarter,
dbo.vnpFacultyCount.Instructors
FROM dbo.tnpFacultyClass INNER JOIN
dbo.tnpClass ON dbo.tnpFacultyClass.Segment =
dbo.tnpClass.Segment AND dbo.tnpFacultyClass.CourseID =
dbo.tnpClass.CourseID AND
dbo.tnpFacultyClass.Quarter = dbo.tnpClass.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.tnpClass.[Year] INNER JOIN
dbo.tnpClassType ON dbo.tnpClass.ClassTypeID =
dbo.tnpClassType.ClassTypeID INNER JOIN
dbo.vnpFacultyCount ON dbo.tnpFacultyClass.Segment =
dbo.vnpFacultyCount.Segment AND
dbo.tnpFacultyClass.CourseID = dbo.vnpFacultyCount.CourseID
AND dbo.tnpFacultyClass.Quarter =
dbo.vnpFacultyCount.Quarter AND
dbo.tnpFacultyClass.[Year] = dbo.vnpFacultyCount.[Year]
WHERE (NOT (dbo.tnpClassType.ClassType = 'Directed Study'
OR
dbo.tnpClassType.ClassType = 'Thesis' OR
dbo.tnpClassType.ClassType = 'Seminar')) AND
(dbo.tnpFacultyClass.IsLabInstructor = 'Y') AND
(dbo.tnpFacultyClass.IsLectureInstructor = 'Y')

```

C. FIRST, SECOND & THIRD CHOICE

These three views are used in computing professor scores. They create logical tables of the number of first second or third place votes a professor receives for all ballots submitted (Z1, Z2 & Z3).

```

CREATE VIEW dbo.vnpFirstChoice
AS
SELECT FirstChoice, COUNT(*) AS FirstChoiceVotes,
CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY FirstChoice, CalendarYear
HAVING (NOT (FirstChoice IS NULL))
*****

CREATE VIEW dbo.vnpSecondChoice
AS
SELECT SecondChoice, COUNT(*) AS SecondChoiceVotes,
CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY SecondChoice, CalendarYear
HAVING (NOT (SecondChoice IS NULL))
*****

CREATE VIEW dbo.vnpThirdChoice
AS
SELECT ThirdChoice, COUNT(*) AS ThirdChoiceVotes,
CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY ThirdChoice, CalendarYear
HAVING (NOT (ThirdChoice IS NULL))

```

D. NUMBER OF SUPPORT STATEMENTS

This view creates a logical table of the number of support statements (D) professors receive when they are a first choice candidate on a ballot.

```

CREATE VIEW dbo.vnpNumberSupportStatements
AS

SELECT FirstChoice, COUNT(*) AS NumberStatements,
CalendarYear

FROM   dbo.tnpSchieffelinBallot
WHERE  (HasStatement = 'Y')
GROUP BY FirstChoice, CalendarYear
HAVING (NOT (FirstChoice IS NULL))

```

E. SCHIEFFELIN CURRIC ID

This view creates a logical table of curriculum ID's and the departments they belong to. It is used to collect demographic information about the voter on the 3rd balloting page (ballot3.asp).

```
CREATE VIEW dbo.vnpSchiefCurricID
AS
SELECT DISTINCT
        dbo.tnpSchieffelinBallot.CurriculumID,
        dbo.tnpCurriculum.CurricID, dbo.tnpCurriculum.CurricNumber,
        dbo.tnpCurriculum.CurricShortName,
        dbo.tnpCurriculum.DepartmentID
FROM  dbo.tnpCurriculum INNER JOIN
        dbo.tnpSchieffelinBallot ON
        dbo.tnpCurriculum.DepartmentID =
        dbo.tnpSchieffelinBallot.DeptID
```

F. SCHIEFFELIN RANKING

This view creates a logical table used for ranking the candidates and used for displaying the history reports.

```
CREATE VIEW dbo.vnpSchieffelinRanking
AS
SELECT EmployeeID, CalendarYear, FirstChoiceScore,
        SecondChoiceScore, ThirdChoiceScore, SelectedOnBallotScore,
        FinalScore, Rank, EligibilityCode,
        NBallots, NumberSupportStatements, PCRank,
        FirstChoiceVotes, SecondChoiceVotes, ThirdChoiceVotes,
        PCScore
FROM  dbo.tnpSchieffelinHistory
WHERE (NOT (FinalScore IS NULL))
```

G. SCHIEFFELIN FIRST,SECOND & THIRD CHOICE

The next three views are used for constructing logical tables for displaying the voters top 3 choices on ballot3.asp.

