SW Engineering CSC648/848 Fall 2020

Application: SFSU Buy and Sell Website

Name: SFSU CONNECT

Team: 1

Robert Clarkson - team lead - rclarkso@mail.sfsu.edu

Kevin Fung - gitmaster

Cody Camp - backend lead

Taylor Luke - frontend lead

Gerardo Ochoa - developer

Hong Li - developer

Milestone 2

October 13, 2020

September 23, 2020	First Revision Done

Table of Contents

Executive summary

The idea behind 'SFSU connect' is to provide a platform akin to craigslist which facilitates the buying and selling of items and services. For example, say a student finishes a class and no longer needs a textbook, he will be able to post a listing detailing the book on our platform. Our website will facilitate the initiation of contact with other SFSU community members who are interested in acquiring the book. The uniqueness of our product stems from how niche the market we are looking to target is. By targeting a market as specialized as the SFSU community specifically, we can build a product that caters to their needs. For example, by creating a system which allows the users to filter by either class, course number or subject, they will be able to efficiently find listings relevant to their needs. Unlike other platforms, search results will not be bloated with irrelevant results such as textbooks which are not used at SFSU. For other items and services on the platform, a user can be assured that they can be procured on the SFSU campus. By limiting the ability to list items and contacting others users that have signed up, we can ensure that those participating are affiliated with the school and in this way we can provide a measure of security for our users.

Our team consists of six senior level students at SFSU. As such we think we are uniquely positioned to understand and properly address the needs of the SFSU community. Our experience as students allows us to know what type of features will be best in addressing the needs of our fellow students. Our narrow target market combined with our understanding of it, ensures that our website will be able to compete with other platforms.

List of main data items and entities

1. Unregistered user

    1.1. Can search for Items without logging in

2. Registered user

    2.1. Can buy, sell, request for Items on the site

    2.2. Report other registered ussr to admin for review

3. Admin

    3.1. Must approve user post before being publish

    3.2. Can delete registered user and their post for violating TOS

4. Registration Record

    4.1. Keep the list of registered user

5. Public View

    5.1. Viewable by anyone. i.e. unregistered users and registered users

6. Private View

    6.1. Viewable only to registered users. Not viewable to unregistered users.

7. SFSU Email

    7.1. Any email address which belongs to the sfsu.edu domain

    7.2. Registration on the website is only possible with an SFSU email

8. Message

    8.1. The messaging system built into the website allowing registered users to communicate with each other

    8.2. Discourage users from giving out contact info in favor of the safer messaging system provided by the website

    8.3. Give a notification when you get a message(sound or tab notice)

9. Category

    9.1. Class

        9.1.1. A sub category that we have for item like book or tutoring service, they can be looked up be the class

    9.2. Other

        9.2.1. Any non class related item will go under here

10. Meet Info

    10.1. The meeting location and meeting time regardless of whether the exchange is for an item or a service

10.2. Only visible to buyer and seller

11. Meet Location

11.1. The specified location where the exchange will take place regardless of whether the exchange is for an tem or a service

11.2. Only visible to buyer and seller

12. Meet Time

12.1. The specified time when the exchange will take place regardless of whether the exchange is for an tem or a service

12.2. Only visible to buyer and seller

13. Item

13.1. This is the item for sale

13.2. Has a title, a price, a description, a category, zero or more tags, zero or more photos

13.3. Items are *public view*

14. Title

14.1. The name given to an item and is the main focus for searching. (e.g. green couch)

15. Photo

15.1. Photos of users visible only to registered users

15.2. Photos of items are public

16. Description

16.1. A text description of each Item

16.2. Descriptions are public

17. Tag

17.1. Special words used to categorize the Items. A couch would have the tag "furniture" so that it would be placed among other furniture during a search

17.2. Tags are public

18. Service

18.1. A service that a user could provide to another user as opposed to a physical object

19. Tutoring

19.1. A service in which one user provides extra opportunities for learning to another user

19.2. Have a list of course available for tutor

19.3. Contain the tutor profile information

Functional Requirements

prioritized (1-must have; 2 – desired; 3 – opportunistic as defined in the class)

Priority 1:

   1.Unregister User

- User shall be able to browse all listings

- User shall be able to create an account

- User shall be able to search for listings based on category or using text search.

