

# SW Engineering CSC648/848 Fall 2022

## SFSU MEDIA STORE

### TEAM 3

*Himani Varshney (Team Lead)* [hvarshney1@sfsu.edu](mailto:hvarshney1@sfsu.edu)

*Donnovan Jiles (Back-end Lead)*

*Olimpia Aguillon (Front-end Lead)*

*Josef Fiedler (Github Master)*

*Yasaman Pakdel*

*Robert Peter Swanson*

### Milestone 5

12/09/2022

### Revisions History

Date Submitted	Date Revised
12/17/2022	

## **1) PRODUCT SUMMARY:**

Name of the product – SFSU Media Store

When we joined this college, we needed to get access to the previous year recordings of the courses we had enrolled for. We went and checked on iLearn which is a go to platform for any course related contents. Unfortunately, it contains the content of the present semester only. We then had to ask my peers in college to share any recordings they had. This is when we decided to develop and launch a web platform “**SFSU Media Store**” for sharing media which is used only by SFSU students and faculty. SFSU Media Store will let SFSU students and faculty browse through numerous digital media present on the web platform. Users can directly download the content if it is available for free. In case the content, is not available for free, the user can contact the seller by sending them a message via the application itself Any student or faculty can upload the content and on submission they need to register on the application. To make sure no unsafe content goes live to the audience, every uploaded item will need the post approval for publishing from the admin.

### **Functionalities:**

1. Unregistered users will be able to browse items.
2. Unregistered users will be able to search items using media categories.
3. Unregistered users will be able to view item details.
4. Unregistered users will be able to search media using the item's title, description.
5. Unregistered users will be able to register to the website.
6. Registered Users will be able to download the free item.
7. Registered Users will be able to login into their personal account.
8. Registered Users will be able to contact sellers to buy paid media item.
9. Registered Users will be able to post an item.
10. Registered Users will be able to view the dashboard for their posts.
11. Registered Users will be able to view the dashboard for their messages.
12. Admin will approve the appropriate posts (before they go live).
13. Admin will reject the posts (from going live) which are inappropriate.

URL: <http://35.88.249.97:3000/>

### **3) Milestone documents – M1-M4**

#### **Milestone 1**

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

### **SFSU MEDIA STORE**

#### **TEAM 3**

***Himani Varshney (Team Lead) [hvarshney1@sfsu.edu](mailto:hvarshney1@sfsu.edu)***

***Donnovan Jiles (Back-end Lead)***

***Olimpia Aguillon (Front-end Lead)***

***Josef Fiedler (Github Master)***

***Yasaman Pakdel***

***Robert Peter Swanson***

10/08/2022

#### **Revisions History**

Date Submitted	Date Revised
10/08/2022	10/23/2022

## **1) EXECUTIVE SUMMARY:**

When we joined this college, we needed to get access to the previous year recordings of the courses we had enrolled for. We went and checked on iLearn which is a go to platform for any course related contents. But unfortunately, it contains the content of the present semester only. We then had to ask my peers in college to share any recordings they had. In between all this painful process, we felt there was the need for a platform where any student could get access to any recordings for the courses they were looking for, as well as any other digital media useful for studying, without troubling others. This is when we decided to develop and launch a web platform “SFSU Media Store” for sharing media across SFSU students and faculty. Not only this, users can buy and sell any kind of digital media such as images, audios, videos, etc. on our web platform. It will be like a One Stop Store for all SFSU students and faculty to gain access to digital media. Our application will make searching for tutorials and class recordings/ presentations a lot easier since it will all be in one place. It will make the hassle of trying to find different information easier.

SFSU Media Store will let students and faculty browse through numerous digital media present on the web platform. Users can directly download the content if it is available for free. In case the content, is not available for free, the user can contact the seller by sending them a message via the application itself to discuss moving forward with downloading the contents. Any student or faculty can upload the content and on submission they need to register on the application. To make sure no unsafe content goes live to the audience, every uploaded item will need the post approval for publishing from the admin.

To make browsing simpler, users will be provided with a search bar wherein they can search by title, course number, class, etc.. Also the contents on the web application are listed in different categories, which will help users navigate to the content easily. Users can also rate the content which will help other users in making wise decisions about buying that content in the future. Also, there will be a like and comment button on each content on the application. This will give users ideas on what topics are getting popular at SFSU.

We are a group of 6 SFSU students who started this Startup as a part of one of our course projects. Two of us are graduate students whereas the others are in their senior year of undergrad. Our expertise lies in diverse skills ranging from backend technologies like Python and Flask to front end technologies like Bootstrap and ReactJS. We have divided our group into three frontend engineers and three backend engineers. This way we have an equal work distribution within our group. Not only that but some have a lot of experience in one area but are curious about the other so this project gives us the opportunity to learn and explore different technologies. We also get to strengthen the skills we already have and create something unique.

## 2. PERSONAE AND MAIN USE CASES:

### A. Jane - an SFSU student



Photo source : [Unsplash](#)

About Jane	Goals and senarios
<ul style="list-style-type: none"><li>• Computer Science student at SFSU</li><li>• Has a busy class schedule</li><li>• Sufficiently familiar with web applications, and sometimes uses various applications to buy the products she needs online, or download books or videos for her classes.</li><li>• She is not too patient and often needs access to her purchased product (class material) immediately.</li></ul>	<ul style="list-style-type: none"><li>• She has an assignment deadline by tonight.</li><li>• She is looking for the class recordings she missed.</li></ul>

**B. John - a faculty at SFSU**



Photo source : [Unsplash](#)

About John	Goals and scenarios
<ul style="list-style-type: none"><li>● He teaches multiple classes and has a busy schedule</li><li>● He is not very familiar with web applications and only has some basic skills</li><li>● He is not very patient with learning how to navigate through a website</li></ul>	<ul style="list-style-type: none"><li>● He wants to make his class recordings to be available to all students of SFSU.</li><li>● He also has some eBooks which he wants to sell to SFSU students.</li></ul>

**C. Jim - a student at SFSU**



Photo source : [Unsplash](#)

About Jim	Goals and senarios
<ul style="list-style-type: none"><li>● He is a senior year student</li><li>● He is very busy preparing for full time job interviews</li><li>● He has basic web app skills</li><li>● He is not very patient.</li></ul>	<ul style="list-style-type: none"><li>● He is looking for free resources for interview preparation.</li></ul>

#### D. Joe - admin (moderator)



Photo source : [Unsplash](#)

About Joe	Goals and senarios
<ul style="list-style-type: none"><li>• This is his job and he allocates a certain amount of time just for his admin tasks</li><li>• He has good knowledge of web applications</li><li>• He has good knowledge of database.</li></ul>	<ul style="list-style-type: none"><li>• He is looking for a database moderator role in a startup/company.</li></ul>

#### Use cases:

1. Instructor posts a media on the website

John is an **instructor** at SFSU. He uses our web app to **post** the class material (lecture videos, books, pdfs, etc.) for his students. These class materials are to be used only by SFSU students.



He uses the **upload** function to select and upload the materials. He will choose which class and subject the material should be categorized as, and he will enter the price for each material. On submission he will then be asked to register or log in (if he has previously made an account on our web app) using his SFSU email or ID. The upload will be then pending approval from the website **admin** (moderator). John will see the message that his post is pending for approval.

## 2. Student contacts the seller to buy paid item

Jane is a **student** at SFSU. She uses our web app to find and buy the media she needs. She browses the website and looks at the different media. She uses the **search** function to search for a certain SFSU class or a specific SFSU product. She can choose the category of the product when searching. Once she finds the product and clicks purchase she will be asked to **register** or **login** (if she has previously made an account on our web app) using her SFSU email or ID. She will then be able to **send a message** to the seller with her contact details and will wait for seller to contact her via the contact info she provided in the message.

## 3. Student posts the item

Jane is a **student** at SFSU, she is part of a student organization. She creates resources and media for this student organization. She uses our web app to share these media with other SFSU students, faculty or staff. She uses the **post** function to select and upload the media. She will choose the category and subject of the media, and she will enter the price for it. After clicking upload she will then be asked to register or login (if she has previously made an account on our web app) using her SFSU email or ID. The upload will be then pending approval from the website admin (moderator). Jane will see the message that her post is pending for approval.

## 4. Admin approves / denies the upload request

Joe is the **admin** for this web app. He will use **Workbench** to monitor the activity of the website. He approves or denies all the **upload requests** before they are posted on the website. He will **suspend** a user's account if they try to upload inappropriate media more than once or if they don't follow the website guidelines.

## 5. Student downloads a free item

Jim is a senior year **student** at SFSU. He is preparing for interviews. He **browses** our web app to **search** for free videos available on the topic he is interested in. He attempts to download the content but is prompted to register/ login. He then **downloads** the content he is interested in.

### **3) LIST OF MAIN DATA ITEMS AND ENTITIES**

- Item
  - (title, category, price, description, file)
- Rating
- Comments
- Post\_category
  - i.e. audio, video, image
- Messages
- Downloads
- User registration record
- Users
  - see types of users below

#### **Type of users**

- Unregistered User
  - a customer who visits the webpage but hasn't registered
  - Viewing permissions
  - Not able to post nor send message
- Registered User
  - a customer who has registered for the web application and can post / upload a media
  - Elevated permissions than unregistered user. Eg: Able to post and send message

- Admin

- An administrator with different set of permissions than an usual user
- Has obligation to approve / reject posts before they go live.

