**Course Resource Library 1.0**

**Introduction**

Introduction

1.1 Overview

The course resource library is the 1.0 is the first release of the system. The web based library will allow instructors to organize resources, tools, and assignments for students.The goal of this project is to create a user friendly web based course resource library that will enable learners to easily access a curated list of resources, tools, and activities to further their learning journey. In order to use this application users will be asked to sign in or create an account. User accounts will be divided into two groups, learners and educators. Learners will be able to enter the course id of the course they are interested in and educators will be allowed to accept students into the course. Students will then be able to access the curated library to aid their studies. Educators will be able to organize information into categories and subcategories.

1.2 Goals

The goal of the 1.0 system system release is the implementation of the following key features:

1. Login and registration
2. Database implementation
3. Course page creation for educators
4. Ability to create categories and subcategories to organize information
5. Learner access to add a course
6. Educator access to accept learners
7. File management
8. Educators will be able to upload and download files
9. Students will allowed to download files from components and submit into files into assignment components

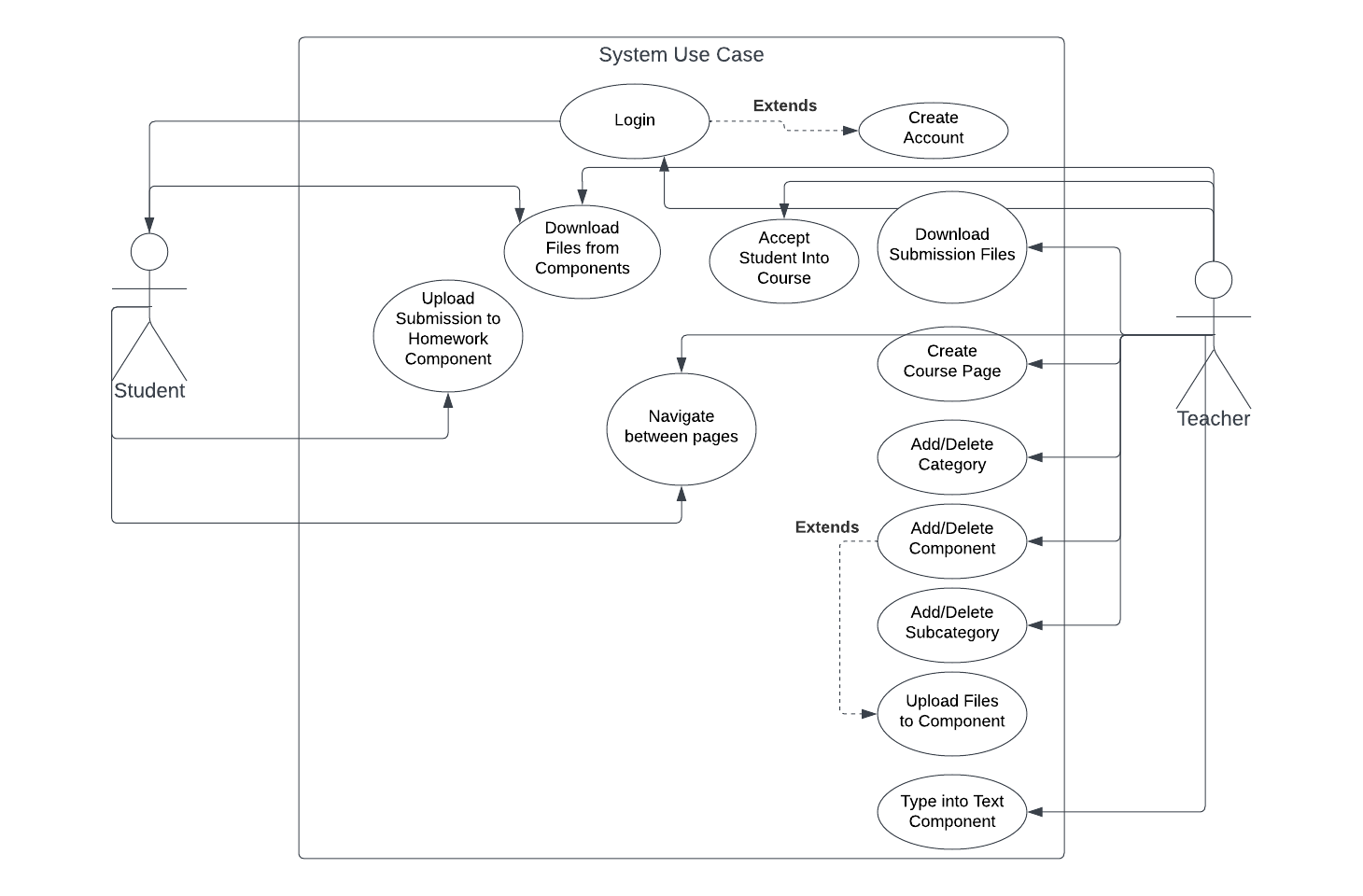
1.3 Constraints

The team has one semester to build a cross browser course resources library and implement key features. Due to the needs of different browsers the team will need to use an array of both browser specific and cross compatible libraries and frameworks. Additionally, the website capacity may be limited as the server will run from a local host.

**Glossary**

* Database: An organized collection of related data
* Text Field: An area where people can enter or edit pieces of text
* File Type: A name given to a specific kind of file
* UI/User Interface: The point of human-computer interaction and communication
* File: A container in a computer system for storing information
* Control: Interactive element of the UI

