SW Engineering CSC 648 Fall 2019

Section 2

Team 8

Month NN, 2019

Milestone 2

Team:

- Oleksii Butakov -- Team Lead / Front-end Lead (obutakov@mail.sfsu.edu)
- Andrew Copas ------ GitHub Master / Front-end
- Alex Lee -------------------------------------- Front-end
- Emanuel Saunders ------------------ Back-end Lead
- Kevin Truong -------------------------------- Back-end
- Tim Lei -- Back-end

History Table:
10/10/19 - Document created
10/19/19 - M2 submitted for review

Content

1. Functional Requirements - prioritized

The website's functional requirements have been trimmed and readjusted to fit better in-line with our projects data-entities, being more clear and concise. Numbering remains the same as the original requirements.

Priority One:

Unregistered Users:

      2. Unregistered users shall be able to browse the posts.

      4. Unregistered users shall be able to access posts details.

      5. Unregistered uses shall be able to register.

Registered Users:

      7. +Functions of unregistered Users.

      8. Registered users shall be able to login.

            8.1. Registered users will require an Email and Password consistent with register information to login.

      9. Registered Users shall be able to make posts.

            9.1. Posts shall require a name, photo, price, description, pickup locations, contact information, and category.

      10.Registered Users shall be able to buy items from posts.

      16. Registered users shall be able to choose their pickup location.

Admin:

      18. Admin shall be able to accept selling requests.

Priority Two:

Unregistered Users:

1. Unregistered Users shall be able to search.

Registered User:

10. Registered Users shall be able to make modifications on their posts.

15. Registered users shall be able to contact the seller.

Admin:

17. Admin shall be able to reject selling requests.

Priority Three:
Unregistered Users:

3. Users shall be able to filter posts by category.

Registered Users:

11. Registered users shall be able to request take down of their posts.

    11.1 Posts shall be taken down by default after 60 days.

14. Registered users shall be able to buy many items at once.

Admin:

19.Admin shall be able to take down selling posts.

20. Admin shall be able to delete accounts.

2. UI Mockups and Storyboards

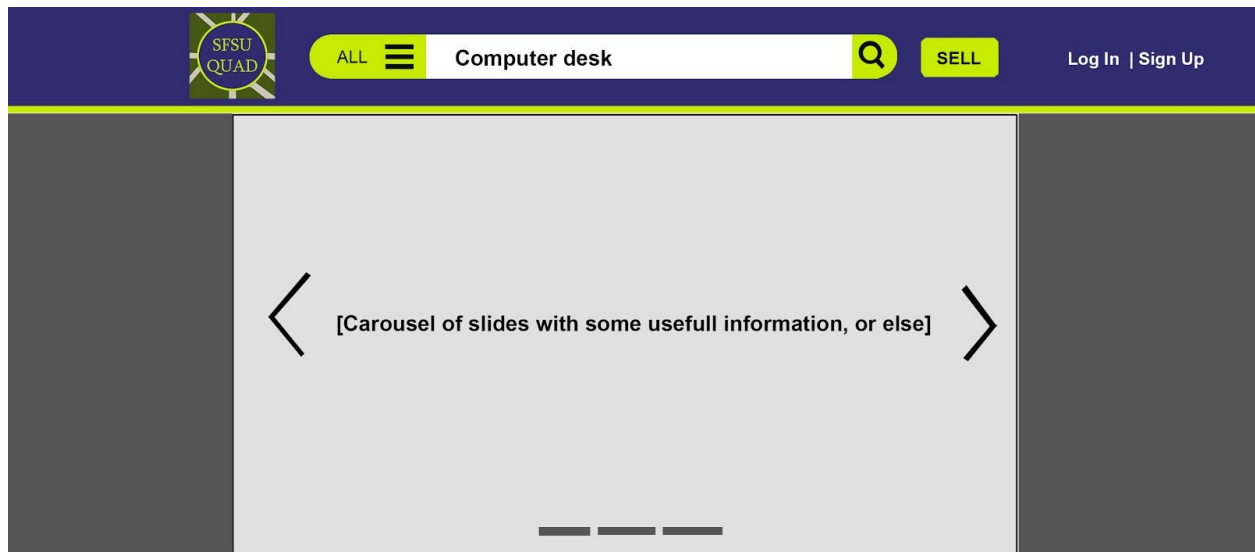Jason is an SFSU student and part time worker.
   About Jason:

- Busy
- Works at night and studies during the daytime from 9-5.
- Wants to utilize the purchasing of most pristine conditioned items from other students mainly due to the low budget costs compared to other shopping methods.
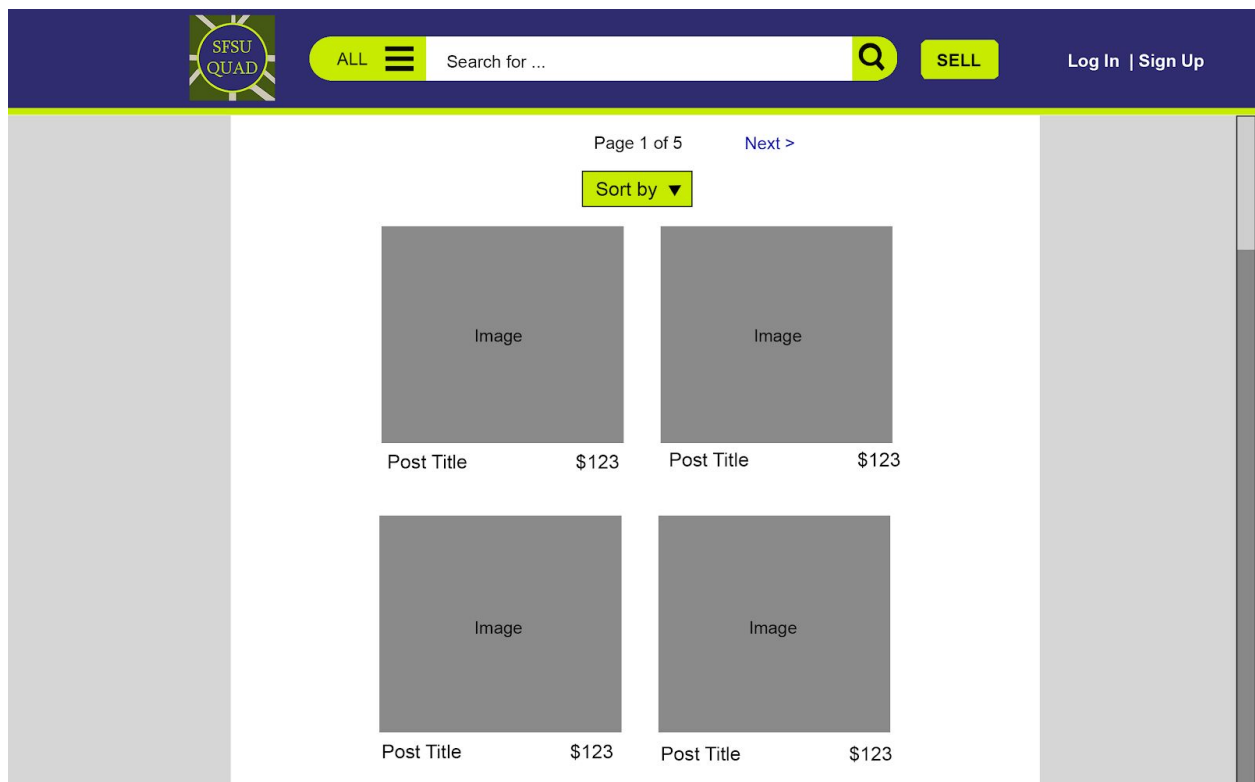
Use cases:

1) Jason uses the search bar to search for a specific item, and review the results.
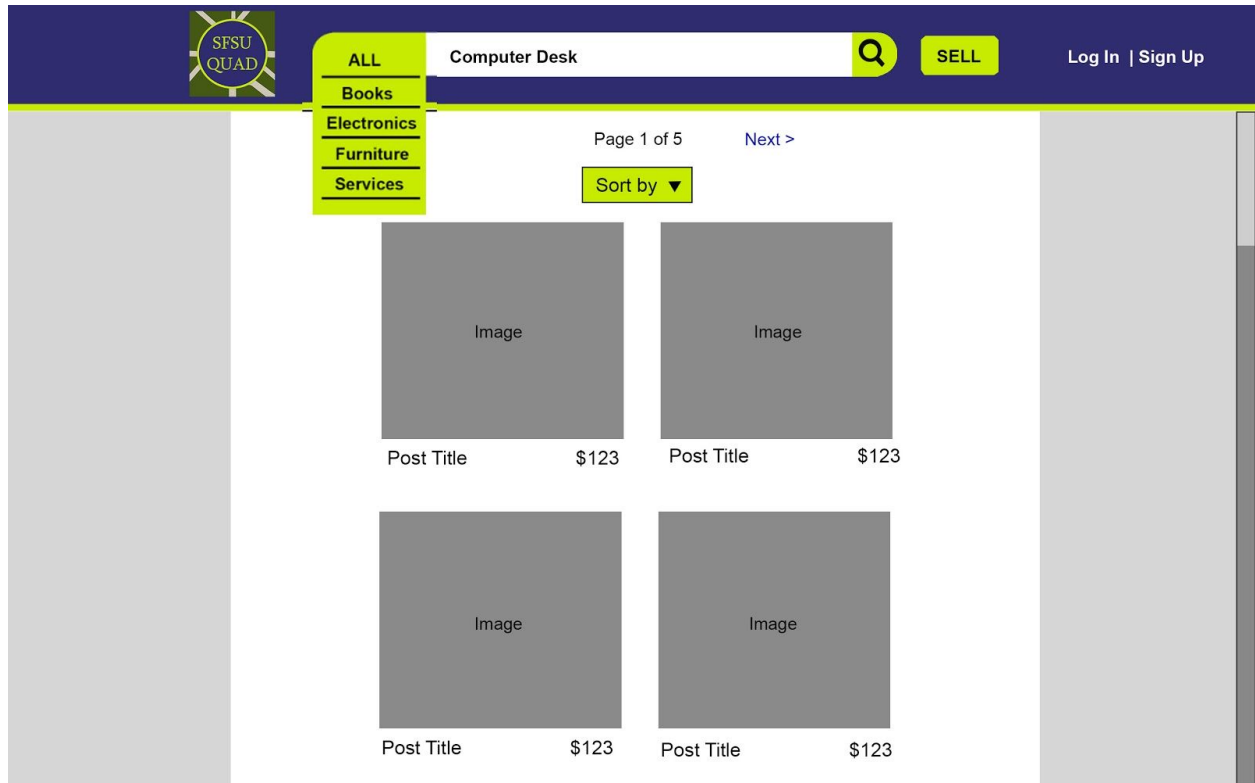   a) Jason types "Computer desk" query into the search bar, and press enter (or clicks on a magnifier icon)

ALL

Computer desk

SELL

Log In | Sign Up

< [Carousel of slides with some usefull information, or else] >

Items Currently for Sale

b) He reviews the results

SFSU
QUAD

ALL

Search for ...

