

# **SW Engineering CSC648/848 Summer 2022**

## **College Essentials**

### **Team 4**

| <b>Student Name</b> | <b>Student Email</b>   | <b>Github Username</b> | <b>Student ID</b> | <b>Student Roles</b>       |
|---------------------|------------------------|------------------------|-------------------|----------------------------|
| Ryan Murphy         | rmurphy4@mail.sfsu.edu | deadlepricon           | 921366560         | Team Lead,<br>Backend Lead |
| Duncan Forniss      | Dforniss@mail.sfsu.edu | DunkeyFunkey           | 917678507         | Frontend Lead              |
| Wee Yang Leong      | wleong@mail.sfsu.edu   | ktleong2000            | 918614936         | Backend Assistant          |
| Darren Wong         | dwong19@mail.sfsu.edu  | dawong025              | 918565679         | Database Master            |
| Justin Wong         | jwong66@mail.sfsu.edu  | jwong66                | 918516578         | Github Master              |

**Milestone:** Milestone 5

**Date:** 08/02/22

## **Version History**

| <b>Milestone</b> | <b>Date</b> |
|------------------|-------------|
| M5V1             | 08/02/2022  |
| M4V2             | 08/02/2022  |
| M4V1             | 07/28/2022  |
| M3V2             | 08/01/2022  |
| M3V1             | 07/19/2022  |
| M2V2             | 07/19/2022  |
| M2V1             | 07/06/2022  |
| M1V2             | 07/01/2022  |
| M1V1             | 06/21/2022  |

## **Table of Contents**

|   |            |
|---|------------|
| <b>[Section I] Product Summary</b>                | <b>5</b>   |
| <b>[Section II] Milestone Documents</b>           | <b>9</b>   |
| <b>[Section III] Screenshots of Final Demo</b>    | <b>210</b> |
| <b>[Section IV] Screenshots of DB Tables</b>      | <b>224</b> |
| <b>[Section V] Screenshots of Task Management</b> | <b>231</b> |
| <b>[Section VI] Team Member Contributions</b>     | <b>232</b> |
| <b>[Section VII] Post analysis</b>                | <b>233</b> |

## [Section I] Product Summary

### Product Name: College Essentials

College Essentials is an online marketplace that is catered towards students, professors or anyone who has products or services to sell related to college.

College students currently have a difficult time trying to buy textbooks at fair and reasonable prices. Students have to either buy their books in new condition at full price, or find it used for close to original price. It is also difficult for students to be able to sell their old textbooks that they no longer need. College Essentials aims to fix these problems by implementing a marketplace catered to users who want to buy new and used textbooks at reasonable prices, and be able to sell their old textbooks for prices that they choose. The three main sections that College Essentials aims to focus on is an online marketplace, forum posts, and service listings.

### What is unique in our product:

To set College Essentials apart from its competition, we aim to implement a user based online marketplace that allows registered users to be able to buy and sell new and used items from people similar to them at reasonable prices. The marketplace will have grown to include not only textbooks, but other college essential items as well. College Essentials will aim to stand out from their competition by allowing users to sell their used products and let them set their own prices. We plan to have a ratings system to have users know that they are buying from a trusted source. Our goal is to make College Essentials an all in one place for College students and professors alike. Forum postings and service listings will help connect these user groups by giving users an area for communication, and allowing them to buy and sell services related to college that they might have. Finally, we

want to be accessible from any university, by allowing users to search the marketplace based off of location.

### **Itemized List of All Major Committed Functions :**

1. General User
  - 1.1. A general user shall be able to browse forum posts.
  - 1.2. A general user shall be able to browse items on the marketplace.
  - 1.3. A general user shall be able to browse service listings.
  - 1.4. A general user shall be able to search for their needed service.
  - 1.5. A general user shall accept our terms of conditions.
  - 1.6. A general user shall be able to search by item condition in the marketplace.
2. Registered User
  - 2.1. A registered user shall be able to log in with their username.
  - 2.2. A registered user shall be able to review another user.
  - 2.3. A registered user shall be able to create their own profile with text.
  - 2.4. A registered user shall be able to create their own profile with pictures.
  - 2.5. A registered user shall be able to edit their own profile with text.
  - 2.6. A registered user shall be able to edit their own profile with pictures.
  - 2.7. A registered user shall be able to post a picture as their profile picture.
  - 2.8. A registered user shall be able to post a service listing.
  - 2.9. A registered user shall be able to list items for sale in the marketplace.
  - 2.10. A registered user shall be able to post on the forums.
  - 2.11. A registered user shall be able to log out.
  - 2.12. A registered user shall be able to rate other general users.

3. Service Listing
  - 3.1. A service listing shall have a title.
  - 3.2. A service listing shall have the location of where the service will be needed.
  - 3.3. A service listing post shall have characters in its content body.
  - 3.4. A service listing post shall be able to have images in its content body.
  - 3.5. A service listing post shall have basic information like time posted.
  - 3.6. A service listing post shall be able to have its content body edited.
  - 3.7. A service listing post shall be able to have its title edited.
  - 3.8. A service listing post shall be able to be unlisted.
  - 3.9. A service listing post shall be able to have filters.
  - 3.10. A service listing post shall have optional tags.
4. Shopping Cart
  - 4.1. A shopping cart shall remember what has been stored within it.
  - 4.2. A shopping cart shall remember how many of one entity have been stored within it.
  - 4.3. A shopping cart shall allow users to edit products within it.
  - 4.4. A shopping cart shall allow for check out.
  - 4.5. A shopping cart shall give descriptions of products stored within it
5. Checkout
  - 5.1. Checkout shall allow a user to input their payment information.
  - 5.2. Checkout shall allow a user to input their shipping address.
  - 5.3. Checkout shall allow a user to input their billing address.

5.4. Checkout shall allow users to review their purchase on last time.

## 6. Navigation Bar

- 6.1. A navigation bar shall have a section for students selling their products.
- 6.2. A navigation bar shall have a section for teachers to look for student assistants.
- 6.3. A navigation bar shall have a section for the student forums.
- 6.4. A navigation bar shall have a section for the about us page.

## 7. Marketplace Listing (Buy/Sell Listing)

- 7.1. A marketplace listing shall have a title.
- 7.2. A marketplace listing shall have a description.
- 7.3. A marketplace listing shall have a photo (or many).
- 7.4. A marketplace listing shall have a price.
- 7.5. A marketplace listing shall be able to be unlisted.
- 7.6. A marketplace listing shall have filters to categorize.

## 8. Ratings

- 8.1. User ratings shall rate a profile on a scale of 1 to 5.
- 8.2. User ratings shall be visible on a registered user's profile.
- 8.3. User ratings shall be visible on a registered user's marketplace listing.
- 8.4. User ratings shall be visible on a registered user's service listing.
- 8.5. User ratings shall be able to be posted by registered users.

**URL to our product:** <http://my-app-dev.us-west-1.elasticbeanstalk.com/>

## [Section II] Milestone Documents

**SW Engineering CSC648/848 Summer 2022**

### **College Essentials**

#### **Team 4**

| <b>Student Name</b> | <b>Student Email</b>   | <b>Github Username</b> | <b>Student ID</b> | <b>Student Roles</b>       |
|---------------------|------------------------|------------------------|-------------------|----------------------------|
| Ryan Murphy         | rmurphy4@mail.sfsu.edu | deadlepricon           | 921366560         | Team Lead,<br>Backend Lead |
| Duncan Forniss      | Dforniss@mail.sfsu.edu | DunkeyFunkey           | 917678507         | Frontend Lead              |
| Wee Yang Leong      | wleong@mail.sfsu.edu   | ktleong2000            | 918614936         | Backend Assistant          |
| Darren Wong         | dwong19@mail.sfsu.edu  | dawong025              | 918565679         | Database Master            |
| Justin Wong         | jwong66@mail.sfsu.edu  | jwong66                | 918516578         | Github Master              |

**Milestone:** Milestone 1

**Date:** 06/21/2022

| <b>Milestone</b> | <b>Date</b> |
|------------------|-------------|
| M1V2             | 07/01/2022  |
| M1V1             | 06/21/2022  |

## **Table of Contents**

|   |           |
|---|-----------|
| <b>[Section I] Executive Summary</b>                                      | <b>3</b>  |
| <b>[Section II] Main Use Cases</b>  | <b>4</b>  |
| <b>[Section III] List of Main Data Items and Entities</b>                 | <b>22</b> |
| <b>[Section IV] Initial List of Functional Requirements</b>               | <b>24</b> |
| <b>[Section V] List of Non-Functional Requirements</b>                    | <b>29</b> |
| <b>[Section VI] Competitive Analysis</b>                                  | <b>34</b> |
| <b>[Section VII] High-level System Architecture and Technologies Used</b> | <b>40</b> |
| <b>[Section VIII] Checklist</b>   | <b>41</b> |
| <b>[Section IX] List of Team Contributions</b>                            | <b>42</b> |

## [Section I] Executive Summary

College is a strange place where students come and can get completely lost because of how different it can be from their previous lives. This new large pond offers many advantages for students who want to further their educational goals. We as software engineers want to make the college experience simpler and possibly cheaper. We are designing a website that has three major uses: selling peer to peer, looking for work opportunities, and enhancing the social aspect of college.

College textbooks can be extremely overpriced, and the worst part is that students will likely only use their books for a semester. When the semester is over they have two options; sell the books back to the school for a much lower price than they initially bought for, or keep the textbook but never use them again. The first aspect of our website College Essentials is to completely change those two options and create a major opportunity for students. Peer to peer selling of textbooks and other items is a key function of our application. This will allow students to bid or buy books from other students who overpaid for textbooks from the school, creating a cycle allowing students to resell their books with little loss.

The next major application of our service is to provide postings for jobs or research opportunities to students. This can allow students to get their names on important research for their field of choice early on in their careers, in order to help get a better job once they graduate. The second advantage is that those who post now have a wider range of applicants and are able to choose the best candidate for their needs.

Finally we have the social aspect of college, where we are giving the users a way to communicate with each other through a forum board. This allows postings for meetings such as study groups for classes and creating new clubs. This will get the word out far better than grass root operations for these groups.

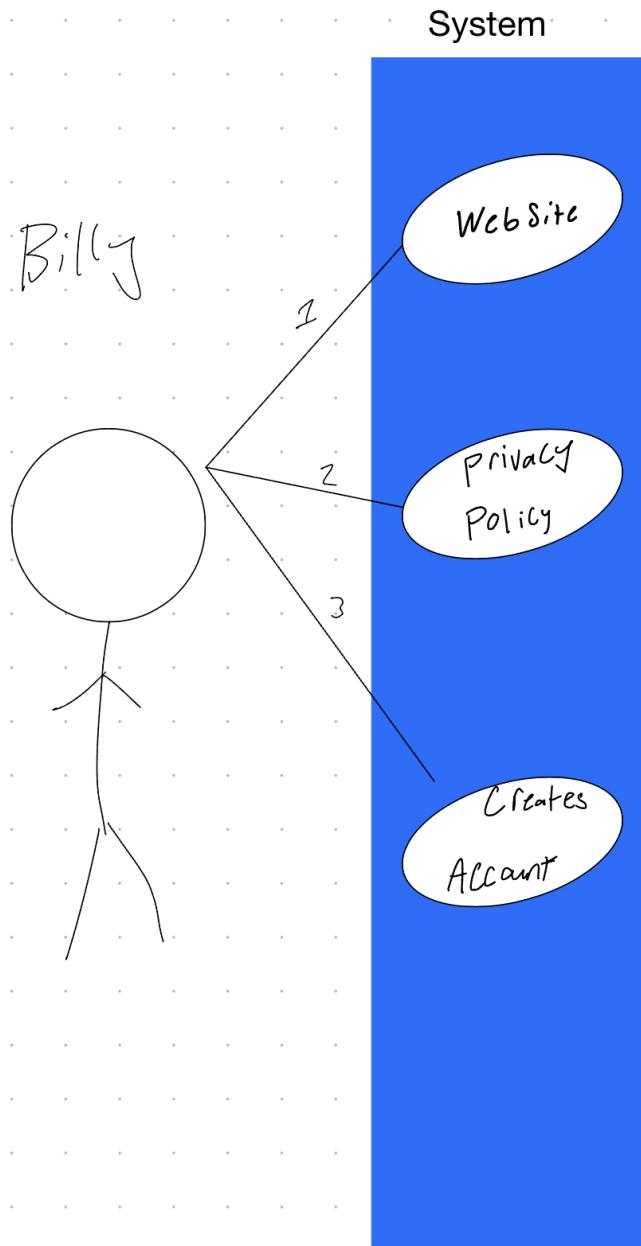
## [Section II] Main Use Cases

**1:[Use Case]** Reading the privacy policy

**[Actor]** Billy

**[Description]** Billy is a freshman in college who just heard about this new forum website from some of his peers in class and wants to sign up. Billy is a very aware and conscious person who always makes sure to read the privacy policies before signing up with a website. Billy makes his way over the College Essentials website and after looking at the general home page and understanding what the website would offer since the home page gives a basic layout and for ear-to-mouth talks with some peers he decides to create a profile. Right before creating a profile, Billy decides to read the Privacy Policies for College Essentials.

College Essentials will be open and honest with the consumers of their products and will tell the users what it uses the data they input into the website for. It shows the users how we protect the data they input into the website to give them ease of mind. The ability for us as College Essentials is to be transparent with the users and all aspects of the products we provide.

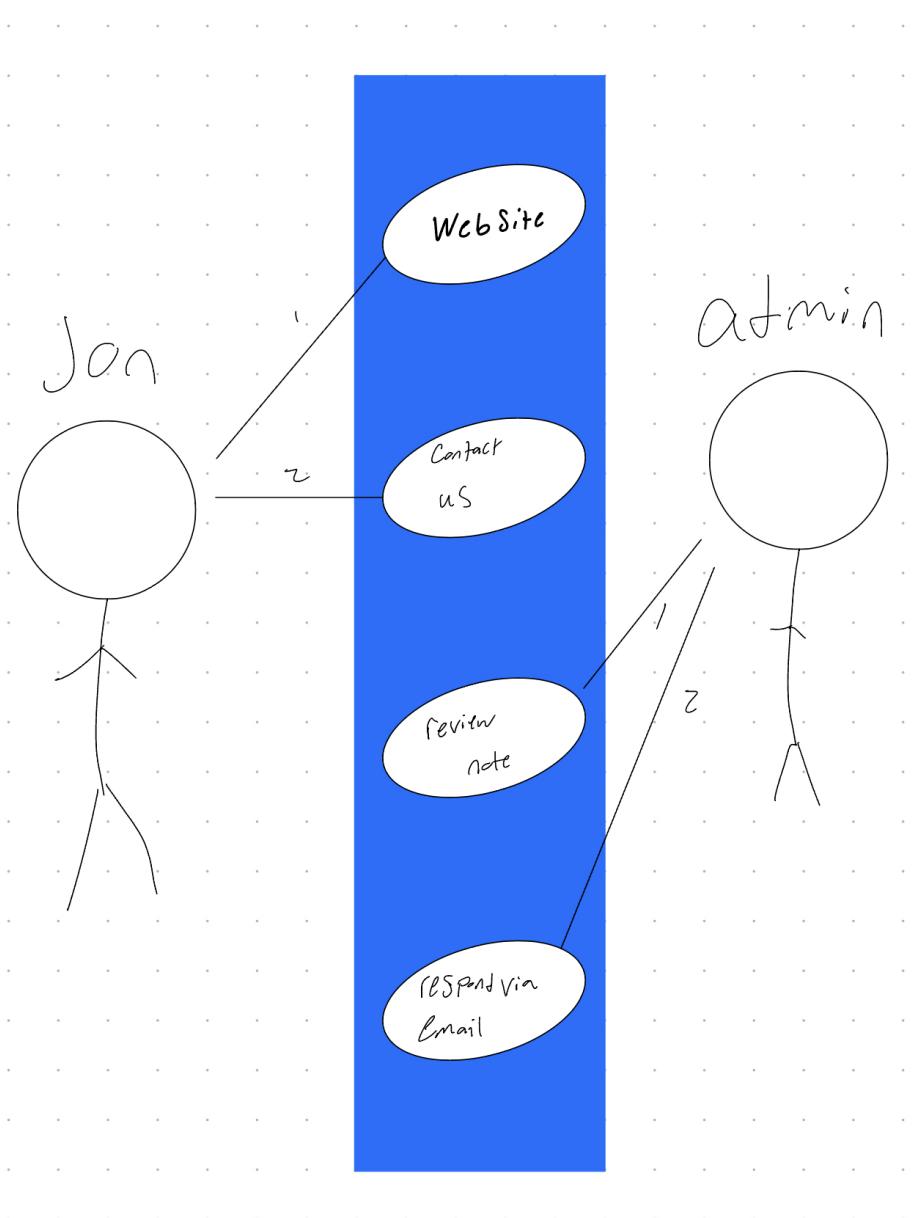


## **2: [Use Case] Using the contact us**

**[Actor]** Jon

**[Description]** Jon is a freshman at SFSU and is also a computer engineering major who is very ahead of the curve and is trying to look everywhere for a coding internship. Jon really loves the use of the forum function on College Essentials from getting study groups together to buying some items off of the marketplace. Jon decides to reach out to the College Essentials personality via the contact us function on the footer of the website. Jon is reaching out to see if they are willing to teach him some pointers and to see if Jon can get an internship. Jon goes to the footer of the website and clicks on the Contact Us link, after that Jon fills out the form.

College Essentials has the option to contact the team at any time for criticism or help. We want to build a community that builds off of itself. Allowing the ability to contact the College essentials team gives the user comfortability while being on the platform.



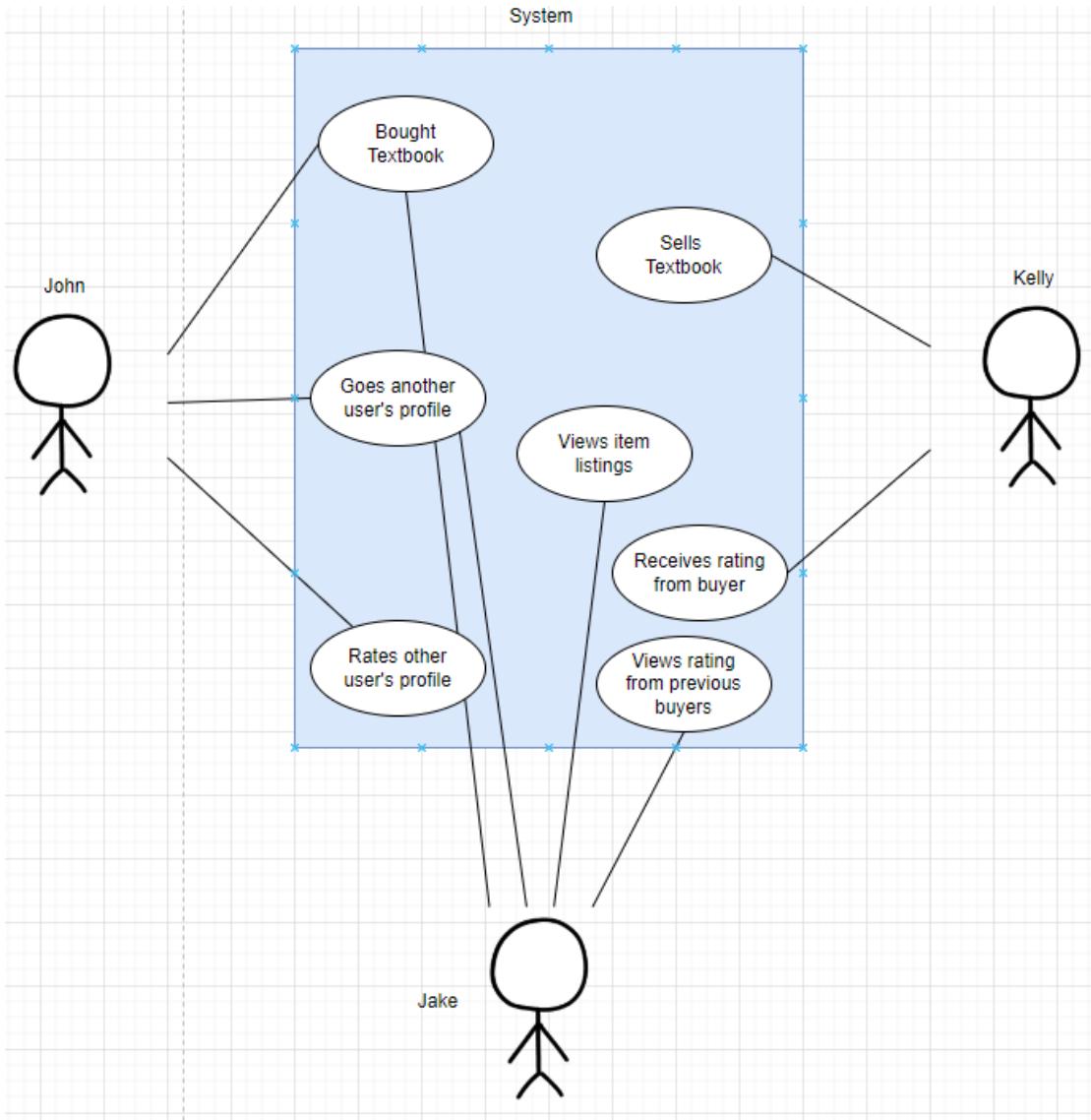
### 3: [Use Case] Reviewing another user

**[Actor]** John, Kelly, Jake

**[Description]** John is a freshman in college. He had been very satisfied with his textbook that he had just bought from another student, Kelly, from College Essentials. The textbook looked new inside, and out. There also didn't seem to be any graffiti or damage found within the book as well. Since

John was so happy with the condition of the textbook he had purchased, he decided to leave a review on Kelly's profile detailing the textbook's condition and how smoothly the transaction went. A few days later, Jake sees another one of Kelly's textbooks up for sale in the marketplace. He is skeptical of Kelly at first, so he goes to Kelly's profile to see if there are any ratings. Much to Jake's delight, he finds that John loved his textbook that he purchased from Kelly a few days ago. This review is enough for Jake to think that Kelly is credible, and he buys that textbook moments later.

Through College Essentials, John can leave a review on the seller's profile to detail their transaction. This is not only beneficial to sellers like Kelly, but to other future buyers from Kelly as well. By leaving a review, future buyers from Kelly know that she conducts smooth transactions and that she is reputable/trustworthy as a seller.



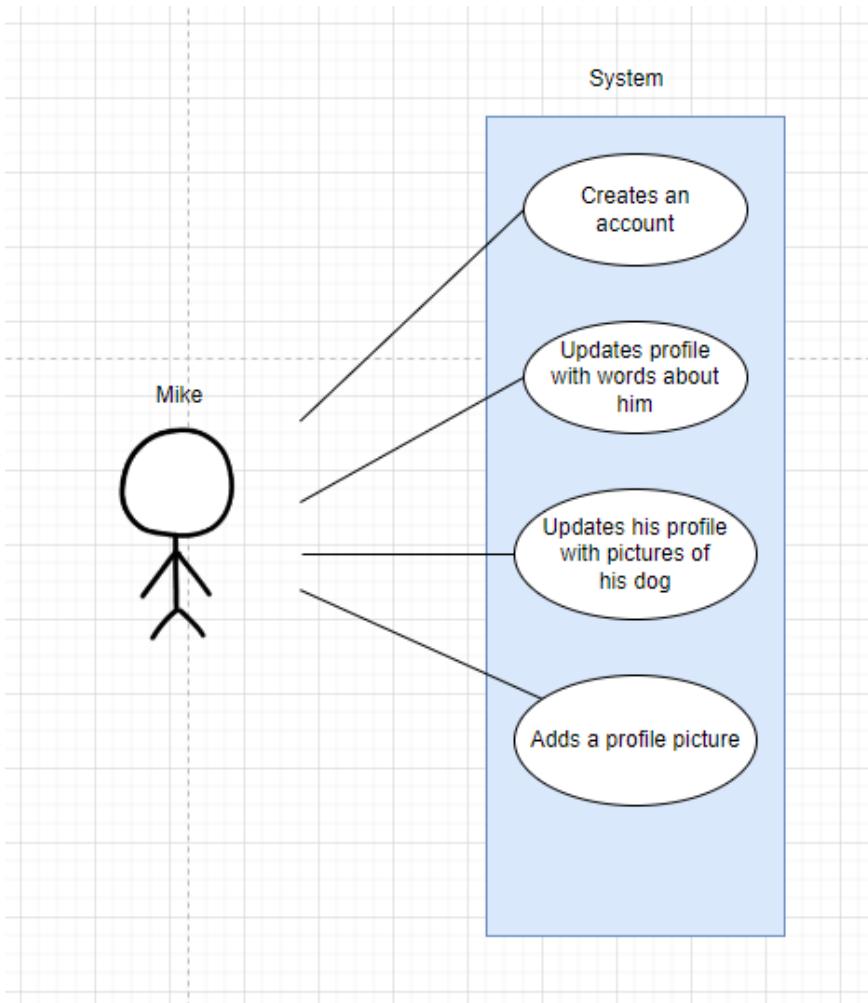
**4: [Use Case]** Users creating an account and updating their profile

**[Actor]** Mike

**[Description]** Mike just ended his first semester of college and has no use for that semester's textbooks. He decided to sell it back to the school to make some of the money back, but the school offered him ten percent of the original cost of the book. Mike then signs up for College Essentials in order to post a listing on the marketplace stating he wants to sell his books. Before posting an advertisement, Mike decides to update his bare profile on College

Essentials in order to establish a greater sense of credibility. The last thing Mike would want is for potential buyers to think he is a scammer. In his profile, he writes some information about himself, including his passions in life as well as his social media accounts. Mike follows it up by adding some pictures of him and his dog to his bio, as well as putting a picture of himself as his profile picture.

Our website would allow everyday users to easily create an account and post their items, whether it be a textbook or a desk they aren't using anymore. Unlike other websites, College Essentials allows everyday users to create and edit their profiles to create a sense of individuality as well as establish some credibility. For potential buyers, getting to know the seller will be a great first step in the door to buying something.



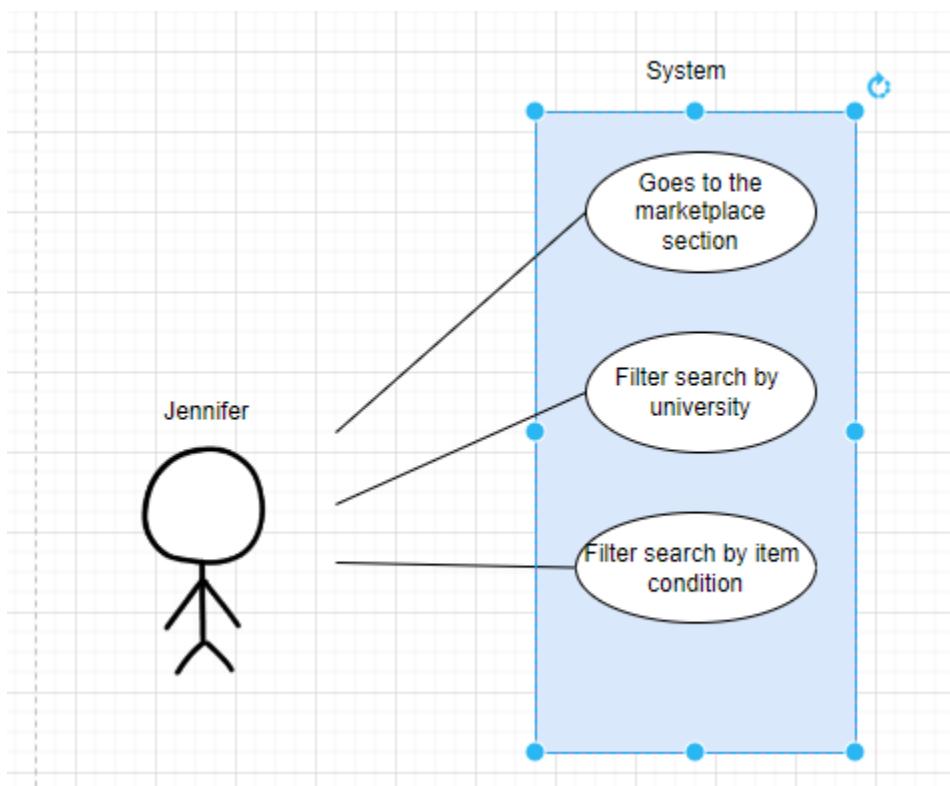
**5: [Use Case]** Searching for listings based on different filters

**[Actor]** Jennifer

**[Description]** Jennifer is a junior in college about to transfer to a four year university. Jennifer needs to buy textbooks for her new classes, so she goes to her community college bookstore to look for the textbooks that she needs. Unfortunately, her community college doesn't offer any courses that use those textbooks. Jennifer doesn't want to buy a new condition textbook for a class that she would be taking once, so she wants to find someone at her new school that has a used copy of the textbook. However, she lives in a different city than her new college, so she wouldn't be able to go to the bookstore to

buy a textbook before the semester starts. Jennifer decides to go to College Essentials's marketplace, where she is able to filter her search based on various universities as well as the condition of the textbook she needs.

Because her local community college doesn't offer the textbooks she needs, our website would allow her to search different colleges in different cities to find someone who is selling those textbooks. If she wants to find the textbook from a closer city, she can search from schools that are closer to her. If she wants to find the textbook from her new school, she would be able to search from that school directly.

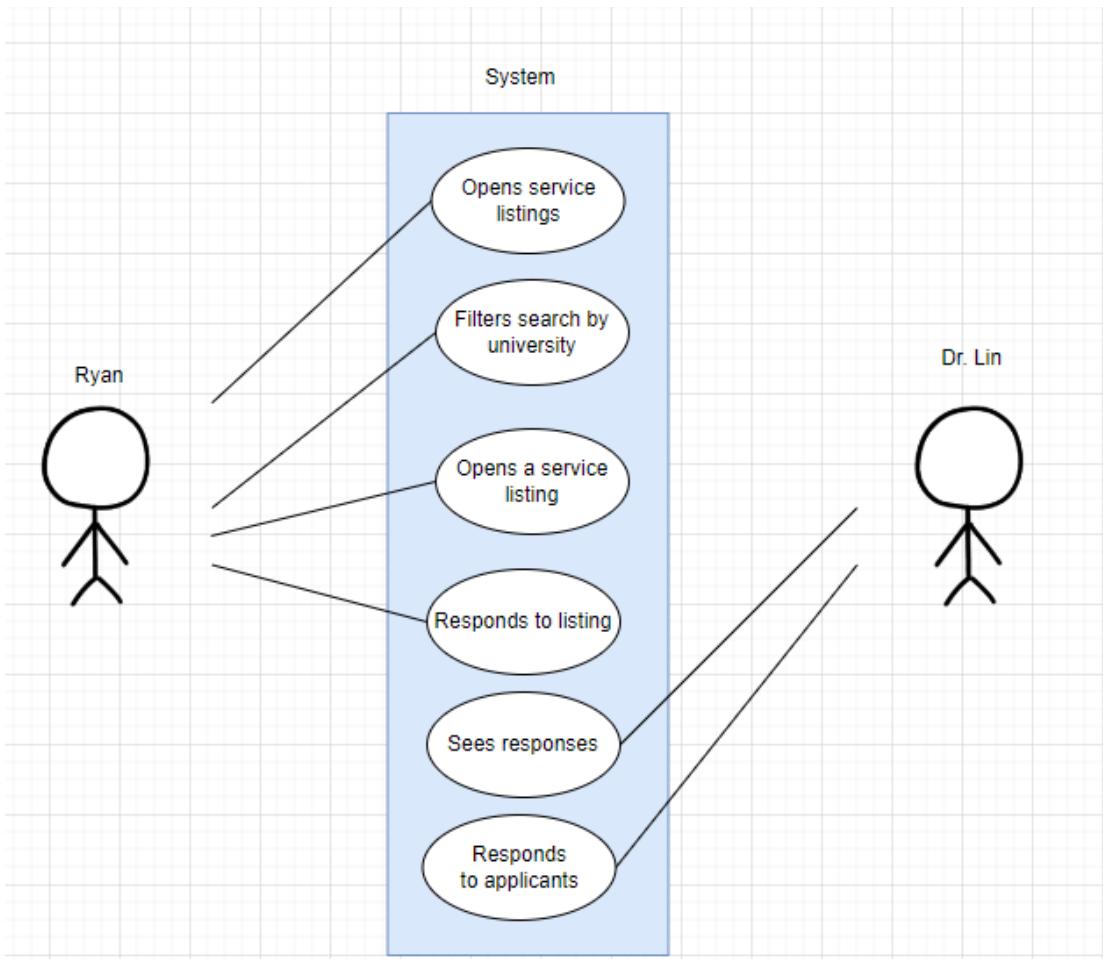


**6: [Use Case]** Users can limit their searches to only people from their own institute

**[Actor]** Ryan, Dr. Lin

**[Description]** Ryan is looking for a possible opportunity to work under a professor at his college. Ryan has a poor experience with other websites, struggling to find opportunities for his university amongst the thousands of results that show for many other universities. However with College Essentials, he notices a menu that can easily return results from a specific institution. He therefore enables the filter and gets new results that are all from his college that he is attending. He was able to find a post from one of his favorite professors and applied. After a few days, Dr. Lin opens his messages and sees the various responses sent to him about his posting.

College Essentials will enable everyday users like Ryan to easily filter among many search results for services that are related to his university. Not only is this beneficial for Ryan, but it is also beneficial for professors of that university to get the help that they need and deserve. In short, College Essentials will make it much easier for everyday users to get the results they need, as well as making it faster for services/jobs to be potentially filled.

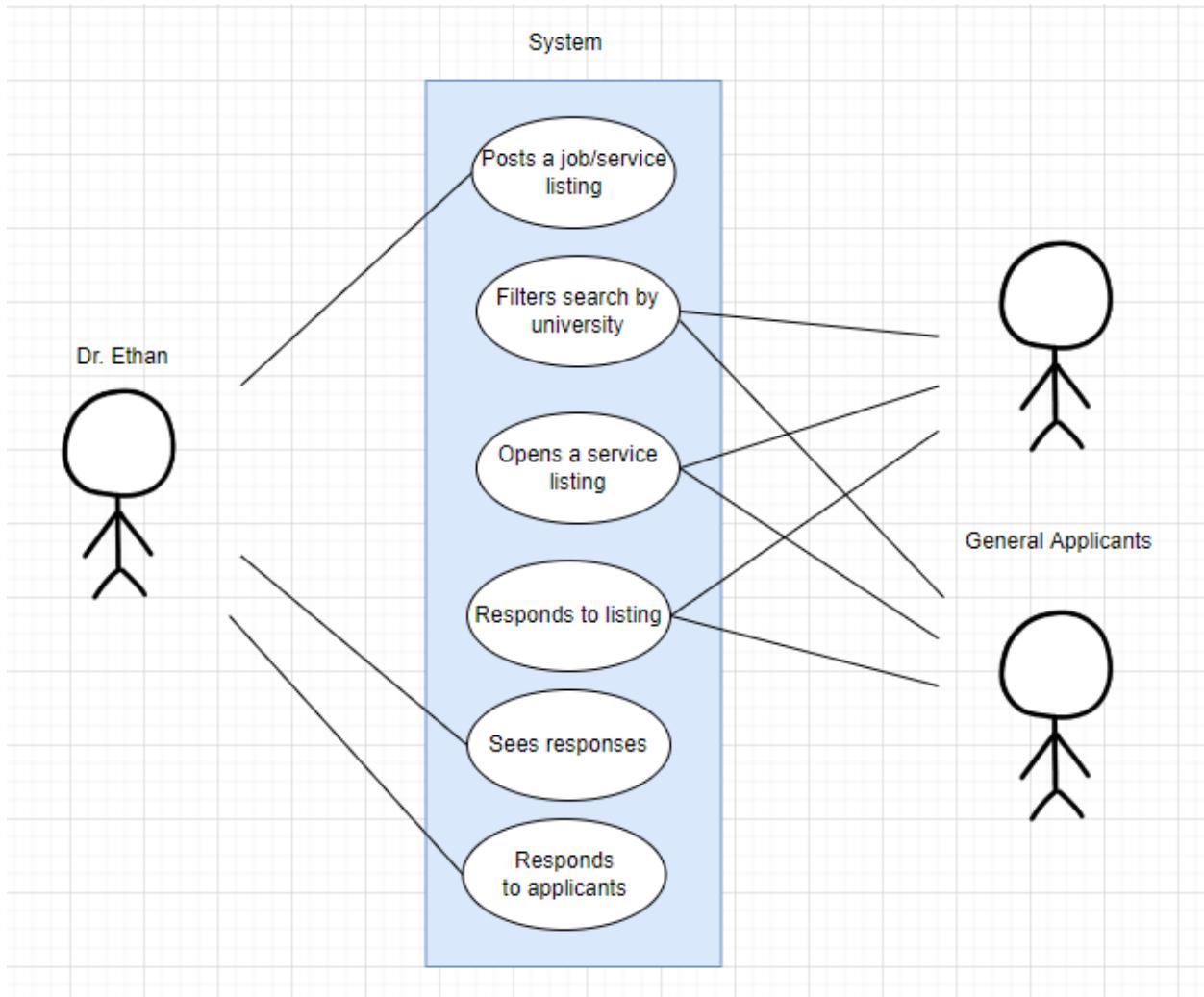


**7:[Use Case]** University staff can use it as a place to look for teacher assistants or research assistants

**[Actor]** Dr. Ethan

**[Description]** Dr. Ethan is a professor at San Francisco State University who teaches biology. He comes up with a brilliant idea and needs a few students to help him bring his idea to life. He decides to go on College Essentials and posts a job offer for students who are willing to help him with his experiments. Students from all over the college reply to his advertisement for seeking help and he is able to choose the best-qualified student to hire.