- User shall be able to view each listing's details

   2.Register User

- User shall inherit all functional requirements of a unregistered user

- User shall be able to look at all listings

- User shall be able to view each listing's details

- User shall be able to message listing's owner

- User shall be able to message with potential buyers

- User shall be able to login

- User shall be able to post tutoring information

   3.Admin

- Admin shall inherit all functional requirements of a registered user

- Admin shall be able to remove postings the admin does not own

- Admin shall be require to approve each listing before it is posted publicly

- Admin shall be able to delete a registered user's account

Priority 2:

   1.Unregister User

- User shall be able to search with a filter for price

   2.Register User

- User shall be able to search for listings based on category

- User shall be able to remove a posting they own

- User shall be able to delete their account

       3.Admin

          -   Admin shall be able to message registered users

Priority 3:

       1.Unregister User

          -   None

       2.Register User

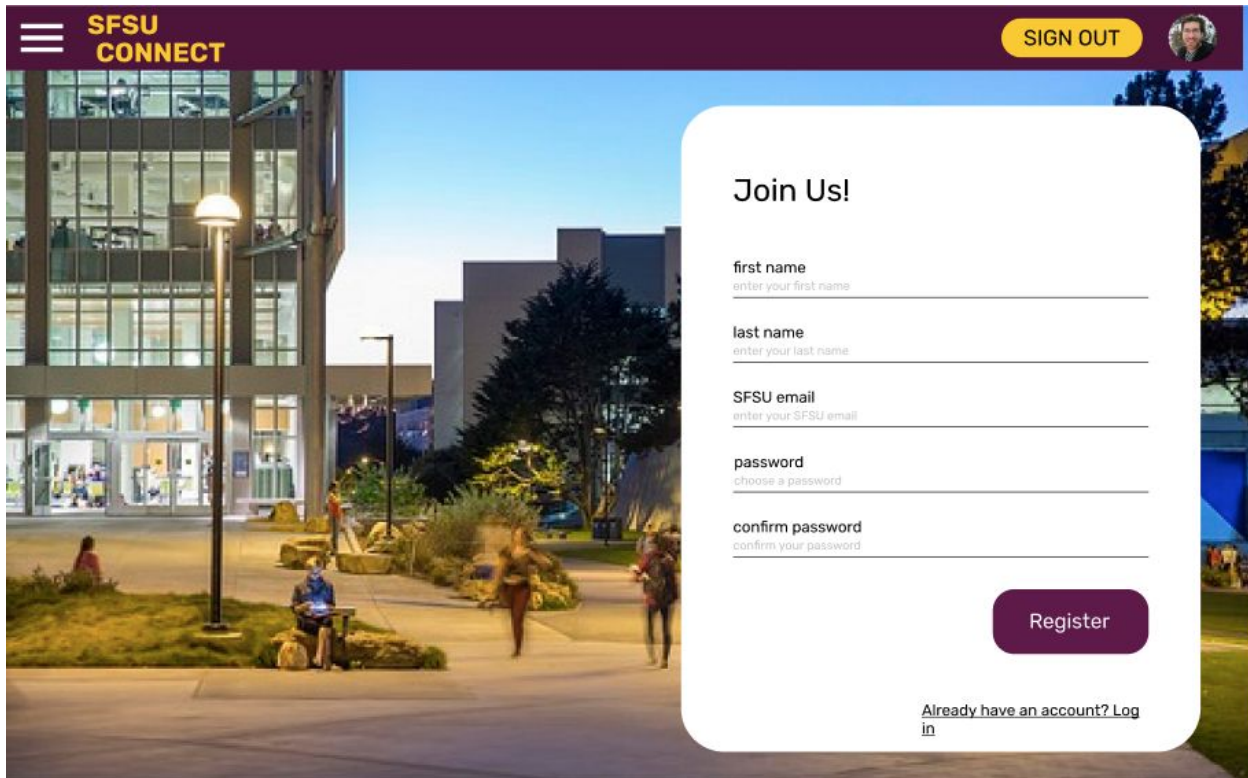   -   User shall be able to create a listing for an item

       3.Admin

          -   Admin shall take request from users

UI Mockups and Storyboards

- Use Case: Number two from Milestone one. Mary attends her first day at SFSU and learns that she needs a certain book for her Archeology class. Mary's professor tells her about the *SFSU Connect* website which she can use to get a textbook for a better price from another student. Mary does just that and browses the website. She uses her class title <u>Intro to Archeology</u> to filter the items listed. Sure enough there are 3 listings for the book she needs. Mary creates an account and contacts all three sellers regarding their books. Mary uses the Register page to create an account on this website.

