Capstone Project Report

Capstone Project Unit Management System

Group 14 - Yoon Kim, Sean Tria, Seyoung Kim, Yeongje Park

Executive Summary

Current management system for capstone projects is challenging to use and very inefficient. An optimal solution to this is to create a new comprehensive and optimised system. The new system, a web service that covers all of the stakeholders for capstone unit, will ultimately replace most of the current system including project application and management and will work alongside the QUT Blackboard. Technical requirements for this new system and how it will be replacing the current system were thoroughly explained throughout this report. A number of prototypes and technical demonstration mockup were successfully created in order to examine the feasibility of the project. The new web based service will have a similar look to the QUT Blackboard and the most of its users will be able to navigate through the website with ease. Sprint plan and other methods were used to record the estimated time for the final release of the project. Using these methods, our team decided that it is safe to assume that the final product will be successfully delivered by the end of given time frame; 16 Weeks (13 teaching weeks and 2 weeks mid semester holidays plus an extra week during Week 0 if deemed).

1 Project Description

1.1 Real World Problem

IFB398/IFB399 Capstone is a unit where every IT student in QUT will eventually have to undertake before graduating. However, the current project management system for the unit has a number of flaws which makes the system incredibly inefficient, disarranged, and error-prone. The unit currently uses too many different programs and services to manage different assets. For example, google forms to record fortnight reflection, Inplace for project application which unit coordinators and tutors have very minimum control over; and QUT BlackBoard and Excel spreadsheets for project allocation and management. The current system also forces its stakeholders; students, unit coordinators, tutors and industry partners; to use obsolete, scattered services to manage their projects. For instance, project supervisors; including industry partners and some of the unit coordinators throughout the university; currently use Inplace to propose projects and manage project applications. Team allocation and marking assessments are currently all done manually in a very inefficient manner using Excel spreadsheets. When the project applications are due, cover letters and CVs are downloaded from Inplace and saved locally for project allocation and marking purposes. When the review is finished, teams are then allocated to projects manually according to the team rankings. A new, efficient and error-free system is required to replace the current system and how the system is going to replace the services currently used needs to be also determined.

1.2 Project Goal

The goal of this project is to develop a comprehensive and optimised web based service that will work alongside with BlackBoard, which will ultimately replace most of the features of the current system and significantly benefit the stakeholders by improving the accessibility and usability. The service will also greatly improve the navigability between different assessments. Technical requirements and explanation of the project design to achieve this goal will be further discussed in section 1.3 and 4 respectively.

1.3 Project Requirements

To determine which of the current system should be replaced by new system, we firstly created a number of user stories to cover all of use cases. Then we made a list of the technical requirements of the project.

1.3.1 User Stories

Refer ***Appendix A*** for user stories

1.3.2 Technical Requirements

i. Students must be able to sign up to the web service, with a number of prefilled personal information such as name, GPA and major during Week 0 – 1.

ii. Students must be able to form teams and add group members on their own until the end of Week 1.

iii. The unit coordinator must be able to add,edit and remove users including students, tutors, industry partners and other unit coordinators.

iv. The users must be able to view/edit their profiles

v. The users must be able to reset passwords

vi. The unit coordinator must be able to see a list of students who are not in a team at the end of the week 1 and generate teams automatically, populating the database with student details at the end of Week 1 (Groups can be generated randomly or by a given algorithm such as similar GPA, different combinations of majors or other preferences).

vii. The unit coordinator must be able to edit teams.

viii. The unit coordinator and industry partners must be able to upload projects and students must be able to view and apply for them during Week 2.

ix. Students must be able to upload Cover letters and CVs.

x. One of the group members must be able to sign up the team for fortnightly meetings and presentations.

xi. One of the group members must be able to rank projects on the behalf of the team during Week 3.

xii. Project supervisors must be able to rank teams during Week 4

xiii. Students must be able to submit assessments including fortnight reflection and edit them before the deadline.

xiv. Students must be able to view previous assessments and marks.

xv. The unit coordinator and tutors must be able to download and mark different assessments.

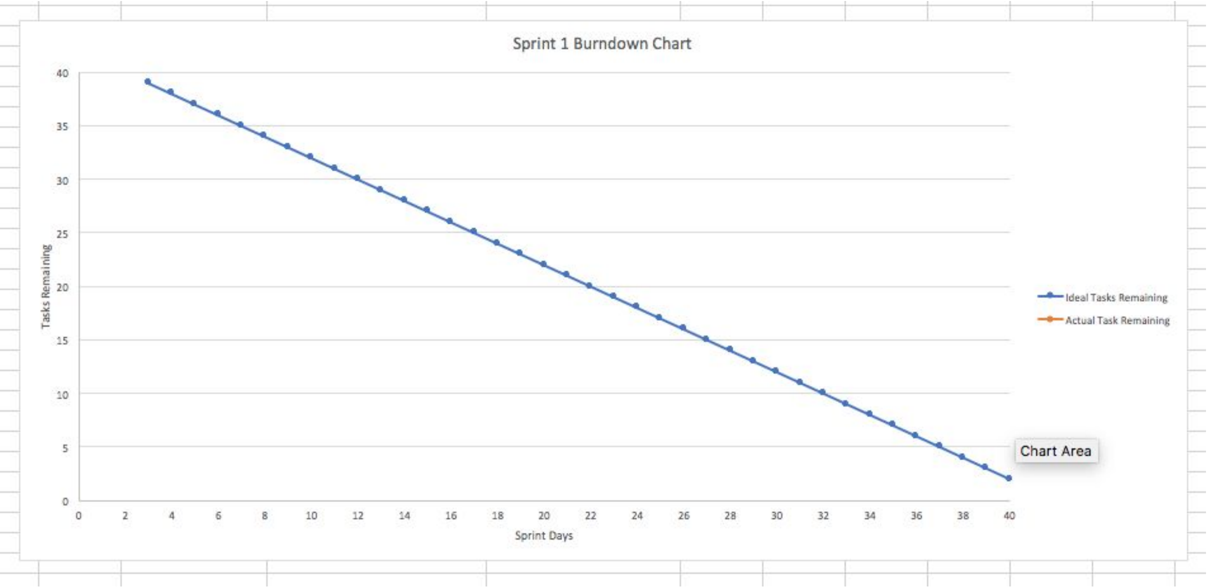
xvi. Tutors and Students must be able to see the meeting time that is allocated to them.

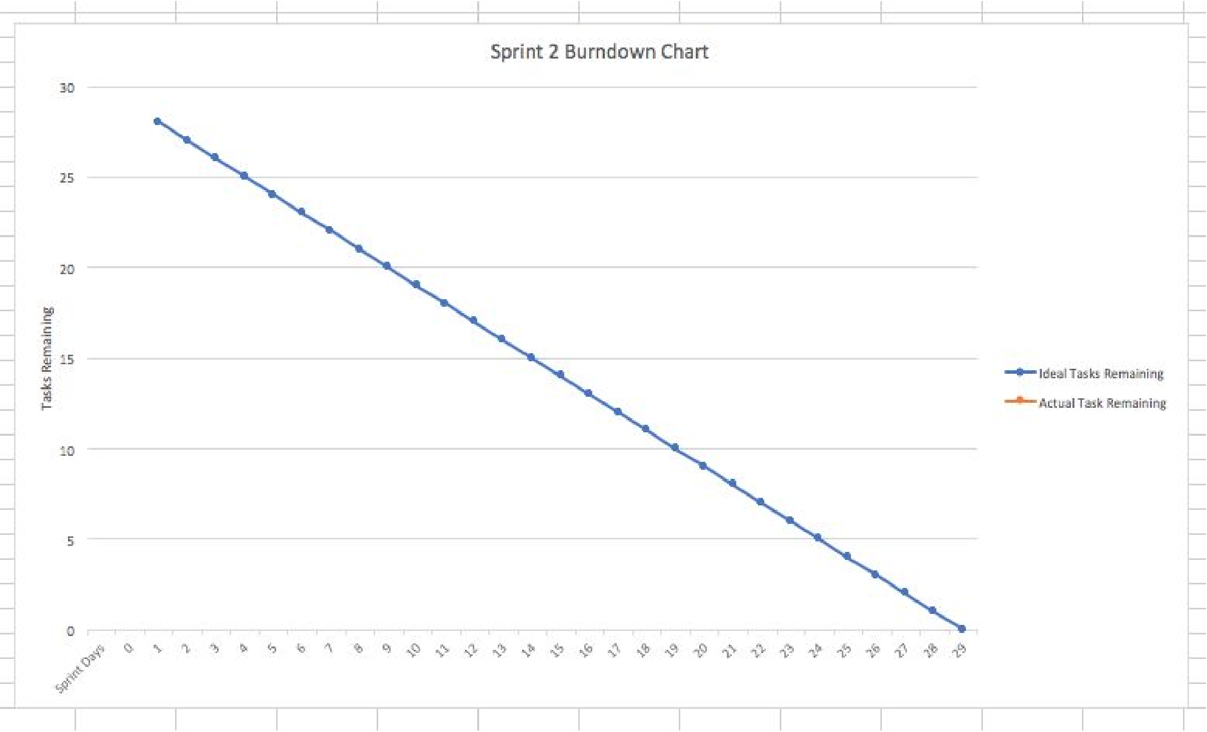
xvii. Users must be able to see their and other project related users' details.

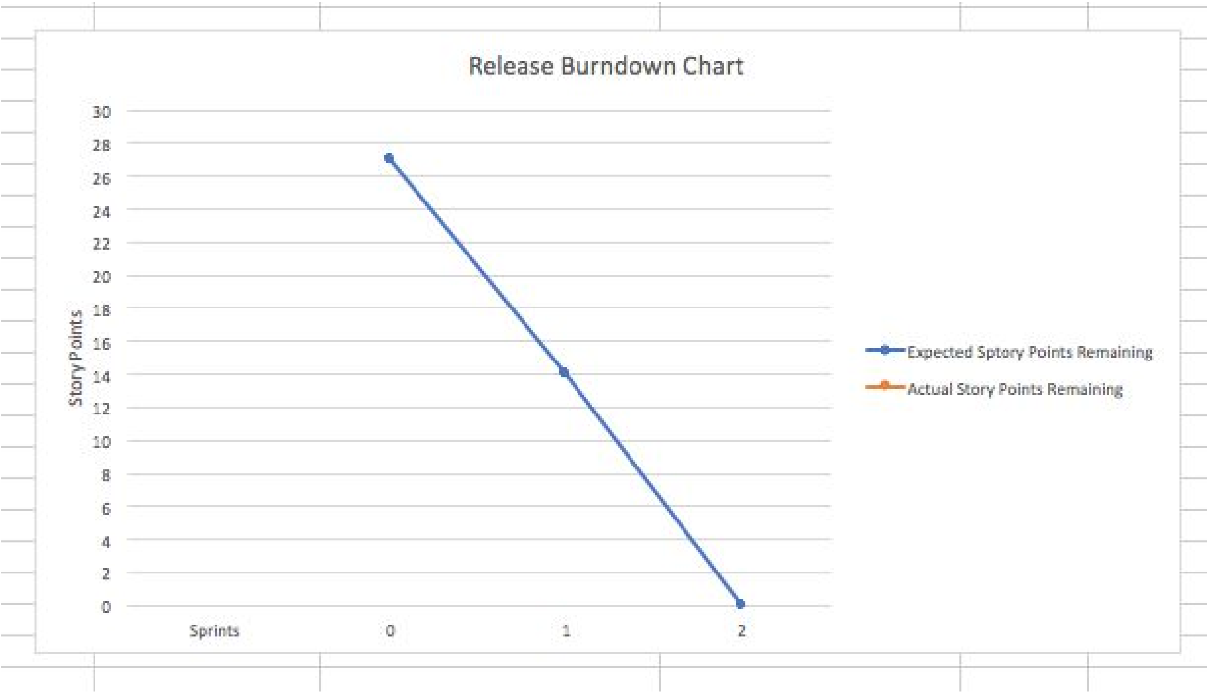
xviii. At the end of the semester, the unit coordinator must be able to release cohort 2 and download related information (Student details and etc…)

2 Project Feasibility

2.1 Estimated Hours to Complete







To keep a track and record of the progress of work, we have decided to use the burndown chart method. This will not only keep record of the progress, will also tell us the remaining time/hours left for completion. A Sprint burndown chart is a graphical representation of work left and the actual work left to do versus time. A Release burndown chart is similar to a sprint burndown chart, but it shows the project progress of each releases.

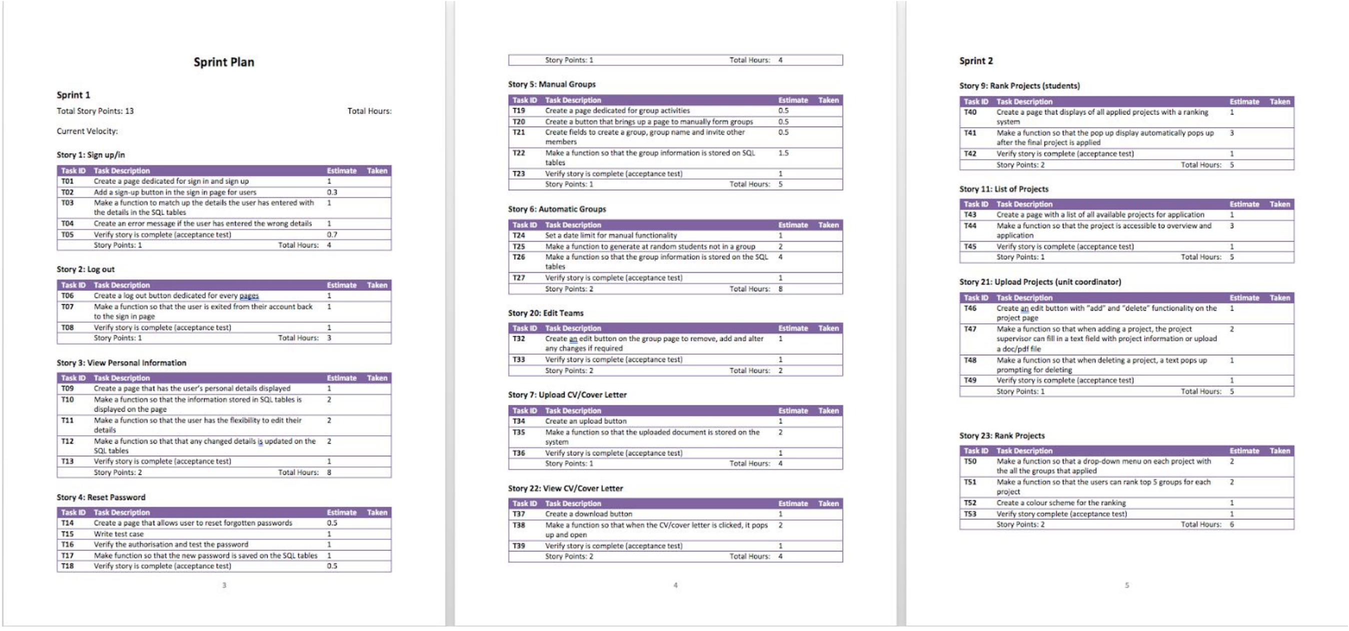
The total number of tasks for sprint one is 39 tasks. This then corresponds to 39 working days. Similar to sprint two, with 28 tasks, it corresponds to 28 working days. There is a total of two releases with a total of 27 story points in this development project and it will also be mentioned in the sprint plan.

