Final Report for Geographically

UNSW Sydney 2019T1 SENG2021 - TeamA

Our Purpose and The Problem

Purpose of system:
Geographically is a centralised hub that revolves around providing a user-friendly, holistic experience when planning a trip around Australia. The application contains a wide variety of functions that alleviate the pressure of deciding on a destination, such as the recommendations feature on the dashboard - displaying tourist attractions nearby the user, as well as more popular ones within road trip distance. During the trip planning stage, additional information about the locations is provided, such as the weather to allow for more efficient planning. The Users are also able to share their experiences and trips through the blog feature, allowing them to highlight underrated attractions, so other users that stumble upon the blog can learn about these locations.

Problems Addressed:
Many trip planning applications currently have limited functionality, and only serve one main purpose - which is the planning of the trip itself. This is where Geographically has an edge over the competition, as there is a lot of integrated additional functionality, highlighted below:
Overview

1) In the process of planning a trip, when deciding the location and date and time, it is inconvenient to constantly check the weather for that particular instance.
2) Users passionate about travel and sharing their experiences now have access to an application that allows them to to do so, integrated with the main functionality (trip planning) to minimise the need to use multiple sites. The blog feature also allows other users to
3) Manually picking out a playlist specific to the trip can be time consuming, so the process is automated based on the destination.

Problem Statements

1) Maps do not tell users information such as the weather on a chosen location in real time.
2) There is no application that combines a map application with a blog that allows people to share their trips and experiences.
3) There is no trip planning application that combines a specialised music system according to your location.

User Stories

Priorities in MOSCOW notation
User-story point = 2 hours

Feature: Plan a trip

As a user

So that I can pay a visit to a location
I want to find a route to get to the location

Scenario: Plan a trip by searching location in the search bar
Given I have chosen a location I want to travel to on the 'Plan' page
When I have type in the chosen location I want to go to
Then I am given the shortest route to get to the location on a map from my current location

Scenario: Plan a trip by being given a recommendation
Given I have chosen a particular location on the 'Dashboard' page
When I have clicked on the location
Then I am redirected to the 'Plan' page and given the shortest route to get to the location on a map from my current location

Priority: Must
Time estimate: 3 story points

Feature: Get recommended tourist attractions

As a user
So that I can easily pick a tourist attraction I would like to go to
I want to be able to find tourist attractions I can go to

Scenario: Receive recommendations on 'Dashboard'
Given I want to get recommendations on where to go
When I am on the 'home' page
Then I am able to see recommendation of tourist attractions under the 'Discover' section or the 'Plan a trip' section along with how it would take to get to that location from home

Priority: Could
Time estimate: 1 story point

Feature: Check weather of chosen location

As a user
So that I can decide when to go to a certain location
I want to check the weather of a location

Scenario: Check weather of chosen location
Given I am on 'Plan a Trip'
When I have chosen the destination I want to go
Then I can view the weather for the location

Priority: Must
Time estimate: 1 story point

Feature: Save a trip

As a user
So that I can possibly revisit the details of the trip plan
I want to be able to save the details of a trip I have planned

Scenario: Save a trip
Given that I am on the 'Plan' page
When I click on save trip
Then I am able to to save various details of a trip I have taken, including the route, location weather and playlist

Priority: Should
Time estimate: 1 story point

Feature: View a saved trip plan

As a user
So that I can read my trip plan
I want to be able to view the details of a trip I have planned

Scenario: View a saved trip plan
Given that **I** am on the 'My Profile' page
When I click on the 'view trip details' button
Then I am able to see the various details of a trip **I** have taken, including the route, location weather and playlist on the 'Plan' page

Priority: Should
Time estimate: 2 story points

Feature: Make a blog post

As a user
So that I can let other users know whether or not my trip was enjoyable
I want to be able to share the details of my trip with other users

Scenario: Make a blog post
Given I am on my 'My Blog'
When I have written a blog post under 'Add new post'
And I click 'save and post'
Then I can share the post with other users on my blog

Priority: Must
Time estimate: 2 story points

Feature: Delete a blog post

As a user
So that the blog post is no longer shared with other users online
I want to be able to delete a blog post

Scenario: Delete a blog post from the profile page
Given I am on my profile page
When I click on trash can icon on one of my blog posts
Then I can delete the chosen blog post

Scenario: Delete a blog post from the 'My Blog' page
Given I am on 'My Blog'
When I click on the trash can icon on one of my blog posts
Then I can delete the chosen blog post

Priority: Should
Time estimate: 1 story point

Feature: Edit a blog post

As a user
So that I can make amendments to what the blog post will share with other users online

I want to be able to edit a blog post

Scenario: Edit a blog post from the 'My Blog' page
Given I am on 'My Blog'
When I click on the pencil icon on one of my blog posts
Then I can edit the chosen blog post

Priority: Should
Time estimate: 2 story points

Feature: View my blog post

As a user
So that I can read what I published on my blog post.
I want to be able to view my blog post

Scenario: View a blog post from the 'My Blog' page
Given I am on my 'My Blog' page
When I click on the 'view blog post' button of a particular blog post I have published
Then I can view the chosen blog post on a new page

Scenario: View a blog post from the 'My Profile' page
Given I am on my 'My Profile' page
When I click on the 'view blog post' button of a particular blog post I have published
Then I can view the chosen blog post

Priority: Must
Time estimate: 1 story point

Feature: Rate a location

As a user
So that I can let other users know the quality of those attractions
I want to be able to rate the attractions

Scenario: Rate a location
Given I am on the 'my blog' page
When I write a blog via click the 'add new blog' button
Then I am able to add type down the location I want to rate and provide that rating to the chosen location

Priority: Could
Time estimate: 2 story points

Feature: View blog post options (by title, most recent date, oldest date)

As a user
So that I can find the blog post I'm looking for easier
I want to be able to view my blog posts in a particular order

Scenario: View blog post by blog post title
Given I am on the 'my blog' page
When I click on the 'by blog post title' button
Then I am able to view my blog posts in alphabetical order by blog post title

Scenario: View blog post by most recent date
Given I am on the 'my blog' page

When I click on the ‘by most recent date’ button
Then I am able to view my blog posts in order of most recent date

Scenario: View blog post by oldest date
Given I am on the ‘my blog’ page
When I click on the ‘by oldest date’ button
Then I am able to view my blog posts in order of oldest date

Priority: Could
Time estimate: 1 story point

Feature: Search blog posts (case-insensitive)

As a user
So that I can find the blog post I’m looking for easier
I want to be able to search for my blog posts by their title

Scenario: Search blog posts
Given I am on the ‘my blog’ page
When I type in the search bar
Then I am able to view the blog posts which match the what I typed in the search bar.

Priority: Could
Time estimate: 1 story point

Feature: Automatically generate road-trip playlist

As a user
So that I can get in the mood to enjoy the location by listening to the playlist
I want to be able to generate a recommended playlist based on the attraction

Scenario: Make a playlist
Given I am the ‘Plan’ page
When I click under ‘My Trip Playlist’ where the music icon is found
Then I am linked to a page where I can link a playlist for that trip
When I click on “Generate a Geographically planned playlist”
Then I should see a playlist generated automatically for my trip

Priority: Must
Time estimate: 2 story points

Feature: Find recommended road-trip playlists

As a user
So that I can get in the mood to enjoy the location by listening to the playlist
I want to see some public playlists that are related to my location

Scenario: Make a playlist
Given I am the ‘Plan’ page
When I click under ‘My Trip Playlist’ where the music icon is found
Then I should see a list of recommended public playlists specifically for my trip

Priority: Must
Time estimate: 2 story points

Feature: Login

As a user
So that I can access the features of the website, such as my saved trips and trip playlist
I want to be able to login to my account

Scenario: Login
Given I am on the Home Page
When I click on the ‘Login’ button’ at the header
Then I fill in my login details in the give white spaces from the login pop up (email and password)
And I click on ‘Login’
Then I am able to login to my account and access the features of the website

Priority: Must
Time estimate: 1 story point

Feature: Log out

As a user
So that I can leave my website
I want to be able to log out of my account

Scenario: Log out
Given I am logged in on any page (except for the Home Page)
When I click the arrow icon from the side bar where ‘logout’ is read on my toolbar
Then I am able to log out of my account

Priority: Must
Time estimate: 1 story point

Feature: Sign up

As a user
So that I can login and access the features of the website, such as my saved trips and trip playlist
I want to be able to sign up for a new account

Scenario: Sign up
Given I am on the Home Page
When I click on the ‘Sign up’ button at the header
Then I fill in my details in the given white spaces
And I click the ‘Sign up’ button
Then I am able to create an account and be read as logged in such that I am able to access features of the website

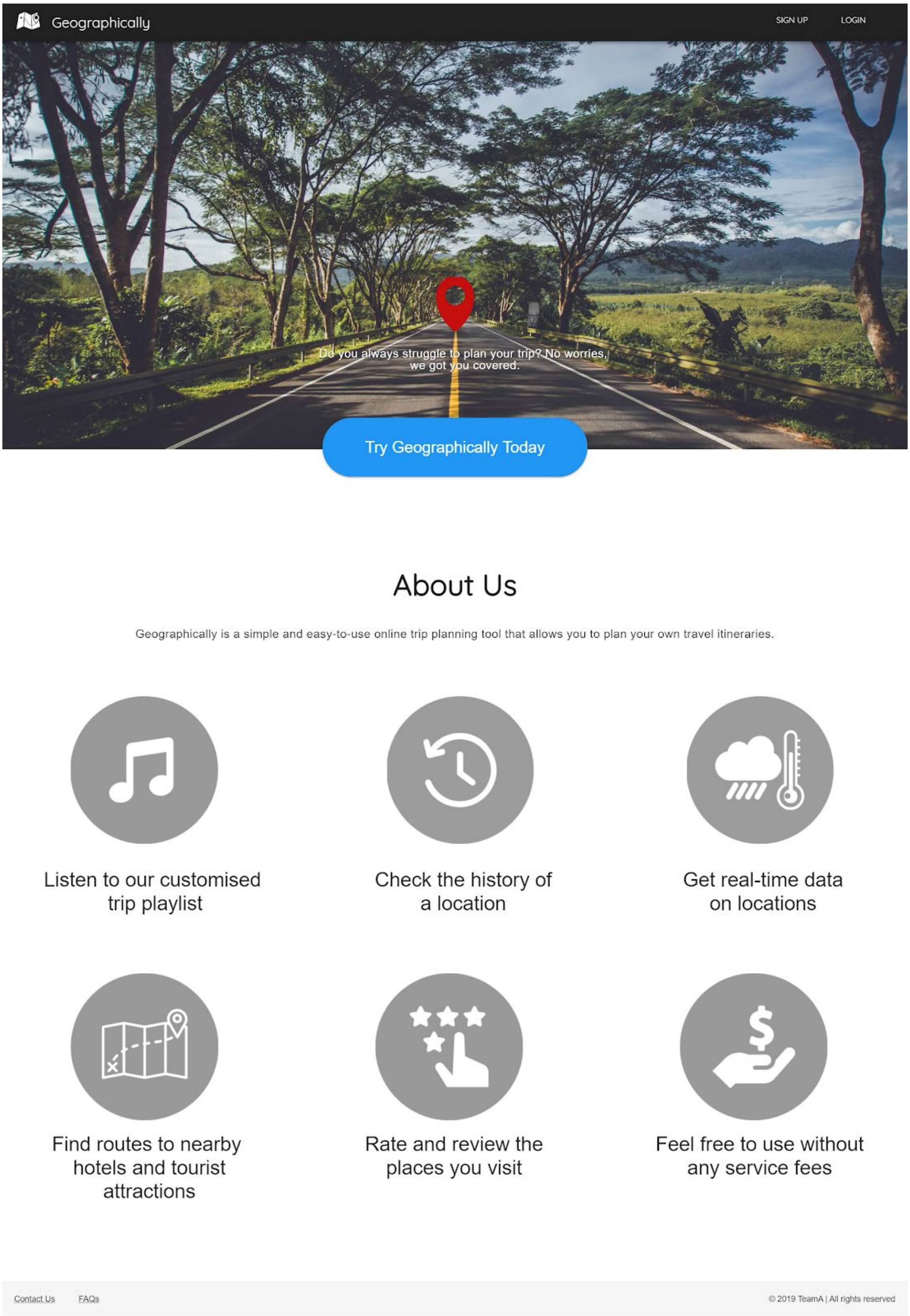
Priority: Must
Time estimate: 2 story points

Interface Design

Our website was set up in a way that makes it easy to navigate through each features due to the consistent sidebar and footer on every page. The sidebar contains a link to every page so the user is never lost. Each feature can work independently on its own page but also contains links to other features, passing the same information to support the holistic feature of our site so that users can plan their trip efficiently.

Home Page

This is the first page that the users would encounter when they go to our page. It provides a simple Login and Signup button at the top of the page. As the user scrolls down, they can see the different features of the site. The “Try Geographically Today” also brings up the Signup card.



Do you always struggle to plan your trip? No worries, we got you covered.

About Us

Geographically is a simple and easy-to-use online trip planning tool that allows you to plan your own travel itineraries.

