Project Report

Title - Rate My Seat
Group Members - Carson Mattei, Derek Marraudino, Jordan Law, Aidan Rotter
Github Usernames:
Jordan Law: jola7852
Carson Mattei: carsomatte
Derek Marraudino: DerekMarr
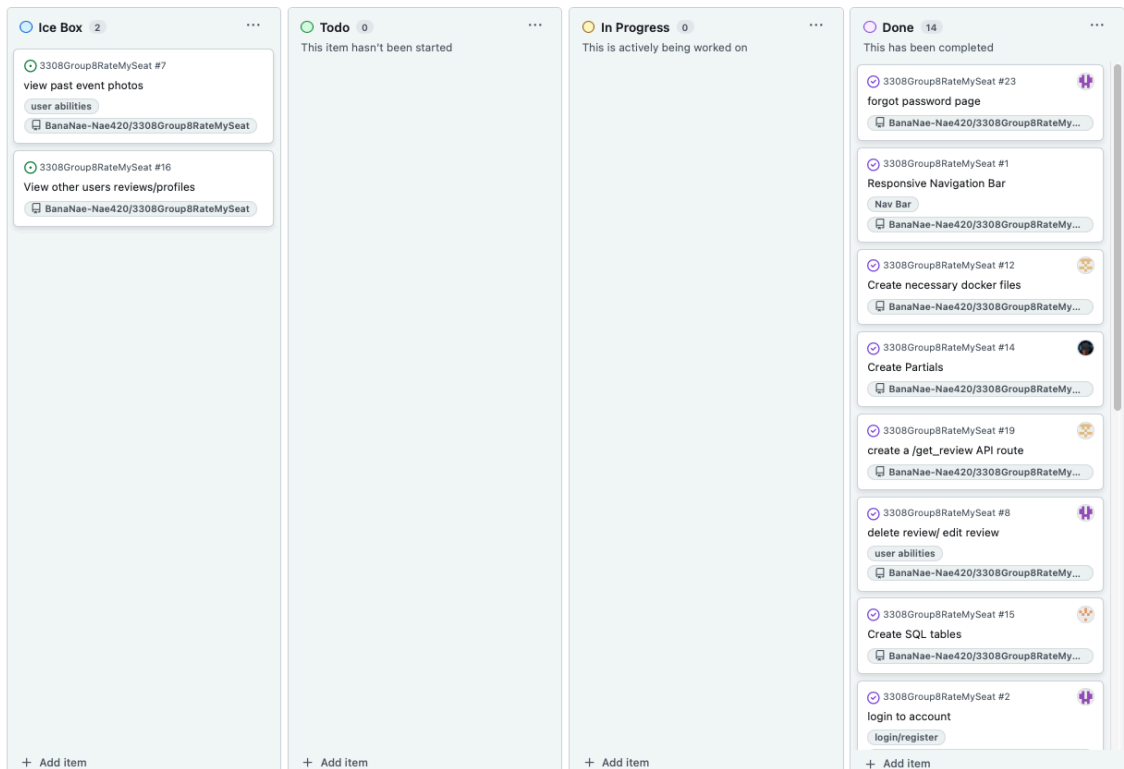Aidan Rotter: BanaNae-Nae420