```
CREATE VIEW dbo.vnpSchiefFirstChoice
```



```

AS
SELECT TOP 100 PERCENT dbo.vnpEligibleFaculty.EmployeeID,
dbo.tnpSchieffelinBallot.FirstChoice,
dbo.tnpSchieffelinBallot.BallotID,
        dbo.vnpEligibleFaculty.LastName,
dbo.vnpEligibleFaculty.FirstName,
dbo.vnpEligibleFaculty.MiddleName,
dbo.vnpEligibleFaculty.Description
FROM    dbo.tnpSchieffelinBallot INNER JOIN
        dbo.vnpEligibleFaculty ON
dbo.tnpSchieffelinBallot.FirstChoice =
dbo.vnpEligibleFaculty.EmployeeID
ORDER BY dbo.tnpSchieffelinBallot.BallotID
*****

```

```

CREATE VIEW dbo.vnpSchieffSecondChoice
AS
SELECT TOP 100 PERCENT dbo.vnpEligibleFaculty.EmployeeID,
dbo.tnpSchieffelinBallot.BallotID,
dbo.vnpEligibleFaculty.LastName,
        dbo.vnpEligibleFaculty.FirstName,
dbo.vnpEligibleFaculty.Description,
dbo.vnpEligibleFaculty.MiddleName,
        dbo.tnpSchieffelinBallot.SecondChoice
FROM    dbo.tnpSchieffelinBallot INNER JOIN
        dbo.vnpEligibleFaculty ON
dbo.tnpSchieffelinBallot.SecondChoice =
dbo.vnpEligibleFaculty.EmployeeID
ORDER BY dbo.tnpSchieffelinBallot.BallotID
*****

```

```

CREATE VIEW dbo.vnpSchieffThirdChoice
AS
SELECT TOP 100 PERCENT dbo.vnpEligibleFaculty.EmployeeID,
dbo.tnpSchieffelinBallot.ThirdChoice,
dbo.tnpSchieffelinBallot.BallotID,
        dbo.vnpEligibleFaculty.LastName,
dbo.vnpEligibleFaculty.FirstName,
dbo.vnpEligibleFaculty.Description,
dbo.vnpEligibleFaculty.MiddleName
FROM    dbo.tnpSchieffelinBallot INNER JOIN
        dbo.vnpEligibleFaculty ON
dbo.tnpSchieffelinBallot.ThirdChoice =
dbo.vnpEligibleFaculty.EmployeeID
ORDER BY dbo.tnpSchieffelinBallot.BallotID

```

H. SCHIEFFELIN HISTORY

This view creates a logical table used in displaying the chairman and committee history reports on history.asp and history1.asp

```
CREATE VIEW dbo.vnpSchiefHistory
AS
SELECT TOP 100 PERCENT dbo.tnpSchieffelinHistory.NBallots,
dbo.tnpSchieffelinHistory.NumberSupportStatements,
        dbo.tnpSchieffelinHistory.EligibilityCode,
dbo.tnpSchieffelinHistory.FinalScore,
dbo.tnpSchieffelinHistory.PCScore,
dbo.tnpEmployee.LastName,
        dbo.tnpEmployee.FirstName,
dbo.tnpEmployee.MiddleName,
dbo.tnpSchieffelinHistory.CalendarYear,
dbo.vnpSchieffelinRanking.Rank,
        dbo.tnpEmployee.EmployeeID,
dbo.tnpSchieffelinHistory.PCRank
FROM   dbo.tnpSchieffelinHistory INNER JOIN
        dbo.vnpSchieffelinRanking ON
dbo.tnpSchieffelinHistory.EmployeeID =
dbo.vnpSchieffelinRanking.EmployeeID INNER JOIN
        dbo.tnpEmployee ON
dbo.tnpSchieffelinHistory.EmployeeID =
dbo.tnpEmployee.EmployeeID
ORDER BY dbo.vnpSchieffelinRanking.Rank,
dbo.tnpSchieffelinHistory.CalendarYear DESC
```

I. SELECT 1-25

These views are used by the scoring stored procedure. The count votes (snpCountVotes) counts the number of ballots each professor was selected on (Z4) for all ballots submitted.

