SW Engineering CSC648/848 Fall 2019

“Gator Store”

Section 02

Milestone 2

Team 12

Beibei Jiang - Team Lead / Github Master / Back-end Developer
Email: bjiang1@mail.sfsu.edu
Jinying Ren - Front-end Lead
Jesse Smick - Back-end Lead
Angelo Solitario - Front-end Developer
Melissa Endang - Front-end Developer
Jinghan Cao - Back-end Developer

Submitted Date	10/18/2019
Revised and Frozen Date	TBD

1. Functional Requirements

Priority 1

Unregistered Users:

1. Users shall be able to create an account
2. Users shall be able to view listed items for sale
3. Users shall be able to browse through category
6. Users shall be able to be redirected to register page if they want to post items for sale

Registered Users:

7.1. Users shall be able to create an account
7.2. Users shall be able to view listed items for sale
7.3. Users shall be able to browse through category
7.6. Users shall be able to be redirected to register page if they want to post items for sale
8. Users shall be able to log into the website
9. Users shall be able to have their own dashboard
11. Users shall be able to post items attached with images for sale on this website
12. Users shall be able to post items attached with prices for sale on this website
14. Users' dashboard shall be able to view the in-site messages
15. Users' dashboard shall be able to view past posted items
17. Users' dashboard shall be able to delete past posted items
18. Users' posted items shall be reviewed by an administrator before it is officially posted on website
20. Users shall be able to send in-site messages among each other

Admins:

24. Administrators shall be able to log in the website
26. Administrators shall be able to review the items before they are officially posted
27. Administrators shall be required to approve items before they are officially posted

Priority 2

Unregistered Users:

4. Users shall be able to browse through tag

Registered Users:

7.4.Users shall be able to browse through tag
10. Users shall be able to post items attached with descriptions for sale on this website
16. Users' dashboard shall be able to modify past posted items

19. Users’ posted items shall be reviewed again if they update the information about items

Admins:

25. Administrators shall be able to have administrator dashboard

Priority 3

Unregistered Users:

5. Users shall be able to see the remaining amount of items

Registered Users:

7.5.Users shall be able to see the remaining amount of items

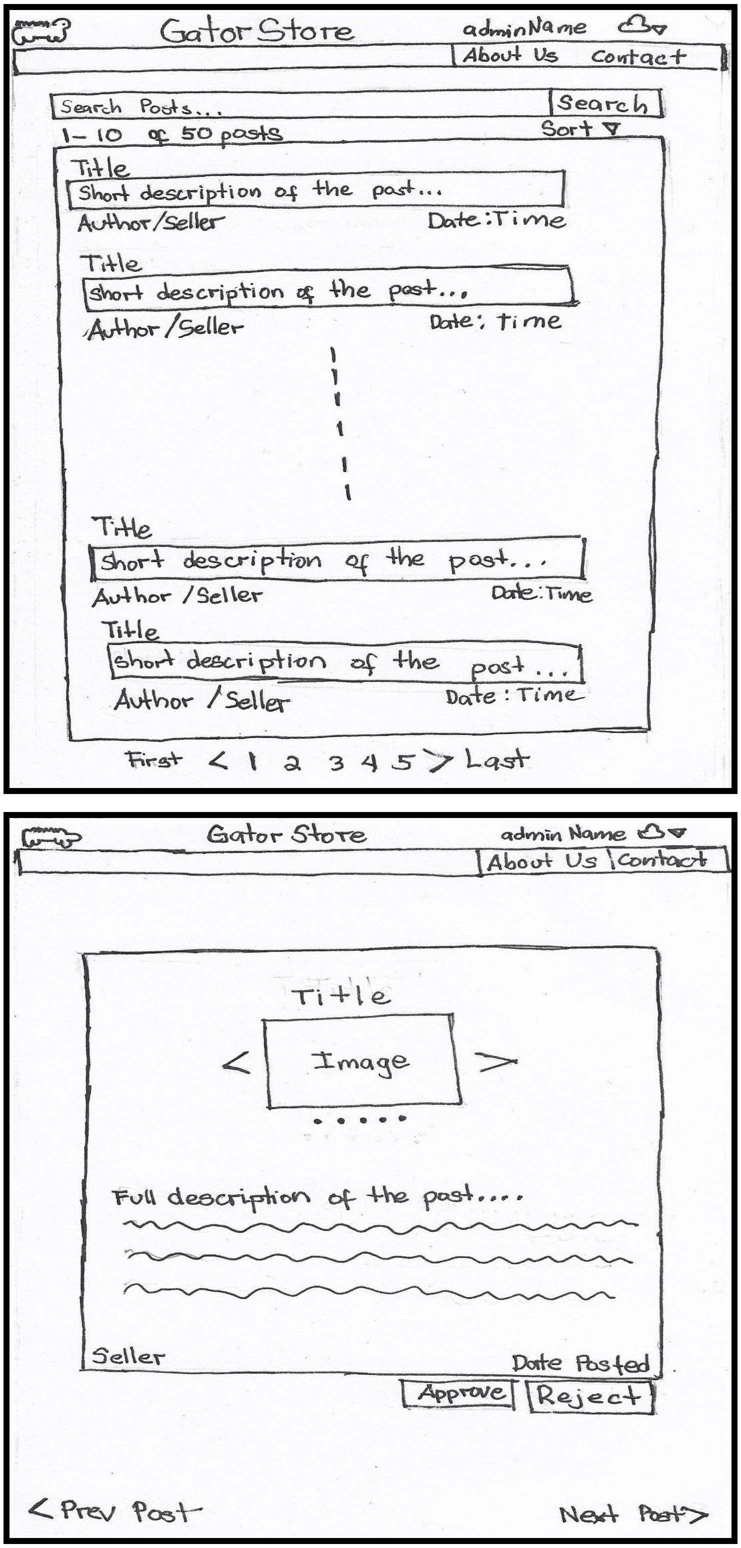
13. Users shall be able to post items attached with tags for sale on this website
21. Users shall be able to view approval results from an administrator regarding their proposed items for sale
22. Users shall be able to choose the location within campus to trade off-line
23. Users shall be able to receive confirmation message sent to their in-site message box

