

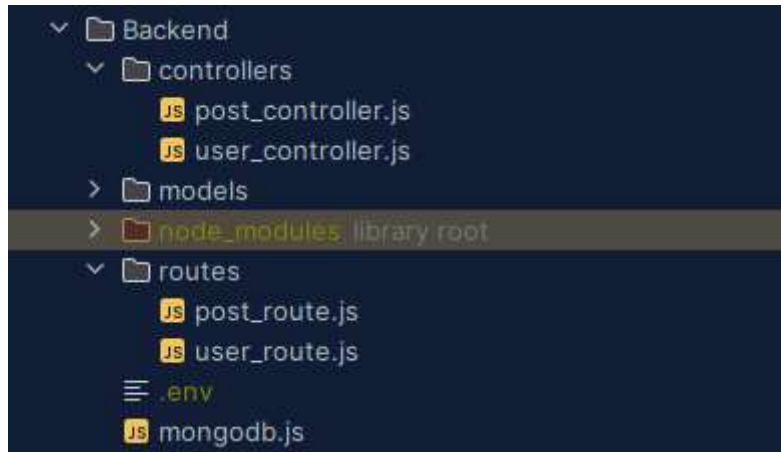
# Core-Marker Design Report

---

By: Jimmy Delgado-Hernandez, Peter Lu, Luke Nguyen, Shiva Shashank Adlagatta

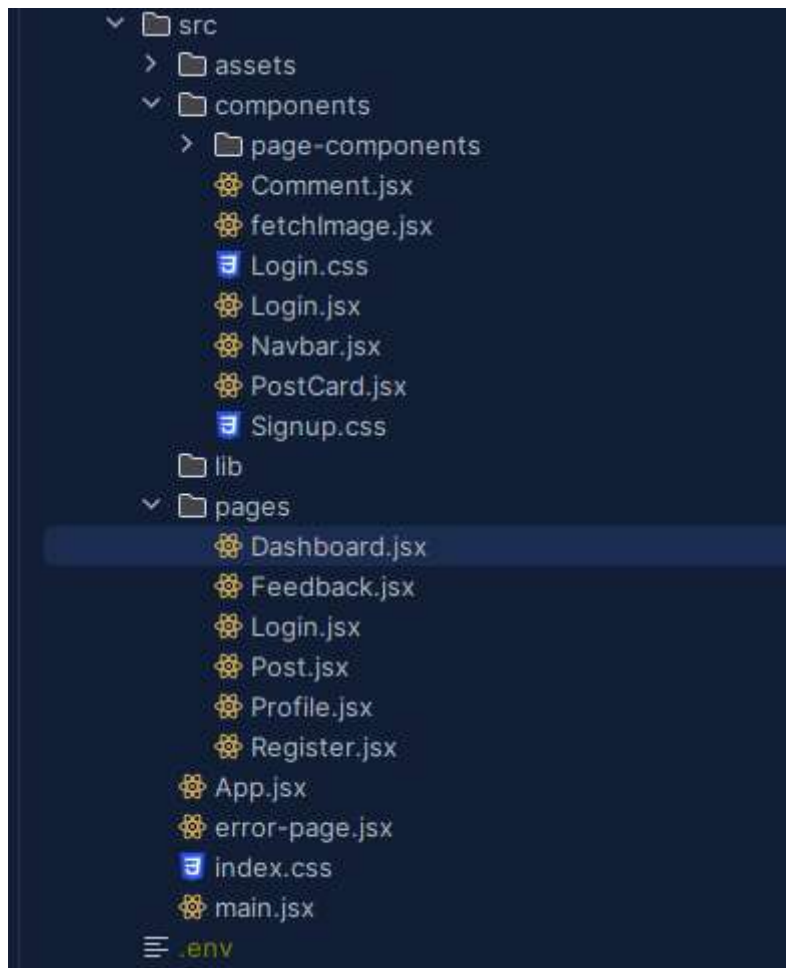
## Design Description

Our app architecture uses a modular monolithic approach. We split our backend and frontend into separate directories and this allows us to create a separation of concerns for our app. We choose to do this because it allows our team to split into a frontend and backend sub teams. This not only helped determine our skills and weaknesses but also how we could debug issues by knowing where errors happened either from the frontend or backend. In the backend our folders use routes and controllers to handle API requests and return responses.



