

---

# LMS PROJECT

---

Software Engineering



Alex Anthony

Juan Reyes

John Baer

Lucy Tea



# TABLE OF CONTENTS

## **1 Intro**

### **1.1 Purpose**

### **1.2 Model**

### **1.3 Scope**

## **2 Description**

### **2.1 Goals**

### **2.2 Functions**

### **2.3 Use Cases**

### **2.4 Operating Environment**

### **2.5 Design Constraints**

## **3. Interface Requirements**

### **3.1 User Interface**

## **4. System Design**

### **4.1 UML Diagram**

### **4.2 Use Case Design**

### **4.3 Database Design**

## **5. Team Coordination**

### **5.1 Github**

### **5.2 Other Communications**

## **6. Software Product Management Plan**

### **6.1 Software Product Management Plan**

## **1. Introduction**

### **1.1 Purpose**

The purpose of this document is to describe our software with unique details under the long process of software development and the software engineering process that our group has embarked into ultimately developing a user-friendly Learning Management System desktop application.

The various methods and strategies used in the project have guided us to fully develop a learning management system program that varies in its utilizations in according to the user.

### **1.2 Model Used for the Project:**

Our project utilized the Democratic approach, with no designated leader. Our project positively exploited the organic ability for various individuals to shine in their respective fields of expertise.

Additionally, our project used a combination of the Rapid Prototype life cycle model and the Iterative and Incremental life cycle model. The model was used due to the nature of the goals within the limited scope of the project itself. Specifically, this models' goal was to build a rapid prototype to let the client and future users interact and experiment with it. Once the initial prototype was created, then we were able to test certain features and make adjustments where needed to fit the clients needs, and also to fit any other constraints such as time, working knowledge of the programming language, etc.

### **1.1 Scope:**

The scope of this project was not only to improve our skills as future computer programmers, but to also to teach us how to work together on major programs and projects with others and to learn the software engineering process throughout the life of a product.

## 2. Project Description

### 2.1 Product Goals

The Goal for the manufacturing of the software has been to develop a learning management system. The functionalities of this system would work under the umbrella of a desktop application. The program would purposefully bifurcate the targeted users into an administrator and a student. Each who would have their respective and limited possibilities delineated by various levels of access, based on the appropriate responsibilities.

### 2.2 Product Functions

The main function is to pragmatically let students and administrators manage the sequential and tiresome bureaucratic reality of test-taking into a user interface user-friendly way of managing grades. The Administrator would be able to without restriction, add, delete, and update the student as well as his grades. The Student on the other hand, will be able with some restrictions to see his grades, and his Grade point Average ( GPA).

#### 1) Log-In:

- 1.1) If User is Admin: Can log-in to edit students and courses
- 1.2) If User is Student Can log-in to view grades courses and GPA.

#### 2) Add-Student

- 2.1) Once logged in only the admin can add a student.

#### 3) Remove Student:

- 3.1) Once logged in only the admin can remove a student

#### 4) Update Student:

- 4.1) Once logged in only the admin can update a student.

#### 5) View Students:

- 5.1) Once logged in, the Admin can view all the students and their respective grades.

#### 6) View Grades and Courses:

**6.1)** The user can view his exam grades and courses.

## **7) View GPA**

**7.1)** The user can view and calculate his GPA of a particular course.

## **2.3 USE Cases**

**STUDENT:** View Student; View Course; View GPA; View Grades

**ADMIN:** View Student; View Course; View GPA; View Grades; Calculate GPA; Modify Grades; Modify Student; Modify Course

## **2.4 Operating Environment**

The Software project was created by the utilization of Visual Studio as a tool to write our program in the C# language, with the interface WPF, to ultimately store the data in a Microsoft SQL server. This means that our product can only be operated in Windows Machines only.

More Specifically

- Operating System: Windows
- Database: SQL
- Platform: .NET/C#

## **2.5 Design and Implementation Constraints.**

1. Ideally the Database would be centralized.
2. Limited SQL knowledge

## **3. Interface Requirements**


### **3.1 User Interface**

For the user interface, there simply needed to be an professor view (admin), and a student view. The student view only has the means to see his grades, classes, and averages, while the professor has the ability to view grades, view student IDs, view/add/delete from the

class roster, and view/add/delete/update an exam. Then, once those changes are made. the student can view the updated changes.

