# CITS3200 Team 15

# Minutes of Meeting 2 held on August 3, 2023

Present: Joshua, Muslim, Genevie, Ryan

Apologies: Jonathan

Absent: None

Meeting Started: 12:50 pm

1.1 Introduction and Overview

The team is introduced to the client, Associate Professor Dianne Hesterman.

The client discussed the topic of prerequisites and how the school of engineering currently publishes study plans to ensure students take the required prerequisite units to graduate within the shortest amount of time.

The client proposes an interactive study plan that will help students better visualize the prerequisites they have to adapt to changes in their study plans.

1.2 Issues with the current system

The current study plans are static pdfs that fail to account for prerequisite requirements and can be unhelpful in circumstances such as part-time students, or unit failures. The client highlighted the need for a more dynamic study plan system that caters to individual needs and circumstances.

2.1 Client's Project Description

The client presented their vision for the desired application, highlighting the following features:

Course Management:

1. Staff should be able to add new courses.
2. Staff should be able to update course requirements.
3. Staff should be able to upload new units and study plans.
4. Consideration for course changes.

Study Plan Generation:

1. Display study plans based on student's course start date.
2. Display alternate new units for students that started in previous years with altered study plans.
3. Restrict new students from accessing old units and old study plans.
4. Study plan changes to find the fastest route to completion based on completed units.

Student Interaction:

1. Students can click on completed units.
2. Show available units based on completed courses.
3. Students can create their own study plan.
4. Students can see the order of unit completion.
5. Input degree and ATAR requirements.
6. Aesthetically resemble a study plan.
7. PDF export feature for saving study plans. \*
8. Drag-and-drop functionality.
9. Easy-to-use interface for students.

System Features:

1. Study plan effects visible after changes.
2. Ability to select units worth 6 points.
3. Allow multiple unit selections. \*
4. Highlight important pathways and unit prerequisites.
5. Help students navigate their study plan.

Items with the \* have been noted as not necessary but nice to have.

The initial focus should be on a few engineering majors, including at least mechanical and chemical engineering as they are the most complex engineering majors to integrate into a system. The project should begin with the framework and undergo a trial phase.

It is important to consider that:

1. In some cases, students only require a specific point total to fulfill prerequisites. (e.g., 12 points of programming.)
2. The study plan will be based on student input and the team will not have access to any records.
3. The application is a website.
4. The client believes it is unnecessary to store study plans in a database.
5. Only the units for each major need to be stored by the application.
6. Client interested in determining study plan solely on prerequisite ordering.
7. In certain instances, it might be implied that a student should be in their third year before attempting a particular third-year unit, even if it's not explicitly stated as a prerequisite. Solutions should be explored.
8. Explore the option of introducing bridging units for cases where students don't meet the minimum ATAR requirements.

The Client stated, if a student goes part time, the study plan should adapt to the units they have done, so they can see in what order they need to do their units to finish in the minimum amount of time.

2.2 Team queries and further discussion

Genevie inquired about the process of acquiring curriculum information and prerequisites for the software. The client explained that a curriculum management system generates PDFs with unit information, including prerequisites.

Joshua asked if there was a way to get the units and their requirements on a spreadsheet, so it could be uploaded by staff if they wanted to make updates or add a new course. The client said she’ll see what she can do. He proposed a method where staff members could upload PDFs or Excel spreadsheets with unit names and prerequisites.

Muslim added that the software could work as a tool to ensure continuity in course information and updates.

Joshua proposed staff members could upload new course outlines or changes in prerequisites, and the program could adapt accordingly to ensure the study plans that finish the course in the least amount of time, based on unit prerequisites. The client was extremely interested in the suggestion.

Genevie suggested a tree-like structure to visualize the prerequisites, but Client believes it will be hard to read, unless highlighting is instead of arrows to signify prerequisites.

Muslim suggested a tree structure and a study plan but has concerns he may be over complicating it.

Muslim suggested downloading the study plan as a PDF for students. The client liked the idea.

Client stated it doesn’t matter how the prerequisites are visualized, so long as it’s easy for students to understand.

2.3 Members Asked Variety of Questions

Genevie questioned how the software would handle discontinued majors, units, or changes in courses. The client acknowledged that this is a consideration, especially in light of units being replaced or changed mid-course. Muslim explored and discussed different possibilities.

Muslim asked if study plans should persist. She stated a system where students could save their plans for future reference isn’t necessary but can be implemented if it is believed to be of value.

Muslim asked how the students ought to be notified if their created study plan on the application is invalid. Client stated it can either be left to the student to figure out what they did wrong, but it would be better to signify what they did wrong.

Joshua suggested notifying students using a debugging string.

Joshua asked if we are allowing the students to create their own study plan. The client confirmed this is the case.

Muslim asked if she wanted weekly updates, she said every 2-3 weeks, if there is something to showcase.

When asked if this is solely for engineering students, she stated, if this application is successful, it may be implemented in other majors. For now, the application will be mainly focused on the Bachelor of Engineering majors with integrated honours. The hardest of which to implement are Mechanical and Chemical specializations.

The client stated she’s happy either with email or face-to-face if more information is needed.

3. Meeting ended

The meeting went overtime due to extensive project-related discussions. The meeting closed to allow for the Client to attend another meeting scheduled at 2pm.

Meeting Closed: 2:10 pm

Action Items:

|  |  |
| --- | --- |
| All | * Update weekly deliverables in the shared spreadsheet. * A risk register will be prepared to address potential project risks. * Work on improving communication and asking for help when needed. * Research and decide on the technology stack for frontend and backend development. * Team members to contribute and collaborate on GitHub using separate branches. * Seek clarification from Client regarding the project specifications and get approval for our design ideas. * Upload team documentation on GitHub. |
| Muslim | * Begin Flask development, create welcome page, and options page. * Send Client the requirements document for her approval. * Follow up with Client about the availability of Excel spreadsheets containing unit and prerequisite information. |
| Ryan | * Complete Project requirements document for the project. Research how to write a requirements document Assign monetary value for each requirement based on client desires, send to Muslim send it to the Client for her approval. |
| Jonathan | * Investigate the necessity of login functionality and API login for Microsoft. (CANCELLED for now as Client stated login not necessary requirement). * Organize files on teams for now. |
| Genevie | * Follow up on Code of Conduct. * Research databases for the project. |
| Josh | * Complete this week’s deliverables. * Implement dynamic table in JS or React. Implement drag and drop functionality for the cells. |