

JUNGLE

SW Engineering
CSC 648/848
Fall 2022
Milestone 5
Team 2

TEAM LEAD	Nanda Pandian (npandian@mail.sfsu.edu)
GITHUB LEAD	Troy Carloni
FRONT-END LEAD	Farid Mehdipour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu
DATE SUBMITTED	12/17/22
URL TO DEMO	http://35.233.185.244:8080/

Product Summary

Name: Jungle

Description of product:

Jungle is a platform used to sell San Francisco State University graphic media, eTextbooks, and a few miscellaneous items. SFSU media includes videos, photos, and other visual content. The purpose of our application is to bring convenience to the customer's finger tips. Oftentimes SFSU students and faculty are struggling to find time between their class or work schedules to get certain media. Our application allows customers to easily purchase and get access to SFSU's media as they are needed.

Itemized list of ALL major committed functions:

1. Unregistered users are able to create temp registration account
2. Unregistered users are able to search items
3. Unregistered users are able to perform general browse
4. Unregistered users are able to check product details
5. Unregistered user are able to browse by category
6. Registered users are able to log in their account
7. Registered users are able to create posts
8. Registered user are able to change email
9. Registered user are able to change password
10. Registered user are able to message other registered users in regards to their posting

What is unique in your product:

Jungle's unique feature is its ability to filter the product search by SFSU class and professor names. Oftentimes SFSU students and faculty have a tough time trying to find out which media belongs to which class and professor. Therefore, Jungle helps narrow down SFSU students and faculty' search to find what they need by adding another filter option.

URL to your product:

<http://35.233.185.244:8080/>

Milestone Documents

(Attached below in order - M1, M2, M3, M4)

JUNGLE

SW Engineering
CSC 648/848
Fall 2022
Milestone 1
Team 2

TEAM LEAD	Nanda Pandian (npandian@mail.sfsu.edu)
GITHUB LEAD	Troy Carloni
FRONT-END LEAD	Farid Mehdi Pour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu
DATE SUBMITTED	10/07/22
DATE REVISED	10/14/22, 10/28/22

Executive Summary

Jungle is a platform used to sell San Francisco State University graphic media, eTextbooks, and a few miscellaneous items. SFSU media includes videos, photos, and other visual content. The purpose of our application is to bring convenience to the customer's finger tips. Oftentimes SFSU students and faculty are struggling to find time between their class or work schedules to get certain media. Our application allows customers to easily purchase and get access to SFSU's media as they are needed.

Jungle offers plenty of different features to help SFSU SFSU students and faculty navigate our application with ease and get access to the media they want without any additional work. First of, we provide an online store where SFSU students and faculty and faculty can browse or specifically search for media they need. If the media they are requesting for is not sold on our application, then we would provide media of similar interest to help SFSU students and faculty find what they still might want. Next, we provide a simple filter system that helps sort out specific types of media based on SFSU students and faculty' interests. SFSU students and faculty do not need to spend much time learning how to navigate the application to find a specific product. Rather using the filter tool helps with categorizing the type and style of media they need.

Jungle's main goal for its users is to ease their interaction to purchase SFSU graphic media by using our application. Two of our main features that help the users are expanding the search tool to not just a specific media, but media that falls under the similar category and the filter tool. By utilizing the expanded search tool, users are able to add depth to their search. If the media they search for is not specifically there, the application will state that the media is not available followed by media that could be of similar interest to the user. In addition, the filtering tool helps SFSU students and faculty either narrow down what they are searching for or to find exactly what they are shopping for. This tool will help SFSU students and faculty' time from having to randomly browse the application for what they are looking for.

Jungle's unique feature is its ability to filter the product search by SFSU class and professor names. Oftentimes SFSU students and faculty have a tough time trying to find out which media belongs to which class and professor. Therefore, Jungle helps narrow down SFSU students and faculty' search to find what they need by adding another filter option.

Our team is a group of six members formed in the course CSC 648/848 at San Francisco State University. Our team's focus is to minimize students' time to get access to SFSU media by just a few clicks on our application. Our team is built to handle any crisis and set goals and strategies often to remind ourselves of our project mission.

Personas

1. About John

He is a student who wants to find medias that his classes required.

Wants to use a new SFSU website

Is patient to use WWW apps

Has basic knowledge for using WWW apps



Goals and scenario:

Browse the website, and wants to browse all items

Was attracted by the items that the website displayed.

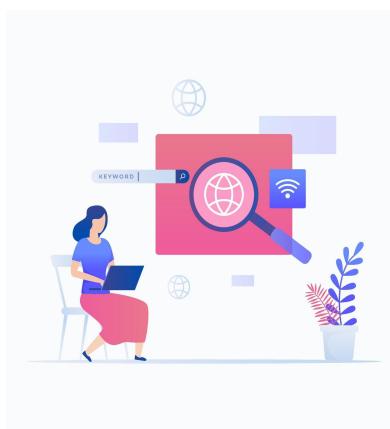
Since he wants to place orders online, he wants to create an account to contact sellers.

2. About Mary

She is a student who wants to sell the media she doesn't need anymore.

Cares feedback on her postings of selling.

Cares about her privacy so doesn't want to give her home address.



Goals and Scenario:

Selling her post.

Needs a secure transaction method that keeps her address private.

If she decides to sell her post, she also will care about her own safety as well as her selling.

3. About Levi

Is familiar with how to use the website to achieve the desired effect

Cares that whether his posting meets the site's specifications since his posting must be reviewed by the admin, he needs to be careful that he is communicating appropriately.



Goals and Scenario:

He wants to sell his items on this website.

If he wants to sell his items more efficiently, He will learn about how to use these WWW apps.

4. About Elise and Alex

Elise is the 1st admin

Alex is the 2nd admin

Elise created an account for Alex.

Alex's account has a temp username and password, so he needs to change them when he logs in, to make this account private.

Alex needs to review posts, and he has the right to deny posts if they don't meet guidelines of the website.

Always navigates to the sites page that outlines posting guidelines to check things out.



Goals and Scenario:

Alex needs an admin account to work on reviewing posts. He also will use the site pages that have posting guidelines to help him checkout the posting guidelines.

Use Cases

1. first time site goer, buys item from post

"John" hears about the new SFSU website and wants to check it out. He navigates to the site with no item in mind and wants to browse. Unfortunately he cannot find a way to browse all items and goes to browse by category. He finds something he wants and goes to message the seller, he is then prompted to create an account in order to proceed. He creates an account and is then returned to the screen of the item he was browsing. He contacts the seller they arrange to meet at a location, and the transaction is carried out.

2. Registered user, sells item

A user of the site "Mary" logs into the site to check if she has any messages in regards to her postings. She sees she does have a message for one of her postings. She replies and the buyer asks for her address, however "she" is not comfortable giving her home address and instead recommends meeting at one of the predetermined locations specified on the website. They agree upon a location and the transaction is carried out. "she then goes back to the site and records that her post has been sold.

3. Unregistered user, creates post

"Levi" wants to create a new posting on the site. He goes to create a post, provides a summary of the item and an asking price. After filing everything out he then goes to post and is prompted to first create an account. After creating an account he is now able to post, Prior to posting he is prompted w/ the guidelines for appropriate posts and then finalizes the post. He receives a notification that his post will be reviewed by an admin and will be posted once approved.

4. Admin, admin responsibilities

"Elise" is an admin for the site, they just hired an additional admin "Alex" and needs to create an admin account for her. Elise navigates to create an admin account option and completes the steps in order to create an account. Elise provides Alex with temp username and password. Alex logs in and changes her username and password. She then starts reviewing posts that require approval prior to posting. She finds a post she believes does not abide by the post guidelines. She navigates to the site's page that outlines posting guidelines. She finds that she is correct and it is not appropriate so she declines the posting, sending a notification to the user that their posting was denied.

Data Glossary

- Unregistered User:
 - can view items
 - can search items
 - create registered account
- Registered User:
 - Create/Manage posts
 - contact other users in regards to posts
 - inherits all properties of unregistered users
- Admin:
 - Required to approve/disapprove registered users posting
 - can create new admin accounts
 - inherits all properties of Registered User
- Posts:
 - contain information regarding the item posted, item description, category, asking price, image of item.
 - Also will have an associated message inbox which will store communication between users in regards to the associated post.
 - Posts will be associated with the user that posted them.
- User registration record:
 - contain information regarding registered user
 - email
 - first/last name

Functional Requirements

- Unregistered User
 - Shall be able to search for posts
 - Shall be able to browse posts
 - Shall be able to perform general browse
 - Shall be able to view post details
 - Shall be able to create registered account
- Registered User
 - Shall inherit properties of unregistered user
 - Shall be able to create posts
 - Shall be able to edit their posts after submitting
 - Shall be able to send and receive post messages
 - Shall be able to remove their own posts
 - Shall be able to see posts they have contacted without having to search
 - Shall be able to message other registered users in regards to their posting
- Admin
 - Shall inherit properties of Registered User
 - Shall be required to approve/disapprove registered users postings
 - Shall be able to create Admin accounts for new Admins

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, Fall 2022. For Demonstration Only" at the top of the WWW page nav bar. (Important so as to not confuse this with a real application).