```
CREATE VIEW dbo.vnpSelect1
AS
SELECT Select1, COUNT(*) AS Vote1, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select1, CalendarYear
HAVING (NOT (Select1 IS NULL))
```

```

CREATE VIEW dbo.vnpSelect2
AS
SELECT Select2, COUNT(*) AS Vote2, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select2, CalendarYear
HAVING (NOT (Select2 IS NULL))

CREATE VIEW dbo.vnpSelect3
AS
SELECT Select4, COUNT(*) AS Vote4, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select4, CalendarYear
HAVING (NOT (Select4 IS NULL))

CREATE VIEW dbo.vnpSelect5
AS
SELECT Select5, COUNT(*) AS Vote5, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select5, CalendarYear
HAVING (NOT (Select5 IS NULL))

CREATE VIEW dbo.vnpSelect6
AS
SELECT Select6, COUNT(*) AS Vote6, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select6, CalendarYear
HAVING (NOT (Select6 IS NULL))

CREATE VIEW dbo.vnpSelect7
AS
SELECT Select7, COUNT(*) AS Vote7, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select7, CalendarYear
HAVING (NOT (Select7 IS NULL))

CREATE VIEW dbo.vnpSelect8
AS
SELECT Select8, COUNT(*) AS Vote8, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select8, CalendarYear
HAVING (NOT (Select8 IS NULL))

CREATE VIEW dbo.vnpSelect9
AS
SELECT Select9, COUNT(*) AS Vote9, CalendarYear

```

```

FROM  dbo.tnpSchieffelinBallot
GROUP BY Select9, CalendarYear
HAVING (NOT (Select9 IS NULL))

CREATE VIEW dbo.vnpSelect10
AS
SELECT Select10, COUNT(*) AS Vote10, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select10, CalendarYear
HAVING (NOT (Select10 IS NULL))

CREATE VIEW dbo.vnpSelect11
AS
SELECT Select11, COUNT(*) AS Vote11, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select11, CalendarYear
HAVING (NOT (Select11 IS NULL))

CREATE VIEW dbo.vnpSelect12
AS
SELECT Select12, COUNT(*) AS Vote12, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select12, CalendarYear
HAVING (NOT (Select12 IS NULL))

CREATE VIEW dbo.vnpSelect13
AS
SELECT Select13, COUNT(*) AS Vote13, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select13, CalendarYear
HAVING (NOT (Select13 IS NULL))

CREATE VIEW dbo.vnpSelect14
AS
SELECT Select14, COUNT(*) AS Vote14, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select14, CalendarYear
HAVING (NOT (Select14 IS NULL))

CREATE VIEW dbo.vnpSelect15
AS
SELECT Select15, COUNT(*) AS Vote15, CalendarYear
FROM  dbo.tnpSchieffelinBallot
GROUP BY Select15, CalendarYear
HAVING (NOT (Select15 IS NULL))

```

```

CREATE VIEW dbo.vnpSelect16
AS
SELECT Select16, COUNT(*) AS Vote16, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select16, CalendarYear
HAVING (NOT (Select16 IS NULL))

CREATE VIEW dbo.vnpSelect17
AS
SELECT Select17, COUNT(*) AS Vote17, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select17, CalendarYear
HAVING (NOT (Select17 IS NULL))

CREATE VIEW dbo.vnpSelect18
AS
SELECT Select18, COUNT(*) AS Vote18, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select18, CalendarYear
HAVING (NOT (Select18 IS NULL))

CREATE VIEW dbo.vnpSelect19
AS
SELECT Select19, COUNT(*) AS Vote19, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select19, CalendarYear
HAVING (NOT (Select19 IS NULL))

CREATE VIEW dbo.vnpSelect20
AS
SELECT Select20, COUNT(*) AS Vote20, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select20, CalendarYear
HAVING (NOT (Select20 IS NULL))

CREATE VIEW dbo.vnpSelect21
AS
SELECT Select21, COUNT(*) AS Vote21, CalendarYear
FROM   dbo.tnpSchieffelinBallot
GROUP BY Select21, CalendarYear
HAVING (NOT (Select21 IS NULL))