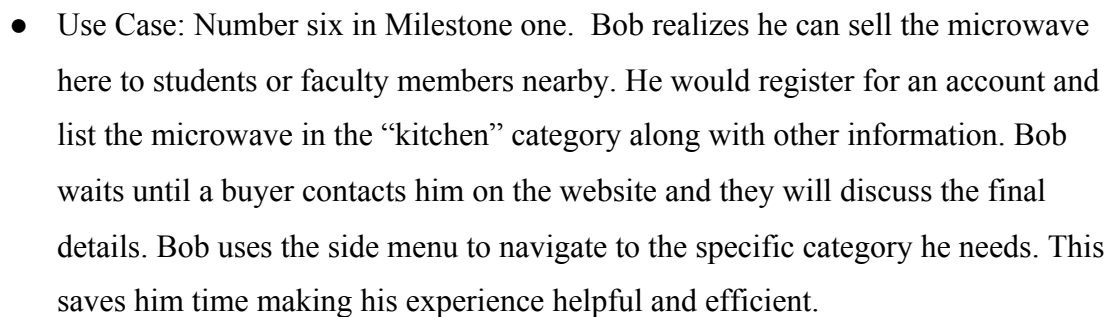## Register Mockup

- Use Case: Number one from Milestone one. Mary makes an account online and posts her book for sale. She includes the class name when she posts the book onto the website. The next day she looks on the website and sees another user has contacted her regarding her book. Mary uses the Login Page to access the website to continue to check if someone is interested in buying her book.

Login Mockup
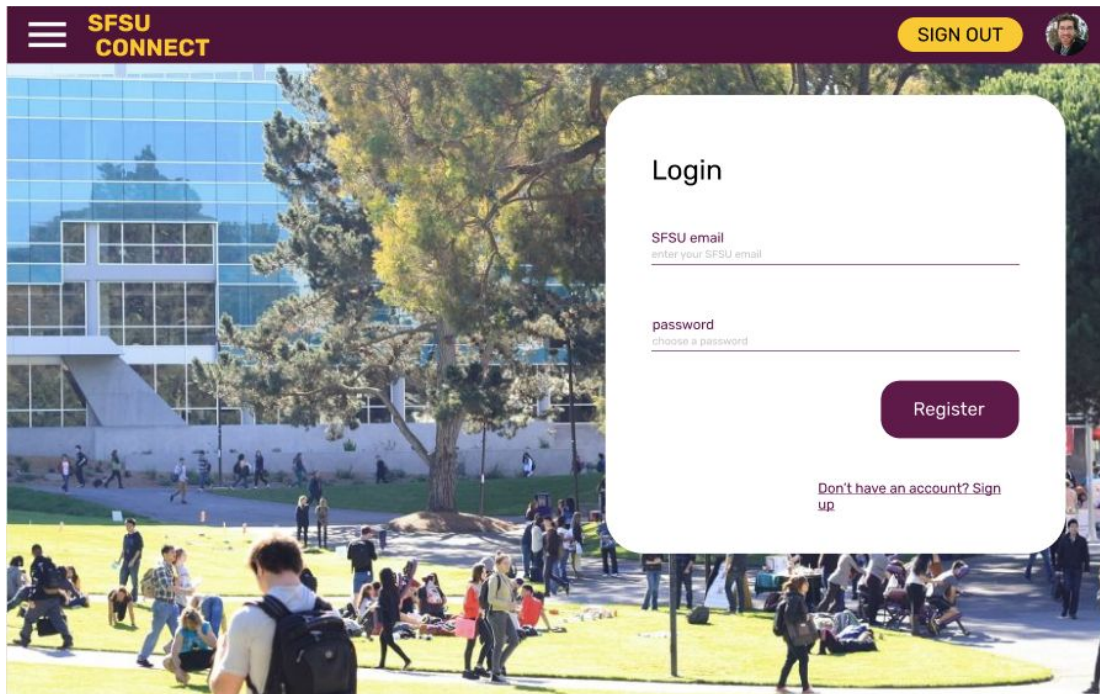
- Use Case: Number three from Milestone one. Mary doesn't know exactly what she needs so she looks to SFSU Connect (our website) for inspiration. Mary doesn't need to be a member to browse and can find what she needs quickly without obligation. When she enters the site she finds that she can search for items by category. Mary can use the navigation bar in order to find her way around the website. Since she is just browsing the navigation bar provides easy organized subjects to go through.

