**Introduction & Background**

In an era driven by rapid technological advancements and artificial intelligence, software solutions that uphold human values and foster individual growth are more essential than ever. Taylor's University is dedicated to this mission by embedding holistic education into its core, preparing students to lead and make meaningful contributions on a global scale. This assignment aligns with Taylor's commitment, challenging students to create a well-defined educational software solution that enhances learning experiences, promotes self-development, and prepares students for impactful roles in society.

*Objective*

The primary goal of this assignment is to assess and build students' proficiency in Agile project management and requirements analysis within the context of educational software design. Students will leverage Agile methodologies to create a comprehensive Software Requirement Specification (SRS), demonstrating:

- Expertise in iterative development, task planning, and requirements adaptation.
- Skill in translating high-level project concepts into structured, actionable requirements.
- Ability to craft a detailed SRS that aligns with holistic educational principles, providing a strong foundation for subsequent project stages.

## Table of Contents


**1.0 Project Selection and Agile Planning** **7**
1.1 Project Selection 7
1.2 Agile Planning 8
1.2.1 Product Backlog 8
1.2.2 Sprint Goals 10
1.2.3 Prioritized Tasks 10
1.2.4 Main Features Justification 11

**2.0 Ideation and Executive Summary** **14**
2.1 Executive Summary 14
2.1.1 Project Concept 14
2.1.2 Project Overview 14
2.1.3 Intended User Base 14
2.1.4 Educational Goals & Impacts 14
2.2 Visual Aids for Enhanced Clarity 16
2.3 Interface Sketches 17
2.3.1 Student 17
2.3.2 Student Leader 18
2.3.3 Lecturer 19
2.3.4 System Staff 20

**3.0 Prototyping and Iterative Development** **21**
3.1 UI/UX Design 22
3.1.1 General 22
3.1.2 Student 24
3.1.3 Lecturer 40
3.1.4 System Admin 48
3.2 Documentation of Iterative Development 52
Overview of Iterative Process 52
Iterative Documentation 52

**SRS Document** **54**

**Appendix** **70**
Appendix A: Activity Diagram 70
Appendix B: Use Case Diagram 71
Appendix C: Architecture Diagram 72
Appendix D: Active In-Development Prototype 72

## 1.0 Project Selection and Agile Planning

### 1.1 Project Selection

Taylor's University project-oriented approach and its implementation within the courses and modules that are offered poses a beneficial, yet also consequential concept. With the implementation of a project-intensive approach within an academic institution, it would come to a realization that within its core, group projects would be an inevitable core academic activity directly imposed on both students and lecturers.

Through time, ranging from difficulties in forming a proper group, difference in work styles, to the lack of communication and social compatibility, difficulties that are encountered within a group project could start to be identified. This, accompanied by a hyper-diversity intercultural environment that Taylor's University inhibits, makes the difficulties encountered within a group project, now not only an academic situation, but also relates to the student's social life and behavior. And with the aforesaid statement, it is important for us to recognize that a student's social condition relates tightly with their mental health and academic performance.

In regard to this problem, our project, **MyGroups*, is designed to tackle and minimize the drawbacks of a project-oriented approach by facilitating students with proper resources and features they would be able to utilize in order to complete their group projects comfortably and seamlessly.** It will act as a centralized platform for any group projects related matter. Starting from group pairings, project consultations, peer-to-peer rating system, to benefits and rewards based on a student's performance, MyGroups* is designed to facilitate any group projects–promoting ease of use, accessibility, engagement, and comfort.

### 1.2 Agile Planning

The development of *MyGroups** will adopt the Agile Software Development Life Cycle (SDLC), a flexible and iterative approach that emphasizes collaboration and adaptive planning. This method is ideal for our student-focused platform, allowing us to respond effectively to real-life challenges in managing group projects.

Our Agile plan includes a structured product backlog, story-point estimation, sprint goals, prioritized tasks, and outlined sprint cycles where each is carefully designed to enhance clarity and development efficiency.

### *1.2.1 Product Backlog*

The table below shows the product backlog for *MyGroups*, listing key features based on student needs to support better group project management and holistic development.

| ID | Feature | Feature Description | Estimated Story Point |
|---|---|---|---|
| 1 | User Registration | Allows students and lecturers securely sign up, log in, and access the platform. | 2 |
| 2 | Dashboard | Displays an overview of all core features, so users can easily navigate and access everything from a single page. | 3 |
| 3 | Courses Module | Displays all the subjects the student is enrolled in, with details about group assignments, latest activity, and any alerts. | 3 |
| 4 | Projects Module | Lists all ongoing and upcoming assignments, deadlines, consultation status, and feedback alerts. | 4 |
| 5 | Project Tracking Module | Allows students to track project deadlines, submission statuses, and see the latest updates from their teams. | 4 |
| 6 | Consultation Module | Enables students to request for consultations with lecturers. A text box would appear in the lecturer's view for them to respond to the students. | 4 |
| 7 | Peer Feedback Module | Lets students give and receive peer reviews after completing each project. | 3 |

| | | | |
|---|---|---|---|
| 8 | GroupedIn™ | A smart search tool to find, view, join, or recruit members for assignment groups in each subject for the semester. | 5 |
| 9 | Profile | Displays student details and their performance. Also lets students manage visibility settings of their info. | 3 |
| 10 | Announcements Page | Displays all system announcements, planned maintenance, and important updates in one place. | 2 |
| 11 | Notifications & Alerts | Sends reminders for deadlines, unread messages, and urgent updates related to group work and platform activity. | 2 |
| 12 | Guide Page | Display users with information on how to use the system, suitable for new users. | 2 |
| 13 | Settings | Allow students and lecturers to manage their account. | 2 |

*Table 1.2.1 Product Backlog*

### *1.2.2 Sprint Goals*

The table below outlines the sprint goals for *MyGroups*, with each sprint focused on building and improving features that support student collaboration and leadership.

| ID | Goal | Duration |
|---|---|---|
| Sprint 1 | Develop core access and navigation features, including User Registration, Dashboard, and Profile. | 1 week |
| Sprint 2 | Build content and overview modules like Courses, Projects, and Announcements Page. | 2 weeks |

| | | |
|---|---|---|
| Sprint 3 | Implement interaction tools such as Project Tracking, Consultation, and Peer Feedback modules. | 2 weeks |
| Sprint 4 | Integrate GroupedIn™, Notifications & Alerts, Guide Page, and Settings, and finalize testing. | 2 weeks |

*Table 1.2.2 Sprint Goals*

### *1.2.3 Prioritized Tasks*

The table below represents the sprint task in more detail along with their priority.

| ID | Tasks | Priority |
|---|---|---|
| Sprint 1 | Develop core access modules to enable students and lecturers to register, log in, and view their profiles.<br>**Features:** User Registration, Profile | High |
| | Develop navigation features to help users access main sections easily.<br>**Features:** Dashboard, Announcements Page | Medium |
| Sprint 2 | Build the Projects module that lists assignments, deadlines, and submission statuses for each course.<br>**Features:** Projects Module | High |
| | Develop the Courses module to display enrolled subjects and related alerts.<br>**Features:** Courses Module | Medium |
| Sprint 3 | Implement project management tools to help track progress and encourage collaboration.<br>**Features:** Project Tracking, Consultation, Peer Feedback modules | High |
| | Check and implement lecturer views to allow editing, grading, and progress monitoring. | Medium |

| | | |
|---|---|---|
| Sprint 4 | Integrate the GroupedIn™ smart search tool for group formation and member recruitment.<br>**Features:** GroupedIn™ | High |
| | Create the notifications and alerts module, guide page, and settings module to improve user experience.<br>**Features:** Notifications & Alerts, Guide Page, Settings | Low |
| | Finalise, test, and improve all developed features to ensure the system works as expected and meets user needs. | High |

*Table 1.2.3 Prioritised Task*

### *1.2.4 Main Features Justification*

The core features of our system were justified as follows:

1. **Project Tracking Module**

   This feature helps students keep track of their project progress with a simple to-do list. Each group member can view tasks, add new ones, and mark them as completed. The group leader can manage how the workload is divided among the team. Lecturers can also monitor each group's progress and reach out if something doesn't look right.

2. **Consultation Module**

   The Consultation Module lets students easily request help from lecturers and get responses right on the platform. It's a simple way to clear up questions and feel supported throughout the project. It is also flexible to the lecturers as they can check the project when they have free time on just one platform.

3. **Peer Feedback Module**

   This module lets students share feedback on their teammates after finishing a project. If there are any problems, like free riders or conflicts, students can report it directly here. It helps promote accountability and makes sure everyone contributes fairly.

4. **GroupedIn™**

   GroupedIn™ is a smart tool for finding, joining, or creating groups. It solves the hassle of trying to form teams by showing who's available and what subjects they're working on. The request to join the group feature can help the group leader to filter members based on desired workstyles. This can resolve conflicts and smoothen the journey of the project.

5. **Profile Module**

   The Profile displays key details like contribution points, grades, and project history. It builds transparency and helps students show their strengths to potential group members. This will motivate students to be more productive and engaging in projects for them to be able to receive a nice profile view. Students are flexible to hide certain information for display.

## 2.0 Ideation and Executive Summary

### 2.1 Executive Summary

#### *2.1.1 Project Concept*

At its core, MyGroups* is designed to be a MyTimes (Taylor's University official platform for educational resources and lecturer-students communication) counterpart but with group projects and/or group assignment management as its point of interest. With the accomplishment of the project, it is well hoped that students will be able to work on their group projects easier, more accessible, and with more enjoyment. We believe one of the key aspects toward a strong academic performance is the student's enjoyment/comfort as well.

#### *2.1.2 Project Overview*

With current existing one such as MyTimes and CampusCentral, MyGroups* will be the third addition to the set of platforms that Taylor's University will be offering to their students as part of a digital-based learning and experience. Along with group formations, progress tracking, automated priority/queue-based consultation system, and customizable personal profile and statistics among others, MyGroups* will incorporate a set of features that is specifically tailored for the cause of "Ease of use, Accessibility, Engagement, and Comfort".

Other supplementary features that would support more of Taylor's Holistic Education Objective will also be implemented within the platform, such as the Leadership and Contribution Point feature. We aim not only for group assignments to be finished quickly and easily, but to also encourage students to take proactive initiatives within their group project environment.

#### *2.1.3 Intended User Base*

Our main user base would be Taylor's University Students and Lecturers as this would be designated as the official platform/portal for group assignments.

#### *2.1.4 Educational Goals & Impacts*

In terms of the educational impacts and goals the project is trying to achieve, we believe in some baselines that would be able to improve and accelerate the quality of an education. This includes developing, among others, the Individual Characteristics (Leadership Quality, Social

Skills, etc.) of the student and their Learning Resources the institution has to offer. Hence, some of the educational goals and impacts we hope this project would achieve, are:

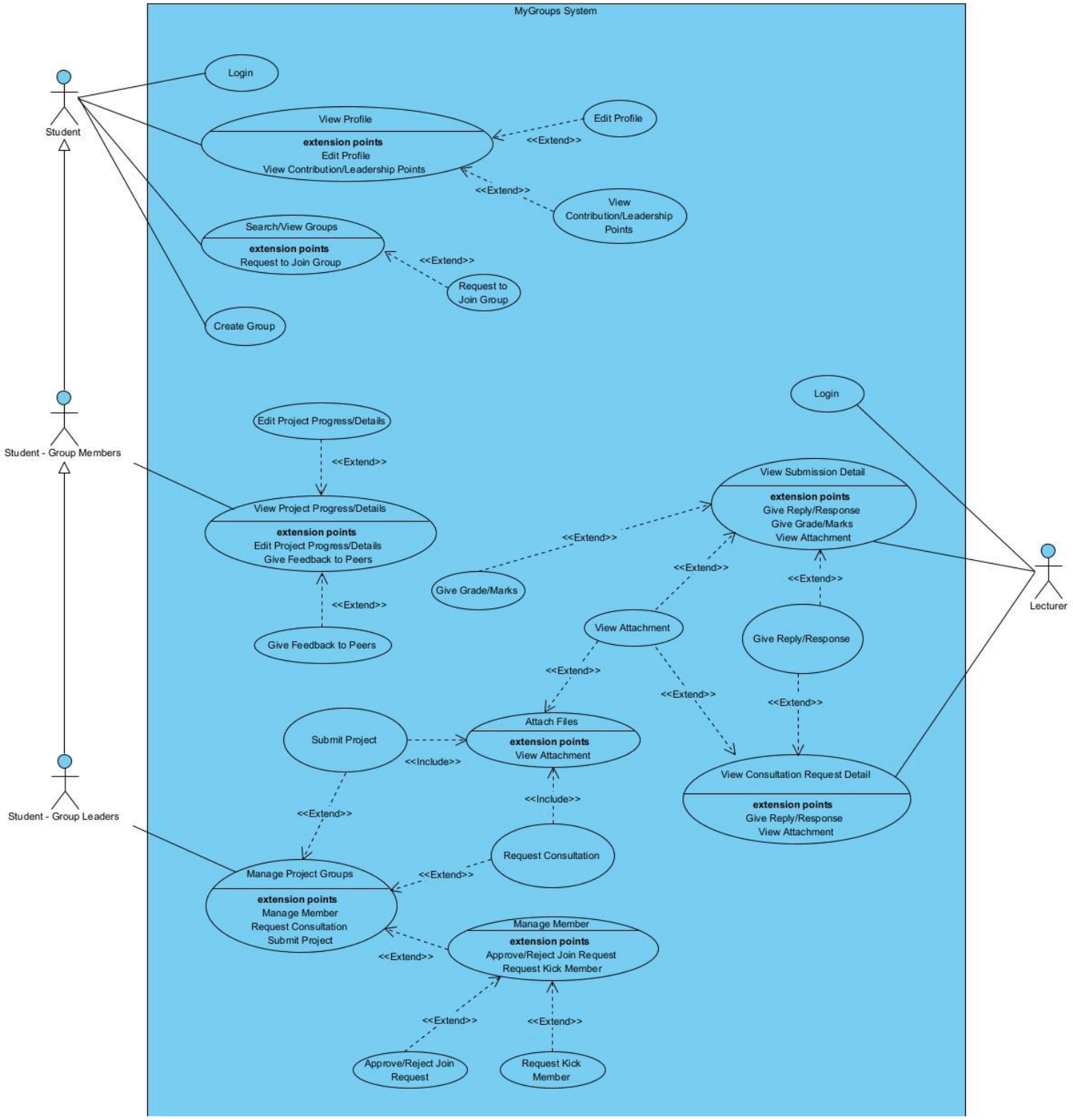
1. Increasing educational accessibility and ease of use through the introduction of a designated platform/portal specifically designed for the matter of group projects. This would also be amplified by an emphasis on student experiences and the problems they often encounter.

2. Accelerating the character development of students through pushing students on taking initiatives and/or proactive parts on group projects. In turn they may gain rewards or benefits based on their activity.

3. Creates opportunities for students to experience hands-on project management techniques through various complementary tools and features, which would help them in the long run of having experience of using some project management methods/techniques.

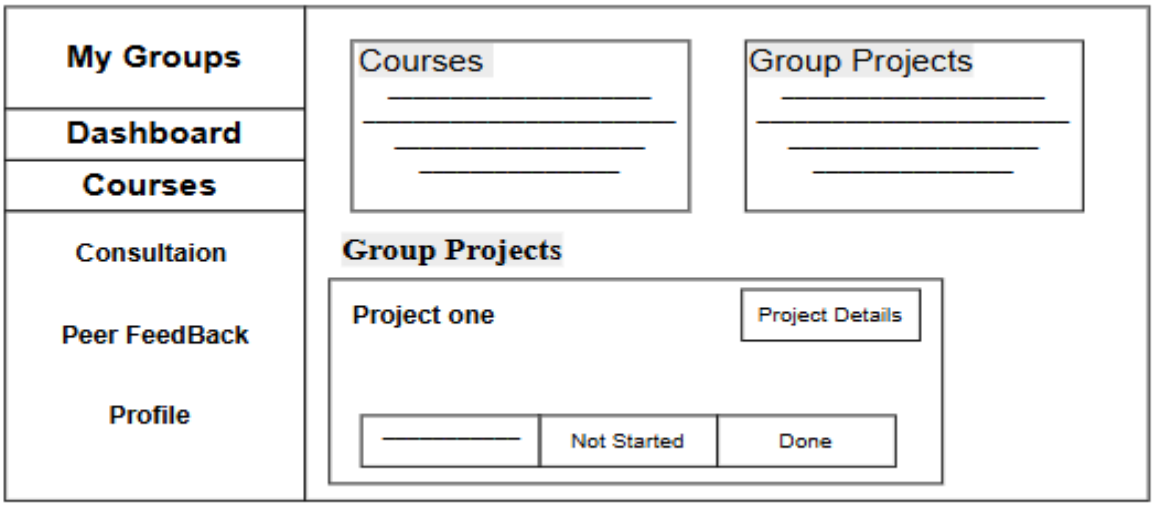
## 2.2 Visual Aids for Enhanced Clarity

### *Use Case Diagram (Details on Section 4 - SRS Documentation)*



## 2.3 Interface Sketches

### *2.3.1 Student*




**Summary**

This is a wire frame of the Student Dashboard of the MyGroups application. It will allow university students to plan their group work, monitor their progress and conveniently orientate in the various academic aids that they will need.

**Main Characteristics:**

- Sidebar Menu:
- Allows user access to primary parts without issues dashboard, courses, consultation, peer feedback, and profile with ease.
- Courses Section:
- Displays subjects and group work they are enrolled in.
- Explication of Group Projects:
- Displays active group projects in one page.

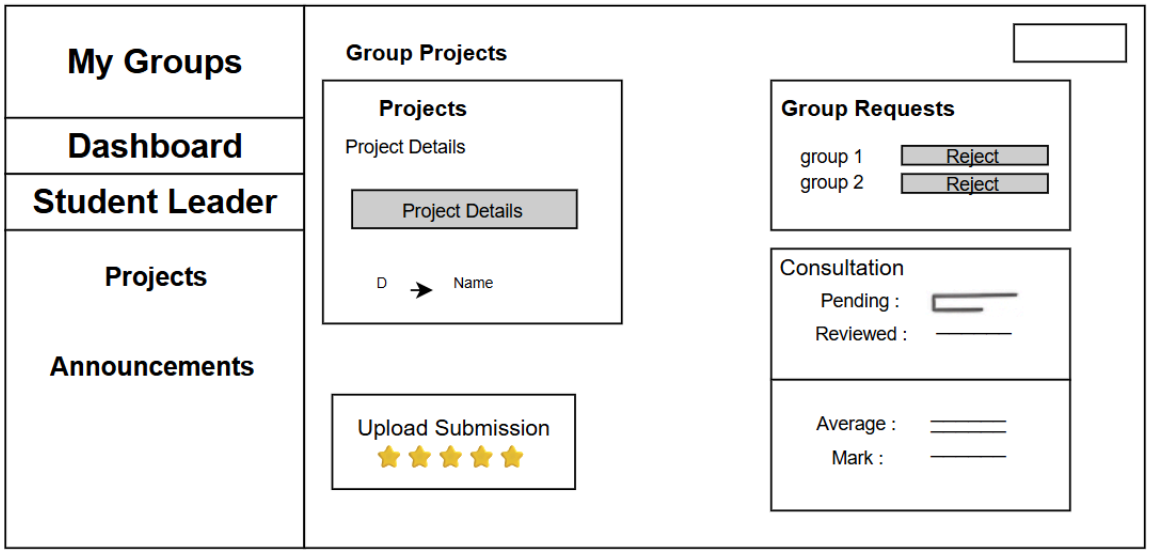
**Project Details Panel:**

Allows students to see details of a particular project, see the progress of a task (e.g., Not Started, Done) and utilize submission or feedback options.

**Primary objective:**

As a means of organizing all the group project activities in a single location so that students are less likely to be scattered, are more likely to collaborate well and finish their academic projects at the right time.

### *2.3.2 Student Leader*




**Summary**

The wireframe below is the display of Student Leader Dashboard in MyGroups platform, which is aimed at structuring and facilitating the work of the group leader in managing team activities and deliverables.