Competitive Analysis Chart

Feature	Competitor A "Getty Images"	Competitor B "Shutterstock"	Competitor C "Adobe Stock"	Our future product
Text Search	++	+	+	+
Browse	++	+	++	+
Simplicity	+	+	+	++
Sorting by	+	+	++	++
School Related Filtering	-	-	-	++

Simplicity:

The two most frequently stuck websites are "Getty Images" and "Shutterstock." For every category, they provide graphics, videos, music, and editing tools. It leads to a complicated user interface, yet they still manage to design it so that people can search for what they're looking for and pursue the product from the home page at the same time.

For our future website, we'll simplify and compress the home page: We'll have a well-organized categories section based on the classes or majors, and even based on the clubs.

Although the visitor can still browse the products, our main goal is to avoid dumping unnecessary products on the home page. It would be better organized, allowing for a more convenient experience for the user rather than being overrun with items they are not interested in.

Sorting by:

Since we interact with San Francisco State University, it would be beneficial to add a sort based on semesters and years. We would also have different filters based on classes or majors so that people can find their content more easily. This comes in especially handy when looking for specific images for each class.

Browse:

All of these similar websites have great browsing features. They have clear layouts and are organized into clear categories so that people can find what they are looking for much easier. Their images are all clear and organized and we will do this with our product too. We will look to implement new designs and layouts when browsing through our website to keep things looking fresh.

Text Search:

Services like Getty Images has a very good search tool where you can search very specific events and will provide images and content on a very specific search.

School Related Content:

Our product will be completely catered toward San Francisco State University. We will allow all people from SF State to submit items to the media catalog so that there can be content there specifically from the school. Since these other services have mostly stock photos this will be great when people who go to SF State need images or media that comes directly from their own surroundings. Since all these other products/services aren't specifically tailored to schools they will not have the kind of content that will be present on ours.

High-level system architecture and technologies used

SERVER HOST	Google Compute Engine 1vCPU 2 GB RAM
OPERATING SYSTEM	Ubuntu 22.0.4
SERVER DATABASE	MySQL v8.0
WEB SERVER	Apache 2.4.54
SERVER-SIDE LANGUAGE	Javascript v16.17.0
ADDITIONAL TECHNOLOGIES	VScode

Team and roles

TEAM LEAD & DOCUMENT MASTER	Nanda Pandian
GITHUB MANAGER	Troy Carloni
FRONT-END LEAD	Farid Mehdipour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu

Checklist (*completed by Team Lead - Nanda*)

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

JUNGLE

SW Engineering

CSC 648/848

Fall 2022

Milestone 2, Part I

Team 2

TEAM LEAD	Nanda Pandian (npandian@mail.sfsu.edu)
GITHUB LEAD	Troy Carloni
FRONT-END LEAD	Farid Mehdi Pour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu
DATE SUBMITTED	10/28/22
DATE REVISED	TBD

Executive Summary

Jungle is a platform used to sell San Francisco State University graphic media, eTextbooks, and a few miscellaneous items. SFSU media includes videos, photos, and other visual content. The purpose of our application is to bring convenience to the customer's finger tips. Oftentimes SFSU students and faculty are struggling to find time between their class or work schedules to get certain media. Our application allows customers to easily purchase and get access to SFSU's media as they are needed.

Jungle offers plenty of different features to help SFSU SFSU students and faculty navigate our application with ease and get access to the media they want without any additional work. First of, we provide an online store where SFSU students and faculty and faculty can browse or specifically search for media they need. If the media they are requesting for is not sold on our application, then we would provide media of similar interest to help SFSU students and faculty find what they still might want. Next, we provide a simple filter system that helps sort out specific types of media based on SFSU students and faculty' interests. SFSU students and faculty do not need to spend much time learning how to navigate the application to find a specific product. Rather using the filter tool helps with categorizing the type and style of media they need.

Jungle's main goal for its users is to ease their interaction to purchase SFSU graphic media by using our application. Two of our main features that help the users are expanding the search tool to not just a specific media, but media that falls under the similar category and the filter tool. By utilizing the expanded search tool, users are able to add depth to their search. If the media they search for is not specifically there, the application will state that the media is not available followed by media that could be of similar interest to the user. In addition, the filtering tool helps SFSU students and faculty either narrow down what they are searching for or to find exactly what they are shopping for. This tool will help SFSU students and faculty' time from having to randomly browse the application for what they are looking for.

Jungle's unique feature is its ability to filter the product search by SFSU class and professor names. Oftentimes SFSU students and faculty have a tough time trying to find out which media belongs to which class and professor. Therefore, Jungle helps narrow down SFSU students and faculty' search to find what they need by adding another filter option.

Our team is a group of six members formed in the course CSC 648/848 at San Francisco State University. Our team's focus is to minimize students' time to get access to SFSU media by just a few clicks on our application. Our team is built to handle any crisis and set goals and strategies often to remind ourselves of our project mission.

Data Items and Entities

- Unregistered User
- Registered User:
 - Email (Key)
 - login/Username
 - Alphanumeric
 - unique
 - password
 - encrypted
 - Alphanumeric
- Admin:
 - Email (Key)
 - login/Username
 - Alphanumeric
 - unique
 - password
 - encrypted
 - Alphanumeric
- Posts:
 - Post_id (Key)
 - unique
 - Email (foreign Key)
 - Alphanumeric
 - unique
 - Category
 - Alphanumeric
 - Description
 - Alphanumeric
 - Asking Price
 - numeric (\$0.00)
 - Image (not required)
 - can be null

- jpeg, png
- Message Inbox:
 - Post_id (Foreign Key)
 - unique
 - Message
 - Alphanumeric

Functional Requirements V2

Here I expand the M1's functional requirements into M2, with the priority of each requirement with 1, 2, 3(1-must have; 2 – desired; 3 – opportunistic as defined in the class).

-Priority 1

- Unreg user

Shall be able to create temp registration account

Shall be able to perform general browse

Shall be able to browse by category

- Reg user

Shall be able to create posts

Shall be able to send and receive post messages

Shall be able to see posts they have contacted without having to search

Shall be able to message other registered users in regards to their posting

Shall be able to edit their posts after submitting

Shall be able to remove their own posts

- Admin

Shall be required to approve/disapprove registered users postings

-Priority 2

- Unreg user

Shall be able to search for posts

Shall be able to browse posts

- Reg user

Shall inherit properties of unregistered user

- Admin

Shall inherit properties of Registered User

-Priority 3

- Unreg user

Shall be able to view post details

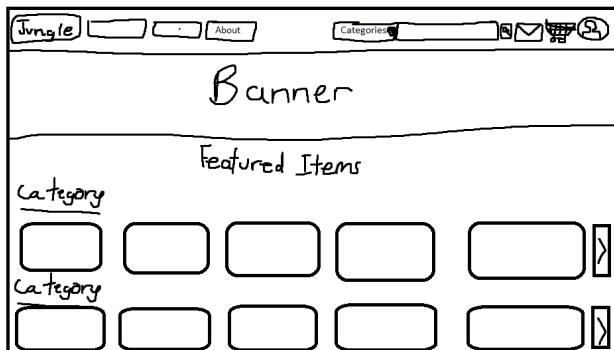
- Reg user
- Admin
- Shall be able to create Admin accounts for new Admins

○

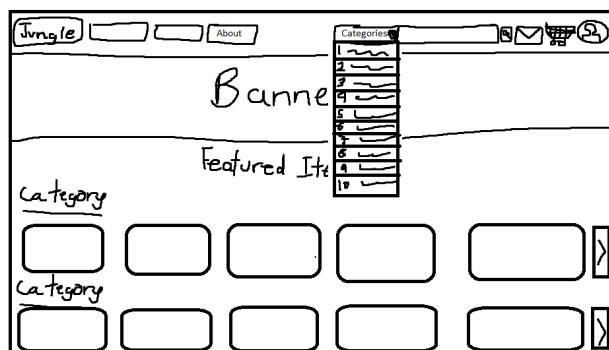
UI Storyboards for each main use case

Use Case 1:

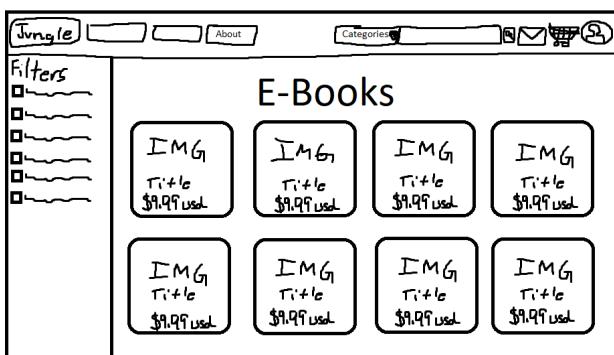
Home->Categories->Browse->Item Page->Register->Message



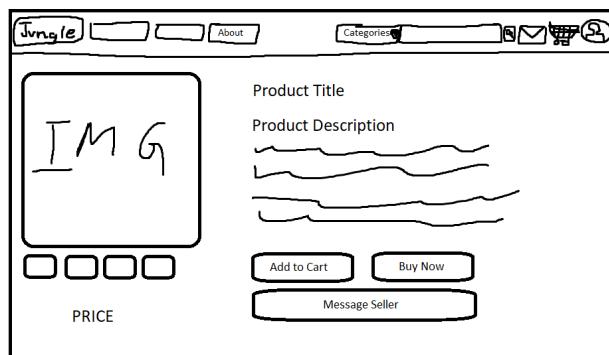
(HOME)