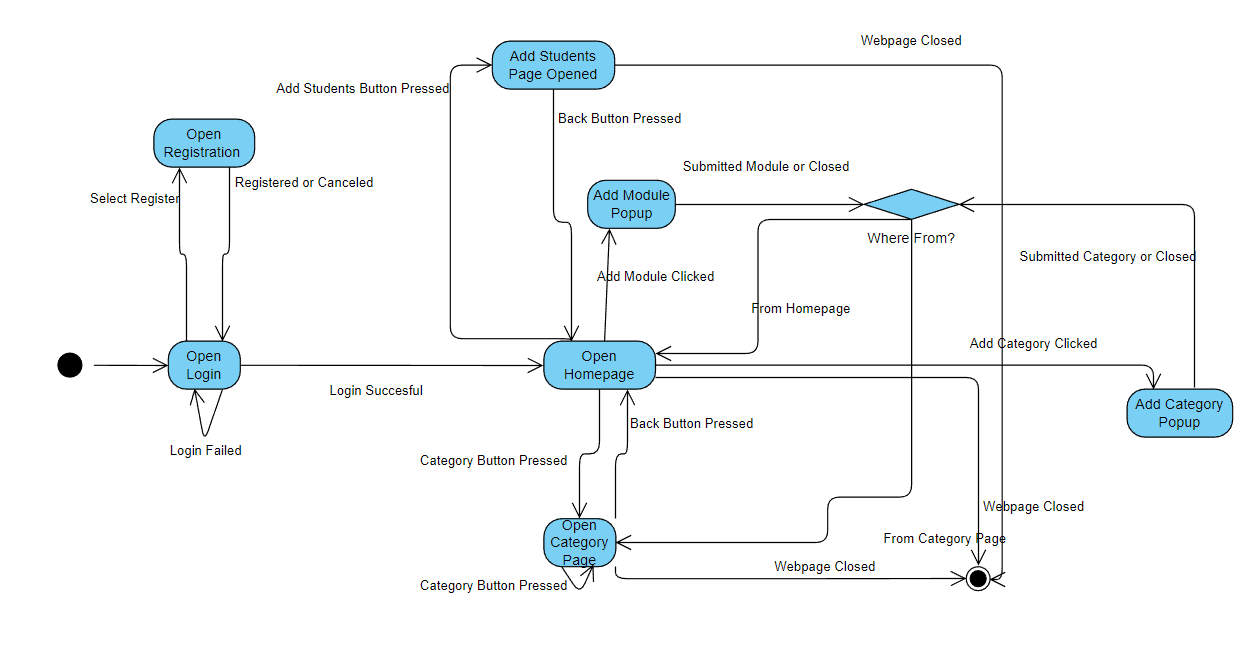
**User Requirements**

****

**List of Functional Requirements**

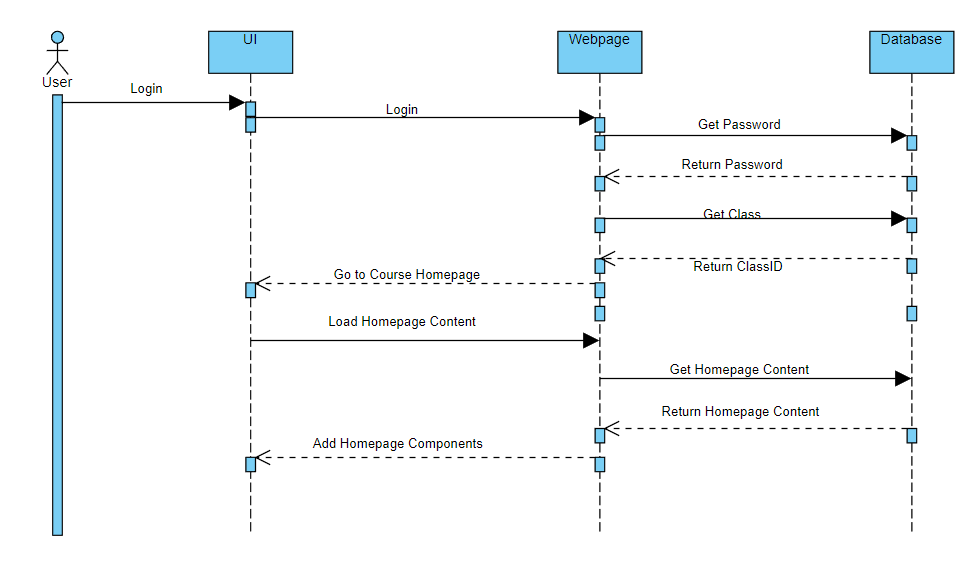
1. User can login to a user created account with a username and password or create an account if they don’t have one.
2. Teachers can add or delete a new category and add or delete students to a category as well.
3. Teachers have the ability to add or delete subcategories as well as add or delete components to any category or subcategory.
4. Users are able to upload and download files from components.
5. Teachers can type into text components.
6. User is able to navigate through pages.