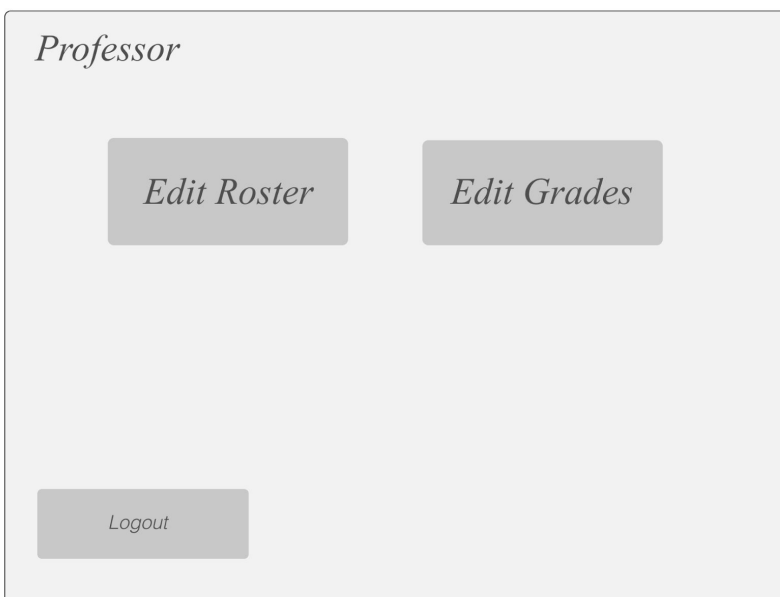
While in the design workflow, in addition to working on the the logical design, we also planned out how we wanted our user interface to work and perform.

Below are the mock ups for each of the views:



A login form mockup with a light gray background and rounded corners. At the top center is the word "LOGIN" in a large, bold, black serif font. Below it are two input fields: "Username" on the left and "Password" on the right, both in a medium gray rounded rectangle. At the bottom center is a "SUBMIT" button in a medium gray rounded rectangle with bold black text.

Professor dashboard:



A professor dashboard mockup with a light gray background and rounded corners. At the top left is the word "Professor" in an italicized black serif font. Below it are two buttons: "Edit Roster" on the left and "Edit Grades" on the right, both in a medium gray rounded rectangle with italicized black text. At the bottom left is a "Logout" button in a medium gray rounded rectangle with italicized black text.

## Professor: Edit Grades

Select Class:

Math 23XX

Current Grades:

Test 1 80

Homework 1 90

add

delete

edit

Select Student

Anthony, Alex	123456
Alvarado, Ricardo	234567
Caulter, Sam	345678
Smith, Kelly	456789

## Professor: Edit Roster

Select Class:

Math 23XX

Roster for class 23XX

Anthony, Alex  
Alvarado, Ricardo  
Caulter, Sam  
Smith, Kelly

last

first

Add

Remove

## Student Dashboard:

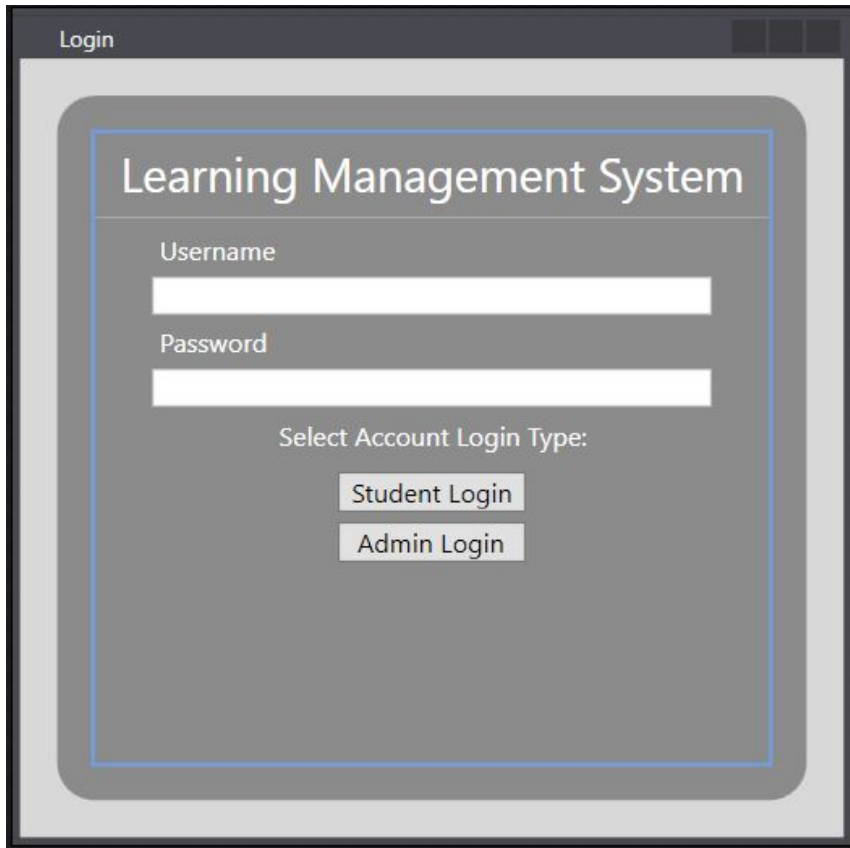
<i>Student</i>	
<i>Classes:</i>	
<i>Graphical Analysis</i>	<i>Test 1 90</i>
<i>Statistical Quantisis</i>	<i>Homework 1 85</i>
<i>Qualitative Initiative</i>	
	<i>GPA: 87.5</i>

