



# Welcome to UniCollab!

## Attack on HD | Final Report

Tiana Douroudis | Fintan O'Shea | Xiaoyan Xie (Christina) | Xiang Ji (James) | Kanishka G Yamani (Kenny) | Haochen Shi (Aiden)

### CONTENTS

<b>Business and Requirement Analysis</b>	<b>4</b>
Purpose of UniCollab	4
Problems UniCollab Addresses	4
UniCollab Features	6
Updated User Stories	13
High Priority: Unique to UniCollab	13
Problem Statement 1:	13
Requirement 1: (Use Case: Group Leader)	13
Feature 1 - Create a group within a UniCollab room	13
Feature 2.1 - Search for users in room based on availabilities	13
Feature 2.2 - Search for users in room based on distance	14
Feature 2.3 - Search for users in room based on skillset compatibility	14
Feature 2.4 - Search for users in a room based on all 3 criteria	15
Feature 3 - Invite users to group	15
Feature 4 - Update my group's information	16
Feature 5 - Remove members from the group	16
Requirement 2: (Use Case: Groupless Room Member)	16
Feature 1.1 - Search for groups in a room based on availabilities	16
Feature 1.2 - Search for groups in a room based on location	17
Feature 1.3 - Search for groups in a room based on skillset	17
Feature 1.4 - Search for groups in a room based on all 3 criteria	17
Feature 2 - Users may request to join a group	18
Problem Statement 2:	18
Requirement 1: (Use Case: Any user)	18
Feature 1 - Sync UniCollab Calendar with Google Calendar	18
Feature 2 - Enter a preferred meeting location to profile	19
Problem Statement 3	19
Requirement 1: (Use Case: Course Administrator)	19
Feature 1 - Create a Unicollab room	19



Requirement 2: (Use Case: Any user)	20
Feature 1: Users can access rooms and groups they are in	20
Feature 2: Students can search for existing rooms	20
Low Priority: Additional Features which enhance the UniCollab experience	21
Problem Statement 1 (Additional Features)	21
Requirement 3: (Use Case: Group member)	21
Feature 1: Group member can leave a group	21
Problem Statement 2 (Additional Features)	21
Requirement 1: (Use Case: Any user)	21
Feature 3 - Enter courses taken	21
Requirement 2	22
Feature 1: Students can update their profile page biography	22
Feature 2: Students can update their profile photo	22
Feature 3: Students can link external sources of Social Media	23
Requirement 3	23
Feature 1: Able to navigate another user's profile from any instance they appear on the UniCollab site to view information about them	23
Problem Statement 3 (Additional Features)	23
Requirement 1	23
Feature 3: Joining UniCollab rooms	24
Feature 4: Leaving UniCollab rooms	24
Requirement 2	24
Feature 1: Students can sign up for a UniCollab account	24
Feature 2: Students can log in to their UniCollab account	25
Feature 3: Students can log out of their UniCollab account	25
Problem Statement 4	25
Requirement 1	25
Feature 1: Group leaders are notified when a user requests to join their group	25
Feature 2: Group leaders can accept or decline requests	26
Feature 3: Students are notified that their request is accepted	26
Feature 4: Students are notified that their request is declined	26
Requirement 2	27
Feature 1: Users receive a notification if a UniCollab group leader has sent an invitation to join their group	27
Feature 2: Users can accept or decline invitations sent by group leaders	27
Feature 3: When a user has responded to an invitation to join a group, the group leader who sent the invitation is notified of the outcome	27
Requirement 3	28
Feature 1: Students receive notifications when group chats possess messages that a user has not yet viewed	28



<b>Requirement 4</b>	<b>28</b>
Feature 1: A red circle appears above the notification icon for a student who has received an unviewed notification	28
Feature 2: Students may view notifications	29
Achieved Components: User Interface	29
<b>Design and Code Inspection</b>	<b>35</b>
Final Software Architecture	35
Sequence Diagrams for each use case	38
Other relevant information and key technologies used	45
Key benefits of design/implementation	47
<b>Team Organisation + Appraisal of Work</b>	<b>48</b>
Responsibilities of the Team	48
How did the project go?	50
Issues and Problems Encountered	51
What would be done differently?	52
<b>Peer Group Assessment Form</b>	<b>54</b>

---



# 1. Business and Requirement Analysis

## Purpose of UniCollab

UniCollab is a web application aiming to simplify and centralise the process of group formation for UNSW Computer Science and Engineering students. UniCollab possesses all of the features that students require to not only form groups that match their needs, but also interact and communicate with team mates once a group has been made, hence reducing the need for external applications and providing students with a centralised portal for UNSW CSE groups.

## Problems UniCollab Addresses

- 1. Current alternatives of university group making utilise different features of multiple platforms, which is ineffective and delocalised.**  
UniCollab centralises all forms of collaboration and communication, so that users do not need to resort to foreign platforms to form groups. Not only this, but once groups are made, they must currently resort to alternative platforms to continue their form of communication and project development, since WebCMS lacks a communication tool. Whether it be via Facebook, Discord, Wechat, or another application, students currently must form accounts on other platforms to interact with their group mates. Additionally, users have no way to communicate with other students doing the same course or project. To communicate with other students participating in similar projects, UNSW students must make a forum post on WebCMS or send a message to a Facebook group chat. Often, group chats do not exist among cohorts of a course, and it may be difficult to communicate with other students via a WebCMS forum post.
- 2. Current methods of group making for CSE students are often randomised, often leading to ineffective groups**

Take this course, SENG2021 for example. There are approximately 112 students and yet only 3 people made posts reaching out to others on the WebCMS forums page. The rest were randomly allocated, completely disregarding the synergy between group members. Even within our group alone we had conflicting schedules and members from other parts of the world. To solve this problem, UniCollab is designed to facilitate the group formation process for UNSW CSE students by taking into consideration three key pieces of information:

1. The distance between a user's inputted preferred meeting location and a group's preferred meeting location (which is inherited from the group leader)
2. A user's calendar availability matched against a group's preferred meeting times
3. The skills a group currently has and is looking for in other members

Not only will UniCollab allow students to filter groups based on highly effective algorithms for these 3 criteria, but also allows for an overall compatibility search. Group leaders will also be able to search for members to invite to their group based on these algorithms. All filtered results will be ranked based on a score / distance, as well as colour coded for user simplicity.



**3. Current alternatives of university group making lack the ability to discover more about potential group candidates, forcing groups to be made based upon insubstantial information**

For example, on WebCMS, each student has an incredibly basic profile page, consisting of their name, profile image, and student email address. This lacks any personal information about a user, hence other users do not gain much insight on another's personal interests, university involvement or potential assets to a group. To solve this issue, UniCollab provides each user with a profile page that contains multiple fields where students can display their personal information, in order to advertise their interests and personality. They may edit their profile image, biography, completed courses, interests and their profile will also display the current groups that they are a member of. This allows other users to gain insight into a potential group member, and make an informed decision towards the group making process.

**4. Current methods of group making lack the ability to communicate personal information such as location and preferred meeting times, which may impede group communication and efficiency**

Due to the random allocation of group members to groups that is currently widely implemented, students often struggle to agree upon preferred meeting times and locations, depending on their personal, work and other commitments. Our group for example, did not know of each other's availability or even global location when we formed our group, which has caused difficulty in planning group meetings. UniCollab solves this issue, allowing users to interact with Google Maps to enter a postcode/suburb location, and Google Calendar to sync their personal calendar. Each of these pieces of information will be used by UniCollab behind the scenes to generate recommended groups for users, based on their availability to join meetings, and/or the distance between themselves and a group's meeting locations. Before joining a group, users can browse the many options that UniCollab offers them, taking time to consider a group's meeting times and location before making any decisions.

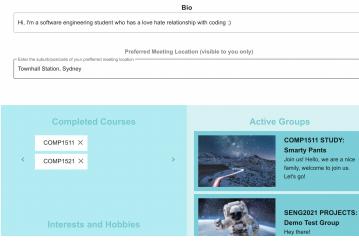
**5. It requires a lot of time and numerous messages from students for course coordinators to allocate students to groups**

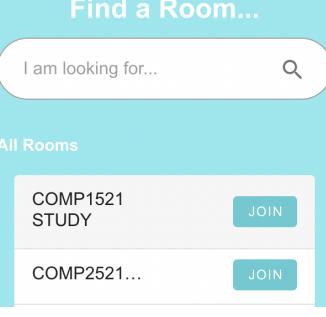
Currently, there are over 5000 students in CSESoc. On average, each student will take four team-based courses each throughout their UNSW careers. That is 20 000 potential unique usages for UniCollab for the current cohort alone. This only increases when one thinks about the future CSE students, and potentially the scalability of expanding to include additional majors and even universities. Head lecturers often take time to respond to individual student messages and manually sort them into teams where any availability is present. UniCollab significantly simplifies this process, allowing for students to form groups themselves and leave course coordinators to focus on other tasks.

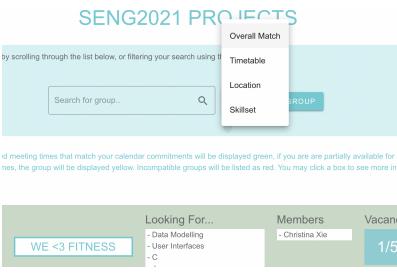
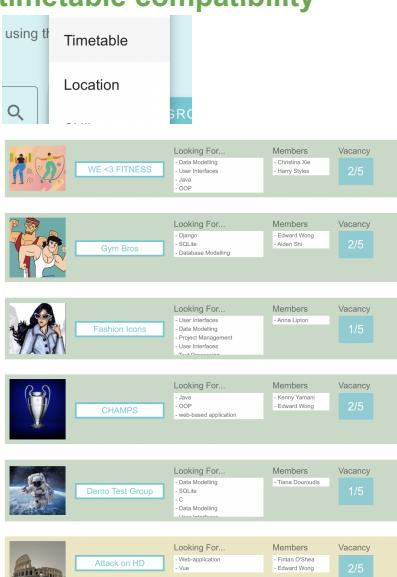
---

## UniCollab Features

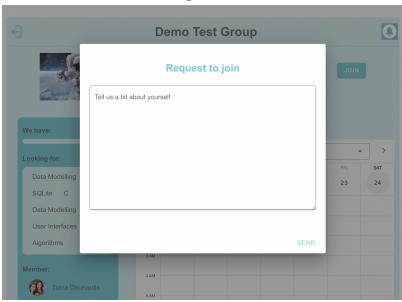
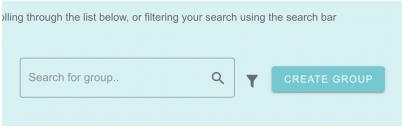
Note: **green** features indicate components we have successfully achieved in our final application, **yellow** indicates partial implementation, **red** features were deemed not essential to the key purpose of UniCollab and were therefore not implemented

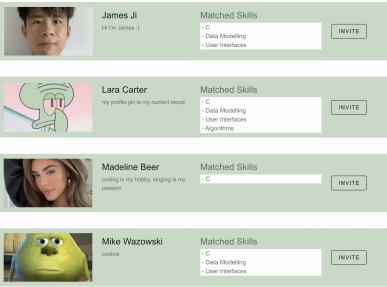
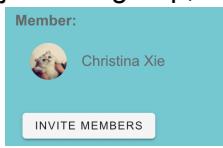
Feature	Description
<b>1. Register</b> Sign-up to start collaborating 	<ul style="list-style-type: none"> <li>- User enters full name, email and password to create a UniCollab account</li> </ul>
<b>2. Login</b> 	<ul style="list-style-type: none"> <li>- User can enter their username (email) and password to login to UniCollab - error thrown if incorrect username or password</li> </ul>
<b>3. Logout</b>  	<ul style="list-style-type: none"> <li>- Clicking the logout button from the user's main dashboard page will logout the user and redirect them to welcome page</li> </ul>
<b>4. Sign in with social media</b>   	<ul style="list-style-type: none"> <li>- Allows users personal details including email, location, bio, profile photo to be uploaded to their UniCollab profile</li> </ul>
<b>5. View profile</b>  	<ul style="list-style-type: none"> <li>- Clicking profile photo in top right corner of any page will allow you to view your own profile including sections for bio, interests + hobbies, completed courses, active groups (groups you are a member/owner of), Preferred Meeting Location + Calendar button (view your UniCollab calendar)</li> </ul>
<b>6. Edit all publicly viewable profile information</b> 	<ul style="list-style-type: none"> <li>- Once click 'Edit Profile' button:  </li> <li>- Able to enter bio (functional), interests + hobbies, completed courses (not functional)</li> <li>- Add courses was implemented on back-end but not connected to frontend UI (based on name of course entered function would loop through all courses in</li> </ul>

	<p>UniCollab database and attach that course object to the student)</p>
<b>7. Sync Google Calendar</b> 	<ul style="list-style-type: none"> <li>- Able to click 'Sync' button on calendar view which will ask you to authorise your Google account and then will upload all your google calendar events (for next 10 weeks - duration of a typical UNSW term)</li> <li>- Able to resync calendar later on by clicking 'Sync' again, which will delete all current calendar events and upload new ones</li> </ul>
<b>8. Enter Preferred Meeting Location</b> 	<ul style="list-style-type: none"> <li>- Clicking edit profile button will allow user to enter suburb of their preferred meeting location for group matching</li> </ul>
<b>9. View Rooms and Groups user is in from their main dashboard</b> 	<ul style="list-style-type: none"> <li>- Main Dashboard page contains 'My Rooms' displaying all rooms the user has joined</li> <li>- 'My Groups' section contains any groups the user has joined and is a member of (either through joining themselves or accepting an invite)</li> <li>- These sections include horizontal scrolling for if the rooms/groups do not fit across the screen</li> <li>- The items also dynamically update as the user leaves or joins a room/group</li> </ul>
<b>10. View a list of all rooms the user is not in</b> 	<ul style="list-style-type: none"> <li>- Able to see a list of all UniCollab rooms - this list only displays the rooms the user is not in (note see no. 9 screenshot, rooms in My Rooms section are not contained in All Rooms list in screenshot to the left)</li> <li>- Also able to use search bar to find a room relating to a particular subject (not implemented)</li> </ul>
<b>11. Create Room (Course Admin only feature)</b> 	<ul style="list-style-type: none"> <li>- Course Admins will have a separate interface with the ability to create a room and enter a description to direct their students to join the room</li> <li>- The image to the left shows that the button is disabled for students</li> <li>- The course admin will decide on the skills groups need (automatically populate the 'Looking For' skills section for newly created groups) and the capacity of</li> </ul>

