

Final Project for SW Engineering Class CSC 648-848 Spring 2022

Team 2

GatorMart

Shane Waxler - Team Lead - swaxler@mail.sfsu.edu

Chuting Yan - Front-End Lead

Robert Garcia - Back-End Lead

Minggu Ma - GitHub Master

Melissa Ho - Front-End

Xiaoqing Yao - Front-End

Joe Guan - Back-End

<http://ec2-54-237-46-1.compute-1.amazonaws.com/>

5/21/2022

I. Product Summary

GatorMart is an online marketplace for the members of the San Francisco State students, professors, and faculty. It allows users with a validated SFSU email to buy and sell items and services. Once registered, validated, and logged in, a user may view product details, make offers on listings, as well as post their own. Interest shown in a user's listings will come in the form of messages, which is viewed on the user's profile page.. Users of the website can be assured that they are interacting with other people from the SFSU community due to the site administration validation.

GatorMart allows users to

- View the current items in the marketplace
- Search for items
- Filter Listings
- Register for an account with SFSU email

Registered users can

- Login to their account
- Upload data to post their own listings
- Message other sellers to request purchase
- Accept or decline offers on their current listings

Administrators can

- Remove listings
- Approve or deny listings
- Approve or deny new user registration, or delete user accounts

GatorMart's unique feature is to allow posters to choose between several on campus options to meet up. These locations are main hubs in busy areas of the campus.

<http://ec2-54-237-46-1.compute-1.amazonaws.com/>

II. Milestone Documents

Software Engineering 648 Spring 2022

GatorMart

An online marketplace for SFSU students/faculty

Team #2

Shane Waxler - Team Lead

Chuting Yan - Front-End Lead

Robert Garcia - Back-End Lead

Minggu Ma - GitHub Master

Melissa Ho - Front-End

Xiaoqing Yao - Front-End

Joe Guan - Back-End

Milestone #1

3/5/22

Original Date	Revise 1 Date
3/5/2022	3/9/2022

I. Executive Summary

In most hallways on campus at San Francisco State University, we see multiple bulletin boards looking to buy, or sell goods and services. Slips of paper, in all sizes, shapes, and colors can be found posted for the passby to glance at. A person looking for a particular item or service could spend hours roaming the hallways of every building, looking across all these bulletin boards. Not only is this time consuming and frustrating, these efforts could prove to be futile. In the fall of 2020, San Francisco State University enrolled 24,025 students. With a large and diverse body of staff and students, the university lacks a centralized hub for them to buy, and sell goods and services. We have developed a website service, GatorMart which provides this service exclusively to current students, faculty, and staff.

GatorMart is a vast online “bulletin board” that is organized into specific categories, allowing registered users to quickly find a specific item. To register, all one needs is a sfsu.edu email that confirms they are a part of the San Francisco State community. Once a user’s registration has been confirmed, they are free to browse various categories for the items or services they are seeking. These include old textbooks, tutoring services, electronics, and job listings. The bulletin board is maintained by an administrator who must approve all postings before they go public. This way, we are able to filter out any inappropriate or spam postings. The administrator’s responsibilities also include confirming a user as a current staff or student of the university. A user can be distinguished by badges on their profile, indicating whether they are part of the staff or student body. Specifics of how a user can fully utilize GatorMart can be found in the following sections.

As former members of the SFSU community, this project means more convenience and accessibility for current and future members of the community. Our team believes that

GatorMart will pave the way for a streamlined online community of users at San Francisco State University to purchase, and sell.

II. Personae and Use Cases

Personae

<p>Persona #1 Dylan</p>  <p>Student at SFSU The Buyer</p>	<p>About Dylan:</p> <ul style="list-style-type: none"> • He has been a full-time student at SFSU for 2 years now and also has a part-time job at Peet's Coffee • He rarely buys his books new as he prefers to save more money buying used books. • He lives on campus. • Has good tech skills <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • Since he's a full-time student and has a job, he is normally very busy; therefore most of his day is spent on campus or at work. Because of this, he normally orders his books online as he doesn't have enough free time to pick books up. However he is finding that the costs of used books online are much more than they would be to pick up in person. • Additionally, he is also incredibly concerned about his safety in picking up books from strangers; if he were to pick up a used book in person; he would want to know that he would be safe/the seller is reputable.
---	--

<p>Persona #2 Sophia</p>  <p>Not a member of SFSU</p>	<p>About Sophia:</p> <ul style="list-style-type: none"> • Sophia is not a student at SFSU, however, she is visiting the school to see whether she would like to attend after she finishes high school. • Has decent experience in using tech and knows how to browse websites that sell items <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • Because she may become a student at SFSU later; Sophia has an okay idea of what classes she may take if she were to enroll; She would like to check the prices of books/notes for classes she may be interested in.
---	--

<p>Persona #3</p> <p>Daniel</p>  <p>Student at SFSU The Seller</p>	<p>About Daniel:</p> <ul style="list-style-type: none"> • He has been a student at SFSU for 3 years • He has collected a lot of school materials over the years as a student and wants to get rid of them • He has difficulty traveling, commutes to campus <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • From being a student for 3 years, Daniel has accumulated a lot of school material, such as textbooks, he no longer has a need for. He thought about throwing them away but then he thought other students might want some of the school material that he has. Knowing the cost of the material, Daniel believes he could sell what he has at a lower price so that other students will pay for it. • However, Daniel doesn't have an easy way to travel so he would want people to meet him for pick-up at a place that is accessible to him and the person he's selling to.
--	---

<p>Persona #4</p> <p>Isabell</p>  <p>Staff/Faculty at SFSU Offer TA jobs</p>	<p>About Isabell:</p> <ul style="list-style-type: none"> • Is a math professor at SFSU • Look for a TA to help with grading • Has difficulty with computers <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • Being a professor at SFSU, there's a lot of work that needs to be done so that students can have the best class experience. As a solution, Isabell wants to hire graders so she can focus on other aspects of the class. She wants the grader to be a SFSU student who has experience in the class she is teaching. • Isabell wants to post the position as a TA online but doesn't know how to navigate between programs well on the computer.
--	--

<p>Persona #5</p> <p>Tom</p>  <p>A student at SFSU Provides tutoring services</p>	<p>About Tom:</p> <ul style="list-style-type: none"> • A senior at SFSU, Tom has taken a number of high level CS courses and has done very well at them. • Doesn't have a job, but wants money to buy a new computer. • Has prior experience tutoring. <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • In his last semester at SFSU, Tom has learned a lot from the Computer Science courses he's taken so far. He wants to use the information he's learned so far to help others, but also wants to make some money in order to buy a new computer. He is interested in becoming a tutor but finds that the school doesn't pay the amount of money he'd like; however he deeply desires the ability to tutor on campus.
---	---

<p>Persona #5</p> <p>Jack</p>  <p>An Admin for GatorMart</p>	<p>About Jack:</p> <ul style="list-style-type: none"> • The admin user who has control of all the information released on the website. • Actively monitoring the item being posted • Notice the user when modifying or deleting the item posted. • He will send systematic notifications to all or a particular group of users. <p>Goals and Scenario:</p> <ul style="list-style-type: none"> • Jack starts his day by signing in to the GatorMart. He can see how many new items are posted on the website on the previous day. Then he will turn on the filter system to let him track the items that will need to modify potentially. He will do this the whole day once inappropriate items are posted. He will also choose the appropriate item to send a systematic notification to all the users to show them the good stuff that was posted in previous days.
--	--

Use-Cases

Use Case #1 - Dylan - Full Time student looking to buy Used Books

Dylan is a full-time student at SFSU and works a part-time job. Instead of buying his books new per semester, he prefers to buy used books as it saves him a lot of money. Normally, Dylan does not have enough time to leave campus or his job to pick up used books, so he resorts to ordering them online; which he finds is normally more expensive than picking them up in person. Additionally, Dylan is also someone who prefers to be very safe, so if he is picking up a used book, he would like it to be by a fellow student or by a seller who is reputable. By registering for our website as he is a student, he can use the app whenever he needs to buy a used book/item. Whenever he sees an item that he likes, he can send an offer he sees fit and a message to the seller. Additionally, Dylan's safety is also assured as once the seller and buyer are in agreement over an offer, he is able to select a meet-up spot on campus as to where he would like to pick up the book. Dylan is also able to look at the review of a seller and check whether they are reputable; in which he is also able to put his own review once the transaction is done.

Use Case #2 - Sophia - Not a student nor faculty at SFSU; Interested in browsing website

Sophia is neither a student nor staff at SFSU, however she is hopeful that she may be accepted to the college soon. She would like to browse our website to see how the prices listed for books/notes compare to other online sellers. Because she is an unregistered visitor, she is able to browse user listings but is unable to do a number of things such as: Put items for sale, send an offer, message users, and any other permissions allowed to users of the website. Additionally, any time she clicks on an item when she's browsing, she should be able to see any details related to that item but be blocked from seeing any of the seller's data (such as name, email, etc.).

Use Case #3 - Daniel - A 3rd year student looking to get rid of unneeded school materials

Daniel has been a student at SFSU for 3 years and has gathered a lot of school material he no longer has a use for. Instead of throwing away the school material, Daniel decided to see if he could give it to other students at a lower price than he got it for. Additionally, Daniel doesn't own anything like a car for easy travel so he wants a meeting place that's accessible to him and the person he's selling to. Through the website, Daniel is able to sell his school materials because he is a student at SFSU. He can make posts for each of his items and mark the price he wants to sell it at. Also, Daniel set the meeting point to be on campus as an accessible place for both him and the student he is selling to.

Use Case #4 - Isabell - A math professor at SFSU looking for a TA

Isabell is a math professor at SFSU and is looking for a TA as a grader to lighten the class load she has. She wants a SFSU student who has experience in the math classes she teaches to be her TA. Through the website, Isabell is able to make a job request for students to see. If a student is interested, they can contact her through her information on the job request page. Also, the website will provide easy navigation for professors to post job listings and these listing will be featured to the students so that students will be able to see them

Use Case #5 - Tom - A last year student at SFSU looking to provide tutoring services

Tom is a senior at SFSU in his last year. He's taken a number of high level Computer Science courses and is willing to tutor those who need it. However, he doesn't think the school pays the wage he would like, which is a shame because he really wants to tutor on campus. By using our website, Tom can register as he is a student. He can post a service to the user listings where he can talk about his tutoring services and the wage he would like to be paid. He can also choose a spot on campus where he would like to meet with whoever is interested. Once he fills out the post, if he is not registered, he will be asked to register and once he does, the post will be sent to an admin for approval. Once someone is interested in his service, they could hammer out the details in messages.

Use Case #6 - Jack - An Admin for GatorMart

Jack is the admin user who has control of all the information released on the website. He is an employee of GatorMart Company. His goal is to monitor the content quality and send a systematic notification to all the users. He will constantly check the filter which gives him the item that needs to be changed or deleted immediately. Once there is an update note for the GatorMart, or there are some items that need to be promoted, Jack will send them as a systematic notification to all the users.

III. Data Items and Entities

Entities	Data Items	Permissions
Unregistered Visitor		View user listings (services, selling, buying, surveys, job listings).
User	UserID, email, password, User Type (student/teacher/admin) profile, fullname, profile picture, payment method, buyer rating, seller rating	List services, items for sale, items they're looking to buy. Make offers / buy items, privately message users after making an offer.
Student (inherits user)	Student ID, major, minor, student standing (undergraduate/graduate and lower/upper division)	
Staff (inherits user)	Department, job title	List job/help positions, surveys (w/ or w/o reward),
Site Administrator (subclass of user)		Remove/Approve listings, remove users, view other users' information, feature items
Website	URL, server, database	

Registration	List of items, List of users,	
Items	Post ID, cost, item ID, image, tags, item name, description	
Message	Sender, recipient, date, time, status	
Reviews	Posted by, date, time	
Meet-up spots	Location ID, marker	
Offer	Offer ID,	
Notifications		

IV. List of Functional Requirements

1. **View Marketplace** - All users shall be able to view the full marketplace.

Unregistered users will not be able to make offers or purchase items.
2. **Register Account** - Unregistered visitors shall be able to register for an account with their active sfsu.edu email address and a desired password. This email address will be verified via a link in an email. Registered users will have more privileges than unregistered, as described in the Data Items and Entities 'Permissions' column. Additional optional information will be requested, such as major, minor, and interests.
3. **Log in** - Registered users shall be able to log in to our site using the information they registered with. This will allow them the privileges of registered users.
4. **List item for sale** - Registered users shall be able to post an item they would like to sell with a title, a description, a price, and, optionally, tags and photos. They shall also be able to decide what information of theirs is made public on the listing (i.e., do they want their email/phone number/name on the listing). Some form of contact information must be available.
5. **List Item Needed** - Registered users shall be able to list items they are looking to purchase. Required information includes a title, description, desired buying price, and, optionally, tags and photos.
6. **List Service (Needed)** - Registered users shall be able to list services they need help with. Payment for these types of transactions will happen off-site. Service listings must have a title, description, a desired price, and some form of contact information (implementing private messaging is a tier 2 priority), and, optionally, tags and photos.

7. **List Service (Provided)** - Registered users shall be able to list services they are able to provide to other students. Payment for these transactions will happen off-site. Provided services must have a title, description, desired pay, and some form of contact information, and, optionally, tags and photos.
8. **Make an Offer** - Registered users shall be able to make offers on items available for purchase. This must be an offer below the listed price; the site will not host auctions. Offers can be accepted or declined by the user who listed the item (when on-site private messaging is implemented, users will be able to include messages with their offers).
9. **Purchase** - Registered users shall be able to purchase an item for its listed price. A notification of this will be sent to the seller with the buyer's contact information.
10. **Accept or Decline Offer/Purchase** - Registered users who have listed an item must be able to accept or decline an offer or purchase. If they accept the offer, the buyer and seller must determine a pickup location and payment method off-site.
11. **Rate Transaction (Buyer)** - Buyers will be able to rate transactions (from 1 to 5 stars, with a description) after they have taken place. These ratings will be listed on the opposing user's profile.
12. **Rate Transaction (Seller)** - Similar to the last list-item, the only difference being that posters will be able to rate their experience with the responder. This rating will also show up on the opposing user's profile.
13. **Search Marketplace** - All users shall be able to search the marketplace via a typed search-bar. Priority 1 will be simply getting the Regex to match exact strings to find items, and priority 2 will be getting the regex to match misspelled items (fuzzy search).

14. **Filter Listings** - All users shall be able to filter the marketplace by tags placed on listings. This way, users can more properly find the items / services they're looking for.
15. **View Profile** - All users shall be able to view other users' profiles. The user profile will show any information the user has made public (i.e., contact information, name, etc.), and will always show their poster / responder ratings as well as the other items they have listed.
16. **Remove Listing** - Site administrators will be able to remove listings which have already been made public if there is a good reason to do so. This will permanently remove that listing from the marketplace, and users will no longer be able to interact with it.
17. **Approve / Disapprove Listing** - Site Administrators must review listings before they are made public on the site. If the content of the listing is not approved, the administrator will be able to prevent that item from being uploaded to the site. A second priority would be adding the ability to include a message (written by the administrator) informing the user on the reason for their listing being removed.
18. **Remove User** - Administrators shall be able to permanently remove a user from the website. This will block them from signing up to the site in the future. Their username and email will remain in the database, but will permanently be flagged as unable to login. When the previously registered user attempts to log in in the future, they will be signaled of their account's removal.
19. **Report Listing** - All users shall be able to report a listing which contains content they find offensive/harmful/etc. This will notify the administrators that a post has been flagged, and allow them to decide whether to keep it up or remove it permanently.

20. Change User Settings - Registered users shall be able to modify their settings (contact information, major, description, etc.) after their account has been created.

