“TuneSign” - Pairing users with their musical zodiac match

Group:

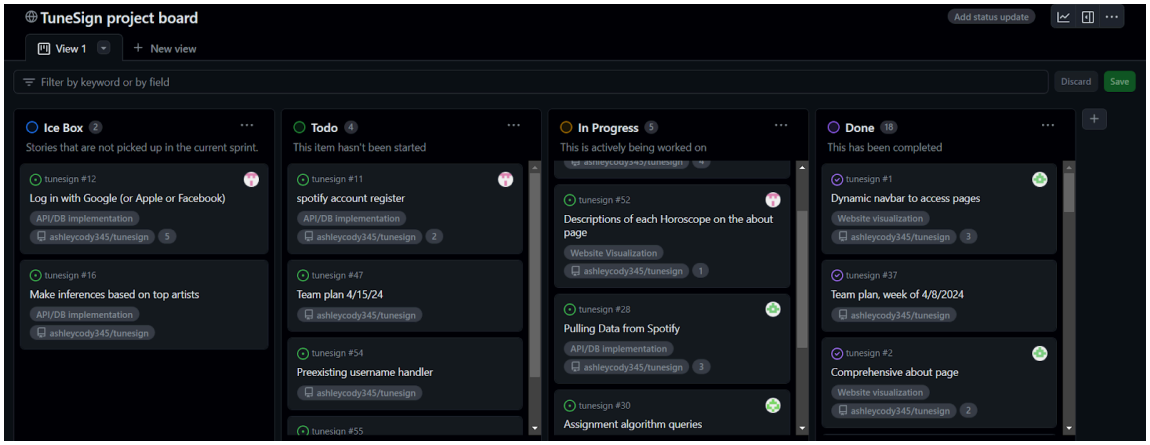
- Ashley Cody
- Langston Denning
- Lachlan Kotarski
- Jasper Shen
- Benedict Antonious

Description:

TuneSign is a web application that uses Spotify data to pair users with their musical zodiac match. Our program has a front end built using Handlebars, and our back end is powered by the Spotify API web service. Our program allows users to have their data stored in our database using PostgreSQL, and connects users to their Spotify account using a session token. This is then used to pull a user’s top tracks from their profile, and the genres of their top tracks is used to assign a zodiac sign to the user. Our database has several genres and associated zodiac signs that are related to them, and we count the number of tracks that the user listens to frequently that match each genre. Their top genres are then used to make an association with a zodiac sign, which is then presented to the user. This presentation of the zodiac sign can be exported via copying the image link, or downloading the image.

Project Tracker:

https://github.com/users/ashleycody345/projects/1



Video:

https://www.youtube.com/watch?v=3Hgr9zjyXgs

VCS:
https://github.com/ashleycody345/tunesign/tree/main

Contributions:

- Ashley:
  - Worked on database schema, testing and implementation of queries and storage techniques. Worked with the rest of the group on utilizing database storage to interact with backend, web services, etc.. Also was heavily involved in project coordination and planning.