Listen to our customised trip playlist

Check the history of a location

Get real-time data on locations

Find routes to nearby hotels and tourist attractions

Rate and review the places you visit

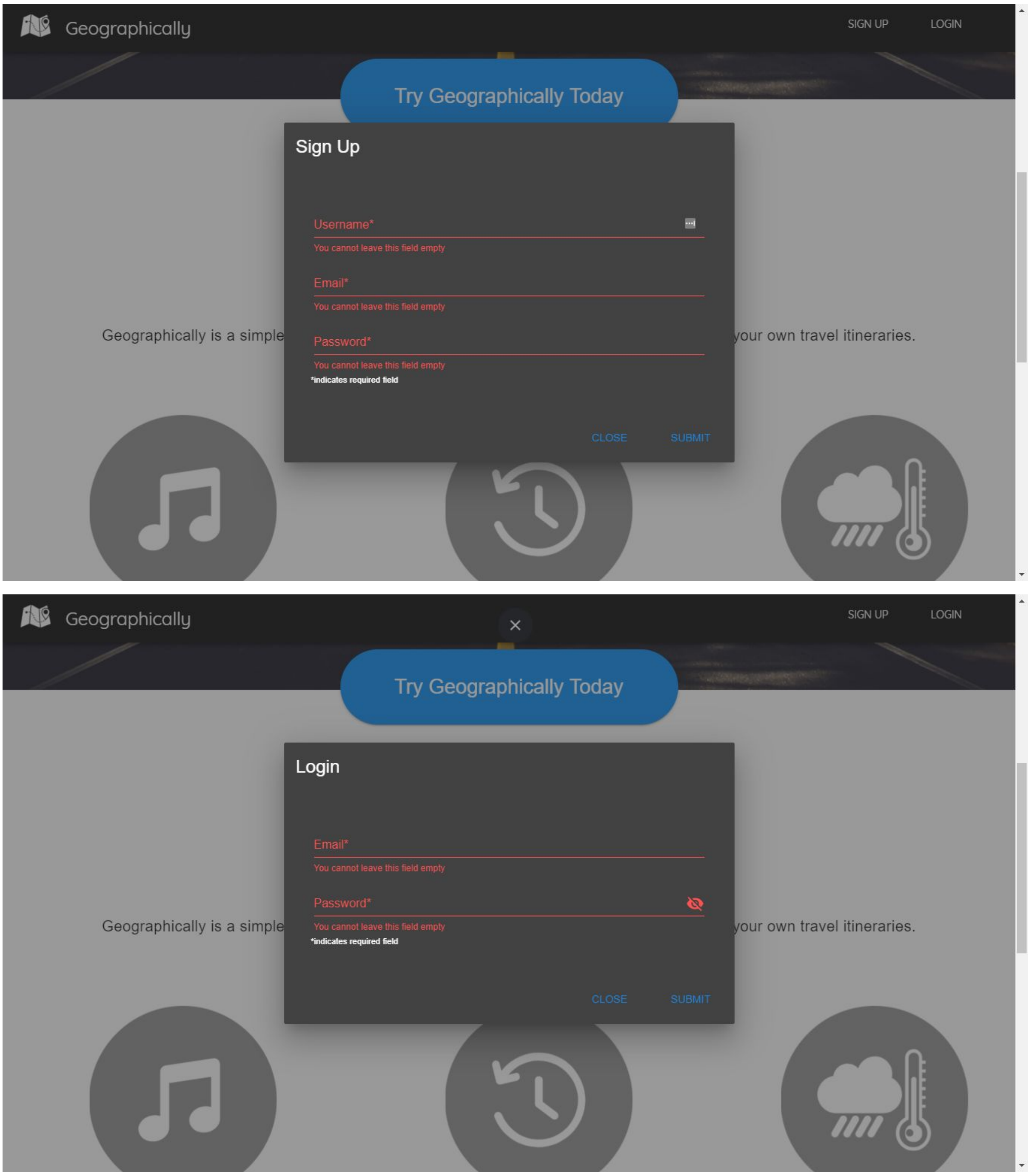
Feel free to use without any service fees

© 2019 TeamA | All rights reserved

Home Page

SignUp / Login

These two cards pop up from the Home Page allowing users to sign up and login. Once logged in, they would be access all Geographically features. There is extensive data validation built in so that user does not leave a field empty, provide an invalid email address, use an already used email to sign up etc.



Sign Up

Username*

You cannot leave this field empty

Email*

You cannot leave this field empty

Password*

You cannot leave this field empty

*indicates required field

Login

Email*

You cannot leave this field empty

Password*

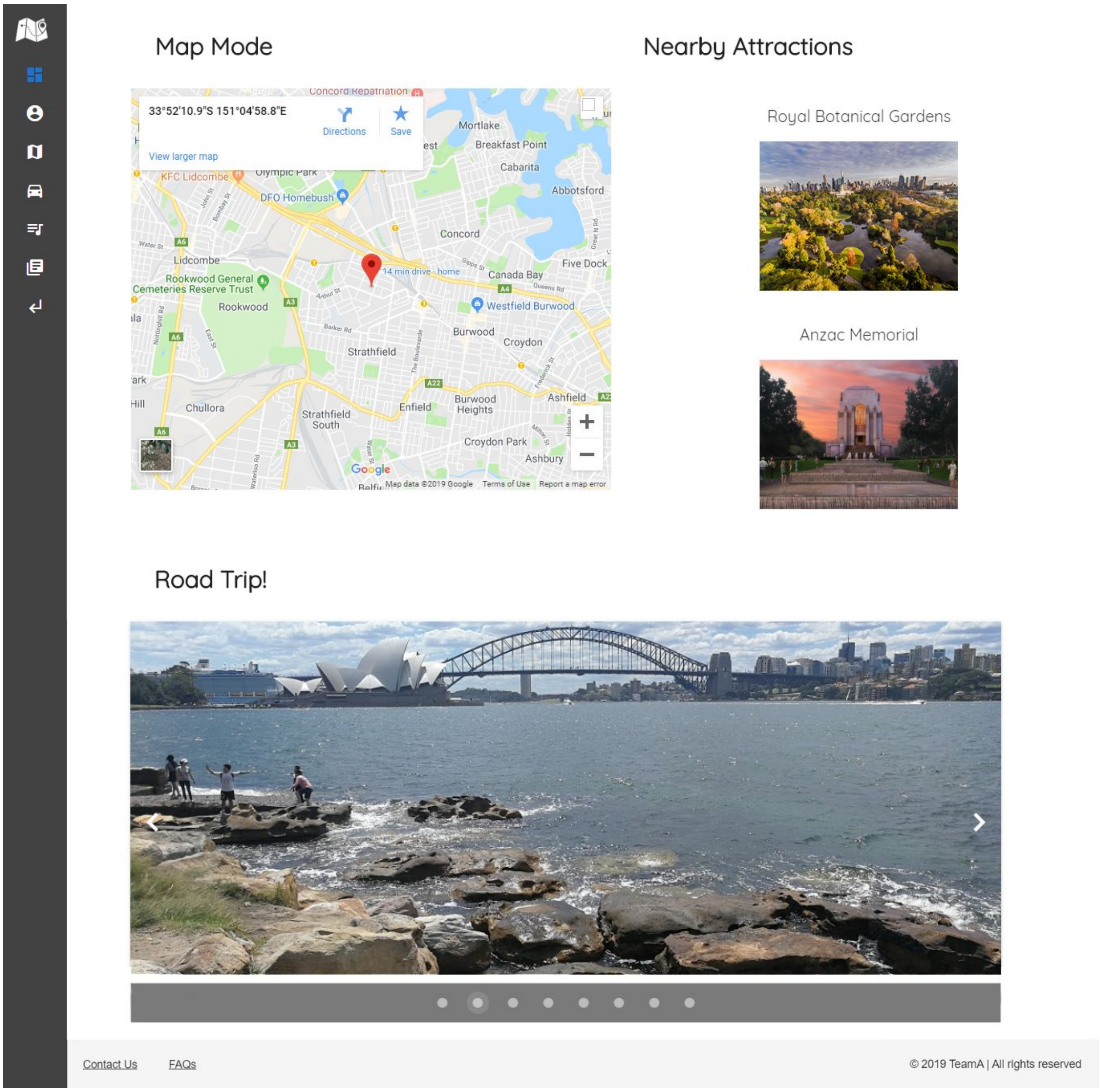
You cannot leave this field empty

*indicates required field

Sign in and Login

Dashboard

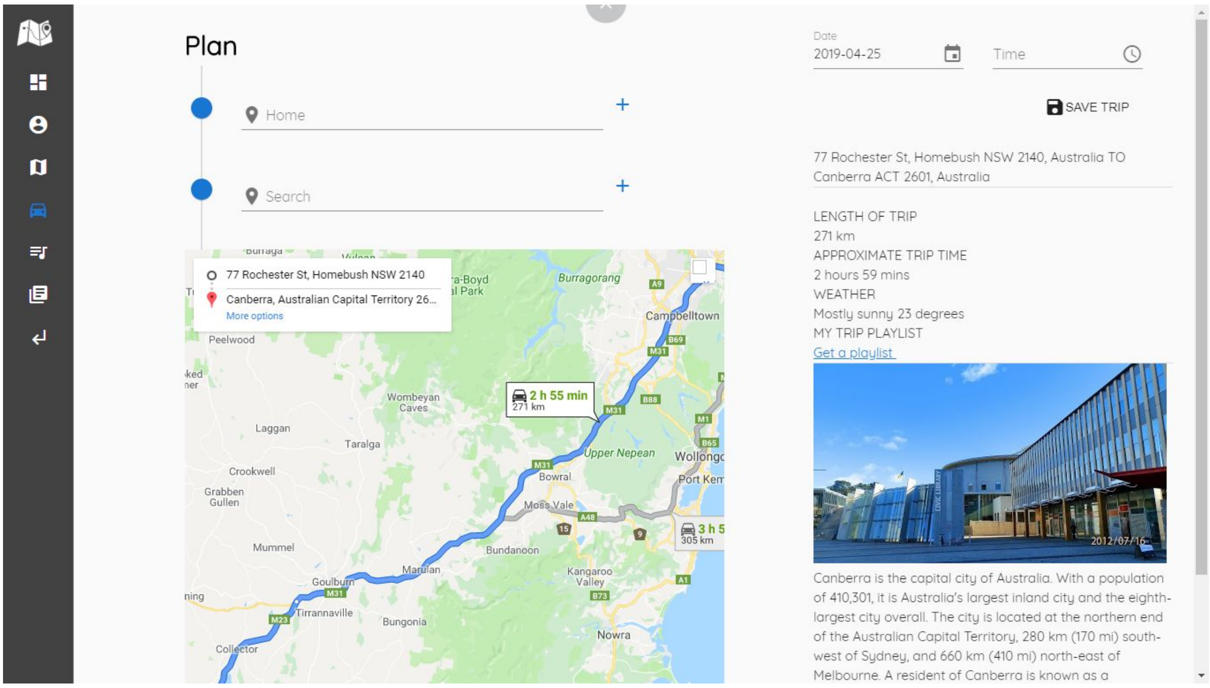
The Dashboard is the first page the user sees after they log in. It contains a small map to show their current location. On the right of the map, it shows a selection of closer tourist attractions whilst the ones at the bottom show further locations that the user can plan a road trip to. Clicking any of the attractions would direct the user to the trip planning page to see trip details.



Dashboard

Trip Planning

The Trip Planning page allows users to specify a start and end location and then would display route information such as the length of trip, weather, pictures and description of the location. If not start location is specified, the start location would be defaulted as the user's current location. There is also an option to save the trip for future reference and to get a playlist for the trip.



Plan

Home +

Search +

Date
2019-04-25

Time

SAVE TRIP

77 Rochester St, Homebush NSW 2140, Australia TO Canberra ACT 2601, Australia

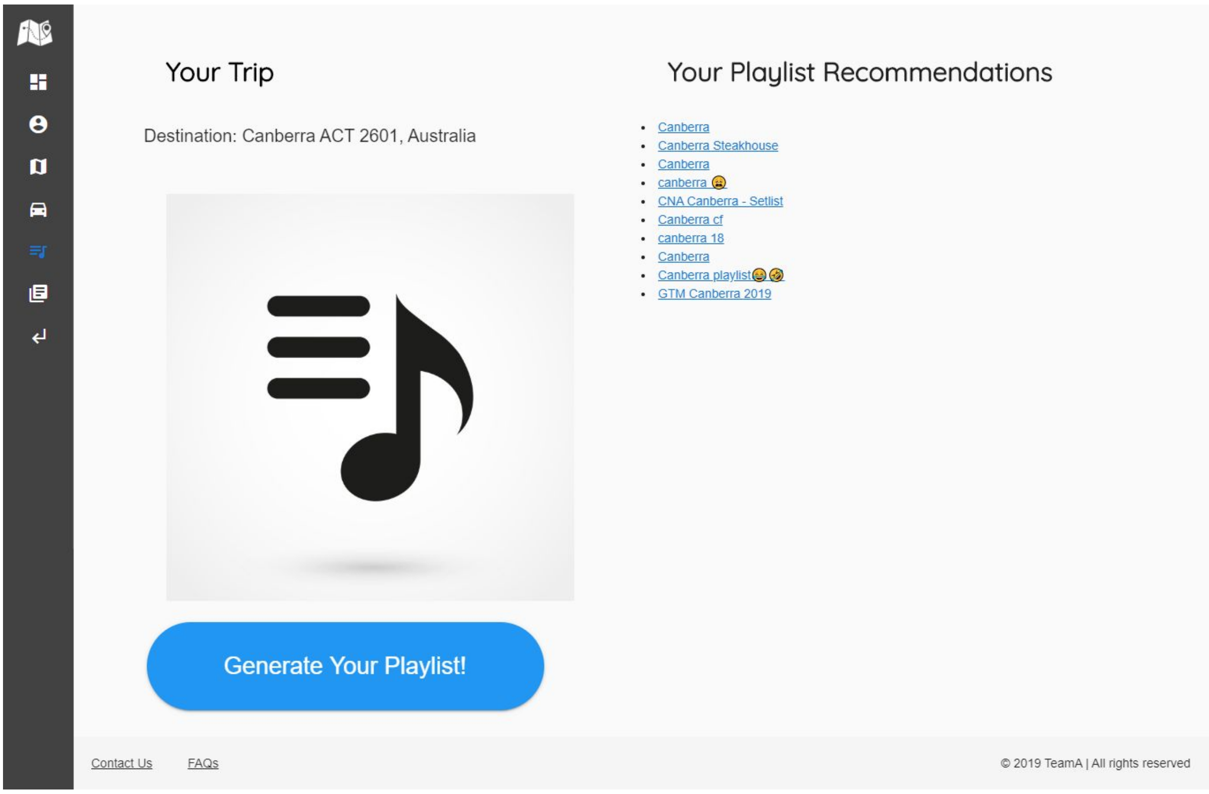
LENGTH OF TRIP
271 km
APPROXIMATE TRIP TIME
2 hours 59 mins
WEATHER
Mostly sunny 23 degrees
MY TRIP PLAYLIST
Get a playlist

Canberra is the capital city of Australia. With a population of 410,301, it is Australia's largest inland city and the eighth-largest city overall. The city is located at the northern end of the Australian Capital Territory, 280 km (170 mi) south-west of Sydney, and 660 km (410 mi) north-east of Melbourne. A resident of Canberra is known as a

Trip Planner

Playlist

When the user clicks “Get a playlist” from the trip planner, they will be redirected to this page which automatically gives a list of recommended public playlists from Spotify which relates to their trip destination. Alternatively, if they click “Generate Your Playlist!”, a private playlist would be generated on Spotify for the user, containing songs specifically for their trip.