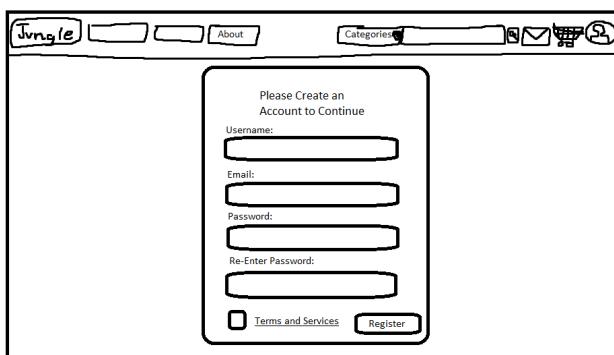
(CATEGORY PULL DOWN MENU)



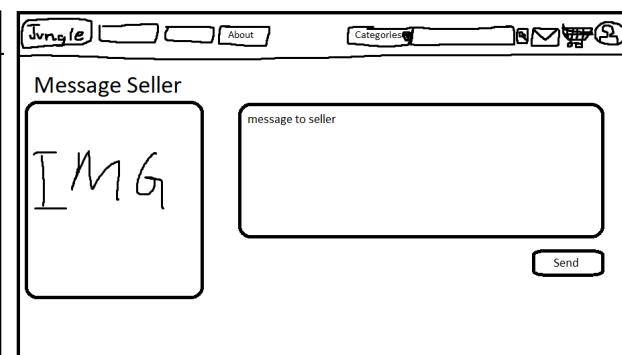
(BROWSE PAGE)



(ITEM PAGE)



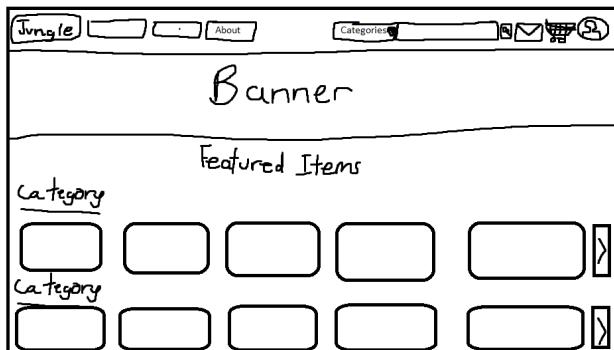
(REGISTRATION)



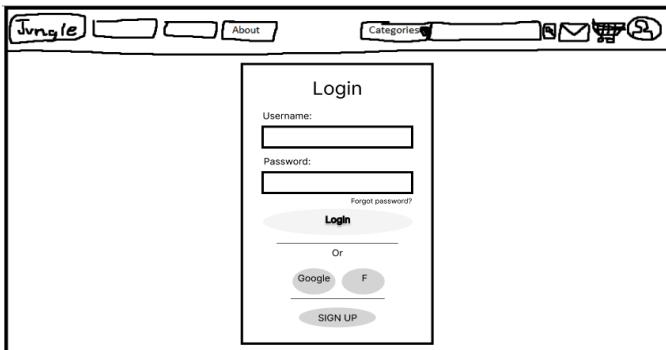
(MESSAGE)

Use Case 2:

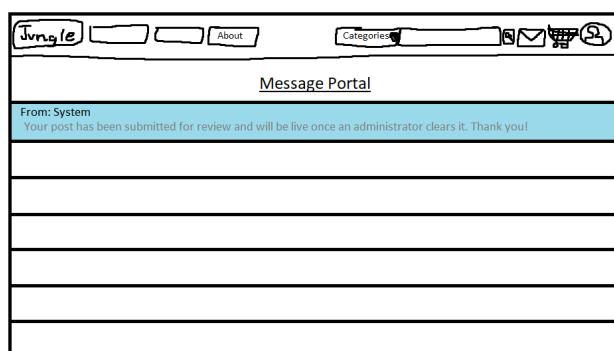
Home->Login->Message Portal->Message->Schedule Meetings->Edit Posts



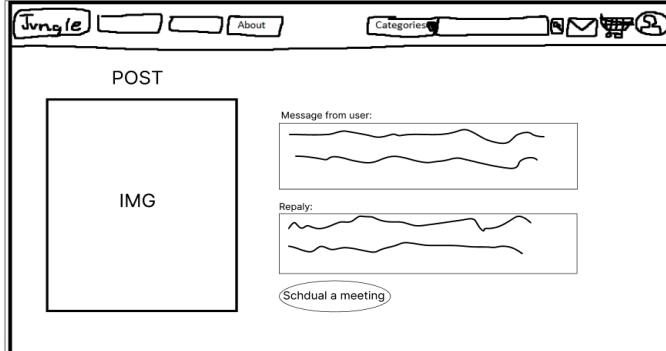
(HOME)



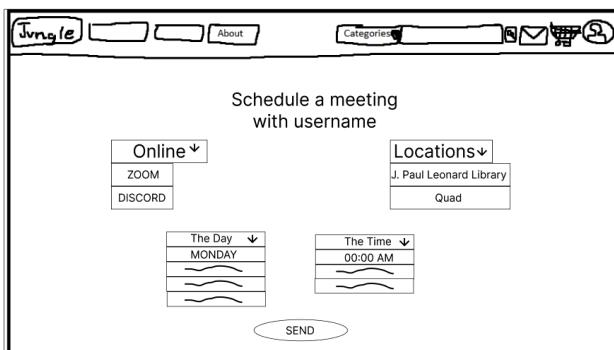
(LOGIN)



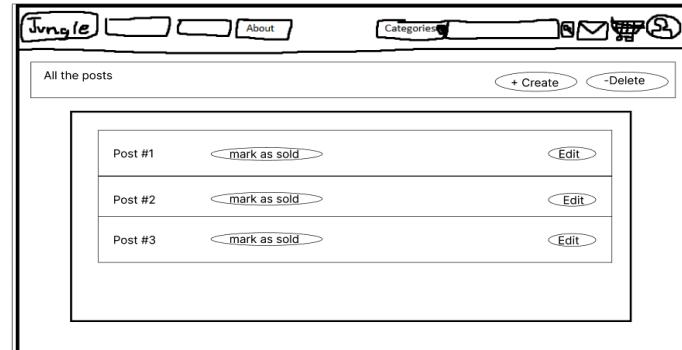
(MESSAGE PORTAL)



(Message)



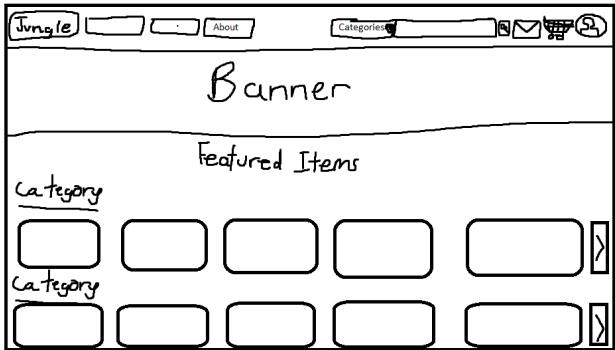
(Schedule Meetings)



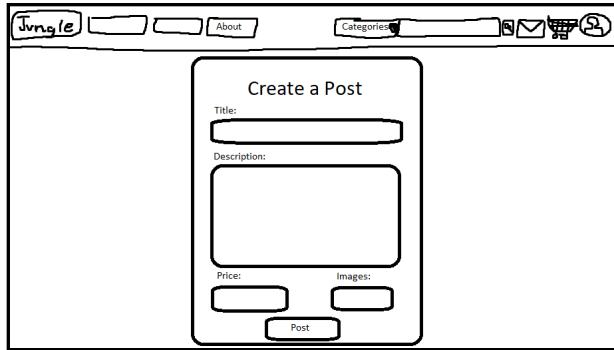
(Edit Posts)

Use Case 3:

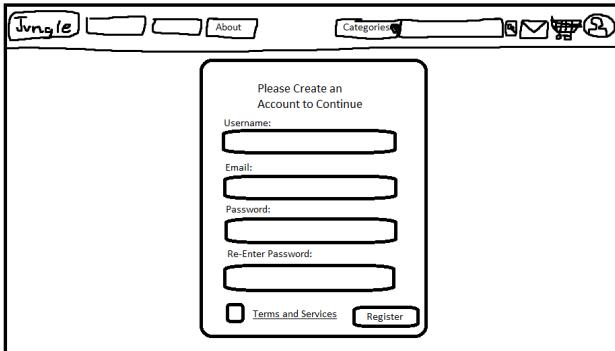
Home->Create Post->Register->Create Post->Guidelines->Message Portal



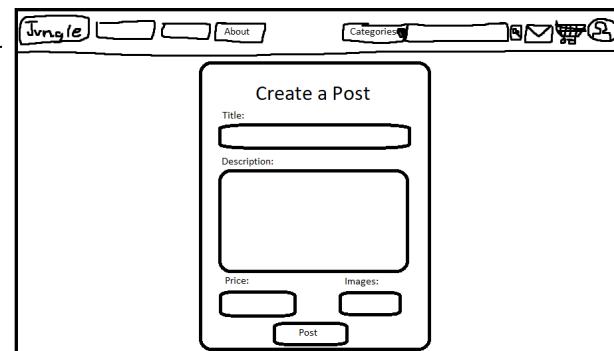
(HOME)



