Detailed Design

Automated Grading System for Microsoft Excel Spreadsheets

Clients:
Dr. Scott Hunter
Professor - Department of Computer Science
Siena College

Ms. Jami Cotler
Professor - Department of Computer Science
Siena College

Delivered By:
Performance Software
“Custom Solutions for the Real World”
www.performancesoftware.org

Performance Software Team:
Whitney Cave
Patrick Durham
Raymond Navarette
Nicholas Sitterly
Joseph Strube

February 21st, 2006
Microsoft Excel Grading System
Preliminary Design

Table of Contents:

Section 1 – External Design Specifications

Section 1.1 – User Displays and Command Summary 3
Section 1.2 – Detailed Data Flow Diagrams 33
Section 1.3 – Logical Data Dictionary 38
Section 1.4 – Logical Data Stores 53
  Section 1.4.1 – Logical Data Stores 53
  Section 1.4.2 – ERD (Entity-Relationship Diagram) 58
Section 1.5 – Logical Format of Data Stores and Databases 59
Section 1.6 – Structure Charts 59

Section 2 – System User Specifications

Section 2.1 – User Case Scenarios 67
Section 2.2 – Functional Requirements 70

Section 3 – Testing Requirements

Section 3.1 – Project Testing 73
Section 3.2 – Test Plan 74
  Section 3.2.1 – Test Plan Identifier 74
  Section 3.2.2 – References 74
  Section 3.2.3 – Introduction 74
  Section 3.2.4 – Test Items 75
  Section 3.2.5 – Software Risk Issues 75
  Section 3.2.6 – Features to be Tested 76
  Section 3.2.7 – Approach 81
  Section 3.2.8 – Item Pass/Fail Criteria 82
  Section 3.2.9 – Suspension Criteria and Resumption Requirements 82
  Section 3.2.10 – Test Deliverables 83
  Section 3.2.11 – Testing Tasks 83
  Section 3.2.12 – Environmental Needs 84
  Section 3.2.13 – Responsibilities 84
  Section 3.2.14 – Staffing and Training Needs 84
  Section 3.2.15 – Schedule 86
  Section 3.2.16 – Risk and Contingencies 86
  Section 3.2.17 – Approvals 87
Section 3.3 – Unit Tests 89
Section 3.4 – Integration Tests 111
Section 1: External Design Specification

Section 1.1: User Displays and Command Summary

Log In Screen

This is the generic log in screen that all users of the system will interact with. The user will enter their username and password and then click on the “Log in to the AGS” to proceed to their appropriate landing page. If the user has never used the system before they will need to follow the Login Here link to create their account.
New User Account Creation

Automated Grading System  
for CSIS-010 / CSIS-011

Welcome to the AGS. Please enter your first and last name as well as your Sienna email address. The system password you were given for your lecture and lab sections is also needed to gain access to the system. The system will remember this information and you will not be asked for it again.

First Name

Last Name

Sienna Email

Password

Lecture Section: Please select your lecture section from the list below.

Lab Section: Please select your lab section from the list below.

This is where a new user will create their account by entering all of their information including a password for their lecture and lab sections that will be provided to them upon entering the course.
After logging in, the Course Coordinator will be taken to this landing page.
Course Coordinator – User Tasks (Search, Add, Batch Add)

After selecting to Add/Edit Users from the Tasks menu, the Course Coordinator will be taken to this page. The Course Coordinator has the ability to search for users and add users.
Course Coordinator – User Search Results and Editing

This page is displayed after the Course Coordinator clicks on the Search button on the User Search page. The Search Results are displayed and the Course Coordinator has the option of performing another search.
After selecting Add/Edit Sections from the Tasks menu, the Course Coordinator will be taken to this page. The Course Coordinator has the ability to edit the type (lecture or lab), the description (date and time), and the instructor for each section. The Course Coordinator also has the option of adding a section.
Course Coordinator – Assignment Category Creation and Editing

After selecting Add/Edit Categories from the Tasks menu, the Course Coordinator will have the option of removing categories from courses and also creating new categories with specific weights.
Course Coordinator – Create New Assignment Shell

After selecting Add New from the Assignments column in the Tasks bar, the Course Coordinator will be able to choose a name for the new assignment and submit it to the database.
After creating a name for the new assignment, the Course Coordinator is allowed to edit its properties. Here, the assignment description may be chosen and also its category which in turn has a corresponding weight. Below this it may be specified how many files this assignment has and their weights. In addition to this the grading files for the assignment may be submitted by clicking “Submit Grading Files”.
Course Coordinator – Grading Files Submission

Here, the Course Coordinator has selected “Submit Grading Files” from the previous screen. The number of files that the assignment contains shows up with an option for the selection of a grading Template and grading Key for each. After selecting all files the Course Coordinator may submit the files via the “Submit Files” button.
Here the course coordinator has the option to archive the current database or change the semester without updating the database and providing a new semester password for student logins.
Student – Section Identification

This screen is presented to the student the first time they attempt to login. Here they are asked to choose both their Lecture Section and Lab Section, this information is then submitted to the database.
Student Landing Page – Grade Overview

Here we see the student interface where grades for various pieces of work they have done in the course of the semester are populated from the database. In this example the “All Grades” option was selected.
Student – View Graded Assignment Details

Upon selecting a particular assignment, the student will then be able to view information on the specific files. Here we see that Lab 1 contained three files each with specific scores and weights which correspond to the student’s grade for that assignment.
Here the student has the option to view specific grade details for assignments that have not been submitted yet. This allows the student to be aware of the due dates and the weights of the various files.
**Student – Submit Assignment Files**

The student submission form is accessed by choosing to submit a particular assignment. The number of files required shows up and the student may browse their system for each one individually before submitting them.
Lab Instructor Landing Page – Lab Section Overview

The Lab Instructor is logged in here and has access to the grades of everyone from their specific course as well as from individual sections. This is simply for viewing purposes of students’ grades, and is the screen that is initially brought forth upon login.

Grade Report

Section: CSIS010 - Lab 6
Description: F, 10:25-12:25

<table>
<thead>
<tr>
<th></th>
<th>Lab 1</th>
<th>Lab 2</th>
<th>Midterm</th>
<th>Lab 3</th>
<th>Lab 4</th>
<th>Final Exam</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>LastName1, FirstName1</td>
<td>86</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>86</td>
</tr>
<tr>
<td>LastName2, FirstName2</td>
<td>93</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>93</td>
</tr>
<tr>
<td>LastName3, FirstName3</td>
<td>90</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>90</td>
</tr>
<tr>
<td>LastName4, FirstName4</td>
<td>87</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>87</td>
</tr>
<tr>
<td>LastName5, FirstName5</td>
<td>94</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>94</td>
</tr>
<tr>
<td>LastName6, FirstName6</td>
<td>99</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>99</td>
</tr>
<tr>
<td>LastName7, FirstName7</td>
<td>96</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>96</td>
</tr>
<tr>
<td>LastName8, FirstName8</td>
<td>71</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>71</td>
</tr>
<tr>
<td>LastName9, FirstName9</td>
<td>62</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>62</td>
</tr>
<tr>
<td>LastName10, FirstName10</td>
<td>68</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>68</td>
</tr>
<tr>
<td>LastName11, FirstName11</td>
<td>55</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>55</td>
</tr>
<tr>
<td>LastName12, FirstName12</td>
<td>101</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>101</td>
</tr>
<tr>
<td>LastName13, FirstName13</td>
<td>83</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>83</td>
</tr>
<tr>
<td>LastName14, FirstName14</td>
<td>80</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>80</td>
</tr>
<tr>
<td>LastName15, FirstName15</td>
<td>79</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>79</td>
</tr>
<tr>
<td>LastName16, FirstName16</td>
<td>81</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>81</td>
</tr>
<tr>
<td>LastName17, FirstName17</td>
<td>80</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>80</td>
</tr>
<tr>
<td>LastName18, FirstName18</td>
<td>81</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>81</td>
</tr>
</tbody>
</table>
Lab Instructor – Lab Section Detail

After choosing to see either the entire course or the specific sections in detail the Lab Instructor is presented with this screen. Here individual files are shown as opposed to just the overall grade for the assignment.
Here the Lecture Instructor is logged in and has access to all of the student’s grades in their classes or in just specific sections.
### Lecture Instructor – Lecture Section Detail

![Automated Grading System](image)

<table>
<thead>
<tr>
<th>Detailed Grade Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section CSIS-010 - 3</td>
</tr>
<tr>
<td>Description MW, 10/25-11/20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lab 1</th>
<th>Lab 2</th>
<th>Midterm</th>
<th>Lab 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>File A</td>
<td>File B</td>
<td>File C</td>
<td>File A</td>
</tr>
<tr>
<td>LName1, FirstName1</td>
<td>10</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td>LName2, FirstName2</td>
<td>15</td>
<td>20</td>
<td>25</td>
</tr>
<tr>
<td>LName3, FirstName3</td>
<td>20</td>
<td>25</td>
<td>30</td>
</tr>
<tr>
<td>LName4, FirstName4</td>
<td>25</td>
<td>30</td>
<td>35</td>
</tr>
<tr>
<td>LName5, FirstName5</td>
<td>30</td>
<td>35</td>
<td>40</td>
</tr>
<tr>
<td>LName6, FirstName6</td>
<td>35</td>
<td>40</td>
<td>45</td>
</tr>
<tr>
<td>LName7, FirstName7</td>
<td>40</td>
<td>45</td>
<td>50</td>
</tr>
<tr>
<td>LName8, FirstName8</td>
<td>45</td>
<td>50</td>
<td>55</td>
</tr>
<tr>
<td>LName9, FirstName9</td>
<td>50</td>
<td>55</td>
<td>60</td>
</tr>
<tr>
<td>LName10, FirstName10</td>
<td>55</td>
<td>60</td>
<td>65</td>
</tr>
<tr>
<td>LName11, FirstName11</td>
<td>60</td>
<td>65</td>
<td>70</td>
</tr>
<tr>
<td>LName12, FirstName12</td>
<td>65</td>
<td>70</td>
<td>75</td>
</tr>
<tr>
<td>LName13, FirstName13</td>
<td>70</td>
<td>75</td>
<td>80</td>
</tr>
<tr>
<td>LName14, FirstName14</td>
<td>75</td>
<td>80</td>
<td>85</td>
</tr>
<tr>
<td>LName15, FirstName15</td>
<td>80</td>
<td>85</td>
<td>90</td>
</tr>
<tr>
<td>LName16, FirstName16</td>
<td>85</td>
<td>90</td>
<td>95</td>
</tr>
</tbody>
</table>

Here the Lecture Instructor has chosen to see detailed information about the students so the individual files as opposed to just the assignment grade are shown.
All Users – Change Password

This is the form for which all users will see upon choosing to change their password from the side menu. A Username, Old Password, and New Password must be submitted to change a user’s password in the database.
**All Users – Successful Password Changes**

If the information entered in the new password form was correct and verified by the database the user is brought to this screen confirming the change.
This is the screen where the Instructor may reset a student’s password.
Here we have the downloadable report which will be presented to the Course Coordinator, displaying all of the grade information for the students registered in the courses.
Here is the report for which the Lecture Instructor will have access to download which will provide grade information for everyone in their classes.
Lab Instructor Downloadable Report

Here is the report for which the Lab Instructor will have access to download which will provide grade information for everyone in their labs.

<table>
<thead>
<tr>
<th>Last Name</th>
<th>First Name</th>
<th>Lecture Section</th>
<th>Lab1</th>
<th>Lab2</th>
<th>Lab3</th>
<th>Lab4</th>
<th>Lab5</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitterly</td>
<td>Nicholas</td>
<td>CSIS 010-01</td>
<td>56%</td>
<td>70%</td>
<td>94%</td>
<td>95%</td>
<td>95%</td>
<td>83.6%</td>
</tr>
<tr>
<td>Mancosse</td>
<td>Raymond</td>
<td>CSIS 010-01</td>
<td>95%</td>
<td>69%</td>
<td>87%</td>
<td>76%</td>
<td>76%</td>
<td>80.6%</td>
</tr>
<tr>
<td>Duranton</td>
<td>Patrick</td>
<td>CSIS 010-02</td>
<td>85%</td>
<td>76%</td>
<td>57%</td>
<td>97%</td>
<td>84%</td>
<td>79%</td>
</tr>
<tr>
<td>Care</td>
<td>Whitney</td>
<td>CSIS 010-02</td>
<td>56%</td>
<td>67%</td>
<td>98%</td>
<td>100%</td>
<td>87%</td>
<td>81.0%</td>
</tr>
<tr>
<td>Case</td>
<td>Kevin</td>
<td>CSIS 010-03</td>
<td>70%</td>
<td>84%</td>
<td>72%</td>
<td>78%</td>
<td>93%</td>
<td>84.0%</td>
</tr>
<tr>
<td>Perry</td>
<td>Brenden</td>
<td>CSIS 010-03</td>
<td>67%</td>
<td>77%</td>
<td>83%</td>
<td>77%</td>
<td>85%</td>
<td>81.6%</td>
</tr>
<tr>
<td>Neksa</td>
<td>Jennifer</td>
<td>CSIS 010-03</td>
<td>87%</td>
<td>99%</td>
<td>87%</td>
<td>89%</td>
<td>77%</td>
<td>87.6%</td>
</tr>
</tbody>
</table>
Here we have a view of the grading system software. The Lab Instructor uses this software to query the database and initialize grading. Clicking the “Assignments: Need Review” tab will result in a list of files that have already been graded, and now need to be approved by the instructor. By clicking “View File” the Lab Instructor will see the graded assignment, and the first wrong answer will appear in the “Actual” section, and the correct answer will be in the “Expected” section. Navigating through these answers can be done using the Entry arrows. Each answer is then either approved, or a new point value can be entered by the Lab Instructor in the “Points Awarded:” category.
Lab Instructor – Lab Section Detail

The Lab Instructor uses the red login box shown here to connect to the system database. Once connected, the database is queried and the results are presented in the window. By clicking on the “Assignments: Ready to Grade” tab, the Lab Instructor will see a list of files that are ready to be graded by the software. The Lab Instructor then highlights any files he/she wishes to grade, and adds them to the list in the lower section of the screen by clicking “Add”. He/she may also choose to simply add all the files, and this works similarly for removing the items from the lower list. Once all files are selected, by clicking “Grade Selected Assignments” the software will begin grading each file, and once finished, the files will be put into the “Need Review” list.
This is the general form that a File Grading Key will take. Each line of the key will indicate a specific area of the student’s Excel project to be checked against the submission provided by the Course Coordinator. Entries in this file will follow a specified format. There will first be the range of the Excel file in question. This is followed by a comma and the number of points this entry is worth. Another comma follows this and then a list of the properties to check over the given range. Each entry will end with a semi-colon.
Section 1.2: Detailed Data Flow Diagrams

A data flow diagram (DFD) is graphic representation of the "flow" of data through Sources or processes. More generally, a data flow diagram is used for the visualization of data processing. It illustrates the processes, data stores, external entities, data flows in a system and the relationships between these things.

**Metrica and SSADM Methodology**

A **Process** signifies that something is happening to transform data. Processes have numbers, and those numbers reflect the decomposition hierarchy.

A **Data Store** is a place where data is kept while it is not actively being processed. Data can only enter a data store from a process and can only leave a data store to a process.

An **External Entity** is something outside the boundary of the system you are modeling that either sends data to your system or receives data from it. External entities are optional.

A **Data Flow** depicts the movement of one to many items of data. Data can enter a system from the outside.
Context Diagram:

Level 1 Diagram:
Level 2 Diagram:
Level 3 Diagram: Decomposition of Display Web Interface

- **D1 System Database**
  - **Login Interface**
    - **Username**
    - **Verification**
    - **Password**
  - **Login Interface**
    - **Reject**
    - **Login**
    - **Reject**
    - **Accept Lecture Instructor**
    - **Accept Course Coordinator**
  - **Display Lecture Instructor Interface**
    - **Grade Info**
    - **Login**
    - **Reject**
    - **Accept Lecture Instructor**
  - **Display Student Interface**
    - **Grade Info**
    - **Assignment**
    - **Accept Student**
  - **Display Lecture Instructor Interface**
    - **Grade Info**
    - **Login**
    - **Reject**
    - **Accept Lecture Instructor**
  - **Display Coordinator Interface**
    - **Grade Info**
    - **Accept Course Coordinator**
    - **Accept Lab Instructor**
  - **Display Lab Instructor Interface**
    - **Grade Info**
    - **Accept Lab Instructor**
  - **Student**
    - **Assignment**
    - **Grade Info**
    - **Login**
    - **Reject**
    - **Accept Student**
  - **Lab Instructor**
    - **Grade Info**
    - **Accept Lab Instructor**
    - **Login**
    - **Reject**
  - **Course Coordinator**
    - **Key**
    - **Reject**
    - **Login**
    - **Reject**
    - **Accept Course Coordinator**
  - **Lecturer Instructor**
    - **Key**
    - **Reject**
    - **Login**
    - **Reject**
    - **Accept Lecture Instructor**

**Performance Software**

**Preliminary Design** 37
Section 1.3: Logical Data Dictionary

Location – Context Diagram:

Assignment Data Flow
Description:
Lab assignments can be submitted to the system.
Source: User (Source/Sink)
Dest: System (Process)

Grade Data Flow
Description:
The final Grade can be viewed by the users.
Source: System (Process)
Dest: User (Source/Sink)

Graded Assignment Data Flow
Description:
Graded Assignment can be viewed by the users.
Source: System (Process)
Dest: User (Source/Sink)

Key Data Flow
Description:
Grading Key can be submitted to the system.
Source: User (Source/Sink)
Dest: System (Process)

Report Data Flow
Description:
Reports can be viewed by users.
Source: System (Process)
Dest: User (Source/Sink)

System Process
Description:
Allows for grading of lab assignments and viewing of student grades.
Input Flows:
Assignment
Key
Output Flows:
Graded Assignment
Grade
Report
User Source/Sink

Description:
Represents the users of the system; Student, Lecture Instructor, Course Coordinator, and Lab Instructor.