Your Trip

Destination: Canberra ACT 2601, Australia

Generate Your Playlist!

Your Playlist Recommendations

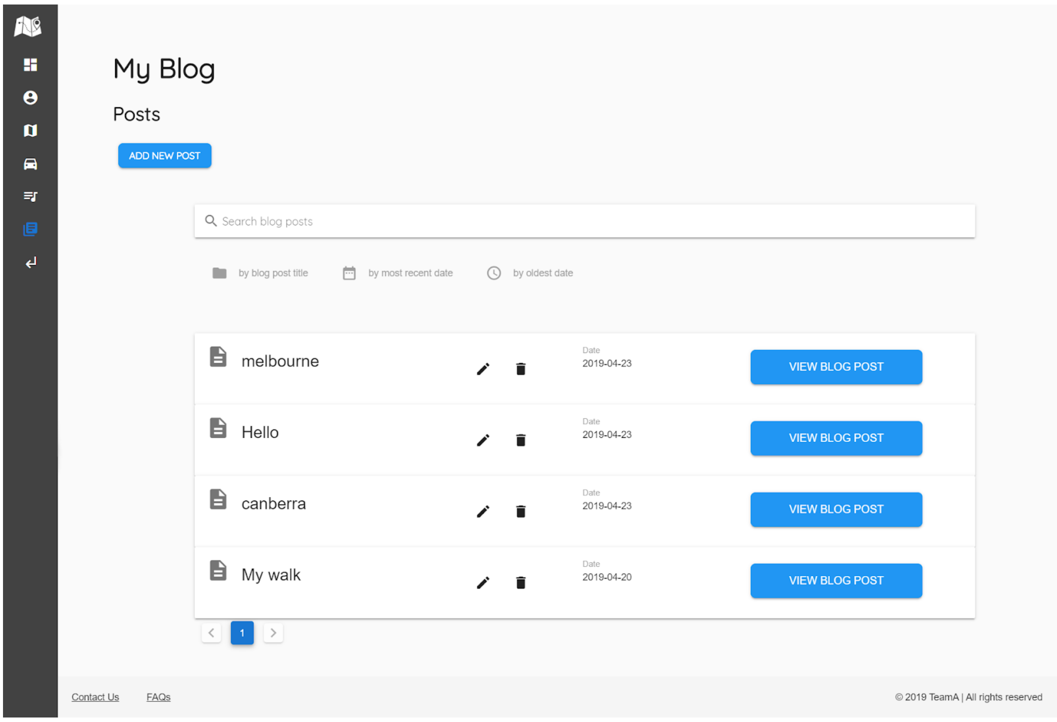
- Canberra
- Canberra Steakhouse
- Canberra
- canberra 😩
- CNA Canberra - Setlist
- Canberra cf
- canberra 18
- Canberra
- Canberra playlist😂🤣
- GTM Canberra 2019

Contact Us FAQs © 2019 TeamA | All rights reserved

Trip Playlist

Blog

The blog page is collection of all the blog posts the user has made. Blogs can be searched for and sorted for easy viewing. Additionally, you can make new posts or edit and delete existing blogs.



My Blog

Posts

ADD NEW POST

Search blog posts

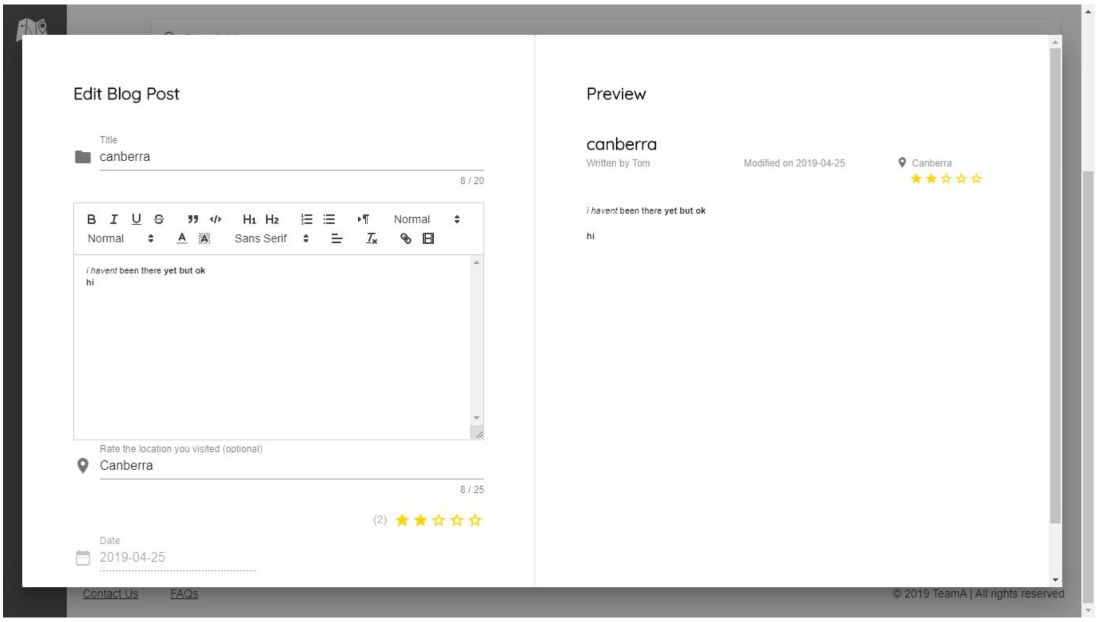
by blog post title by most recent date by oldest date

	Date	
melbourne	2019-04-23	VIEW BLOG POST
Hello	2019-04-23	VIEW BLOG POST
canberra	2019-04-23	VIEW BLOG POST
My walk	2019-04-20	VIEW BLOG POST

1

Contact Us FAQs © 2019 TeamA | All rights reserved

My Blog page



Edit Blog Post

Title
canberra
8 / 20

i havent been there yet but ok
hi

Rate the location you visited (optional)
Canberra
8 / 25

(2) ★★☆☆☆

Date
2019-04-25

Preview

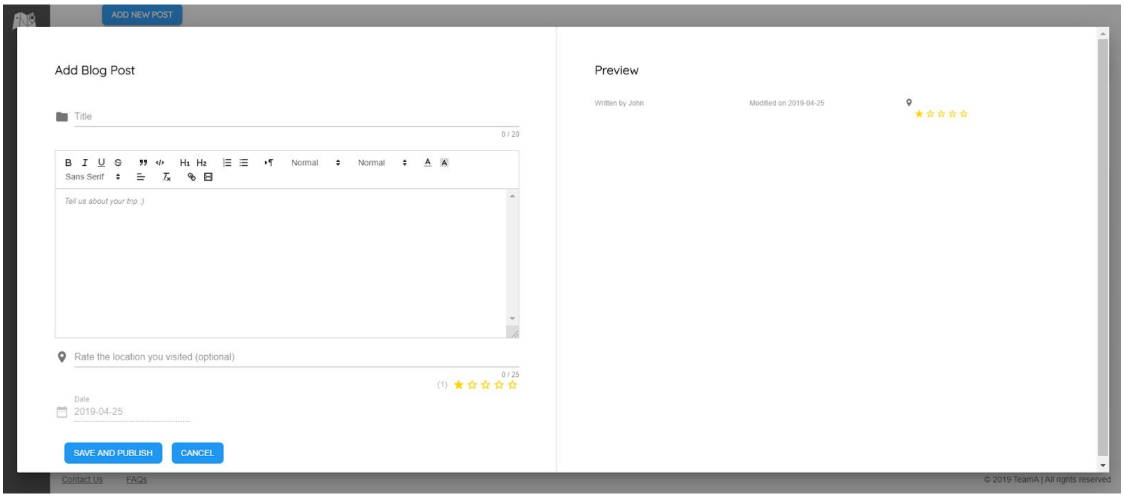
canberra
Written by Tom
Modified on 2019-04-25
Canberra
★★☆☆☆

i havent been there yet but ok

hi

© 2019 TeamA | All rights reserved

Edit Blog



Add Blog Post

Title
0 / 20

Tell us about your trip :)

Rate the location you visited (optional)
0 / 25

(1) ★☆☆☆☆

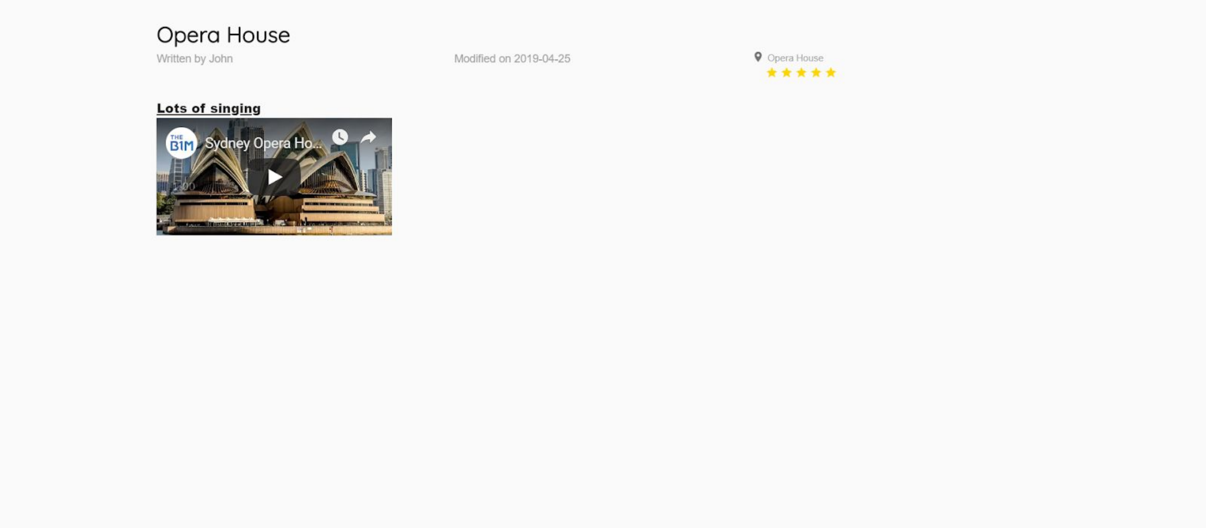
Date
2019-04-25

Preview

Written by John
Modified on 2019-04-25
★☆☆☆☆

© 2019 TeamA | All rights reserved

Add Blog



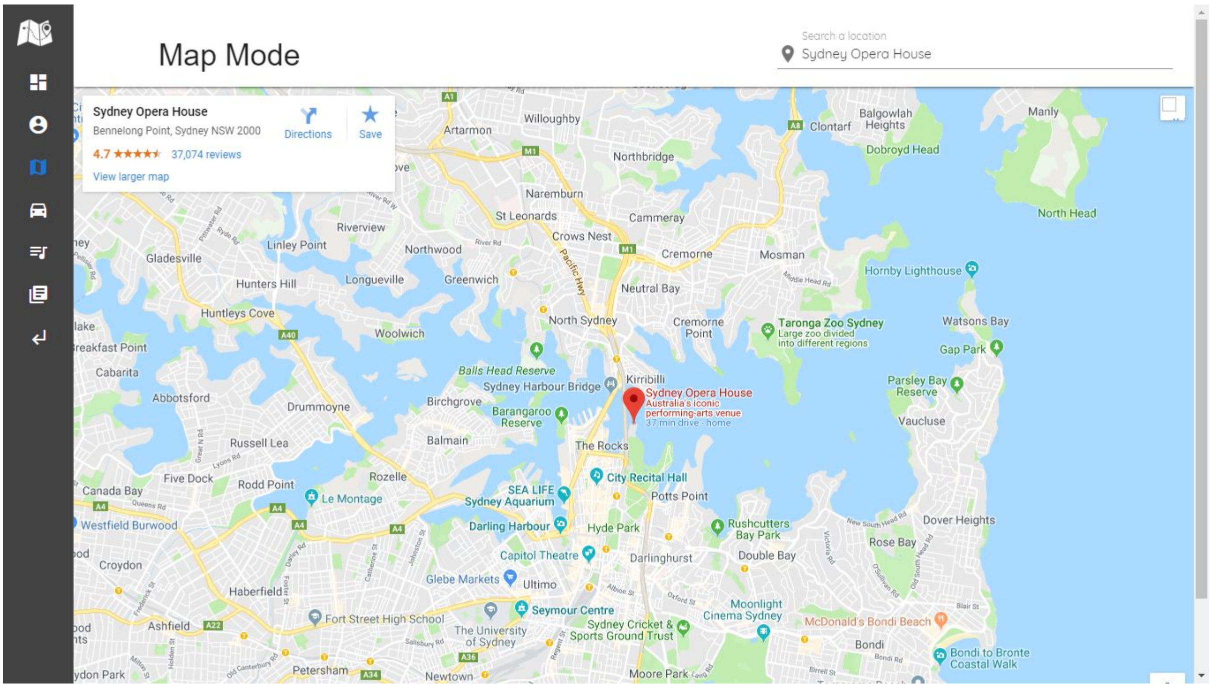
Opera House
Written by John
Modified on 2019-04-25
Opera House
★★★★★

Lots of singing

View Blog Post Example

Map Mode

The Map Mode page provides a quick google map for users to search up any location. The large display makes it easier for users to see surroundings.