CREATE VIEW dbo.vnpSelect22
AS
SELECT Select22, COUNT(*) AS Vote22, CalendarYear
FROM   dbo.tnpSchieffelinBallot

```

```
GROUP BY Select22, CalendarYear  
HAVING (NOT (Select22IS NULL))
```

```
CREATE VIEW dbo.vnpSelect23  
AS  
SELECT Select23, COUNT(*) AS Vote23, CalendarYear  
FROM   dbo.tnpSchieffelinBallot  
GROUP BY Select23, CalendarYear  
HAVING (NOT (Select23 IS NULL))
```

```
CREATE VIEW dbo.vnpSelect24  
AS  
SELECT Select24, COUNT(*) AS Vote24, CalendarYear  
FROM   dbo.tnpSchieffelinBallot  
GROUP BY Select24, CalendarYear  
HAVING (NOT (Select24 IS NULL))
```

```
CREATE VIEW dbo.vnpSelect25  
AS  
SELECT Select25, COUNT(*) AS Vote25, CalendarYear  
FROM   dbo.tnpSchieffelinBallot  
GROUP BY Select25, CalendarYear  
HAVING (NOT (Select25 IS NULL))
```

APPENDIX N. ELIGIBILITY REPORTS

These are the screen shots of Web pages used for viewing eligible professors.

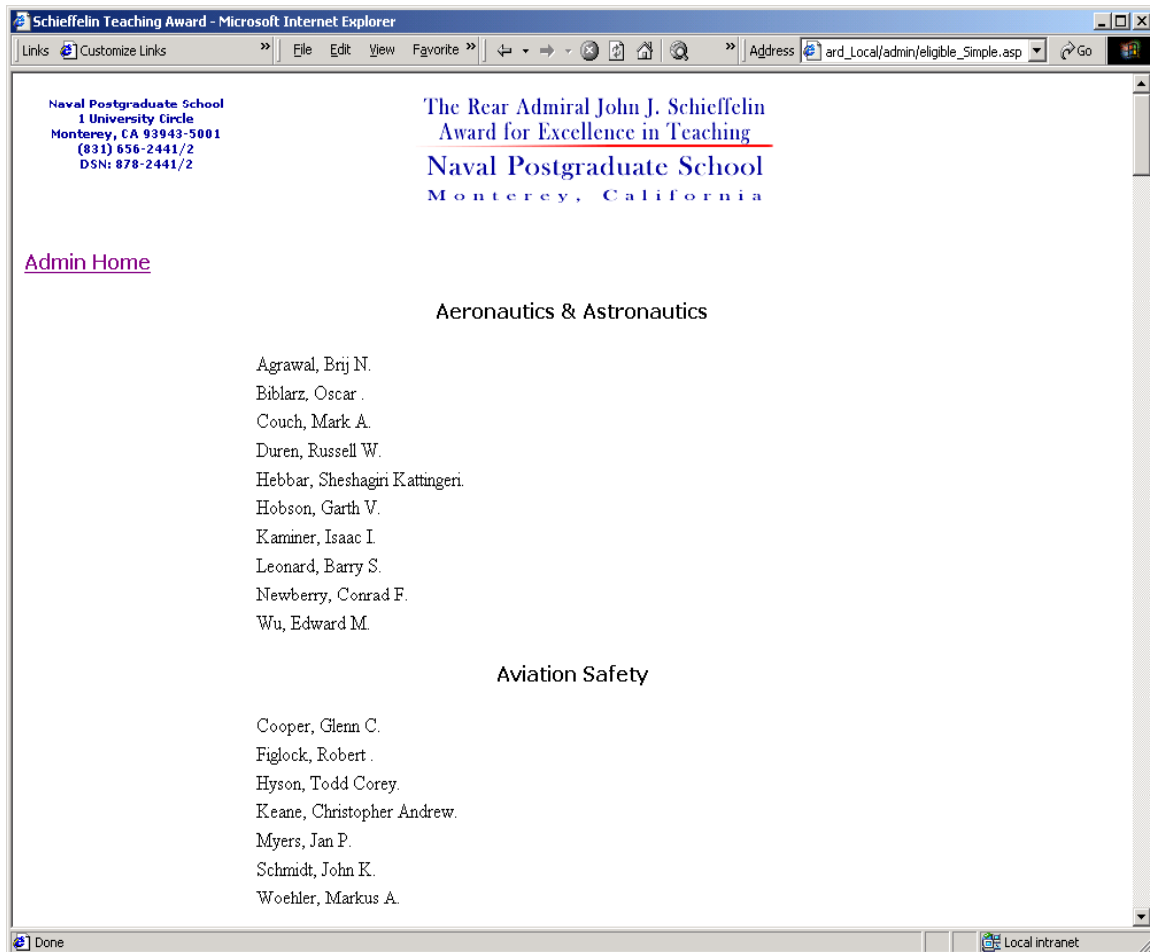


Figure 27. Eligible_simple.asp (Eligibility Report)

A simple view of eligible professors in order of department.

Schieffelin Teaching Award - Microsoft Internet Explorer

rd_Local/admin/eligible_detailed.asp

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

Aeronautics & Astronautics	Segment	Qtr	LecHrs	LabHrs	#Instr	Year	Type
<i>Agrawal, Brij N.</i>							
AA4871		2	2	2	1	2000	Resident
AA3811		3	1	2	1	2000	Resident
AA4816		3	4	0	1	2000	Resident
AA4871		4	2	2	1	2000	Resident
<i>Biblarz, Oscar.</i>							
AA2043		2	3	2	1	2000	Resident
AA4506	1	3	4	0	1	2000	Resident
AA3851		4	3	2	1	2000	Resident
AA4505		4	3	2	1	2000	Resident
<i>Couch, Mark A.</i>							
AA3202		1	3	2	1	2001	Resident
AA2440		2	3	2	2	2000	Resident
AA3251		2	4	1	1	2000	Resident
AA2339		3	3	2	1	2000	Resident
AA2440		4	3	2	1	2000	Resident
AA3251		4	4	1	1	2000	Resident
AA3250	1	4	3	0	1	2000	Resident
AA2440	A	4	3	2	1	2000	Resident
AA2440	B	4	3	2	1	2000	Resident

Done Local intranet

Figure 28. Eligible_detailed.asp (Eligibility Report)