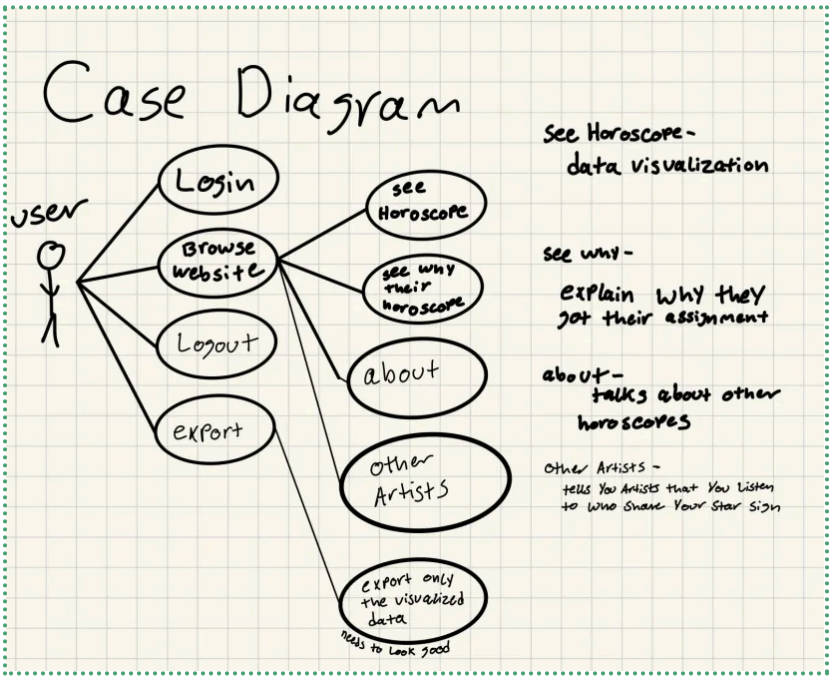
- Lachlan:
  - Lachlan worked mostly on the front end using tools like handlebars and HTML. He initiated the docker yaml file and packages associated with it, mapped the relationships between zodiacs and Spotify genres, and built the export functionality and other parts of the about page.

- Jasper:
  - Jasper dealt with implementing the Spotify API into the application, with workings mainly on the backend using JavaScript and Axios API calls. He additionally implemented the algorithm for assigning a zodiac sign through utilizing SQL and the database.

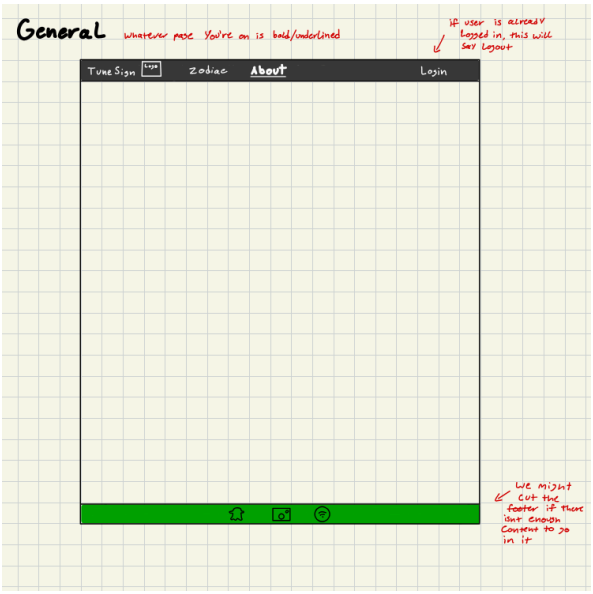
- Benedict:
  - Benedict handled implementing the Spotify API into the application, with a focus mainly on taking in data from Spotify, then parsing the data to make it usable. Additionally helped with the formatting/layout for the about page.

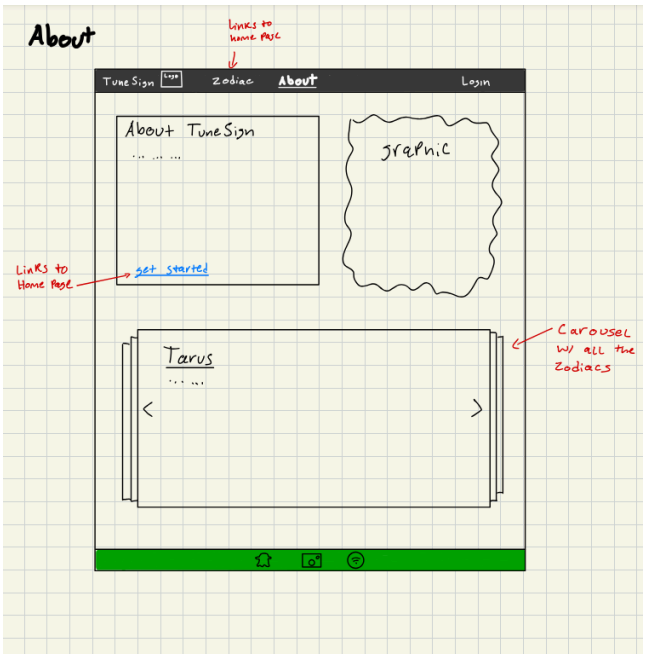
- Langston:
  - Langston bolstered progress on the front end. He was the main front end developer of the group and had the vision of UI experience. He mainly used HTML and handlebars to handle the front end. As well as creating most of the paths between pages and using our zodiac algorithm to create custom paths to images.