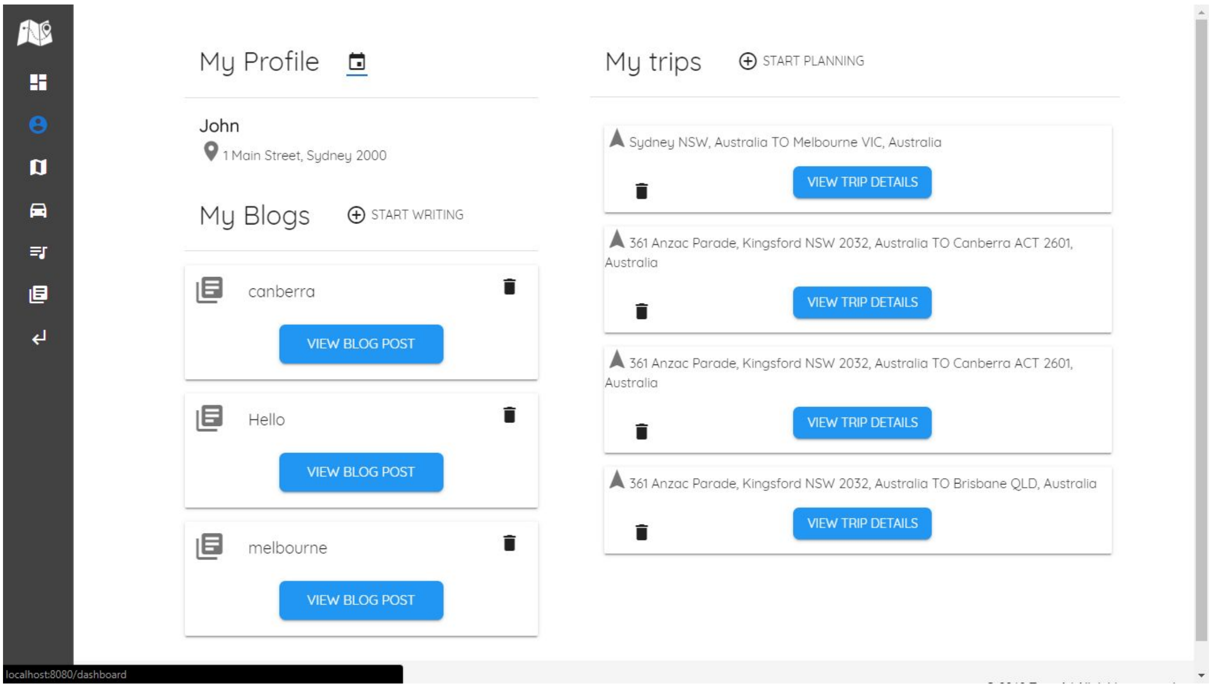
Map Mode

Search a location: Sydney Opera House

Sydney Opera House
Bennelong Point, Sydney NSW 2000
4.7 ★★★★★ 37,074 reviews
View larger map

My Profile

This page shows all the information relating to the user logged in. It displays the user's current location, their save blogs and also saved trips. Each button takes the user to the respective saved blog/trip.



John
1 Main Street, Sydney 2000

canberra

Hello

melbourne

Sydney NSW, Australia TO Melbourne VIC, Australia

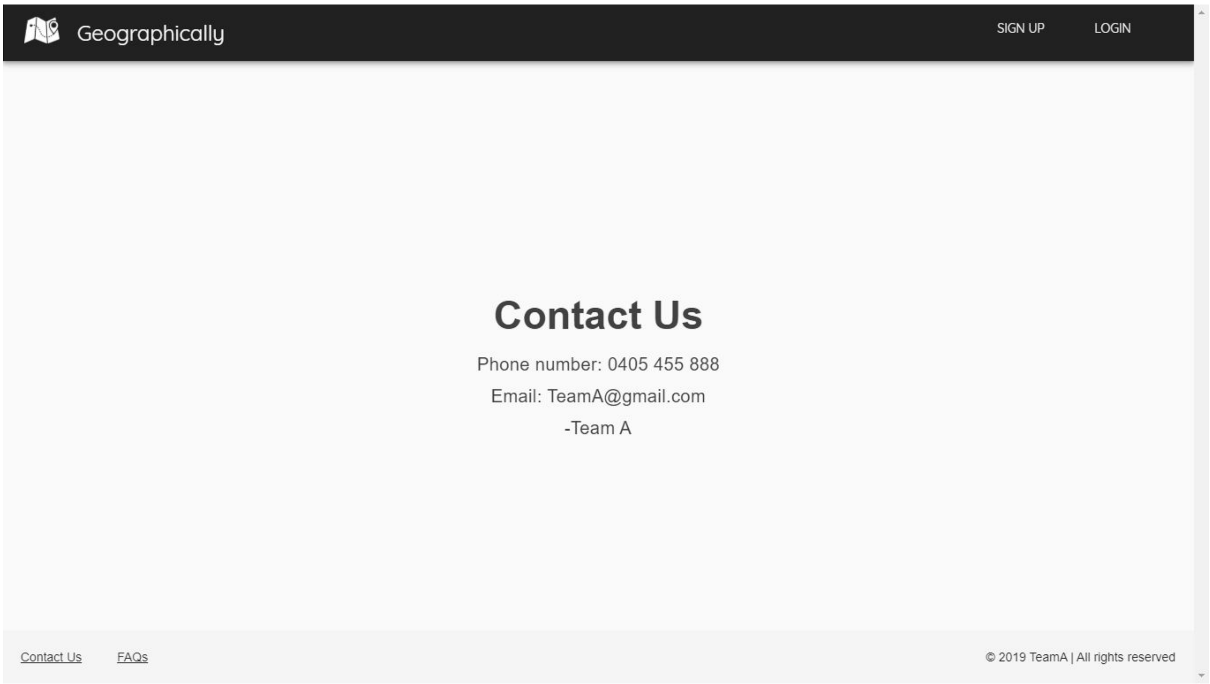
361 Anzac Parade, Kingsford NSW 2032, Australia TO Canberra ACT 2601, Australia

361 Anzac Parade, Kingsford NSW 2032, Australia TO Canberra ACT 2601, Australia

361 Anzac Parade, Kingsford NSW 2032, Australia TO Brisbane QLD, Australia

Contact Us

This page provides contact details for our users to provide feedback or ask questions to the developers. It can be accessed from the footer of any page.



Contact Us

Phone number: 0405 455 888

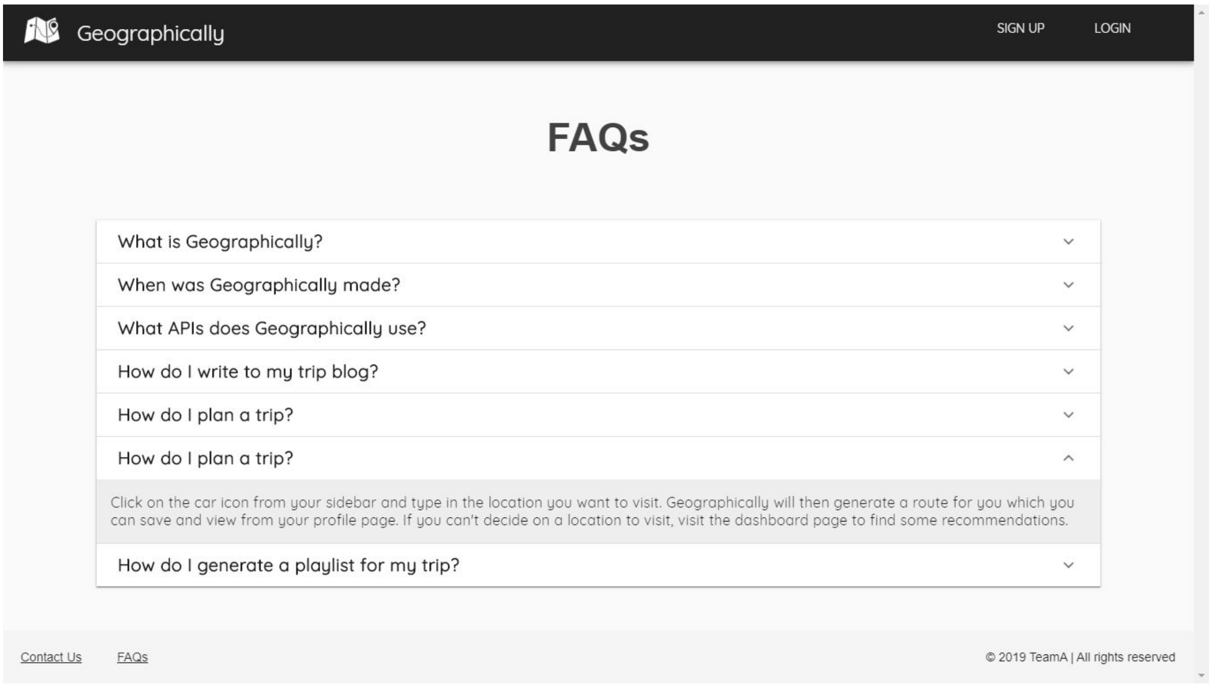
Email: TeamA@gmail.com

-Team A

© 2019 TeamA | All rights reserved

FAQs

This page provides answers to the most commonly asked questions by our users and can be accessed from the footer of any page.



Geographically

SIGN UP LOGIN

FAQs

- What is Geographically?
- When was Geographically made?
- What APIs does Geographically use?
- How do I write to my trip blog?
- How do I plan a trip?
- How do I plan a trip?

Click on the car icon from your sidebar and type in the location you want to visit. Geographically will then generate a route for you which you can save and view from your profile page. If you can't decide on a location to visit, visit the dashboard page to find some recommendations.

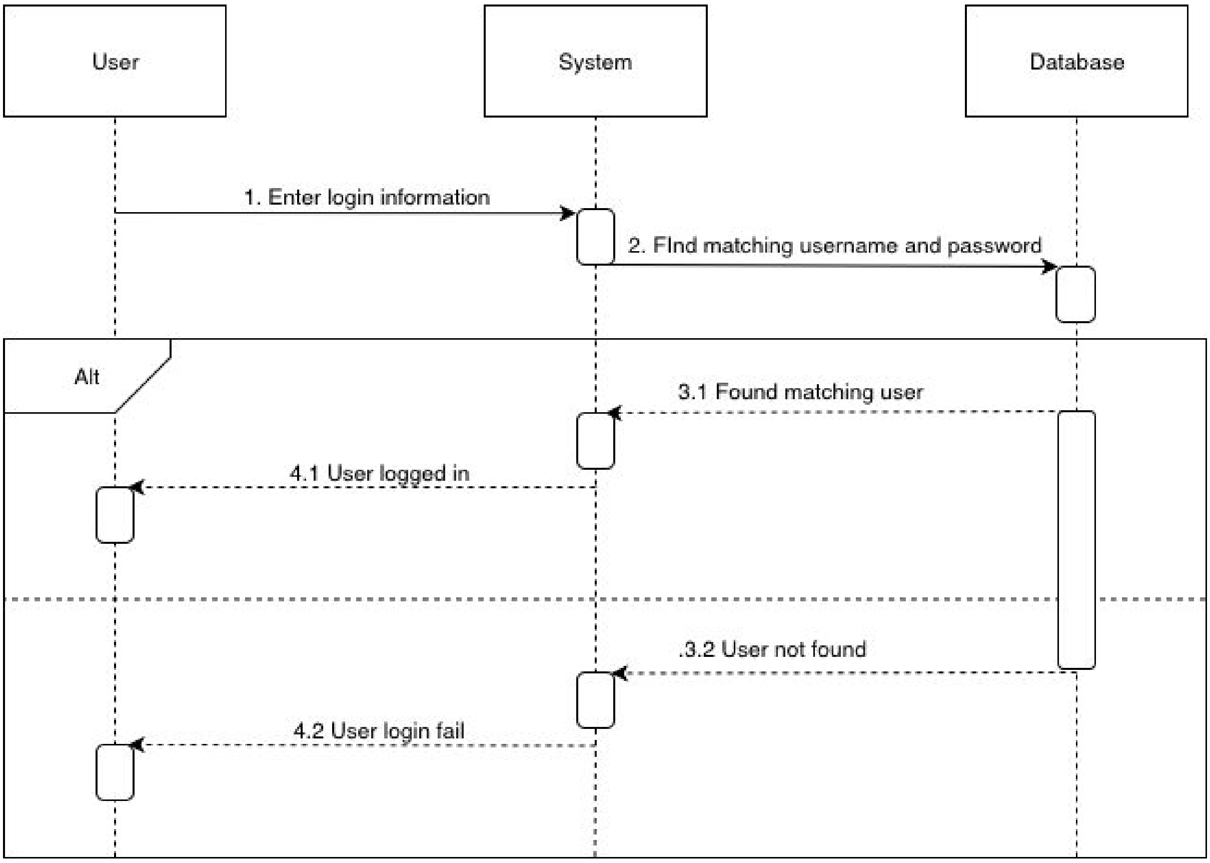
- How do I generate a playlist for my trip?