College Essentials will make it easier for Dr. Ethan to seek help for his upcoming research project. For any professor like Dr. Ethan, finding time out of his day away from holding lectures and grading work is already difficult as is. With our website, we can help users like Dr. Ethan reduce the time spent on seeking volunteers/students, and allow him to focus more of his time and efforts on figuring out how to conduct his research project.

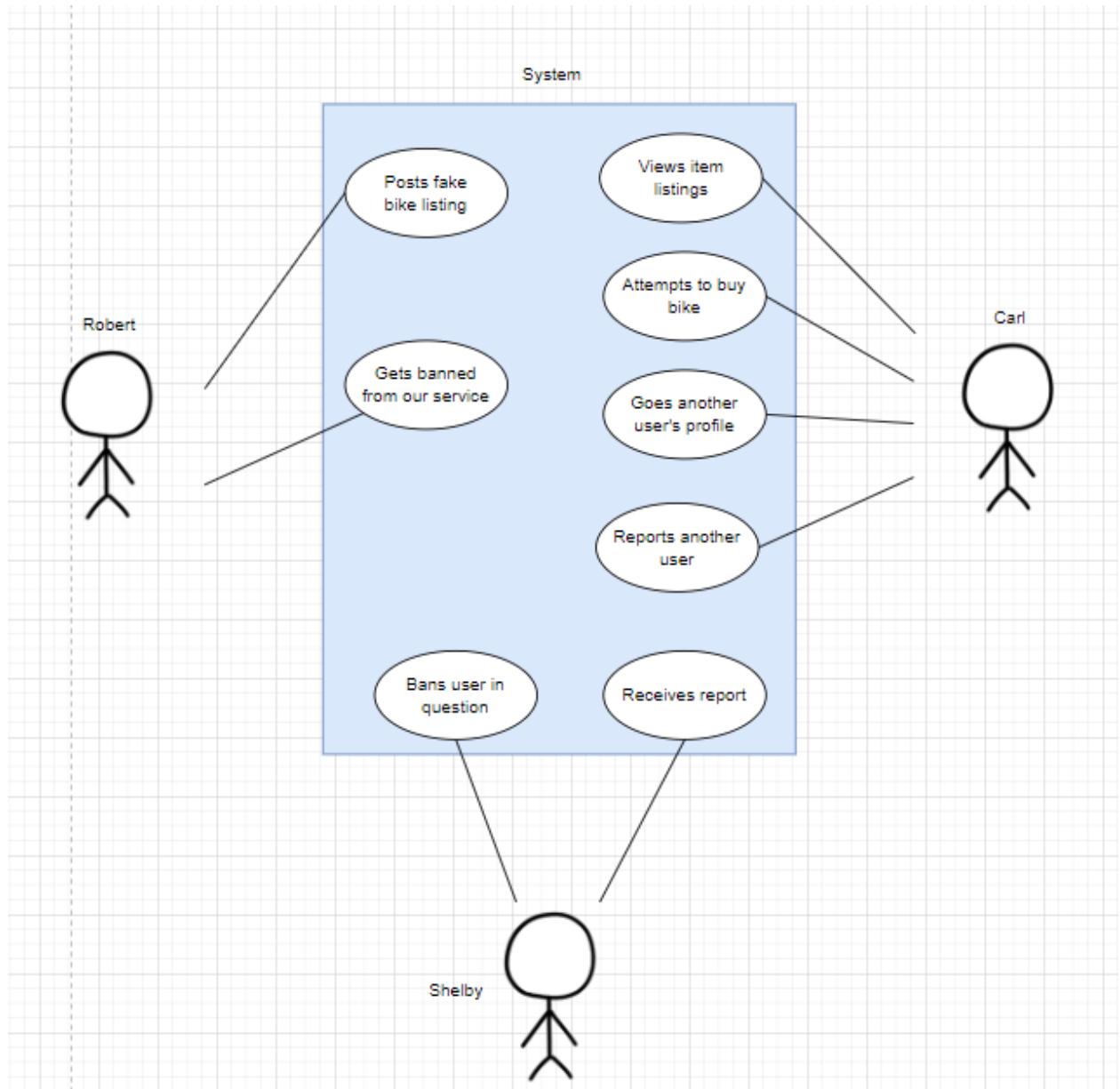


## **8:[Use Case] Reporting a Scammer**

**[Actor]** Robert, Carl, Shelby

**[Description]** Robert is a scammer who is looking for some college students to scam. He posts an advertisement for a brand new bike for much cheaper than the average selling price. Shortly after, Carl decides to meet Robert, only to have his (Carl's) money taken without the bike he (Carl) was promised. Before Robert can scam someone else, Carl decides to report Robert. Shelby, an admin at College Essentials, sees Carl's report, including proof about the scam, and promptly bans Robert before he can commit more scams.

College Essentials will ensure that transactions and services offered are genuine. People like Robert are detrimental in terms of the reputation of College Essentials, and most importantly, the everyday users like Carl. The ability to report other users can ensure that everyday users can have the easiest and best experience possible when buying items, selling items, or exploring various services offered.



## **9:[Use Case] Buying a ad**

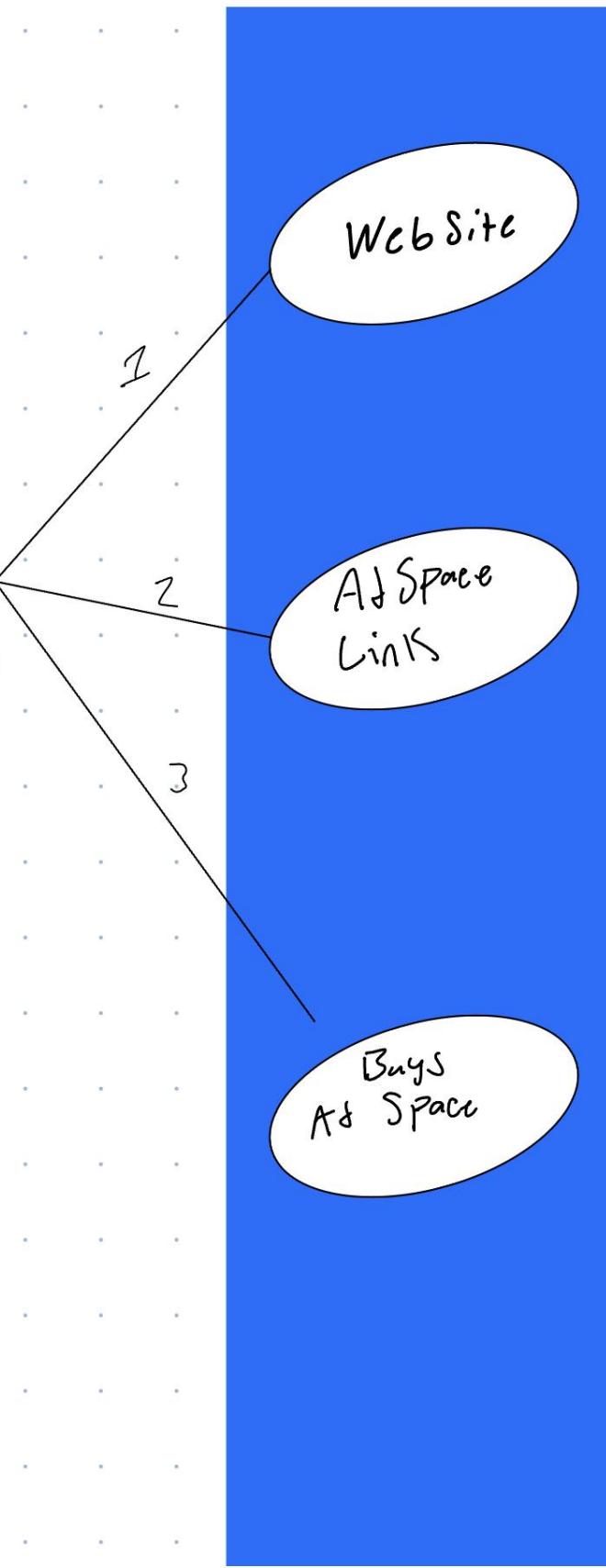
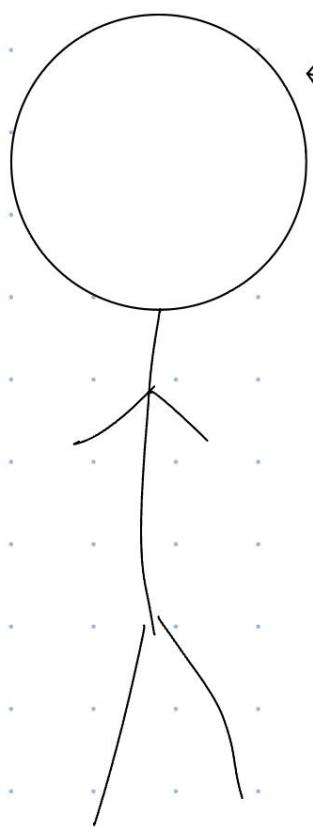
**[Actor]** Bill

**[Description]** Bill is a senior at sfsu. He has been making Youtube videos since he was a freshman. After joining College Essentials he had found out about our service for buying student Ad space. After going over the concept of buying an ad space on a college website. Bill finally went to our website and navigated to the footer of the page and found the link for Ad space. After clicking on the link bill is taken to a new page where the website gives Bill the information and form to fill out of the Ad on the website.

College Essentials can give the users the ability to buy Ads in order to advertise what the student wants within our confines and rules. The platform gives the users a platform to advertise what they want from the ease of accessing the website. When applying for Ad space the user can choose the type of demographic to advertise to.

# System

Bill



## **10:[Use Case] Changing college**

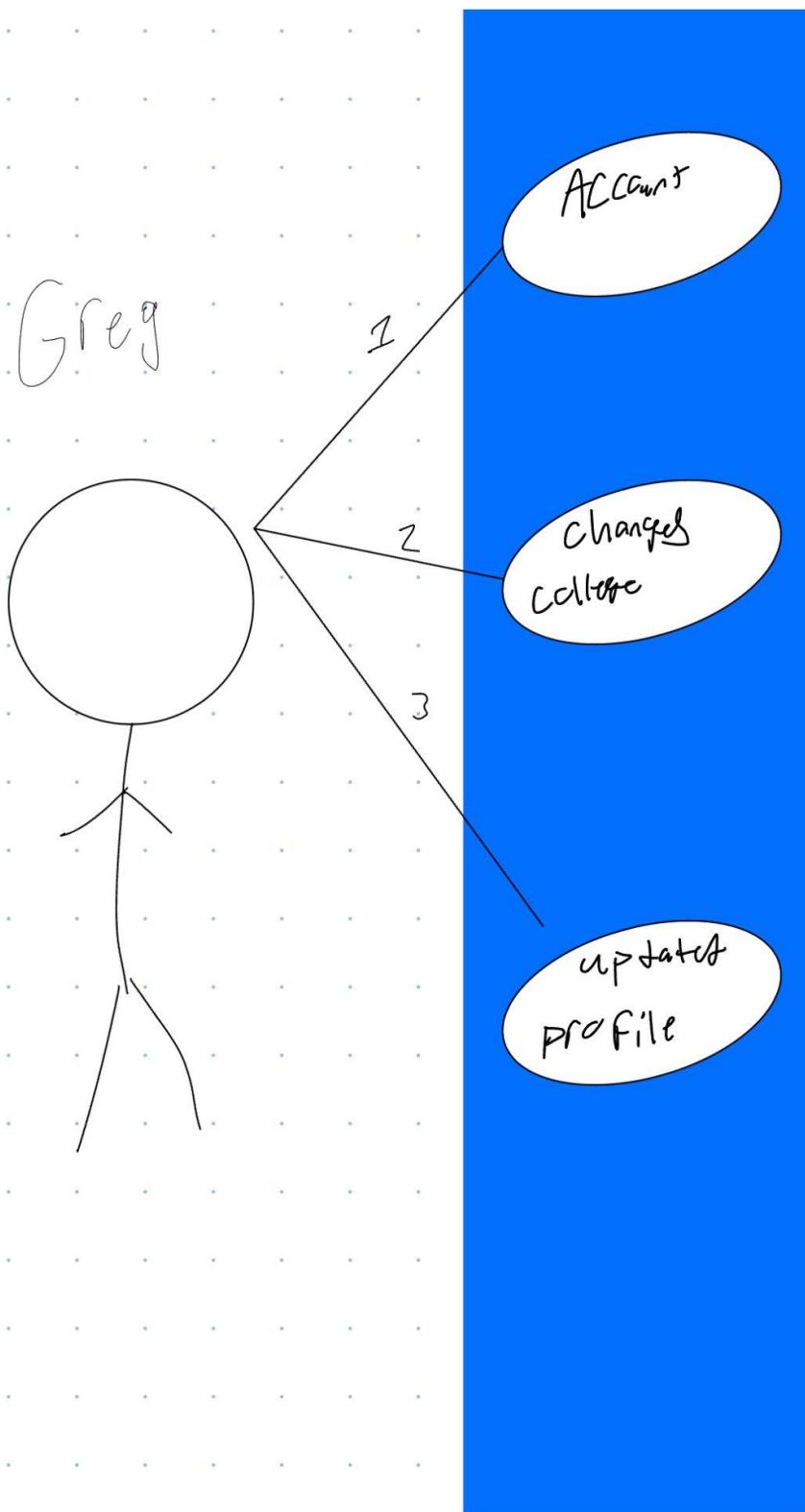
**[Actor]** Greg

**[Description]** Greg is transferring colleges and wants to change his account from SFSU to UCLA. He would log in to the account and after logging in then he would edit his profile and change his college. The ability to change the college on your profile will let other users know what college you are going to. Greg is excited to go to his new college.

College Essentials has the ability to change your college on your profile. College Essentials profiles have the college in their profile so when looking at someone's profile you can tell where someone is.

---

## System



### [Section III] List of Main Data Items and Entities

| Name              | Definition   | Usage  |
|-------------------|--|--|
| General User      | A person who is not signed in or does not have an account with College Essentials.                   | A general user can do basic functions on College Essentials, including scrolling through forums and the marketplace. They are not allowed to do actions like make comments on these posts. |
| Registered User   | A person who has signed up to use College Essentials.  | Registered users will have user information such as: username and school.  |
| Moderator         | A person who moderates content on the website.   | Moderators will be allowed to moderate forum content, such as deleting posts.  |
| Guest             | This is a user who is unregistered or not logged in who is able to view all posts but not reply.     | Guests will be ghosts within the site, they are able to search and look but not interact.  |
| User Registration | This is the section that would allow users to register.  | Users would need to register with a username, password, school, and school email.  |
| Private Message   | Direct messages between two different parties.   | Allows two different people to communicate privately.  |
| Forum Post        | This is where users are able to post miscellaneous forms that don't fit within the other categories. | These forms are not within the categories already such as people to go to games with or even making friends with similar interests   |

|                  |  |   |
|------------------|--|---|
| Marketplace      | A place for users to post what products they are trying to sell.   | Users will be able to post the products that they are trying to sell, which would include their own prices, and how the product would be delivered. |
| Service Listings | A place for users to post if they need or provide services.  | This is the link to the information on the job or item. This also provides the ability to privately message the service provider.                   |
| User Rating      | In order to fight against fraud we have user ratings where other users are able to grade the service provider. | This will be a rating out of 5 and an optional description report.  |
| Payment System   | This will be a system for collecting online payments for College Essentials.                                   | The payment system would use an API to collect listing fees and online transaction payments.  |

## **[Section IV] Initial List of Functional Requirements**

1. General User
  - a. A general user shall be able to register their own unique account(s) with a unique username.
  - b. A general user shall be able to register their own unique account(s) with a password.
  - c. A general user shall be able to browse forum posts.
  - d. A general user shall be able to browse items on the marketplace.
  - e. A general user shall be able to browse service listings.
  - f. A general user shall be able to search for their needed service.
  - g. A general user can search by categories in the marketplace.
  - h. A general user can search by college institution in the service listing section.
  - i. A general user can search by item condition in the marketplace.
  - j. A general user shall accept our terms of conditions.
  - k. A general user shall be able to submit their own ads to be promoted on our website.
2. Registered User
  - a. A registered user shall be able to log in using their username.
  - b. A registered user shall be able to log in using their email.
  - c. A registered user shall be able to review another user.
  - d. A registered user shall be able to create their own profile with text.
  - e. A registered user shall be able to create their own profile with pictures.

- f. A registered user shall be able to edit their own profile with text.
  - g. A registered user shall be able to edit their own profile with pictures.
  - h. A registered user shall be able to post a picture as their profile picture.
  - i. A registered user shall be able to post a service listing.
  - j. A registered user shall be able to list items for sale in the marketplace.
  - k. A registered user shall be able to post on the forums.
  - l. A registered user shall be able to edit a forum post.
  - m. A registered user shall be able to edit their service listing.
  - n. A registered user shall be able to edit their sales listing.
  - o. A registered user shall be able to log out.
  - p. A registered user shall be able to email other registered users.
  - q. A registered user shall be able to pay with different online payment methods.
  - r. A registered user shall be able to opt in to pay in cash in person through email
  - s. A registered user shall be able to report another user.
  - t. A registered user shall be able to block another user.
3. Admin
- a. An admin shall be able to review reports.
  - b. An admin shall be able to ban registered users.
  - c. An admin shall be able to unban registered users.
  - d. An admin shall be able to delete forum posts of other users.
  - e. An admin shall be able to remove service listings from the website.
  - f. An admin shall be able to remove marketplace listings from the website

#### 4. Forum Listing

- a. A Forum listing shall have a title.
- b. A Forum listing post shall have characters in its content body.
- c. A Forum listing post shall have basic information like time posted.
- d. A Forum listing post shall be able to have its content body edited.
- e. A Forum listing post shall be able to have its title edited.
- f. A Forum listing post shall have optional tags that categorize the post.
- g. A Forum listing post shall be able to be unlisted.

#### 5. Service Listing

- a. A service listing shall have a title.
- b. A service listing post shall have characters in its content body.
- c. A service listing post shall have basic information like time posted.
- d. A service listing post shall be able to have its content body edited.
- e. A service listing post shall be able to have its title edited.
- f. A service listing post shall be able to be unlisted.
- g. A service listing post shall be able to have filters based on location, and type of service.
- h. A service listing post shall have optional tags that categorize the post.

#### 6. Receipt

- a. A receipt shall have the product bought.
- b. A receipt shall have the price of the product bought.
- c. A receipt shall have the time of the transaction.
- d. A receipt shall have the email of the user involved.
- e. A receipt shall have the method of contacting the seller via email.

- f. A receipt shall have the method of contacting the seller via phone number.

## 7. Shopping Cart

- a. A shopping cart shall remember what has been stored within.
- b. A shopping cart shall remember how many of one entity have been stored within.
- c. A shopping cart shall allow users to edit products within.
- d. A shopping cart shall allow for check out.
- e. A shopping cart shall give descriptions of products stored within.

## 8. Navigation Bar

- a. A navigation bar shall have a section for users selling their products
- b. A navigation bar shall have a section for teachers look for student assistants
- c. A navigation bar shall have a section for the student forums
- d. A navigation bar shall have a section for the about us page

## 9. User Ratings

- a. A rating shall be rate a profile on a scale of 1 to 5
- b. Ratings shall be visible on a registered user's profile.
- c. Ratings shall be visible on a registered user's marketplace listing.
- d. Ratings shall be visible on a registered user's service listing.
- e. Ratings shall be able to be posted by registered users.

## 10. Marketplace Listing (Buy/Sell Listing)

- a. A marketplace listing shall have a title.
- b. A marketplace listing shall have a description.

- c. A marketplace listing shall have a photo (or many).
- d. A marketplace listing shall have a price.
- e. A marketplace listing shall have no expiration date.
- f. A marketplace listing shall have a set price.
- g. A marketplace listing shall be able to be unlisted.
- h. A marketplace listing shall have filters to categorize.
- i. A marketplace listing shall have optional tags such as condition and type of listing.

## 11. Messages

- a. A message shall have characters in its content body.
- b. A message shall have a timestamp.
- c. A message shall have a status (sent/received/read).

## **[Section V] List of Non-Functional Requirements**

1. Functionality
  - a. The website shall use the tools and frameworks that the group has agreed on and approved by the CTO.
  - b. The website shall use Amazon Web Services for deployment.
  - c. The website shall be user friendly and simple.
  - d. The website shall be easy to navigate for everyone.
  - e. The website shall be available for students and professors at any school.
2. Security
  - a. The user shall need a username/email and password to be able to post and respond to posts on the website.
  - b. The user shall need an account in order to interact with posts.
  - c. User registration shall only be available to those with emails.
  - d. The user shall need a registered account in order to write reviews for other users.
  - e. Admins shall be able to access all aspects of the websites.
  - f. Admins shall be able to moderate the website.
3. Privacy
  - a. The user shall agree to a terms of service agreement in order to access our services.
  - b. Passwords and other personal information shall be kept hidden.
  - c. Users shall be authenticated through username and password.
  - d. Users shall only be able to sign up if they have an email.

- e. User information shall be only accessible to the user.
- f. Payments shall be collected through an API.

#### 4. Performance

- a. The website shall be up at all times.
- b. The website shall make it easy to post listings.
- c. The website shall make it easy to see other users ratings.
- d. The website shall make it easy to delete old listings.
- e. The website shall make it easy to allow the user to go through payment system

#### 5. System Requirements

- a. The website shall work on version 1 to 98 of Mozilla Firefox.
- b. The website shall work on version 1 to 99 of Google Chrome.
- c. The website shall work on version 1 to 99 of Microsoft Edge.
- d. The website shall work on version 1.0 to 15.2 of Safari.
- e. The website shall work on version 2.0 83 of Opera.

#### 6. Marketing

- a. The website shall be viewable for both guests and users.
- b. The website shall have its own logo.
- c. The website shall be easy to navigate for any user.
- d. The website shall be seen as a familiar alternative for other online marketplaces.
- e. The website shall be seen as friendly for students and faculty of any school.

f. The website shall be not directly linked with any specific school.

## 7. Content

- a. The website shall allow users to search posts by categories.
- b. The website shall allow users to search by object name.
- c. The website shall allow users to search by listing name.
- d. The website shall allow users to leave reviews on other users.
- e. The website shall allow users to see the reviews of other users.
- f. The website shall have a section for miscellaneous postings.

## 8. Scalability

- a. The website shall be able to handle a large number of users from any school.
- b. The website shall have enough categories to sort through the growing number of posts.
- c. The website shall be able to accommodate larger amounts of listings.
- d. The website shall be able to accommodate large amounts of user reviews.
- e. The website shall be able to accommodate large amounts of forum posts.

## 9. Capability

- a. The website shall be able to provide data requested by the user.
- b. The website shall allow posts to be uploaded in a timely manner.
- c. The website shall be able to moderate content within a timely manner.
- d. The website shall be able to recover from failures in a timely manner.

- e. The website shall be able to communicate directly and effectively with their users.
- f. The website shall clean up old content within timely manners.

## 10. Look and Feel

- a. The website shall have readable fonts.
- b. The website shall have a simple layout.
- c. The website shall be easy to navigate.
- d. The website shall load quickly.
- e. The website shall differentiate between different types of listings through service listings and marketplace listings.
- f. The website shall differentiate between listings and miscellaneous forum postings.
- g. The website shall visibly show the ratings of other users on their profile and listings.

## 11. Coding Standards

- a. The code shall be organized and understandable.
- b. The code shall have proper working functions.
- c. The code shall use git to be pushed and pulled from.
- d. The code shall use a framework.
- e. The code shall use a uniform style.

## 12. Availability

- a. The website shall always be available for users.
- b. The website shall generate error messages when errors occur.
- c. The website shall refresh when it fails to load.

- d. The website shall allow postings to be available whenever for users.
- e. The website shall be available to view for unregistered users.

### 13. Fault Tolerance

- a. The website shall refresh requests for products if request for purchasing fails.
- b. The website shall re-attempt to access the database for listings if querying fails.
- c. The website shall refresh access requests if users' internet fails.
- d. The website shall notify users and allow user's to retry if payment fails.
- e. The website shall notify users if accessing the database fails.

### 14. Storage

- a. The website shall store usernames in the database.
- b. The website shall store passwords in the database.
- c. The website shall store schools in the database.
- d. The website shall store the user's listings in the database.
- e. The website shall store user's ratings in the database.

## [Section VI] Competitive Analysis

| Competitive Analysis   | Craigslist  | Reddit  | RamNation   | SFSU Bookstore  | Facebook MarketPlace  | College Essentials                                |
|------------------------|---|---|---|---|---|---|
| Reputable              | X   | X   | ✓   | ✓   | X   | ✓   |
| URL                    | <a href="https://sfbay.craigslist.org/">https://sfbay.craigslist.org/</a> | <a href="https://www.reddit.com/">https://www.reddit.com/</a> | <a href="https://coloradostate.sportswar.com/">https://coloradostate.sportswar.com/</a> | <a href="https://www.bkstr.com/sanfranciscostatestore/home">https://www.bkstr.com/sanfranciscostatestore/home</a> | <a href="https://www.facebook.com/marketplace/">https://www.facebook.com/marketplace/</a> |   |
| Target Audience        | Anyone in a particular area.  | Anyone on a particular forum.                                 | Mainly Colorado State University Students.  | San Francisco State University Students.  | Anyone in a particular area.  | Students who are currently enrolled in a college. |
| Forums                 | ✓   | ✓   | ✓   | X   | X   | ✓   |
| Product Listings       | ✓   | ✓   | ✓   | ✓   | ✓   | ✓   |
| Service listings       | ✓   | X   | X   | X   | X   | ✓   |
| User Listings          | ✓   | ✓   | X   | X   | ✓   | ✓   |
| Direct Messaging       | X   | ✓   | X   | X   | ✓   | ✓   |
| User Reviews           | X   | X   | X   | X   | X   | ✓   |
| Geographic Preferences | X   | X   | X   | X   | ✓   | ✓   |

## Features Implemented:

**Not Implemented: -**

**Implemented: +**

**Better Implemented: ++**

| Features                         | Craigslist | Reddit | RamNation | SFSU Bookstore | Facebook MarketPlace | College Essentials |
|----------------------------------|------------|--------|-----------|----------------|----------------------|--------------------|
| <b>Auction</b>                   | -          | -      | -         | -              | -                    | +                  |
| <b>Store Listings</b>            | -          | -      | +         | +              | -                    | +                  |
| <b>User Marketplace Listings</b> | +          | -      | -         | -              | +                    | +                  |
| <b>Service Listings</b>          | -          | -      | -         | -              | -                    | +                  |
| <b>Payment Options</b>           | +          | -      | +         | +              | +                    | +                  |
| <b>Search Bar</b>                | +          | +      | +         | +              | +                    | ++                 |
| <b>User Reviews</b>              | +          | -      | -         | -              | -                    | ++                 |
| <b>Geographic Preferences</b>    | +          | -      | -         | -              | -                    | ++                 |

### 1. [Craigslist] <https://sfbay.craigslist.org/>

- Taking into account the years Craigslist has behind its belt (26 years), it has gained a reputation as a trustworthy source for finding things you might need or like. However, one of its cons is the amount of

services it provides. Craigslist has been a website created by the people for the people with the amount of things you can do with the site it is very easy to get lost and with its outdated UI its pull isn't for the average college student. Craigslist offers services from looking for communities through forums to finding jobs. The reach of Craigslist is that it is truly accessible for everyone but because of how big the platform is, there isn't any regulation for posts. As for how Craigslist makes money, Craigslist uses posting fees as well as charging companies to have Craigslist promote their listings (mainly in jobs forums). Despite the amount of advertising for the website it stands in the top 20% of websites in the U.S with its main user base being 78% based in the U.S. College Essentials shall have a more updated UI, as well as having more regulation for what is posted. Because we will have a user rating system, users can feel safer knowing that they will be able to see which sellers are trustworthy or not.

## 2. [Reddit] <https://www.reddit.com/>

- Reddit has been one of the most common forum websites we see among college students. Reddit has been active for roughly 16 years with this Reddit has built up a cult following with forms from very peculiar niche groups to general communities. When compared to College Essentials, Reddit is a more non conventional competitor. When on Reddit you have to go into it knowing exactly what you are looking for so you can find the correct forum. Reddit themselves doesn't directly offer any ways to purchase things, however many smaller communities have their own systems in place that would allow them to buy, sell, and trade products. These communities do all

their purchases within private messages, so Reddit has no accountability for any of the transactions that involve users. College Essentials will be directly involved with moderating listings and thus will have a more reliable and trustworthy service than Reddit. We shall be able to show how trustworthy a seller would be, as well as allowing users to easily search for the products that they might want.

### 3. [RamNation] <https://coloradostate.sportswar.com/>

- RamNation is the closest concept of a website to our idea for College Essentials. Some of RamNation's most notable functions are its marketplace and forum boards for posting messages. The down side for the website are the main UI and its target audience. The target audience for RamNation is that it is marketed towards Colorado State students. It narrows down its own audience by focusing mainly on those who are fans of Colorado State sports. It is not a very user friendly website based on the front end of the website being very clunky and outdated. The concept of the website seems to work very well within its niche of its community. One of the features that seemed very interesting is that RamNation builds community strength through its ability to submit a 'class note'. This feature is only for alumni and it works as a shout out that gets published on the website as well as the Colorado State Magazine. College Essentials will aim to do a similar thing by building a community. However, we aim to target the entire college student population in general, regardless of what school that people attend. College Essentials will be more focused on the marketplace aspect, without alienating our target audience.

### 4. [SFSU Bookstore] <https://www.bkstr.com/sanfranciscostatestore/home>

- The SFSU bookstore is one of our main competitors we are trying to compare to when adding implementations and add usability to College Essentials. The SFSU bookstore is only a market place used to sell school supplies, SFSU orientated products and other miscellaneous products. There are no forum posts or any way to contact anyone on the SFSU bookstore website. SFSU bookstore is purely an online and in-person store front. College Essentials in comparison shall reimplement the online store aspect by allowing users to buy and sell products directly to other users, while adding more usability in how the marketplace itself functions. Users shall be able to resell their second hand products at prices of their choices, as well as interact with other users through direct messaging and user reviews. We shall also add more usability through variations in what is allowed to be listed as well as having different sections for forum posts for users to interact with one another.

## 5. [Facebook Market] <https://www.facebook.com/marketplace/>

- Facebook Marketplace functions similarly to Craigslist. It is centered around user interactions, with very little third party interactions, and therefore more likely that there is no liability for any scams that occur. This portion of the social media platform is more often than not considered as afterthought. Listed items are barely curated in any shape or form, so there can be fake listings which could lead to possible scams. Some of the features that it provides are, a location of the seller, details on the seller, price of the item, the condition of the item, and when the seller joined Facebook. However, the user details seems to be lacking since it provides little to no detail on the user

besides the info on their profile page; ie. when they joined facebook and where they live. There is a lack of a reviewer functionality such as the ability to validate the authenticity of the seller. The only way to check a review on someone is to do it through the facebook application on your phone, which is not available on pc. Some features are clearly missing from Facebook Marketplace for unknown reasons such as verified sellers and scam protection. It is more often than not a waste of time, and people are cautioned to use at your own risk. Comparatively we want to strive for a smaller audience with a more trustworthy system. A smaller audience curates a more professional environment, especially since it can be run by either students or teachers. We also want to minimize the amount of risk that users take so that they feel more comfortable returning to College Essentials again.

## [Section VII] High-level System Architecture and Technologies Used

|                             |  |
|-----------------------------|--|
| <b>Server Host:</b>         | Corretto 11 running on 64bit Amazon Linux 2/3.2.15 |
| <b>Operating System:</b>    | Windows  |
| <b>Server Database:</b>     | MySQL  |
| <b>Web Server:</b>          | NGINX 1.12.2                                       |
| <b>Server-Side Language</b> | Java   |
| <b>Web Framework:</b>       | Vaadin   |
| <b>IDE:</b>                 | Visual Studio Code                                 |
| <b>Web Analytics:</b>       | Google Analytics                                   |
| <b>SSL Cert:</b>            | AWS Certificate Manager                            |
| <b>SASS:</b>                | 3.5.5  |

## [Section VIII] Checklist

|           | <b>Task</b>  | <b>Status</b> | <b>Additional Comments</b>  |
|-----------|--|---------------|---|
| <b>1.</b> | Team found a time slot to meet outside of the class  | DONE ✓        |   |
| <b>2.</b> | Github master chosen   | DONE ✓        | Our team had some hiccups after our initial Github master left, however new Github master has already been appointed              |
| <b>3.</b> | Team decided and agreed together on the using the listed SW tools and deployment server  | DONE ✓        |   |
| <b>4.</b> | Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing | DONE ✓        | Deemed on track since the project will require consistent learning about back and front end frameworks for consistent development |
| <b>5.</b> | Team lead ensured that all team members read the final M1 and agree/understand it before submission                                    | DONE ✓        |   |
| <b>6.</b> | Github is organized as discussed in class (e.g. master branch, development branch, folder for milestone documents, etc.)               | DONE ✓        |   |

## [Section IX] List of Team Contributions

| Name           | Contributions  | Complaints |
|----------------|--|------------|
| Ryan Murphy    | Worked on Section: 1, 2, 3, 4, 5, 6, 7, 8 and 9. Helped set up the framework/website and about us page. Also helped with AWS set up. |            |
| Duncan Forniss | Worked on Section: 1,2,3,4,5,6,7,8.  |            |
| Darren Wong    | Worked on section: 2, 3, 4, 5, 6, 7 and 8. Worked on Framework/website and database.   |            |
| Brendan Leong  | Worked on section: 2,3,4,5,6,7,8.<br>Helped with framework/website setup.  |            |
| Justin Wong    | Worked on section: 1, 2,3,4,5,6,7,8. Helped with AWS set up, framework/website and database.   |            |

# **SW Engineering CSC648/848 Summer 2022**

## **College Essentials**

### **Team 4**

| <b>Student Name</b> | <b>Student Email</b>   | <b>Github Username</b> | <b>Student ID</b> | <b>Student Roles</b>       |
|---------------------|------------------------|------------------------|-------------------|----------------------------|
| Ryan Murphy         | rmurphy4@mail.sfsu.edu | deadlepricon           | 921366560         | Team Lead,<br>Backend Lead |
| Duncan Forniss      | Dforniss@mail.sfsu.edu | DunkeyFunkey           | 917678507         | Frontend Lead              |
| Wee Yang Leong      | wleong@mail.sfsu.edu   | ktleong2000            | 918614936         | Backend Assistant          |
| Darren Wong         | dwong19@mail.sfsu.edu  | dawong025              | 918565679         | Database Master            |
| Justin Wong         | jwong66@mail.sfsu.edu  | jwong66                | 918516578         | Github Master              |

**Milestone:** Milestone 2

**Date:** 07/06/2022

## **Version History**

| <b>Milestone</b> | <b>Date</b> |
|------------------|-------------|
| M2V2             | 07/19/2022  |
| M2V1             | 07/06/2022  |
| M1V2             | 07/01/2022  |
| M1V1             | 06/21/2022  |

## Table of Contents

|  |    |
|--|----|
| [Section I] Data Definitions   | 4  |
| [Section II] Prioritized Functional Requirements                       | 10 |
| [Section III] UI Mockups and Storyboards (High Level Only)             | 17 |
| [Section IV] High Level Database Architecture Organization             | 27 |
| [Section V] High Level APIs and Main Algorithms                        | 41 |
| [Section VI] High Level UML Diagrams                                   | 44 |
| [Section VII] High Level Application Network and Deployment Diagrams   | 45 |
| [Section VIII] Identify Actual Key Risks For Your Project At This Time |    |
| 47   |    |
| [Section IX] Project Management  | 50 |
| [Section X] Detailed List of Contributions                             | 51 |

## [Section I] Data Definitions

1. **General User:** A user who is not signed in or does not have an account with College Essentials. A general user has the minimum permissions on the website and can **register** for more permissions.
  - 1.1. **Registration:** A general user shall be able to create an account to get more permissions.
  - 1.2. **Account Contains:**
    - 1.2.1. **Username:** Needs a username to create an account
    - 1.2.2. **Password:** Needs a password that is encrypted to create an account.
    - 1.2.3. **Email:** Needs a unique edu email to create an account.
    - 1.2.4. **School:** Can choose to register a school to the account.
    - 1.2.5. **Accept Terms of Service:** Must accept the terms of service to create an account.
2. **Registered User:** A user who has signed up to use College Essentials. This user gains the extra permissions needed to access the other features on the website. A general user shall be able to **login** to gain access to all these features.
  - 2.1. Can login to the system.
    - 2.1.1. Username: Needs a unique username to login.
    - 2.1.2. Password: Needs a unique password to login.
  - 2.2. Has the ability to post Marketplace Posts.
    - 2.2.1. **Buyers:** All registered users are buyers by default. They can see and interact with Marketplace posts.