Admins:

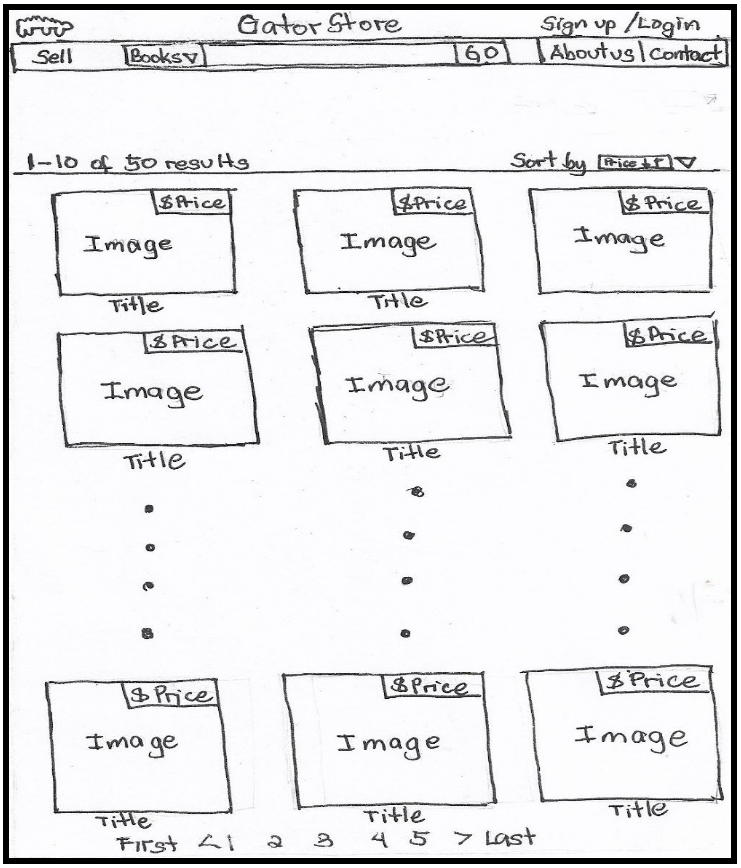
28. Administrators shall be able to send messages to sellers if the items are against the policies