Git Repository Link - https://github.com/BanaNae-Nae420/3308Group8RateMySeat
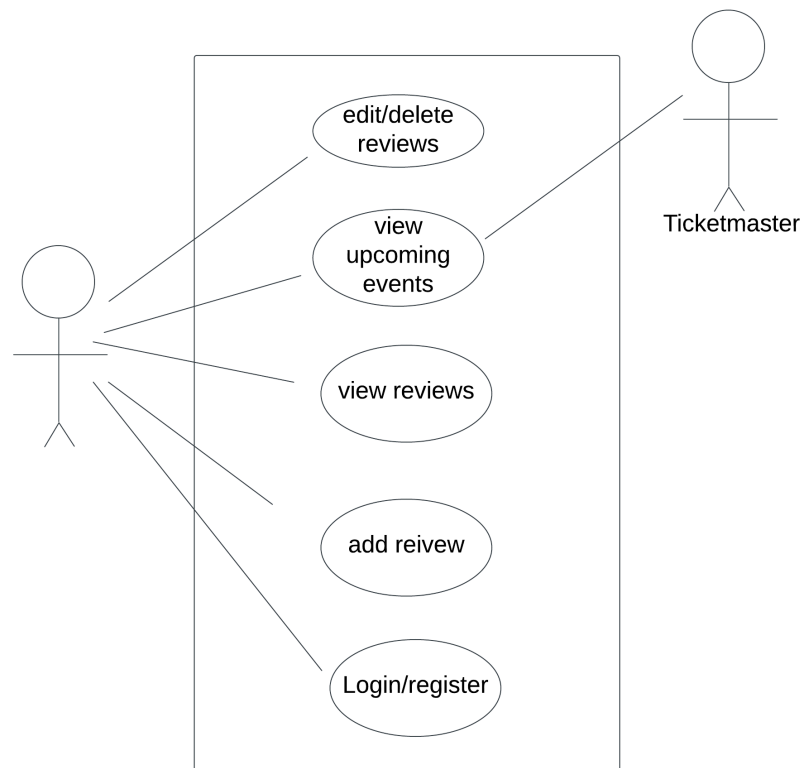**Deployment Link** - https://ratemyseat.onrender.com

Project Description - Our group decided on creating a website in which the main purpose is to view reviews and ratings for seats in a given stadium. If you have ever hesitated while buying a ticket, unsure if the seat you're choosing will offer a great view, then this website aims to eliminate your uncertainty. The website relies on user-submitted reviews to populate reviews in a particular stadium. For instance, when adding a review, you'll be asked for the event name, seat section, seat row, seat number, a 1-5 rating, a review of the seat, and an image of your view from the seat. The website requires you to be logged in to submit a review, and it's also possible to delete or edit one of your reviews. To do this simply navigate to your own reviews under your profile and click the button for your task. Rate My Seat also has a feature to view upcoming events at a given stadium using an external API. The API used is Ticket Master's discovery API and it allows us to grab as many future events as possible and create cards with information such as name, date, and a url to buy tickets. Rate My Seat lets you make sure you get the best seat to meet your needs!

Project Tracker Link/Screenshot - https://github.com/users/BanaNae-Nae420/projects/1

Project Report

Kanban Board

Ice Box 2

- 3308Group8RateMySeat #7
  view past event photos
  user abilities
  BanaNae-Nae420/3308Group8RateMySeat

- 3308Group8RateMySeat #16
  View other users reviews/profiles
  BanaNae-Nae420/3308Group8RateMySeat

Todo 0
This item hasn't been started

In Progress 0
This is actively being worked on

Done 14
This has been completed

- 3308Group8RateMySeat #23
  forgot password page
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #1
  Responsive Navigation Bar
  Nav Bar
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #12
  Create necessary docker files
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #14
  Create Partials
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #19
  create a /get_review API route
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #8
  delete review/ edit review
  user abilities
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #15
  Create SQL tables
  BanaNae-Nae420/3308Group8RateMy...

- 3308Group8RateMySeat #2
  login to account
  login/register

[Video](#)