- 2.2.2. **Sellers:** All registered users can put up their own Marketplace posts to become a seller. Sellers will be differentiated on the Marketplace post that they have posted.
- 2.3. Has the ability to post forum Posts.
- 2.4. **Profile:** Contains username, school, ratings, private messaging section and marketplace posts.
  - 2.4.1. **Username:** Shows the unique username of the user.
  - 2.4.2. **School:** Shows the listed schools linked with the user.
  - 2.4.3. **Ratings:** Shows the ratings on the user posted by other users.
  - 2.4.4. **Private Message:** Has an area that allows users to private message other users.
  - 2.4.5. **Marketplace Posts:** Has a list of marketplace posts made by the user.
  - 2.4.6. **Service Listings:** Has a list of service listings made by the user.
- 3. **Admin:** A user who moderates content on the website. A moderator still has the same permissions that registered users have, but have additional **admin tools** on our website.
  - 3.1. **Admin Tools:** Contains the ability to delete posts, and temporarily restrict users.
    - 3.1.1. **Delete Posts:** Can choose to delete posts by other users.
    - 3.1.2. **Restrict Users:** Can restrict user accounts who break the terms of service.
- 4. **Guest:** This is a user who is unregistered or not logged in who is able to view all posts but not reply. Guests will be ghosts within the site, they are able to search and look but not interact.

- 4.1. **Username:** Will show as a guest with numbers ie. Guest###
  - 4.2. **Browsing:** A guest will be able to browse listings, and add them to cart but not complete the transaction
  - 4.3. **Ads:** A guest shall be able to contact, and pay us to have their ads on the site.
5. **Login / User Registration:** This is the section that would allow users to login or register accounts with College Essentials.
  - 5.1. **Login Section:** Users have the ability to login to their accounts here.
    - 5.1.1. **Username:** Users will have to enter a unique username to login.
    - 5.1.2. **Password:** Users will have to enter a unique password to login.
  - 5.2. **Registration Section:** Users have the ability to register new accounts here.
    - 5.2.1. **Username:** Needs a username to create an account
    - 5.2.2. **Password:** Needs a password that is encrypted to create an account.
    - 5.2.3. **Email:** Needs a unique edu email to create an account.
    - 5.2.4. **School:** Can choose to register a school to the account.
    - 5.2.5. **Accept Terms of Service:** Must accept the terms of service to create an account.
6. **Private Message:** This is the section that allows users to private message each other through their profiles.
  - 6.1. **Contacting a Person:** Will allow the contacting of a user, client, seller, guest, etc.
  - 6.2. **Timestamp:** Shows when a message was sent.

6.3. **Read Status:** Will show if a message was seen

7. **Forum Post Section:** This is where users are able to post miscellaneous forum posts that don't fit within the other categories. Users must be registered users to make a forum post.

7.1. **Forum Posts:** Contains a post body and post replies.

7.1.1. **Post Body:** A registered user will be able to post a forum post that contains username, post body, and timestamps.

7.1.2. **Post Replies:** A registered user will be able to reply underneath the forum posts that contains a body, the replier's username, and a timestamp.

8. **Marketplace Listings:** A place for registered users to post if they want to buy or sell marketplace listings.

8.1. **Marketplace Listing:** Contains title, location, photo, price, description, expiration date, bidding(If allowed by seller), and tags.

8.1.1. **Title:** Gives a small description of the item at hand.

8.1.2. **Location:** Gives the location of where the deal is to be held.

8.1.3. **Photo:** If provided there will be a picture of the product.

8.1.4. **Price:** Displays the price of the item being sold.

8.1.5. **Description:** A body section that describes the listing.

8.1.6. **Expiration Date:** Displays how long the listing is valid.

8.1.7. **Bidding:** Shows the current minimum bid.

8.1.8. **Tags:** Key words appended to a listing to categorize for searching.

9. **Service Listings:** A place for registered users to post if they need or provide

services.

- 9.1. **Service Listings:** Contains, a Title, Location, Photo, Price, Description, an expiration date, and tags.
  - 9.1.1. **Title:** Gives a small description of the item at hand.
  - 9.1.2. **Location:** Gives the location of where the deal is to be held.
  - 9.1.3. **Photo:** If provided there will be a picture of the product.
  - 9.1.4. **Price:** Displays the price of the item being sold.
  - 9.1.5. **Description:** A body section that describes the listing.
  - 9.1.6. **Expiration Date:** Displays how long the listing is valid.
  - 9.1.7. **Tags:** Key words appended to a listing to categorize for searching.

10. **User Rating:** This is a section after completing a listing that will allow registered users to leave ratings for the sellers after a purchase. We will provide the section in order to fight against fraud.

- 10.1. **User Ratings:** Will contain the sellers username, the buyers username, a star rating on the seller, and a description body.
  - 10.1.1. **Seller Username:** This is the username of the seller who provided the listing.
  - 10.1.2. **Buy Username:** This is the username of the buyer who purchased the listing.
  - 10.1.3. **Star Ratings:** This is the section where buyers are allowed to rate a seller out of 5.
  - 10.1.4. **Description:** This is a description box with a body for a short description of the rating.

11. **Payment System:** The place where you choose your payment method for

the services or items involved.

11.1. **Type:** type of the payment.

11.2. **Amount:** The shown amount needed to be paid.

## [Section II] Prioritized Functional Requirements

### Priority 1:

1. General User
  - 1.1. A general user shall be able to browse forum posts.
  - 1.2. A general user shall be able to browse items on the marketplace.
  - 1.3. A general user shall be able to browse service listings.
  - 1.4. A general user shall be able to search for their needed service.
  - 1.5. A general user can search by categories in the marketplace.
  - 1.6. A general user shall accept our terms of conditions.
  - 1.7. A general user shall be able to submit their own ads to be promoted on our website.
  - 1.8. A general user shall be able to search by categories in the marketplace.
2. Registered User
  - 2.1. A registered user shall be able to log in with their username.
  - 2.2. A registered user shall be able to review another user.
  - 2.3. A registered user shall be able to create their own profile with text.
  - 2.4. A registered user shall be able to create their own profile with pictures.
  - 2.5. A registered user shall be able to edit their own profile with text.
  - 2.6. A registered user shall be able to edit their own profile with pictures.
  - 2.7. A registered user shall be able to post a picture as their profile picture.
  - 2.8. A registered user shall be able to post a service listing.
  - 2.9. A registered user shall be able to list items for sale in the marketplace.
  - 2.10. A registered user shall be able to post on the forums.
  - 2.11. A registered user shall be able to edit a forum post.

- 2.12. A registered user shall be able to edit a service listing.
- 2.13. A registered user shall be able to edit a sales listing.
- 2.14. A registered user shall be able to log out.
- 2.15. A registered user shall be able to message other registered users.
- 2.16. A registered user shall have an edit account option.
- 2.17. A registered user shall be able to rate other general users.

### 3. Admin

- 3.1. An admin shall be able to review reports.
- 3.2. An admin shall be able to ban registered users.
- 3.3. An admin shall be able to unban registered users.
- 3.4. An admin shall be able to delete forum posts of other users.
- 3.5. An admin shall be able to remove service listings from the website.
- 3.6. An admin shall be able to remove marketplace listings from the website.

### 4. Forum Listing

- 4.1. A Forum listing shall have a title.
- 4.2. A Forum listing post shall have characters in its content body.
- 4.3. A Forum listing post shall have basic information like time posted.
- 4.4. A Forum listing shall be visible to all general users.
- 4.5. A Forum listing post shall be able to be unlisted.

### 5. Service Listing

- 5.1. A service listing shall have a title.

- 5.2. A service listing shall have the location of where the service will be needed.
- 5.3. A service listing post shall have characters in its content body.
- 5.4. A service listing post shall be able to have images in its content body.
- 5.5. A service listing post shall have basic information like time posted.
- 5.6. A service listing post shall be able to have its content body edited.
- 5.7. A service listing post shall be able to have its title edited.
- 5.8. A service listing post shall be able to be unlisted.
- 5.9. A service listing post shall be able to have filters.
- 5.10. A service listing post shall have optional tags.

## 6. Receipt

- 6.1. A receipt shall have the product bought.
- 6.2. A receipt shall have the price of the product bought.
- 6.3. A receipt shall have the time of the transaction.
- 6.4. A receipt shall have the email of the user involved.

## 7. Shopping Cart

- 7.1. A shopping cart shall remember what has been stored within it.
- 7.2. A shopping cart shall remember how many of one entity have been stored within it.
- 7.3. A shopping cart shall allow users to edit products within it.
- 7.4. A shopping cart shall allow for check out.
- 7.5. A shopping cart shall give descriptions of products stored within it.

## 8. Checkout

- 8.1. Checkout shall allow a user to input their payment information.

- 8.2. Checkout shall allow a user to input their shipping address.
- 8.3. Checkout shall allow a user to input their billing address.
- 8.4. Checkout shall allow users to review their purchase one last time.

## 9. Navigation Bar

- 9.1. A navigation bar shall have a section for users selling their products.
- 9.2. A navigation bar shall have a section for teachers to look for student assistants.
- 9.3. A navigation bar shall have a section for the student forums.
- 9.4. A navigation bar shall have a section for the about us page.

## 10. Marketplace Listing (Buy/Sell Listing)

- 10.1. A marketplace listing shall have a title.
- 10.2. A marketplace listing shall have a description.
- 10.3. A marketplace listing shall have a photo.
- 10.4. A marketplace listing shall have a price.
- 10.5. A marketplace listing shall have filters to categorize.

## 11. Ratings

- 11.1. User ratings shall rate a profile on a scale of 1 to 5.
- 11.2. User ratings shall be visible on a registered user's profile.
- 11.3. User ratings shall be visible on a registered user's marketplace listing.
- 11.4. User ratings shall be visible on a registered user's service listing.
- 11.5. User ratings shall be able to be posted by registered users.

## **Priority 2:**

1. General User

- a. A general user shall be able to leave comments on products/services.
  - b. A general user can search by college institution in the service listing section.
2. Registered User
- a. A registered user shall do anything an unregistered user can.
  - b. A registered user shall rate products they have used.
  - c. A registered user shall be allowed to save payment information.
  - d. A registered user shall be able to save products they like.
  - e. A registered user shall be able to log in using their user id.
  - f. A registered user shall be able to log in using their email.
  - g. A registered user shall be able to pay with different methods, ie. Cash, debit, etc.
  - h. A registered user shall be able to pay to be on the top of the forums.
  - i. A registered user shall be able to report another user.
  - j. A registered user shall be able to block another user.
3. Admin
- a. An admin shall be able to view a chat log between two users.
  - b. An admin shall be able to do all that a registered user can.
  - c. An admin shall be able to see personal information of users.
  - d. An admin shall be able to close group posts.
  - e. An admin shall be able to re-open posts.
4. Receipt
- a. A receipt shall have a way to contact admin service support.
  - b. A receipt shall have the method of contacting the seller via email.

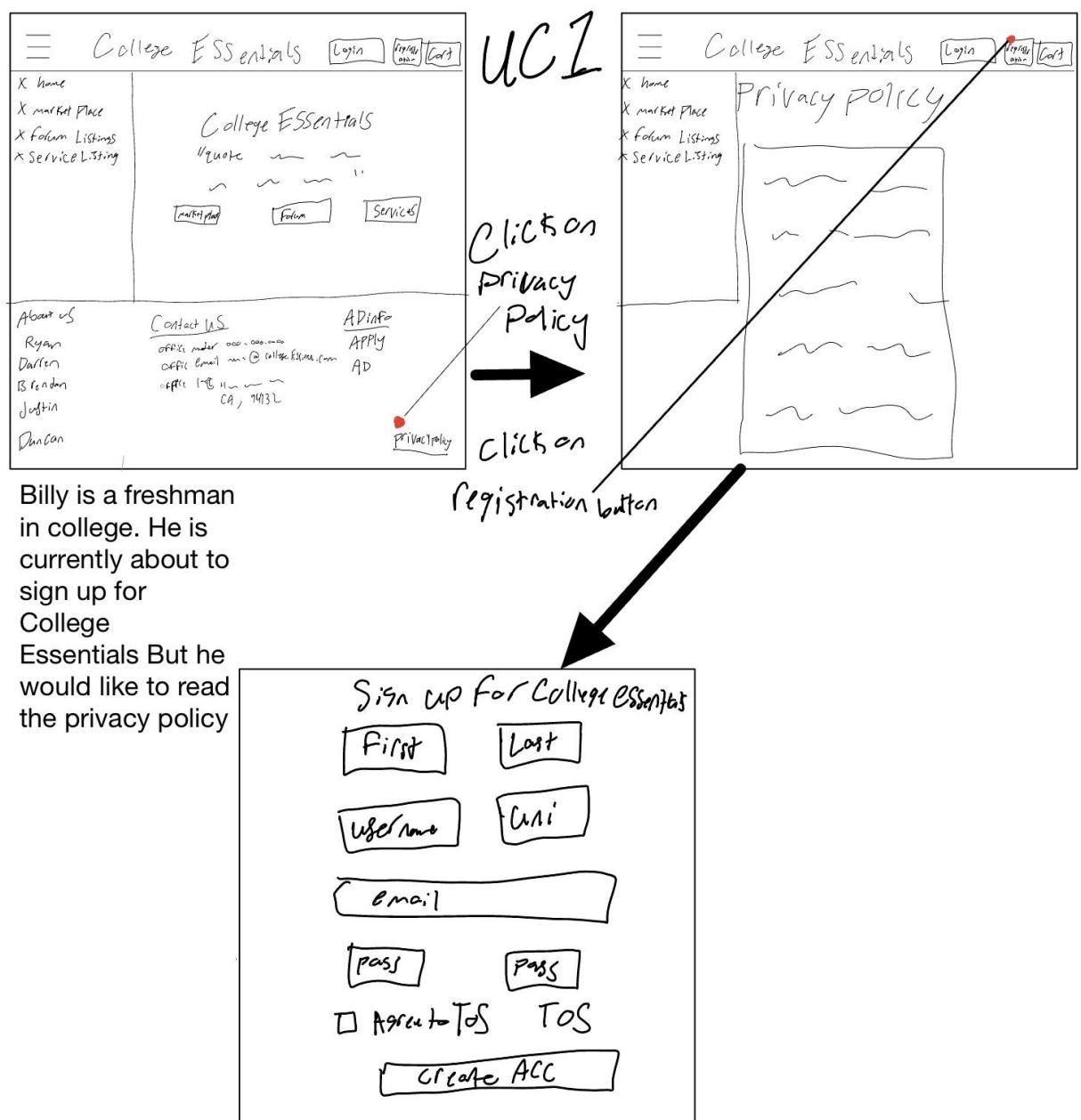
- c. A receipt shall have the method of contacting the seller via phone number.
5. Shopping Cart
- a. The shopping cart shall have a button to relocate you to the marketplace
6. Navigation Bar
- a. The Navigation Bar shall have zip code search.
7. Messages
- a. A message shall have characters in its content body.
  - b. A message shall have a timestamp.
  - c. A message shall have a status (sent/received/read).
  - d. A message shall allow a user to edit it.
  - e. A message shall allow a user to delete it.
8. Marketplace Listing (Buy/Sell Listing)
- a. A marketplace listing shall have the location of where the deal is going to be held.
  - b. A marketplace listing shall have an expiration date.
  - c. A marketplace listing shall have a bidding minimum.
  - d. A marketplace listing shall have optional tags such as condition.
  - e. A marketplace listing shall be able to be unlisted.
  - f. A marketplace listing shall have the option to be returned or not.
9. Forum Listing

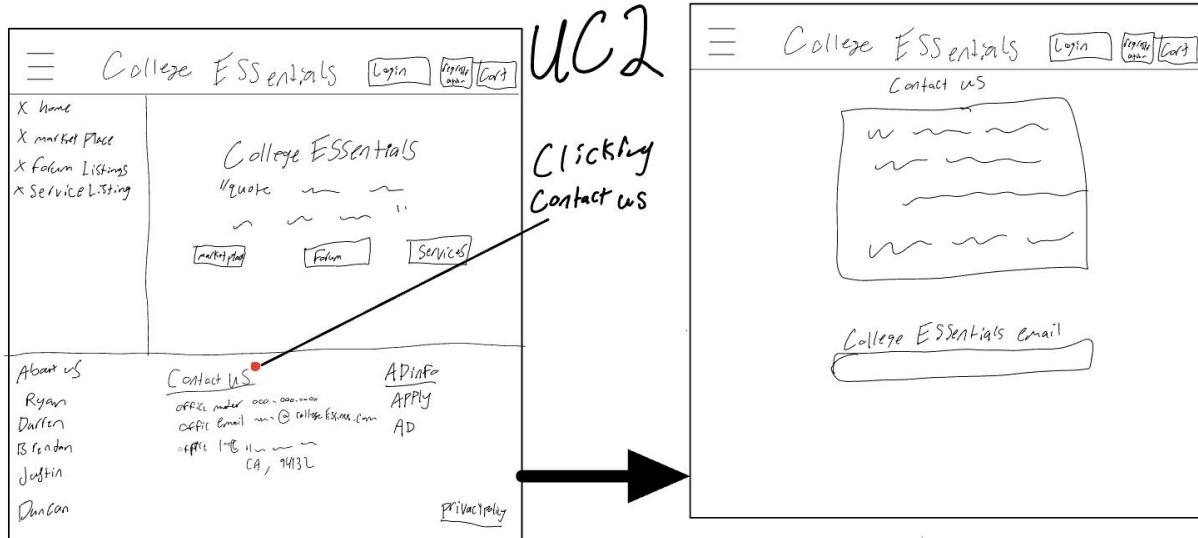
- a. A Forum listing post shall be able to have its content body edited.
- b. A Forum listing shall have the location of where the service will be needed.
- c. A Forum listing post shall be able to have images in its content body.
- d. A Forum listing post shall be able to have its title edited.
- e. A Forum listing post shall be able to be unlisted.
- f. A Forum listing post shall be able to have filters.

### **Priority 3:**

- 1. Shopping Cart
  - a. A shopping cart will have a little number showing how many items are within it, while you are not on the checkout screen.
- 2. Checkout
  - a. Checkout shall prompt users to create an account if not yet registered.

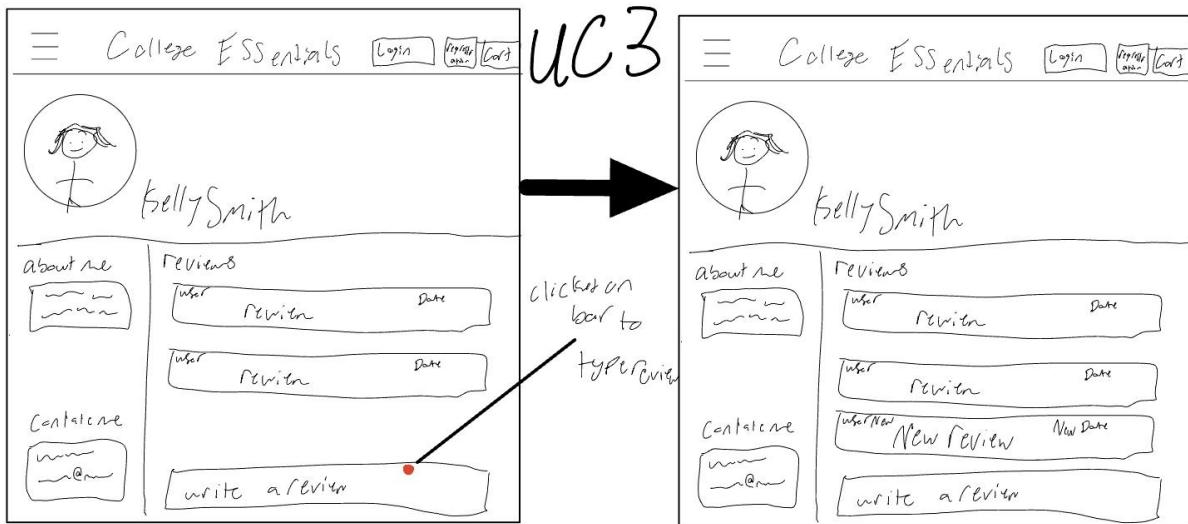
## [Section III] UI Mockups and Storyboards (High Level Only)





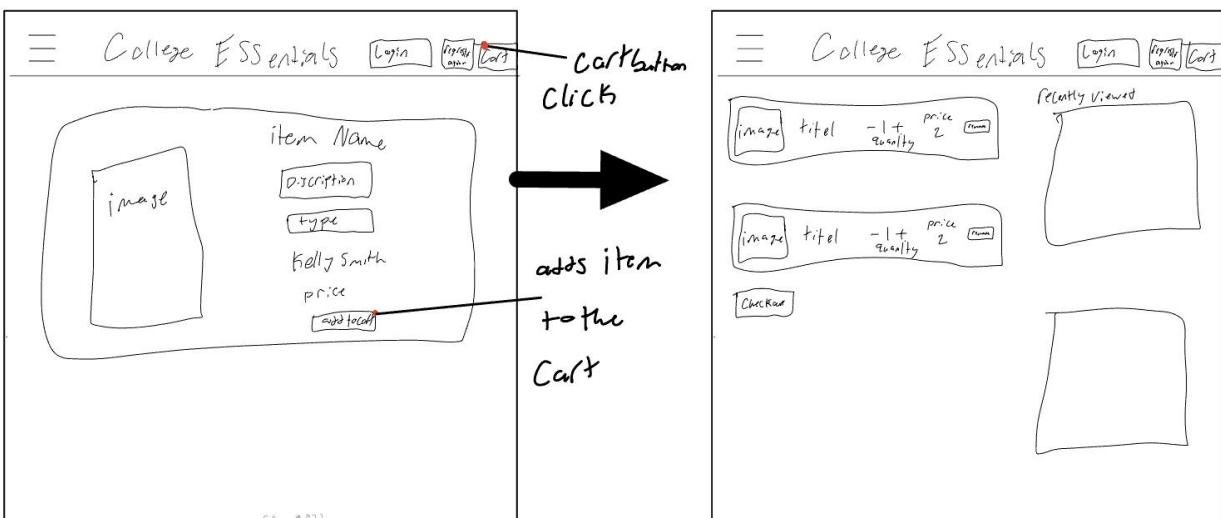
John is a freshman in college. He had been very satisfied with the website. And would love to contact the team at College Essentials he clicks on Contact Us

John can use this page to contact the team



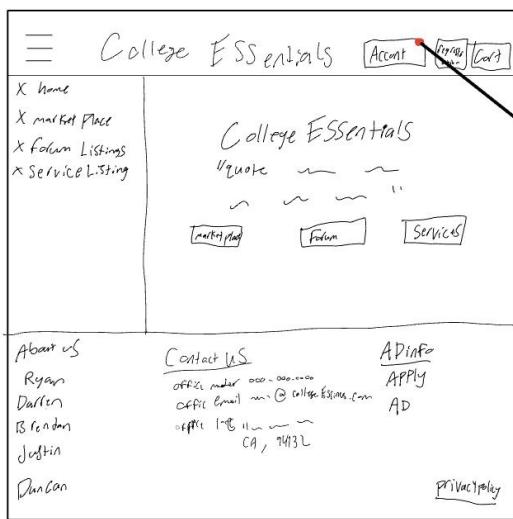
John is a freshman in college. He had been very satisfied with his textbook that he had just bought from another student, Kelly, from College Essentials. The textbook looked new inside, and out. There also didn't seem to be any graffiti or damage found within the book as well.

Since John was so happy with the condition of the textbook he had purchased, he decided to leave a review on Kelly's profile detailing the textbook's condition and how smoothly the transaction went.



A few days later, Jake sees another one of Kelly's textbooks up for sale in the marketplace. He is skeptical of Kelly at first, so he goes to Kelly's profile to see if there are any ratings. Much to Jake's delight, he finds that John loved his textbook that he purchased from Kelly a few days ago.

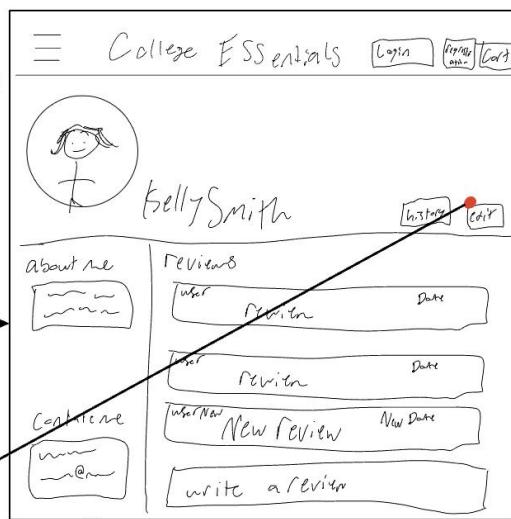
This review is enough for Jake to think that Kelly is credible, and he buys that textbook moments later.



UC 4

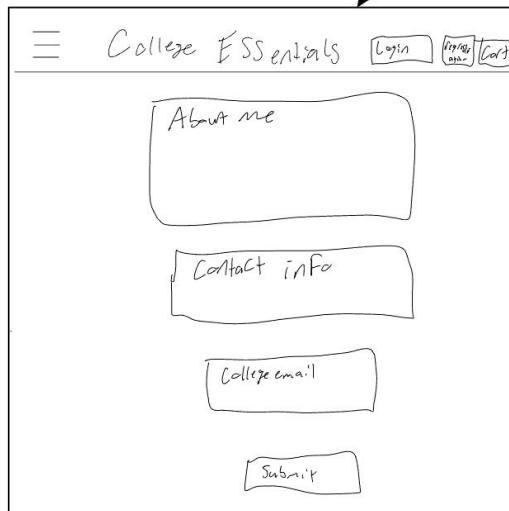
Clicking  
Account  
Personal ACC

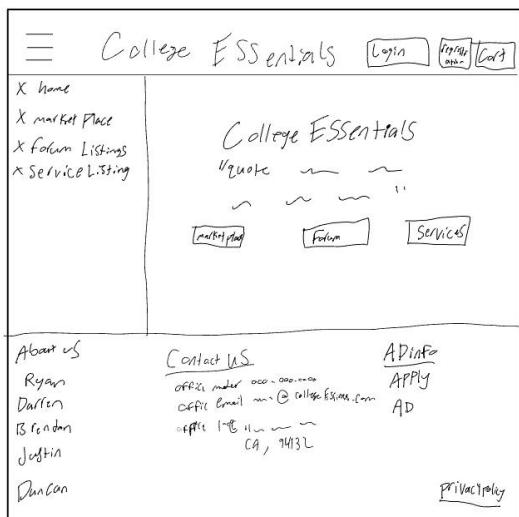
→  
Button to  
Edit Personal  
Account



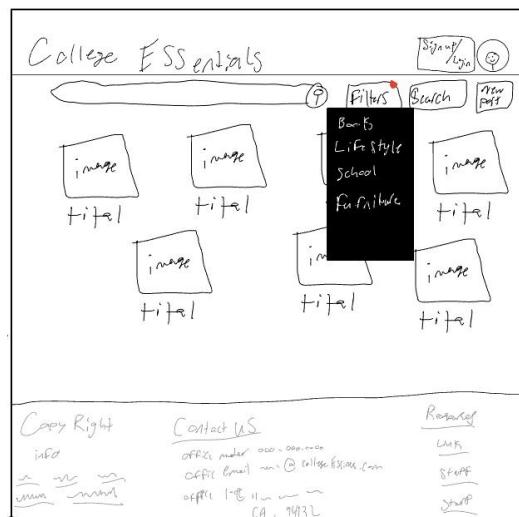
Mike then signs up for College  
Essentials in order to post a listing on  
the marketplace stating he wants to  
sell his books

Mike decides to update his  
bare profile on College  
Essentials in order to  
establish a greater sense of  
credibility. The last thing Mike  
would want is for potential  
buyers to think he is a  
scammer. In his profile, he  
writes about some  
information about himself,  
including his passions in life  
as well as his social media  
accounts. Mike follows it up  
by adding some pictures of  
him and his dog to his bio, as  
well as putting a picture of  
himself as his profile picture.

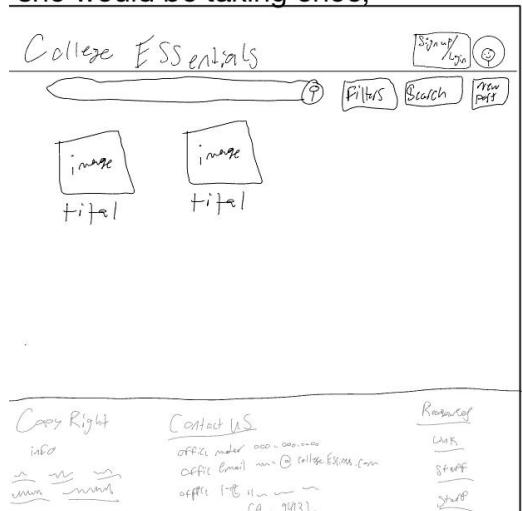




UC5

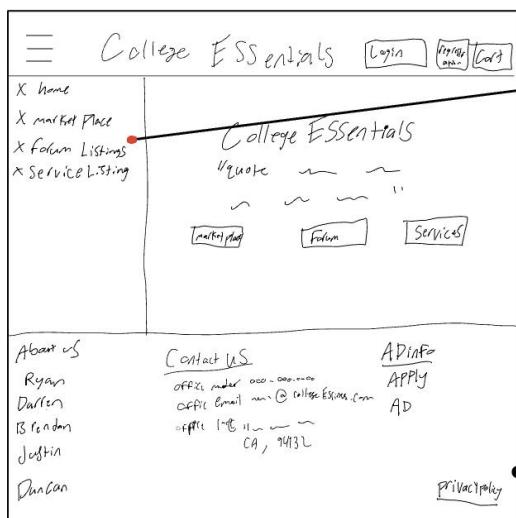


Jennifer needs to buy textbooks for her new classes, so she goes to her community college bookstore to look for the textbooks that she needs. Unfortunately, her community college doesn't offer any courses that use those textbooks. Jennifer doesn't want to buy a new condition textbook for a class that she would be taking once,



Jennifer decides to go to College Essentials's marketplace, where she is able to filter her search based on various universities as well as the condition of the textbook she needs.

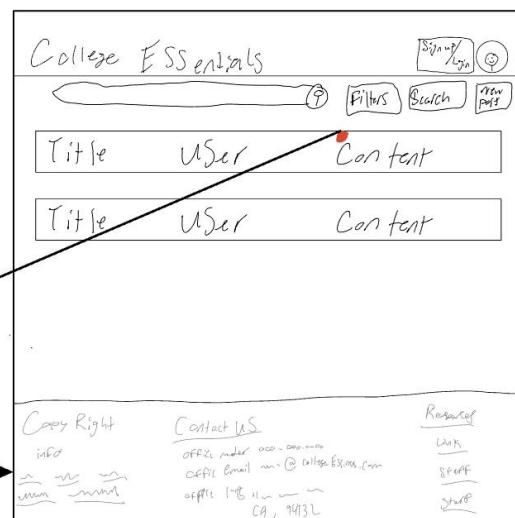
Because her local community college doesn't offer the textbooks she needs, our website would allow her to search different colleges in different cities to find someone who is selling those textbooks. If she wants to find the textbook from a closer city, she can search from schools that are closer to her. If she wants to find the textbook from her new school, she would be able to search from that school directly.



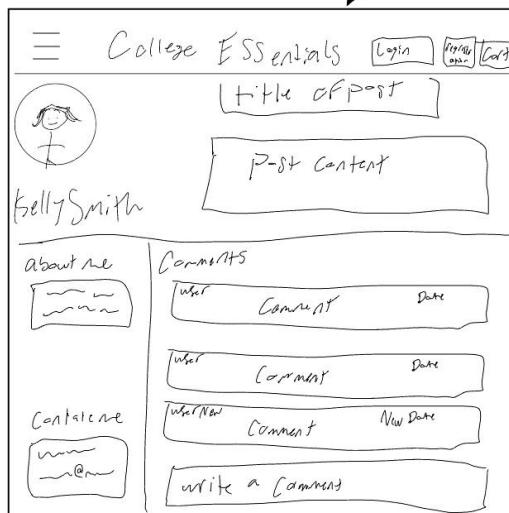
Ryan is looking for a possible opportunity to work under a professor at his college. Ryan has a poor experience with other websites, struggling to find opportunities for his university amongst the thousands of results that show for many other universities. However with College Essentials, he notices a menu that can easily return results from a specific institution.

UC6

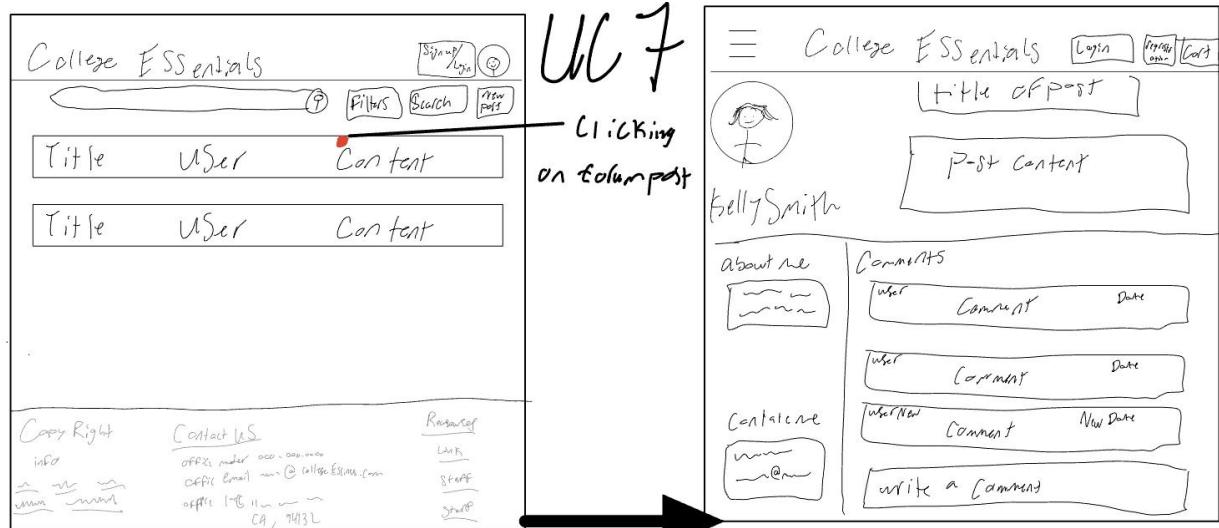
clicking on menu and forum listing  
clicking on a forum post



He therefore enables the filter and gets new results that are all from his college that he is attending. He was able to find a post from one of his favorite professors and applied.



