



G R A D E

Beta Version

User Manual

Ivorine Do, Andy Jing, Albert Sze

TABLE OF CONTENTS

CONTENTS	
DESCRIPTION	2
MOTIVATION	2
USERS	2
SYSTEM ARCHITECTURE	2
ENTITY CLASSES	2
IMPORT CLASS	3
DATABASE	3
GUI CLASSES	4
USER INTERFACE – FUNCTIONAL FLOW	6
LOGIN	6
NEWUSER	7
DASHBOARD	7
ADD COURSE	8
IMPORT PAGE	8
BROWSE FILES	9
COURSE INFO	9
LAB INFO	10
ADD STUDENTS	11
COURSE PAGE	11
ADD COURSEWORK	12
EDIT COURSE	12
LAB PAGE	13
EDIT GRADES	13
EDIT LAB	14
STUDENT PROFILE	14

DESCRIPTION

Grade A is a program intended to assist college level professors to evaluate students in multiple courses, analyze the statistics of how the class is performing, maintain and secure student data, and adjust grades as he/she sees fit. This document details the system architecture and the user manual of the system's interface.

MOTIVATION

Although the idea of a grading program is not novel, there doesn't yet exist a free system that perfectly fulfills the needs of college professors. Many professors tend to rely on Excel to keep track of their student's grades. Excel is a great tool, with an abundant number of functionalities. However, having too many options can make it difficult or confusing for a professor with very specific requirements to use. Grade A is designed to meet this need by keeping what the benefits of Excel and simplifying it to be a user-friendly tool specifically catered towards college professors.

USERS

The target users for Grade A are university professors. Professors shall be able to create a username and password to create a profile for the courses they are teaching. Upon the main menu being pulled up (see User Interface section), a professor's dashboard will be load. A dashboard shall contain all the courses a professor is teaching, which includes the labs and students associated within them. A dashboard shall need a correct username and password in order to be accessed.

SYSTEM ARCHITECTURE

ENTITY CLASSES

The following are the classes can be found in the entity folder:

- Assignment.java
- Assistant.java
- Calculations .java
- Course.java
- GradeBreakDown.java
- Lab.java
- Student.java
- User.java

These files can be thought of as the core class that allow the front-end and back-end of the system to talk with one another. All these files' variables are private and have getter and setters to set variables from different classes.

The Assignment class keeps track of all the coursework or assignments that each student has. This includes variables for: coursework type, coursework name, total points, weight of coursework, course name the coursework is associated with, type percentage (categorical grade percentage), and coursework percentage. The grade type percentage and percentage keep track of both the undergraduate and graduate percentages. This class also calls a comparable interface to compare different coursework from one another.

Assistant class will keep track of a course's teaching fellow's or teaching assistant's information. The only information that will be kept are their names and emails.

The calculation class is a public abstract class. The main purpose is house the method to calculate the average of a coursework. This class is called upon in EditGrades, which will be discussed later.

The Course class has information of a course, which will be used to be displayed on the front end and will connect to the database. This class will maintain information of the courseID, course name, the time of the class, class duration, what days in the week the course is held, the teaching fellows/assistants associated with the course, and the lab section associated with the class.

GradeBreakDown is an important class, which is used to keep track of a coursework's name, type percentage, percentage, the type of coursework, the weight or extra points associated with it, total points, and points lost on the assignment. This class also calls a comparable interface, in order to compare different GradeBreakDown instances. The main distinguisher between Assignment and GradeBreakDown is that Assignment's purpose is to store the points lost, total points, and weights, while GradeBreakDown maintains the percentage and type percentage of each coursework.

A lab section's information can be called through the Lab class. Similar to the Course, this class will have variables for the lab section name, class time, teaching assistant assigned to the lab, and the days of the weekday that the lab is held.

Student class have variables for a student's name, student ID, the student type, photo path, their graduating year, and email. In order to get a student's list of coursework and grade, the system will gather the student's coursework information from the database using the student ID and the final grade will be calculated. This will be discussed a bit further in the Database section.

The User class is a very simple class, which only contains the variables for a username and the password associated with the username.