Now, of course, the mock ups would be the fully polished version of our program, but due to time constraints and lack of expertise in the area of GUIs, we had to revise our design and functionality to meet the time limitations of the project.

Here is the result of our work:



Login Screen:



The image shows a login screen for a Learning Management System. It features a dark gray header bar with the word "Login" on the left and three small square icons on the right. The main content area is a light gray rounded rectangle containing a darker gray rounded rectangle. Inside this, the text "Learning Management System" is displayed in a large, white, sans-serif font. Below this, there are two white input fields: the first is labeled "Username" and the second is labeled "Password". Under the password field, the text "Select Account Login Type:" is centered. Below this text are two buttons: "Student Login" and "Admin Login", both with a light gray background and dark gray text. The entire interface is enclosed in a black border.

Login

# Learning Management System

Username

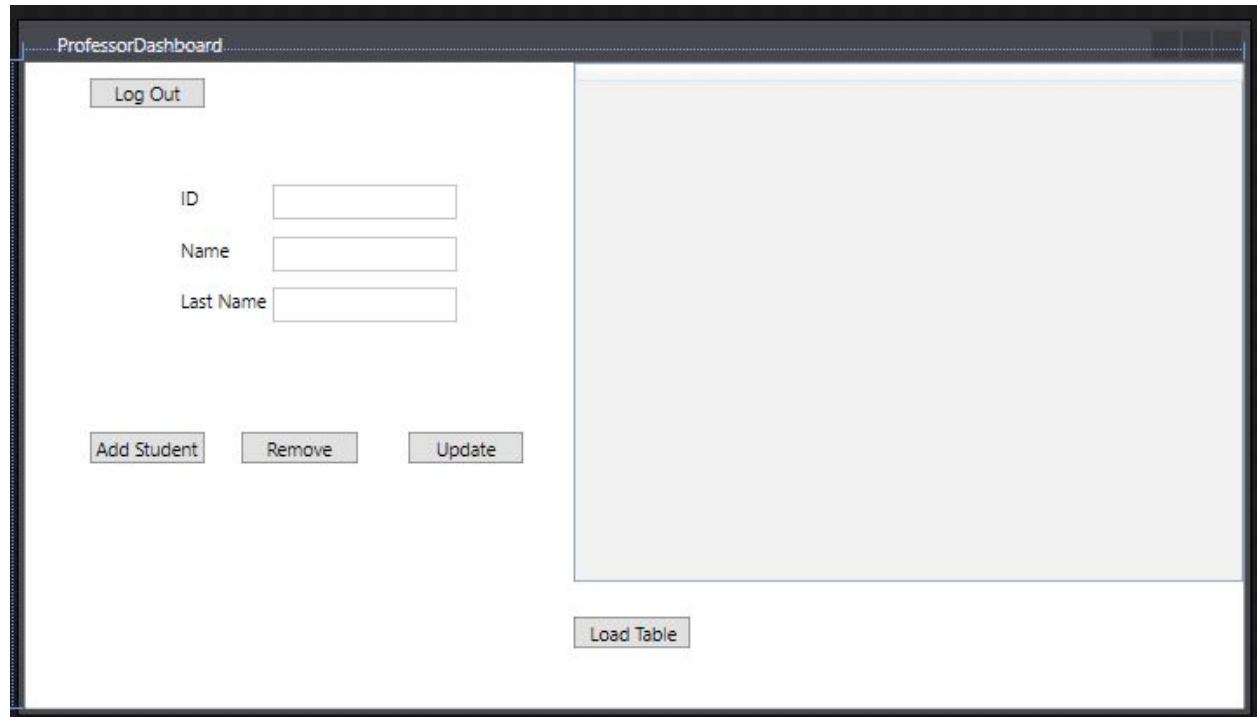
Password

Select Account Login Type:

Student Login

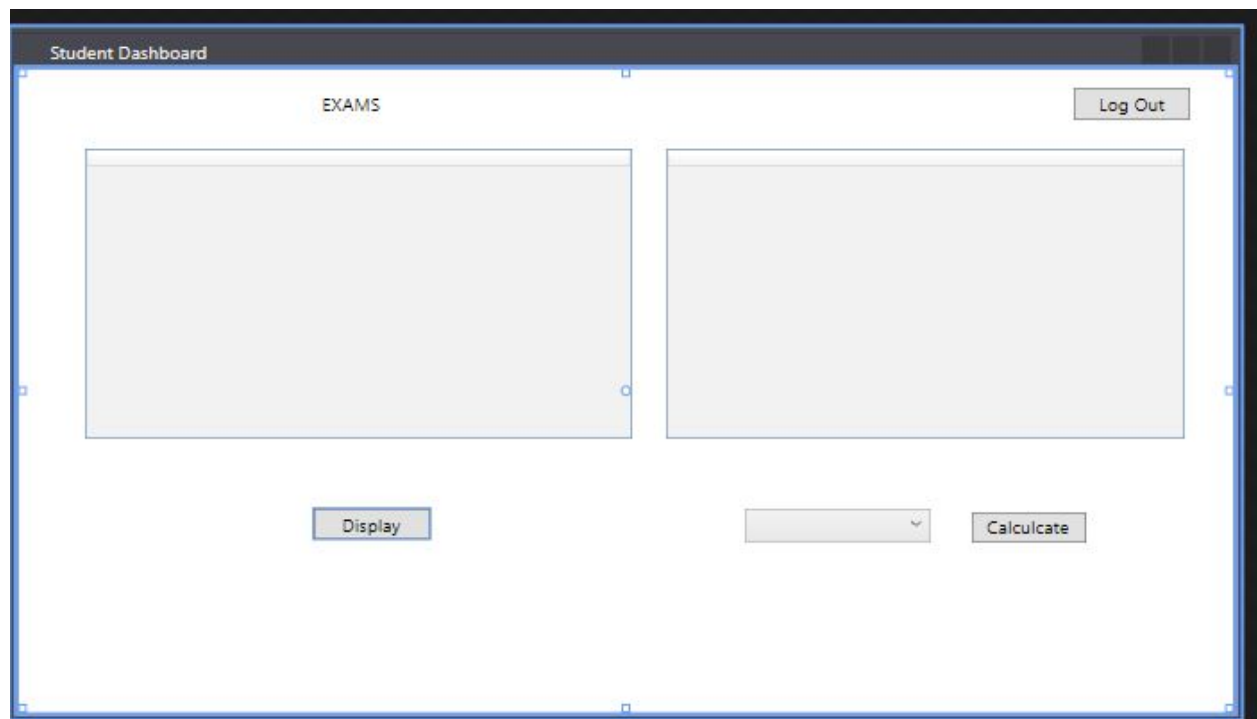
Admin Login

The professor dashboard:



The ProfessorDashboard window has a title bar labeled "ProfessorDashboard...". Inside, there is a "Log Out" button in the top left. Below it are three input fields labeled "ID", "Name", and "Last Name". At the bottom left are three buttons: "Add Student", "Remove", and "Update". A large, empty rectangular area occupies the right half of the window. At the bottom center is a "Load Table" button.