V. List of Non-functional Requirements

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. All or selected application functions must render well on mobile devices
4. Data shall be stored in the database on the team's deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected
7. The language used shall be English (no localization needed)
8. Application shall be very easy to use and intuitive
9. Application should follow established architecture patterns
10. Application code and its repository shall be easy to inspect and maintain
11. Google analytics shall be used
12. No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging. One round of messaging (from user to seller) is enough for this application
13. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
14. Site security: basic best practices shall be applied (as covered in the class) for main data items
15. Media formats shall be standard as used in the market today
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

17. The application UI (WWW and mobile) shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2022. For Demonstration Only" at the top of the WWW page nav bar. (Important so as to not confuse this with a real application).

VI. Competitive Analysis

Feature	Craigslist	eBay	Amazon	Facebook Marketplace	Our Future Product
Text Search	+	+	++	+	+
Browse	-	+	+	+	++
Sell/Purchase Products and Services	+	-	-	-	+
List/Apply Jobs	+	-	-	-	+
Pickup Option	+	-	-	+	+

+ feature exists; ++ superior; - does not exist

Summary:

GatorMart is an online hub that we plan to develop exclusively for SFSU students, faculty, and staff to buy and sell products and services. One of the planned advantages of our product compared to the competitors is that registered users can buy and sell not only products on the website, but also services. Our competitors such as eBay, Amazon, and Facebook Marketplace only allow registered users to buy and sell products. To provide a convenient and secure platform for the SFSU community, registered users on GatorMart will pick up the items or have the service done on campus. While eBay and Amazon only provide shipping options. Another planned competitive advantage is job listing. Registered faculty can list job openings, which will eventually make GatorMart an go-to online hub for SFSU community.

VII. High-Level System Architecture and Technologies Used

21. Server Host: Amazon AWS EC2 T3 (t3.small, 12 months free package):

- i. 2 vCPU w/ 24 CPU Credits/hour
- ii. 2 GiB Memory
- iii. EBS-Only Storage
- iv. Up to 5Gbps

22. Operating System: Ubuntu 20.04 LTS Server

23. Database: MySQL v8.0

24. Web Server: Apache httpd 2.4

25. Server-Side Language: Javascript ES12

26. Additional Technologies:

- i. Node.js and Express.js
- ii. Sequelize for communication between JS and MySQL
- iii. VSCode

VIII. Team and Roles

Name	Role	Github Name
Shane Waxler	Team Lead	ShaneWaxler1
Robert Garcia	Back-End Lead	RGarcia35
Joe Guan	Back-End	Joeguan1
Melissa Ho	Front-End	melissah717
Minggu Ma	GitHub Master	Macgoogle
Chuting Yan	Front-End Lead	cctina516
Xiaoqing Yao	Front-End	yao961002

IX. Checklist

- III. So far all team members are engaged and attending ZOOM sessions when required
DONE
- IV. Team found a time slot to meet outside of the class **DONE**
- V. Back end, Front end leads and Github master chosen **DONE**
- VI. 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**
- VII. Team lead ensured that all team members read the final M1 and agree/understand it before submission **DONE**
- VIII. Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.) **DONE**

Software Engineering 648/848 Spring 2022

Milestone 2

03/19/2022

Team #2 - GatorMart

Shane Waxler - Team Lead - swaxler@mail.sfsu.edu

Chuting Yan - Front-End Lead

Robert Garcia - Back-End Lead

Minggu Ma - GitHub Master

Melissa Ho - Front-End

Xiaoqing Yao - Front-End

Joe Guan - Back-End

Date submitted for review	Date revised (after feedback)
03/20/2022	3/31/2022

I. Executive Summary

In most hallways on campus at San Francisco State University, there's multiple bulletin boards looking to buy or sell goods and services. Colorful pieces of paper litter these boards.. A person looking for a particular item or service could potentially spend hours trying to find one item. Not only is this time consuming and frustrating, these efforts could prove to be futile. In the fall of 2020, San Francisco State University enrolled 24,025 students. With a large and diverse body of staff, students, and faculty members, the university lacks a centralized hub for them to buy, and sell goods and services. We have developed a website service, GatorMart which provides this service exclusively to current students, faculty, and staff.

GatorMart is a vast online “bulletin board” that is organized into specific categories, allowing registered users of the SFSU community to quickly find a specific item. To register, all one needs is a sfsu.edu email that confirms they are a part of the San Francisco State community. Once a user’s registration has been confirmed, they are free to browse various categories for the items or services they are seeking. These include old textbooks, tutoring services, electronics, and job listings. The bulletin board is maintained by an administrator who must approve all postings before they go public. This way, we are able to filter out any inappropriate or spam postings. The administrator’s responsibilities also include confirming a user as a current staff or student of the university. A user can be distinguished by badges on their profile, indicating whether they are part of the staff or student body. Specifics of how a user can fully utilize GatorMart can be found in the following sections.

As former members of the SFSU community, this project means more convenience and accessibility for current and future members of the community. Our team believes that GatorMart will pave the way for a streamlined online community of users at San Francisco State University to purchase, and sell.

II. List of Main Data Items and Entities

Entities	Data Items	Definition
Unregistered Visitor		View user listings (i.e., services, selling, buying, job listings) and listing details. Create Account
Registered User	UserID, email, password, User Type (student/teacher/admin) profile, fullname, profile picture, payment method, buyer rating, seller rating.	Log in Every category of data that a user can view/access, post item and message on any operation of the website.
Registration Record	UserID, password, full name	Required data for registering as a user.
Items	Post ID, cost, item ID, image, tags, item name, description, date, owner.	These data define an item.
Message	Sender, recipient, date, time, status	These data categories will be stored when sending messages to other users.
Reviews	Posted by, date, time	These date categories will be stored when posting and

		viewing reviews.
Meet-up spots	Location ID, marker	These data categories will be stored when users are using the meet-up spot function.
Offer	Offer ID	This data category will be stored when the new offer is made.

III. Prioritized Functional Requirements

Priority: 1	
Unregistered User	<p>2) Register Account - Unregistered visitors shall be able to register for an account with their active sfsu.edu email address and a desired password. This email address will be verified via a link in an email. Registered users will have more privileges than unregistered, as described in the Data Items and Entities 'Permissions' column. Additional optional information will be requested, such as major, minor, and interests. This is priority 1, but remains secondary to listing items.</p> <p>1) View Marketplace - All users shall be able to view the full marketplace. Unregistered users will not be able to make offers or purchase items.</p> <p>13) Search Marketplace (Simple Regex) - All users shall be able to search the marketplace via a typed search-bar. Priority 1 will be simply getting the Regex to match exact substrings to find items.</p> <p>14) Filter Listings - All users shall be able to filter the marketplace by category placed on listings as well as string search. This way, users can more properly find the items / services they're looking for.</p>
Registered User (Inherit all above)	<p>3) Log in - Registered users shall be able to log in to our site using the information they registered with. This will allow them the privileges of registered users.</p> <p>4) List item for sale - Registered users shall be able to post an item they would like to sell with a title, a description, a price, and, optionally, tags and photos. They shall also be able to decide what information of theirs is made public on the listing (i.e., do they want their email/phone number/name on the listing). Some form of contact information must be available.</p> <p>7) List Service (Provided) - Registered users shall be able to list services they are able to provide to other students. Payment for these transactions will happen off-site. Provided services must have a title, description, desired pay, and some form of contact information, and, optionally, tags and photos.</p> <p>9) Message - Registered users shall be able to message the seller. A notification of this will be sent to the seller with the buyer's contact information.</p>
Admin	<p>16) Remove Listing - Site administrators will be able to remove listings which have already been made public if there is a good reason to do so. This will permanently remove that listing from the marketplace, and users will no longer be able to interact with it.</p>

	<p>17) Approve / Disapprove Listing - Site Administrators shall review listings before they are made public on the site. If the content of the listing is not approved, the administrator will be able to prevent that item from being uploaded to the site. A second priority would be adding the ability to include a message (written by the administrator) informing the user on the reason for their listing being removed.</p> <p>18) Remove User - Administrators shall be able to permanently remove a user from the website. This will block them from signing up to the site in the future. Their username and email will remain in the database, but will permanently be flagged as unable to login. When the previously registered user attempts to log in in the future, they will be signaled of their account's removal.</p>
--	---

Priority: 2	
Registered User	<p>5) List Item Needed - Registered users shall be able to list items they are looking to purchase. Required information includes a title, description, desired buying price, and, optionally, tags and photos.</p> <p>6) List Service (Needed) - Registered users shall be able to list services they need help with. Payment for these types of transactions will happen off-site. Service listings must have a title, description, a desired price, and some form of contact information (implementing private messaging is a tier 2 priority), and, optionally, tags and photos.</p> <p>8) Make an Offer - Registered users shall be able to make offers on items available for purchase. This must be an offer below the listed price; the site will not host auctions. Offers can be accepted or declined by the user who listed the item (when on-site private messaging is implemented, users will be able to include messages with their offers).</p> <p>15) View Profile - All users shall be able to view other users' profiles. The user profile will show any information the user has made public (i.e., contact information, name, etc.), and will always show their poster / responder ratings as well as the other items they have listed.</p> <p>19) Report Listing - All users shall be able to report a listing which contains content they find offensive/harmful/etc. This will notify the administrators that a post has been flagged, and allow them to decide whether to keep it up or remove it permanently.</p>

Priority: 3

Registered User	<p>11) Rate Transaction (Buyer) - Buyers will be able to rate transactions (from 1 to 5 stars, with a description) after they have taken place. These ratings will be listed on the opposing user's profile.</p>
	<p>12) Rate Transaction (Seller) - Similar to the last list-item, the only difference being that posters will be able to rate their experience with the responder. This rating will also show up on the opposing user's profile.</p>
	<p>13.5) Search Marketplace (Fuzzy) - Search shall be improved with a more advanced search algorithm. Excluding unlikely characters, permutation matching, etc.</p>
	<p>20) Change User Settings - Registered users shall be able to modify their settings (contact information, major, description, etc.) after their account has been created.</p>

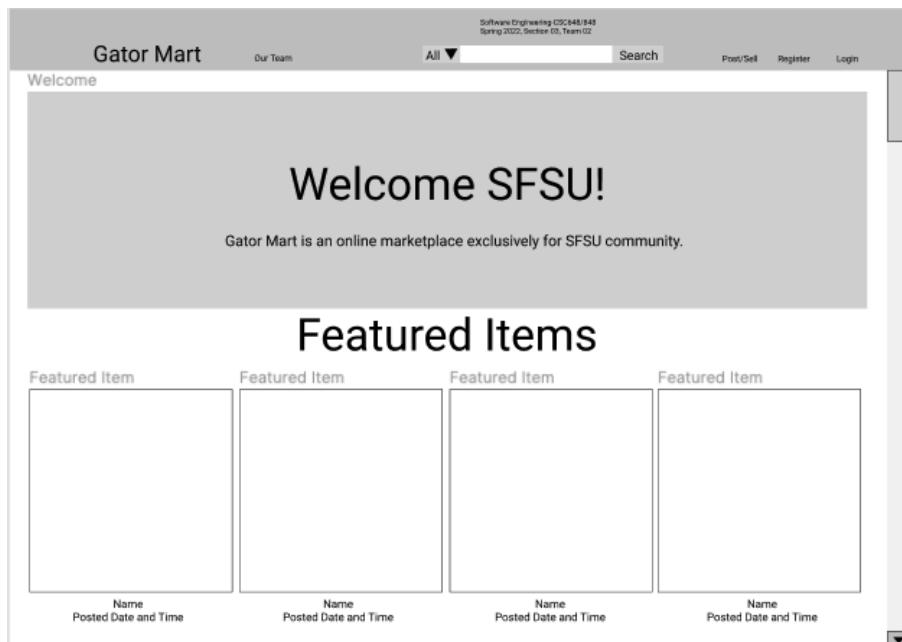
IV. UI Storyboards for Main Use Cases

Use Case #1 - Dylan

Dylan is a full-time student at SFSU and works a part-time job. He prefers to buy used books every semester. He doesn't have time to leave campus or his job to pick up used books, so he resorts to ordering them online. But he finds it's more expensive to order online than picking them up in person.

He finds GatorMart a great place for him to buy used books, as GatorMart is an online hub exclusively for the SFSU community. All the used books he needs can be found on GatorMart.

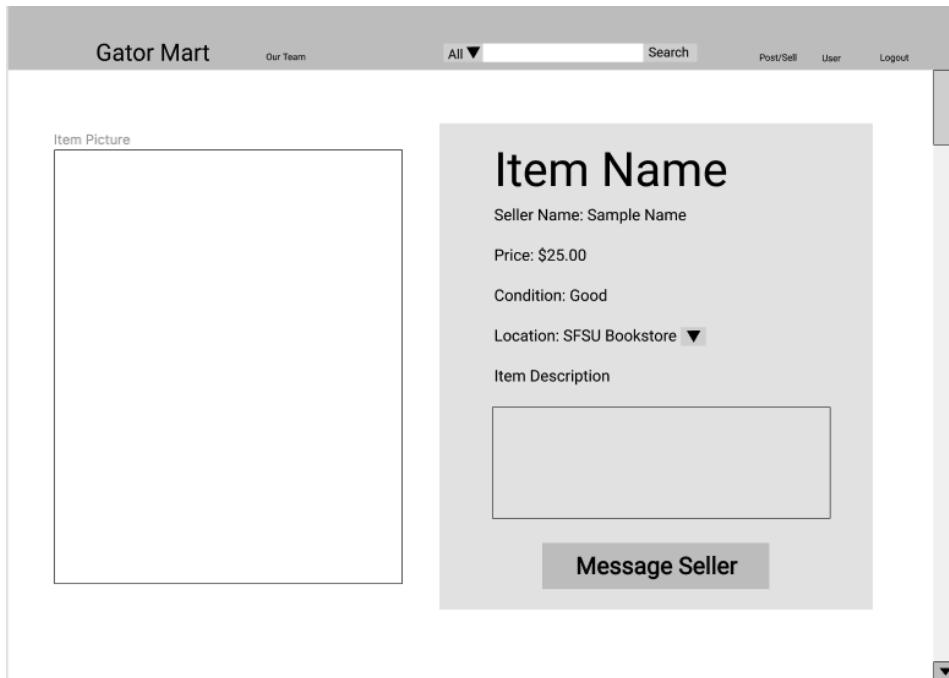
He visits GatorMart's home page. He sees featured items from various categories.



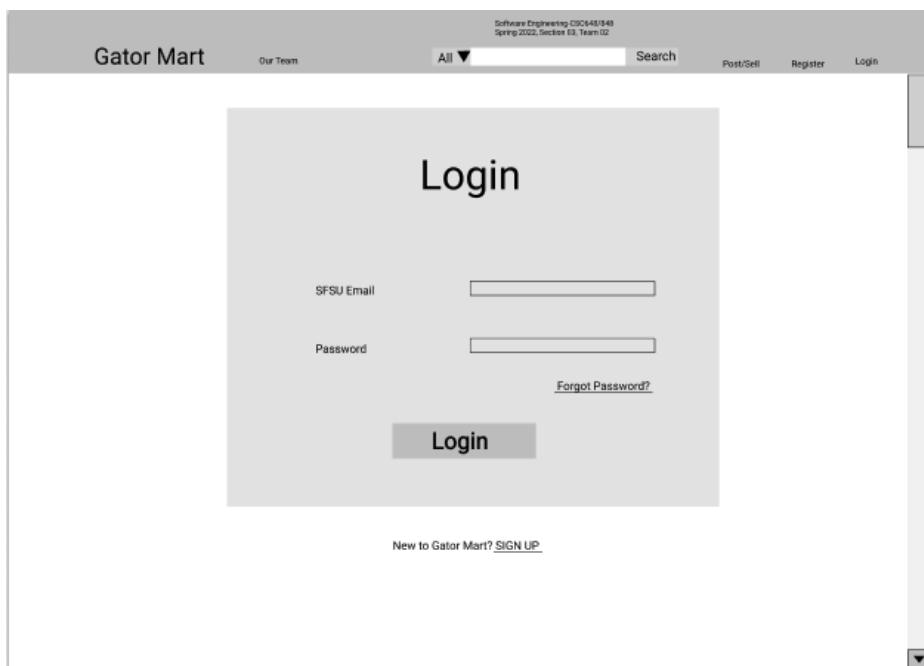
Then he types in the book name he is looking to buy in the search bar. The books that match the keywords show:

The screenshot shows a web browser displaying the Gator Mart website. At the top, there is a navigation bar with links for "Our Team", "All" (with a dropdown arrow), "Search", "Post/Sell", "Register", and "Login". Below the navigation bar, a message reads "1-6 of 6 results for 'Intro to CS'". A "sort by price ▼" button is also present. The main content area displays six search results, each represented by a large rectangular box labeled "matching item". Underneath each box, there are two smaller boxes showing "Name" and "Posted Date and Time". The results are arranged in two rows: the first row contains four items, and the second row contains two items.