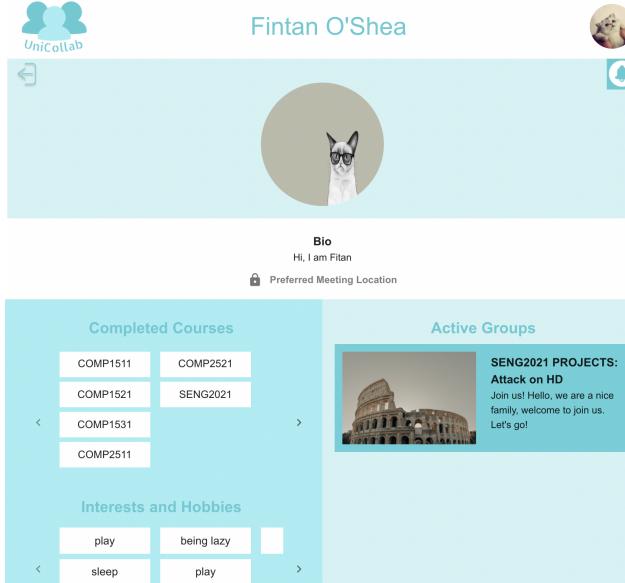
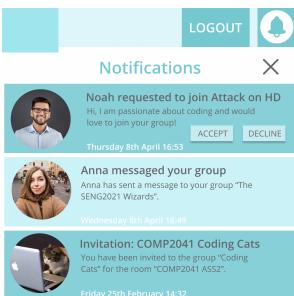
	<p>the groups (e.g. for this subject would be 5)</p>
<b>12. Join Room</b> 	<ul style="list-style-type: none"> <li>- Clicking the 'Join' button next to the room name in the list of all rooms will display a popup with the description of the room, then take them to the room page once they click 'Join' again (student will become a member of the room)</li> </ul>
<b>13. Search groups by overall compatibility</b> 	<ul style="list-style-type: none"> <li>- Clicking the funnel icon and selecting <b>Overall Match</b> will display a list of groups ranked by combination of how well they mesh with your preferred meeting times (<b>timetable</b>), preferred meeting locations (<b>location</b>) and skill compatibility (<b>skillset</b>).</li> <li>- The groups will be sorted from top to bottom, colour coded green → yellow → red (best overall match → worst overall match)</li> <li>- The algorithm equally weighs all 3 factors. How it calculates each factor will be explained below in Feature 14, 15, 16.</li> </ul> <p>Note: as you will see below the list of groups changes ordering and colouring based on what filter has been applied</p>
<b>14. Search groups by timetable compatibility</b> 	<ul style="list-style-type: none"> <li>- Selecting Timetable will get events of a students UniCollab calendar (via their Google Calendar API) and compare them with the groups preferred meeting times (put forward by group leader)</li> <li>- A percentage is calculated based on the number of meeting times the student is available for and the groups are colour coded from top to bottom: <ul style="list-style-type: none"> <li>- <b>Green:</b> A student is available for 90% or more of a group's meeting times.</li> <li>- <b>Yellow:</b> A student is available for less than 90% of the group's preferred meeting times</li> <li>- <b>Red:</b> A student is available for less than 60% of the group's preferred meeting times</li> </ul> </li> </ul> <p><b>availability.py</b> (glimpse into matching algorithm)</p> <pre> percentageAvailable = nAvailable / nMeetings  matchedScore = 0 if percentageAvailable &gt;= 0.6:     matchedScore = 1 if percentageAvailable &gt;= 0.9:     matchedScore = 2  user['score'] = percentageAvailable user['match'] = matchedScore sortedUsers.append(user) </pre>

<h3>15. Search groups by location compatibility</h3>  <table border="1"> <thead> <tr> <th>Group Name</th> <th>Skills</th> <th>Members</th> <th>Vacancy</th> </tr> </thead> <tbody> <tr> <td>Fashion Icons</td> <td>User Interfaces, Data Modeling, Project Management, User Interfaces</td> <td>Anna Linton</td> <td>1/5</td> </tr> <tr> <td>Gym Bros</td> <td>Java, SQL, Database Modeling</td> <td>Edward Wong</td> <td>2/5</td> </tr> <tr> <td>CHAMPS</td> <td>Java, GOF, web-based application</td> <td>Kenny Yamani, Edward Wong</td> <td>2/5</td> </tr> <tr> <td>Attack on HD</td> <td>Web-application, Vue</td> <td>Firan O'Shea, Edward Wong</td> <td>2/5</td> </tr> <tr> <td>WE &lt;3 FITNESS</td> <td>Data Modeling, User Interfaces, Java, GOF</td> <td>Christina Xie, Harry Styles</td> <td>2/5</td> </tr> <tr> <td>Demo Test Group</td> <td>Data Modeling, SQL, C, Data Modeling</td> <td>Tara Dounouds</td> <td>1/5</td> </tr> </tbody> </table>	Group Name	Skills	Members	Vacancy	Fashion Icons	User Interfaces, Data Modeling, Project Management, User Interfaces	Anna Linton	1/5	Gym Bros	Java, SQL, Database Modeling	Edward Wong	2/5	CHAMPS	Java, GOF, web-based application	Kenny Yamani, Edward Wong	2/5	Attack on HD	Web-application, Vue	Firan O'Shea, Edward Wong	2/5	WE <3 FITNESS	Data Modeling, User Interfaces, Java, GOF	Christina Xie, Harry Styles	2/5	Demo Test Group	Data Modeling, SQL, C, Data Modeling	Tara Dounouds	1/5	<ul style="list-style-type: none"> <li>- Selecting the <b>Location</b> filter will compare the student's preferred meeting location to the preferred meeting location of each group (inherited from the group leader)</li> <li>- A distance between the two locations is calculated using the Google Maps API and the groups are ranked in this order colour coded from top to bottom:       <ul style="list-style-type: none"> <li>- <b>Green:</b> A group's preferred meeting location is within 10km of the student's preferred meeting location</li> <li>- <b>Yellow:</b> A group's preferred meeting location is within 30km of the student's preferred meeting location</li> <li>- <b>Red:</b> A group's preferred meeting location is more than 30km from the student's preferred meeting location</li> </ul> </li> <li>- See <b>meeting.py</b> for algorithm</li> </ul>
Group Name	Skills	Members	Vacancy																										
Fashion Icons	User Interfaces, Data Modeling, Project Management, User Interfaces	Anna Linton	1/5																										
Gym Bros	Java, SQL, Database Modeling	Edward Wong	2/5																										
CHAMPS	Java, GOF, web-based application	Kenny Yamani, Edward Wong	2/5																										
Attack on HD	Web-application, Vue	Firan O'Shea, Edward Wong	2/5																										
WE <3 FITNESS	Data Modeling, User Interfaces, Java, GOF	Christina Xie, Harry Styles	2/5																										
Demo Test Group	Data Modeling, SQL, C, Data Modeling	Tara Dounouds	1/5																										
<h3>16. Search groups by skill set compatibility</h3>  <table border="1"> <thead> <tr> <th>Group Name</th> <th>Skills</th> <th>Members</th> <th>Vacancy</th> </tr> </thead> <tbody> <tr> <td>CHAMPS</td> <td>Java, GOF, web-based application</td> <td>Kenny Yamani, Edward Wong</td> <td>2/5</td> </tr> <tr> <td>Attack on HD</td> <td>Web-application, Vue</td> <td>Firan O'Shea, Edward Wong</td> <td>2/5</td> </tr> <tr> <td>WE &lt;3 FITNESS</td> <td>Data Modeling, User Interfaces, Java, GOF</td> <td>Christina Xie, Harry Styles</td> <td>2/5</td> </tr> <tr> <td>Fashion Icons</td> <td>User Interfaces, Java, C, Project Management, User Interfaces</td> <td>Anna Linton</td> <td>1/5</td> </tr> <tr> <td>Demo Test Group</td> <td>Data Modeling, SQL, C, Data Modeling</td> <td>Tara Dounouds</td> <td>1/5</td> </tr> <tr> <td>Gym Bros</td> <td>Django, SQL, Database Modeling</td> <td>Edward Wong, Aiden Shi</td> <td>2/5</td> </tr> </tbody> </table>	Group Name	Skills	Members	Vacancy	CHAMPS	Java, GOF, web-based application	Kenny Yamani, Edward Wong	2/5	Attack on HD	Web-application, Vue	Firan O'Shea, Edward Wong	2/5	WE <3 FITNESS	Data Modeling, User Interfaces, Java, GOF	Christina Xie, Harry Styles	2/5	Fashion Icons	User Interfaces, Java, C, Project Management, User Interfaces	Anna Linton	1/5	Demo Test Group	Data Modeling, SQL, C, Data Modeling	Tara Dounouds	1/5	Gym Bros	Django, SQL, Database Modeling	Edward Wong, Aiden Shi	2/5	<ul style="list-style-type: none"> <li>- Selecting the <b>Skillset</b> filter will match a group's "Looking For" skills (skills the group is looking for) and "We Have" skills (skills possessed by members of the group) to the skills a student has.</li> <li>- The skills a student has is determined by examining the courses a student has taken and finding matching keywords, scraped from WebCMS course outlines</li> <li>- A synergy score is calculated based on 2 factors:       <ul style="list-style-type: none"> <li>- If a student possesses the skills a group is looking for (how much they are needed by that group) <b>Currently weighted 1</b></li> <li>- If a student possesses the skills a group currently has (how much they can collaborate with that group) <b>Currently weighted 0.5</b></li> </ul> </li> <li>- The more a skill appears in the student's courses (e.g. if the word <b>Python</b> appears in 3 of their courses), the more they will be deemed to be capable at this skill and the more they will be matched with groups that are looking for e.g. Python</li> <li>- The colour of the group is based on synergy score (see <b>skills.py</b> for more about our algorithm) and is again, sorted from top to bottom in order of green → yellow → red</li> </ul>
Group Name	Skills	Members	Vacancy																										
CHAMPS	Java, GOF, web-based application	Kenny Yamani, Edward Wong	2/5																										
Attack on HD	Web-application, Vue	Firan O'Shea, Edward Wong	2/5																										
WE <3 FITNESS	Data Modeling, User Interfaces, Java, GOF	Christina Xie, Harry Styles	2/5																										
Fashion Icons	User Interfaces, Java, C, Project Management, User Interfaces	Anna Linton	1/5																										
Demo Test Group	Data Modeling, SQL, C, Data Modeling	Tara Dounouds	1/5																										
Gym Bros	Django, SQL, Database Modeling	Edward Wong, Aiden Shi	2/5																										
<h3>17. View Group page</h3>	<ul style="list-style-type: none"> <li>- Clicking on a group's name will direct the user to that group's page. The user will be able to view the following information about the group:</li> <li>- <b>About Us</b> → group leader's description of group</li> <li>- <b>We have skills</b> → populated as members join the</li> </ul>																												

	<p>group from groups initial ‘Looking for’ skills list</p> <ul style="list-style-type: none"> <li>- <b>Looking for</b> skills → inherited from the room             <ul style="list-style-type: none"> <li>- Group leader is able to add/remove skills (non-currently implemented)</li> </ul> </li> <li>- <b>Member</b> → lists all members of the group, clicking their profile icon will take the user to their publicly viewable profile page</li> <li>- <b>Preferred Meeting Times</b> → Meeting times put forward by the group leader</li> </ul>
<h3>18. Join Group</h3> 	<ul style="list-style-type: none"> <li>- When a student clicks the join button on a group page, a popup will appear prompting them to enter in a short message for the group leader. (see 27. Notifications).</li> <li>- Joining not only adds the student to the member section, but updates the group’s skills by removing the skills the new member has from the Looking for section (based on the courses they have completed using our custom web scraper API) and inserting them to the groups ‘We have section’</li> <li>- We will model a specific example of this in <u><a href="#">Achieved Components</a></u> section below</li> </ul>
<h3>19. Leave Group</h3> 	<ul style="list-style-type: none"> <li>- When the leave button is clicked, the member is removed from the group. This will allow them to be ‘groupless’ again and the group’s skills section will update based on the skills they have lost. See <u><a href="#">Achieved Components</a></u> section below for example</li> </ul>
<h3>20. Create Group</h3> <p>COMP1531 PROJECTS</p> 	<ul style="list-style-type: none"> <li>- When the <b>Create Group</b> button is clicked, the student will be prompted to enter a name for the group and the group will be created</li> <li>- The <b>Looking For</b> skills section will automatically populate based on the room the group has been created in - see no 11. Create Room</li> <li>- The <b>Capacity</b> will also be automatically set based on the room</li> <li>- The group creator will also be added to the member list</li> </ul>
<h3>21. Group leader can edit group information - description, put forward preferred meeting times, add/remove skills</h3> 	<ul style="list-style-type: none"> <li>- Through Clicking the <b>Settings</b> button the user will be able to edit the ‘About Us’ field and click save</li> </ul> <div style="border: 1px solid #ccc; padding: 10px; width: fit-content;"> <p>About us</p> <p>This group is to demonstrate the features of UniCollab for our 1</p> <p style="text-align: right;"><b>SAVE</b></p> </div>

	<p><b>About us</b></p> <p>This group is to demonstrate the features of UniCollab for our final report</p> <p><b>SETTING</b></p> <ul style="list-style-type: none"> <li>- Adding preferred meeting times and adding or removing skills from either skills section has not been implemented</li> </ul>
<p><b>22. Group leader can view a list of groupless members in and their matched skills with the group</b></p> 	<ul style="list-style-type: none"> <li>- When the group leader clicks the <b>invite members</b> button below the member section within their group page, they will be taken to a page with a searchable list of members that are in the room but have not yet joined a group, as seen to the left</li> </ul>  <ul style="list-style-type: none"> <li>- This list contains the students profile picture, name, bio and matched skills which are found by searching for the Looking For skills of the group in the courses the students have completed (see <b>matched_skills.py</b>)</li> <li>- e.g. Madeline Beer has only completed COMP1511 whilst Lara Carter has completed COMP1511, COMP1531 and COMP1521 so has more skills</li> </ul>
<p><b>23. Group leader can search members by overall, location and skill set compatibility</b></p> 	<ul style="list-style-type: none"> <li>- We have coded algorithms to sort a list of users based on timetable, location, skillset and overall with the information of the group creators group (these can be seen in the <b>availability.py</b>, <b>meeting.py</b>, <b>skills.py</b> and <b>overall.py</b> files in separate functions to the group search)</li> <li>- As these algorithms were implemented and connected with the system for group searching (the main functionality of our system) we didn't implement it for user searching</li> </ul>
<p><b>24. Group Leader can invite members to the group</b></p> 	<ul style="list-style-type: none"> <li>- When clicking <b>Invite</b> the student is invited to join the group (see no 27. Notifications)</li> <li>- Our current implementation joins them to the group instantly, performing same processes as No. 18</li> </ul>
<p><b>25. View other users profile pages</b></p>	<ul style="list-style-type: none"> <li>- Clicking a users profile image from anywhere on UniCollab e.g. from Member section of a group's page will take you to their publicly viewable profile:</li> </ul>