Navigation Bar Mockup

- Use Case: Number six in Milestone one. Bob realizes he can sell the microwave here to students or faculty members nearby. He would register for an account and list the microwave in the "kitchen" category along with other information. Bob waits until a buyer contacts him on the website and they will discuss the final details. Bob uses the side menu to navigate to the specific category he needs. This saves him time making his experience helpful and efficient.

Side Menu Mockup

- Use Case: Number two on Milestone one. Mary is looking for a new textbook. Mary messages with the seller and negotiates a better price. The seller and Mary then agree on a location and time to meet using the website. The website recommends the SFSU Library and both Mary and the seller agree as they will be on campus regardless. On the Admin Page Mary is able to easily navigate to her message and even more vet listings. She can quickly look at new listings for a better price. If she doesn't find the price she wants she then can continue to negotiate in her inbox with other sellers.
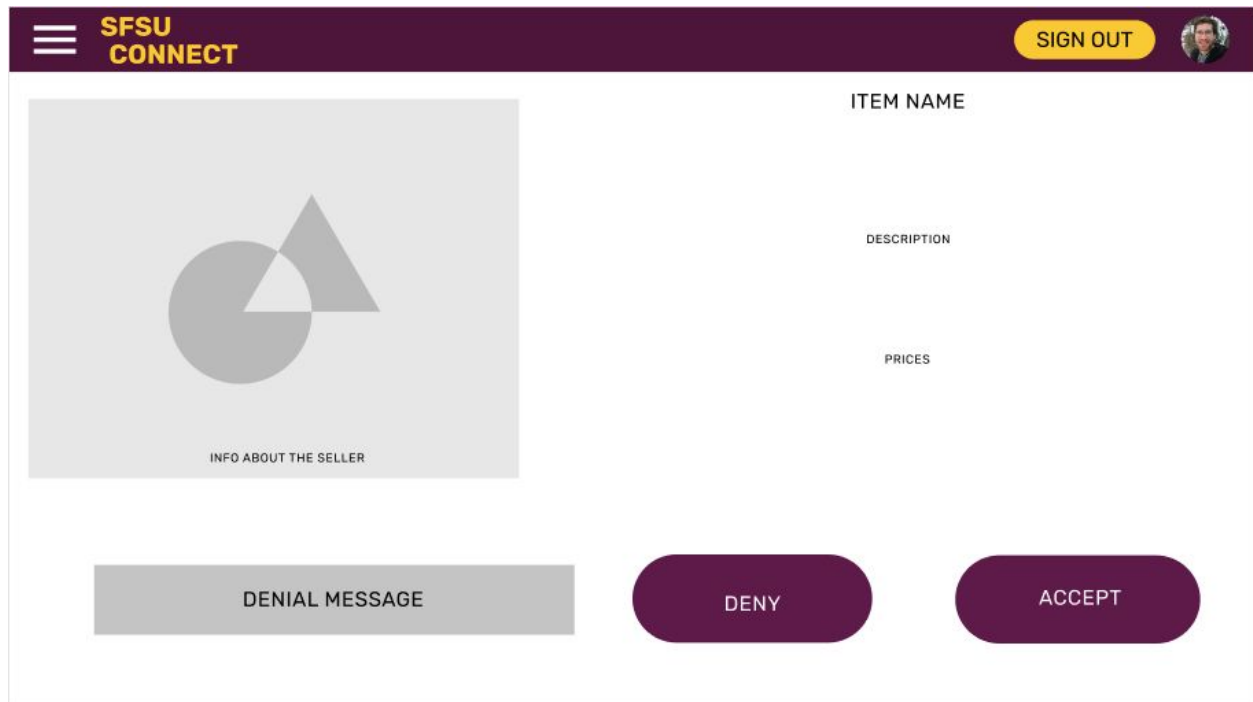
Admin Page Mockup
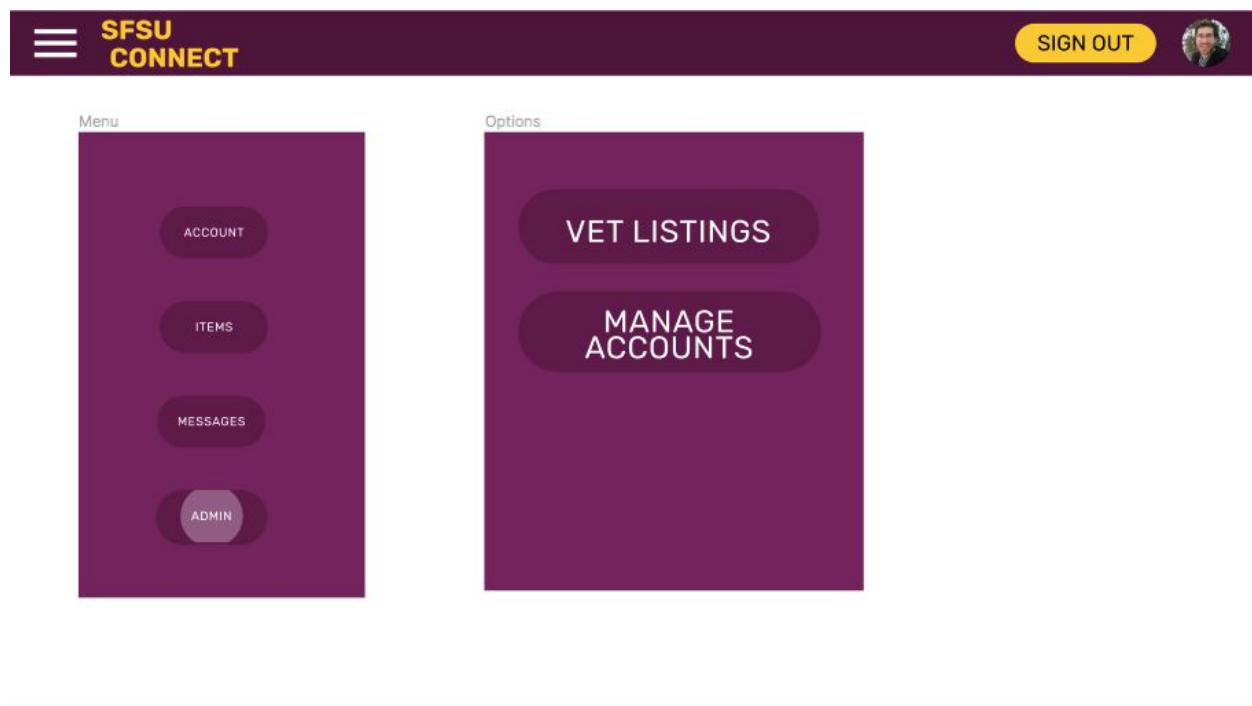
- Use Case: Number three on Milestone one. Mary browses the Landing Page to see any new products posted to the website. When she enters the site she finds that she can search for items by category. She sees for example that there is a "kitchen" category and selects that to see all related listings. She sees a "cheese grater" listed and realizes that it is something she needs. This page makes it easy to find exactly what the student is looking for.