**IV. Design and Implementation Modules**

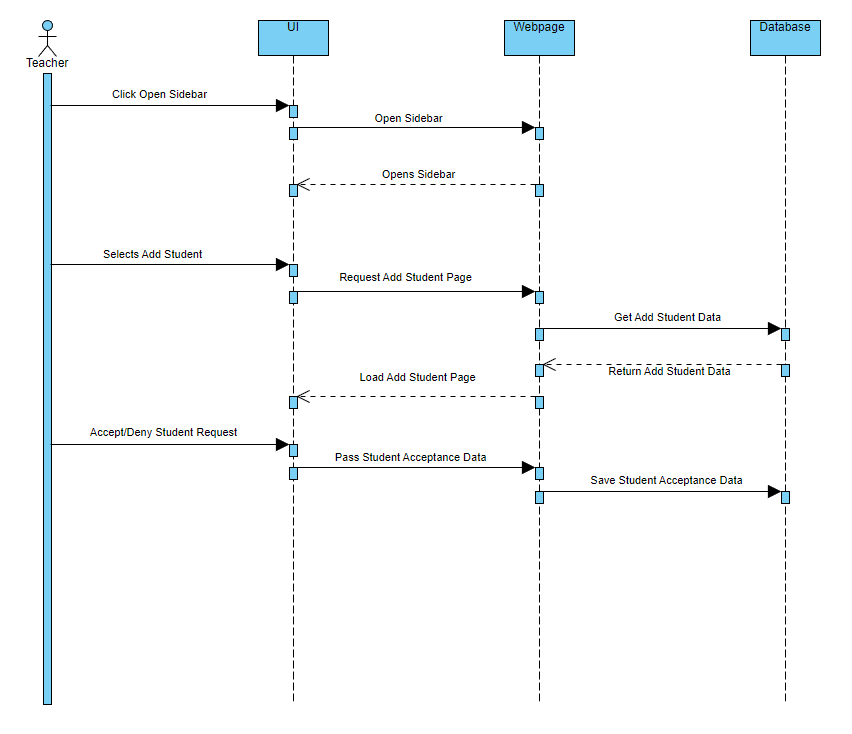


1. Pictured above is the teacher state diagram. It demonstrates the user’s actions flow when using the product.

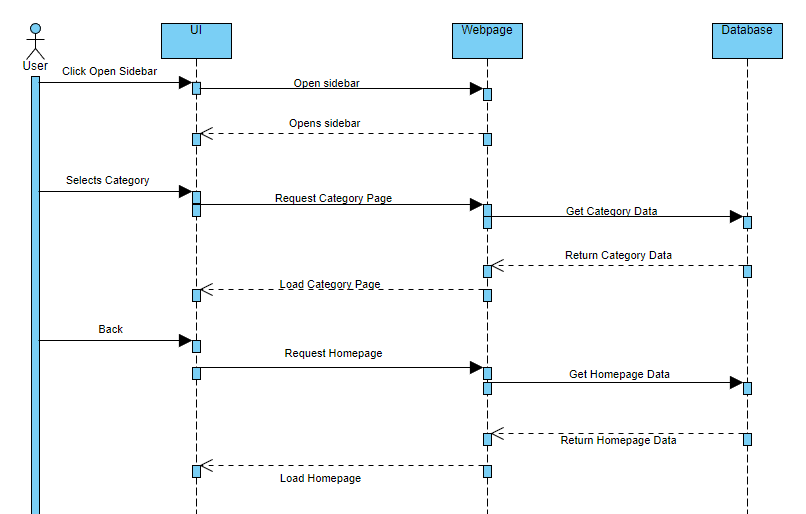
\*Note: Sequence models were extremely useful for visualizing js load sequence when retrieving information for dynamic applications.