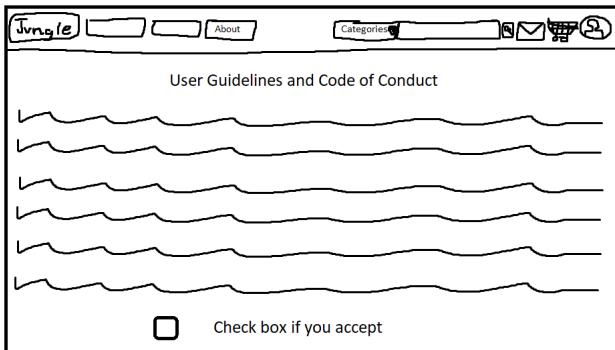
(CREATE POST)



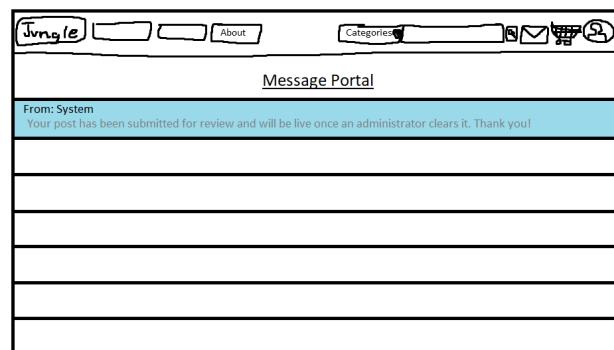
(REGISTER)



(CREATE POST)



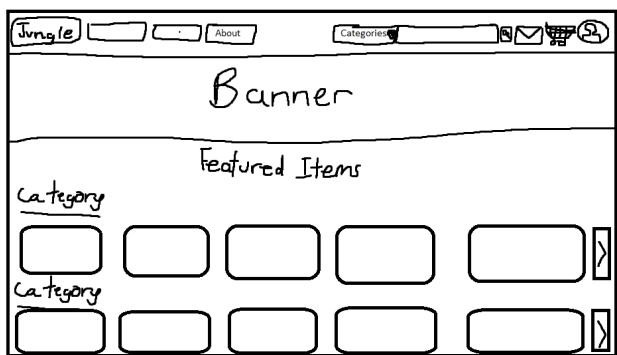
(USER GUIDELINES)



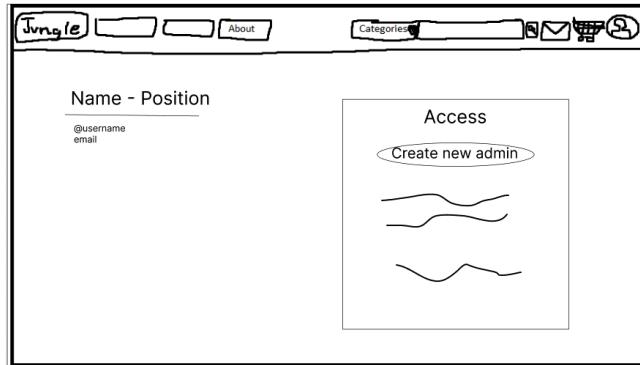
(MESSAGE PORTAL)

Use Case 4:

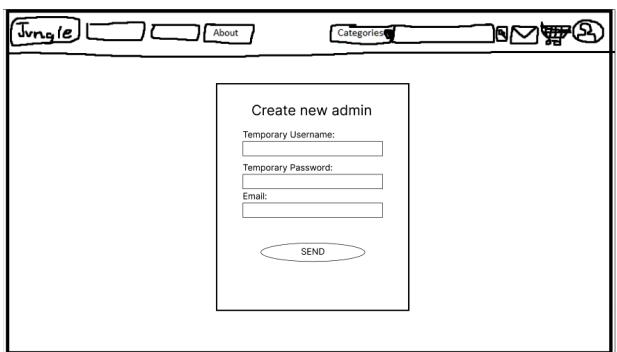
Home->Admin->Create admin->Login->Change Username & Password->Post Statuses->Guidelines



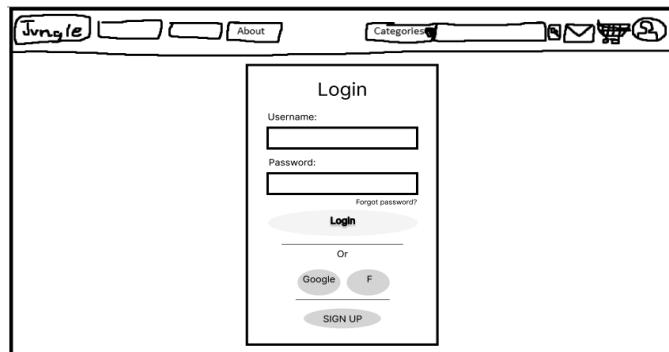
(HOME)



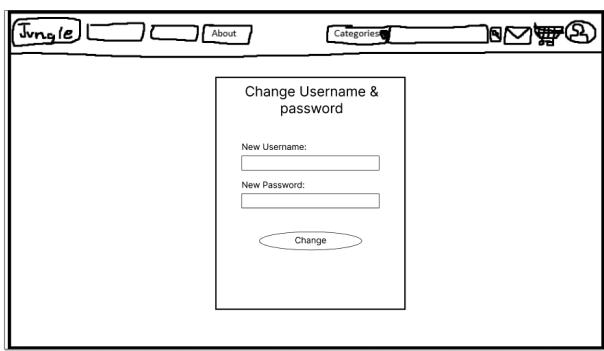
(Admin)



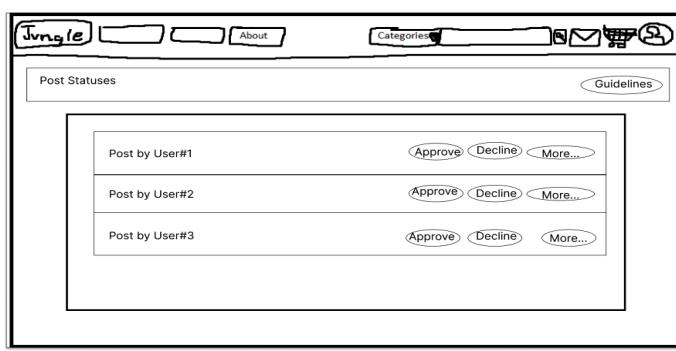
(Create admin)



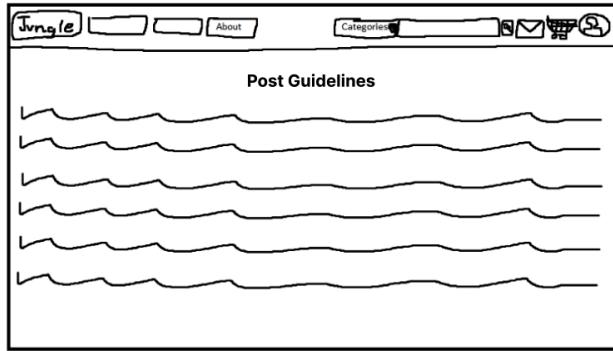
(Login)



(Change Username & Password)



(Post Statuses)



(Guidelines)

High-level system architecture and technologies used

SERVER HOST	Google Compute Engine 1vCPU 2 GB RAM
OPERATING SYSTEM	Ubuntu 22.0.4
SERVER DATABASE	MySQL v8.0
WEB SERVER	Apache 2.4.54
SERVER-SIDE LANGUAGE	Javascript v16.17.0
ADDITIONAL TECHNOLOGIES	VScode

Key Risks

There are no high-level risks at the moment for our project, but there are some small risks we are facing either as a team or individually. For instance, some of the team members are new to MySQL, so understanding how the DB works and more of the back-end takes longer to comprehend. However, we set time for the back-end to educate the rest of the team on the back-end tasks, so the team is knowledgeable on it and gets the opportunity to ask questions, rather than having to learn it on our own. Next, working with git is a new skill for some of our team members, so sometimes there might be a confusion on how some file/code should be committed. Once again, those members who are thorough with git, step up to work with the others who have questions. As for the schedule risks, we have not faced many because we communicate well to each other. If a team member is ill or can not show up for any reason, the tasks are immediately delegated to other members to lessen the weight for a temporary period. Since the team is good with informing each other of any hurdles, there has not been any road bumps for meeting deadlines. Lastly, regarding legal/content risks, since we are still a bit early on with our application, we have been using placeholder content for the media. We are starting to look into proper licensing and will explore.