Landing Page Mockup
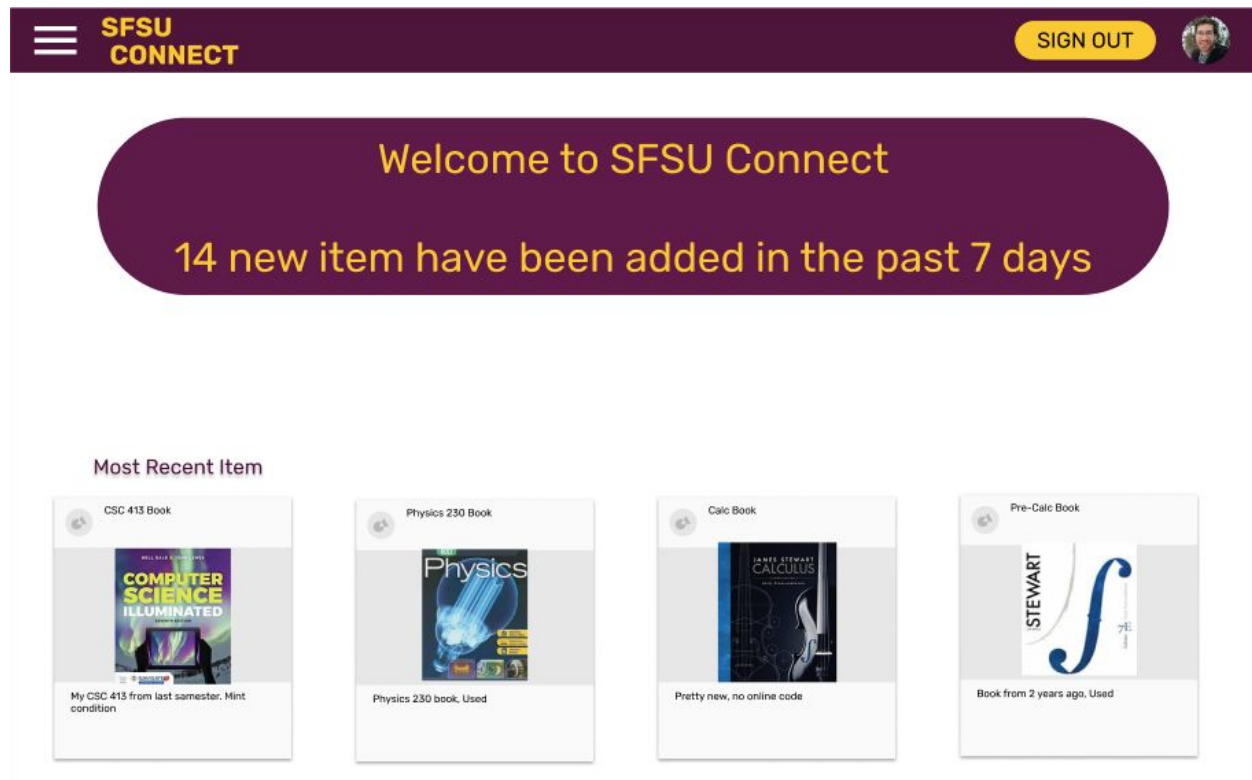
- Use Case: Number one on Milestone one. Mary makes an account online and posts her book for sale. She includes the class name when she posts the book onto the website. The Listing Veting page allows the student to list the item name, description, price and picture. It gives the student the chance to either deny or accept the offer. With all this given information on the page there should be enough for a student to make a quick decision. This helps both the seller and the buyer get what they need in an efficient manner.
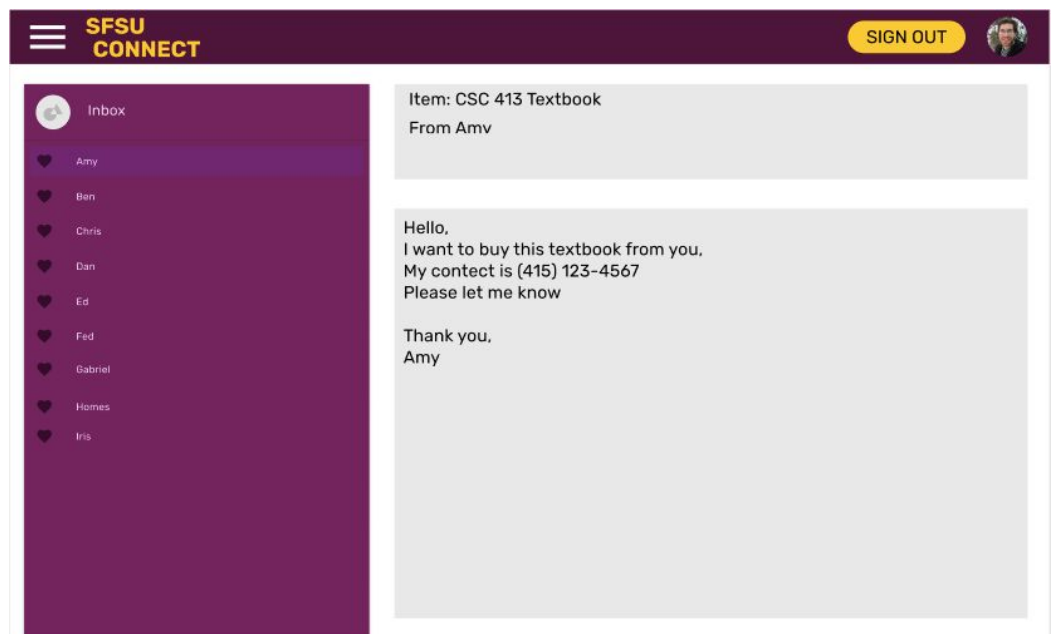
Listing Veting Mockup

