Project Proposal
Sight See

Tourist Map

ECE: 452
Spring 2025

Team

Rayyan Barbhuiya

Abhinav Bollu

Aaryan Handa

Sean Johnson

Saahil Patel

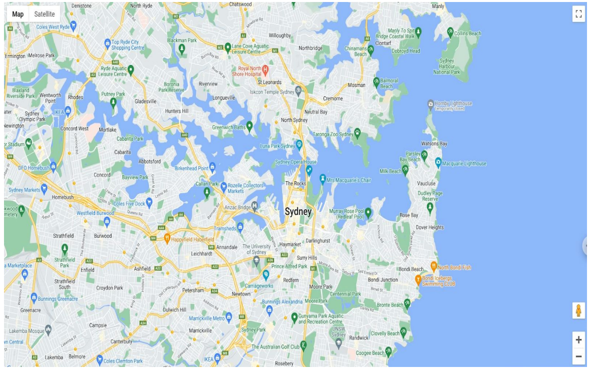
Nirshanth Kiritharan

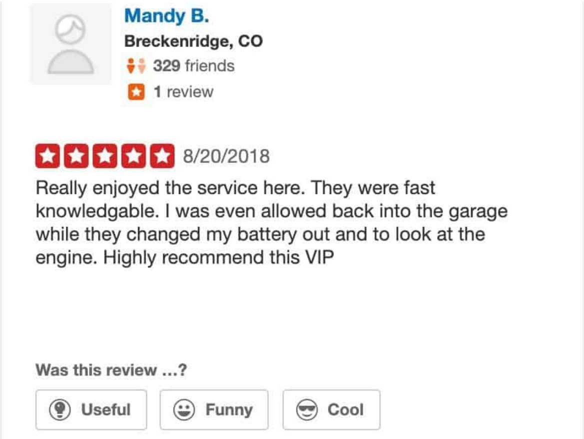
Sachit Nigam

Adiel Torres

Project Description - Map Widget

- Displays interactive map that users can intuitively use to get information useful to a visitor
  - Highest rated tourist attractions in the area
  - Live weather updates
  - Local restaurants
- Displays helpful location reviews from around the area
  - Rate reviews by confidence and filter bot reviews
- Search bar with interactive filters
  - Filter by budget, average review rating or distance





Mandy B.
Breckenridge, CO
329 friends
1 review

★★★★★ 8/20/2018

Really enjoyed the service here. They were fast knowledgable. I was even allowed back into the garage while they changed my battery out and to look at the engine. Highly recommend this VIP

Was this review ...?

Useful Funny Cool

Requirements

- API Implementation (Earth Engine, Google Places API, OpenWeather API)
- CSS/HTML Web Design
- Javascript for interactables
- Library (quality, efficiency)
- Testing/Debugging
- Web Server (Github Pages)
- Github repository

Use Cases

- Users can use the interactable map to find restaurants and attractions near them.
- Users can plan a trip by finding businesses, attractions and locations of interest in advance with the help of search filters.
- Users can add reviews of locations of interest as feedback that could be incorporated into the system.
- Users can report weather and other conditions at attraction sites.

Business Policies & System Rules

- Review Moderation → Reviews flagged as "low-confidence" by the bot are hidden
- API Usage Rules → Cached data is used if API requests fail or exceed limits
- Weather Reporting Rules → Users must confirm their location before submitting reports
- Search Filter Enforcement → Confidence rating thresholds are applied to avoid spam
- User Roles → Visitors vs. Registered users (who can write reviews and submit reports)

Sub-teams

- Frontend Team (UI/UX & Interactions) - Aaryan, Saahil
- Backend Team (Data Handling & API Integration)
  - Data Handling: Sean, Rayyan
  - API Integration: Adiel, Abhinav
- Testing & Optimization Team - Sachit, Nirshanth

Frontend Rough Plan

1. Building a User friendly Interface
   - Design a clean and easy-to-navigate layout using HTML, CSS, and JavaScript.
   - Add features like a search bar, filters, and interactive buttons to enhance usability.

2. Integrating Maps and Location Features
   - Use Google Maps API to display attractions, routes, and user locations.
   - Make sure users can zoom, pan, and click on locations for more details.

3. Displaying Attractions and weather updates.
   - Show tourist spots, weather forecasts, and ratings in an engaging way.
   - Pull real-time data from APIs to keep information fresh and accurate.

Data Management & Storage

- Attractions Collection → Place name, description, category, user ratings, location data
- Weather Reports Collection → User reported weather conditions, timestamps, API fetched data for caching
- User Reviews Collection → User submitted reviews, confidence rating, filters (budget, distance, etc.)
- Guest Access Policy → Visitors can view attractions, but reviews & reports require login

System Failure Handling

- Google Places API Down → Show a "No data available" message & suggest popular attractions
- Weather API Fails → Display cached weather data or general climate trends
- User Input Invalid → Restrict blank queries, suggest autofill city names
- Spam Reviews → Reviews below a set confidence level are filtered out
- Map Service Issues → Default to a text based list of attractions if the map fails to load