**SW Engineering CSC648/848**

**ADSPets Adoption and Blog Site**

**Section 04 Team 01**

Khadija Khan - Team lead

Qin Chen - Backend lead

Katie Chang - Scrum master

Elyod Abraham - Frontend lead

Avitaj Singh Bhangoo - Github master

Brahamjyot Kaur - Frontend lead

**October 11, 2023**

| **Version** | **Date** |
| --- | --- |
| Milestone V.1 | 10/11/2023 |

**DATA DEFINITIONS V2**

| **Data Name** | **Definition** | **Usage** |
| --- | --- | --- |
| normal\_user | A user for our application that will be looking for pets, applying to adopt them, or making blog posts related to pet care or their adoption process. | Users will be sharing their information to be in contact with shelters regarding adopting pets. Their names will be needed to have post authors |
| registered\_animal | An animal will be any type of pet that will be set for adoption on our website. They will have information on their species/breed | List unadopted animals |
| post | Users can post some information for the animals. | Normal users can post the status of their animals.  Shelter users can post their animals. |
| Shelter\_user | The shelter owner. | They can add animal data into the database and approve the adoption application. |
| application | Normal users can request their application to shelter users when they want to adopt an animal. | To adopt. A bridge through users， shelter and animals. |

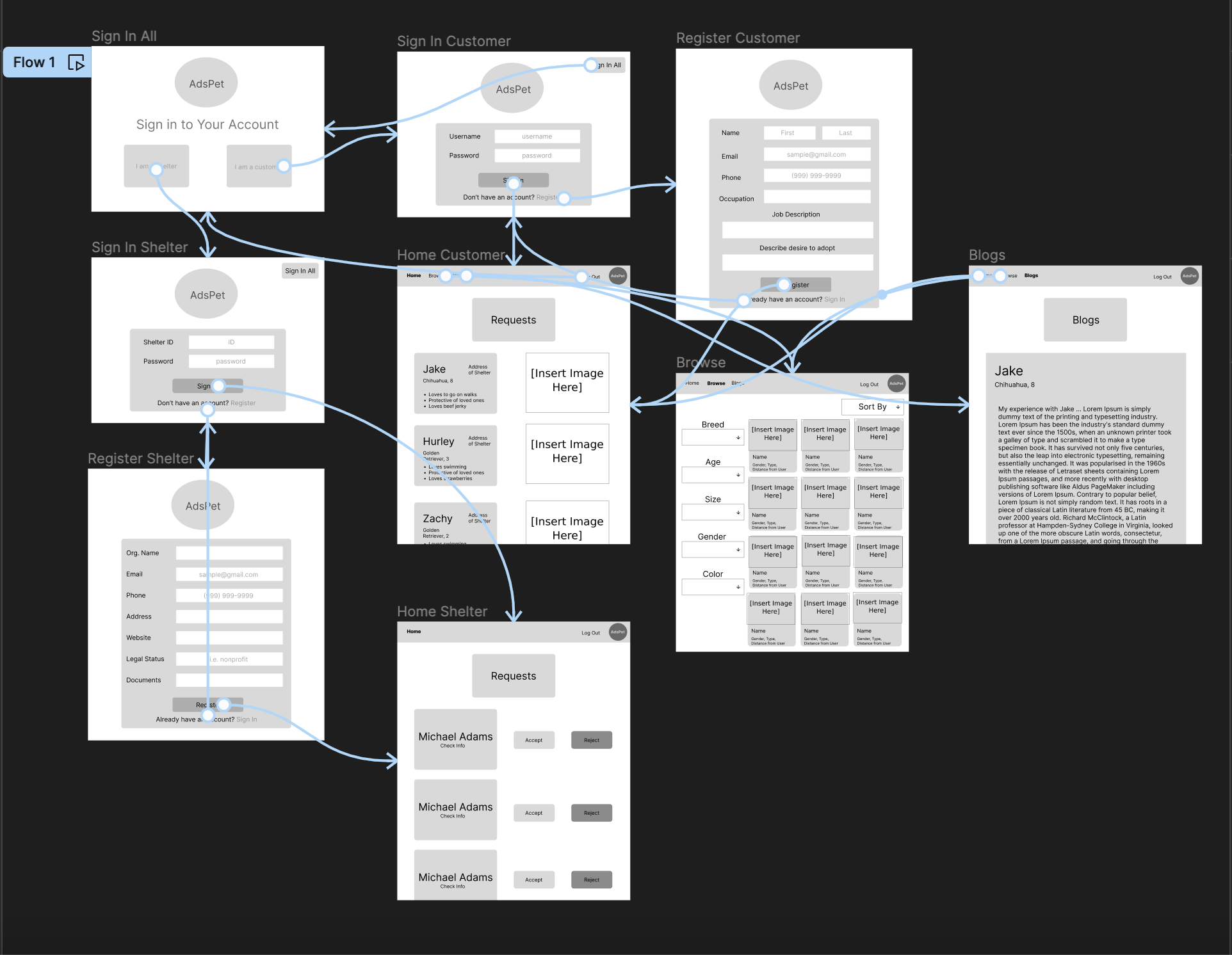
| **Data** | **Attributes** |
| --- | --- |
| normal\_user | id(PK)  Username(unique)  password  Firstname  Lastname  email(unique)  Age  Phone  Password  Occupation  desire |
| animal | id(PK)  Name  Species  Breed  Age  Status  Gender  image  comment |
| Post | id(PK)  Title  description  Picture  Createby  uid(FK)  sid(FK)  aid(FK) |
| Shelter\_user | id(PK)  Name  Address  email(unique)  Phone  website |
| application | id(PK)  sid(FK)  uid(FK)  aid(FK)  status |