## **4) INITIAL LIST OF FUNCTIONAL REQUIREMENTS**

### Unregistered users:

9. Shall be able to browse items.
10. Shall be able to search items using media categories.
  1. Can search media using a category menu that will present media based on the type of content the media shows.
11. Shall be able to view item details.
12. Shall be able to search media using the author's name.
  1. The first and/or last name of an author can be searched.
13. Shall be able to search media using the item's title, class number, etc..
14. Shall be able to view comments on posts.
15. Shall be able to view likes on posts.
16. Shall be able to view ratings on posts.
17. Shall be able to preview media.
18. Shall be able to register to the website.

### Registered Users

All of the above functionalities of unregistered users plus the following:

19. Shall be able to download the free item.

- 20. Shall be able to login into their personal account.
- 21. Shall be able to contact sellers to buy paid media item.
- 13. Shall be able to post an item.
- 14. Shall be able to delete item.
- 15. Shall be able to edit item.
  - 1. Users will be able to change the title of published media.
- 16. Shall be able to delete their account.
- 17. Shall be able to post ratings on item.
- 18. Shall be able to change their personal account settings.
  - 1. This includes changing: password, user details.
- 19. Shall be able to comment on other user's posts.
- 20. Shall be able to like other user's posts.

Admin:

- 22. Admin is required to approve the posts (before they go live) which are appropriate.
- 23. Admin shall be able to reject the posts (from going live) which are inappropriate.
- 24. Admin shall have access to all registered users' data.
- 25. Admin shall have the permission to suspend registered users accounts.
- 26. Admin shall be able to view users' data (Data including media downloaded, media rating given, etc.)

**5) 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, 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).

## 6) COMPETITIVE ANALYSIS

	Facebook Market Place	Amazon	Ebay	Etsy	Our Future Product
Text Search	+	++	+	++	+
Boolean Search	+	+	+	+	+

Browse	+	++	+	++	+
Download	-	+	-	+	++
Digital Media Variety	-	+	-	+	++
Messaging	++	-	+	++	+

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

Looking at our competitors, our planned advantages is that our product will be geared towards SFSU students, by giving an option to search using class number as well. It will be fully digital, i.e. we will be supporting all kinds of digital media ranging from audio, videos, images, etc.. Users will be able to buy, sell, and share digital media. Users will also have the ability to download any free material and message sellers for buying the paid content. Looking at our competitors such as Facebook market place and Ebay, those are features they lack in. Some companies such as Amazon and Etsy have those features but not at the level that our future product will be at. That being said, we do want to get inspiration from their browsing abilities and their search abilities.

## **7) HIGH LEVEL SYSTEM ARCHITECTURE AND TECHNOLOGIES USED:**

Sever Host: Amazon AWS

Operating System: Ubuntu 16.04 Server

Database: MySQL

Web Server: NGINX 1.12.2

Server-Side Language: Python 3

Additional Technologies: Web Framework: Flask , Bootstrap

IDE: PyCharm and VSCode,

Web Analytics: Google Analytics

## **8) TEAM & ROLES**

Member Name	Role
Himani Varshney	Team Lead, Document Master, Back-end developer
Donnovan Jiles	Back-end Lead
Olimpia Aguillon	Front-end Lead
Josef Fiedler	Github Master, Back-end developer
Yasaman Pakdel	Front-end developer
Robert Peter Swanson	Front-end developer

## 9) CHECKLIST

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

## Milestone 2

# SW Engineering CSC648/848 Fall 2022

## SFSU MEDIA STORE

### TEAM 3

*Himani Varshney (Team Lead)* [hvarshney1@sfsu.edu](mailto:hvarshney1@sfsu.edu)

*Donnovan Jiles (Back-end Lead)*

*Olimpia Aguillon (Front-end Lead)*

*Josef Fiedler (Github Master)*

*Yasaman Pakdel*

*Robert Peter Swanson*

## Milestone 2

10/29/2022

### Revisions History

Date Submitted	Date Revised
10/29/2022	11/10/2022



## **1) EXECUTIVE SUMMARY:**

When we joined this college, we needed to get access to the previous year recordings of the courses we had enrolled for. We went and checked on iLearn which is a go to platform for any course related contents. But unfortunately, it contains the content of the present semester only. We then had to ask my peers in college to share any recordings they had. In between all this painful process, we felt there was the need for a platform where any student could get access to any recordings for the courses they were looking for, as well as any other digital media useful for studying, without troubling others. This is when we decided to develop and launch a web platform “SFSU Media Store” for sharing media which is used only by SFSU students and faculty. Users can buy and sell any kind of digital media such as images, audios, videos, etc. on our web platform. It will be like a one stop store for all SFSU students and faculty to gain access to digital media. Our application will make searching for tutorials and class recordings/ presentations a lot easier since it will all be in one place. It will make the hassle of trying to find different information easier.

SFSU Media Store will let SFSU students and faculty browse through numerous digital media present on the web platform. Users can directly download the content if it is available for free. In case the content, is not available for free, the user can contact the seller by sending them a message via the application itself. To discuss moving forward with downloading the contents. Any student or faculty can upload the content and on submission they need to register on the application. To make sure no unsafe content goes live to the audience, every uploaded item will need the post approval for publishing from the admin.

To make browsing simpler, users will be provided with a search bar wherein they can search by title, course number, class, etc.. Also the contents on the web application are listed in different categories, which will help users navigate to the content easily. Users can also rate the content which will help other users in making wise decisions about buying that content in the future. Also, there will be a like and comment button on each content on the application. This will give users ideas on what topics are getting popular at SFSU.

We are a group of 6 SFSU students who started this Startup as a part of one of our course projects. Two of us are graduate students whereas the others are in their senior year of undergrad. Our expertise lies in diverse skills ranging from backend technologies like Python and Flask to front end technologies like Bootstrap and ReactJS. We have divided our group into three frontend engineers and three backend engineers. This way we have an equal work distribution within our group. Not only that but some have a lot of experience in one area but are curious about the other so this project gives us the opportunity to learn and explore different technologies. We also get to strengthen the skills we already have and create something unique.

## **2. LIST OF MAIN DATA ITEMS AND ENTITIES (EXPAND AS NECESSARY):**

- 1) Item (Any image or a video, etc)
  - Item\_id
  - Item\_title
  - Item\_blob
  - Item\_size
  - Item\_category (I.e. Audio, Video, Class, etc. Item category will be derived from this information)
  - Item\_created\_time
  - Item\_description
  - Item\_edited
  - Item\_rating
  - Item\_likes\_count
  - Item\_comments\_count
- 2) Item\_likes
  - Item\_id
  - User\_id
- 3) Item\_comments
  - Item\_id
  - User\_id
  - Comment\_text
- 4) Message
  - Message\_id
  - Message\_created\_date
  - Message\_sender\_user\_id
  - Message\_recipient\_user\_id
  - Message\_text
- 5) Downloads
  - Download\_id
  - Item\_id
  - User\_id
- 6) Category
  - Item\_category
- 7) User\_records
  - User\_id
  - User\_type (Registered User, Admin)
  - User\_username

- User\_first\_name
- User\_last\_name
- User\_email
- User\_password

8) Rating

- Item\_id
- User\_id
- Rating

### **3) FUNCTIONAL REQUIREMENTS - PRIORITIZED**

#### **- Priority 1**

##### Unregistered users:

22. Shall be able to browse items.

23. Shall be able to search items using media categories.

1. Can search media using a category menu that will present media based on the type of content the media shows.

24. Shall be able to view item details.

25. Shall be able to search media using the author's name.

1. The first and/or last name of an author can be searched.

26. Shall be able to search media using the item's title, description, category.

27. Shall be able to register to the website.

28. Shall be able to sort items by price.

##### Registered Users

All of the above functionalities of unregistered users plus the following:

29. Shall be able to download the free item.

30. Shall be able to login into their personal account.

- 31. Shall be able to contact sellers to buy paid media item.
- 10. Shall be able to post an item.
- 11. Shall be able to view the dashboard for their posts.
- 12. Shall be able to view the dashboard for their messages.

#### Admin

- 13. Admin is required to approve the appropriate posts (before they go live).
- 14. Admin shall be able to reject the posts (from going live) which are inappropriate.

#### **- Priority 2**

##### Unregistered users:

- 15. Shall be able to view comments on posts.
- 16. Shall be able to view likes on posts.
- 17. Shall be able to view ratings on posts.

##### Registered Users

All of the above functionalities of unregistered users plus the following:

- 18. Shall be able to post ratings on item.
- 19. Shall be able to comment on other user's posts.
- 20. Shall be able to like other user's posts.

#### Admin :

#### **- Priority 3**

##### Unregistered Users

21.Shall be able to view thumbnail.

#### Registered Users

All of the above functionalities of unregistered users plus the following:

22.Shall be able to delete item.

23.Shall be able to edit item.

1. Users will be able to change the title of published media.

24.Shall be able to delete their account.