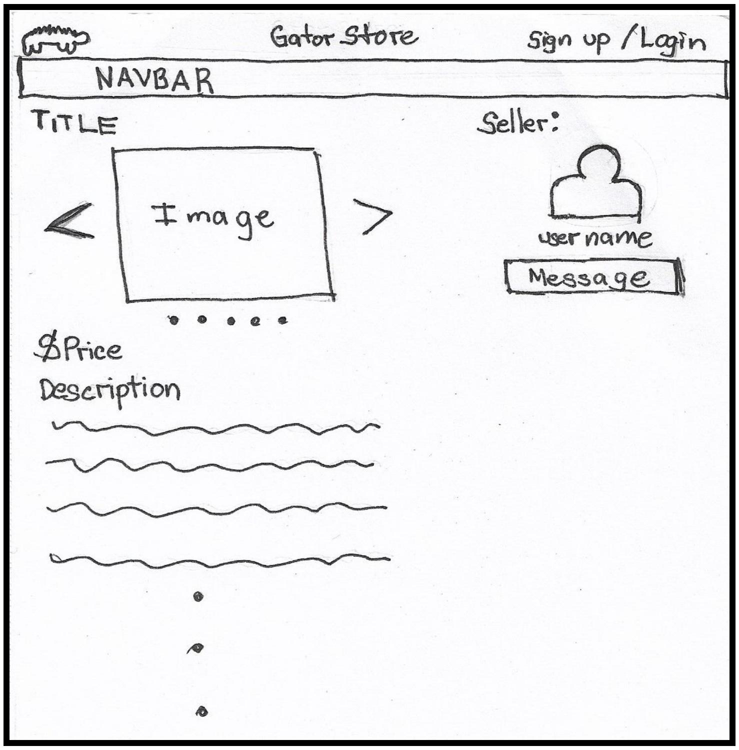
2. UI Mockups and Storyboards(high level only)

2.1 Ali the admin goes through the posts to review them one by one to approve or reject them.

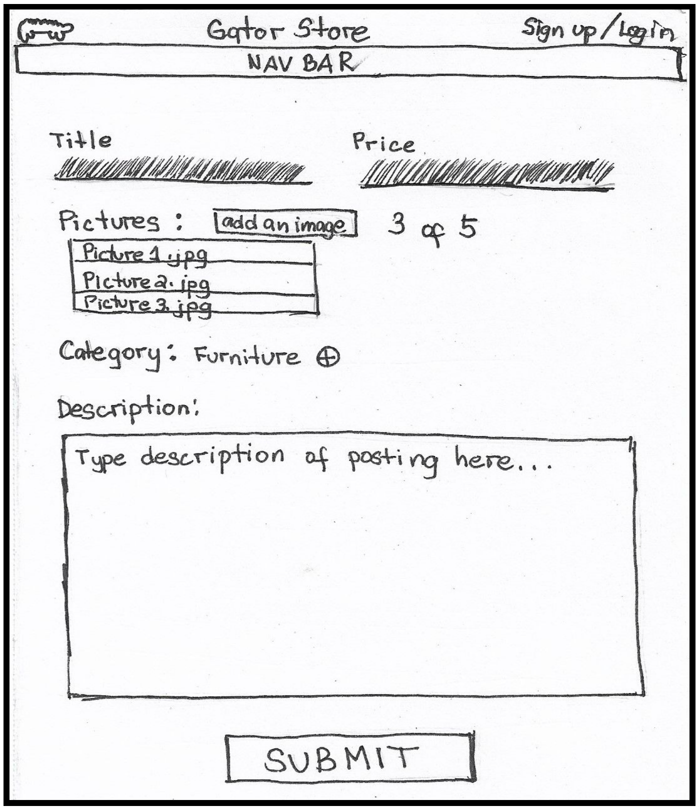


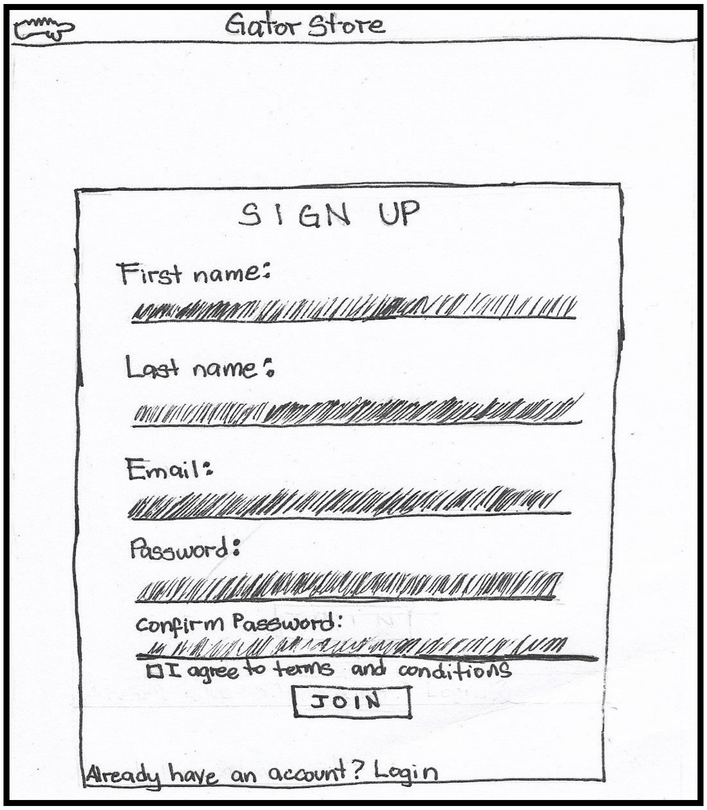
2.2 Bella went on the website to browse by category. Bella views one item.