1. Pictured above is the homepage load sequence. It demonstrates a user logging in and being redirected to the homepage.

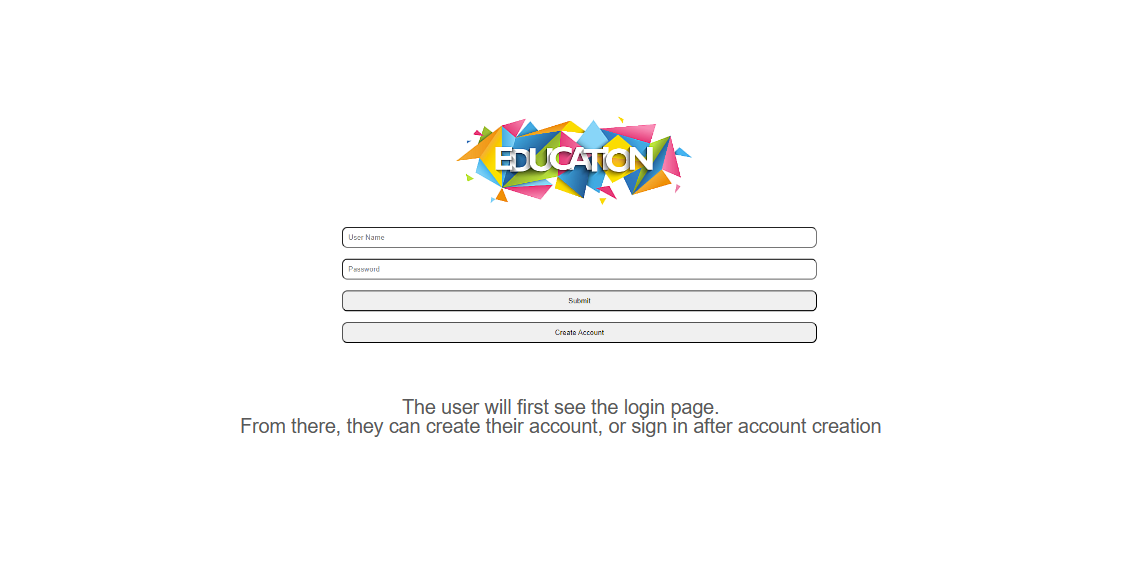
****

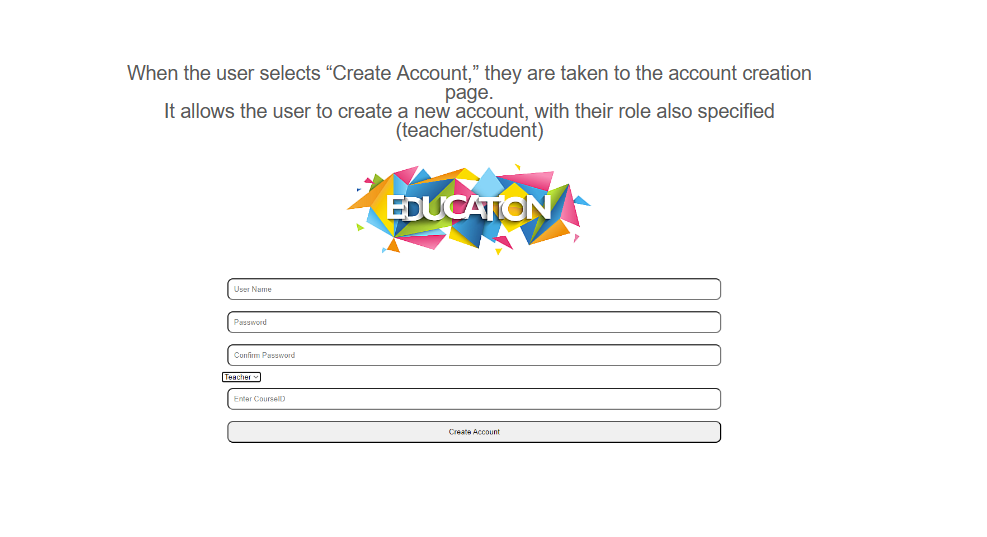
1. Pictured above is the add student sequence model. It demonstrates a teacher accessing the student acceptance page from the homepage and then accepting/denying student requests.