25.Shall be able to change their personal account settings.

a. This includes changing: password, user details.

#### Admin:

26. Admin shall have the permission to suspend registered users accounts.

### **4) UI STORYBOARDS FOR EACH MAIN USE CASE :**

1) User Browses the items on home page

Logo


About Us

Post

Sign in/Register

Welcome to our website. You can find useful media for your SFSU classes on our website

Category



Price

Contact seller

Price

Contact seller

Price

Contact seller

Price









Contact seller

Contact Us

Policy

Logo

2) User Search results using item title and then all items with that search string will come

Logo	About Us	<input type="text" value="✓   🔍 Product Name"/>	Post	Sign in/Register
# results found		Sort By		
				
<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	
				
<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	<div>Price</div> <div>Contact seller</div>	
Contact Us	Policy	Logo		

3) Unregistered user shall register

Sign Up

First name

Last name

SFSU Id

Email

Password

☐ Accept terms and conditions

Already have an account? [Sign in](#)

Sign Up

:



4) User shall sign in :

### Sign In

Email Address

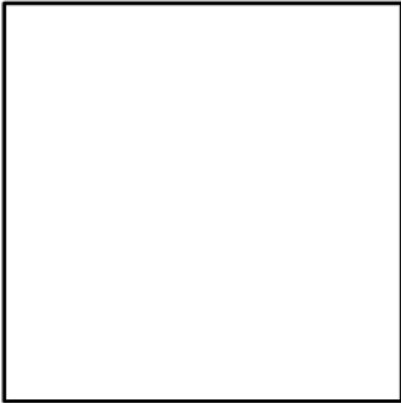
Password

[Forgot Password?](#)

Do not have an account? [Sign up](#)

Submit

5) User shall view item details :

Logo	About Us	<input type="text" value="✓   🔍"/>	Post	<div>User Initials</div>
		<h1>Name of Product</h1> <p>Category</p> <p>Price</p> <p>~~~~~</p>		
<p>Want to buy " ~~~~ " ?</p>		<div>Contact Seller</div>		
Contact Us	Policy	Logo		

6) User shall Contact Seller to buy product after signing in :  
User searches for the item

Logo

About Us

☐ |

Post

Sign in/Register

Sort By



Contact Us

Policy

Logo

Logo

About Us

✓

|

○

Post

Sign in/Register

Name of Product

Category

Price

Want to buy "  "?

Contact Seller

Contact Us

Policy

Logo

click of Contact Seller he is asked to sign in

On

## Sign In

Email Address

Password

[Forgot Password?](#)

Do not have an account? [Sign up](#)

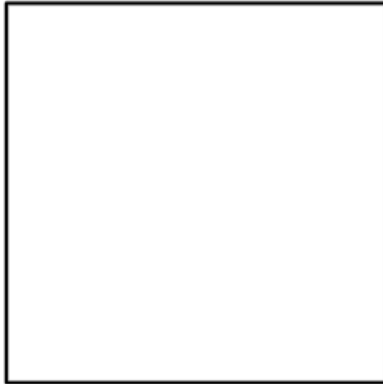
Submit

Logo

About Us

Post

User  
Initials




# Name of Product

Category

Price



Want to buy "  "?

Send Message to: Seller Name

Date

Title: Product Name

Message:

cancel

Send




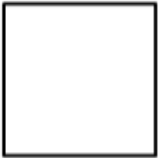












Contact Us

Policy

Logo

7) Download free product:

User search and finds the item

Logo	About Us	<input type="checkbox"/>   <input type="text" value="Product Name"/>	Post	Sign in/Register
				Sort By
 	 	 	 	
 	 	 	 	
Contact Us	Policy	Logo		

LogoAbout Us

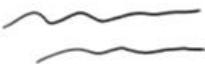
✓ | 🔍

PostSign in/Register

# Name of Product

Category

Free



Download

Contact UsPolicyLogo

On click of download he is asked to sign in:



## Sign In

Email Address

Password

[Forgot Password?](#)

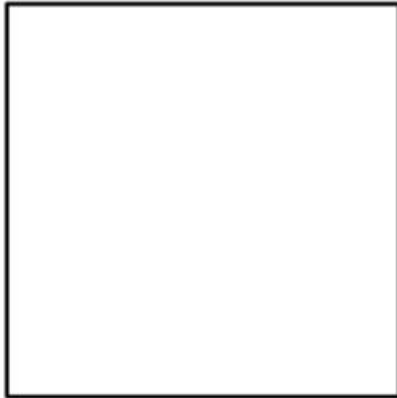
Do not have an account? [Sign up](#)

Submit

Logo

About Us

Post



# Name of Product

Category

Free



Download

Contact Us

Policy

Logo

8)Dashboard:

Logo	About Us	<input type="checkbox"/> <input type="text"/>	Post	User Initials
Hello Name !				
My Messages	First Name			
My Posts	Last Name			
Edit profile	<div></div> <div></div>			
Contact Us	Policy	Logo		

Dashboard my messages:

Logo

About Us

✓

|

🔍

Post

User

Initials

Hello Name !

My Messages	My Messages	Sort By
	Message 1	<div>Delete</div>
My Posts	Message 2	<div>Delete</div>
	Message 3	<div>Delete</div>
Edit profile		

Contact Us

Policy

Logo

Dashboard my posts:

Logo

About Us

Post



Hello Name !

My Messages	My Posts			Sort By
	Post Name	Status	Date	Delete
My Posts	Post Name	Status	Date	Delete
Edit profile				

Contact Us

Policy

Logo

9)Post an Item :  
User opens the home page

The wireframe depicts a web page layout. At the top, there is a navigation bar with links: "Logo", "About Us", "Post", and "Sign in/Register". Below this bar is a large rectangular search area. On the left side of this search area, the word "Category" is written above a checkmark icon. To the right of "Category" is a vertical line, followed by a magnifying glass icon, indicating a search function. The main body of the page contains four square boxes arranged horizontally, each with a wavy line underneath it, representing placeholders for item images. At the bottom of the page, there is a footer with links: "Contact Us", "Policy", and "Logo".

User can then post an item

[Logo](#)[About Us](#)

✓

|

🔍

[Sign in/Register](#)

Product Name:

Category:

Select category ✓

Price:

\$

Description:

Upload File

Post

[Contact Us](#)[Policy](#)[Logo](#)

On click of Post he is asked to Sign In

## Sign In

Email Address

Password

[Forgot Password?](#)

Do not have an account? [Sign up](#)

Submit



Logo

About Us

✓ | 🔍

Post

User Initials

Post an item

Product Name: \*

Category: \*

Select category ✓

Price: \*

\$

Description:

Upload File

Post

Contact Us

Policy

Logo

User is informed that post is pending for approval

Thank you your post is pending for approval!

[Continue Shopping](#)

For admin use cases we will be using workbench.

## **5) HIGH LEVEL ARCHITECTURE, DATABASE ORGANIZATION SUMMARY**

- *DB Organization* – Our MySQL db will have following tables (high level):
  - 9) Item (Any image or a video, etc)
    - Item\_id
    - Item\_title
    - Item\_blob
    - Item\_size
    - Item\_category (I.e. Audio, Video, Class, etc. Item category will be derived from this information) Foreign key to Category table
    - Item\_created\_time
    - Item\_description
    - Item\_edited
    - Item\_rating
    - Item\_likes\_count
    - Item\_comments\_count
    - Item\_created\_date
  - 10) Item\_likes
    - Item\_id
    - User\_id
  - 11) Item\_comments
    - Item\_id
    - User\_id
    - Comment\_text
  - 12) Message
    - Message\_id
    - Message\_created
    - Message\_sender\_user\_id
    - Message\_recipient\_user\_id
  - 13) Downloads
    - Download\_id
    - Item\_id
    - User\_id
  - 14) Category
    - Item\_category
  - 15) User\_records
    - User\_id
    - User\_type (Registered User, Admin)
    - User\_username
    - User\_first\_name
    - User\_last\_name

- User\_email
- User\_password

#### 16) Rating

- Item\_id
  - User\_id
  - Rating
- *Media Storage* – We will be storing our media as File paths in MYSQL database. Files will be stored at a safe place on the drive. We will be using standard file types for each like mp4 for videos, mp3 for audios, and so on.
  - *Search filter and architecture* – Item category will be used to filter search results. We will be using SQL precise search for categories from DB column called Item\_category and with MYSQL “%LIKE” search on text field like item\_description, item\_title, etc. In case our MYSQL query returns 0 search results we will be displaying a page of items with 6 popular items.

### **6) ACTUAL KEY RISKS FOR PROJECT AT THIS TIME:**

- **Skills risks:** The team is not very strong in front end development as no one from the team had earlier worked on any of the front-end frameworks. That being said, the team front end developers are learning front end frameworks like React and Bootstrap, as well as trying hands on React before the actual website development.
- **Schedule risks:** Given this covid times, there is a risk associated of being a team member getting infected which may impact his/ her deliverables. However, everyone is asked to let team lead know about this situation at the earliest so that lead can re-assign the tasks to other team members based on bandwidth.
- **Technical risks:** The team is not very familiar with the deployment part on AWS. However, team is learning how to deploy websites on AWS using tutorials.
- **Teamwork risks:** As of now we don't see any risk associated with team work. Team has found time other than the class meetings to collaborate and update on tasks.