2.2 Technical Demonstration

Alongside the initial prototypes, we have designed a database in Python Django from ERD and DB tables as a proof of concept and integrated that into one of our high fidelity prototype webpages. Refer ***Appendix B*** for the technical demonstration.

3 Project Management and Plan

3.1 Sprint/Release Plan



Sprint Plan

Sprint 1

Total Story Points: 13 Total Hours:

Current Velocity:

Story 1: Sign up/in

Task ID	Task Description	Estimate	Taken
T01	Create a page dedicated for sign in and sign up	1	
T02	Add a sign-up button in the sign in page for users	0.3	
T03	Make a function to match up the details the user has entered with the details in the SQL tables	1	
T04	Create an error message if the user has entered the wrong details	1	
T05	Verify story is complete (acceptance test)	0.7	
	Story Points: 1 Total Hours:	4	

Story 2: Log out

Task ID	Task Description	Estimate	Taken
T06	Create a log out button dedicated for every pages	1	
T07	Make a function so that the user is exited from their account back to the sign in page	1	
T08	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	3	

Story 3: View Personal Information

Task ID	Task Description	Estimate	Taken
T09	Create a page that has the user's personal details displayed	1	
T10	Make a function so that the information stored in SQL tables is displayed on the page	2	
T11	Make a function so that the user has the flexibility to edit their details	2	
T12	Make a function so that that any changed details is updated on the SQL tables	2	
T13	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	8	

Story 4: Reset Password

Task ID	Task Description	Estimate	Taken
T14	Create a page that allows user to reset forgotten passwords	0.5	
T15	Write test case	1	
T16	Verify the authorisation and test the password	1	
T17	Make function so that the new password is saved on the SQL tables	1	
T18	Verify story is complete (acceptance test)	0.5	

3

	Story Points: 1 Total Hours:	4	

Story 5: Manual Groups

Task ID	Task Description	Estimate	Taken
T19	Create a page dedicated for group activities	0.5	
T20	Create a button that brings up a page to manually form groups	0.5	
T21	Create fields to create a group, group name and invite other members	0.5	
T22	Make a function so that the group information is stored on SQL tables	1.5	
T23	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

Story 6: Automatic Groups

Task ID	Task Description	Estimate	Taken
T24	Set a date limit for manual functionality	1	
T25	Make a function to generate at random students not in a group	2	
T26	Make a function so that the group information is stored on the SQL tables	4	
T27	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	8	

Story 20: Edit Teams

Task ID	Task Description	Estimate	Taken
T32	Create an edit button on the group page to remove, add and alter any changes if required	1	
T33	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	2	

Story 7: Upload CV/Cover Letter

Task ID	Task Description	Estimate	Taken
T34	Create an upload button	1	
T35	Make a function so that the uploaded document is stored on the system	2	
T36	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	4	

Story 22: View CV/Cover Letter

Task ID	Task Description	Estimate	Taken
T37	Create a download button	1	
T38	Make a function so that when the CV/cover letter is clicked, it pops up and open	2	
T39	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	4	

4

Sprint 2

Story 9: Rank Projects (students)

Task ID	Task Description	Estimate	Taken
T40	Create a page that displays of all applied projects with a ranking system	1	
T41	Make a function so that the pop up display automatically pops up after the final project is applied	3	
T42	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	5	

Story 11: List of Projects

Task ID	Task Description	Estimate	Taken
T43	Create a page with a list of all available projects for application	1	
T44	Make a function so that the project is accessible to overview and application	3	
T45	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

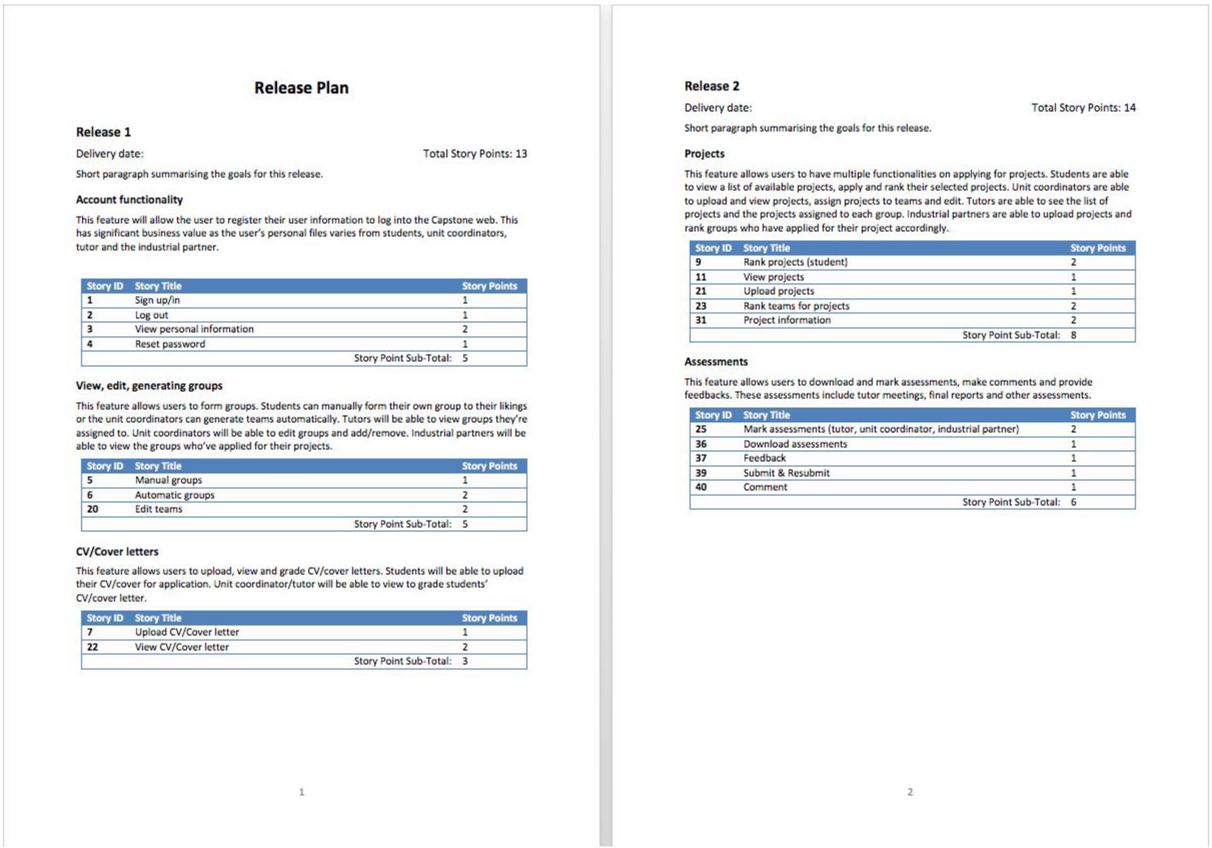
Story 21: Upload Projects (unit coordinator)

Task ID	Task Description	Estimate	Taken
T46	Create an edit button with "add" and "delete" functionality on the project page	1	
T47	Make a function so that when adding a project, the project supervisor can fill in a text field with project information or upload a doc/pdf file	2	
T48	Make a function so that when deleting a project, a text pops up prompting for deleting	1	
T49	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

Story 23: Rank Projects

Task ID	Task Description	Estimate	Taken
T50	Make a function so that a drop-down menu on each project with the all the groups that applied	2	
T51	Make a function so that the users can rank top 5 groups for each project	2	
T52	Create a colour scheme for the ranking	1	
T53	Verify story complete (acceptance test)	1	
	Story Points: 2 Total Hours:	6	

5



Release Plan

Release 1

Delivery date: Total Story Points: 13

Short paragraph summarising the goals for this release.

Account functionality

This feature will allow the user to register their user information to log into the Capstone web. This has significant business value as the user's personal files varies from students, unit coordinators, tutor and the industrial partner.

Story ID	Story Title	Story Points
1	Sign up/in	1
2	Log out	1
3	View personal information	2
4	Reset password	1
	Story Point Sub-Total:	5

View, edit, generating groups

This feature allows users to form groups. Students can manually form their own group to their likings or the unit coordinators can generate teams automatically. Tutors will be able to view groups they're assigned to. Unit coordinators will be able to edit groups and add/remove. Industrial partners will be able to view the groups who've applied for their projects.

Story ID	Story Title	Story Points
5	Manual groups	1
6	Automatic groups	2
20	Edit teams	2
	Story Point Sub-Total:	5

CV/Cover letters

This feature allows users to upload, view and grade CV/cover letters. Students will be able to upload their CV/cover for application. Unit coordinator/tutor will be able to view to grade students' CV/cover letter.

Story ID	Story Title	Story Points
7	Upload CV/Cover letter	1
22	View CV/Cover letter	2
	Story Point Sub-Total:	3

1

Release 2

Delivery date: Total Story Points: 14

Short paragraph summarising the goals for this release.

Projects

This feature allows users to have multiple functionalities on applying for projects. Students are able to view a list of available projects, apply and rank their selected projects. Unit coordinators are able to upload and view projects, assign projects to teams and edit. Tutors are able to see the list of projects and the projects assigned to each group. Industrial partners are able to upload projects and rank groups who have applied for their project accordingly.

Story ID	Story Title	Story Points
9	Rank projects (student)	2
11	View projects	1
21	Upload projects	1
23	Rank teams for projects	2
31	Project information	2
	Story Point Sub-Total:	8

Assessments

This feature allows users to download and mark assessments, make comments and provide feedbacks. These assessments include tutor meetings, final reports and other assessments.

Story ID	Story Title	Story Points
25	Mark assessments (tutor, unit coordinator, industrial partner)	2
36	Download assessments	1
37	Feedback	1
39	Submit & Resubmit	1
40	Comment	1
	Story Point Sub-Total:	6

2

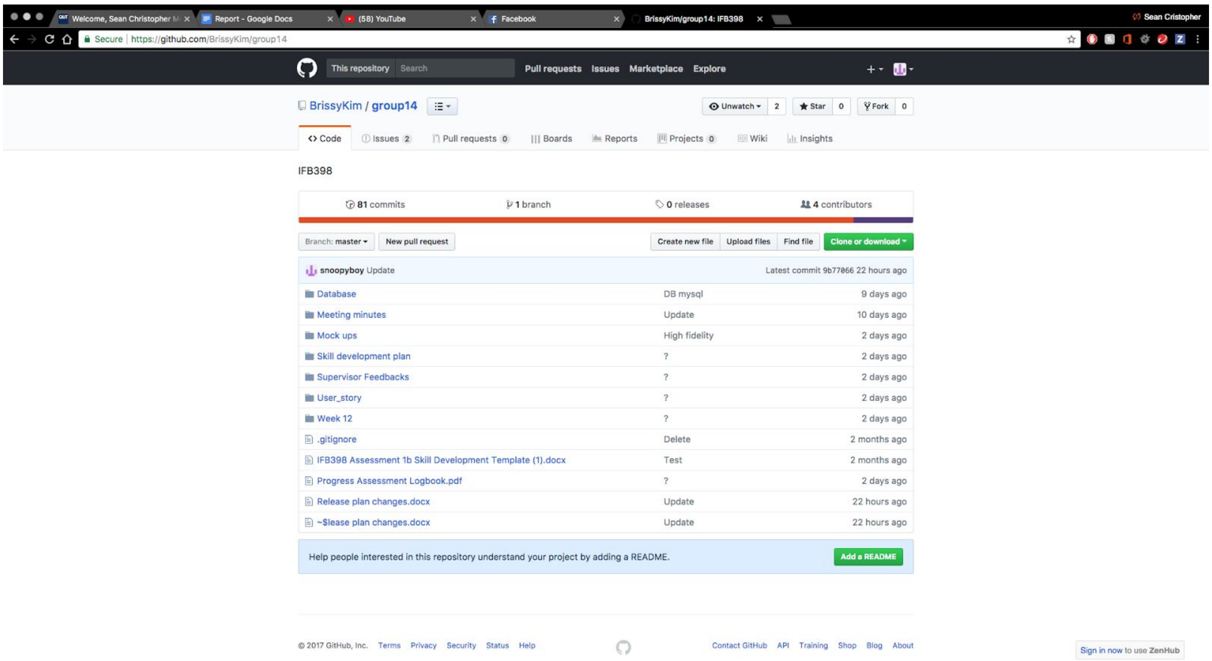
For this project development our group has decided to use the skills we learnt throughout different units. By using the release/sprint plan method, we were able to thoroughly organise and plan in a cohesive and precise manner ahead of the time. The release/sprint plan is composed of two characteristics:

1. Release plan - this consists of user stories with story points that are grouped into one subheading. A story point is equivalent to four hours of work.
2. Sprint plan - this consists of a detailed breakdown of each user story and its functionality with tasks and properties with an estimate number of minutes/hours required to finish.

These are then collectively arranged depending on how a group would work. For our scenario, we have two releases and two sprints, both equally divided to evenly spread the workload. Refer ***Appendix C*** for full sprint/release plan.

3.2 GitHub

A repository was necessary to store all our documents and files. For this development we have chosen to use GitHub as our repository. in our repository we have grouped into different folders to keep files and documents organised rather than being scattered. This is accessible to all members of the group.