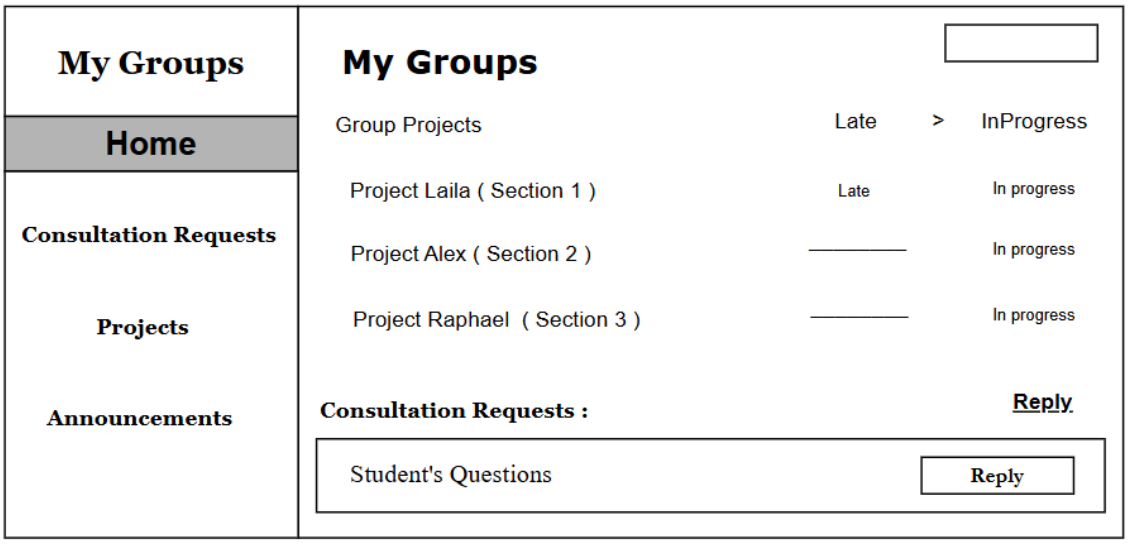
**Most Important Features:**

- Navigation Menu: go to Dashboard, go to Projects, go to Announcements.
- Group Projects Panel: Check the project and group progress.
- Upload entry: Upload entries with a star-based feedback structure.
- Group Requests: Deny or approve students who wanted to join.
- Consultation Panel: Monitor the consultations awaiting and consultations done by the lecturers.
- Performance Metrics: Check project marks as well as average feedback.

**Purpose:**

- So that the group coordination, submissions, and communication can be under the full control of student leaders--to ensure that the process of group work becomes more efficient and transparent.

### *2.3.3 Lecturer*





**Summary**

The Lecturer Dashboard depicted by this wireframe is part of the MyGroups platform, the main purpose of which is to monitor the readiness of group projects and address student consultations.


**Major Characteristics:**

Navigation Menu: Home, Consultation Requests, Projects and Announcements links.

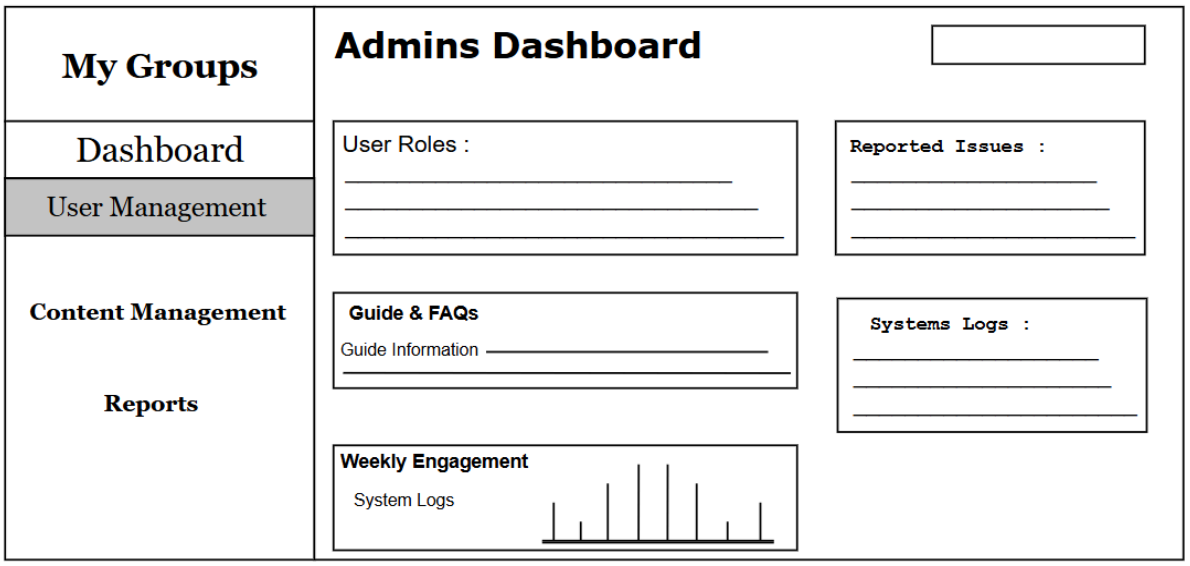
Group Projects Panel: Lists the names of student groups by project and section, their status (Late, In Progress)' their Scholarship status.

Consultation Requests: Allows the lecturers to answer the questions of students directly on the platform.

**Purpose**:

To assist lecturers to easily follow group progress, detect problems (such as late submission) and answer consultation requests- all in the same place.notes or reminders which can be seen by all students in courses taken.

### *2.3.4 System Staff*




**System Staff Interface summary :**

The following wireframe represents the Admin Dashboard in the MyGroups** system where the staff of the system can manage the users, track activity, and update the system.

**Features:**

Navigation Menu: Dashboard, User, Content, Report access.

User Roles Panel: edit/ assign role (student, lecturer, admin.):

Reported Issues: See and monitor the technical or user reported problems.

System Logs: Track the use of system and the event occurring in the system.

Guide & FAQs: revise or fix platform documentation.

Weekly Engagement: The use of the graph is to see the overall user activity throughout the system.

**Purpose:**

In order to provide managing administrators with complete control over the operations of the platform and support content, user activity observation, stability, and usability of the system.

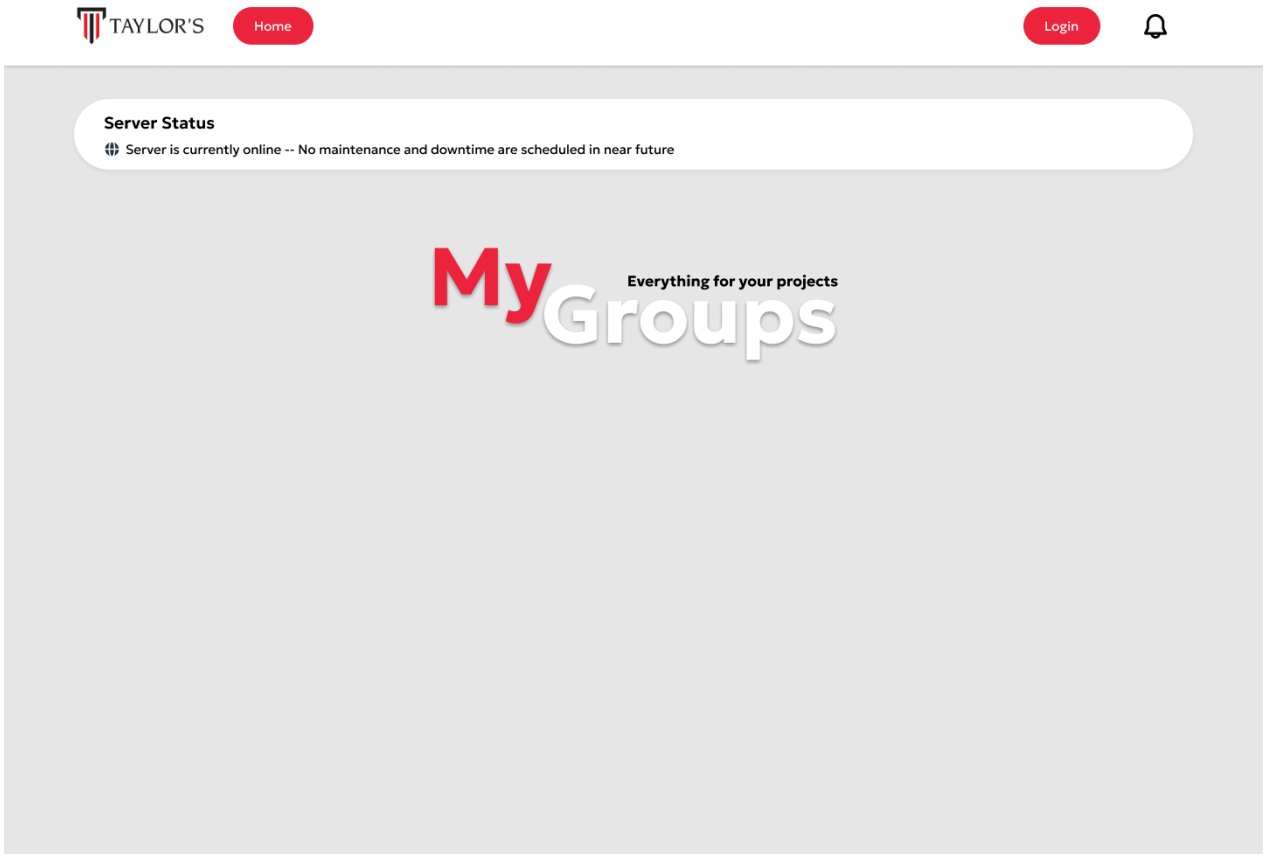
## 3.0 Prototyping and Iterative Development

The following section constitutes the UI/UX design created for the proposed project. It will showcase major features and core flows that are designed to fulfill requirements as well as objectives for the project

Due to the evolving nature of the project, this section of the document is not final, and **may not consist** of every functionalities depicted in other parts of this document. This section is dedicated only to illustrate major user stories and flows which consists of interaction with major features/systems.

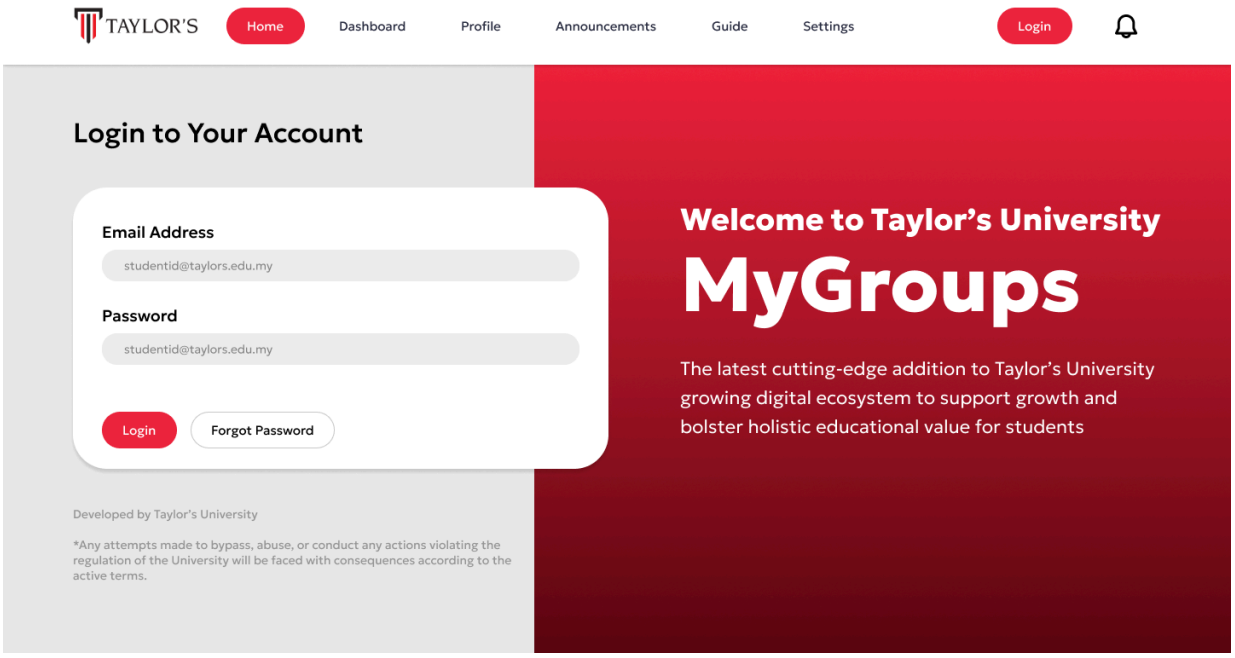
## 3.1 UI/UX Design

### *3.1.1 General*




General Main Page

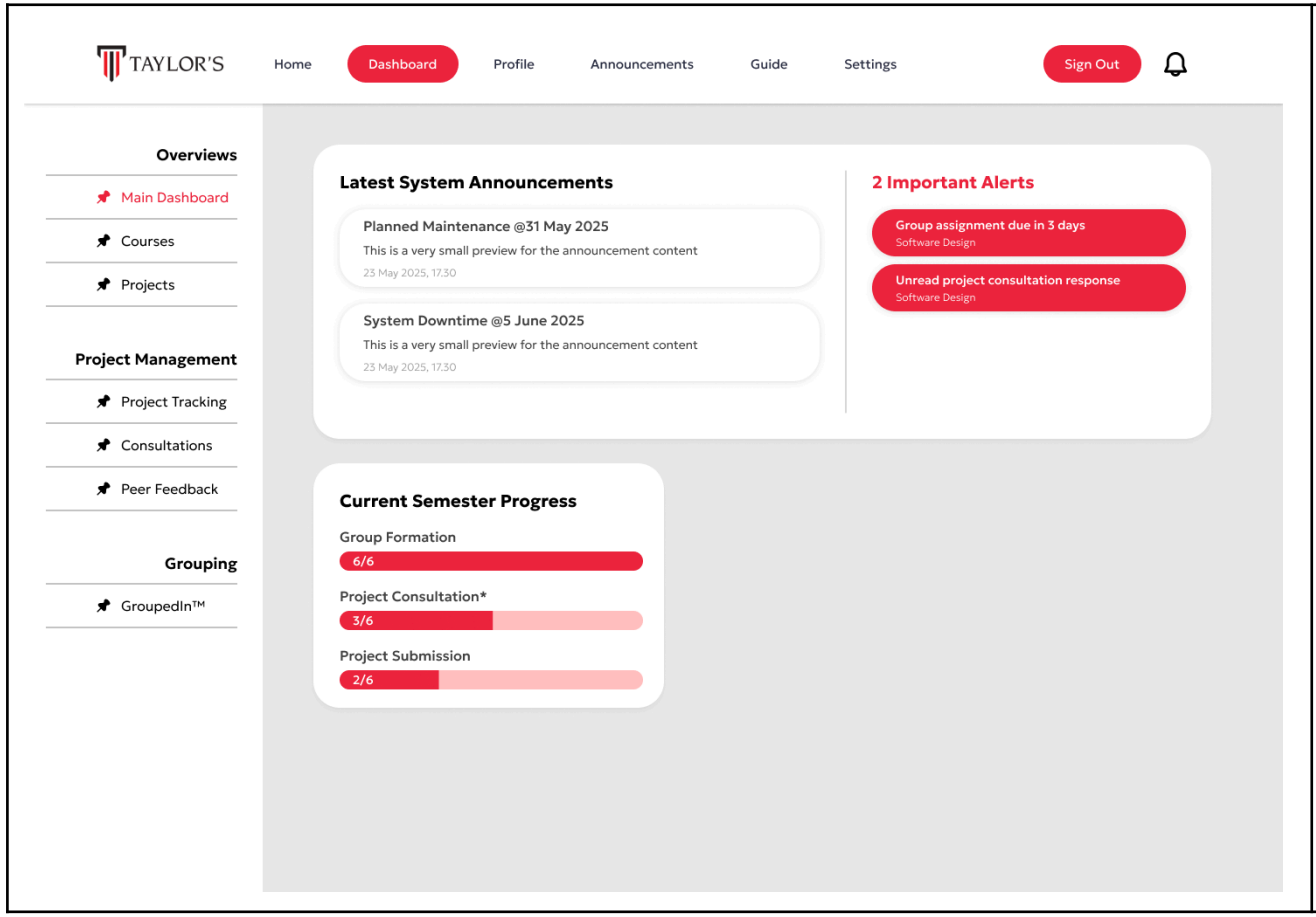
The General Main Page consists of a landing page showing the server status, and a consistent Navbar which will be shown on each page, showing features/page selections depending on the roles




General Login Page

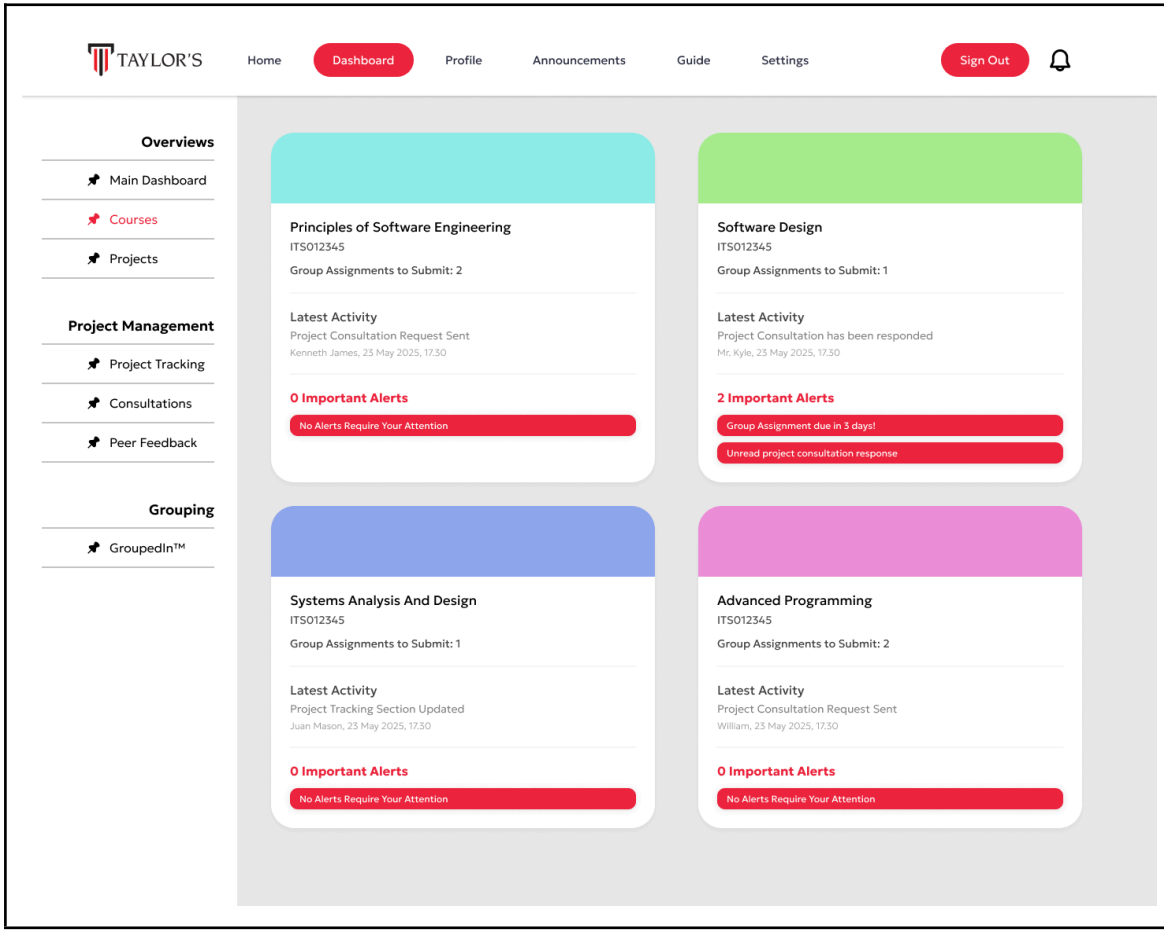
The General Login Page will be the main page where all users will do their login activity with their credentials.