Project Management

First, the team scanned through the entire Milestone 2 document, including Part II. The team lead went through the milestone 2 requirements and what each task requires as a group. As we ran through the set of tasks, the team lead delegated them respectively to each of the team members. We set a soft deadline for each of the development groups (front-end, back-end, devs) by our next virtual meeting. During our virtual meeting, we discussed our statuses on the milestone 2 tasks and worked through on any issues the groups had. The entire team stayed on call despite which group the issue is from, so that the entire team is knowledgeable about what is going on with each other. At the end of the meeting, we set another soft deadline by the next class meeting to ensure that the feedback and corrections made in the previous meeting are enhanced. During our class meeting, the team lead checked in with each team member on their status with their tasks and making sure front-end and back-end members are synchronized. Finally, at the end of our class meeting, we set a hard deadline to have the tasks completed and polished by our next virtual meeting, in which we will review as a team and submit the milestone. In addition to that, the team lead has delegated tasks for milestone 2, Part II and planned to discuss in detail at our next meeting.

Team and roles

TEAM LEAD & DOCUMENT MASTER	Nanda Pandian
GITHUB MANAGER	Troy Carloni
FRONT-END LEAD	Farid Mehdipour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu

Checklist for M2, Part I (*completed by Team Lead - Nanda*)

<p>So far all team members are engaged and attending ZOOM sessions when required</p>	<p>DONE</p>
<p>Team found a time slot to meet outside of the class</p>	<p>DONE</p>
<p>Back end, Front end leads and Github master chosen</p>	<p>DONE</p>
<p>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</p>	<p>DONE</p>
<p>Team lead ensured that all team members read the final M2 and agree/understand it before submission</p>	<p>DONE</p>
<p>Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)</p>	<p>DONE</p>

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

Team number: 02

Meeting date: Nov 16

- • **Summary of feedback on UI (record all pages that need revision)**
 - *REGISTRATION PAGE*
 - No need to check for 13+
 - “TOS and Privacy Rules” should be linked to a dummy link
 -
 - *LOGIN PAGE*
 - Remove “form” from the Login Form title
 - *HOME PAGE*
 - Add introduction to the company instead of just “Jungle” blindly stated
 - Logo for Jungle
 - Differentiation between nav-bar and the products listed (for example a line)
 - Search and category in the nav bar must be connected
 - Move LOGIN to the right side of the nav-bar
 - Design a “card-layout,” not just a dump of images and their info
 - Have a button in the card to access for more info
 - Remove “Item for sale” for every product; make it a generic title or something
 - Must have an option to Create Post directly on this page
- • **Summary of feedback on code and architecture**
 - App structure is good so far
 - Express and ejs used and deployed; working
 - File paths are working
 - Passowrds are hashed
 - Thumbnails are being used
- • **Summary of feedback on github usage**
 - Should have a backend and frontend directory, in which the respective folders should be moved to
 - Improve commit comments (too vague by the prof)
 - Check for proper indentation
 - Must have header comments for each page
- • **Summary of feedback on DB**
 - Should have a live in the DB (True if LIVE)
 - Display when user is created
- • **Summary of feedback on teamwork and risk management**
 - Overall good, but team lead should push team more towards hard deadlines
 - More team communication that is clear to everyone

- Confirm that you have done architecture review to check that developers adhere to MVC pattern, coding style, minimal agreed documentation etc. Record if OK or list the issues found. Request developers follow up on corrections and follow up later by doing code reviews. OK
- 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
 1. Unregistered users are able to create temp registration account
 2. Unregistered users are able to search items
 3. Unregistered users are able to perform general browse
 4. Unregistered users are able to check product details
 5. Unregistered user are able to browse by category
 6. Registered users are able to log in their account
 7. Registered users are able to create posts
 8. Registered user are able to change email
 9. Registered user are able to change password
 10. Registered user are able to message other registered users in regards to their posting

- Any other comments and issues

- Information in credential folder needs to be updated
- Access to DB not given in credential folders
- Check Point (CP) if given, DUE: 11-22-22

JUNGLE

SW Engineering
CSC 648/848
Fall 2022
Milestone 4
Team 2

TEAM LEAD	Nanda Pandian (npandian@mail.sfsu.edu)
GITHUB LEAD	Troy Carloni
FRONT-END LEAD	Farid Mehdipour
BACK-END LEAD	Richard Aguilar
SOFTWARE DEVELOPER (1)	Hugo Moreno
SOFTWARE DEVELOPER (2)	Yuwei Liu
DATE SUBMITTED	12/09/22
DATE REVISED	TBD

Product Summary

Name: Jungle

Description of product:

Jungle is a platform used to sell San Francisco State University graphic media, eTextbooks, and a few miscellaneous items. SFSU media includes videos, photos, and other visual content. The purpose of our application is to bring convenience to the customer's finger tips. Oftentimes SFSU students and faculty are struggling to find time between their class or work schedules to get certain media. Our application allows customers to easily purchase and get access to SFSU's media as they are needed.

Itemized list of ALL major committed functions:

1. Unregistered users are able to create temp registration account
2. Unregistered users are able to search items
3. Unregistered users are able to perform general browse
4. Unregistered users are able to check product details
5. Unregistered user are able to browse by category
6. Registered users are able to log in their account
7. Registered users are able to create posts
8. Registered user are able to change email
9. Registered user are able to change password
10. Registered user are able to message other registered users in regards to their posting

What is unique in your product:

Jungle's unique feature is its ability to filter the product search by SFSU class and professor names. Oftentimes SFSU students and faculty have a tough time trying to find out which media belongs to which class and professor. Therefore, Jungle helps narrow down SFSU students and faculty' search to find what they need by adding another filter option.

URL to your product:

<http://35.233.185.244:8080/>

Usability Test Plan

Test Objectives:

For our usability test plan we decided to use our create post function feature. In this feature we will be testing the ease of creating a post. Specifically how fast it is to create the post, how easy the create a post tool is to use and how clear the feature itself is to use.

Test Objectives and Setup:

Starting Point: Users will begin at the home page of our program and will need to find and navigate to create a post page themselves.

Intended Users: The intended users for this page are San Francisco State University students who want to sell their digital media on our website. This feature is also only available for users who have already registered on our website. If you do not have an account you will be sent to the registration page.

URL to Create Post Page for students who want to sell: <http://35.233.185.244:8080/createPost>

What is to be measured: For this usability test we are going to have the user's answer a survey with a series of statements and give them multiple answer options including Strongly Agree, Agree, Neutral, Disagree, and Strongly Disagree.

Usability Task Description:

For this task you will create and publish a post on the Jungle website which you will then later evaluate. Creating a post requires you to set a title, description and give a price along with including an image of your product

Evaluation of Effectiveness:

To evaluate effectiveness we will need to see if the user was successfully able to create a post with a title, description price and photo. If the user was able to successfully create the post we would consider the feature effective.

Evaluation of Efficiency:

To evaluate efficiency we would first need to know if the user was able to complete the task in a timely manner and without any complication. We would also need to know if the user was able to

complete the task easily with a few amount of clicks. Lastly we would need to know if the task itself was easy and simple to understand.

Evaluation of User Satisfaction:

Prompt 1:

The task of creating and publishing a post was simple and easy to complete. *

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

Prompt 2:

Creating a post was quick and efficient. *

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

Prompt 3:

The instructions for the task were clear and easy to understand. *

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

QA Test Plan

Using the same function as the usability test, below is the QA test plan.

Test objectives - what is being tested:

- Search
- Ensure all methods of search queries function correctly
- Ensure search is protected against sql injection

HW and SW setup (including URL):

- Browsers: Microsoft Edge, Google Chrome
- URL: <http://35.233.185.244:8080/>

Feature to be tested:

- Search

QA Test plan: Search

#	Title	Description	Test Input	Expected Output	PASS/FAIL
1	All products	Query DB for all posts through search bar	Leave search bar input field empty and category should be set to “All”	Get 5 results. Name of expected items; “test”, “horror”, “literature”, “movietime”, “America”	Edge: PASS Chrome: PASS
2	Input field	Use search bar input field to search for post	In search input field type “test” and hit enter	Get 1 result. Result should be “test” with category “video” and price “\$80.00”	Edge: PASS Chrome: PASS
3	category option	Use category option to search for posts	Leave search bar input field empty and category should be set to “audio”	Get 1 result. Result should be “america” with category “Audio” and price “\$80.00”	Edge: PASS Chrome: PASS
4	Input field and category option	Use search bar input field and category option to search for posts	Set category option to “Video” and type “test” in search bar input field	Get 1 result. Result should be “test” with category “video” and price “\$80.00”	Edge: PASS Chrome: PASS
5	Sql Injection	Attempt to manipulate DB with sql injection in search bar	Set category option to “All” and type “SELECT * FROM Users” in search bar input field	Get “No Post Found, Showing All Posts”, and will show all 5 posts.	Edge: PASS Chrome: PASS
6	Character limit	Attempt a search with more than 45 characters	In search bar input field type “1” 41 times.	Get zero results. “exceeded character search length” should show on page	Edge: PASS Chrome: PASS
7	No post found	Search for a post that does not exist	In the search bar input field type “helloworld”	Get “No Post Found, Showing All Posts”, and will show all 5 posts.	Edge: PASS Chrome:PASS

Code Review

Email sent to teammate for code review

CSC 648-848 Fall 2022 Milestone 4 Code Review

↶ ↵ →



✉ Nanda Sivagnanam Pandian <npandian@mail.sfsu.edu>
To: ✉ Farid Mehdipour

Today at 10:21 42AM

Hi Farid,

Hope all is well.