The first book looks like the one he needs. He clicks on the first book to see more detail about it.



It looks like this is exactly the book he wants. He wants to send a message to the seller to purchase the book, so he clicks the “Message Seller” button. Since he is not logged in, it prompts him to login or register:



Since he doesn't have an account with Gator Mart. He clicks the register button to register as a new user:

Software Engineering CSD448/B448
Spring 2022, Section 03, Team 02

All ▼ Search Post/Sell Register Login

Registration

Full Name*

SFSU Email*

Username / Display Name*

Password*

Confirm Password*

Register

Already have an account? [LOGIN here](#)

After he registers, the website directs him back to the previous page.

Gator Mart Our Team All ▼ Search Post/Sell User Logout

Item Picture

Item Name

Seller Name: Sample Name

Price: \$25.00

Condition: Good

Location: SFSU Bookstore ▼

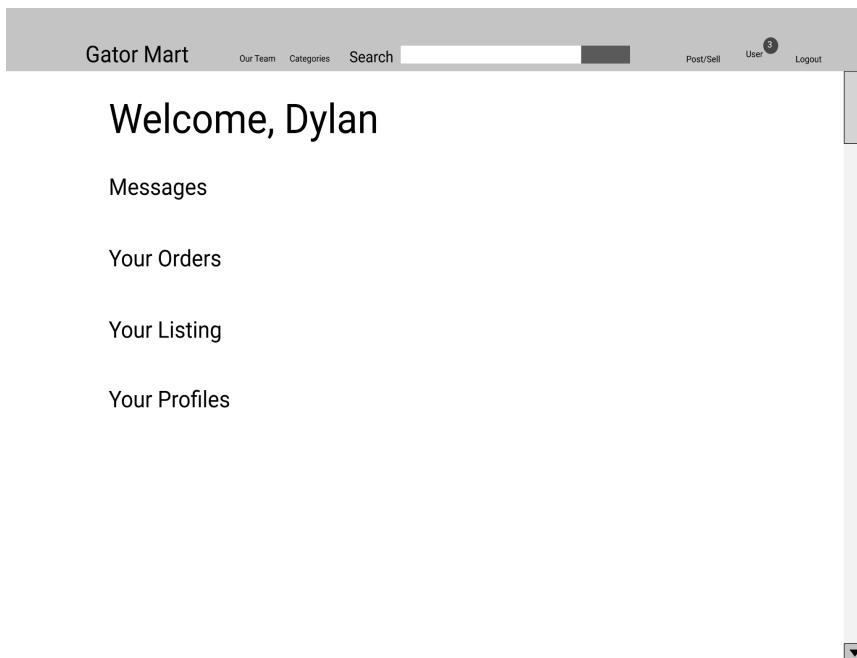
Item Description

Message Seller

He selects the meeting location as SFSU Bookstore. He clicks the "Message Seller" button and message the seller if they can meet tomorrow at 12pm and he offers to pay cash.



Later that day, he logs into Gator Mart to see if the seller replies to him. He sees that he has one notification. He clicks on the notification and it directs him to his dashboard.



He sees the seller agrees to his term. He doesn't reply anything more and just closes the website.

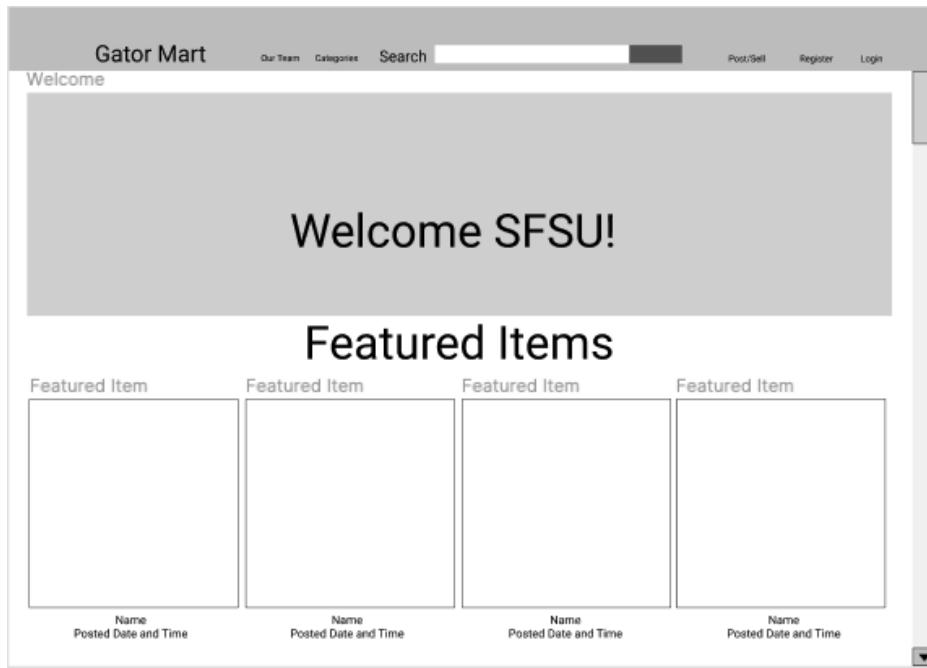


The next day, he and the buyer meet at the time and place they agreed on. He purchases the book and he is happy with the experience he has with Gator Mart.

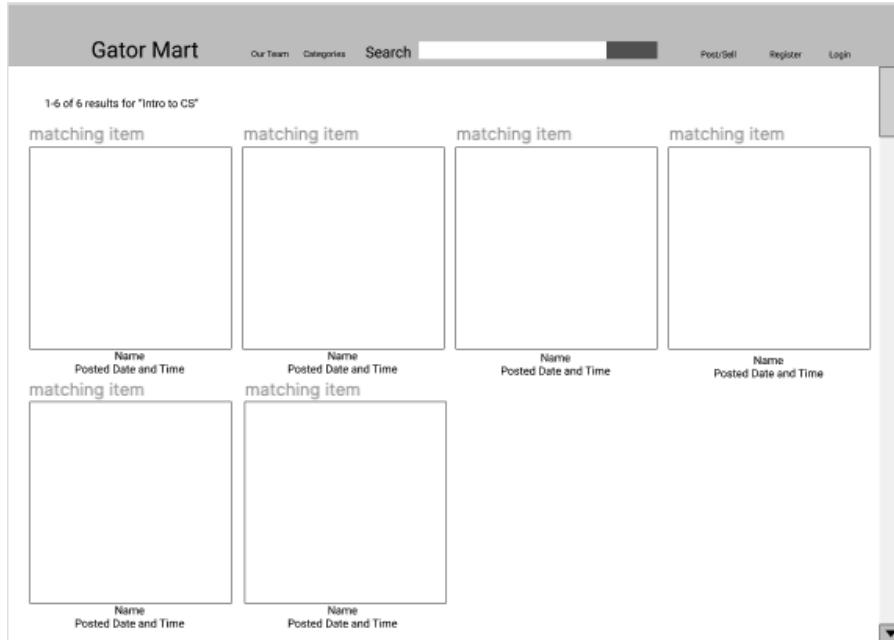
Use Case #2 – Sophia

Sophia is neither a student nor a staff at SFSU, however she is hopeful that she may be accepted to SFSU soon. She has an education plan and some classes that she is interested in taking after successfully enrolled. She heard of the good reputation of Gator Mart as the online hub for SFSU community. So she would like to check the prices of books/notes for the classes that she is interested in or planned to take.

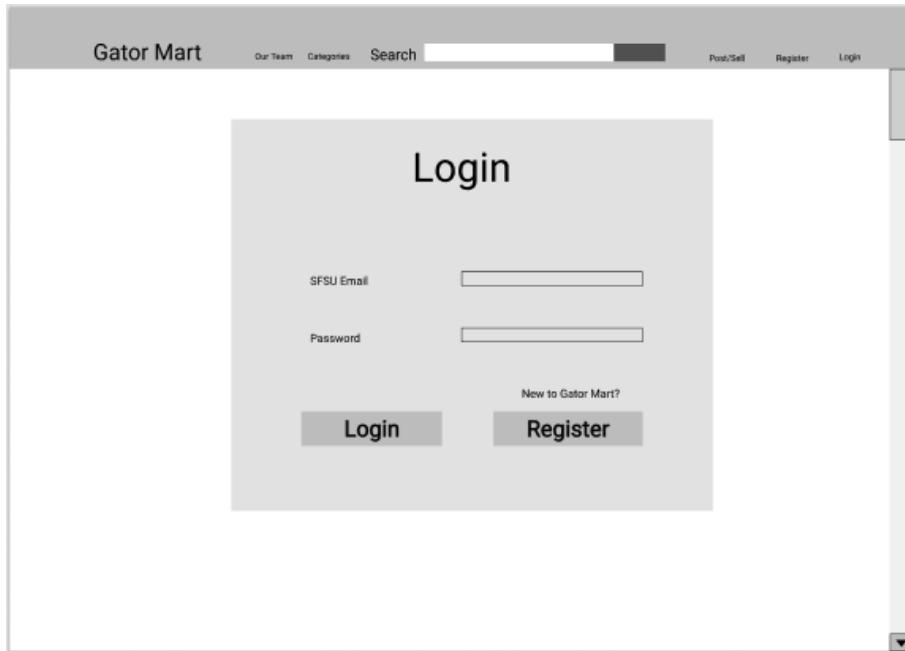
She visits GatorMart's home page. She sees featured items from various categories.



She browses the homepage and searches one of the books for the class that she is interested in. The matching results for the book shows.

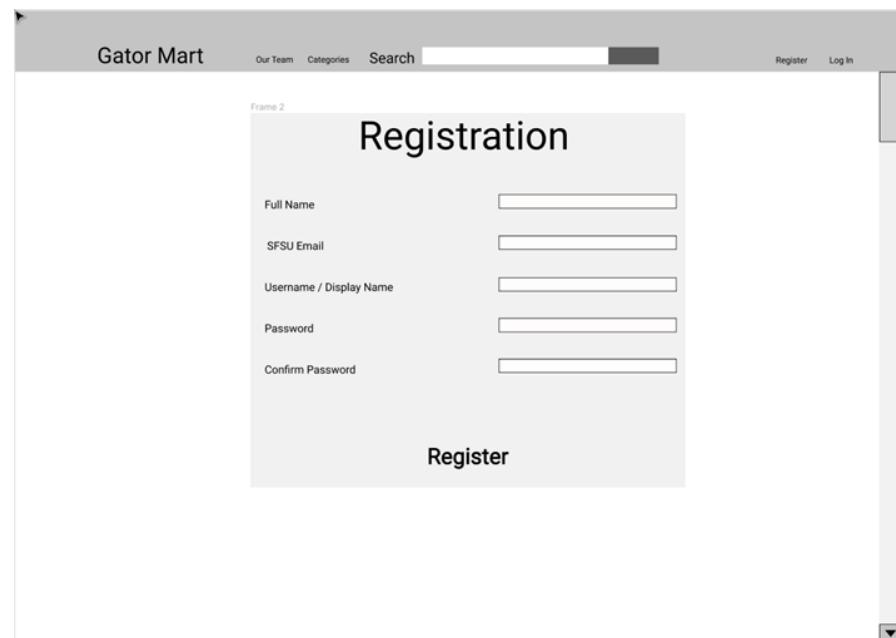


She clicks on the second book to see more detail. As she is not logged in, the website prompts her to login or register.



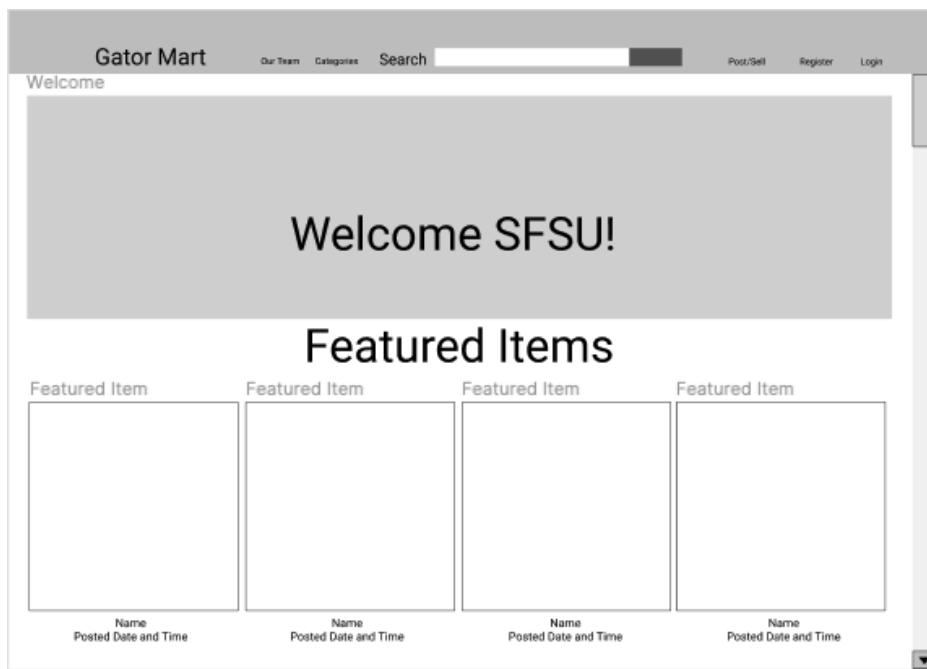
The screenshot shows the Gator Mart login page. At the top, there is a navigation bar with links for "Our Team", "Categories", "Search", "Post/Sell", "Register", and "Login". The main content area has a light gray background and features a large "Login" heading. Below it are two input fields: one for "SFSU Email" and one for "Password". To the right of the password field is a link "New to Gator Mart?". At the bottom of the form are two buttons: "Login" and "Register".

Since she doesn't have an account with Gator Mart, she clicks on the register button. Yet she finds out that she will need a SFSU email to register.



The screenshot shows the Gator Mart registration page. At the top, there is a navigation bar with links for "Our Team", "Categories", "Search", "Register", and "Log In". The main content area has a light gray background and features a large "Registration" heading. Below it are five input fields: "Full Name", "SFSU Email", "Username / Display Name", "Password", and "Confirm Password". At the bottom of the form is a single "Register" button.

Since she is not a SFSU student yet, she can't register for now without the SFSU email. She clicks back to the home page to browse the items.