<p><b>Member:</b></p>  <p>Fintan O'Shea</p>  <p>Edward Wong</p>	 <p><b>Fintan O'Shea</b></p> <p>Bio Hi, I am Fintan</p> <p>Preferred Meeting Location</p> <p>Completed Courses</p> <table border="1"> <tr><td>COMP1511</td><td>COMP2521</td></tr> <tr><td>COMP1521</td><td>SENG2021</td></tr> <tr><td>COMP1531</td><td></td></tr> <tr><td>COMP2511</td><td></td></tr> </table> <p>Active Groups</p> <p>SENG2021 PROJECTS: Attack on HD Join us! Hello, we are a nice family, welcome to join us. Let's go!</p> <p>Interests and Hobbies</p> <table border="1"> <tr><td>play</td><td>being lazy</td></tr> <tr><td>sleep</td><td>play</td></tr> </table>	COMP1511	COMP2521	COMP1521	SENG2021	COMP1531		COMP2511		play	being lazy	sleep	play				
COMP1511	COMP2521																
COMP1521	SENG2021																
COMP1531																	
COMP2511																	
play	being lazy																
sleep	play																
<p><b>26. Can navigate to main dashboard page at any time</b></p>  <p>Find a user that is currently groupless by scrolling through the list below search using the search bar filter tool</p> <p>Search for User... <input type="text"/></p>	<ul style="list-style-type: none"> <li>- Clicking the logo from any page within UniCollab will take the user to their personal main dashboard page</li> </ul>  <p>welcome Christina Xie</p> <p>LOGOUT</p> <p>Find a Room... I am looking for... <input type="text"/></p> <p>All Rooms</p> <table border="1"> <tr><td>COMP1521...</td><td>JOIN</td></tr> <tr><td>COMP2521...</td><td>JOIN</td></tr> <tr><td>COMP6080...</td><td>JOIN</td></tr> </table> <p>My Rooms</p> <table border="1"> <tr><td>SENG2021 PROJECTS</td><td>LEAVE</td></tr> <tr><td>COMP1511 STUDY</td><td>LEAVE</td></tr> <tr><td>COMP1531 PROJECTS</td><td>LEAVE</td></tr> <tr><td>COMP2511 PROJECTS</td><td>LEAVE</td></tr> <tr><td>COMP3331 LABS</td><td>LEAVE</td></tr> </table> <p>My Groups</p>	COMP1521...	JOIN	COMP2521...	JOIN	COMP6080...	JOIN	SENG2021 PROJECTS	LEAVE	COMP1511 STUDY	LEAVE	COMP1531 PROJECTS	LEAVE	COMP2511 PROJECTS	LEAVE	COMP3331 LABS	LEAVE
COMP1521...	JOIN																
COMP2521...	JOIN																
COMP6080...	JOIN																
SENG2021 PROJECTS	LEAVE																
COMP1511 STUDY	LEAVE																
COMP1531 PROJECTS	LEAVE																
COMP2511 PROJECTS	LEAVE																
COMP3331 LABS	LEAVE																
<p><b>27. Notifications</b></p>  <p>Noah requested to join Attack on HD Hi, I am passionate about coding and would love to join your group! ACCEPT DECLINE Thursday 8th April 16:53</p> <p>Anna messaged your group Anna has sent a message to your group "The SENG2021 Wizards". Wednesday 7th April 16:49</p> <p>Invitation: COMP2041 Coding Cats You have been invited to the group "Coding Cats" for the room "COMP2041 ASS2". Friday 25th February 14:32</p>	<ul style="list-style-type: none"> <li>- When a student clicks the <b>Join</b> button and enters in a message in the popup, a join request will appear in the group leader's notifications with the ability to accept or decline the request. The student will be notified of the group leaders decision</li> <li>- When a group leader invites a student to join a group, the student receives a notification that allows them to navigate to the group page and decide if they would like to join the group. They can similarly accept or decline the invite request → group leader notified</li> </ul>																
<p><b>28. Group Chat</b></p>  <p>ROOM: SENG2021 PROJECTS</p> <p>CHAT - Attack on HD</p> <p>Harry: Hey, hey, hey... I'm having fun in Tokyo! Send me some pictures.</p>	<ul style="list-style-type: none"> <li>- From within a group page there is a chat icon that will navigate group members to a chat page that allows them to talk to their new group members!</li> <li>- This was not implemented as we prioritised the main problem of solving the group formation process and many existing chat platforms already exist</li> </ul>																



*Please feel free to test out any of these features yourself! To run our application:*

Frontend folder: **npm dev run**

Backend folder: **python ./manage.py runserver**

To login with any existing students/see or add data to system → <http://127.0.0.1:8000/admin>

---

## Updated User Stories

**Problem statements are concise and correctly reflect the project topic. Description of the Feature and description of the Scenario, each with correct notational language.**

High Priority: Unique to UniCollab

Problem Statement 1:

**Forming compatible groups based on location, scheduling and skill-matching for specific UNSW projects and courses is difficult as people are not sure what they are looking for**

Requirement 1: (Use Case: Group Leader)

**As a group leader of a UniCollab group, I would like to manage the information regarding my group and control the list of members it holds.**



Feature 1 - Create a group within a UniCollab room

**As** a groupless user looking to become a group leader

**So that** I can form my own group and manage who is a part of it

**I want to** create a new UniCollab group

**SCENARIO:** A groupless member in the SENG2021 room wants to form their own group and appeal to people with similar mindsets.

**Given** I am in a UniCollab room and I am not in a group

**When** I click the 'Create Group' button from the Room Page

**Then** I see a prompt prompting me to type in a group name

**When** I type in a group name

**And** I click 'Create'

**Then** my group is created with this information (inheriting the preferred meeting location from my own preferred meeting location), meaning it will appear in the groups list of the UniCollab room where it is viewable by others

**And** I am taken to my group's Group Page

Feature 2.1 - Search for users in room based on availabilities

**As** a group leader looking for people to join my group

**So that** I can find people who are able to attend my group's meetings

**I want to** filter-sort users based on their compatibility with group's preferred meeting times

**SCENARIO:** A group leader in SENG2021 21T1 of the group, 'Attack on HD' wants to recruit people who are available to meet on Wednesdays at 5-8pm.

**Given** I am a group leader on my group's Group Page

**When** I click 'Invite Members'

**Then** I am shown a scrollable list of users in the room who do not currently belong to a group

**When** I click the filter icon

**And** I click 'Timetable'

**Then** I see a sorted list of users from most compatible (top) to least compatible (bottom) colour coded green → yellow → red based on how compatible their free times are with my group's preferred meeting times

- **Green:** They are available for 90% or more of the group's meeting times
- **Yellow:** They are available for 60-90% of the group's meeting times
- **Red:** They are available for less than 60% of the group's meeting times

Feature 2.2 - Search for users in room based on distance

**As** a group leader looking for people to join my group

**So that** I can find people who are able to easily attend my group's meetings

**I want to** filter-sort users based on their distance to my group's preferred meeting location



**SCENARIO:** A group leader in SENG2021 21T1 of the group, 'Attack on HD' wants to recruit people who can meet at Town Hall Station

**Given** I am a group leader on my group's Group Page

**When** I click 'Invite Members'

**Then** I am shown a scrollable list of users in the room who do not currently belong to a group

**When** I click the filter icon

**And** I click 'Location'

**Then** I see a sorted list of users from nearest (top) to furthest (bottom) colour coded green → yellow → red based on their distance to my group's preferred meeting location

- **Green:** Their location is less than 10km from the group's preferred meeting location
- **Yellow:** Their location is between 10-30km from the group's preferred meeting location
- **Red:** Their location is more than 30km from the group's preferred meeting location

Feature 2.3 - Search for users in room based on skillset compatibility

**As a group leader looking for people to join my group**

**So that** I can find people who will be able to work well with my group

**I want to** filter-sort users based on their synergy with the skills my group is looking for and the skills my group currently has

**SCENARIO:** A group leader in SENG2021 21T1 of the group, 'Attack on HD' wants to recruit someone who knows Python and Django and as a bonus, SQL (since many members in the group know it and will be using it)

**Given** I am a group leader on my group's Group Page

**When** I click 'Invite Members'

**Then** I am shown a scrollable list of users in the room who do not currently belong to a group

**When** I click the filter icon

**And** I click 'Skillset'

**Then** I see a sorted list of groups from most compatible (top) to least compatible (bottom) colour coded green → yellow → red based on how well my skillset synergises with their needs and current members (In depth algorithm can be viewed inside the code base)

Feature 2.4 - Search for users in a room based on all 3 criteria

**As a group leader looking for people to join my group**

**So that** I can find people who will be able to work well with my group

**I want to** filter-sort groups based on all 3 previously mentioned factors (timetable, location, skills)

**SCENARIO:** A group leader in SENG2021 21T1 of the group, 'Attack on HD' wants to recruit a suitable candidate based on a mix of all 3 factors (availability, location, skillset)

**Given** I am a group leader on my group's Group Page

**When** I click 'Invite Members'

**Then** I am shown a scrollable list of users in the room who do not currently belong to a group



**When** I click the filter

**And** I click 'Overall'

**Then** I see a sorted list of groups from most suitable (top) to least suitable (bottom) colour coded green → yellow → red, ranked by combining all 3 prior criteria in an equally weighted manner. (In depth algorithm can be viewed inside the code base)

Feature 3 - Invite users to group

**As** the leader of a UniCollab group

**So that** I can notify suitable individuals of my intent to recruit them

**I want to** invite them to join my group

**SCENARIO:** A group leader wishes to invite a certain individual who they believe will be a good fit for their group

**Given** I am a group leader on my group's Group Page

**When** I click 'Invite Members'

**Then** I am shown a scrollable list of users in the room who do not currently belong to a group

**When** I click 'Invite' next to the user I wish to invite

**Then** a popup appears informing 'Invite Successful'

**And** the invitation is sent to that user

Feature 4 - Update my group's information

**As** the leader of a UniCollab group

**So that** I can update my group's information

**I want to** edit my group's image, description and looking for skills and maximum capacity

**SCENARIO:** A group has hit the maximum capacity, but the group leader wishes to change the capacity and description to advertise one more vacancy

**Given** I am a group leader on my group's Group Page

**When** I click the 'Settings' button

**Then** all fields become editable to the group leader only

**When** I increase the maximum capacity of the group and change the description text,

**And** I click 'Save Changes'

**Then** all the changes will be saved and reflected when others visit my Group Page

Feature 5 - Remove members from the group

**As** a UniCollab group leader

**So that** I can ensure my group is only filled with people of complementary skills and similar mindsets to me

**I want to** remove users from my group



**SCENARIO:** A group leader no longer feels like a certain group member meshes well with their group atmosphere

**Given** I am a group leader on my group page

**When** I click the 'Settings' button

**Then** I can see red crosses next to my members' names

**When** I click on a the cross next to their name

**Then** that member is removed from the group

Requirement 2: (Use Case: Groupless Room Member)

**As a UniCollab member who does not belong to a group for the room I am in, I would like a system that assists me to find appropriate groups and allow me to join them.**

Feature 1.1 - Search for groups in a room based on availabilities

**As a groupless member of a UniCollab room**

**So that** I can find a group who meets when I am free

**I want to** filter-sort groups based on their compatibility with my availabilities

**SCENARIO:** A SENG2021 groupless member can only meet on Wednesdays and Thursdays

**Given** I am on the Room Page of a UniCollab room (SENG2021 21T1)

**When** I click the filter icon

**And** I click 'Timetable'

**Then** I see a sorted list of groups from most compatible (top) to least compatible (bottom) colour coded green → yellow → red based on how compatible their preferred meeting times are with my free times

- **Green:** I am available for 90% or more of the group's meeting times
- **Yellow:** I am available for 60-90% of the group's meeting times
- **Red:** I am available for less than 60% of the group's meeting times

Feature 1.2 - Search for groups in a room based on location

**As a groupless member of a UniCollab room**

**So that** I can find a group who meets a location convenient to me

**I want to** filter-sort groups based on the distance of their preferred meeting location to my location

**SCENARIO:** A SENG2021 groupless member lives in Bondi and wishes to find groups who meet somewhere close to them for their convenience

**Given** I am on the Room Page of a UniCollab room (SENG2021 21T1)

**When** I click the filter icon

**And** I click 'Location'

**Then** I see a sorted list of groups from closest (top) to furthest (bottom) colour coded green → yellow → red based on the distance of their preferred meeting location to my location



- **Green:** My location is less than 10km from the group's preferred meeting location
- **Yellow:** My location is between 10-30km from the group's preferred meeting location
- **Red:** My location is more than 30km from the group's preferred meeting location

Feature 1.3 - Search for groups in a room based on skillset

**As** a groupless member of a UniCollab room

**So that** I can find a group who synergises with my skillset

**I want to** filter-sort groups based on their need for my skills and my compatibility with their existing skills.

**SCENARIO:** A SENG2021 groupless member is looking for a group who is looking for their skillset whose members have skills which synergises with their skillset

**When** I click the filter icon

**And** I click 'Skillset'

**Then** I see a sorted list of groups from most compatible (top) to least compatible (bottom) colour coded green → yellow → red based on how well my skillset synergises with their needs and current members (In depth algorithm can be viewed inside the code base)

Feature 1.4 - Search for groups in a room based on all 3 criteria

**As** a groupless member of a UniCollab room

**So that** I can find a group which groups suit me the best

**I want to** filter-sort groups based on all 3 previously mentioned factors (timetable, location, skills)

**SCENARIO:** A SENG2021 groupless member wishes to find the most suitable group for themselves

**Given** I am on the Room Page of a UniCollab room (SENG2021 21T1)

**When** I click the filter icon

**And** I click 'Overall'

**Then** I see a sorted list of groups from most suitable (top) to least suitable (bottom) colour coded green → yellow → red, ranked by combining all 3 prior criteria in an equally weighted manner.

Feature 2 - Users may request to join a group

**As a** groupless UniCollab user

**So that** I can communicate my desire to join a group I am interested in

**I want to** send a join request message to the group leader

**SCENARIO:** A groupless user in the room 'SENG2011 Project' wishes to join a specific group

**Given** I am a groupless UniCollab user the room 'SENG2011 Project'

**When** I click on a group's name or image



**Then** I will be taken to their group page

**When** I click on the 'Join' button

**Then** a popup will appear, prompting me to type a message

**When** I type in a message and click 'Join'

**Then** the group leader will be notified of my request

## Problem Statement 2:

**It is currently difficult to view information of other UNSW CSE students, as there is no common interaction application.**

Requirement 1: (Use Case: Any user)

**As a UniCollab user, I would like to enter relevant private data about myself such as upload my personal calendar, location and courses so that I can optimise group matching by forming groups with people whose availability, location and skill sets align with mine.**

Feature 1 - Sync UniCollab Calendar with Google Calendar

**As a user looking to form groups with people that suit my schedule**

**So that** I can easily find groups with compatible availabilities for group meetings

**I want to sync my UniCollab Calendar with my personal calendar**

**SCENARIO:** A student wants their UniCollab calendar to align with their Google calendar including all their personal, work and university commitments.

**Given** I am on my Profile Page

**When** I click the Calendar button

**Then** a Calendar Popup appears of the current week with arrows to cycle to the next

**When** I click the Sync button

**Then** I am prompted to enter my Google Account details by a popup

**When** I enter my details

**Then** the UniCollab calendar syncs with my Google Calendar, meaning it now accurately displays all my Google Calendar commitments

Feature 2 - Enter a preferred meeting location to profile

**As a UniCollab user**

**So that** I can easily find groups and members who want to meet near the same place

**I want to enter my location into UniCollab**

**SCENARIO:** A student wants to update their preferred meeting location to make it easier to form groups with members closer to them



**Given** I am on my Profile Page

**When** I click on “Preferred Meeting Location”

**Then** a Google Maps popup appears showing me my preferred meeting location

**When** I type an address of my preferred meeting location into the search field

**And** I press enter

**Then** my location is updated for group matching purposes

## Problem Statement 3

**There is no central hub for me to organise my group formation activities, requiring me to use a variety of different platforms for each course**

Requirement 1: (Use Case: Course Administrator)

**As a UNSW CSE Course Administrator, I would like to create and manage a space where my students can form compatible groups.**

Feature 1 - Create a Unicollab room

**As a UNSW CSE course administrator**

**So that** students can organise themselves into groups for my course

**I want to** create a UniCollab Room which people can join and form groups with others in the room

**SCENARIO:** At the start of the term, the lecturer of SENG2021 wishes to create a UniCollab Room for his students to form themselves into groups of 5-6 for the term project

**Given** I am logged in to my UniCollab and on the Main Dashboard Page for my account

**When** I click on the ‘Create Room’ button

**Then** I am prompted to enter a room name and description

**When** I enter a room name (‘SENG2021 21T1’) and description (‘Form groups of 5-6 for the SENG2021 term project here!’) and click ‘Create Room’

**Then** I am taken to the newly created UniCollab Room Page

**And** my room is added to the existing rooms list viewable on the Dashboard Page

Requirement 2: (Use Case: Any user)

**As a user of UniCollab, I would like to access a personalised dashboard page so that I can access my rooms and groups as well as search for and join rooms from one location**

Feature 1: Users can access rooms and groups they are in

**As a UniCollab user**

**So that** I can directly navigate to groups and rooms I am a part of

**I want to** view a scrollable list of my current groups and rooms on my dashboard page



**SCENARIO:** A student wishes to have quick access to their groups and rooms

**Given** I am an active UniCollab user

**When** I log in to my account

**Then** I see a scrollable list of my groups and rooms from my dashboard page

**When** I click on a particular group or room

**Then** I will be transported to that group/room's page

Feature 2: Students can search for existing rooms

**As** an existing UniCollab user

**So that** I can find rooms to join

**I want to** view a searchable list of existing rooms from my dashboard page

**SCENARIO:** A student wants to find a room relating to their course/project

**Given** I am an active UniCollab user

**When** I log in to my account

**Then** I see a scrollable list of existing rooms with a search bar on my dashboard

**When** I enter a query into the search bar

**Then** I see a filtered list of rooms matching my query

Low Priority: Additional Features which enhance the UniCollab experience

Problem Statement 1 (Additional Features)

**Forming compatible groups based on location, scheduling and skill-matching for specific UNSW projects and courses is difficult as people are not sure what they are looking for**

Requirement 3: (Use Case: Group member)

**As a member of a UniCollab group, I would like to be able to leave the group if I deem them to no longer be a suitable fit for me in terms of location, scheduling, skillset or other factors**

Feature 1: Group member can leave a group

**As** a member of a UniCollab group

**So that** I can look for a more compatible group which suits my needs better

**I want to** be able to leave the group I am currently in

**SCENARIO:** A group member of Attack on HD is no longer available for the group's preferred meeting time of Thursday 6-8pm and they wish to leave the group to find another group which can accomodate their availabilities

**Given** I am on my group's Group Page



**When** I click 'Leave'

**Then** a popup prompt appears asking me if I am sure I want to leave the group

**When** I click Yes

**Then** I am removed from the group

**And** I am on the Room Page

## Problem Statement 2 (Additional Features)

**It is currently difficult to view information of other UNSW CSE students, as there is no common interaction application.**

Requirement 1: (Use Case: Any user)

**As a UniCollab user, I would like to enter relevant private data about myself such as upload my personal calendar, location and courses so that I can optimise group matching by forming groups with people whose availability, location and skill sets align with mine.**

Feature 3 - Enter courses taken

**As a UniCollab user**

**So that** I can easily find groups and members who synergise well with my existing skillset  
**I want to** enter in the courses I have completed into UniCollab

**SCENARIO:** A student wants others to know that they have done COMP1531 so that they can find a group who requires 'Python' skills.