Input Flows:
Graded Assignment
Grade
Report

Output Flows:
Assignment
Key

Location – Level 1 Diagram:

Assignment Data Flow

Description:
Student submits the lab assignment to the system for grading.

Source: 1) Student (Source/Sink)
2) Excel Grading System (Process)
3) System Database (Data Store)

Dest: 1) Excel Grading System (Process)
2) System Database (Data Store)
3) Excel Grading System (Process)

-------------------------------------------------------------------------------------------------------------------------------

Course Coordinator Source/Sink

Description:
The Course Coordinator is responsible for creating user accounts and for submitting the key and template files to be used for grading each lab assignment.

Input Flows:
Grade Information

Output Flows:
Key

-------------------------------------------------------------------------------------------------------------------------------

Excel Grading System Process

Description:
Allows for submission of lab assignments, submission of key files, grading of lab assignments, and viewing of student grades.

Input Flows:
Assignment
Key
Verified Grade Results

Output Flows:
Assignment
Grade Info
Key
Unverified Grade Results
Verified Grade Results
Grade Info Data Flow

Description:
The Excel Grading System returns the grade information for viewing.

Source: Excel Grading System (Process)
Dest: Student (Source/Sink)
      Lecture Instructor (Source/Sink)
      Course Coordinator (Source/Sink)
      Lab Instructor (Source/Sink)

Key Data Flow

Description:
Key is submitted to the system to serve as criteria for grading the lab assignments.

Source: 1) Course Coordinator (Source/Sink)
        2) Excel Grading System (Process)
        3) System Database (Data Store)
Dest: 1) Excel Grading System (Process)
      2) System Database (Data Store)
      3) Excel Grading System (Process)

Lab Instructor Source/Sink

Description:
The Lab Instructor teaches lab sections and is in charge of grading their students’ labs.
The Lab Instructor can also view their students’ grades.

Input Flows:
Grade Information
Unverified Grade Results

Output Flows:
Verified Grade Results

Lecture Instructor Source/Sink

Description:
The Lecture Instructor teaches one or more lecture sections and can view the grades of all
students enrolled in their lecture sections.

Input Flows:
Grade Information

Student Source/Sink

Description:
Student that is enrolled in a lab section.

Input Flows:
Grade Information

Output Flows:
Assignment
**System DB Data Store**

*Description:*
Data store; Database containing software usernames, software passwords, submitted lab files, graded lab files, grading key, and the grading template.

*Input Flows:*
- Assignment
- Verified Grade Results
- Key

*Output Flows:*
- Assignment
- Verified Grade Results
- Key

Unverified Grade Results Data Flow

*Description:*
The Unverified Grade Results display what the Excel Grading System marked incorrect on each student’s lab. The Unverified Grade Results can be overridden by the Lab Instructor.

*Source:* Excel Grading System (Process)

*Dest:* Lab Instructor (Source/Sink)

Verified Grade Results Data Flow

*Description:*
The Verified Grade Results are the final graded lab assignment for the student.

*Source:* 1) Lab Instructor (Source/Sink)  
2) Excel Grading System (Process)  
3) System Database (Data Store)

*Dest:* 1) Excel Grading System (Process)  
2) System Database (Data Store)  
3) Excel Grading System (Process)

**Location – Level 2 Diagram:**

Assignment Data Flow

*Description:*
Student submits the lab assignment to the system for grading.

*Source:* 1) Student (Source/Sink)  
2) Display Web Interface (Process)  
3) System Database (Data Store)

*Dest:* 1) Display Web Interface (Process)  
2) System Database (Data Store)  
3) Grade Software (Process)
**Class List Data Flow**

*Description:* List of students in each lab section.

*Source:* System Database (Data Store)  
*Dest:* Grade Software (Process)

**Course Coordinator Source/Sink**

*Description:* The Course Coordinator is responsible for creating user accounts and for submitting the key and template files to be used for grading each lab assignment.

*Input Flows:*  
Grade Information

*Output Flows:*  
Key

**Display Web Interface Process**

*Description:* Deals with user registration, user permission verification, submission of files, and access to reports, grades, and assignments.

*Input Flows:*  
Verified Grade Results  
Assignment  
Key

*Output Flows:*  
Assignment  
Grade Info  
Key

**Grade Info Data Flow**

*Description:* The Excel Grading System returns the grade information for viewing.

*Source:* Excel Grading System (Process)  
*Dest:* Course Coordinator (Source/Sink)  
Lab Instructor (Source/Sink)  
Lecture Instructor (Source/Sink)  
Student (Source/Sink)

**Grade Software Process**

*Description:* Handles the grading of lab assignments.

*Input Flows:*  
Assignment  
Class List  
Key  
Verified Grade Results

*Output Flows:*  
Unverified Grade Results  
Verified Grade Results
**Key Data Flow**

*Description:*  
Key is submitted to the system to serve as criteria for grading the lab assignments.

*Source:*  
1) Course Coordinator (Source/Sink)  
2) Display Web Interface (Process)  
3) System Database (Data Store)

*Dest:*  
1) Display Web Interface (Process)  
2) System Database (Data Store)  
3) Grade Software (Process)

**Lab Instructor Source/Sink**

*Description:*  
The Lab Instructor teaches lab sections and is in charge of grading their students’ labs. The Lab Instructor can also view their students’ grades.

*Input Flows:*  
Grade Information  
Unverified Grade Information

*Output Flows:*  
Verified Grade Information

**Lecture Instructor Source/Sink**

*Description:*  
The Lecture Instructor teaches one or more lecture sections and can view the grades of all students enrolled in their lecture sections.

*Input Flows:*  
Grade Information

**Results Data Flow**

*Description:*  
Grade Results generated by the Grade Software.

*Source:*  
Grade Software (Process)

*Dest:*  
System Database (Data Store)

**Student Source/Sink**

*Description:*  
Student that is enrolled in a lab section.

*Input Flows:*  
Grade Information

*Output Flows:*  
Assignment
System DB Data Store

Description:
Data store; Database containing software usernames, software passwords, submitted lab files, graded lab files, grading key, and the grading template.

Input Flows:
Assignment
Verified Grade Results
Key

Output Flows:
Assignment
Verified Grade Results
Key

Unverified Grade Results Data Flow

Description:
The Unverified Grade Results display what the Excel Grading System marked incorrect on each student’s lab. The Unverified Grade Results can be overridden by the Lab Instructor.

Source: Grade Software (Process)
Dest: Lab Instructor (Source/Sink)

Verified Grade Results Data Flow

Description:
The Verified Grade Results are the final graded lab assignment for the student.

Source:
1) Lab Instructor (Source/Sink)
2) Excel Grading System (Process)
3) System Database (Data Store)

Dest:
1) Excel Grading System (Process)
2) System Database (Data Store)
3) Display Web Interface (Process)

Location – Level 3 Diagram - Decomposition of Display Web Interface

Accept Course Coordinator Data Flow

Description:
If a valid Course Coordinator login is entered, user is taken to Display Course Coordinator Interface

Source: Login Interface (Process)
Dest: Display Course Coordinator Interface (Process)

Accept Lab Instructor Data Flow

Description:
If a valid Lab Instructor login is entered, user is taken to Display Lab Instructor Interface

Source: Login Interface (Process)
Dest: Display Lab Instructor Interface (Process)
Accept Lecturer Data Flow
Description:
If a valid Lecturer login is entered, user is taken to Display Lecturer Interface
Source: Login Interface (Process)
Dest: Display Lecturer Interface (Process)

Accept Student Data Flow
Description:
If a valid Student login is entered, user is taken to Display Student Interface
Source: Login Interface (Process)
Dest: Display Student Interface (Process)

Assignment Data Flow
Description:
Student submits assignment through the Display Student Interface, which in turn sends
the assignment to the System Database
Source: Student (Source)
Dest: Display Student Interface (Process)
System Database (Data Store)

Course Coordinator Source/Sink
Description:
The Course Coordinator is responsible for creating user accounts and for submitting the
key and template files to be used for grading each lab assignment.
Input Flows:
Grade Information
Output Flows:
Key

Display Course Coordinator Interface Process
Description:
Interface that gives the Course Coordinator all the capabilities that they are allowed
Input Flows:
Key
Accept Course Coordinator
Output Flows:
Key
**Display Lab Instructor Interface Process**

*Description:*
Interface that gives the Lab Instructor all the capabilities that they are allowed

*Input Flows:*
- Accept Lab Instructor
- Grade Info

*Output Flows:*
- Grade Info

**Display Lecturer Interface Process**

*Description:*
Interface that gives the Lecturer all the capabilities that they are allowed

*Input Flows:*
- Accept Lecturer
- Grade Info

*Output Flows:*
- Grade Info

**Display Student Interface Process**

*Description:*
Interface that gives the Student all the capabilities that they are allowed

*Input Flows:*
- Assignment
- Accept Student
- Grade Info

*Output Flows:*
- Grade Info
- Assignment

**Grade Info Data Flow**

*Description:*
The System Database returns the grade information for viewing.

*Source:*
System Database (Data Store)

*Dest:*
- Display Lab Instructor Interface (Source)
- Display Lecture Interface (Source)
- Display Student Interface (Source)
- Lab Instructor (Process)
- Lecturer (Process)
- Student (Process)

**Key Data Flow**

*Description:*
Key is entered by the Course Coordinator, through the Display Course Coordinator Interface, and entered into the System Database

*Source:*
Course Coordinator (Source)

*Dest:*
Display Course Coordinator Interface (Process)
Lab Instructor Source/Sink

Description:
The Lab Instructor teaches lab sections and is in charge of grading their students’ labs. The Lab Instructor can also view their students’ grades.

Input Flows:
Grade Information
Unverified Grade Results

Output Flows:
Verified Grade Results

---

Lecture Instructor Source/Sink

Description:
The Lecture Instructor teaches one or more lecture sections and can view the grades of all students enrolled in their lecture sections.

Input Flows:
Grade Information

---

Login Data Flow

Description:
User enters a login, it is then accepted or rejected by the Login Interface Process

Source:
Lecturer Instructor (Source)
Course Coordinator (Source)
Student (Source)
Lab Instructor (Source)

Dest:
Login Interface (Process)

---

Password Data Flow

Description:
Login sends a Username and Password to the System Database for validation

Source:
Login Interface (Process)

Dest:
System Database (Data Store)

---

Reject Data Flow

Description:
If an incorrect Username and Password are entered, this user will be rejected from the System

Source:
Login Interface (Process)

Dest:
Lecturer Instructor (Source)
Course Coordinator (Source)
Student (Source)
Lab Instructor (Source)
Student Source/Sink
Description: Student that is enrolled in a lab section.
Input Flows: Grade Information
Output Flows: Assignment

System Database Data Store
Description: Data Store that holds, Assignments, Grade Info, Usernames, Passwords, Keys
Input Flows: Assignments, Username, Password, Key
Output Flows: Grade Info, Verification

Username Data Flow
Description: Login sends a Username and Password to the System Database for validation
Source: Login Interface (Process)
Dest: System Database (Data Store)

Verification Data Flow
Description: When a Username and Password are entered, the System Database will check to see if they are valid, if so the Login will be verified
Source: System Database (Data Store)
Dest: Login Interface (Process)

Location – Level 3 Diagram - Decomposition of Grade Software

Approve Report Data Flow
Description: When a Grade Report is returned to the Lab Instructor they must validate its contents
Source: Lab Instructor (Source)
Dest: User Interface (Process)

Approved Results Data Flow
Description:
Grading results are sent to the Lab Instructor to be approved, after they are approved they are sent to the System Database

**Source:** User Interface (Process)
**Dest:** System Database (Data Store)

---

### Assignment Data Flow

**Description:**
User Interface requests an Assignment from the File Grabber Process

**Source:** User Interface (Process)
**Dest:** File Grabber (Process)

---

### Assignment Data Flow

**Description:**
File Grabber Process pulls the assignment from the System Database

**Source:** File Grabber (Process)
**Dest:** System Database (Data Store)

---

### Assignment Request Data Flow

**Description:**
User requests an assignment to be graded

**Source:** User Interface (Process)
**Dest:** File Grabber (Process)

---

### Engine Process

**Description:**
Process that does the actual grading of assignments, based on the assignment entered and the key for that assignment

**Input Flows:**
- Assignment
- Key

**Output Flows:**
- Grading Data
- Message
File Grabber Process

Description:
Process that gets the assignments that are to be graded from the System Database

Input Flows:
Assignment Request
Key
Results
Assignment

Output Flows:
Assignment
Key
Assignment Request
Result Request
Key Request

Grading Data Data Flow

Description:
After the grading has been completed, the Engine gives all the Grade Data to the System Database

Source: Engine (Process)
Dest: System Database (Data Store)

Grading Request Data Flow

Description:
The Lab Instructor requests to grade a variable number of assignments to be graded

Source: Lab Instructor (Source)
Dest: User Interface (Process)

Key Data Flow

Description:
Upon a request from the File Grabber, the System Database sends the Key

Source: System Database (Data Store)
Dest: File Grabber (Process)

Key Request Data Flow

Description:
File Grabber process sends a request to the System Database for the Key

Source: File Grabber (Process)
Dest: System Database (Data Store)
Lab Instructor ID Data Flow
Description:
A Lab Instructor ID can be entered to find a list of students in the Lab Instructors lab section(s)
Source: User Interface (Process)
Dest: System Database (Data Store)

Lab Instructor Source
Description:
Any Lab Instructor has the ability to initiate the grading process
Input Flows:
Result Report
Output Flows:
Grading Request
Approved Report

Message Data Flow
Description:
A message is sent to the interface that alerts Lab Instructor users that grades are completed and are waiting validation
Source: Engine (Process)
Dest: User Interface (Process)

Results Data Flow
Description:
Grading results are sent to the File Grabber upon request
Source: System Database (Data Store)
Dest: File Grabber (Process)

Result Report Data Flow
Description:
A report containing all the grade results is sent to the Lab Instructor to be validated
Source: User Interface (Process)
Dest: Lab Instructor (Source)

Result Request Data Flow
Description:
File Grabber process sends a request to the System Database for the grading results
Source: File Grabber (Process)
Dest: System Database: (Data Store)
**Student ID Data Flow**

*Description:*  
Student ID can be entered to do a search for a specific student  
*Source:* User Interface (Process)  
*Dest:* System Database (Data Store)

**System Database Data Store**

*Description:*  
Data Store that holds, Assignments, Grade Info, Usernames, Passwords, Keys  
*Input Flows:*  
Student ID  
Lab Instructor ID  
Approved Results  
Assignment Request  
Result Request  
Key Request  
Grading Data  
*Output Flows:*  
Assignment  
Key  
Results

**User Interface Process**

*Description:*  
Interface that allows the Lab Instructor the capabilities to search for, and grade specific lab assignments  
*Input Flows:*  
Grading Request  
Approved Report  
Message  
Results  
Assignment  
*Output Flows:*  
Student ID  
Lab Instructor ID  
Approved Results  
Result Request  
Assignment Request  
Result Report
Section 1.4: Logical Data Stores

Section 1.4.1: Logical Data Stores

The following is a list of tables, their elements, and the modules for which they will be used within the Excel Grading System. Field Name corresponds to the attribute name the entities are expected to have within the actual database.