### *3.1.2 Student*



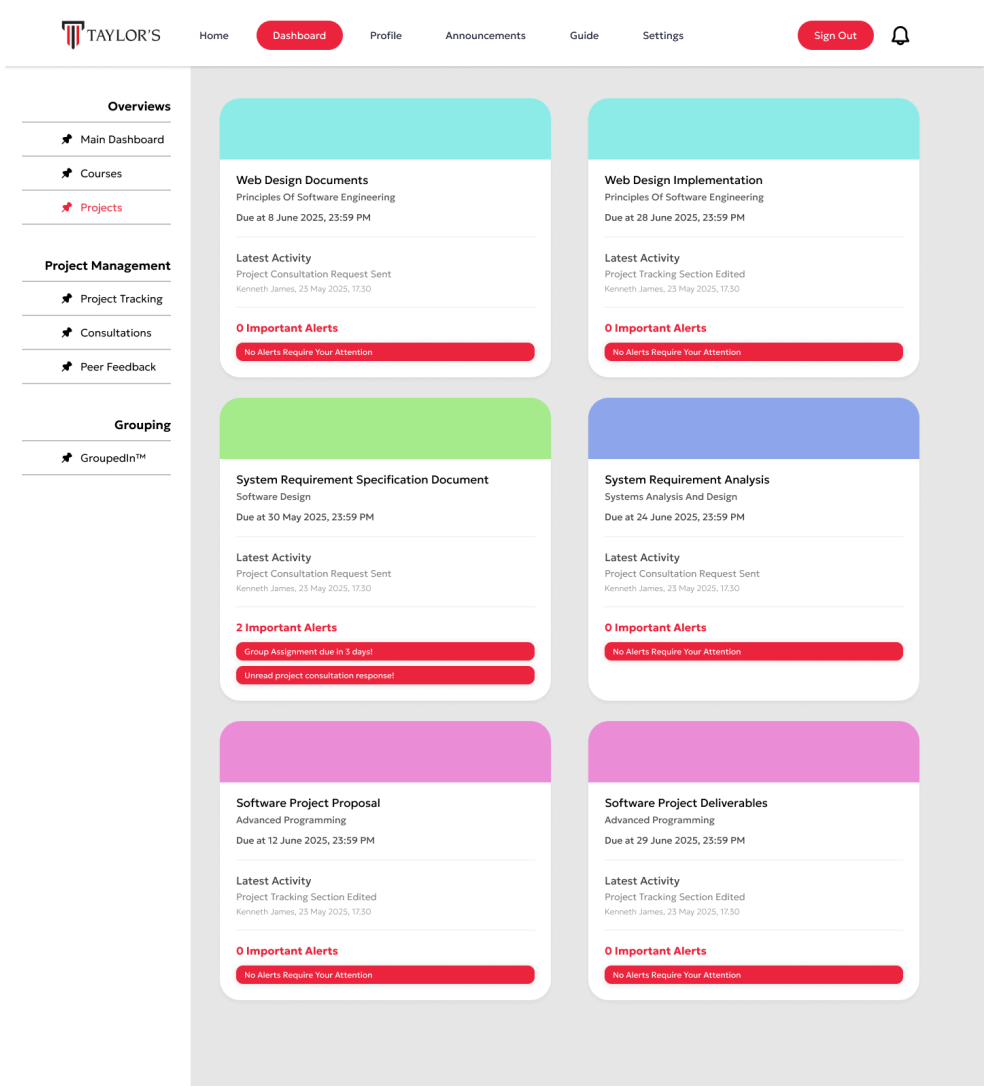

Student Main Dashboard

The Student Main Dashboard interface is designed to show students the latest System Announcements as well as their Group Assignment Progress visually.



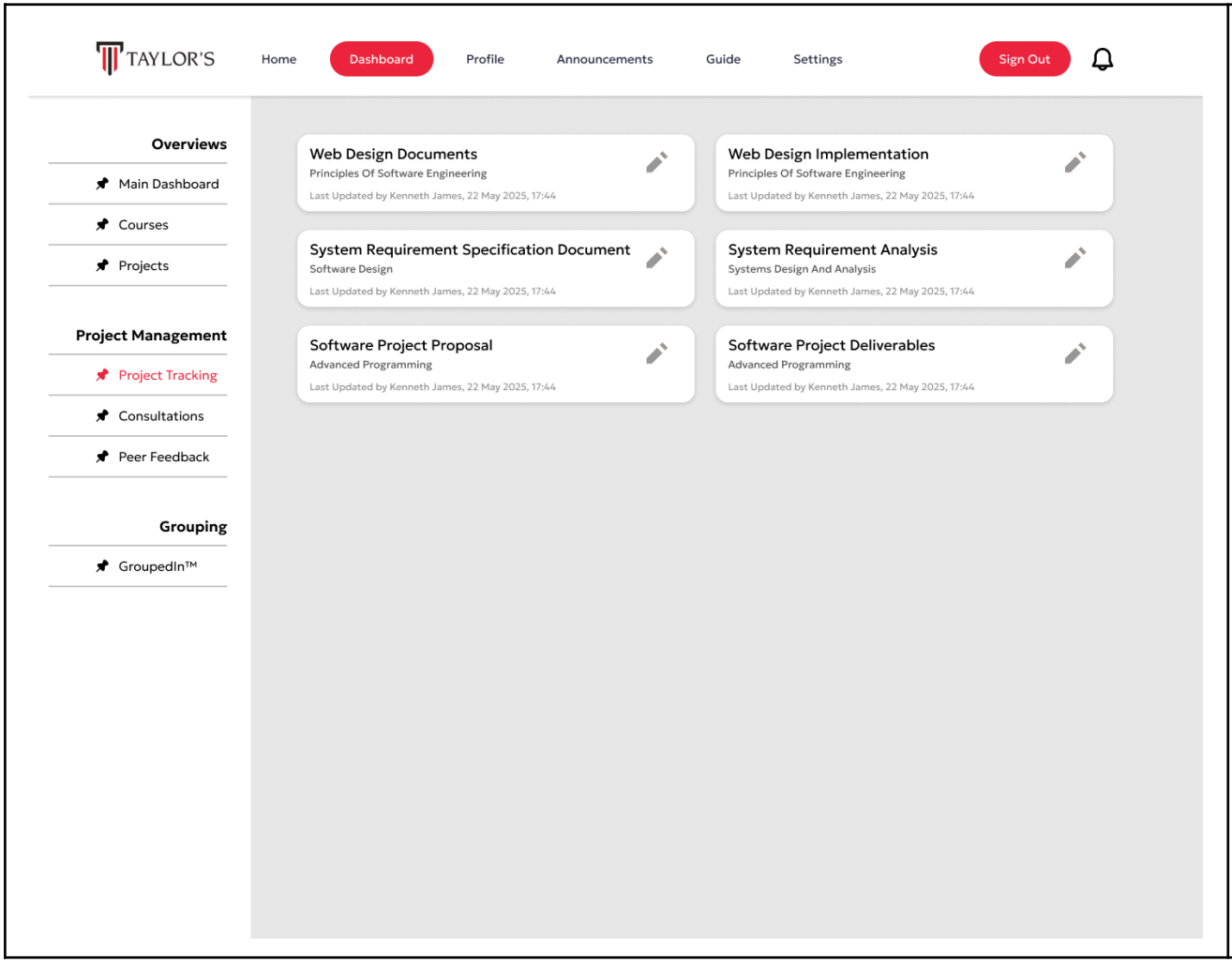

Student Course Dashboard

The Student Course Dashboard interface is designed to show students the overview of their Courses




Student Project Dashboard

The Student Project Dashboard interface is designed to show students the overview of their Projects

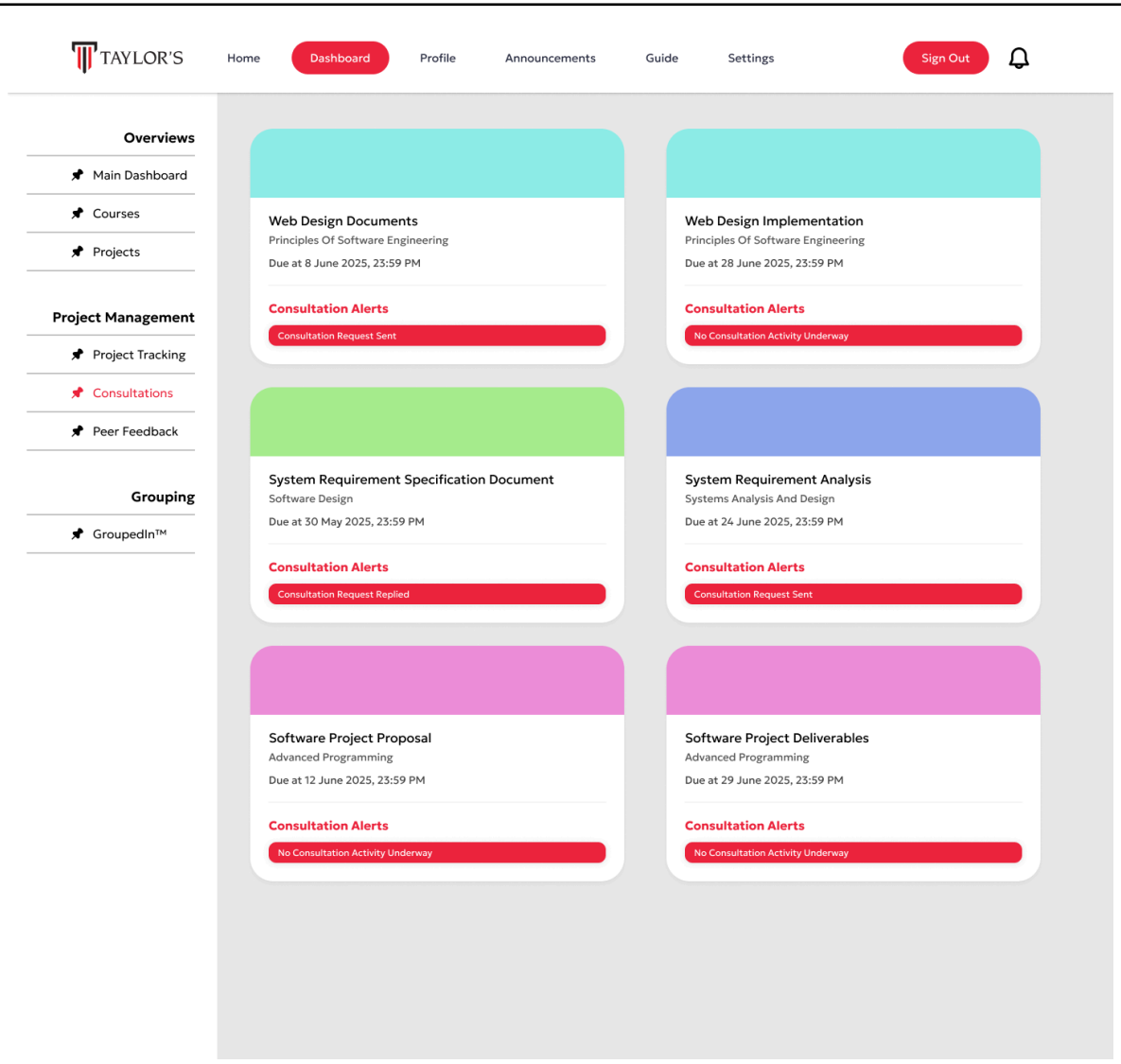



## Student Project Tracking

The Student Project Tracking interface is designed to show students the details of their projects, along with accessing various project management tools that are provided by the system

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on one of the cards, it will open a drop-down consisting of to-do lists, and editable fields to assign group members to a respective task*

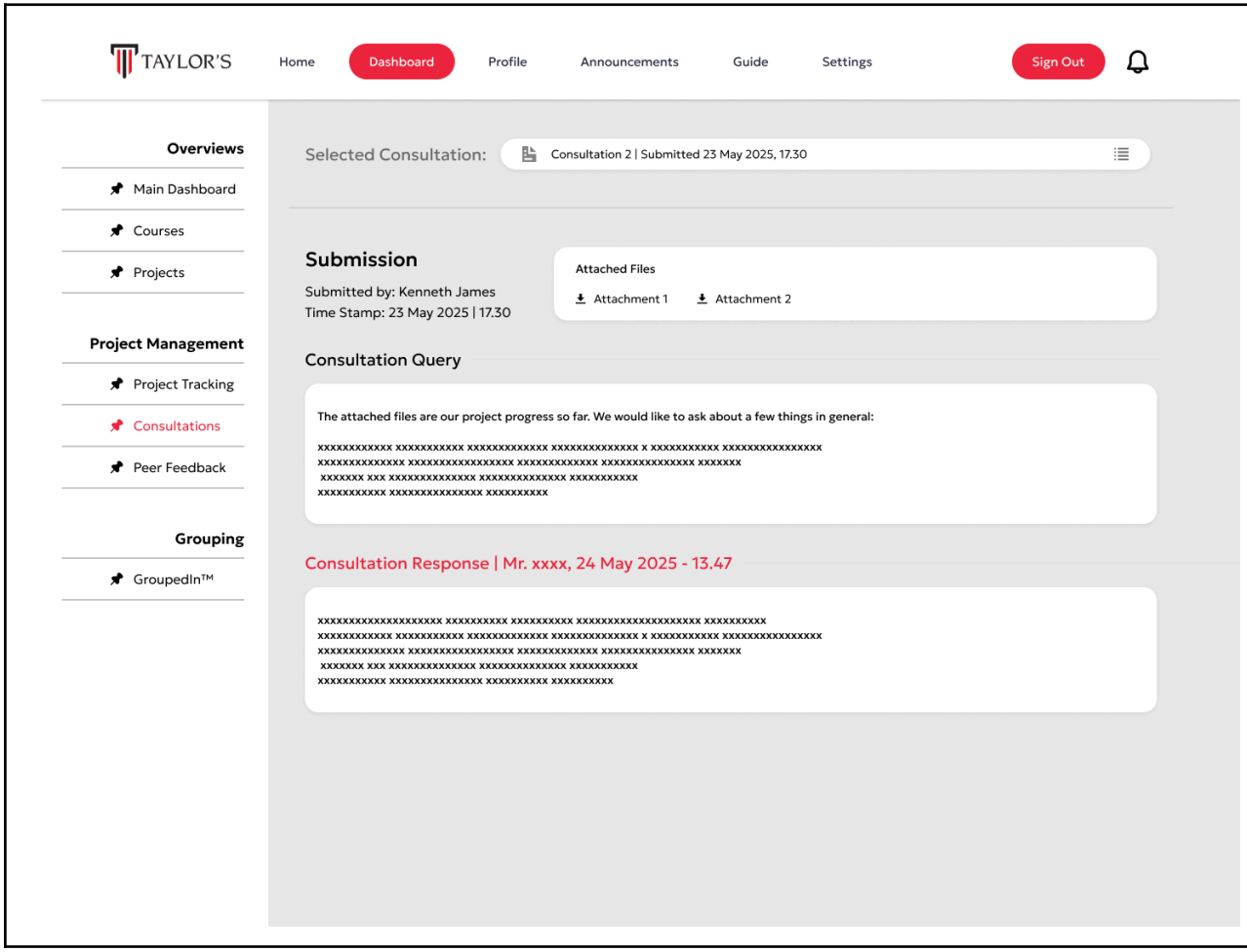



Student Consultation Main Page

The Student Consultation Main Page interface is designed to show students each of their projects and some alerts related to their consultation

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on one of the cards, it will open the Student Consultation Detail Page*

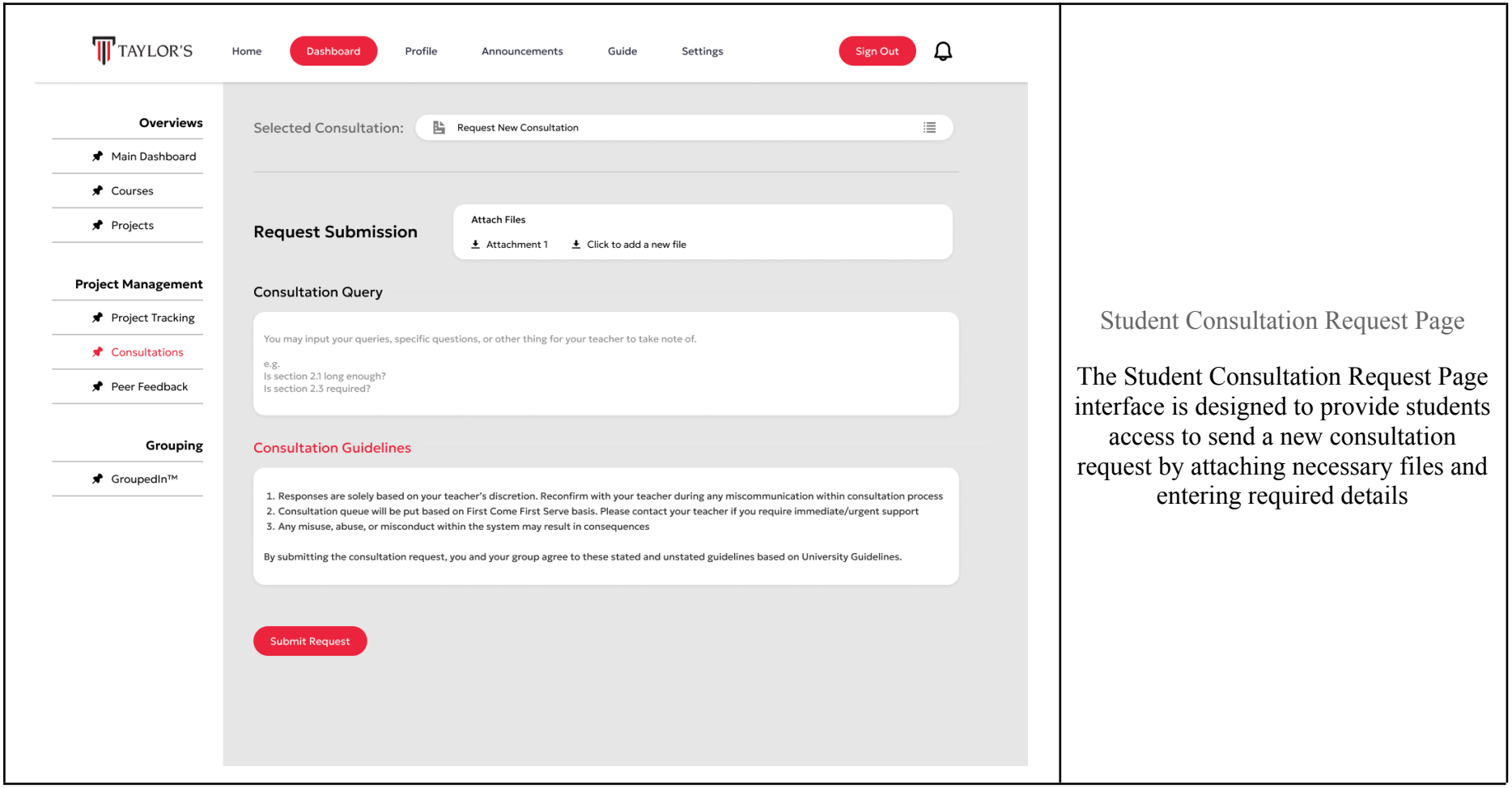



Student Consultation Detail Page

The Student Consultation Detail Page interface is designed to show students their consultation details