**Given** I am on my Profile Page

**When** I click on '+' next to 'Completed Courses'

**Then** a prompt appears prompting me to type in a course code

**When** I type in a course I have taken

**And** I press enter

**Then** my courses are updated for skillset matching purposes

## Requirement 2

**As a UniCollab user, I would like to be able to add and update my personal information, so that it is always relevant and accurately reflects who I am to other users viewing my profile**

Feature 1: Students can update their profile page biography

**As a student using UniCollab**

**So that** I can convey information about myself to other UniCollab users

**I want to** add/edit my own biography



**SCENARIO:** A student wishes to update their biography to reflect their interests, hobbies, co-curricular commitments and other personal information

**Given** I am on my own profile page

**When** I click the 'Edit' button

**Then** I can edit the text of my biography

**When** I type in a biography and click the 'Confirm Changes' button

**Then** my biography will be updated and viewable to other users

Feature 2: Students can update their profile photo

**As a student using UniCollab**

**So that** I can display an updated image of myself

**I want to** change my existing profile page to a different image

**SCENARIO:** A student wishes to update their profile picture, to one that has been taken more recently

**Given** I am on my own profile page

**When** I click the 'Upload Profile Picture' button

**Then** a popup will open, prompting me to update my profile picture

**When** I select an image from my device and click the 'Upload' button

**Then** my profile picture will be updated and viewable to other users

Feature 3: Students can link external sources of Social Media

**As a student using UniCollab**

**So that** I can provide other users with additional information about myself without entering it manually

**I want to** be able to link my Facebook, Instagram and LinkedIn profiles

**SCENARIO:** A student has created a UniCollab profile

**Given** I am a registered UniCollab user

**When** I access my profile page

**Then** I will see the option to link to Facebook, Instagram and LinkedIn

**When** I click on one of these icons

**And** pass the authentication process

**Then** my profile will be adjusted to link my profiles to these external sources

Requirement 3

**As a UniCollab user, I would like to view a person's profile page so that I can learn more about their interests, groups and their courses to determine if I want to group with them**



Feature 1: Able to navigate another user's profile from any instance they appear on the UniCollab site to view information about them

**As a UniCollab user**

**So that** I can view information regarding another user

**I want to** be able to click their name or image and be navigated to their profile page

**SCENARIO:** A student sees a groupless member in a room that they are both part of and wishes to learn more about them to determine if they would like to form a group with them

**Given** I am viewing a list of members in a UniCollab room

**When** I click on another user's name

**Then** I will be navigated to their profile page

**And** able to view their profile picture, biography, completed courses and active groups

### Problem Statement 3 (Additional Features)

**There is no central hub for me to organise my group formation activities, requiring me to use a variety of different platforms for each course**

#### Requirement 1

**As a user of UniCollab, I would like to access a personalised dashboard page so that I can access my rooms and groups as well as join and leave rooms from one location**

#### Feature 3: Joining UniCollab rooms

**As a user of UniCollab**

**So that** I can form a group with other people mutually interested in the room topic

**I want** to join an existing UniCollab room

**SCENARIO:** A student wishes to join the SENG2021 room to form a SENG2021 project group with other SENG2021 members

**Given** I am on my main dashboard page

**When** I click the 'Join' button next to a room name on the scrollable list

**Then** I am taken to that room page

#### Feature 4: Leaving UniCollab rooms

**As a student using UniCollab**

**So that** I can remove myself from a room I no longer wish to be a part of

**I want** to leave that UniCollab room

**SCENARIO:** A student dropped COMP1531 and no longer needs to form a group for that course



**Given** I am on my main dashboard page

**When** I click on the Leave button next to one of my rooms (e.g. COMP1531)

**Then** I see a popup confirming if I want to leave room

**When** I click 'Leave'

**Then** I will be removed from that room and any UniCollab group I am involved with in that room

**And** the room will be removed from my UniCollab dashboard page

## Requirement 2

**As a UNSW CSE student, I want to use a personal UniCollab account so I can participate in the UniCollab group formation ecosystem with my own identity**

Feature 1: Students can sign up for a UniCollab account

**As a student who wishes to be involved in the UniCollab group formation process**

**So that** I can manage my UniCollab groups safely and securely

**I want to** create a personal, password protected UniCollab account

**SCENARIO:** A student wishes to join UniCollab and participate in a group formation process

**Given** I am on the the UniCollab Welcome Page

**When** I click 'Get Started'

**Then** I am taken to the Sign Up Page

**When** I enter my 'First Name', 'Last Name', 'Email', 'Password' and 'Confirm Password'

**And** I click the 'Sign Up' button

**Then** I am logged in to my newly created UniCollab account

**And** am taken to the Main Dashboard Page

Feature 2: Students can log in to their UniCollab account

**As an existing UniCollab user**

**So that** I can access and manage my UniCollab groups

**I want to** login to my UniCollab account

**SCENARIO:** A student wishes to login to their UniCollab account to view their UniCollab groups

**Given** I am on the UniCollab Welcome page

**When** I click 'Log In'

**Then** I am taken to the Login Page

**When** I enter the correct email and password for my account

**Then** I am logged in to my UniCollab account

**And** I am taken to the UniCollab Main Dashboard Page



Feature 3: Students can log out of their UniCollab account

**As a logged in UniCollab user**

**So that I can discontinue using UniCollab and prevent unauthorised actions**

**I want to** logout of my UniCollab account

**SCENARIO:** A student wishes to logout of their UniCollab account

**Given** I am on my UniCollab Main Dashboard Page (home)

**When** I click the 'Logout' button

**Then** I am taken to the UniCollab Welcome Page

## Problem Statement 4

**Current alternatives in place (such as posting on a course forum) do not have responsive notification systems.**

Requirement 1

**As a UniCollab group leader, I would like to be notified when a fellow UniCollab user requests to join my group.**

Feature 1: Group leaders are notified when a user requests to join their group

**As a UniCollab group leader**

**So that I may stay informed in the group-making process**

**I want to receive notifications when users request to join my group**

**Given** I am a group leader of a UniCollab group and have received a join request

**When** I open my notifications inbox

**Then** I can see the request in my notifications along with the user who sent it and their message

Feature 2: Group leaders can accept or decline requests

**As a UniCollab group leader**

**So that I may proceed in forming my ideal group,**

**I want to be able to accept or decline invitation requests that users have sent.**

**SCENARIO:** The leader of a group has received a request from another student who wishes to join their group.

**Given** I am a group leader of a UniCollab group and am viewing a request

**When** I click the 'Accept Request' or 'Decline Request' button to accept or decline

**Then** my decision will be communicated to the user that sent the request

Feature 3: Students are notified that their request is accepted

**As a user who sent a request to join a group**



**So that** I know if I have successfully formed a group or need to keep searching  
**I want to** be notified if my request to join a group has been accepted

**SCENARIO:** A group member is awaiting approval from their request to join a group

**Given** I have requested to join a group

**When** the group leader has decided to accept my request

**Then** a red circle will appear on the notification icon in the top right of the screen

**When** I click the notification icon

**Then** I can view the notification

**And** see that my request was accepted

**And** be granted full access to this group via my main dashboard page and profile page

Feature 4: Students are notified that their request is declined

**As a user who sent a request to join a group**

**So that** I know if I have successfully formed a group or need to keep searching

**I want to** be notified if my request to join a group has been declined

**SCENARIO:** A group member is awaiting approval from their request to join a group

**Given** I have requested to join a group

**When** the group leader has decided to decline my request

**Then** a red circle will appear on the notification icon in the top right of the screen

**When** I click the notification icon

**Then** I can view the notification

**And** see that my request was declined

Requirement 2

**As a group leader, I would like to request for a specific user to join my group if I think they will be a good fit.**

Feature 1: Users receive a notification if a UniCollab group leader has sent an invitation to join their group

**As a UniCollab user**

**So that** I can be recruited by group leaders

**I want to** be notified when a group leader invites me to join their group

**SCENARIO:** A UniCollab user receives an invitation to join a group for the room 'COMP1511 Assignment 2'

**Given** I am a participant of the 'COMP1511 Assignment 2' room (grouped or ungrouped)

**When** a group leader under this room sends me an invitation to join their room

**Then** I can view the notification by clicking on the Notification Icon



Feature 2: Users can accept or decline invitations sent by group leaders

**As a UniCollab user**

**So that** I can progress in the group-making process

**I want to** be able to accept or decline an invitation to a group I am not currently in

**SCENARIO:** A user receives a notification to join a group for a room they are active in, and would like to respond to this invitation

**Given** I am a UniCollab user who has received an invitation to a group

**When** I click on the group's name or image

**Then** I am transported to the publically viewable group page

**When** I click 'Accept'

**Then** I become a member of the group

**When** I click 'Reject'

**Then** I do not become a member of the group

**And** in either circumstance, the group leader is informed via a notification

Feature 3: When a user has responded to an invitation to join a group, the group leader who sent the invitation is notified of the outcome

**As a UniCollab group leader**

**So that** I can determine whether users have joined my group

**I want to** receive a notification regarding the response of my group invitation

**SCENARIO:** A group leader has sent an invitation to another UniCollab user. This user has decided to decline the invitation.

**Given** I am a group leader who has sent an invitation to a user

**When** that user has declined my invitation

**Then** I will receive a notification that informs me that this invitation to said user has been declined

### Requirement 3

**As a member of a UniCollab group, I would like to be notified when new messages are sent between members of said group.**

Feature 1: Students receive notifications when group chats possess messages that a user has not yet viewed

**As a UniCollab group member**

**So that** I can stay connected to my group and informed on our project's developments

**I want to** be notified when chat messages are sent to my group that I have not yet seen

**SCENARIO:** Anna is in a group for COMP1531 and Bob sends a link for a recorded lecture

**Given** that Anna and Bob are in the same group for COMP1531

**When** Bob sends a message containing a URL to the group



**Then** all group members who are not currently viewing the chat component will receive a notification immediately (if browsing another UniCollab page) or next time they log in (if the group member has logged out of UniCollab)

**And** said notification will be added to their notification list

#### Requirement 4

**As a UniCollab user, I would like to be able to easily keep track of my notifications and be informed when I receive new ones.**

Feature 1: A red circle appears above the notification icon for a student who has received an unviewed notification

**As a UniCollab user**

**So that** I can stay informed on all updates relevant to me

**I want to** know when I have new notifications that I have not opened

**SCENARIO:** While a student was logged out, a member of one of their groups sent a message to said group

**Given** I am a UniCollab user who is part of a group that has been communicating since I was last logged in

**When** I log into my UniCollab account

**Then** my notifications icon will possess a small red circle to indicate that I have received unviewed notifications

Feature 2: Students may view notifications

**As a UniCollab user wanting to browse my notifications**

**So that** I can view all updates I have missed

**I want to** be able to access a list of notifications

**SCENARIO:** A student receives a notification and would like to view it

**Given** I am a user of UniCollab and have received an notification

**When** I click on the notification icon to the top right beneath my profile box

**Then** I will view a list of all my notifications

**When** I press an individual notification

**Then** it expands to show the full message (if the message was cut off)

---



## Achieved Components: User Interface

Please refer to our [\*\*UniCollab Features\*\*](#) table for an indication of all the functionality that was implemented in our final user interface. Below we will show a small selection of screens modelling some of our favourite features and the key functionality of our system as well as validating its correctness and effectiveness.

We have decided to log in as Lara Carter. Her profile reveals all the courses she has completed and also that she is not in any Active Groups. Her Preferred Meeting Location is hidden and can be changed and revealed using the 'Edit Profile' button.



# Lara Carter



## Bio

my profile pic is my current mood

Preferred Meeting Location

### Completed Courses

COMP1511 COMP2521  
COMP1521  
COMP1531 COMP2511

### Active Groups

## Bio

my profile pic is my current mood

Preferred Meeting Location (visible to you only)

Enter the suburb/postcode of your preferred meeting location

Liverpool, Sydney

This is Lara's view when selecting to filter groups by location in the SENG2021 Projects Room (which she has navigated to from her main dashboard page)



**SENG2021 PROJECTS**

Find a group by scrolling through the list below, or filtering your search using the tool

Search for group...

Overall Match  
Timetable  
Location  
Skillset

Groups with proposed meeting times that match your calendar commitments will be displayed green, if you are partially available for a group's proposed meeting times, the group will be displayed yellow. Incompatible groups will be listed as red. You may click a box to see more information about the group.