I was hoping you could take a look at a portion of code that I'd like to get some feedback on. Due to security reasons, I will not be sending the code directly through email. So, please go to the repo link and follow the code path as stated below. The code highlighted is what I'd like to get reviewed on. Please let me know if there are any questions or issues finding the specific code.

Repo link: <https://github.com/CSC-648-SFSU/csc648-03-fa22-team02>

- A. **application >> myapp >> routes >> index.js**
 - a. inside index.js, please find the `router.get('/search')`
- B. **application >> my app >> public >> javascript >> frontend.js**
 - a. inside of frontend.js, please find `searchNav`

Thank you for your time and help!

Best regards,
Nanda

Email response (1): Code review performed by teammate

Re: CSC 648-848 Fall 2022 Milestone 4 Code Review

↶ ↷ →



✉ Farid Mehdipour <fmehdipour@mail.sfsu.edu>

To: ✉ Nanda Sivagnanam Pandian

Today at 10:27 25AM

Hi Nanda,

Hope all is well too.

Thanks for reaching out, I would be happy to do the code review, and please let me know if you have any questions in this process.

The search in the Navigation bar has three sections:

1. The first part is categories, where the users choose what kind of element they are looking for, these categories are:
 - ebook
 - Video
 - Image
 - Audio
2. The middle part or the second part is where the user enters the name of the product.
3. The third part is the button that executes the search.

The following functions are what make the searchNav execute accordingly.

Repository: application/myapp/routes/index.js

The function `router.get('/search', async (req, res, next)` is the get function that receives the `search_term` that the user enters:

In order to determine whether the search term and category are valid inputs, it first accepts them. If they are, it then takes the category input and search term, retrieves the result from the database, and passes it back.

This is how the query Prompt for the DB would appear.

1) If a user requests all the components in a particular search name:

```
'SELECT * FROM Posts WHERE name LIKE '%${search_term}%' AND approved = 1';
```

2) If the users want all the elements inside of DB with no specification on the category or the search term:

```
'SELECT * FROM Posts WHERE approved = 1'
```

3) Output all the elements in that category if the user only specified the category and not the search term:

```
'SELECT * FROM Posts WHERE category = "${category}" AND approved = 1'; }
```

4) Take all the outputs and pass them to the DB if neither of these options is chosen:

```
'SELECT * FROM Posts WHERE category = '${category}' AND (name LIKE '%${search_term}%'') AND approved = 1'; '
```

Email response (2): Continuation of code review performed by teammate

Re: CSC 648-848 Fall 2022 Milestone 4 Code Review

← ↵ →



✉ Farid Mehdipour <fmehdipour@mail.sfsu.edu>

Today at 10:28 29AM

To: ✉ Nanda Sivagnanam Pandian

Repository: application/myapp/public/javascript/frontendjs.js

There are three functions inside of the frontendjs.js that are components of SearchNav's front end.

The submit button on the search initiates the execution of this function initially.

This function is the first one to receive the search phrase and category inputs before passing them on to the following function.

By using getElementById, it retrieves the elements from the HTML (user interface), examines their value, and saves them. Additionally, it develops searchURL using the values in the

`'/search?search=${searchTerm}&category=${categoryText}'`

After that, it fetches the data on that URL and creates the cards which are going to be the product thumbnails in the search result.

executeSearchNavbar():

This function takes the search term and the category inputs and passes them to the backend in the HTTP format of `/result-page?search=\${searchTerm}&category=\${categoryText}`

Where we navigate the location of the window to that URL to it.

Before that, it also checks the URL to see if it is validated.

executeSearchNavbar_onPage():

This function gets the URL parameter on the location of the window tab, then takes the two variable of search term and category from that URL and pass it back as a new URL:

`'/search?search=${searchTerm}&category=${categoryText}'`

The category value is then examined to determine its length and the number of post products that can be returned to the user.

With that knowledge, we can determine how many card products to display, so we execute a retrieve function, display the number of results, and then feed the remaining product information back to the build card method (this function creates the product post thumbnails on the screen)

Self check on best practices for security

Asset to be protected	Types of possible/expected attacks	Your strategy to mitigate/protect the asset
User Login Credentials	<ul style="list-style-type: none"> ● SQL Injection ● Leaking user credentials from malicious parties 	<ul style="list-style-type: none"> ● To prevent a SQL Injection attack the queries executed use SQL Parameters to prevent the possibility of a malicious person to ruin the database . ● To prevent major leaks of user credential data the password is hashed once going to the database. The reason it isn't encrypted is because an encryption can be decrypted. To assure full safety hashing the password is best.
Search bar	<ul style="list-style-type: none"> ● SQL Injection ● XSS Attack 	<ul style="list-style-type: none"> ● Similar to the login credential and SQL Injection takes advantage of how SQL queries or executes within the code. Similar to the login credentials I will use SQL parameters because they are known to prevent these attacks.
Post Creation	<ul style="list-style-type: none"> ● XSS Attack ● Users post malicious content or spam the website with posts that violate the website's terms and conditions they agreed to. 	<ul style="list-style-type: none"> ● XSS is usually a foreign party trying too input a script in an input area within the website to access information regularly not allowed. To prevent this any post a user wants to add will be

		validated before added to the website which will prevent bad actors from accessing information
Changing user credentials	<ul style="list-style-type: none"> Bad actor would want to change the password or email of victims account 	<ul style="list-style-type: none"> To prevent the attempt to change the users password they will have to have access to their account. When changing the password you will be asked to enter the old password, new password and to confirm the new password. With the changing of email or password when forgotten the user will email the website support for the request

Self-check original Non-functional specs (*performed by team lead*)

NON-FUNC SPEC	DONE	ON TRACK	ISSUE
Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0	YES		
Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers		YES All the pages render their sizes correctly, except the CSS is a little off. So the company banner or the feed page will not be centered perfectly. Front end team is on the fix!	
All or selected application functions must render well on mobile devices		YES All the pages are rendering their sizes proportionally to the mobile device, except About & Profile pages. Front end team is debugging the code to find the minor error.	
Data shall be stored in the database on the team's deployment server.	YES		
No more than 50 concurrent users shall be accessing the application at any time	YES		
Privacy of users shall be protected	YES		

The language used shall be English (no localization needed)	YES		
Application shall be very easy to use and intuitive	YES		
Application should follow established architecture patterns	YES		
Application code and its repository shall be easy to inspect and maintain	YES		
Google analytics shall be used	YES		
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	YES		
Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.	YES		
Site security: basic best practices shall be applied (as covered in the class) for main data items	YES		
Media formats shall be standard as used in the market today	YES		
Modern SE processes and practices shall be used as specified in the class,	YES		

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

Product Screenshots

Home Page:

The screenshot shows the homepage of the Jungle platform. At the top, there is a navigation bar with links for 'Profile', 'About', 'Create Post', and 'Login'. Below the navigation bar is a search bar with dropdown options 'All' and 'Search...'. The main header features the word 'Jungle' in a large, stylized font, with the subtitle 'Where you can sell or share your projects Exclusively for students' underneath. Below this, a sub-header reads 'SFSU Software Engineering Project CSC 648-848, Fall 2022 For Demonstration Only'. The page displays four project cards arranged in a grid. The first card, titled 'post', shows two human skeletons with internal organs highlighted in blue. It is categorized as an eBook and priced at \$80.00. The second card, titled 'test', shows a dark, abstract image of a landscape. It is categorized as a Video and priced at \$80.00. The third card, titled 'horror', shows a red triangular warning sign. It is categorized as an Image and priced at \$80.00. The fourth card, titled 'literature', shows two human skeletons with internal organs highlighted in blue. It is categorized as an eBook and priced at \$80.00. Each card includes a 'CONTACT SELLER' button. At the bottom of the page, a dark footer bar contains the copyright notice 'Copyright © Team 2. All Rights Reserved'.

Login:

The screenshot shows the login page of the Jungle platform. At the top, there is a navigation bar with links for 'Profile', 'About', 'Create Post', and 'Login'. Below the navigation bar is a search bar with dropdown options 'All' and 'Search...'. The main content area features a central 'Login' form. The form has fields for 'Email:' and 'Password:', each with a corresponding input field. Below the password field is a 'LOGIN' button. At the bottom of the form, there are links for 'Create a new account' and 'Forgot Password'. The background of the page is light beige, and the overall design is clean and modern.

Registration:

The screenshot shows the JUNGLE registration interface. At the top, there is a navigation bar with the logo "JUNGLE Team 2", a search bar, and links for "Profile", "About", "Create Post", and "Login". The main content area is titled "Registration". It contains fields for "Email" (with placeholder "...@email.com"), "Password" (with placeholder "!@#"), and "Confirm your Password" (with a redacted placeholder). Below these fields is a checkbox labeled "By checking the box, you agree to our [TOS](#) and [Privacy Rules](#)". A green "Submit" button is at the bottom right of the form.

Create Post:

The screenshot shows the JUNGLE create post interface. At the top, there is a navigation bar with the logo "JUNGLE Team 2", a search bar, and links for "Profile", "About", "Create Post", and "Login". The main content area is titled "Create Post". It contains fields for "Title" (redacted), "Description" (redacted), and "Price" (with placeholder "\$ \$\$"). Below these fields is a dropdown menu for "Choose a category" set to "eBook", a file upload field with placeholder "Browse... No fil...", and a note "New Posts can take up to 24 hours to be approved". At the bottom are two buttons: a green "Create Post" button and a red "CANCEL" button.