- Use Case: Case two on Milestone one. Mary is trying to buy a book on the website.Mary creates an account and contacts all three sellers regarding their books. Within the day, one of the sellers gets back to Mary with a steep price. Mary messages with the seller and negotiates a better price. The sidebar shows Mary all of the names of the people that she is messaging. The inbox is easy to scroll through and navigate the person she is trying to reach.

Inbox Mockup

- Use Case: Case five on Milestone one. Mary is looking for a tutor on the website. A day goes by and Marys already has inquiries sent to her by direct message through the website. She is able to communicate with each student and make sure they are the right fit for each other. The chat is an efficient way for the users to contact without having to meet up first.

Chat Mockup
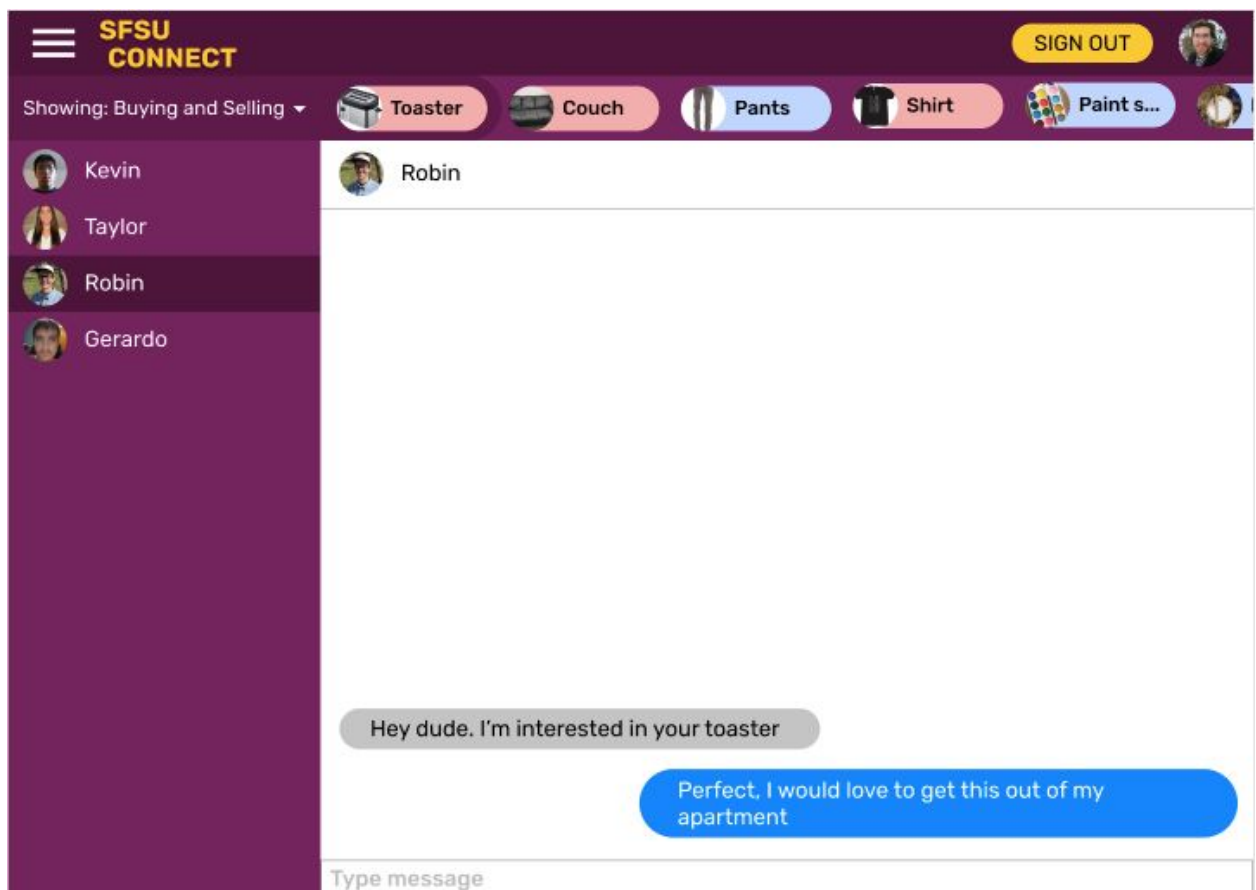
- Use Case: Number two on Milestone one. Mary attends her first day at SFSU and learns that she needs a certain book for her Archeology class. On her account page she has an option to manage items and watch items. On the watched items she can continuously check back on the items she has viewed. This option is good to check if the product has been sold or if the price goes down. This page also allows Mary to update any of her personal information at any time.