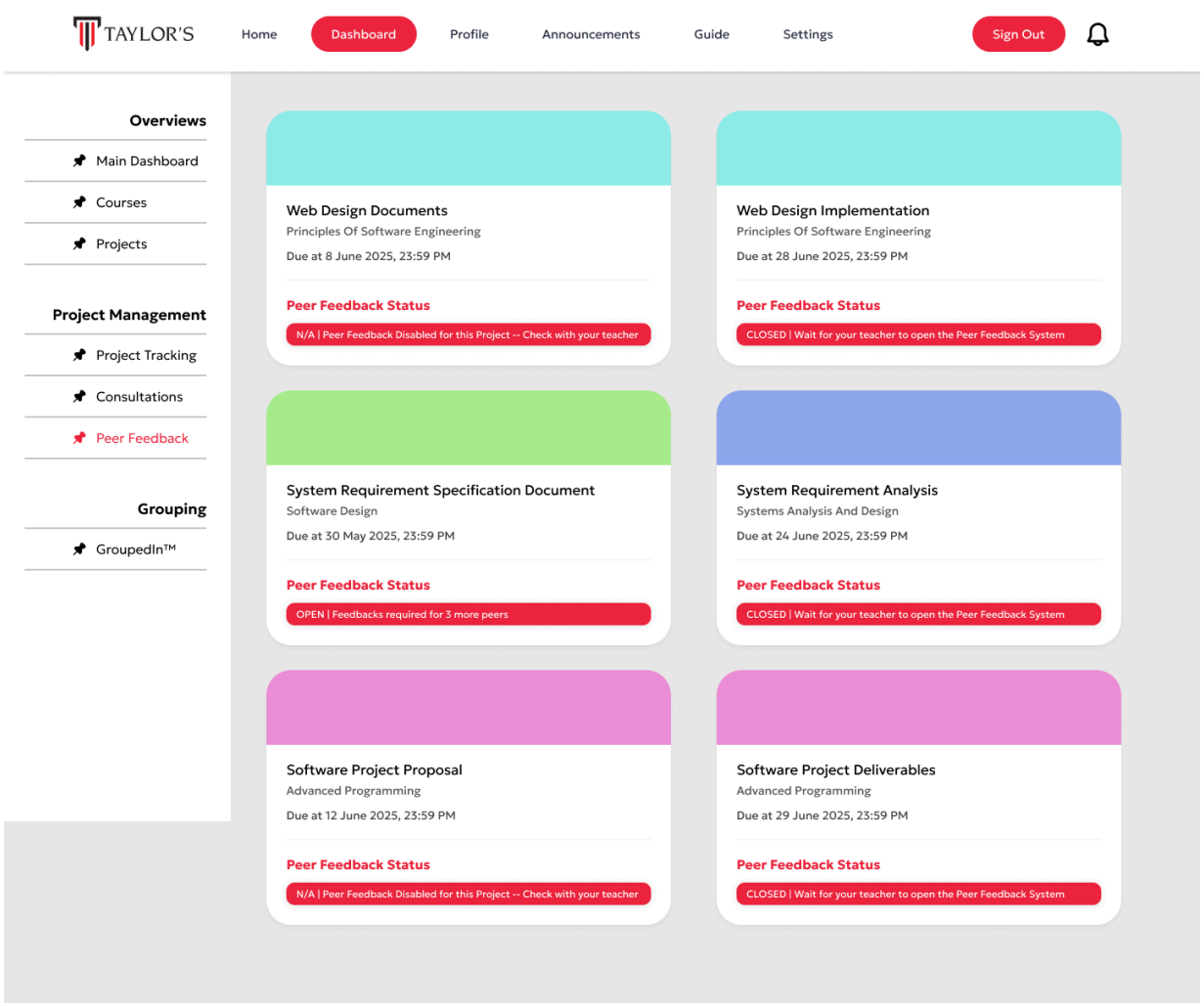
*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on Selected Consultation, a drop down will appear for them to select their previous consultations or make a new one. Making a new one will open Student Consultation Request Page*




Student Consultation Request Page

The Student Consultation Request Page interface is designed to provide students access to send a new consultation request by attaching necessary files and entering required details

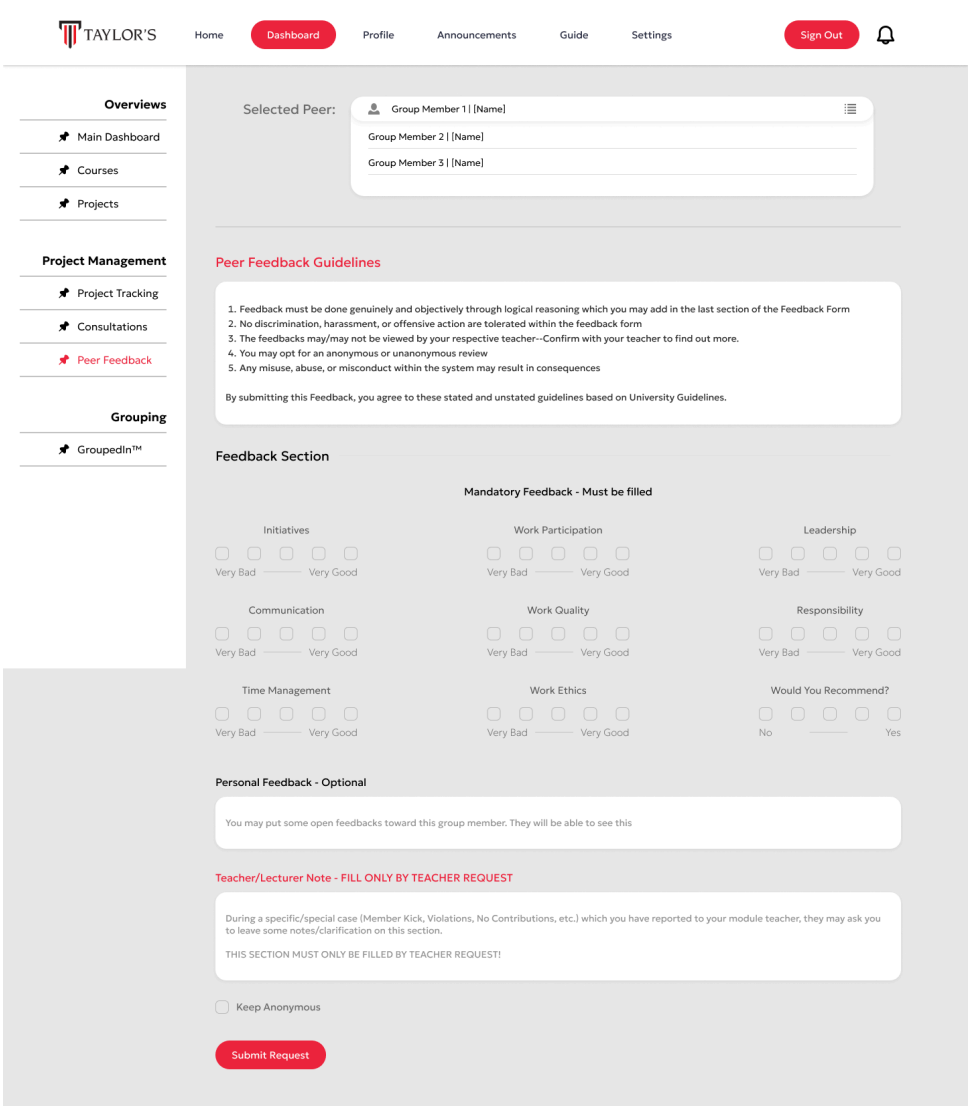



Student Peer Feedback Main Page

The Student Peer Feedback Main Page interface is designed to show students each of their projects and some alerts related to their Peer Feedbacks.

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on one of the cards, it will open the Student Peer Feedback Detail Page*



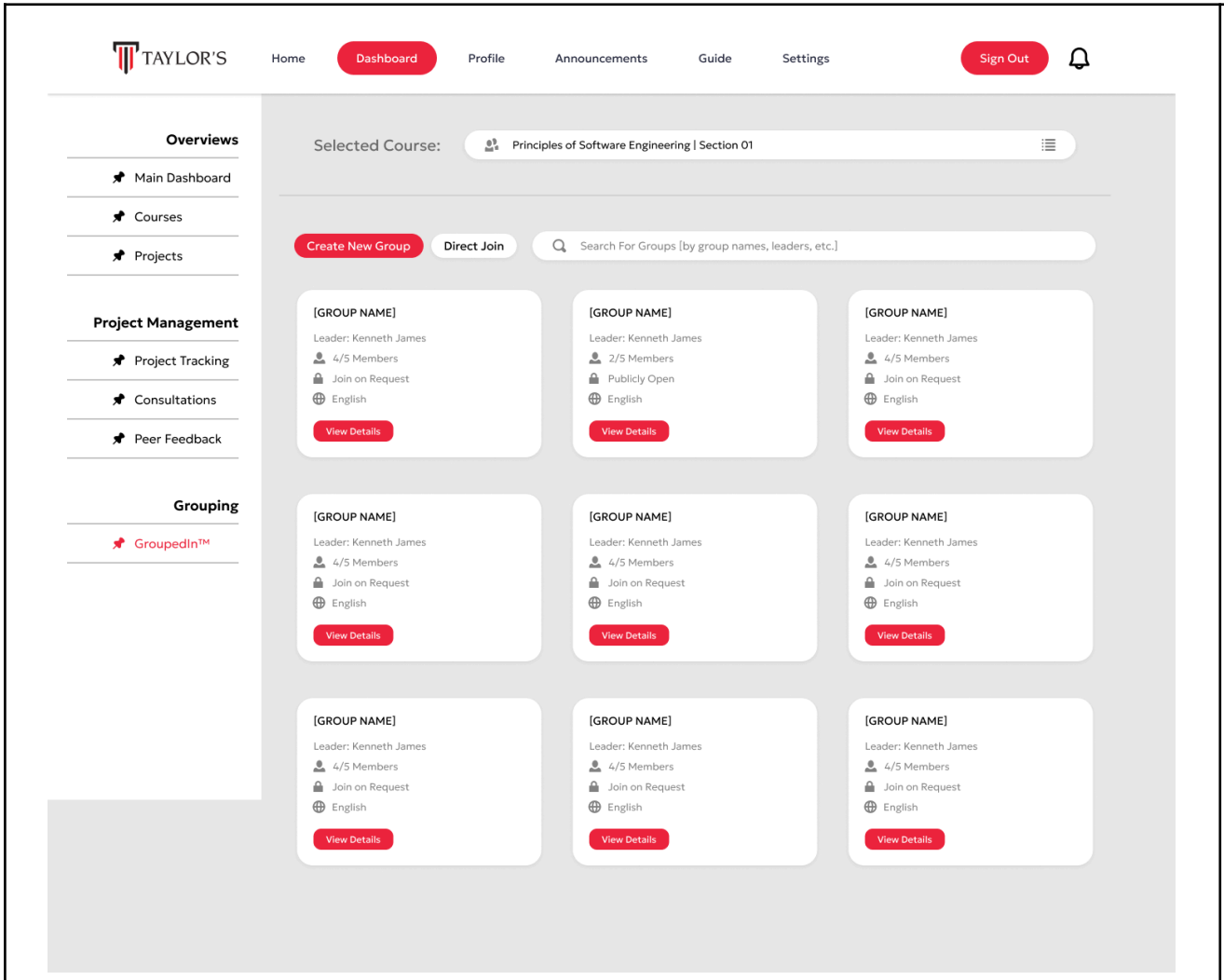

Student Peer Feedback Detail Page

The Student Peer Feedback Detail Page interface is designed to provide students an integrated Peer Feedback System.

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking Selected Peer, a drop down panel will appear consisting of member lists they need to give feedback.*

*In the Feedback Section, a "scale button" system has been implemented. In the last section, users can click the checkbox to keep the feedback anonymous.*




## Student GroupedIn™ Page

The Student GroupedIn Page interface is designed to show students lists of group opened to public for them to be able to join

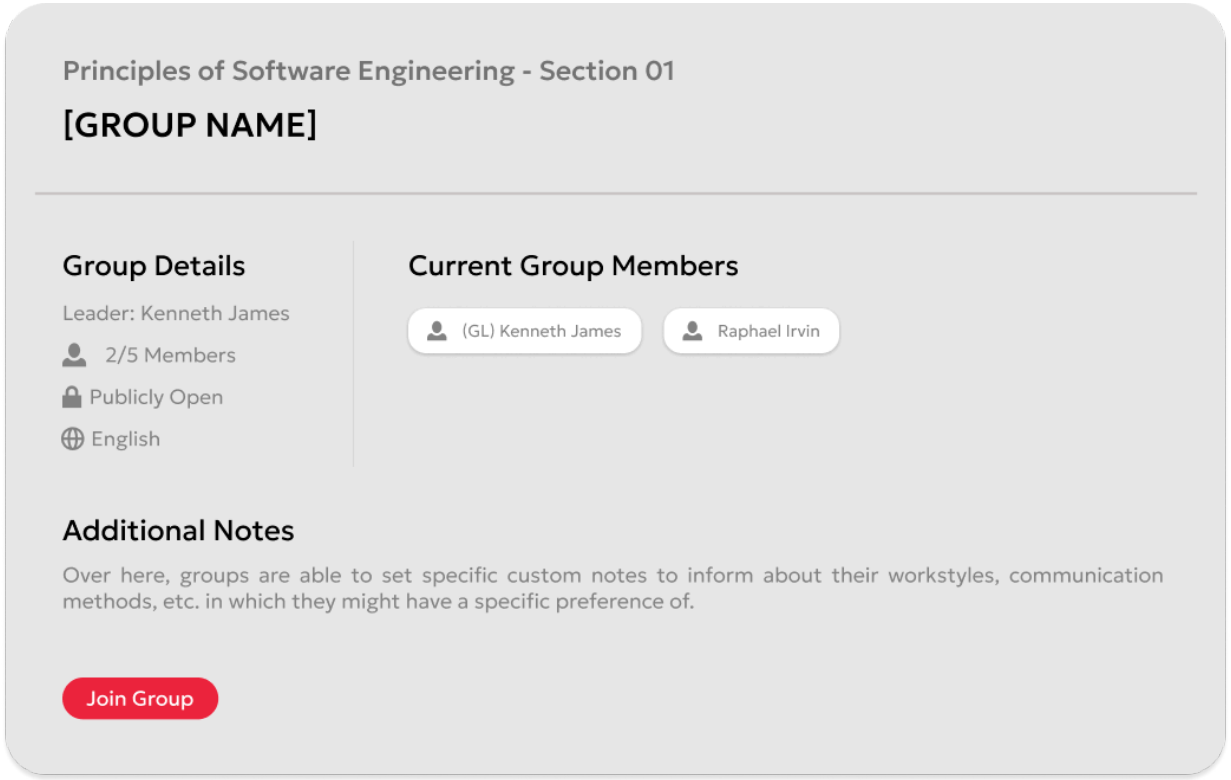
*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on Selected Course, it will open a drop down panel of lists of module*

*Upon clicking on View Details, it will open the Group Detail Overlay*

*Upon clicking on Create New Group, it will open the Student Group Creation Page*

*Upon clicking on Direct Join, it will show the Direct Join Overlay*

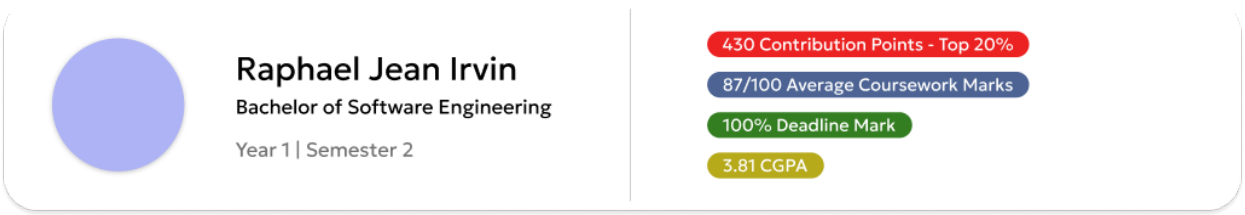



Group Detail Overlay

The Group Detail Overlay interface will be shown upon a user clicking View Details within the Student GroupedIn™ Page

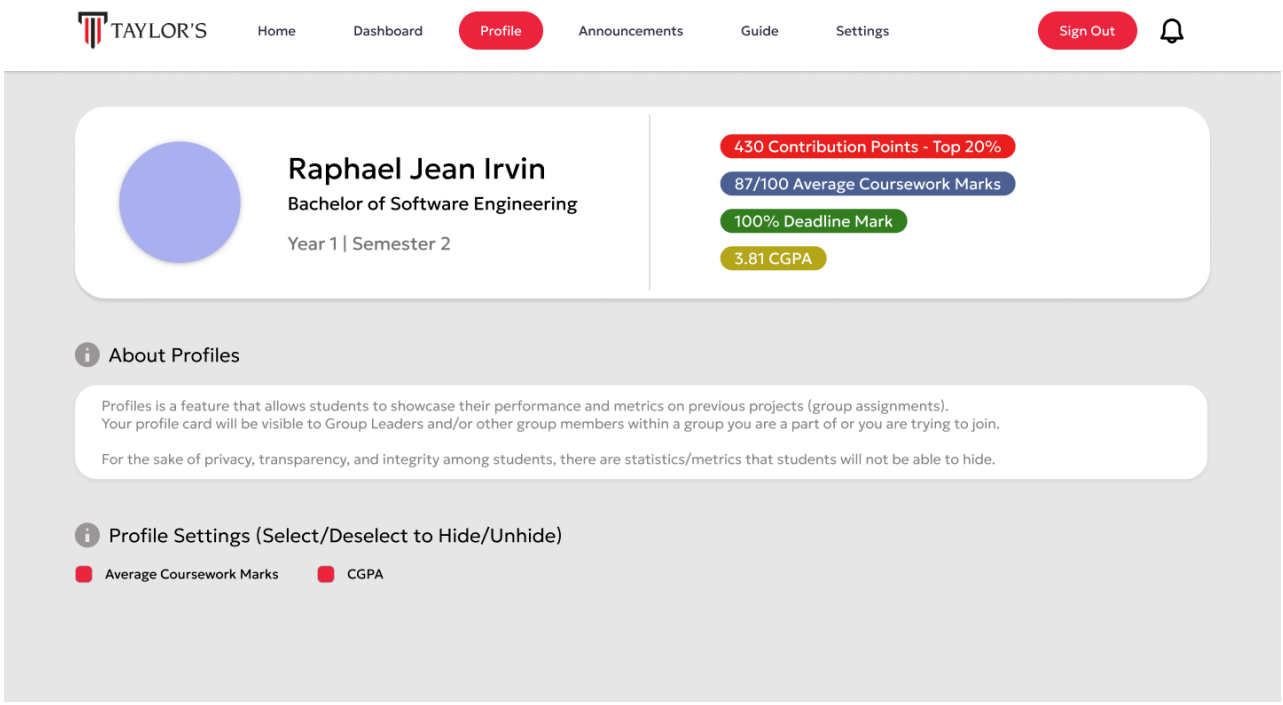
*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on a Profile Card/Button, it will open the Profile Card Overlay*




Profile Card Overlay

The Profile Card Overlay interface will be shown upon a user clicking a student profile in Group Detail Overlay according to the profile settings they set in the Student Profile Page

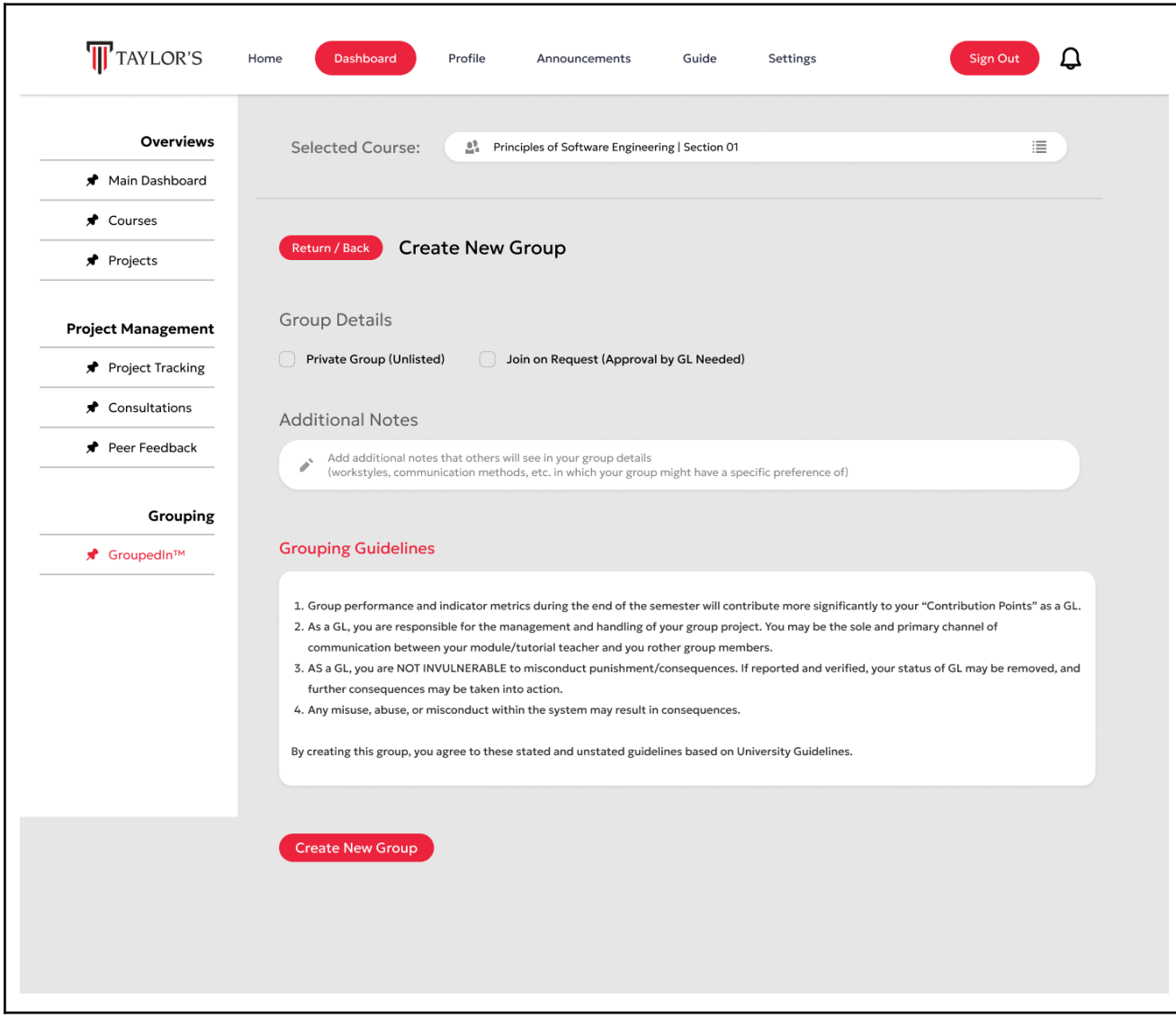



Student Profile Page

The Student Profile Page interface is designed to show them visualization of their Profile Card