A detailed view of eligible professors and all of the courses they taught during the award period.

APPENDIX O. EDIT PROFESSOR ELIGIBILITY

These are the Web pages used in editing professor award eligibility.

The screenshot shows a Microsoft Internet Explorer window titled "Schieffelin Teaching Award - Microsoft Internet Explorer". The address bar shows "http://accelerate/Awar". The page content includes the Naval Postgraduate School logo and contact information on the left, and the award title "The Rear Admiral John J. Schieffelin Award for Excellence in Teaching" on the right. Below the award title is a link "Admin Home" and a form with the label "Enter Year:" followed by a text input field and a "Submit" button.

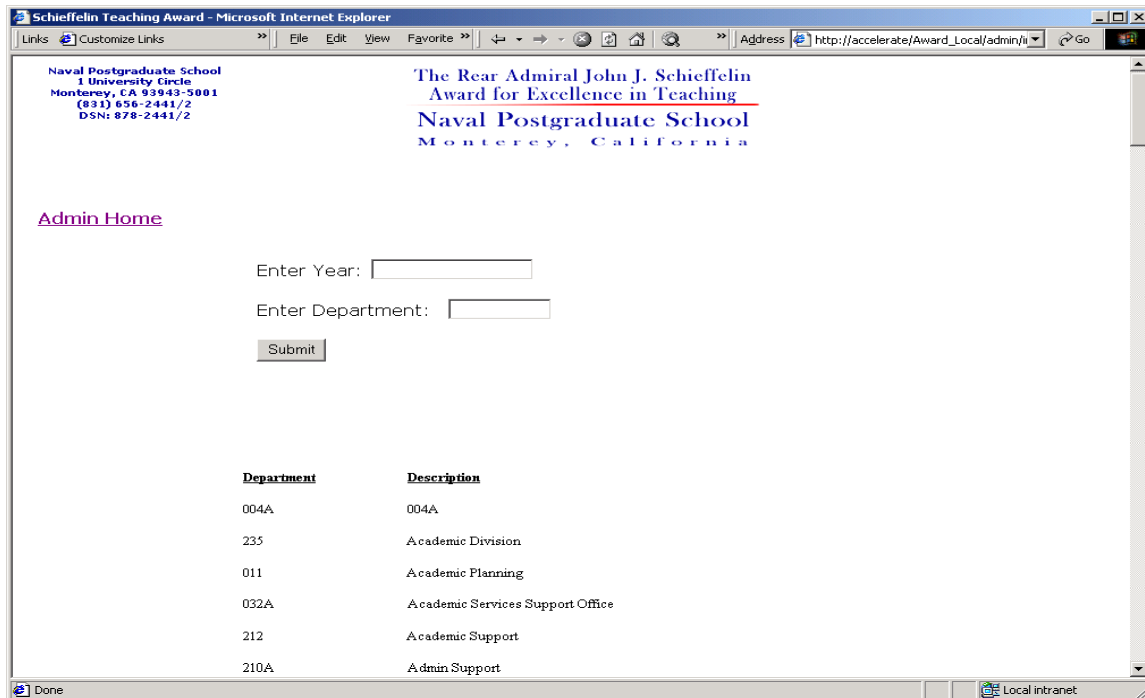
Figure 29. Input_form4.

This search form finds all professors in a given year.

The screenshot shows the same Microsoft Internet Explorer window as Figure 29, but the form is now for data entry. It includes the same header information. Below the "Admin Home" link, there are several labeled text input fields: "Department:" (Academic Division), "EmployeeID:" (98839), "Last Name:" (Kersh), "First Name:" (Kevin), "Middle Name:" (G), "Year:" (2000), and "Eligible:" (N). At the bottom of the form are four navigation buttons: "<|", "<", ">", and ">|".

Figure 30. Data_entry_eligibility_all.asp

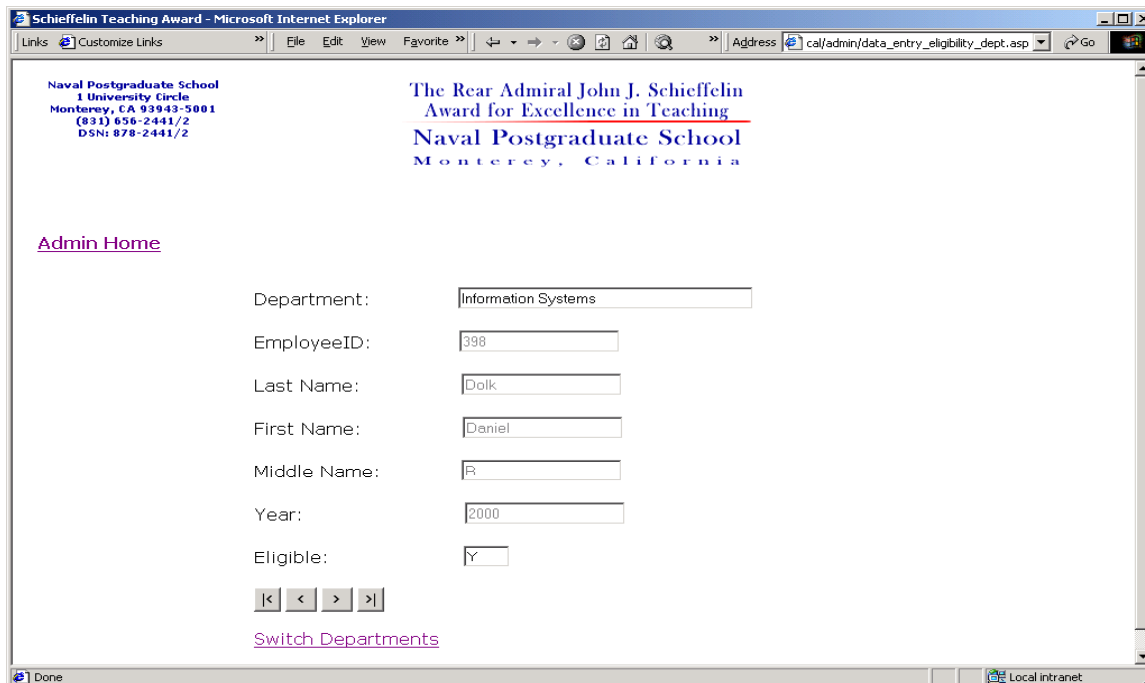
This form is used to edit eligibility by all department.



Department	Description
004A	004A
235	Academic Division
011	Academic Planning
032A	Academic Services Support Office
212	Academic Support
210A	Admin Support

Figure 31. Input_form5.

This search form is used to find professors by department in a given year.



Department: Information Systems

EmployeeID: 398

Last Name: Dolk

First Name: Daniel

Middle Name: R

Year: 2000

Eligible: Y

Navigation: [Left] [Right] [First] [Last]

[Switch Departments](#)

Figure 32. Data_entry_eligibility_dept.

