



Contributors: Ashley Cody, Langston Denning, Jasper Shen, Ben Antonious, & Lachlan Kotarski

Description

For **Spotify** users, who want to learn more about their **music tastes**.

TuneSign is an entertainment service that highlights a **user's individuality**.

Unlike other Spotify statistics websites, our product will **celebrate users' uniqueness** by using data to infer their **horoscope** information.

Tools Used


Front End:  

- Handlebars: (4/5 stars)
 - template software that mitigates repetitive code.
- HTML: (4/5 stars)
 - Easy to understand front end creation software.

Back End:     

- Visual Studio Code: (5/5 stars)
 - Easily edit different file types and support for multiple languages
- NodeJS: (4/5 stars)
 - Asynchronous and has large library through npm
- Spotify API: (4/5 stars)
 - Easy to use API with large amount of available data
- Docker (4/5 stars)
 - Manage project in a single container for simplified development
- PostgreSQL (4/5 stars)
 - Manage database to work with NodeJS and Docker

Tools

Project Tracker: 

- Github Project Board (5/5 stars)
 - Ability to organize tasks into to-do, working on, and completed

VCS: 

- Github (5/5 stars)
 - Great for illustrating plans, ensures communication throughout the extent of the project and serves as a main hub for all ideas.

Testing: 

POSTMAN

- Postman: (5/5 stars)
 - used for ensuring database connectivity and API routes

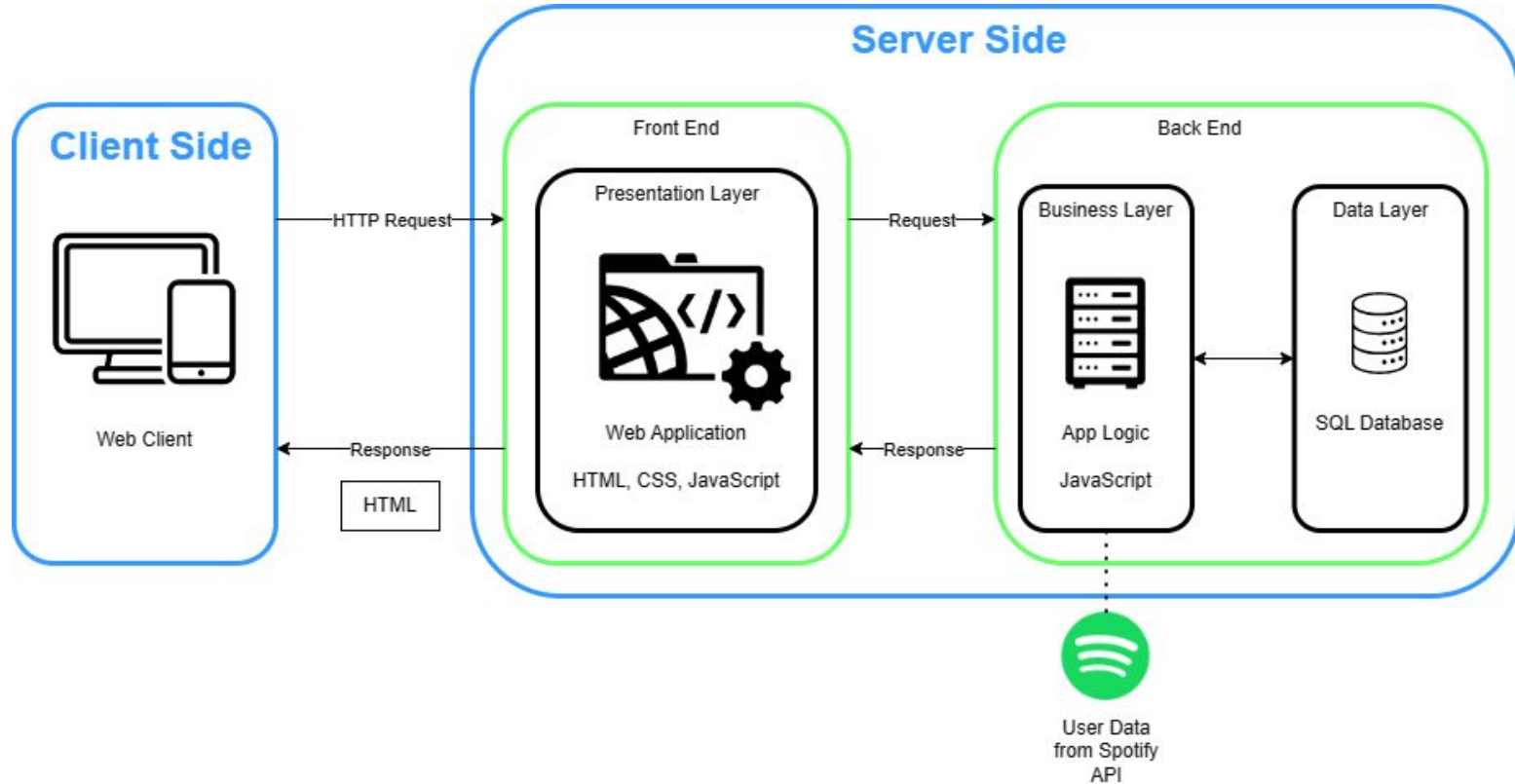
Deployment: `:://localhost`

- Localhost (5/5 stars)
 - Easy deployment on local machine for testing and updates in a very reasonable amount of time