IMPORT CLASS

The following file is in the Import folder:

- ImportCSV.java

This file is used to import CSV file and upload the data into the database. The code will check if the file is a valid CSV file and that it is capable of being upload. If the incorrect file format is being upload into the system, the file will not be upload and error message will appear.

DATABASE

The files that will be used to talk to the SQL database are the following:

- | | |
|--------------------------|-------------------|
| • AssignmentDAO.java | • GradeDAO.java |
| • AssistantDAO.java | • LabDAO.java |
| • ConnectorDAO .java | • StudentDAO.java |
| • CourseDAO.java | • UserDAO.java |
| • GradeBreakDownDAO.java | |

All usernames and passwords shall be stored in a SQL database. An additional SQL database shall be created to maintain data regarding student information, students' grades, Teaching Fellows/Assistants, and Course Assistants. The databases shall be updated when changes have been made to a User's profile.

The following attributes will be part of the database:

- LID- Lab ID, lab section
- LabTime – Lab Time
- sname – Student Name
- sid –Student ID
- S_type – Student type (Undergraduate or Graduate Student)
- Photo – photo of student
- Email – Email of individual
- Cwname – course work name
- wid – work ID
- post_date – date course work is posted
- due_date –due date of course work
- CW_type – Course Work type (exam, homework, project, etc.)
- cname – course name
- cid – course ID
- classTime – class Time
- tfname – Teaching fellow/assistant name
- tfid – Teaching fellow/assistant ID
- username – username of User
- password – password associated with username

The attributes that each table contains is detailed below is the Entity Relation-diagram (ER Diagram). M and n represent the instance of an entity, which can appear more than once in the relationship table. The preliminary ER-diagram shown below and is subject to change.

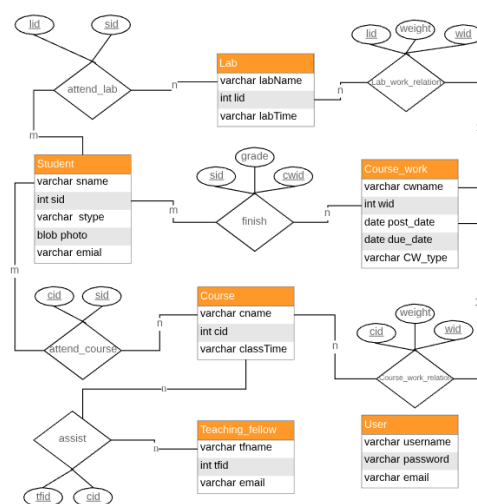


Figure 1: Database Structure

GUI CLASSES

The following are the GUI classes can be found in the GUI folder and are used for Grade A system:

- Login.java
- NewUser.java
- Dashboard.java
- AddCourse.java
- ImportPage.java
- BrowseFiles.java
- DatePicker.java
- CourseInfo.java
- LabInfo.java
- AddStudents.java
- CoursePage.java
- AddCoursework.java
- EditCourse.java
- LabPage.java
- StudentProfile.java
- EditGrades.java
- EditLab.java

Each of these class will be discussed a bit further in the User Interface section of this document. In order to run the program, the Login file needs to be run, which is the only file with a main function call. The login page is the start file, which will open a window with the login in page. In this class, the user will be asked to enter his/her username

and password in order to login. In addition, the user can ask to create a new user, which will direct him/her to the `NewUser` class.

In the `NewUser` class, the user will be asked to enter in a username and a password, which needs to be re-entered. Three security questions will be prompt to the user to answer. The purpose of the security question is for the event if the user forgets his/her username or password. However, the system currently does not allow for retrieval of password or username. The “forget username or password” functionality is planned to be included during the next iteration of this system.

After either login into the system or creating a new user, the professor will be sent to the `Dashboard` code. This code will call a frame to appear, which will have a table with all the professor’s courses and labs. The purpose of this code is to display the professor’s courses. Depending on what the user selects, he/she will be directed to `AddCourse`, `CoursePage`, or `LabPage`.

`AddCourse` is very simple, which will ask the user if he/she plans to create a new course from scratch or wants to import a CSV file. If the user wants to import a file he or she will be directed to `ImportPage`, `CourseInfo` if he/she wants to create a course from scratch, otherwise he/she will be returned to the `Dashboard`.