SELL

Log In | Sign Up

Page 1 of 5   Next >

Sort by ▼

Image

Image

Post Title $123   Post Title $123

Image

Image

Post Title $123   Post Title $123
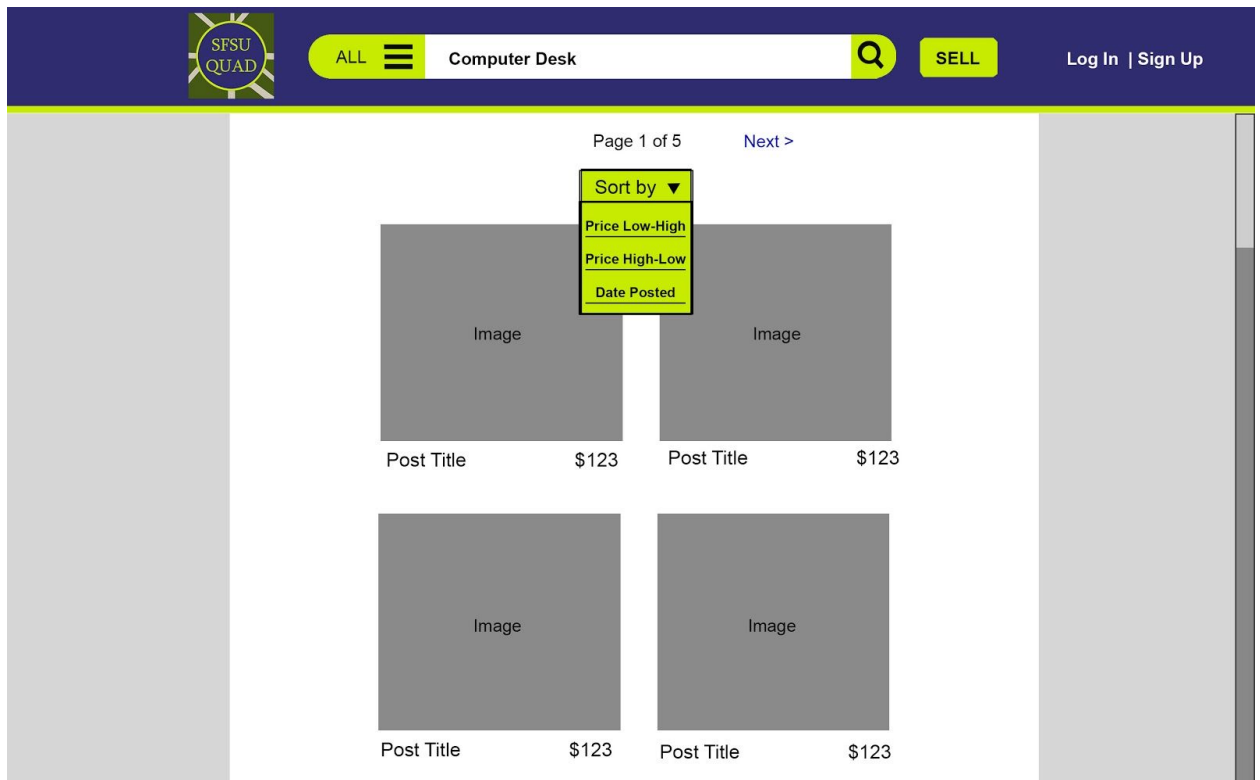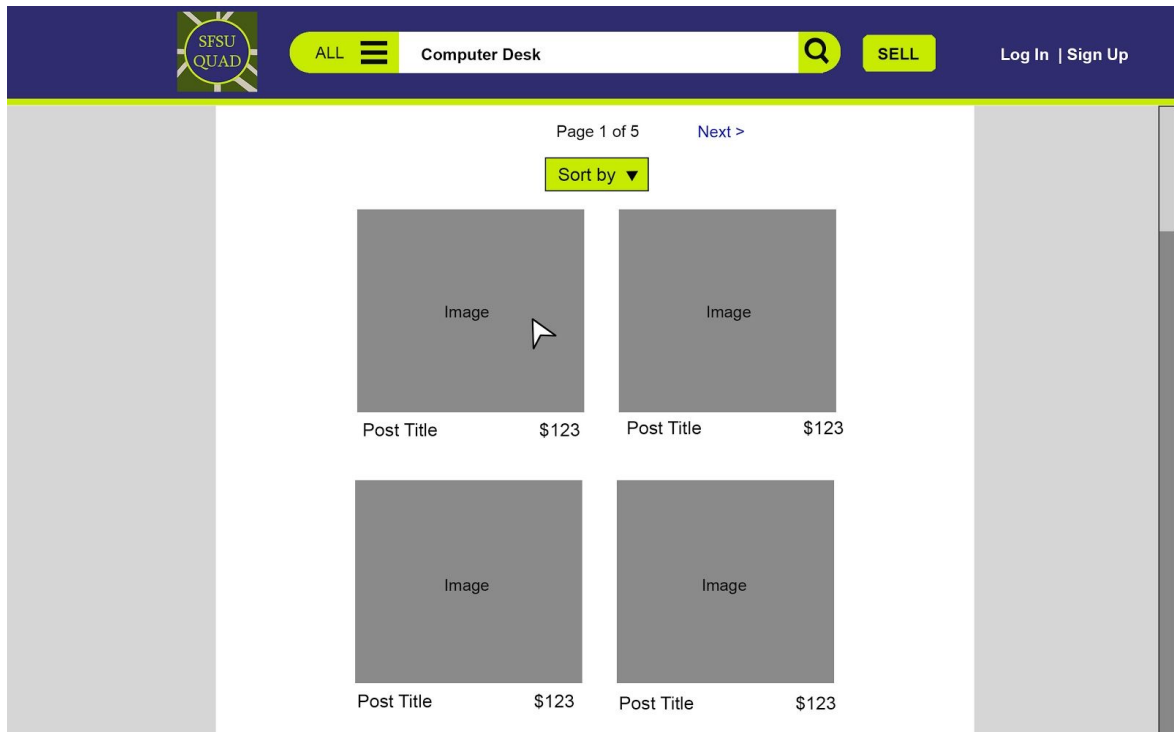
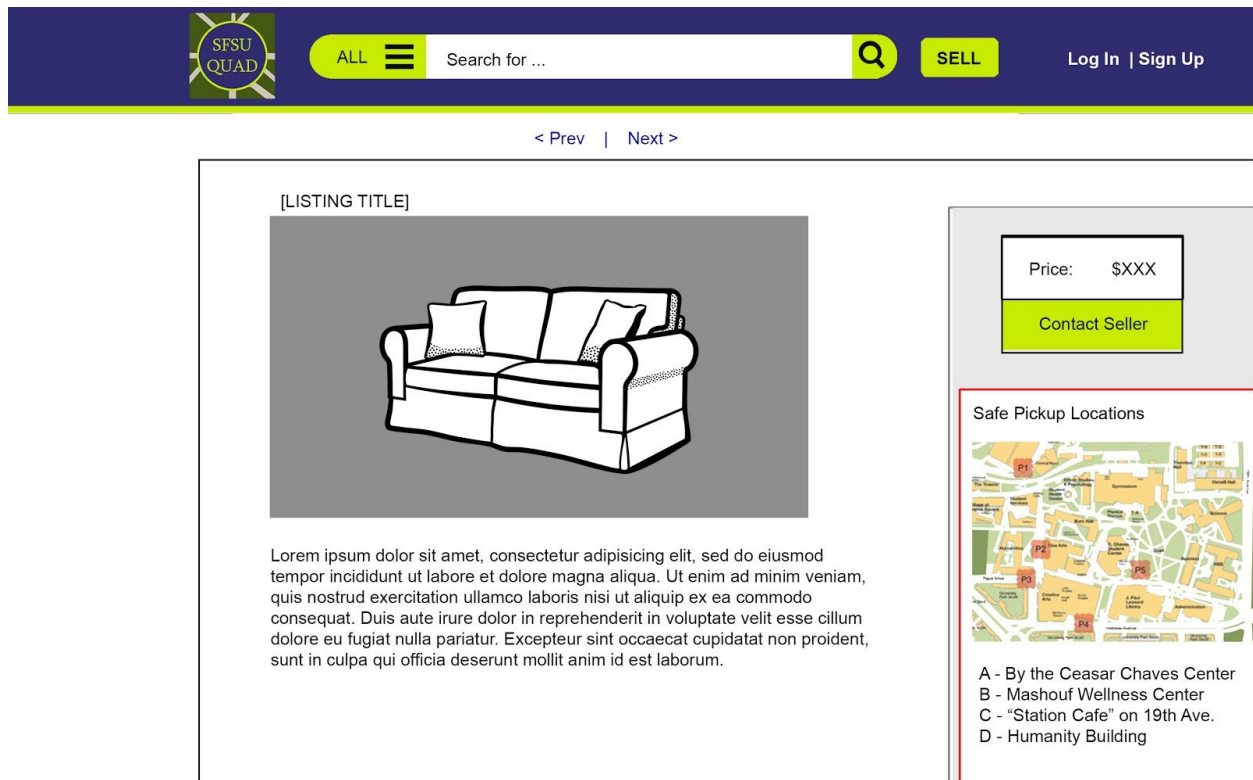2) Jason uses the category tab if he wants to narrow down the results.

