

# Group 8 Project Report

Application Title:

**Fotos4Family**

Group Members:

- Emiliano Medina Gonzalez
  - Github: Emilianomedina10
- Daria Ruchala
  - Github: Daria-ruchala
- Brandon Muhlenbruch
  - Github: brandonMuhlenbruch
- Christian Vandenburg
  - Github: christianvandenburg
- Vijay Khatri
  - Github: VKH2004

Project Description:

Fotos4Family is an interactive travel map designed for users to visually document and share their experiences with family and friends. Instead of just listing past trips, users can pin locations on a dynamic map, attaching images, notes, and personal stories to each spot. When a user clicks on a pinned location, they can view photos and details of the experience, creating a visual travel diary that is engaging and easy to navigate. This allows travelers to relive their memories while providing their close circle with authentic recommendations and insights about places they've visited. This will also enable users to share photos from the same event.

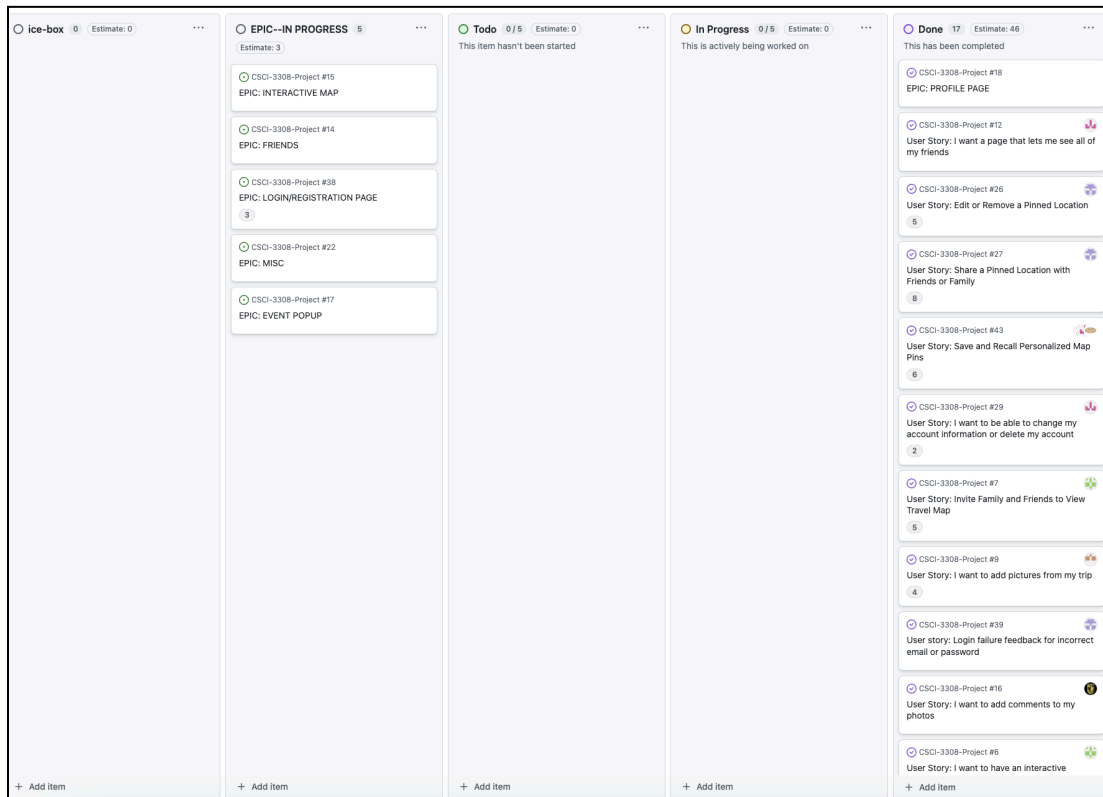
Unlike public social media platforms, Fotos4Family is designed for private sharing, ensuring users can securely share their travel experiences only with family and friends. Whether it's a breathtaking hike, a favorite café, or a hidden gem discovered on vacation, users can build a personalized travel history that remains interactive, visual, and meaningful. The app enhances social connectivity by allowing users to explore each other's journeys, fostering conversations around shared experiences and future travel plans. With an intuitive design and real-time updates, Fotos4Family makes travel memories more than just photos, they become interactive experiences.