This form is used to edit professor eligibility by department.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

Enter Year:

Enter Last Name:

Figure 33. Input_form11.

This search form finds professors by last name.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

Department:

EmployeeID:

Last Name:

First Name:

Middle Name:

Year:

Eligible:

[Search another Last Name](#)

Figure 34. lname.asp.

This form is used to edit eligibility by last name.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX P. INPUT PROFESSOR HISTORY

Web page forms used for inputting professor history.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

Enter Year:

Enter Last Name:

Figure 35. Input_form3.

This search form finds professors by last name for the given year.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

[Admin Home](#)

ID:

Last Name:

First Name:

Middle Name:

Eligibility Code:

Year:

Press Navbar button to update record.

[Add History for another Professor](#)

Figure 36. Data_entry_history.asp

This form is used to edit the eligibility code for updating professor history.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX Q. VIEWING AND EDITING SCORING WEIGHTS

These Web pages are used for viewing and editing the scoring weights.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

Admin Home
Edit Weights

These are the weights currently used in calculating the Schieffelin Scores.

W ₁	W ₂	W ₃	W ₄	PValue
7	4	3	1	0.9

Figure 37. Weights.asp (View Weights)

This Web page is used to view the current weights.

Naval Postgraduate School
1 University Circle
Monterey, CA 93943-5001
(831) 656-2441/2
DSN: 878-2441/2

The Rear Admiral John J. Schieffelin
Award for Excellence in Teaching
Naval Postgraduate School
Monterey, California

Admin Home
View Weights

FirstChoiceWeight:
SecondChoiceWeight:
ThirdChoiceWeight:
SelectedWeight:
PValue:
CalYear:

Figure 38. Weights.asp (Edit Weights).

This form is used to change the weights.

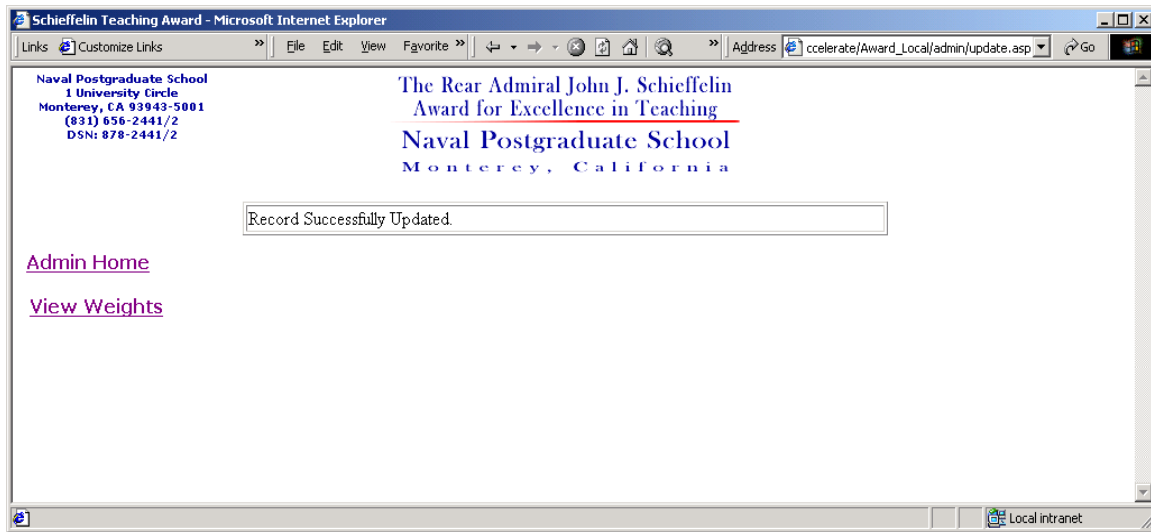


Figure 39. Update.asp.