3) Jason uses sort button to refine the results and searches for the most affordable option

4) Jason selects item posting, reviews it, and then contacts the seller for more information on item he is considering.

    a) Jason selects the item from the results that he is interested in

    b) Jason reviews the item

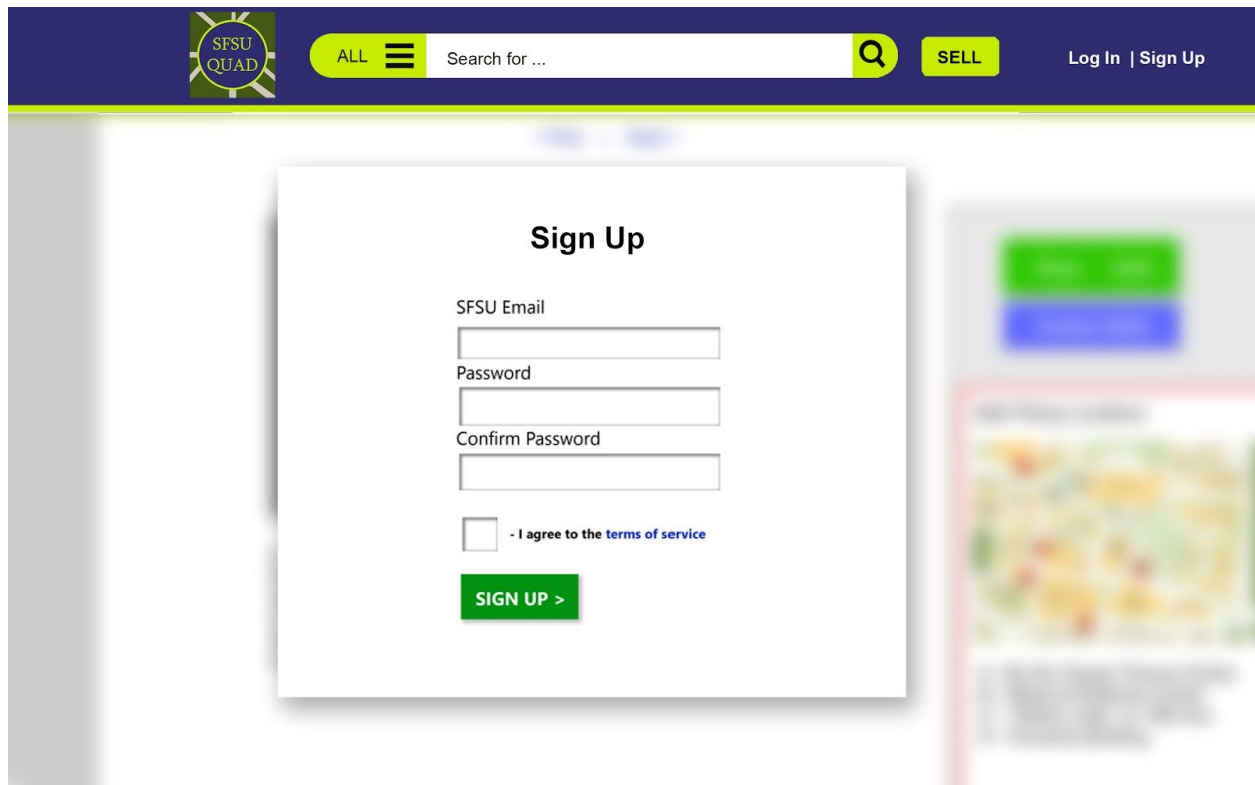c) Jason decides to contact the seller and presses "Contact Seller" button. Since he is not logged in, website offers him a chance to sign-up or log-in before showing seller's contact info.
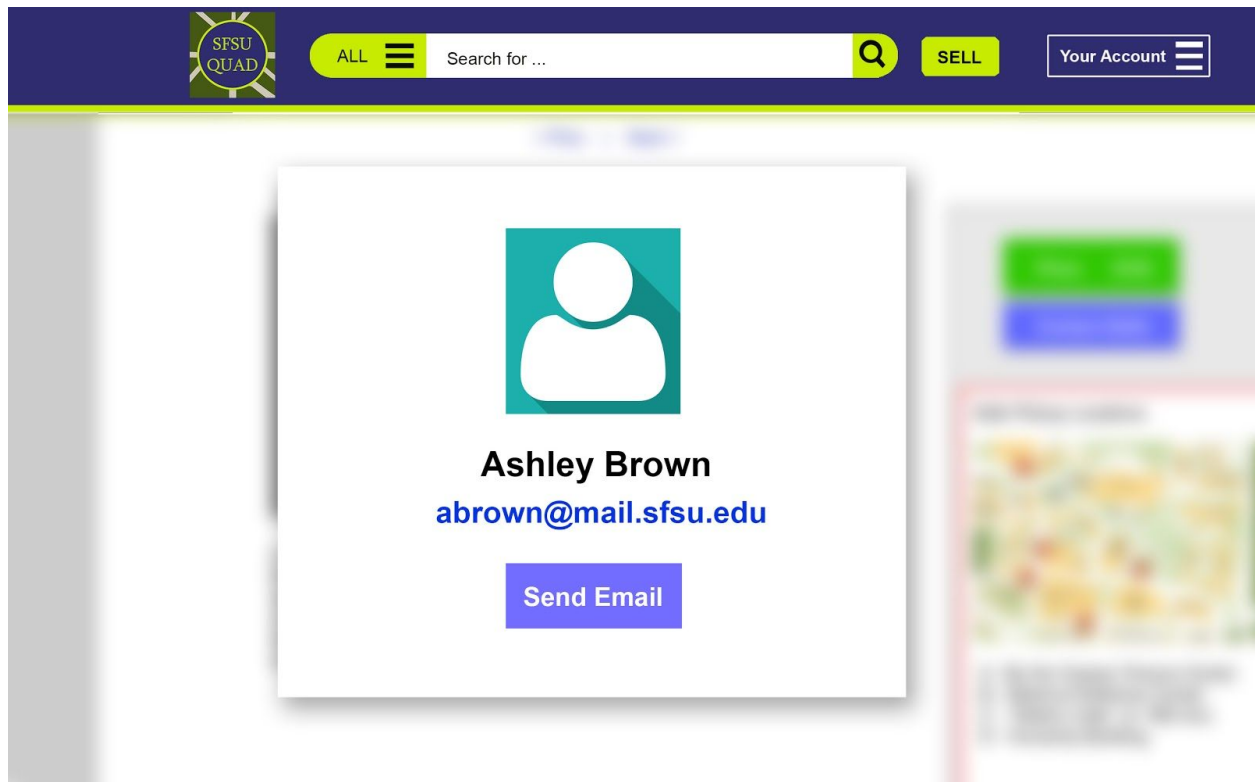