Our project is primarily backend focused so much of the rendering logic is done “near” the database which allows us to load HTML and pages on the backend before they are sent to the frontend. While this may show slight load times between 0.5-1 seconds, this allows us to keep our code secure because all the operations are handled on the server side which is more secure and less prone to JS injections and HTML manipulation from the client side.

In the frontend we use component based architecture which is a strength of React which is our frontend library. This allows us to develop our frontend very quickly by making reusable components and splitting our pages into their own files.



We have 5 main pages in our application and a multitude of components. Some of the major components are the Navbar, Login, and PostCard. The Navbar renders on all pages inside the app after logging in. The Login component holds the forms and buttons that actually perform client side actions such as sending a request for data and routing a user to the homepage after a successful login. The PostCard is the actual posts that show up on the home and profile page. Most or all post structures are the same so all we had to do was create a component and add the data fetching to display unique post information.

### **Libraries:**

We had used libraries such as React, Tailwind, and Chakra UI to help us perform

asynchronous web application under the styling of Tailwind Library under CSS.

Tailwind and Chakra UI helped speed up UI development and allow us to focus on the more challenging frontend aspects such as RESTful practices and sending HTML methods to the backend.

## **API**

The APIs we used was a combination of NodeJS and ExpressJS and Mongoose. Together they serve as our REST API which combined with CRUD operations in the database made it easy to handle full-stack operations where we managed requests and responses between client and server. Express also has middleware support which primarily helps with form and database validation. Database validation is a crucial part of our project because we need to handle updates to the database and valid correct requests and sent responses. Express added a layer of security by helped reduce boiler and heavy code if we only used nodejs which helps us debug code that improves our development experience.

Mongoose is an ODM we used to handle model and schema and We referred to their APIs to make database operations easier in terms of code writing. Mongoose allowed us to create a database connection and make CRUD operations to our database such as updating user avatar and banner or updating and deleting a post more streamlined. Mongoose also is nicely integrated with Mongodb as their own ODM which allows us to perform object oriented programming instead of writing mongodb raw queries.

## **Languages:**

We had coded our project under a base of HTML, CSS, and JavaScript, as we coded under the react file extension .jsx as our web application is made as a react vite App. JSX is just a javascript extension that allows us to write HTML in between our javascript code, a unique feature of React applications. Nodejs and Express are both written in javascript so overall our application was almost a single language app which played to our strengths in a language everything was comfortable in and without introducing other languages.

## **System Setup:**

To be able to run our web application, the user must install git and git clone our project such as “git clone <https://github.com/CS-179K/Core-Marker.git>” onto their terminal. The user must have had their folder routing set up to be able to cd their way into our app so that the path can look like “[User personal pathway]/Core-Marker/core-marker-app”. Once the user has gotten a path similar to “../Core-Marker/core-marker-app”, the user must then be able to run “npm install” so that the user can have Frontend capabilities. Additionally, the user must have a path similar to “../Core-Marker/core-marker-app/Backend” and run “npm install express” so that the user can have Backend capabilities to connect our Database with our Frontend. The user must then open two terminals, one for the Frontend with a path like “../Core-Marker/core-marker-app” to run it with command “npm run dev”, and a second one for the Backend with a path like “../Core-Marker/core-marker-app/Backend” to run it with command “node mongodb.js” to have the web app run as intended.

### **User Interaction:**

Once the User is able to set up our application on their end, the User starts at the Login Page. The Login page is a normal login page with email and password fields to fill in and a “Login” button to actually login into your account. For a first time user, there’s an extra button labeled “Sign up” underneath the Login button that would take them to the Register page to register their account for the first time. Once filling in the register fields, the app routes the user back to the Login page to Login with their new account. Once the user logs in, they can see the Home page which should display a navigation bar on the top of the page with the user’s avatar on the top right along with the logout button. The main Home page displays all the posts any user had been able to create a post for everyone to see in the home page. For a user to create a post, a user must press the “Post” button to fill in the fields to create and share a post with everyone. User’s can also send a feedback form by clicking on “Support” on the navigation bar on the top of any page to fill in the form and send the form to us developers. Users can also see their profile page by clicking their username on the top of the page which leads them to their profile page with their personally created posts.