(PK) - Represents a Primary Key
(FK) - Represents a Foreign Key

StudentUser Table
Modules:
Web Interface User Login, Grading Application User Login.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>Text</td>
<td>PK of StudentUser table/ This will be the Student's email address</td>
</tr>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of StudentUser table/ FK to Semester table</td>
</tr>
<tr>
<td>password</td>
<td>Text</td>
<td>Must be alphanumeric and be &lt;=20 vars</td>
</tr>
<tr>
<td>firstname</td>
<td>Text</td>
<td>Must be only characters, &lt;=40 chars</td>
</tr>
<tr>
<td>lastname</td>
<td>Text</td>
<td>Must be only characters, &lt;=50 chars</td>
</tr>
<tr>
<td>lecturesection</td>
<td>Number</td>
<td>FK to lecture section, &lt;=10 decimal value</td>
</tr>
<tr>
<td>labsection</td>
<td>Number</td>
<td>FK to lab section, &lt;=10 decimal value</td>
</tr>
<tr>
<td>lastlogin</td>
<td>Date/Time</td>
<td>The date and time of the last login</td>
</tr>
<tr>
<td>attempts</td>
<td>Number</td>
<td>If Attempts &gt; 3 username will be locked out of system</td>
</tr>
</tbody>
</table>

This table will store information regarding all Students. This includes login information, such as username, password, the last login date and time, and the number of login attempts. Other Student information such as first name, last name and both the lecture and the lab section to which the Student belongs is also stored in this table.
OtherUser Table

Modules:
Web Interface User Login, Grading Application User Login.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of OtherUser table/ FK to Semester table</td>
</tr>
<tr>
<td>username</td>
<td>Text</td>
<td>PK of OtherUser table/ This will be the User’s email address</td>
</tr>
<tr>
<td>password</td>
<td>Text</td>
<td>Must be alphanumeric and be &lt;=20 varchar</td>
</tr>
<tr>
<td>firstname</td>
<td>Text</td>
<td>Must be only characters, &lt;=40 chars</td>
</tr>
<tr>
<td>lastname</td>
<td>Text</td>
<td>Must be only characters, &lt;=50 chars</td>
</tr>
<tr>
<td>lastlogin</td>
<td>Date/Time</td>
<td>The date and time of the last login</td>
</tr>
<tr>
<td>attempts</td>
<td>Number</td>
<td>If Attempts &gt; 3 username will be locked out of system</td>
</tr>
<tr>
<td>admin</td>
<td>Yes/No</td>
<td>0 for non-admin, 1 for admin</td>
</tr>
</tbody>
</table>

This table will store information regarding the users of the system who are not Students (Lecture Instructor, Lab Instructor, and Course Coordinator). This includes login information, such as username, password, the last login date and time, and the number of login attempts.

Sections Table

Modules:
StudentUser and OtherUser Tables for Web Interface and Grading Software.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sectionid</td>
<td>Number</td>
<td>PK of the Sections table</td>
</tr>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of Sections table/ FK to Semester table</td>
</tr>
<tr>
<td>sectiontype</td>
<td>Text</td>
<td>The type of the section(lecture/lab)</td>
</tr>
<tr>
<td>instructor</td>
<td>Text</td>
<td>FK to the OtherUser’s table/ Username of instructor</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
<td>User specified course description</td>
</tr>
</tbody>
</table>

This table will store information regarding the class Sections (either lab or lecture). This includes information such as the instructor of the class and a description.
**StudentSubmissions Table**

Modules:
- Web Interface, Grading Application.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>assignmentName</td>
<td>Text</td>
<td>PK of StudentSubmissions table/ FK to the AssignmentDetail table</td>
</tr>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of StudentSubmissions table/ FK to Semester table</td>
</tr>
<tr>
<td>username</td>
<td>Text</td>
<td>PK of the StudentSubmissions table/ FK to the StudentUser table</td>
</tr>
<tr>
<td>filename</td>
<td>Text</td>
<td>PK of StudentSubmissions table/ FK to the AssignmentDetail table</td>
</tr>
<tr>
<td>fileLocation</td>
<td>Text</td>
<td>Address of where the Student’s file is stored</td>
</tr>
<tr>
<td>submitted</td>
<td>Date/Time</td>
<td>The date and time the file was submitted</td>
</tr>
<tr>
<td>finalized</td>
<td>Date/Time</td>
<td>Null if not yet graded</td>
</tr>
<tr>
<td>submissionId</td>
<td>Number</td>
<td>An unique id for the StudentSubmissions table</td>
</tr>
</tbody>
</table>

This table will store information regarding the Student submission of files. This includes information such as the assignment name, the username of the Student submitting, and the date and time that the file was submitted.

**Assignment Table**

Modules:
- Web Interface, Grading Application.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>assignmentName</td>
<td>Text</td>
<td>PK of the Assignment table</td>
</tr>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of Assignment table/ FK to Semester table</td>
</tr>
<tr>
<td>category</td>
<td>Text</td>
<td>FK to Category table</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
<td>An optional description of the assignment</td>
</tr>
<tr>
<td>created</td>
<td>Date/Time</td>
<td>The date the assignment was created on</td>
</tr>
</tbody>
</table>

This table will store information regarding the assignment. This includes information such as the assignment name, the assignment description, and the assignment category.
AssignmentDetail Table

Modules:

Grading Application

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename</td>
<td>Text</td>
<td>PK of AssignmentDetail table</td>
</tr>
<tr>
<td>assignmentName</td>
<td>Text</td>
<td>PK of AssignmentDetail table/ FK to the Assignment table</td>
</tr>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of AssignmentDetail table/ FK to Semester table</td>
</tr>
<tr>
<td>description</td>
<td>Text</td>
<td>An optional description of this part of the assignment</td>
</tr>
<tr>
<td>templatelocation</td>
<td>Text</td>
<td>Address of where the template file(Excel file) is stored</td>
</tr>
<tr>
<td>templatesubmitdate</td>
<td>Date/Time</td>
<td>The date the template was submitted</td>
</tr>
<tr>
<td>keylocation</td>
<td>Text</td>
<td>Address of where the key file(text file) is stored</td>
</tr>
<tr>
<td>keysubmitdate</td>
<td>Date/Time</td>
<td>The date the key was submitted</td>
</tr>
<tr>
<td>pointsworth</td>
<td>Number</td>
<td>The amount of points that this file is worth</td>
</tr>
</tbody>
</table>

This table will store information regarding the key and template files used in the grading process. This includes information such as the assignment name, the locations of both the key and template files, and the amount of points that each file (part of the assignment) is worth.

Category Table

Modules:

Assignment Table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of Category table/ FK to Semester table</td>
</tr>
<tr>
<td>category</td>
<td>Text</td>
<td>PK of Category table</td>
</tr>
<tr>
<td>weight</td>
<td>Number</td>
<td>All assignment files add up to 100</td>
</tr>
</tbody>
</table>

This table will store information regarding the category and weight of the assignment files.
GradeElements Table
Modules:
  Grading Application

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>submissionId</td>
<td>Number</td>
<td>PK of the GradeElements table / FK to StudentSubmissions table</td>
</tr>
<tr>
<td>elementId</td>
<td>Text</td>
<td>PK of the GradeElements table</td>
</tr>
<tr>
<td>pointsEarned</td>
<td>Number</td>
<td>The number of points earned for the element</td>
</tr>
<tr>
<td>notes</td>
<td>Text</td>
<td>Optional for the Lab Instructor</td>
</tr>
</tbody>
</table>

This table will store information regarding the graded elements of the submitted files. This includes information such as the number of points earned for each graded element and any notes left by the Lab Instructor for the Student to view.

Release Table
Modules:
  Web Interface

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>semesterId</td>
<td>Text</td>
<td>PK of Release table / FK to the Semester table</td>
</tr>
<tr>
<td>section</td>
<td>Number</td>
<td>The section number</td>
</tr>
<tr>
<td>assignment</td>
<td>Text</td>
<td>The assignment to be released</td>
</tr>
<tr>
<td>released</td>
<td>Yes/No</td>
<td>0 if no, 1 if yes</td>
</tr>
</tbody>
</table>

This table will store information regarding the release of assignments.

Semester Table
Modules:
  All database tables, excluding the GradeElements table.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>semesterId</td>
<td>Number</td>
<td>PK of the Semester table</td>
</tr>
<tr>
<td>year</td>
<td>Number</td>
<td>The semester year</td>
</tr>
<tr>
<td>semester</td>
<td>Text</td>
<td>The semester season (Fall/Spring)</td>
</tr>
<tr>
<td>semesterPassword</td>
<td>Text</td>
<td>The password for the semester</td>
</tr>
<tr>
<td>isCurrentSemester</td>
<td>Yes/No</td>
<td>0 if no, yes if 1</td>
</tr>
</tbody>
</table>
This table will store information regarding the Semester. This will include information such as the Semester year and season (Fall/Spring).

Section 1.4.2: ERD (Entity-Relationship Diagram)

The following ERD (Entity-Relationship Diagram) graphically depicts the conceptual structure of the database. The table headings represent the names of the database tables. The table contents represent the names of the database table attributes. A bolded attribute represents a primary key. A line connecting two tables represents the use of a foreign key from one table to reference an attribute of another.

Note: The StudentSubmission_1 table is a duplicate of the StudentSubmission table and the StudentUser_1 table is a duplicate of the StudentUser table. Both the StudentSubmission_1 table and the StudentUser table do not exist in the database. Their sole purpose below is to provide a more comprehensive depiction of the database structure.
Section 1.5: Logical Format of Data Stores and Databases

The information for the Excel Grading System will be contained in an Oracle database. As we have determined from our clients we will need to store the submitted Excel files, grading keys, and grading specifications in a separate directory. The Oracle database will then contain the locations of these specific files as opposed to containing the files directly in the database. This means we will only require the storage of Decimals, VarChars, and Booleans.

Section 1.6: Structure Charts

Structure charts are graphical representations of the subroutine and function hierarchy in the program's Procedure Division. By viewing structure charts, you can identify and fully understand the strengths and weaknesses of the program structure.

**Structure Chart Methodology**

A **Module** represents a group of instructions that carry out an operation, such as a computer program or subroutine.

A **Library Module** behaves in the same manner every time that it is called upon.

An **Off-Page Connector** is used to link charts that span multiple pages.
View Grades

- Lecture Section
- Download Sections
Section 2: System User Specifications

Section 2.1: User Case Scenarios

This section describes the functions of each user in the EGS (Excel Grading System) fully and in detail. The abilities of each user are laid out in entirety to specify exactly what capabilities each user will have in the system.

Course Coordinator:

The Course Coordinator (CC) will be administrator of the EGS. This user will log into an account created by the developers. Once logged into the system, the CC has many tasks to perform, including management of users, setup of lab and lecture sections, setup of assignment list, submission of grading keys and template files, and report generation.

The CC is responsible for creating a system profile for each instructor. This profile is independent of what type of section (lecture or lab) that instructor leads. The CC must also import a list of student users so that the system can create user profiles for the students. The CC will have the ability to reset passwords for all users in the system and to change the assignment of student user lab and lecture sections. (Note: If the system ends up implementing the LDAP protocol, the CC ability would be limited only to importing user lists, identifying which users are instructors, and changing student lab and lecture sections.)

The CC must create Lecture and Lab sections. During the creation process, the CC will identify what type of section is being made (lab or lecture), the details of the class (this includes meeting day(s), time, section number), and assign an instructor user to the class. This process identifies the instructor as a Lab Instructor or Lecture Instructor. Keep in mind that an instructor may be assigned multiple sections, and so may end up being both a Lab Instructor and Lecture Instructor.

To begin the process of setting up the assignment list, the CC must set up assignment categories and category weights. This allows the system to track student grades based on the CC specified weight criteria. Categories may be added at any time; however, an assignment cannot be created for a category that does not exist. Categories not designated to any assignments may be deleted. Percentage weights to existing categories may be changed. The percentages must add up to 100%. A typical set of category/weight pairs may be:

\{Regular Labs: 50%; Midterm: 25%; Final:25%;\}

The CC will set up assignments in the system. For a new assignment, this process includes choosing to create a new assignment, choosing a category for the assignment, identifying what files will be expected from the students for submission, and assigning a point value for each file in the assignment. The CC will have the ability to modify or remove created assignments from the system.

The CC must submit template files and key files for each expected file in an assignment. The template file will be a correct copy of the assignment that is submitted by the
students. The key file will describe what will be checked in the students’ submitted labs and what point values each part of the template is worth.

Reports will be available for the CC to view. Some possible reports include lists of students by section (lab or lecture) and class averages by section (lab or lecture).

The ability to change the user password will be formed so that the Course Coordinator may enter the Course Coordinator’s current password and the new password twice for confirmation. There will be further precautions placed on this process for the Course Coordinator to ensure security.

**Student:**

The Student will be a user of the EGS. This user will log in through the web interface using an account previously established by the Course Coordinator. Once logged into the system the Student will have the option of submitting a lab, viewing graded labs, viewing individual lab grades, as well as overall course grades, and changing the student’s password.

The Student’s ability to submit a lab will include a view of what lab is currently due and the deadline for submission. Once this deadline is reached the system will no longer accept the lab from this user. Lab submissions may include multiple files which will be uploaded individually through a submission box by the student.

The ability to view graded labs will allow the student to see labs which have been graded and approved by the Lab Instructor. These grades will come in the form of the submitted lab assignment file that has been measured against a key provided by the Course Coordinator as well as any comments provided by the Lab Instructor.

A section in the Student user account will be included that will contain a table of the labs and the separate lab files the student has submitted. This table is where the student will go to open and view a graded lab, but the grade of each file will also be visible from the table. Along with this there will be a table which includes both lab grades as well as other course grades and therefore gives the student’s entire course grade, as well as the lab and lecture section averages the student is enrolled in.

The ability to change the user password will be a form where the student may enter the student’s current password and the new password twice for confirmation.

**Lab Instructor:**

The Lab Instructor will be a user of the EGS. The lab instructor is an instructor who teaches either a CS 010 or CS 011 lab section. This user will log in through the web interface using an account previously established by the Course Coordinator. Once logged into the system the Lab Instructor will have the option of viewing graded labs, changing the lab instructor’s password, grading a specific set of labs or grading all labs, leaving comments on graded labs, manually overriding our software grade, viewing
reports of the lab instructor’s class averages and individual student grades, and including outside grades.

The ability to view graded labs will allow the lab instructor to see labs which have been graded and approved for each student. These grades will come in the form of submitted lab assignment file that has been measured against a key provided by the Course Coordinator.

The ability to change the user password will be a form where the lab instructor may enter the lab instructor’s current password and the new password twice for confirmation.

The lab instructor will have the option of grading a specific set of labs or grading all the labs that have been submitted. When the lab instructor logs onto the system the lab instructor will see a list of all the labs that have been submitted by students. At this point the lab instructor will have the option to select all of the labs that are waiting to be graded, or to select a specific set of labs that are to be graded using check boxes next to each assignment.

The lab instructor will have the ability to leave comments on labs that have been graded. After an assignment has been graded by our system, the lab instructor will have the ability to view the lab to see what has been marked wrong. At this time the lab instructor will have the option of leaving comments on an assignment telling the student any pertinent information.

The lab instructor will be able to manually override our software. After a lab has been graded the lab instructor will be able to view the assignment, seeing what has been marked as wrong and where a student has lost points. At this time the lab instructor will be able to select certain portions of an assignment and override what our grading system has marked as being wrong.

The lab instructor will have a sidebar with the lab instructor’s lab sections listed. Upon clicking on a section a list of the students enrolled in this section will appear in a table format. Along side each student’s name will be the labs the students have submitted and the scores the students have received for those labs. The bottom of the table will show the class average for each lab, as well as for all the labs together.

The lab instructor will be able to include outside grades. An instructor will be able to manually enter grades into our system without the submission of a file. The system will allow for lab instructor to enter grades that are not excel documents, and include nothing but a grade, this will allow lecture instructors to view all of a students grades in the course, not just the student’s excel lab grades.

**Lecture Instructor:**

The lecture instructor is a user of the EGS. After logging into the system, under the account setup by the Course Coordinator, the user will be able to change the lecture
instructor’s password, view the students in each of the lecture instructor’s lecture sections and also download student grades.

To view the student information, the lecture instructor will select the desired lecture section link from the left menu pane. After choosing a section to view, the lecture instructor will see a list of the lecture instructor’s students and the lab section that the students are in. The lecture instructor will also be able to view the total score that each student received on each lab and the score that the student received on each individual part of the lab.

To download the student grades, the lecture instructor will select the link in the left menu pane. A file will be downloaded to the lecture instructor’s computer containing the student grades for all of the sections that the lecture instructor teaches. The file that is downloaded will either be an excel document or some form of text document.

The ability to change the user password will be a form where the lecture instructor may enter the lecture instructor’s current password and the new password twice for confirmation.

Section 2.2: Functional Requirements

This section describes the functions of each user in the EGS and what abilities are involved within each function. It summarizes the above User Case Scenarios.