IFB398

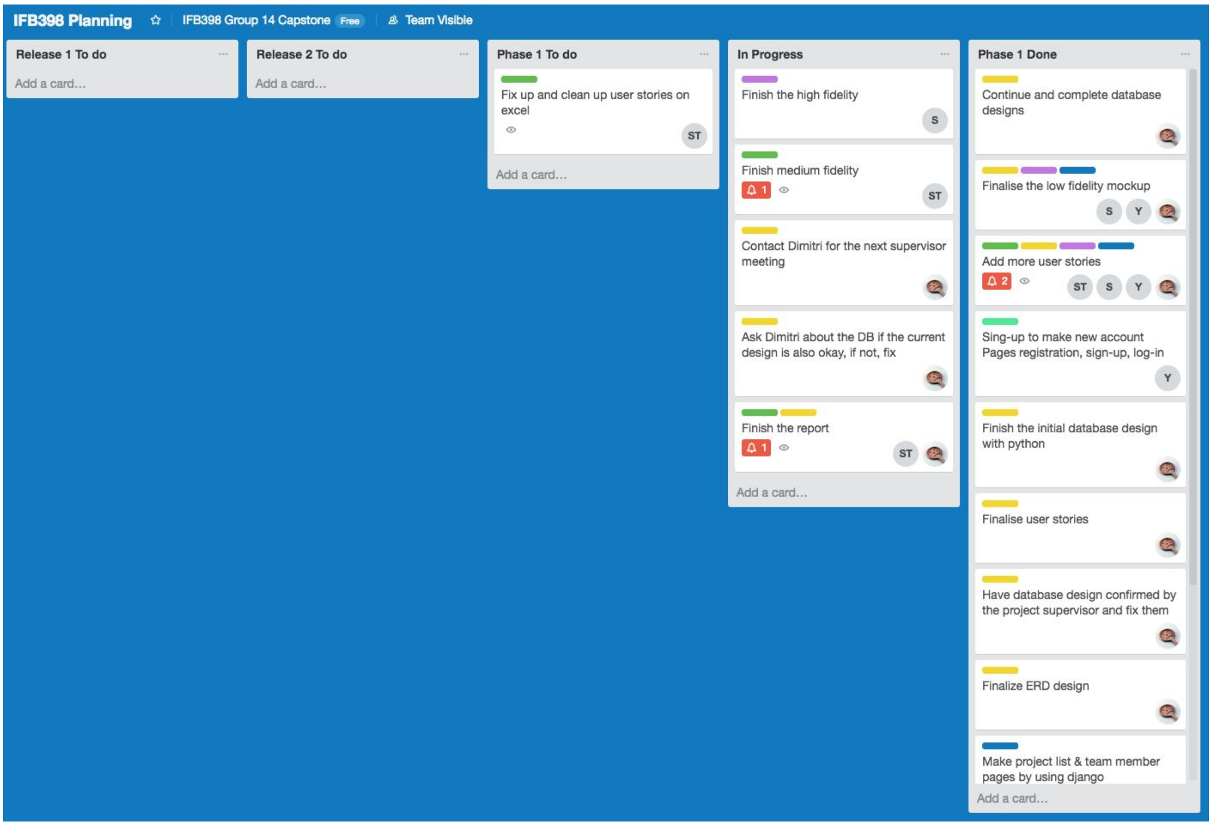
81 commits   1 branch   0 releases   4 contributors

snoopyboy Update   Latest commit 9b77066 22 hours ago

Database	DB mysql	9 days ago
Meeting minutes	Update	10 days ago
Mock ups	High fidelity	2 days ago
Skill development plan	?	2 days ago
Supervisor Feedbacks	?	2 days ago
User_story	?	2 days ago
Week 12	?	2 days ago
.gitignore	Delete	2 months ago
IFB398 Assessment 1b Skill Development Template (1).docx	Test	2 months ago
Progress Assessment Logbook.pdf	?	2 days ago
Release plan changes.docx	Update	22 hours ago
~$lease plan changes.docx	Update	22 hours ago

Help people interested in this repository understand your project by adding a README.

3.3 Trello



IFB398 Planning ☆ IFB398 Group 14 Capstone Free Team Visible

Release 1 To do

Add a card...

Release 2 To do

Add a card...

Phase 1 To do

- Fix up and clean up user stories on excel — ST

Add a card...

In Progress

- Finish the high fidelity — S
- Finish medium fidelity — 1 — ST
- Contact Dimitri for the next supervisor meeting
- Ask Dimitri about the DB if the current design is also okay, if not, fix
- Finish the report — 1 — ST

Add a card...

Phase 1 Done

- Continue and complete database designs
- Finalise the low fidelity mockup — S Y
- Add more user stories — 2 — ST S Y
- Sing-up to make new account Pages registration, sign-up, log-in — Y
- Finish the initial database design with python
- Finalise user stories
- Have database design confirmed by the project supervisor and fix them
- Finalize ERD design
- Make project list & team member pages by using django

Add a card...

We have decided to use Trello as our task log system to keep a track of our progress. This contains what needs to be done, what is in progress and what has been done. It has the flexibility to assign a particular task to individuals or groups of people.

4 Project Design

4.1 Languages, APIs and Hosting Services

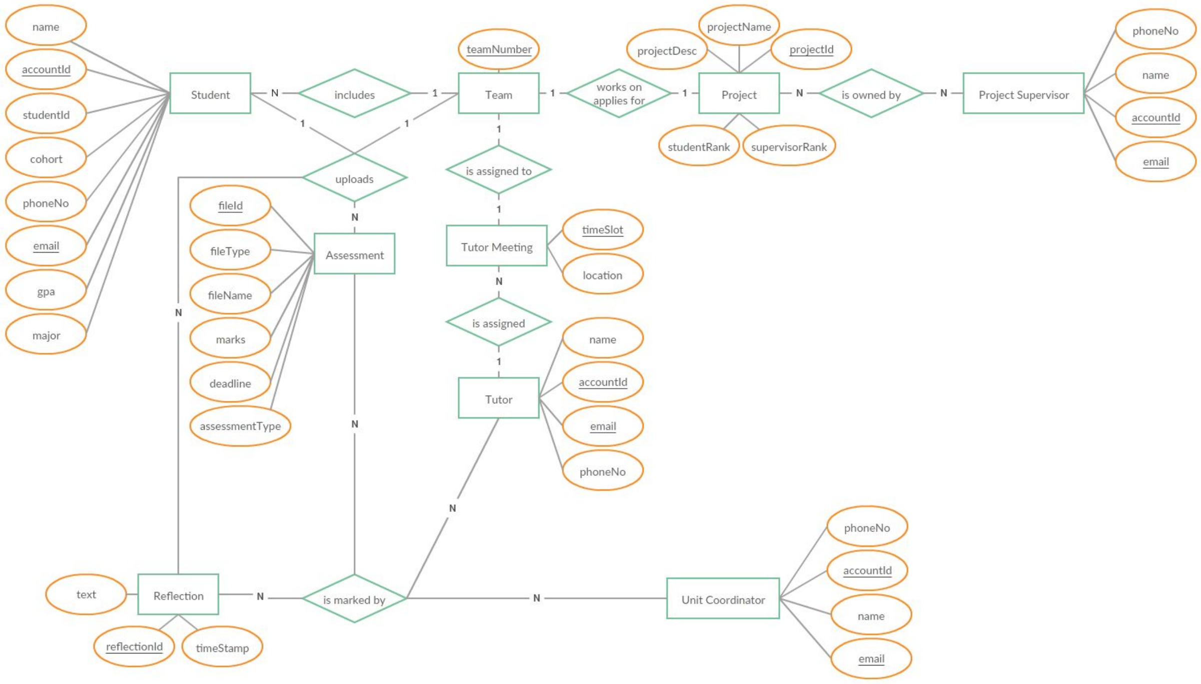
Out of three very popular languages for web development; Python Django, Ruby on Rails and PHP, for our project, we have decided that Python Django has a slight edge over its competitors. Even though PHP is a conventional language for web development, for a database heavy service like our project, either Python DJango or Ruby on Rails would be a better option. Both Python Django and Ruby on Rails have great readability and flexibility, however, Python DJango was chosen as most of our group members are familiar with Python language, to reduce challenges we could run into when developing with an unfamiliar language.

Since our project is nonprofit and for educational purposes, we had a number of open source APIs and libraries to choose from. For design and file upload, we will be using W3.CSS, Bootstrap and DropzoneJS. W3.CSS and Bootstraps are widely used, free, open source css libraries which support all browsers and devices. DropzoneJS is a library for file uploading which supports both upload by drag and drop and file explorer and was chosen because of its compactness and programmability. DropzoneJS can also be used in a modular form where it is needed. A third party authentication API, such as OAuth would also be used for provide better security.

During the development phase, the website will be locally hosted, however, at the end of the development phase, third party web hosting services such as C9 and Azure could be used to prove the functionality of the web service on a larger scale.

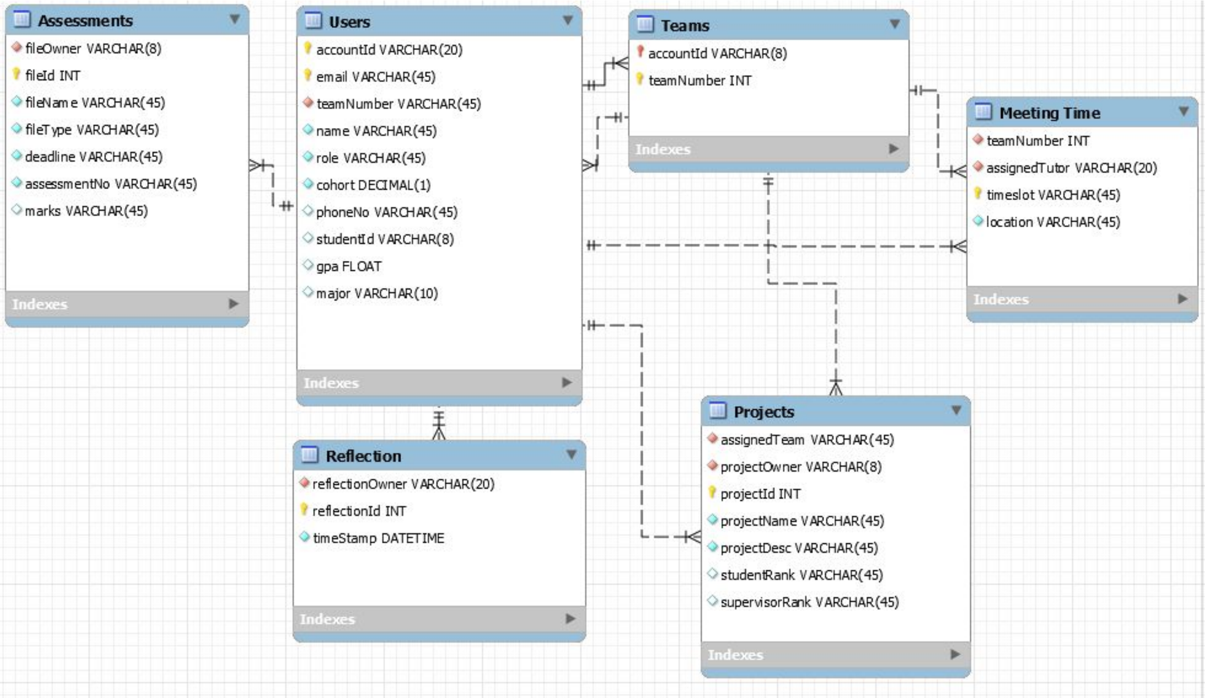
4.2 Database design

Entity Relationship Diagram



We firstly created Entity Relationship Diagram (ERD) to clarify the different privileges of different users before designing the initial database tables to make sure the design is clear and sound. The ERD outlines a number of entities, including stakeholders, projects and assessments. Each entity has attributes that serve the purpose of recording information about the entity.

Database Tables



From use case scenarios and ERD, we then created initial database tables which consist a number of primary keys using mySQL workbench . These tables were utilised to determine the types of attributes and then used in the technical demonstration prototype using Python Django.

5 Prototypes

5.1 Low/Medium Fidelity prototypes

Our team created a number of low and medium fidelity prototypes which outline the functionality and the design of what would be the final prototype for our project. A number of different iterations of Low Fidelity prototypes were drawn in paper, then Medium fidelity prototypes were created to match the design of our web service to the QUT BlackBoard, for better usability and navigability.

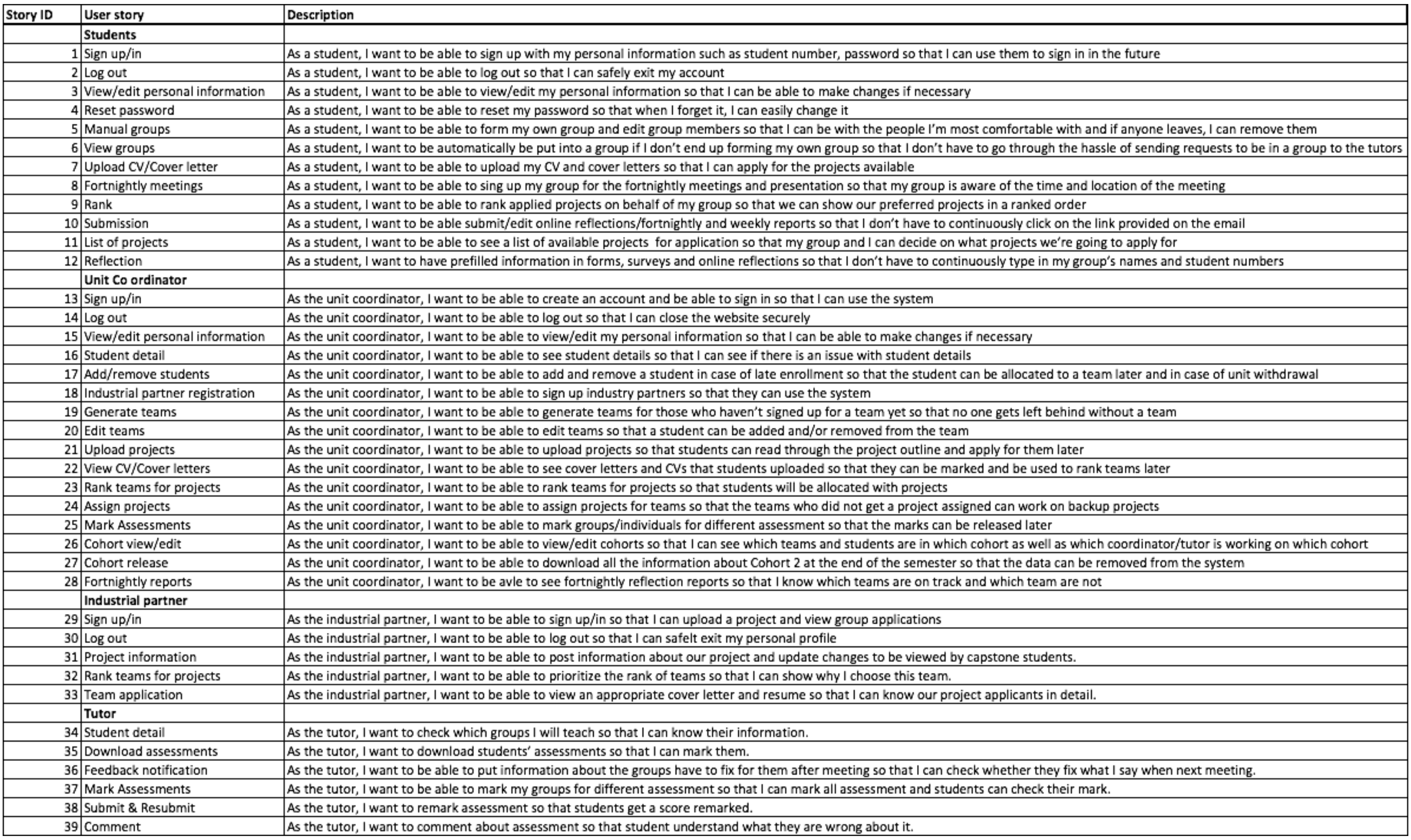
Low Fidelity prototypes is the initial part of any designs which includes the use of a paper and pencil to roughly sketch a page or a screen layout of a particular development, in this case webpages. Medium Fidelity prototypes is a step up from low fidelity and requires more complexity in terms of detail, colour and design. It is a close to representation of the final look/design of the page or layout of something. This involves intricate collaboration of designs and choice of colourings. Refer ***Appendix D and E*** for low and medium fidelity prototypes.

5.2 High Fidelity Prototype

In response to the development of the low fidelity and medium fidelity, high fidelity prototypes is the final stage of mock-ups. High fidelity mock up is done by coding the medium fidelity designs and as well as the functionality and features of the final product. Refer ***Appendix F*** for the high fidelity prototype.