### **7) PROJECT MANAGEMENT :**

We have been assigning tasks in our class team meetings and using discord and ZOOM meetings to update our progress on tasks up until now. For keeping record of who is doing which task, the team lead assigns the tasks to each team member based on their preferences and skills. Team keeps track of it using Google Spreadsheet. The team lead then shares a Google doc which is used by the team to work on their task. The work of each task member can be seen by everyone else using this Google doc. However, we will be soon moving to Trello (from next team meeting) to manage tasks as it will provide a more accurate and up to date account of the progress of each task and who is responsible for what. With Trello we can see who is responsible for what task, their progress on said task, and what tasks are completed or have yet to be started.

## **Milestone 3**

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

**Team number:** 3

**Meeting date:** 11/16/2022

- **Summary of feedback on UI (record all pages that need revision)**

Although overall Prof. was happy with the work we had done he gave us minor UI comments for revision of the following pages:

Home page – To add disclaimer at top and list top items sorted by date. Sign in and Register should be closer.

Search Results - Give button to contact seller on the Search Result page.

Contact Seller – Details of the item should be pre-filled.

Sign Up – Checkbox to agree with Terms of Service should be before Submit button.

Dashboard – Align the dashboard to center.

Post page – Right justify the field names in post form.

- **Summary of feedback on code and architecture**

Prof. reviewed the code and the folder structure. He helped us in pointing out that Header comments were missing in some files.

- **Summary of feedback on github usage**

Prof. reviewed our github repository and was satisfied with our commits history. He told us the importance of maintaining meaningful Commits messages.

- **Summary of feedback on DB**

Prof. reviewed our database on Mysql workbench and gave us the following feedback:

Some of the tables like User\_records were missing from the database. We told him that since we haven't done the backend integration of Sign up page, that table is missing. Having said that, we have updated our db and added these missing tables.

- **Summary of feedback on teamwork and risk management**

Prof. enquired about how the team work is going on and strongly told the team lead to report any kind of negligence to him. He made us realize that very less time is left for the final delivery and we need to gear up to complete it by the due date.

For risk management, he told us to keep the P1 requirements to the minimum so that we can complete the project on time.

- **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**

Developers are adhering to MVC pattern by segregating the Front end, Backend Code and Database.

• **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**

#### Unregistered users:

1. Shall be able to browse items.
2. Shall be able to search items using media categories.
3. Shall be able to view item details.
4. Shall be able to search media using the item's title, description.
5. Shall be able to register to the website.
6. Shall be able to sort items by price.

#### Registered Users

All of the above functionalities of unregistered users plus the following:

7. Shall be able to download the free item.
8. Shall be able to login into their personal account.
9. Shall be able to contact sellers to buy paid media item.
10. Shall be able to post an item.
11. Shall be able to view the dashboard for their posts.
12. Shall be able to view the dashboard for their messages.

#### Admin

13. Admin is required to approve the appropriate posts (before they go live).
14. Admin shall be able to reject the posts (from going live) which are inappropriate.

- Any other comments and issues

N/A

- Check Point (CP) if given, DUE:

N/A

#### Milestone 4

## SW Engineering CSC648/848 Fall 2022

### SFSU MEDIA STORE

#### TEAM 3

*Himani Varshney (Team Lead)* [hvarshney1@sfsu.edu](mailto:hvarshney1@sfsu.edu)

*Donnovan Jiles (Back-end Lead)*

*Olimpia Aguillon (Front-end Lead)*

*Josef Fiedler (Github Master)*

*Yasaman Pakdel*

*Robert Peter Swanson*

#### Milestone 4

12/09/2022

#### Revisions History

Date Submitted	Date Revised
----------------	--------------

12/09/2022	
------------	--

## 1) PRODUCT SUMMARY:

Name of the product – SFSU Media Store

When we joined this college, we needed to get access to the previous year recordings of the courses we had enrolled for. We went and checked on iLearn which is a go to platform for any course related contents. But unfortunately, it contains the content of the present semester only. We then had to ask my peers in college to share any recordings they had. This is when we decided to develop and launch a web platform “**SFSU Media Store**” for sharing media which is used only by SFSU students and faculty. Users can buy and sell any kind of digital media such as images, audios, videos, etc. on our web platform. It will be like a one stop store for all SFSU students and faculty to gain access to digital media. Our application will make searching for tutorials and class recordings/ presentations a lot easier since it will all be in one place. It will make the hassle of trying to find different information easier.

### Functionalities:

1. Unregistered users will be able to browse items.
2. Unregistered users will be able to search items using media categories.
3. Unregistered users will be able to view item details.
4. Unregistered users will be able to search media using the item's title, description.
5. Unregistered users will be able to register to the website.
6. Registered Users will be able to download the free item.
7. Registered Users will be able to login into their personal account.
8. Registered Users will be able to contact sellers to buy paid media item.
9. Registered Users will be able to post an item.
10. Registered Users will be able to view the dashboard for their posts.
11. Registered Users will be able to view the dashboard for their messages.
12. Admin will approve the appropriate posts (before they go live).
13. Admin will reject the posts (from going live) which are inappropriate.

URL: <http://35.88.249.97:3000/>

## 2) Usability test plan:

### **Test Objectives:**

To test the usability of the SEARCH functionality. It will help to measure the efficiency, effectiveness and satisfaction of the search usage for the users. Feedback of the same will help in designing a more user centred design for the product.

### **Test background and setup –**

The user needs to have a Windows/Mac PC. He / She can use Windows OS / Linux OS. The user needs to have a web browser like Google Chrome / Microsoft Edge. The URL of the system to be tested is <http://35.88.249.97:3000/Home>

The intended users of the system are the SFSU students and the faculty who can have zero to advanced level of computer knowledge. This usability testing is done to measure the effectiveness, efficiency and satisfaction of users using Likert test.

### **Usability Task description:**

The tester should access the URL in the web browser and go to the home page of the website. The user needs to test usability of the search bar by giving different input to the search bar.

### **Evaluation of Effectiveness:**

To measure the effectiveness of the search functionality, we will measure the percentage of people who were able to search for the desired items within 2 minutes. We will also record the count of errors per task by the users.

### **Evaluation of efficiency:**

To measure the efficiency of the search functionality, we will measure the average time users took to search for an item. We will also measure efficiency by the average time in which the image is loaded on click of search.

### **Evaluation of user satisfaction:**

To measure the user satisfaction, we will use the results of the Likert survey.

Likert Survey:

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
It is easy to change category in the search bar.					
I had no problem in finding the search bar.					



I can easily see the number of results returned on click of search					
--	--	--	--	--	--

### **3) QA test plan**

#### **Test objectives:**

To test the Search functionality of the SFSU Mediastore as per specs defined

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

Windows OS

Google Chrome Version 108.0.5359.98

Microsoft Edge Version 108.0.1462.46

URL : <http://35.88.249.97:3000/Home>

#### **Feature to be tested:**

Search Bar

#### **QA Test plan:**

Test #	Test Title	Test Description	Test Input	Expected Correct Output	Test Results (PASS / FAIL)
0001	Search_test_like	Test % like in search for the search field	Go to the specified URL. On the home page, click on search bar. Type "gator" in the search bar. Click on Search button.	2 items should be displayed having gator in their title	
0002	Search_test_null	Test the search field by giving blank input	Go to the specified URL. On the home page, click on search	8 items should be displayed.	

			bar. Do not type anything in the search bar and click on Search button.		
0003	Search_test_dropdown	Test for the category dropdown in the search bar	Go to the specified URL. On the home page, select 'Image' from the Search dropdown menu and enter 'gator' in the search bar. Click on Search button.	Only 1 item should be displayed having gator in their title	

QA test results on Google Chrome Version 108.0.5359.98

Test #	Test Title	Test Description	Test Input	Expected Correct Output	Test Results (PASS / FAIL)
0001	Search_test_like	Test % like in search for the search field	Go to the specified URL. On the home page, click on search bar. Type "gator" in the search bar. Click on Search button.	2 items should be displayed having gator in their title	PASS

0002	Search_test_null	Test the search field by giving blank input	Go to the specified URL. On the home page, click on search bar. Do not type anything in the search bar and click on Search button.	8 items should be displayed.	PASS
0003	Search_test_dropdown	Test for the category dropdown in the search bar	Go to the specified URL. On the home page, select 'Image' from the Search dropdown menu and enter 'gator' in the search bar. Click on Search button.	Only 1 item should be displayed having gator in their title	PASS