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on the Profile Settings checkboxes, their options will be reflected on the profile card*

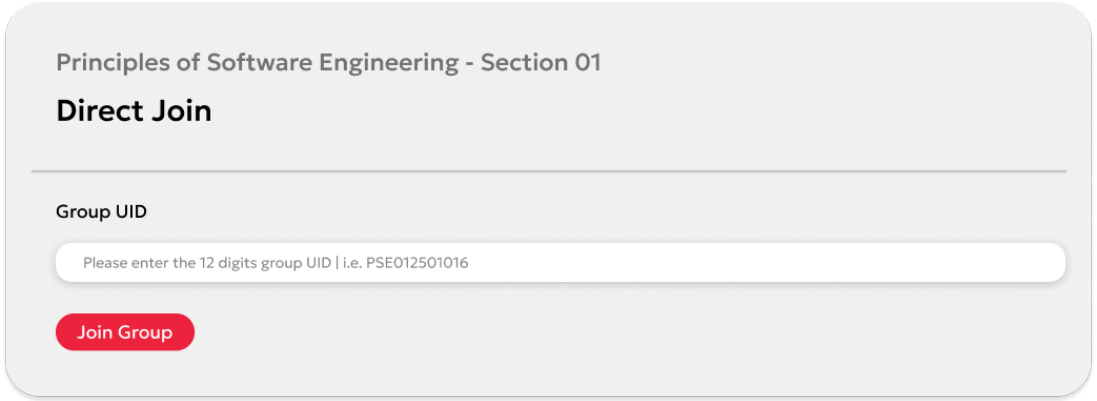



Student Group Creation Page

The Student Group Creation Page Interface is designed to give students access towards creating their own groups with respective details

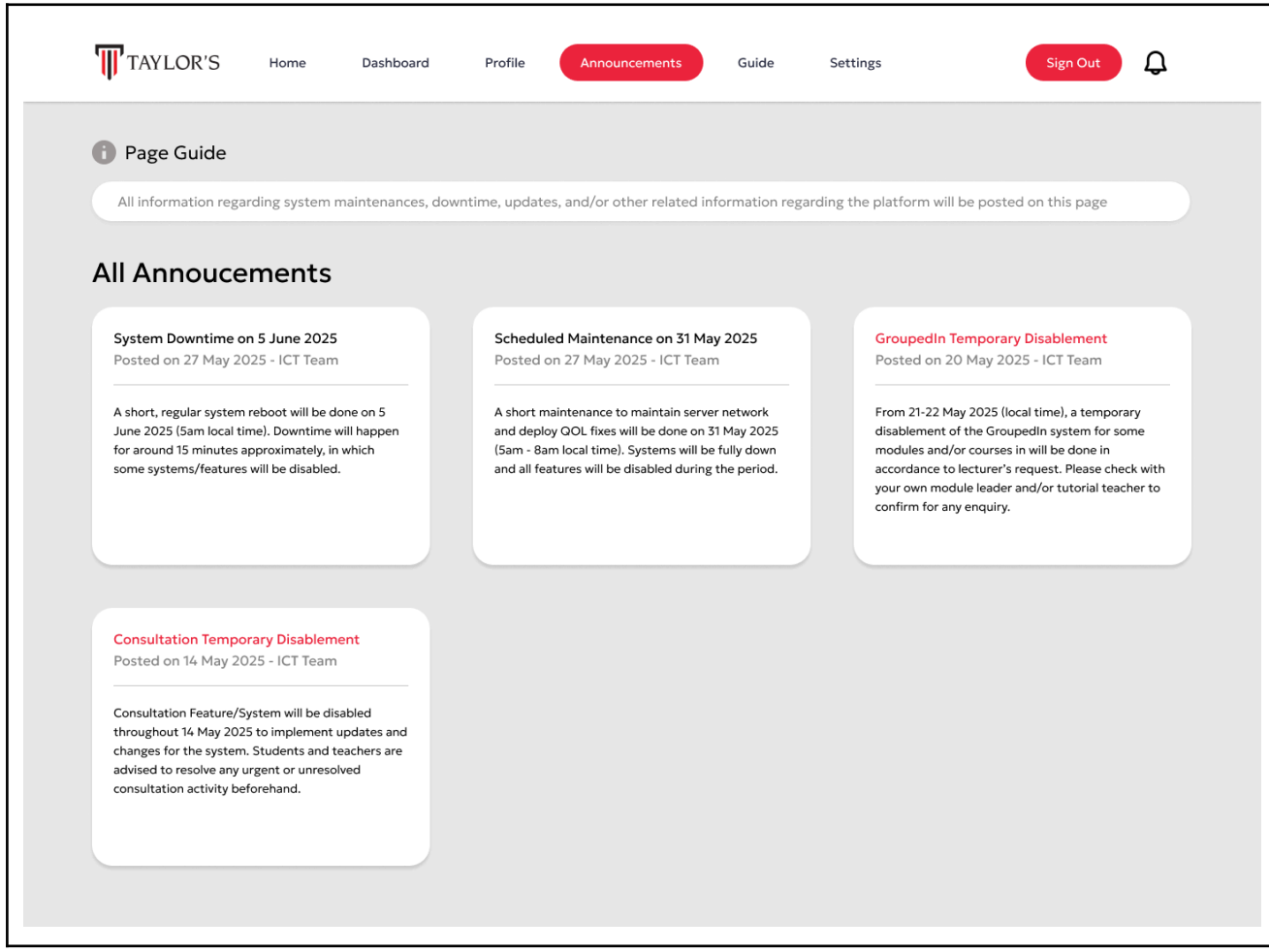
*Functionality:*
*(Accessible from prototype–Appendix D)*

*Checkboxes in the Group Details section can be enabled/disabled for students to choose whether they want to make their group public, restricted, or even private*



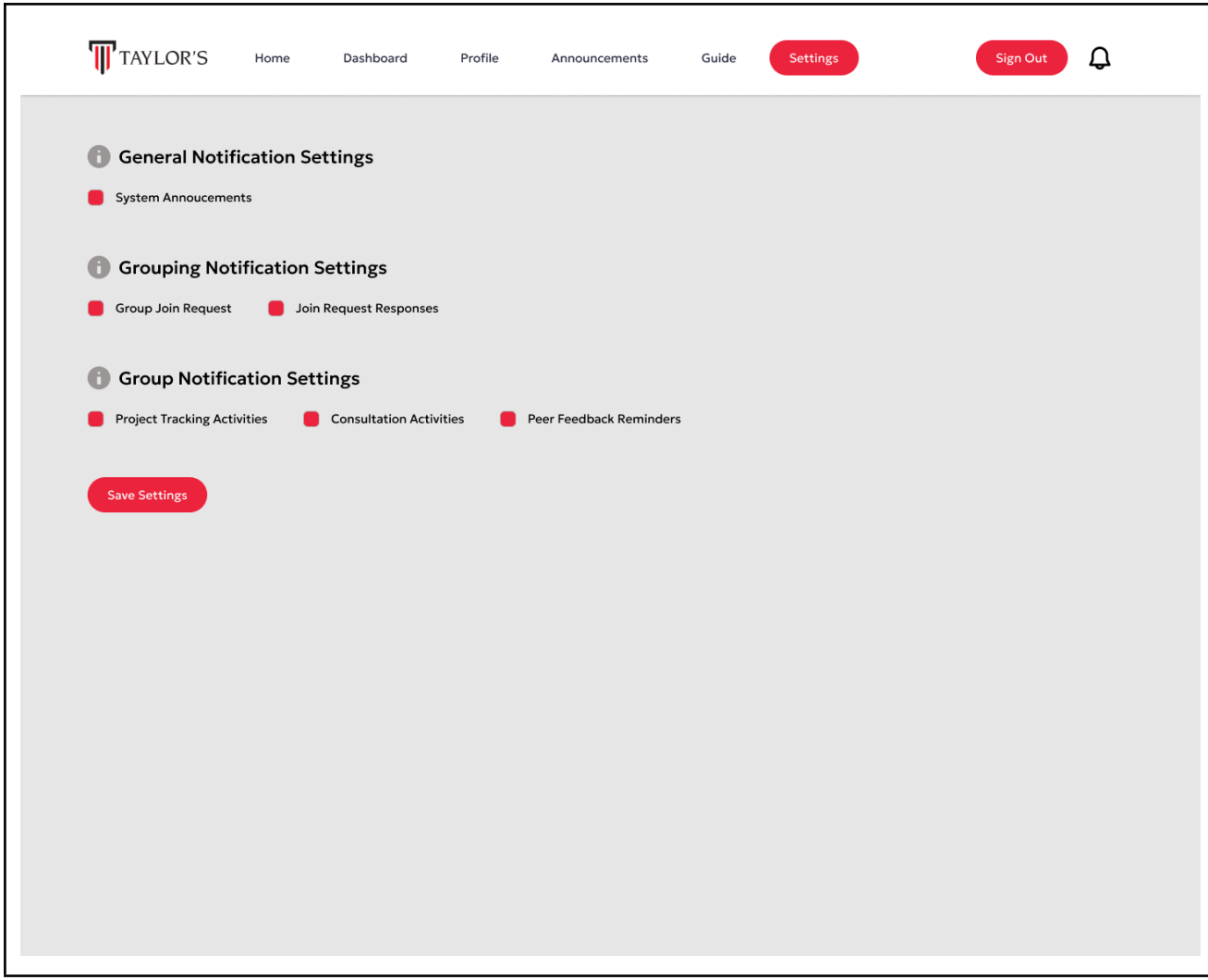

Direct Join Overlay

The Direct Join Overlay interface is shown when a person chooses Direct Join as a method of joining a group.




Student Announcement Page Page

The Student Announcement Page interface is designed to show latest System Announcements



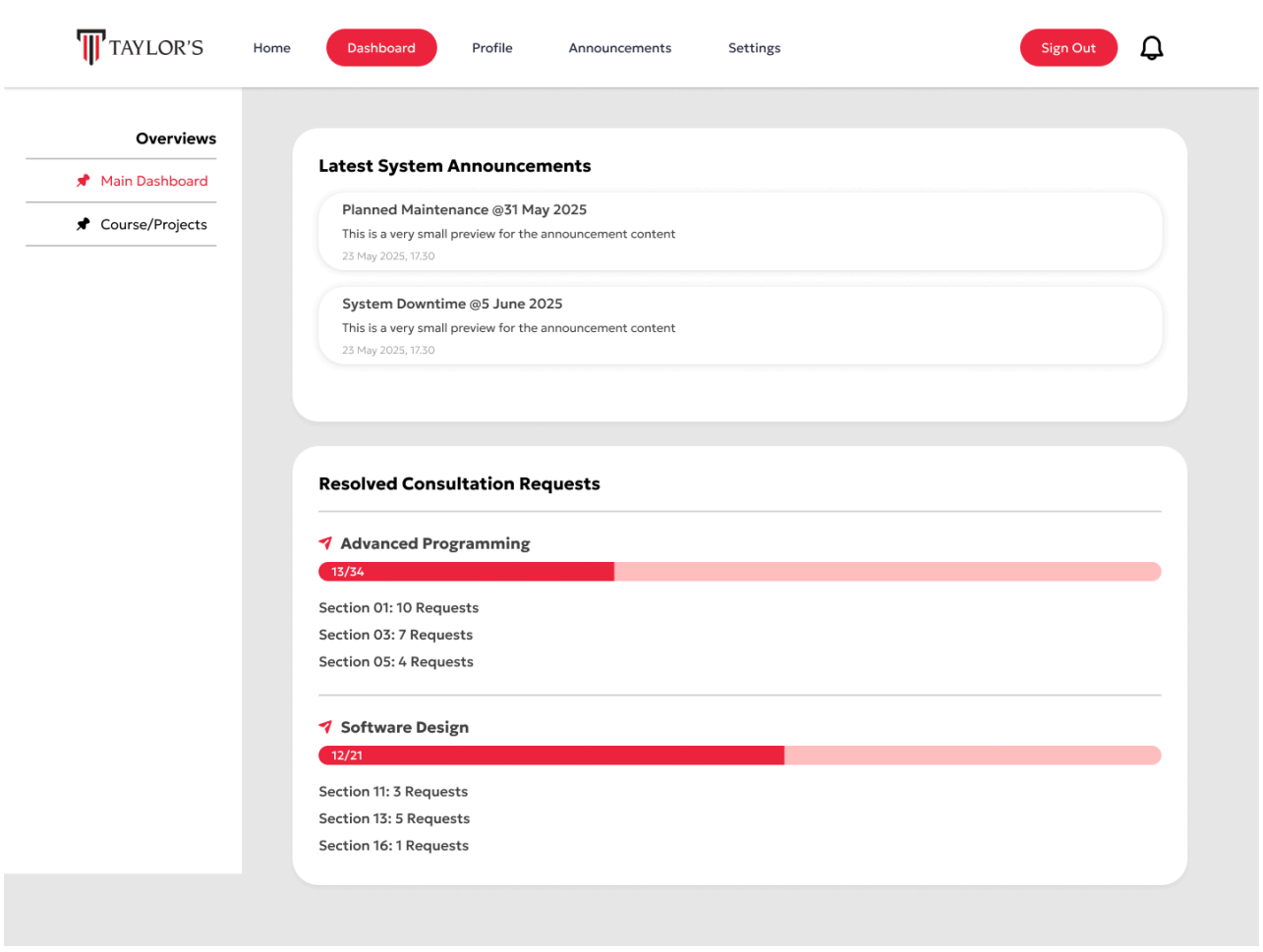

Student Settings Page

The Student Setting Page interface is designed to show Students of settings they could choose, in particular of notifications.

*Functionality:*
*(Accessible from prototype–Appendix D)*

*The checkboxes will are clickable to activate/deactivate the settings*

### *3.1.3 Lecturer*

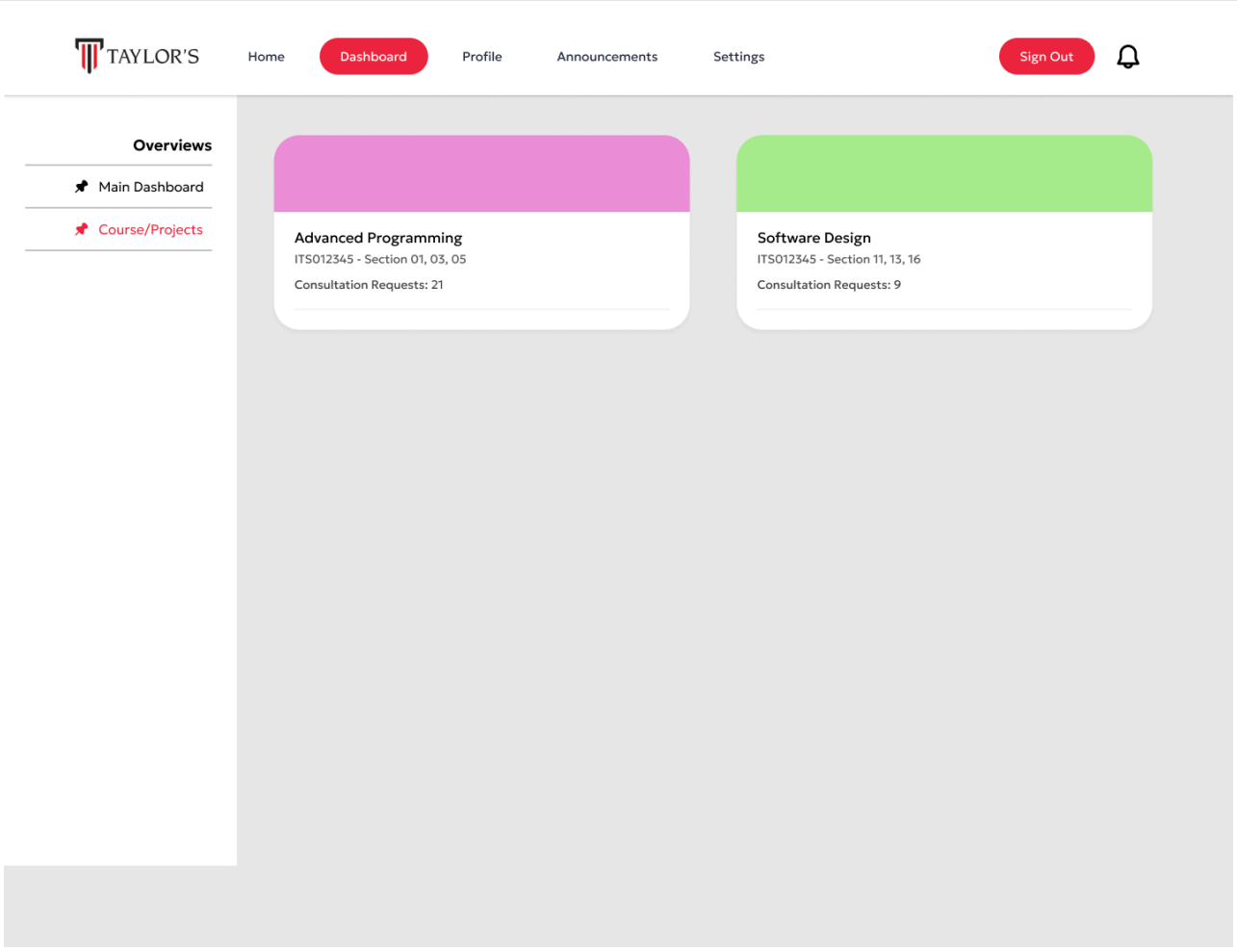



Lecturer Main Dashboard Page

The Lecturer Main Dashboard Page interface is designed to show Lecturers latest System Announcements and their progress of resolving consultation requests

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on the Module Name, they'll be redirected to the Lecturer Submission and Consultation Detail Page*