Account Page Mockup

High level Architecture and Database Organization

1. Email_Messages
   1.1. idMessages
   1.2. sender

1.3. receiver

1.4. message

1.5. subject

1.6. created

1.7. read/not read

2. Instant_Messages

    2.1. idMessages

    2.2. idChats

    2.3. idUsers

    2.4. message

    2.5. created

3. Chats

    3.1. idChats

4. Users

    4.1. idUsers

    4.2. name

    4.3. email

    4.4. permissions

    4.5. created

5. Permissions

    5.1. idPermissions

    5.2. name

    5.3. description

6. Items

    6.1. idItems

    6.2. idUsers

    6.3. name

    6.4. description

    6.5. price

    6.6. image

    6.7. created

    6.8. approved/not approved

    6.9. adminMessage

    6.10. available/sold

7. Tags
   7.1. name
8. Categories
   8.1. name
9. ItemsTags
   9.1. idItems
   9.2. tagName
10. CategoriesItems
    10.1. categoryName
    10.2. idItems
11. Watchlists
    11.1. idUsers
    11.2. idItems

All Images will be stored in the server's file system. The database will store links to the location of each image in the file system. We will not be storing video or audio.

The search algorithm to find items will be an SQL %LIKE search.

Identify actual key risks for your project at this time

- skills risks (do you have the right skills),
    - Most members have no experience with html framework "pug"
    - Most members have no experience with "MaterializeCSS" framework
    - Some members have little experience with web development
    - Some members have little to no prior experience with javascript
    - Some members have little experience with database management
    - Some members have little experience with SQL
- schedule risks (can you make it given what you committed and the resources),
    - Frontend schedule risk is fairly high because some frontend members have little web development experience. It will be hard to gauge how long it will take each page to be transformed into code. That being said, I think that our frontend mockups are as basic as possible for now and are about as good as we can get it to be simple but still satisfy the functional requirements
    - Backend schedule risk is fairly high because only a few members have worked on an express app before. However, I believe that expressjs is the easiest server framework to use. Therefore, while the risk is high, I do think this is the best option given the circumstances.
- technical risks (any technical unknowns to solve),
    - One technical unknown we have is how we are going to categorize items posted. There is an ongoing discussion on whether the use of traditional categories is better or worse than a tag system. At the current moment, we the team, are thinking that using a root category followed by a tag system.
    - Some members have never saved a client side image to the server side. However, one member does have experience in this and will hopefully be able to handle the issue using skills he has previously learned.
- teamwork risks (any issues related to teamwork);
    - Just need to make sure we meet our due dates and ask for help when we need it.
- legal/content risks (can you obtain content/SW you need legally with proper licensing, copyright).
    - We do not have any foreseeable legal issues at this time.

Project Management

- WireFrames & Mockups for M2 → Gerardo, Hong, Kevin, Taylor
- DB management → Cody, Gerardo
- Test AWS Expressjs connection to AWS MySQL Database → Robert