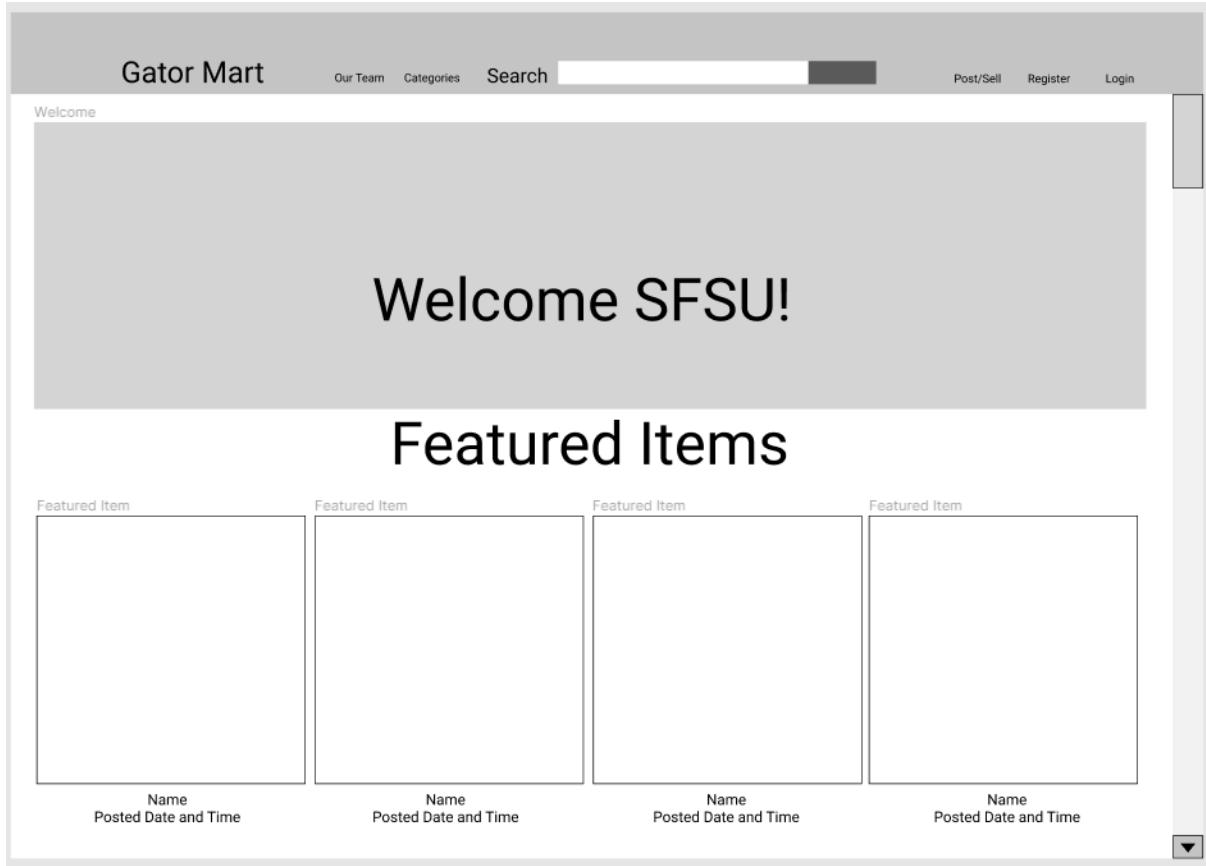


Seeing all the items available on the website, she knows that she will be a regular user of Gator Mart once she becomes a SFSU student.

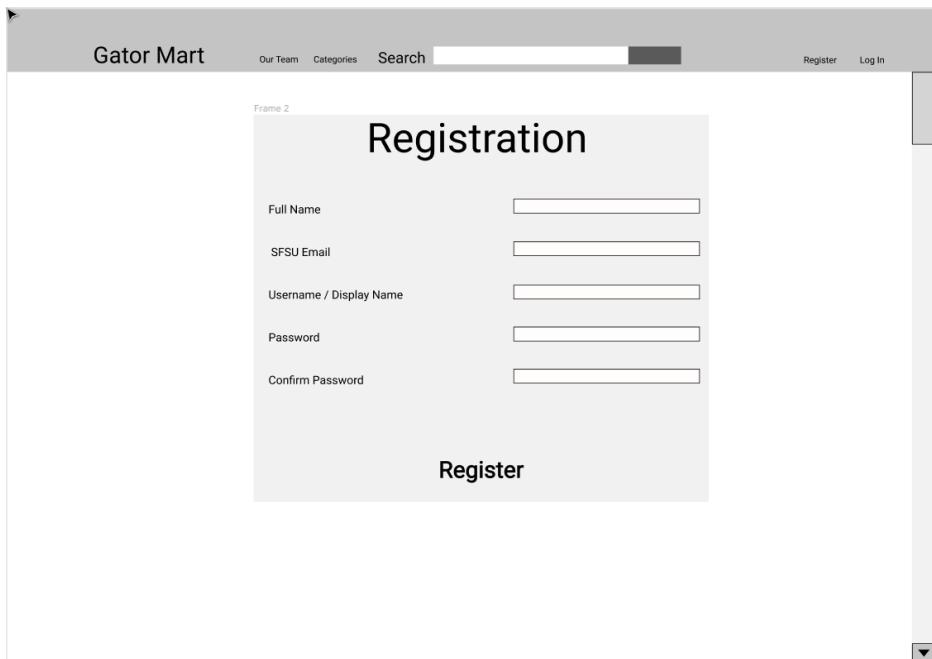
Use Case #3 - Daniel

Daniel is a 3rd year SFSU student who has accumulated many books over the years. Knowing it cost him a lot of money to purchase these materials, he wants to sell them instead of throwing them away. However, he doesn't have an easy mode of transportations since he has to commute to campus. He needs to meet at locations that are convenient to him on campus.

Daniel visits GatorMart, and on the homepage, he sees some recently popular items. Some of them are textbooks. He decides that this marketplace centered around SFSU is a perfect platform to post his books.

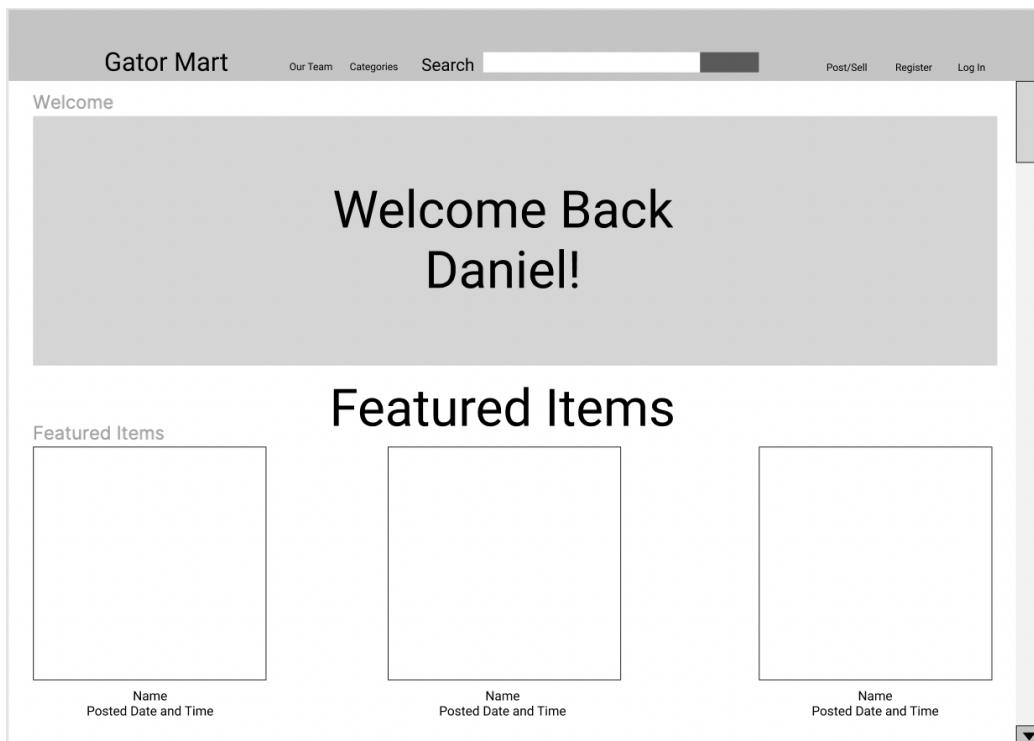


He clicks the register link on the top right, and is prompted to go to make an account. It gives him peace of mind that he has to register with his SFSU email.



The screenshot shows a registration form titled "Registration" within a "Frame 2" frame. The form includes fields for Full Name, SFSU Email, Username / Display Name, Password, and Confirm Password, each with an associated input field. A "Register" button is located at the bottom of the form. The top navigation bar of the website includes links for "Our Team", "Categories", "Search", "Register", and "Log In".

After the admin approves his account, he can log in. The user home page asks him if he would like to post/sell.



He clicks “post/sell”. It directs him to the post/sell page, which prompts him for information about his item. He uploads a picture of “Transcendental Calculus”, and in the location field puts down that he would like to meet up at the Cesar Chavez Center or outside Thornton Hall.

Gator Mart Our Team Categories Search Post/Sell User Logout

POST ITEM FOR SALE

*Item Name:

*Category:

*Price:

*Condition:

Location(s):

*Upload Image:

Item Description:

May take up to 24 hours for approval

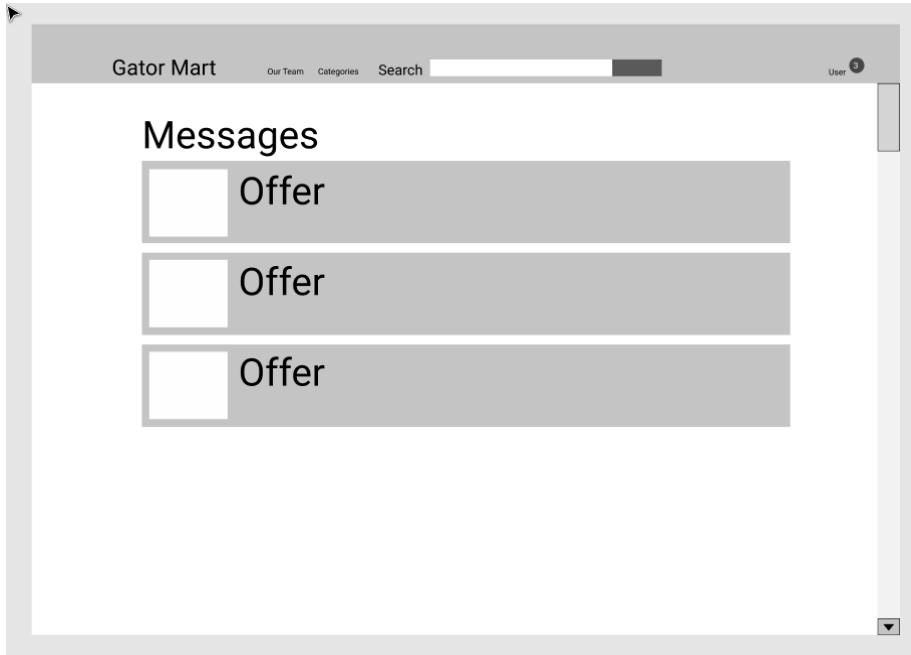
Wanting to make sure his post was successful, he searches “Transcendental Calculus” and sorts by most recent posts. His post is the first one. He is satisfied.

Gator Mart Our Team Categories Search Post/Sell User Logout

Filter by: Most Recent

Home page item 5	Home page item 6	Home page item 7	Home page item 8
Name Posted Date and Time			
Home page item 9			
Name Posted Date and Time			

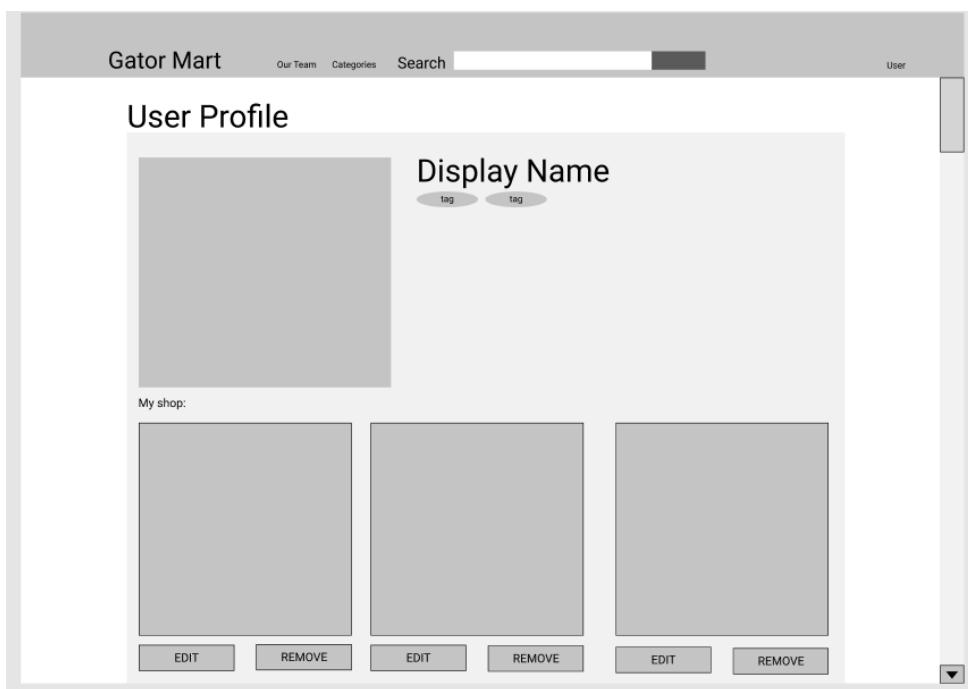
The next day, David can check if he has received any messages by clicking on messages in the navigation bar and he'll be directed to the messages page. There, if he has received any messages, they will show up and he can click on them to see the message.



Daniel reads the message sent by the person making the offer. They ask if they can pick it up on the steps of Thornton Hall the next day, and that they would be paying with cash. Daniel prefers cash, so he accepts the offer.



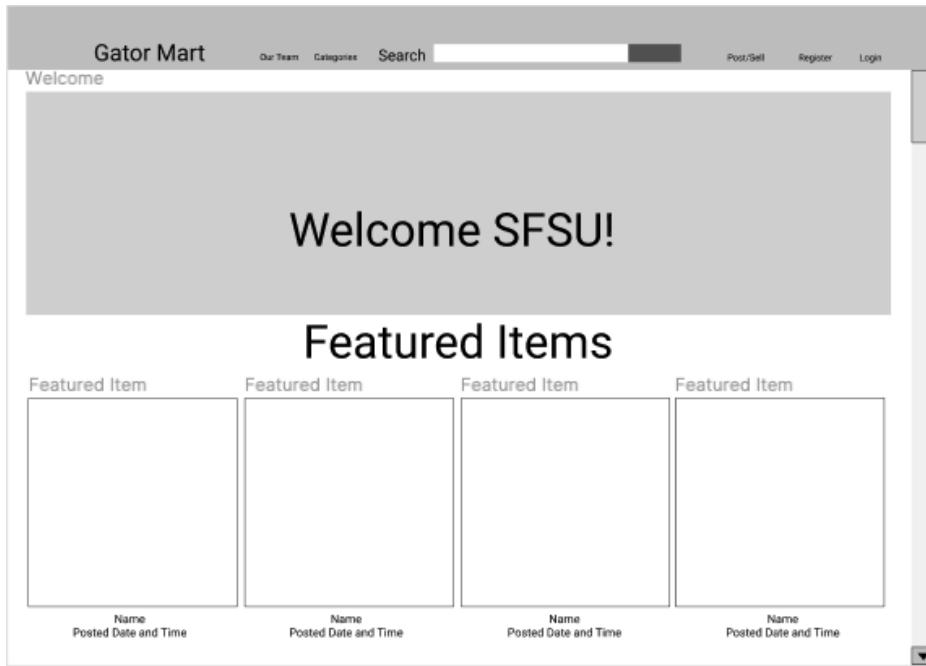
The next day, he meets the buyer and the transaction is complete. He logs into GatorMart and removes the listing from the marketplace, so people don't message him anymore about the Calculus textbook.