Contact Us FAQs

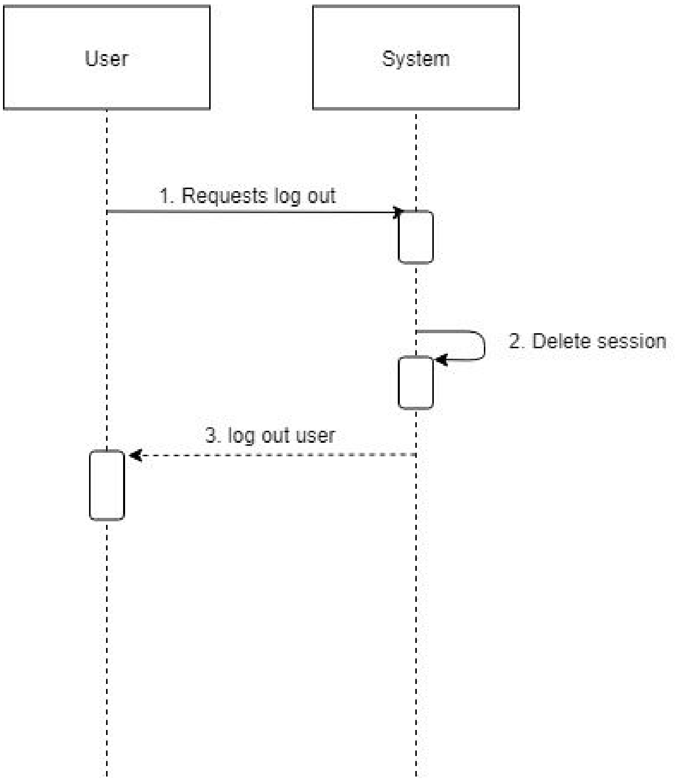
© 2019 TeamA | All rights reserved

Sequence diagrams for each use case/story point

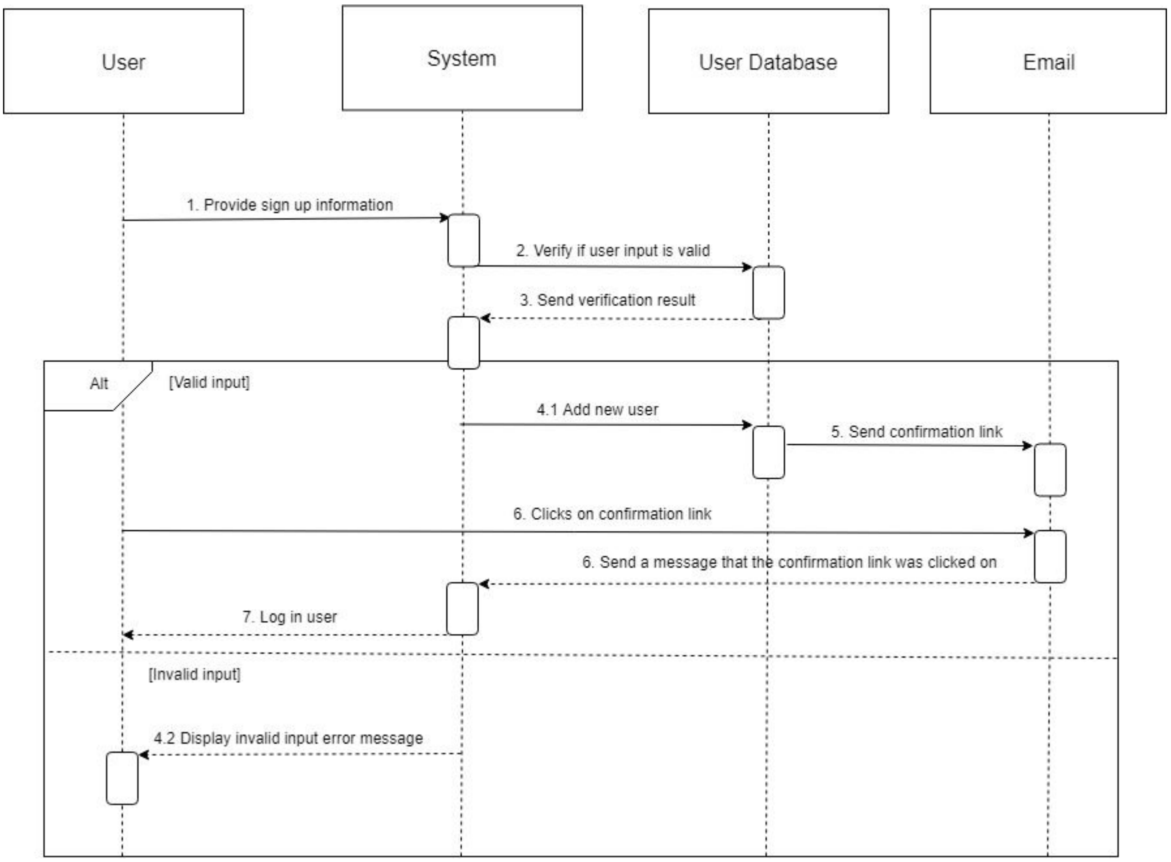
Login sequence diagram:
https://drive.google.com/file/d/18t0NXIe-wQLXM3g9rCjHVeQhjE6Gy1a-/view?usp=sharing



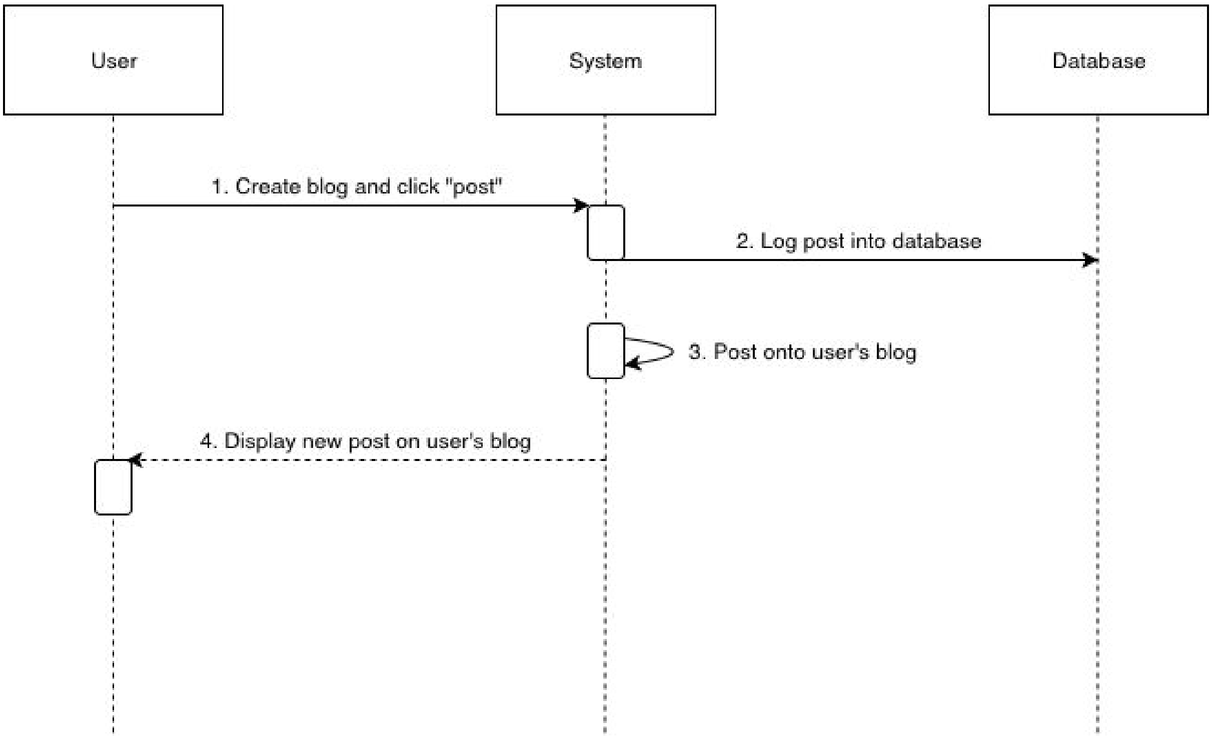
Log out sequence diagram:
https://www.dropbox.com/s/d8ezs1qggsaroa6=/Logout.drawio?dl=0



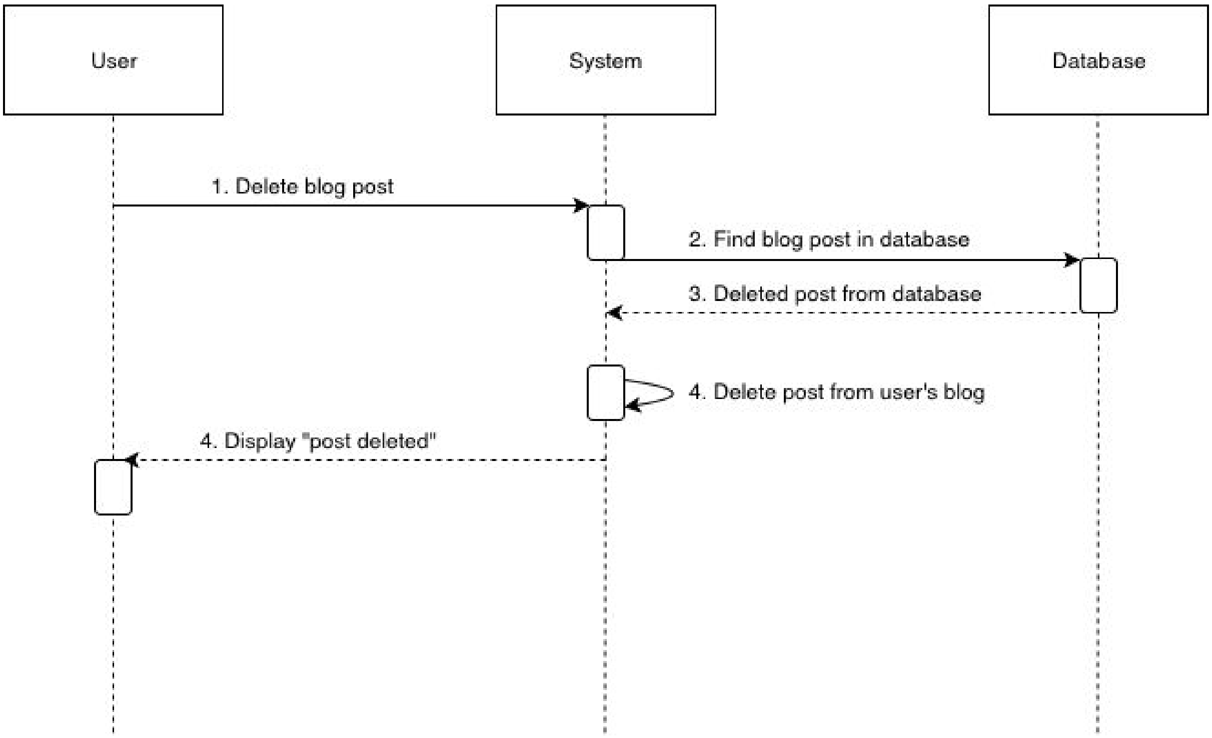
Sign up sequence diagram:
https://www.dropbox.com/s/eb5h02vvpak6mnq/Signup.-drawio?dl=0


User
System
User Database
Email
1. Provide sign up information
2. Verify if user input is valid
3. Send verification result
Alt
[Valid input]
4.1 Add new user
5. Send confirmation link
6. Clicks on confirmation link
6. Send a message that the confirmation link was clicked on
7. Log in user
[Invalid input]
4.2 Display invalid input error message

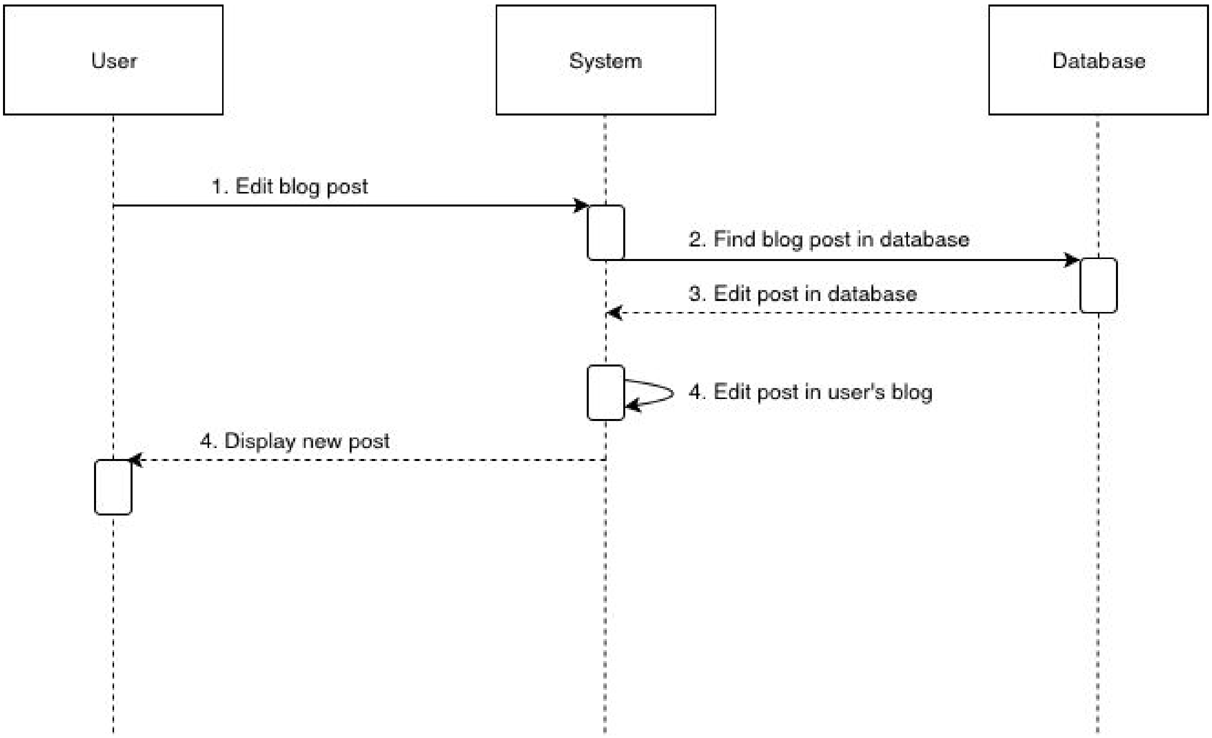
Create blog sequence diagram:
https://drive.google.com/file/d/1CZC1_mVuz5V2wJI48KSHSXgDgG-WZyqs/view?usp=sharing


User
System
Database
1. Create blog and click "post"
2. Log post into database
3. Post onto user's blog
4. Display new post on user's blog