**FUNCTIONAL REQUIREMENTS V2**

**Key:** 🟩Priority 🟨Desired 🟦Opportunistic

|  | **ID** | **Description** | **Details** |
| --- | --- | --- | --- |
| 🟩 | 1 | Users will be able to sign up and register for an account | 1.1. User will be able to sign in with this account |
| 🟩 | 2 | Shelters will be able to make shelter accounts | 2.1. Shelters will be able to sign in with this account |
| 🟩 | 3 | Shelters will be able to upload new pets to the website | 3.1. These pets will have information about breed/species, age, name, location  3.2. The pets will have a picture |
| 🟩 | 4 | Users will be able to submit applications for pets | 4.1. Users will fill out an online form  4.2. Users will know if they are approved or denied through their account profile |
| 🟨 | 5 | Shelters will be able to approve or deny applications for pets | 5.1. The application will be shown on the shelter profile  5.2. The shelter can approve or deny applicants  5.3. The shelter can have preset questions that approve or deny applicants |
| 🟩 | 6 | Registered users can make posts on the website blog | 6.1. Posts will need a title and tag about what it is about  6.2. Posts can be about animal care, adoption, or advertisements |
| 🟩 | 7 | Guest users will be able to see posts and pets but not apply or post | 7.1. Guest users won't be able to apply or make posts until they make an account |
| 🟦 | 8 | Shelters will be able to update pet status | 8.1. Shelters can say if a pet is pending adoption or has been adopted  8.2. Shelters will be able to add in new information on an animal |
| 🟦 | 9 | Users will be able to change their profile and get a list of compatible pets | 9.1. Users will be able to include what kind of pet they are looking for |
| 🟨 | 10 | Users will get an application update on their profiles | 10.1. Users will be told if they were approved or denied for whichever pet they applied for |
| 🟨 | 11 | Shelters and users will need to pass a pre-screening questionnaire to make an account | 11.1. If they don’t answer the questions properly then they won’t be able to make an account  11.2. Used to make sure that bad people and shelters don’t do business |