**Fashion Icons**

Looking For...  
 - User Interfaces  
 - Text Processing  
 - Data Modelling  
 - Project Management  
 - User Interfaces

Members  
 - Anna Lipton

Vacancy **1/5**



**Gym Bros**

Looking For...  
 - Django  
 - SQLite  
 - Database Modelling

Members  
 - Edward Wong  
 - Aiden Shi

Vacancy **2/5**

As you can see from the json data that is sent to the front end, the groups are sorted from closest to furthest and the match scores are correctly applied with match scores of 0 for 'Attack on HD', 'We <3 Fitness' and 'Demo Test Group' as they are over 30km away from Lara.  
 (Reminder the group's preferred meeting location is set to the group leader's preferred meeting location, in this case Anna's) Note the url path below: Lara is student id 14, SENG2021 is room id 1

```
← → ⌂ 127.0.0.1:8000/rooms/14/location/1
Apps Moodle webcms Facebook Mail - Tiana Dourou... Messenger discord Google Docs
distance| 1/6 ▲ ▼ × Reading List
[{"id": 5, "name": "Fashion Icons", "members": [{"name": "Anna Lipton"}], "description": "I love fashion do you? Join me!", "location": "Cabrannatta, Sydney", "preferredMeetingTimes": [{"eventName": "Test Meeting", "start": "2021-04-28T16:00:00Z", "end": "2021-04-28T19:00:00Z"}, {"eventName": "Initial Meeting", "start": "2021-04-28T16:00:00Z", "end": "2021-04-28T19:00:00Z"}], "photo": "\\\groupAvatar\\Screen_Shot_2021-04-21_at_11.02.06_pm.png\"", "lookingFor": ["User Interfaces", "Text Processing", "Data Modelling", "Project Management", "User Interfaces"], "weHave": {}, "capacity": 5, "vacancy": 4, "distance": 4.9, "match": 2}, {"id": 4, "name": "Gym Bros", "location": "Hurstville, Sydney", "preferredMeetingTimes": [{"eventName": "Trial run", "start": "2021-04-23T08:00:00Z", "end": "2021-04-23T11:00:00Z"}, {"eventName": "Trial run", "start": "2021-04-23T08:00:00Z", "end": "2021-04-23T11:00:00Z"}], "photo": "\\\groupAvatar\\Screen_Shot_2021-04-21_at_10.57.27_pm.png\"", "lookingFor": ["Django", "SQLite", "Database Modelling"], "weHave": ["Python"], "capacity": 5, "vacancy": 3, "distance": 20.1, "match": 1}, {"id": 6, "name": "CHAMPS", "members": [{"name": "Kenny Yamani"}, {"name": "Edward Wong"}], "description": "We are the champions!", "location": "Ashfield, Sydney", "preferredMeetingTimes": [{"eventName": "Discussion Meeting", "start": "2021-04-25T08:00:00Z", "end": "2021-04-25T09:00:00Z"}, {"eventName": "Brainstorm Meeting", "start": "2021-04-26T18:00:00Z", "end": "2021-04-26T19:00:00Z"}], "photo": "\\\groupAvatar\\Screen_Shot_2021-04-21_at_11.00.04_pm.png\"", "lookingFor": ["Java", "OOP", "web-based application"], "weHave": ["C", "Project Management", "Java"], "capacity": 5, "vacancy": 3, "distance": 26.3, "match": 1}, {"id": 1, "name": "Attack on HD", "members": [{"name": "Fintan O'Shea"}, {"name": "Edward Wong"}], "description": "Join us! Hello we are a nice family, welcome to join us. Let's get started", "location": "NSW, Sydney", "preferredMeetingTimes": [{"eventName": "Weekly Meeting", "start": "2021-04-23T05:00:00Z", "end": "2021-04-23T05:00:00Z"}, {"eventName": "Weekly Meeting", "start": "2021-04-27T15:00:00Z", "end": "2021-04-27T15:00:00Z"}], "photo": "\\\groupAvatar\\A9F54B97-DF5F-4AD9-AAE3-EA3FA138CD88.jpeg\"", "lookingFor": ["Web-application", "Vue"], "weHave": ["Programming", "Data Modelling"], "capacity": 5, "vacancy": 3, "distance": 35.2, "match": 0}, {"id": 3, "name": "WE <3 FITNESS", "members": [{"name": "Christina Xie"}], "description": "we love fitness, wanting to implement a project in the health lifestyle category.", "location": "UNSW, Sydney", "preferredMeetingTimes": [{"eventName": "Project Brainstorm Meeting", "start": "2021-04-25T09:00:00Z", "end": "2021-04-25T15:00:00Z"}, {"eventName": "Presentation", "start": "2021-04-27T09:00:00Z", "end": "2021-04-27T11:00:00Z"}, {"eventName": "Project outline", "start": "2021-04-27T09:00:00Z", "end": "2021-04-27T11:00:00Z"}, {"eventName": "Presentation", "start": "2021-04-27T09:00:00Z", "end": "2021-04-27T11:00:00Z"}, {"eventName": "Project outline", "start": "2021-04-25T09:00:00Z", "end": "2021-04-27T15:00:00Z"}], "photo": "\\\groupAvatar\\Screen_Shot_2021-04-22_at_9.08.19_am_TMD0uM.png\"", "lookingFor": ["Data Modelling", "User Interfaces", "C", "Java", "OOP"], "weHave": {}, "capacity": 5, "vacancy": 4, "distance": 35.2, "match": 0}, {"id": 7, "name": "Demo Test Group", "members": [{"name": "Tiana Douroudis"}], "description": "Hey there!", "location": "UNSW, Sydney", "preferredMeetingTimes": [], "photo": "\\\groupAvatar\\\\\"", "lookingFor": ["Data Modelling", "SQLite", "C", "Data Modelling", "User Interfaces", "Algorithms"], "weHave": {}, "capacity": 5, "vacancy": 4, "distance": 35.2, "match": 0}]
```

Similarly, here is the data sent to the frontend when selecting skillset and timetable respectively:

Note: The frontend views (UI's) corresponding to the data sent to the server through the url paths above and below can be seen from items 14, 15 and 16 of the **UniCollab Features** table. You will notice that the groups appear in the exact order as the screenshots here with the match attribute corresponding to the correct colours. Also see **overall.py** for information on how the overall list is calculated



score 1/6 ^ v x Reading List

For more information on how this score and match is calculated see [skills.py](#)

A score of 1 indicates Lara is available for all of the group meeting times. It can be seen that she is only available for 75% of 'Attack on HD' preferred meeting times, thus it is ranked at the bottom.

## Calendar clashes:

## 'Attack on HD' preferred meeting times calendar

### Preferred Meeting Times

weekdays Sun - Sat

SUN	MON	TUE	WED	THU	FRI	SAT
25	26	27	28	29	30	1

1 AM      2 AM      3 AM

Weekly Meet 01 - 02      Weekly Meet 01 - 02

### Your Calendar

weekdays Sun - Sat

JN	MON	TUE	WED	THU	FRI	SAT
5	26	27	28	29	30	May 1
	Monday Lunch	01 Weekly CSESoS Me	09 Breakfast with the g			
2	3	4	5	6	7	8
		01 Weekly CSESoS Me	18 COMP35311 EXAM			
9	10	11	12	13	14	15
		01 Weekly CSESoS Me				

## Lara's personal calendar

As you can see Lara has a weekly meeting on Tuesdays 'Weekly CSEsoc Meeting' that clashes with the weekly meeting put forward by the group leader on Tuesday.

As form above, Fashion Icons was the top ranked group based on location, and after visiting the group page and Anna's profile page she decides this is the group she would like to join.



# Room: SENG2021 PROJECTS



Fashion Icons

About us

I love fashion do you? Join me!

JOIN

We have:

---

Looking for:

- User Interfaces
- Text Processing
- Data Modelling
- Project Management
- User Interfaces

Member:

Anna Lipton

Preferred Meeting Times

	SUN	MON	TUE	WED	THU	FRI	SAT
	25	26	27	28	29	30	1
11 AM							
12 PM					Initial Meet 12 - 14		
1 PM							
2 PM							
3 PM							

Now we will display screenshots of how the group page transforms when Anna joins in relation to our skill matching process based on our course scraper API. (See `scrape.py`)

Take into consideration that she has completed COMP1531

[webcms3.cse.unsw.edu.au/COMP1531/21T1/outline](https://webcms3.cse.unsw.edu.au/COMP1531/21T1/outline)

Moodle [webcms](#) [Facebook](#) [Mail - Tiana Dourou...](#) [Messenger](#) [discord](#) [Google Docs](#)

Linux on Windows	The students are exposed to agile software practices, team collaboration and effective communication through implementing a group project based on agile software methodologies that requires them to analyse, design, build and deploy a web-based application. This course is typically taken in the semester after completing COMP1511, but could be delayed and taken later. It provides essential background for the teamwork and project management required in many later courses.
Python Style Guide	
Tiana Douroudis Public	

products in effectively functioning teams.

The students are exposed to agile software practices, team collaboration and effective communication through implementing a group project based on agile software methodologies that requires them to analyse, design, build and deploy a web-based application. This course is typically taken in the semester after completing COMP1511, but could be delayed and taken later. It provides essential background for the teamwork and project management required in many later courses.

**2.1. Assumed Knowledge**

We assume all students have completed COMP1511 (or equivalent).

Students should be familiar with the basic concepts of programming, including loops, functions, libraries of code, compiling, and writing code to follow specifications.

**2.2. Student Learning Outcomes**

After completing this course, students will be able to:

1. Demonstrate effective use of applying the python programming language to solve problems in relation to web-based applications
2. Demonstrate proficiency in use of system schematics, data modelling, and state modelling, to analyse complex software

## And COMP2511

[webcms3.cse.unsw.edu.au/COMP2511/20T2/outline](https://webcms3.cse.unsw.edu.au/COMP2511/20T2/outline)

Moodle  webcms  Facebook  Mail - Tiana Dour...  Messenger  discord  Google Docs

---

- Understand the importance of team organization and communication
- Be able to work within a small team in the context of a software development project
- Be able to plan and execute a software project according a systematic software process
- **User Interfaces**
  - Become familiar with principles of effective user interface design
  - Be able to implement a user interface in Java

This course contributes to the development of the following graduate capabilities:

This course contributes to the following UNSW graduate attributes.

Lara Joins →

We have:

- Data Modelling
- Project Management
- User Interfaces

Looking for:

- Text Processing
- User Interfaces

Member:

 Anna Lipton  
 Lara Carter

leave button clicked →

We have:

Looking for:

- Text Processing
- User Interfaces
- Data Modelling
- Project Management
- User Interfaces

Member:

 Anna Lipton

You can see that the skills Lara possesses based on her completed courses have populated the **We have** section and when she leaves the group, the skills repopulate the **Looking for** section.

*Note: we use regex to ignore case and consider characters that might get in the way of a skill being detected (., ; “ - etc.). Also as Anna (group leader) had listed User Interfaces twice in her Looking for skills, this suggests she would like two people that possess this thus the skill is only removed once from the looking for list allowing it to be fulfilled by another member*

The **Fashion Icons** group will now appear on Lara's main dashboard as well as on her profile

**My Groups**

Fashion Icons	Final Report Group!
<a href="#">LEAVE</a>	<a href="#">LEAVE</a>

**Active Groups**



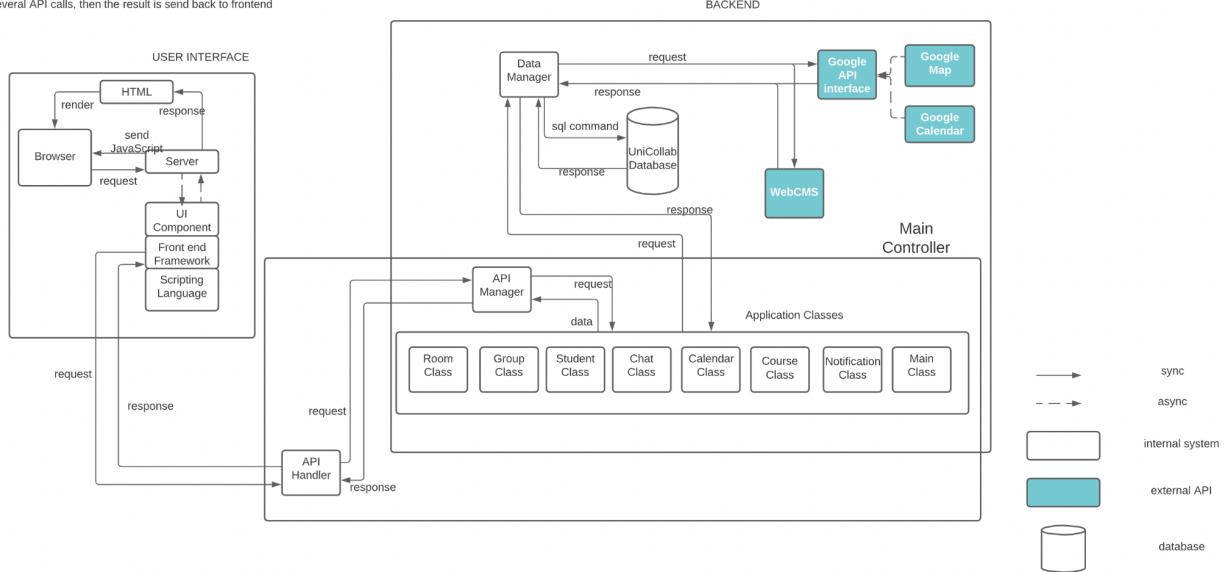
**SENG2021 PROJECTS: Fashion Icons**  
I love fashion do you? Join me!

## 2. Design and Code Inspection

### Final Software Architecture

#### UniCollab Software Architecture

Frontend receives data and send to backend for the OO structure and database to process which involves several API calls, then the result is send back to frontend



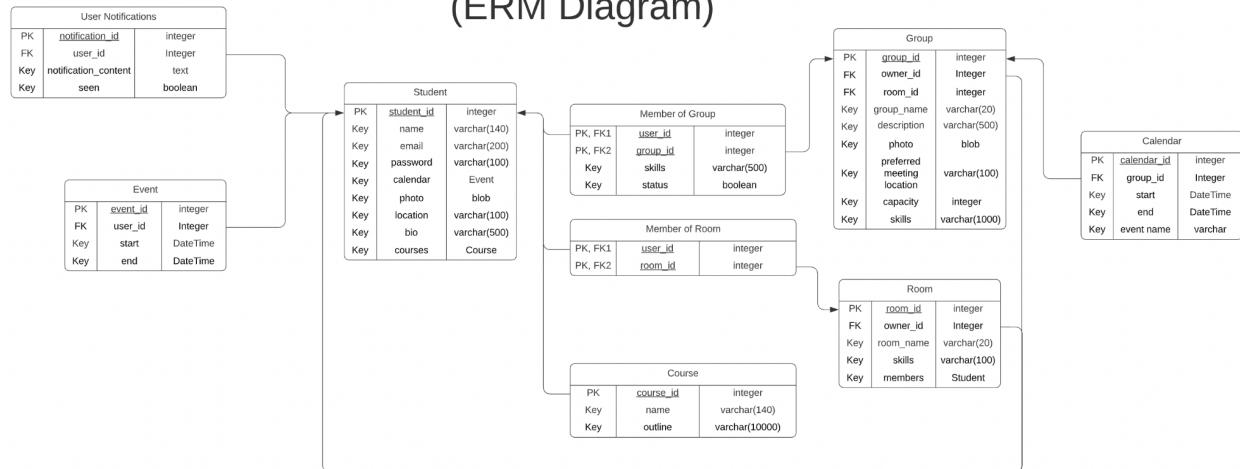
UniCollab relies on multiple components to collect information from external sources and an internal database. WebCMS is one of the external sources we used to collect the course outline by web scraping technique, then the data manager has a csv class to convert the course and course outline to course object. The Google Map API is queried on the fly by backend algorithms to fetch the distances between locations (extracted from the user class and the group class) when searching via the location filter. The Google Calendar API fetches the user's events and stores it in an array list, saving it to the user's calendar class (which is then fed into the search filter functions).

To summarise, our frontend user interface relies on JavaScript, CSS and HTML to communicate a frontend framework to the user's browser with the help of Vue framework. The frontend will interact with the backend via a REST API handler, to utilize python classes and the aforementioned API and web scraper to retrieve relevant data.

Within this architecture, our SQLite database will interact with Python classes, which uses the Django framework to transfer to a JSON format while interacting with the frontend. Then, Vue, Nuxt will be used to translate this JSON information into JavaScript content that will be accessed from a web browser.