The student dashboard:

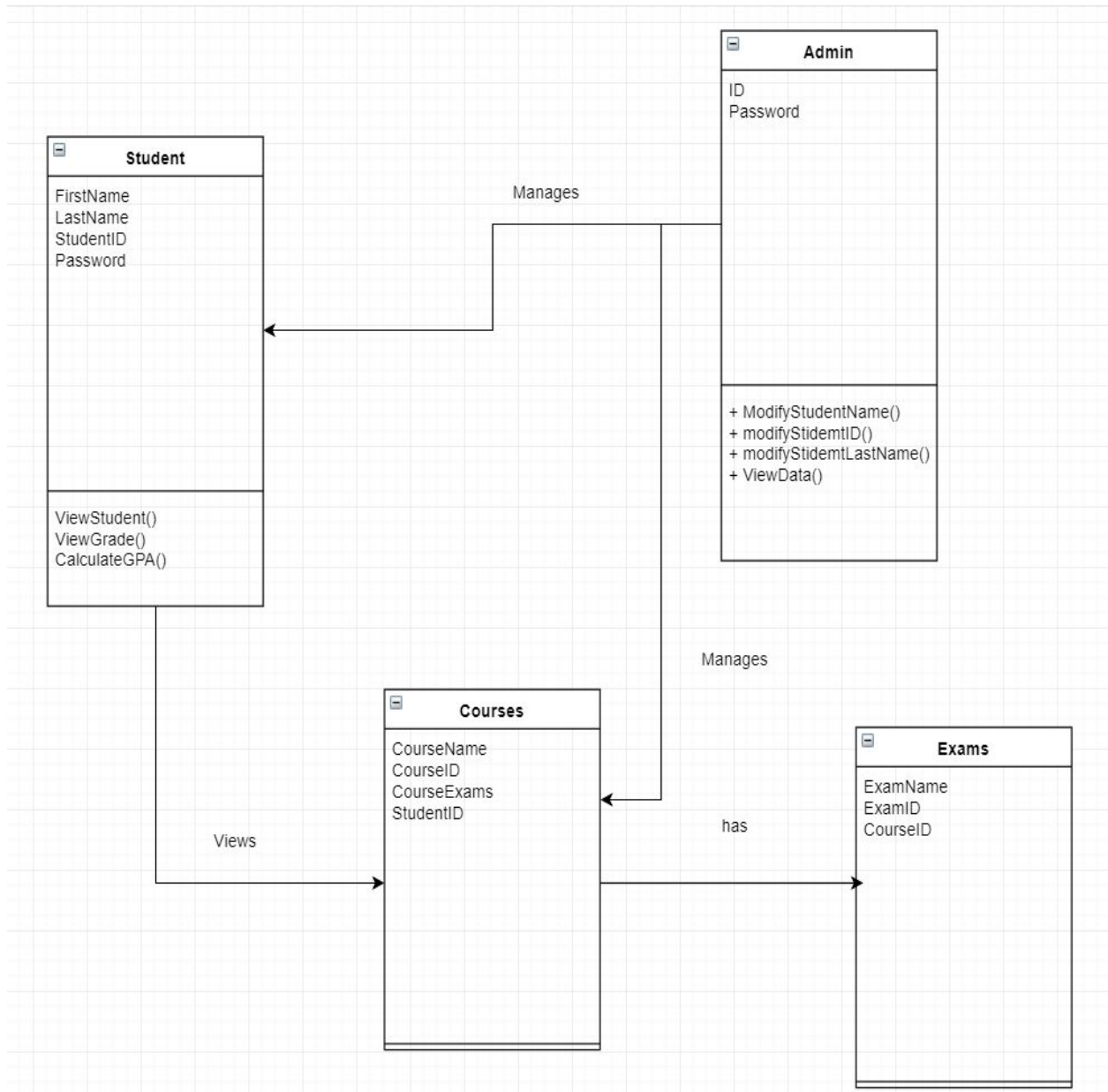


The Student Dashboard window has a title bar labeled "Student Dashboard". It features a "Log Out" button in the top right corner. Below the title bar, the word "EXAMS" is centered. The main area contains two large, empty rectangular boxes side-by-side. At the bottom, there is a "Display" button on the left, a dropdown menu in the center, and a "Calculate" button on the right.