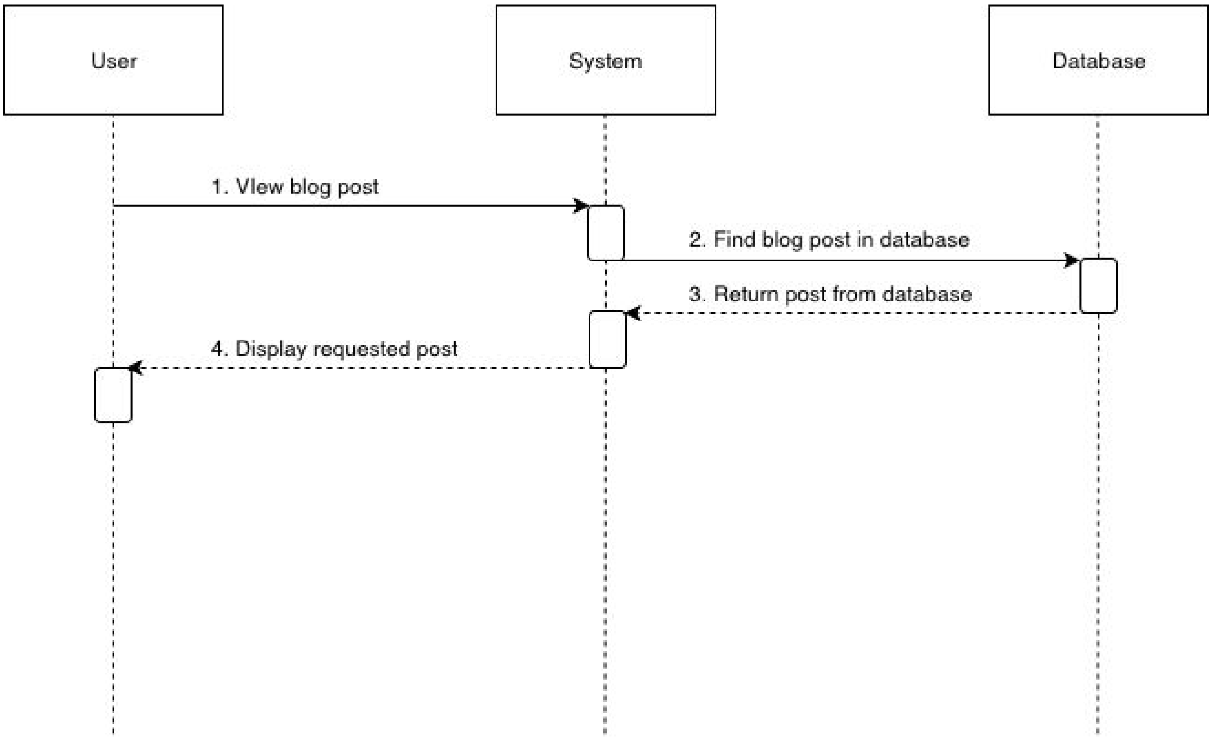
Delete blog sequence diagram:
https://drive.google.com/file/d/1pJ_PY8lP4hJEulMWuLpKPnOeY270cFIo/view?usp=sharing


User
System
Database
1. Delete blog post
2. Find blog post in database
3. Deleted post from database
4. Delete post from user's blog
4. Display "post deleted"

Edit blog sequence diagram:
https://drive.google.com/file/d/1A9IHs9WeXGxWZNpBVKJkgaqKF_wfj7g4/view?usp=sharing

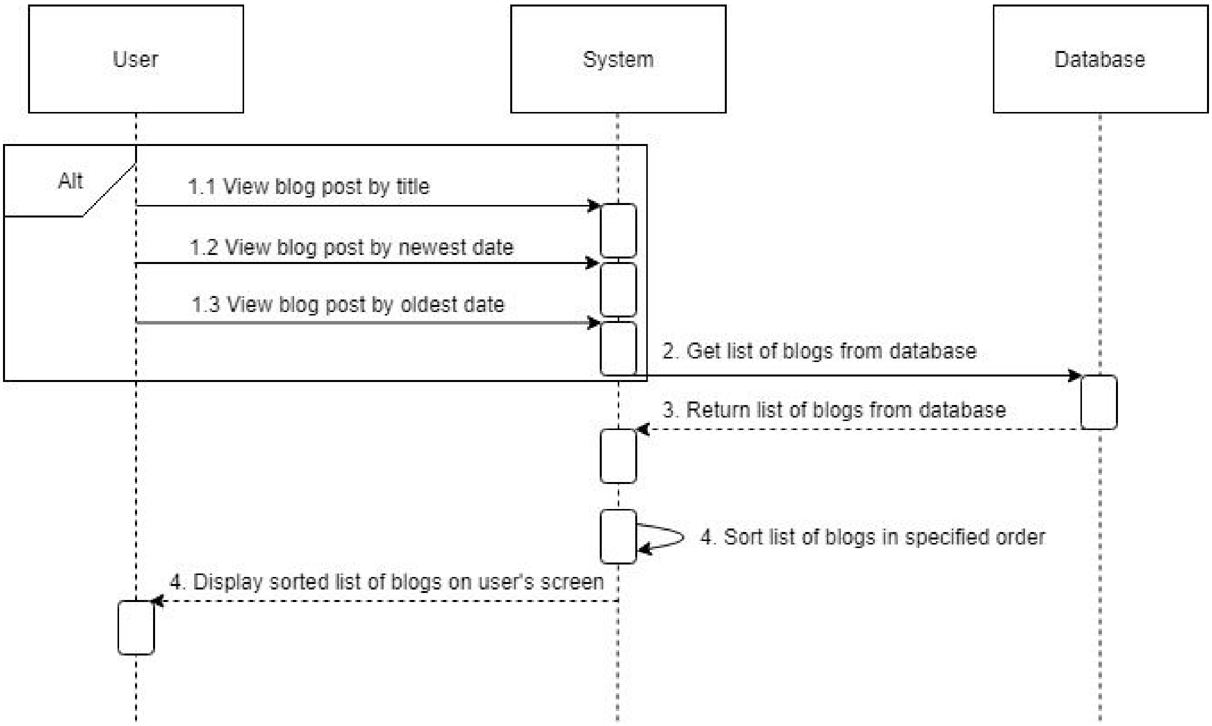


View blog post sequence diagram:
https://drive.google.com/file/d/1obN0ZC21SitdTLREDlFPJqJ_7RlpcydT/view?usp=sharing


User
System
Database
1. View blog post
2. Find blog post in database
3. Return post from database
4. Display requested post

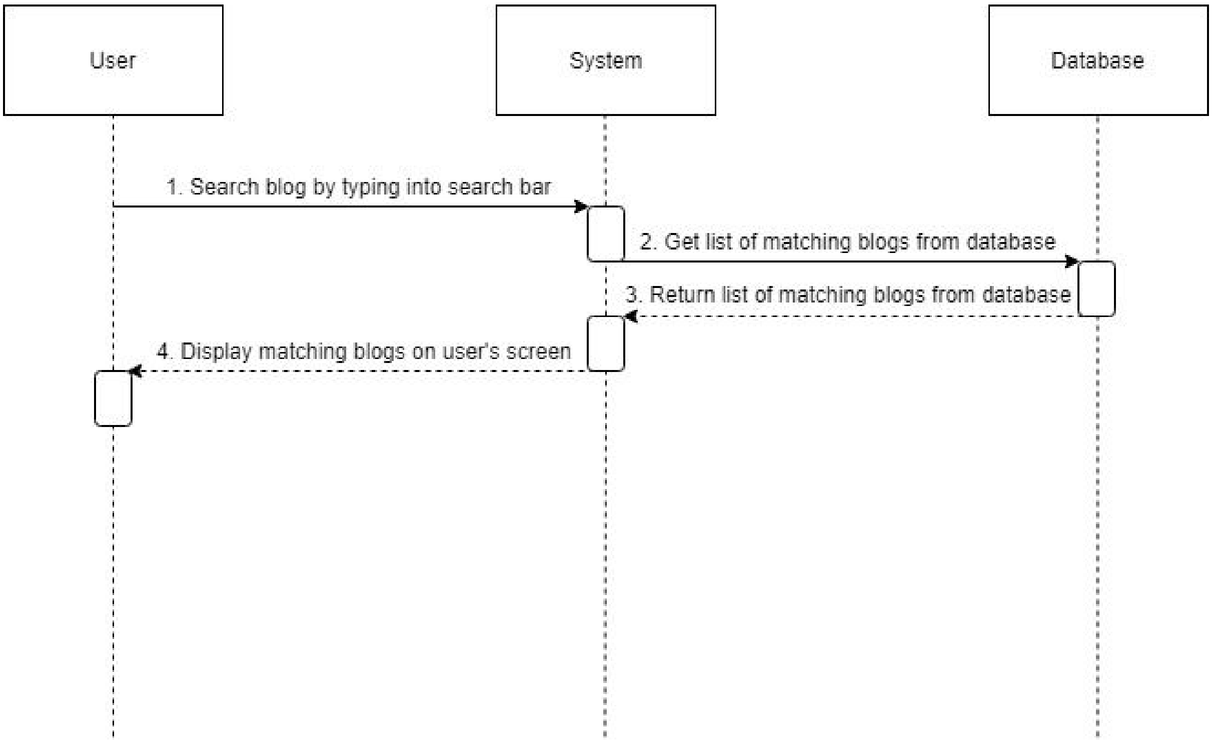
View blog posts with options sequence diagram

https://drive.google.com/file/d/18mlltO4oFvmwhD9mvMUywsMVtZuEa3Y6/view?usp=sharing

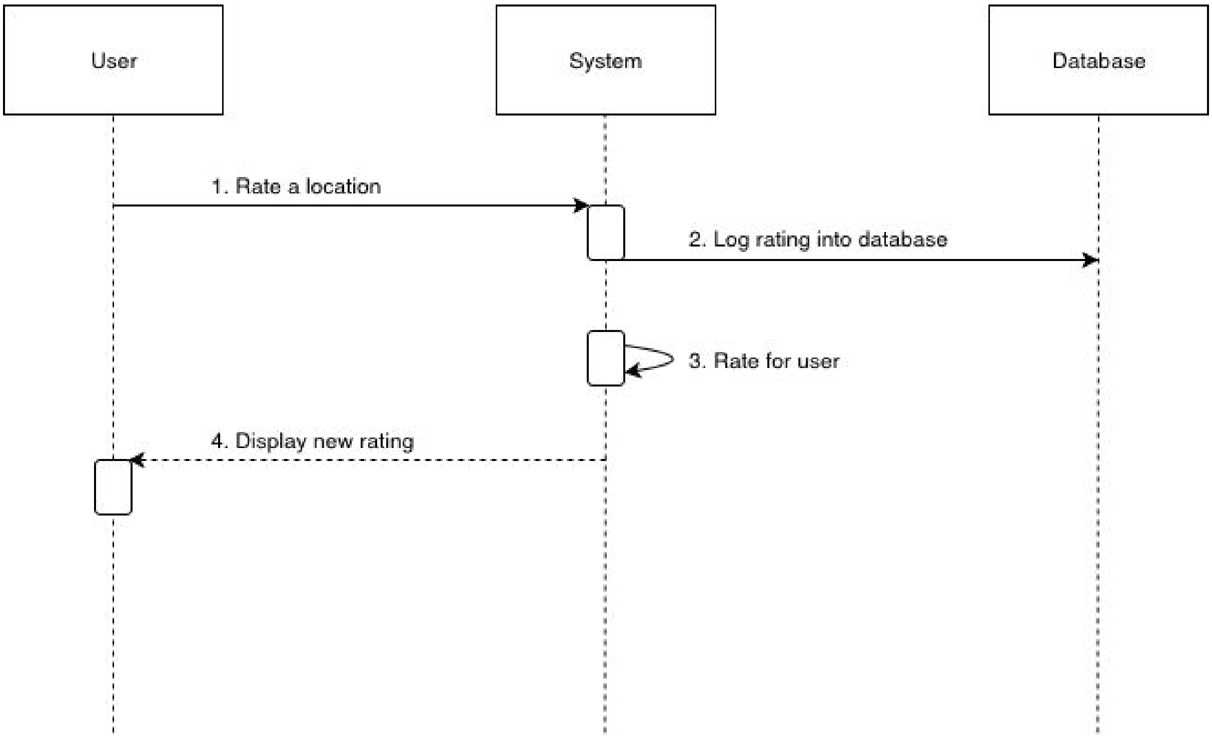

User
System
Database
Alt
1.1 View blog post by title
1.2 View blog post by newest date
1.3 View blog post by oldest date
2. Get list of blogs from database
3. Return list of blogs from database
4. Sort list of blogs in specified order
4. Display sorted list of blogs on user's screen

Search blog sequence diagram:

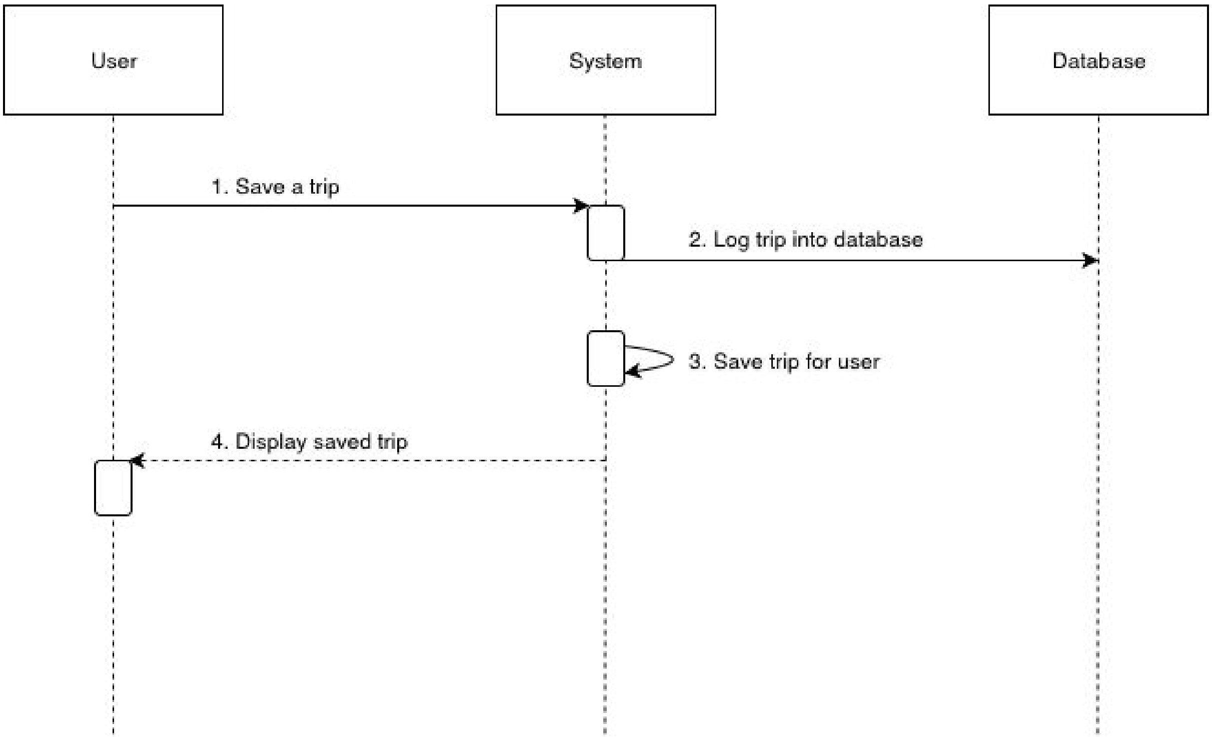
https://drive.google.com/file/d/1UPIqUnPWj-f1FSDdSKmCakWagDvE9AB9/view?usp=sharing