### **Use**

Our Web application is a small social media app for anyone to login and communicate with other users through posts and comments about their life experiences they had gone through in their local communities or elsewhere.

## **Evaluation**

This section goes over our test cases and evaluation of our user stories and how many of our test cases passed.

### **Overview:**

Our test cases are organized by the major pages and features in our app.

Login: Test that you are able to login with a registered account using correct email, username, and password. Test that any incorrect fields will not prompt a successful login. A successful login must redirect a user to the homepage.(3)

Register: Test that you are able to register an account. Test that any incorrect fields will not allow successful creation of a new account. A successful registration should prompt the user success and redirect to the login page. The account must be saved to the database. The password must be hashed before being sent to the database.(3)

Profile: Test that you are able to see the posts you created on your own profile and not other users' posts. Test that you are able to see personalization such as correct account username and email. Test that you are able to edit your profile but updating the avatar or banner. Test that the database validates a correct image type. Test that a successful update of avatar or profile changes the image immediately after database validation.(5)

Comment and Likes: Test that you are able to like and unlike a post. Test that you are able to see the count of likes go up or down depending on interactions. Test that you are able to see a comment field when clicking on viewing all comments. Test that comments and likes persist and save to the database.(4)

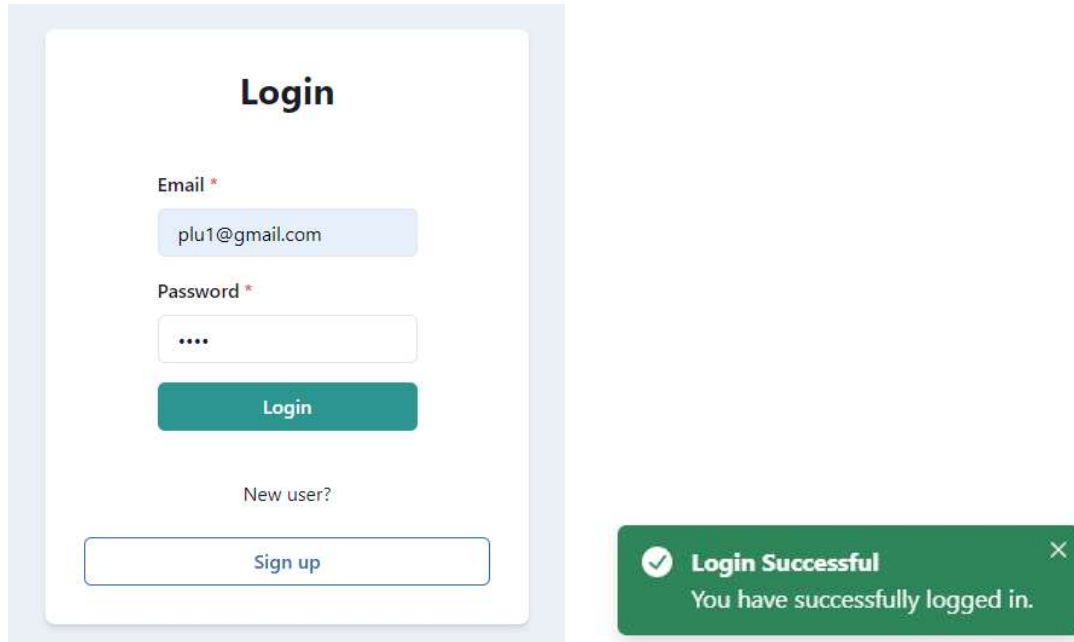
Posts: Test that users are able to create a post. Test that users are only able to create a post when they fill in all parts of the form. Test that the database saves a newly created post. Test that posts show up on the homepage for all users.(4)

Feedback: Test that users are able to fill out the form. Test that users are able to send a feedback form. Test that developers are able to receive an email of the feedback form. (3)