The `ImportPage` will prompt the user for a file path to a CSV file to upload. This file path can be given by selecting the browse button, which will send the user to the `BrowseFile`. `BrowseFile` will allow the user to visualize the folders and cycle to find the file. After selecting the file path, the user will return to the `Dashboard`.

The purpose of `CourseInfo` page is to allow the user to enter in all information regarding course, this will allow Grade A to enter the information into the database. Once the information has been entered, the user will be directed to `LabInfo`.

Like `CourseInfo`, `LabInfo` will ask the user to enter in the information for a lab section. At least one lab section needs to exist in order to add students to a course. `LabInfo`’s intention is to be able to add multiple labs at one time, however this feature has not been added to the beta version and is planned for future iterations. After all information has been entered, the user will then be directed to `AddStudents`.

`AddStudents`, as the filename suggests, is to add students to a class. On this page the user can add multiple students to a class. To add students to a course, the user must enter in the student’s name, email, lab section, student type, etc. The system will check the information and send the user back to either the dashboard or `Coursepage`, depending where he/she is coming from.

`Coursepage` can be accessed from the `Dashboard`. This code will once again show a window frame and will show an existing course information. This page will show averages for coursework and grade break down, in terms of type percentage. On this page, the user will be given many options to change the grade break down, add lab (redirect to `LabInfo`), add student (redirect to `AddStudents`), add coursework (redirect to `AddCoursework`), edit course (redirect to `EditCourse`), and delete the course. In order to implement edits to grade breakdown, the save button needs to be selected.

`AddCoursework` is a simple code, whose purpose is to add a new coursework to a course. The user will be asked what type of coursework he/she would like to add and the total amount of points of the assignment. Upon completion of this information, the user will return to `CoursePage`.

`EditCourse` will show all the information of an existing course and will look like `CourseInfo`. However, `EditCourse` will also display all the coursework and its information such as grade break down (for undergraduate and graduate students), extra points or weight, and total points. After edits have been made the user can save the information and will return to `CoursePage`.

`LabPage` can be accessed from the `Dashboard`. The purpose of this page is to display the students within a lab section and the Teaching Fellow associated with the lab section. Student profiles will be displayed on a button

panel. From here the user can redirect back to the dashboard, to an individual student's profile (StudentProfile), edit grades (EditGrades), or edit the lab's information (EditLab).

The purpose of EditGrades is to allow the user to change multiple student's grades on one page. Students that will be displayed on this page is dependent on a lab section. The students' grades as well as the average for a lab section will be displayed on a table. Once all the edits have been completed, the user can save the information and return to LabPage.

EditLab will display the all the information for a given lab and will look similar to LabInfo, but will only display one lab section's information. The user can edit information of a given lab section and is able to save the information via a "Save" button. Once information has been saved, the user will return to LabPage.

StudentProfile as the class name insinuate, will display an individual student's information and grades. The purpose of this page is to allow the user to edit a student's lab section, grades, and delete the student from the course. Once edits have been finalized for a student, the user will need to save the information, via "Save" button. the User can then return either to the LabPage or to the Dashboard (via home button).

USER INTERFACE – FUNCTIONAL FLOW

This section of the manual describes how to interface with the Grade A system.

LOGIN

Item	Description
1	Username
2	Password
3	Login button
4	Create User button

Figure 2: Login Page

Upon starting up the program, the professor shall be asked for their username and password, this is shown in Figure 2. User will enter he/she username and password in items 1 and 2; once completed, he/she can select the "Login" button. Credential will check if the username and password match prior to continuing to the dashboard.

If the professor does not have a username and password, he/she shall be given the option of creating one by selecting the "Create User" button, this will bring him/her to the NewUser page.

NEWUSER

Item	Description
1	Username
2	Password
3	Reenter Password
4.1	Question 1
4.2	Answer to Question 1
5.1	Question 2
5.2	Answer to Question 2
6.1	Question 2
6.2	Answer to Question 2
7	Finish Button
8	Cancel Button

Figure 3: New User Page

The New User page will ask for a username, password, and three security questions. A username will be entered into item 1 and a password will be entered into items 2 and 3. Three different questions will be asked from the combo box from items 4.1, 5.1, and 6.1. The answer of these question must be entered in in items 4.2, 5.2, and 6.2.