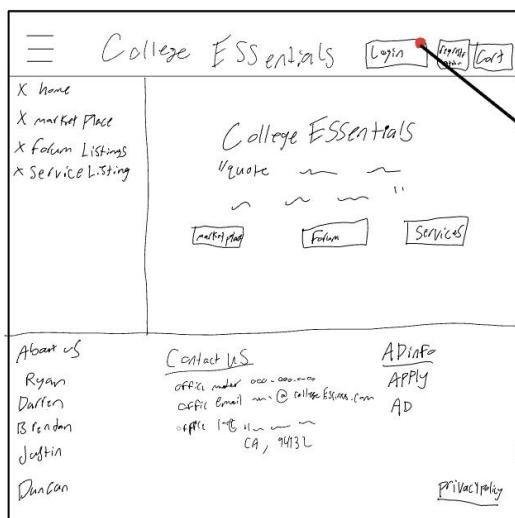
College Essentials will enable everyday users like Ryan to easily filter among many search results for services that are related to his university.



Dr. Ethan is a professor at San Francisco State University who teaches biology. He comes up with a brilliant idea and needs a few students to help him bring his idea to life. He decides to go on College Essentials and posts a job offer for students who are willing to help him with his experiments.

Students from all over the college reply to his advertisement for seeking help and he is able to choose the best qualified student to hire.

College Essentials will make it easier for Dr. Ethan to seek help for his upcoming research project. For any professor like Dr. Ethan, finding time out of his day away from holding lectures and grading work is already difficult as is.



UC8

clicking  
on login

button  
After user  
and pass

Log in button

College Essentials

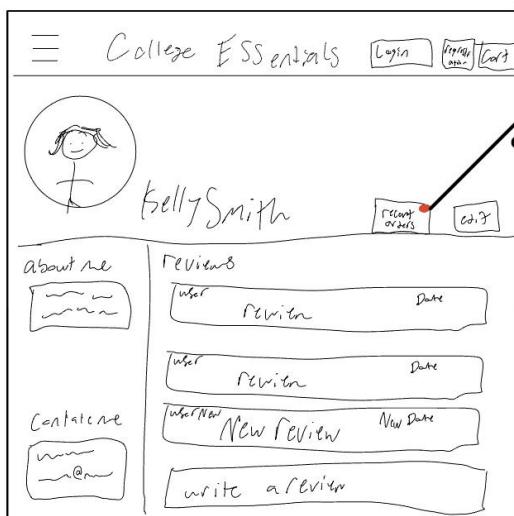
user

pass

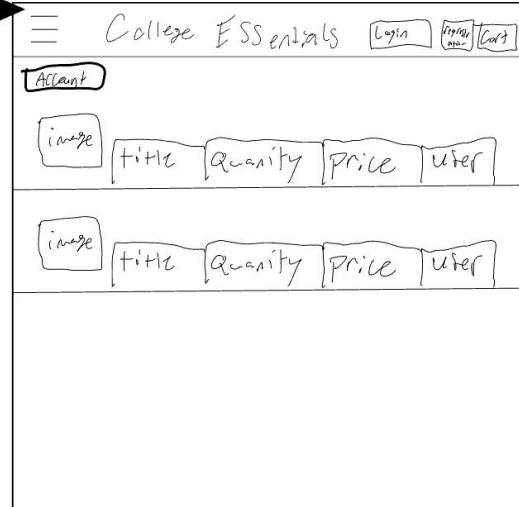
Forgot pass

Robert is loving College Essentials  
and has bought a couple of items  
from the website,

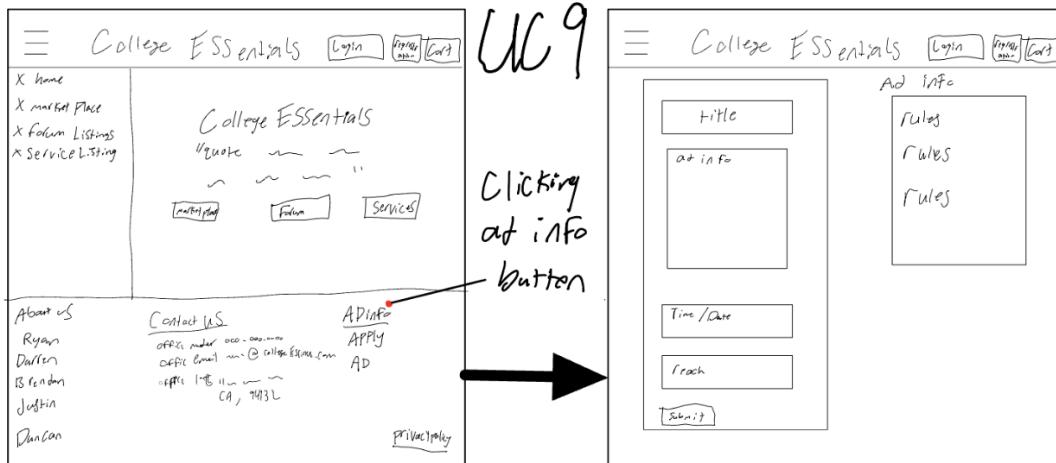
Robert is logging in to his  
account to check on his recent  
orders



Recent orders  
button pressed

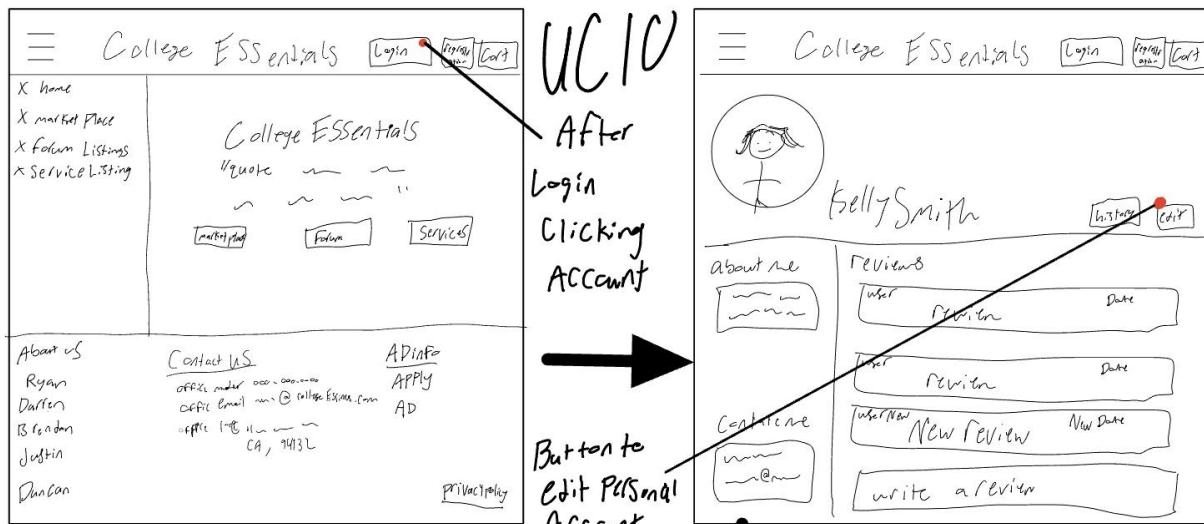


Robert goes to his account and  
clicks on recent orders to see the  
orders he has placed recently



The user Bill wants to purchase an Student Ad on the College Essentials website in order to promote his new YouTube video he made on his college

Bill clicks on our sib-link in order to purchase an Ad, adds all his ad information



The user Greg is transferring colleges and wants to change his account from SFSU to UCLA he clicks on log in after that he would click account

On his account setting page he clicks on the button edit account after that takes him to the edit page

A wireframe of a form titled "Edit Personal Account". It contains three main sections: "About me", "Contact info", and "College email". The "College email" field is highlighted with a red dot.

The College email is the way we verify the college

## [Section IV] High Level Database Architecture Organization

### 1. Business Rules

- a. GeneralUser
  - i. A GeneralUser shall be able to log into a registered account from many devices. (RegisteredUser)
  - ii. A GeneralUser shall have a ShoppingCart.
  - iii. A GeneralUser shall be able to checkout their purchases.
  - iv. A GeneralUser shall be able to view the marketplace.
  - v. A GeneralUser shall be able to view the service listings.
  - vi. A GeneralUser shall be able to view the forum listings.
  - vii. A GeneralUser shall receive many receipts for their transactions.
  
- b. RegisteredUser
  - i. A RegisteredUser shall be able to be an Admin
  - ii. A RegisteredUser shall have one or many Colleges associated with their accounts.
  - iii. A RegisteredUser shall be able to post zero or many ServiceListings.
  - iv. A RegisteredUser shall be able to post zero or many ServiceReplies
  - v. A RegisteredUser shall be able to post zero or many ForumPosts
  - vi. A RegisteredUser shall be able to post zero or many ForumReplies
  - vii. A RegisteredUser shall have zero or many Ratings.
  - viii. A RegisteredUser shall be able to post zero or many Ratings.

- ix. A RegisteredUser shall have many forms of Payment
- c. Rating
  - i. A Rating shall belong to one RegisteredUser.
  - ii. A Rating shall be posted by zero or many RegisteredUsers.
  - iii. A Rating shall be tied to zero or one Transaction
- d. Admin
  - i. An Admin shall be considered a RegisteredUser.
- e. College
  - i. A College shall be associated by many RegisteredUsers (e.g. college student or college faculty)
- f. MarketplaceListing
  - i. A Marketplace Listing shall be listed by one RegisteredUser
  - ii. A Marketplace Listing shall list at least one Product
- g. ServiceListing
  - i. A ServiceListing shall be listed by one RegisteredUser.
  - ii. A ServiceListing shall be followed by many ServiceReplies
- h. ServiceReply
  - i. A ServiceReply will be tied to many ServiceListings
  - ii. A ServiceReply is posted by a RegisteredUser
- i. Transaction

- i. A Transaction shall have many Receipts.
  - ii. A Transaction shall be linked to one Payment method.
  - iii. A Transaction shall be linked to a Buyer
  - iv. A Transaction shall be linked to a Seller
  - v. A Transaction shall be linked to a Rating
  - vi. A Transaction shall be linked to many TransactionItems
- 
- j. TransactionItems
    - i. A TransactionItems is tied to a Transaction.
    - ii. A TransactionItems is tied to a Product
- 
- k. ShoppingCart
    - i. A ShoppingCart is tied to one GeneralUser.
    - ii. A ShoppingCart is linked to one Transaction.
- 
- l. Payment
    - i. A Payment shall belong to one RegisteredUser.
    - ii. A Payment shall be linked to a Transaction.
- 
- m. Receipt
    - i. A Receipt shall be given to one RegisteredUser (buyer).
    - ii. A Receipt shall be given to one RegisteredUser (seller).
    - iii. A Receipt shall belong to one Transaction.
- 
- n. Product
    - i. A Product is ordered by zero or many GeneralUsers
    - ii. A Product is added to many ShoppingCarts

- iii. A Product is listed on a MarketplaceListing
  - iv. A Product is tied to many TransactionItems
- 
- o. ForumPost
    - i. A ForumPost is posted by one RegisteredUser
    - ii. A ForumPost shall be followed by many ForumReplies
- 
- p. ForumReply
    - i. A ForumReply will be tied to many ForumPost
    - ii. A ForumReply is posted by a RegisteredUser

## 2. Entities

- a. GeneralUser (Strong)
  - i. General\_user\_id: key, numeric; This is the unique id registered to each GeneralUser
  - ii. ip\_address: numeric; This is the IP address associated with a GeneralUser
  
- b. RegisteredUser (Weak)
  - i. registered\_user\_id: key, numeric; This is the unique id registered to each RegisteredUser.
  - ii. general\_user\_id: key, numeric; This is the id associated to the generalUser tied to RegisteredUser
  - iii. first\_name: alphanumeric; This is the first name registered to a RegisteredUser.
  - iv. last\_name: alphanumeric; This is the last name registered to a RegisteredUser

- v. password: alphanumeric; This is the unique password registered to each RegisteredUser.
- vi. email: alphanumeric; This is the unique username registered to each RegisteredUser.
- vii. about: alphanumeric; This is the about info tied to a RegisteredUser
- viii. contact: alphanumeric; This is the contact info tied to a RegisteredUser
- ix. is\_banned: numeric; This is a 0 or 1 value that denotes if the RegisteredUser is banned or not
- x. pfp: alphanumeric; This is the image link tied to a RegisteredUser's profile picture

c. Ratings (Weak)

- i. rating\_id: key, numeric; This is the unique id given to the specific rating.
- ii. rating: numeric; This is a value between 0 and 5 given to the rating.
- iii. description: alphanumeric; This is a short description of the given rating.
- iv. transaction\_id: key, numeric; This is the transaction linked to the rating.
- v. registered\_user\_sender\_id: key, numeric; This is the ID of the user giving/sending a rating
- vi. registered\_user\_receiver\_id: This is the ID of the user receiving the rating
- vii. posted\_at: timestamp; This is the timestamp of when the rating

was posted

d. Admin (Weak)

- i. admin\_id: key, numeric; This is the unique id linked to an Admin account.
- ii. registered\_user\_id: key, numeric; This is the RegisteredUser id associated with an Admin

e. College (Strong)

- i. college\_id: key, numeric; This is the unique college id associated with each college
- ii. name: alphanumeric; This is the name for a college

f. MarketplaceListing (Strong)

- i. marketplace\_listing\_id: key, numeric; This is the unique id linked with the MarketplaceListing.
- ii. title: alphanumeric; This is the title given to the MarketplaceListing.
- iii. body: alphanumeric; This is the body given to the MarketplaceListing.
- iv. created\_at: timestamp; This is the timestamp given for when a marketplace listed was posted
- v. seller\_id: key, numeric; This is the unique seller id used in a transaction.
- vi. price: numeric; This is the price associated with a MarketplaceListing
- vii. quantity: alphanumeric; This is the quantity of the product

listed

g. ServiceListing (Strong)

- i. service\_listing\_id: key, numeric; This is the unique id linked with the ServiceListing.
- ii. title: alphanumeric; This is the title given to the Service Listing.
- iii. description: alphanumeric; This is the description given to the ServiceListing.
- iv. created\_at: timestamp; This is the timestamp given to the ServiceListing.

h. ServiceReply (Weak)

- i. service\_reply\_id: key, numeric; This is the unique id linked with the ServiceReply.
- ii. description: alphanumeric; This is the description given to the ServiceReply
- iii. posted\_at: timestamp; This is when the ServiceReply was posted

i. Transaction (Strong)

- i. transaction\_id: key, numeric; This is the unique id of the transaction.
- ii. completed\_at: timestamp; This is the timestamp of when a transaction is completed
- iii. payment\_id: key, numeric; This is the unique id of a payment
- iv. seller\_id: key, numeric; This is the unique id associated with the seller (RegisteredUser) in the transaction

- v. buyer\_id: This is the unique id associated with the buyer (RegisteredUser) in the transaction
- j. TransactionItems (Weak)
  - i. Transaction\_id: numeric; This is the transaction id associated with a Transaction
  - ii. product\_id: numeric; This is the product id associated with a TransactionItem
  - iii. title: alphanumeric; This is the title associated with a product
  - iv. image: alphanumeric; This is the image associated with a product
  - v. Price: alphanumeric; This is the price associated with a product
  - vi. quantity: alphanumeric; This is the quantity associated with the product
- k. ShoppingCart (Strong)
  - i. shopping\_cart\_id: key, numeric; This is the unique id for a ShoppingCart
  - ii. buyer\_id: key, numeric; This is the unique id for a Buyer (GeneralUser)
- l. Receipt (Strong)
  - i. receipt\_id: key, numeric, This is the unique id of the receipt given in the transaction.
  - ii. seller\_id: key, numeric; This is the unique id of the seller in the transaction.

iii. buyer\_id: key, numeric; This is the unique id buyer in the transaction.

iv. transaction\_id: key, numeric; This is the unique id of the transaction.

m. Payment (Strong)

i. Payment\_id: key, numeric; This is the unique id given to the payment.

ii. payment\_type: alphanumeric; This is the type of payment used by the buyer.

iii. card\_number: numeric; This is the number associated with a card

n. Product (Strong)

i. product\_id: key, numeric; This is the unique id of the product

ii. general\_type: alphanumeric; This is the general type of the product (e.g. a textbook, bicycle, furniture, etc.)

iii. picture: alphanumeric; This the picture link associated with the product

iv. Condition: alphanumeric; This is the condition associated with a product

o. ForumPost (Strong)

i. forum\_id: key, numeric; This is the unique id of a ForumPost

ii. title: alphanumeric; This is the title of a ForumPost

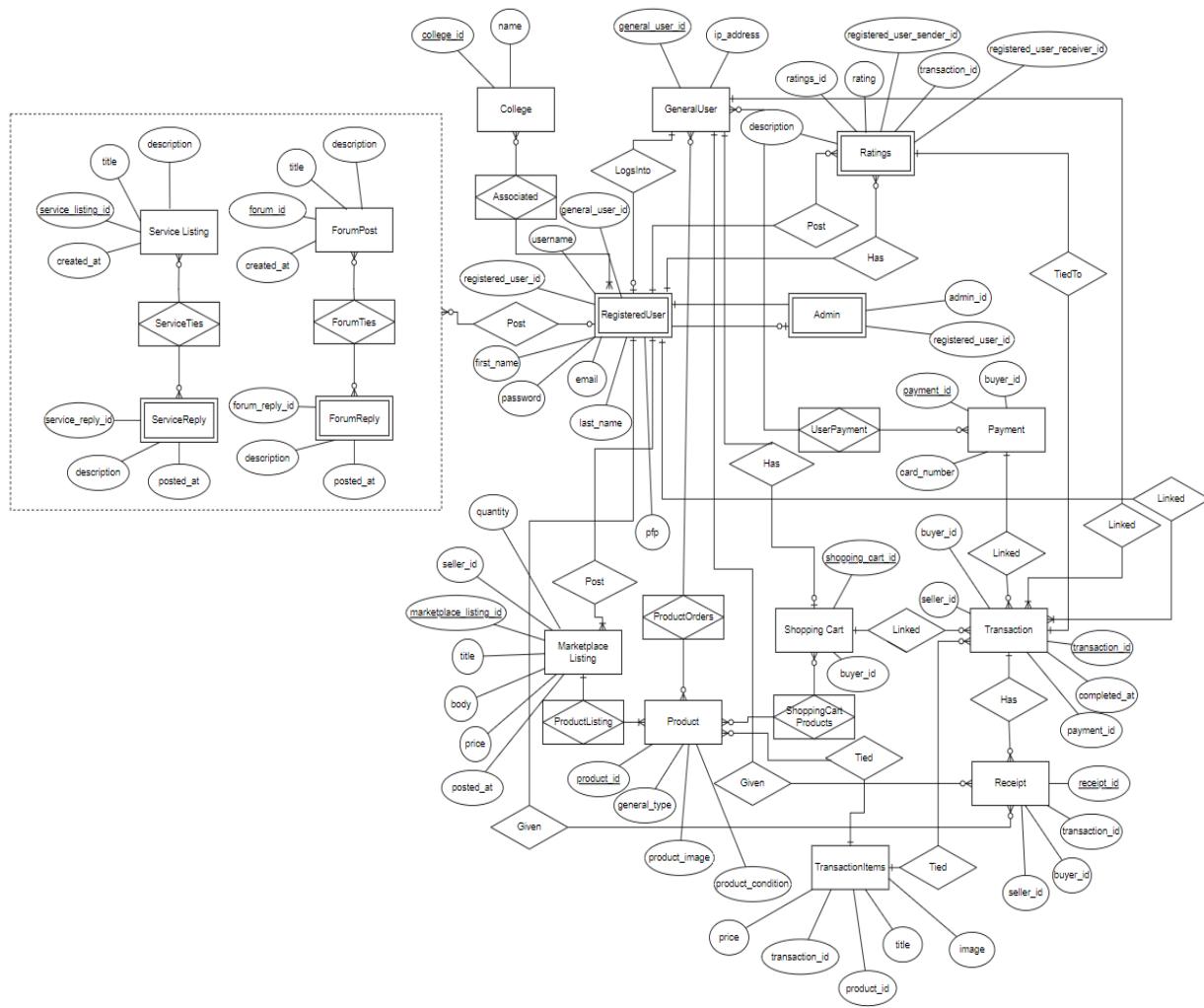
iii. description: alphanumeric; This is the description of a ForumPost

iv. `created_at`: timestamp; This is when a ForumPost was created

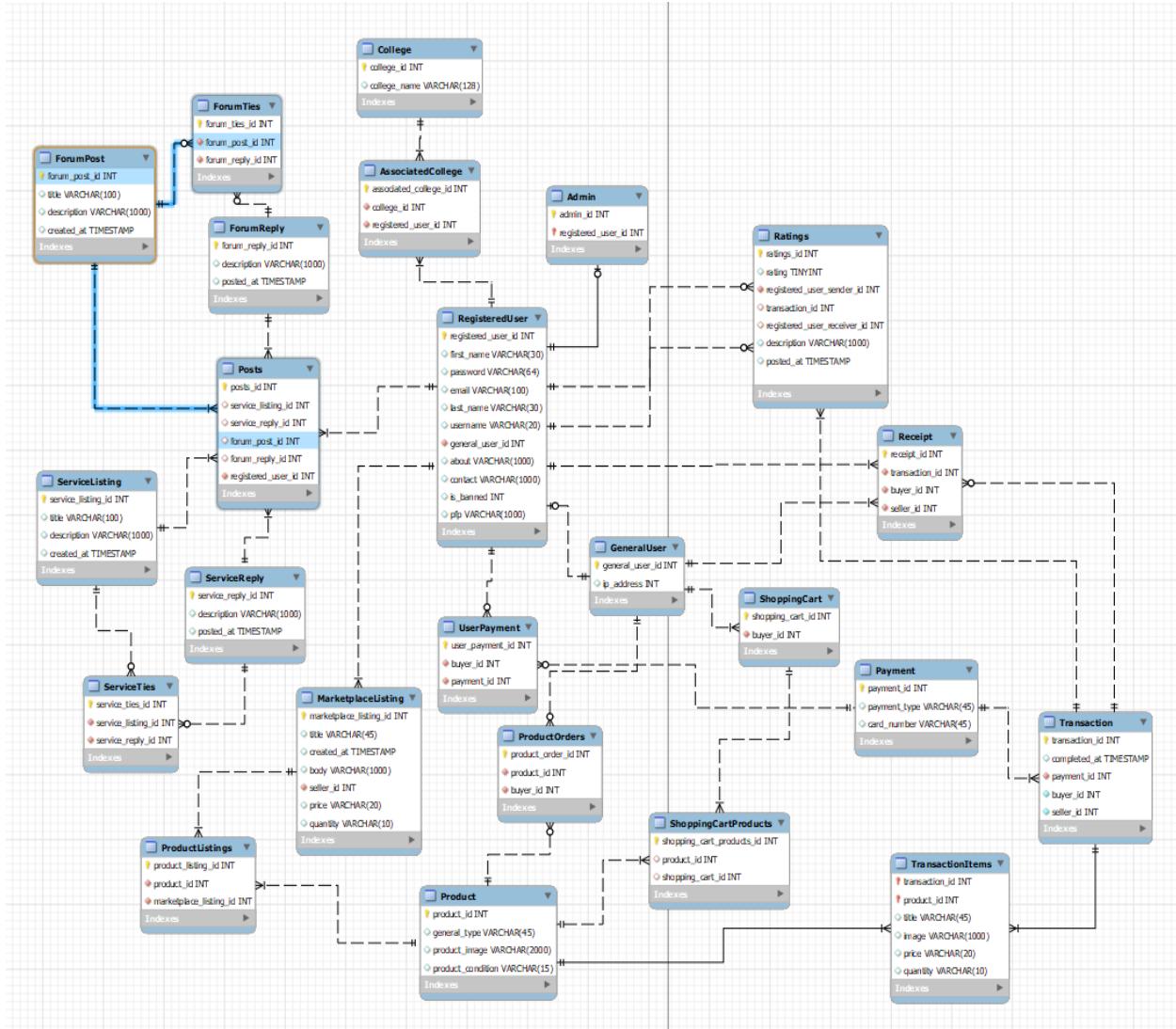
p. ForumReply (Weak)

- i. `forum_reply_id`: key, numeric; This is the unique id of a ForumReply
- ii. `description`: alphanumeric; This is the description of a ForumReply
- iii. `posted_at`: timestamp; This is when a ForumReply was posted

3. Entity Relationship Diagram (ERD)



#### 4. Database Model (EER)



## 5. DBMS

College Essentials will use MySQL as its DBMS, given its widely compatible and easy to use nature. MySQL has many tools available, such as MySQL Workbench, and continues to grow alongside cloud data platforms.

## 6. Media Storage

The College Essentials database will store images in links within the Product table, but the image and files themselves will be stored within the project's file system. This will allow us to keep the files in a consistent location, and allow us to

easily be able to locate and reference these files. Because all files will be in the file system in a possible “uploads” folder, we would be able to just implement a path in the SQL database to be able to point to the file.

## 7. Search/Filter Architecture and Implementation

- a. Our search algorithm will allow the user to use a search bar for terms within a marketplace listing. There will also be a drop down menu for specifying the condition of a product that will work in tandem with the search bar. At this time, we have filters that allow the user to search based on condition, but we plan to allow users to search based on college as well.
- b. In order to implement the search filters, we will have a table in the database that would show the specific keywords linked to each marketplace listing. Here is an example of how the search would be implemented based on no filters, using SQL LIKE to search for the keyword within the title of the marketplace listings:

```
SELECT MarketplaceListing.title, Product.product_image  
FROM ProductListings  
JOIN MarketplaceListing ON  
ProductListings.marketplace_listing_id =  
MarketplaceListing.marketplace_listing_id  
JOIN Product ON ProductListings.product_id =  
Product.product_id  
WHERE MarketplaceListing.title LIKE '%tes%';
```

- c. When searching for products using a condition filter, we will use a

similar SQL LIKE to search for the specific keyword given, however, the product condition will be added using the drop down menu with the SQL AND function. A Query example would be:

```
SELECT MarketplaceListing.title, Product.product_image,  
Product.condition FROM ProductListings  
JOIN MarketplaceListing ON  
ProductListings.marketplace_listing_id =  
MarketplaceListing.marketplace_listing_id  
JOIN Product ON ProductListings.product_id =  
Product.product_id  
WHERE MarketplaceListing.title LIKE '%searchterm%' AND  
Product.condition = "NEW";
```

## [Section V] High Level APIs and Main Algorithms

### 1. Authentication API

#### a. Registration

- Post Request: When the user wants to register for the website, the backend will receive a post request that will contain user information such as username, password, and email. The backend will check if the user information is valid, and then send a success message. If the information is not valid, the backend will send a failure message.

#### b. Login

- Post Request: When the user wants to login, the backend will receive a post request that will contain the user's information such as username, password, and email. The backend will check the database if the user information is valid and then send a success message. If the information is not valid, the backend will send a failure message and request that user information is incorrect or does not exist.

### 2. Buying / Selling API (Marketplace/Service Listings):

#### a. Selling Post Request

- When the user wants to sell a product, the backend will receive a post request with all the required information for the product. The backend will save that information in the database and return a success.

#### b. Buy Get Request

- When the user wants to buy a product, the backend will receive a get request with all the required information for the product. The backend will then return that information in a hashmap to be saved in the shopping cart.

### 3. Forum API:

#### a. Forum Post Request

- When the user wants to post a forum post, the backend will receive a post request with the necessary information for the forum post. The backend will save that information to the database and return a success message.

### 4. Search Bar API:

#### a. Search Get Request

- When the user wants to search for a marketplace item or service listing, the backend will receive a get request that contains all of the search filters requested by the user. The backend will then check the database for the information that the user is requesting, and then return that information and a success to the user if possible, otherwise will return a failure.

### 5. User Ratings API

#### a. Post Request

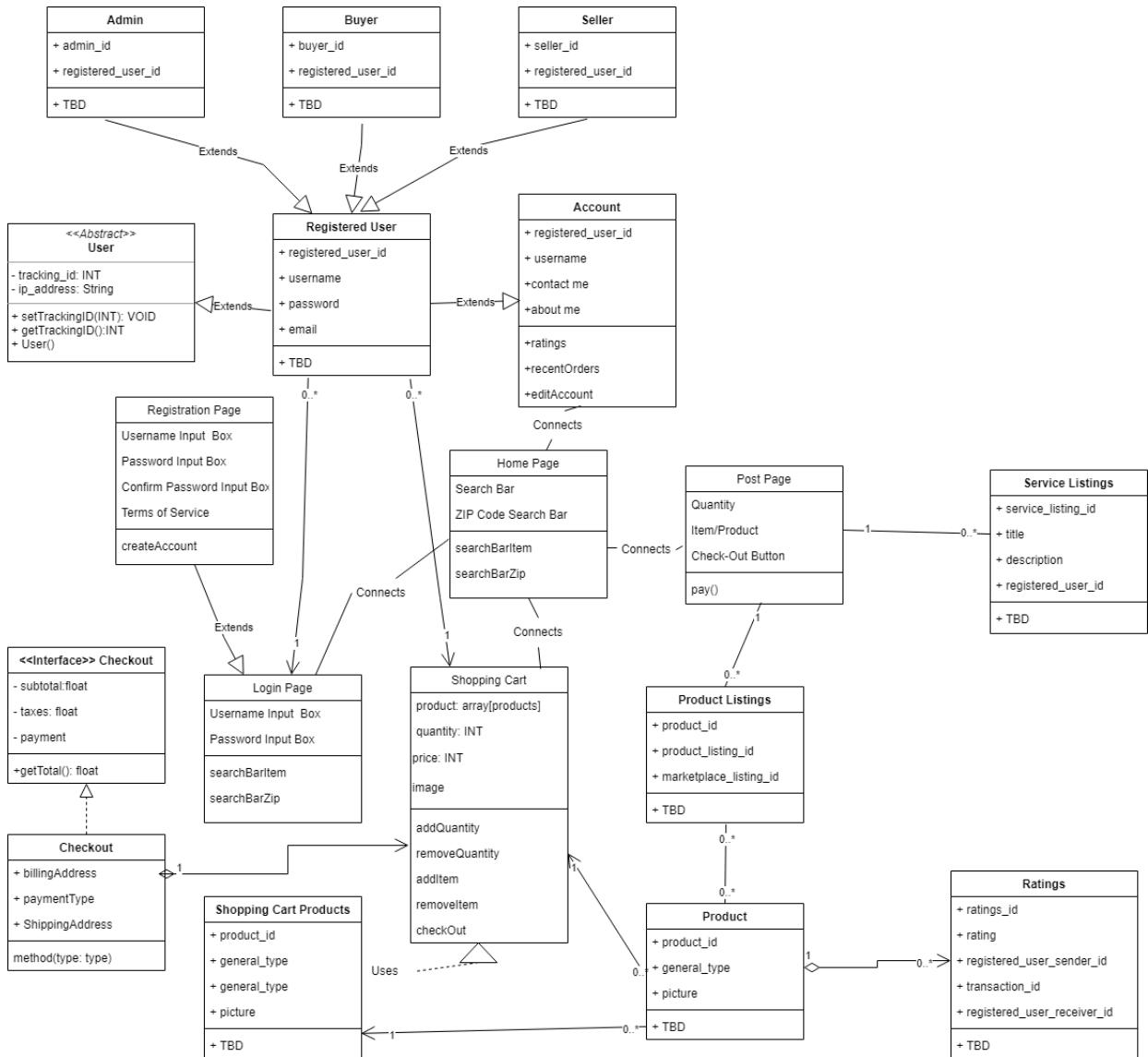
- When the user wants to post a user rating on another user's profile, the backend will receive a post request that

will receive the rating and other details of the post. The backend will then save in the database and send a success message.

b. Get Request

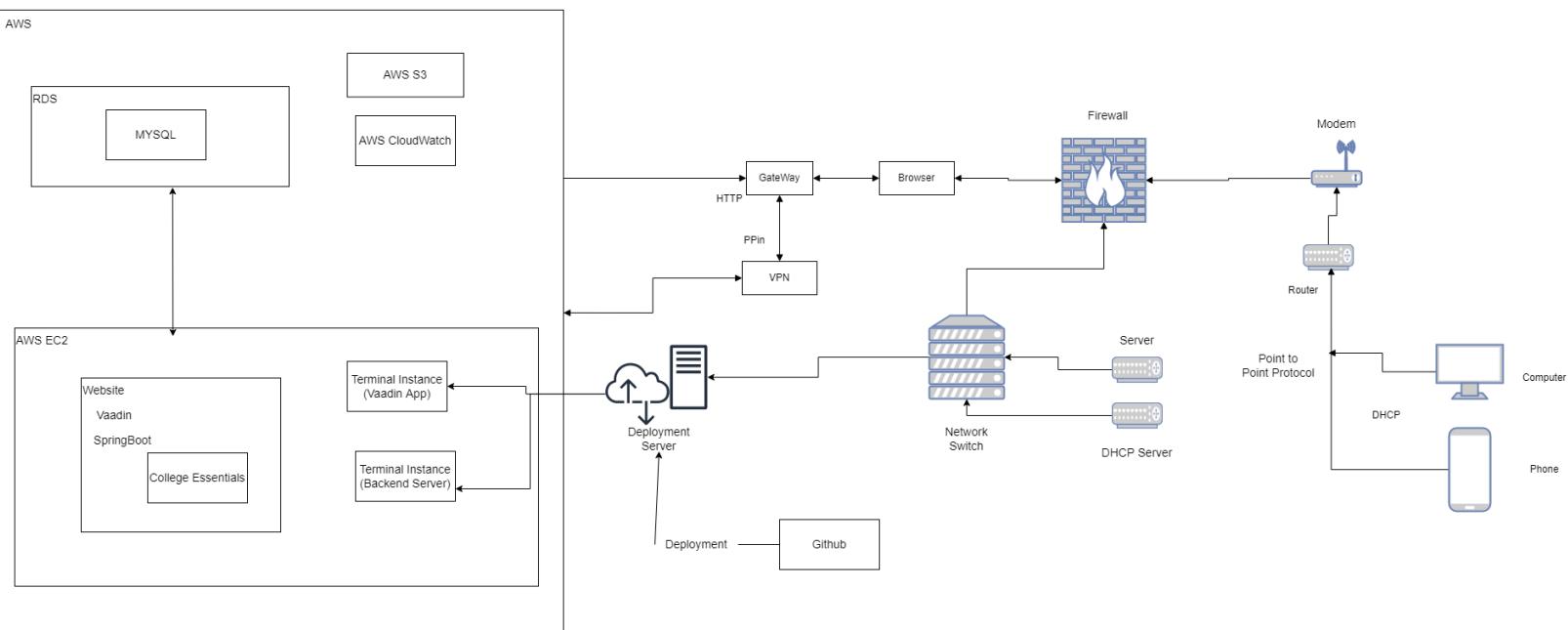
- When the user wants to view a user's profile, the backend will automatically receive a get request for the user's ratings, and then display on the user's profile page.