## **Results:**

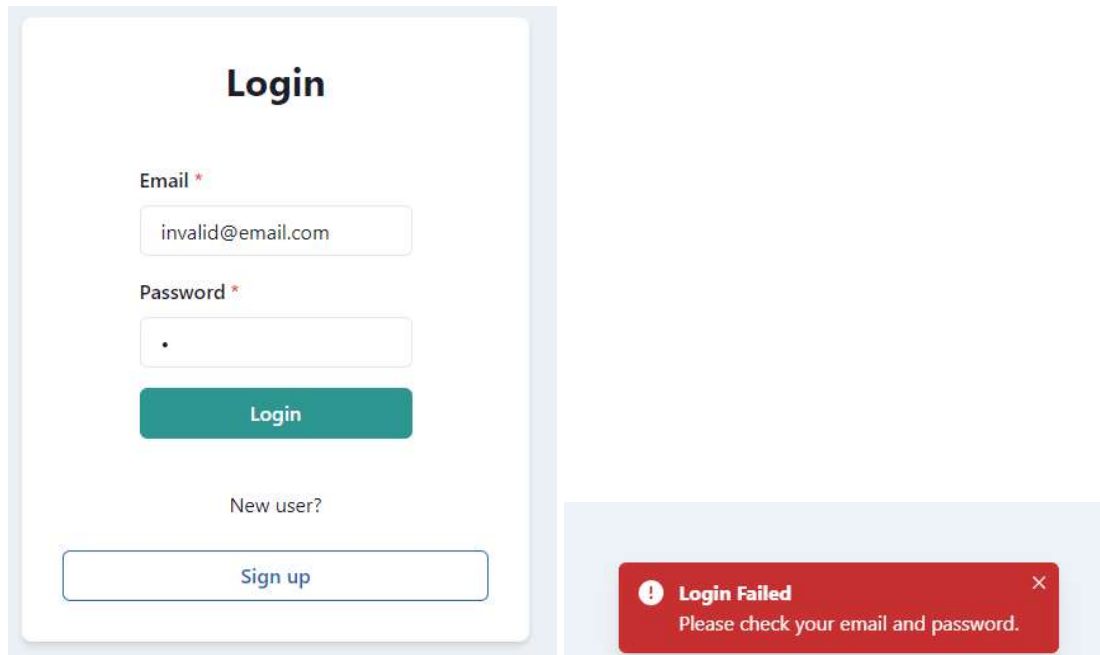
Login:

1. Test successful login with correct email, username, and email(pass)



The screenshot shows a login form titled "Login" with two input fields: "Email \*" containing "plu1@gmail.com" and "Password \*" containing four dots. Below the fields is a teal "Login" button. At the bottom of the form is a "New user?" link and a "Sign up" button. To the right of the form is a green notification box with a checkmark icon, titled "Login Successful", and the message "You have successfully logged in." with a close button (X).

2. Test any incorrect fields will not prompt a successful login(pass)



The screenshot shows the same login form as above, but with incorrect data: the "Email \*" field contains "invalid@email.com" and the "Password \*" field contains a single dot. The "Login" button is teal. Below the form is a "New user?" link and a "Sign up" button. To the right of the form is a red notification box with an exclamation mark icon, titled "Login Failed", and the message "Please check your email and password." with a close button (X).


3. Test that successful login redirects user to home page(pass)

Core Marker


Post

Support

plu1



Logout



Golden Gate Bridge at Dawn!!


plu1

Caught the first light of day over the Golden Gate Bridge. The early morning fog was slowly lifting, revealing the iconic bridge against a backdrop of soft, golden hues. The serene and quiet atmosphere was a perfect start to the day, with the bay's still waters reflecting the majestic structure. A moment that perfectly captures the beauty and grandeur of San Francisco

Location: Golden Gate Bridge, San Francisco, California

0 Likes

View All Comments



A Stroll Through Kyoto's Bamboo Forest


plu3

Walking through the towering bamboo groves of Arashiyama feels like stepping into another world. The gentle rustle of bamboo leaves and the dappled sunlight create a peaceful and enchanting atmosphere. Kyoto's natural beauty never ceases to amaze. 🌿🇯🇵 #Kyoto #BambooForest #JapanTravel

Location: Arashiyama Bamboo Grove, Kyoto, Japan

0 Likes

View All Comments



Discovering the Streets of Istanbul


plu3

Strolled through the vibrant streets of Istanbul, where the blend of historic and modern elements creates a unique atmosphere. From the bustling Grand Bazaar to the serene Blue Mosque, the city's rich cultural heritage was on full display.