d)
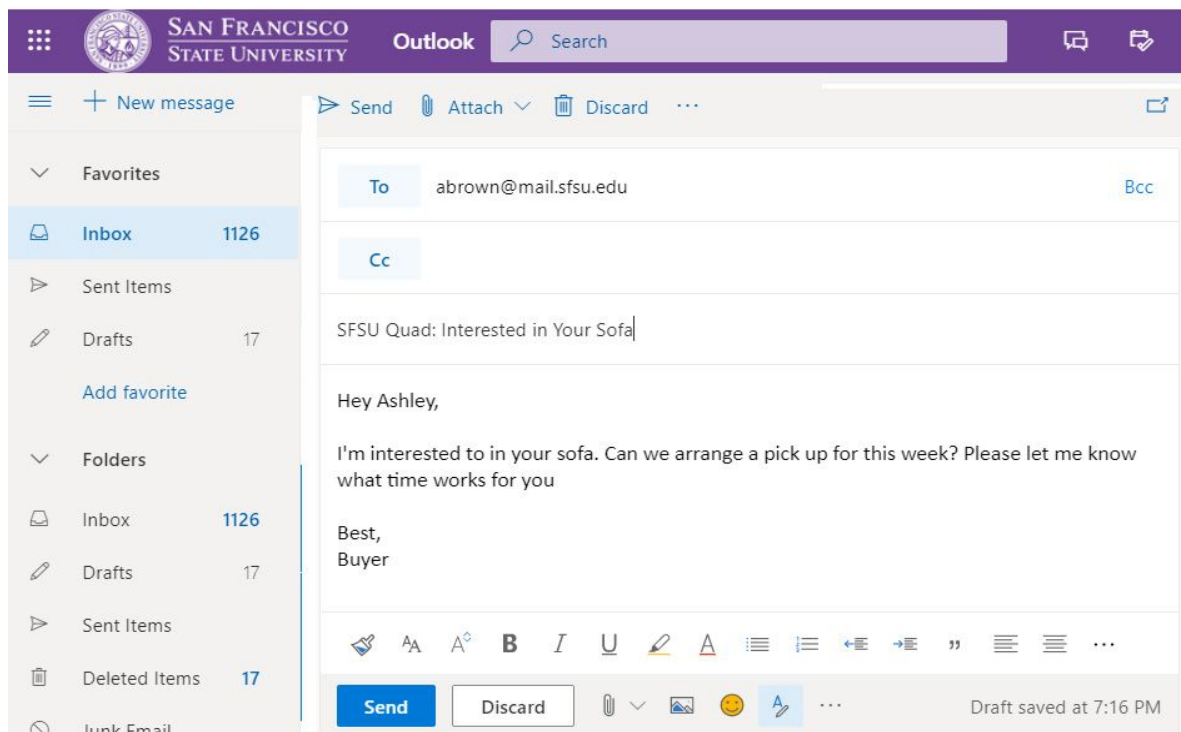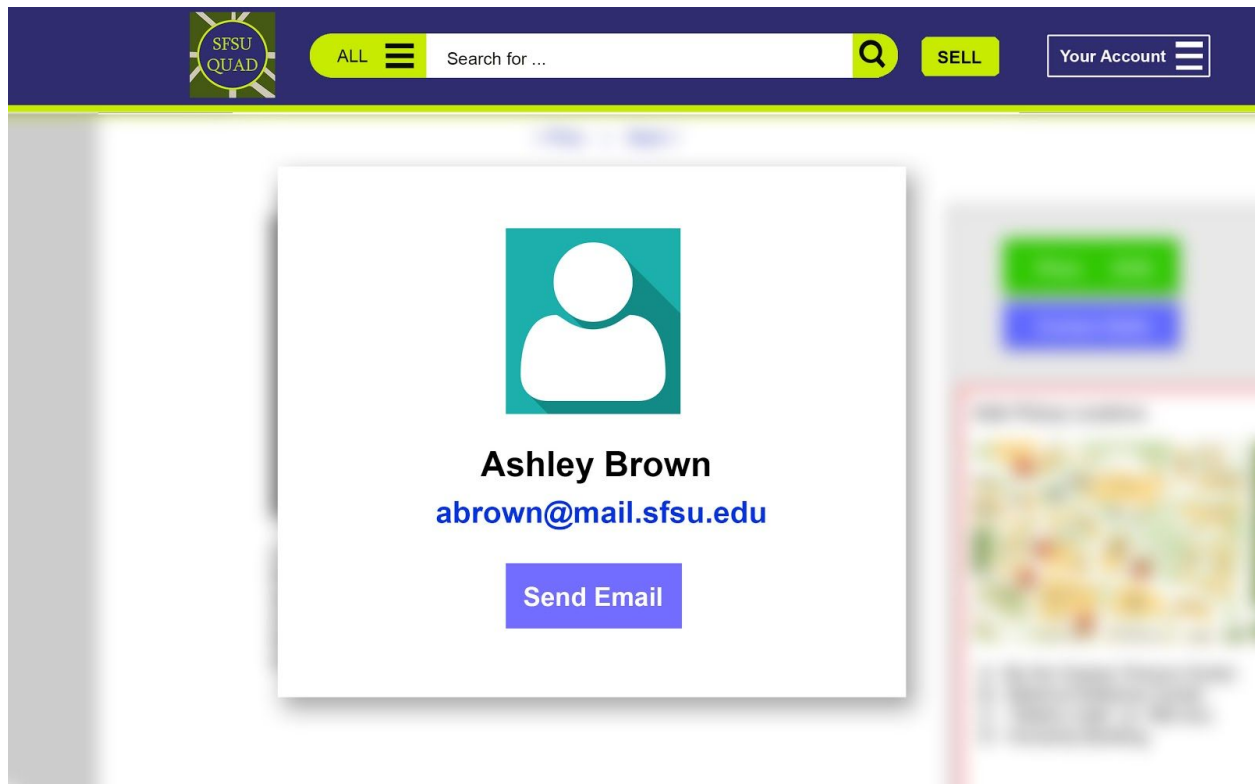i) If he chooses Log In option
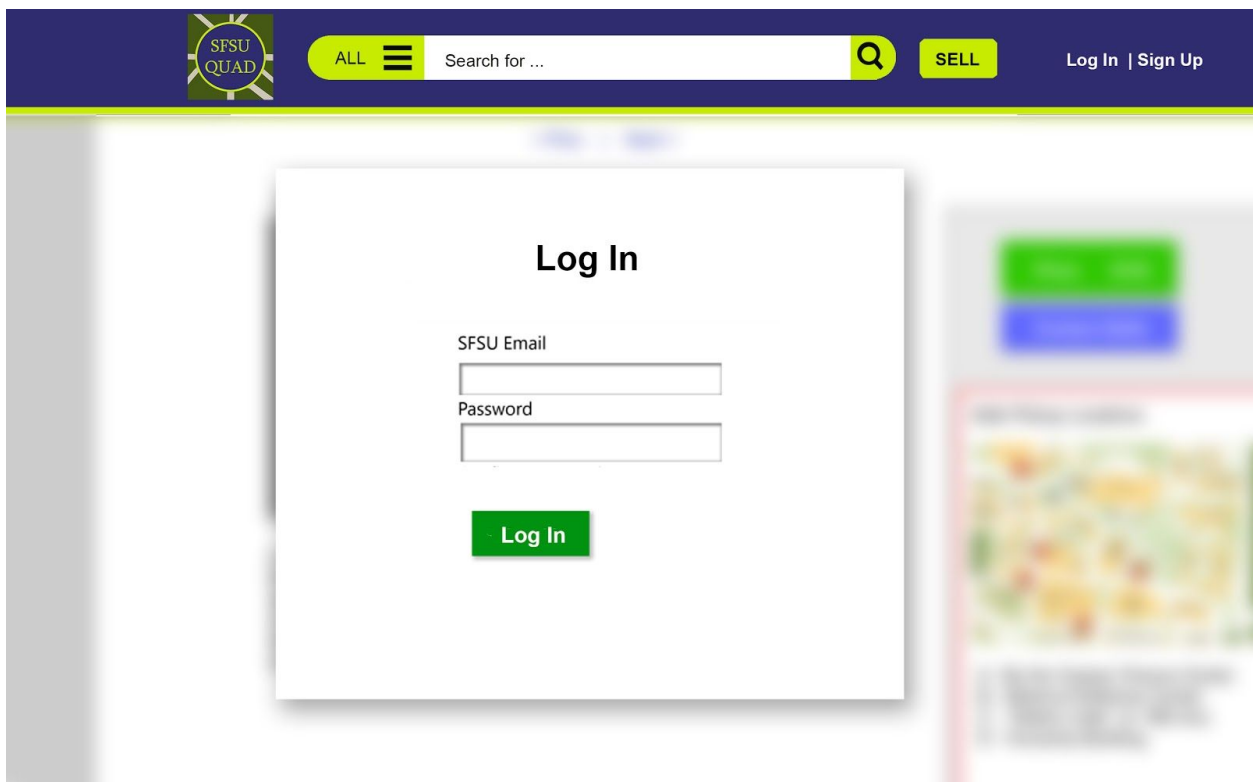
ii) If he chooses Sign Up option