Course Coordinator Functions:

- Log in under an account created by the developers
- Create a system profile for each instructor
  - Change instructor account passwords
  - Remove profiles already created
- Import a list of students to be used by the system to create student profiles
  - Change student account passwords
  - Change lab and/or lecture sections chosen by the students at initial login
  - Remove profiles already created
- Create lecture and lab sections
  - Identify if the section is lab or lecture
  - Identify details of the section (meeting days, time, and section number of each)
  - Identify instructor of each section
  - Change or remove sections already created
- Set up assignment categories and category weights
  - Specify the type of assignment being accepted and the percentage of the overall class grade it will count for
  - Modify or remove empty categories
- Set up assignments for the system
  - Choose the category for the assignment to fall under
• Identify expected files for grading
  • Assign point value for each file
  • Modify or remove the assignments already created

• Submit template files and key files for each expected file in assignments
  • The Template file is the correct version of the files that will be submitted by the students
  • The Key file will specify which parts of each file are to be graded and the point values for each

• View reports
  • Lists of students by lab or lecture sections
  • Class average based on lab or lecture sections

• Change current password

**Student Functions:**

• Login to an account established by Course Coordinator
• Submit a lab
  • View deadlines for the lab
  • Multiple files in each lab will be uploaded separately
• View graded labs
  • See submitted lab file as compared to the key lab submitted by Course Coordinator, as well as see comments from instructor
• View table of lab grades
  • This is where the files can be opened, and grade is visible from table
  • Table includes tests and other course grades so overall grade found here
• Change current password

**Lab Instructor Functions:**

• Login to an account established by Course Coordinator
• Choose to grade a specified set of labs, or all the labs ready for grading
  • Select all files ready to grade or specify with check boxes
  • View files in the grading process, waiting to be graded, or already graded, but not yet reviewed
• Review the labs graded by software
  • Override the answers given by the grading system
  • Make comments about the lab for the individual student
• Manually enter grades into the system without submission of a file
  • These may not be excel files and may include only a grade specified by instructor
• View graded labs for each student
  • View submitted lab file with corrections corresponding to key lab submitted by Course Coordinator, as well as comments made
• View grades of each student
  • When clicking on class section on sidebar table is opened with list of students
- In table includes all grades for individual students on each lab file
- Change current password

**Lecture Instructor Functions:**

- Login to an account established by Course Coordinator
- Select a lecture section of students to view
- View lab section of each student and individual scores of each submitted lab, as well as each section of the lab
- Download the students’ grades
  - A file will be downloaded to the instructor’s computer containing the grades
- Change current password
Section 3: Testing Requirements

Section 3.1: Project Testing

The testing that we will be conducting in order to ensure a complete and correct working system will include several specific types of analysis.

The first of these four testing steps will be Unit Testing, and for convenience and further understanding of this stage we have included an example of a Unit Test in the following section (Section 3.2) of this document. Unit Testing is used to make certain the system has no data, logic, or standard errors. It checks that each function that the users should be able to perform can be performed with the correct outcome.

The next stage of testing will be Integration Testing. This step combines the Unit Testing to ensure that the separate functions that were tested in Unit Testing follow through correctly when connecting them throughout the system. So a test that may affect another part of the system is checked not only in the present section of the system, but also all other affected portions.

The third stage is System Testing. This stage involves testing that everything that was promised of the system by the software developers is in tact and working correctly. The Requirements Specification document, which has been completed previously, will be a guideline for this stage of testing, and everything that was specified in that document will be tested to be in working order at this point.

The final stage of testing is Acceptance Testing. This stage of testing provides certainty for our clients, Dr. Hunter and Ms. Cotler, that all techniques for building the software promised were enforced, the Requirements Specification was met, and the final product is as it should be. Also, in this stage we test not only for success, but for failure. Therefore, we make sure that when we perform a function that should not allow work (i.e. logging in under the wrong username) the system will not perform the function. All in all, this stage guarantees that our clients are satisfied with the outcome of the system, and feel their requests were met.
Section 3.2: Test Plan

Section 3.2.1: Test Plan Identifier

This test plan is the Master Plan of the Automated Grading System. It describes the testing requirements we, Performance Software, will follow throughout the creation of our program, which includes Unit Testing, Integration Testing, System Testing, as well as Acceptance Testing. All Unit Testing, as well as some Acceptance Testing will be included in this document, while System and Integration Testing will be described thoroughly.

Contact Information for Performance Software, the authors of this test plan document, can be found at <www.performancesoftware.org>.

Section 3.2.2: References

Our test plan outline is based upon the definition and description of test plan from Wikipedia Encyclopedia, found at <http://en.wikipedia.org/wiki/Test_Plan#Test_plan_template.2C_IEEE_829_format>.

Other resources we used for information include: <http://www.coleyconsulting.co.uk/testplan.htm>.

We also modeled our test plan structure on Spartacus Computing Solutions Detailed Design Document from the Software Engineering class of 2004-2005.

Section 3.2.3: Introduction

The function of this test plan is to identify the exact steps we will be taking in testing our software. In this document we will describe what items in our software will be tested. We will examine these items and decide what areas of the software are critical in our testing process, as well as what risks present themselves. We will also look at the software from the user’s perspective and find important testing areas from this point of view, as well as what testing areas are less important and will not be tested. We will identify our exact strategy in going about our testing, including the unit, integration, system, and acceptance testing. The steps we will take upon encountering problems during our testing will be identified. For instance, at what points may we need to stop and go back a few steps because of failed testing. We will specify what will be delivered upon the completion of this document, and what can be expected from it. Necessary environmental needs will be stated, along with the essential staff and staff preparation for the project. Our schedule for the project and testing thereof will be laid out in full. Risks and restrictions of our testing process will be defined. Lastly, required approvals for moving further will be explained, including who must give the go ahead on proceeding forward. Documents that may be referenced in this test plan include the Software Plan, Requirements Specification, and the Preliminary Design.
Section 3.2.4: Test Items

Our testing of this program will begin with unit testing. The unit testing is to include testing that each function we described in the Requirements Specification document will be carried out correctly. The following is a working list of unit tests we will perform:

-- Log in of every type of account
-- Course Coordinator pages:
  - All side bar functions navigate the user to the correct desired screen
  - Log out
  - Change of Password
  - Listing of Assignments for Viewing
  - Adding users, class sections, and grading categories
  - Editing users, class sections, and grading categories
  - Viewing Reports
  - Searching for a User
  - Creating New Assignments
  - Submitting Grading Files
-- Student pages:
  - Student ID upon initial log in
  - All side bar functions navigate the user to the correct desired screen
  - Log out
  - Change of Password
  - View All Grades for Course
  - View Individual Assignment Grades
    - Ungraded
    - Graded
  - Submit Assignment Files
-- Lab Instructor pages:
  - All side bar functions navigate the user to the correct desired screen
  - Log out
  - Change of Password
  - View Grade Reports by Class Section
  - View Detailed Reports with all file grades in each Assignment
-- Lecture Instructor pages:
  - All side bar functions navigate the user to the correct desired screen
  - Log out
  - Change of Password
  - View Grade Reports by Class Section
  - View Detailed Reports with all file grades in each Assignment

Section 3.2.5: Software Risk Issues

Difficulties in our testing of this software will occur when testing that all actions are carried out throughout the entire system. For instance, if a student user is deleted from the system by the Course Coordinator, the student should no longer be listed under any of
the instructors’ accounts in lab or lecture sections, the student should no longer be able to log on to the system, and all traces of the student being part of the courses should be gone in the Course Coordinator account as well. Along with this, the student user’s grades should no longer be included in the total averages in the all student user reports, or in any of the specific section reports that the student was a part of previously. These areas will be more difficult to test for. Other examples of actions being carried out throughout the system include a category percentage being changed; it must be changed in all student accounts when viewing their personal grades, as well as in overall reports viewed by the instructors. If a section is added by the Course Coordinator, it should be added to the drop down menu for the students’ initial log on and viewable by the instructor who teaches the section. Once a student submits his/her files for grading, they should show on the student account as submitted, and also on the lab instructor’s account as ready to grade. A lab instructor overwriting what the grading software has counted as incorrect, or changing a previous score, should also change in the reports to the instructors and Course Coordinator, as well as on that student’s account. These examples are several risk areas of our software that will have to be examined carefully for certain success of each. Another area in which testing will need to be tedious is in the clarification of correct formulas calculating the grades and scores of the students. There will be no room for error in the area of grading the labs and other assignments.

Section 3.2.6: Features to be Tested

Grading Software

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click 'Connect'</td>
<td>Log user in to system</td>
<td></td>
</tr>
<tr>
<td>Click 'Files to Grade'</td>
<td>Get updated list of available files</td>
<td></td>
</tr>
<tr>
<td>Click 'Send Approved Files'</td>
<td>Update server with graded assignments</td>
<td></td>
</tr>
<tr>
<td>Click 'Ready to Grade' Tab</td>
<td>Switch view to 'Ready to Grade'</td>
<td></td>
</tr>
<tr>
<td>Click 'Need Review' Tab</td>
<td>Switch view to 'Need Review'</td>
<td></td>
</tr>
<tr>
<td>Click 'Add'</td>
<td>Add record to pending list</td>
<td></td>
</tr>
<tr>
<td>Click 'Add All'</td>
<td>Add all records to pending list</td>
<td></td>
</tr>
<tr>
<td>Click 'Remove'</td>
<td>Remove record from pending list</td>
<td></td>
</tr>
<tr>
<td>Click 'Remove All'</td>
<td>Remove all records from pending list</td>
<td></td>
</tr>
<tr>
<td>Click 'Grade Selected Assignments'</td>
<td>Process assignments in pending list</td>
<td></td>
</tr>
<tr>
<td>Click 'View File'</td>
<td>Open selected file</td>
<td></td>
</tr>
<tr>
<td>Click 'Previous Arrow'</td>
<td>Move to previous entry</td>
<td></td>
</tr>
<tr>
<td>Click 'Next Arrow'</td>
<td>Move to next entry</td>
<td></td>
</tr>
<tr>
<td>Click 'Approved'</td>
<td>Set file to be ready for submission</td>
<td></td>
</tr>
</tbody>
</table>

Login All Users

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Log In&quot; Button</td>
<td>Logs user into the system with the correct permissions</td>
<td></td>
</tr>
</tbody>
</table>
## Course Coordinator – Side Panel Functions

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Log Out&quot;</td>
<td>User logged out of system</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Change Password&quot;</td>
<td>Navigate to change password page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;All&quot;</td>
<td>Remain on page</td>
<td></td>
</tr>
<tr>
<td>Click a specific assignment</td>
<td>Navigate to page containing details on assignment</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Add New&quot;</td>
<td>Navigate to page for adding new assignments</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Add/Edit Users&quot;</td>
<td>Navigate to page to add or edit users</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Add/Edit Section&quot;</td>
<td>Navigate to page to add or edit class sections</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Add/Edit Category&quot;</td>
<td>Navigate to page to add or edit categories</td>
<td></td>
</tr>
<tr>
<td>Click “All Users”</td>
<td>Navigate to page displaying grade reports of all students</td>
<td></td>
</tr>
<tr>
<td>Click &quot;By Lecture Section&quot;</td>
<td>Navigate to page allowing choice of lecture section to view grades of</td>
<td></td>
</tr>
<tr>
<td>Click &quot;By Lab Section&quot;</td>
<td>Navigate to page allowing choice of lab section to view grades of</td>
<td></td>
</tr>
<tr>
<td>Click &quot;All Assignments&quot;</td>
<td>Navigate to page displaying details of all assignment grades</td>
<td></td>
</tr>
</tbody>
</table>

## Course Coordinator – Class Section Creation and Editing

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Remove&quot;</td>
<td>Ask if the section should really be removed</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Update&quot; by section</td>
<td>Generate new page with updated section listed</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Update&quot; by instructor</td>
<td>Generate new page with updated instructor listed</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Add&quot;</td>
<td>Generate new page with updated course</td>
<td></td>
</tr>
</tbody>
</table>

## Course Coordinator – User Search Results and Editing

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Search&quot;</td>
<td>Create new page with search results</td>
<td></td>
</tr>
</tbody>
</table>
Course Coordinator – User Tasks

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Search&quot;</td>
<td>Create new page with search results</td>
</tr>
<tr>
<td>Click &quot;Add&quot; single user</td>
<td>Show message that user was added</td>
</tr>
<tr>
<td>Click &quot;Add&quot; group</td>
<td>Show message that users were added</td>
</tr>
</tbody>
</table>

Course Coordinator – Assignment Category Creation and Editing

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Remove&quot; Button</td>
<td>Removes a category from the page displayed</td>
</tr>
<tr>
<td>Click on &quot;Add&quot; Button</td>
<td>Adds a category to the page that is displayed</td>
</tr>
<tr>
<td>Click on &quot;Update&quot; Button</td>
<td>Reloads the page to display the changed values</td>
</tr>
</tbody>
</table>

Course Coordinator – Create New Assignment Page

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Create Assignment&quot; Button</td>
<td>Creates a new assignment shell</td>
</tr>
</tbody>
</table>

Course Coordinator – Edit Assignment

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Description&quot; Dropdown Menu</td>
<td>Menu appears offering a list of possible descriptions for the assignment that is being edited</td>
</tr>
<tr>
<td>Click on &quot;Category&quot; Dropdown Menu</td>
<td>Menu appears offering a list of possible Categories that the assignment may be</td>
</tr>
<tr>
<td>Click on &quot;Submit Changes&quot; Button</td>
<td>Reloads the page to display the changed information</td>
</tr>
<tr>
<td>Click on &quot;Add File&quot; Button</td>
<td>Adds another file to an assignment</td>
</tr>
<tr>
<td>Click on &quot;Delete Assignment&quot; Button</td>
<td>Deletes the entire assignment from the system</td>
</tr>
<tr>
<td>Click on &quot;Delete File&quot; Button</td>
<td>Deletes a single file from the assignment</td>
</tr>
</tbody>
</table>

Course Coordinator – Grading Files Submission

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Browse&quot; Button</td>
<td>Opens an explorer window to locate the files to be added</td>
</tr>
<tr>
<td>Click on &quot;Submit Files&quot; Button</td>
<td>Files are sent to the system, a message informs the user of their status</td>
</tr>
</tbody>
</table>

Student – Section Identification
## Test Case

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Continue&quot; with fields entered</td>
<td>Navigate to log in page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Continue&quot; without fields entered</td>
<td>Error message</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Cancel&quot;</td>
<td>Navigate to exit page</td>
<td></td>
</tr>
</tbody>
</table>

### Student – View Graded Assignment Details

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Log Out&quot;</td>
<td>User logged out of system</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Change Password&quot;</td>
<td>Navigate to change password page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Click Here&quot;</td>
<td>Navigate to student grades page</td>
<td></td>
</tr>
<tr>
<td>Click an assignment</td>
<td>Navigate to grade details on assignment page</td>
<td></td>
</tr>
</tbody>
</table>

### Student – View Ungraded Assignment Details

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Log Out&quot;</td>
<td>User logged out of system</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Change Password&quot;</td>
<td>Navigate to change password page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Click Here&quot;</td>
<td>Navigate to student grades page</td>
<td></td>
</tr>
<tr>
<td>Click an assignment</td>
<td>Navigate to grade details on assignment page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Submit Files&quot;</td>
<td>Navigate to file submission page</td>
<td></td>
</tr>
</tbody>
</table>

### Student – Landing Page Grade Overview

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Log Out&quot;</td>
<td>User logged out of system</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Change Password&quot;</td>
<td>Navigate to change password page</td>
<td></td>
</tr>
<tr>
<td>Click &quot;Click Here&quot;</td>
<td>Navigate to student grades page</td>
<td></td>
</tr>
<tr>
<td>Click an assignment</td>
<td>Navigate to grade details on assignment page</td>
<td></td>
</tr>
</tbody>
</table>

### Student - Submit Assignment Files

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;Submit&quot; Button</td>
<td>Submits users files to the database and informs user</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;Submit&quot; Button with incorrect file locations</td>
<td>Informs user and requests a resubmit</td>
<td></td>
</tr>
</tbody>
</table>

### Lab Instructor Landing Page

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;log-out&quot; Link</td>
<td>Logs Lab Instructor out of system and brings to log-in</td>
<td></td>
</tr>
</tbody>
</table>
### Performance Software