## [Section VI] High Level UML Diagrams

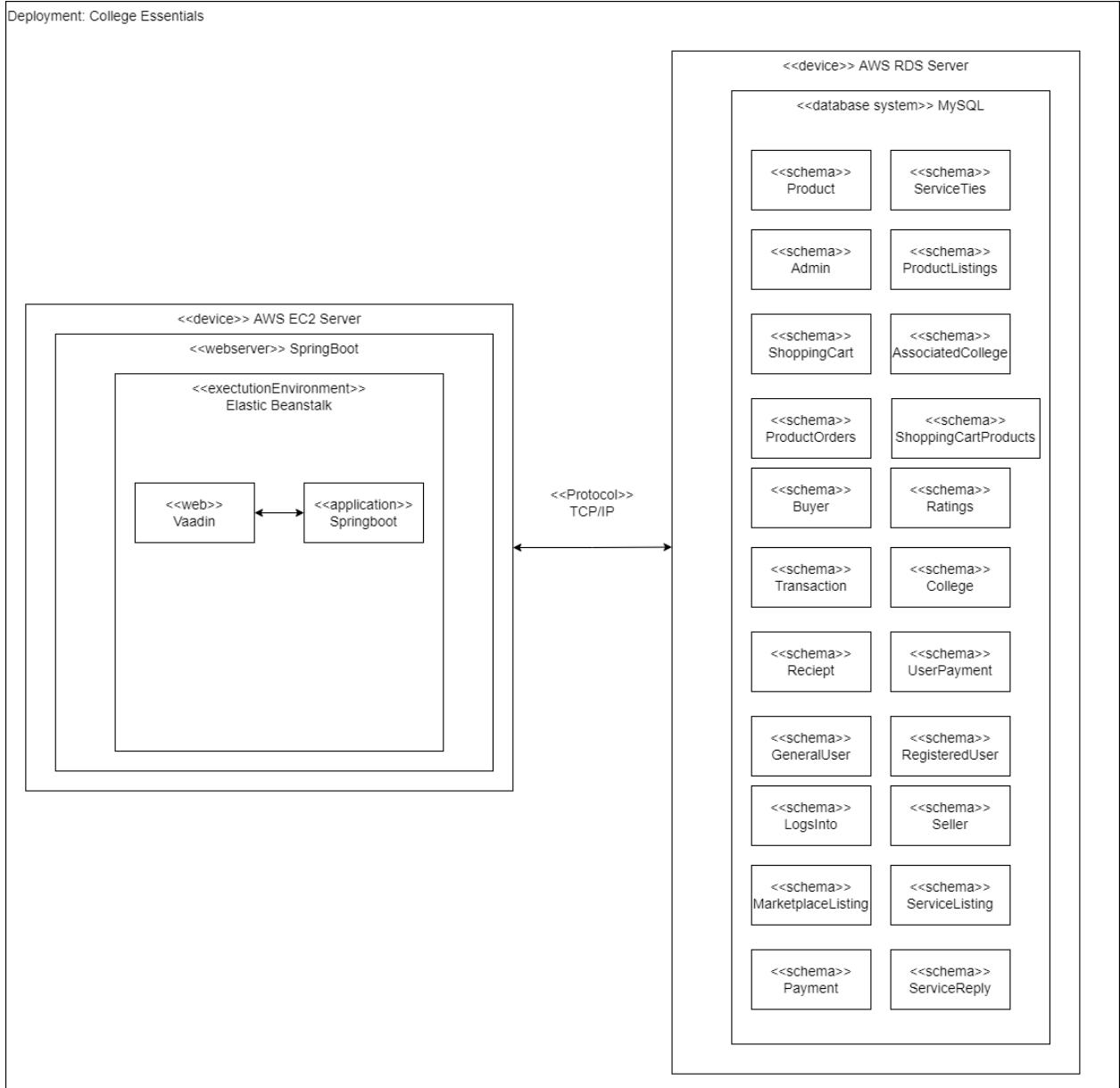


## [Section VII] High Level Application Network and Deployment Diagrams

### Network Diagram



# Deployment Diagram



## **[Section VIII] Identify Actual Key Risks For Your Project At This Time**

### **1. Skills Risks**

The main areas that we are whether or not our team are all on the same skill level when it comes to working on the code. We have two teams; one for front-end and one for back-end, and the risks that come with it are whether each member is on the same page.

To mitigate these risks, we are making sure that we all learn how to work with Vaadin and that all our team members are working with the same tech stack. We are making sure that we learn from each other to help the teammates that might have more skill risks than others.

### **2. Schedule Risks**

There are two main problems that we face with schedule risks. The first main problem is whether the time schedules of each individual member lines up with one another. The second problem is whether or not we are able to allocate enough time to do the project.

To address these problems, we have shared our schedules with each other in order to determine the best times to meet as a group. We are working together to keep each other responsible for the work that needs to be done. We are also using Trello to keep track of which tasks need to be done, and when we should have it done by.

### **3. Technical Risks**

The main areas that we are technical skill risks are related to the deployment of our app. The majority of our team has no experience with Vaadin, which is the development platform we are using to build our web app. Our other points of risks are in regards to setting up the database, and our understanding of networking.

To address these issues, our team is putting in the effort to research each area on our own, while working together to teach each other how each section should be done. We continue to work as a group to learn how to use Vaadin properly, and will continue our research on how to set up the database and broaden our knowledge of networking.

#### **4. Teamwork Risks**

The only major risks that can be considered teamwork risks are making sure that each member of the team is putting in enough work. We all have our strengths and weaknesses, the only risk is whether each member is held accountable for the work that they should be doing.

Our solution to this is to have all the members hold each other accountable for the work that they should be doing. Our team shouldn't be afraid to make sure that we are all doing the work, and that we are comfortable telling each other if they have a task that they need to do.

#### **5. Legal/Content Risks**

Some of the Legal Risks that we face would be the risks of users abusing our website. The two major problems related to this would be suspicious service listings, and scamming other users.

To mitigate these risks, we have a moderation system where users would be able to report listings as suspicious, and then moderators would be able to delete

these suspicious listings. We also have a rating system that would allow people to make sure that the people they are buying from are safe.

## **[Section IX] Project Management**

As a team, we started using Trello in order to determine which tasks we needed to get done, and how we wanted to manage our time when doing so. We have trello split our tasks into three sections: TODO, in progress, and done. Along with this, we consistently met on Discord to discuss which sections we needed to work on, as well as how we wanted to split the work. Initially, we would all take a look at each task, attempt working on it as a group, and then split individually to finish up every task. After that, we would regroup and add our completed tasks Discord and Github, and then as a group we would go over final checks before fully submitting the task as completed. Then we would update Trello and continue on working with the other tasks. We used Discord to maintain communication, giving continuous updates on our own tasks and helping each other in areas that we needed help in. We shall continue to use Discord as our main way of communicating along with consistent meetings, and using Trello as our main checklist for which tasks we still need to complete.

## **[Section X] Detailed List of Contributions**

### **I. Ryan**

- A. Facilitated group meetings and plans
- B. Contributed to the initial network diagram.
- C. Researched and documented the High Level APIs
- D. Documented the list of contributions for each team member.
- E. Worked on frontend for the vertical prototype.
  - 1. Worked on the CSS for the login, register page, and home view.
  - 2. Designed the file system that the project will be using throughout the web application.
  - 3. Troubleshooted the remaining problems that occurred in the backend code.

### **II. Duncan**

- A. Research and created all the UI Mockups needed for the use cases.
- B. Created and organized some prioritized functions needed.
- C. Worked on the frontend for the vertical prototype.
  - 1. Worked on some CSS for the home view page.

### **III. Justin**

- A. Wrote the data definitions needed for each data group.
- B. Created and organized some prioritized functions needed.
- C. Worked on the business rules, search/filter function and media storage for the web application.
- D. Helped edit the high level UML diagrams.
- E. Created and edited the network and deployment diagrams.
- F. Discussed and documented the key risks and project management sections of Milestone 2.

- G. Edited the Milestone 2 document to be accurate and consistent.
- H. Worked on the deployment of the webapp to AWS and some backend code needed for the deployment.

#### **IV. Brendan**

- A. Wrote the data definitions needed for each data group.
- B. Created and organized some prioritized functions needed.
- C. Created the first draft for the high level UML diagram.
- D. Created and documented the guidelines for the UML diagram.

#### **V. Darren**

- A. Created and organized some prioritized functions needed.
- B. Worked on the business rules needed for the database.
- C. Created the ERD and EER diagrams for the database.
- D. Worked on creating the database and connecting the database to the website.
- E. Edited the Milestone 2 document to be accurate and consistent.
- F. Worked on the backend for the prototype.
  - 1. Connected the database to the website, including the backend needed for the diagram.
  - 2. Troubleshooted the remaining problems that occurred in the backend code.

# **SW Engineering CSC648/848 Summer 2022**

## **College Essentials**

### **Team 4**

| <b>Student Name</b> | <b>Student Email</b>   | <b>Github Username</b> | <b>Student ID</b> | <b>Student Roles</b>       |
|---------------------|------------------------|------------------------|-------------------|----------------------------|
| Ryan Murphy         | rmurphy4@mail.sfsu.edu | deadlepricon           | 921366560         | Team Lead,<br>Backend Lead |
| Duncan Forniss      | Dforniss@mail.sfsu.edu | DunkeyFunkey           | 917678507         | Frontend Lead              |
| Wee Yang Leong      | wleong@mail.sfsu.edu   | ktleong2000            | 918614936         | Backend Assistant          |
| Darren Wong         | dwong19@mail.sfsu.edu  | dawong025              | 918565679         | Database Master            |
| Justin Wong         | jwong66@mail.sfsu.edu  | jwong66                | 918516578         | Github Master              |

**Milestone:** Milestone 3

**Date:** 07/19/2022

## **Version History**

| <b>Milestone</b> | <b>Date</b> |
|------------------|-------------|
| M3V2             | 07/28/2022  |
| M3V1             | 07/19/2022  |
| M2V2             | 07/19/2022  |
| M2V1             | 07/06/2022  |
| M1V2             | 07/01/2022  |
| M1V1             | 06/21/2022  |

## **Table of Contents**

|   |           |
|---|-----------|
| <b>[Section I] Data Definitions</b>                                   | <b>4</b>  |
| <b>[Section II] Functional Requirements</b>                           | <b>10</b> |
| <b>[Section III] Wireframes Based on Mockups/Storyboards</b>          | <b>16</b> |
| <b>[Section IV] High Level Database Architecture and Organization</b> | <b>23</b> |
| <b>[Section V] High Level Diagrams</b>                                | <b>36</b> |
| <b>[Section VI] List of Contributions</b>                             | <b>37</b> |

## [Section I] Data Definitions

1. **General User:** A user who is not signed in or does not have an account with College Essentials. A general user has the minimum permissions on the website and can **register** for more permissions.
  - 1.1. **Registration:** A general user shall be able to create an account to get more permissions.
  - 1.2. **Account Contains:**
    - 1.2.1. **Username:** Needs a username to create an account
    - 1.2.2. **Password:** Needs a password that is encrypted to create an account.
    - 1.2.3. **Email:** Needs a unique edu email to create an account.
    - 1.2.4. **School:** Can choose to register a school to the account.
    - 1.2.5. **Accept Terms of Service:** Must accept the terms of service to create an account.
2. **Registered User:** A user who has signed up to use College Essentials. This user gains the extra permissions needed to access the other features on the website. A general user shall be able to **login** to gain access to all these features.
  - 2.1. Can login to the system.
    - 2.1.1. Username: Needs a unique username to login.
    - 2.1.2. Password: Needs a unique password to login.
  - 2.2. Has the ability to post Marketplace Posts.
    - 2.2.1. **Buyers:** All registered users are buyers by default. They can see and interact with Marketplace posts.

- 2.2.2. **Sellers:** All registered users can put up their own Marketplace posts to become a seller. Sellers will be differentiated on the Marketplace post that they have posted.
- 2.3. Has the ability to post forum Posts.
- 2.4. **Profile:** Contains username, school, ratings, private messaging section and marketplace posts.
  - 2.4.1. **Username:** Shows the unique username of the user.
  - 2.4.2. **School:** Shows the listed schools linked with the user.
  - 2.4.3. **Ratings:** Shows the ratings on the user posted by other users.
  - 2.4.4. **Private Message:** Has an area that allows users to private message other users.
  - 2.4.5. **Marketplace Posts:** Has a list of marketplace posts made by the user.
  - 2.4.6. **Service Listings:** Has a list of service listings made by the user.
- 3. **Admin:** A user who moderates content on the website. A moderator still has the same permissions that registered users have, but have additional **admin tools** on our website.
  - 3.1. **Admin Tools:** Contains the ability to delete posts, and temporarily restrict users.
    - 3.1.1. **Delete Posts:** Can choose to delete posts by other users.
    - 3.1.2. **Restrict Users:** Can restrict user accounts who break the terms of service.
- 4. **Guest:** This is a user who is unregistered or not logged in who is able to view all posts but not reply. Guests will be ghosts within the site, they are able to search and look but not interact.

- 4.1. **Username:** Will show as a guest with numbers ie. Guest###
  - 4.2. **Browsing:** A guest will be able to browse listings, and add them to cart but not complete the transaction
  - 4.3. **Ads:** A guest shall be able to contact, and pay us to have their ads on the site.
5. **Login / User Registration:** This is the section that would allow users to login or register accounts with College Essentials.
  - 5.1. **Login Section:** Users have the ability to login to their accounts here.
    - 5.1.1. **Username:** Users will have to enter a unique username to login.
    - 5.1.2. **Password:** Users will have to enter a unique password to login.
  - 5.2. **Registration Section:** Users have the ability to register new accounts here.
    - 5.2.1. **Username:** Needs a username to create an account
    - 5.2.2. **Password:** Needs a password that is encrypted to create an account.
    - 5.2.3. **Email:** Needs a unique edu email to create an account.
    - 5.2.4. **School:** Can choose to register a school to the account.
    - 5.2.5. **Accept Terms of Service:** Must accept the terms of service to create an account.
6. **Private Message:** This is the section that allows users to private message each other through their profiles.
  - 6.1. **Contacting a Person:** Will allow the contacting of a user, client, seller, guest, etc.
  - 6.2. **Timestamp:** Shows when a message was sent.

6.3. **Read Status:** Will show if a message was seen

7. **Forum Post Section:** This is where users are able to post miscellaneous forum posts that don't fit within the other categories. Users must be registered users to make a forum post.

7.1. **Forum Posts:** Contains a post body and post replies.

7.1.1. **Post Body:** A registered user will be able to post a forum post that contains username, post body, and timestamps.

7.1.2. **Post Replies:** A registered user will be able to reply underneath the forum posts that contains a body, the replier's username, and a timestamp.

8. **Marketplace Listings:** A place for registered users to post if they want to buy or sell marketplace listings.

8.1. **Marketplace Listing:** Contains title, location, photo, price, description, expiration date, bidding(If allowed by seller), and tags.

8.1.1. **Title:** Gives a small description of the item at hand.

8.1.2. **Location:** Gives the location of where the deal is to be held.

8.1.3. **Photo:** If provided there will be a picture of the product.

8.1.4. **Price:** Displays the price of the item being sold.

8.1.5. **Description:** A body section that describes the listing.

8.1.6. **Expiration Date:** Displays how long the listing is valid.

8.1.7. **Bidding:** Shows the current minimum bid.

8.1.8. **Tags:** Key words appended to a listing to categorize for searching.

9. **Service Listings:** A place for registered users to post if they need or provide

services.

- 9.1. **Service Listings:** Contains, a Title, Location, Photo, Price, Description, an expiration date, and tags.

- 9.1.1. **Title:** Gives a small description of the item at hand.
- 9.1.2. **Location:** Gives the location of where the deal is to be held.
- 9.1.3. **Photo:** If provided there will be a picture of the product.
- 9.1.4. **Price:** Displays the price of the item being sold.
- 9.1.5. **Description:** A body section that describes the listing.
- 9.1.6. **Expiration Date:** Displays how long the listing is valid.
- 9.1.7. **Tags:** Key words appended to a listing to categorize for searching.

10. **User Rating:** This is a section after completing a listing that will allow registered users to leave ratings for the sellers after a purchase. We will provide the section in order to fight against fraud.

- 10.1. **User Ratings:** Will contain the sellers username, the buyers username, a star rating on the seller, and a description body.

- 10.1.1. **Seller Username:** This is the username of the seller who provided the listing.
- 10.1.2. **Buy Username:** This is the username of the buyer who purchased the listing.
- 10.1.3. **Star Ratings:** This is the section where buyers are allowed to rate a seller out of 5.
- 10.1.4. **Description:** This is a description box with a body for a short description of the rating.

11. **Payment System:** The place where you choose your payment method for

the services or items involved.

11.1. **Type:** type of the payment.

11.2. **Amount:** The shown amount needed to be paid.

## [Section II] Functional Requirements

### Priority 1:

1. General User
  - 1.1. A general user shall be able to browse forum posts.
  - 1.2. A general user shall be able to browse items on the marketplace.
  - 1.3. A general user shall be able to browse service listings.
  - 1.4. A general user shall be able to search for their needed service.
  - 1.5. A general user shall accept our terms of conditions.
  - 1.6. A general user shall be able to search by category in the marketplace.
2. Registered User
  - 2.1. A registered user shall be able to log in with their username.
  - 2.2. A registered user shall be able to review another user.
  - 2.3. A registered user shall be able to create their own profile with text.
  - 2.4. A registered user shall be able to create their own profile with pictures.
  - 2.5. A registered user shall be able to edit their own profile with text.
  - 2.6. A registered user shall be able to edit their own profile with pictures.
  - 2.7. A registered user shall be able to post a picture as their profile picture.
  - 2.8. A registered user shall be able to post a service listing.
  - 2.9. A registered user shall be able to list items for sale in the marketplace.
  - 2.10. A registered user shall be able to post on the forums.
  - 2.11. A registered user shall be able to log out.
  - 2.12. A registered user shall be able to email other registered users.
  - 2.13. A registered user shall be able to rate other general users.
3. Service Listing

- 3.1. A service listing shall have a title.
  - 3.2. A service listing shall have the location of where the service will be needed.
  - 3.3. A service listing post shall have characters in its content body.
  - 3.4. A service listing post shall have basic information like time posted.
  - 3.5. A service listing post shall be able to be unlisted.
  - 3.6. A navigation bar shall have a section for teachers to look for student assistants.
4. Shopping Cart
  - 4.1. A shopping cart shall remember what has been stored within it.
  - 4.2. A shopping cart shall remember how many of one entity have been stored within it.
  - 4.3. A shopping cart shall allow users to edit products within it.
  - 4.4. A shopping cart shall allow for check out.
  - 4.5. A shopping cart shall give descriptions of products stored within it
5. Receipt
  - 5.1. A receipt shall have a way to contact admin service support.
  - 5.2. A receipt shall have the product bought.
  - 5.3. A receipt shall have the price of the product bought.
6. Checkout
  - 6.1. Checkout shall allow a user to input their payment information.
  - 6.2. Checkout shall allow a user to input their shipping address.
  - 6.3. Checkout shall allow a user to input their billing address.

6.4. Checkout shall allow users to review their purchase on last time.

## 7. Navigation Bar

7.1. A navigation bar shall have a section for students selling their products.

7.2. A navigation bar shall have a section for the student forums.

7.3. A navigation bar shall have a section for the about us page.

## 8. Marketplace Listing (Buy/Sell Listing)

8.1. A marketplace listing shall have a title.

8.2. A marketplace listing shall have a description.

8.3. A marketplace listing shall have a photo (or many).

8.4. A marketplace listing shall have a price.

8.5. A marketplace listing shall be able to be unlisted.

8.6. A marketplace listing shall have filters to categorize.

## 9. Ratings

9.1. User ratings shall rate a profile on a scale of 1 to 5.

9.2. User ratings shall be visible on a registered user's profile.

9.3. User ratings shall be visible on a registered user's marketplace listing.

9.4. User ratings shall be visible on a registered user's service listing.

9.5. User ratings shall be able to be posted by registered users.

## 10. Forum Listing

10.1. A Forum listing shall have a title.

10.2. A Forum listing post shall have characters in its content body.

10.3. A Forum listing post shall have basic information like time posted.

10.4. A Forum listing shall be visible to all general users.

10.5. A Forum listing post shall be able to be unlisted.

## **Priority 2:**

### 1. General User

- a. A general user shall be able to leave comments on products/services.
- b. A general user can search by college institution in the marketplace.
- c. A general user can search by college institution in the service listing section.
- d. A general user shall be able to submit their own ads to be promoted on our website.
- e. A registered user shall be able to edit a forum post.
- f. A registered user shall be able to edit a service listing.
- g. A registered user shall be able to edit a sales listing.
- h. A registered user shall be able to pay with different methods, ie. Cash, debit, etc.
- i. A registered user shall be able to pay to be on the top of the forums.
- j. A registered user shall be able to report another user.
- k. A registered user shall be able to block another user.

### 2. Registered User

- a. A registered user shall do anything an unregistered user can.
- b. A registered user shall rate products they have used.
- c. A registered user shall be allowed to save payment information.
- d. A registered user shall be able to save products they like.
- e. A registered user shall be able to log in using their user id.
- f. A registered user shall be able to log in using their email.

g. A registered user shall be able to message other registered users.

### 3. Service Listings

- a. A service listing post shall be able to have filters.
- b. A service listing post shall have optional tags.
- c. A service listing post shall be able to have images in its content body.
- d. A service listing post shall be able to have its content body edited.
- e. A service listing post shall be able to have its title edited.

### 4. Admin

- a. An admin shall be able to review reports.
- b. An admin shall be able to ban registered users.
- c. An admin shall be able to unban registered users.
- d. An admin shall be able to delete forum posts of other users.
- e. An admin shall be able to remove service listings from the website.
- f. An admin shall be able to remove marketplace listings from the website.
- g. An admin shall be able to view a chat log between two users.
- h. An admin shall be able to do all that a registered user can.
- i. An admin shall be able to see personal information of users.
- j. An admin shall be able to close group posts.
- k. An admin shall be able to re make posts.
- l. An admin shall be able to see IP's.

### 5. Receipt

- a. A receipt shall have a way to contact admin service support.
- b. A receipt shall have the method of contacting the seller via email..

- c. A receipt shall have the method of contacting the seller via phone number.
- d. A receipt shall have the time of the transaction.

## 6. Shopping Cart

- a. The shopping cart shall have a button to relocate you to the marketplace

## 7. Navigation Bar

- a. The Navigation Bar shall have zip code search.

## 8. Messages

- a. A message shall have characters in its content body.
- b. A message shall have a timestamp.
- c. A message shall have a status (sent/received/read).
- d. A message shall allow a user to edit it.
- e. A message shall allow a user to delete it.

## 9. Marketplace Listing (Buy/Sell Listing)

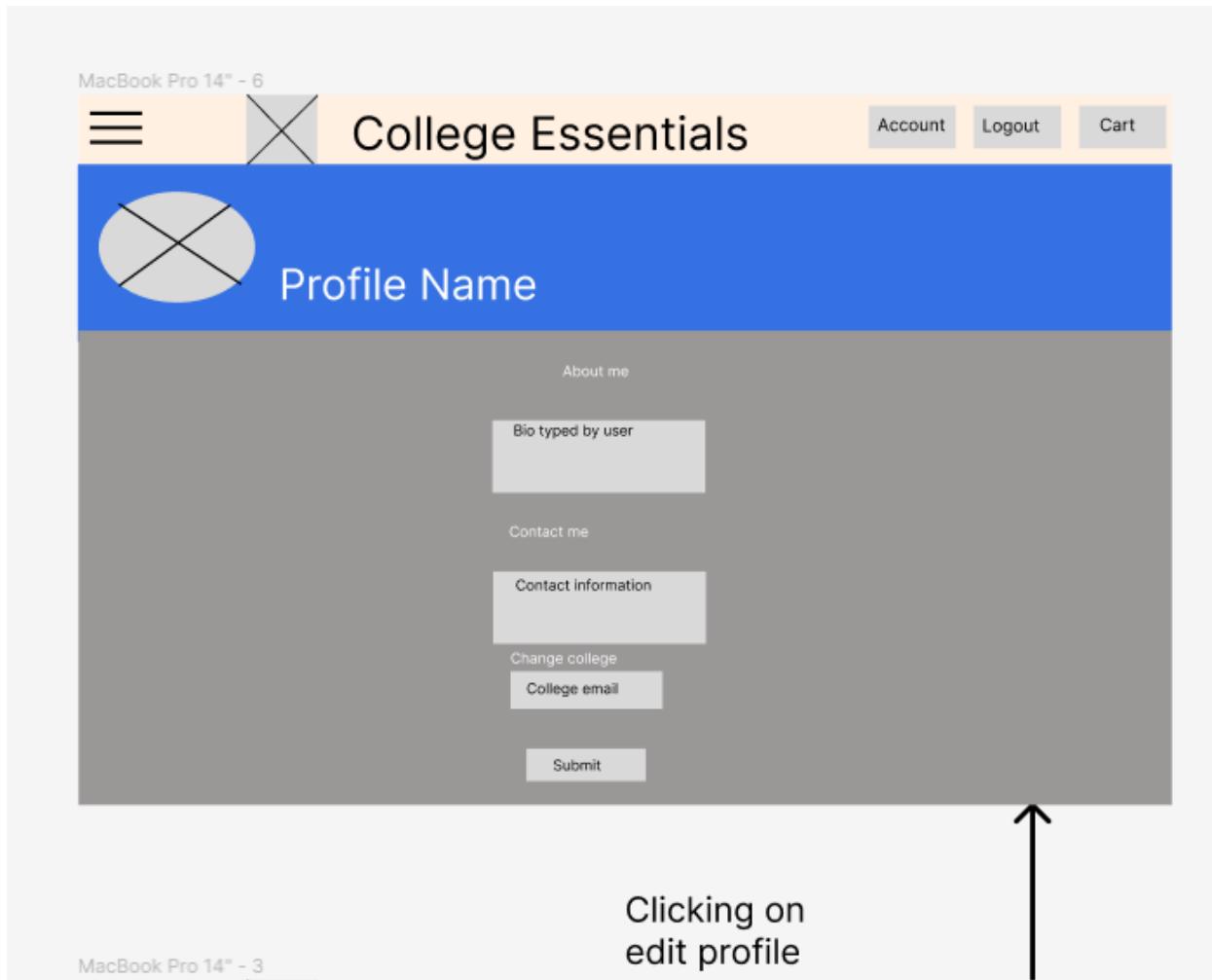
- a. A marketplace listing shall have the location of where the deal is going to be held.
- b. A marketplace listing shall have an expiration date.
- c. A marketplace listing shall have a bidding minimum.
- d. A marketplace listing shall have optional tags.
- e. A marketplace listing shall have the option to be returned or not.

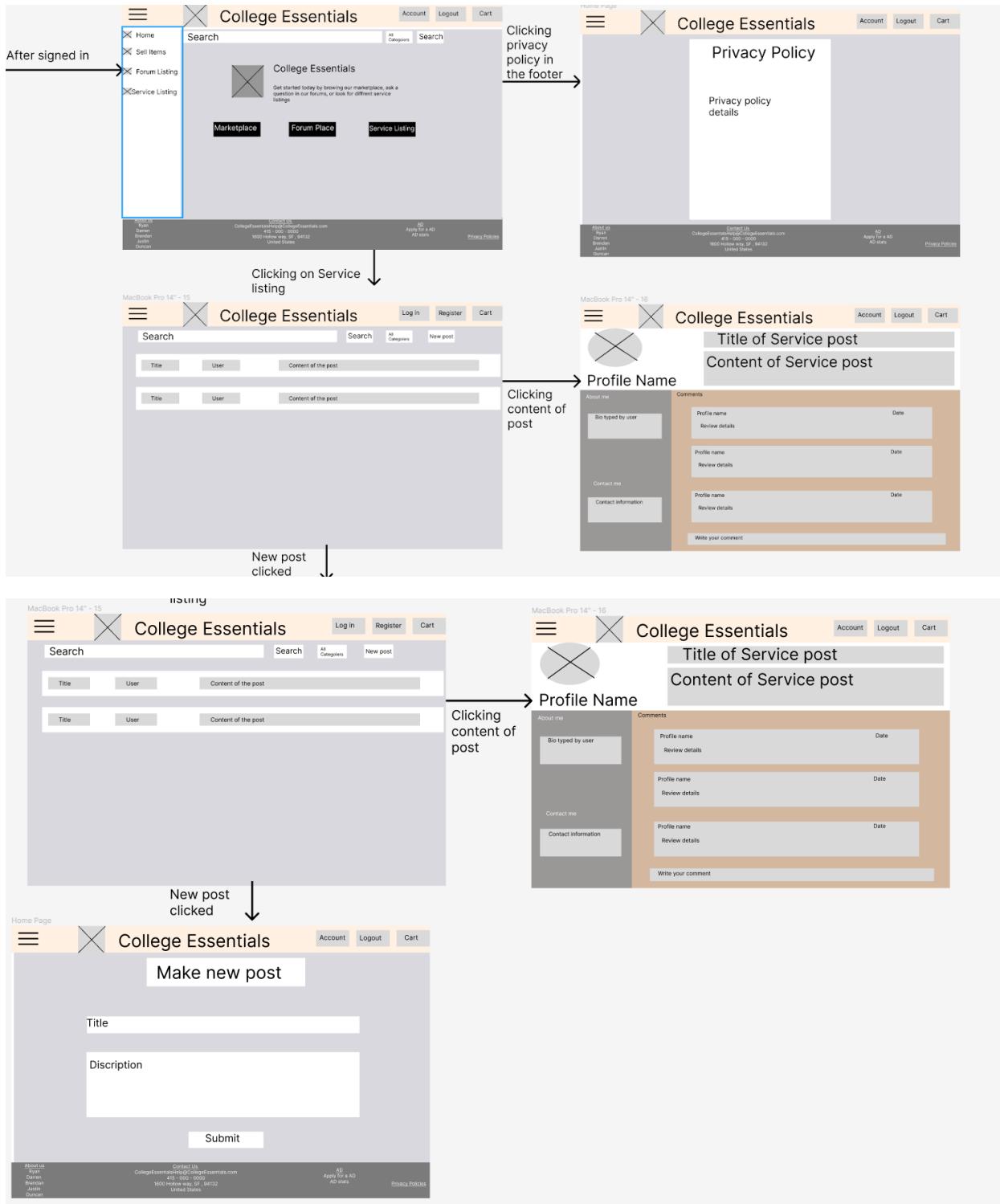
**Priority 3:**

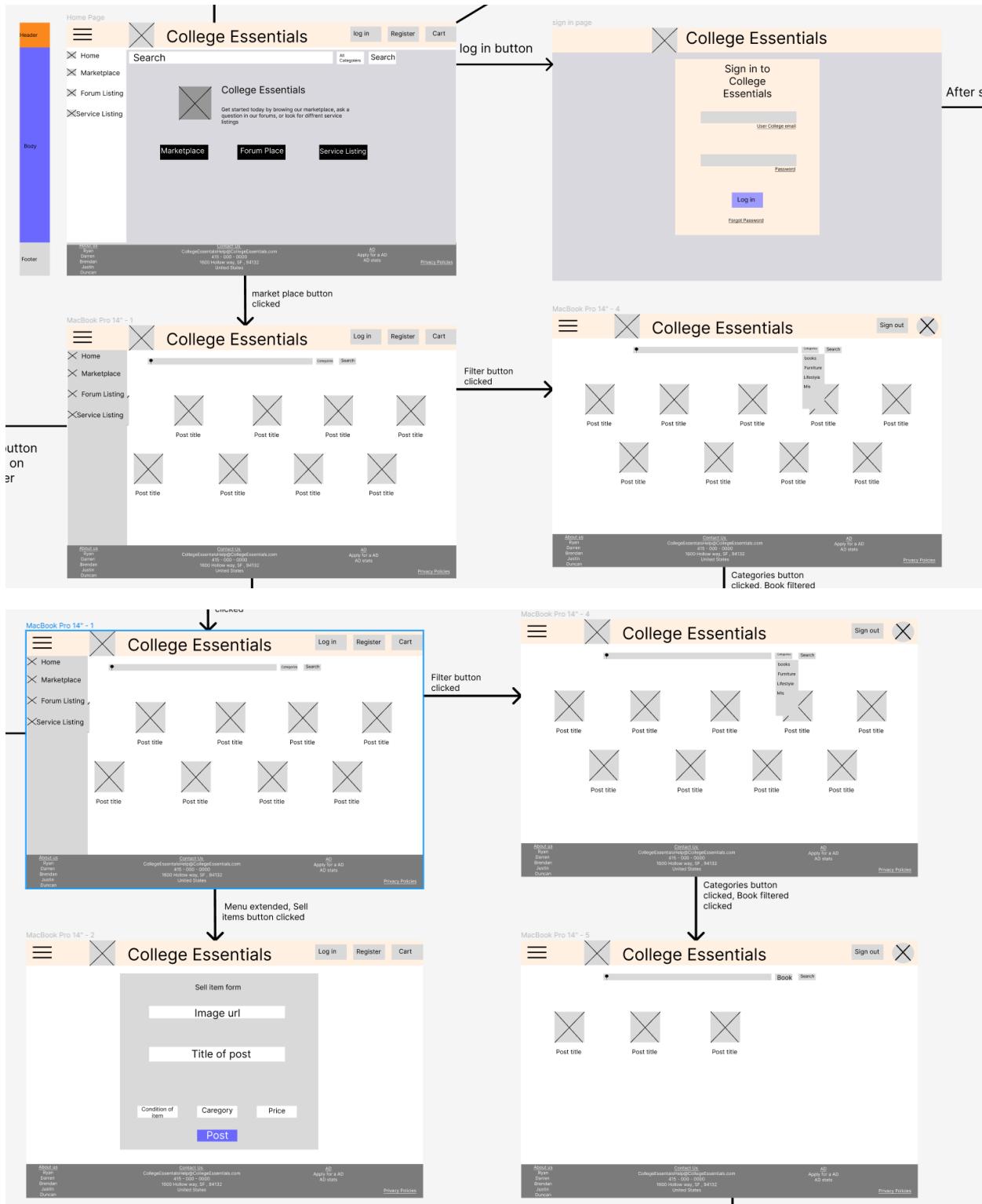
1. Shopping Cart
  - a. A shopping cart will have a little number showing how many items are within it, while you are not on the checkout screen.
2. Checkout
  - a. Checkout shall prompt users to create an account if not yet registered.