Once all information has been entered the user can press the “Finish” button. However, before moving onto the Dashboard, Grade A will check that the everything has been filled out and that the password and reentered password from items 2 and 3 match. The user can return to the Login page, by selecting the “Cancel” button.

DASHBOARD

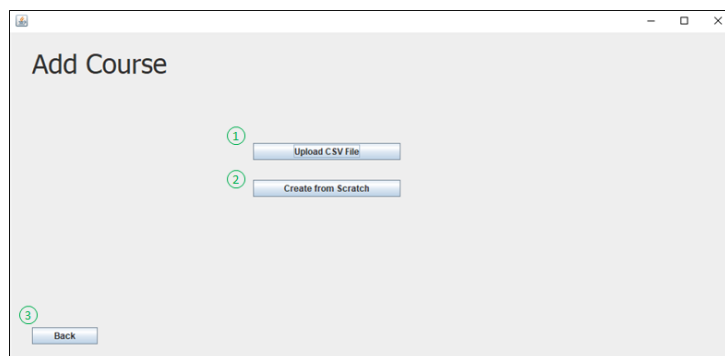
Item	Description
1	List of course and labs
2	Performance table
3	Add course button

Figure 4: Dashboard

After login in to the program, a user will be brought to the Dashboard. On this page, it shall show all the professor's courses as well as all the lab section in item 1. Item 2, it will show all the averages for classes and lab

sections, however this function has not been added to the beta version of Grade A and will be implement in future versions. The ability to add a new course will be available by selecting the “Add Course” button in the lower right-hand corner, this will direct the user to the AddCourse page. In addition, the user can cycle to the CoursePage and LabPage, by double clicking on a course or lab section.

ADDCOURSE

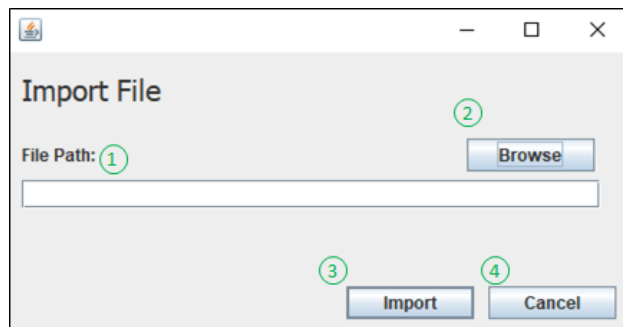


Item	Description
1	Upload CSV File button
2	Create from Scratch button
3	Back button

Figure 5: Add Course Page

If the “Add Course button” is selected on the CoursePage, it will bring the user to a window and ask if the user would like to import data from a CSV file or to create a new class from scratch. If the user selects “Upload CSV file”, he/she will be brought to the ImportPage, which will ask for the file path. The “Create from Scratch” button, item 2, will direct the user to the CourseInfo page to generate a new course. The back button in the lower left-hand corner will take the professor back to the Dashboard and will not create a new course.

IMPORTPAGE



Item	Description
1	File Path
2	Browse button
3	Import button
4	Cancel button

Figure 6: Import Page

The Import Page allows the user to import a CSV file. The File path in item 1, will allow the user to enter in a file path to a CSV file. This can also be done by browsing the user’s computer to locate the file by selecting the “Browse” button. Once the file path has been given, the user may select the “Import” button. The “Import: button (item 3) will check to ensure that the file path is indeed to a CSV file and that the CSV file imports properly. If the user would like to return to the AddCourse page, he/she can select the “Cancel” button in the bottom right-hand corner.