Location: Istanbul, Turkey

0 Likes

View All Comments



Historic Charm of Dubrovnik


plu3

Wandered through the historic streets of Dubrovnik, Croatia. The well-preserved medieval architecture, stunning coastal views, and charming old town created a picturesque setting. The city's rich history and vibrant culture were evident at every turn.

Location: Dubrovnik, Croatia

0 Likes

View All Comments



Majestic Views of the Great Wall

py1u

Hiking along the ancient stones of the Great Wall of China was an awe-inspiring experience. The sweeping views of the rugged mountains and the rich history embedded in every brick made it a journey to remember. Standing on this historical marvel, it's easy to feel connected to the past and inspired by the strength and ingenuity of those who built it. 🇨🇳 #GreatWallOfChina #HistoricalWonder #TravelChina

Location: Mutianvu. Great Wall of China.

Total: 3/3

Register:

1. Test that you are able to create a new account:(pass)

## Core Marker

Username \*

Email \*

Password \*

Register

Already have an account?

✓ **Registration Successful**

You have successfully registered. Redirecting to login.

×

2. Prompt a user with success and redirect after creating a account(pass)
3. Save a new user to the database and password must be encrypted before saving to db(pass)

```
_id: ObjectId('66d57f75f89a6c5a3713e5e4')
name : "plu5"
email : "plu5@ucr.edu"
password : "$2a$10$UuA67VV8t6zaV6PZxI82NuvBWghIlu7FTLK95VuPXFjQc/prq8csq"
avatar : "44.png"
banner : "22.png"
createdAt : 2024-09-02T09:03:49.432+00:00
updatedAt : 2024-09-02T09:03:49.432+00:00
__v : 0
```

\* password is already hashed when saved to db

Total: 3/3

Profile:

1. Profile shows posts that you created and not others(pass)





**plu1**  
plu1@gmail.com



**Golden Gate Bridge at Dawn!!**


Caught the first light of day over the Golden Gate Bridge. The early morning fog was slowly lifting, revealing the iconic bridge against a backdrop of soft, golden hues. The serene and quiet atmosphere was a perfect start to the day, with the bay's still waters reflecting the majestic structure. A moment that perfectly captures the beauty and grandeur of San Francisco.

Edit


Delete

Core Marker

PostSupportplu1



Logout



Golden Gate Bridge at Dawn!!


plu1

Caught the first light of day over the Golden Gate Bridge. The early morning fog was slowly lifting, revealing the iconic bridge against a backdrop of soft, golden hues. The serene and quiet atmosphere was a perfect start to the day, with the bay's still waters reflecting the majestic structure. A moment that perfectly captures the beauty and grandeur of San Francisco

Location: Golden Gate Bridge, San Francisco, California

0 Likes

View All Comments



A Stroll Through Kyoto's Bamboo Forest 🌿🌿


plu3

Walking through the towering bamboo groves of Arashiyama feels like stepping into another world. The gentle rustle of bamboo leaves and the dappled sunlight create a peaceful and enchanting atmosphere. Kyoto's natural beauty never ceases to amaze. 🇯🇵 #Kyoto #BambooForest #JapanTravel

Location: Arashiyama Bamboo Grove, Kyoto, Japan

0 Likes

View All Comments



Discovering the Streets of Istanbul

plu3

Strolled through the vibrant streets of Istanbul, where the blend of historic and modern elements creates a unique atmosphere. From the bustling Grand Bazaar to the serene Blue Mosque, the city's rich cultural heritage was on full display.


Location: Istanbul, Turkey

0 Likes

View All Comments

\* only one post should show given that the account has only made on post

2. Show correct username and email for user profile(pass)



plu1

plu1@gmail.com

3. Test you are able to update avatar or banner(pass)



**plu1**  
plu1@gmail.com

✓ **Avatar updated.** ×  
Your avatar has been successfully updated.

✓ **Banner updated.** ×  
Your banner has been successfully updated.