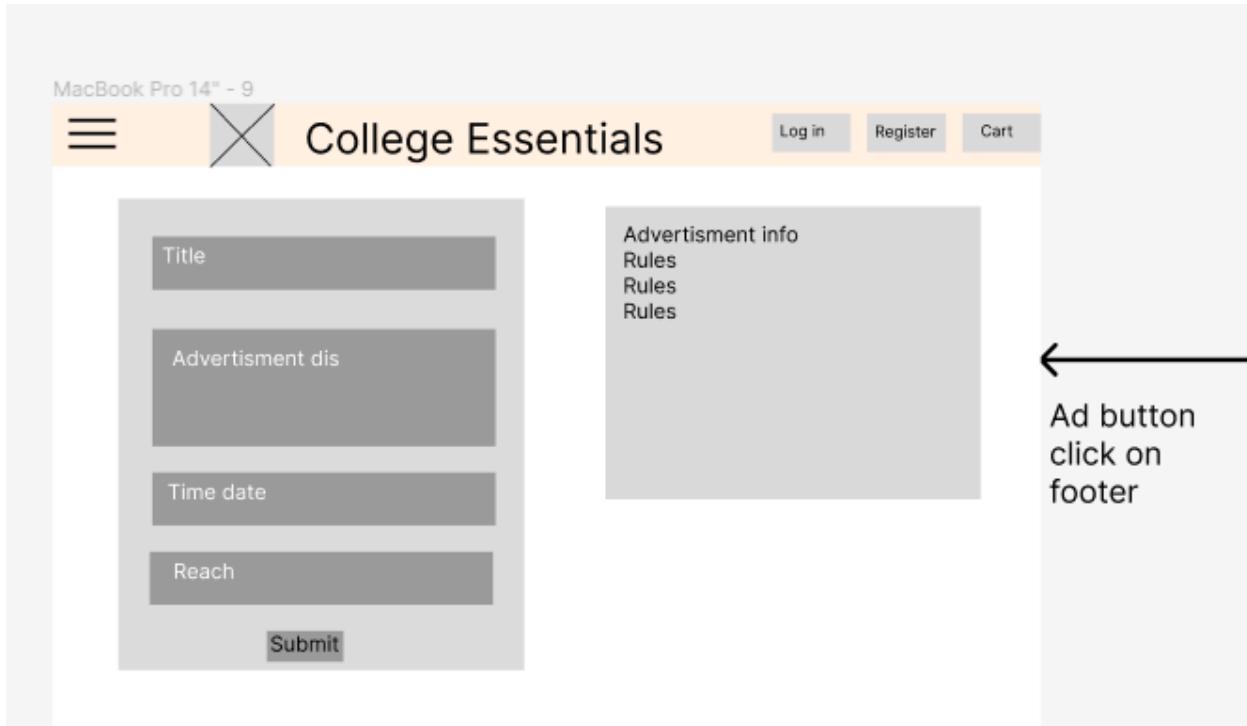
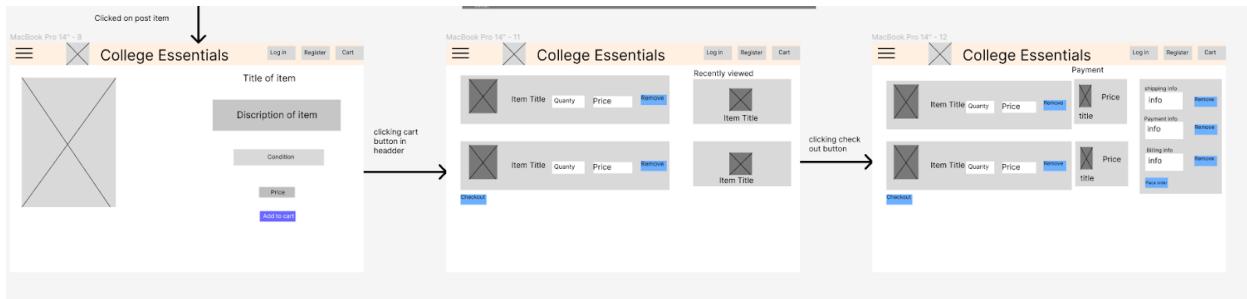
## [Section III] Wireframes Based on Mockups/Storyboards











MacBook Pro 14" - 10

College Essentials

Title of forum post

Content of forum post

Profile Name

About me

Bio typed by user

Contact me

Contact information

Comments

Profile name

Review details

Date

Profile name

Review details

Date

Profile name

Review details

Date

Write your comment

MacBook Pro 14" - 7

Home Page

College Essentials

Account Logout Cart

Make new post

Title

Description

Submit

New post clicked

Menu extended, Forum listing button clicked

Home Page

Search

Title User Content of the post

Title User Content of the post

Log in Register Cart

Categories New post

About us

Ryan

Dave

Brendan

John

Danica

Contact Us

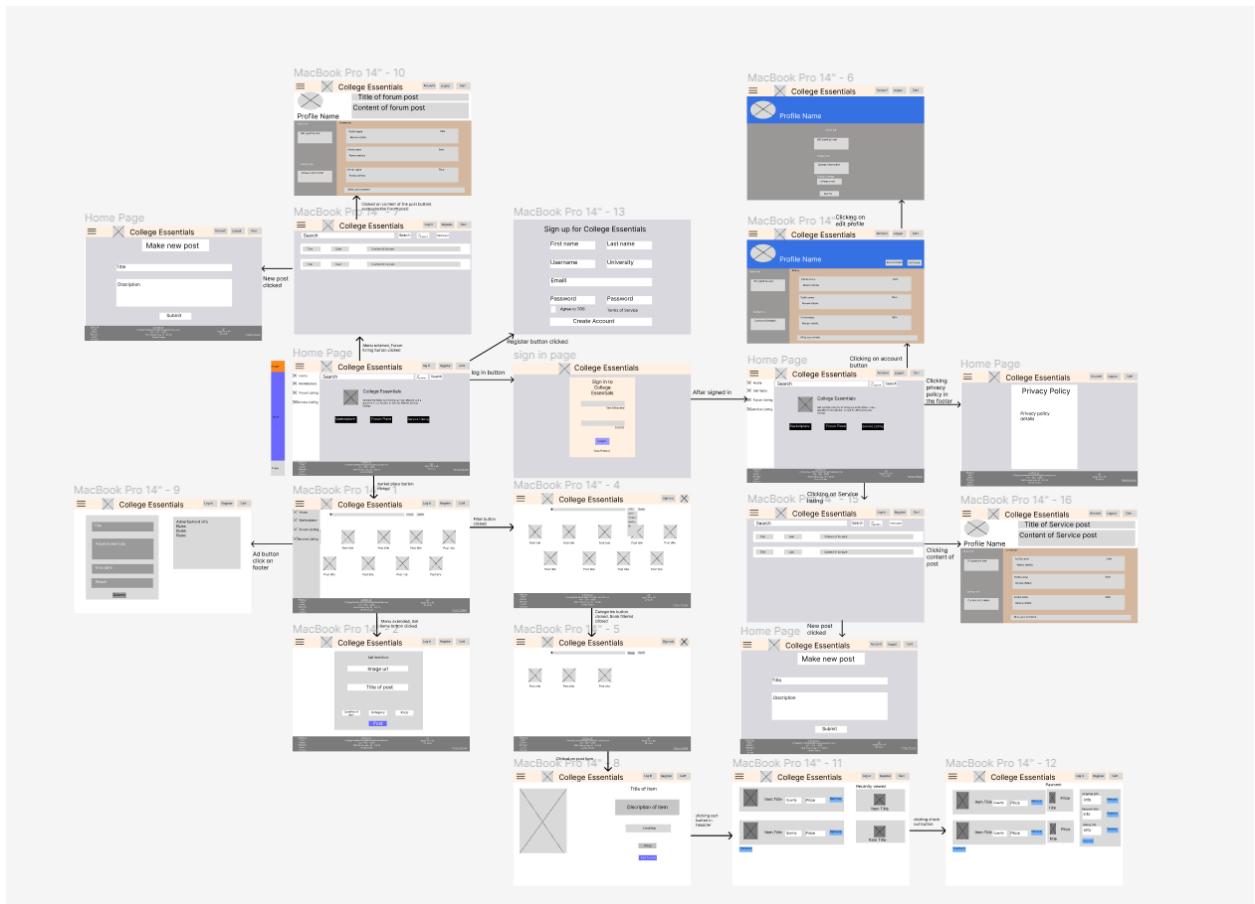
CollegeEssentialsCollegeEssentials.com

1600 Nellis Way, Ste. 14132

United States

50 AD items

Privacy Policies



## Link to all the diagrams:

<https://www.figma.com/file/nq07WHQeXEGWxPcPPLDjI7/Untitled?node-id=0%3A1>

## [Section IV] High Level Database Architecture and Organization

### 1. Business Rules

- a. GeneralUser
  - i. A GeneralUser shall be able to log into a registered account from many devices. (RegisteredUser)
  - ii. A GeneralUser shall have a ShoppingCart.
  - iii. A GeneralUser shall be able to checkout their purchases.
  - iv. A GeneralUser shall be able to view the marketplace.
  - v. A GeneralUser shall be able to view the service listings.
  - vi. A GeneralUser shall be able to view the forum listings.
  - vii. A GeneralUser shall receive many receipts for their transactions.
  
- b. RegisteredUser
  - i. A RegisteredUser shall be able to be an Admin
  - ii. A RegisteredUser shall have one or many Colleges associated with their accounts.
  - iii. A RegisteredUser shall be able to post zero or many ServiceListings.
  - iv. A RegisteredUser shall be able to post zero or many ServiceReplies
  - v. A RegisteredUser shall be able to post zero or many ForumPosts
  - vi. A RegisteredUser shall be able to post zero or many ForumReplies
  - vii. A RegisteredUser shall have zero or many Ratings.
  - viii. A RegisteredUser shall be able to post zero or many Ratings.

- ix. A RegisteredUser shall have many forms of Payment
- c. Rating
  - i. A Rating shall belong to one RegisteredUser.
  - ii. A Rating shall be posted by zero or many RegisteredUsers.
  - iii. A Rating shall be tied to zero or one Transaction
- d. Admin
  - i. An Admin shall be considered a RegisteredUser.
- e. College
  - i. A College shall be associated by many RegisteredUsers (e.g. college student or college faculty)
- f. MarketplaceListing
  - i. A Marketplace Listing shall be listed by one RegisteredUser
  - ii. A Marketplace Listing shall list at least one Product
- g. ServiceListing
  - i. A ServiceListing shall be listed by one RegisteredUser.
  - ii. A ServiceListing shall be followed by many ServiceReplies
- h. ServiceReply
  - i. A ServiceReply will be tied to many ServiceListings
  - ii. A ServiceReply is posted by a RegisteredUser
- i. Transaction

- i. A Transaction shall have many Receipts.
  - ii. A Transaction shall be linked to one Payment method.
  - iii. A Transaction shall be linked to a Buyer
  - iv. A Transaction shall be linked to a Seller
  - v. A Transaction shall be linked to a Rating
  - vi. A Transaction shall be linked to many TransactionItems
- 
- j. TransactionItems
    - i. A TransactionItems is tied to a Transaction.
    - ii. A TransactionItems is tied to a Product
- 
- k. ShoppingCart
    - i. A ShoppingCart is tied to one GeneralUser.
    - ii. A ShoppingCart is linked to one Transaction.
- 
- l. Payment
    - i. A Payment shall belong to one RegisteredUser.
    - ii. A Payment shall be linked to a Transaction.
- 
- m. Receipt
    - i. A Receipt shall be given to one RegisteredUser (buyer).
    - ii. A Receipt shall be given to one RegisteredUser (seller).
    - iii. A Receipt shall belong to one Transaction.
- 
- n. Product
    - i. A Product is ordered by zero or many GeneralUsers
    - ii. A Product is added to many ShoppingCarts

- iii. A Product is listed on a MarketplaceListing
  - iv. A Product is tied to many TransactionItems
- 
- o. ForumPost
    - i. A ForumPost is posted by one RegisteredUser
    - ii. A ForumPost shall be followed by many ForumReplies
- 
- p. ForumReply
    - i. A ForumReply will be tied to many ForumPost
    - ii. A ForumReply is posted by a RegisteredUser

## 2. Entities

- a. GeneralUser (Strong)
  - i. General\_user\_id: key, numeric; This is the unique id registered to each GeneralUser
  - ii. ip\_address: numeric; This is the IP address associated with a GeneralUser
  
- b. RegisteredUser (Weak)
  - i. registered\_user\_id: key, numeric; This is the unique id registered to each RegisteredUser.
  - ii. general\_user\_id: key, numeric; This is the id associated to the generalUser tied to RegisteredUser
  - iii. first\_name: alphanumeric; This is the first name registered to a RegisteredUser.
  - iv. last\_name: alphanumeric; This is the last name registered to a RegisteredUser

- v. password: alphanumeric; This is the unique password registered to each RegisteredUser.
- vi. email: alphanumeric; This is the unique username registered to each RegisteredUser.
- vii. about: alphanumeric; This is the about info tied to a RegisteredUser
- viii. contact: alphanumeric; This is the contact info tied to a RegisteredUser
- ix. is\_banned: numeric; This is a 0 or 1 value that denotes if the RegisteredUser is banned or not
- x. pfp: alphanumeric; This is the image link tied to a RegisteredUser's profile picture

c. Ratings (Weak)

- i. rating\_id: key, numeric; This is the unique id given to the specific rating.
- ii. rating: numeric; This is a value between 0 and 5 given to the rating.
- iii. description: alphanumeric; This is a short description of the given rating.
- iv. transaction\_id: key, numeric; This is the transaction linked to the rating.
- v. registered\_user\_sender\_id: key, numeric; This is the ID of the user giving/sending a rating
- vi. registered\_user\_receiver\_id: This is the ID of the user receiving the rating
- vii. posted\_at: timestamp; This is the timestamp of when the rating

was posted

d. Admin (Weak)

- i. admin\_id: key, numeric; This is the unique id linked to an Admin account.
- ii. registered\_user\_id: key, numeric; This is the RegisteredUser id associated with an Admin

e. College (Strong)

- i. college\_id: key, numeric; This is the unique college id associated with each college
- ii. name: alphanumeric; This is the name for a college

f. MarketplaceListing (Strong)

- i. marketplace\_listing\_id: key, numeric; This is the unique id linked with the MarketplaceListing.
- ii. title: alphanumeric; This is the title given to the MarketplaceListing.
- iii. body: alphanumeric; This is the body given to the MarketplaceListing.
- iv. created\_at: timestamp; This is the timestamp given for when a marketplace listed was posted
- v. seller\_id: key, numeric; This is the unique seller id used in a transaction.
- vi. price: numeric; This is the price associated with a MarketplaceListing
- vii. quantity: alphanumeric; This is the quantity of the product

listed

g. ServiceListing (Strong)

- i. service\_listing\_id: key, numeric; This is the unique id linked with the ServiceListing.
- ii. title: alphanumeric; This is the title given to the Service Listing.
- iii. description: alphanumeric; This is the description given to the ServiceListing.
- iv. created\_at: timestamp; This is the timestamp given to the ServiceListing.

h. ServiceReply (Weak)

- i. service\_reply\_id: key, numeric; This is the unique id linked with the ServiceReply.
- ii. description: alphanumeric; This is the description given to the ServiceReply
- iii. posted\_at: timestamp; This is when the ServiceReply was posted

i. Transaction (Strong)

- i. transaction\_id: key, numeric; This is the unique id of the transaction.
- ii. completed\_at: timestamp; This is the timestamp of when a transaction is completed
- iii. payment\_id: key, numeric; This is the unique id of a payment
- iv. seller\_id: key, numeric; This is the unique id associated with the seller (RegisteredUser) in the transaction

- v. buyer\_id: This is the unique id associated with the buyer (RegisteredUser) in the transaction
- j. TransactionItems (Weak)
  - i. Transaction\_id: numeric; This is the transaction id associated with a Transaction
  - ii. product\_id: numeric; This is the product id associated with a TransactionItem
  - iii. title: alphanumeric; This is the title associated with a product
  - iv. image: alphanumeric; This is the image associated with a product
  - v. Price: alphanumeric; This is the price associated with a product
  - vi. quantity: alphanumeric; This is the quantity associated with the product
- k. ShoppingCart (Strong)
  - i. shopping\_cart\_id: key, numeric; This is the unique id for a ShoppingCart
  - ii. buyer\_id: key, numeric; This is the unique id for a Buyer (GeneralUser)
- l. Receipt (Strong)
  - i. receipt\_id: key, numeric, This is the unique id of the receipt given in the transaction.
  - ii. seller\_id: key, numeric; This is the unique id of the seller in the transaction.

iii. buyer\_id: key, numeric; This is the unique id buyer in the transaction.

iv. transaction\_id: key, numeric; This is the unique id of the transaction.

m. Payment (Strong)

i. Payment\_id: key, numeric; This is the unique id given to the payment.

ii. payment\_type: alphanumeric; This is the type of payment used by the buyer.

iii. card\_number: numeric; This is the number associated with a card

n. Product (Strong)

i. product\_id: key, numeric; This is the unique id of the product

ii. general\_type: alphanumeric; This is the general type of the product (e.g. a textbook, bicycle, furniture, etc.)

iii. picture: alphanumeric; This the picture link associated with the product

iv. Condition: alphanumeric; This is the condition associated with a product

o. ForumPost (Strong)

i. forum\_id: key, numeric; This is the unique id of a ForumPost

ii. title: alphanumeric; This is the title of a ForumPost

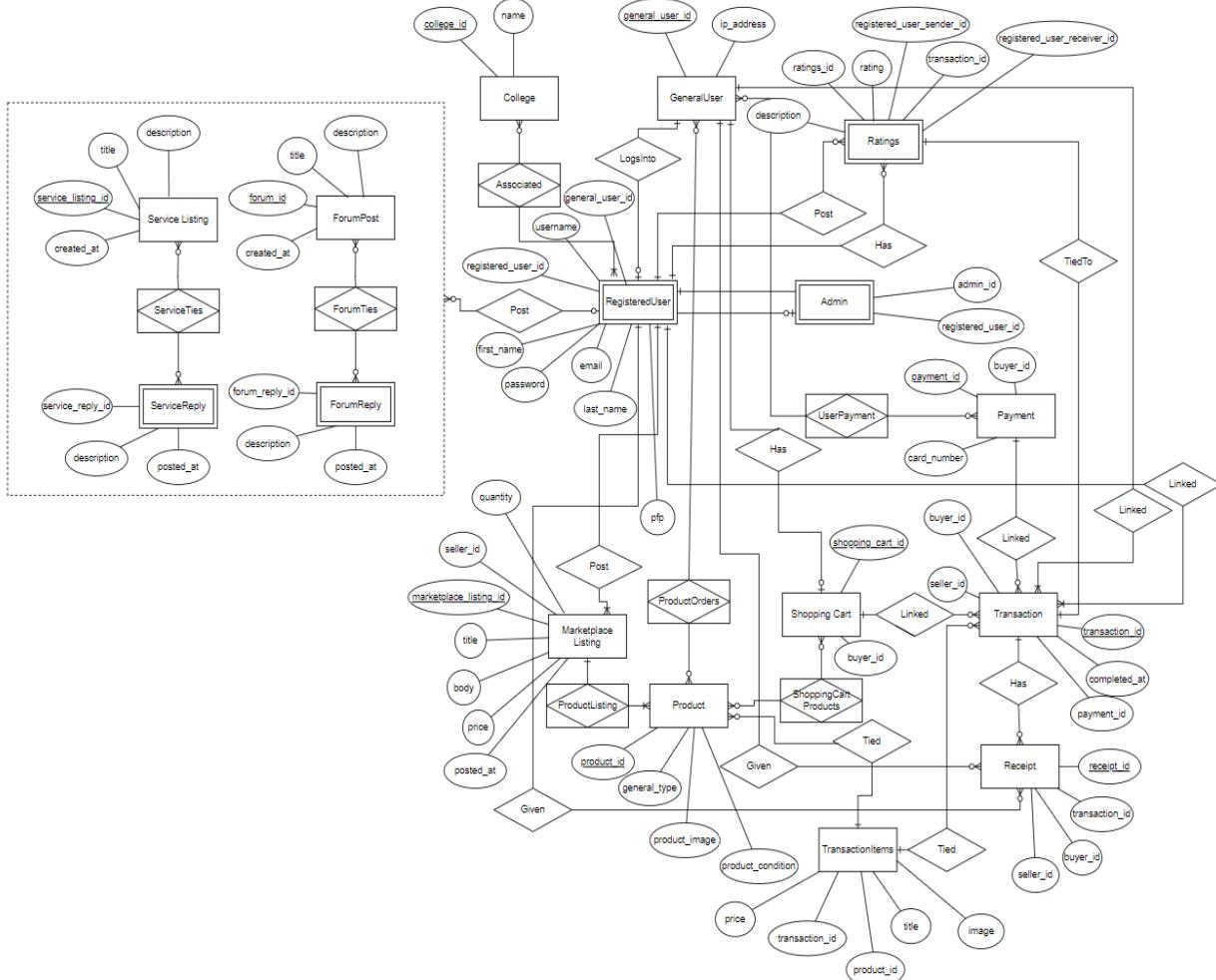
iii. description: alphanumeric; This is the description of a ForumPost

iv. `created_at`: timestamp; This is when a ForumPost was created

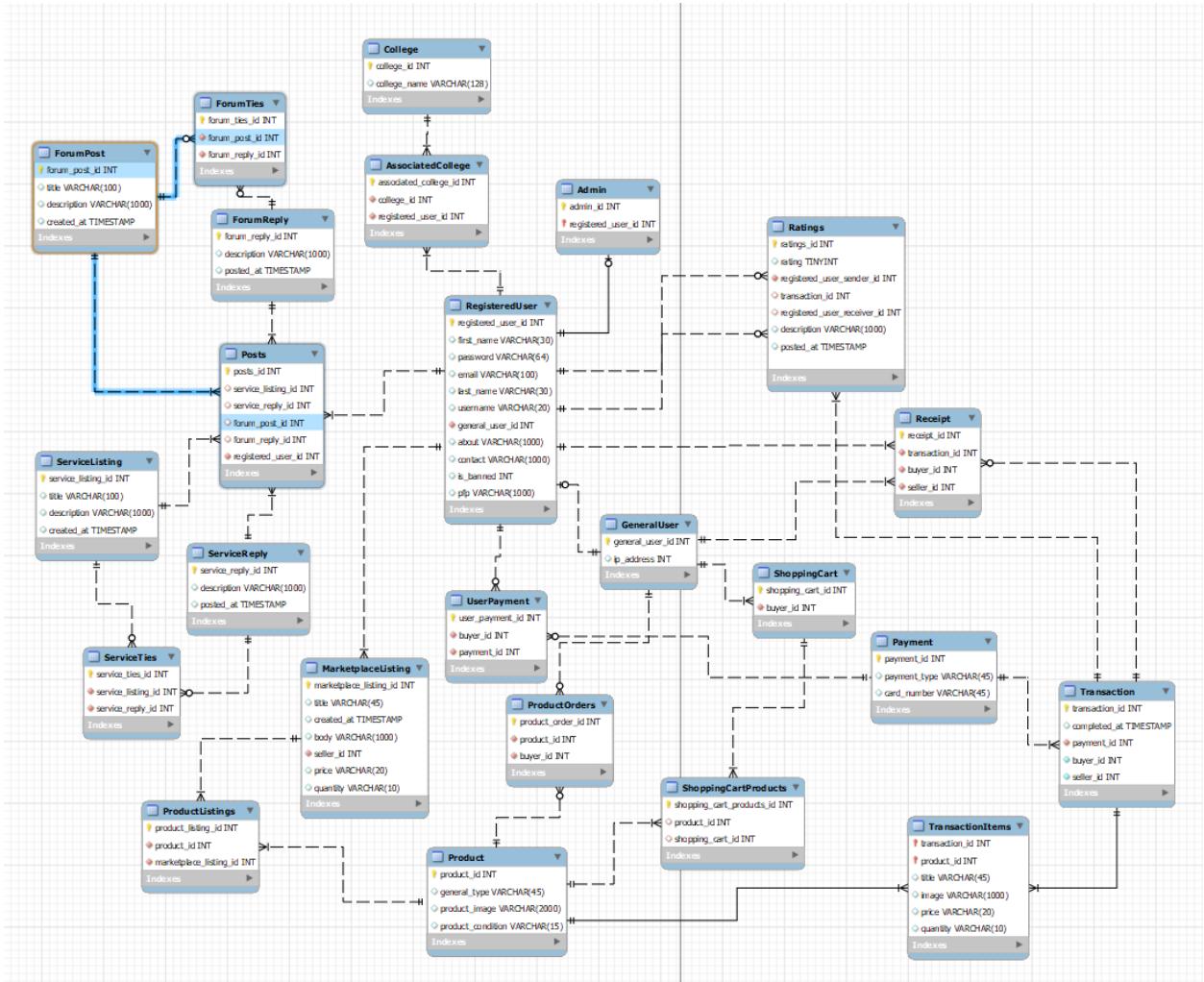
p. ForumReply (Weak)

- i. `forum_reply_id`: key, numeric; This is the unique id of a ForumReply
- ii. `description`: alphanumeric; This is the description of a ForumReply
- iii. `posted_at`: timestamp; This is when a ForumReply was posted

### 3. Entity Relationship Diagram (ERD)



## 4. Database Model (EER)



## 5. DBMS

College Essentials will use MySQL as its DBMS, given its widely compatible and easy to use nature. MySQL has many tools available, such as MySQL Workbench, and continues to grow alongside cloud data platforms.

## 6. Media Storage

The College Essentials database will store images in links within the Product

table, but the image and files themselves will be stored within the project's file system. This will allow us to keep the files in a consistent location, and allow us to easily be able to locate and reference these files. Because all files will be in the file system in a possible "uploads" folder, we would be able to just implement a path in the SQL database to be able to point to the file.

## 7. Search/Filter Architecture and Implementation

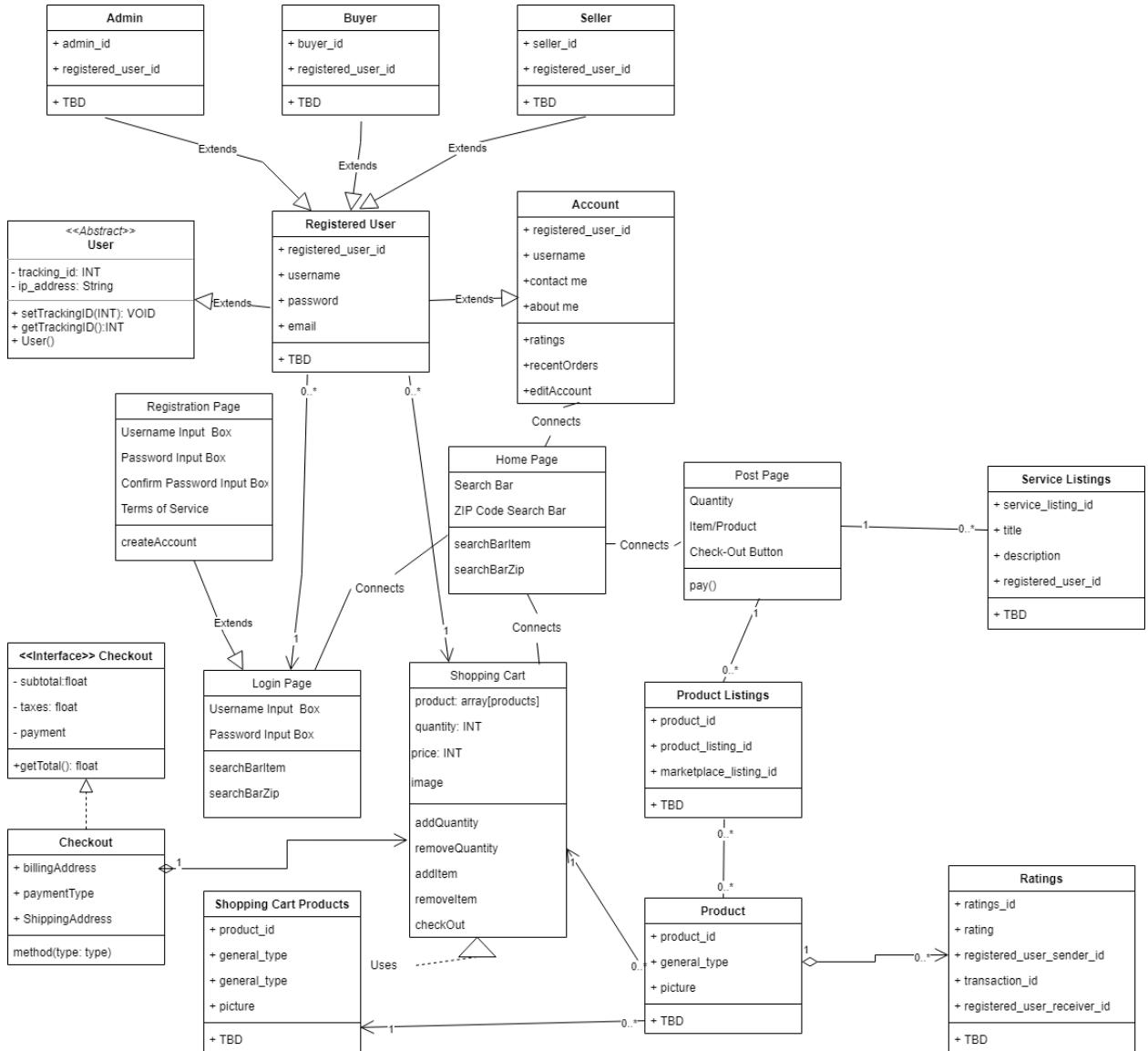
- a. Our search algorithm will allow the user to use a search bar for terms within a marketplace listing. There will also be a drop down menu for specifying the condition of a product that will work in tandem with the search bar. At this time, we have filters that allow the user to search based on condition, but we plan to allow users to search based on college as well.
- b. In order to implement the search filters, we will have a table in the database that would show the specific keywords linked to each marketplace listing. Here is an example of how the search would be implemented based on no filters, using SQL LIKE to search for the keyword within the title of the marketplace listings:

```
SELECT MarketplaceListing.title, Product.product_image  
FROM ProductListings  
JOIN MarketplaceListing ON  
ProductListings.marketplace_listing_id =  
MarketplaceListing.marketplace_listing_id  
JOIN Product ON ProductListings.product_id =  
Product.product_id  
WHERE MarketplaceListing.title LIKE '%tes%';
```

- c. When searching for products using a condition filter, we will use a similar SQL LIKE to search for the specific keyword given, however, the product condition will be added using the drop down menu with the SQL AND function. A Query example would be:

```
SELECT MarketplaceListing.title, Product.product_image,  
Product.condition FROM ProductListings  
JOIN MarketplaceListing ON  
ProductListings.marketplace_listing_id =  
MarketplaceListing.marketplace_listing_id  
JOIN Product ON ProductListings.product_id =  
Product.product_id  
WHERE MarketplaceListing.title LIKE '%searchterm%' AND  
Product.condition = "NEW";
```

## [Section V] High Level Diagrams



## **[Section VI] List of Contributions**

### **I. Ryan**

- A. Facilitated group meetings and plans
- B. Worked on categorizing the priorities with the group
- C. Worked on frontend for the horizontal prototype
  - 1. Worked on the CSS for the login, register page, PostItem, SellItem, Home Page, and item View
  - 2. Worked on the CSS for the footer along with the java
  - 3. Worked on fixing Home backend, itemView, login, registration
- D. Worked on backend for the vertical/prototype
  - 1. Worked to refactor putting marketplace listing products into the remote database
  - 2. Worked to fix the search features for Marketplace listings (back-end)

### **II. Duncan**

- A. Created all the wireframes drawings for Milestone 3.
- B. Revised all High level UI Mock up drawings in Milestone 2.
- C. Revised use cases that were given in Milestone 1.
- D. Created the UI design of the Front end of each page (not the coding).

### **III. Justin**

- A. Worked on categorizing the priorities with the group.
- B. Revised the functional requirements on Milestone 3.
- C. Revised the data definitions for Milestone 3.
- D. Revised the UML diagram for Milestone 3.
- E. Worked on the frontend for the horizontal prototype.

1. Designed and wrote the code for the UI for the Shopping Cart and Checkout pages.
2. Worked on the routing for the Advertisements, Privacy Policies, and Term of Service pages.
3. Created the UI for the Advertisement page, designing an easy forum for users to fill out.
4. Added CSS for the Shopping Cart and Checkout pages.

#### **IV. Brendan**

- A. Assisted in altering CSS for various part of the site
  1. Login CSS assistance
  2. Home page search bar CSS assistance
- B. Worked on routing related to the forum and service listing
  1. Routing to proper external links
  2. Routing to proper dummy user pages mentioned in service and forum listing
- C. Created dummy user account pages for proper routing.
- D. Worked on UI for the pages of forum and service listing
  1. Implemented grid design to forum pages for a for fulfilling design
  2. A smoother design for forum and service listing for easy reading, and proper design to fit with background

## V. Darren

- A. Worked on categorizing the priorities with the group and reworked the documentation on the database (section IV)
- B. Worked on revising the database (on MySQL and AWS) based on the feedback from M2V1
- C. Worked on frontend for the horizontal prototype
  1. Made the home page more user friendly
  2. Made the account the page
  3. Added css to various details in the website, including the footer, buttons, and menu
- D. Worked on backend for the vertical/prototype
  1. Worked to refactor putting marketplace listing products into the remote database
  2. Worked to fix the search features for Marketplace listings (back-end)

## **[Appendix I] Milestone 3 Review: Project Status and UI Review**

### **1. UI and Functionality Feedback**

- a. Home screen has a good hook
- b. Website is relatively user friendly and intuitive
  - i. e.g. Sell Item leads to a form to post items to sell
  - ii. e.g. Forum Listings lead to a centralized hub for forum posts
  - iii. e.g. Service Listings lead to a centralized hub for service listings
  - iv. Most buttons and entry fields are responsive
- c. Input fields should be validated
- d. Too much blank space for ForumPost description and ServicePost listing
  - i. Needs more dynamic sizing
- e. Home screen doesn't utilize space well, needs better alignment and spacing
  - i. Needs images or something to display what each button represents (something for marketplace, another for forum listing, and another for service listing)
- f. Website needs a logo instead of plain text "College Essentials"

### **2. Brief Review of Coding, Github, Database, etc.**

- a. NOTE: CTO did not review code, Github, Database during the meeting. This is based on feedback from the meeting about our practices in general along with feedback from the milestone doc in regards to the database
- b. Input fields must be validated to ensure they are not empty or pass in illogical values (e.g. negative values for quantity)

- c. Database is done relatively well, “Relationships are beautifully implemented as per database requirements “ (per M2 feedback too)

### **3. Project Status**

- a. Teamwork:
  - i. The team dynamic has been overall positive so far. Some of the problems that have arisen in regards to team dynamic is the amount of work that some teammates have compared to others. As a team overall, we should try to abide by timelines better, and communicate with the group the progress on our individual parts. Overall, we are working well together as a team. We are able to keep each other accountable, and make sure that even if we are late on our deadlines, we are still getting the work done in the end.
- b. Risks:
  - i. As a team, our main risk is that we are attempting to have more functional and nonfunctional requirements than what we should have attempted. Because our group is smaller, each person has to contribute more work than a group with more members would have to. That means that we will inevitably fail to meet some functional and nonfunctional requirements. To mitigate these issues we have been pushing back some functional requirements to priority 2, and focusing on being able to deliver all of our priority 1 requirements.
- c. Coding Practices:

- i. Our team's coding style has solidified itself to be an eXtreme Programming style. We demonstrated many if not all components of XP such that we had come to the conclusion that this coding style is what fit our situation the best when compared to the other styles. These components include communication, simplicity, feedback, courage, and respect. First being communication which our team has an abundance of. We had plenty of meetings, which albeit had no webcam still had plenty of communications. The communication did not stop there as we had plenty outside of voice chats as well, through things like chatting by text. Overall, it helped keep us on track as to what tasks were at hand. Second feature we practiced was simplicity. We all thought it best that we stuck with what was best for our skills. It worked off our specialties such that we could finish our work in an efficient manner. We also tend to not work towards future prospects, rather we worked with what we were comfortable with to make a simple working project which later we can build upon. Third feature we practiced was feedback, which we were definitely not lacking in. If we found ourselves dissatisfied with a certain feature we would not be afraid to voice our opinions on the subject, and provide feedback. It helped refine our product to what we thought it should be. Which leads me to the fourth we practiced, courage. None of us were shy of courageousness, and were not afraid to voice any of our opinions. Through those opinions we were not afraid of change either, and were willing to change any of our code in a heartbeat if it meant a better project. Last but not least

is the respect aspect. No one on the team lacked a sense of respect for the other person. This builds upon courage and feedback because we never put each other down. We always respected each other's time, skill, and availability which enabled courage, and feedback. Overall, all of these components helped build up to how we got our coding style, which in the end we concluded fit the most into extreme programming.

d. Usage of Proper SE Code Management Practices:

- i. For code submissions, we work on individual parts that need to be done, and test them with the rest of the application to make sure that everything still works. The workflow is done by assigning each individual member a section to work on, then having all the other team members available to review the code before pushing it to the development branch on github. This process repeats for all team members and any work that they have to do. After all the work is done for a section, we review the working prototype, and finally push to master and deploy into AWS.

e. How did you address Site Security and Safe Coding Practices

- i. For site submissions, passwords have the ability to be shown or hidden on registration and login. Passwords are also hashed and stored into the database instead of plain text. Insert statements to the remote DB and protected from SQL injections using VALUES(?, ?, ..) etc.

f. Digital Content:

- i. All images that we use for the application use external urls that are saved in the database. That means that the database only saves the url of the image, and does not have to store the image itself. Any page that needs the images pulls the url from the database to populate the pages. Any formats or resolutions are allowed, but the resolutions are chosen by the website.