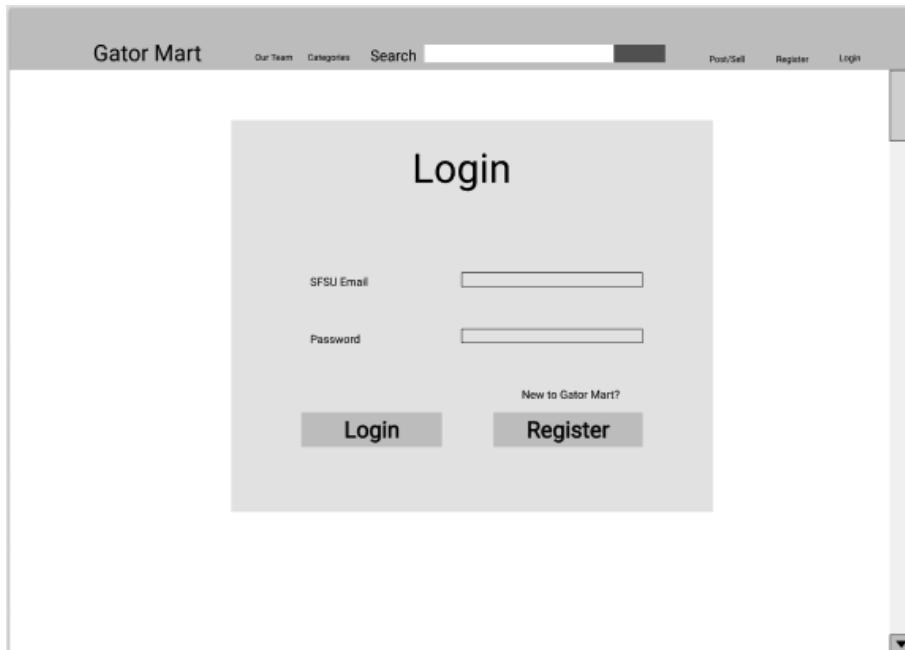
Use Case #4 – Isabell

Isabell is a math professor at SFSU and is looking for a TA as a grader to lighten the class load she has. She wants a SFSU student who has experience in the math classes she teaches to be her TA. She finds GatorMart a great place for her, as the users for the website are all part of the SFSU community.

She visits GatorMart's home page. She sees different items, services and job posts listed there.

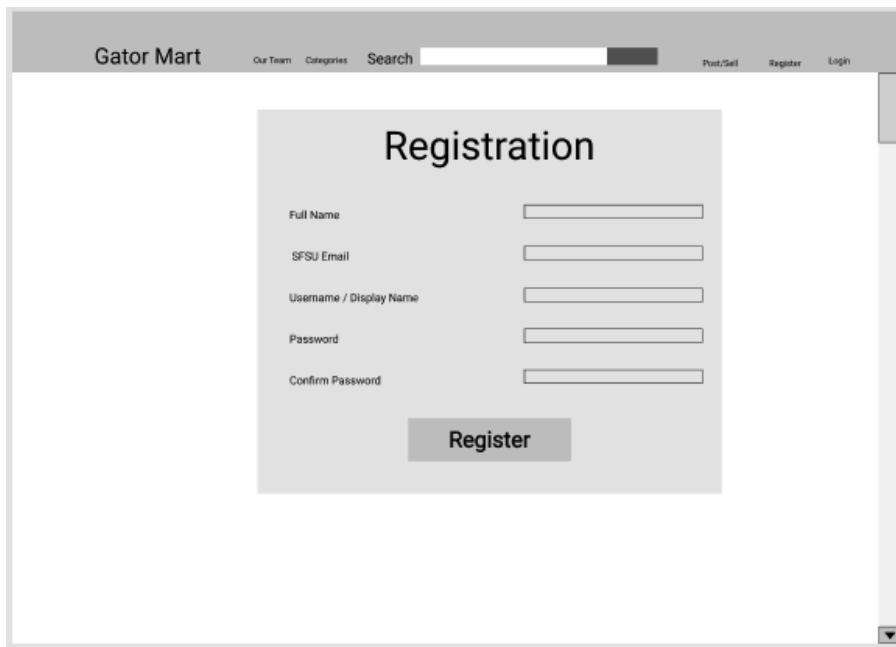


She sees the post/sell button on the top right and clicks it. Since she's not logged in yet, the website prompts her to log in or register.



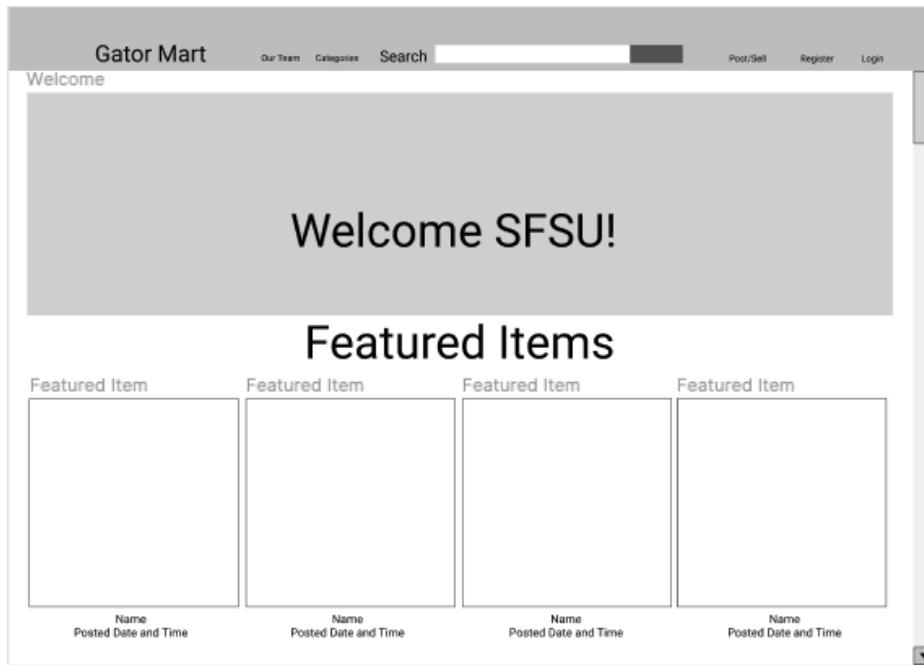
The screenshot shows the Gator Mart login page. At the top, there is a navigation bar with links for "Our Team", "Categories", "Search", "Post/Sell", "Register", and "Login". The main content area has a light gray background and features a large "Login" heading at the top center. Below it are two input fields: one labeled "SFSU Email" and another labeled "Password", each with a corresponding text input box. To the right of the password field is a link "New to Gator Mart?". At the bottom of the form are two buttons: "Login" on the left and "Register" on the right.

Since she doesn't have an account with Gator Mart. She clicks the register button to register an account.



The screenshot shows the Gator Mart registration page. The layout is similar to the login page, with a navigation bar at the top. The main content area has a light gray background and features a large "Registration" heading at the top center. Below it are five input fields: "Full Name", "SFSU Email", "Username / Display Name", "Password", and "Confirm Password", each with a corresponding text input box. At the bottom of the form is a single "Register" button.

After the admin approves her account, she can log in. The website directs her back to the previous page, the home page. She clicks on the post/sell button again.



It directs her to the post page, which prompts her for information about her post. First, she puts “Math 325 TA” in the item name box as the keyword. She uploads SFSU’s logo as the image, and chooses the correct category. Then she puts all the job description, and her contact information in the description box.

Since it’s a job posting, she would like to have the students email her their CVs and transcripts via email. Thus, she writes in the description to contact her via email. She also puts her SFSU email in the location text box.

Gator Mart

Our Team Categories Search

Post/Sell User Logout

Frame 9

Upload Image:

Item Name:

Category ▾

Price:

Condition:

Location(s):

Item Description:

She wants to make sure her post was successful, she searches “Math 325 TA” and sorts by most recent posts. Her post is the first one. She is happy.

Gator Mart

Our Team Categories Search

Post/Sell User Logout

Filter by: ▾ Most Recent

Home page item 5

Home page item 6

Home page item 7

Home page item 8

Name
Posted Date and Time

She is impressed by the user experience of Gator Mart. She closes the website, and she is confident that she'll find a TA soon.

V. High Level Architecture, Database Organization

Media Storage: Our team will be storing images in the file system and storing links to those images in the DB. This way, video and image upload/retrieval can be handled with the same function.

DB Schema:

User:

Column	Datatype	Valid Entries
id (PK)	INT	(set to auto_increment from 0)
email	VARCHAR(255)	Exclusively “*.sfsu.mail.edu” or “*.sfsu.edu”
pw	VARCHAR(255)	Any string encrypted with SHA256.
bio	VARCHAR(255)	“My name is... I am a... I enjoy...”
phone_number	VARCHAR(255)	“+1-000-000-0000”, or “+1-000-000-0000 ext 0000”

Review:

Column	Datatype	Valid Entries
id (PK)	INT	(set to auto_increment from 0)
user_id (FK)	INT	1, 2, 3, etc.
post_id (FK)	INT	1, 2, 3, etc.
direction	VARCHAR(255)	Exclusively “buyer” or “seller”
review	VARCHAR(255)	“The transaction went... I would recommend buying from...”
rating	INT	Limited from 1 to 5 (i.e., a 1

		to 5 star rating).
--	--	--------------------

Post:

Column	Datatype	Valid Entries
id (PK)	INT	(set to auto_increment from 0)
user_id (FK)	INT	1, 2, 3, etc.
title	VARCHAR(255)	Title of post
category (FK)	INT	1, 2, 3, etc.
available	NUMBER(1) (essentially a boolean)	Exclusively 0 or 1.
description	VARCHAR(255)	“This is a new chair with...” etc.
price	FLOAT	34.50, 0.00, 900.12, etc.
type	VARCHAR(255)	Exclusively “service” or “product”.
direction	VARCHAR(255)	Exclusively “requesting” or “providing”.

Categories:

Column	Datatype	Valid Entries
id (PK)	INT	(set to auto_increment from 0)
category	VARCHAR(255)	“Books”, “School Supplies”, “Services”, ect.

Messages:

Column	Datatype	Valid Entries
id(PK)	INT	(set to auto_increment from 0)

body	VARCHAR(10000)	“Hello, my name is Daniel and I would like to...” etc.
post_id	INT	1, 2, 3, etc.
sender_id	INT	1, 2, 3, etc.
reciever_id	INT	1, 2, 3, etc.
creation_time	DATETIME	Current timestamp at time of creation

Media:

Column	Datatype	Valid Entries
id (PK)	INT	(set to auto_increment from 0)
post_id (FK)	INT	0, 1, 2, etc.
file_link	VARCHAR(255)	“images/image.jpg”, “videos/vid.mp4”, etc.

VI. Key Risks

While the team certainly has the necessary skills to develop the application outlined in this document, the time constraints pose genuine risks. Balancing usability, effectiveness, and warranted-uniqueness within the given time frame will be difficult; as such, proper time management and task allocation is necessary. We will prioritize app functions to allow for minimal functionality before moving on to other functions that make the app more accessible for all to use. Another risk is that our team is likely only going to collaborate online. Meeting in person is difficult and slightly risky considering the ongoing pandemic, so building close relationships is difficult. That being said, our team has made strides to get to know each other better and to not treat every meeting as a strictly formal matter.

VII. Project Management

So far, our team has been collaborating and managing tasks over Discord. How this works is, there are separate chat rooms for each section of the team (backend, frontend, etc.). There is also a weekly goals chat room and a meeting minutes chat room. In the weekly goals chat room, groups and individual members are assigned tasks to complete throughout the week. These are usually planned in our weekly meetings outside class, which usually take place on mondays. This system has been effective; however, we all believe we will be more effective once we transition to Trello. Trello allows for much more flexibility in role-assignment and task completion. The transition will likely happen in between submitting Milestone 2 and receiving the description of Milestone 3. This transition did not take place in the midst of finishing Milestone 2 so as to not disrupt the flow of work. Team members have generally been responsive and hard-working.

Software Engineering 648/848 Spring 2022

Milestone 3

04/13/2022

Team #2 - GatorMart

Shane Waxler - Team Lead - swaxler@mail.sfsu.edu

Chuting Yan - Front-End Lead

Robert Garcia - Back-End Lead

Minggu Ma - GitHub Master

Melissa Ho - Front-End

Xiaoqing Yao - Front-End

Joe Guan - Back-End

Date written
04/13/2022

Summary of Milestone 3 ZOOM meeting review with Prof. Petkovic and plans for further development:

- **Team number:** 2
- **Meeting date:** April 13, 2022
- **Summary of feedback on UI (list if items, note that details are available in ZOOM recording):**
 - a. All pages
 - i. Add header which states that this is a class project.
 - ii. Search queries must remain after search is completed.
 - iii. Navbar order shall be Post/Sell, My Page, Login/ Registration
 - b. Home page
 - i. Remove banner image and replace with better design
 - ii. Add a short description to the banner
 - c. Search Results page
 - i. Whole item cards should be clickable, not just a “View Details” button.
 - ii. Add contact seller button to items.
 - d. My Page
 - i. Add number of messages received
 - ii. In the message, show which product it refers to
 - e. Post page
 - i. Add red asterisk to define necessary fields
 - ii. Add category dropdown menu
 - iii. Add space between the two buttons
- **Summary of feedback on code and architecture:**
 - a. Generally
 - i. Add comments/header comments.
 - ii. Transition to a more Object Oriented structure and break up the main JS file into smaller files.
 - iii. Don't repeat functions.
- **Summary of feedback on github usage:**
 - a. Good, told us our commits had detailed messages and to continue to do so.
- **Summary of feedback on DB:**
 - a. Create images & messages table.
 - b. Messages should be a P1 feature; no need to allow responses back.
 - c. Add Category table that allows drop down menu for categories to pull from it.
 - d. Use Apt titles.
- **Summary of feedback on teamwork:**
 - a. Professor said teamwork is good so far. He also noted that when issues are encountered we should contact him.
- **Any other comments and issues:**
 - a. None that we are aware of at the moment.

List below agreed upon P1 list of features for final delivery which constitute product plan. NOTE: after this meeting the team focuses solely on this P1 list of features, e.g. the development is in “feature freeze mode”. All listed P1 features (no more no less) MUST be delivered in usable way, free of bugs

Priority: 1	
Unregistered User	<p>1) Register Account - Unregistered visitors shall be able to register for an account with their active sfsu.edu email address and a desired password. This email address will be verified via a link in an email. Registered users will have more privileges than unregistered..</p> <p>2) View Marketplace - All users shall be able to view the full marketplace. Unregistered users will not be able to make offers or purchase items.</p> <p>3) Search Marketplace - All users shall be able to search the marketplace via a typed search-bar.</p> <p>4) Filter Listings - All users shall be able to filter the marketplace by category placed on listings as well as string search. This way, users can more properly find the items / services they're looking for.</p>
Registered User (Inherit all above)	<p>1) Log in - Registered users shall be able to log in to our site using the information they registered with. This will allow them the privileges of registered users.</p> <p>2) List item or service for sale - Registered users shall be able to post an item they would like to sell with a title, a description, a price, a quality, and a photo. Some form of contact information must be available.</p> <p>3) Message - Registered users shall be able to message the seller. A notification of this will be sent to the seller with the buyer's contact information.</p> <p>4) Delete Former Posts - Registered users shall be able to delete posts they no longer are selling.</p> <p>5) View User Accounts - Only registered users shall be able to see accounts of other users.</p>
Admin	<p>1) Remove Listing - Site administrators will be able to remove listings which have already been made public if there is a good reason to do so. This will permanently remove that listing from the marketplace, and users will no longer be able to interact with it.</p> <p>2) Approve / Disapprove Listing - Site Administrators shall review listings before they are made public on the site. If the content of the</p>

	listing is not approved, the administrator will be able to prevent that item from being uploaded to the site.
	3) Remove User - Administrators shall be able to permanently remove a user from the website.