4. Test database has correct data type of string with base64 conversion(pass)

```
_id: ObjectId('66d3c5b5e5f1cd97bc10bde8')
title: "Golden Gate Bridge at Dawn!!"
description: "Caught the first light of day over the Golden Gate Bridge. The early m..."
likes: 0
location: "Golden Gate Bridge, San Francisco, California"
imageUrl: "data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAQABAAQ/4gHYSUNDX1BST0ZJTEU..."
userId: ObjectId('66ccfaf5a671ef42a492fbb4')
createdAt: 2024-09-01T01:39:01.437+00:00
updatedAt: 2024-09-02T03:39:10.297+00:00
__v: 0
```

5. Check that profile and avatar update after database validation(pass)



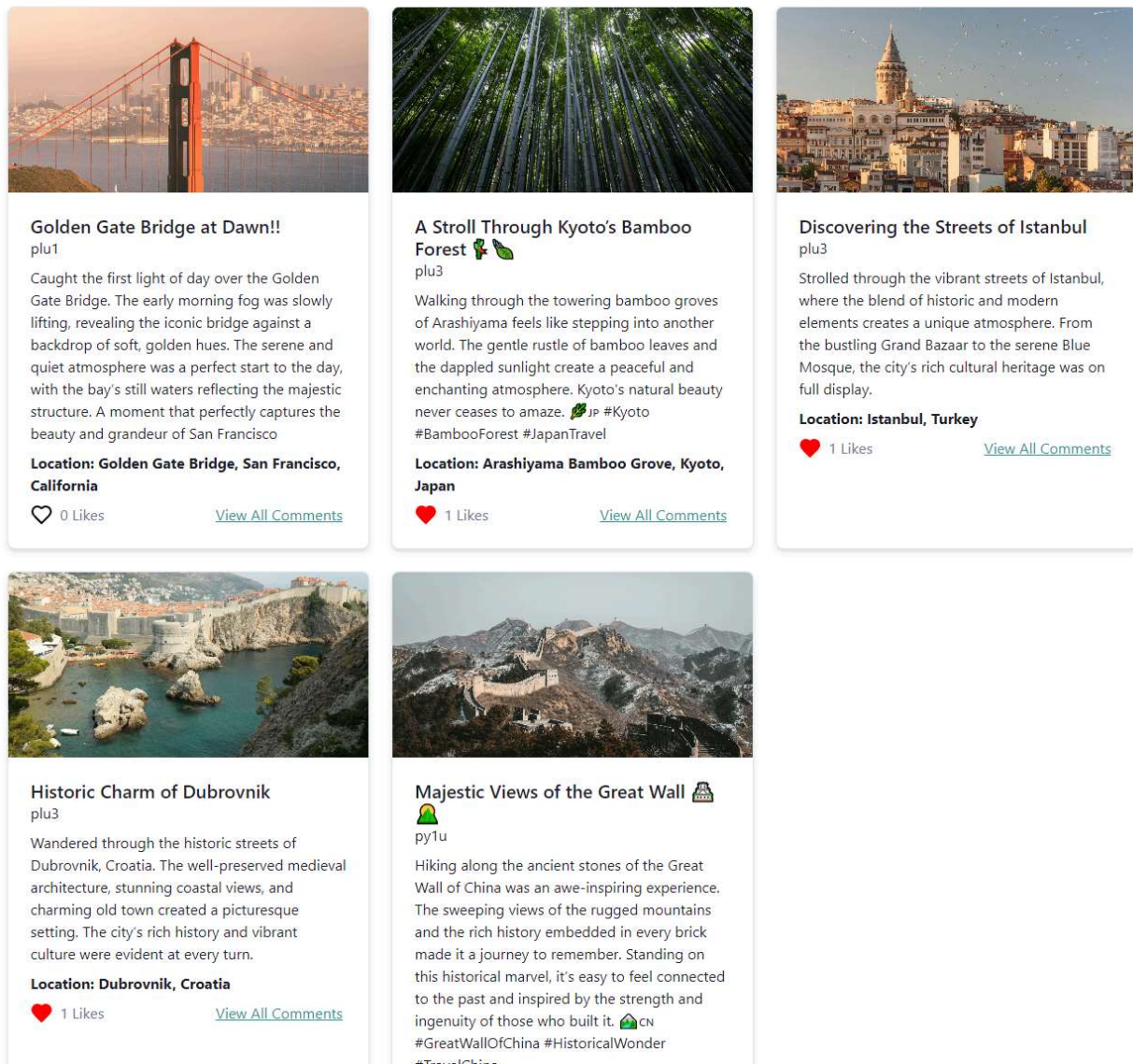
\*note the update in the navbar avatar as well