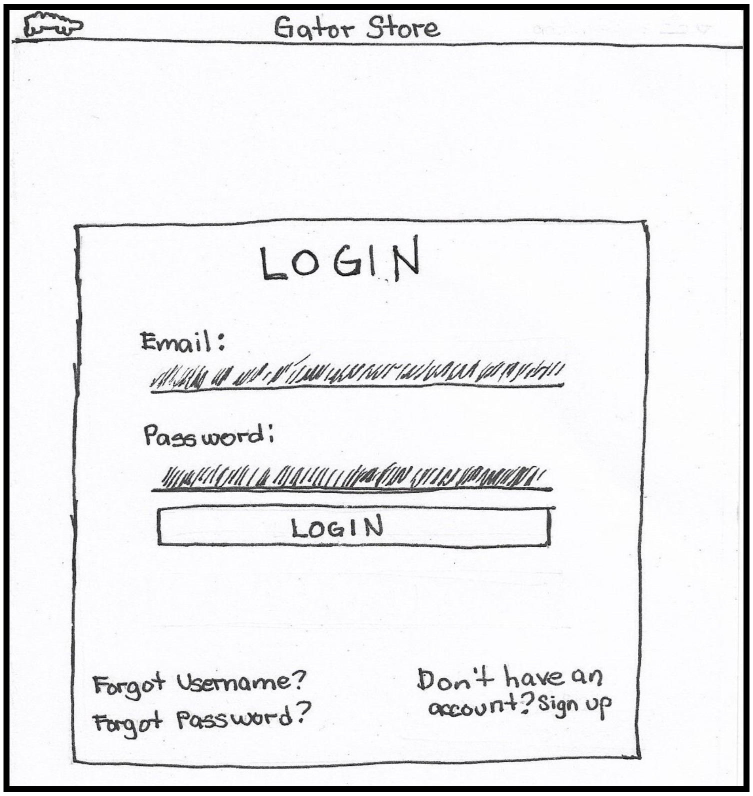


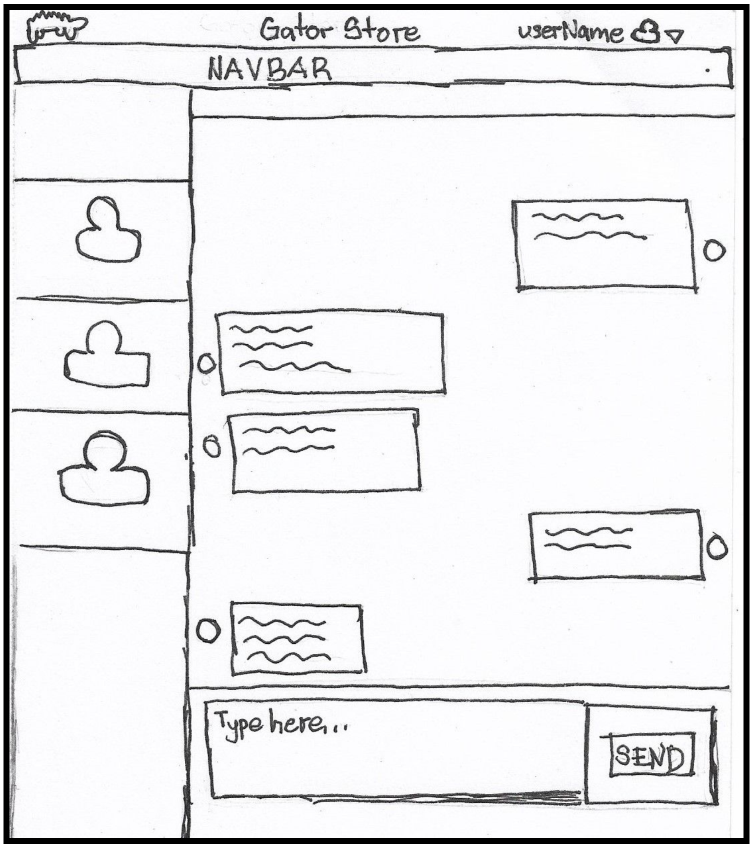
2.3 Sylvester goes to the website to sell items and is then prompted to fill up the form and then sign up.





