Patrick Canny
Ellis Springe
Liam Ormiston

Landscrape Sprint Log

Overview

The primary goal of the Landscrape web application is to provide users with information on the best things around them. What differentiates our application from competitors is support for a variety of search queries spanning several popular searching sites in order to assist users in developing a list of top attractions in the area. This allows for a simple application that can be easily accessed and utilized by many users.

Project Planning

Development of our application will adhere to the agile methodology and the principles of iterative design. After our initial prototyping sprint (2 weeks) we will focus on one-week sprints with the goal of adding a single feature with each sprint. We will utilize both a gantt chart alongside of the Busybot Slack integration to manage task delegation and progress tracking.

Sprint 1 | Prototyping | 10/22-11/5

Prototype Goals

The prototype of our application will focus on baseline functionality, that is focusing on a single query and scraping results from a single site (Yelp).

We chose to use the Flask microframework for development of our application. Flask is a bare-bones web development framework based in Python. Flask allows for a high level of flexibility and customization for small-scale applications. We knew that our web-scraper would most easily be implemented with Python as an engine, which is a primary reason that we went with Flask. We also wanted the option to use ReactJS for our frontend, which is not as easily supported in a more full-featured Python framework such as Django. Though our prototype doesn't use React for the frontend, we are hoping to add this in a future iteration.

Prototype Features

Landing Page: The landing page of the application is simple, and allows the user to directly enter and search quickly and easily.

Search Page: Barebones search page with a string entry field. The query entered in this field is then thrown to the Python backend, which handles all the heavy lifting.

About Page: Documentation Landing Page

Search Functionality: The search function is implemented in Python, and queries the Yelp API in order to return results and display them to the frontend.
Navbar: We learned about HTML templating in development of this application, and utilized templating to allow for a persistent navbar layout on each page.