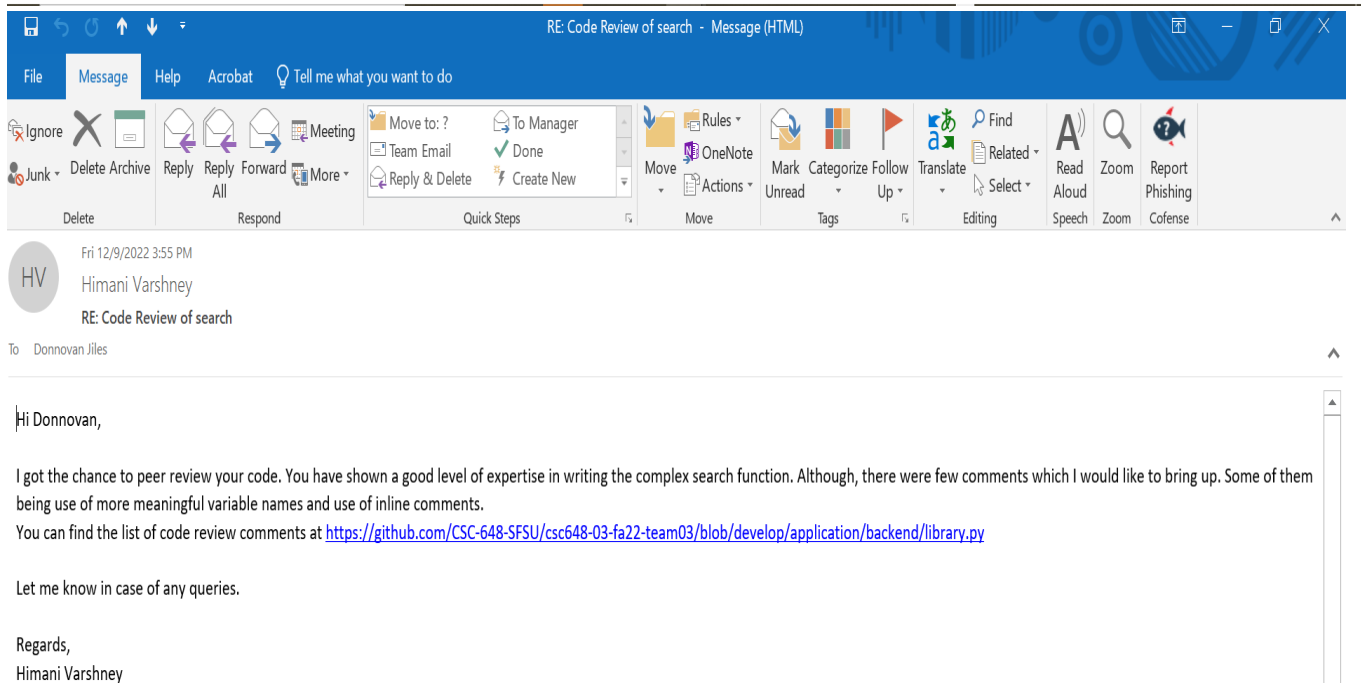
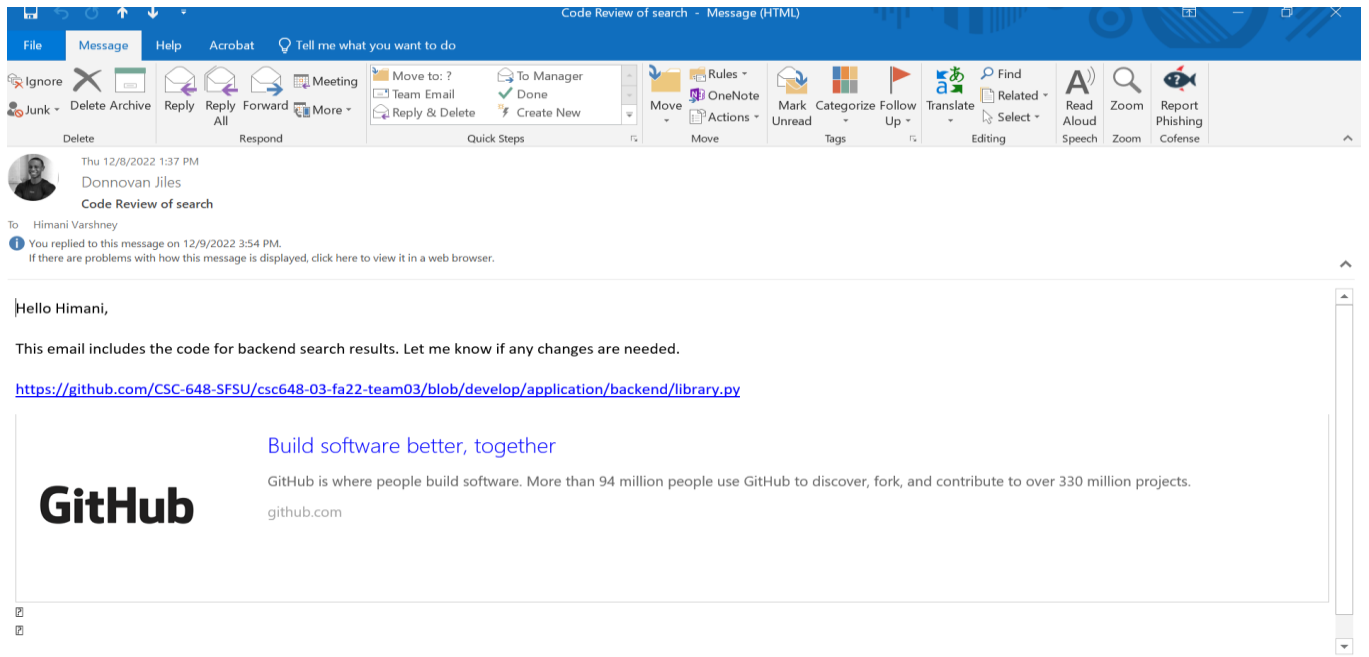
QA test results on Microsoft Edge Version 108.0.1462.46

Test #	Test Title	Test Description	Test Input	Expected Correct Output	Test Results (PASS / FAIL)
0001	Search_test_like	Test % like in search for the search field	Go to the specified URL. On the home page, click on search bar. Type	2 items should be displayed having gator in their title	PASS

			"gator" in the search bar. Click on Search button.		
0002	Search_test_null	Test the search field by giving blank input	Go to the specified URL. On the home page, click on search bar. Do not type anything in the search bar and click on Search button.	8 items should be displayed.	PASS
0003	Search_test_dropdown	Test for the category dropdown in the search bar	Go to the specified URL. On the home page, select 'Image' from the Search dropdown menu and enter 'gator' in the search bar. Click on Search button.	Only 1 item should be displayed having gator in their title	PASS

#### **4) Code Review:**

Back end code for Search -



Github Repo screenshot:

```

145 '''
146 Code reviewed by - Himani Varshney
147 Comments - 1) Remove print statements before deploying the code on server.
148           2) Use more meaningful variable names.
149           3) Use more inline comments for each if else condition.
150 '''
151 '''
152 #endpoint for search
153 @app.route('/search', methods=['GET', 'POST'])
154 def search():
155     print("Hello World")
156     if request.method == "POST":
157         print(request)
158         # request user input from <input> with name="book"
159         print(dir(request))
160         print("#####")
161         #request.json['book']
162         #print(request.body)
163         #book = request.form['book']
164         book = request.get_json()
165         print(request.get_json())
166         print(book['book'])
167         print("#####")
168         conn = mysql.connect()
169         cursor = conn.cursor()
170         if(book['book']=="" and book['Category'] == 'all' ):
171             cursor.execute("SELECT item_title,item_description, user_username, item_path, item_price, item_category FROM item JOIN user_records ON item_creator_id = u
172             conn.commit()
173             data = cursor.fetchall()
174             print(data)

```

## Front end code for Search :

Code Review Request - Message (HTML)

File Message Help Acrobat Tell me what you want to do

Ignore X Delete Archive Reply Reply Forward Meeting Move to: ? Team Email To Manager Done OneNote Mark Categorize Follow Translate Find Junk Delete Reply All Respond Reply & Delete Create New Move Actions Unread Tags Select

Tue 12/6/2022 1:45 PM

RP Robert Peter Swanson

Code Review Request

To Himani Varshney

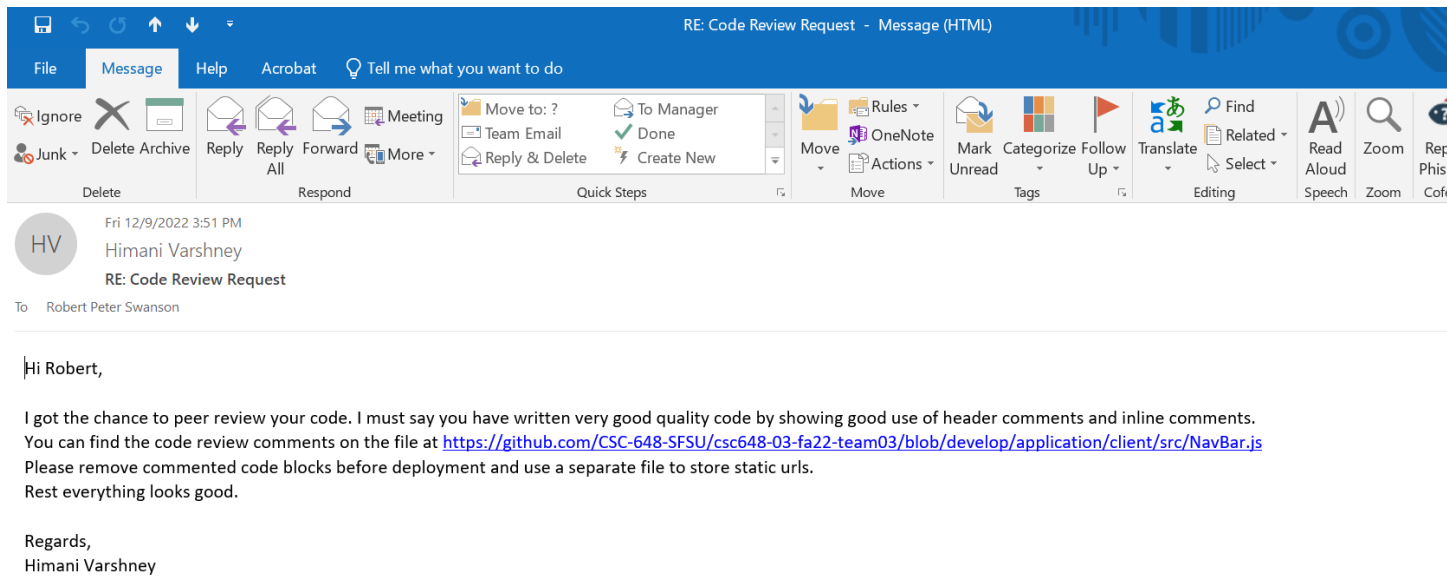
You replied to this message on 12/9/2022 3:50 PM.