Software Engineering 648/848 Spring 2022

GatorMart

Team 2

Shane Waxler - Team Lead - swaxler@mail.sfsu.edu

Chuting Yan - Front-End Lead

Robert Garcia - Back-End Lead

Minggu Ma - GitHub Master

Melissa Ho - Front-End

Xiaoqing Yao - Front-End

Joe Guan - Back-End

Milestone 4

5/14/2022

I. Product Summary

GatorMart is an online marketplace for the members of the San Francisco State students, professors, and faculty. It allows users with a validated SFSU email to buy and sell items and services. Once registered, validated, and logged in, a user may view product details, make offers on listings, as well as post their own. Interest shown in a user's listings will come in the form of messages, which is viewed on the user's profile page.. Users of the website can be assured that they are interacting with other people from the SFSU community due to the site administration validation.

GatorMart allows users to

- View the current items in the marketplace
- Search for items
- Filter Listings
- Register for an account with SFSU email

Registered users can

- Login to their account
- Upload data to post their own listings
- Message other sellers to request purchase
- Accept or decline offers on their current listings

Administrators can

- Remove listings
- Approve or deny listings
- Approve or deny new user registration, or delete user accounts

GatorMart's unique feature is to allow posters to choose between several on campus options to meet up. These locations are main hubs in busy areas of the campus.

<http://ec2-54-237-46-1.compute-1.amazonaws.com/>

II. Usability Test Plan (max 2 pages)

Test Objectives - Uploading and Posting Data

The objective of our usability test is to test user satisfaction on the ease of posting data to GatorMart. GatorMart is an online marketplace, where registered users are expected to post multiple items or services to the database to be displayed to other members. The test will help us determine future areas of improvement to optimize the user experience.

Test Background and Setup

System setup

A browser on a PC, laptop, or other mobile device with internet access.

Starting point

A user that is registered with GatorMart but not currently logged in.

Intended Users

San Francisco State University students, faculty, and staff. Those with basic knowledge of how to navigate a browser on the PC or a mobile device.

URL <http://ec2-54-237-46-1.compute-1.amazonaws.com/>

/item/post

User Satisfaction

For our user satisfaction survey, we would present the focus group with this survey:

Task	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
The process of locating the link to the post/uploading data page was simple	1				
Process of navigating to the “POST” page was easy	1				
The form to post and upload data was intuitive and aesthetically pleasing	1				
The mandatory sections were clearly marked	1				
The clickable sections were not too small	1				
I was able to complete the task in a reasonable amount of time	1				
The process of clearing all the data was simple and intuitive	1				
The buttons and fields were clearly marked so that there were no ambiguities on what to fill in	1				

Usability Task Description

Please log in to your GatorMart account.

Evaluation of Effectiveness

To evaluate the effectiveness of posting a listing, we would measure the percentage of users that completed these following tasks successfully, as well as note the errors that users could encounter in the process. We would also give users an option to leave a comment on how effectively they were able to complete the task in a reasonable amount of time.

To evaluate effectiveness of this function, we would adhere to the following example chart:

Test/Use case	%completed	errors	comments
Login	100		
Navigation to /item/post page	100		
Filling out the forms	100		
Uploading a picture	100		
Clearing the form	100		
Submitting the form	100		

Evaluation of Efficiency

To measure efficiency of the feature, we would check the following tasks: Login, navigation to page, filling out the form, upload of pictures, clearing the form, and submitting the form. We would test for how long a user takes to complete each task, the number of clicks, and the number of errors a user encounters.

To evaluate the efficiency of our function, we would adhere to the following example chart:

Test/Use case	Time of completion	Number of clicks	Number of errors
Login			
Navigation to /item/post page			
Filling out the forms			
Uploading a picture			
Clearing the form			
Submitting the form			

III. QA Test Plan

Test Objectives

To test the reliability of the posting function on GatorMart when a registered and logged in user attempts to create a posting. We are testing the success rates of when a user attempts to post and upload new data to GatorMart's marketplace.

Hardware and Software Setup (include URL)

Hardware setup - A PC, laptop, or mobile device with internet connection

Software setup - Chrome or Firefox browser

<http://ec2-54-237-46-1.compute-1.amazonaws.com/>

Feature to be tested

Priority 1 function "Post/Sell". We are testing to see if filling out the form, filling it out incompletely or incorrectly, and clearing the form are successful.

QA test plan example chart:

Test	Title	Description	Test Input	Expected Correct Output	Results (P/F) Firefox	Results (P/F) Chrome
1	Full inputs on the form	When a user completes all the fields on the form, they click submit to upload the data to the marketplace	1. Completely and correctly fill all the fields on the entire post form. 2. Press Submit	Redirected to user's profile page, where the item shows up on the user's profile under "current listings" section. When searching for the item, the item appears in the marketplace search function.		
2	Only one field	When a user only completes one field on the form and attempts to click the submit button.	1. Fill out the name of the listing only 2. Press Submit	A message appears reminding the user that all fields marked with a red asterisk are mandatory. No redirect. Submission is unsuccessful.		
3	Too many words in the description box	The user writes too many words in the description box, which states that the maximum is 250 characters	1. Write 251 characters in the description box 2. Press Submit	A message appears reminding that the user shall not write more than 250 letters for the description of the item. There is no redirect, submission is unsuccessful.		
4	Clearing the fields	The user fills out one or more fields on the form, but	1. Completely and correctly fill all the fields on the entire post form.	All of the fields on the post form are cleared, the page is redirected to the same page as before the user has filled out		

		changes mind about posting at that moment.	2. Press “clear” button.	anything.		
--	--	--	--------------------------	-----------	--	--

IV. Code Review

RE: Code Review Request

MH Melissa Ho <mho10@mail.sfsu.edu> 9:36 PM

To: Chuting Yan

Hello Chuting,

I've reviewed your code for the post page. Here is the summary:

M4 Code Review - post.hbs. Reviewer - Melissa Ho. The code is very organized in terms of layers, divs, and TODO actions. The way Chuting set up her comments really helps for when merge happens. The back end team will clearly be able to see where the JS elements are needed to complete the page. Overall, really great job. The github commit comments she included are succinct, but descriptive enough for me to understand how changes were implemented and in what order. I suggested some minor detail changes on where certain text fields are positioned on the page so that the user has an easier time reading down the page. Great job, Chuting!

I've also made comments in the code itself via GitHub, as well as uploaded the summary on the commit.

Have a great weekend!

Best Regards,
Melissa

From: Chuting Yan
Sent: Thursday, May 12, 2022 9:12 PM
To: Melissa Ho
Subject: Code Review Request

Hi Melissa,

Hope you are doing well.

Can you please review the post page?

Thanks,
Chuting

M4 Code Review

M4 Code Review - post.hbs. Reviewer - Melissa Ho. The code is very organized in terms of layers, divs, and TODO actions. The way Chuting set up her comments really helps for when merge happens. The back end team will clearly be able to see where the JS elements are needed to complete the page. Overall, really great job. The github commit comments she included are succinct, but descriptive enough for me to understand how changes were implemented and in what order. I suggested some minor detail changes on where certain text fields are positioned on the page so that the user has an easier time reading down the page. Great job, Chuting!

frontend

mellissa17 committed 5 minutes ago Verified

1 parent 3cef807 commit 459586ebfc358093eb77d784cf62e0309c2516

Showing 1 changed file with 36 additions and 1 deletion.

```
diff --git application/views/post.hbs application/views/post.hbs
--- a/application/views/post.hbs
+++ b/application/views/post.hbs
@@ -5,-6 +5,+11 @@
     ...
     ...
     ...
@@ -13,+14 @@
     ...
     ...
     ...
@@ -15,+16 @@
     ...
     ...
     ...
@@ -18,+19 @@
     ...
     ...
     ...
@@ -20,+21 @@
     ...
     ...
     ...
@@ -21,+22 @@
     ...
     ...
     ...
@@ -22,+23 @@
     ...
     ...
     ...
@@ -23,+24 @@
     ...
     ...
     ...
@@ -24,+25 @@
     ...
     ...
     ...
@@ -25,+26 @@
     ...
     ...
     ...
@@ -26,+27 @@
     ...
     ...
     ...
@@ -27,+28 @@
     ...
     ...
     ...
@@ -28,+29 @@
     ...
     ...
     ...
@@ -29,+30 @@
     ...
     ...
     ...
@@ -30,+31 @@
     ...
     ...
     ...
@@ -31,+32 @@
     ...
     ...
     ...
@@ -32,+33 @@
     ...
     ...
     ...

```

```
29 +
30 +         <!--HELISSA:
31 +             I really like how this portion of the code is set up.
32 +             It's easy to read and understand. However, the text in front of the
33 +             banner is a little hard to read. I recommend changing the color or
34 +             font size.
35 +             -->
36 +
37 +         <div class="hl-positions">Post Item For Sale Or List Jobs</div>
38 +
39 +
40 +         <!--HELISSA:
41 +             A suggestion I have is to put the *labels denoted with * are mandatory"
42 +             line underneath the header instead of by itself on the right. That way, it's
43 +             easier for the user to read as he or she browses down the page.
44 +             -->
45 +
46 +         <div class="item float-right">
47 +             Labels denoted with <span>*</span> are mandatory.
48 +         </div>
49 +
50 +         <!--HELISSA:
51 +             In my opinion, the labels for the forms should be a little larger to look like headings.
52 +             -->
53 +
54 +         <div class="item">
55 +             <label for="name"> item/job Name <span>*</span></label>
56 +             <input id="name" type="text" name="name" required />
57 +             @0 -65,6 +51,11 @0
58 +
59 +             <label for="instructions">Description <span>*</span></label>
60 +             <textarea id="instructions" rows="3" required></textarea>
61 +             </div>
62 +
63 +         <!--HELISSA:
64 +             I like the feature that limits users to a certain number of characters for the description.
65 +             It's a good security implementation. However, I feel as if 100 characters is too little. Maybe you could
66 +             change it to 250 characters, to keep it short but still allow the user to say what he or she needs.
67 +             -->
68 +             <p>(small)<100 characters</small></p>
69 +             <div class="mb-block">
70 +                 <button type="submit" class="certain " href="/profile">Submit</button>
71 +             @0 -51,4 +120,4 @0
72 +
73 +             <script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.0/dist/js/bootstrap.min.js"
74 +                   integrity="sha384-T7eQqjZqJLWVx/9/7fYsH+XnZB+q5+QXxU9PwRy+D+gZDfJ93HvOg+u4Cz" data-bbox="820 930 880 948">
```

V. Self-check on best practices for security

Asset to be protected	Types of possible/expected attacks	Your strategy to mitigate/protect the asset
Password	Unsecured accounts	Encryption, track system usage, two factor authentication. Using session handling.
Database	SQL injection	Input validation and queries using prepared statements.
User Accounts	Spam or robot users.	A valid SFSU email is required to register.
Postings	Dangerous postings, threats, scams	Administrator's validation before each posting, giving a window of 24 hours.
Confidential data	Data leak	Using HTTPS because it is a secure protocol.
Search		Limit search bar input to 40 alphanumeric characters

VI. Self-check of the adherence to original Non-functional specs

Non Functional Specifications	Status	Comments
Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0	On track	
Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers	Done	
All or selected application functions must render well on mobile devices	Done	
Data shall be stored in the database on the team's deployment server.	Done	
No more than 50 concurrent users shall be accessing the application at any time	On track	
Privacy of users shall be protected	On track	
The language used shall be English (no localization needed)	Done	
Application shall be very easy to use and intuitive	Done	
Application should follow established architecture patterns	Done	
Application code and its repository shall be easy to inspect and maintain	Done	
Google analytics shall be	On track	

used		
No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging. One round of messaging (from user to seller) is enough for this application	Done	
Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.	Done	
Site security: basic best practices shall be applied (as covered in the class) for main data items	On track	
Media formats shall be standard as used in the market today	Done	
Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development	Done	
The application UI (WWW and mobile) shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2022. For Demonstration Only" at the top of the WWW page nav bar. (Important so as to not confuse this with a real application).	Done	

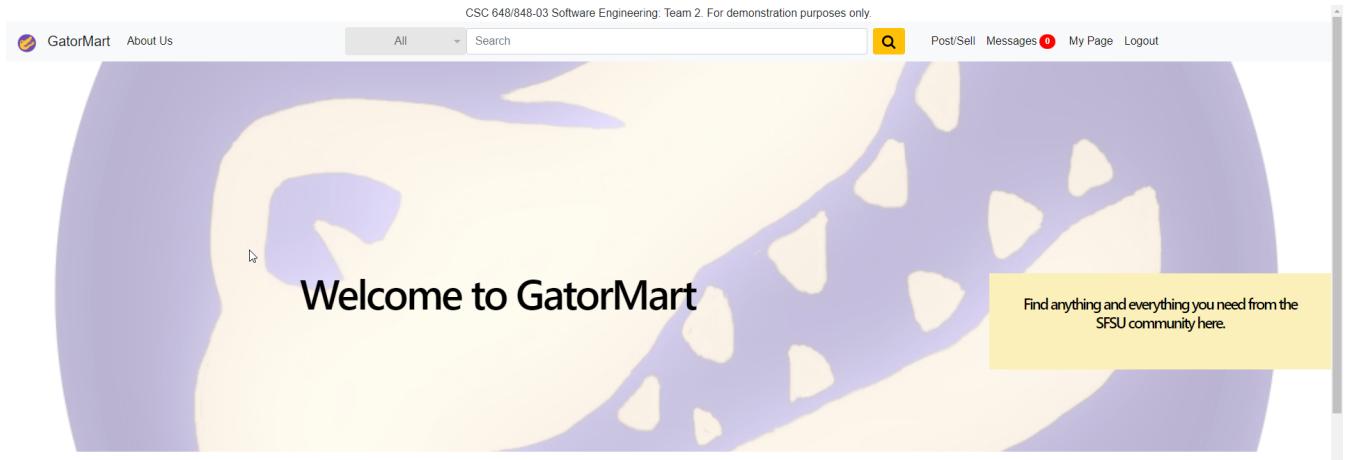
IX. Product Screenshots



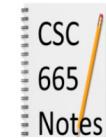
The screenshot shows the 'Register for GatorMart' page. At the top, there is a navigation bar with links for 'GatorMart' and 'About Us'. A search bar is located next to the navigation bar. On the right side of the header, there is a 'Login/Register' button. Below the header, a message states 'Fields marked with * are required.' The registration form consists of several input fields:

- Full Name***: An input field labeled 'First Last'.
- SFSU email***: An input field labeled 'Email' with the note 'Must be an official SFSU email.'
- Username***: An input field labeled 'Username' with the note 'This is the display name that will be used on GatorMart.'
- Phone Number (optional)**: An input field labeled 'Phone Number'.
- Password***: An input field labeled 'Password' with the note 'Password must be 8 or more characters, and contain upper and lowercase letters.'
- Confirm Password***: An input field labeled 'Confirm Your Password' with the note 'Passwords must match.'

At the bottom of the form, there is a 'Submit' button and a link 'Already have an account? [Log In Here](#)'.



FEATURED ITEMS



About

Full Name: Melissa Ho
SFSU Email: mho10@mail.sfsu.edu
Username: meow717



CSC 648/848-03 Software Engineering: Team 2. For demonstration purposes only.

All Search Post/Sell Messages 1 My Page Logout

Post Item For Sale Or List Jobs

Labels denoted with * are mandatory.

Title *

Category *

Price *

Condition

Upload Picture * No file chosen

Description *

0/100 characters

May take up to 24 hours for approval

CSC 648/848-03 Software Engineering: Team 2. For demonstration purposes only.

