Star-Crossed Backend README

Project Overview

Star-Crossed is a zodiac dating app that matches users based on compatibility of their sun sign and gender preference. A user is required to include their DOB (to find their sun sign on the backend), as well as gender and gender preference for filtering results. Two algorithms on the backend to facilitate matches — one finds a sun sign's compatible signs based on API data. The other creates matches with that compatibility, then cross-references against both matched users gender and gender preference. In their profile, a user can view all matches, click to view a match's profile, and decline or accept a match. Declining will nix the match for both users obviously, but accepting will generate a status of either "pending" or "accepted", contingent on whether or not the matched user has also accepted. If a user edits their details (gender preference, for example) matches will be regenerated, but declined matches will never be re-created or re-added. A user can of course view all of their accepted matches and soon (working stages), be able chat with them as well. 🔮💌

Technologies Used

React, Redux, Ruby on Rails, Zodiacal API, HTML, CSS, JSON, Materialize React CDN & GoogleFonts, WebSockets/ActionCable, ActiveModel Serializers & other Ruby gems

Getting Started

Prerequisites

To use, clone down this repo and open with your preferred text editor. This project uses Ruby on Rails, so you'll need to make sure you have both installed (if you don't, visit this link for instructions: Digital Ocean: Install Ruby and Set Up Local Environment), as well as PostgreSQL as the database resource.

Installing

The project gemfile specifies Ruby 2.6.0, so you may need to install that version if you don't have it. Open terminal and in the root of the project directory run:

```
bundle install
```

or simply

```
bundle
```

to install gem dependencies. Once that completes successfully, run:

```
rake db: create && rake db:migrate
```

to facilitate backend set up. You can then:

```
rails c
```

and jump into the console to test out that tables and relationships were established correctly, that class instances can be created and related successfully, etc. Zodiac data is gathered from https://zodiacal.herokuapp.com, which does not require an API key so you're all set to run:

```
rails db:seed
```

to initialize sun signs and generate compatibilities. If the development database is running correctly, then the seed data should be successfully created and final step is to test the server. To activate it, run:

```
rails s
```

and once the terminal says it's running, navigate to http://localhost:3000. If there's a "Yay! You're on Rails!" welcome page, then you can visit http://localhost:3000/api/v1/suns or api/v1/compatibilities or api/v1/users (if you seeded any) to check that all data is being rendered properly. If so, it's time for the frontend (link below). 🤙🏼

Frontend Link

star-crossed--frontend

Demo Video

star-crossed.mov

Authors

Hamilton Hudson

» *ehamiltonhudson@gmail.com*
↳ *LinkedIn*: https://www.linkedin.com/in/hamiltonhudson
↳ *Website*: https://hamiltonhudson.myportfolio.com
↳ *Blog*: https://ehhudson.wordpress.com
↳ *Twitter*: https://twitter.com/HamiltonHudson

License