e) After signing up or loggin in, Jason will be given seller's contact info

5) Jason contacts the seller to arrange a meeting time on campus.

3. High level Architecture, Database Organization

DB organization:

Registered Users:

- User ID : Unique ID which identifies the user
- Email : Used for login
- Password : Used for login
- Nickname : Name displayed
- Admin (Boolean) : Identifies whether user is admin

Post:

- Post ID : Unique ID which identifies the post
- Name : Name of the item
- Price : Listing price for the item
- Photo : Images of the photo
- Description : Details about the item
- Seller Contact : Seller's contact information
- Pickup Locations : Available pickup locations
- Category : Foreign key to the category table

Category:

- Category ID : Unique ID which identifies the category
- Item category : Category name for item

Our database will have three tables: registered users, post, and category. The Registered Users table contains attributes of user id, user SFSU email, user password, a boolean field to determine whether or not a registered user is an admin and the nickname that the user wants to show . The Post table contains attributes of item name, item price, photo(s) of item, item description, seller's contact information, pickup locations for the item, and category the item is in. The Category table contains attribute of the category of the item.

Media Storage: We will only allow users to add photos and store them in the file system. The way the photos are stored is by storing in a column named "photo" in the Post table using relative link to the location of the photo in a file system.

Search algorithm: For the searching and filtering architecture, we will be using SQL and %like to implement the searching and filtering functions. The user will enter the name of an item or a type of item into the search bar so the backend can run a query

to search the database using %like so that relevant results will pop up to the user. We are planning to implement Flask - MySQLDB.

We have no plans to create/use any new API at this point in time.

By default, the posts are organized and displayed by the order of time added to the database from most recent to latest.