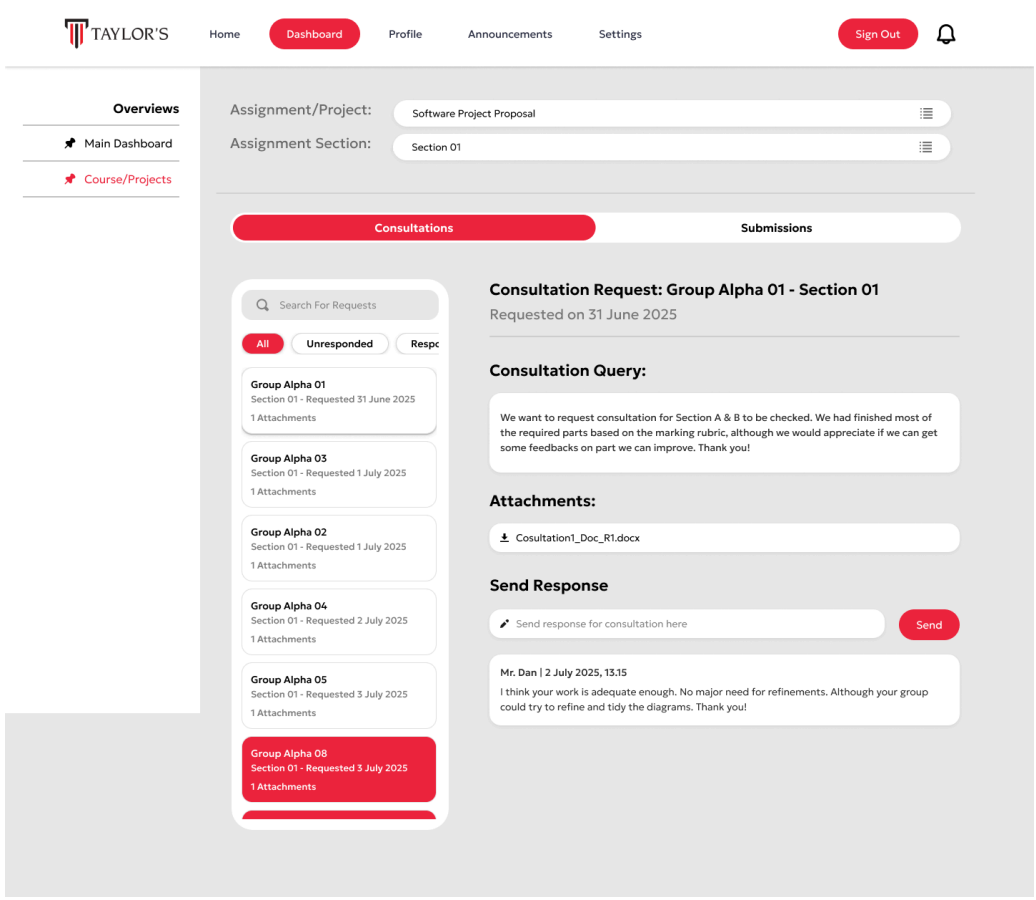



Lecturer Dashboard Course Page

The Lecturer Dashboard Course Page interface is designed to show Lecturers overview/informations about the modules they are currently handling

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on the Card, they will be redirected to the Lecturer Submission and Consultation Detail Page*



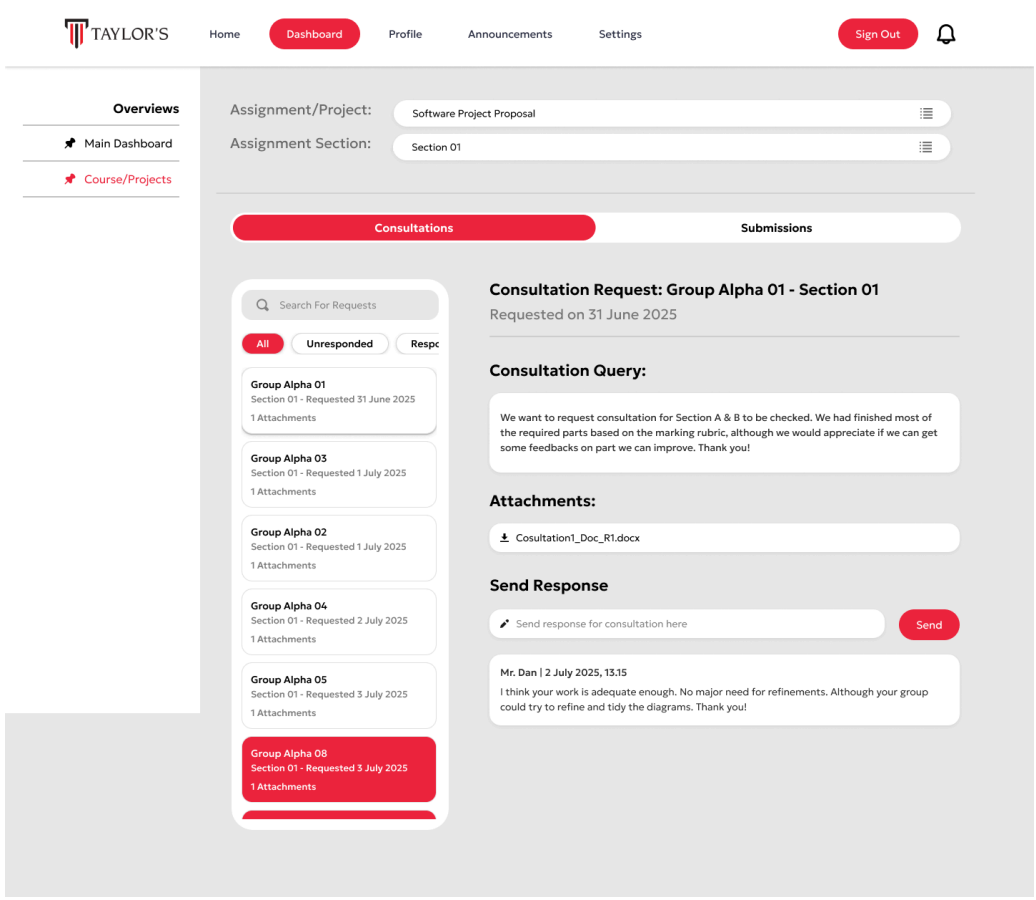

Lecturer Submission & Consultation Detail Page | Submission Page

The Lecturer Submission & Consultation Detail Page | Submission Page interface is designed to show Lecturers overview/informations all the consultation requests that needs to be responded

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on Submissions, it will open the Submission part of this page.*

*Filters can be pressed to sort consultation requests based on whether it has been Responded or not*



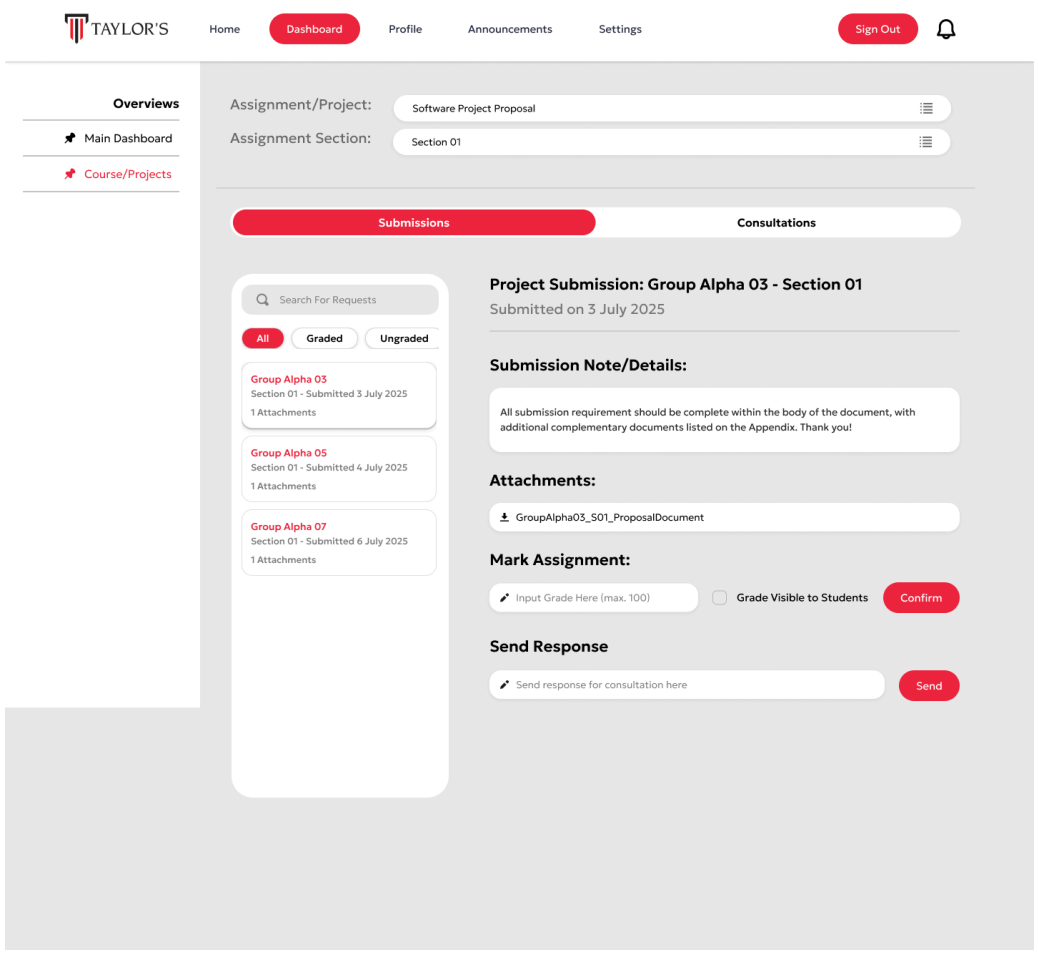

Lecturer Submission & Consultation Detail Page | Consultation Page

The Lecturer Submission & Consultation Detail Page | Submission Page interface is designed to show Lecturers overview/informations all the consultation requests that needs to be responded

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Upon clicking on Submissions, it will open the Submission part of this page.*

*Filters can be pressed to sort consultation requests based on whether it has been Responded or not*



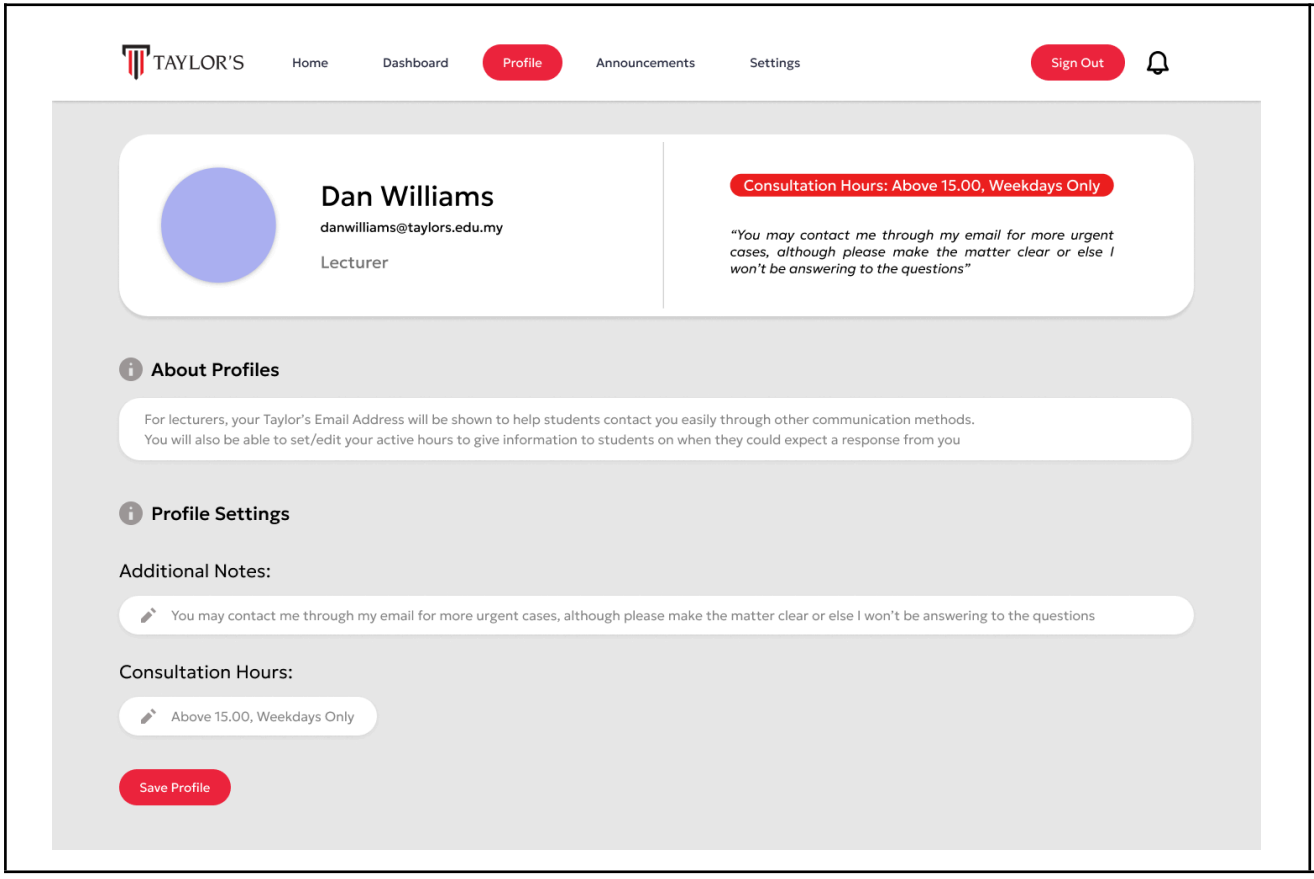

Lecturer Submission & Consultation Detail Page | Submission Page

The Lecturer Submission & Consultation Detail Page | Submission Page interface is designed to show Lecturers overview/informations all the submissions that needs to be checked

*Functionality:*
*(Accessible from prototype–Appendix D)*

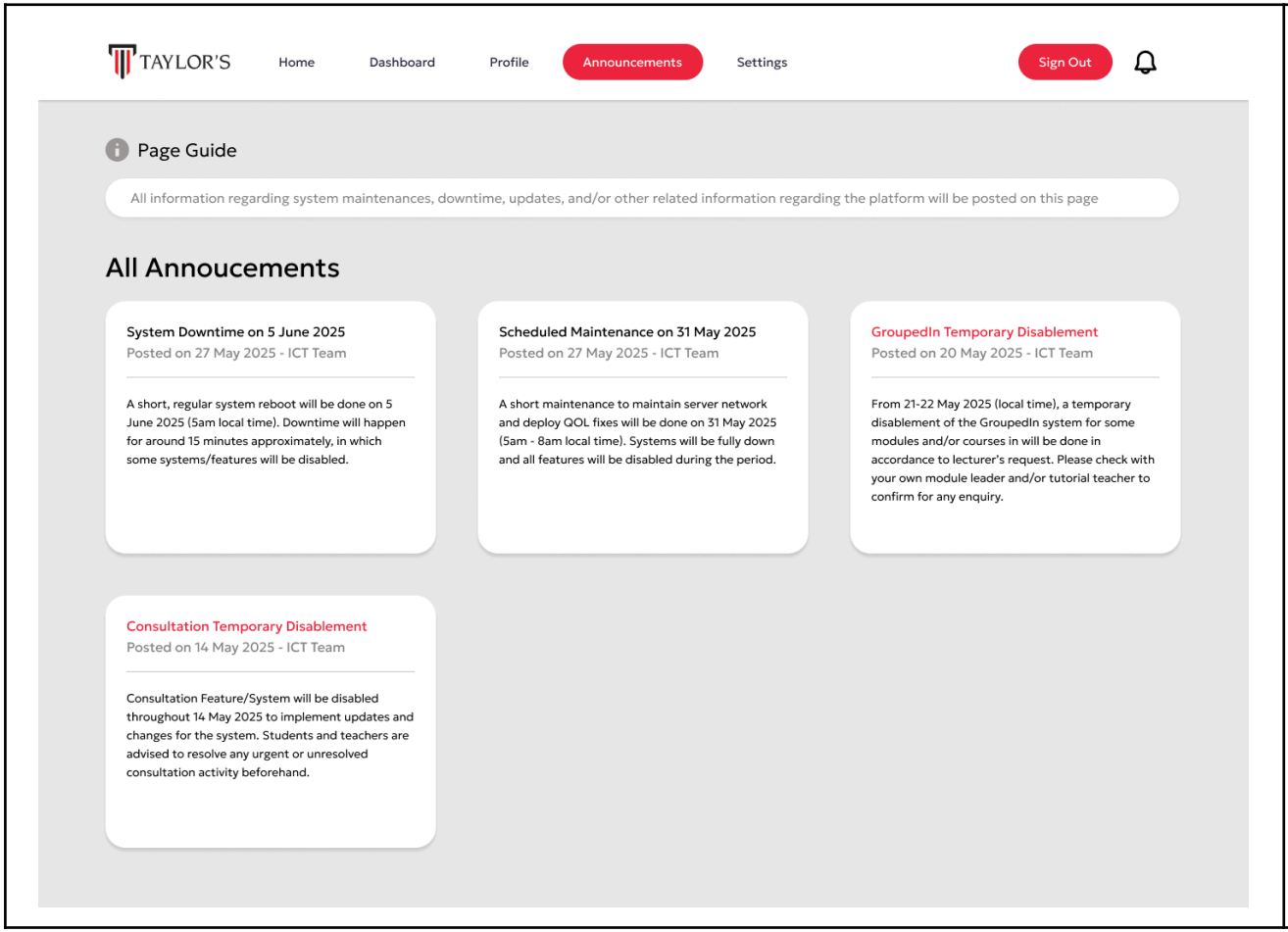
*Upon clicking on Consultations, it will open the Consultations part of this page.*

*Filters can be pressed to sort consultation requests based on whether it has been Graded or not*



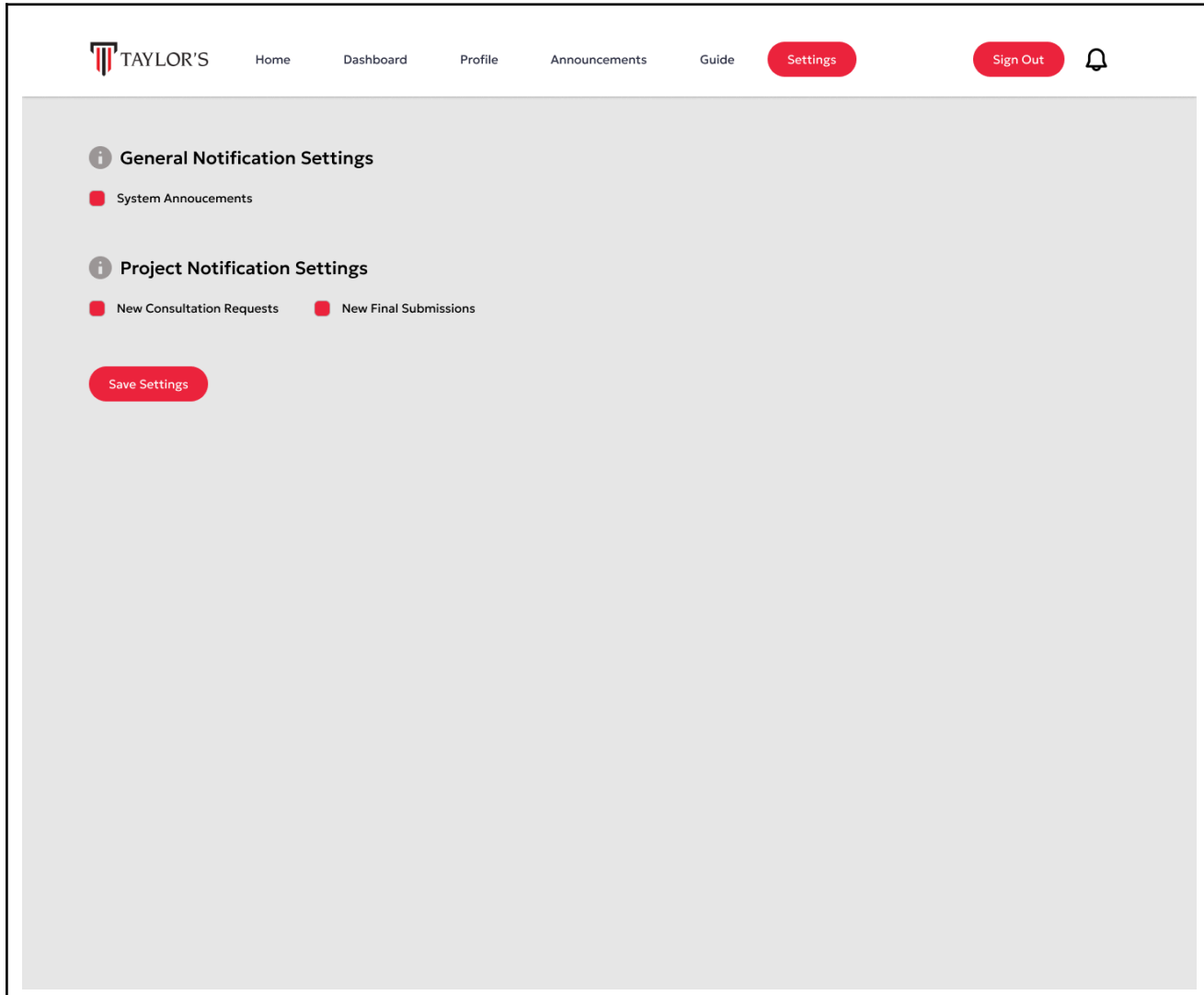

Lecturer Profile Page

The Lecturer Profile Page interface is designed to show them visualization of their Profile Card