BROWSEFILES

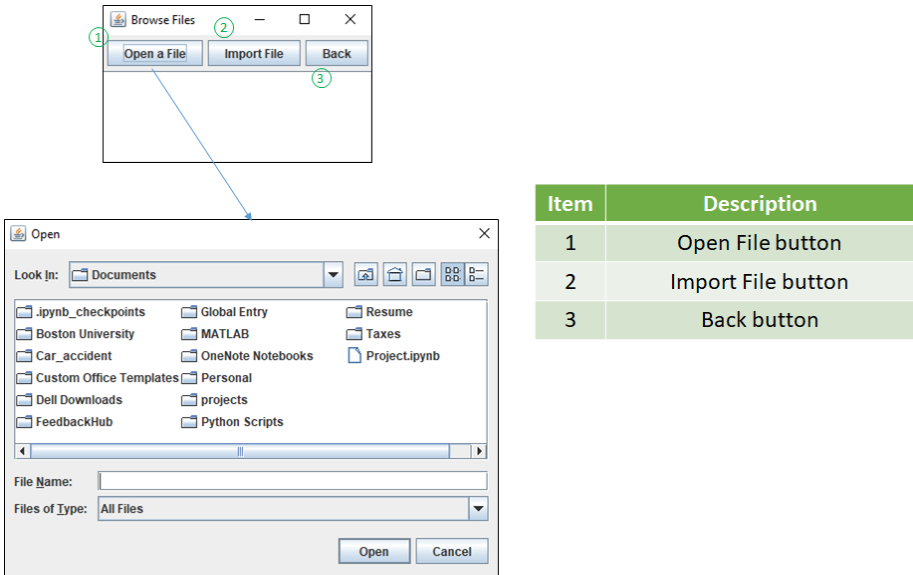


Figure 7: Browse Page

The BrowseFiles page allows the user to visually look through his/her computer to locate the csv file. Item 1, “Open a File” button will pop up a new window to browse through a computer’s folders. Once the file has been selected, the Import button (item 2), will import the file. To return to the ImportPage, the user needs to select item 3, “Back” button.

COURSEINFO

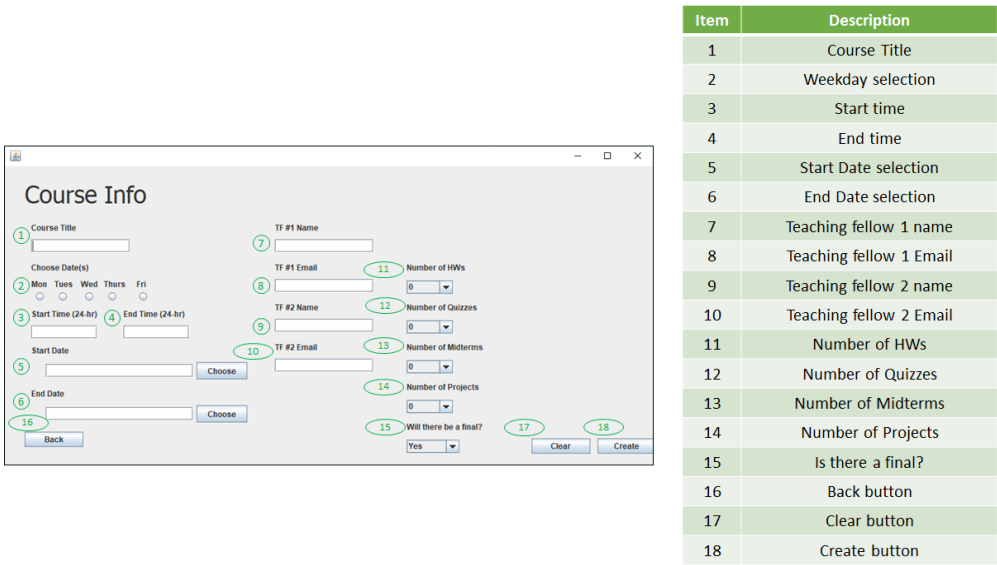
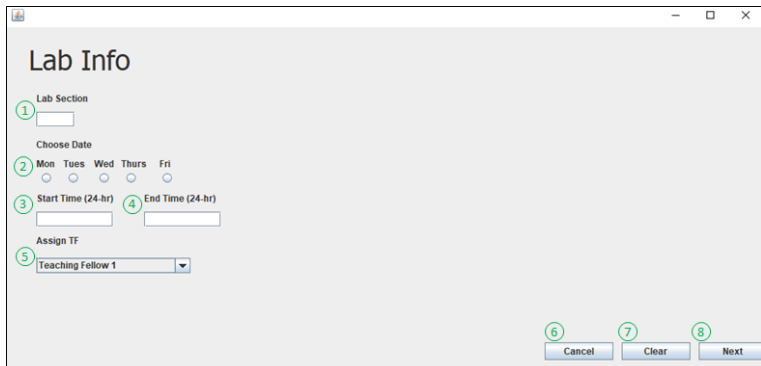