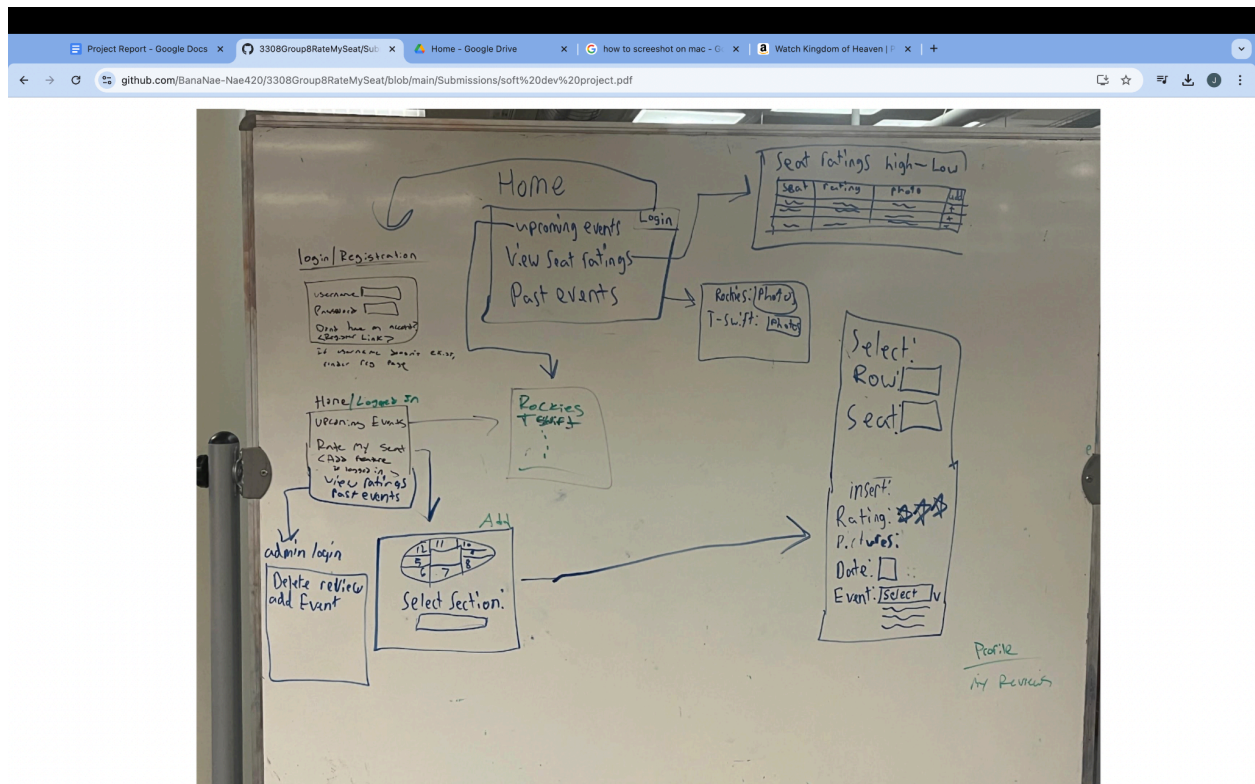
Use Case Diagram
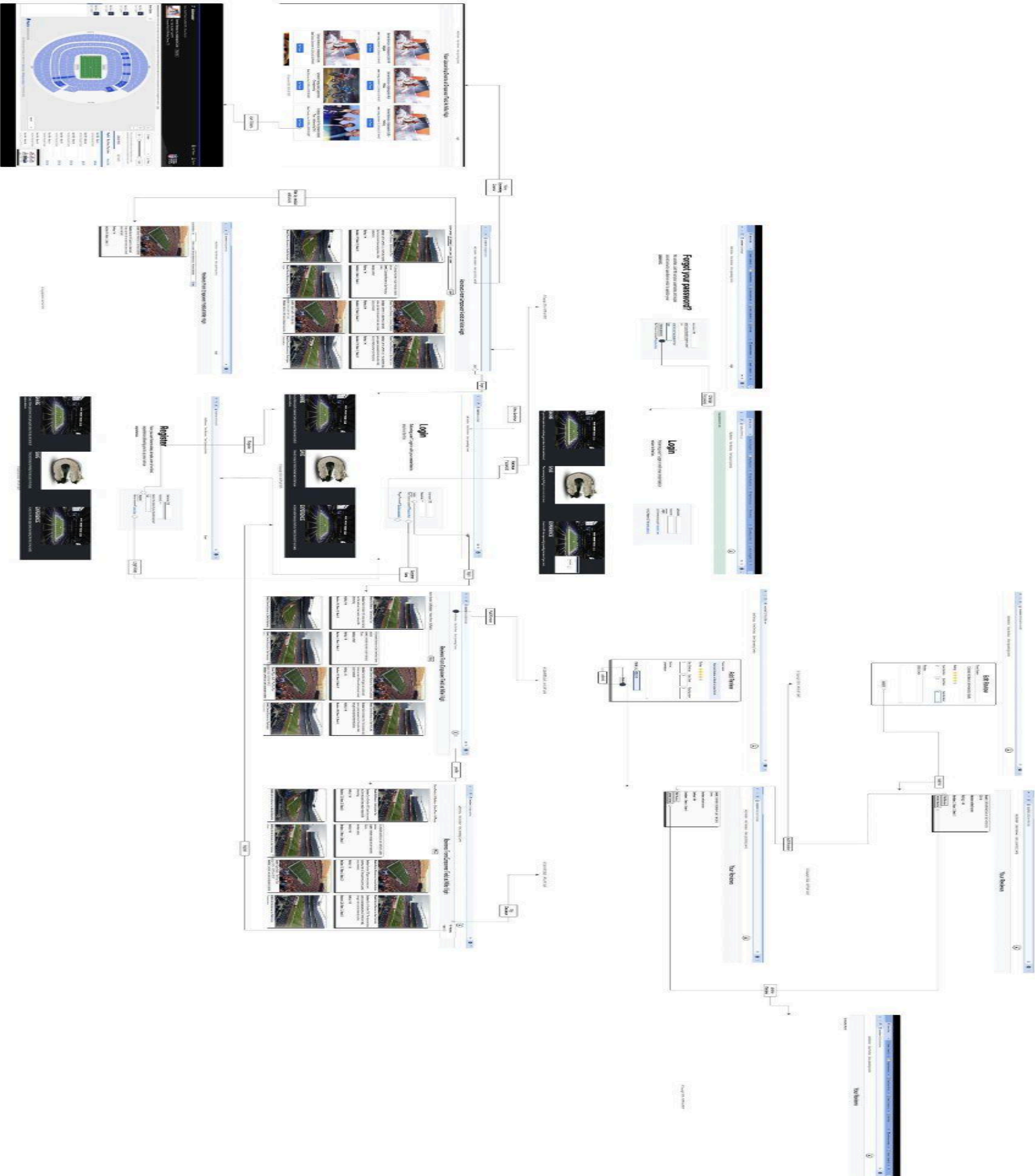
Project Report