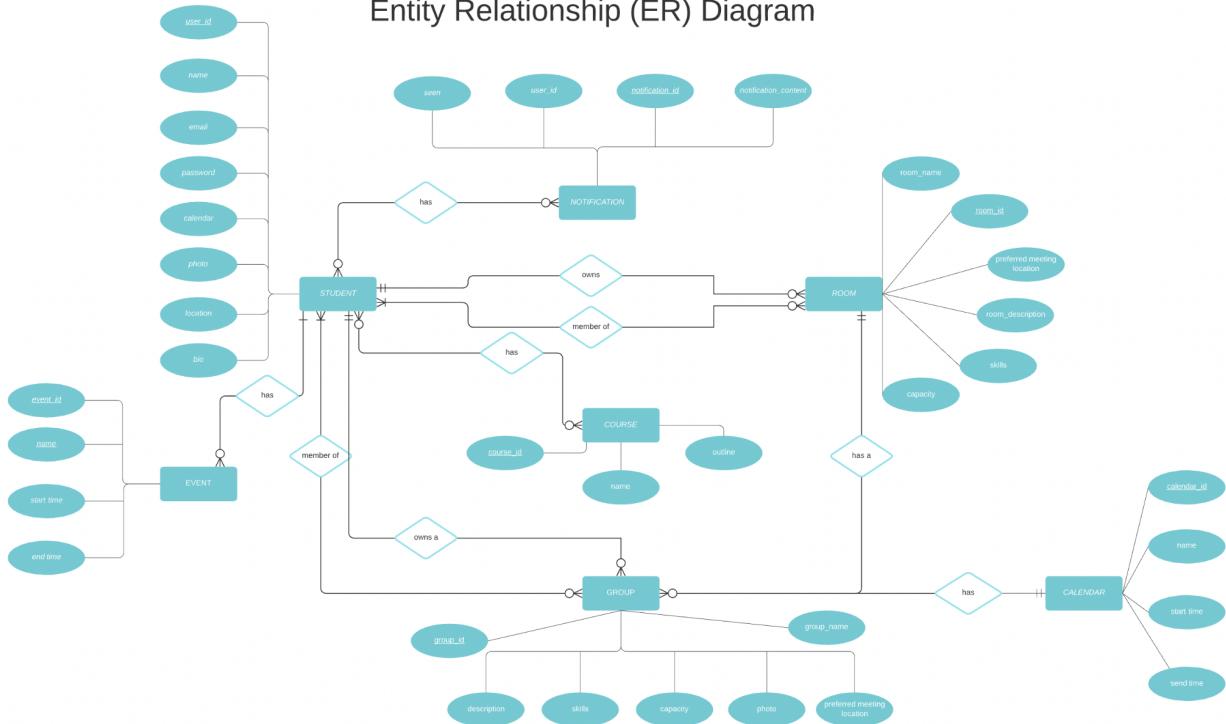
Below is our modelling table, it directly aligns with the django models in our backend which make up our sqlite database (see `models.py` in each entity folder)

## Entity Relationship Model of Database (ERM Diagram)



**Note:** The **skills** field of the **Group** object contains the groups ‘Looking For’ skills (charfield with skills separated by commas) which are by default set to skills of the room (decided by Course Admin when creating room). The ‘We Have’ skills are found by compiling the skills of each group member (stored in the **GroupMember** object) and displayed as a list on the group page.

## Entity Relationship (ER) Diagram





These entities can also be seen through the django administration page:

→ C 127.0.0.1:8000/admin/ ☆

Apps Moodle webcms Facebook Mail - Tiana Dour... Messenger discord Google Docs

Django administration WELCOME, TIANADOUROUDIS VIEW SITE / CHANGE PASSWORD

Site administration

**COURSES**

Courses + Add Change

**GROUPS**

Calendars + Add Change

Group members + Add Change

Groups + Add Change

**ROOMS**

Members + Add Change

Rooms + Add Change

**STUDENTS**

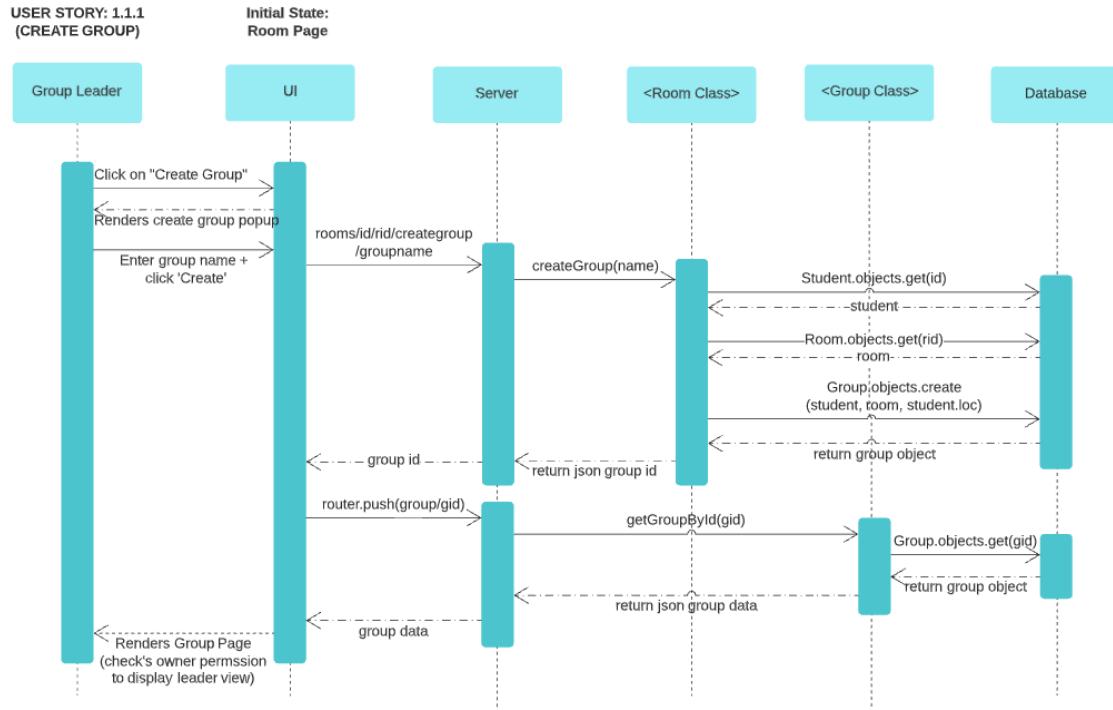
Events + Add Change

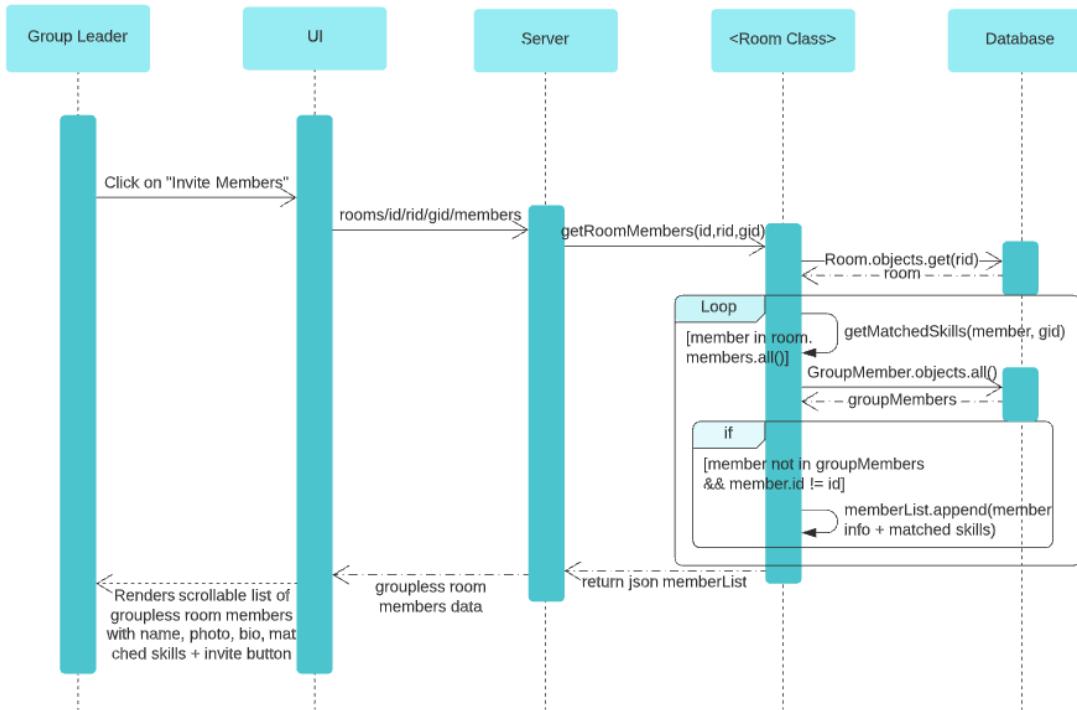
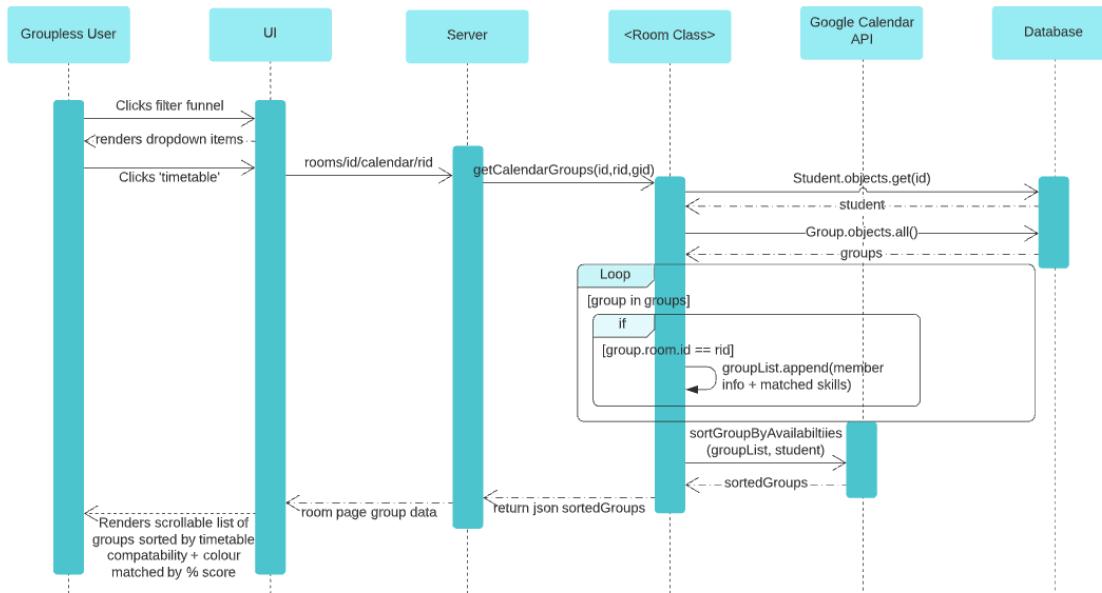
Students + Add Change

---

## Sequence Diagrams for each use case

As our application involves an extensive amount of use cases, **as instructed by our mentor**, we have decided to display the diagrams that map to our highest priority user stories → features important & unique to UniCollab.

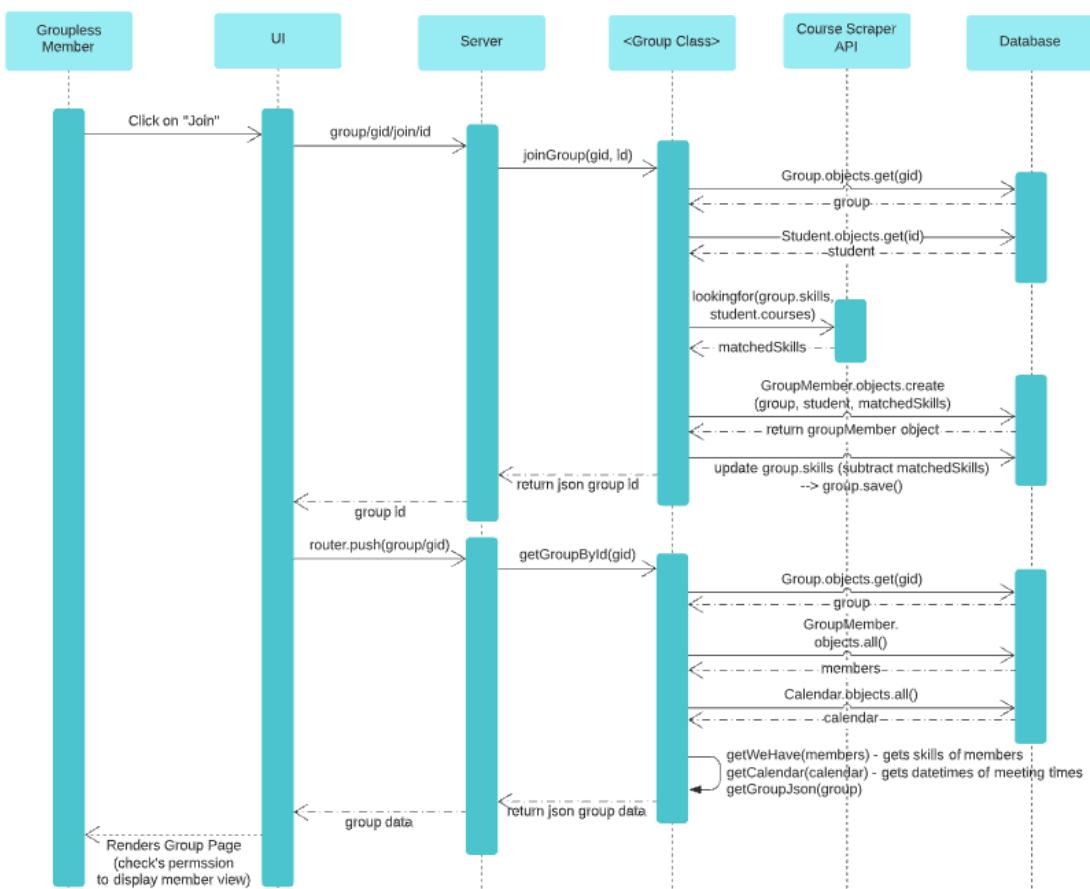


**USER STORY: 1.1.2  
(MEMBER SEARCH)**
**Initial State:  
Group Page**

**USER STORY: 1.2.1  
(GROUP SEARCH)**
**Initial State:  
Room Page**


**Note:** this is an example for when 'timetable' filter is applied, exact same process for 'location', 'skillset' and 'overall' with talking to relevant API's e.g. 'location' will call `sortGroupByDistance(groupList, student)` to Google Maps API (meeting.py)  
 Similar process for member search with instead of `sortGroupBy...` fn called will be `sortUserBy...` to the relevant API file