4. High Level UML Diagrams

```
                    ┌─────────────────────────────┐
                    │      Unregistered User       │
                    ├─────────────────────────────┤
                    │                             │
                    ├─────────────────────────────┤
                    │         +search()           │
                    │         +browse()           │
                    │         +filter()           │
                    │         +register()         │
                    │         +sort()             │
                    └─────────────────────────────┘
                                 ▲
                                 │
        ┌─────────────────────────────┐      ┌─────────────────────────────┐
        │      Registered User        │      │            Post             │
        ├─────────────────────────────┤      ├─────────────────────────────┤
        │         -int uID            │      │         +int pID            │
        │         -String email       │      │         +String name        │
        │         -String password    │──────│         +float price        │
        │         -bool admin         │      │         +String photoLink    │
        ├─────────────────────────────┤      │         +String description  │
        │         -login()            │      │         +String location     │
        │         -post()             │      │         +int cID            │
        │         -editPost()         │      │         -bool approved       │
        │         -deletePost()       │      ├─────────────────────────────┤
        │         +buy()              │      │                             │
        │         +bid()              │      └─────────────────────────────┘
        │         +contact()          │                    │
        └─────────────────────────────┘                    │
                    ▲                          ┌─────────────────────────────┐
                    │                          │          Category           │
        ┌─────────────────────────────┐        ├─────────────────────────────┤
        │           Admin             │        │         -int cID            │
        ├─────────────────────────────┤        │         -String name        │
        │                             │        ├─────────────────────────────┤
        ├─────────────────────────────┤        │                             │
        │         +approvePost()      │        └─────────────────────────────┘
        │         -deleteUser()       │
        └─────────────────────────────┘
```
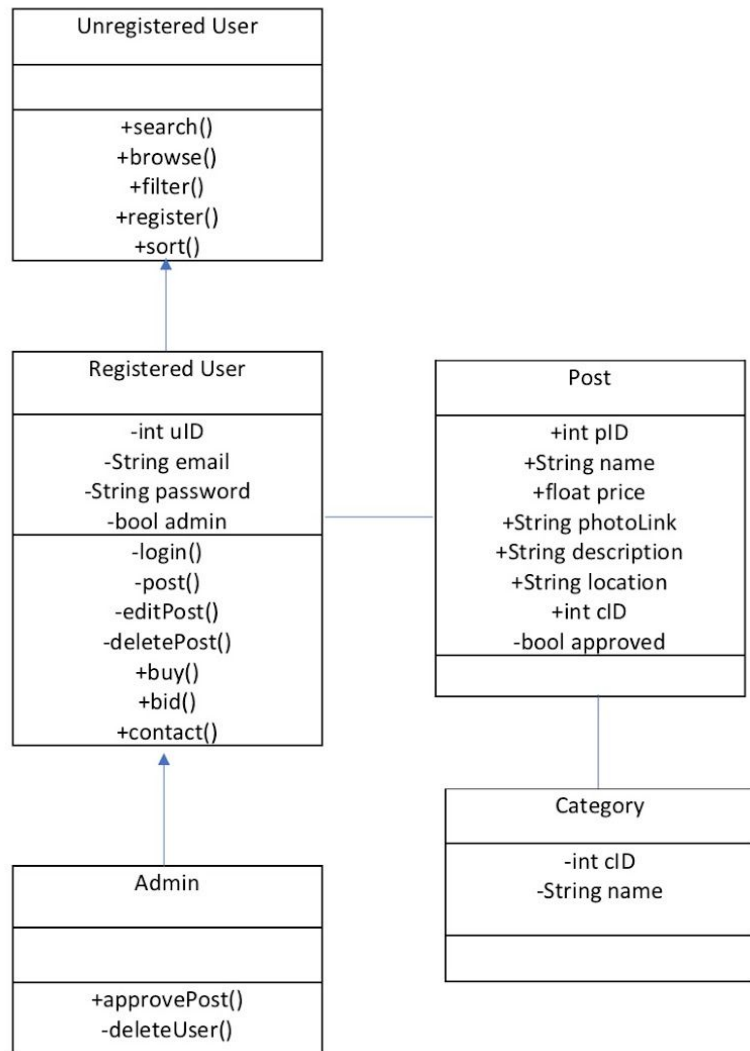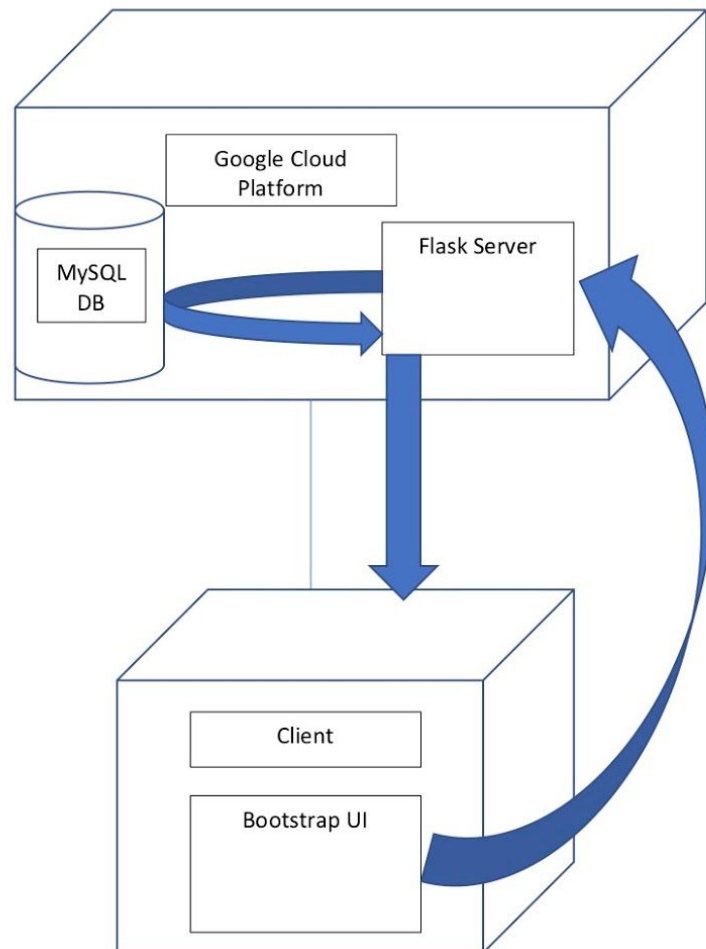
b) UML Component and deployment diagrams

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

Skill risks: Although team will have the current skills, sometimes certain components might require slightly higher calibration of skills, which can be challenging. Some tasks might even require outside of meeting work.
- Depending on the difficulty of tasks, we might need to allocate more time to the tasks even if we have the right skills to tackle it.
- Setting earlier deadlines before the official due date to be sure tasks are up to par then conducting final touches since some tasks are challenging even if possessing the right skills.

Schedule risks: Team members might not start on time due to complications. Sometimes personal issues can delay set schedules as well as programming issues that might arise unexpectedly.
- Use online chat to communicate with our teams, perform individual work outside of meetings in contribution to milestones.
- Set earlier deadlines in means of ensuring that the individual tasks are at least up to par to use time to polish up tasks before official deadlines set by CTO/CEO.
    - If necessary, under most extenuating circumstances, will request extensions from CTO to resolve any complications of tasks due to schedule and technical complications.

Technical risks: code or items programmed might fail suddenly or even last minute. Sometimes systems might work on one device but not other devices.
- Conduct multiple tests on various systems
- Conduct extensive tests for every piece of implementations

Teamwork risks: There can potentially be disputes between team members depending on scenario. Some ideas of team members can differ from others in team.
- Address issue at hand
- Conduct a survey of which ideas team would like to go with.

Legal/content risks: Might need to request permission to use certain contents of certain items depending on the content. Some content might not render well on certain interfaces.
- Try to find content related to projects where the content is royalty-free, or try to use clip art to describe the scope of project.
- Create original clips of pictures or content which would tailor to the project
- Test the content on interfaces that project will be tailored to.

6. Project management

We have started using Trello to assign the tasks to our team. Slack is still used to when it comes to chatting with our team members.

Trello is used to assign earlier deadlines prior to official deadline so we can make progress on the tasks at hand. Since there is a lot to cover with Milestone 2, we set these earlier deadlines to ensure our tasks can be up to par before the upcoming class meetings with CTO/CEO. That way we can get some feedback, and use the feedback to polish up whatever we need before the official deadline. We can also take a look at our assigned sections which would need to be completed or up to par prior to the assigned deadline by Team lead.

Slack is still used to communicate with the entire team, and to be informed of any circumstances that the team members might run into. There are circumstances that can possibly delay work progress or ideas which can help work progress. At times, we might use that to communicate the team of our progress, and for any guidance or tips on our tasks at hand. Sometimes we might show our images of our work or our renderings of the interface we are trying to go forth with. That way, we can get feedback from one another of our ideas outside of class meetings.

We have urged the team to start work on Milestone 2 immediately as soon as the document was online. During our last meeting, we have used the library board to sketch our user interface and functionalities. This way we have an idea of how to structure our website.