Lecturer Announcement Page

The Lecturer Announcement Page interface is designed to show latest System Announcements



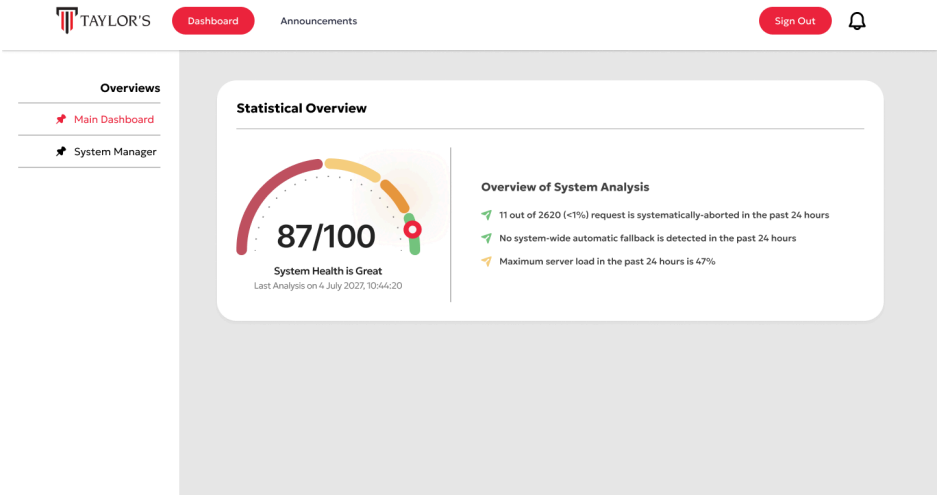
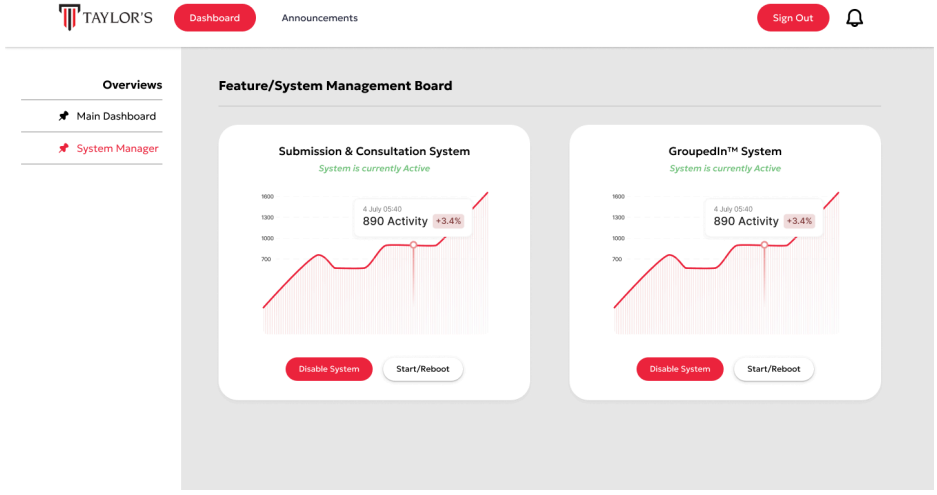

Lecturer Settings Page

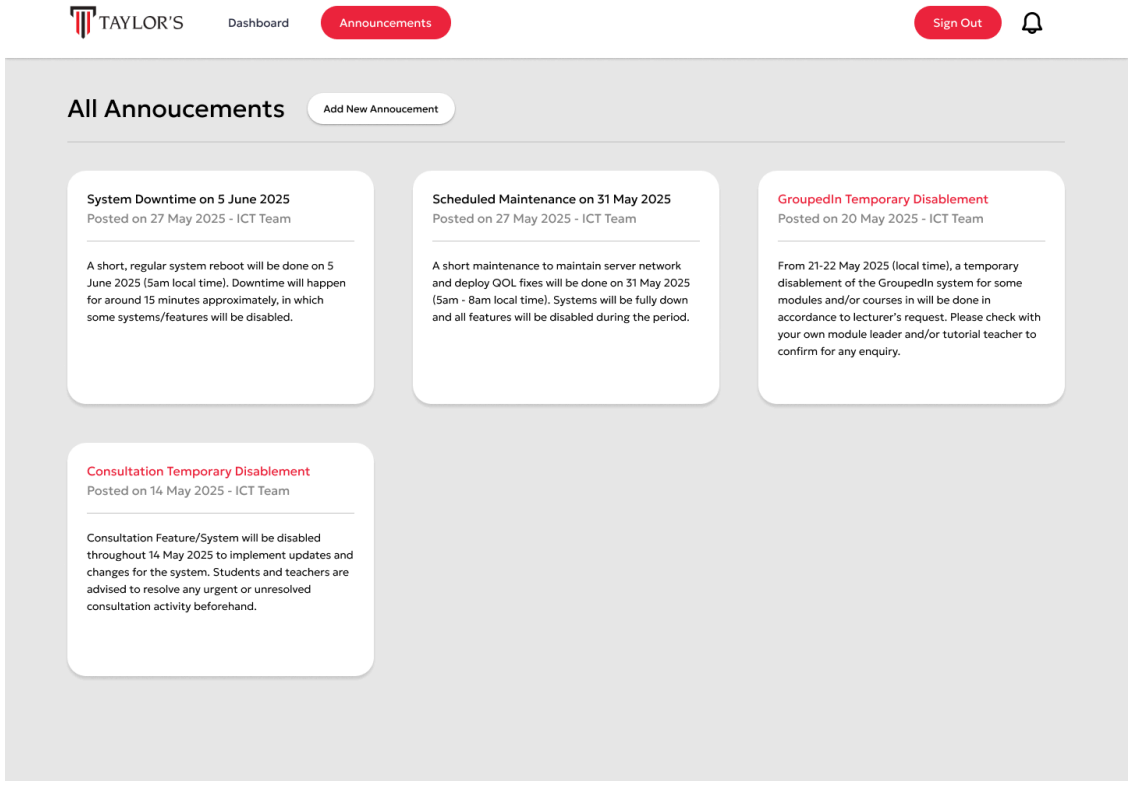
The Lecturer Setting Page interface is designed to show Lecturers of settings they could choose, in particular of notifications.

*Functionality:*
*(Accessible from prototype–Appendix D)*

*The checkboxes will are clickable to activate/deactivate the settings*

### *3.1.4 System Admin*

| | |
|---|---|
|  | System Admin Main Dashboard Page<br><br>The System Admin Main Dashboard Page interface is designed to show admins the overall statistical analysis and health of the system |
|  | System Admin Feature Dashboard Page<br><br>The System Admin Feature Dashboard Page interface is designed to show admins the detailed activity status of each feature, and for them to Disable or Reboot/Start a specific System/Feature for maintenance purposes |

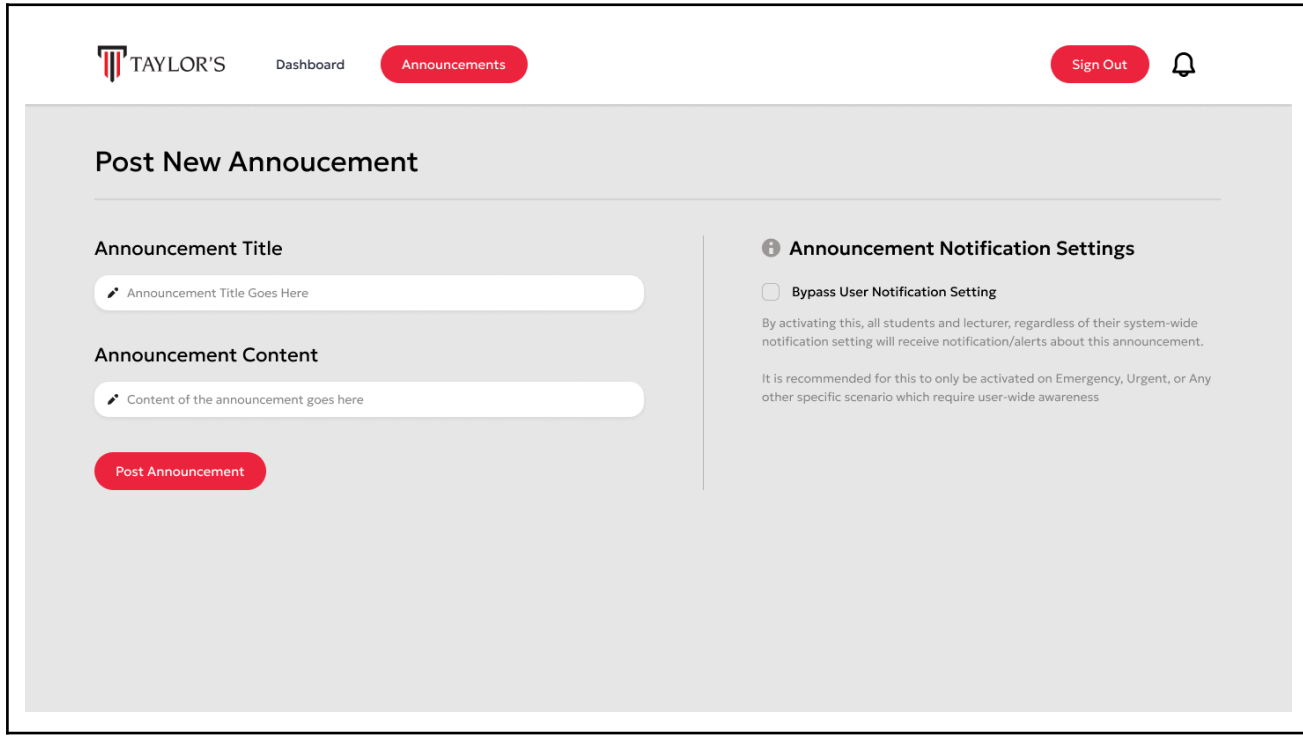



System Admin Announcement Page

The System Admin Announcement Page interface is designed to show latest System Announcements

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Clicking on Add New Announcement will open the System Admin New Announcement Page*




System Admin New Announcement Page

The System Admin New Announcement Page interface is designed to give access to admins for them to post new announcements

*Functionality:*
*(Accessible from prototype–Appendix D)*

*Clicking on Bypass User Notification Setting will enable the interactive checkbox*

### 3.2 Documentation of Iterative Development

***Overview of Iterative Process***

The development of MyGroups followed an Agile Iterative Model, divided into four sprint cycles. Each sprint focused on a set of feature goals, ensuring incremental delivery, continuous feedback incorporation, and progressive refinement.

Although the system was not deployed or tested in real-time environments, the iteration cycles were simulated based on planned features, anticipated user experience improvements, and evolving design insights during the design phase.

***Iterative Documentation***

**Disclaimer:*

*Due to the nature of the project/assignment, where practical regulated/routine Iterative Activity are not likely to be conducted as expected, the following table showcasing Iterative Documentation may not fully reflect the iterative development done by the team (mostly in Author column).* ***The table shown below may serve as a baseline of changes that occur during the prototype development.***

| ***Version*** | ***Date & Time*** | ***Type*** | ***Description*** |
|---|---|---|---|
| *v0.1.0* | *2025-05-01 10:15 AM* | *Added* | *Initial wireframes (Student & Lecturer UI) created and uploaded to Figma.* |
| *v0.1.1* | *2025-05-03 3:40 PM* | *Changed* | *Navigation bar layout revised from top-only to sidebar approach.* |
| *v0.1.2* | *2025-05-05 11:25 AM* | *Fixed* | *UI spacing inconsistencies on mobile resolved.* |
| *v0.2.0* | *2025-05-06 6:00 PM* | *Added* | *GroupedIn™ mockups introduced. Consultation system flow drafted.* |
| *v0.2.1* | *2025-05-07 8:30PM* | *Changed* | *Reworded Group* |

| | | | |
|---|---|---|---|
| | | | *Request logic and added filters for smart grouping.* |
| *v0.3.0* | *2025-05-10 9:00AM* | *Added* | *Admin dashboard concepts and system monitoring cards added.* |
| *v0.3.1* | *2025-05-11 12:20 PM* | *Changed* | *Peer Feedback System changed to include anonymity toggle and scale UI.* |
| *v0.4.0* | *2025-05-13 4:00PM* | *Added* | *Notification Center added; status-based color badges finalized.* |
| *v1.0.0* | *2025-05-15 10:00 AM* | *Final* | *Interface prototype and SRS synced for initial submission phase.* |

# SRS Document

The following section constitutes the comprehensive Software Requirements Specification (SRS) for the project. While it outlines the current requirements and system design details, it is intended to serve as a baseline reference and not as the final deliverable.

Due to the evolving nature of the project, this document may be updated or refined as new information emerges or requirements change.

# Table of Contents


**1. Introduction** **55**
1.1. Purpose 55
1.2. Scope of Project 55
1.3. Document Overview 55

**2. System Overview** **56**
2.1. System Use Case Overview 56
2.2. Functional Requirement Specifications 57
2.2.1. General Use Case 57
2.2.2. Student Use Case 57
Use Case: View Profile 57
Use Case: Create Group 57
Use Case: View Project Progress/Details - Group Members 57
Use Case: Manage Projects/Groups - Group Leaders 58
2.2.3. Lecturer Use Case 58
Use Case: View Submission Detail 58
Use Case: View Consultation Request Detail 59
2.2.4. Staff Use Case 59
Use Case: Post Announcements 59
Use Case: Enable/Disable Feature (Systems) 59
2.3. Non-Functional Requirement Specifications 59
2.3.1. Load Balancing 59
2.3.2. Security 60
2.3.3. Performance and Responsiveness 60
2.3.4. Accessibility and Usability 60
2.4. Requirement Traceability Matrix 61
2.5. System Architecture Overview 66
2.6. Interface Overview 68

# 1. Introduction

## 1.1. Purpose

The purpose of this document is to define the software requirements for the MyGroups platform, an educational group project management system intended to support students and lecturers at Taylor's University. This Software Requirements Specification (SRS) outlines the functional and non-functional requirements, system architecture overview, use case models, and design considerations necessary for successful development and deployment. It shall serve as a mutual agreement between stakeholders, including developers, clients, and users.

## 1.2. Scope of Project

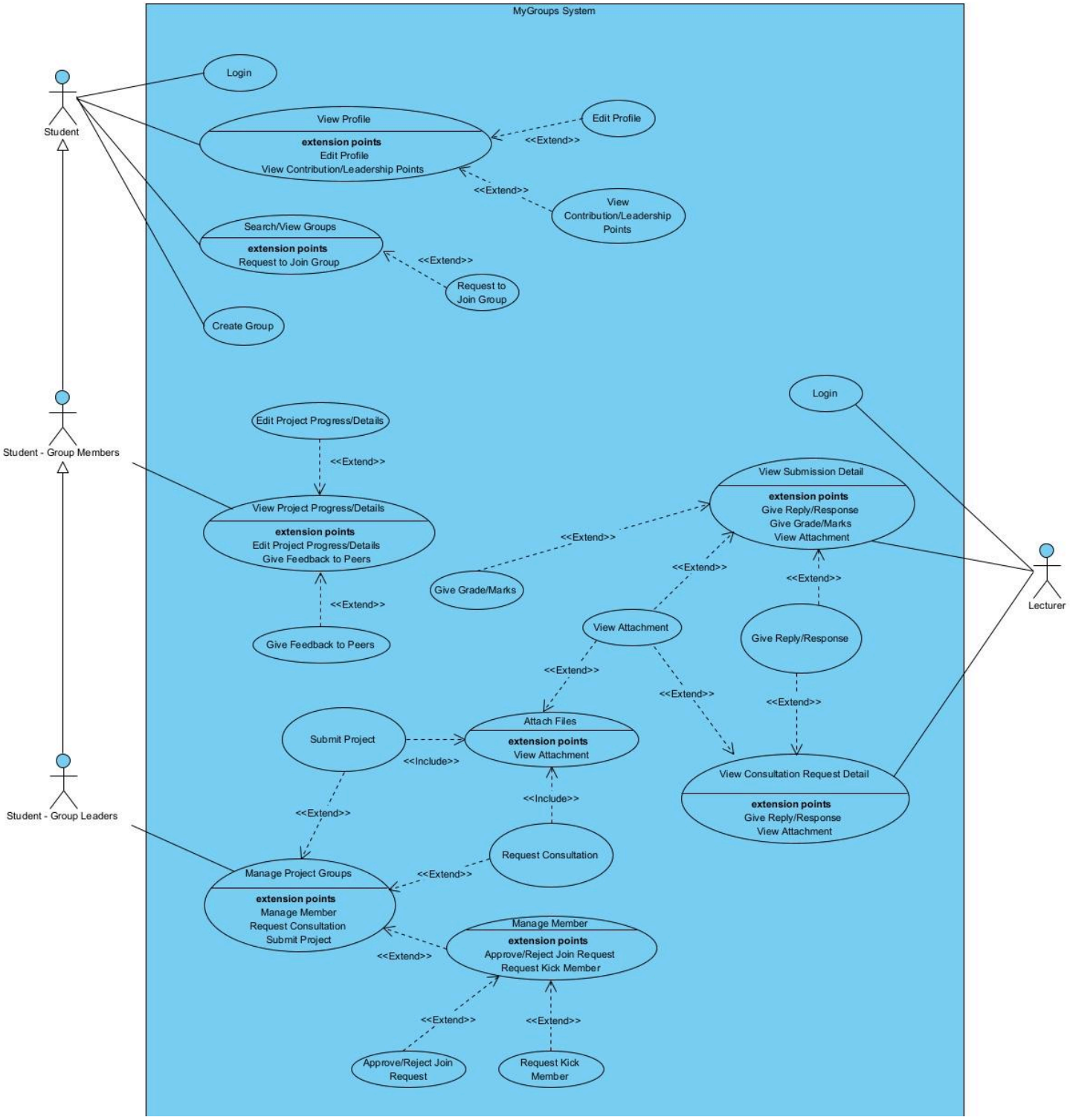
MyGroups is a centralized platform designed to assist students and lecturers in managing group assignments and academic collaboration more effectively. It aims to provide features such as group pairing, project tracking, peer evaluations, and consultation management, all while promoting holistic student development. The system will support different user roles (students, lecturers, staff) and deliver personalized functionalities for each. It will be integrated into Taylor's University's existing digital ecosystem, complementing platforms like MyTimes and CampusCentral.

## 1.3. Document Overview

This document is organized into several sections that describe the project's objectives, functional and non-functional requirements, use case scenarios, system design, and implementation strategies. It includes project concepts, system overview, comprehensive explanations, and a detailed breakdown of use cases per user role. It is intended for both technical and non-technical stakeholders involved in the software design and development process.

# 2. System Overview

## 2.1. System Use Case Overview



## 2.2. Functional Requirement Specifications

### *2.2.1. General Use Case*

**Use Case: Login**

The login feature is integrated as a general feature to verify/assign users their specific role (student/lecturer/staff) and grant them access to the respective features.

### *2.2.2. Student Use Case*

**Use Case: View Profile**

Allows students to view their own profile information including name, student ID, group status, and activity summary.

Extension 1: Edit Profile

Allows students to update their personal information.

Extension 2: Search/View Groups

Lets students search for existing groups or view public group listings for a specific subject.

Extension 2.1: Request to Join Group

Enables students to send a join request to the selected group.

**Use Case: Create Group**

Allows students to create a new group under a specific subject with optional group description and settings.

**Use Case: View Project Progress/Details - Group Members**

Students can view the project status, task breakdown, and deadlines for their current group project.

Extension 1: Edit Project Progress/Details

Group members can update the project timeline, mark tasks as complete, or revise project details.