#### Preliminary Design

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;change password&quot; Link</td>
<td>Brings Lab Instructor to change password page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;lecture sections&quot; Link</td>
<td>Brings Lab Instructor to the lecture sections page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;download all&quot; Link</td>
<td>Brings Lab Instructor to the download all page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;printer friend version&quot; Link</td>
<td>Brings Lab Instructor to the printer friendly version page of current page</td>
<td></td>
</tr>
<tr>
<td>Mouse-Over &quot;students name&quot;</td>
<td>Provides additional information on student</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;students name&quot; Link</td>
<td>Brings Lab Instructor to a page with additional information on the student</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lab Instructor – Lab Section Detail</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Result</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click on &quot;log-out&quot; Link</td>
<td>Logs Lab Instructor out of system and brings to log-in page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;change password&quot; Link</td>
<td>Brings Lab Instructor to change password page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;lecture sections&quot; Link</td>
<td>Brings Lab Instructor to the lecture sections page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;download all&quot; Link</td>
<td>Brings Lab Instructor to the download all page</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;printer friend version&quot; Link</td>
<td>Brings Lab Instructor to the printer friendly version page of current page</td>
<td></td>
</tr>
<tr>
<td>Mouse-Over &quot;students name&quot;</td>
<td>Provides additional information on student</td>
<td></td>
</tr>
<tr>
<td>Click on &quot;students name&quot; Link</td>
<td>Brings Lab Instructor to a page with additional information on the student</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lecture Instructor Landing Page - Lecture Section Overview</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Results</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click “Log Out”</td>
<td>Lecture Instructor is logged out of system.</td>
<td></td>
</tr>
<tr>
<td>Click “Change Password”</td>
<td>Takes user to a page with a change password form.</td>
<td></td>
</tr>
<tr>
<td>Click a specific lecture section</td>
<td>Generate Grade Report for the specified Lecture Section in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click lecture section “detail”</td>
<td>Generate Grade Report with details about each assignment for the specified Lecture Section in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click “Download All”</td>
<td>Generate Grade Report for all Lecture Sections combined in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click a specific lab section</td>
<td>Generate Grade Report for all Lab Sections combined in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click lab section “detail”</td>
<td>Generate Grade Report for all Lab Sections combined in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click “ Printer Friendly Version”</td>
<td>Generate a web page that is formatted to print properly.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lecture Instructor - Lecture Section Detail</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Results</th>
<th>Pass/Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click “Log Out”</td>
<td>Lecture Instructor is logged out of system.</td>
<td></td>
</tr>
<tr>
<td>Click “Change Password”</td>
<td>Takes user to a page with a change password form.</td>
<td></td>
</tr>
<tr>
<td>Click a specific lecture section</td>
<td>Generate Grade Report for the specified Lecture Section in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click lecture section “detail”</td>
<td>Generate Grade Report with details about each assignment for the specified Lecture Section in the main pane.</td>
<td></td>
</tr>
<tr>
<td>Click “Download All”</td>
<td>Generate Grade Report for all Lecture Sections combined</td>
<td></td>
</tr>
<tr>
<td>Test Case</td>
<td>Expected Results</td>
<td>Pass/Fail</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Click &quot;Change Password&quot;</td>
<td>User is taken to successful password changed page</td>
<td></td>
</tr>
</tbody>
</table>

**All Users - Change Password**

**Test Case** | **Expected Results** | **Pass/Fail** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Click &quot;Return&quot;</td>
<td>User is taken back to homepage.</td>
<td></td>
</tr>
</tbody>
</table>

**All Users - Successful Password Changes**

**Exception Testing:**

We will also perform exception testing to make sure that the program properly handles all possible situations that could occur. This will help ensure that no data is lost or corrupted due to loop-holes in the program or possible user error. The testing will aim to test possible exceptions to the different units. One such example of a possible exception test is trying to create a user that already exists.

**Section 3.2.7: Approach**

**Overall Approach to Testing:**

The overall approach to testing will include unit level testing. The units will primarily be tested individually. However, if a given unit has dependencies with one or more other units, an integration test will occur to ensure their compatibility.

**Testing Approach to be Used for Each Major Group of Features:**

The testing approach to be used with all major groups and features will be based on testing the individual units in the groups. The units will be tested by themselves, and when integrated they will then be tested to ensure that they function together properly. This bottom up approach will help ensure that problems are identified early, and will cut down on debugging time required.

**Major Activities, Techniques, and Tools Which are Going to be Used to Test the Groups:**

The major activities, techniques, and tools which will be used to test groups will be determined as the project progresses further into the development stages.

**Minimum Degree of Comprehensiveness Required:**
The minimum degree of comprehensiveness required will depend on the major activities, techniques and tools which will be decided upon at a later date.

**Techniques Which Will be Used to Judge Comprehensiveness:**

Techniques which will be used to judge comprehensiveness will be decided at a later date.

**Additional Completion Criteria:**

Any additional completion criteria will be determined at a later date.

**Techniques Which Will be Used to Trace Requirements:**

Techniques which will be used to trace requirements will be decided at a later date.

**Significant Constraints on Testing:**

The significant Constraints on testing are as follows:
- The limited number of people which are available for testing the application
- The deadline for the project is the Academic Celebration

**Section 3.2.8: Item Pass/Fail Criteria**

**Criteria to be Used to Determine Whether Each Test Passes or Fails:**

The criteria which will be used to determine whether the test item passes or fails testing is as follows:
- Task has to perform action specified
- Task has to perform action in a reasonable amount of time
- Task has to be performed with correct data
- Task has to be performed without errors
- Task has to display output correctly

The percentage of unit testing which will be considered acceptable before continuing forward will be discussed with Ms. Cotler and Dr. Hunter and will depend upon what items are being tested and the importance of the absolutely correctness of those items.

**Section 3.2.9: Suspension Criteria and Resumption Requirements**

**Criteria to be Used to Suspend the Testing Activity:**

Testing activity will be suspended under the following circumstances:
- Major code problems
- Minor code problems linked to only one module
- Extenuating circumstances which result in the inability of all team members to continue testing

Other criteria for testing suspension will be determined in the future upon discussion with our clients, Ms. Cotler and Dr. Hunter.

**Testing Activities Which Must be Redone When Testing is Resumed:**

- In the event of a major code problem all testing that can in anyway be related to the change should be retested, or if the change was extensive testing should be restarted
- In the event of a minor code problem all testing related to the module that was changed should be retested

**Section 3.2.10: Test Deliverables**

**Deliverable Documents:**

Documents that will be delivered upon the completion of testing will include:
- Updated test plan document (If any revisions are made)
- Test incident reports (SPRs)
- Test summary

**Test Input and Output Data:**

Test input data will be derived from the following locations:
- User input

Test data will be output to the following locations:
- Printer
- GUI

**Section 3.2.11: Testing Tasks**

**Tasks Necessary to Prepare for and Perform Testing:**
- Program needs to be in final stages of development
- All critical modules need to be completed
- Database need to be developed
- Database connections must be operational
- Files and user records must be stored in database

**Task Interdependencies:**
Task interdependencies exist between the database and the programs. The database needs to be developed and configured in order for other aspects to be designed to access and stored data. Likewise, parts of the program must be operational in order for users to retrieve and input data into the database.
Section 3.2.12: Environmental Needs

Environmental Needs:

- Server
  - Database software
  - Ethernet cards to connect to internet
- Desktop Computers
  - Allow users to access application
  - Ethernet cards to connect to internet
  - Microsoft Excel
  - Internet Browser

Required Level of Security:

Security information is as follows:
- Security level is MEDIUM
  - Program is not critical to operation of college
  - Limited access through Usernames and Passwords to preserve integrity of data
  - Usernames can be changed as frequently as needed
  - User Passwords can be changed as frequently as needed

Section 3.2.13: Responsibilities

Groups Responsible for Managing, Designing, Preparing, Executing, Witnessing, Checking, and Resolving Issues Involving Testing:

Performance Software Solutions is responsible for all aspects of testing

Groups Responsible for Providing Test Items Identified in the Test Items Section

Performance Software Solutions is responsible for providing all test items identified in the Test Items section

Groups Responsible for Providing the Environmental Needs Identified in the Environmental Needs section:

Siena College is responsible for providing all environmental needs identified in the Environmental Needs section

Section 3.2.14: Staffing and Training Needs

Staffing and Training Needs:

Staffing and training needs are as followed
  - Course Coordinator user
- Requires most training
- Responsible for overseeing other users
- Responsible for creating other users
- Responsible for deleting other users
- Responsible for editing other users
- Responsible for creating Lab/Lecture Sections
- Responsible for setting up the assignment list and weights
- Responsible for uploading the key files and template files

- Lab Instructor
  - Medium training
  - Responsible for submitting labs to the system for grading
  - Responsible for viewing graded labs
  - Responsible for leaving comments on graded labs
  - Responsible for manually overriding the grading software
  - Responsible for viewing lab section Grade Reports
  - Responsible for adding student non-lab grades into system

- Lecture Instructor
  - Basic training
  - Responsible for viewing their lecture section Grade Reports
  - Responsible for downloading student grades from all sections taught

- Student
  - Medium training
  - Responsible for submitting lab assignment
  - Responsible for viewing graded lab assignments
  - Responsible for viewing individual lab grades and overall course grades

Training Options for Providing Necessary Skills

Training for the application will be provided through the following means:
  - A typed step-by-step tutorial will be provided
    - How to add users
    - How to delete users
    - How to edit users
    - How to create Lab/Lecture Sections
    - How to set up the assignment list and weights
    - How to upload the key files and template files
    - How to submit labs to the system for grading
    - How to log in
    - How to view graded lab assignments
    - How to leave comments on graded labs
    - How to manually override the grading software
    - How to view Grade Reports
    - How to add non-lab grades into system
    - How to download grades from all sections
    - How to submit a lab assignment
    - How to view individual lab grades and overall course grades
- One-on-one training can be provided also initially to cover the same processes as are listed above

Section 3.2.15: Schedule

Test Milestones:

Test Milestones are as follows:
- Excel Grading System Database tested
- Database and application interaction tested
- Course Coordinator Login and abilities tested
- Created accounts for lecturer, lab instructor, and students tested
- Lab instructor abilities tested
- Lecturer abilities tested
- Student abilities tested
- Final pre acceptance test check
- Acceptance test

Estimate Time Required to do Each Testing Task:

Testing time required for each task will vary. However, the typical range of time to complete any given task should be between 30 seconds and 10 minutes.

Schedule for all Testing Tasks and Test Milestones:

Schedule for testing tasks and test milestones are still pending. All work and testing will be completed by the Siena College Academic Celebration.

Section 3.2.16: Risk and Contingencies

High-Risk Assumptions of the Test Plan:

High-risk assumptions of the test plan are that if one case for a given user works correctly, then the same case should work for all users. This is a high-risk assumption that is only being made because of the time constraints on our group.

Contingency Plans:

If this high-risk event came into reality then steps would have to be taken to correct the error in coding as quickly as possible. Then tests would need to be undertaken to ensure that the problem was correctly fixed.
Section 3.2.17: Approvals

Names and Titles for Approval:

Whitney Cave, Performance Software Team Leader

Ms. Cotler, Siena College Instructor

Dr. Hunter, Siena College Instructor

SPR Form  SPR #________

Name of individual(s) who identified problem:______________________________

Date problem was identified:  /  / 05

Time problem was identified:   :    AM / PM

Issue:

Possible Solution:

Initials:____
Initials: ______

**Actual Solution:**

Initials: ______

Person(s) that implemented solution: ________________________________

Date solution was implemented:  /  / 05

Time solution was implemented:  :  AM / PM

Amount of time spent on problem solution: _____Days   _____Hours   _____Minutes
## Performance Software
### Automated Grading System for Microsoft Excel Spreadsheets

**UNIT TEST**

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Proper Log-In (Student)</td>
<td>Attempt Log-In using valid student username and its corresponding password</td>
<td>Student Username &amp; Password</td>
<td>Generate web page for the student</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>Proper Log-In (Lecture Instructor)</td>
<td>Attempt Log-In using valid lecture instructor username and its corresponding password</td>
<td>Lecture Instructor Username &amp; Password</td>
<td>Generate web page for the lecture instructor having all of their classes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.3</td>
<td>Proper Log-In (Lab Instructor)</td>
<td>Attempt Log-In using valid lab instructor username and its corresponding password</td>
<td>Lab Instructor Username &amp; Password</td>
<td>Generate web page for the lab instructor having all of their classes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.4</td>
<td>Proper Log-In (Lecture &amp; Lab Instructor)</td>
<td>Attempt Log-In using valid lecture &amp; lab instructor username and its corresponding password</td>
<td>Lecture &amp; Lab Instructor Username &amp; Password</td>
<td>Generate web page for the lecture &amp; lab instructor having all of their classes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5</td>
<td>Proper Log-In (Course Coordinator)</td>
<td>Attempt Log-In using valid course coordinator username and its corresponding password</td>
<td>Course Coordinator Username &amp; Password</td>
<td>Generate web page for the Course Coordinator.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.6</td>
<td>Invalid Username</td>
<td>Attempt Log-In using invalid username.</td>
<td>Invalid Username and any password.</td>
<td>Generate message saying that either username or password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.7</td>
<td>Invalid Password</td>
<td>Attempt Log-In using valid username and invalid password.</td>
<td>Valid Username &amp; invalid password</td>
<td>Generate message saying that either username or password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.8</td>
<td>Invalid Passwords</td>
<td>Attempt to Log-In 3 times using valid username and invalid password.</td>
<td>Same username each time and any incorrect password</td>
<td>Generate message saying that either username or password is incorrect and that the account has been locked.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.9</td>
<td>Proper Username and Password to locked account</td>
<td>Attempt Log-In to a locked account using valid username and password.</td>
<td>Valid username &amp; password where username is an account that has been locked (see 1.8);</td>
<td>Generate message saying that the account has been locked.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Automated Grading System for Microsoft Excel Spreadsheets

## UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Log Out</td>
<td>The &quot;Log Out&quot; button is clicked</td>
<td>None</td>
<td>Log out of the Course Coordinator account and generate web page with appropriate message</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.2</td>
<td>Change Password</td>
<td>The &quot;Change Password&quot; button is clicked</td>
<td>None</td>
<td>Generate web page to change the Course Coordinator's password</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.3</td>
<td>All Assignments</td>
<td>The &quot;All&quot; Button is clicked under &quot;Assignments&quot;</td>
<td>None</td>
<td>The current web page remains</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.4</td>
<td>Specific Assignment</td>
<td>A specific assignment is clicked under &quot;Assignments&quot;</td>
<td>None</td>
<td>A web page is generated with the details of the specified assignment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>Add New Assignment</td>
<td>The &quot;Add New&quot; button is clicked under &quot;Assignments&quot;</td>
<td>None</td>
<td>Generate web page to add a new assignment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.6</td>
<td>Add/Edit Users</td>
<td>The &quot;Add/Edit Users&quot; button is clicked under &quot;Tasks&quot;</td>
<td>None</td>
<td>Generate a web page to add a user to the system or edit settings of current users</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.7</td>
<td>Add/Edit Sections</td>
<td>The &quot;Add/Edit Sections&quot; button is clicked under &quot;Tasks&quot;</td>
<td>None</td>
<td>Generate a web page to add a lab or lecture section, or to edit a current lab or lecture section</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.8</td>
<td>Add/Edit Categories</td>
<td>The &quot;Add/Edit Categories&quot; button is clicked under &quot;Tasks&quot;</td>
<td>None</td>
<td>Generate a web page to add a category for grading, or edit the settings of a current category</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.9</td>
<td>All User Reports</td>
<td>The &quot;All Users&quot; button is clicked under &quot;Tasks&quot;</td>
<td>None</td>
<td>Generate a web page with detailed reports based on all students' grades</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.10</td>
<td>Reports By Lab Section</td>
<td>The &quot;By Lab Section&quot; button is clicked under &quot;Reports&quot;</td>
<td>None</td>
<td>Generate a web page allowing the Course Coordinator to choose a lab section to create a report for</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.11</td>
<td>Reports By Lecture Section</td>
<td>The &quot;By Lecture Section&quot; button is clicked under &quot;Reports&quot;</td>
<td>None</td>
<td>Generate a web page allowing the Course Coordinator to choose a lecture section to create a report for</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.12</td>
<td>All Assignment Reports</td>
<td>The &quot;All Assignments&quot; button is clicked under &quot;Reports&quot;</td>
<td>None</td>
<td>Generate a web page providing details about each assignments from all students</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Performance Software
### Automated Grading System for Microsoft Excel Spreadsheets

#### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Search All Users</td>
<td>The “Search” button is clicked without any input entered</td>
<td>None</td>
<td>Generate web page with a listing of all users in the system</td>
<td></td>
</tr>
<tr>
<td>3.2</td>
<td>Search a Last Name</td>
<td>The “Search” button is clicked with only a last name entered</td>
<td>User last name</td>
<td>Generate web page listing all users with the entered last name</td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>Search a First Name</td>
<td>The “Search” button is clicked with only a first name entered</td>
<td>User first name</td>
<td>Generate a web page listing all users with the entered first name</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>Search a lecture section</td>
<td>The “Search” button is clicked with only a lecture section entered</td>
<td>Lecture Section</td>
<td>Generate a web page listing all users in the specified lecture section</td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>Search a lab section</td>
<td>The “Search” button is clicked with only a lab section entered</td>
<td>Lab Section</td>
<td>Generate a web page listing all users in the specified lab section</td>
<td></td>
</tr>
<tr>
<td>3.6</td>
<td>Search a specific user</td>
<td>The “Search” button is clicked with all or some fields entered</td>
<td>User last name, and/or User first name, and/or Lecture Section, and/or Lab Section</td>
<td>Generate a web page listing all users matching the specified categories</td>
<td></td>
</tr>
<tr>
<td>3.7</td>
<td>Invalid Search</td>
<td>The “Search” button is clicked with all or some fields entered</td>
<td>Incorrect User last name, and/or User first name, and/or Lecture Section, and/or Lab Section</td>
<td>Generate a web page with message explaining that no matching users were found in the system</td>
<td></td>
</tr>
<tr>
<td>3.8</td>
<td>Add a User</td>
<td>The “Add” button is clicked with all fields entered</td>
<td>Correct Last Name, First Name, User Name, User Type, Lecture Section, and/or Lab Section</td>
<td>Generate a web page with a message saying the new user has been entered</td>
<td></td>
</tr>
<tr>
<td>3.9</td>
<td>Add a User Incorrectly</td>
<td>The “Add” button is clicked with all or some fields entered</td>
<td>Last Name, and/or First Name, and/or User Name, and/or User Type, and/or Lecture Section, and/or Lab Section</td>
<td>Generate a web page with message explaining either not all fields were specified or the user already exists</td>
<td></td>
</tr>
<tr>
<td>3.10</td>
<td>Add a Group of Users</td>
<td>The “Add” button is clicked with all fields entered</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.11</td>
<td>Add a Group of Users Incorrectly</td>
<td>The “Add” button is clicked with all or some fields entered</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Performance Software

### Automated Grading System for Microsoft Excel Spreadsheets

#### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Search All Users</td>
<td>The “Search” button is clicked without any input entered</td>
<td>None</td>
<td>Generate web page with a listing of all users in the system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.2</td>
<td>Search a Last Name</td>
<td>The “Search” button is clicked with only a last name entered</td>
<td>User last name</td>
<td>Generate web page listing all users with the entered last name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>Search a First Name</td>
<td>The “Search” button is clicked with only a first name entered</td>
<td>User first name</td>
<td>Generate a web page listing all users with the entered first name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Search a lecture section</td>
<td>The “Search” button is clicked with only a lecture section entered</td>
<td>Lecture Section</td>
<td>Generate a web page listing all users in the specified lecture section</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>Search a lab section</td>
<td>The “Search” button is clicked with only a lab section entered</td>
<td>Lab Section</td>
<td>Generate a web page listing all users in the specified lab section</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.6</td>
<td>Search a specific user</td>
<td>The “Search” button is clicked with all or some fields entered</td>
<td>User last name, and/or User first name, and/or Lecture Section, and/or Lab Section</td>
<td>Generate a web page listing all users matching the specified categories</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.7</td>
<td>Invalid Search</td>
<td>The “Search” button is clicked with all or some fields entered</td>
<td>Incorrect User last name, and/or User first name, and/or Lecture Section, and/or Lab Section</td>
<td>Generate web page with message explaining that no matching users were found in the system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.8</td>
<td>Reset User Password</td>
<td>The “Reset Password” button is clicked next to a user</td>
<td>None</td>
<td>Generate web page allowing Course Coordinator to reset a user’s password</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.9</td>
<td>Delete User</td>
<td>The “Delete User” button is clicked next to a user</td>
<td>None</td>
<td>Generate web page asking if the Course Coordinator is sure he/she wants to delete the user from the system</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.10</td>
<td>Change Lab Section</td>
<td>“Submit” is clicked after selecting a different Lab Section next to a user in the drop down list</td>
<td>The selected Lab Section</td>
<td>Generate web page with a message stating the changes to the user</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.11</td>
<td>Change Lecture Section</td>
<td>“Submit” is clicked after selecting a different Lecture Section next to a user in the drop down list</td>
<td>The selected Lecture Section</td>
<td>Generate web page with a message stating the changes to the user</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Remove a lecture or lab section</td>
<td>The &quot;Remove&quot; button is clicked next to a lab or lecture section</td>
<td>None</td>
<td>Generate a web page asking if the section should really be removed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>Change Section Type</td>
<td>The &quot;Update&quot; button is clicked after changing the section type in the drop down menu</td>
<td>None</td>
<td>Generate an updated web page with the new type of section listed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>Change the Section Instructor</td>
<td>The &quot;Update&quot; button is clicked after changing the instructor in the drop down menu</td>
<td>None</td>
<td>Generate an updated web page with the new instructor listed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>Add Section</td>
<td>&quot;Add&quot; button is clicked after entering the categories</td>
<td>Section number, type, day and time, and instructor</td>
<td>Generate an updated web page with the new course entered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.5</td>
<td>Add Section Incorrectly</td>
<td>&quot;Add&quot; button is clicked after entering some or all of the categories</td>
<td>Section number, and/or type, and/or day and time, and/or instructor</td>
<td>Generate web page with a message indicating the fields were entered incorrectly</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Automated Grading System for Microsoft Excel Spreadsheets

## UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>Invalid Percentage Value for Assignments</td>
<td>Total percentage values for the categories do not add up to 100%</td>
<td>Invalid percentage is entered</td>
<td>Generate Message saying the percentages of the categories don't add up to 100%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.2</td>
<td>Invalid Percentage Value</td>
<td>The value entered for each of the categories</td>
<td>Value that is not a numeric value, or is greater than 100% or less than 0%</td>
<td>Generate message saying an invalid % has been entered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td>Proper Remove Button</td>
<td>Attempt to remove an assignment category</td>
<td>Request to remove a category</td>
<td>Generate webpage minus the removed category</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td>Invalid Remove Button</td>
<td>Attempt to remove an assignment category</td>
<td>Invalid request to remove a category</td>
<td>Generate message saying why the assignment couldn't be removed</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.5</td>
<td>Proper Add Button</td>
<td>Attempt to create an assignment category</td>
<td>Valid request to create a new category</td>
<td>Generate webpage with the specified category added to the category list</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.6</td>
<td>Invalid Add Button</td>
<td>Attempt to create an assignment category</td>
<td>Invalid request to create a new category</td>
<td>Generate message saying why the assignment couldn't be created</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.7</td>
<td>Proper Update Button</td>
<td>Attempt to update page with changed category values</td>
<td>Valid request to update</td>
<td>Generate webpage with all the updated values</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.8</td>
<td>Invalid Update Button</td>
<td>Attempt to update page with changed category values</td>
<td>Invalid request to update</td>
<td>Generate message saying why the values could not be updated</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Name</td>
<td>Description</td>
<td>Input</td>
<td>Expected Output</td>
<td>Pass</td>
<td>Comments</td>
</tr>
<tr>
<td>-----</td>
<td>--------------------</td>
<td>--------------------------------------</td>
<td>------------------------------------</td>
<td>---------------------------------------------------------------------</td>
<td>------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>7.1</td>
<td>Proper Create Assignment</td>
<td>Attempt to create a new assignment shell</td>
<td>Valid new assignment name</td>
<td>Generate webpage with the assignment shell created</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>Invalid Create Assignment</td>
<td>Attempt to create a new assignment shell</td>
<td>Invalid new assignment name</td>
<td>Generate message saying why the new assignment shell has not been created</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Performance Software

### Automated Grading System for Microsoft Excel Spreadsheets

#### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1</td>
<td>Proper Description</td>
<td>Attempt to select a description of the assignment that is being edited</td>
<td>Valid Description of the file that is to be added/edited</td>
<td>Generate the description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.2</td>
<td>invalid Description</td>
<td>Attempt to select a description of the assignment that is being edited</td>
<td>Invalid description of the file that is to be added/edited</td>
<td>Generate message saying the description is invalid</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.3</td>
<td>Proper Category is selected</td>
<td>Attempt to select a category from the dropdown list</td>
<td>Valid category is selected</td>
<td>Generate the category</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.4</td>
<td>Invalid Category is selected</td>
<td>Attempt to remove an assignment category</td>
<td>Invalid category is selected</td>
<td>Generate message saying the category is invalid</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.5</td>
<td>Proper Submit Changes Button</td>
<td>Attempt to submit changes that have been made to an assignment</td>
<td>Valid request to submit the changes</td>
<td>Generate webpage with the proper changes being made</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.6</td>
<td>Invalid Submit Changes Button</td>
<td>Attempt to submit changes that have been made to an assignment</td>
<td>Invalid request to submit the changes</td>
<td>Generate message saying why the changes could not be submitted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.7</td>
<td>Proper Add File</td>
<td>Attempt to add a file to an assignment</td>
<td>Valid request to add a file to an assignment</td>
<td>Generate webpage with the file being added</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.8</td>
<td>Invalid Add File</td>
<td>Attempt to update page with changed category values</td>
<td>Invalid request to add a file to an assignment</td>
<td>Generate message saying why the file could not be added</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.9</td>
<td>Proper Submit Grading Files</td>
<td>Attempt to submit all the files that are required for an assignment to be graded</td>
<td>Valid request to submit the files</td>
<td>Generate message saying the files have been submitted for grading</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.10</td>
<td>Invalid Submit Grading Files</td>
<td>Attempt to submit all the files that are required for an assignment to be graded</td>
<td>Invalid request to submit the files</td>
<td>Generate message saying why the files can not be submitted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.11</td>
<td>Proper Delete Assignment</td>
<td>Attempt to delete an entire assignment from the assignment list</td>
<td>Valid request to delete an entire assignment from the assignment list</td>
<td>Generate webpage with the entire assignment being deleted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.12</td>
<td>Invalid Delete Assignment</td>
<td>Attempt to delete an entire assignment from the assignment list</td>
<td>Invalid request to delete an entire assignment from the assignment list</td>
<td>Generate message saying why the assignment could not be deleted</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.13</td>
<td>Proper Delete File</td>
<td>Attempt to delete a single file from an existing assignment</td>
<td>Valid request to delete a single file from an assignment list</td>
<td>Generate webpage with the specified file deleted from the list</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.14</td>
<td>Invalid Delete File</td>
<td>Attempt to delete a single file from an existing assignment</td>
<td>Invalid request to delete a single file from an assignment list</td>
<td>Generate message saying why the file could not be deleted from the list</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Name</td>
<td>Description</td>
<td>Input</td>
<td>Expected Output</td>
<td>Pass</td>
<td>Comments</td>
</tr>
<tr>
<td>----</td>
<td>---------------</td>
<td>--------------------------------------------</td>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------</td>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>9.1</td>
<td>Proper Browse</td>
<td>Opens up an explorer window to locate the file to be submitted</td>
<td>Valid request to browse for files to be submitted</td>
<td>Explorer window opens to locate where the files are</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.2</td>
<td>Invalid Browse</td>
<td>Opens up an explorer window to locate the file to be submitted</td>
<td>Invalid request to browse for files to be submitted</td>
<td>Generate message saying that the explorer window could not be opened</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.3</td>
<td>Proper Submit Files</td>
<td>Attempt to submit files to server to be graded</td>
<td>Valid request to submit files to the server</td>
<td>Generate message saying the files have been submitted to the server for grading</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.4</td>
<td>Invalid Submit Files</td>
<td>Attempt to submit files to server to be graded</td>
<td>Invalid request to submit files to the server</td>
<td>Generate message saying why the files have not been submitted to the server for grading</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Performance Software

**Automated Grading System for Microsoft Excel Spreadsheets**

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>Correct Submission</td>
<td>A Lecture and Lab Section is selected and the &quot;Continue&quot; button clicked</td>
<td>None</td>
<td>Bring student to log in page</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.2</td>
<td>Incorrect Submission</td>
<td>Either Lecture and/or Lab Section not selected</td>
<td>None</td>
<td>Show an error message asking student to fill in both fields</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.3</td>
<td>Cancel</td>
<td>The &quot;Cancel&quot; button is clicked</td>
<td>None</td>
<td>A web page is generated with the option to go back to initial log in or close the window</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Unit Test

### Unit Category:  Students - Side Panel Functions

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>Log Out</td>
<td>The &quot;Log Out&quot; button is clicked</td>
<td>None</td>
<td>Log out of the Course Coordinator account and generate web page with appropriate message</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.2</td>
<td>Change Password</td>
<td>The &quot;Change Password&quot; button is clicked</td>
<td>None</td>
<td>Generate web page to change the Course Coordinator's password</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.3</td>
<td>All Grades</td>
<td>The &quot;Click Here&quot; button is clicked</td>
<td>None</td>
<td>Generate web page showing all the student user's grades for all assignments</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.4</td>
<td>Assignments</td>
<td>A specific assignment is clicked under the Assignments category</td>
<td>None</td>
<td>Generate web page showing details of the specific assignment or the submission options</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Performance Software

#### Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.1</td>
<td>Submit Files</td>
<td>The &quot;Submit Files&quot; button is clicked</td>
<td>None</td>
<td>Generate web page with options for student to browse files to submit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Performance Software

Automated Grading System for Microsoft Excel Spreadsheets

UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.1</td>
<td>Proper Submit(Single File)</td>
<td>Attempt to submit a single file in slot A</td>
<td>Single File Slot A</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.2</td>
<td>Proper Submit(Single File Slot B)</td>
<td>Attempt to submit a single file in slot B</td>
<td>Single File Slot B</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.3</td>
<td>Proper Submit(Single File Slot C)</td>
<td>Attempt to submit a single file in slot C</td>
<td>Single File Slot C</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.4</td>
<td>Proper Submit(Two Files Slots A and C)</td>
<td>Attempt to submit two files in slots A and C</td>
<td>Two Files, Slots A and C</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.5</td>
<td>Proper Submit(Two Files Slots A and B)</td>
<td>Attempt to submit two files in slots A and B</td>
<td>Two Files, Slots A and B</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.6</td>
<td>Proper Submit(Two Files Slots B and C)</td>
<td>Attempt to submit two files in slots B and C</td>
<td>Two Files, Slots B and C</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.7</td>
<td>Proper Submit(Three Files Slots A, B and C)</td>
<td>Attempt to submit three files in slots A, B and C</td>
<td>Three Files, Slots A, B and C</td>
<td>Generate successful upload webpage</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.8</td>
<td>Incorrect Submit(File A Error)</td>
<td>Attempt to submit files with slot A pointing to an incorrect address.</td>
<td>Incorrect address in slot A</td>
<td>Generate webpage informing that file A could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.9</td>
<td>Incorrect Submit(File B Error)</td>
<td>Attempt to submit files with slot B pointing to an incorrect address.</td>
<td>Incorrect address in slot B</td>
<td>Generate webpage informing that file B could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.10</td>
<td>Incorrect Submit(File C Error)</td>
<td>Attempt to submit files with slot C pointing to an incorrect address.</td>
<td>Incorrect address in slot C</td>
<td>Generate webpage informing that file C could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.11</td>
<td>Incorrect Submit(File A and B Error)</td>
<td>Attempt to submit files with slot A and slot B pointing to incorrect addresses.</td>
<td>Incorrect address in slot A and slot B</td>
<td>Generate webpage informing that file A and file B could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.12</td>
<td>Incorrect Submit(File A and C Error)</td>
<td>Attempt to submit files with slot A and slot C pointing to incorrect addresses.</td>
<td>Incorrect address in slot A and slot C</td>
<td>Generate webpage informing that file A and file C could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.13</td>
<td>Incorrect Submit(File B and C Error)</td>
<td>Attempt to submit files with slot B and slot C pointing to incorrect addresses.</td>
<td>Incorrect address in slot B and slot C</td>
<td>Generate webpage informing that file B and file C could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.14</td>
<td>Incorrect Submit(File A, B and C Error)</td>
<td>Attempt to submit files with slot A, B and slot C pointing to incorrect addresses.</td>
<td>Incorrect address in slot A, B and slot C</td>
<td>Generate webpage informing that file A, B and file C could not be found and request resubmit</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

**Unit Category:** Lab and Lecture Instructors - Side Panel Functions

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.1</td>
<td>Logout</td>
<td>Lecturer is logged out of system.</td>
<td>Lecture Instructor Username &amp; Password</td>
<td>Generate web page confirming that the Lecture Instructor has logged out</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.2</td>
<td>Change Password</td>
<td>Lecture Instructor creates a new password for their account.</td>
<td>Lecture Instructor Username, current Password &amp; the new desired Password entered twice.</td>
<td>Generate web page that has a change password form.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.1</td>
<td>Select Specific Lecture Section to view</td>
<td>An individual Lecture Section is selected on for viewing.</td>
<td>Lecture Instructor Username &amp; Password &amp; the Course Identification Number.</td>
<td>Generate Grade Report for the specified Lecture Section in the main pane.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.2</td>
<td>Select Specific Lecture Section to view in detail</td>
<td>An individual Lecture Section is selected on for viewing in detail</td>
<td>Lecture Instructor Username &amp; Password &amp; the Section Identification Number.</td>
<td>Generate Grade Report with details about each assignment for the specified Lecture Section in the main pane.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.3</td>
<td>Download All course sections available</td>
<td>All Lecture sections are selected for viewing.</td>
<td>Lecture Instructor Username &amp; Password &amp; the Section Identification Numbers for each section.</td>
<td>Generate Grade Report for all Lecture Sections combined in the main pane.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.4</td>
<td>View Lab Section</td>
<td>An individual Lab Section is selected for viewing.</td>
<td>Lecture Instructor Username &amp; Password &amp; the Lab Identification Number.</td>
<td>Generate Grade Report for the specified Lab Section in the main pane.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.5</td>
<td>View Lab Section in detail</td>
<td>An individual Lab Section is selected for viewing in detail.</td>
<td>Lecture Instructor Username &amp; Password &amp; the Lab Identification Numbers for each lab section.</td>
<td>Generate Grade Report for all Lab Sections combined in the main pane.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.6</td>
<td>View Printer Friendly Version</td>
<td>The page is selected to be viewed in a format that will print correctly.</td>
<td>No Input.</td>
<td>Generate a web page that is formatted to print properly.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Performance Software

## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.1</td>
<td>Link(lecture sections)</td>
<td>Lab instructor clicks on &quot;lecture sections&quot; link</td>
<td>mouse click</td>
<td>Brings user to the lecture sections screen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.2</td>
<td>Link(download all)</td>
<td>Lab instructor clicks on &quot;download all&quot; link</td>
<td>mouse click</td>
<td>Brings user to the download all screen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.3</td>
<td>Link(lab sections)</td>
<td>Lab instructor clicks on &quot;lab sections&quot; link</td>
<td>mouse click</td>
<td>Brings user to the lab sections screen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.4</td>
<td>Link(printer friendly version)</td>
<td>Lab instructor clicks on &quot;printer friendly version&quot; link</td>
<td>mouse click</td>
<td>Generates a printer friendly version of the current page</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.5</td>
<td>MouseOver(students name)</td>
<td>Lab instructor moves cursor over &quot;students name&quot;</td>
<td>mouse over</td>
<td>Provides additional information about the student without leaving current page</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.6</td>
<td>Link(students name)</td>
<td>Lab instructor clicks on &quot;students name&quot; link</td>
<td>mouse click</td>
<td>Brings user to a screen providing additional information about the student</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Automated Grading System for Microsoft Excel Spreadsheets

## UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>17.1</td>
<td>Proper Login (Lab Instructor)</td>
<td>Attempt login using valid lab instructor username and its corresponding password</td>
<td>Lab Instructor username &amp; Password</td>
<td>Login panel color changes and text changes to reflect status of logged in. Populate list of assignments ready to grade, sorted by section, assignment, file name. Assignments should come only from the instructors assigned classes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.2</td>
<td>Invalid Username</td>
<td>Attempt login using invalid username</td>
<td>Invalid username &amp; password</td>
<td>Generate message saying that either username or password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.3</td>
<td>Non-Lab Instructor Username</td>
<td>Attempt login using valid username for an identity that is not a Lab Instructor</td>
<td>Valid non-Lab instructor username &amp; password</td>
<td>Generate message saying that the user does not have the proper privileges.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.4</td>
<td>Invalid Password</td>
<td>Attempt login using valid username and invalid password</td>
<td>Valid username &amp; invalid password</td>
<td>Generate message saying that either username or password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.5</td>
<td>3 Invalid Passwords</td>
<td>Attempt to login 3 times using valid username and invalid password</td>
<td>Same username each time and any incorrect password</td>
<td>Generate message saying that either username or password is incorrect and that the account has been locked.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.6</td>
<td>Proper Username and Password to Locked Account</td>
<td>Attempt login to a locked account using valid username and password</td>
<td>Valid username &amp; password where username is an account that has been locked (see 1.8).</td>
<td>Generate message saying that the account has been locked.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Automated Grading System for Microsoft Excel Spreadsheets

## UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.1</td>
<td>Login Panel / Login Timeout</td>
<td>Let 10 minutes pass without having the program communicate with the server. Server independent tasks should not influence this.</td>
<td>None required</td>
<td>Panel color &amp; text should change to reflect state of &quot;logged out&quot;. Additionally, message describing that a session timeout occurred.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.2</td>
<td>Connect Button - Logged in State</td>
<td>Click login button before timeout occurs.</td>
<td>None required</td>
<td>Nothing should change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.3</td>
<td>Connect Button - Logged Out State</td>
<td>Attempt login using valid username (Lab Instructor) and password.</td>
<td>Valid username &amp; password.</td>
<td>Panel color &amp; text should change to reflect state of &quot;logged in&quot;. Other than login panel, screens should not refresh.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.4</td>
<td>Check for Files to Grade Button (Logged in)</td>
<td>Click button.</td>
<td>None required</td>
<td>Populate list of assignments ready to grade, sorted by section, assignment, file name. Assignments should come only from the instructors assigned classes.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.5</td>
<td>Check for Files to Grade Button (Logged out)</td>
<td>Click button.</td>
<td>None required</td>
<td>Generate message indicating session timeout and instruct user to sign in before continuing.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.6</td>
<td>Send Approved Files Button (Logged in)</td>
<td>Click button.</td>
<td>None required</td>
<td>Submit grade detail to server for all files that have been marked as having their final grade approved.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.7</td>
<td>Send Approved Files Button (Logged out)</td>
<td>Click button.</td>
<td>None required</td>
<td>Generate message indicating session timeout and instruct user to sign in before continuing.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.8</td>
<td>Ready to Grade Tab Handle</td>
<td>Click button.</td>
<td>None required</td>
<td>Change main panel to ‘Ready to Grade’ panel.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.9</td>
<td>Need Review Tab Handle</td>
<td>Click button.</td>
<td>None required</td>
<td>Change main panel to ‘Grade Assignments’ panel.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.1</td>
<td>&quot;Add&quot; button</td>
<td>Click button.</td>
<td>None required</td>
<td>Add highlighted records from 'Ready to Grade' list box (top) to 'Selected for Grading' list box (bottom).</td>
</tr>
<tr>
<td>19.2</td>
<td>&quot;Add All&quot; button</td>
<td>Click button.</td>
<td>None required</td>
<td>Add all records from 'Ready to Grade' list box (top) to 'Selected for Grading' list box (bottom).</td>
</tr>
<tr>
<td>19.3</td>
<td>&quot;Remove&quot; button</td>
<td>Click button.</td>
<td>None required</td>
<td>Remove highlighted records from 'Selected for Grading' list box (bottom) to 'Ready to Grade' list box (top).</td>
</tr>
<tr>
<td>19.4</td>
<td>&quot;Remove All&quot; button</td>
<td>Click button.</td>
<td>None required</td>
<td>Remove all records from 'Selected for Grading' list box (bottom) to 'Ready to Grade' list box (top).</td>
</tr>
<tr>
<td>19.5</td>
<td>'Grade Selected Assignments' button (Logged in)</td>
<td>Click button.</td>
<td>None required</td>
<td>Download files listed in list box as well as all necessary keys to temporary local folder. Process assignments through grading engine and populate initial grade reports for the 'Need Review' panel.</td>
</tr>
<tr>
<td>19.6</td>
<td>'Grade Selected Assignments' button (Logged out)</td>
<td>Click button.</td>
<td>None required</td>
<td>Generate message indicating session timeout and instruct user to sign in before continuing.</td>
</tr>
</tbody>
</table>
## Performance Software
### Automated Grading System for Microsoft Excel Spreadsheets

**UNIT TEST**

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.1</td>
<td>&quot;Student&quot;, &quot;Assignment&quot;, Files' labels</td>
<td>Click on a record in the list box.</td>
<td>None required</td>
<td>Labels update to reflect currently selected file.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.2</td>
<td>'View File' button</td>
<td>Click button.</td>
<td>None required</td>
<td>Open the currently selected file for review in read-only mode.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.3</td>
<td>'Previous Arrow' button</td>
<td>Click button.</td>
<td>None required</td>
<td>Entry number description should change to reflect current entry. Points worth label should reflect point value of current entry. Points awarded should reflect either score assigned by grading engine or persist the value entered by the user. Expected value text box should be populated from the grading key excel file. Actual value text box should be populated from the student’s file.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.4</td>
<td>'Next Arrow' button</td>
<td>Click button.</td>
<td>None required</td>
<td>Entry number description should change to reflect current entry. Points worth label should reflect point value of current entry. Points awarded should reflect either score assigned by grading engine or persist the value entered by the user. Expected value text box should be populated from the grading key excel file. Actual value text box should be populated from the student’s file.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.5</td>
<td>'Approved' Checkbox</td>
<td>Click box.</td>
<td>None required</td>
<td>This checked value should persist if a different record is selected for review. It will only be unset when the assignment is submitted or when the user unselects it.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Performance Software

## Automated Grading System for Microsoft Excel Spreadsheets

### UNIT TEST

**Unit Category:** All Users - Change Password

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
<th>Pass</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>21.1</td>
<td>Proper Password Change (Student)</td>
<td>Attempt to Change Password using valid Student username, its corresponding old password &amp; the new desired password</td>
<td>Student Username, old Password &amp; the new desired Password</td>
<td>Generate web page confirming successful password change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.2</td>
<td>Proper Password Change (Lecture Instructor)</td>
<td>Attempt to Change Password using valid lecture instructor username, its corresponding old password &amp; the new desired password</td>
<td>Lecture Instructor Username, old Password &amp; the new desired Password</td>
<td>Generate web page confirming successful password change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.3</td>
<td>Proper Password Change (Lab Instructor)</td>
<td>Attempt to Change Password using valid lab instructor username, its corresponding old password &amp; the new desired password</td>
<td>Lab Instructor Username, old Password &amp; the new desired Password</td>
<td>Generate web page confirming successful password change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.4</td>
<td>Proper Password Change (Lecture &amp; Lab Instructor)</td>
<td>Attempt to Change Password using valid lecture &amp; lab instructor username, its corresponding old password &amp; the new desired password</td>
<td>Lecture &amp; Lab Instructor Username, old Password &amp; the new desired Password</td>
<td>Generate web page confirming successful password change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.5</td>
<td>Proper Password Change (Course Coordinator)</td>
<td>Attempt to Change Password using valid course coordinator username, its corresponding old password &amp; the new desired password</td>
<td>Course Coordinator Username, old Password &amp; the new desired Password</td>
<td>Generate web page confirming successful password change.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.6</td>
<td>Invalid Username</td>
<td>Attempt to Change Password using invalid username.</td>
<td>Invalid Username, any old password &amp; any new password</td>
<td>Generate message saying that either username or old password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.7</td>
<td>Invalid Old Password</td>
<td>Attempt to Change Password using valid username and invalid old password</td>
<td>Valid Username, invalid old password &amp; any new password</td>
<td>Generate message saying that either username or old password is incorrect.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.8</td>
<td>Invalid Old Passwords</td>
<td>Attempt to Change Password 3 times using valid username and invalid old password</td>
<td>Same username each time, any invalid old password, &amp; new password</td>
<td>Generate message saying that either old password is incorrect and that the account has been locked.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.9</td>
<td>Invalid Retyped New Password</td>
<td>Attempt to Change Password using valid username and invalid retyped new password</td>
<td>Valid username, valid old password &amp; invalid retyped new password</td>
<td>Generate message saying that the retyped new password does not match the new desired password.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Performance Software

#### Automated Grading System for Microsoft Excel Spreadsheets

**UNIT TEST**

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Description</th>
<th>Input</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.1</td>
<td>Return (Student)</td>
<td>Attempt to return to Student homepage.</td>
<td>Student Username &amp; Password</td>
<td>Generate web page for the student.</td>
</tr>
<tr>
<td>22.2</td>
<td>Return (Lecture Instructor)</td>
<td>Attempt to return to Lecture Instructor homepage.</td>
<td>Lecture Instructor Username &amp; Password</td>
<td>Generate web page for the lecture instructor.</td>
</tr>
<tr>
<td>22.3</td>
<td>Return (Lab Instructor)</td>
<td>Attempt to return to Lab Instructor homepage.</td>
<td>Lab Instructor Username &amp; Password</td>
<td>Generate web page for the lab instructor.</td>
</tr>
<tr>
<td>22.4</td>
<td>Return (Lecture &amp; Lab Instructor)</td>
<td>Attempt to return to the Lecture &amp; Lab Instructor homepage.</td>
<td>Lecture &amp; Lab Instructor Username &amp; Password</td>
<td>Generate web page for the lecture &amp; lab instructor.</td>
</tr>
<tr>
<td>22.5</td>
<td>Return (Course Coordinator)</td>
<td>Attempt to return to the Course Coordinator homepage.</td>
<td>Course Coordinator Username &amp; Password</td>
<td>Generate web page for the course coordinator.</td>
</tr>
</tbody>
</table>
Section 3.4: Integration Tests

Integration testing is the phase of software testing in which individual software modules are combined and tested as a group. It follows unit testing and precedes system testing. During unit testing all the individual modules are tested separately, after these tests have been completed and the defects have been eliminated, the individual modules are grouped together in larger aggregates. These aggregates are then subjected to various tests that have been defined in the integration test plan. The output of integration testing is an integrated system that is ready for system testing. Many units are systematically combined into components, which are in turn aggregated into even larger parts of the program. The idea is to test combinations of pieces and eventually expand the process to test your modules with those of other groups. Eventually all the modules making up a process are tested together.

Ultimately, the goals of integration test are to:

- bring together the multiple pieces of a system
- find and fix defects that couldn’t be found earlier

The Importance of Integration Testing

Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis.

Common Strategies

- The top-down approach to integration testing requires the highest-level modules be test and integrated first. This allows high-level logic and data flow to be tested early in the process and it tends to minimize the need for drivers. However, the need for stubs complicates test management and low-level utilities are tested relatively late in the development cycle. Another disadvantage of top-down integration testing is its poor support for early release of limited functionality.

- The bottom-up approach requires the lowest-level units be tested and integrated first. These units are frequently referred to as utility modules. By using this approach, utility modules are tested early in the development process and the need for stubs is minimized. The downside, however, is that the need for drivers complicates test management and high-level logic and data flow are tested late. Like the top-down approach, the bottom-up approach also provides poor support for early release of limited functionality.
The third approach, sometimes referred to as the umbrella approach, requires testing along functional data and control-flow paths. First, the inputs for functions are integrated in the bottom-up pattern discussed above. The outputs for each function are then integrated in the top-down manner. The primary advantage of this approach is the degree of support for early release of limited functionality. It also helps minimize the need for stubs and drivers. The potential weaknesses of this approach are significant, however, in that it can be less systematic than the other two approaches, leading to the need for more regression testing.

Performance Software’s Approach:

Due to the various pros and cons of each of the different techniques, we will be selecting different pieces of each of the three approaches that have been described in order to create the most effective and thorough Integration Tests.

Examples of Integration Tests to Complete:

- If an instructor deletes a student from his/her class, the student does not show up as in that class in the course coordinator's account.

- If an instructor deletes a student from his/her class, the student does not have access to log into the system anymore.

- If the course coordinator deletes an instructor, they do not have access to log into the system, or view their old course.

- If the course coordinator deletes an assignment from the assignment list, the assignment does not show up in the student accounts.

- If the course coordinator adds users, class sections, or grading categories, the changes are spread throughout the system to make the pertinent changes to Students, Lecture Instructors and Lab Instructors.

- If the student submits an assignment, the course coordinator receives the files to be graded.

- When an assignment is graded, the updates are reflected in the students’ grade history.

- When an assignment is graded, the updates are reflected in the generated reports.
Section 3.5: System Tests

In the most general logic our System Testing will be an assessment process which takes place under artificial circumstances created for the Automated Grading System. This means we will set up example accounts, files for grading, etc., and carry out the Unit and Integration Tests in full in order to guarantee that the system creates the expected results, which have been identified in previous documents (specifically, the Requirements Specification Document).

More specifically, the following process will take place during our System Testing:

- Accounts will be created by the software developers for Students, Lab Instructors, and Lecture Instructors, with purely fictional details, from the point of view of the Course Coordinator. These accounts will all be logged into to ensure that they were created successfully and properly.
- The software developers will create assignment categories, expected lab assignments, key files for comparison in grading, and all necessary weights for categories, files, and lab portions.
- The fake Student accounts will be logged into and files will be submitted for grading.
- The Lab Instructor accounts will be logged into and the submitted student files will be downloaded so that the software developers can use the Grading Software to grade the submitted files.
- The Grading Software will be carried out by the software developers and overwriting, as well as accepting grades as correct will be attempted, along with making comments to the students.
- Once grades have been uploaded all associated accounts will be logged into (including the Course Coordinator, the Student, and the Student’s Lab and Lecture Instructor’s) in order to ensure the grades appear for each account. Other accounts will be logged into as well in order to make certain the grades do not appear on these accounts.