Figure 8: Course Info Page

If the user decided to select the “Create from Scratch” button it will bring him/her to a new page asking for the course information. In the course title section (item 1), the user can input the Course’s name. The radio buttons in item 2 allow the user to select what days the class is held during a week. Information of the start-time and end-time of the class can be added in items 3 and 4; these times must be entered in a 24-hour scale (00:00:00 –

23:59:59). Items 5 and 6 are inputs for date of the class duration, since classes normally run by a semester or trimester schedule. Pressing the choose button will bring up a pop-up window of a calendar, users can select the date of the class. A date can also be entered in manually in the following format “DD-MM-YYYY”. Teaching Fellows name and email can be added in items 7, 9, 8, and 10. Currently in the beta version, only two teaching fellows may be added per course. The number of course works (homework, quizzes, midterms, projects) and where there is a final can be entered using the combo boxes in items 11-15. The “Back” button in the lower left-hand corner will return the user to the AddCourse page. The “Clear” button, item 17, will clear all the information from all the text fields and combo boxes. The last button “Create”, item 18, checks that all items have been filled out and will then proceed to the LabInfo page.

LABINFO



Item	Description
1	Lab Section
2	Weekday selection
3	Starting time
4	Ending Time
5	Teaching Fellow Selections
6	Cancel button
7	Clear button
8	Next button

Figure 9: Lab Info Page

On the LabInfo page, the user can add a lab section’s information, like the CourseInfo page. Item 1 is where the lab section name can be specified. Similar to the Course Info page, item two allows to specify what days of the week the lab is held. Items 3 and 4 are where the starting and end time can be implemented, once again the time must be added in 24-hour time. Item 5 allows for the selection of a teaching fellow, which was entered in the Course Info page. Currently in the beta version labs can only be added one at a time. In the lower right-hand corner are three buttons: cancel, clear, and next. Selecting the “Cancel” button will not save any information and will return the user back to the Dashboard or Course Page, depending what page it had come from. The “Clear” button, like on the CourseInfo page, will clear the information currently present on the page. The “Next” button will save all the information that has been entered and will proceed to the AddStudents page.

ADDSTUDENTS

Item	Description
1	Add Student table
1.1	Lab selection
1.2	Student Type
2	Add Row button
3	Finish button
4	Cancel button

Figure 10: Adding a New Students

On this page the user can manual add students to a course. In table 1 (initializes with 10 rows) the user must enter in a student's last name, first name, student ID, their email, and their expected graduation year. Item 1.1 and 1.2 are combo boxes where the user can select which lab the student is a part of (item 1.1) and if the student is a graduate or undergraduate (item 1.2). If more students need to be added, the user can press on the "Add Row" button, which will add 1 more row. The "Finish" button will create the new students and add them according to the lab section. The "Finish" button will also check that all the information is filled out properly, otherwise it will warn the user. The "Cancel" button will not save the information that has been entered and will return to the Dashboard or CoursePage, depending where it came from.

COURSEPAGE

	hw_1	quiz_1	project_1	midterm_1	hw_2	final_1	quiz_2
Averages	42.5	52.67	68.89	70.67	50.33	61.0	63.0

	quiz	final	project	hw	midterm
Undergraduate	0.0	30.0	20.0	25.0	25.0
Graduate	20.0	20.0	20.0	20.0	20.0

Item	Description
1	Coursework Statistic table
2	Grade Breakdown table
3	Home button
4	Trash button
5	Save button
6	Add Lab button
7	Add Student button
8	Add Coursework button
9	Edit Course button

Figure 11: Course Page

On the course page the user will see the average of each coursework in table 1, the user is unable to edit this table. Table 2 consist of the grade breakdown for undergraduate and graduate students and is editable. The home button, item 3, will return the user to the Dashboard. A prompt will appear, upon selecting the trash button (item 4), which will ask if the user wants to delete the course. The "Save" button (item 5), will be enabled when changes to the grade break down table are detected. However, if the percentages do not equal 100% then the save button will not be enabled. Selecting the "Add Lab" button, item 6, will bring the user to the LabInfo page to add more labs to the course. When the "Add Student", item 7, will bring the user to the AddStudents Page. Item 8, "Add Coursework" button, will direct the user to the AddCoursework page. Finally, the "Edit Course" button, item 9, will go to the EditCourse Page, discussed later in this document.