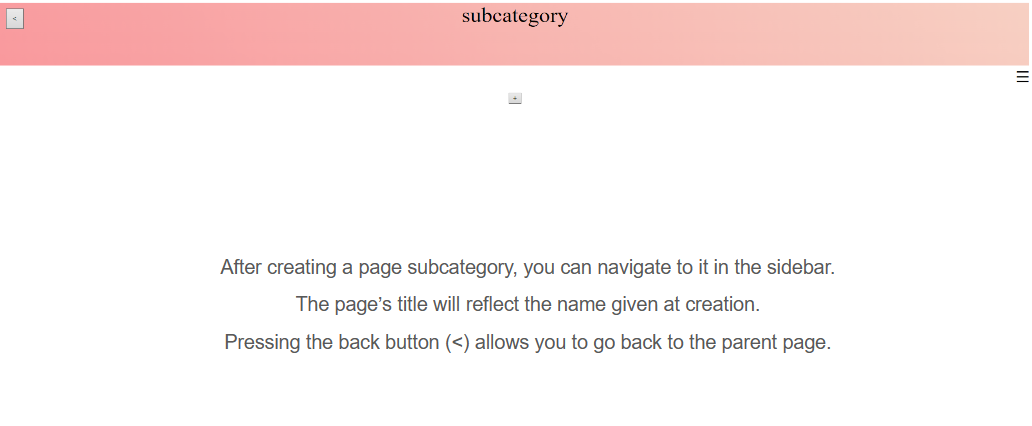
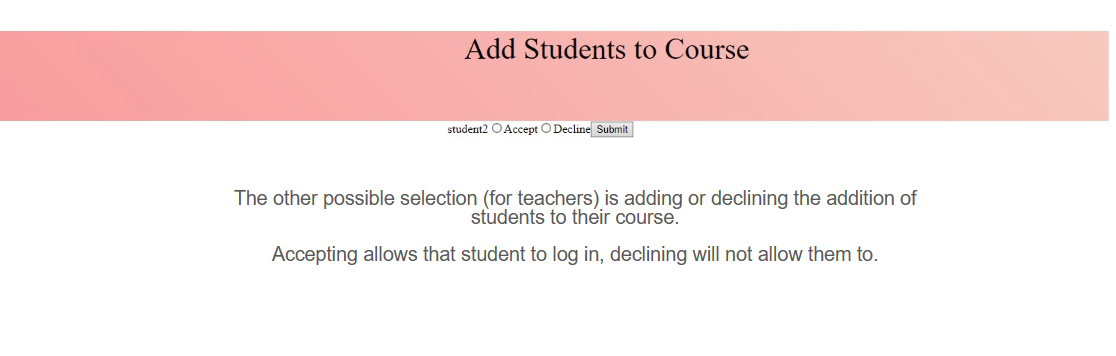
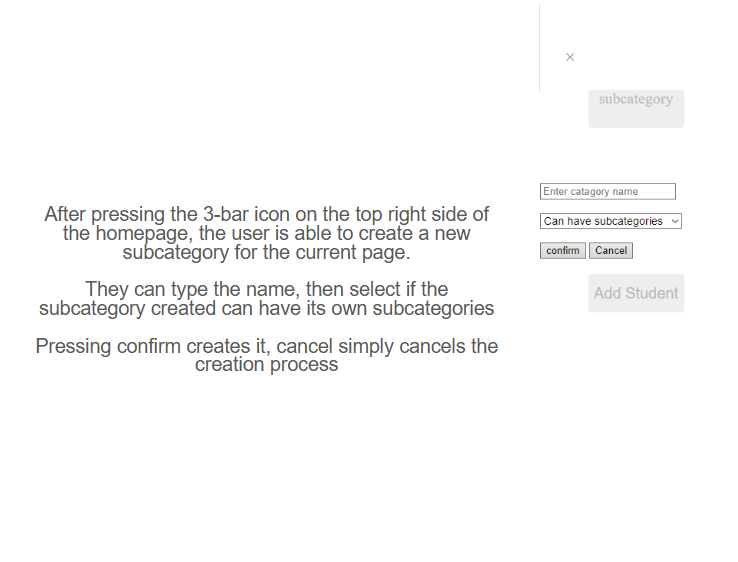
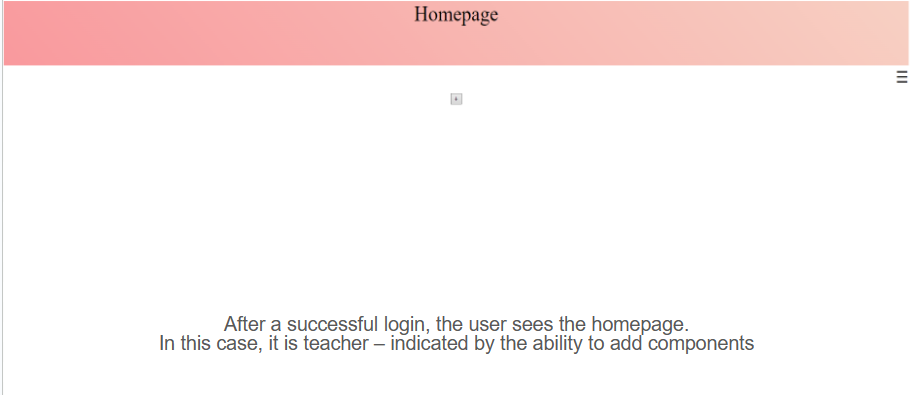
****