## **4. List of P1 Features Committed for Delivery**

- a. General User
  - i. A general user shall be able to browse forum posts.
  - ii. A general user shall be able to browse items on the marketplace.
  - iii. A general user shall be able to browse service listings.
  - iv. A general user shall be able to search for their needed service.
  - v. A general user shall accept our terms of conditions.
  - vi. A general user shall be able to search by item condition in the marketplace.
  
- b. Registered User
  - i. A registered user shall be able to log in with their username.
  - ii. A registered user shall be able to review another user.
  - iii. A registered user shall be able to create their own profile with text.
  - iv. A registered user shall be able to create their own profile with pictures.
  - v. A registered user shall be able to edit their own profile with text.
  - vi. A registered user shall be able to edit their own profile with pictures.
  - vii. A registered user shall be able to post a picture as their profile picture.
  - viii. A registered user shall be able to post a service listing.

- ix. A registered user shall be able to list items for sale in the marketplace.
- x. A registered user shall be able to post on the forums.
- xi. A registered user shall be able to log out.
- xii. A registered user shall be able to rate other general users.

c. Service Listing

- i. A service listing shall have a title.
- ii. A service listing post shall have characters in its content body.
- iii. A service listing post shall have basic information like time posted.
- iv. A service listing post shall be able to be unlisted.

d. Forum Listing

- i. A Forum listing shall have a title.
- ii. A Forum listing post shall have characters in its content body.
- iii. A Forum listing post shall have basic information like time posted.
- iv. A Forum listing shall be visible to all general users.
- v. A Forum listing post shall be able to be unlisted.

e. Shopping Cart

- i. A shopping cart shall remember what has been stored within it.
- ii. A shopping cart shall remember how many of one entity have been stored within it.
- iii. A shopping cart shall allow users to edit products within it.
- iv. A shopping cart shall allow for check out.

- v. A shopping cart shall give descriptions of products stored within it

- f. Checkout

- i. Checkout shall allow a user to input their payment information.
- ii. Checkout shall allow a user to input their shipping address.
- iii. Checkout shall allow a user to input their billing address.
- iv. Checkout shall allow users to review their purchase on last time.

- g. Navigation Bar

- i. A navigation bar shall have a section for students selling their products.
- ii. A navigation bar shall have a section for teachers to look for student assistants.
- iii. A navigation bar shall have a section for the student forums.
- iv. A navigation bar shall have a section for the about us page.

- h. Marketplace Listing (Buy/Sell Listing)

- i. A marketplace listing shall have a title.
- ii. A marketplace listing shall have a description.
- iii. A marketplace listing shall have a photo (or many).
- iv. A marketplace listing shall have a price.
- v. A marketplace listing shall be able to be unlisted.
- vi. A marketplace listing shall have filters to categorize.

- i. Ratings

- i. User ratings shall rate a profile on a scale of 1 to 5.
- ii. User ratings shall be visible on a registered user's profile.
- iii. User ratings shall be visible on a registered user's marketplace listing.
- iv. User ratings shall be visible on a registered user's service listing.
- v. User ratings shall be able to be posted by registered users.

## [Appendix II] Milestone 3 Review: SW Review

- Git/Github organization (e.g. organization of branches)
  - Our group has only been pushing to development, and then pushing to master only at the end of each milestone. We can keep working in the development branch, but we need to push to master more often, such as once a week.
- Git/Github usage
  - Our github usage is in general good. We have an even number of submissions between the team members, and we post to github very often.
- Code documented (header, in code) with good coding style
  - Our code follows a consistent coding style that all of us have adapted since milestone 2. We based our coding style off of one another, where the comments are in line with the code.
- MVC/OO patterns followed up
  - In terms of MVC, all pages use a MVC pattern where the user sees the original page, and then updates the page based on their inputs. After that, the new updated page is generated using the information from the database, and then is sent back to the user on the new page.
- Frameworks (back end front end) deployed correctly
  - The back end and front end are both deployed correctly to AWS and the application. They both work together when getting a page, and testing all functionality of the page.
- Database organization (tables, naming...)
  - Our database is well organized, where all tables that are needed are named similarly to their respective pages.

- Blobs being used? If so, is it working?
  - Our project does not use any blobs.
- Adherence to best practices of security (PW encrypted, search inputs verified etc.)
  - Passwords are hashed and stored in the database instead of plain text
  - Input fields, such as fields in Registration and postItem are validated to ensure that the fields are not empty, and contain logical values if applicable (e.g. quantity value must be above one, a price shall not be negative, etc.)
- Efficiency (proper use of image thumbnails, efficient search etc.)
  - Images are stored as links within our database, and are posted on the website using the image link
  - Search throughout our project utilizes the SQL keyword LIKE ‘keyword’
    - Searching for marketplace listings has an optional categories tag based on the product being listed
    - Searching for forum and service listings has an optional dropdown menu for searching with all (title and description), just title, or just description

# **SW Engineering CSC648/848 Summer 2022**

## **College Essentials**

### **Team 4**

| <b>Student Name</b> | <b>Student Email</b>   | <b>Github Username</b> | <b>Student ID</b> | <b>Student Roles</b>       |
|---------------------|------------------------|------------------------|-------------------|----------------------------|
| Ryan Murphy         | rmurphy4@mail.sfsu.edu | deadlepricon           | 921366560         | Team Lead,<br>Backend Lead |
| Duncan Forniss      | Dforniss@mail.sfsu.edu | DunkeyFunkey           | 917678507         | Frontend Lead              |
| Wee Yang Leong      | wleong@mail.sfsu.edu   | ktleong2000            | 918614936         | Backend Assistant          |
| Darren Wong         | dwong19@mail.sfsu.edu  | dawong025              | 918565679         | Database Master            |
| Justin Wong         | jwong66@mail.sfsu.edu  | jwong66                | 918516578         | Github Master              |

**Milestone:** Milestone 4

**Date:** 07/28/22

## **Version History**

| <b>Milestone</b> | <b>Date</b> |
|------------------|-------------|
| M4V1             | 07/28/2022  |
| M3V2             | 07/28/2022  |
| M3V1             | 07/19/2022  |
| M2V2             | 07/19/2022  |
| M2V1             | 07/06/2022  |
| M1V2             | 07/01/2022  |
| M1V1             | 06/21/2022  |

## **Table of Contents**

|  |           |
|--|-----------|
| <b>[Section I] Product Summary</b>   | <b>4</b>  |
| <b>[Section II] Usability Test Plan</b>                                    | <b>8</b>  |
| <b>[Section III] QA Test Plan</b>  | <b>28</b> |
| <b>[Section IV] Code Review</b>  | <b>32</b> |
| <b>[Section V] Self-check: Best Practices for Security</b>                 | <b>37</b> |
| <b>[Section VI] Self-check: Adherence to Original Non-Functional Specs</b> | <b>40</b> |
| <b>[Section VII] List of Contributions</b>                                 | <b>53</b> |

## [Section I] Product Summary

### Product Name: College Essentials

College Essentials is an online marketplace that is catered towards students, professors or anyone who has products or services to sell related to college.

College students currently have a difficult time trying to buy textbooks at fair and reasonable prices. Students have to either buy their books in new condition at full price, or find it used for close to original price. It is also difficult for students to be able to sell their old textbooks that they no longer need. College Essentials aims to fix these problems by implementing a marketplace catered to users who want to buy new and used textbooks at reasonable prices, and be able to sell their old textbooks for prices that they choose. The three main sections that College Essentials aims to focus on is an online marketplace, forum posts, and service listings.

### What is unique in our product:

To set College Essentials apart from its competition, we aim to implement a user based online marketplace that allows registered users to be able to buy and sell new and used items from people similar to them at reasonable prices. The marketplace will have grown to include not only textbooks, but other college essential items as well. College Essentials will aim to stand out from their competition by allowing users to sell their used products and let them set their own prices. We plan to have a ratings system to have users know that they are buying from a trusted source. Our goal is to make College Essentials an all in one place for College students and professors alike. Forum postings and service listings will help connect these user groups by giving users an area for communication, and allowing them to buy and sell services related to college that they might have. Finally, we

want to be accessible from any university, by allowing users to search the marketplace based off of location.

### **Itemized List of All Major Committed Functions :**

1. General User
  - 1.1. A general user shall be able to browse forum posts.
  - 1.2. A general user shall be able to browse items on the marketplace.
  - 1.3. A general user shall be able to browse service listings.
  - 1.4. A general user shall be able to search for their needed service.
  - 1.5. A general user shall accept our terms of conditions.
  - 1.6. A general user shall be able to search by item condition in the marketplace.
2. Registered User
  - 2.1. A registered user shall be able to log in with their username.
  - 2.2. A registered user shall be able to review another user.
  - 2.3. A registered user shall be able to create their own profile with text.
  - 2.4. A registered user shall be able to create their own profile with pictures.
  - 2.5. A registered user shall be able to edit their own profile with text.
  - 2.6. A registered user shall be able to edit their own profile with pictures.
  - 2.7. A registered user shall be able to post a picture as their profile picture.
  - 2.8. A registered user shall be able to post a service listing.
  - 2.9. A registered user shall be able to list items for sale in the marketplace.
  - 2.10. A registered user shall be able to post on the forums.
  - 2.11. A registered user shall be able to log out.
  - 2.12. A registered user shall be able to rate other general users.

3. Service Listing
  - 3.1. A service listing shall have a title.
  - 3.2. A service listing shall have the location of where the service will be needed.
  - 3.3. A service listing post shall have characters in its content body.
  - 3.4. A service listing post shall be able to have its content body edited.
  - 3.5. A service listing post shall be able to be unlisted.
4. Forum Listing
  - 4.1. A Forum listing shall have a title.
  - 4.2. A Forum listing post shall have characters in its content body.
  - 4.3. A Forum listing post shall have basic information like time posted.
  - 4.4. A Forum listing shall be visible to all general users.
  - 4.5. A Forum listing post shall be able to be unlisted.
5. Shopping Cart
  - 5.1. A shopping cart shall remember what has been stored within it.
  - 5.2. A shopping cart shall remember how many of one entity have been stored within it.
  - 5.3. A shopping cart shall allow users to edit products within it.
  - 5.4. A shopping cart shall allow for check out.
  - 5.5. A shopping cart shall give descriptions of products stored within it
6. Checkout
  - 6.1. Checkout shall allow a user to input their payment information.

- 6.2. Checkout shall allow a user to input their shipping address.
- 6.3. Checkout shall allow a user to input their billing address.
- 6.4. Checkout shall allow users to review their purchase on last time.

## 7. Navigation Bar

- 7.1. A navigation bar shall have a section for students selling their products.
- 7.2. A navigation bar shall have a section for teachers to look for student assistants.
- 7.3. A navigation bar shall have a section for the student forums.
- 7.4. A navigation bar shall have a section for the about us page.

## 8. Marketplace Listing (Buy/Sell Listing)

- 8.1. A marketplace listing shall have a title.
- 8.2. A marketplace listing shall have a description.
- 8.3. A marketplace listing shall have a photo (or many).
- 8.4. A marketplace listing shall have a price.
- 8.5. A marketplace listing shall be able to be unlisted.
- 8.6. A marketplace listing shall have filters to categorize.

## 9. Ratings

- 9.1. User ratings shall rate a profile on a scale of 1 to 5.
- 9.2. User ratings shall be visible on a registered user's profile.
- 9.3. User ratings shall be visible on a registered user's marketplace listing.
- 9.4. User ratings shall be visible on a registered user's service listing.
- 9.5. User ratings shall be able to be posted by registered users.

**URL to our product:** <http://my-app-dev.us-west-1.elasticbeanstalk.com/>

## [Section II] Usability Test Plan

### Search

- Test Objectives
  - The main objective for this test is to make sure that our search function works in relation to our database. The search bar should be able to search through the database based off of the title of the object searched. It should also be able to search for items with specific categories, and return only items in those categories. These functions are important to be tested because the search bar is the main way to be able to search for items on College Essentials. Without the search bar working, users will not be able to aggregate the page with the specific items that they want, but will instead be flooded with every single item in the database.
- Test Description
  - The system that will be used for the test will be a Windows 10 operating system. Multiple web browsers will be used, from Google Chrome, Firefox, and Microsoft Edge. The tests should be able to run on all the browsers with the same end results.
  - The starting point for the test will be a user who has not logged in. The search bar should work the same regardless if the user is logged in or not, so having a user not be logged in yet should emulate the ability of having majority functionality on the website without needing to be registered.
  - The intended users for this test are anyone who visits this website. Because our starting point was having the user not be logged in, this

will be the same as having the average person being able to access College Essentials and search for what they need first, before deciding if they want to create an account with our services.

- The URL that is going to be tested is:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/home>.

We are aiming to measure the aggregation of the website based on the users inputs or lack of inputs. This means that regardless of what the user searches for, the website should be able to interact with the database in order to populate the page with what the user aims to find.

- Usability Test Table

| <b>Number of pages needed</b> | <b>Test/Use case</b> | <b>% completed</b> | <b>Errors</b> | <b>Comments</b> | <b>% of time to complete task (efficiency)</b> |
|-------------------------------|----------------------|--------------------|---------------|-----------------|--|
| 1                             | Search               | 100%               | N/A           | N/A             | 100%   |
| 1                             | View Items           | 100%               | N/A           | N/A             | 100%   |
| 2                             | Select Item          | 100%               | N/A           | N/A             | 100%   |
| 2                             | Add to Cart          | 100%               | N/A           | N/A             | 100%   |
| 2                             | View Seller          | 100%               | N/A           | N/A             | 100%   |

- User Satisfaction

I was able to search for items with ease. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

i think there should be a marketplace tab on the left section too.

The information for each search item was easily visible. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

one pic didnt load

The items displayed after searching is in a good format. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

pictures could be bigger

## **Item Post**

- Test Objectives
  - Posting has the objective of giving a user who is unfamiliar with technology the ability to post on College Essentials without the prior knowledge of how to. Even without prior experience with other sites, posts should have easy and straightforward labels on all the input fields so that they are clear to the user in what they do. There shouldn't be a question as to how our posting should work, and should allow for users to be able to post whatever items they want on our site (within guidelines) as long as they are logged in. The importance of testing these functions is that College Essential relies on user interaction with buying and selling items, so users must be able to sell items on the website.
- Test Description
  - The system that will be used for the test will be a Windows 10 operating system. Multiple web browsers will be used, from Google Chrome, Firefox, and Microsoft Edge. The tests should be able to run on all the browsers with the same end results.
  - The starting point for the test will be a user who has logged in. Only users who have logged in should be able to make posts. However, any user can find posts. This helps the buying and selling system work on College Essentials, where only registered users are allowed to post, but anyone is allowed to buy.
  - The intended users for this test are any registered users on College Essentials. Because our starting point is having the user be logged in,

only registered users should be able to sell. Registered users are thus our intended test group for this section.

- The URL that is going to be tested is:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/postItem>.

We are aiming to measure the ability of the user to add an item to the database to be able to sell, as well as add any information related to said item. The user must be able to find their item, and edit the information if needed.

- Usability Test Table

| <b>Number of pages needed</b> | <b>Test/Use case</b>  | <b>% completed</b> | <b>Errors</b>                                     | <b>Comments</b>   | <b>% of time to complete task (efficiency)</b> |
|-------------------------------|-----------------------|--------------------|---|---|--|
| 1                             | Access Sell Item Form | 100%               | N/A   | N/A   | 100%   |
| 1                             | Sell Item             | 100%               | N/A   | N/A   | 100%   |
| 2                             | Search for Item       | 90%                | Item appears in item view as new instead of used. | This can be implemented by changing the item information. | 80%  |

|   |                     |     |                                      |  |     |
|---|---------------------|-----|--------------------------------------|--|-----|
|   |                     |     |                                      |  |     |
| 3 | See Item on Account | 20% | Selling history not implemented yet. | Selling history needs to pull from user id and item id.                    | 20% |
| 3 | Edit Item           | 0%  | Edit item is not yet implemented.    | We can implement an edit function based off of selling history on account. | 0%  |

- User Satisfaction

| Item Post Survey   |                       |                       |                       |                       |                                  |                |
|--|-----------------------|-----------------------|-----------------------|-----------------------|----------------------------------|----------------|
| <p>I was able to post a product to sell with ease. *</p>                       |                       |                       |                       |                       |                                  |                |
| 1  | 2                     | 3                     | 4                     | 5                     |                                  |                |
| Strongly Disagree  | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | Strongly Agree |
| <p>Comments:</p> <p>straightforward</p>  |                       |                       |                       |                       |                                  |                |
| <p>I was able to easily find what items I am selling in the marketplace. *</p> |                       |                       |                       |                       |                                  |                |
| 1  | 2                     | 3                     | 4                     | 5                     |                                  |                |
| Strongly Disagree  | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | Strongly Agree |
| <p>Comments:</p> <p>if it was there i found it</p>                             |                       |                       |                       |                       |                                  |                |

I am easily able to track what items I am selling. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

it wasn't working

## **Forum Posts**

- Test Objectives
  - The main objective of this test is to verify that all the functions regarding forum postings are functional. Forums are another feature that College Essentials should have that sets them apart from the competition. Forum posts allow for multiple users to interact with one another just through the use of forum posts. Forum posts consist of a topic that the main user is posting, and a section for comments from other users to reply with. The importance of testing forum posts is to allow another way for users to interact with one another for different topics, without the need to purchase items.
- Test Description
  - The system that will be used for the test will be a Windows 10 operating system. Multiple web browsers will be used, from Google Chrome, Firefox, and Microsoft Edge. The tests should be able to run on all the browsers with the same end results.
  - The starting point for the test will be for users who have logged in. Similar to item posting, only registered users should be able to post forum posts, however, all users should be able to view forum posts regardless of logged in or not. Forum posts have a section for the user who posted them, and thus they need to be logged in.
  - The intended users for this test are anyone registered with the website. In this test, the user must have already been registered and logged in with College Essentials, however the test shall be accessible for users who are not adept with technology. This means that being able to post a forum post shall be easy for any user to post it.

- The URL that is going to be tested is:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/forumList>.

We are aiming to measure the ability of the forum list to be populated with forum posts, as well as allowing any user to be able to search for a specific forum post. Users should also have an easy time making posts of their own.

- Usability Test Table

| <b>Number of pages needed</b> | <b>Test/Use case</b> | <b>% completed</b> | <b>Errors</b>      | <b>Comments</b>  | <b>% of time to complete task (efficiency)</b> |
|-------------------------------|----------------------|--------------------|--------------------|--|--|
| 1                             | View Forum Lists     | 100%               | N/A                | N/A  | 100%   |
| 1                             | Make a New Post      | 80%                | Username is wrong. | Username is hard coded and does not change based on login. | 70%  |
| 2                             | View Forum Post      | 80%                | Username is wrong. | Username is hard coded and does not change based on login. | 70%  |

|   |                          |     |   |   |     |
|---|--------------------------|-----|---|---|-----|
|   |                          |     |   |   |     |
| 2 | Comment on<br>Forum Post | 60% | Comment<br>section not<br>yet<br>implemented. | There is no<br>button to<br>submit a<br>comment.                              | 50% |
| 1 | Search Forum<br>Posts    | 0%  | Search bar<br>not<br>implemented.             | Search bar<br>does not work,<br>search button<br>does not return<br>anything. | 20% |

- User Satisfaction

I was able to post a forum post with ease. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

I was able to easily find forum posts. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

search option didn't work

I am easily able to navigate through forum posts. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

it was organized by title but user was wrong

## ShoppingCart

- Test Objectives
  - The main objective of this test is to verify that all the functions in regards to the shopping cart are working properly. The main functions that we want to test are adding items to the shopping cart, being able to access and manipulate the shopping cart itself, and being able to exit the shopping cart for checkout. We need to make sure that the shopping cart is working as intended since commerce is the central part of College Essentials. Without a working shopping cart, the main functionality of College Essentials is lost.
- Test Description
  - The system that will be used for the test will be a Windows 10 operating system. Multiple web browsers will be used, from Google Chrome, Firefox, and Microsoft Edge. The tests should be able to run on all the browsers with the same end products.
  - The starting point for the test will be a user who has not logged in. In our website design, an unregistered user shall be able to access the main functions of the website, including the ability to make purchases without being logged in. This means that a user who is not logged in should be able to use the shopping cart and beagle to checkout.
  - The intended users for this test are anyone who visits this website. Because our starting point was having the user not be logged in, this will be the same as having the average person being able to access College Essentials and be able to purchase products without having the need to sign up.

- The URL that is going to be tested is:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/ShoppingCart>.

We are aiming to measure the ability of the shopping cart by allowing the user to add items to the shopping cart, being able to show the details of the items in the shopping cart, and being able to interact with said items as well as checkout.

- Usability Test Table

| <b>Number of pages needed</b> | <b>Test/Use case</b> | <b>% completed</b> | <b>Errors</b>                   | <b>Comments</b>                             | <b>% of time to complete task (efficiency )</b> |
|-------------------------------|----------------------|--------------------|---------------------------------|---|---|
| 2                             | Add to Cart          | 100%               | N/A                             | N/A   | 80%   |
| 1                             | Access Shopping Cart | 100%               | N/A                             | N/A   | 100%  |
| 1                             | Edit Cart            | 100%               | N/A                             | N/A   | 90%   |
| 2                             | Checkout Cart        | 90%                | Checkout not fully implemented. | Checkout does not create a new transaction. | 90%   |
| 3                             | Check Purchase       | 30%                | Purchase                        | Purchase history                            | 20%   |

|  |         |  |                        |  |  |
|--|---------|--|------------------------|--|--|
|  | History |  | history not appearing. | page is on account, but the purchase history is not implemented. |  |
|--|---------|--|------------------------|--|--|

- User Satisfaction

I was able to access my shopping cart with ease. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

.....

I was able to interact with my shopping cart with ease. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

i can add and remove items easily

.....

I am easily able to checkout my shopping cart. \*

1      2      3      4      5

Strongly Disagree

Strongly Agree

Comments:

completed my order

## Account

- Test Objectives
  - The main objective of this test is to verify that different functions of the user account are working. These functions include being able to view account information, and edit said account information.  
Accounts shall have an about me that is editable. There should also be a section for recent orders, user listings, and user ratings. The importance for this is because everything that a user does while logged in should be linked to their user account. Users who are also making purchases should be able to view a user's account before making said purchase in order to make sure that they are not getting scammed.
- Test Description
  - The system that will be used for the test will be a Windows 10 operating system. Multiple web browsers will be used, from Google Chrome, Firefox, and Microsoft Edge. The tests should be able to run on all the browsers with the same end products.
  - The starting point for the test will be a user who is logged in. A user must be logged in in order to access the account page of the website. Without being logged in, a user wouldn't have an account, and therefore not have any information that needs to be tracked.
  - The intended users for this test are anyone registered users on the website. Because our starting point is a user who is logged in, any registered user on the website should be able to do the actions needed for the test. The test should be easy enough for any user with any experience in technology to be able to perform.

- The URL that is going to be tested is:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/Account>.

We are aiming to measure the usability of the accounts page, which would include populating the page with user information, and allowing them to edit their own user information. Users should also be able to look at any history regarding their account and any other information.

- Usability Test Table

| <b>Number of pages needed</b> | <b>Test/Use case</b> | <b>% completed</b> | <b>Errors</b>                             | <b>Comments</b>  | <b>% of time to complete task (efficiency)</b> |
|-------------------------------|----------------------|--------------------|---|--|--|
| 1                             | Access Account       | 100%               | N/A                                       | N/A  | 100%   |
| 2                             | Edit Account         | 50%                | Edit account does not change information. | Text fields do not change user information on the main page. | 50%  |
| 2                             | View Recent Orders   | 30%                | Recent orders not yet implemented.        | Recent orders do not yet appear.                             | 10%  |
| 2                             | View User Items      | 30%                | User items not                            | User items being   | 10%  |

|   |              |     |                              |   |     |
|---|--------------|-----|------------------------------|---|-----|
|   |              |     | yet implemented.             | sold does not yet appear.                           |     |
| 1 | Post Ratings | 40% | Ratings not yet implemented. | Drop down has a number section for posting ratings. | 10% |

- User Satisfaction

I was able to access my account information with ease. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

.....

I was able to edit my account information with ease. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

about me and contact me section didn't work

.....

I am able to access my account history with ease. \*

1      2      3      4      5

Strongly Disagree



Strongly Agree

Comments:

.....

## [Section III] QA Test Plan

### Test objectives:

- The purpose for these QA tests are to make sure our superior functions are working properly for College Essentials. Since college essentials is designed to replace ebay with its superior usability, UI and features. We want to make sure our Search and Upload functions are working as intended. Without these features being properly implemented, the entire website will shut down. To ensure this does not happen we are designing tests to test our multiple search features including marketplace listings, form listings and service listings. These are what allows the user to communicate and to profit within the website. The next features are the upload functions for marketplace, forms and service listings. These allow registered users to post their products or services so other users can purchase or communicate with the poster.

### HW and SW setup (including URL):

- Hardware requirements need to be any laptop. As for software there needs to be an OS of any type such that the device is setup properly ie. Windows or Mac. They will also need to have any browser in order to access our website, such as Google, Opera GX, Microsoft Edge, etc. As well as a proper connection such that they can interact with our website, and the other users involved. Once a user all the prerequisites in order to access our website, they are able to visit it here at:

<http://my-app-dev.us-west-1.elasticbeanstalk.com/>

### Feature to be tested:

- Search
  - High level searching such as individual item or category search

- Item Posting
  - Testing input fields such as URL, Title, Price, Description, Quantity, and Quality. Also if logged in
- Form Posting
  - Test Input fields such as title and description. Also tests if user is logged in
- Shopping Cart
  - This testing field is for adding times, removing or adding more quantity, and completing a checkout
- Account
  - Check if editing account is working, ability to use all features within, and see others accounts

### **QA Test plan:**

| <b>Number</b> | <b>Test Title</b> | <b>Description</b>                                      | <b>Test Input</b>   | <b>Expected Output</b>                        | <b>Pass/Fail</b>      |
|---------------|-------------------|---|---|---|-----------------------|
| 1             | Search            | Make sure the search function for their respective page | marketplace/<br>home page<br>(Book)<br>-----<br>Forum page/<br>Service<br>Listing<br>(Any Post) | A post for a book<br>-----<br>Respective Post | Pass<br>-----<br>Fail |
| 2             | Item              | Make sure the item is                                   | Test by   | (Respective                                   | Pass                  |

|   |               |  |   |   |      |
|---|---------------|--|---|---|------|
|   | Posting       | posted properly on the marketplace, and properly stored in database with its proper values (URL, Title, Price, Description, Quantity, and Quality) | adding image url, title, description, quantity, quality and price | item) is posted on the marketplace, and appears on our database with proper URL, Title, Price, Description, Quantity, and Quality |      |
| 3 | Form Post     | Makes sure that the new form post is properly added to the form listing and is stored in the database()  | Enter title and description                                       | There should be a pop up saying that the form was properly posted along with being able to see it on formlistings                 | Pass |
| 4 | Shopping Cart | Adding items to the cart with the ability to   | Adding selected item  | (Respective item) is  | Pass |

|   |         |   |  |   |      |
|---|---------|---|--|---|------|
|   |         | add or reduce quantity<br>then complete the<br>checkout or remove<br>an item. | in full item<br>view                         | present in<br>cart with<br>ability to add<br>or reduce<br>quantity, and<br>the ability to<br>remove item<br>or remove<br>checkout |      |
| 5 | Account | See if account pull up<br>is accurate for same<br>user that clicked           | Click on<br>account<br>button after<br>login | Account<br>page should<br>open up with<br>your<br>personal info   | pass |

## [Section IV] Code Review

### I. eXtreme Programming

A. Our team's coding style has solidified itself to be an eXtreme Programming style. We demonstrated many if not all components of XP such that we had come to the conclusion that this coding style is what fit our situation the best when compared to the other styles. These components include communication, simplicity, feedback, courage, and respect. First being communication which our team has an abundance of. We had plenty of meetings, which albeit had no webcam still had plenty of communications. The communication did not stop there as we had plenty outside of voice chats as well, through things like chatting by text. Overall, it helped keep us on track as to what tasks were at hand. Second feature we practiced was simplicity. We all thought it best that we stuck with what was best for our skills. It worked off our specialties such that we could finish our work in an efficient manner. We also tend to not work towards future prospects, rather we worked with what we were comfortable with to make a simple working project which later we can build upon. Third feature we practiced was feedback, which we were definitely not lacking in. If we found ourselves dissatisfied with a certain feature we would not be afraid to voice our opinions on the subject, and provide feedback. It helped refine our product to what we thought it should be. Which leads me to the fourth we practiced, courage. None of us were shy of courageousness, and were not afraid to voice any of our opinions. Through those opinions we were not afraid of change either, and were willing to change any of our code in a heartbeat if it meant a better project. Last but not least is the respect aspect. No one on the team

lacked a sense of respect for the other person. This builds upon courage and feedback because we never put each other down. We always respected each other's time, skill, and availability which enabled courage, and feedback. Overall, all of these components helped build up to how we got our coding style, which in the end we concluded fit the most into extreme programming.

## II. (Code portion review which needs the QA test plan done)

We are doing a code review for our post item checking algorithm. For this we are using flags and null checking to make sure all the fields are filled out so users can't post blank items. Ryan is asking for this to be reviewed because he wanted the others to see if they could understand the code since he felt like it made no sense to the outside viewer.

 RM

Ryan Michael Murphy



...

To: Darren Lok-Lu... +1 other

Wed 7/27/2022 6:27 PM

Here is our checking method for posting an item to the website, How does it look?

```
129     if(LoginView.logStatus()){
130         float p = Float.parseFloat(price.getValue());
131         int q = Integer.parseInt(quant.getValue());
132         if(price.getValue().contains(s: "-") || quant.getValue().contains(s: "-")){
133             showNeg(userBean);
134             flag++;
135         }
136     }
137
138     if(p == 0 || q == 0){
139         showZero();
140         flag++;
141     }
142
143     }else{
144         flag++;
145         showFail(userBean);
146     }
147
148     if(flag == 0 && url.getValue() != "" && title.getValue() != "" && description.getValue() != ""
149     && category.getValue() != null && condition.getValue() != null && price.getValue() != ""){
150         db.StorePostItem(title.getValue(), url.getValue(), condition.getValue(), category.getValue(),
151         price.getValue(), description.getValue(), quant.getValue());
152         showSuccess(userBean);
153         userBean = new PostItemDetail();
154         this.getUI().ifPresent(ui ->
155             ui.navigate(location: "/home"));
156     }
157
158 );
159 }
```

Brendan sent a response of :

 WL

Wee Brendan Leong



...

To: Ryan Michael Murphy

Thu 7/28/2022 10:50 AM

Good morning Ryan. I think whatever you have here is really good. I can see that this function will be used to populate a site with a new item. I can also see that a user will not be able to populate the price field with a negative pricing. However, there seems to be a lack of comments inside this function. I think without proper comments it will be hard for people without proper context to grasp what this function does in a timely manner. Otherwise, I think this function looks amazing!

...

Because of this response Ryan edited the variable names and added comments to our postItemView.java class. He made the code more readable and spaced things

out. The code is now working fully as expected and the other members of the team can now understand his thought process.

The next code review we as a group checked was Darren's AccountView.java. Darren chose this segment of code because he worked on it a while ago and wanted us to understand what his thought process was and to see if we could see any logic errors.

DW Darren Lok-Lum Wong  
To: Ryan Michael Murphy  
Wed 7/27/2022 6:44 PM

AccountView.java  
8 KB

Hi Ryan,

I'm not 100% sure what the format of this email should be like. However, I was looking for some code review for the AccountView.java file within the application folder. More specifically, from lines 31 to 191. I've also attached the AccountView.java folder for easier viewing.

Thanks and much appreciated,  
-Darren Wong

Ryan responded with this:

RM Ryan Michael Murphy  
To: Darren Lok-Lum Wong  
Wed 7/27/2022 6:54 PM

Hello Darren,  
From what i can see the account view looks good, The only thing missing is comments, Because of this I am having a hard time following what is going on, I see you have a default case to avoid nulls which I like. Could you do some commenting for the other members of the group?

...

Sure, I can do that. | I will take a look. | Here is the screenshot.

Are the suggestions above helpful? Yes No

Reply | Forward

Darren went back to accountView and added comments and made his code legible and spaced out. There were also no major logic errors any of the team mates could see at the moment.

## **[Section V] Self-check: Best Practices for Security**

- I. Account Information
  - A. A general user can view other user profiles
  - B. Only a registered user can edit their own user profile
- II. Forum Posts / Replies
  - A. A general user can view forum posts and replies
  - B. Only a registered user can post forum posts and replies
- III. Service Listings / Replies
  - A. A general user can view service listings and replies
  - B. Only a registered user can post service listings and replies
- IV. Ratings
  - A. A registered user can rate another registered user if they have a transaction with that user.
- V. Payment
  - A. Payment methods will be hashed upon entry, and stored inside the database as hashed values instead of plain text (depending on the payment API)
- VI. Password
  - A. Passwords are hashed upon registration, and stored inside the database as hashed passwords instead of plain text. The following screenshots illustrate how a password from the registration form is stored as a hashed password within the database.

Signup for College Essentials

First name \* Kevin

Last name \* K

Username \* tig777

University SFSU

Email \* kevink@sfsu.edu

Password \* \*\*\*\*\*

Confirm password \* team4\_2022

Agree To Terms & Services?

[Terms & Services](#)

**Create Account**

B.

|      |       |                                 |                 |      |        |
|------|-------|---------------------------------|-----------------|------|--------|
| 23   | Kevin | 44873c4ef315d2a4884cd3a29eabfc5 | kevink@sfsu.edu | K    | tig777 |
| NULL | NULL  | NULL                            | NULL            | NULL | NULL   |

## VII. Data Input Validation

- A. Registration form ensures that all the fields must be filled out, including having an 8+ character password
- B. Login form ensures that the email/username exists, and if so, the password entered will be hashed and matched with the hashed password in the database.
- C. Sell item ensures that all the fields must be filled out, and numerical values such as the quantity of the item listed and its price are logical values (above 0 for quantity, and above \$0).