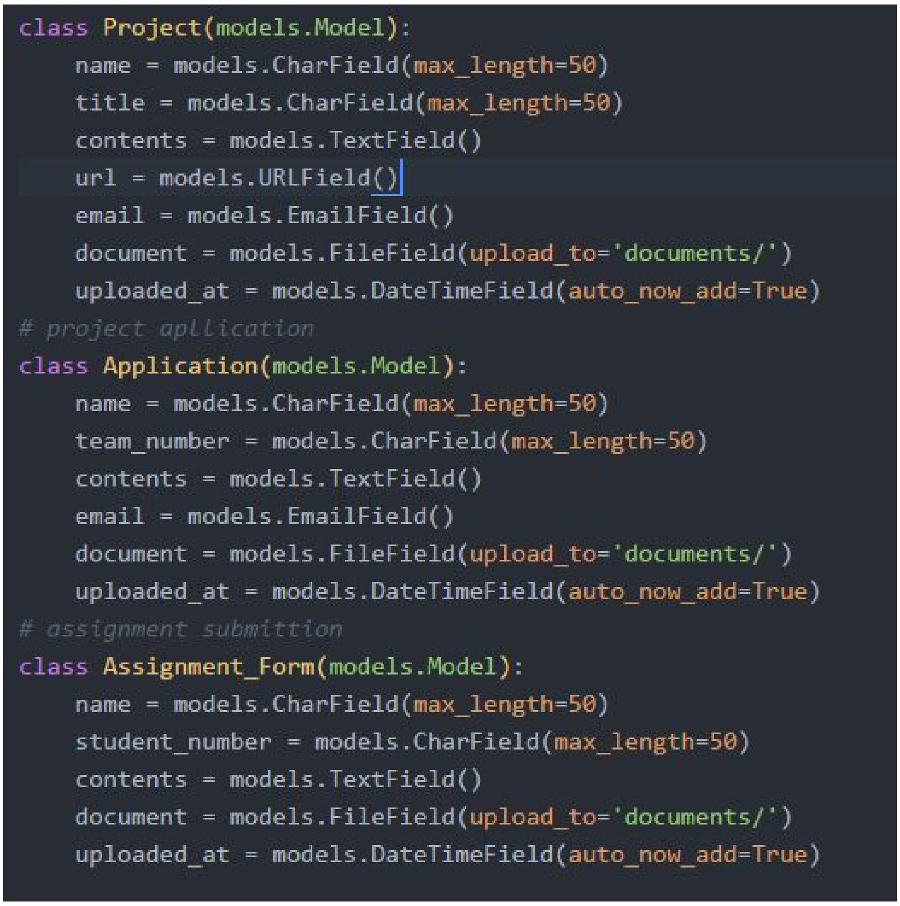
Appendix

Appendix A - User Stories

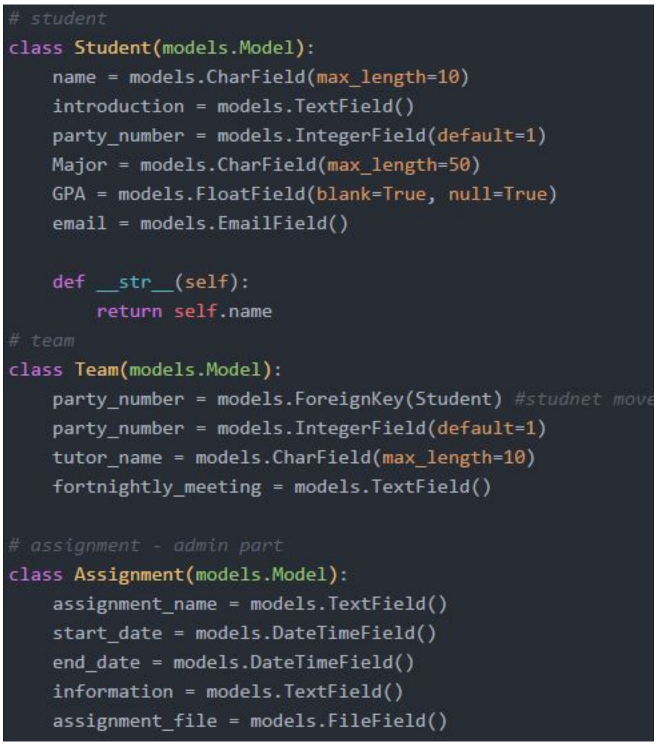


Story ID	User story	Description
	Students	
1	Sign up/in	As a student, I want to be able to sign up with my personal information such as student number, password so that I can use them to sign in in the future
2	Log out	As a student, I want to be able to log out so that I can safely exit my account
3	View/edit personal information	As a student, I want to be able to view/edit my personal information so that I can be able to make changes if necessary
4	Reset password	As a student, I want to be able to reset my password so that when I forget it, I can easily change it
5	Manual groups	As a student, I want to be able to form my own group and edit group members so that I can be with the people I'm most comfortable with and if anyone leaves, I can remove them
6	View groups	As a student, I want to be automatically be put into a group if I don't end up forming my own group so that I don't have to go through the hassle of sending requests to be in a group to the tutors
7	Upload CV/Cover letter	As a student, I want to be able to upload my CV and cover letters so that I can apply for the projects available
8	Fortnightly meetings	As a student, I want to be able to sing up my group for the fortnightly meetings and presentation so that my group is aware of the time and location of the meeting
9	Rank	As a student, I want to be able to rank applied projects on behalf of my group so that we can show our preferred projects in a ranked order
10	Submission	As a student, I want to be able submit/edit online reflections/fortnightly and weekly reports so that I don't have to continuously click on the link provided on the email
11	List of projects	As a student, I want to be able to see a list of available projects for application so that my group and I can decide on what projects we're going to apply for
12	Reflection	As a student, I want to have prefilled information in forms, surveys and online reflections so that I don't have to continuously type in my group's names and student numbers
	Unit Co ordinator	
13	Sign up/in	As the unit coordinator, I want to be able to create an account and be able to sign in so that I can use the system
14	Log out	As the unit coordinator, I want to be able to log out so that I can close the website securely
15	View/edit personal information	As the unit coordinator, I want to be able to view/edit my personal information so that I can be able to make changes if necessary
16	Student detail	As the unit coordinator, I want to be able to see student details so that I can see if there is an issue with student details
17	Add/remove students	As the unit coordinator, I want to be able to add and remove a student in case of late enrollment so that the student can be allocated to a team later and in case of unit withdrawal
18	Industrial partner registration	As the unit coordinator, I want to be able to sign up industry partners so that they can use the system
19	Generate teams	As the unit coordinator, I want to be able to generate teams for those who haven't signed up for a team yet so that no one gets left behind without a team
20	Edit teams	As the unit coordinator, I want to be able to edit teams so that a student can be added and/or removed from the team
21	Upload projects	As the unit coordinator, I want to be able to upload projects so that students can read through the project outline and apply for them later
22	View CV/Cover letters	As the unit coordinator, I want to be able to see cover letters and CVs that students uploaded so that they can be marked and be used to rank teams later
23	Rank teams for projects	As the unit coordinator, I want to be able to rank teams for projects so that students will be allocated with projects
24	Assign projects	As the unit coordinator, I want to be able to assign projects for teams so that the teams who did not get a project assigned can work on backup projects
25	Mark Assessments	As the unit coordinator, I want to be able to mark groups/individuals for different assessment so that the marks can be released later
26	Cohort view/edit	As the unit coordinator, I want to be able to view/edit cohorts so that I can see which teams and students are in which cohort as well as which coordinator/tutor is working on which cohort
27	Cohort release	As the unit coordinator, I want to be able to download all the information about Cohort 2 at the end of the semester so that the data can be removed from the system
28	Fortnightly reports	As the unit coordinator, I want to be avle to see fortnightly reflection reports so that I know which teams are on track and which team are not
	Industrial partner	
29	Sign up/in	As the industrial partner, I want to be able to sign up/in so that I can upload a project and view group applications
30	Log out	As the industrial partner, I want to be able to log out so that I can safelt exit my personal profile
31	Project information	As the industrial partner, I want to be able to post information about our project and update changes to be viewed by capstone students.
32	Rank teams for projects	As the industrial partner, I want to be able to prioritize the rank of teams so that I can show why I choose this team.
33	Team application	As the industrial partner, I want to be able to view an appropriate cover letter and resume so that I can know our project applicants in detail.
	Tutor	
34	Student detail	As the tutor, I want to check which groups I will teach so that I can know their information.
35	Download assessments	As the tutor, I want to download students' assessments so that I can mark them.
36	Feedback notification	As the tutor, I want to be able to put information about the groups have to fix for them after meeting so that I can check whether they fix what I say when next meeting.
37	Mark Assessments	As the tutor, I want to be able to mark my groups for different assessment so that I can mark all assessment and students can check their mark.
38	Submit & Resubmit	As the tutor, I want to remark assessment so that students get a score remarked.
39	Comment	As the tutor, I want to comment about assessment so that student understand what they are wrong about it.

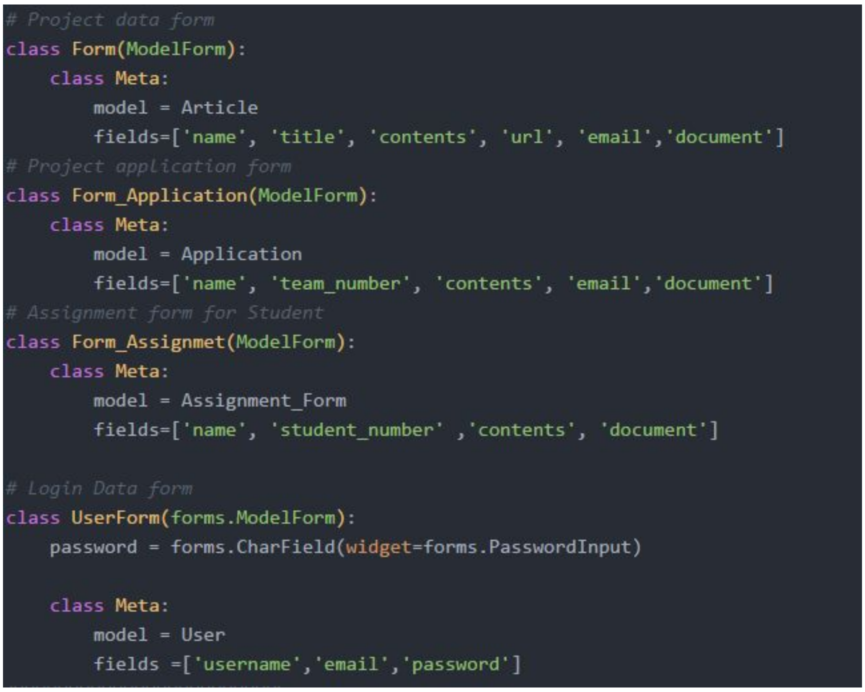
Appendix B - Technical Demonstration



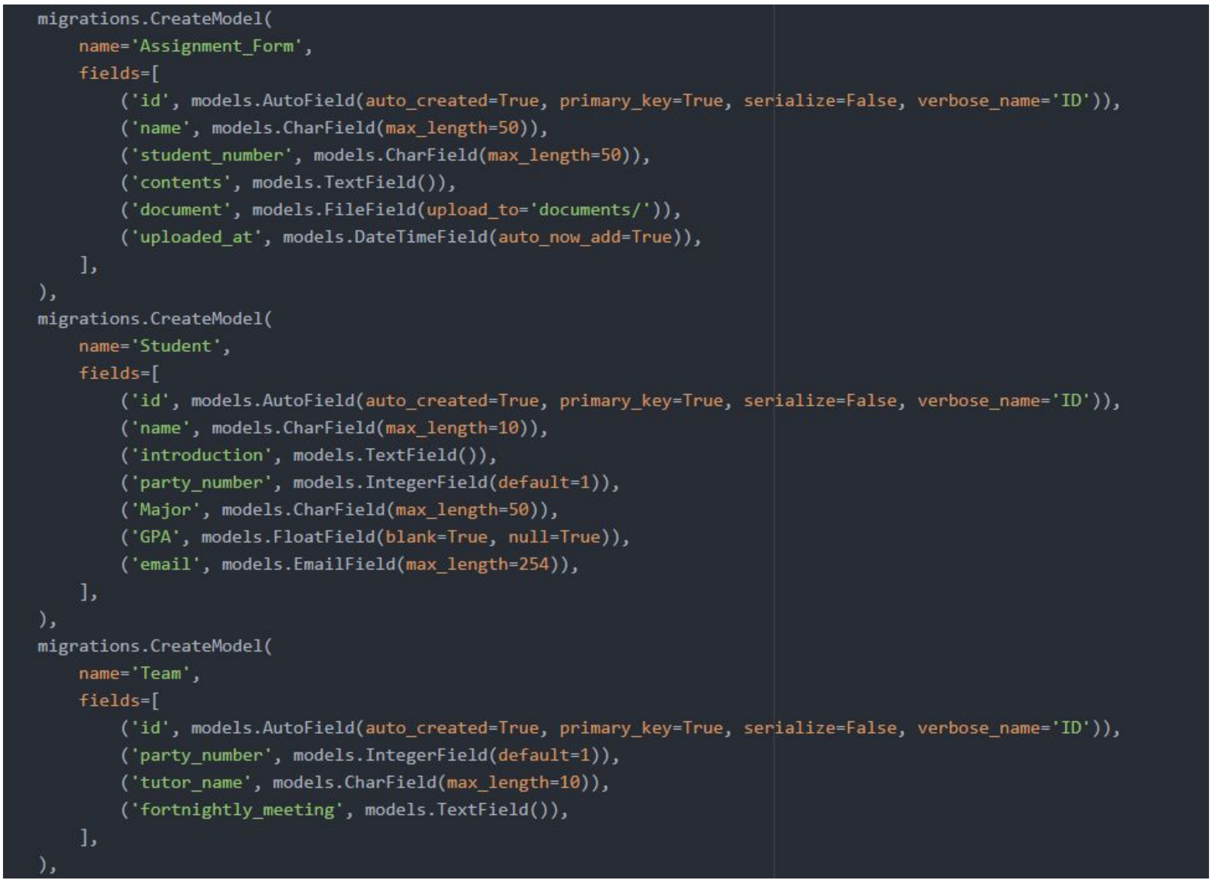
```python
class Project(models.Model):
    name = models.CharField(max_length=50)
    title = models.CharField(max_length=50)
    contents = models.TextField()
    url = models.URLField()
    email = models.EmailField()
    document = models.FileField(upload_to='documents/')
    uploaded_at = models.DateTimeField(auto_now_add=True)
# project apllication
class Application(models.Model):
    name = models.CharField(max_length=50)
    team_number = models.CharField(max_length=50)
    contents = models.TextField()
    email = models.EmailField()
    document = models.FileField(upload_to='documents/')
    uploaded_at = models.DateTimeField(auto_now_add=True)
# assignment submittion
class Assignment_Form(models.Model):
    name = models.CharField(max_length=50)
    student_number = models.CharField(max_length=50)
    contents = models.TextField()
    document = models.FileField(upload_to='documents/')
    uploaded_at = models.DateTimeField(auto_now_add=True)
```