User Profile Page:

Screenshot of the User Profile Page. The header includes the logo "JUNGLE Team 2", a search bar, and navigation links for Profile, About, Create Post, and Logout.

Profile

Profile settings section:

- Email: test@example.com (undefined)
- Change Password
- Change Email

PRODUCT TITLE	DATE POSTED	PRICE	VIEW
post	2022-11-22T23:52:50.000Z	\$80.00	View
test	2022-11-22T23:52:50.000Z	\$80.00	View
horror	2022-11-22T23:52:50.000Z	\$80.00	View
literature	2022-11-22T23:52:50.000Z	\$80.00	View
movietime	2022-11-22T23:52:50.000Z	\$80.00	View
america	2022-11-22T23:52:50.000Z	\$80.00	View
Jungle	2022-11-23T03:05:57.000Z	\$0	View

Copyright © Team 2. All Rights Reserved

About Page Directory:

Screenshot of the About Page Directory. The header includes the logo "JUNGLE Team 2", a search bar, and navigation links for Profile, About, Create Post, and Logout.

Team 2

Team Lead NANDA PANDIAN	GitHub Master TROY CARLONI	Backend RICHARD AGUILAR (LEAD) YUWEI LIU	Frontend FARID MEHDIPOUR (LEAD) HUGO MORENO
-----------------------------------	--------------------------------------	---	--

Copyright © Team 2. All Rights Reserved

Search Results:

The screenshot shows a web application interface for a software engineering project. At the top, there is a navigation bar with the logo "JUNGLE Team 2", a search bar containing "Image Search...", and links for "Profile", "About", "Create Post", and "Logout". Below the navigation bar, a banner displays the text "SFSU Software Engineering Project CSC 648-848, Fall 2022" and "For Demonstration Only". The main content area is titled "Showing (2) Out Of (2) Results". It contains two search results cards. The first card features a red triangle icon and is labeled "horror". The second card features a green background with the word "JUNGLE" and is labeled "Jungle". Both cards show the category "Image", price "\$0.00", and a "CONTACT SELLER" button. A "DOWNLOAD" button is also present on the second card. At the bottom of the page, a dark footer bar contains the copyright notice "Copyright © Team 2 All Rights Reserved".

Showing (2) Out Of (2) Results

horror
Category: Image
Price: \$0.00
CONTACT SELLER

JUNGLE
Team 2

Jungle
Category: Image
Price: \$0
DOWNLOAD

Copyright © Team 2 All Rights Reserved

Database Organization

Category Table:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
◆ category_id	int		NO			select,insert,update,references	auto_increment	
◆ category	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		

Message Inbox table:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
◆ message_id	int		NO			select,insert,update,references	auto_increment	
◆ post_id	int		NO			select,insert,update,references		
◆ message_txt	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ date_time	datetime		NO			select,insert,update,references		
◆ message_sender_i...	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		

Post table:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
◆ post_id	int		NO			select,insert,update,references	auto_increment	
◆ email	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ name	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ category	int		NO			select,insert,update,references		
◆ description	varchar(80)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ asking_price	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ image	varchar(80)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ image_thumbnail	varchar(80)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ date_time	datetime		NO			select,insert,update,references		
◆ approved	int		NO			select,insert,update,references		

Users table:

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges	Extra	Comments
◆ email	varchar(45)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ password	varchar(80)		NO	utf8mb4	utf8mb4_090...	select,insert,update,references		
◆ date_time	datetime		NO			select,insert,update,references		

Github Organization

1. Branch Access

- a. Main
 - i. Troy, Nanda
- b. Troy:
 - i. Troy, Nanda
- c. Richard:
 - i. All members
- d. Farid:
 - i. All members
- e. Hugo:
 - i. All members
- f. Yuwei:
 - i. All members
- g. Nanda:
 - i. All members

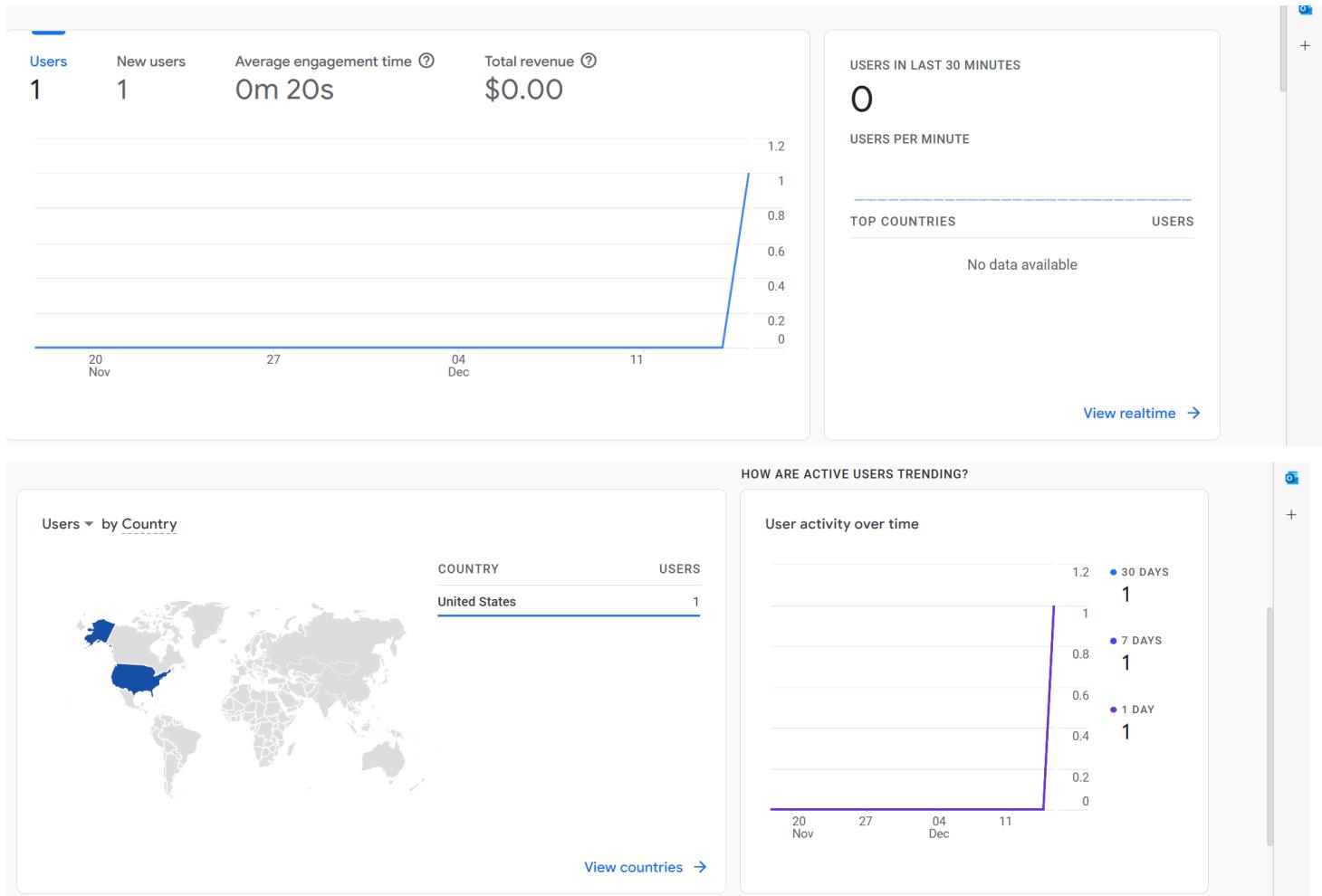
1. Folder structure

- a. Milestones:
 - i. Milestones consist of several folders each pertaining to a specific milestone. All relevant data to that milestone is stored in their respective folder. This made it easy to revisit previous milestones and differentiate between each iteration of the project.
- b. Application:
 - i. Application contained the actual website. The entire application can be run with just the contents within that folder. This directory is what is deployed to the server.
- c. Credentials:
 - i. Credentials provides information regarding access to the server and the database loaded on the server. All information regarding access to the server/database can be found in the readme.md file within this folder. It also contains a key which allows for SSH connection to the server.