**USER STORY: 1.1.3 & 1.2.2  
(JOIN GROUP/  
ACCEPT INVITE)**

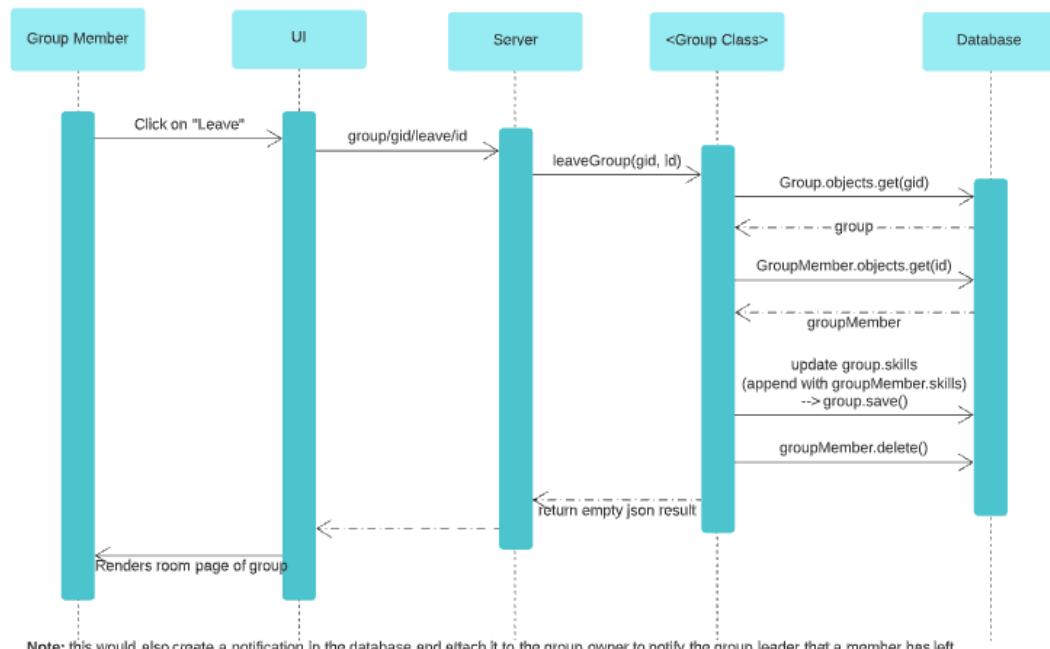
**Initial State:  
Group Page**



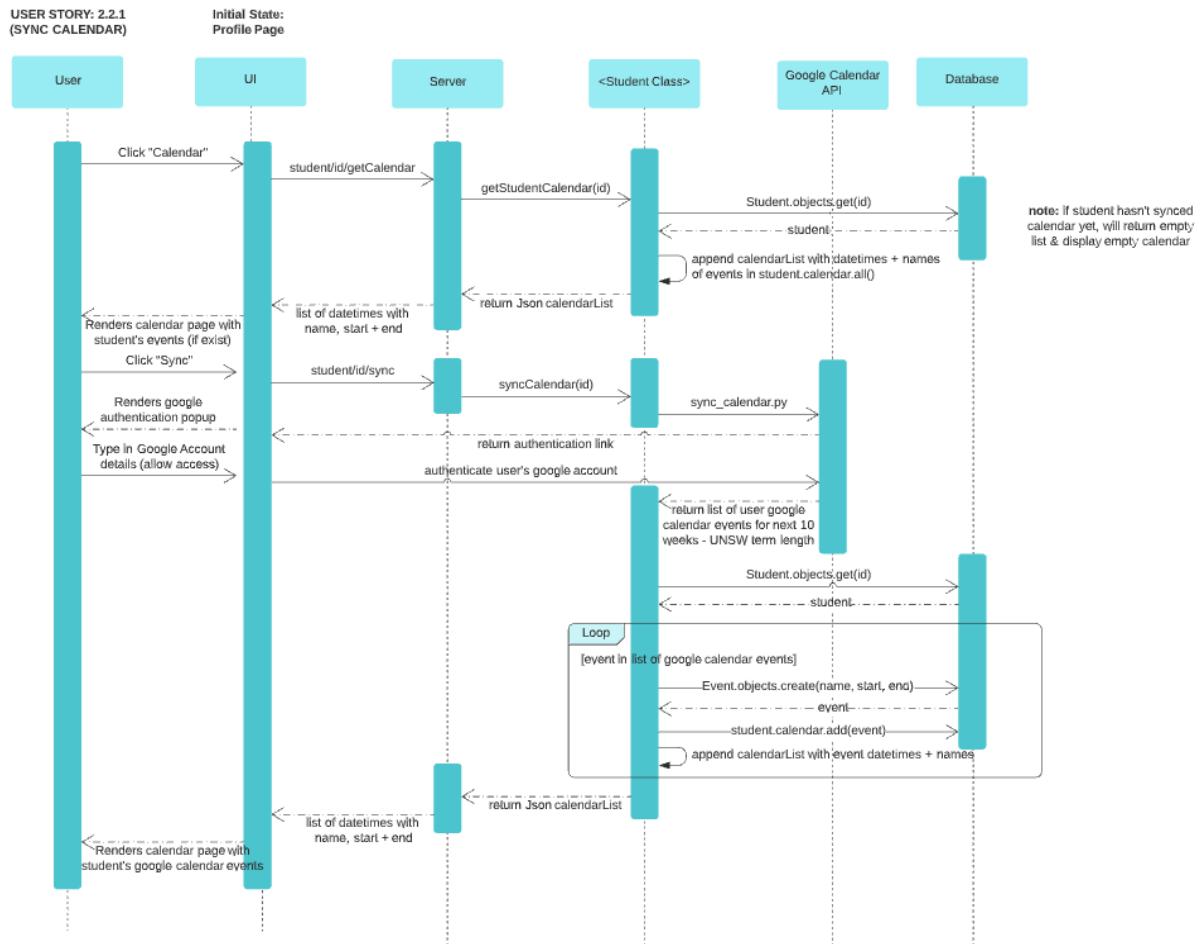
**Note:** this would also create a notification in the database and attach it to the user who clicked 'Join'. This is also the process for when a member is invited (and accepts the invite request notification)

USER STORY: 1.3.1  
(Leave Group)

Initial State:  
Group Page

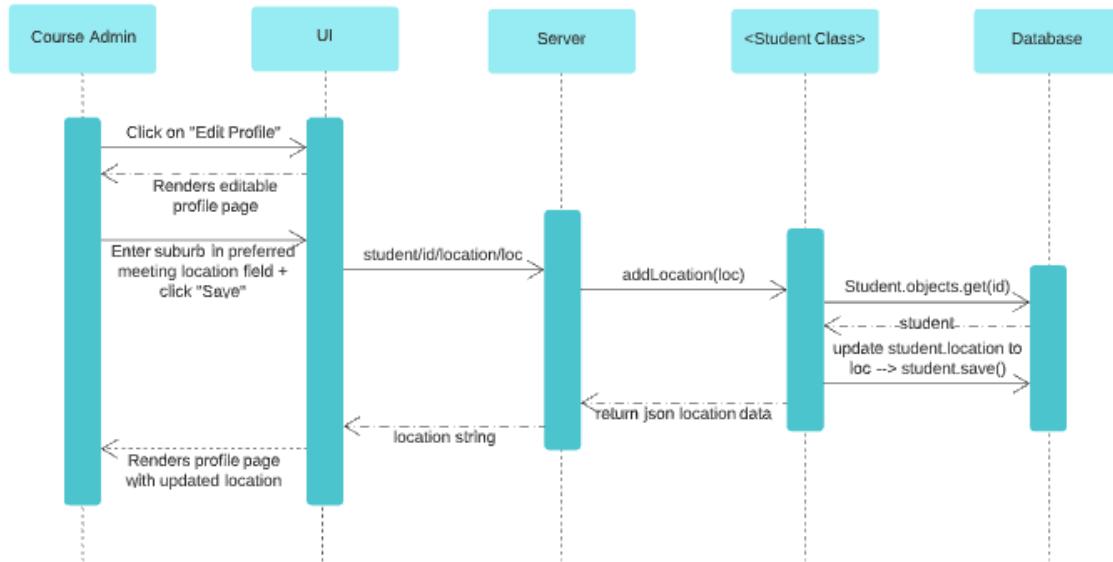


Note: this would also create a notification in the database and attach it to the group owner to notify the group leader that a member has left

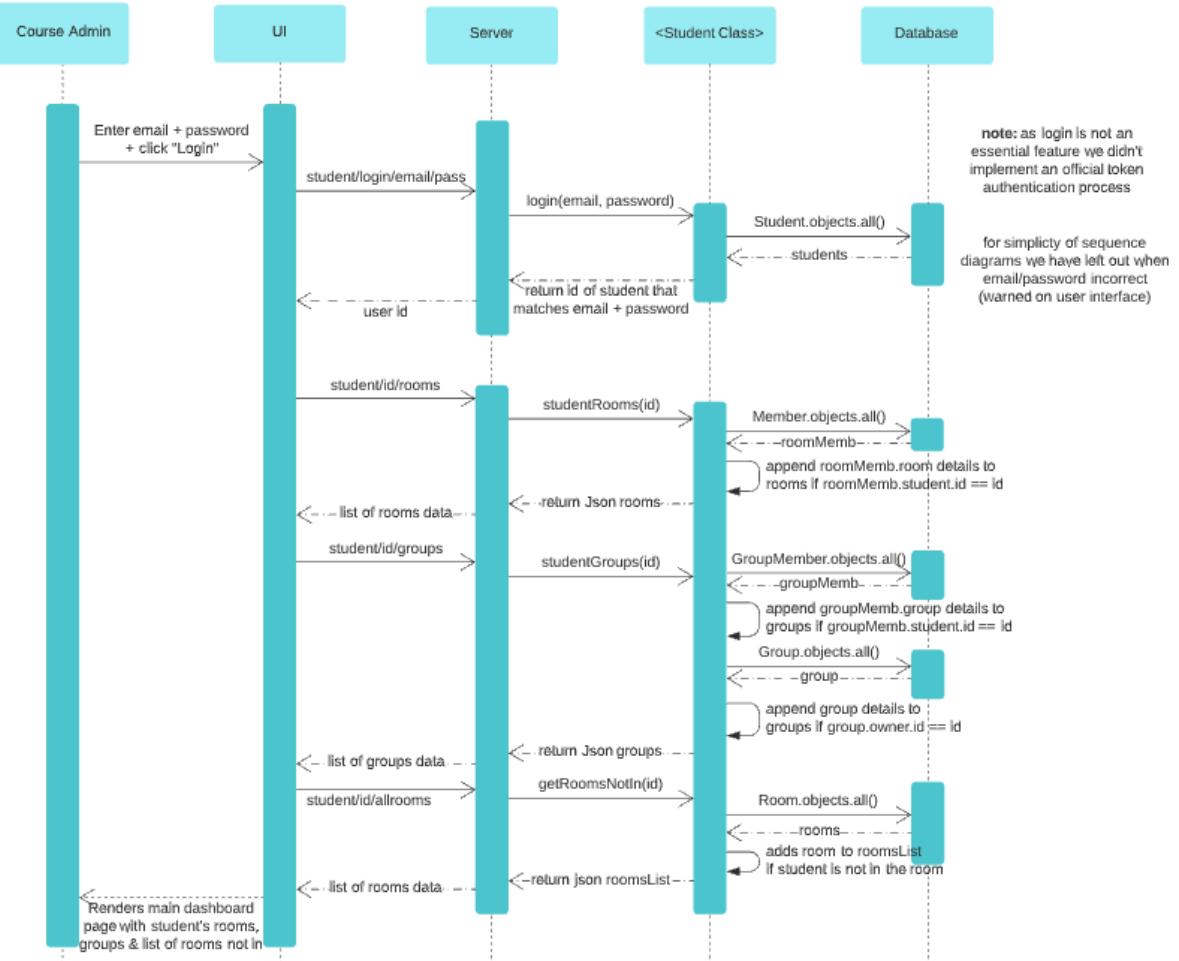


**USER STORY: 2.2.2**  
**(ENTER PREFERRED MEETING LOCATION)**

Initial State:  
Profile Page

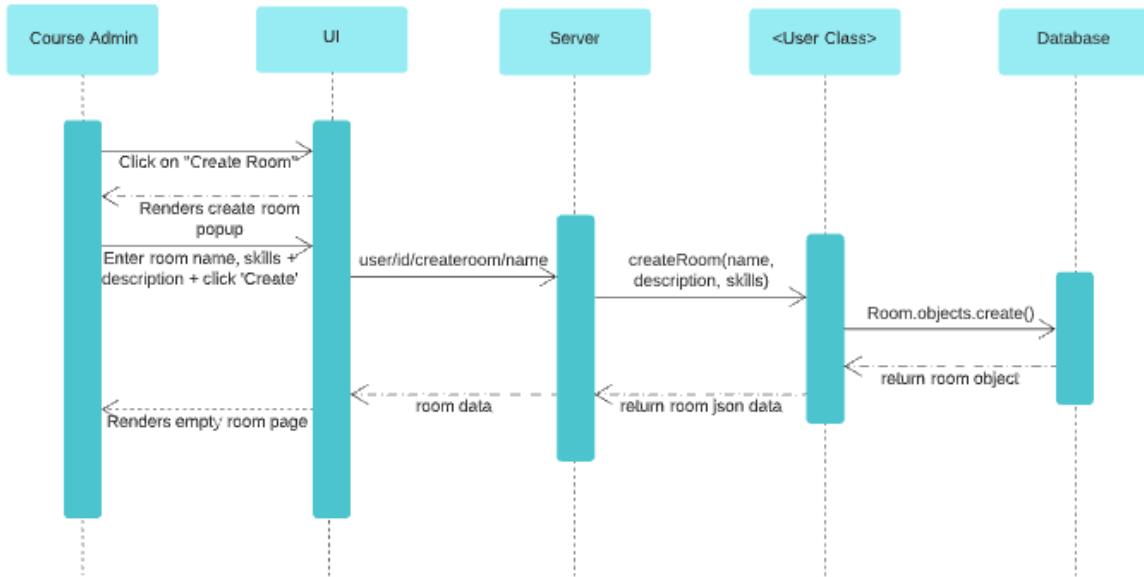


**USER STORY: 8.2.1**  
**HOW THE MAIN DASHBOARD PAGE IS LOADED WHEN A USE LOGS IN TO UNICOLLAB**



**USER STORY: 8.1.1  
(CREATE ROOM)**

**Initial State:**  
Main Dashboard Page





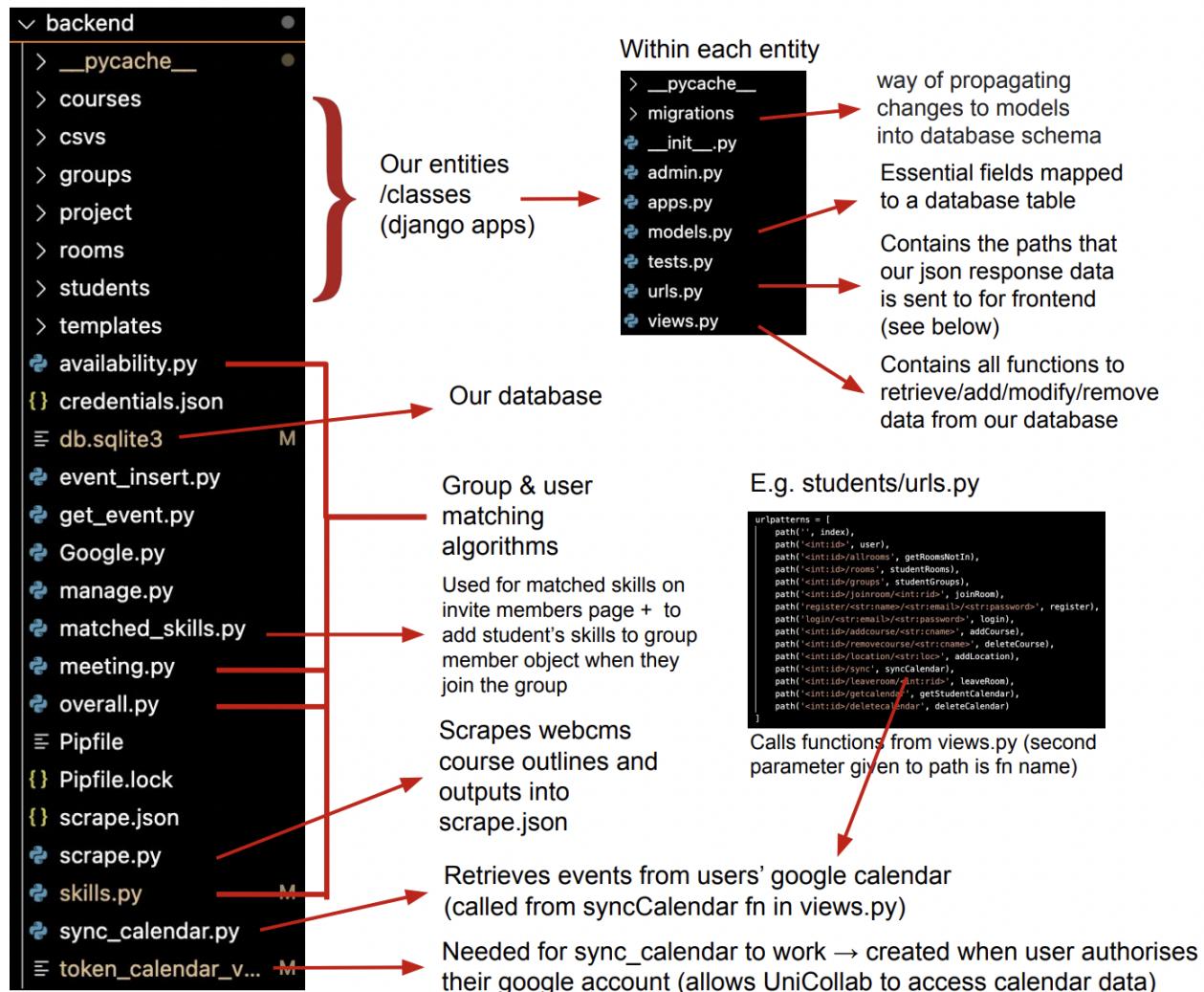
## Other relevant information and key technologies used