ADDCOURSEWORK

Item	Description
1	Type of Coursework
2	Total Points
3	Cancel button
4	Add button

Figure 12: Add Coursework Page

The AddCoursework page is a relatively simple page. There is a drop-down menu (item 1) that allow the selection of an assignment type (Homework, Quiz, Project, Midterm, and Final). In the total points section, the user must add the total amount of points a coursework in area 2. To not save the changes and return to the CoursePage, a user can select the “Cancel” button. To add the coursework to the course itself, the user must select the “Add” button. After this button has been selected it returns to the CoursePage.

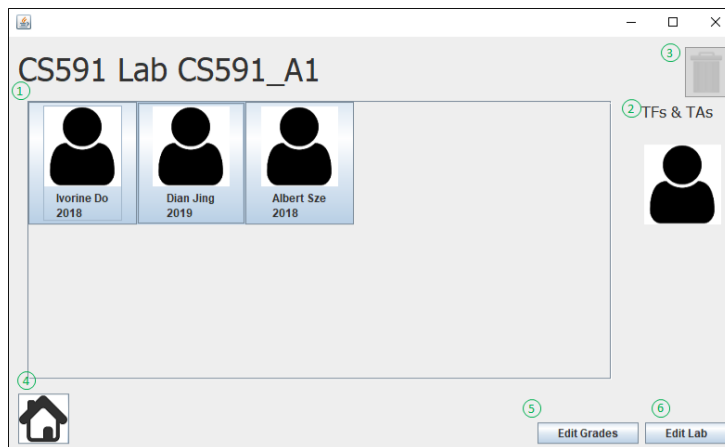
EDITCOURSE

Item	Description
1	Course Title
2	Weekday selection
3	Start time
4	End time
5	Start Date selection
6	End Date selection
7	Teaching fellow 1 name
8	Teaching fellow 1 Email
9	Teaching fellow 2 name
10	Teaching fellow 2 Email
11	Coursework grade breakdown
12	Save button
13	Cancel button
14	Clear button

Figure 13: Edit Course

On the EditCourse page, the course’s current information will be displayed in the proper areas. Item 1 shows the Course Title, which is changed. Item 2 are radio buttons to indicate the days in a week a class is held. Item 3, 4, 5, and 6 display the time the course starts, ends, the start date, and end date of a class respectively, all which can be changed. Like the CoursePage, item 7 and 9 display the Teaching Fellow’s names, while item 8 and 10 display the Teaching Fellow’s Emails. All these items are editable. There is a table, item 11, which lists each coursework and the respective coursework grade percentage (undergraduate and graduate), extra points (or weight), and total points. The percentage of a coursework depends on the type and must add up to 100%. For example, all the Homework percentage must add up to 100%. Once edits are completed the “Save” button must be selected in order for the changes to take effect. However, the changes will not go into effect if the percentages don’t equal up to 100% for each coursework type, instead it will display a warning. The “Cancel button”, item 13, will return the user to the Course Page and will not save the changes that have been made. Finally, the user can select the “Clear button, to clear all the course information, except for the coursework details, out of the text fields.

LABPAGE

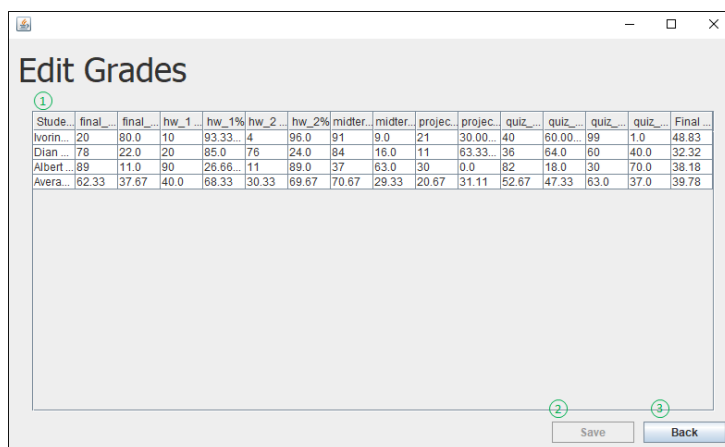


Item	Description
1	Student Profile Table
2	Teaching Fellows
3	Trash button
4	Home button
5	Edit Grades button
6	Edit Lab button