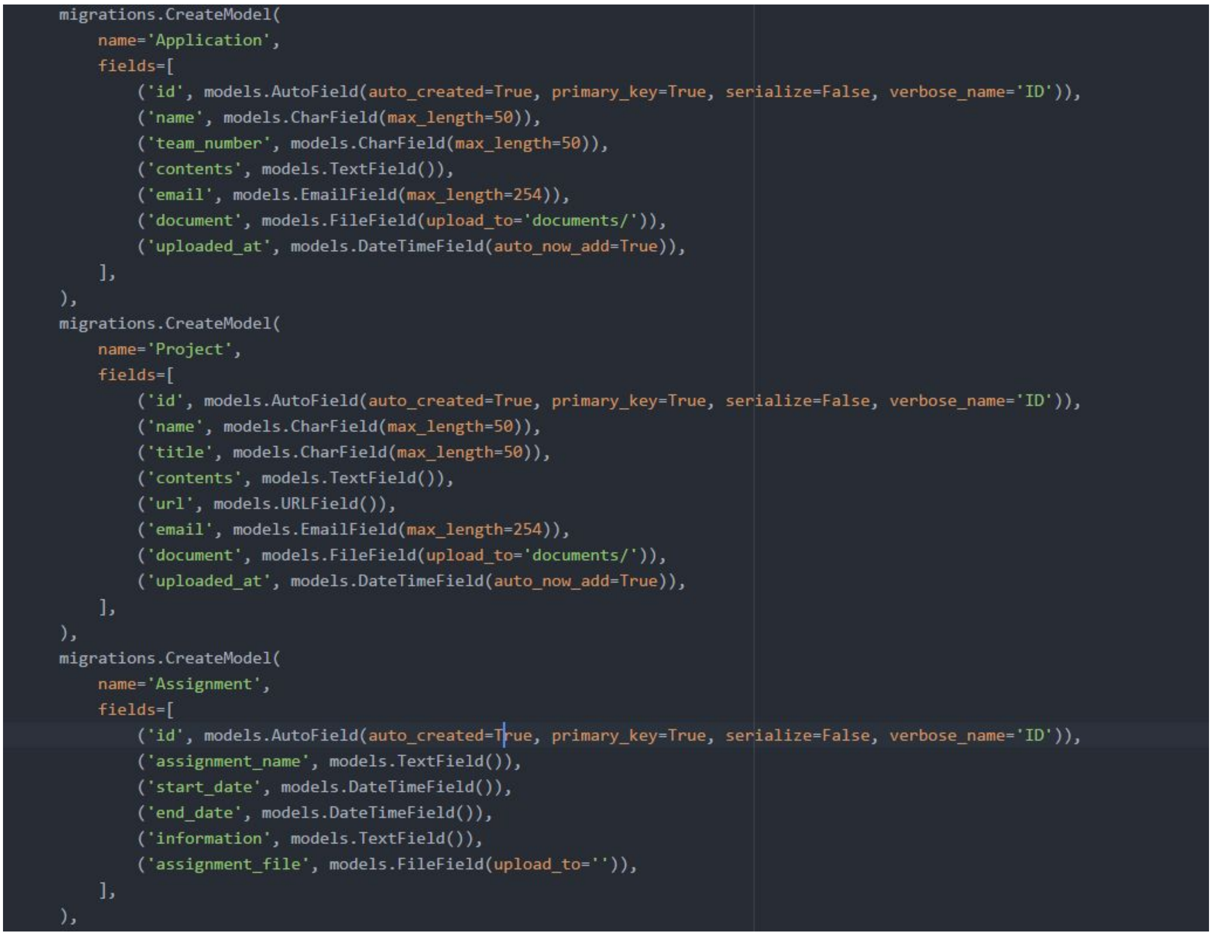
```python
# student
class Student(models.Model):
    name = models.CharField(max_length=10)
    introduction = models.TextField()
    party_number = models.IntegerField(default=1)
    Major = models.CharField(max_length=50)
    GPA = models.FloatField(blank=True, null=True)
    email = models.EmailField()

    def __str__(self):
        return self.name
# team
class Team(models.Model):
    party_number = models.ForeignKey(Student) #studnet move
    party_number = models.IntegerField(default=1)
    tutor_name = models.CharField(max_length=10)
    fortnightly_meeting = models.TextField()

# assignment - admin part
class Assignment(models.Model):
    assignment_name = models.TextField()
    start_date = models.DateTimeField()
    end_date = models.DateTimeField()
    information = models.TextField()
    assignment_file = models.FileField()
```



```
# Project data form
class Form(ModelForm):
    class Meta:
        model = Article
        fields=['name', 'title', 'contents', 'url', 'email','document']
# Project application form
class Form_Application(ModelForm):
    class Meta:
        model = Application
        fields=['name', 'team_number', 'contents', 'email','document']
# Assignment form for Student
class Form_Assignmet(ModelForm):
    class Meta:
        model = Assignment_Form
        fields=['name', 'student_number' ,'contents', 'document']

# Login Data form
class UserForm(forms.ModelForm):
    password = forms.CharField(widget=forms.PasswordInput)

    class Meta:
        model = User
        fields =['username','email','password']
```

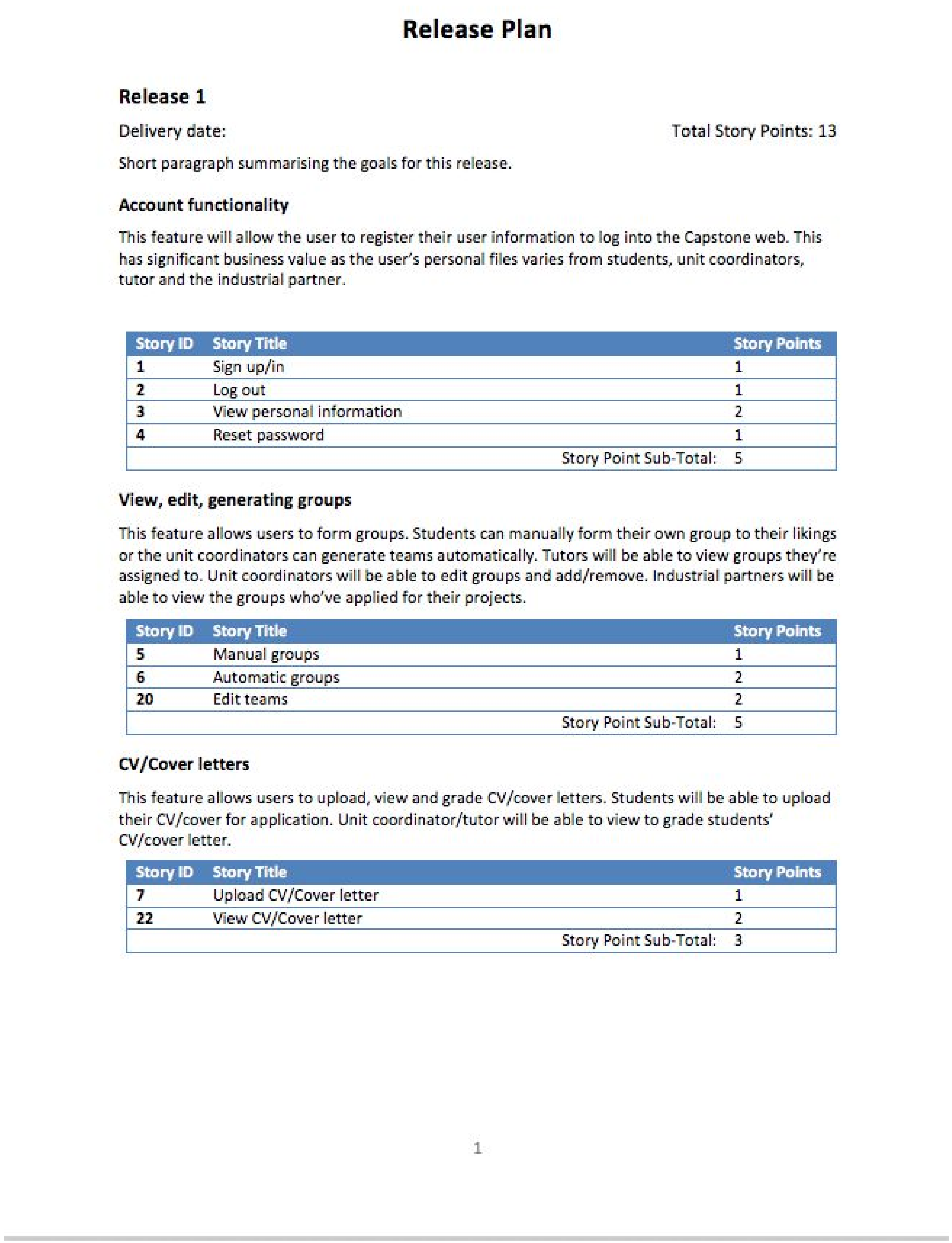


```
    migrations.CreateModel(
        name='Assignment_Form',
        fields=[
            ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
            ('name', models.CharField(max_length=50)),
            ('student_number', models.CharField(max_length=50)),
            ('contents', models.TextField()),
            ('document', models.FileField(upload_to='documents/')),
            ('uploaded_at', models.DateTimeField(auto_now_add=True)),
        ],
    ),
    migrations.CreateModel(
        name='Student',
        fields=[
            ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
            ('name', models.CharField(max_length=10)),
            ('introduction', models.TextField()),
            ('party_number', models.IntegerField(default=1)),
            ('Major', models.CharField(max_length=50)),
            ('GPA', models.FloatField(blank=True, null=True)),
            ('email', models.EmailField(max_length=254)),
        ],
    ),
    migrations.CreateModel(
        name='Team',
        fields=[
            ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
            ('party_number', models.IntegerField(default=1)),
            ('tutor_name', models.CharField(max_length=10)),
            ('fortnightly_meeting', models.TextField()),
        ],
    ),
```



```
        migrations.CreateModel(
            name='Application',
            fields=[
                ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
                ('name', models.CharField(max_length=50)),
                ('team_number', models.CharField(max_length=50)),
                ('contents', models.TextField()),
                ('email', models.EmailField(max_length=254)),
                ('document', models.FileField(upload_to='documents/')),
                ('uploaded_at', models.DateTimeField(auto_now_add=True)),
            ],
        ),
        migrations.CreateModel(
            name='Project',
            fields=[
                ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
                ('name', models.CharField(max_length=50)),
                ('title', models.CharField(max_length=50)),
                ('contents', models.TextField()),
                ('url', models.URLField()),
                ('email', models.EmailField(max_length=254)),
                ('document', models.FileField(upload_to='documents/')),
                ('uploaded_at', models.DateTimeField(auto_now_add=True)),
            ],
        ),
        migrations.CreateModel(
            name='Assignment',
            fields=[
                ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
                ('assignment_name', models.TextField()),
                ('start_date', models.DateTimeField()),
                ('end_date', models.DateTimeField()),
                ('information', models.TextField()),
                ('assignment_file', models.FileField(upload_to='')),
            ],
        ),
```

Appendix C - Release Sprint Plan



Release Plan

Release 1

Delivery date: Total Story Points: 13

Short paragraph summarising the goals for this release.

Account functionality

This feature will allow the user to register their user information to log into the Capstone web. This has significant business value as the user's personal files varies from students, unit coordinators, tutor and the industrial partner.

Story ID	Story Title	Story Points
1	Sign up/in	1
2	Log out	1
3	View personal information	2
4	Reset password	1
	Story Point Sub-Total:	5

View, edit, generating groups

This feature allows users to form groups. Students can manually form their own group to their likings or the unit coordinators can generate teams automatically. Tutors will be able to view groups they're assigned to. Unit coordinators will be able to edit groups and add/remove. Industrial partners will be able to view the groups who've applied for their projects.

Story ID	Story Title	Story Points
5	Manual groups	1
6	Automatic groups	2
20	Edit teams	2
	Story Point Sub-Total:	5

CV/Cover letters

This feature allows users to upload, view and grade CV/cover letters. Students will be able to upload their CV/cover for application. Unit coordinator/tutor will be able to view to grade students' CV/cover letter.

Story ID	Story Title	Story Points
7	Upload CV/Cover letter	1
22	View CV/Cover letter	2
	Story Point Sub-Total:	3

Release 2

Delivery date: Total Story Points: 14

Short paragraph summarising the goals for this release.

Projects

This feature allows users to have multiple functionalities on applying for projects. Students are able to view a list of available projects, apply and rank their selected projects. Unit coordinators are able to upload and view projects, assign projects to teams and edit. Tutors are able to see the list of projects and the projects assigned to each group. Industrial partners are able to upload projects and rank groups who have applied for their project accordingly.

Story ID	Story Title	Story Points
9	Rank projects (student)	2
11	View projects	1
21	Upload projects	1
23	Rank teams for projects	2
31	Project information	2
	Story Point Sub-Total:	8

Assessments

This feature allows users to download and mark assessments, make comments and provide feedbacks. These assessments include tutor meetings, final reports and other assessments.

Story ID	Story Title	Story Points
25	Mark assessments (tutor, unit coordinator, industrial partner)	2
36	Download assessments	1
37	Feedback	1
39	Submit & Resubmit	1
40	Comment	1
	Story Point Sub-Total:	6

Sprint Plan

Sprint 1

Total Story Points: 13 Total Hours:

Current Velocity:

Story 1: Sign up/in

Task ID	Task Description	Estimate	Taken
T01	Create a page dedicated for sign in and sign up	1	
T02	Add a sign-up button in the sign in page for users	0.3	
T03	Make a function to match up the details the user has entered with the details in the SQL tables	1	
T04	Create an error message if the user has entered the wrong details	1	
T05	Verify story is complete (acceptance test)	0.7	
	Story Points: 1 Total Hours:	4	

Story 2: Log out

Task ID	Task Description	Estimate	Taken
T06	Create a log out button dedicated for every pages	1	
T07	Make a function so that the user is exited from their account back to the sign in page	1	
T08	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	3	

Story 3: View Personal Information

Task ID	Task Description	Estimate	Taken
T09	Create a page that has the user's personal details displayed	1	
T10	Make a function so that the information stored in SQL tables is displayed on the page	2	
T11	Make a function so that the user has the flexibility to edit their details	2	
T12	Make a function so that that any changed details is updated on the SQL tables	2	
T13	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	8	

Story 4: Reset Password

Task ID	Task Description	Estimate	Taken
T14	Create a page that allows user to reset forgotten passwords	0.5	
T15	Write test case	1	
T16	Verify the authorisation and test the password	1	
T17	Make function so that the new password is saved on the SQL tables	1	
T18	Verify story is complete (acceptance test)	0.5	
	Story Points: 1 Total Hours:	4	

Story 5: Manual Groups

Task ID	Task Description	Estimate	Taken
T19	Create a page dedicated for group activities	0.5	
T20	Create a button that brings up a page to manually form groups	0.5	
T21	Create fields to create a group, group name and invite other members	0.5	
T22	Make a function so that the group information is stored on SQL tables	1.5	
T23	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

Story 6: Automatic Groups

Task ID	Task Description	Estimate	Taken
T24	Set a date limit for manual functionality	1	
T25	Make a function to generate at random students not in a group	2	
T26	Make a function so that the group information is stored on the SQL tables	4	
T27	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	8	