1. Pictured above is the category load sequence model. It demonstrates a user’s request to display a category page. It also demonstrates the request sent when a user uses the back button that returns to the parent page of the current category.

\*Note: The use of the parent page instead of the homepage is deliberate. Each category has its own page ID. When a category has a subcategory and the user is on that page it returns the user to the parent category page. While on a parent category page, clicking the back button will then return the user to the homepage.

**Demo Screenshots**





Test Cases:

<https://docs.google.com/spreadsheets/d/16rlzUtXLu9PuTeYVcJ34DobT5NNxLqMgvaPGaveGqQ4/edit?usp=sharing>

Tools:

1. Github- code repository used to facilitate working on code together and maintain version control.
2. VS Code- Used as an IDE to edit code.
3. MySQL- Used to create and house the database.
4. Languages
   1. HTML
   2. CSS
   3. JavaScript
   4. SQL
   5. PHP

**Project Contribution**

| Course Resource Library 1.0 | | | | |
| --- | --- | --- | --- | --- |
| Dates |  |  |  |  |
| Alec | Project Team | Srs Doc/Pres | Design/Sprint Pres | Final Project/Pres |
| Group Project Selection  Project Report Documentation | Prototype Design  Presentation  Client Meeting | Database: Creation  Database: Add Modules  Database: Add categories  Database: Add Course | Took Over: UI: Login, Register, and Add Student  Conversion to PHP  Application Demo  Presentation  Final Documentation |
| Alex | Group Project Selection  Project Report Documentation | User Requirements  Presentation  Client Meeting | Database: Login/ Registration  Database: Add Student  Database: Add Join Requests  Database: Add Student to Course  Presentation | Conversion to PHP  Application Demo  Presentation  Final Documentation |
| Megan | Group Project Selection  Project Report Documentation | Non-Functional Requirements  Presentation  Client Meeting | UI: Page Components  UI: File Selection  Presentation | Conversion to PHP  Application Demo  Presentation  Final Documentation |
| Halie | Group Project Selection  Project Report Documentation | Introduction  Presentation  Client Meeting | UI: Page Creation  UI: Page Navigation  Presentation | Conversion to PHP  Application Demo  Presentation  Final Documentation |
| Prasanna |  | Client Meeting | UI: Login/  Registration  Presentation | Application Demo  Presentation  Final Documentation |