- D. Posting a forum post or service listing ensures that all the fields must be filled out

## VIII. SQL Injection

- A. Using preparedStatements and VALUES(?, ?, ?..), the user input given is handled as a parameter instead of pure user input

## [Section VI] Self-check: Adherence to Original Non-Functional Specs

### *Functionality*

| Non-Functional Spec  | Status      | Comment |
|--|-------------|---------|
| The website shall use the tools and frameworks that the group has agreed on and approved by the CTO. | <u>Done</u> | N/A     |
| The website shall use Amazon Web Services for deployment.  | <u>Done</u> | N/A     |
| The website shall be user friendly and simple.   | <u>Done</u> | N/A     |
| The website shall be easy to navigate for everyone   | <u>Done</u> | N/A     |
| The website shall be available for students and professors at any school.                            | <u>Done</u> | N/A     |

## Security

| Non-Functional Spec   | Status      | Comment |
|---|-------------|---------|
| The user shall need a username/email and password to be able to post and respond to posts on the website. | <u>Done</u> | N/A     |
| The user shall need an account in order to interact with posts  | <u>Done</u> | N/A     |
| User registration shall only be available to those with emails.   | <u>Done</u> | N/A     |
| The user shall need a registered account in order to write reviews for other users.                       | <u>Done</u> | N/A     |
| Admins shall be able to access all aspects of the websites.   | <u>Done</u> | N/A     |

|   |             |     |
|---|-------------|-----|
| Admins shall be able to moderate the website. | <u>Done</u> | N/A |
|---|-------------|-----|

### Privacy

| Non-Functional Spec   | Status       | Comment  |
|---|--------------|--|
| The user shall agree to a terms of service agreement in order to access our services. | <u>Done</u>  | N/A  |
| Passwords and other personal information shall be kept hidden.                        | <u>Done</u>  | Passwords are hashed and stored into a database                  |
| Users shall be authenticated through username and password.                           | <u>Done</u>  | N/A  |
| Users shall only be able to sign up if they have an email.                            | <u>Done</u>  | N/A  |
| Payments shall be collected through an API.   | <u>Issue</u> | Further testing and research must be done regarding payment APIs |

### Performance

| Non-Functional Spec   | Status          | Comment |
|---|-----------------|---------|
| The website shall be up at all times.   | <u>Done</u>     | N/A     |
| The website shall make it easy to post listings.                              | <u>Done</u>     | N/A     |
| The website shall make it easy to see other users ratings.                    | <u>On Track</u> | N/A     |
| The website shall make it easy to delete old listings.                        | <u>On Track</u> | N/A     |
| The website shall make it easy to allow the user to go through payment system | <u>Done</u>     | N/A     |

### System Requirements

| Non-Functional Spec   | Status      | Comment |
|---|-------------|---------|
| The website shall work on version 1 to 98 of Mozilla Firefox. | <u>Done</u> | N/A     |

|  |             |     |
|--|-------------|-----|
| The website shall work on version 1 to 99 of Google Chrome.  | <u>Done</u> | N/A |
| The website shall work on version 1 to 99 of Microsoft Edge. | <u>Done</u> | N/A |
| The website shall work on version 1.0 to 15.2 of Safari.     | <u>Done</u> | N/A |
| The website shall work on version 2.0 83 of Opera.           | <u>Done</u> | N/A |

### Marketing

| Non-Functional Spec                                      | Status      | Comment |
|--|-------------|---------|
| The website shall be viewable for both guests and users. | <u>Done</u> | N/A     |
| The website shall have its own logo.                     | <u>Done</u> | N/A     |
| The website shall be easy to navigate for any user.      | <u>Done</u> | N/A     |

|  |             |     |
|--|-------------|-----|
| The website shall be seen as a familiar alternative for other online marketplaces. | <u>Done</u> | N/A |
| The website shall be seen as friendly for students and faculty of any school.      | <u>Done</u> | N/A |
| The website shall be not directly linked with any specific school                  | <u>Done</u> | N/A |

### Content

| Non-Functional Spec  | Status      | Comment |
|--|-------------|---------|
| The website shall allow users to search posts by categories. | <u>Done</u> | N/A     |

|  |             |     |
|--|-------------|-----|
| The website shall allow users to search by object name           | <u>Done</u> | N/A |
| The website shall allow users to search by listing name.         | <u>Done</u> | N/A |
| The website shall allow users to leave reviews on other users.   | <u>Done</u> | N/A |
| The website shall allow users to see the reviews of other users. | <u>Done</u> | N/A |
| The website shall have a section for miscellaneous postings.     | <u>Done</u> | N/A |

### Scalability

| Non-Functional Spec  | Status      | Comment |
|--|-------------|---------|
| The website shall be able to handle a large number of users from any school. | <u>Done</u> | N/A     |

|   |                        |     |
|---|------------------------|-----|
| The website shall have enough categories to sort through the growing number of posts. | <u><b>Done</b></u>     | N/A |
| The website shall be able to accommodate larger amounts of listings.                  | <u><b>On Track</b></u> | N/A |
| The website shall be able to accommodate large amounts of user reviews.               | <u><b>On Track</b></u> | N/A |
| The website shall be able to accommodate large amounts of forum posts.                | <u><b>On Track</b></u> | N/A |

### **Capability**

| Non-Functional Spec  | Status             | Comment |
|--|--------------------|---------|
| The website shall be able to provide data requested by the user. | <u><b>Done</b></u> | N/A     |

|   |             |     |
|---|-------------|-----|
| The website shall allow posts to be uploaded in a timely manner.                    | <u>Done</u> | N/A |
| The website shall be able to moderate content within a timely manner.               | <u>Done</u> | N/A |
| The website shall be able to recover from failures in a timely manner.              | <u>Done</u> | N/A |
| The website shall be able to communicate directly and effectively with their users. | <u>Done</u> | N/A |
| The website shall clean up old content within timely manners.                       | <u>Done</u> | N/A |

### Look and Feel

| Non-Functional Spec                    | Status      | Comment |
|--|-------------|---------|
| The website shall have readable fonts. | <u>Done</u> | N/A     |

|  |                        |     |
|--|------------------------|-----|
| The website shall have a simple layout.  | <u><b>Done</b></u>     | N/A |
| The website shall be easy to navigate  | <u><b>Done</b></u>     | N/A |
| The website shall load quickly.  | <u><b>Done</b></u>     | N/A |
| The website shall differentiate between different types of listings through service listings and marketplace listings. | <u><b>Done</b></u>     | N/A |
| The website shall differentiate between listings and miscellaneous forum postings.                                     | <u><b>Done</b></u>     | N/A |
| The website shall visibly show the ratings of other users on their profile.  | <u><b>On Track</b></u> | N/A |

### Coding Standards

| Non-Functional Spec                                  | Status      | Comment |
|--|-------------|---------|
| The code shall be organized and understandable.      | <u>Done</u> | N/A     |
| The code shall have proper working functions.        | <u>Done</u> | N/A     |
| The code shall use git to be pushed and pulled from. | <u>Done</u> | N/A     |
| The code shall use a framework.                      | <u>Done</u> | N/A     |

### Availability

| Non-Functional Spec  | Status      | Comment |
|--|-------------|---------|
| The website shall always be available for users.             | <u>Done</u> | N/A     |
| The website shall generate error messages when errors occur. | <u>Done</u> | N/A     |

|  |                     |   |
|--|---------------------|---|
| The website shall refresh when it fails to load.                     | <u><b>Issue</b></u> | Error checking includes prevention of as many null exceptions and SQL exception, but does not refresh when it fails to load |
| The website shall allow postings to be available whenever for users. | <u><b>Done</b></u>  | N/A   |
| The website shall be available to view for unregistered users.       | <u><b>Done</b></u>  | N/A   |

### **Fault Tolerance**

| Non-Functional Spec  | Status              | Comment  |
|--|---------------------|--|
| The website shall refresh requests for products if request for purchasing fails. | <u><b>Issue</b></u> | Error checking includes prevention of as many null exceptions and SQL exceptions, but does not refresh when it fails to load. Based on many tests, our website properly loads product listings |
| The website shall re-attempt to access the                                       | <u><b>Issue</b></u> | Difficult to requery if the input fields are the same  |

|  |              |  |
|--|--------------|--|
| database for listings if querying fails.                                   |              |  |
| The website shall refresh access requests if users' internet fails.        | <u>Done</u>  | N/A                                      |
| The website shall notify users and allow user's to retry if payment fails. | <u>Issue</u> | Could not implement or test payment API. |

### Storage

| Non-Functional Spec                                | Status                | Comment   |
|--|-----------------------|---|
| The website shall store usernames in the database. | <u>Done</u>           | N/A   |
| The website shall store passwords in the database. | <u>Done</u>           | N/A   |
| The website shall store schools in the database.   | <u>On Track/Issue</u> | We have some California universities stored in our database, but not a comprehensive list of all the universities in the country or the world. At |

|  |             |   |
|--|-------------|---|
|  |             | this point of our project, we didn't have the time to implement colleges throughout our website, e.g. service listings. Colleges in terms of functional requirements would've been done in a future update. |
| The website shall store the user's listings in the database. | <u>Done</u> | N/A   |
| The website shall store user's ratings in the database.      | <u>Done</u> | N/A   |

## [Section VII] List of Contributions

### I. Ryan

- A. Worked on the QA testing plan and Code review
- B. Worked on form validation for login and registration (e.g. making sure fields were filled out, responsive feedback for passwords)
- C. Worked on a post item for more validation and error checking for user input.
- D. Worked on account for user specific clicks

E. Worked on Adherence to original Non-functional specs

## **II. Duncan**

- A. Worked on the Product Summary
- B. Worked on hashing passwords, including hashing the password upon registration and ensuring login worked with the stored hashed password
- C. Debugged the hashing function for hashing passwords.

## **III. Justin**

- A. Worked on creating and documenting Usability Test Plan.
- B. Worked on product summary and itemized list of committed functions.
- C. Worked on adding a Sell History tab to display the items sold by a user.
- D. Worked on login verification on posting forums and service listings.

## **IV. Brendan**

- A. Worked on QA Test Plan and Code Review sections
- B. Worked on responsive forum listings, such displaying the forums from the database
- C. Made dynamic user pages that conform to a clicked on user from the forum listing page.

## **V. Darren**

- A. Worked on Self-check: Adherence on Best Practices for Security

- B. Worked on posting forums and service listings (e.g. posting in the database)
- C. Worked on making individualized account pages based on user login
- D. Worked on general styling of the website, including to Registration and Sell Item

## [Section III] Screenshots of Final Demo

Home:

The screenshot shows the homepage of the "College Essentials" website. The header features a logo with a stylized 'C' and 'E', followed by the text "College Essentials". On the right side of the header are links for "Account", "Log Out", and "Cart". Below the header is a navigation bar with links: "Home" (which is highlighted in blue), "Sell an Item", "Forum Listings", and "Services Listings". A search bar with placeholder text "Search For Products, and change the category section..." is positioned above a main content area. The content area has a blurred background image of a library or bookstore. It contains text encouraging users to connect students and faculty from colleges all over the world, and provides instructions on how to get started by browsing the marketplace, asking questions in forums, or looking for service listings. Below this text are three buttons: "Marketplace", "Forum Place", and "Service Listings". Under each button is a small thumbnail image: a green square with dollar signs for the Marketplace, a blue circle with a yellow speech bubble for the Forum Place, and a photo of three people at a table for the Service Listings.

## MarketPlace:

The screenshot shows a web-based marketplace interface. At the top, there's a navigation bar with a logo, the title 'College Essentials', and links for 'Account', 'Log Out', and 'Cart'. Below the navigation is a search bar with placeholder text 'Search For Products, and change the category section...' and a dropdown menu for 'All Categories'. A large, blurred background image of a room serves as the header.

The main content area displays a grid of product cards. The first row contains four items: 'Sony Headphones' (\$220), 'Parka Jacket' (\$82), 'Mongoose Bike' (\$60), and 'Skateboard' (\$95). The second row contains three items: a banana, two pencils, and a backpack. The third row contains one item: a red apple.

On the left side of the page, there's a sidebar with links for 'Home', 'Sell an Item', 'Forum Listings', and 'Services Listings'. At the bottom of the sidebar, there are links for 'About Us' and 'Terms of Service'.

itemView:

College Essentials

≡ Item

Account Log Out Cart

Home Sell an Item Forum Listings Services Listings

Parka Jacket

Description  
No description was given

New

By tester6

Quantity  
1

\$82

Add to Cart



About Us

Terms of Service

Shopping cart:

= Shopping Cart

Account Log Out Cart

Recently Viewed

Shopping Cart

Name: Parka Jacket

Quantity: 1

Price: \$82

Final Price: \$82.0

Remove from Cart

checkout

Parka Jacket



Check out:

---

## Checkout

All fields are required unless otherwise noted

Checkout 1/3

### Personal details

Name \*

Email address \*

Phone number \*

Remember personal details for next time

Checkout 2/3

### Shipping address

Country \*

Street address \*

Postal Code \*

City \*

Billing address is the same as shipping address

Remember address for next time

Checkout 3/3

### Personal details

Cardholder name \*

Card Number \*

Security Code \*

What is this?

Expiration month \*

Expiration year \*

[Cancel order](#)

[Pay securely](#)

| Order   |  | Edit                  |
|---|--|-----------------------|
| Parka Jacket  |  | Quantity: 1<br>\$82.0 |
|  |  |                       |
| Price Total: \$82.0   |  |                       |

Personal Account:

College Essentials

≡ My Account

Admin Account Log Out Cart

Home Sell an Item Forum Listings Services Listings

Ryan Murphy @tester7

About Me

THIS IS MY ADMIN ACCOUNT

Contact Me

NULL

Ratings 4.2

@Tester8 2022-08-01 03:26:12  
5: HEY TESTING RATINGS

@Tester8 2022-08-01 03:26:15  
5: HEY TESTING RATINGS

Rate account:

**About Me**

**Contact Me**

**Ratings 3.0**

Tester8 2022-08-02 04:13:25  
1: HEY

Tester8 2022-08-02 17:30:51  
5: HEY

Add a new rating for this user

## Sell Item:

The screenshot shows a web application interface for posting items. The top navigation bar includes links for Admin, Account, Log Out, and Cart. On the left, a sidebar menu lists Home, Sell an Item (which is highlighted), Forum Listings, and Services Listings. Below the sidebar are links for About Us and Terms of Service. The main content area is titled "Post Item" and contains a "Sell Item Form". The form fields include:

- Enter URL of Image •
- Title Of Post •
- Describe the Item •
- Select Condition of Item •
- Select Category of Item •
- Enter Item Quantity
- Enter Price •
-

## Form Listing:

Forum List

Admin Account Log Out Cart

Search for a forum post (Title, Description, All) All Search Make a new post

(1) Title: Things you wished you knew earlier about college? User: tester7 Based on your college experience, what are some things you wished you knew sooner?

(2) Title: Thoughts on SFSU? User: tester7 Looking into SFSU as my college, any thoughts?

(3) Title: testing for comments User: dawongguy hey

(4) Title: Essentials? User: dawongguy What are some essentials you took to college?

(5) Title: College Ad... User: dawong... Hi, I'm a high school student looking for advice about university in general. What are some factors you considered when you cho...

(6) Title: help pls User: dawongguy testtest

(7) Title: yanimal User: dawongguy yanimalscafelogs

(8) Title: hello User: dawongguy go watch my youtube videos @yanimalscafelogs

(9) Title: Brendan is a weeb User: bobnob idk what ur fav anime is

## Form Post:

The screenshot shows a web application interface for a forum page. On the left, there is a sidebar with a logo, navigation links (Home, Sell an Item, Forum Listings, Services Listings), and links to About Us and Terms of Service. The main content area has a header "Forum Page". It features a user profile card for "dawongguy" with a blue circular icon, followed by sections for "About Me" (text: "Hi, I'M DAWONGGUY! NICE TO MEET YOU!", "CS @ NORTHWESTERN") and "Contact Me" (text: "FOR BUSINESS INQUIRIES ONLY: BRENDINI@GMAIL.COM"). Below this is a comment feed. The first comment is from "TESTER7" at "2022-08-02 17:43:21" with the text "HEY WHATS UP?". At the bottom, there is a form field "Add a new comment for this user" with a "Submit" button.

## Service Listing:

College Essentials

Home Sell an Item Forum Listings Services Listings

Admin Account Log Out Cart

Search for a service (Title, Description, User, School) All Search Make a new post

(1) Title: Looking for a job/int... User: tes... Hi, I'm a junior at UCSB right now and I'm looking for an internship or job related to bio. Would love some recommendations of po...

(2) Title: [SFSU] Looking for a BIO349 Study Partner User: c9thefreak As stated as above, looking for a study partner for BIO349 this fall. Thanks.

(3) Title: hey User: dawongguy cwcw

(4) Title: Looking for a tutor for Operating Syste... User: dawongg... Hi, I'm looking for a tutor, or even just help about understanding how to properly work with memory in...

(5) Title: potato User: dawongguy rise of gru



## Service Post:

The screenshot shows a web application interface for a service listing. At the top, there's a navigation bar with links for Account, Log Out, and Cart. On the left, a sidebar titled "College Essentials" includes links for Home, Sell an Item, Forum Listings, and Services Listings. Below the sidebar, a user profile card for "c9thefreak" is displayed, featuring a blue circular icon with a white person symbol. The main content area has a title "[SFSU] Looking for a BIO349 Study Partner". A message from "c9thefreak" states: "As stated as above, looking for a study partner for BIO349 this fall. Thanks." Below this, two comments are shown:

- Comment by "tester7": "HI, I'M C9THEFREAK! NICE TO MEET YOU!" (posted 2022-08-02 17:44:49)
- Comment by "Tester8": "ID LOVE TO BE YOUR STUDY BUDDY" (posted 2022-08-02 01:55:52)

At the bottom, there's a text input field for adding a new comment with a "Submit" button.

New Service Listing:

The screenshot shows a web application titled "Service Post". The header includes a logo for "College Essentials", a navigation menu with links to "Home", "Sell an Item", "Forum Listings", "Services Listings", "Account", "Log Out", and "Cart". A prominent button in the center says "Make a new Service Listing!". Below it is a large input field labeled "Title". Further down is another input field labeled "Description". At the bottom right of the form area is a "Submit" button. The background of the page features a photograph of a room with a whiteboard, a calendar on the wall, and a window showing a starry night sky.

New Form Listing:

College Essentials

Home Sell an Item Forum Listings Services Listings

Account Log Out Cart

## Forum Post

### Make a new Post!

Title

Description

submit

About Us Terms of Service

The screenshot shows a website interface for 'College Essentials'. At the top, there's a navigation bar with links for Home, Sell an Item, Forum Listings, Services Listings, Account, Log Out, and Cart. Below the navigation is a header 'Forum Post' with a sub-header 'Make a new Post!'. There are two input fields: 'Title' and 'Description', both currently empty. Below these fields is a dark button labeled 'submit'. At the bottom left of the main content area, there are links for 'About Us' and 'Terms of Service'. The background of the main content area features a photograph of a room with a whiteboard, a window showing a starry night sky, and a door. To the left of the main content area, there's a sidebar with the 'College Essentials' logo and the same navigation links as the top bar.

## [Section IV] Screenshots of DB Tables

1 • SELECT \* FROM RegisteredUser

| Result Grid |                    |            |                                  |                         |           |               |                 |                     |                       |           |         |
|-------------|--------------------|------------|----------------------------------|-------------------------|-----------|---------------|-----------------|---------------------|-----------------------|-----------|---------|
|             | registered_user_id | first_name | password                         | email                   | last_name | username      | general_user_id | about               | contact               | is_banned | pfp     |
| 1           |                    | Darren     | team4_2022                       | dwong19@mail.sfsu.edu   | Wong      | dawongguy     | 1               | CS @ SFSU F'22      | dwong19@mail.sfsu.edu | 0         | NULL    |
| 2           |                    | Ryan       | team4_2022                       | rmurphy19@mail.sfsu.edu | Murphy    | deadlepricon1 | 2               | NULL                | NULL                  | 0         | NULL    |
| 3           |                    | admin      | team4_2022                       | admin@gmail.com         | admin     | admin         | 3               | NULL                | NULL                  | 0         | NULL    |
| 6           |                    | Ryan       | team4_2022                       | 1234@gmail.com          | Murphy    | getsmurfed    | 6               | NULL                | NULL                  | 0         | NULL    |
| 7           |                    | bob        | team4_2022                       | 324@gmail.com           | nob       | bobnob        | 7               | NULL                | NULL                  | 0         | NULL    |
| 8           |                    | Ryan       | team4_2022                       | rm92691@gmail.com       | Murphy    | deadlepricon  | 8               | NULL                | NULL                  | 0         | NULL    |
| 9           |                    | hi         | aaaaaaaaaa                       | theman@gmail.com        | bye       | dead          | 9               | NULL                | NULL                  | 0         | NULL    |
| 11          |                    | Duncan     | Test                             | stuff@test.edu          | Forniss   | tester_Duncan | 11              | NULL                | NULL                  | 0         | NULL    |
| 12          |                    | test       | BruhTest                         | adminintrest            | test      | test          | 12              | NULL                | NULL                  | 0         | NULL    |
| 14          |                    | test4      | 44873c4ef315d2a4884cd3a29eabfc5  | test4@sfsy.edu          | test4     | test4         | 16              | NULL                | NULL                  | 0         | NULL    |
| 17          |                    | Duncan     | 3bddaa616298fd0e54170f16b188dda0 | dunkey@sfsu.edu         | Forniss   | getdunkedon   | 19              | NULL                | NULL                  | 0         | NULL    |
| 22          |                    | Johnny     | 44873c4ef315d2a4884cd3a29eabfc5  | johnnytest@sfsu.edu     | Test      | johnnytest    | 24              | NULL                | NULL                  | 0         | NULL    |
| 23          |                    | Kevin      | 44873c4ef315d2a4884cd3a29eabfc5  | kevink@sfsu.edu         | K         | tig777        | 25              | NULL                | NULL                  | 0         | NULL    |
| 24          |                    | Justin     | 1d916d68b3053d5c055fd2c7676ec21  | test@email.com          | Wong      | JustinTest    | 26              | This me!            | This me!              | 0         | http... |
| 25          |                    | ryan       | 552e6a97297c53e592208cf97fbb3b60 | tester6@gmail.com       | Murphy    | tester6       | 27              | NULL                | NULL                  | 0         | NULL    |
| 26          |                    | yadanar    | 5f4dcc3b5aa765d61d8327deb882cf99 | email@gmail.com         | a         | na            | 28              | NULL                | NULL                  | 0         | NULL    |
| 29          |                    | Ryan       | 3dbe00a167653a1aaee01d93e77e730e | tester7@gmail.com       | Murphy    | tester7       | 31              | This is my admi...  | null                  | 0         | http... |
| 30          |                    | Ryan       | 3dbe00a167653a1aaee01d93e77e730e | tester8@gmail.com       | Murphy    | Tester8       | 32              | hey this is a te... | @tester8@gmail.com    | 0         | http... |
| 31          |                    | test8      | 5f4dcc3b5aa765d61d8327deb882cf99 | test8@test.com          | test8     | test8         | 33              | NULL                | NULL                  | 0         | NULL    |
| 32          |                    | Johnny     | 44873c4ef315d2a4884cd3a29eabfc5  | c9@ucd.edu              | K         | c9thefreak    | 34              | NULL                | NULL                  | 0         | NULL    |
| 33          |                    | qqq        | 1bbd886460827015e5d605ed44252251 | qqq@gmail.com           | qqq       | qqq           | 35              | NULL                | NULL                  | 0         | NULL    |
| 34          |                    | Carey      | 44873c4ef315d2a4884cd3a29eabfc5  | carey@gmail.com         | L         | pigtailsrock  | 36              | NULL                | NULL                  | 0         | NULL    |
| *           |                    | NULL       | NULL                             | NULL                    | NULL      | NULL          | NULL            | NULL                | NULL                  | NULL      | NULL    |

The screenshot shows a MySQL Workbench interface. At the top, there's a toolbar with various icons. Below the toolbar, a query editor window displays the SQL command: `1 • SELECT * FROM ForumPost`. The main area is a result grid titled "Result Grid". The grid has columns: forum\_post\_id, title, description, and created\_at. It contains 13 rows of data. The last row is a footer row with all columns labeled "NULL".

|   | forum_post_id | title                  | description  | created_at          |
|---|---------------|------------------------|--|---------------------|
| ▶ | 1             | Brendan is a weeb      | idk what ur fav anime is                             | 2022-07-22 00:46:16 |
|   | 4             | LFG?                   | LFGGG  | 2022-07-24 05:39:28 |
|   | 5             | hello                  | go watch my youtube videos @yanimalscafevlogs        | 2022-07-27 04:38:24 |
|   | 6             | yanimal                | yanimalscafevlogs                                    | 2022-07-27 04:39:08 |
|   | 7             | help pls               | testtest   | 2022-07-29 05:58:40 |
|   | 8             | College Advice         | Hi, I'm a high school student looking for advice ... | 2022-07-30 01:41:09 |
|   | 9             | Essentials?            | What are some essentials you took to college?        | 2022-07-30 01:43:55 |
|   | 10            | testing for comments   | hey  | 2022-08-01 03:21:02 |
|   | 11            | test                   | test   | 2022-08-01 18:40:11 |
|   | 12            | Thoughts on SFSU?      | Looking into SFSU as my college, any thoughts?       | 2022-08-02 01:50:23 |
|   | 13            | Things you wished y... | Based on your college experience, what are so...     | 2022-08-02 02:08:21 |
| * | HULL          | HULL                   | HULL   | HULL                |

1 • SELECT \* FROM ForumReply

| forum_reply_id | description                     | posted_at           |
|----------------|---------------------------------|---------------------|
| 8              | i hope this works: Toilet paper | 2022-07-31 08:39:02 |
| 13             | test                            | 2022-08-01 07:37:24 |
| 14             | am i going to get finessed?     | 2022-08-01 07:38:22 |
| 17             |                                 | 2022-08-01 18:32:36 |
| 18             |                                 | 2022-08-01 18:34:25 |
| 19             |                                 | 2022-08-01 18:34:26 |
| 20             |                                 | 2022-08-01 18:34:55 |
| 22             | test                            | 2022-08-02 01:52:57 |
| 23             | hey                             | 2022-08-02 03:34:50 |
| 24             | hey                             | 2022-08-02 04:14:44 |
| *              | NUL                             | NUL                 |



SQL File 3\* ×

File Edit View Insert Tools Options Help

Limit to 2000 rows

1 • SELECT \* FROM Product

2

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

|   | product_id | general_type    | product_image   | product_condition |
|---|------------|-----------------|---|-------------------|
| ▶ | 2          | Lifestyle       | <a href="https://i.imgur.com/CwC7KEi.jpg">https://i.imgur.com/CwC7KEi.jpg</a>                                     | New               |
|   | 3          | Furniture       | <a href="https://i.imgur.com/hlEVDi.jpg">https://i.imgur.com/hlEVDi.jpg</a>                                       | New               |
|   | 4          | Furniture       | <a href="https://i.imgur.com/xKBukIf.jpg">https://i.imgur.com/xKBukIf.jpg</a>                                     | New               |
|   | 5          | Miscellaneous   | <a href="https://i.imgur.com/NLnQCB3.jpg">https://i.imgur.com/NLnQCB3.jpg</a>                                     | Used              |
|   | 9          | Textbook        | <a href="https://i.imgur.com/JEwHfmy.jpg">https://i.imgur.com/JEwHfmy.jpg</a>                                     | Used              |
|   | 10         | School Supplies | <a href="https://i.imgur.com/qGQJ6UV.jpg">https://i.imgur.com/qGQJ6UV.jpg</a>                                     | New               |
|   | 11         | Miscellaneous   | <a href="https://i.imgur.com/D6eNwVv.jpg?1">https://i.imgur.com/D6eNwVv.jpg?1</a>                                 | Used              |
|   | 12         | Lifestyle       | <a href="https://i.imgur.com/IGSremk_d.webp?maxwidth...">https://i.imgur.com/IGSremk_d.webp?maxwidth...</a>       | Used              |
|   | 16         | Lifestyle       | <a href="https://i.imgur.com/BbLt1aH.jpg">https://i.imgur.com/BbLt1aH.jpg</a>                                     | New               |
|   | 24         | Miscellaneous   | <a href="https://i.imgur.com/YjtEzqOh.jpg">https://i.imgur.com/YjtEzqOh.jpg</a>                                   | New               |
|   | 27         | Lifestyle       | <a href="https://i.imgur.com/bS7eD.jpg">https://i.imgur.com/bS7eD.jpg</a>   | Used              |
|   | 28         | Miscellaneous   | <a href="https://preview.redd.it/tuzvdDani7q71.jpg?widt...">https://preview.redd.it/tuzvdDani7q71.jpg?widt...</a> | Used              |
|   | 29         | Textbook        | <a href="https://i.imgur.com/814k8dxg.jpg?1">https://i.imgur.com/814k8dxg.jpg?1</a>                               | Used              |
|   | 30         | Miscellaneous   | <a href="https://i.imgur.com/rYcyLFE.jpeg">https://i.imgur.com/rYcyLFE.jpeg</a>                                   | New               |
|   | 34         | null            | <a href="https://i.imgur.com/U5gpJU1.jpg">https://i.imgur.com/U5gpJU1.jpg</a>                                     | null              |
|   | 35         | Miscellaneous   | <a href="https://www.applesfromny.com/wp-content/upl...">https://www.applesfromny.com/wp-content/upl...</a>       | Used              |
|   | 36         | Textbook        | <a href="https://preview.redd.it/2vb9r737n4o71.jpg?au...">https://preview.redd.it/2vb9r737n4o71.jpg?au...</a>     | Used              |
|   | 37         | Miscellaneous   | <a href="https://www.kapwing.com/resources/content/i...">https://www.kapwing.com/resources/content/i...</a>       | Used              |
|   | 38         | Miscellaneous   | <a href="https://i.imgur.com/0RNfpNv.jpg">https://i.imgur.com/0RNfpNv.jpg</a>                                     | New               |
|   | 39         | Miscellaneous   | <a href="https://i.imgur.com/0RNfpNv.jpg">https://i.imgur.com/0RNfpNv.jpg</a>                                     | New               |
|   | 40         | Miscellaneous   | <a href="https://i.imgur.com/CKExOyL_d.webp?maxwidt...">https://i.imgur.com/CKExOyL_d.webp?maxwidt...</a>         | Used              |
|   | 41         | Miscellaneous   | ik  | New               |
|   | 42         | School Supplies | Delete  | Used              |
|   | 43         | Miscellaneous   | delete  | New               |
|   | 44         | Miscellaneous   | unkown  | New               |
|   | 45         | Miscellaneous   | tet   | Used              |
|   | 46         | Textbook        | test  | New               |
|   | 47         | Textbook        | test  | New               |
|   | 48         | Lifestyle       | <a href="https://i.imgur.com/jYJEDEb.jpg">https://i.imgur.com/jYJEDEb.jpg</a>                                     | New               |
|   | 49         | Textbook        | random  | New               |

Product 9 ×

```

1 •  SELECT * FROM ServiceListing
2

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

|   | service_listing_id | title                                     | description   | created_at          |
|---|--------------------|---|---|---------------------|
| ▶ | 1                  | test                                      | test  | 2022-07-24 06:18:40 |
|   | 2                  | potoato                                   | rise of gru   | 2022-07-24 06:19:43 |
|   | 3                  | Looking for a tutor for Operating Systems | Hi, I'm looking for a tutor, or even just help abo... | 2022-07-30 02:13:52 |
|   | 4                  | hey                                       | cvcv  | 2022-08-01 03:21:33 |
|   | 5                  | [SFSU] Looking for a BIO349 Study Partner | As stated as above, looking for a study partner...    | 2022-08-01 18:52:13 |
| * | 6                  | Looking for a job/internship              | Hi, I'm a junior at UCSB right now and I'm lookin...  | 2022-08-02 02:15:57 |
|   | HULL               | NULL                                      | NULL  | NULL                |

```

101 •  SELECT * FROM ServiceReply;

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

|   | service_reply_id | description     | posted_at           |
|---|------------------|-----------------|---------------------|
| ▶ | 3                | pls             | 2022-08-01 08:41:21 |
|   | 4                | test            | 2022-08-01 08:45:48 |
|   | 5                | ayylmaoo?       | 2022-08-01 08:53:39 |
|   | 7                | hey             | 2022-08-01 18:24:23 |
|   | 8                |                 | 2022-08-01 18:25:12 |
|   | 9                | Hey             | 2022-08-01 18:26:23 |
|   | 10               | hey             | 2022-08-01 18:28:20 |
|   | 11               | test            | 2022-08-01 18:30:01 |
|   | 12               | test            | 2022-08-01 19:30:19 |
|   | 13               | test            | 2022-08-02 01:55:52 |
|   | 14               | sure soun...    | 2022-08-02 17:01:19 |
| * | 15               | I'd love to ... | 2022-08-02 17:44:49 |
|   | HULL             | NULL            | NULL                |

Result Grid | Form Editor | Field Types | Query Stats | Execution

SQL File 3\* ×

1 • SELECT \* FROM TransactionItems

2

<

Result Grid | Filter Rows:  | Edit: | Export/Import: | Wrap Cell Content:

|   | transaction_id | product_id    | title  | image  | price            | quantity |
|---|----------------|---------------|--------|--|------------------|----------|
| ▶ | 3              | 30            | Banana | https://i.imgur.com/rYcyLFE.jpeg               | 5                | 1        |
| 4 | 30             | Banana        |        | https://i.imgur.com/rYcyLFE.jpeg               | 5                | 1        |
| 5 | 27             | Pea Car       |        | https://i.imgur.com/bS7eD.jpg                  | 20000            | 1        |
| 6 | 10             | Sharp Pencils |        | https://i.imgur.com/qGQJ6UV.jpg                | No price was set | 1        |
| 7 | 30             | Banana        |        | https://i.imgur.com/rYcyLFE.jpeg               | 5                | 1        |
| 8 | 30             | Banana        |        | https://i.imgur.com/rYcyLFE.jpeg               | 5                | 1        |
| 9 | 35             | apple         |        | https://www.applesfromny.com/wp-content/upl... | 10               | 1        |
| * | NULL           | NULL          | NULL   | NULL   | NULL             | NULL     |

## [Section V] Screenshots of Task Management

The screenshot shows a task management application interface with a sidebar and four main boards:

- Done:** M4 section 7, M4 section 6, M4 section 5, M4 section 4, M4 section 3, M4 section 2, M4 section 1.
- Review:** + Add a card.
- TODO:** Fix contact info on in div forumposts, fix footer on forumlisting + servicelisting, fix css alignment on comments box, TOS and privacy, show pfp on forum/servicelisting. + Add a card.
- In Progress:** User linking on forumlisting/servicelisting, fix itemview parsing for indiv postings, fix account rating view other users, connection.close() for DB files, M5. + Add a card.

The sidebar includes sections for Boards, Members, Settings, Workspace views (Table, Calendar), Your boards, and College Essentials (selected). A purple button at the bottom left says "Try Premium free".

## **[Section VI] Team Member Contributions**

### **I. Ryan**

A. Since it's myself it's kind of hard to be unbiased for my grade but overall I worked on this project every single day and would help out where I was needed. 10/10

### **II. Justin**

A. Justin was amazing this entire semester. Any time there was an issue, he would volunteer to take it on himself. He also took on a lot of the milestone fixes and quality checked them alongside me. His coding skills also shined here. Justin was able to balance his work and classes perfectly. Therefore he deserves a 10/10.

### **III. Duncan**

A. Duncan was our main designer for our website, He created the wireframes and mockup drawings for the use of our website. But that's where it stopped. He did a great job on what he was assigned but did not contribute to the coding. 6/10

### **IV. Brendan**

A. Brendan was always on top of everything, he would see bugs and try to fix them or tell another member where it was so it could get done in a timely manner. He was also a CSS wizard, he was able to fix any CSS related issue and he was the one who would implement a lot of the website design. When I asked him for help he would do it within an instance. Therefore he deserves a 10/10.

## V. Darren

A. Darren was always around to help. Whenever I needed database help, he was there with the solution. He created the database we use and helped us implement it. Darren was also great at coding, he was probably the most well rounded coder out of us, his front end skills and backend skills are on par with each other. He would jump on any task he saw that wasn't done and would help me with keeping track of all that was happening. Therefore he gets a 10/10.

## [Section VII] Post Analysis

As our project is reaching completion, we are starting to see the diamond in the rough that the team as a whole has created. With the end of the semester and the final product on its way to its final version, we all took a step back and looked at what we accomplished in such a short time. Along with the fun of creating a project that the team came up with out of nowhere and built, there were many drawbacks.

The biggest draw back our team faced was the missing two members. In the beginning we didn't feel any drawbacks with such a small team, if anything we saw it as an advantage. But as the semester moved along, slowly but surely we started to notice the amount of work on our plates. If we had a normal semester length to complete this project with the team size we had, then we would have not felt the pressure nearly as hard. If we could have had two more teammates then we as a team would have won every single milestone since we consisti. That leads me to my next major challenge. Since we took this class over the summer, We had half the time to complete this project other teams who take this class during the regular school year would have. Two months to completely build a functioning website was very rough. The Last main challenge was choosing the wrong framework. Since I was Team Lead I thought vaadin would be the best framework since it was java based. We should have chosen a java script framework since it's created for websites. Java for a frontend and backend language was not the best idea. I believed since everyone knew java that we could easily create the best project in the class. But this hurt us since it was so foreign and everyone had to learn this strange framework. The framework itself is very cool but has some limitations.

Overall this class was my favorite computer science class that I have ever taken. The creative freedom was such a great change from the other classes I've

taken where they make me fit into their framework for the assignments. This class helped me develop my leadership skills along with my software engineering skills. This project is also very amazing since we can use it for our personal use inorder to show off to jobs.