### List of Technologies Used

Technology	Description
APIs	<p>Google Maps API:</p> <ul style="list-style-type: none"><li>Allows users to input their location so that the group-matching and member-matching algorithms can make suggestions based on distance between locations</li></ul> <p>Google Calendar API:</p> <ul style="list-style-type: none"><li>UniCollab allows users to sync personal google calendars to their profiles</li><li>Group leaders enter weekly meeting times</li><li>Group-matching and member-matching algorithms compare a group's meeting times to a user's availability, to suggest matches who can attend meetings the most</li></ul>
External Source	<p>WebCMS and CGI course outlines:</p> <ul style="list-style-type: none"><li>Contains relevant text data regarding what type of content is taught in the course</li><li>Useful for doing keyword matching analysis</li></ul>
Backend	<p>Python:</p> <ul style="list-style-type: none"><li>Since the language was known to all group members, this made collaboration and implementation of algorithms easy and succinct</li><li>Experience with making classes and functions</li><li>Easily integratable with Django</li></ul> <p>Django:</p> <ul style="list-style-type: none"><li>Web-source framework allowing us to easily develop a web-application aligned with our database models and software design</li><li>Allows data to be transformed into JSON objects which is passed to the UniCollab frontend</li><li>Compatible with Vue framework through sending the JSON objects through url paths using the django rest framework</li></ul> <p>SQLite3:</p> <ul style="list-style-type: none"><li>Self contained and server-less</li><li>Local demonstration of UniCollab</li><li>Sufficient to develop a working prototype</li><li>Group experience and access to resources</li></ul>
Frontend	<p>Vue:</p> <ul style="list-style-type: none"><li>Group experience with all three frameworks made this a suitable choice</li><li>Suitable for small development/prototypes due to its</li></ul>

	<p>lightweight nature (unlike heavier frameworks such as React)</p> <p><b>Nuxt:</b></p> <ul style="list-style-type: none"> <li>• provide convenient directory structure</li> <li>• Server side rendering can improve user experience</li> </ul> <p><b>Vuetify:</b></p> <ul style="list-style-type: none"> <li>• Easy to use, aesthetic UI component styling library</li> <li>• Excellent integration with Vue</li> </ul>
<b>Other</b>	<p>Beautiful Soup</p> <ul style="list-style-type: none"> <li>• Python web scraping library</li> <li>• Simplifies the process of scraping data from WebCMS course outlines by making it easy to read HTML tags</li> </ul>

## Code compatibility with design



Overview of our backend code system



## Key benefits of design/implementation

Our design and implementation system works very well for our solution through being clearly structured into the different entities required and allowing us to perform all necessary functions to build a functional web-application and utilise our APIs.

A particular benefit of using django as part of our design meant that it made serialization easy both for communication with the frontend and with our database. Our design is extremely flexible and adaptable in how if any changes are needing to be made to any models e.g. adding a new field to one of our entities, we can perform a migration to the database and set a value we would like all existing objects in our database to have. This also means scalability and implementing additional functionality to improve UniCollab would be a very simple process.

We were able to achieve a wide range of functionality with our member objects through having separate models in django for room members and group members that allowed us to access the student and room/group object directly from this entity. This meant when wanting to retrieve members of a particular room/group, instead of having to go through all the rooms or groups we were able to access any information we needed through directly the group member or room object. Another huge benefit of having the group member entity meant that we were able to store the matched skills of a student with the group as a field in the object. This means matched skills would not have to be constantly calculated, and extremely simplified the process of leaving a group as these skills could be retrieved from the relevant group member object and then that object deleted as the student is no longer a group member.

Our algorithms are also a key benefit of our implementation. Through working in django which was easily integratable with python, we were able to easily work with the student and group data and write functions to provide scores and matching for each student/group through utilising APIs. These were able to be called by our views.py files which in turn are called by our vue frontend framework when the relevant action is performed by the user.

---



### 3. Team Organisation + Appraisal of Work

#### Responsibilities of the Team

***Clear illustration of the structure of the team with description of roles and responsibilities.***

#### Team Roles and Responsibilities

	Deliverable 1	Deliverable 2	Deliverable 3	Deliverable 4	Deliverable 5
<b>Haochen</b>		-Database ER diagram relative to problem statement P2, Modelling Table	- Presentation Google calendar api	-backend Google calendar api set up, Coded get_event.py and event_insert.py	
<b>Christina</b>	- High-Fidelity Prototype (Figma) - User stories about groups	- Software architecture diagram - Database UML Diagram - environment set up - frontend pages(no connect to backend ) - refactor sequence diagram	- present the frontend pages	- frontend pages(group, profile, room, dashboard, calendar) - backend implementation of groups and courses. - convert webScraping content to course object - AJAX (communication with backend developer)	-Refind UML, ER & Software Architecture diagrams
<b>Fintan</b>	- High-Fidelity Prototype (Figma) - User stories brainstorm - User stories refinement	- Database UML Diagram - ER Diagram brainstorm - Part 1: Software Architecture write up	- Presentation Script - Presentation slide information		- Refined user stories - Purpose and problems addressed by UniCollab - Reflection brainstorm

		<ul style="list-style-type: none"> <li>- Summary of benefits of design</li> </ul>			<ul style="list-style-type: none"> <li>- Reflection write-up</li> </ul>
<b>James</b>	<ul style="list-style-type: none"> <li>-User stories</li> <li>-Cross reference table</li> </ul>	<ul style="list-style-type: none"> <li>-Sequence diagrams</li> <li>-ER diagrams</li> <li>-Helped architecture diagram</li> <li>-Helped on report</li> </ul>	<ul style="list-style-type: none"> <li>-Business value</li> </ul>	<ul style="list-style-type: none"> <li>-Backend algorithms (availability.py, skills.py, meeting.py, overall.py, scrape.py)</li> </ul>	<ul style="list-style-type: none"> <li>-User stories</li> </ul>
<b>Kenny</b>	<ul style="list-style-type: none"> <li>- User story refinement</li> <li>- Low-Fidelity Prototype (Storyboard interaction graphs + UI Component sketches on Gravit Designer + Lucid charts)</li> <li>- High-Fidelity Prototype brainstorming</li> </ul>	<ul style="list-style-type: none"> <li>- Aiding part 1.1: External data sources</li> <li>- Aiding part 1.4: Choice of platforms</li> <li>- Aiding part 2.1: updating list of user stories</li> <li>- User profile sequence diagrams</li> </ul>	<ul style="list-style-type: none"> <li>- Presentation slides layout</li> <li>- Concept problem, solution &amp; business value</li> </ul>	<ul style="list-style-type: none"> <li>- Front-end contribution (initial pages &amp; button navigation)</li> </ul>	<ul style="list-style-type: none"> <li>- Contributed to creation of purpose/business aspect</li> <li>- Contributed to/refined Features section</li> </ul>
<b>Tiana</b>	<ul style="list-style-type: none"> <li>- High-Fidelity Prototype (Figma)</li> <li>- User stories about calendar availability</li> <li>- UniCollab Glossary</li> </ul>	<ul style="list-style-type: none"> <li>- Sequence diagrams refinement + creation in lucidcharts</li> <li>- Restructured user stories based on feedback for usage of more APIs</li> </ul>	<ul style="list-style-type: none"> <li>- Constructing figma prototyping for demo 'good flow'</li> <li>- scripting demo part</li> </ul>	<ul style="list-style-type: none"> <li>- Backend implementation of students, rooms + creating fns in views.py and paths in urls.py for communication with frontend</li> </ul>	<ul style="list-style-type: none"> <li>- Features table + Achieved components section</li> <li>- sequence diagrams refinement</li> </ul>



## How did the project go?

**Relevant and appropriate reflection on the experience during the project. Information about their experience.**

**What they have learnt; positive and/or negatives.**

**Overall Summary:** Our team is very satisfied with our teamwork and communication throughout this project, which allowed us to establish an application prototype that we are proud of. The SENG2021 course, along with COVID-19 university adjustments allowed us to learn a wide array of new information, teamwork skills, and software practices that we will take on for future university projects and beyond.

### 1. Working as a group

Being a team-based project course, SENG2021 taught us a lot about what it means to be an effective team member. We learnt that organisation is key, as it allows us to remain clear on what tasks have/have not been completed and updated on all teammates' duties. We organised ourselves very efficiently, by using multiple Discord channels, each one catered to different project components, such as "Backend", 'High-Fidelity', 'Sequence Diagrams' etc. Our decision making was also very efficient, as we hosted frequent meetings and discussed our current problems, proposed solutions and built upon each other's advice to adhere to a common solution. This also boosted our decision making and feedback skills, which are critical to team performance and hence establishing a successful final result.

### 2. Differing levels of experience

Due to our differing levels of programming and software experience, we needed to communicate effectively to ensure that everyone was comfortable with their duties and able to perform them sufficiently. We hosted meetings to discuss our computing experience, and proposed possible combinations of teammates to work on different tasks. Within each group, we then set goals, which were effectively adhered to throughout the project for every deliverable.

### 3. Development of a Web Application (Software Practices)

The actual content of the course and its structure (Deliverables) allowed us to learn an incredible amount regarding Software Design Practices over such a short period of design. Firstly, we learnt that the design phase is one of the most critical portions of the process, as it allows team members to form a cohesive, unified idea of what their end product may look like, so that all members can work towards a common goal. In saying this, we learnt that initial ideas are not always concrete, which was especially true when we needed to drastically change our application after receiving feedback from our new mentor. After splitting into smaller 'teams', we learnt a lot about the front and back ends of web applications, especially how to effectively integrate the two together. Since none of us had prior experience with Django, it was a collective learning experience to use it properly and link our Python backend with the Vue frontend.

---



## Issues and Problems Encountered

***What they have learnt; positive and/or negatives.***

### **1. COVID**

The severity of COVID-19 has obvious implications upon the university student experience, forcing all mentoring sessions and team meetings to be held online. Due to this, we had to establish meeting times in advance, even when members could not all be present. Mentoring sessions were made difficult, since mentors could not physically meet with us and discuss ideas in further detail due to the 20 minute time limitation. Organising team duties was difficult, due to differing locations and hence time zones, thus reaching deadlines was an ongoing concern.

### **2. Mentoring Change**

Our change of mentor in Week 4 had large implications on our design process. Notably, our previous mentor (Decler) said that our idea and API usage was adequate in order to establish a unique and viable product. However, Fethi prompted us to expand upon our initial design, asking for more API usage to be combined so that UniCollab is a more unique platform. In hindsight, this was excellent advice, and we believe it to be a key reason as to our success, however at the time this meant rewriting user stories and diagrams. As a result, we requested an extension for the Deliverable 2 deadline so that we could accomplish this task.

### **3. Levels of Experience**

The levels of software and programming experience differed amongst all team members, which required us to effectively divide work loads and tasks. In some cases, we needed to reshuffle the delegation of tasks, which required a change of group deadlines and individual workloads.

### **4. Git merge difficulties**

One problem we encountered when coding was a large amount of merge conflicts that were almost impossible to fix. This took much time to resolve and set us back in progress. However, we were able to solve this through copying across files from the up to date branch and performing git force push and pulls.

### **5. Django**

Most of the backend developers had actually not had much experience with Django and found it quite difficult to know how to get started. After many youtube tutorials and discussions within the team we were able to start setting up models and views

### **6. Communication of ideas**

Due to a volatile development approach, our team experienced difficulties expressing the exact usage and purpose of Unicollab to mentors, as its core functions were rapidly changing/improving to become a more stable application. As university students, we immediately and collectively understood the lack of applications which allow for efficient group



formation. Hence, our vision was in sync, and we were able to communicate ideas effectively amongst ourselves.

## What would be done differently?

### ***Suggestions for Improvements.***

#### **1. Put a bigger emphasis on code design from the beginning.**

In hindsight, our backend code design was relatively disjoint and unorganised. We tended to use various forms of data storage, including Python dictionaries and also SQLite, in addition to dummy data we provided to ensure that our demonstration could run smoothly. If we were to redo this assignment knowing what we know now, we would have rethought some of our features and design concepts, in a way that we could easily store all information in SQLite so that Python could be reserved for programming. Moreover, our team would prioritise the construction of our script architecture sooner, in order to remain timely with the deliverable due-dates, and produce work more effectively.

#### **2. Get started on backend implementation earlier.**

Our frontend team were very prompt in the early implementation phase, where they began to form interactive front-end pages in Week 7. However, due to university commitments and difficulty understanding the relationship between Python and Django, it took some time to begin implementing the backend of UniCollab. This meant that frontend pages relied on dummy data to be tested and improved, and then changed once more after the implementation and subsequent testing of our backend architecture. If we could improve upon our work, we would ensure to start developing the backend processes much earlier.

#### **3. Perform user testing with our high fidelity prototypes to see what could be improved about the user interfaces from a usability perspective.**

One of our design flaws that was noted after the Deliverable 3 demonstration was that our UniCollab interface used a mix of mobile and desktop user interface components. Things like scrollable lists next to large buttons made this statement accurate. In later stages, we made adjustments to our interface layout and asked external users to provide feedback. Essentially, we were able to adhere to a more desktop-friendly interface, but in hindsight we should have asked external sources to review our high-fidelity prototype, which would have saved us a lot of time.

#### **4. Additional Mentor Meetings**

If we could do over this project, a key thing we would change is to request and participate in more mentor sessions with our mentor, Fethi. We started requesting external meetings (outside of our scheduled time slot) around Week 8, which allowed us to fine-tune many aspects of our application and final report (especially the user stories). This proved to be very beneficial to our understanding, and hence we should have been doing so for early deliverables.



**Overall:** Ultimately, we are very happy with how we performed throughout the duration of the project. Due to things like frequent meetings, clear and efficient communication and effective time-management, we as a team were able to create and demonstrate a product that we feel very satisfied with. In fact, we had additional features and functions to display, but due to the time restraint, we had to select fewer items to showcase. In saying that, there were definitely things that we could have done better, mostly actions that would have saved time for us in later deliverables.

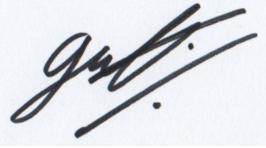
---

## 4. Peer Group Assessment Form

**Project Final Submission**

**Due Date:** 26/04/2021 17:00

**Team Name:** Attack on HD

Student ID	Student Name	Contribution	Signature
z5259202	Tiana Douroudis	100	
z5257532	Fintan O'Shea	100	
z5368339	Kanishka Yamani	100	
z5183932	Xiaoyan(Christina) Xie	100	
z5255218	Haochen Shi	70	
z5254551	Xiang (James) Ji	100	