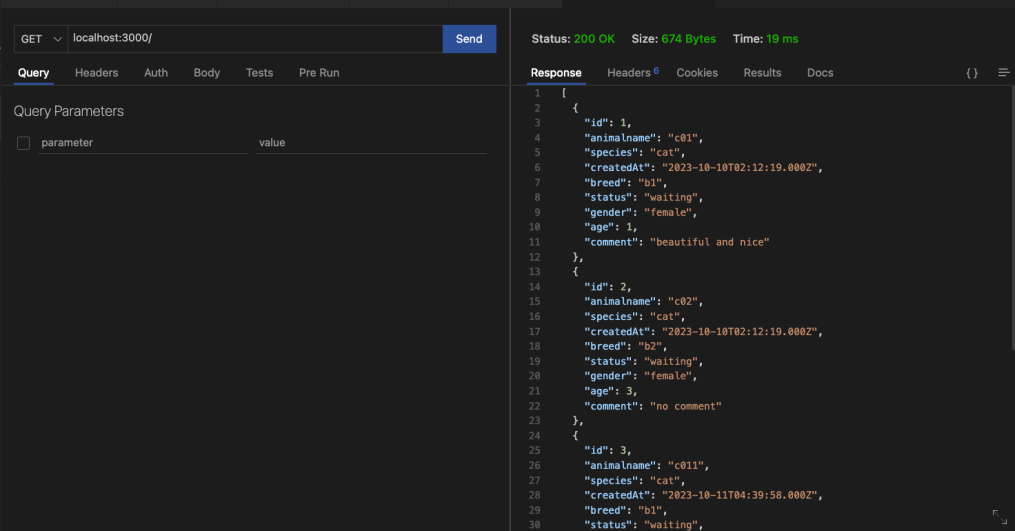
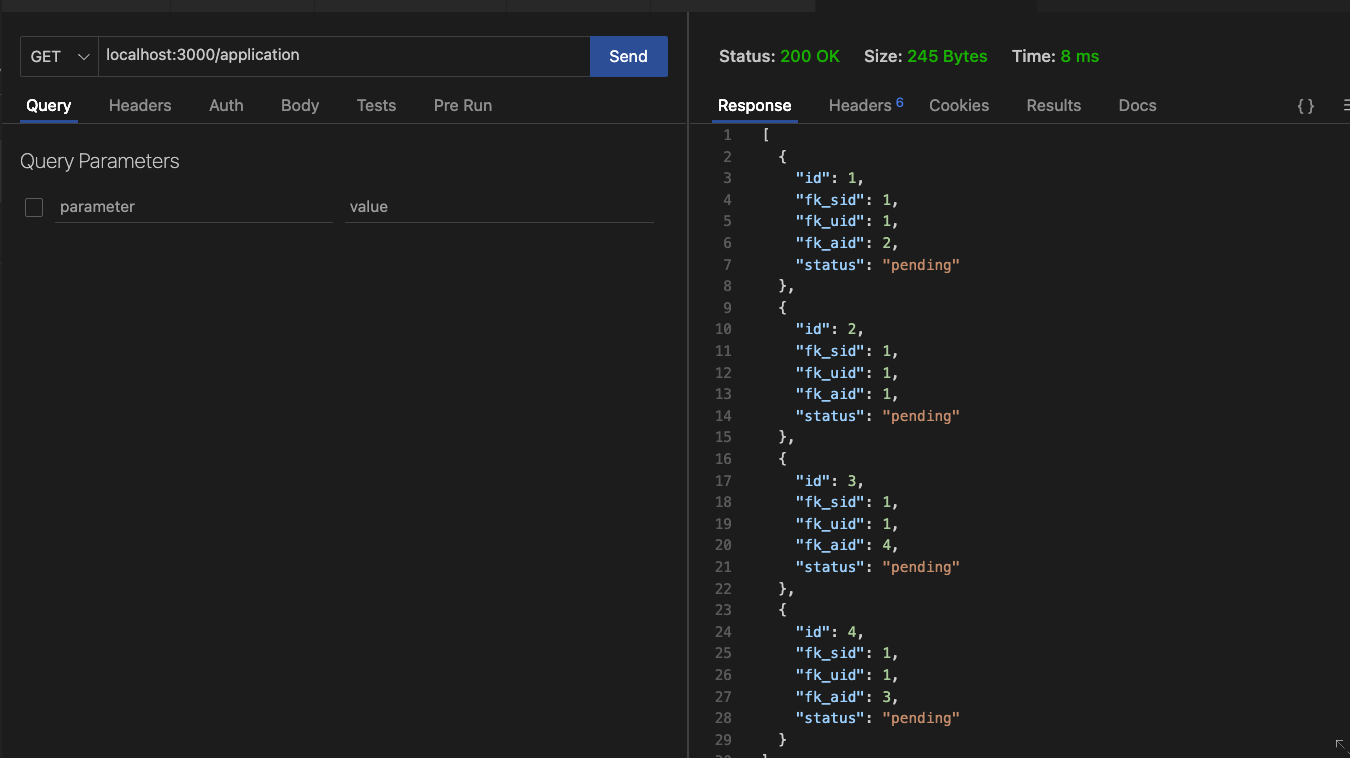
**UI MOCKUPS AND UX FLOWS**

****

Note: Current mockup requires one to sign in to browse the app but this will likely change in the next update.

UX Principles: Meets the necessary UX principles and the emphasis of these mockups was to ensure that. Although some of these design ideas may change, this draft shows a good idea of how we are emphasizing cleanliness and simplicity.