Extension 2: Give Feedback to Peers

Enables peer evaluation by rating or providing feedback on a group member's contribution.

**Use Case: Manage Projects/Groups - Group Leaders**

Group leaders are granted tools to oversee project settings, monitor task completion, and manage team dynamics.

Extension 1: Submit Project

Allows the group leader to submit the final project deliverables.

*Inclusion 1.1: Attach Files*

Enables the leader to attach necessary files (e.g., reports, source code) during submission.

*Extension 1.1.1: View Attachment*

Lecturers or group members can access attached files in the submission.

Extension 2: Request Consultation

Allows the group leader to formally request a consultation with a lecturer regarding the project.

*Inclusion 2.1: Attach Files*

Enables the user to include reference files (e.g., drafts or questions) with the consultation request.

*Extension 2.1.1: View Attachment*

Lecturers or group members can access attached files in consultation threads.

Extension 3: Manage Member

Allows the group leader to monitor and manage team members.

*Extension 3.1: Approve/Reject Join Request*

Leaders can accept or reject incoming join requests to the group.

*Extension 3.2: Request to Kick Members*

Leaders can initiate a removal request for inactive or disruptive members.

### *2.2.3. Lecturer Use Case*

**Use Case: View Submission Detail**

Lecturers can review submitted projects, including attached files and group information.

Extension 1: Give Reply/Response

Enables lecturers to leave comments or feedback on the submitted work.

Extension 2: View Attachments

Allows lecturers to open and assess the files submitted along with the project.

Extension 3: Give Grade/Marks

Enables the lecturer to assign scores or grades to the group submission.

**Use Case: View Consultation Request Detail**

[Description]

Extension 1: Give Reply/Response

Enables lecturers to leave comments or feedback on the submitted work.

Extension 2: View Attachments

Allows lecturers to open and assess the files submitted along with the project.

#### *2.2.4. Staff Use Case*

**Use Case: Post Announcements**

Enables administrative staff to post system-wide announcements for all users.

Inclusion 1: Set Title

Let the staff input a subject line for the announcement.

Inclusion 2: Set Content

Allows staff to write the full content or message body of the announcement.

**Use Case: Enable/Disable Feature (Systems)**

Allows authorized staff to enable or disable specific features in the system temporarily.

Inclusion 1: Set Notice Message

Staff can leave a system-wide note explaining why a feature is disabled or modified.

### 2.3. Non-Functional Requirement Specifications

#### 2.3.1. **Load Balancing**

The requirement for a proper and effective load balancing implementation is deemed critical as the scale and the scope of the project is designed to be used simultaneously across hundreds or even thousands access points in a single moment. Effective load balancing through [ways to implement/make sure of load balancing] is essential to ensure that system downtime and sudden outages are reduced to minimum. Utilizing cloud infrastructure with

auto scaling capabilities (e.g., AWS Elastic Load Balancer or Google Cloud Load Balancer) may also be considered for future scalability and reliability.

### 2.3.2. Security

Given that MyGroups handles user credentials, academic information, and private group communications, strong security measures are mandatory. These include:

1. User authentication via secure protocols (e.g., OAuth 2.0 or two-factor authentication)
2. Data encryption (SSL/TLS for data in transit, AES for data at rest)
3. Role-based access control (RBAC) to ensure users only access permitted functionalities
4. Regular penetration testing and vulnerability assessments
5. Secure API access using tokens or API keys

These measures will safeguard user privacy and ensure system integrity against potential attacks or misuse.

### 2.3.3. Performance and Responsiveness

- The system must ensure consistent and responsive user experiences across a wide range of devices and network conditions.
- Page load times should not exceed 2 seconds under normal conditions.
- System performance should be optimized using lazy loading, asynchronous calls, and efficient database queries to maintain fluid interaction even under high traffic.

### 2.3.4. Accessibility and Usability

As a platform serving a diverse student body, MyGroups must adhere to basic accessibility standards (e.g., WCAG 2.1). This includes support for keyboard navigation, screen readers, and high-contrast UI options. The interface should maintain usability across different screen sizes (responsive design) and be intuitive enough for first-time users to navigate without needing extensive guidance.

## 2.4. Requirement Traceability Matrix

*Disclaimer:*

*As this project does not necessarily integrate with a real Development, Quality Assurance, or other Departments commonly found in industry practices, additional columns such as "Source, Priority, Assigned Department, etc." are not included in this RTM. This Requirement Traceability Matrix section is solely used to demonstrate how RTM could be used within Software Design practicality.*

| ***Major Requirement*** | ***Functional Requirement*** | ***Req. ID*** | ***Success Criteria*** | ***Test Case ID*** | ***Status*** | ***Remarks*** |
|---|---|---|---|---|---|---|
| General Authentication | Users must be able to log in based on their email and passwords, then granted access to features according to their registered role. | GA-01 | The system shall log users in to a valid, registered account based on the account database. | T-GA-01, T-GA-02 | PARTIAL PASS | On T-GA-02, an unregistered email with the same password as another registered email is logged in as the registered email. |
| | | GA-02 | Email and Password registered under "Student" type should be logged in as Student | T-GA-03 | PASS | |
| | | GA-03 | Email and Password registered under "Lecturer" type should be logged in as Lecturer | T-GA-04 | PASS | |
| | | GA-04 | Email and Password registered under | T-GA-05 | PASS | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | “Student” type should be logged in as Staff | | | |
| Profile Management | Students can view their own profile. | PM-01 | Profile page displays correct personal and academic info. | T-PM-01 | PASS | |
| | Students can edit profile details. | PM-01.1 | Updated info is saved and reflected on the profile. | T-PM-02 | PASS | |
| Project Progress | Group members can view project details and timeline. | PP-01 | Timeline and task progress are displayed. | T-PP-01 | PASS | |
| | Group members can edit project progress/tasks . | PP-01.1 | Changes to project details are saved and reflected. | T-PP-02 | PASS | |
| | Members can give peer feedback after milestones. | PP-01.2 | Feedback submission is stored and visible to the group. | T-PP-03 | PASS | |
| Leader Permissions | Group leaders can manage projects and | GL-01 | Leader tools (edit group, monitor members) are | T-GL-01 | PASS | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | members. | | accessible. | | | |
| | Group leaders can submit the project. | GL-01.1 | Project submission is recorded and locked. | T-GL-02 | PASS | |
| | Files can be attached during submission. | GL-01.1.1 | Files are uploaded and previewable. | T-GL-03 | PASS | |
| | Group leaders can request a consultation. | GL-01.2 | Consultation request is sent and visible to lecturer. | T-GL-04 | PASS | |
| | Files can be attached during consultation request. | GL-01.2.1 | Attachment is stored and visible. | T-GL-05 | PASS | |
| | Leaders can approve/reject join requests. | GL-01.3 | Decision is reflected in member list. | T-GL-06 | PASS | |
| | Leaders can request to kick members. | GL-01.4 | Kick request triggers approval process. | T-GL-07 | PASS | |
| Lecturer Features | Lecturer can view submission | LF-01 | Submission is accessible with file list. | T-LF-01 | PASS | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | details. | | | | | |
| | Lecturer can leave responses to submission. | LF-01.1 | Responses are stored and linked to project. | T-LF-02 | PASS | |
| | Lecturer can view attachments in submission. | LF-01.2 | Attachments open correctly. | T-LF-03 | PASS | |
| | Lecturer can give marks. | LF-01.3 | Marks are saved and visible to group. | T-LF-04 | PASS | |
| | Lecturer can view consultation requests. | LF-02 | Request details and attached files are viewable. | T-LF-05 | PASS | |
| | Lecturers can respond to consultation. | LF-02.1 | Response is sent to requesting group. | T-LF-06 | PASS | |
| Staff Controls | Staff can post announcements. | SC-01 | Announcement appears on dashboard. | T-SC-01 | PASS | |
| | Staff can set title and content of announcement. | SC-01.1 | Input fields are editable and stored. | T-SC-02 | PASS | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | Staff can enable/disable features on the platform. | SC-02 | Feature toggle works, status changes system-wide. | T-SC-03 | PASS | |
| | Staff can set notice messages on disabled features. | SC-02.1 | Message appears when trying to access disabled features. | T-SC-04 | PASS | |

### 2.5. System Architecture Overview

The system architecture for **MyGroups** is designed to be a secure, scalable, and high-performance web platform supporting seamless collaboration between students, lecturers, and administrators. It follows a modular, cloud-based architecture that integrates modern front-end technologies, efficient backend logic, and reliable data management systems.

| No. | Component | Description |
|---|---|---|
| 1 | User Browser | The client interface where users interact with the system via web browsers on desktops, laptops, or mobile devices. |
| 2 | Domain Name Servers (DNS) | Resolves the web address (e.g.,mygroups.taylors.edu.my) into the IP address of the web server. |
| 3 | Load Balancer | Distributes incoming user traffic across multiple backend servers to enhance reliability, minimize response time, and support scalability. |
| 4 | Front End (Application) | Developed using **React.js**, the front-end handles user interactions, dynamic content rendering, and real-time UI updates. |
| 5 | Back End (Application) | Built with **Node.js** and **Express.js**, the backend processes user requests, enforces business logic, and interfaces with databases and external services. |
| 6 | Database Layer | Utilizes **MySQL** for structured data (e.g., user info, project data), and **Redis** for caching frequently accessed content. **Google Cloud** |

| | | |
|---|---|---|
| | | **SQL** may be integrated for cloud database services. |
| 7 | Security Services | Implements **JWT (JSON Web Tokens)** for user authentication, **Role-Based Access Control (RBAC)** for permission management, and encryption protocols to ensure data privacy. |
| 8 | Cloud Storage | Used for storing media files and attachments, including project submissions and consultation documents. Supported by **Google Cloud Storage** or **Amazon S3**. |

Additional Technical Details:

- Content Delivery Network (CDN): Tools like Cloudflare or AWS CloudFront will be used to cache and deliver static assets globally for faster load times and reduced latency.
- Monitoring & Observability: Services such as Grafana and AWS CloudWatch will track server health, usage metrics, and system performance in real time.
- API Services: Modular REST APIs will handle Authentication, Profile Management, Group Management, Feedback Systems, Grading, and Consultations.
- Communication Protocols: All data exchanges between users and servers are secured using HTTPS with SSL/TLS encryption.

This system architecture promotes high availability, security, and scalability, essential for delivering an effective and reliable educational platform aligned with NAVIGO's mission of enhancing user experience and maintaining data integrity.

## 2.6. Interface Overview

***Visual & Interaction Consistency***

- The system must maintain a consistent visual identity across all pages, including typography, icon sets, and button styles.
- All roles (Student, Lecturer, Admin) will share a unified layout skeleton but differ in contextual content.
- The navbar, sidebar, and cards should appear in the same style and position across modules for intuitive navigation.

***Color Palette & Harmony***

- Primary colors should align with Taylor's University branding (e.g., dark reds, charcoal greys).
- High contrast maintained for accessibility compliance.

***Responsive Design***

- UI components adapt across screen sizes — ensuring legibility and interaction both on desktops and mobile/tablets.
- Responsive cards, collapsible sections, and flex-based layout are used in design prototypes.

***Accessibility & Inclusivity***

- Fonts must be legible, minimum size 16px.
- Support for keyboard navigation and screen readers should be embedded in development.
- Feedback buttons (scale sliders, checkboxes) designed to be touch-friendly.

## Appendix

### *Appendix A: Activity Diagram*

Activity Diagram

1. **Activity Diagram: System**

This activity diagram outlines the core system-level functionalities that occur regardless of the specific user type (admin, student, etc.). Here's a step-by-step breakdown:

1. **Start**

The system begins operation.

2. **Verify Credentials**

The system authenticates the user's login credentials before allowing access.

3. **Notify Leader of Join Request**

When a student sends a group join request, the system notifies the group leader.

4. **Display Group Progress**

The system visualizes the group’s overall progress (possibly in tasks or project status).

5. **Trigger Notification to Lecturer**

Alerts are sent to the lecturer about key updates, such as submissions or group changes.

6. **Store Project Files**

Project-related files uploaded by students or leaders are stored securely in the system.

7. **Calculate Leadership Points**

The system automatically calculates points based on leadership actions, possibly for gamification or evaluation purposes.

8. **Display Leadership Dashboard**

A dashboard is presented to users with leadership roles to monitor group dynamics and personal contribution metrics.

9. **End**

The system completes the process or awaits further input.

2. **Activity Diagram: System Admin**

This diagram represents the workflow and actions performed by the system administrator to manage and maintain the platform:

1. **Start**

Admin session begins.

2. **Login → Verify Credentials**

Admin logs into the system and their credentials are authenticated.

3. **Select Feature → Select Action (Enable/Disable/Restart)**

The admin chooses a system feature and decides to enable, disable, or reboot it.

4. **Apply Feature Change**

The system applies the chosen action (e.g., enabling or disabling a feature).

5. **Update System Logs**

Every change is logged for future tracking or auditing purposes.

6. **Reboot Set → Confirm Reboot → Restart Feature**

If the reboot option is selected, the system seeks confirmation before executing the restart.

7. **Enable → Confirm Enable → Activate Feature**

If enabling a feature, the system asks for confirmation and then activates it.

8. **Disable → Set Disablement Message → Confirm Action**

If disabling a feature, the admin sets a message and confirms the action.

9. **View System Analytics / Feature Analytics → Display Analytics Chart**

Admin can view visual analytics on system-wide or feature-specific performance.

10. **Logout → End**

The admin logs out, ending the session.

**3. Activity Diagram: Student**

This activity diagram illustrates the user journey of a student using the system:

1. **Start**
2. **Login → Verify Credentials**

The student enters login details and the system verifies them.

3. **Create Profile**

A first-time user sets up their profile.

**4. Search for Group**

The student browses or searches for available project groups.

5. **Request to Join Group**

The student sends a join request to the desired group.

6. **Join Request Accepted?**

A decision point: if the request is accepted, the flow continues; otherwise, it ends.

7. **Track Group Progress**

The student monitors group activities and progress through the dashboard.

8. **Submit Individual Contribution**

Each student submits their personal task or portion of the project.

**9. Rate Members**

Peer evaluation of other group members' performance.

**10. Leave Private Feedback (Optional)**

Optional step to submit confidential feedback regarding the group experience.

**4. Activity Diagram: Student Leader**

This activity diagram outlines the actions taken by a student leader to manage a group and submit the final project through the system:

**1. Start**

The session begins for the student leader.

**2. Login → Verify Credentials**

The leader logs into the platform and is authenticated before gaining access to any features.

**3. Create Profile**

If it's the first login, the student leader creates their personal profile.

**4. Create Group**

The leader establishes a new project group within the system.

**5. Accept/Reject Join Request**

When students request to join the group, the leader reviews and either accepts or rejects their requests.

**6. Manage Members**

The leader manages group composition, including assigning roles or removing members.

**7. Track Group Progress**

The progress of the group's activities and deliverables is monitored using the dashboard.

**8. Book Consultation**

The student leader schedules a consultation session with the lecturer for guidance or project review.

**9. Attach Files**

Project files and documents are uploaded to the system.

**10. Submit Final Project**

The leader officially submits the completed project for evaluation.

**11. View Leadership Dashboard**

A dashboard is displayed showing group performance, task statuses, and leadership-related insights.

**12. End**

The process concludes, marking the end of the leader's activity flow.

## 5. Activity Diagram: Lecturer

This activity diagram illustrates the workflow of a lecturer interacting with the system to monitor and support student project development:

**1. Start**

The lecturer begins their session.

**2. Login → Verify Credentials**

The lecturer logs into the system and their credentials are authenticated before proceeding.

**3. View Project Progress**

Once logged in, the lecturer accesses the project progress dashboard to monitor ongoing student activities and submissions.

**4. Respond to Consultation Request**

If a student group has submitted a request for consultation, the lecturer can review and respond accordingly.

**5. Provide Feedback**

The lecturer offers feedback to guide the student team on their work or clarify specific issues.

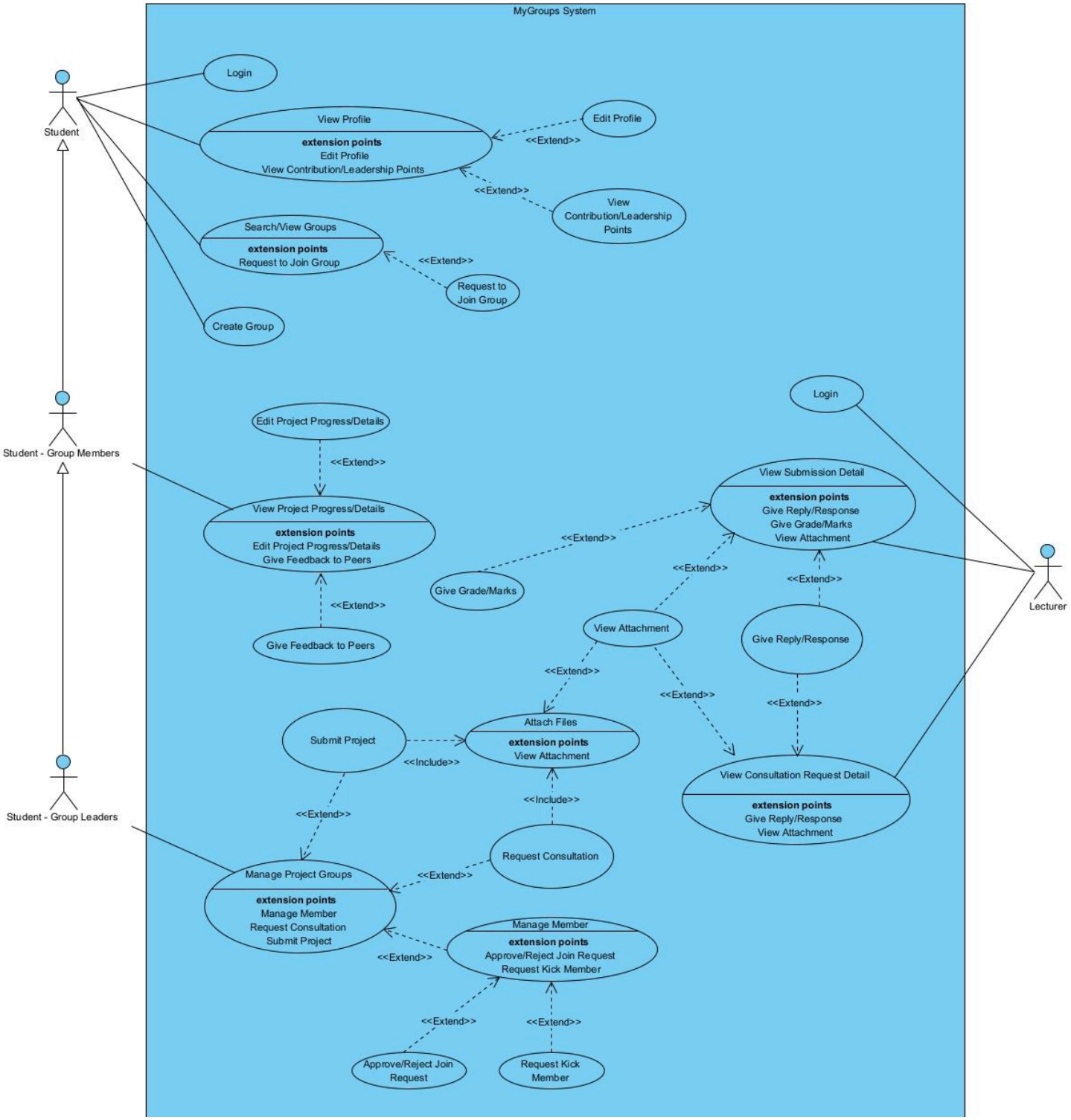
**6. Contact Group**

Direct communication can be initiated by the lecturer to engage with the student group, provide clarification, or give updates.

**7. End**

The lecturer completes the session or exits the platform.

## *Appendix B: Use Case Diagram*



***Appendix C: Architecture Diagram***

Architecture diagram.pdf

***Appendix D: Active In-Development Prototype***

https://www.figma.com/proto/7CctnSe6aCsE6zHgiVGq4n/SoftwareDesign?page-id=0%3A1&node-id=1-3&p=f&viewport=-134%2C-428%2C0.13&t=0NsTVz2rc1HAaUzn-1&scaling=min-zoom&content-scaling=fixed&starting-point-node-id=1%3A3