This form informs the user that the weights were updated.

LIST OF REFERENCES

Barker, Jacquie, *Beginning Java Objects: From Concepts to Code*, Wrox Press Ltd. 2000.

Barman, Dilip, *Dilip's Brief Introduction to Relational Databases*, Department of Computer Science, University of North Carolina, <http://www.cs.unc.edu/Courses/wwwp-s98/members/barman/databaseLesson/>, January 1998.

Brenton, Chris, *Mastering Network Security*, Network Press, 1999.

Brooks, Fredrick P., *The Mythical-Man Month, Anniversary Edition: Essays on Software Engineering*, Addison Wesley, 1995.

Caroland, James, and others, "The Future of NPS Education Management," paper presented to students at the Naval Postgraduate School, Monterey, California, 5 April 2001.

DeAngelis, Carla, *Data Modeling with Erwin: The Authoritive Solution*, Sams Publishing, 2000.

Eden, Timothy, *Active Server Pages: In Plain English*, IDG Book World, 2001.

Espisito, Dino, *Professional ASP Data Access*, Wrox Ltd., 2000.

Gavurin, Stuart L. "Where does prototyping fit in IS development?" *Journal of Systems Management* 42 (1991)

Harrington, James H., *Business Process Improvement Workbook*, McGraw-Hill, 1997.

Harrington, James H., *Business Process Improvement: The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness*, McGraw-Hill, Inc. 1991.

Harrington, Jan L., *Relational Database Design: Clearly Explained*, Academic Press, 1998.

Homer, Alex, et al., *Professional Active Server Pages 3.0*, Wrox Press Ltd., 1999.

Kauffman, John, *Beginning ASP Databases*, Wrox Press Ltd. 1999.

Laudon, Kenneth C. and Jane P. Laudon., *Essentials of Management Information Systems*. Prentice Hall, 1999.

Mitchell, Scott, *Sams Teach Yourself: Active Server Pages 3.0 in 21 Days*, Sams Publishing, 2000.

Mullins, Craig S., "SQL Server Update," *Using Views in Microsoft SQL Server*, Craig S. Mullins & Associates, Inc., 1999.

O'Neil, Joseph, *Visual InterDev 6: From the Ground Up*, Osborne/McGraw-Hill, 1998.