With the process just described being the most important portion of the Automated Grading System, it is the primary area we will focus on during System Testing. However, along with the general stages stated here, in order to ensure detailed correct functionality, we will be taking the created input through our Unit and Integration tests which have been previously described. We will also pay close attention to not only that correct output is occurring during this testing stage, but also that there is not unexpected output being generated in the system (i.e. one student’s account listing the detailed grades of another student, a student having instructor capabilities, etc.) because these are fine points that are important to catch before Acceptance Testing.
Section 3.6: Acceptance Tests

This section describes each function that each user in the EGS will be allowed to perform and describes what should happen upon performing these specific functions. This is how we will go about testing our software in our final testing step.

Course Coordinator Acceptance Test Criteria:

<table>
<thead>
<tr>
<th>Function Performed:</th>
<th>Necessary Outcome:</th>
</tr>
</thead>
</table>
| Log into system                                       | Successfully brings user to opening page  
An incorrect username produces an error message saying either username or password is incorrect  
Locks account after three incorrect passwords |
| Create a system profile for each instructor           | The account is created in the system  
The instructor has all required capabilities |
| Change instructor account passwords                   | The instructor’s previous password no will log the instructor into the system  
The new password now logs the instructor into the system |
| Remove instructor profiles in existence               | The instructor username and password are no longer recognized upon attempted login  
No students are listed as members in any lab or lecture sections of this instructor |
| Import a list of students to create student profiles   | A profile is successfully created for each student in the list  
All students have the ability to login, submit labs, and view grades |
| Change student account passwords                      | The student’s previous password no will log the student into the system  
The new password now logs the student into the system |
<table>
<thead>
<tr>
<th>Change lab and/or lecture section chosen by the students at initial login</th>
<th>The student no longer sees the student as enrolled in the original lab/lecture section under the student’s account. The student sees the student as enrolled in the new lab/lecture section under the student’s account. The instructor(s) of the original lab/lecture section no longer see the student in their list of that section or have the ability to view or grade labs. The instructor(s) of the new lab/lecture section now see the student in the instructor(s) list of that section or have the ability to view or grade labs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remove student profiles in existence</td>
<td>The student username and password are no longer recognized upon attempted login. No instructors see the deleted student as a member of the instructor’s sections.</td>
</tr>
<tr>
<td>Create lecture and lab sections</td>
<td>The section is created in the system and identified as either lab or lecture. The section has all required information (meeting days, time, section number, and instructor). The section is included as a choice in the drop down menu upon initial login for students. The section can be viewed by the instructor on the instructor’s sidebar when logged into the system, as well as accessed by clicking.</td>
</tr>
<tr>
<td>Change or remove sections already created</td>
<td>If the instructor is changed: the change is applied to all relevant parts of the system (dropdown menu for students’ initial login, and the listed instructor in student and course coordinator accounts), the instructor no longer has access to the section in the instructor’s profile, and the new instructor does have access. If the change is applied to meeting days, time, section number, or whether the section is lab or lecture: the change is applied to all relevant parts of the system (dropdown menu for students’ initial login, the listed change in student profiles, and the listed change in instructor and course coordinator account). If the change is a removal: the section is no longer visible as a section in any instructor’s account.</td>
</tr>
<tr>
<td>Set up assignment categories and category weights</td>
<td>A new category of the course is created (i.e. labs, midterm, final) and included in all accounts upon viewing student grades. When submitting files, the files are sent to the specified category. The grades within each category are weighted to account for the specified percentage of the final grade and this is represented through the final grade.</td>
</tr>
<tr>
<td>Modify or remove empty categories</td>
<td>The category is removed from view in all accounts of the system. The category no longer has any affect over the final course grade, or if the percentage is modified, it affects the course grade correctly.</td>
</tr>
<tr>
<td>Set up assignments for the system</td>
<td>The assignment is entered into the correct category in the system and weighted accordingly for the students’ final grade. The expected files are identified and prepared for accepting. Each file in the assignment has a point value attached to it.</td>
</tr>
<tr>
<td>Modify or remove the assignments already created</td>
<td>The modifications are made correctly so that there is a file added, or a point value changed, etc., and the changes apply accordingly to the grade of the assignment. The removal of the assignment makes the users unable to view it any longer and no longer counts toward the students’ grades.</td>
</tr>
<tr>
<td>Submit template files and key files for each expected file in assignments</td>
<td>The template file is added into the system and can be compared correctly to a submitted file by the student. The key file is added into the system and can be used correctly to create the grade outcome using point values of the different parts to the assignment.</td>
</tr>
<tr>
<td>View reports</td>
<td>The reports are shown with correct and up to date grades and averages of students based on either section of lab/lecture or on class average of section.</td>
</tr>
<tr>
<td>Change current password</td>
<td>The Course Coordinator’s previous password no will log the Course Coordinator into the system. The new password now logs the Course Coordinator into the system.</td>
</tr>
</tbody>
</table>
### Student Acceptance Test Criteria:

<table>
<thead>
<tr>
<th>Function Performed:</th>
<th>Necessary Outcome:</th>
</tr>
</thead>
</table>
| Log into system     | Successfully brings student to opening page  
An incorrect username produces an error message saying either username or password is incorrect  
Locks account after three incorrect passwords  
Initial login requires the student to specify lecture and lab sections |
| Submit a lab        | The files for each lab are uploaded separately into the system  
The lab instructor sees the submitted files as ready for grading  
The student sees the files were submitted successfully by having the ability to click on them |
| View graded labs    | When clicking on the date of the desired file, it is opened and the incorrect portions of the file are marked for visibility  
A comment section is included for the student to see specific notes from the instructor |
| View grades         | When click on lab section the table of labs submitted already, and those yet to be submitted is shown, along with grades of submitted labs |
| Change current password | The student’s previous password no will log the student into the system  
The new password now logs the student into the system |
## Lab Instructor Acceptance Test Criteria:

<table>
<thead>
<tr>
<th>Function Performed:</th>
<th>Necessary Outcome:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log into system</td>
<td>Successfully brings lab instructor to opening page</td>
</tr>
<tr>
<td></td>
<td>An incorrect username produces an error message saying either username or password is incorrect</td>
</tr>
<tr>
<td></td>
<td>Locks account after three incorrect passwords</td>
</tr>
<tr>
<td>Choose to grade a specified set of labs, or all the labs</td>
<td>The labs ready to be graded will be viewable when logged in</td>
</tr>
<tr>
<td>ready for grading</td>
<td>Check boxes are provided to indicate which should be graded by system; a select all option is included</td>
</tr>
<tr>
<td></td>
<td>Once marked and entered the files are graded by the system</td>
</tr>
<tr>
<td>Review the labs graded by software</td>
<td>Opens specified file when clicking on the date submitted</td>
</tr>
<tr>
<td></td>
<td>Incorrect portions are marked visibly</td>
</tr>
<tr>
<td></td>
<td>Comments can be seen that were added upon grading</td>
</tr>
<tr>
<td></td>
<td>Answers marked correct or incorrect by the software can be overridden by instructor</td>
</tr>
<tr>
<td>Manually enter grades into the system without submission</td>
<td>A file is shown that is not an excel file and may or may not be opened, but is included as part of the final</td>
</tr>
<tr>
<td>of a file</td>
<td>grade (the percentage of which is specified by what category it is entered under)</td>
</tr>
<tr>
<td>View graded labs for each student</td>
<td>Can open submitted and graded files by clicking on date submitted by student in the table of students and</td>
</tr>
<tr>
<td></td>
<td>view graded files with comments</td>
</tr>
<tr>
<td>View grades for each student</td>
<td>Reports of the grades for each student are visible with by clicking on the section the student is in and looking on the corresponding row of the student table</td>
</tr>
<tr>
<td>Change current password</td>
<td>The lab instructor’s previous password no will log the lab instructor into the system</td>
</tr>
<tr>
<td></td>
<td>The new password now logs the lab instructor into the system</td>
</tr>
</tbody>
</table>
### Lecture Instructor Acceptance Test Criteria:

<table>
<thead>
<tr>
<th>Function Performed</th>
<th>Necessary Outcome</th>
</tr>
</thead>
</table>
| Log into system    | Successfully brings lecture instructor to opening page  
|                    | An incorrect username produces an error message saying either username or password is incorrect  
|                    | Locks account after three incorrect passwords |
| Select a lecture section of students to view | Clicking on the section on the sidebar brings up the list of students in that section to the screen, including the students’ information |
| View lab section of each student and individual scores of each submitted lab, as well as each section of the lab | The list of students in each section includes a table which shows the grades of each file in the lab |
| Download the students’ grades | A file is downloaded to the instructor’s computer containing the grades of the students’ |
| Change current password | The lecture instructor’s previous password no will log lecture instructor into the system  
|                    | The new password now logs lecture instructor into the system |
Grading Software Acceptance Test Criteria:

<table>
<thead>
<tr>
<th>Function Preformed:</th>
<th>Necessary Outcome:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log into system</td>
<td>Successfully brings lab instructor to opening page</td>
</tr>
<tr>
<td></td>
<td>An incorrect username produces an error message saying either username or password is incorrect</td>
</tr>
<tr>
<td></td>
<td>Locks account after three incorrect passwords</td>
</tr>
<tr>
<td></td>
<td>Text and panel color changes</td>
</tr>
<tr>
<td></td>
<td>List of files ready to be graded appears</td>
</tr>
<tr>
<td>Login Timeout</td>
<td>After ten minutes without activity results in a change of panel and text color and a message indicating a login timeout has occurred</td>
</tr>
<tr>
<td>Login Time-in</td>
<td>Must re-enter username and password correctly to re-login</td>
</tr>
<tr>
<td>Check Files to Grade</td>
<td>If logged in a sorted list of files ready to grade appears</td>
</tr>
<tr>
<td></td>
<td>If logged out a message appears indicating the necessity of logging back in</td>
</tr>
<tr>
<td>Sending Approved Files</td>
<td>If logged in send grades to server that have been approved as complete and correct</td>
</tr>
<tr>
<td></td>
<td>If logged out create message indicating the necessity of logging back in</td>
</tr>
<tr>
<td>Ready to Grade/Need Review Buttons</td>
<td>When either button clicked change either to the Ready to Grade panel or the Grade Assignments panel</td>
</tr>
<tr>
<td>Ready to Grade Panel</td>
<td>Add assignments chosen for Ready to Grade to Selected for Grading list by clicking ‘Add’</td>
</tr>
<tr>
<td></td>
<td>Add all the assignments in Ready to Grade to Selected for Grading list by clicking ‘Add All’</td>
</tr>
<tr>
<td></td>
<td>Remove assignments chosen in Selected for Grading and put back in Ready to Grade list by clicking ‘Remove’</td>
</tr>
<tr>
<td></td>
<td>Remove all assignments in Selected for Grading list and put back in Ready to Grade list by clicking ‘Remove All’</td>
</tr>
<tr>
<td></td>
<td>Grade the assignments in Selected for Grading list by clicking ‘Grade Selected Assignments’ button and once done grading add to Need Review list</td>
</tr>
</tbody>
</table>
| Need Review Panel | View the files that have been graded by clicking ‘View File’
|                   | Navigate through assignments using arrow buttons, with each entry coming up with details about the grading performed and the option of overwriting grade or approving it |
Section 4: Performance Requirements

Section 4.1: Development/Production Environment

For the production of our Excel Grading Software, we will be using two machines. Our first machine is a Dell Dimension 4550, running the Windows XP operating system, currently updated with Service Pack 2. This machine has a 2.4-GHz Pentium 4 processor, a 37.2-gigabyte hard drive, and 512 MB of Ram.

The second machine is a Gateway, running the Microsoft Windows 2000 operating system, equipped with Service Pack 3. This machine has a 1300-MHz Pentium 4 processor, a 35.4-gigabyte hard drive, and 654,640 KB of Ram.

We will be using Oraserve, a server located in the CS Department. A server is a computer that handles requests for data, email, file transfers, and other network services from other computers (i.e., clients). We are using Oraserve version # 2.4.21-4.el. It’s operation system is Red Hat Enterprise Linux ES release.

Our software will have to be installed onto the user’s machine before use. Ideally, the software will be installed onto all of the computers in the CS-010 computer lab, as well as the Lab Instructor’s, Lecture instructor’s, and the Course Coordinator’s. This software is initially being designed for the Windows XP operating system, and will be compatible with all common web browsers. Machines running our software will also need Microsoft Office 2003 or a newer release.

We will be using the following software to assist in the creation of our software:

Apache – is open-source (source code is freely available and can be shared) HTTP Web server software. Apache is currently the most popular web server on the Net. Conveniently it supports the PHP language. We will be using Apache version # 2.0.46

Microsoft Visual Basic for Applications (VBA) - is an embeddable programming environment designed to enable developers to build custom solutions using the full power of Microsoft Visual Basic. We will be using VBA version #6.4

Oracle - a relational database management system (RDBMS) developed and copyrighted by the Oracle Corporation. We will be using Oracle version #10G

PHP (PHP Hypertext Pre-processor) - a HTML-embedded scripting language. The goal of the language is to allow web developers to write dynamically generated pages quickly. We will be using PHP version #4.3.11
Section 5: Sources of Information

Our information was gathered from meetings with our clients, Ms. Jami Cotler and Dr. Scott Hunter in addition to Dr. Lederman’s class lectures, the Software Engineering class textbook Software Engineering: A Practitioner’s Approach by Roger S. Pressman, and various Software Engineering teams’ projects from previous years located at: http://www.cs.siena.edu/~lederman/csis410/csis410.html
Section 6: Glossary of Terms

**Code:**
A system of symbols and rules used to represent instructions to a computer.

**Course Coordinator (CC):**
A user in the EGS which is the only account created by the software developers. This user is responsible for creating student and instructor accounts, and adding key and template files to be used to grade the submitted student labs.

**Database:**
A collection of data arranged for ease and speed of search and retrieval.

**Data Flow:**
Depicts the movement of one to many items of data. Data can enter a system from the outside.

**Data Store:**
A place where data is kept while it is not actively being processed. Data can only enter a data store from a process and can only exit a data store to a process.

**EGS:**
Excel Grading System

**ERD (Entity-Relationship Diagram):**
A graphical depiction of the conceptual structure of a database. It contains entities, attributes, and relationships. An attribute may be a primary or foreign key.

**External Entity:**
An entity that is outside the boundary of the system that is being modeled. It can either send data to the system or receive data from it. External entities are optional.

**Foreign Key:**
An attribute or combination of attributes that is not the primary key of the relation, but that is a primary key of another existing relation. A foreign key represents a logical connection between relations.

**Gantt Chart:**
A chart that depicts progress in relation to time, often used in planning and tracking a project.

**GUI - Graphical User Interface:**
A user interface based on graphics (icons, pictures, and menus) instead of text; uses a mouse as well as a keyboard as an input device.
Internet:
An interconnected system of networks that connects computers around the world via the TCP/IP protocol.

Key File:
This is a file added to the system by the Course Coordinator, which is used to specify what parts of the submitted student files to grade and how much each part is worth.

Lab Assignment:
The full assignment of lab included multiple files that all are put together to be one lab assignment as a portion of the final grade.

Lab Instructor:
A user of the EGS, the lab instructor account is set up by the Course Coordinator and has the ability to grade the labs submitted by the students in their lab sections, as well as view the grades of these students. The lab instructor can also override the grading done by the system and make comments on graded labs to their students.

Lab Assignment File:
One file of the lab assignment, each of which will be submitted separately by the students to make up one full lab assignment.

Lecture Instructor:
A user of the EGS, the lecture instructor account is set up by the Course Coordinator and has the ability to view the grades of all the students that are enrolled in their lecture courses.

Metrica:
A specific methodology for structuring and representing relationships in Data Flow Diagrams.

Primary Key:
An attribute or combination of attributes that is recognized as the normal way to identify entities and access records.

Process:
Signifies that something is happening to transform the data. Processes have numbers that reflect the decomposition hierarchy.

Software:
Written programs, procedures, or rules and associated documentation pertaining to the operation of a computer system and that are stored in read/write memory.

SSADM:
Structured Systems Analysis and Design Methodology
**Student:**
A user of the EGS, the student account is set up by the Course Coordinator and has the ability to submit labs to the system, as well as view their graded assignments.

**System:**
A group of independent but interrelated elements comprising a unified whole.

**Template File:**
This is file added to the system by the Course Coordinator, and is a correct copy of the files submitted by the students. It is used by the system to compare the student answers to the template file answers.
## Section 7: Gantt Chart

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Duration</th>
<th>January 2006</th>
<th>February 2006</th>
<th>March 2006</th>
<th>April 2006</th>
<th>May 2006</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team Meeting</td>
<td>74 days</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Client Meeting</td>
<td>96 days</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed Design</td>
<td>26 days</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed Design - Doc</td>
<td>1 day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed Design - Pres</td>
<td>1 day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acceptance Test</td>
<td>51 days</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acceptance Test - Doc</td>
<td>1 day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acceptance Test - Pres</td>
<td>1 day</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Gantt Chart Diagram]