Total: 5/5

Comments and Likes:

1. Test that you are able to like and unlike a post(pass)

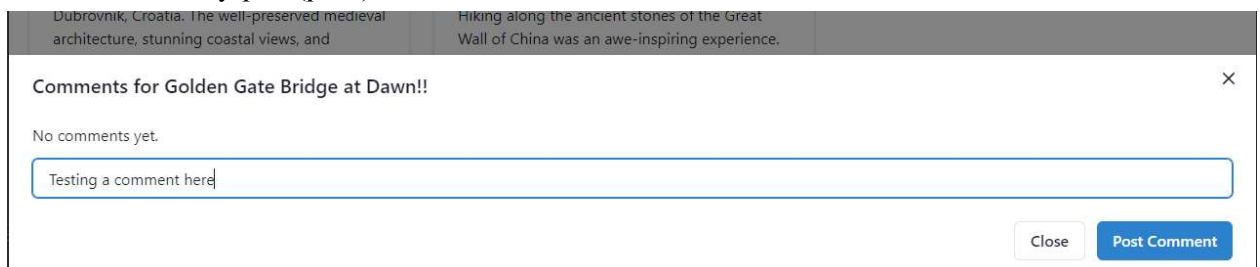




2. Test that like counts change based on interaction(pass)



3. Test you are able to type a comment into the comment field on clicking View All comments for any post(pass)



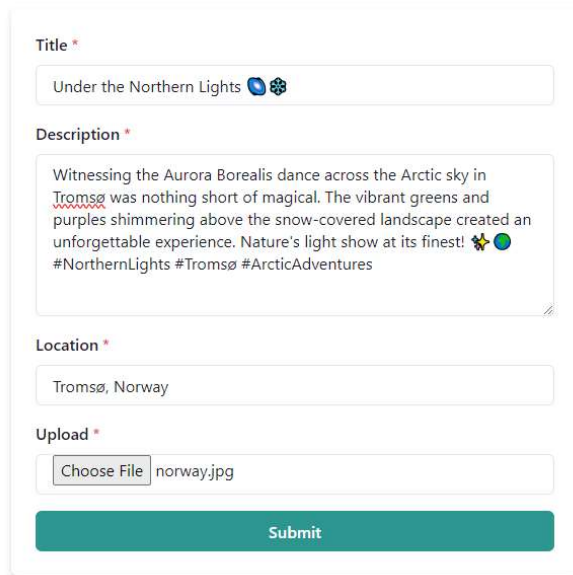
4. Test that likes and comments persist to database(fail)

Total: 3/4

Posts:

1. Test that users are able to create a post(pass)

## Create a New Post

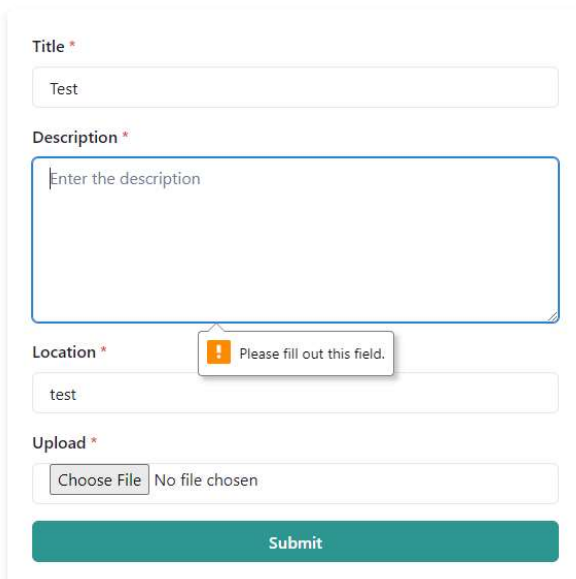


The screenshot shows a form titled "Create a New Post" with the following fields and content:

- Title \***: Under the Northern Lights 🌌🌌
- Description \***: Witnessing the Aurora Borealis dance across the Arctic sky in Tromsø was nothing short of magical. The vibrant greens and purples shimmering above the snow-covered landscape created an unforgettable experience. Nature's light show at its finest! 🌟🌍  
#NorthernLights #Tromsø #ArcticAdventures
- Location \***: Tromsø, Norway
- Upload \***: Choose File norway.jpg
- Submit** button

✓ **Post created.**  
Your post has been successfully created. ✕

2. Test that a post must have all fields filled in before creating a post(pass)

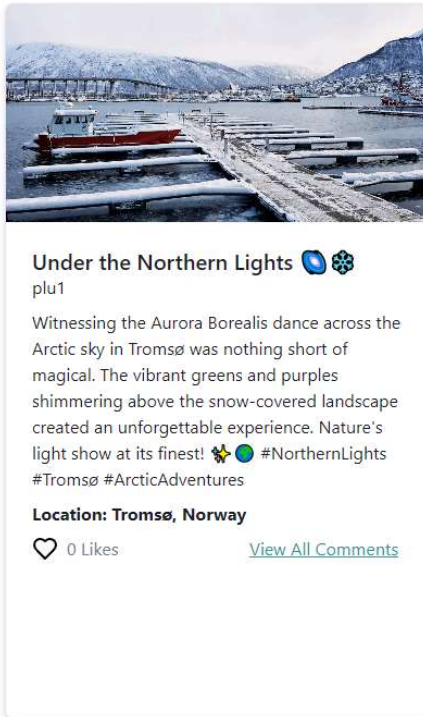


The screenshot shows the same form as above, but with the following changes:

- Title \***: Test
- Description \***: Enter the description (empty text area)
- Location \***: test
- Upload \***: Choose File No file chosen
- Submit** button
- A validation error message is displayed: **!** Please fill out this field. (pointing to the description box)

\*includes missing input in description box

3. Test database successfully saves a created post(pass)
4. Test that a new post shows up on the homepage(pass)



5.

**Feedback:**

1. Test that users are able to fill out a form(pass)

**Name \***

Peter Lu

**Place \***

Los Angeles, California

**Email \***

plu1@gmail.com

**Subject (50 words max) \***

I want to share a post with a URL link

**Context \***

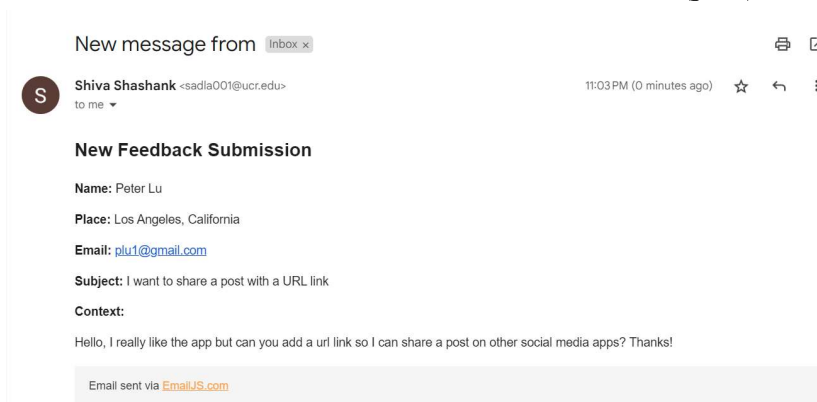
Hello, I really like the app but can you add a url link so I can share a post on other social media apps? Thanks!

**Submit Feedback**

2. Test users can send a feedback form(pass)



3. Test that we receive a feedback form in email(pass)



Total: 4/4

**Assessment:**



Based on our test cases we pass most of our test cases. Except for comments and like features which were unsuccessful in saving to the database.

### **Next Steps:**

Our next steps are to fix the like and comment backend logic to save to the database. We also want to introduce the feature to allow users to view profiles. This would increase user interactions with other users and possibly a follower feature as well.

### **Project Resources**

Peter Lu: <https://github.com/py1u>

Jimmy Delgado-Hernandez: <https://github.com/Maker424>

Shiva Shashank Adlagatta: <https://github.com/sala001>

Luke Nguyen: <https://github.com/lukenguyen353>