Pfleeger, Charles P., *Security in Computing*, Second Edition, Prentice Hall, 1996.

Rahmel, Dan, "Database Security," *Internet Systems*, April 1997

Read, Robert R., *Data Analysis of the Teaching Award Ballots*, Master's Thesis, Naval Postgraduate School, Monterey, California, July 1979.

Scango, Patsy D., *The Third Annual Award for Excellence in Teaching an Analysis of Procedure and Data*, Master's Thesis, Naval Postgraduate School, Monterey, California, June 1972.

Scherkenbach, 1991

Schneiderman, B., *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, 1997.

Setten, William, *Microsoft Windows NT Network Administration: Academic Learning Series*, Microsoft Press, 1998.

Smith, Horace W., *Database Design and Implementation for USMC Manpower Retention Survey*, Thesis, Naval Postgraduate School, September 1999.

Strebe, Matthew, *MCSE: Internet Information Server 4 Study Guide*, Third Edition, Network Press, 2000.

Teorey, Toby J., *Database Modeling & Design: The Fundamental Principles*, Second Edition, Morgan Kaufmann Publishers, Inc. 1996.

Thurrott, Paul, et al., *Visual InterDev 6*, Sams Publishing, 1999.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Fort Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Marine Corps Representative
Naval Postgraduate School
Monterey, California
debarber@nps.navy.mil
4. Director, Training and Education, MCCDC, Code C46
Quantico, Virginia
webmaster@tecom.usmc.mil
5. Director, Marine Corps Research Center, MCCDC, Code C40RC
Quantico, Virginia
ramkeyce@tecom.usmc.mil
strongka@tecom.usmc.mil
sanftlebenka@tecom.usmc.mil
6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
Camp Pendleton, California
doranfv@mctssa.usmc.mil
palanaj@mctssa.usmc.mil
7. Professor Daniel R. Dolk
Naval Postgraduate School
Monterey, CA 93943
drdolk@nps.navy.mil
8. Dennis R. Mar
Naval Postgraduate School
Monterey, CA 93943
mar@nps.navy.mil
9. Major Samuel A. Magliano
713 Porter Street
Fallbrook, CA 92028
jarhed@f2s.com