2.4 Sylvester logs in to view messages.





3. High level Architecture, Database Organization

DB Tables:

1. **Entities**: Users
   Attributes:
   - User_Id(primary key)
   - Email
   - Password
   - Role

2. **Entities**: Items
   Attributes:
   - Item_Id(primary key)
   - Title
   - Photo
   - Price
   - Category_Id(foreigh key)
   - Description
   - Tag
   - User_Id(foreign key)

3. **Entities**: Category
   Attributes:
   - Category_Id(primary key)
   - Title

4. **Entities**: Message
   Attributes:
   - Message_Id(primary key)
   - Content
   - Time_Stamp
   - User_Id(foreigh key)

Media storage:

Images will be stored using the file path in the data field “photo” in table items.

Search/filter architecture and implementation:

Our website's search uses the simple MySQL filter query to access requested content on the database. The query is created by the user who fills out the searching bar then hits the submit button to search for specific items. Items can be filtered based on name, category and other related information(like course number).

In backend, data can be queried by using MySQL LIKE operation. For example:
SELECT Price FROM Items WHERE Category_Id LIKE '1%';

APIs:

No API will be used.

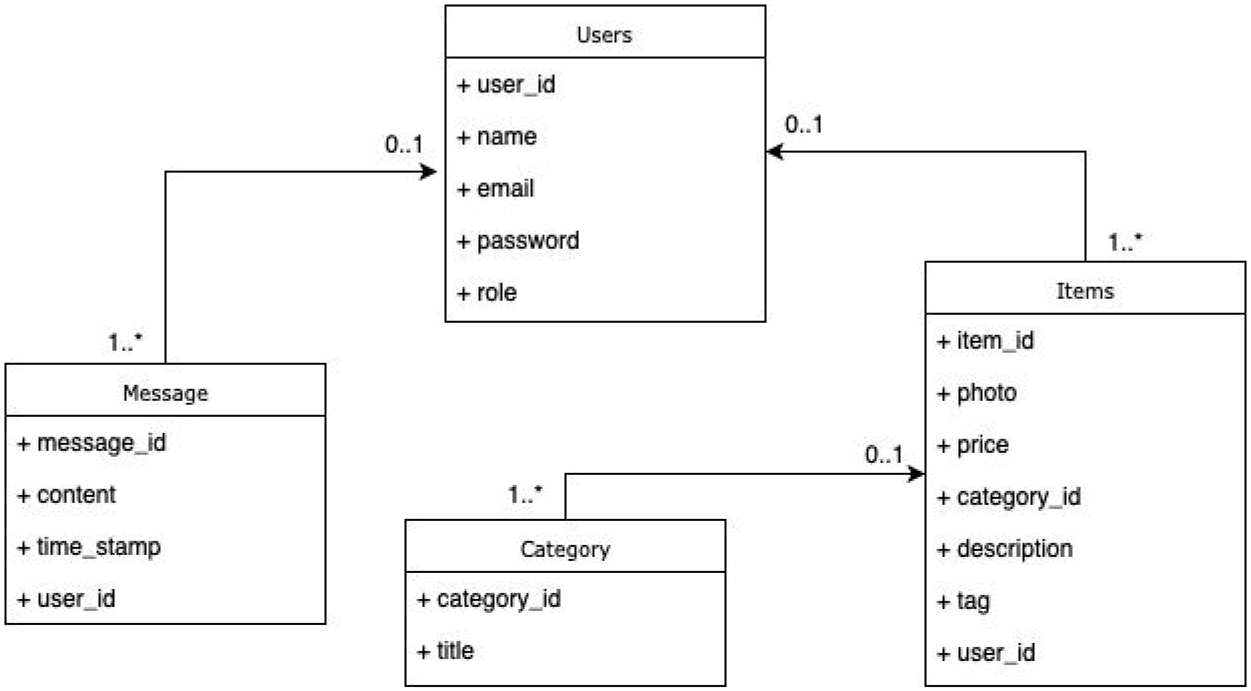
Significant non-trivial algorithm or process:

No significant non-trivial algorithms or process will be used.