Hello Himani,

Please review the code for search results and let me know if any changes are needed.

<https://github.com/CSC-648-SFSU/csc648-03-fa22-team03/blob/develop/application/client/src/NavBar.js>

Best,  
Robert Swanson



## Github Repo screenshot:

```
1  /*
2   * File: NavBar.js
3   * Author: Robert Swanson
4   * Description: NavBar for webapp with search functionality and links to About Us, Post,
5   *             Dashboard, Sign in, and register. And contains disclaimer that this is student
6   *             project
7   */
8
9  /*
10   Code reviewed by - Himani Varshney
11   Date - 12/09/2022
12   Comments - 1) Please remove commented block of code before moving to deployment server.
13              2) Good use of header comments and inline comments
14              3) All URL can be moved to one place and imported from there.
15   */
16   import { Link, useNavigate } from "react-router-dom"
17   import { useEffect, useState } from 'react';
18
19   const NavBar = () => {
20     const [category, setCategory] = useState("all");
21     const [searchText, setSearchText] = useState("");
22     const navigate = useNavigate();
23   }
24
25   function setCategories() {
26     //Categories return from DB goes here
27     const categories = { 'cat': ['Audio', 'Video', 'Image', 'Class'] }; //Hard coded for testing
28     /*
29     const options = []; //Array of <option> to be returned to dropdown
30   */
31 }
```

## **5) Self - check on best practices for security –**

<b>Asset to be protected</b>	<b>Types of possible / expected attacks</b>	<b>Strategy to protect / mitigate the asset</b>
Media items	Loss / Theft of media	All the media items are securely stored on the file disk of server, where in only admin has the access to the server and no one else
User Information	Theft of user's personal data, SQL Injection	DB is on the server with only admin having the access. All the confidential data like password is encrypted. Input Validation (like limiting input length to 40 alpha numeric characters) is done to prevent SQL injection. Only users with sfsu.edu can access the website.

Encrypt PW in the DB – DONE using md5 encryption algorithm

Input data validation –

search bar input for up to 40 alphanumeric characters – ON TRACK

registration e-mail to include “sfsu.edu” at the end – DONE

## **6) Self-check of the adherence to original Non-functional specs –**

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



## 4) Product Screen Shots:


Home page

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout

Welcome to Media Store, the one stop shop for all your digital media needs. Here you will find photos, videos, music, and anything else you could wish for to finish your SFSU projects on time and with style.

Sort By ▾



Gator

CSc648/848 Software Engineering Fall 2022

SW Engineering Teamwork (including global/distributed environment)

CSC648 Class1

Software Engineering CSC 648/848 Fall 2022

Software Architectures and Design Patterns

Multimedia and Search Architectures Design Patterns


CSC648 Class2

Contact Us Policy **MediaStore**

Item details page for free item:

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout



# Gator

Seller: padouch28

Category: Image

Price: \$ 0


Download

Contact Us Policy **MediaStore**

Item details page for paid item:

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout



# Halloween

Seller: testuser12

Category: Image

Price: \$ 25

Contact Seller

Contact Us Policy **MediaStore**

Contact Seller to buy an item :

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout

## Contact Seller

Send message to:

testuser12

Date:

mm/dd/yyyy

Title:

Halloween

Message:

Write message

Send

Contact Us Policy **MediaStore**

Post an Item:

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

MediaStore

About Us

All

Search...

Search

Post

DashBoard

Logout

Post an item

\* Required fields

Product Name:\*

Title

Category:\*

Category

Price:\*

0.00

Description:

Description...

Choose File

No file chosen

Post

Contact Us

Policy

MediaStore

## Dashboard

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

MediaStore

About Us

All

Search...

Search

Post

DashBoard

Logout

Welcome Hermione

My Profile

My Posts

My Messages

My Profile

First Name: Hermione

Last Name: V

Email: himani@sfsu.edu

Contact Us

Policy

MediaStore

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout

## Welcome Hermione

My Profile  
My Posts  
My Messages

### My Posts

1 results found

Post Name	Status	Date		
test image	Pending	Fri, 16 Dec 2022 00:00:00 GMT	Delete	View

Contact Us Policy **MediaStore**

SFSU Software Engineering Project CSC 648-848, Fall 2022. For Demonstration Only

**MediaStore** About Us All Search... Search Post DashBoard Logout

## Welcome Hermione

My Profile  
My Posts  
My Messages

### My Messages

Message	From	Date	Sort		
Hi, I like your post!	padouch28	Tue, 06 Dec 2022 00:00:00 GMT		Delete	View

Contact Us Policy **MediaStore**

## 5) Database Organization:

We used Mysql workbench to organise our data base. Four of our main tables are:

item table – where item\_category is foreign key to category table

item_id	item_title	item_description	item_creator_id	item_file	item_path	item_price	item_category	item_created_date	item_approved
1	Gator	Image of the Gator mascot	5	static/gator	gator.jpeg	0	Image	2022-12-07	1
2	Halloween	Image of halloween celebration	1	static/halloween	halloween.jpeg	25	Image	2022-12-07	1
3	CSC648 Class1	Lecture 1 PPT	1	static/ppt1	ppt1.jpeg	3	Class	2022-12-08	1
4	CSC648 Class2	Lecture 2 PPT	5	static/ppt2	ppt2.jpeg	3	Class	2022-12-08	1
5	Lunar Eclipse Video	Lunar Eclipse Video as observed from SF	1	static/video1	video1.jpeg	15	Video	2022-12-07	1
6	Winter at SF	Experience snowfall	6	static/video2	video2.jpeg	25	Video	2022-12-08	1
7	Gator anthem	Song sung by SFSU students	1	static/audio1	audio1.jpeg	5	Audio	2022-12-08	1
8	Design of Algorithms	CSC810 Textbook	6	static/course3	course3.jpeg	10	Class	2022-12-08	1
10	dbms h/w		6	2c.png	2c.png	0	Image	2022-12-13	0
11	hw3	solution	6	g.PNG	g.PNG	0	Image	2022-12-13	0
12	test post	test image uploaded	6	trelo.PNG	trelo.PNG	5	Image	2022-12-16	0

Category table

	category_id	item_category
▶	2	Audio
	4	Class
	1	Image
	3	Video
*	NULL	NULL

user\_records table

	user_id	user_type	user_username	user_first_name	user_last_name	user_email	user_password
	1	admin	testuser12	Test	User	test@sfsu.edu	5f4dcc3b5aa765d61d8327deb882cf99
	5	registered_user	padouch28	Pa	Douch	padouchpad@sfsu.edu	5f4dcc3b5aa765d61d8327deb882cf99
▶	6	registered_user	himani14	Himani	Varshney	hvarshney1@sfsu.edu	540ded8f44dd6764b6ea664ba68f1e
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

message table

	message_id	message_created_date	message_body_text	message_sender_id	message_receiver_id
▶	1	2022-12-06	Hi, I like your post!	5	6
	2	2022-12-08	Hi, have you received my message?	5	1
	3	2022-12-13	interested	6	5
*	NULL	NULL	NULL	NULL	NULL