Communication & Methodologies

- Agile
 - Peer code Reviews
 - Team meetings used for communication w/ a document for tracking progress
 - Short term and long term goals
 - Urgent to-do's
 - Checked in with each other
 - Tasks assigned through project board
 - Discord Server and email for communication
-

Overall Architecture Diagram



Challenges

- Retrieving and organizing Spotify data
 - Understanding organization of data
 - Extracting needed data from pulled data
 - Took time to understand data
- Organizing genres
 - Over 6k individual categories
 - Filtered into general genres with keywords
 - Ex: **Alternative** has subgenres of Alternative Americana, Alternative Country, Alternative Dance, Alternative Emo, etc.
- Developing flow to access data
 - Need to know what user is logged into
 - Create permissions for each page



Future Scope/Ice Box

- Login with Google or Facebook
- Using user information of their top artists, make inferences of what their artists' zodiacs would be
- Music recommendations based on a user's zodiac





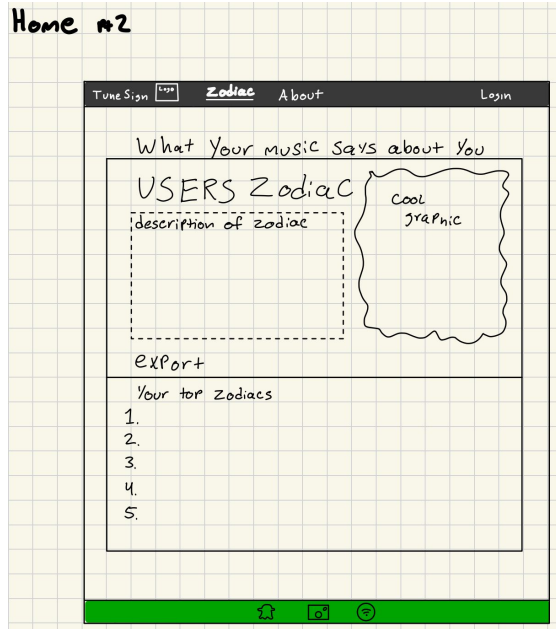
Demo

Questions?

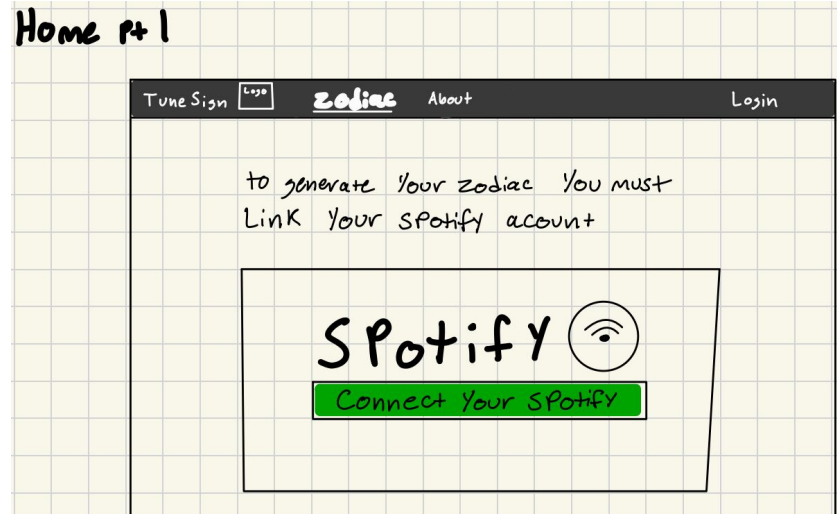
Wireframes

Front End

Home #2

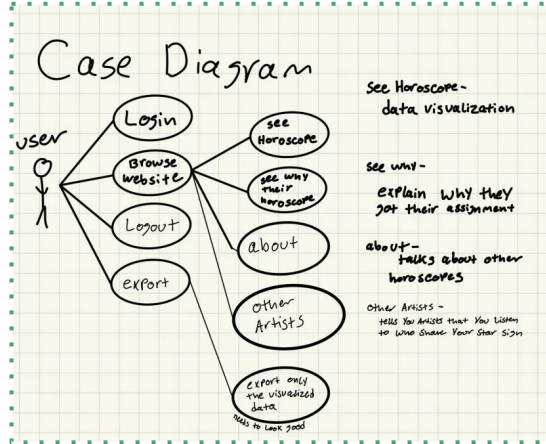


Home #1



Diagrams

Cases



Database

