Motivation

We firmly advocate for the pivotal role of technology in revolutionizing healthcare services. In the state of California, which faces a pronounced challenge with high rates of sexually transmitted diseases (STDs), there exists a pressing demand for enhanced accessibility to accurate information and support for affected individuals. Recognizing this imperative, STDHelp is committed to harnessing diverse data reservoirs to develop an inclusive platform. Our objective is to empower individuals grappling with STDs by furnishing them with comprehensive resources and assistance, thus bridging the gap between information and action in addressing this critical public health concern.

User stories

#34 - Description of STDs - !19
#35 - give information on the price/cost of treatments - !21 - Closed by manually adding insurance information but is also impossible for the future:
- Many medical clinics don't even list their accepted insurances on their websites.
- The data from the API does not provide insurance information.
- Prices for medical treatment wildly vary even between insurances
- Medical centers don't comprehensively list treatments on websites nor is it API information.

#36 - testimonies - !22 - Added samples to home screen
#37 - attribute "zip code" - !14 - Condensed into address
#38 - Implement filtering by gender - !12 - Added with first information
#50 - Media of Testimonies. - !37 On the landing/splash added five testimonies we found from various sources.
#51 - Logo - !33 Added a logo we created to the navbar which is displayed on the landing and all other pages.
#52 - Fix about page - !35 Formatted about page adding more style changes and better overall formatting and links.
#53 - Add more images - !35 Added images for all tools we used during development to the about pages.
#54 - More similarities in Model Pages - !39 All model pages now have a title and clickable cards. Also formatted more similarly.

Models

Model 1: Counties
This model shows the statistics of the number of cases as well as the rates of different types of STDs, such as Primary and Secondary Syphilis, Gonorrhea, Chlamydia, Total Early Syphilis and

more in the year 2021. The model page shows a smaller selection of these statistics, as well as the county seat and the population of the county, with each card describing a county, and each card has a button that leads to the instance pages for the county. The instance pages contained the aforementioned information as well as a map from Google Maps embedded as well as the flag of that particular county. Below that are cards that should link to instances of healthcare locations that are available in that county from the locator model, but for this phase it is just linking to the current 3 instances of locations that are available. The cards in the model page will later be sortable in the number of cases for each STD in the preview, as well as the population of each county in ascending or descending order as well as the default alphabetical order.

Model 2: Prevalence

Model two displays the prevalence of three different types of STDs in the state of California. The dataset we decided to use contained information on Chlamydia, Gonorrhea, and Syphilis. The cards displayed on this model pages are catergorized by three main factors: year, gender, and disease. Each card displays this image along with the number of cases and the rate of each disease. We decided to differentiate male and female entries for the year because that is how the dataset is split and one of the user stories asked for entries to be sortable by gender. Upon clicking on an instance card the instance page opens up. Here the population for that gender in California is displayed. The total cases, rate of cases, and percent of patients tested for Chlamydia, Gonorrhea, or Syphilis is displayed. There is also a link which navigates to the CDC's page on each of said STDs. Underneath that a generic infographic is displayed and at the bottom of the page related content is shown with a link to navigate to said pages.

Model 3: Locator

Model that uses https://locator.hiv.gov/map to find locations and information about the locator. Currently pending access to API because they have not responded to our email request. The main model page has a grid of cards, each with an image and several fields describing the locator. These cards can be clicked on to navigate to an instance page which contains additional information. The model card grid currently holds attributes to: Name of location, address, phone number, website, and Services offered. These are planned to be made filterable at a later date. Information for the cards are currently stored in the Locator file, but the information once the instance is clicked on is stored in the JSON file. The information of the cards is planned on being transferred to the JSON file as well at a later date, once we can scrape it. The instances additionally contain an about section, as well as what insurance providers are accepted.

Tools

Web framework - React
CSS framework - React-bootstrap
DNS - Namecheap
Hosting - AWS Amplify

VCS - Gitlab
CI/CD - Gitlab Pipelines
Testing- Jest, Selenium
Backend API - Hono
Database - Postgres
Backend Hosting - Amazon EC2
Database Hosting - Amazon RDS

Hosting

Hosted on stdhelp.site
Domain bought through Namecheap and the website is hosted on AWS Amplify.

RESTful API

https://www.postman.com/bmaloneut/workspace/stdhelp/api/c441db53-30a6-4e53-91ac-484269
7ffc21/definition/c028b8d1-a096-459d-9f20-ba07177da61e?version=82e7a787-163d-43e6-a62b
-d946216748e2&view=documentation

The current API concept has two types of GETs: explicit and categorical. The explicit GET takes a model (category) and a unique identifier (instance). If the instance given exists, a JSON object is sent with only the information from that specific instance included. At the moment, the JSON object only includes base information from the instance, meaning no media, links to instances, or other special information is included. The Postman link has the schema documented clearly. Any bad input will result in a 404.

The categorical GET takes no input but gives a full list of ALL instance information for a given model. It is structured as an array of instance objects specific to the model.