Wireframes:

First wireframe:

Link to final wireFrame diagram on Lucid:
https://lucid.app/lucidchart/cea4b509-fc89-441d-9265-4a55d29363db/edit?viewport_loc=-4071%2C-3748%2C15063%2C8739%2C0_0&invitationId=inv_3844cf36-5881-4964-b89a-06242ec836dd

Pdf version: 📄 RateMySeatWireframe (3).pdf

Project Report

Project Report

1.

Test results from the UAT plan, we had three tests:

2. User Login
3. Add Review
4. Forgot Password

Since our website is pretty consistent and easy to understand/navigate, there were no deviations from the expected actions outlined in our UAT plan. Users who took part in testing had very consistent behavior and no problems registering, logging in, adding a review, and navigating the website in general. The test user's reasoning for their actions were always straightforward because the website made it that way.

Contributions:

Aidan Rotter: I contributed the majority of my work to the front end side of our project. I created each style for the webpages and kept them similar in layout making the entire website nice and cohesive. I also did general debugging for adding and editing reviews.

Carson Mattei: For starters, I got the coding and docker environment set up by creating the docker-compose file, package.json file, and index.js file (setting up dependencies, handlebars instances, connecting to db, and initializing session variables). I took part in creating some API's such as register, login, getReviews, getSections, getEvents, upcomingEvents, addReview, and logout. I also did the work for the external Ticketmaster API. I took part, along with Derek, in creating SQL tables in create.sql, and I also filled tables in insert.sql. Finally, I created a majority of the tests in server.spec.js and made the README file.

Jordan Law: I primarily worked on backend in the index.js file, I made the ownReviews api and the original ownReviews.hbs to populate a page with all the information from reviews a user has made when logged in. I also created the editReviews API to allow the user to edit reviews they have made, and a delete API which allows the user to delete any of their own reviews. I also added a forgot password login feature which takes the answer of 'who is your favorite teacher' which is taken on the register page and if correct, allows the user to change their password.

Derek Marraudino: I contributed a mix of backend and front end throughout the project. For the front end I designed the getReviews.hbs page and along with Carson did the java to dynamically populate the page and write the backend API. I also did the stylization for the add reviews page and worked on the navbars changes based on user authentication of being logged in or not. I also wrote some global functions and css styles in order to have star rating for the reviews rather than

Project Report

number and then the functions in order to populate the stars across the different pages. Lastly I worked with Carson to design the sql tables in create.sql and write some of the query requests for adding images inputs for going through the addReviews.hbs page and the getReviews API to populate onto the getReviews.hbs page.