All Search Post/Sell Messages 1 My Page Logout

Cat food, Two 5lb bags

Note: this item must be approved by site administrators before it is displayed
on the main page

Miscellaneous

Posted by: [meow717](#) on 2022-05-21

Quality:

My cat died a week ago and I have a bunch of extra cat food. If anyone wants to buy it from me. Friskie's brand, Salmon flavored.

\$20

GatorMart

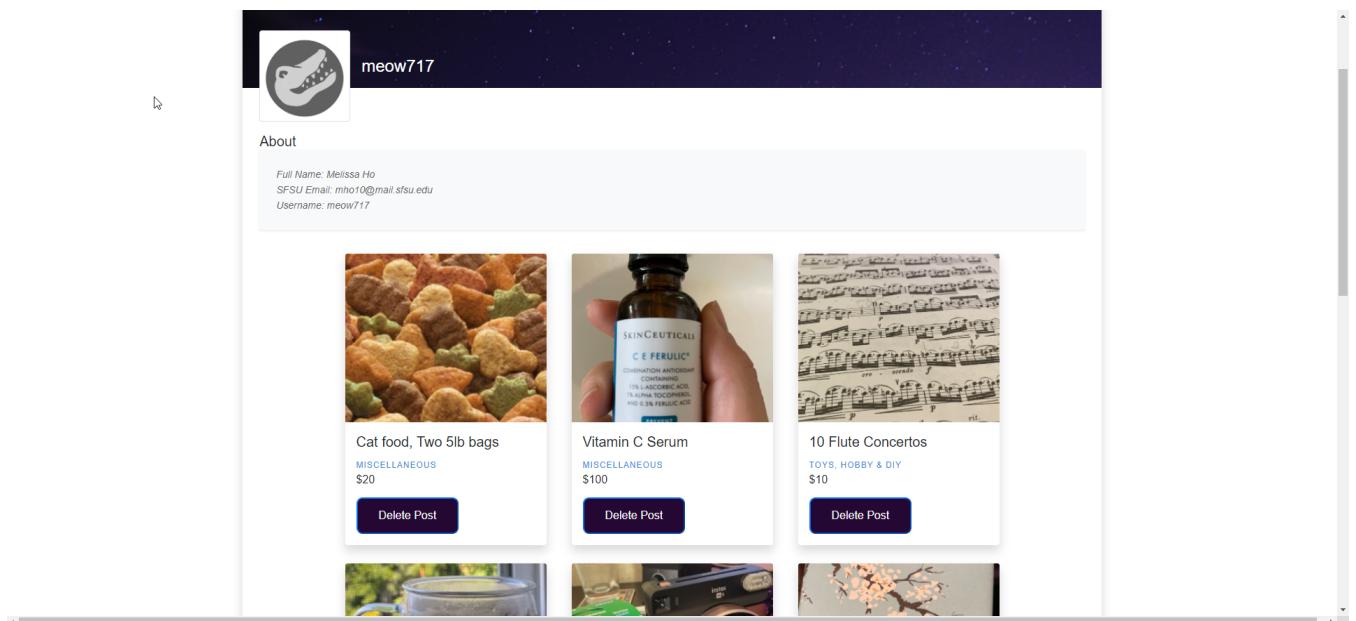
[Home](#) | [About](#) | [Contact](#)
© 2022 GatorMart Pvt. Ltd.

Software Engineering - CSC648/848
Spring, 2022, Section 03, Team 02

support@gatormart.com

About GatorMart

GatorMart is an online hub exclusively for SFSU community. GatorMart provides service to current students, faculty, and staff.



CSC 648/848-03 Software Engineering: Team 2. For demonstration purposes only.

GatorMart About Us Books Post/Sell Messages (0) My Page Logout

We couldn't find anything for "textbooks" in Books. Here are some things you might like. Sort By

Learning Life: The Path to ... BOOKS \$20 Message Seller	Calculus: Early Transcend... BOOKS \$12 Message Seller	Physics for Scientists and ... BOOKS \$5 Message Seller	Understanding and Using ... BOOKS \$10 Message Seller	CSC 648 Notes NOTES \$100 Message Seller	CSC 665 Notes NOTES \$5 Message Seller
CSC 642 Notes NOTES Message Seller	Coffee Table FURNITURE Message Seller	Computer Chair FURNITURE Message Seller	Couch FURNITURE Message Seller	New Guitar! TOYS, HOBBY & DIY Message Seller	Help Me Move this Couch! HELP WANTED Message Seller

CSC 648/848-03 Software Engineering Team 2 For demonstration purposes only.

GatorMart About Us

All calculus

Post/Sell Messages 0 My Page Logout

Showing 1 results for "calculus" in All

Sort By

JAMES STEWART
CALCULUS

Message IDunagan2

Message Body*

Send Phone Number?

Cancel Send Message

GatorMart

Software Engineering - CSC648/848
Spring, 2022, Section 03, Team 02

[Home](#) | [About](#) | [Contact](#)

About GatorMart

GatorMart is an online hub exclusively for SFSU community. GatorMart provides service to current students, faculty, and staff.

[support@gatormart.com](#)

CSC 648/848-03 Software Engineering: Team 2. For demonstration purposes only.

All Search  Post/Sell Messages  My Page Logout



Shane Waxler
Team Lead
[Read More](#)



Minggu Ma
GitHub Lead
[Read More](#)



Chuting Yan
Front-End Lead
[Read More](#)



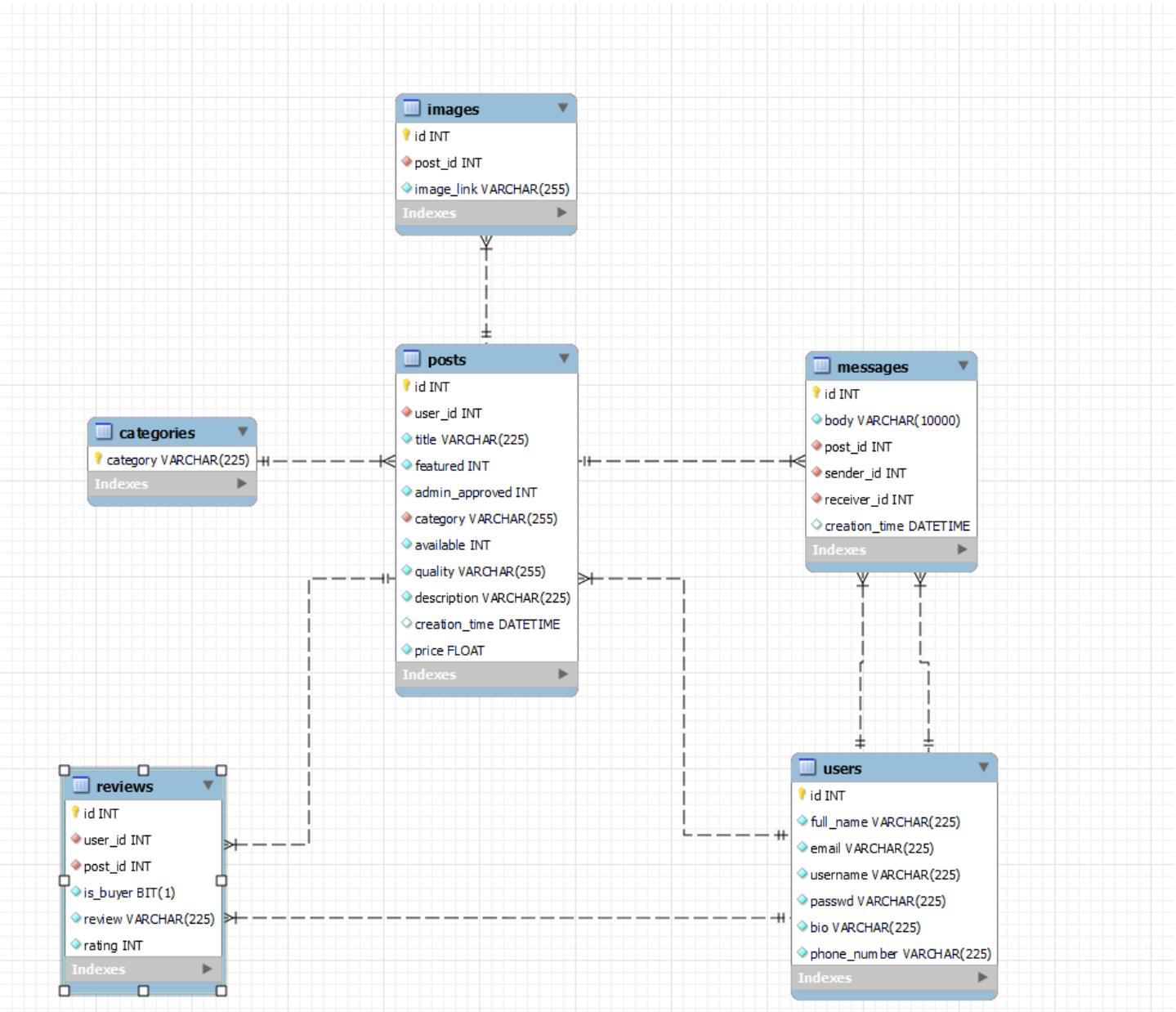
Robert Garcia
Back-End Lead
[Read More](#)



Melissa Ho
Front-End
[Read More](#)



X. Database Organization



XI. Github Organization

The screenshot shows a GitHub repository page for 'CSC-648-SFSU / csc648-03-sp22-team02'. The repository is private. The main navigation bar includes 'Search or jump to...', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the navigation, there are links for 'Code' (selected), 'Issues' (16), 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. The repository name is 'csc648-03-sp22-team02'. It has 13 branches and 0 tags. A message indicates 'develop had recent pushes 5 minutes ago'. The repository is 60 commits ahead of 'main'. A commit from 'Macgogole' merges 'develop' into 'Frontend-Develop'. The repository has 142 commits. The 'About' section includes a 'Readme' file, an MIT license, 1 star, 1 watching, and 0 forks. The 'Releases' section shows no releases published, with a link to 'Create a new release'. The 'Packages' section shows no packages published, with a link to 'Publish your first package'. The 'Contributors' section lists 6 contributors with small profile icons. The 'Languages' section shows a breakdown: HTML 86.3%, CSS 10.7%, and JavaScript 3.0%. The repository description is 'csc648 Repository'. It asks users to add their team's application URL or IP to the description. It also instructs users to do the following steps before completing Milestone 0: change the name of the repository to their respective team number.

A. List main branches

1. Develop - Back and Front End
2. Front End - Front End Team
3. Back End - Back End Team
4. Main - Github Master and Team Lead

All other branches are temporary branches for merge tests. Important merges

B. Describe each top level folder in github file organization (name and what it contains, up to three lines per folder)

/Application

/db: Contains both the db initializing script and the db.js file which establishes a connection between the express app and the database.

/public: Contains the stylesheets, images, and local scripts.

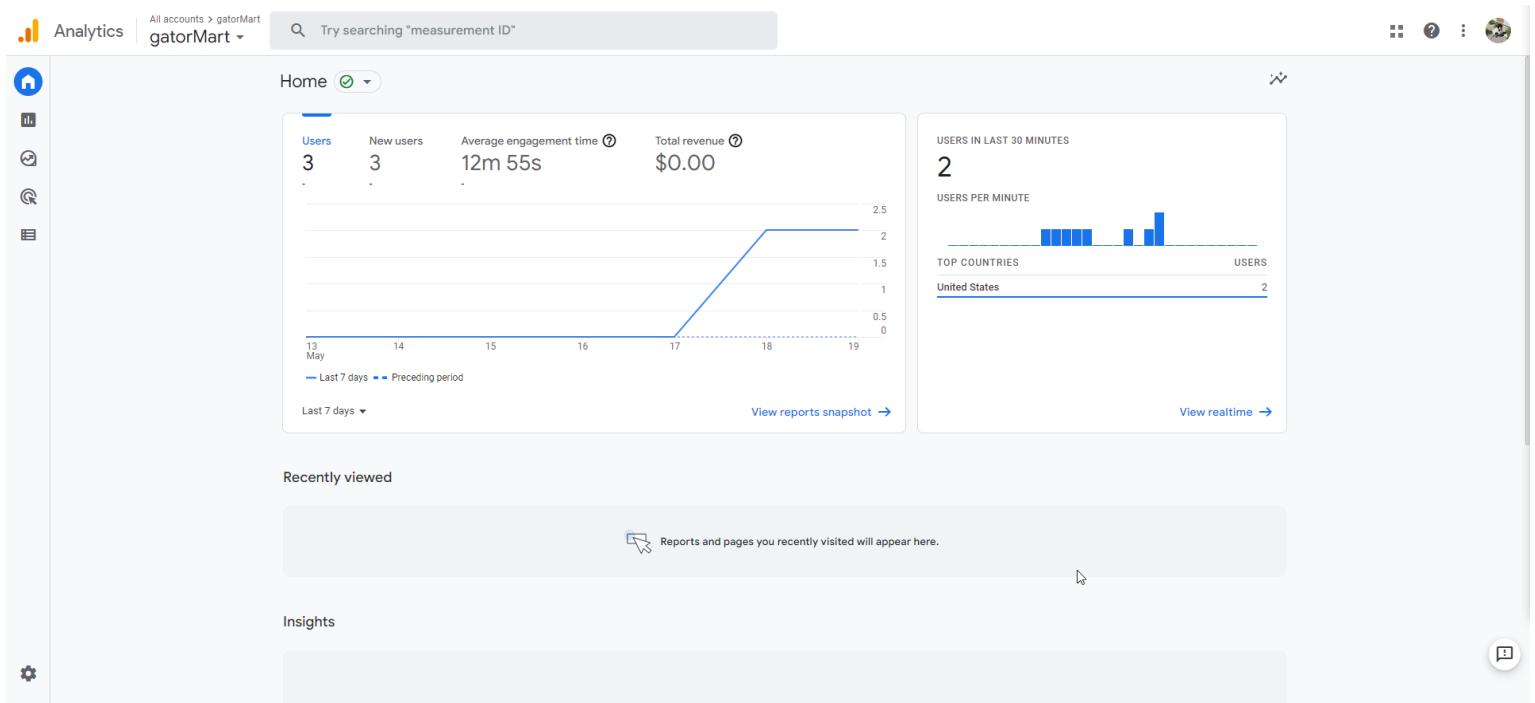
/routes: Contains router-level middleware to reduce the size of app.js. Each file contains classes with relevant functions to the route (e.g., fetch items, send messages, etc.).

/views: Contains the handlebars views for all pages, layouts, and partials (e.g. navbar).