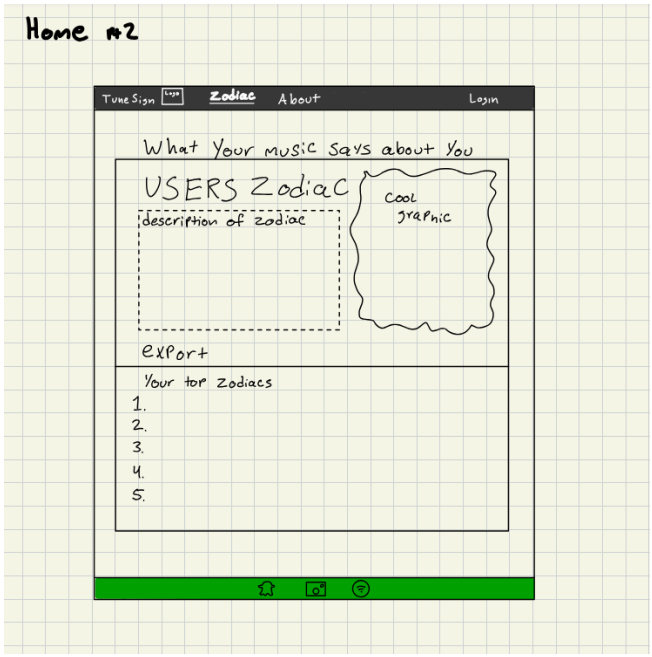
Use Case Diagram:

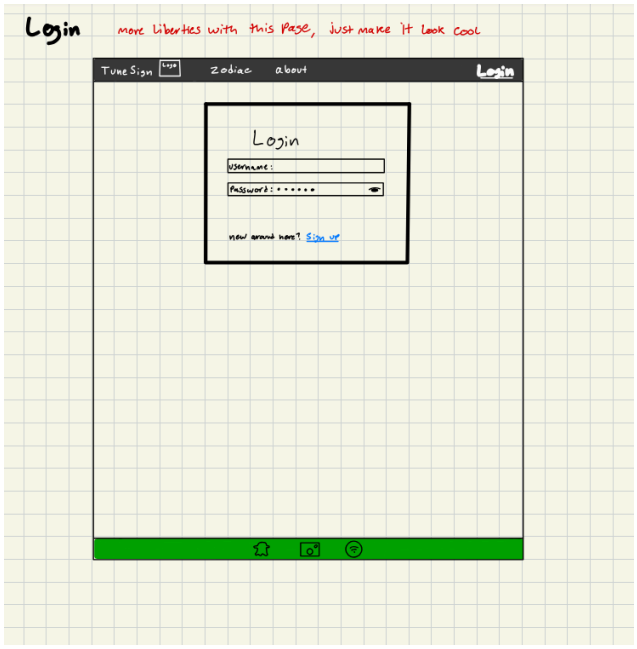


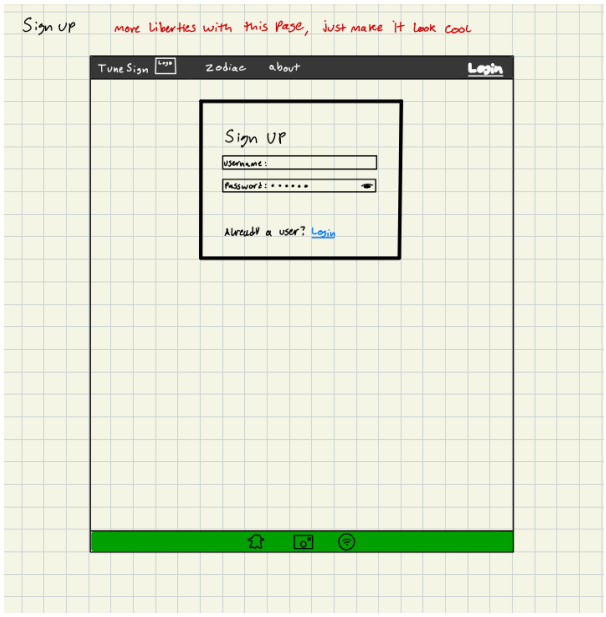
Wireframes:











Test Results:

Test Cases:

1. <u>User Registration</u> - Testing the registration process, including form validation and successful registration.

2. User Login - Testing the login process, including form validation, successful login, and error handling for incorrect credentials.
3. Website Navigation - Testing the navigation bar of the website, including automatic redirections and handling permission levels.
4. Spotify Login - Testing the ability to login using a Spotify account, including form validation and data acquisition through the home page.

Observations:

1. User Registration
    - What are the users doing?
        i. The user knew they didn't already have an account. And immediately clicked on the register button.
    - What is the user's reasoning for their actions?
        i. "I read the login page and clearly saw the line to register if I didn't have an account" -Leah yi (test user 1).
    - Is their behavior consistent with the use case?
        i. Yes, this is exactly what we wanted to happen.
    - If there is a deviation from the expected actions, what is the reason for that?
        i. The user did exactly what was expected.
    - Did you use that to make changes to your application? If so, what changes did you make?
        i. No changes needed.
2. User Login
    - What are the users doing?
        i. The user was immediately drawn to the get started button in the middle of the About page. After that when they were routed to the login page they successfully found the register button as they knew they didn't already have an account.
    - What is the user's reasoning for their actions?
        i. "It was the first thing I saw" -Jon Ciazzio (test user 2)
    - Is their behavior consistent with the use case?

        i. Yes we made a button to get started for the purpose of getting started
    - If there is a deviation from the expected actions, what is the reason for that?
        i. A little part of me was expecting that they would click on the login button in the top right corner but we made both buttons for the same purpose to cover all test cases.
    - Did you use that to make changes to your application? If so, what changes did you make?
        i. No, we want to include both buttons so users have a choice and they don't get confused.
3. <u>Website Navigation</u>
    - What are the users doing?
        i. Users are using the navbar to explore the website.
        ii. Users aren't finding the carousel on the About page
        iii. Users understand the login/registration pages
    - What is the user's reasoning for their actions?
        i. They said everything was laid out in a way that made sense, they just didn't see the left/right arrows for the carousel
    - Is their behavior consistent with the use case?
        i. For the most part yes, except for the About page
    - If there is a deviation from the expected actions, what is the reason for that?
        i. The carousel buttons are a little small, so it makes sense the users didn't see them. They are also at the bottom of the page so the user would often move from the about page before even looking at the rest of the page.
    - Did you use that to make changes to your application? If so, what changes did you make?
        i. We ran out of time at the end when we discovered this qualm but in the future, we will change this issue

4. Spotify Login
    - What are the users doing?
        i. The users are navigating to the OAUTH page for Spotify and allowing our website to access their Spotify data.
    - What is the user's reasoning for their actions?
        i. User saw the outstanding green "Connect your Spotify" button and clicked.
    - Is their behavior consistent with the use case?
        i. Yes, the user needs to login with their Spotify account to fully access our website, so the urgency for Spotify login is heightened.
    - If there is a deviation from the expected actions, what is the reason for that?
        i. No deviations from the expected actions.
    - Did you use that to make changes to your application? If so, what changes did you make?
        i. No changes necessary. Events occurred as expected.

Deployment:
http://recitation-14-team-03.eastus.cloudapp.azure.com:3000/about