Story 20: Edit Teams

Task ID	Task Description	Estimate	Taken
T32	Create an edit button on the group page to remove, add and alter any changes if required	1	
T33	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	2	

Story 7: Upload CV/Cover Letter

Task ID	Task Description	Estimate	Taken
T34	Create an upload button	1	
T35	Make a function so that the uploaded document is stored on the system	2	
T36	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	4	

Story 22: View CV/Cover Letter

Task ID	Task Description	Estimate	Taken
T37	Create a download button	1	
T38	Make a function so that when the CV/cover letter is clicked, it pops up and open	2	
T39	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	4	

Sprint 2

Story 9: Rank Projects (students)

Task ID	Task Description	Estimate	Taken
T40	Create a page that displays of all applied projects with a ranking system	1	
T41	Make a function so that the pop up display automatically pops up after the final project is applied	3	
T42	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	5	

Story 11: List of Projects

Task ID	Task Description	Estimate	Taken
T43	Create a page with a list of all available projects for application	1	
T44	Make a function so that the project is accessible to overview and application	3	
T45	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

Story 21: Upload Projects (unit coordinator)

Task ID	Task Description	Estimate	Taken
T46	Create an edit button with "add" and "delete" functionality on the project page	1	
T47	Make a function so that when adding a project, the project supervisor can fill in a text field with project information or upload a doc/pdf file	2	
T48	Make a function so that when deleting a project, a text pops up prompting for deleting	1	
T49	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	5	

Story 23: Rank Projects

Task ID	Task Description	Estimate	Taken
T50	Make a function so that a drop-down menu on each project with the all the groups that applied	2	
T51	Make a function so that the users can rank top 5 groups for each project	2	
T52	Create a colour scheme for the ranking	1	
T53	Verify story complete (acceptance test)	1	
	Story Points: 2 Total Hours:	6	

Story 31: Project Information (industrial partner)

Task ID	Task Description	Estimate	Taken
T54	Create an edit button with "add" and "delete" functionality on the project page	1	
T55	Make a function so that when adding a project, a pop up display to fill in with the project information	2	
T56	Make a function so that when deleting a project, a text pops up prompting for deleting	1	
T57	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	5	

Story 25: Mark Assessments

Task ID	Task Description	Estimate	Taken
T58	Create a page for assessments	2	
T59	Make a function so that the CRA for assessments can be electronically editable for marking purposes	2	
T60	Make a function so that users can enter numeric values	2	
T61	Verify story is complete (acceptance test)	1	
	Story Points: 2 Total Hours:	7	

Story 36: Download Assessments

Task ID	Task Description	Estimate	Taken
T62	Create a download button for assessments	1	
T63	Verity story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	2	

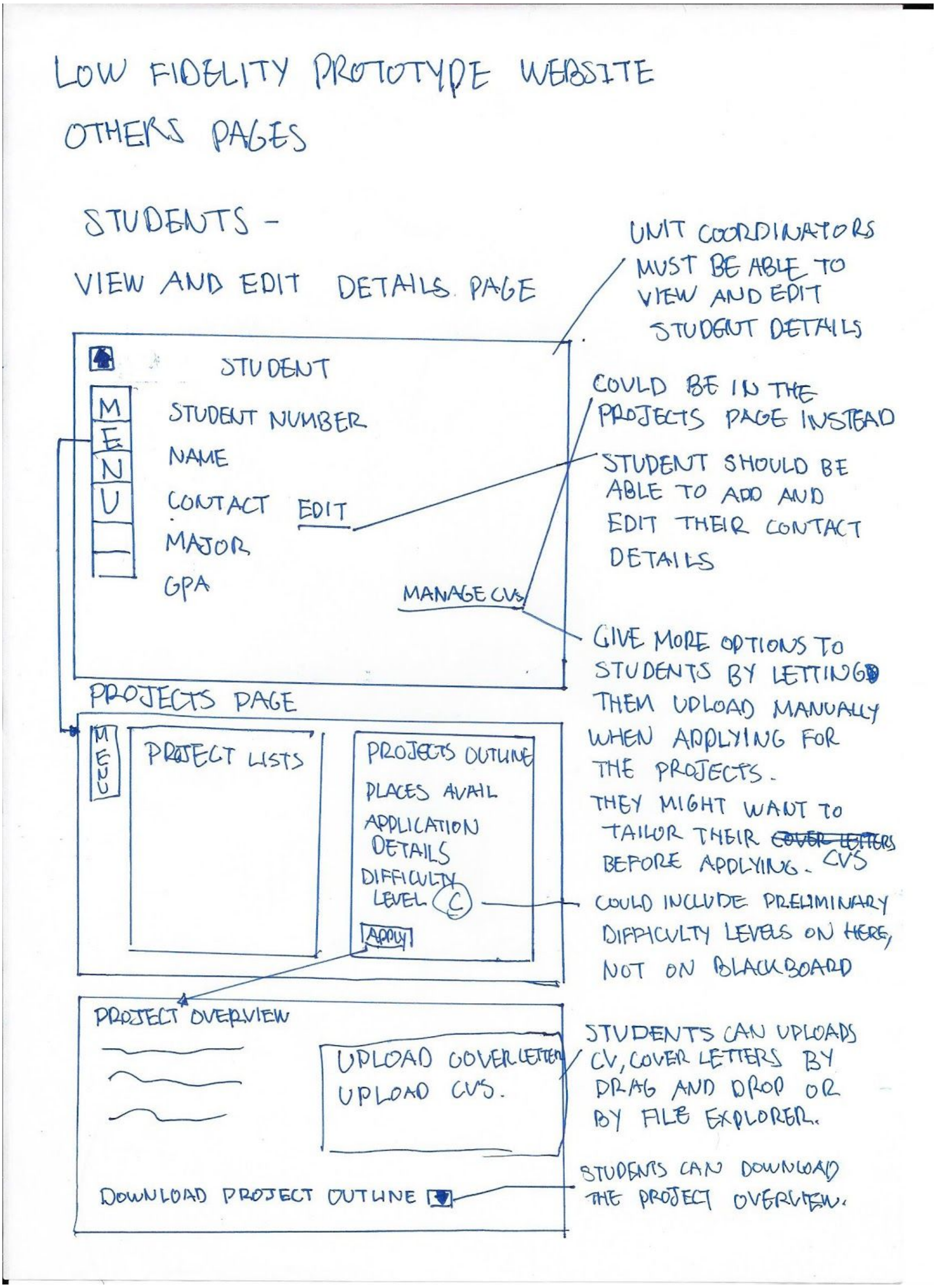
Story 37: Feedback

Task ID	Task Description	Estimate	Taken
T64	Create a text field for unit coordinators and tutors to give feedback	1	
T65	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	2	

Story 40: Comment

Task ID	Task Description	Estimate	Taken
T66	Create a text field for unit coordinators and tutors to comment while marking assessments	1	
T67	Verify story is complete (acceptance test)	1	
	Story Points: 1 Total Hours:	2	

Appendix D - Low fidelity prototypes



LOW FIDELITY PROTOTYPE WEBSITE

OTHERS PAGES

STUDENTS -

VIEW AND EDIT DETAILS PAGE

UNIT COORDINATORS MUST BE ABLE TO VIEW AND EDIT STUDENT DETAILS

COULD BE IN THE PROJECTS PAGE INSTEAD

STUDENT SHOULD BE ABLE TO ADD AND EDIT THEIR CONTACT DETAILS

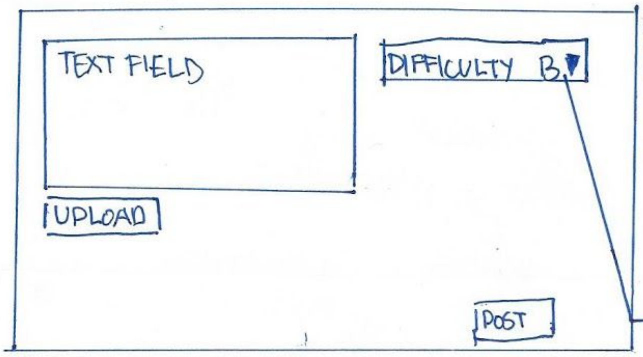
GIVE MORE OPTIONS TO STUDENTS BY LETTING THEM UPLOAD MANUALLY WHEN APPLYING FOR THE PROJECTS. THEY MIGHT WANT TO TAILOR THEIR ~~COVER LETTERS~~ CVS BEFORE APPLYING.

COULD INCLUDE PRELIMINARY DIFFICULTY LEVELS ON HERE, NOT ON BLACKBOARD

STUDENTS CAN UPLOADS CV, COVER LETTERS BY DRAG AND DROP OR BY FILE EXPLORER.

STUDENTS CAN DOWNLOAD THE PROJECT OVERVIEW.

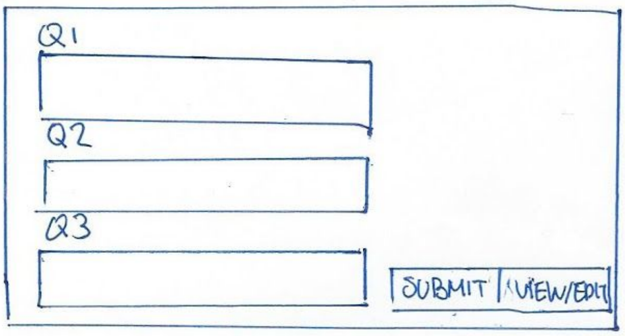
PROJECTS PAGE - COORDINATORS AND INDUSTRY PARTNERS



PROJECTS CAN BE UPLOADED BY EITHER FILLING THE TEXT FIELD OR UPLOADING A DOC/PDF FILE

DROPDOWN MENU.

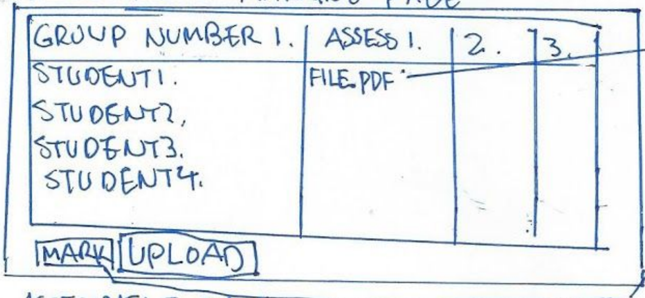
REFLECTION- STUDENTS.



STUDENTS WILL BE GIVEN A NUMBER OF QUESTIONS FOR THE REFLECTION. THEY WILL BE ABLE TO EDIT BEFORE THE NEXT TUTOR MEETING.

TUTORS AND COORDINATORS SHOULD BE ABLE TO SEE THE REFLECTION REPORTS,

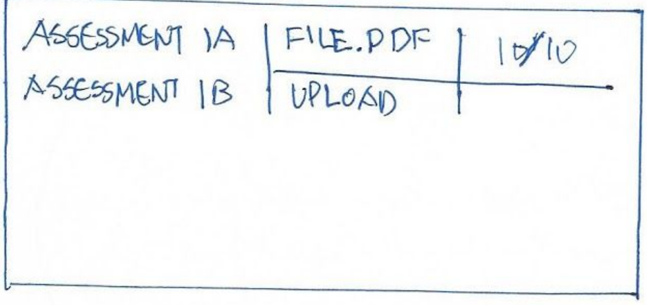
ASSESSMENT MARKING PAGE



FILES ARE IN PDF FORMAT, OPENED WHEN CLICKED, COULD BE DOWNLOADED.

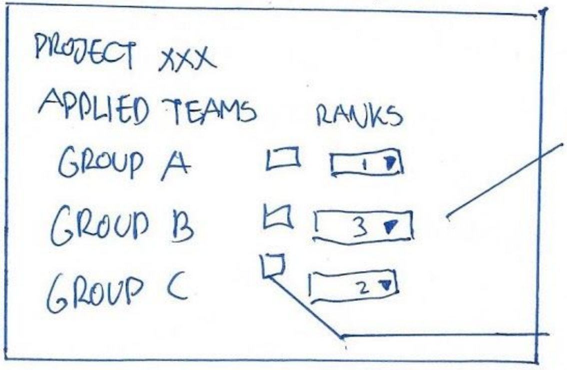
UNIT COORDINATOR CAN MARK ASSESSMENT ON THIS PAGE, OR HE CAN UPLOAD AN EXCEL SPREADSHEET PREFILLED WITH MARKS FOR DISPLAY

ASSESSMENT PAGE.



STUDENTS WILL BE ABLE TO UPLOAD ASSESSMENTS ON THIS PAGE, WILL BE ABLE TO SEE MARKS AS WELL

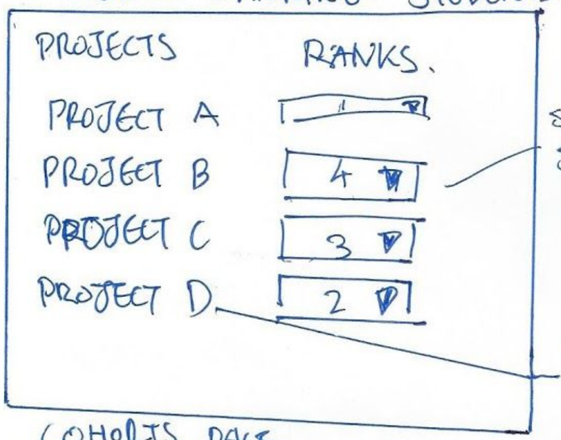
PROJECT SUPERVISOR - RANK TEAMS



PROJECT SUPERVISOR CAN RANK TEAMS USING A DROP DOWN MENU.

DOWN LOAD/VIEW CV/COVER LETTERS BY CLICKING HERE

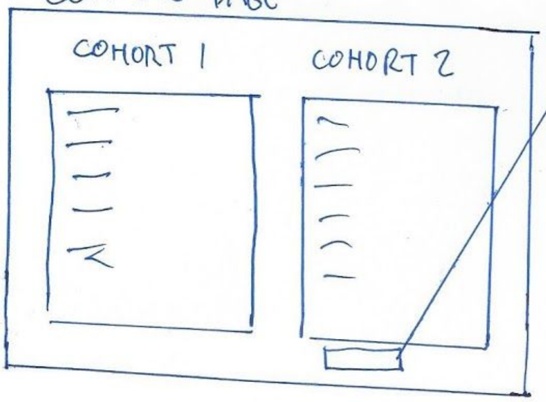
PROJECTS RANKING - STUDENTS.