The screenshot shows a GitHub repository page for 'CSC-648-SFSU / csc648-03-fa22-team02'. The repository has 9 branches and 0 tags. The main branch is selected. The commit history is as follows:

Commit	Author	Date	Message
main	nandap1	4 days ago	M4 documentation by team02
Milestones		4 days ago	M4 documentation by team02
application		4 days ago	Merge branch 'main' of https://github.com/CSC-648-SFSU/csc648-03-fa...
credentials		11 days ago	Merge branch 'main' into Richard
.gitignore		11 days ago	Merge branch 'main' into Richard
LICENSE		3 months ago	Revert "test merge"
README.md		3 months ago	Updated readme (1)

Google Analytics Stats Plot



Project Management

SERVER HOST	Google Compute Engine 1vCPU 2 GB RAM
OPERATING SYSTEM	Ubuntu 22.0.4
SERVER DATABASE	MySQL v8.0
WEB SERVER	Apache 2.4.54
SERVER-SIDE LANGUAGE	Javascript v16.17.0
ADDITIONAL TECHNOLOGIES	VScode

Self Assessments and Contributions

NS

✉ Nanda Sivagnanam Pandian <npandian@mail.sfsu.edu>

To: ✉ Troy Carloni; ✉ Farid Mehdipour; ✉ Hugo Moreno; ✉ Richard William Aguilar; ✉ Yuwei Liu

Today at 1:31 43PM

- **His/her contributions to team project and teamwork (technical and any other) in no more than half a page – list item format is OK.**
 - Putting together the final submission for each milestone
 - Gathering everyone for team meetings and ensuring the work gets done
 - Coded for messaging function for the application
 - Made sure to check in with every team member. If help was needed, made sure to get the appropriate team member online to resolve the issue.
- **Number of submissions he/she made to github team Dev. Branch (explain if this number is very low)**
 - 17 commits
 - The number is low because I was primarily double checking all the requirements were being met for each milestone. If a requirement was not met, then I would go inquire and find out the reasons. Overall managing the team and reviewing code.
- **One brief paragraph on main challenges he/she encountered in team project**
 - I think one of the main challenges that I encountered in the team project was getting everyone together to meet soft deadlines. It is understandable that everyone has other class projects yet gathering everyone to meet their task soft deadlines was beginning at first. Around milestone 3, I was starting to get a hold of going after my teammates if anything was pending or needed work. Other than that, some of us didn't have certain skills or knowledge in an area, so it added on more work for another team member due to lack of time. This connects to team communication. Even though there was emphasis in addressing any issues early on, they were brought up closer to our soft deadline. However, the team did make sure to get online and get the work done as insisted.
- **One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes.**
 - Next time, I would like to enforce deadlines among the team, rather than let it pass at times. I believe I was lenient within the team, which is why we had last-minute meetings for one of our milestone deadlines. Although the work was complete, it wore out my teammates, so I would like to plan better next time. I would like to also inquire about everyone's knowledge in the skills needed for the application, so that we can be aware ahead of time. It was difficult to assign some tasks to certain teammates due to the lack of expertise in an area. Again, it wasn't a negative factor, but being aware of that as a team lead could've helped me assign tasks better. I hope to learn better in organizing and enforcing deadlines. Besides all this, my team was exceptional, and everyone always made things happen!

Best regards,
Nanda (Team Lead)



✉ Troy Carloni <tcarloni@mail.sfsu.edu>

To: ✉ Nanda Sivagnanam Pandian; ✉ Farid Mehdiipour; ✉ Hugo Moreno; ✉ Richard William Aguilar; +1 more ↗

Today at 3:19 47PM

Hello Team,

a) Contributions to team project and teamwork

1. Bycrypt/password hash implementation
2. Passport
3. Database implementation
4. Download files (file.controller.js)
5. Api error handling
6. Usermanager.js
7. Login/logout (backend, frontend js)
8. Registration (backend, frontend js)
9. Most if not all backend routes
10. Search
11. JOI implementation
12. I pretty much worked/touched majority of code that involves the backend or any frontend js. I did not work on UI of site.

b) Number of submissions

1. 117

c) brief paragraph on main challenges

1. Getting started was the tough part. Having only done a similar endeavor in 317 I had to learn a lot starting out. Once I learned the basics of how the frontend and backend communicated and learned how to use postman to test the backend I was able to really hit the ground running and get a lot done. There are a lot of useful YouTube tutorials of how to set up aspects of a website and those helped a lot.

d) Personal Improvement

1. While the way we went about GitHub worked, I feel like there could have been more improvement in the organization of it. As the github master I did learn more about how to navigate/be effective with github but I still feel like I have a lot to go before I can actually call myself a GitHub Master. One other thing I could do better is letting other people take the reigns more, I found once I knew how to implement something I just found it easier to go in and do it myself vs letting my teammates tackle it.



✉ Hugo Moreno <hmoreno1@mail.sfsu.edu>

To: ✉ Nanda Sivagnanam Pandian; ✉ Troy Carloni; ✉ Farid Mehdipour; ✉ Yuwei Liu; ✉ Richard William Aguilar

Today at 3:42 14PM

a. Contributions

- Made profile page with username and user's posts
- Organized layout of cards on the home screen
- Connected backend to front end for user profile page (with Troy's help)
- Created handlebars base structure of login and registration pages
- Contributed to functions to retrieve messages

b. Total number of submissions:

- 22 Commits to Branches with 3,775 insertions and 385 deletions
- Few commits because I would commit bigger chunks of code

c. Main Challenges

I think the biggest challenge for me was learning a new language and working together through github. Before this class the only experience I had with Java Script was the 317 WebDev Class and as we did create a fully functioning website in that class, I feel like the content was very specific towards the website we were creating in that class. Although this website is similar, we went about creating the website in a different way, so I had to learn A lot more of how java script and node js worked. Working together through github was also difficult because in past group projects I mostly just sent code through discord and pieced it together. Besides a small experience during the summer in a group project, I had never used github in a big group setting. Trying to merge everybody's work together to get a new updated version was probably the most difficult part but once I got the hang of it, the use of github wasn't too bad.

d. What to do better next time

Next time I think I could communicate better with my group. There were some instances where we wouldn't be too coordinated or clear on what we needed to get done for the current task and some things would fall through the cracks. I would also try and take more time to learn more about the language we are using to program. I hit a lot of roadblocks during the programming process that slowed down the development of the product.



✉ Farid MehdiPour <fmehdipour@mail.sfsu.edu>

To: ✉ Nanda Sivagnanam Pandian; ✉ Troy Carloni; ✉ Hugo Moreno; ✉ Richard William Aguilar; ✉ Yuwei Liu

Today at 5:13 34PM

Hello Team,

a) Contributions

- Lay out the header body and footer
- Created the Navigation bar
- Created the logo using canva.com
- Created the banner using canva.com
- Created the login page
- Created the registration page
- Created the Product Info and Product user
- Contact The Seller
- Created the search bar
- All the CSS of the pages included above
- Contributed to functions of the product info page (Passing the right data about the product)

b) Number of submissions

- 57

c) Main Challenges

The page's rendering on various platforms with different screen sizes was my biggest challenge. This problem resulted from my failure to read the structure in the beginning; I was unaware that the website needed to be flexible with size, and that created a significant problem because I did not incorporate that throughout the CSS part of my coding, leading to a large number of CSS codes that now need to be edited. This also created another problem because changing the CSS of one element on the page would affect all the others; however, I am still working on that in order to finish it in time for the Demo day.

d) Personal Improvement

- Be flexible and adaptable, as plans may change during the course of the project.
- I can definitely get better at collaborating and communicating so that I can share the task with my team and keep them informed of my progress.



Yuwei Liu <yliu43@mail.sfsu.edu>

Today at 7:39 38AM

To: Nanda Sivagnanam Pandian; Troy Carloni; Hugo Moreno; Richard William Aguilar; Farid Mehdiour

- **His/her contributions to team project and teamwork (technical and any other) in no more than half a page – list item format is OK.**

Participate in design ideas, and document writing.

Participate in functional requirements.

Developed the user's favorite items page.

- **Number of submissions he/she made to github team Dev. Branch (explain if this number is very low)**

- 15 commits
- The number is low because I am not familiar with the technology, and I was always afraid to merge people's work, and I was afraid of the mess project up. I was glad to develop the favorite page even though it was dropped due to the feedback, but I believe I have studied many things and it has helped me.

- **One brief paragraph on main challenges he/she encountered in team project**

The main challenges I encountered I think were a lot, and the communication was one, the other is the technology, and I was trying to study new things and try to help more.

- **One brief paragraph on what he/she do better next time based on what was learned in the class about SE management and processes.**

Next time I think I will use this term's experience to trying to participate more no matter in technology aspect or team communication, I also will ask people more to assign tasks to me and not just wait for assignments.



✉ Richard William Aguilar <raguilar10@sfsu.edu>

To: ✉ Nanda Sivagnanam Pandian; ✉ Troy Carloni; ○ Farid Mehdipour; ✉ Hugo Moreno; ○ Yuwei Liu

Yesterday at 3:43 46PM

Hello Team,

a. Contributions

- Helped with back end code
- Did the initial set up for the server on Google Console
- Allowed the website to be deployed on port 8080
- Installed tools onto the server on demand
- Created SSH profile
- Fixed any back end or front end issues to have them work on the server correctly
- Helped with documentation
- Checked each day to make sure server works and doesn't overcharge

b. Total number of submissions:

- 9 Commits to Branches
- Few commits because I would commit bigger chunks of code

c. Main Challenges

Main challenges for me was with the server management because the server would have constant trouble allowing the user to SSH in and would kick the user out in random times. The other problem is workbench constantly kept crashing on me which hindered some of my work. Personally the challenging part is the feeling of working with a team instead of doing individual work because there has to be communication. As a student I am used to me working on my assignment by myself, but with this project it has to be constant updates everyday on where we are.

d. What to do better next time

In the future I feel like I want to definitely improve on my availability and being able to balance my loads. With this class and others I had to balance a lot of work that could have compromised the team to not being able to finish the project. The second thing I believe I could do better is understanding that I can reach out for help on a problem instead of working on it and getting nothing done