User
System
Database
1. Search blog by typing into search bar
2. Get list of matching blogs from database
3. Return list of matching blogs from database
4. Display matching blogs on user's screen

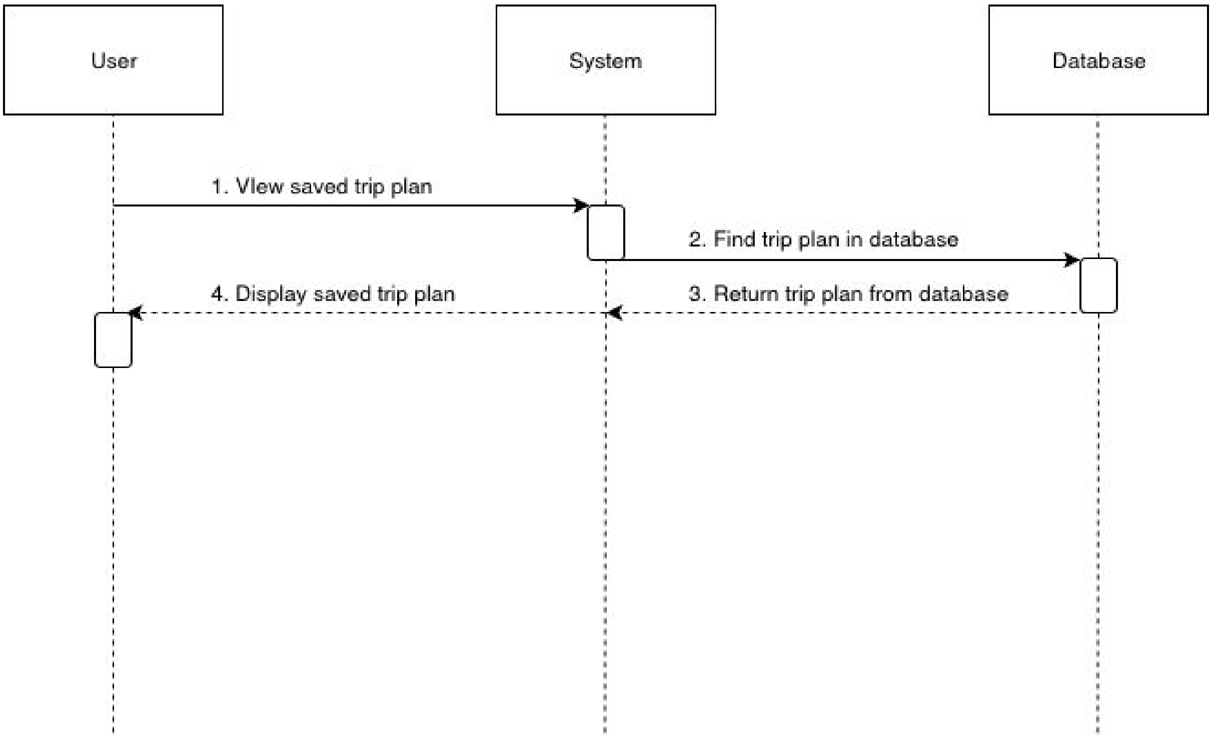
Rate a location sequence diagram:
https://drive.google.com/file/d/1g39oZ2OYRX-r1pcnnSTJDNQvKMgoY_H3/view?usp=sharing



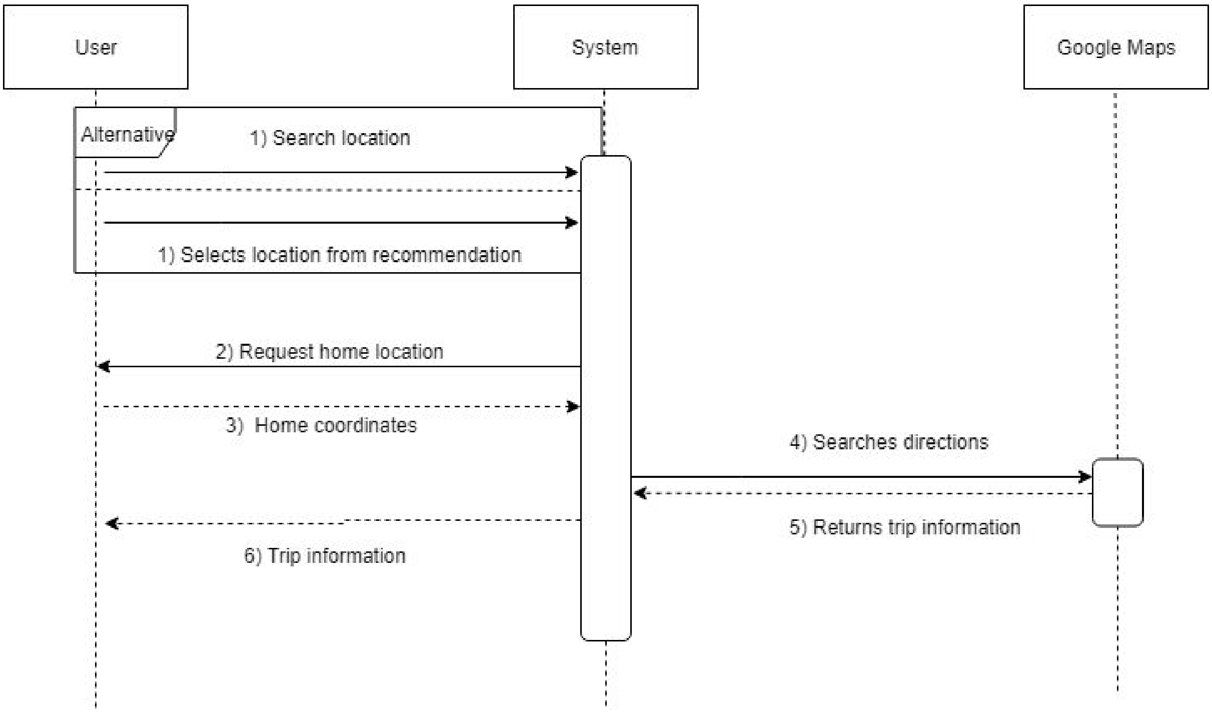
Save a trip sequence diagram:
https://drive.google.com/file/d/19ICIbJtxXut7eGaxXuz3Th_wpzQsHqaL/view?usp=sharing



View saved trip plan sequence diagram:
https://drive.google.com/file/d/1XDjQ71R9U7J1VZ31GPStNuMVMhuAi84l/view?usp=sharing

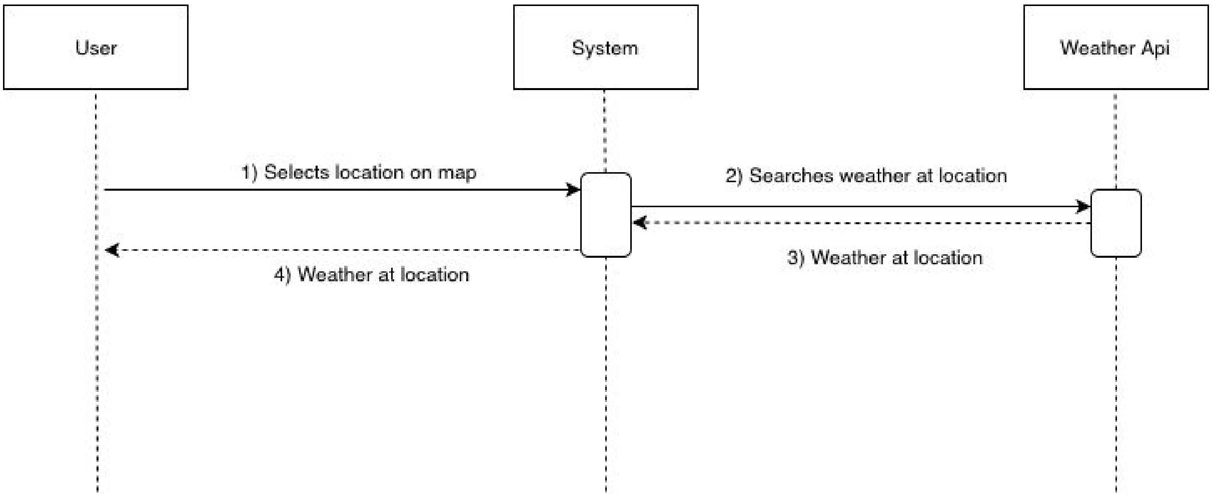

User
System
Database
1. View saved trip plan
2. Find trip plan in database
4. Display saved trip plan
3. Return trip plan from database

Plan a trip sequence diagram
https://drive.google.com/file/d/1YxwOnCy8B46CRvu_o0MzMNSZuQ0PdN91/view?usp=sharing



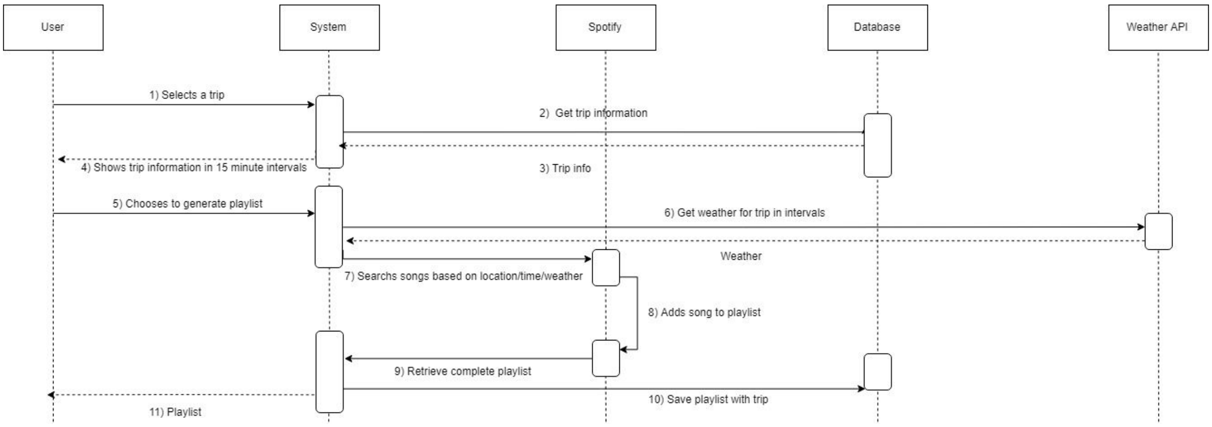
Show weather sequence diagram

https://drive.google.com/file/d/15WdaQs4zd28_n6PLwbCAaHFU5nzDisIz/view?usp=sharing



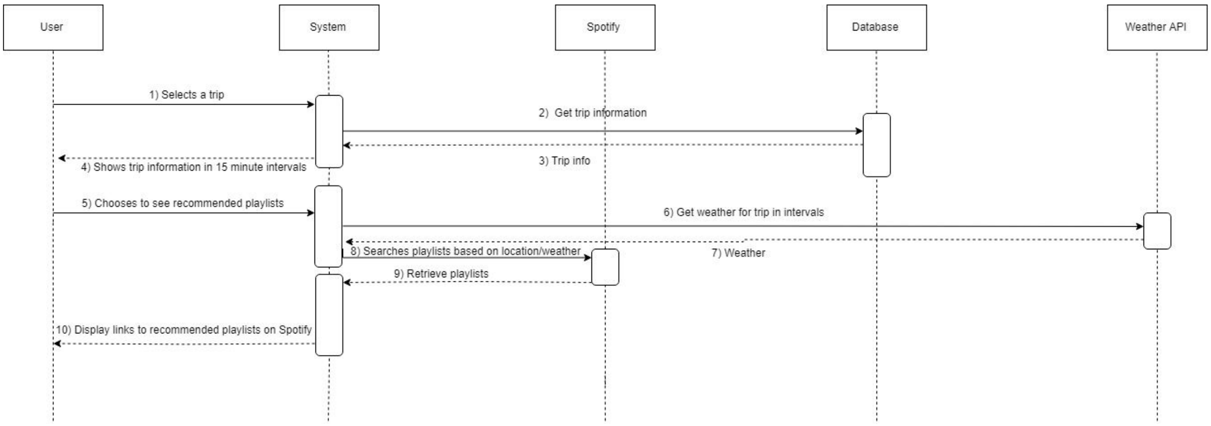
Make playlist sequence diagram

https://drive.google.com/file/d/1DcaWJuAm9r2TTgXHirl2O7rSZyAy94wQ/view?usp=sharing