SIMILAR TO PROJECT SUPERVISORS, STUDENTS SHOULD BE ABLE TO RANK PROJECTS

CLICK TO SEE PROJECT OUTLINE

COHORTS PAGE

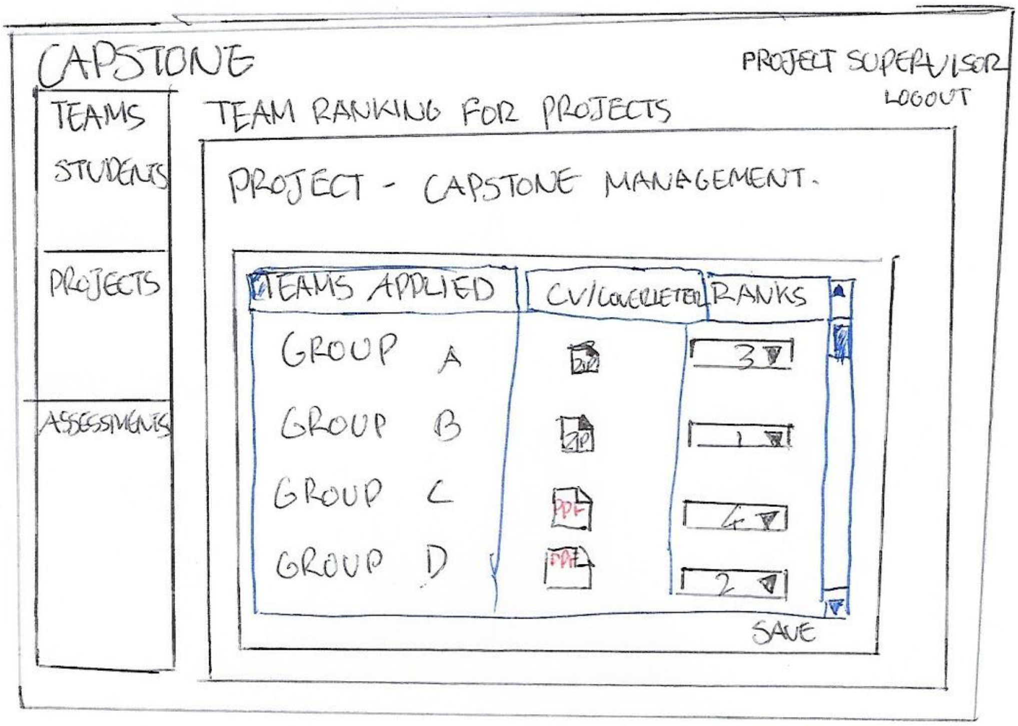


DOWNLOAD / REMOVE COHORT FROM THE SYSTEM.

COHORT 1 WILL BE MOVED AUTOMATICALLY AND BE CALLED COHORT 2 NOW.

COHORT 1 WILL BE LEFT UNTIL THE SEMESTER BEGINS.

RANK TEAMS - PROJECT SUPERVISORS



CAPSTONE

PROJECT SUPERVISOR
LOGOUT

TEAMS
STUDENTS

PROJECTS

ASSESSMENTS

TEAM RANKING FOR PROJECTS

PROJECT - CAPSTONE MANAGEMENT.

TEAMS APPLIED	CV/COVERLETER	RANKS
GROUP A	ZIP	3 ▼
GROUP B	ZIP	1 ▼
GROUP C	PDF	4 ▼
GROUP D	PDF	2 ▼

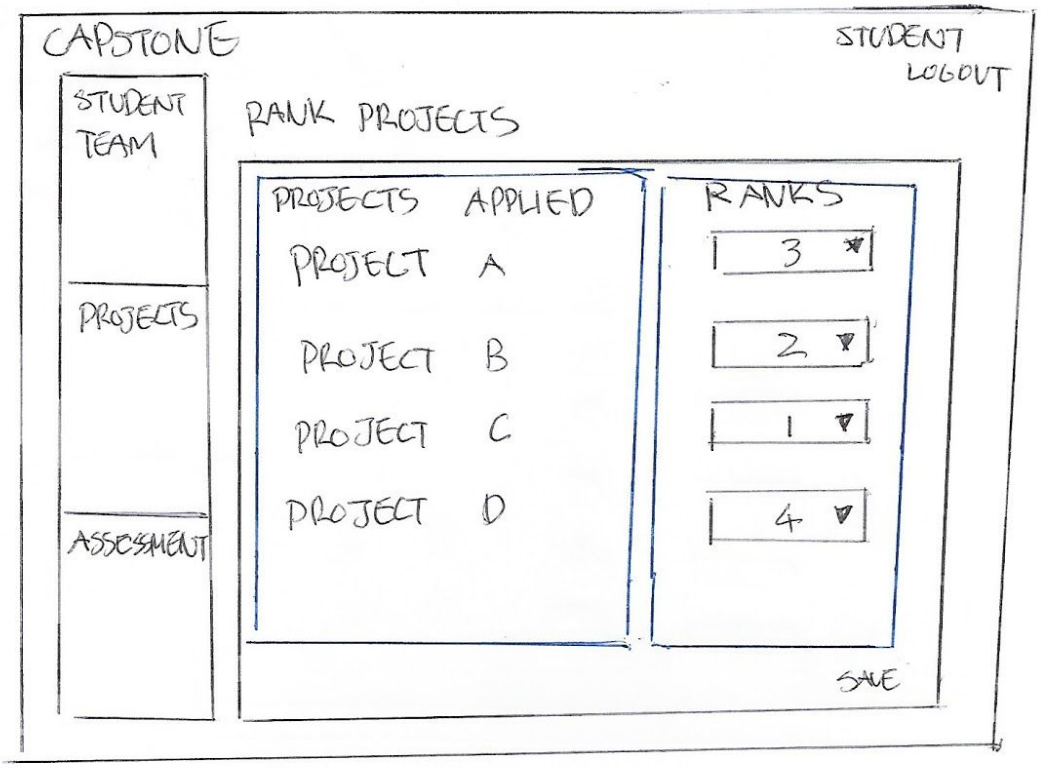
SAVE

WOULD KEEP THE SAME COLOUR SCHEME AND UI LAYOUT AS BB FOR THE USABILITY

RELATED USER STORIES.

- UNIT COORDINATOR, INDUSTRY PARTNERS.
- RANK TEAMS FOR PROJECT.

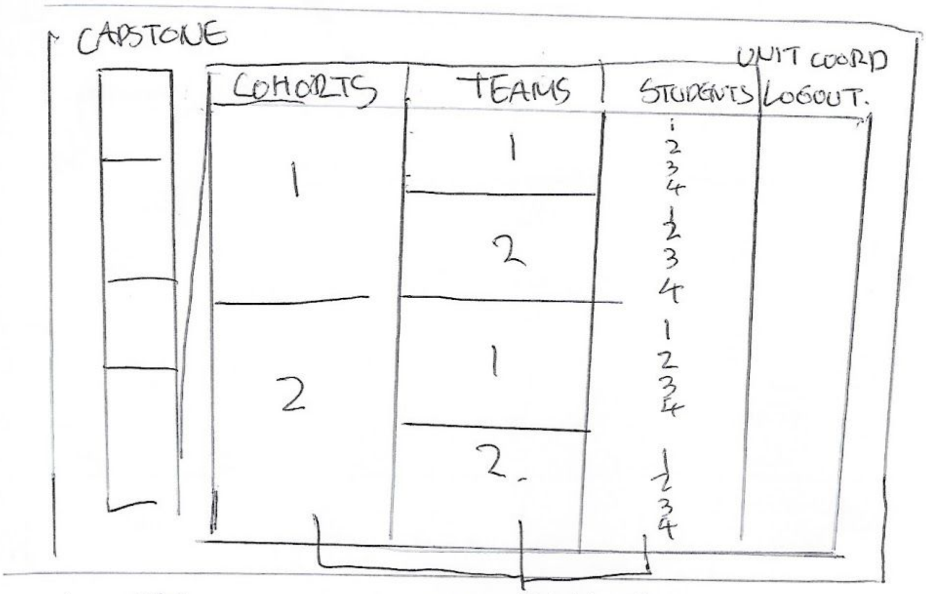
RANK PROJECTS - STUDENTS.



RELATED USER STORIES.

- STUDENTS.
- RANK PROJECTS.

COHORTS - TEAMS - STUDENTS DETAILS PAGE.



RELATED USER STORIES

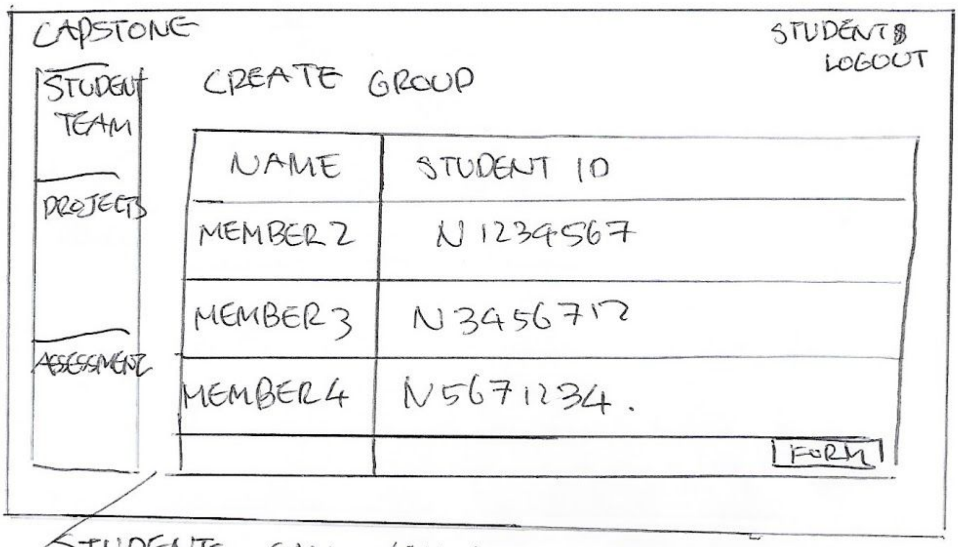
VIEW/EDIT COHORTS.

RELEASE/DOWNLOAD COHORTS.

EDIT STUDENT / ~~TEAM~~ DETAILS.

ADD/ REMOVE STUDENTS/ TEAMS.

STUDENTS - GROUPS PAGE.



NAME	STUDENT ID
MEMBER 2	N1234567
MEMBER 3	N3456712
MEMBER 4	N5671234.

STUDENTS CAN MANUALLY FORM GROUPS BY PUTTING STUDENTS' NAME AND IDS.

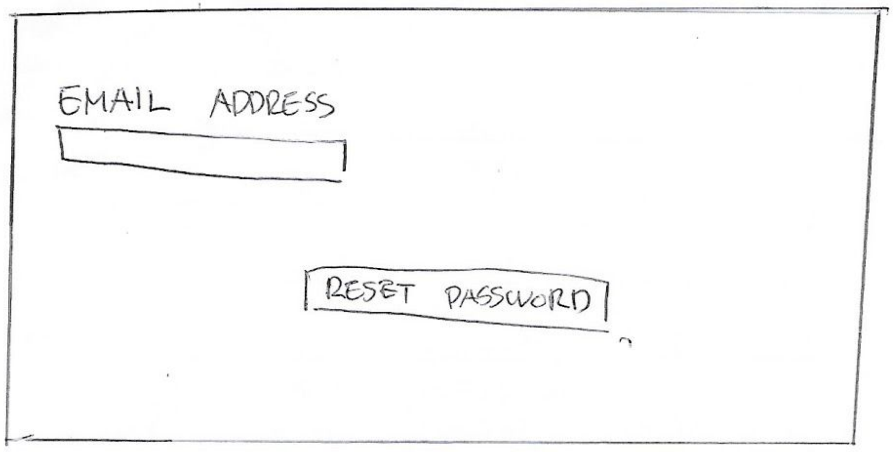
ONCE FORMED/OR ASSIGNED TO GROUP. THEY WILL BE ABLE TO SEE GROUP MEMBER NAMES AND JOIN A TIME SLOT FOR DIFFERENT ASSESSMENTS INCLUDING TUTOR MEETINGS AND THE END OF SEMESTER PRESENTATION.