## Project Tracker - GitHub project board:

Link:

<https://github.com/users/daria-ruchala/projects/2>

Screenshot of Project Board:



## Video Demonstration:

<https://youtu.be/GRCjFl1ViH4>

## VCS:

Link to your git Repository:

<https://github.com/daria-ruchala/CSCI-3308-Project>

## Contributions:

Vijay:

I created the basic skeleton for the friends page as well as the SQL tables we used to store our data. I then implemented the full friend code system using api routes as well as made additions to the profile page, with each user getting a unique random friend code on account creation. I then finished the rest of the friends page with styling and different errors if the user tried to enter a non-existent code or their code.

Daria:

At first, I worked on creating a login and registration page very bare bones. Then transitioned and worked with Brandon to work on storing pins and also adding images with those pins. Adding pins worked to name a pin, delete a pin, and make the comments for pins optional.

Emiliano:

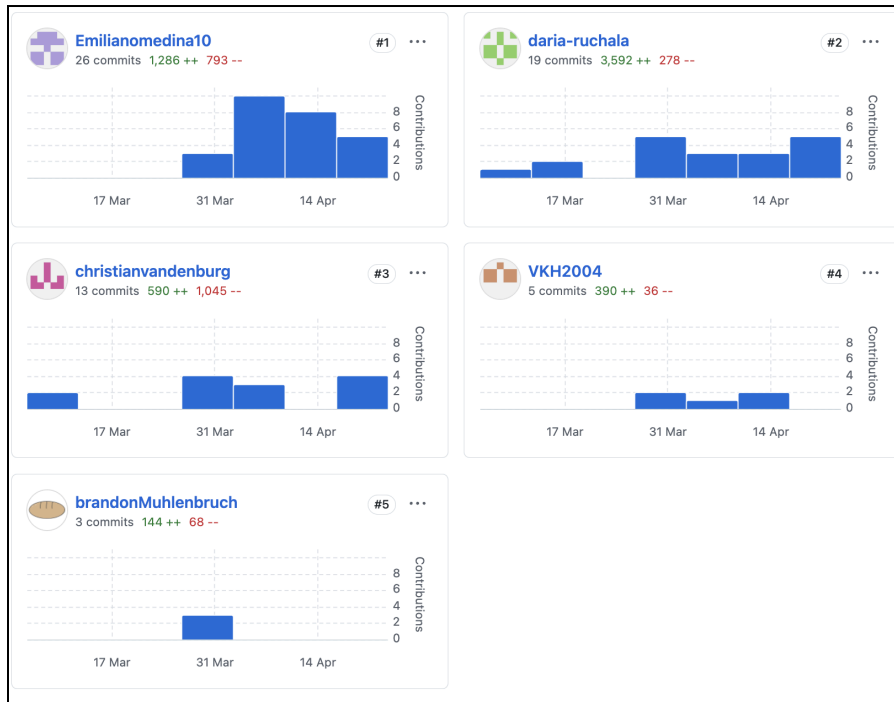
I worked on the front-end design using CSS and Bootstrap, ensuring a clean and responsive layout across all pages. I contributed significantly to the home page, implementing the interactive map features and designing the toggle buttons for adding pins and switching map styles. I also built a filtering system that allows users to view pins from specific friends or only their own. Additionally, I helped with the logout functionality, and the structure of the profile page, and collaborated on the logic behind the add a pin button to ensure seamless integration with the map and database.

Christian:

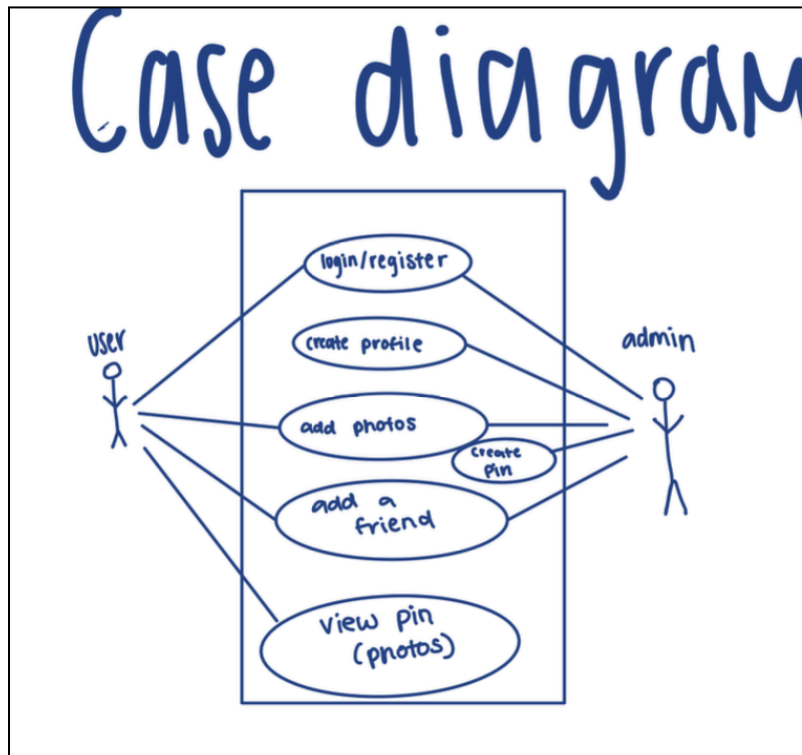
Initially, I was assigned the navigation bar. After initially implementing it, some small changes had to be made as the application evolved with a changed vision. Once I had finished and refined the navigation bar, I changed my focus to allowing a user to edit their profile. Initially, I had let the user change all of their info, but I decided that it was odd to let the user change their email address linked to their account. I had a couple of challenges when implementing this new page, including changing some of the tables to allow the cascading deletion of a user's pins when they delete their account, making sure the password and the verified password held the same data, and rerouting after the session was destroyed.

Brandon: N/A

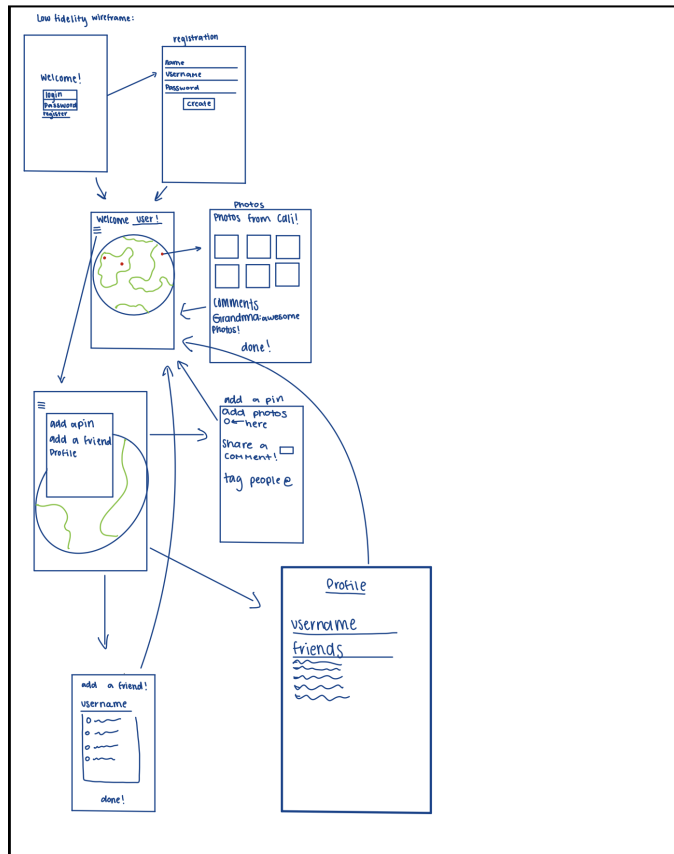
## Screenshot of GitHub Contributions:



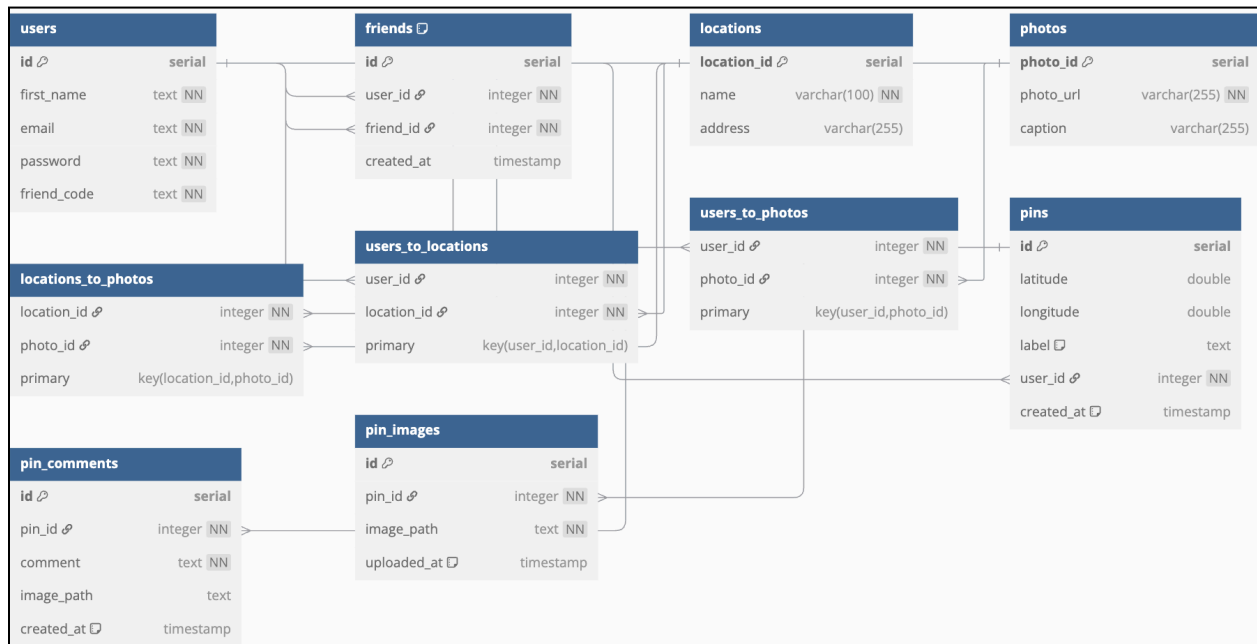
Use Case Diagram:



## Wireframes:



## SQL TABLES:



## Test results:

- <https://github.com/daria-ruchala/CSCI-3308-Project/blob/test-Lab11/MilestoneSubmissions/UATplans.txt>
- <https://github.com/daria-ruchala/CSCI-3308-Project/blob/test-Lab11/ProjectSourceCode/test/server.spec.js>

**(Aside Note: Everything from lab 11 (testing) is on the branch named: test-Lab11:)**

Link for branch: <https://github.com/daria-ruchala/CSCI-3308-Project/tree/test-Lab11>

```
emilianom@Emilianos-MacBook-Pro ProjectSourceCode % docker compose exec web npx mocha test/server.spec.js

Server!
  ✓ Returns the default welcome message

Testing /register API
(node:99) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
(Use `node --trace-deprecation ...` to show where the warning was created)
  ✓ Positive: should register user with valid input (90ms)
  ✓ Negative: should return 400 for missing fields

Redirect testing
  ✓ GET /test should return login page with 200

Render testing
  ✓ GET /login should return HTML

GET /profile (authenticated vs unauthenticated)
  ✓ Should return 401 if not authenticated
  ✓ Should return profile data when authenticated (64ms)

7 passing (237ms)
```

## Test Reports:

During our project development, we incorporated unit and route testing using Mocha and Chai to validate the core functionality of the backend. These tests were run using chai-http to simulate HTTP requests and were executed within our Docker environment. The testing process included verifying the root route response, testing that the: /register, /login returned HTML correctly. One of the more important components was verifying the /profile route. In total, 7 distinct tests were tried and indeed passed without error, confirming that the server's routing and authentication logic were functioning. This process not only helped us validate functionality but also highlighted the importance of regression protection and future test scalability. Putting the Mocha and Chai tests to use was very important to our team to increase our application's dependability. Before putting modifications into production, it allowed us to identify problems early, especially regarding session-based access control and registration validation.

## Deployment:

<https://csci-3308-project-dblu.onrender.com>