**APIs**

* When the clients try to visit the home page”/”, the back end will pull out the database and get all the elements from the animal table. Then translate the data into the array and send the array to the front end. 
* Some as above, but this time is for the “/application” router, and the back end pull out the application table for this time.

**Authentication**

* User registration: /register
* User login: /login
* User profile: /user/{id}
* User logout: /logout

**Pet Management**

* Get all pets: /pets
* Get pet by ID: /pets/{id}
* Search pets by filters: /pets?filter=value

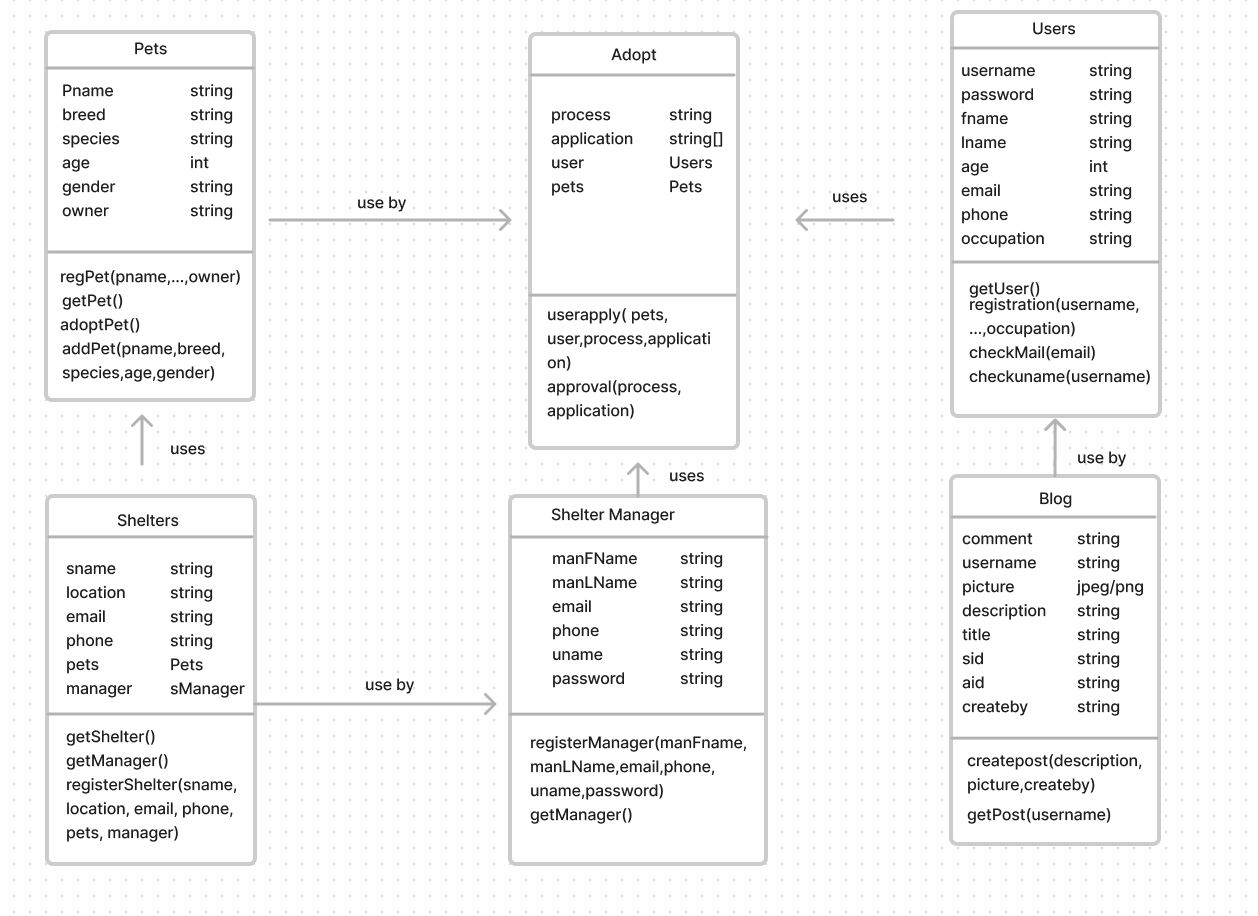
**Shelter Management**

* Create a new shelter: /shelters
* Add pet: /shelters/pets/{id}
* Delete a shelter: /shelters/{id}
* Get all shelters: /shelters
* Get shelter by ID: /shelters/{id}