## **6) Github organization:**

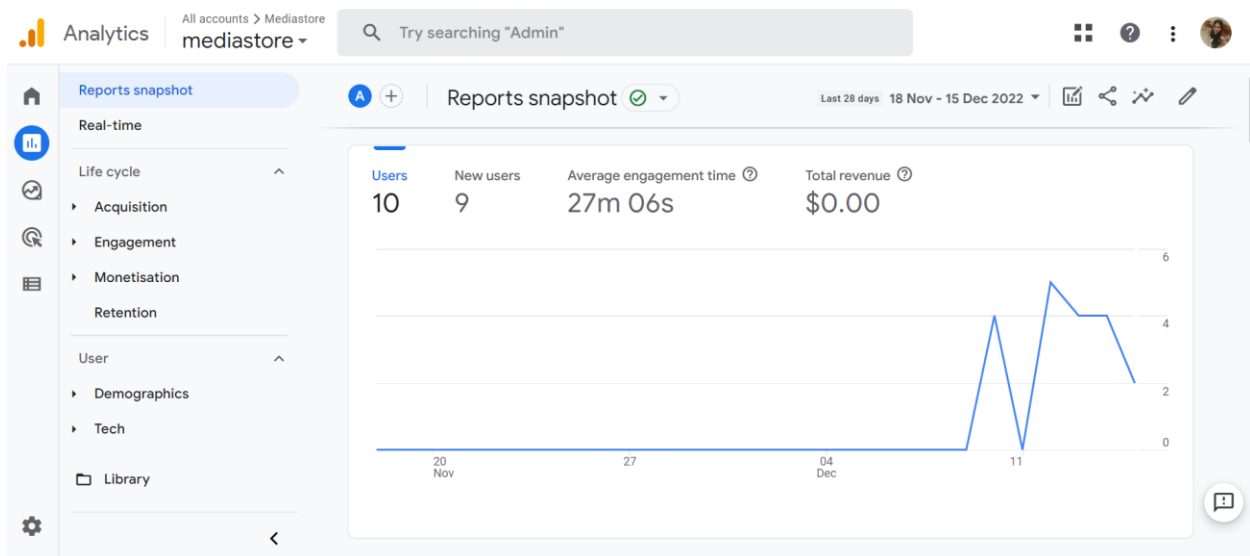
In our github repository, we have a main branch and a develop branch. We used develop branch for development by the individual developers and the code was pushed to main branch only after integration testing of a milestone. Time to time some other branches were made from develop by a specific developer to push his/her changes. All team members had access to the develop branch whereas only the team lead and the github master had access to master branch.

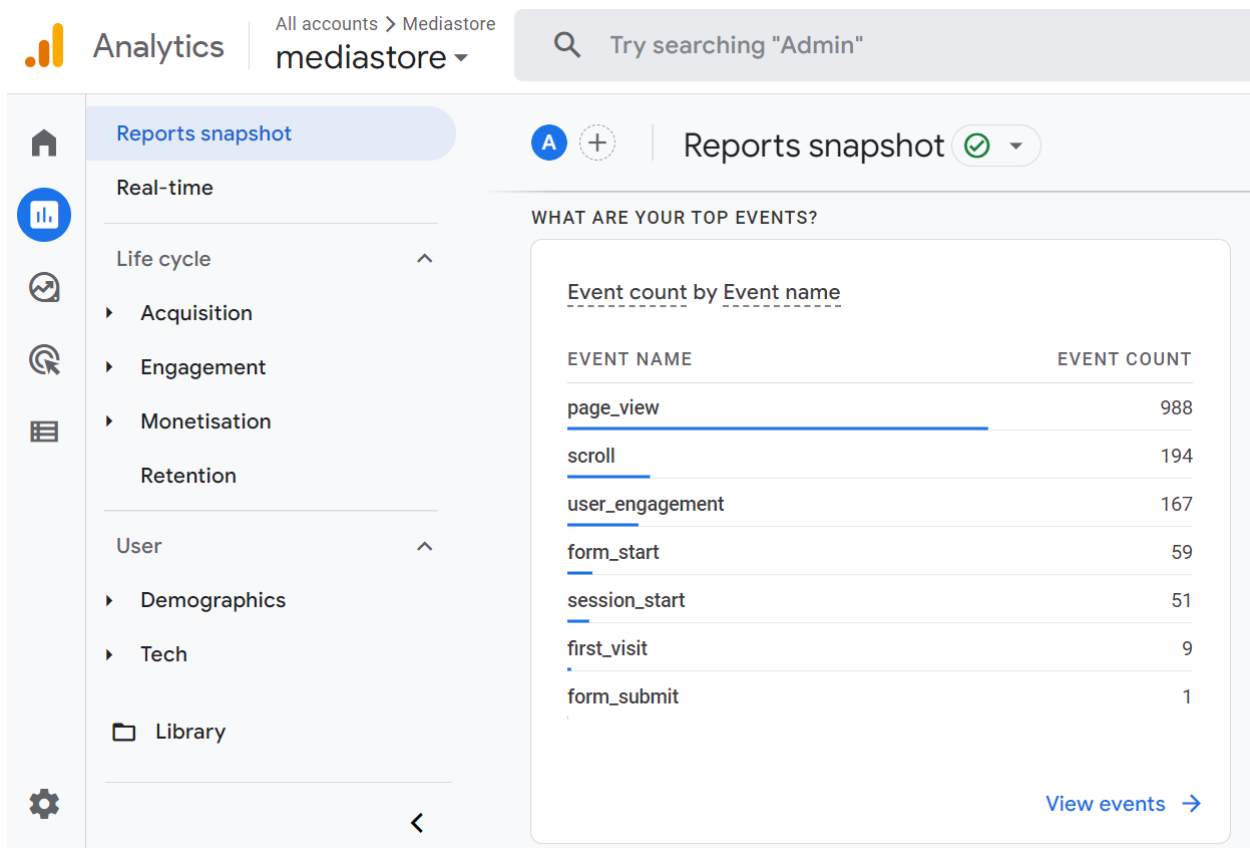
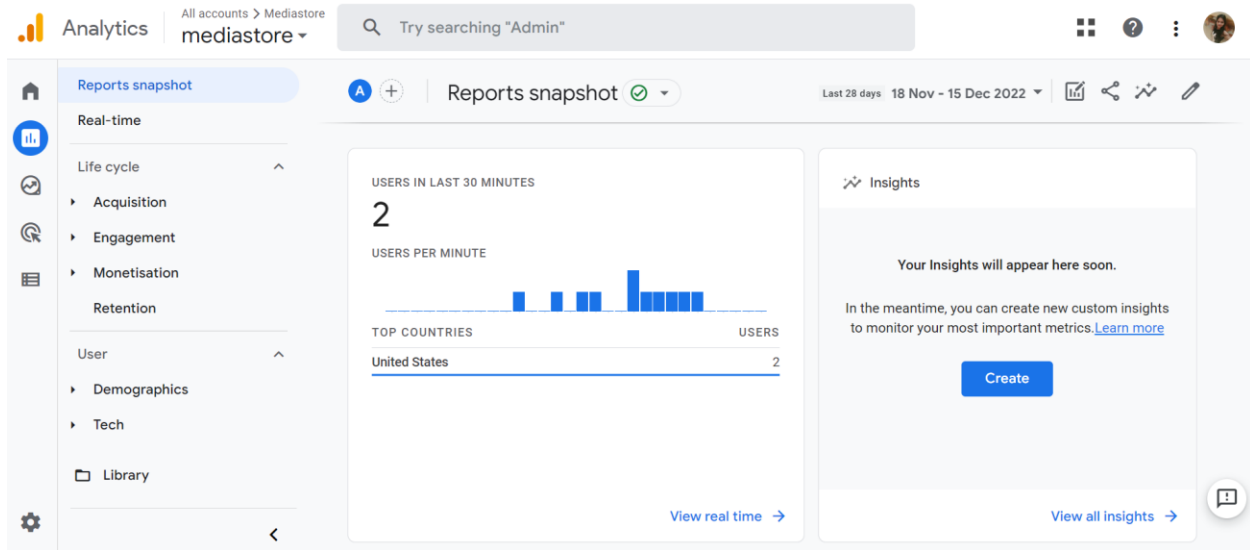
Our github application folder is organised as per the MVC architecture to have front and code separated from the back-end code. Inside the application folder we have a “client” folder for front end code and the “backend” folder for backend code.

The screenshot shows a GitHub repository page for 'CSC-648-SFSU / csc648-03-fa22-team03'. The repository is private and has 174 commits. The 'Code' tab is selected, showing a list of files and folders: Milestones, application, credentials, LICENSE, and README.md. The 'About' section on the right indicates the repository was created by GitHub Classroom and has 1 star and 0 forks. The 'Releases' section shows no releases published.