TEAM 4	STUDENT 1 STUDENT 2 STUDENT 3

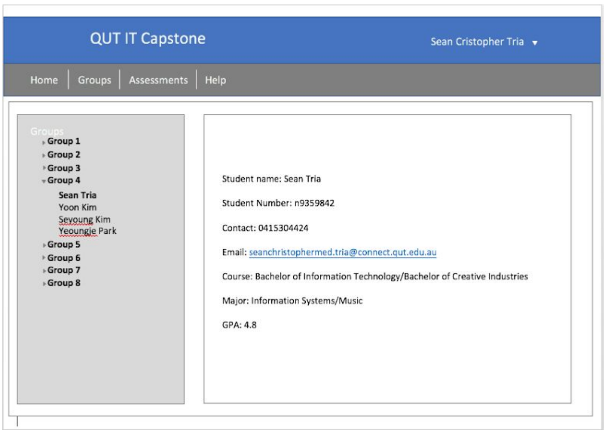
·TIME SLOT

TIME	VENUE	TUTOR
TUES 10:25	0604	JIMMY

ALL USERS - PASSWORD RESET



Appendix E - Medium fidelity prototype



Student name: Sean Tria

Student Number: n9359842

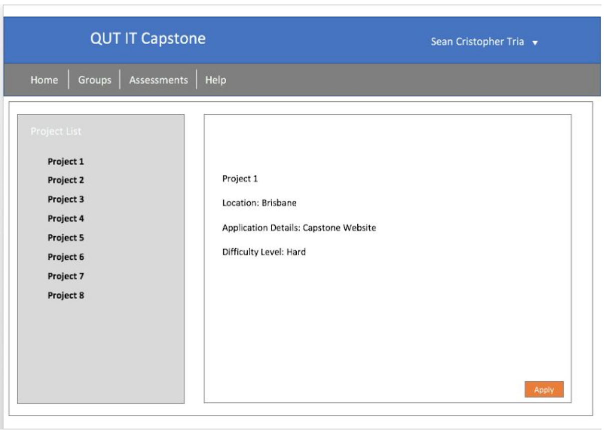
Contact: 0415304424

Email: seanchristophermed.tria@connect.qut.edu.au

Course: Bachelor of Information Technology/Bachelor of Creative Industries

Major: Information Systems/Music

GPA: 4.8

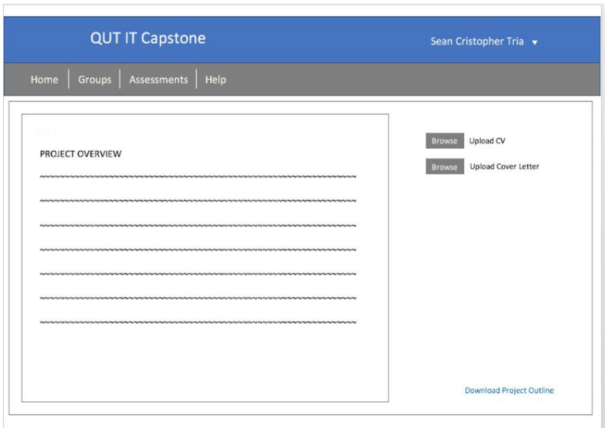


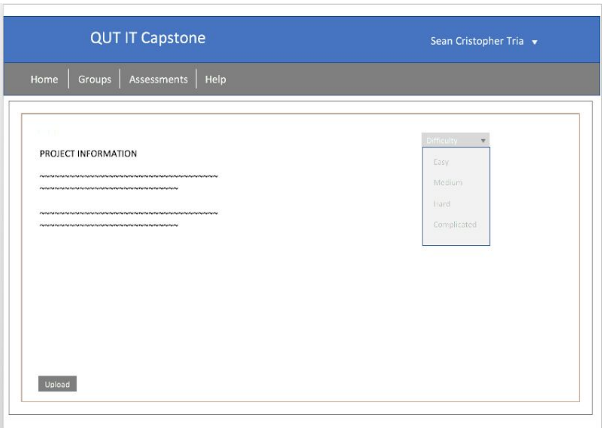
Project 1

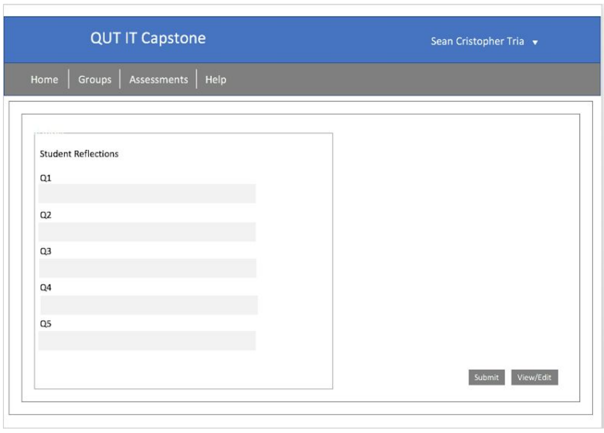
Location: Brisbane

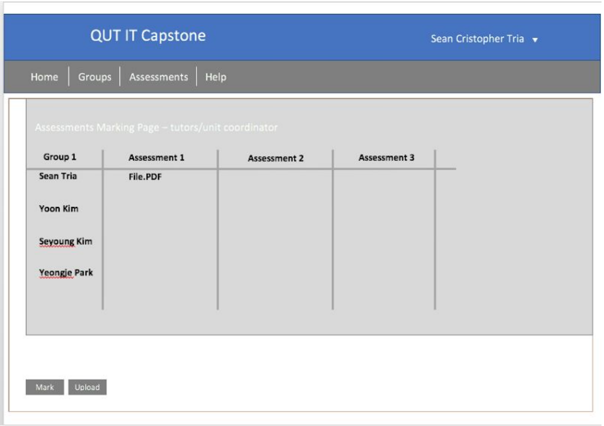
Application Details: Capstone Website

Difficulty Level: Hard



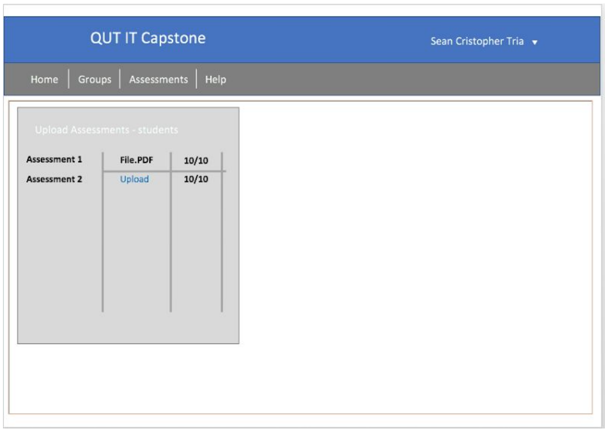






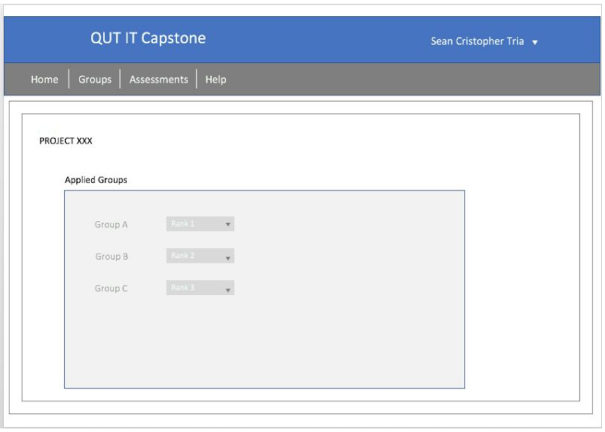
Assessments Marking Page – tutors/unit coordinator

Group 1	Assessment 1	Assessment 2	Assessment 3
Sean Tria	File.PDF		
Yoon Kim			
Seyoung Kim			
Yeongje Park			



Upload Assessments - students

Assessment 1	File.PDF	10/10
Assessment 2	Upload	10/10



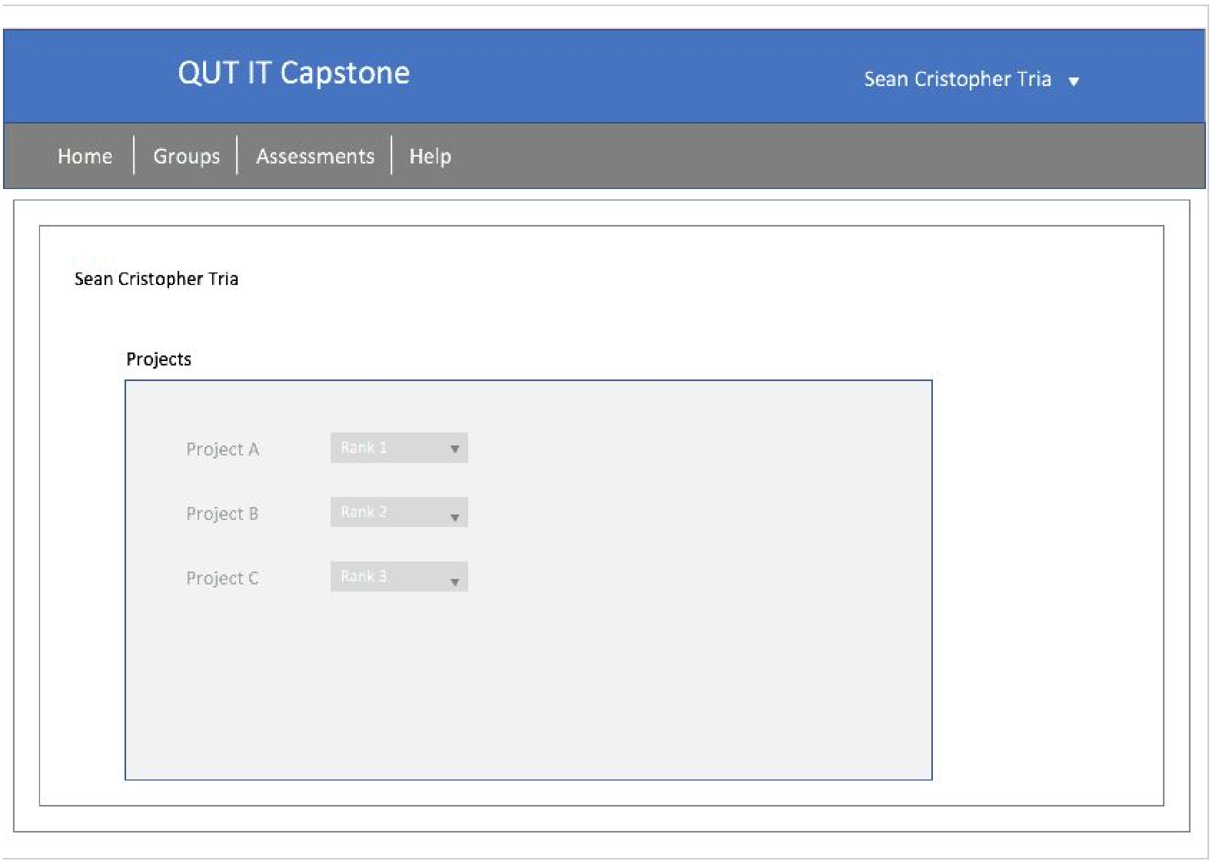
PROJECT XXX

Applied Groups

Group A Rank 1 ▾

Group B Rank 2 ▾

Group C Rank 3 ▾



Sean Cristopher Tria

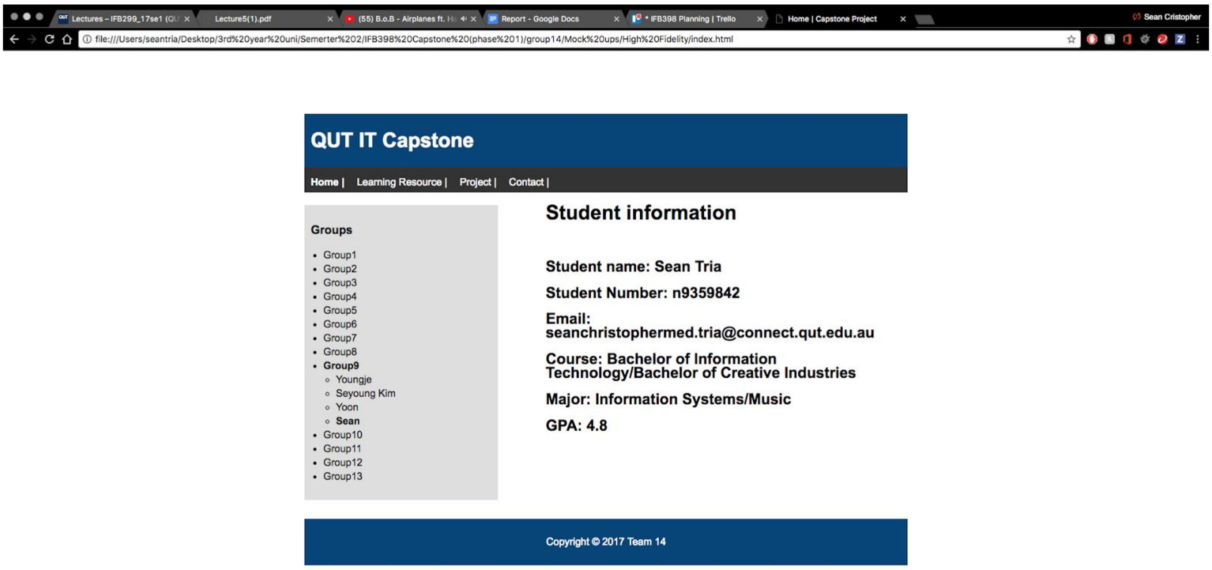
Projects

Project A Rank 1 ▾

Project B Rank 2 ▾

Project C Rank 3 ▾

Appendix F - High fidelity prototype


Lectures – IFB299_17se1 (QU
Lecture5(1).pdf
(55) B.o.B - Airplanes ft. Ha
Report - Google Docs
* IFB398 Planning | Trello
Home | Capstone Project
Sean Cristopher
file:///Users/seantria/Desktop/3rd%20year%20uni/Semerter%202/IFB398%20Capstone%20(phase%201)/group14/Mock%20ups/High%20Fidelity/index.html
Home | Learning Resource | Project | Contact |

QUT IT Capstone

Groups

- Group1
- Group2
- Group3
- Group4
- Group5
- Group6
- Group7
- Group8
- **Group9**
  - Youngje
  - Seyoung Kim
  - Yoon
  - **Sean**
- Group10
- Group11
- Group12
- Group13

Student information

Student name: Sean Tria

Student Number: n9359842

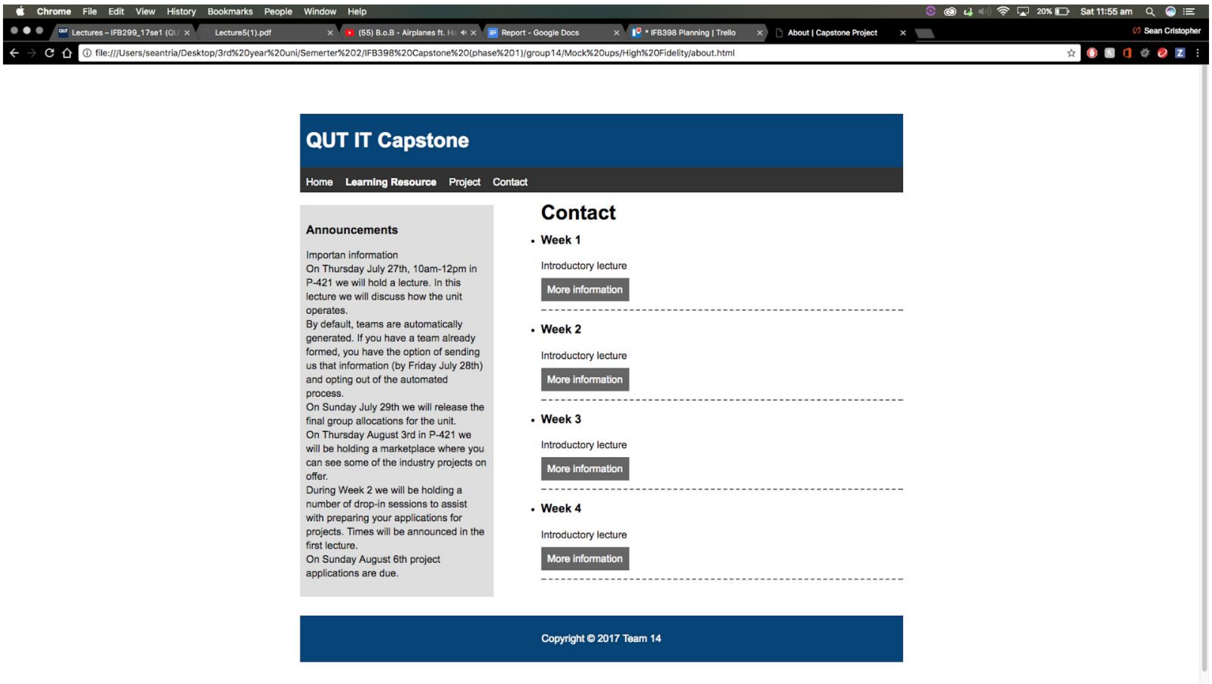
Email: seanchristophermed.tria@connect.qut.edu.au

Course: Bachelor of Information Technology/Bachelor of Creative Industries

Major: Information Systems/Music

GPA: 4.8

Copyright © 2017 Team 14


Chrome
File
Edit
View
History
Bookmarks
People
Window
Help
20%
Sat 11:55 am
Lectures – IFB299_17se1 (QU
Lecture5(1).pdf
(55) B.o.B - Airplanes ft. Ha
Report - Google Docs
* IFB398 Planning | Trello
About | Capstone Project
Sean Cristopher
file:///Users/seantria/Desktop/3rd%20year%20uni/Semerter%202/IFB398%20Capstone%20(phase%201)/group14/Mock%20ups/High%20Fidelity/about.html
Home
Learning Resource
Project
Contact
More information

QUT IT Capstone

Announcements

Importan information
On Thursday July 27th, 10am-12pm in P-421 we will hold a lecture. In this lecture we will discuss how the unit operates.
By default, teams are automatically generated. If you have a team already formed, you have the option of sending us that information (by Friday July 28th) and opting out of the automated process.
On Sunday July 29th we will release the final group allocations for the unit.
On Thursday August 3rd in P-421 we will be holding a marketplace where you can see some of the industry projects on offer.
During Week 2 we will be holding a number of drop-in sessions to assist with preparing your applications for projects. Times will be announced in the first lecture.
On Sunday August 6th project applications are due.

Contact

- **Week 1**

  Introductory lecture

- **Week 2**

  Introductory lecture

- **Week 3**

  Introductory lecture

- **Week 4**

  Introductory lecture

Copyright © 2017 Team 14