**Blog Management**

* Create a new blog post: /blog/posts
* Retrieve all blog posts: /blog/posts
* Retrieve blog post by ID: /blog/posts/{id}
* Delete a blog post: /blog/posts/{id}

**High Level UMI**

****

**Note \*\* this is our current High Level class UMI it can subject to change.**

**KEY RISKS**

**Individual risks**

* External circumstances, like personal emergencies or commitments from other classes, can disrupt a team member's study schedule, potentially delaying feature implementation.
  + To resolve:
    - Begin preparations in advance and avoid procrastination
    - Strategize and schedule ahead of time.
    - Communicate with the team as soon as possible so that the team can plan accordingly
* When a team member faces a critical code error and gets stuck on it , it can lead to frustration, increased stress, and result in postponement of features and milestones.
  + To resolve:
    - Ask for help from team members.
    - Ask for guidance from the Professor or class tutor, if needed.
    - Ensure to clearly understand the current code before progressing.
    - Seek clarification in the team discussion (or Discord channel) if any code segment is unclear.

**Skills issue**

* If a team leader doesn't clearly define the deliverables for an upcoming milestone, it can lead to delays in subsequent tasks.
  + To resolve:
    - When determining deliverables, take into account comments from instructors about prior milestones.
    - Become familiar with the priorities, additional obligations, and availability of your teammates.
    - Create plans that are adaptable and can be adjusted as needed.
* Some team members might be technically proficient but lack the necessary team collaboration skills or soft skills to work effectively in a team setting, leading to potential conflicts or miscommunication, decreasing the overall productivity of the team.
  + To resolve:
    - Conduct team-building activities like playing games, asking a fun question that everyone answers at the start of meeting or having coffee together.
    - Promote open communication and feedback within the team.
* Learning a new concept, specific software or methodology can become more time-consuming than anticipated, this could result in delaying of the project features.
  + To resolve:
    - Begin preparations ahead of time.
    - Study a range of sources and manuals to solidify basic concepts.
    - Team members should share the resources that helped them in the group chat.

**Teamwork risks**

* Interpersonal conflicts can be detrimental for the project and decrease team morale. Disagreements or personal issues between team members can affect project progress.
  + To resolve:
    - Set clear team norms and expectations for behavior.
    - Address conflicts promptly and professionally.
    - If needed, take into consideration talking to the team lead or scrum master.
* There can be issues during code integration if a team member changes the decided plans without telling team members. Integrating work from different team members can lead to bugs or conflicts.
  + To resolve:
    - Use version control best practices e.g., regular pull requests, code reviews, using branches .
    - Set clear coding standards and guidelines to make sure everyone is on the same page.
    - Github master should regularly do code integration testing by communicating with the team members.
* Tasks that are handed off between team members might lack clarity or completeness, making it hard for the team members to understand code functionality and work on it.
  + To resolve:
    - Code review and document (comments and feedback) handoff processes .
    - When files start getting complicated, start organizing and segmenting the code.
    - Allowing time for questions and clarifications.
    - learn and follow the code style documentations for the prog. language that the team use
* When miscommunication leads to two team members tackling the same project feature, it can cause delays in other tasks.
  + To resolve:
    - Communication is the key to success.
    - Ensure all members are updated on individual responsibilities for each task and milestone at every weekly meeting.
    - Using Trello application regularly , where our Team-lead assigns roles and tasks for each team member.

**PRODUCT MANAGEMENT**

In our team, we have three frontend leads and two backend leads that have been working together to get our web app running. To assign these tasks, I (team lead), divide the tasks before our weekly Monday meetings and then also put them on our Trello page so that everyone can remember what tasks they were assigned. Besides assigning things on our Trello page, I also asked our team to post their daily progress on our discord page so we know what they are doing and what exactly is happening with their code.

I believe we have some issues about people sharing when certain tasks are done and what exactly was done. When some new code is pushed, it is often a bit difficult to understand since there are issues with sharing what is exactly going on. Since our meetings are on Monday, it is usually focused on what we are going to do in the upcoming week and then on Wednesdays in class, we check in on how we are doing and through the week. On Monday’s, I will also start to ask what progress was made along with what new tasks we need to do. One issue we had with this milestone is that we are still dealing with the issue of only having a local database and not having our webpage setup.