File/Folder	Commit Message	Time Ago
Milestones	Added revised M2 document	17 hours ago
application	Adding static folder	37 minutes ago
credentials	Updated server ip	17 hours ago
LICENSE	Initial commit	3 months ago
README.md	Updated ip address of server	17 hours ago

## 7) Google analytics stats plot for web site

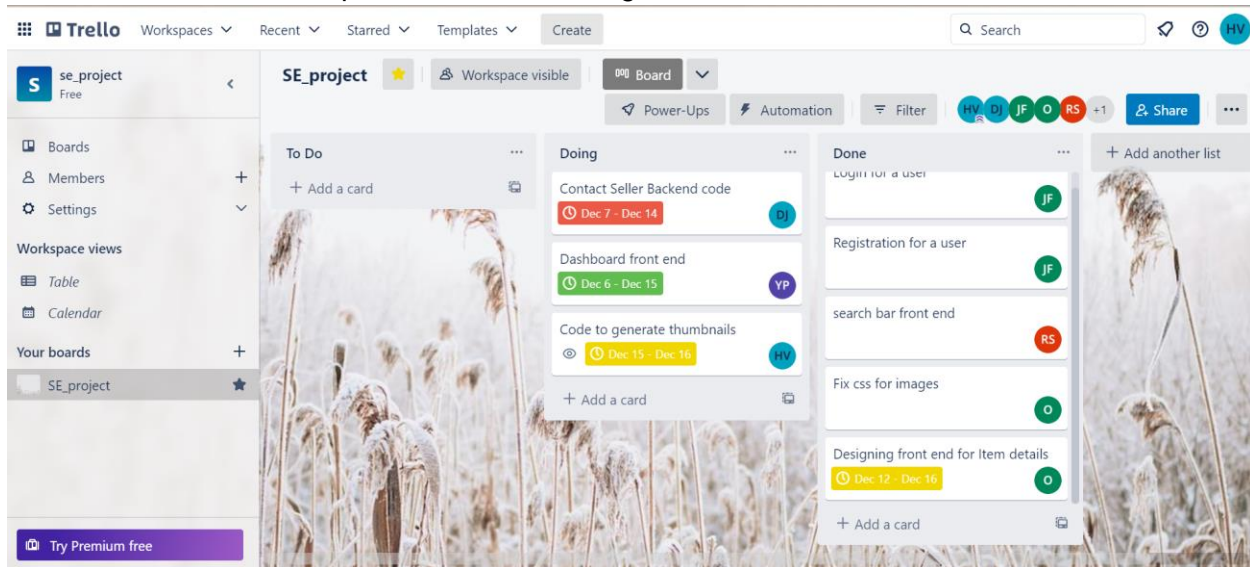




## 8) Project Management

We have been assigning tasks in our class team meetings and using discord and ZOOM meetings to update our progress on tasks up until now. For keeping record of who is doing which task, the team lead assigns the tasks to each team member based on their preferences and skills. Team keeps track of it using Trello. With Trello we can see who is responsible for what task, their progress on said task, and what tasks are completed or have yet to be started.

For the documentation of product we used Google docs.



## 9) Team member self-assessment and contributions

RP

Fri 12/16/2022 8:30 PM

Robert Peter Swanson

CSC 648 Fall 2022 Milestone 5 Section 9 Team 3- Robert Swanson

To Himani Varshney; Donnovan Jiles; Olimpia Aguilon; Yasaman Pakdel; Josef Fiedler

1. Contributions I made

a. Created the navbar and linked search functionality with the backend

b. Created the footer

c. Created the homepage

d. Created the search results page

e. Created the Upload page and linked functionality with backend

f. Created state tracking to track whether a user is logged in

g. Implements lazy registration on all pages that requires the user to be logged in

h. implemented log out functionality

2. Commits

a. I have made 39 commits to dev branch

3. Challenges

a. My main challenge was working in a frontend framework for the first time. I have only ever coded front end in html and it took some adjustment using a framework instead of plain html. My other main challenge was figuring out how to pass data and maintain state between pages while working in react.

4. What would i do differently

a. The main thing I would do differently is to learn the framework i am using at a much higher level beforehand. I ended up coding things that could have been implemented better from the start if I had a better understanding of the full picture of how to do things in react. Instead, I had to make the decision of rebuilding things from the ground up or trying to find different ways to be able to use the code I have already written. So, to sum it up I would say I needed to understand the framework better to make everything more scalable.





Fri 12/16/2022 5:13 PM

Donovan Jiles

CSC 648 Fall 2022 Milestone 5 Section 9 Team 3- Donovan Jiles

To Dragutin Petkovic

Cc Himani Varshney; Robert Peter Swanson; Yasaman Pakdel; Josef Fiedler

The contributions I made to the team project include: development of functional requirements (prioritized and non-prioritized), implementation of search functionality, implementation of post functionality, implemented message functionality, established flask server configuration and database connection, QA testing.

I made 9 commits to develop. My commit count is relatively low due to Himani committing collaborative work.

One of the main challenges I ran into while doing this project was correctly saving files locally. Our React server was not sending the form data to our Flask server and I was confused on whether it was a front-end or back-end issue. After tons of google searches I found that there exists a React constructor called `FormData()` that creates a new `FormData` object which enables React to send form data properly to the Flask server.

What I would Do differently next time is focus more on time management and getting things done in a timely manner. I would also have read documentation more thoroughly, because the learning curve for React was steep.



Sat 12/17/2022 11:31 AM

Josef Fiedler

Team Contributions

To Himani Varshney

Cc Robert Peter Swanson; Yasaman Pakdel; Donovan Jiles; Olimpia Aguilon

The main areas in which I contributed are server deployment, setting up a CI CD pipeline, creating the database, implementation of login and registration functionalities and dashboard info (user, messages, posts) functionality.

I made 55 commits to the develop branch.

The biggest challenge was making the several pieces of software (react, node, nginx, flask) work together on our AWS server. Another big challenge was to assure good practices using Github, but the team has handled this task very well.

I would implement a swagger as soon as backend development started for an easier work with the flask APIs.

Hope to see you guys all today! 😊

Josef



Olimpia Aguillon

Dragutin Petkovic; Himani Varshney; Robert Peter Swanson; Yasaman Pakdel; Josef Fiedler; Donovan Jiles

1 4:21 PM

CSC 648 Fall 2022 Milestone 5 Section 9 Team 3- Olimpia Aguillon



CSC 648 Milestone 5 Section 9.pdf  
27 KB

- a)
1. Created login page- Requires SFSU email and password. Provided options for users to click on forgot password link or signup link.
  2. Created the forgot password page- Requires users to input their email.
  3. Created signup page- Requires first name, last name, SFSU id, SFSU email, password, and checking off the terms and conditions box. Provided a link to go to the login page for users with existing accounts.
  4. Created details page for both free posts and contact seller posts- Displays the image, the title, the author, the category, and the price. Also shows the download button or the contact seller button according to the post selected.
  5. Created the contact seller messaging page- This is a form that has prefilled the seller's name and the title of the item. Which then leaves the user to provide the date and the message they want to send.
- b)
- I made 17 commits to the develop branch.
- c)
- The first main challenge I had was getting familiar with react.js and bootstrap. I had never used it and setting up my ide was a bit challenging to figure out, but once I knew which commands to run I was able to get started. The next thing that was challenging was creating the contact seller button. For some reason when I tried to use the button tag it wouldn't link to my message page so instead, I dressed up the link to look like a button using css. The most challenging thing was connecting the backend. I had to figure out how to send data as well as how to receive data but luckily there were many tutorials online and I was also able to look at my classmate's code for reference.
- d)
- If I had to do this again I would have not hard coded, to begin with. I know that during milestone 3 we had to show what the design would be. So I know that's why we did it, but I think it was really challenging transiting from it being hardcoded to having the backend provide data. I feel like the time I spent hardcoding could have been spent thinking about receiving and giving data from/to the backend.

Best,

Olimpia Aguillon (Team 3)



Sat 12/17/2022 8:37 AM

Yasaman Pakdel

Milestone 5 part 9 - my team contributions

To Himani Varshney; Robert Peter Swanson; Olimpia Aguillon; Donovan Jiles; Josef Fiedler

- a) My contributions were:
- a. Created user profile page
  - b. Created user posts page
  - c. Created user messages page
- b) I have made 3 commits to the develop branch. This number is small because I made most of my commits to other sub-branches and merged them later on.
- c) The main challenge for me was learning React and Bootstrap, as this was the first time I was working on the front end. Connecting the front end to backend was also very challenging for me.
- d) One thing I would do differently is that I would study the front end framework more in advance and spend more time on it before our project actually begins. I also wished I could help my team more and do more tasks, which could have been possible if I had spent more time learning the framework.

Happy Holidays, Gators! ITS wishes you a safe and Happy New Year 2023!!



**My contributions:**

- Integrating Google Analytics to the website
- Deploying the app on AWS server
- Integrating front end to back end apis
- Adding backend code to generate thumbnails
- Helping both front end developers and backend developers in resolving some showstopper technical issues
- Document editor for milestone documents.
- Effectively managing the team by ensuring team members deliver on time.
- Motivating team members time to time.
- Enhancing collaboration among the team
- Setting up softwares like Trello for team management and encouraging the team to use them.
- Setting up Zoom meeting for continuous communication within the team and getting daily updates.

**The number of submissions I made to github dev branch** is 40. It is so because I was more involved in the integration and the deployment part of the application.

**Main challenges encountered:**

One of the main challenges I faced while doing this project was the deployment of application on server. I had limited knowledge of AWS and had to go through several online tutorials to get the app deployed for milestone 0. In fact, there were times when the app was running fine on local but gave issues on the server. As a team lead, the main challenge I faced was the communication gap within the team. There were times when I had to coordinate between them to get things done. However, I saw improvement over time and the team collaboration improved. They were helping in resolving each other's technical issues. One more challenge which I noticed was, some developers pushing the code to github without proper unit testing. This created problem while integration and I had to spend lot of time in fixing bugs.

**What I would do differently** based on my learnings in the class would be better Usability testing to make the application more user centered. I would also have frequent QA to detect bugs in early stage. As a team lead, I would encourage team members to better collaborate with the team and bring up issues at the earlier stage. I would use team management tools like Trello more effectively. I have learnt a lot through this course and I would definitely carry these learnings when I start working.