Figure 14: Lab Page

If a lab section is selected, it displays the lab section with all the students enrolled in that lab. This student profiles are displayed in table 1. Selecting one of these Student profiles will direct the user to an individual Student's profile, discussed later in StudentProfile. On the right of the table, item 2, is where the Teaching Fellow associated with the lab section is displayed. Currently, in the beta version only display a default icon image of students and teaching fellows, in future iterations, the ability to upload individual images for students and teaching fellow will be an option. A trash can icon is at the top right of the page (item 3), which will be enabled if there are no students in the lab. If the trash button is selected a pop-up window will appear confirming that the user wants to delete the lab. The lower left corner is a home button, item 4, which will return a user to the Dashboard. Located at the lower right-hand corner there are an "Edit Grades" and "Edit Lab" button, item 5 and 6. The "Edit Grades" button directs a user to the EditGrades Page, while the "Edit Lab" button will take him/her to the EditLab page.

EDITGRADES



Item	Description
1	All Students data
2	Save button
3	Back button

Figure 15: Edit Grades

This page will show all the students associated in a lab section with their grades on each assignment in a table, item 1. The table will have the points lost on an assignment, the score that the students received on each, and the final grade of each student. At the bottom of the table there is a row to show the averages of each assignment and final grade. In this table only, the points lost column can be manipulated. "Save" button (item 2) will save the changes to grade and will only be enabled once changes in the table have been detected with a valid value. Pressing the "Back" button, item 3, will return the user to the lab page, without saving changes to the grades.

EDITLAB

Item	Description
1	Lab Section
2	Weekday selection
3	Starting time
4	Ending Time
5	Teaching Fellow
6	Clear Button
7	Next Button

Figure 16: Edit Lab

This page will look like the LabInfo page but will only have the information for one lab section. Lab section (item 1) is a text field with the lab section name. Radio buttons, in item 2, display the days in the week that the lab is held. The start and end time of the lab will be displayed in item 3 and 4. Finally, a drop-down menu of Teaching Fellows for the class are displayed in item 5. All these items are editable and can be saved using the finish button, item 7, located in the lower right corner. The finish button will also return the user to the LabPage. The clear button, item 6, will clear the information from each cell, must like the other clear buttons.

STUDENTPROFILE

Assignment	Points Lost	Extra Points	Total Points Available	Percentage	Class Average
hw_1	2	2	120	100.0	92.0
hw_2	11	0	100	89.0	69.0
quiz_1	4	0	100	96.0	73.0
quiz_2	8	0	100	92.0	44.0
project_1	5	0	30	83.0	58.0
midterm_1	9	0	100	91.0	38.0
final_1	5	0	100	95.0	65.0

Item	Description
1	Lab Section Selection
2	Student's grades
3	Trash button
4	Home button
5	Save button
6	Back button

Figure 17: Individual Student Page

Selecting on Student's image will direct a user to an Individual student's profile page. Here the grades of each coursework, as well as the current status of the student is shown in table 2. In this table only, the Points Lost can be adjusted, all other cells are non-editable. The lab student's lab section is displayed in item 1 and can be changed through a drop-down menu of all the labs in the course. Other information such as name, graduating year, final grade, and image are shown in the top left corner. Currently only a default image is displayed, in future iterations, the image of the student should be changeable. In the top right corner is a trash button (item 3), when selected will ask the user to confirm if he/she wishes to delete the student from the course. In the bottom left hand corner, is a home button (item 4) to return the user to the Dashboard. Like the other pages, item 5 is a "Save" button, which will only be enable upon detecting a valid change to the table or lab section. If the "Save" button is selected, all the changes made to a student profile will be saved to the database. Lastly the "Back" button, item 6, will return the user to the LabPage and will not save the edits.