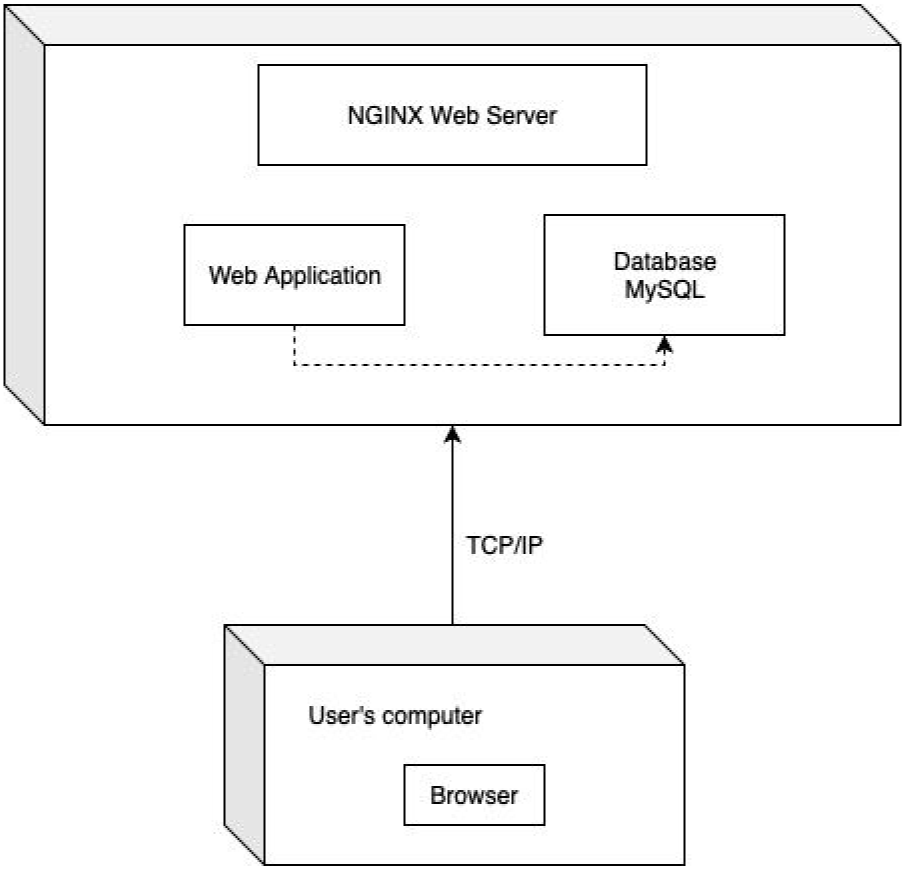
Any changes in Software:

There is no significant change to the software stack at this time.

4. High Level UML Diagrams



Deployment diagram:


NGINX Web Server
Web Application
Database
MySQL
TCP/IP
User's computer
Browser

5. Identify actual key risks for your project at this time

- Skills risks:
  Team members are all well informed with web development concepts and practices needed to complete the project.
- Schedule risks:
  Each team member is responsible for making sure they complete tasks on time before the deadline so there are no schedule risks.
- Technical risks:
  There are no technical risks that will affect the project because all team members will inform each other when there are any potential risks.
- Teamwork risks:
  All team members may be busy with other commitments such as school and work which makes finding a good time to meet outside of class more difficult.
- Legal/content risks:
  Content posted such as images shall be approved by Admin before appearing on the website. Those images must be cleared for any copyright issues and need proper licensing before posting.

6. Project management

In milestone 2, we chose two members from back-end to take charge of vertical prototype and the rest of us write documentation. We divided the work according to the workload of each subsection. Milestone 2 documentation has six subsections and among them, the second and fourth part have relatively harvier workload, we assigned two people for each of them and other members are responsible for two subsections respectively.

In future, after M2 is approved, we are planning to start working based on whether it is for frontend or backend. Each team member will be given a specific task to follow. We have started to use github issues and projects to manage our tasks. After each milestone has been distributed, we will create specific issues and assign them to respective team members. From github project, every team member can easily know the process of every issue, whether it is just started, in processing or finished.