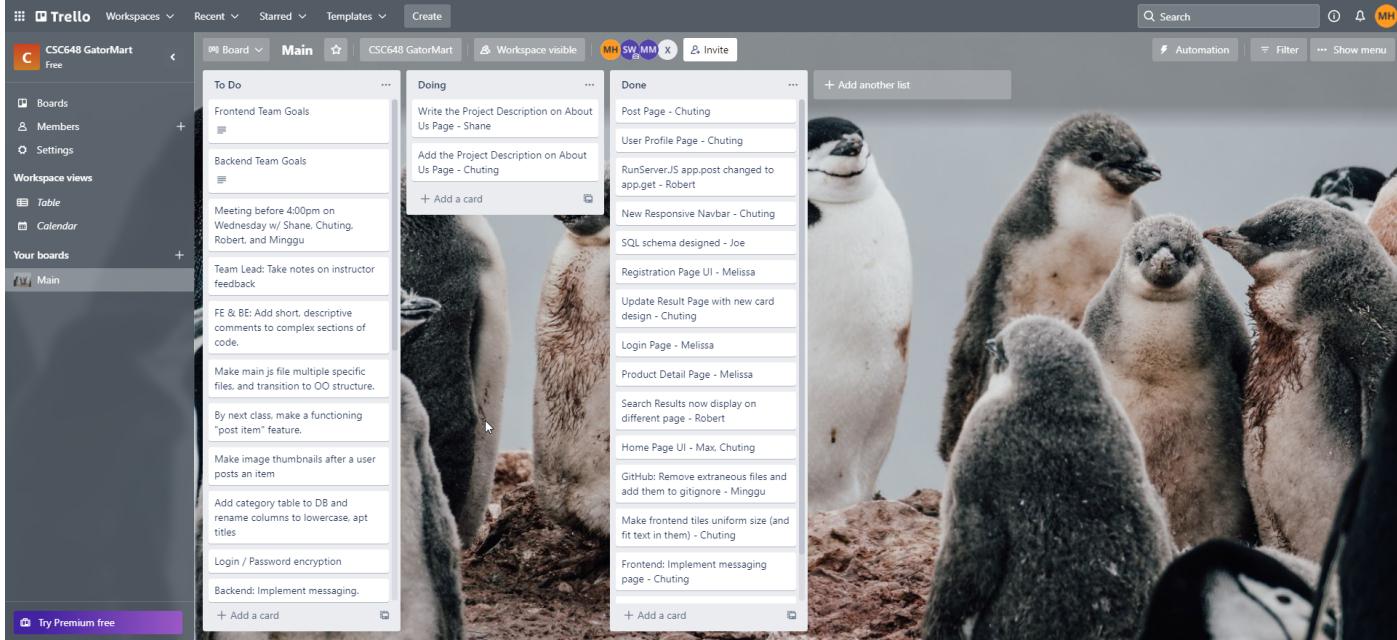
/Milestone - Contains each milestone document submitted.

/Credentials - Contains necessary credentials to access the server.

XII. Google Analytics Stats Plot for WWW Site



XIII. Project Management



important-documents

Revision for milestone 2

1

May 5, 2022

Melissa (FE) 05/05/2022 <https://docs.google.com/document/d/1hdjTtCG4qxHOCxSsOjboX3NyfI4yGBSe4AVnhS16QA/edit?usp=sharing>

Google Docs

GatorMart Milestone4

Software Engineering 648/848 Spring 2022 GatorMart Team 2
Shane Waxler - Team Lead - swaxler@mail.fsu.edu Chuting Yan - Front-End Lead Robert Garcia - Back-End Lead Minggu Ma - GitHub Master Melissa Ho - Front-End Xiaoqing Yao - Front-End Joe Guan - Back-End Milestone 4 5/14/2...

May 15, 2022

Melissa (FE) 05/15/2022 https://docs.google.com/document/d/1rIkC96f_lsLkXcIvgX5jxI4-k6SVc-hw89_TQ78U/edit?usp=sharing

Google Docs

GatorMart Milestone5

Software Engineering 648/848 Spring 2022 Team 2 GatorMart
Shane Waxler - Team Lead - swaxler@mail.fsu.edu Chuting Yan - Front-End Lead Robert Garcia - Back-End Lead Minggu Ma - GitHub Master Melissa Ho - Front-End Xiaoqing Yao - Front-End Joe Guan - Back-End //TODO CLICKABLE URL HERE 5/21/2...

meow

Message #important-documents

ONLINE - 3

Melissa (FE)

Robert G. (BE Lead)

Shane W. Sharing their screen

OFFLINE - 4

Chuting (FE Lead)

Joe (FE)

Max (FE)

Minggu (GitHub Lead)

XIV. Team member self-assessments and contributions

CSC648/848 M5 Self Assessment

SW Shane Christopher Waxler
To: Melissa Ho
Cc: Minggu Ma; Robert Johnathan Garcia; Joe Guan; Xiaoqing Yao; Chuting Yan

Contributions to team project and teamwork

- Organizing and scheduling group meetings
- Ensuring team members are contributing as much as they happily can
- Creating an environment where working is engaging and interesting
- Assigning fitting tasks for each team member
- Keeping track of course requirements and submission details
- Writing large portions of backend code with Robert
- Revising frontend code to work properly with the backend
- Reviewing code and adding comments where necessary
- Ensuring that directory structure was proper and that SW principles were followed throughout

Number of submissions made to GitHub team Development Branch: 88 (though, much of these commits were done in a shared VSCode session with Robert. I do not wish to take full credit for all 88. It was a collaborative effort).

Main Challenges: The most significant challenge I faced as a team lead was achieving decisiveness and clarity of objective. The decisiveness part may be obvious. I learned fairly quickly to rely on best principles and then trust instinct for decision making. Clarity of objectives, on the other hand, was a bit more difficult. At first, it seemed as a team lead that I had very little responsibility. Simply assigning tasks did not seem helpful enough from my point of view, and so I decided early on to be a collaborative member of the team. This decision was made not only because I had an excess of time, but also because I felt that my contribution would speed up the process and alleviate extra work for the rest of the team. A majority of my collaborative efforts were directed towards the backend. Despite having written an Express.js application previously, I found it difficult to structure our backend in a way that made sense. Luckily, at one point we decided to move forward with express-handlebars, whose documentation was far more clear and concise about best practices for Express apps than the Express documentation. Starting to develop before having a fundamental understanding of the systems we were using (e.g., what is middleware, what are routers, when to use GET vs POST, etc.) was an utter mess. One long file with numerous functions which were essentially just used because they showed up in the documentation. Luckily, a comprehensive reading of the documentation led to a very quick understanding of what the actual purpose of Express is and how to best structure an Express application. This also allowed me—as a team lead—to better guide the rest of the team on project directory layout. What I found most interesting was that, to be a good team lead, I had to also be a principled engineer. Without understanding the fundamentals of the software we were using, there was no way I could properly advise the team on what/how to develop. Perhaps this was a personal obstacle though, as I am aware of many successful team leaders who had very little technical knowledge (or, at least far less technical knowledge than business and design prowess. See Steve Jobs). Anyways, I found it difficult to operate without. Having the knowledge made the rest of the process quite smooth.

What I Would do Better Next Time: Judging by my previous answer, the answer to this is quite simple: always read documentation before writing code (in addition to a host of other design/specification-related things which—luckily—we were guided through via the Milestones laid out for us). The design elements were, for this project, mostly ostensible/taste based, though the introduction to HCI given in this class gave a better understanding of nearly objective principles in design. More generally, though, than reading documentation though is knowing that principled approaches *inherently* make design choices simpler. For instance, if we had continued our misguided backend approach of keeping all express-related code in one file, relating that to the frontend views and routing requests would have been a nightmare. Corners would have been cut on the number of pages and functions we were able to write. Perhaps if this project had a longer development cycle, I could have assumed a further position from the technical work and focused significantly more on task allocation, focus grouping, and design; however, judging by the fact that we only had 6 months to deploy, I found it necessary to be an engineer as well.

- Shane Waxler

CSC648/848 M5 Self Assessment

RG Robert Johnathan Garcia
To: Melissa Ho; Minggu Ma; Shane Christopher Waxler; Joe Guan; Xiaoqing Yao; Chuting Yan

Robert Garcia - Back End Lead

Self Assessment and Contributions

Contributions to team project and teamwork -
Was the back-end lead for the team and worked on a number of different files related to the backend alongside the team-lead. These files include functionalities relevant to: registering, logging in, searching for an item, posting items, receiving messages, sending messages, rendering the page for an item, and rendering the user page for a specific user. Was also involved in the initial creation of the AWS server as well as contributing to the setup of the DB. Active and Communicative with the team throughout meetings and through discord.

Number of Contributions to Dev Branch - 39 Total

Main challenges encountered in team project -
In terms of main challenges in terms of the backend coding; there were definitely a number of them. As someone who has had a bit of experience in working with JavaScript but am not super familiar with the language, coding for the backend took a ton of review and research to get right. In terms of the lead for the backend; there were challenges at times getting everyone in the team involved. This project is one of the few times I've been a lead for a project so I'm still learning with every experience, and while there were times I felt the challenges of the responsibilities that come with being a backend lead, I thought our team worked very well alongside one another.

What would I do better next time based on what was learned in the class about SE management and processes -
If I were to try to do this same project after knowing everything I do now, I think I would have spent a lot more time in the planning of the functionalities in the backend. I believe that planning more would have made splitting tasks much simpler and could have made working alongside one another much more efficient. Also, while we set up meetings generally for the team fairly often, I could have done a better job in setting up meetings for the backend team specifically to ensure that we were all making good progress. Overall though, I was very proud of the progress that we as a team all made.

Best,
Robert Garcia



Chuting Yan

To: Joe Guan; Minggu Ma; Robert Johnathan Garcia; Melissa Ho; Shane Christopher Waxler; Xiaoqing Yao



Fri 5/20/2022 4:23 PM

Chuting Yan - Front End Lead

Contributions to team project and teamwork

- Designed and implemented several pages/files
- Worked on group milestone documentations and use cases
- Attended all required team meetings and shared Front End team progress
- Worked with team lead to set up goals for each milestone for Front End team
- Communication with team lead, Back End Lead, and Front End team on goals and expectations
- Broke down each milestone goal into actionable tickets, assigned to Front End team members or asked them to choose the tickets they want to work on
- Checked in with Front End teammates on their tickets, gave suggestions and helped make decisions when needed
- Active and responsive on Discord and Trello

Number of submissions made to the github team Development Branch: 45

Main challenges encountered in team project :

As a Front End lead, I had more responsibilities as well as more challenges. One challenge I had is to have all teammates contribute to the same level, or at least similar level. Another challenge I had is to be aware of the different levels of communication and guidance needed for different teammates. Some teammates might need more guidance and reminders on the delivery expectations and delivery deadline. I'm still learning the balance between making sure teammates are on track for their tickets and micromanagement.

What would I do better next time based on what was learned in the class about SE management and processes:

I would communicate with each member to have a better understanding of the level of communication they prefer and need early on, and I would check in with them accordingly. I would set up meetings more often with the Front End team to help if anything blocking them from making progress. I would do a better job of assigning tickets and setting deadlines for each teammate, and I would push more to have all teammates contribute to a similar level. Overall, I'm grateful for the team members that I worked along with and made contributions to this project.

Best,
Chuting

...

CSC648/848 M5 Self Assessment



Melissa Ho

To: Minggu Ma; Shane Christopher Waxler; Robert Johnathan Garcia; Joe Guan; Xiaoqing Yao; Chuting Yan



Thu 5/19/2022 5:57 PM

Melissa Ho - Front End

Contributions to team project and teamwork

- Communication with FE/BE lead and team lead on goals for the upcoming week
- Attended all required team meetings and had a valid excuse for why I could not attend the optional ones.
- Set up the main storyboards for early project development so that the other members of the Front End team could use the same templates.
- Authored the executive summary for documentation
- Worked with FE lead to discuss design elements of the website and collaborated with her to find inspiration from other sites we were browsing.
- Worked with back end to set up routes, handlebars, and express for the eventual merging of the branches.
- Worked consistently and checked in with Chuting (FE lead) throughout the semester to assess goals and what needed to be done for the upcoming week(s).
- Active on both Discord and Trello about personal progress and feedback, and asked for help if necessary and in a timely manner before deadlines.
- Worked consistently to revise and help with all documentation processes.

Number of submissions made to the github team Development Branch: 36

One of the main challenges I encountered doing this project was finding the time to discuss things with the lead, because I work full time and I usually get off work after 10pm. Sometimes I did not want to message because it was out of normal business hours. Another challenge I encountered was the lack of communication between some of the team members. It often felt like the same people were working or communicating week to week, and not everybody on the team was on the same page.

Next time, I would have more week to week goals consistently set up for myself and for my team. I would love to have more communication with members from the back end team as well as communication within the front end team so that we are all on the same track. In the case that we are not, I would push for more active communication or try to find solutions as to why it's not being achieved.

Best Regards,
Melissa Ho

Reply Reply all Forward



Minggu Ma

To: Robert Johnathan Garcia; Melissa Ho; Shane Christopher Waxler; Joe Guan; Xiaoqing Yao; Chuting Yan

↳ ↙ ↖ ↘ ⋮

Thu 5/19/2022 9:34 PM

Minggu Ma - GitHub/Backend

Contributions to team projects and teamwork:

Was initially in charge of GitHub master then move to the backend. I've been helping with the initial GitHub branch setup as well as merging big changes from backend/frontend to the develop branch and main. Was helping to develop standalone functions for backend use. Was responsive to the backend lead and team lead and other team members whenever they reached out. Was attend every team meeting and group meeting on time.

Number of submissions made to the Dev branch: 12

Main challenges encountered in the team project:

Since I do not have too much backend-related knowledge, it took me quite a long time to understand what needs to be done and the functions already done by other people in our team. Luckily our backend lead and team lead is very supportive and gives us time to figure things out and answer our questions as well.

What would I do better next time based on what was learned in the class about SE management and processes?

I think the thing I have to do better next time is to be less passive. Throughout the semester, I am mainly doing the stuff that was assigned to me. I was counting too much on the direction of the leads. Next time I am working in a team I will be more active. Reach out to people more and get myself more involved in things that the team does. That might be the more efficient way to get things done.

Best,
Minggu Ma

...

CSC648/848 M5 Self Assessment



Joe Guan

To: Minggu Ma; Robert Johnathan Garcia; Melissa Ho; Shane Christopher Waxler; Xiaoqing Yao; Chuting Yan

↳ ↙ ↖ ↘ ⋮

Fri 5/20/2022 9:09 AM

Joe Guan - Back-End

Contributions to team projects and teamwork:

- Created some of the use cases and personas
- Aided in creating database
- Tried to help in implementing a few backend functions
- Almost attended every team meeting and told team when I wasn't able to meet up
- Responded to messages within the discord

Number of contributions to the Dev Branch: 15

Main challenges encountered with the project:

The main challenges I encountered during the project was related to time. First was the amount of time trying to figure out backend functions as I've only had a mild experience with backend and even less with trying to use javascript to connect with the backend. The other was time management with my outside of class commitments such as other classes. As to be expected as the semester came closer to ending, classes became busier, and I became overwhelmed with work on other projects that took a lot of time to do and was struggling to finish. This cause me to have focus on what I could complete but I'm thankful that the backend team gave some tasks that allow me the capacity to be able to focus on being to do my work in other classes.

What would I do better next time based on what was learned in the class about SE management and processes?

For next time, I believe I could have been more communicative with my team about the things I could do and where I was at. During times I was stressed with work from other classes or was feeling sick, I didn't speak up about it and took the tasks they asked me to do. However, this led to me not being able to finish some of them before the next time we meet and me having to say why. This was especially a problem near the end as I tried to take on tasks, but I had to focus on finish my projects from other classes as they were due around the same time as the when the team asked me to have the task done and those projects ended taking more time than I expect. During these times, I should have let the team know where I was act so that they could do something about rather than me telling them I didn't finish because I got overwhelmed with work.

Best,
Joe Guan

...

↳ Reply

◀◀ Reply all

→ Forward



Xiaoqing Yao
Re: CSC648/848 M5 Self Ass...
To: Chuting Yan & 5 more

11:19 AM

[Details](#)

Xiaoqing Yao - Front End

Self Assessment and Contributions

Contributions to team project and teamwork -

Was the front-end member of the team. Worked on the task given by the front-end leader and team leader and complete each goal. Worked on part of UI design for the homepage. Finish given task on every milestone documentation. Active and communicative with the team throughout participating in every zoom meeting and discord.

Number of Contributions to Dev-Branch - 5 Total

Main challenges encountered in team project -

Since I don't have much knowledge and little experience about front-end related work from class 317, I only know some very basic stuff like login/register/search with javascript. It's pretty difficult to get started on something not familiar but I also enjoy learning and accumulating new experiences throughout working on the project.

What would I do better next time based on what was learned in the class about SE management and processes -

what would I do better for next time based on what was learned in the class, I will apply more learned concepts and knowledge in the project. What I also need to do better is not rely too much on leaders. Because, what I did this semester is only doing the tasks assigned to me by my leaders, I am grateful for how they helped me so much but I need to learn to stop relying on others too much.

Best,
Xiaoqing Yao