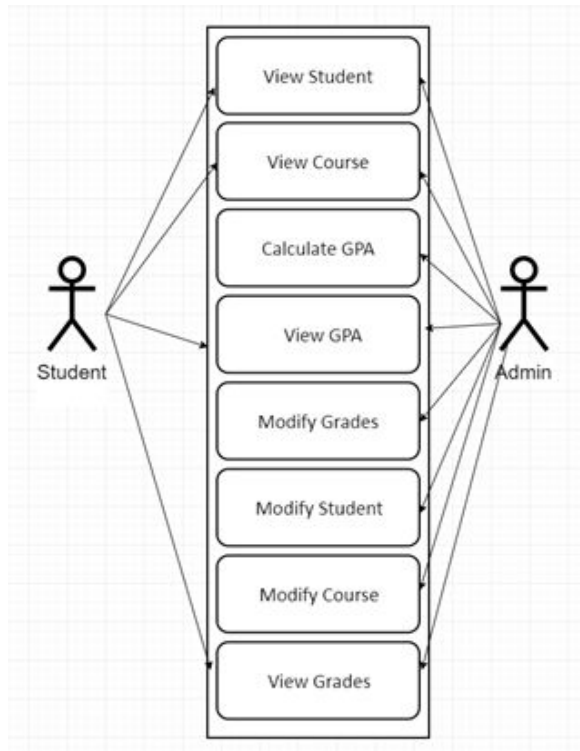
Although not as polished, the working prototype still displays all requirements for the LMS.

## 4. System Design

### 4.1 UML Design




## 4.2 Use Case Design



## 4.3 Database Design

AdminTable	
Username	
Password	

UserTable	
UserName	
Password	

StudentTable	
 StudentID	
StudentName	
StudentLastName	
GPA	

Students	
Student	
Exams	
Courses	
Professor	
Grades	

Exams	
ExamID	
Score	
CourselD	
StudentID	

Courses	
CourselD	
CourseName	
StudentID	

## 5. Team Coordination

### 5.1 Github

Our team utilized Github as our version control system. Using Github as our VCS served to our advantage in the fact that each of us was able to work on part of the project separately, and then merge all of our changes together and resolve any disputes.

The screenshot displays the GitHub interface for the repository **AVmachine / SoftwareProject**. At the top, there are navigation links for Code, Issues (0), Pull requests (1), Projects (0), Wiki, Insights, and Settings. The repository has 2 Unwatch, 2 Stars, and 0 Forks. The current branch is **master**.

The commit history for November 25, 2018, is shown below:

Commit Message	Author	Commit Hash	Actions
Merge pull request #1 from AVmachine/Ita_testing	Verified	d06c9ae	<>
Ita29 committed 6 days ago			
Edited display information for the student dashboard to show more	Ita29 committed 6 days ago	b8aff54	<>
Just to be safe, don't think this actually did anything	AVmachine committed 6 days ago	18e04bc	<>
Merge branch 'master' of https://github.com/AVmachine/SoftwareProject	Ita29 committed 6 days ago	ae74011	<>
Push for changes	Ita29 committed 6 days ago	939e464	<>
Changed the GPA for a class data table	AVmachine committed 6 days ago	3a58bba	<>
Changed the display Exams Data Table using new table	AVmachine committed 6 days ago	5231658	<>
Made some tweaks to FillCombo	AVmachine committed 6 days ago	36b8a08	<>
Added to FillCombo function and gave comboBox a name for student dash...	AVmachine committed 6 days ago	c88f675	<>
Revert manual changes back before the merge	Ita29 committed 6 days ago	4a5d3c6	<>
Revert	Ita29 committed 6 days ago	e07ef9e	<>
Resolved Merge Conflicts	AVmachine committed 6 days ago	ab2da27	<>
Doing this because I have to	AVmachine committed 6 days ago	f4de801	<>
Added the log out button on the professor dashboard	Ita29 committed 6 days ago	d5d9cd3	<>

Below the list of commits for Nov 25, 2018, there is a section for "Commits on Nov 24, 2018", which is currently empty.

## 5.2 Other Communications

In addition to collaborating on Github, we made use of a group messaging app to communicate, in addition to telephone calls and meeting in person work on the project. Staying on top of communication really helped us to keep an overall flow of the project.

## 6 Software Product Management Plan

### 6.1 Software Product Management Plan

Deliverable	Due Date
Use Case Diagram	10/28/2018
UML Design	11/04/2018
Github Setup	11/11/2018
First Prototype	11/18/2018
Revision 1	11/25/2018
Final Product	11/30/2018