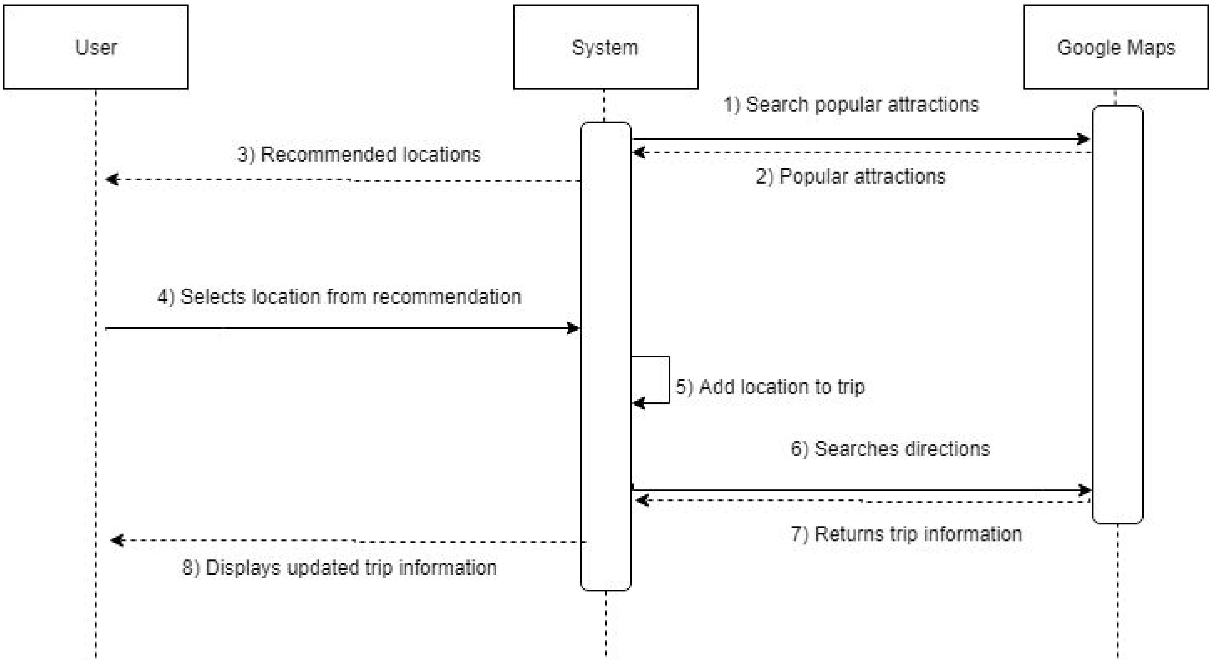

User
System
Spotify
Database
Weather API
1) Selects a trip
2) Get trip information
3) Trip info
4) Shows trip information in 15 minute intervals
5) Chooses to generate playlist
6) Get weather for trip in intervals
Weather
7) Searchs songs based on location/time/weather
8) Adds song to playlist
9) Retrieve complete playlist
10) Save playlist with trip
11) Playlist

Recommended playlists sequence diagram

https://drive.google.com/file/d/1JA5J4UpinSTmr8_UK-dIUT2yw9wtCO0-/view?usp=sharing


User
System
Spotify
Database
Weather API
1) Selects a trip
2) Get trip information
3) Trip info
4) Shows trip information in 15 minute intervals
5) Chooses to see recommended playlists
6) Get weather for trip in intervals
7) Weather
8) Searches playlists based on location/weather
9) Retrieve playlists
10) Display links to recommended playlists on Spotify

Recommended attractions sequence diagram
https://drive.google.com/file/d/1m_tlmY7ZagVwM3ZsqOlgJX_9PEMFKCSB/view?usp=sharing


User
System
Google Maps
1) Search popular attractions
2) Popular attractions
3) Recommended locations
4) Selects location from recommendation
5) Add location to trip
6) Searches directions
7) Returns trip information
8) Displays updated trip information

Design and Key Technologies

Design of the website (GUI)

The website’s design achieved a simple and attractive look because basic design principles such as white space (padding) and a consistency in the use the fonts, basic style of icons, colour and layout was implemented throughout the website. We chose the classic colour scheme of white, blue and black inspired by companies, Google and Apple, to establish a sense of familiarity and trust to our application and brand. The simple icons implemented gave the design a cleaner look, and providing labelled buttons and a navigation sidebar assisted with making the website user-friendly.

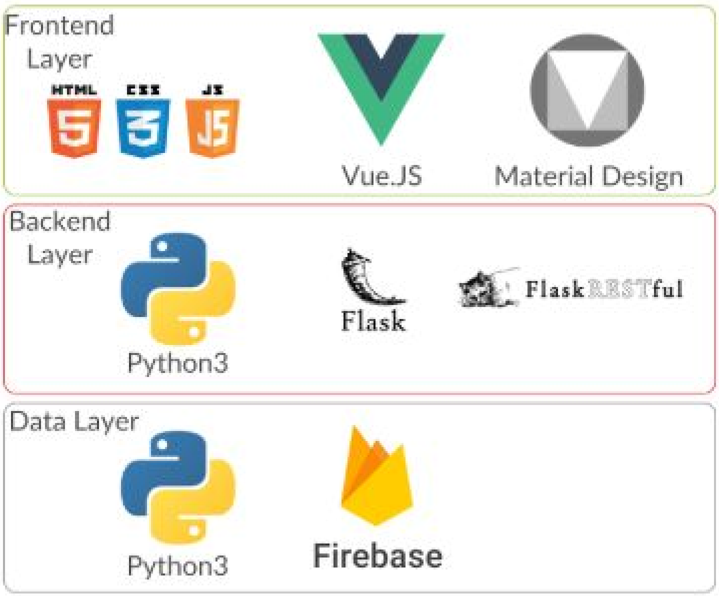
Implementation: Key Technologies and their benefits

Summary

Frontend	**Backend**
● Vue.js: Progressive JS framework	● Python 3 ● Flask with Flask RESTful library

● Vuetify (Material Design Framework) ● APIs: Quill editor, Google maps	● Firebase (Cloud storage) Libraries: - Firebase auth: allowed for users to login, sign up and sign out. - Firestore: allows for data collection and storage. ● APIs: Sanitize-html, Google maps, willy weather, Spotify

Technology stack diagram



Front End

Vue CLI 3 (Vue.js)

Vue.js is a progressive JS framework that was used to create the user interface for our web application. It had the advantage of being easy to learn (need to be familiar with HTML and

JS), which is very beneficial on a tight schedule). It was also very flexible to use, allowing for the option to write templates in either HTML or JS, and was able to be integrated simply with the rest of the app as it is built on JS. Furthermore, vue.js allowed for easy data binding between HTML and javascript which allowed for real-time updates to be made on our websites. Components coded could be reused easily across multiple webpages to reduce code repetition.

Vuetify (Material Design)

Vuetify is a plugin that was compatible with vue and provided a material design component framework for the website. This reduced the need for manual CSS implementation and a cleaner and attractive GUI by providing a library of animated HTML components and CSS classes.

Back End

Python 3 and Flask

Group members were more comfortable with using Python and Flask for backend, as they have been used before for previous courses. This allowed us to cut time required to learn new languages. It allowed for interaction with RESTful APIs, which is necessary for the web app.The RESTful APIs enabled communication between the frontend and backend.

Firebase

Firebase had many advantages including its wide range of libraries and SDKs to help with user authorisation and its ability in detecting real-time updates across multiple clients so that data on the website could be change accordingly. It also was easy to use in relation to connecting the front end to the back end due to its detailed documentation and high popularity among tutorials online. It also had an error checking system within its Firebase Auth library which produced certain error messages when the user sign up or logged in. This allowed error handling of user input to be completed easily.

External data sources: APIs

As our web application is multi-purpose, we require access to several APIs, with them accomplishing one purpose.

Google Maps API

This is the core of the application, and we used multiple Google Maps APIs, such as Embedded and Places. Google Maps Embedded allowed us to embed the Google Maps interface into our app, which was crucial in order to view destinations on a scale, and also to demonstrate the route a user would take. This was be combined with Google Places in order to convert user input into a specific location which could be interpreted by other APIs such as the Wikipedia and Weather API. Finally Google Routes provided all the required route information. Google Maps is the most well-known maps application, with good reason as well. The amount of utility (detailed above) that it provides is unrivalled from other Maps applications, and implementing the API greatly enhanced the quality of the web app. The documentation is also clean and simple, being written in digestible segments for each API function.

Wikipedia API

This was used to retrieve a brief description of a location. Wikipedia is considered the largest easily accessible database of information on the web, so utilising it's API to retrieve information about travel destinations was a good idea. Furthermore, the Wikipedia API had good documentation detailing easy to follow endpoints, as well as guides on how to handle inputs/outputs and errors.

Spotify API

The primary objective was to get Spotify Playback and access to playlists on the application, but also utilise the Spotify API to generate playlists based on the trip. Upon looking into several music player APIs, Spotify was the most viable for several reasons. The lengthy and descriptive documentation is easy to understand, and details several API for each Spotify function. The endpoints were described clearly and thus were easier to implement. However the main advantage that Spotify has was that its endpoints that allowed us to retrieve user information and statistics which was handy in allowing us to generate a recommended playlist.

WillyWeather API

Utilising a weather API is also crucial for the web app because it greatly assisted the user in choosing the right time to visit certain locations. Upon research into the Bureau of Meteorology data services, we found that they currently provide long term forecasts and information through FTP however their real time data services required a paid subscription. Subsequently, we found an api which sources its data from the Bureau of Meteorology. WillyWeather satisfies this criteria thus provided detailed information about the weather forecast of certain locations in Australia down to the hour in an easy to use api. This was also essential for the generation of road trip playlists as we needed to select songs based on the weather at a certain location at a certain time.

Quill Editor

Using the Quill Editor was very useful as it provided video embedding and text formatting such that the user could customise their blog post. By providing an rich text editor which contained formatting icons similar to Microsoft Word, it also allowed blog post making to remain user-friendly. Furthermore, text editors were customisable which allowed us to remove unnecessary formatting options from the toolbar of the text editor.

Sanitize-html

Since a text editor was used to collect the user's blog content as well as HTML and CSS input, Sanitize-html was used to help filter and clean out HTML fragments, and unwanted CSS and Js code to keep the website safe from XSS attacks and other malicious code from potential users. This ensured that only the blog post content and text formatting features available from the blog text editor were collected from the user when creating or editing a blog post. Sanitize-html was also easy to implement and set up as it had clear documentation and examples in respect to how it could be used.

Team organisation

Organization of the team
Initially, the team was organised into two groups, people doing the front end, and people doing the back end of the web application based upon everyone's preferences, strengths and weaknesses. For example, if one team member had a strong sense in design, they were assigned to work on the front end to help with the user interface.

Front End	Back End
Winston Darmawan Mandy Chen Alana Hua	Galen Rowan Elizabeth Zhong

Once the basic front end was implemented, front end team members were re-assigned to a certain webpage such as the 'My Blog' page to help with the team members who were working on the back end get the functionality of the website's features to work. Often times, team members would help each other with the implementation when a team member for example was struggling or transfer over responsibilities.

Summary of everyone's Responsibilities

Student ID	Student	Responsibilities/Contribution
z5209503	Alana	Designed the website's GUI and implemented the code for the home page, blog, sign up, login, navigation sidebar, contact and FAQ page; Worked on the user interface for the maps page and functionality of the profile page (showing saved blog posts).
z5209166	Elizabeth	Worked on the functionality of location recommendations, trip planner, trip playlist and profile page (showing saved trip plans).
z5214822	Galen	
z5205689	Mandy	Helped with implementing the user interface for the profile page, trip planner and playlist; Worked on the functionality of the trip playlist.
z5205439	Winston	Helped with implementing the user interface for sign up, login and dashboard; Worked on the functionality of log out and profile page. (Editing user's details)

How the project went

From the beginning, the concept of forming an idea and making of architectural choices was completely foreign to most members of the group and thus progress was initially slow and the success of the project seemed uncertain. During the idea formation, idea of an outfit manager was settled upon because it was innovative and unique. However, the more the group thought about the idea, the more unrealistic the idea seemed given the time and skill level we had. Thus, the idea was changed to a trip planning application.

From there, a rough division of roles was assigned and we began research into the optimal software architecture and API choices that would best optimise our application, with some recommendations from our tutor. Initially opting for the MEAN stack, time constraints caused us to change to a framework that was more familiar to the group, utilising Vue.js and Python+Flask and Flask RESTful.

Beginning development, there were several team members stacked working on the front-end compared to the back-end, and thus our first iteration and demonstration of the project had a very clean and attractive front-end but with minimal connection from the back-end. This meant that the majority of the front-end was missing its core functionality from back-end.

By re-assigning responsibilities of members to assist in the backend, we were able to implement more of the features before the final iteration, although some features were not able to be finished in time due to time constraints and the steep learning curves we faced when implementing new APIs and using Vue.js.

Therefore, the project was a good effort, however it was not a complete success since the project was missing some features that we had previously planned. In terms of group work, the group made decisions appropriate to the project and worked well together overall.

Issues/problems encountered

Lack of understanding of new frameworks

Initially, we planned to use the MEAN software stack as it was innovative and seemed to be used quite often to develop projects similar to our own. As it was built on Javascript, it also seemed to allow easy integration from frontend to backend. Upon recommendations by our mentor and research and discussion between group members, we opted to use Vue.js as our front end and Python/Flask on our backend as we were more familiar with Python. Despite this, we still used quite a few new technologies such as APIs which lead us to spend a lot of time learning how to use them and debugging simple issues that would not have occurred if we had the time to learnt it properly.

Segregation of front end and back end

Our allocation of tasks as frontend and backend cause team members to focus on learning how to code for their specific task. As a result, by our first demonstration, we had a strong frontend as more people were allocated to do it but the backend was lacking greatly and there was little communication between the frontend and backend as no one was well versed in what was happening on the other side.

Would you do it any differently now ?
In light of the issues experienced throughout the project, we would:

1. Use more familiar software and programming languages
2. Spread out the allocation of tasks so team members were more aware of the project as a whole
3. Make a to do list so members could choose which tasks they wanted to do and pass on tasks that they couldn’t complete

Potential Future Improvements to the website

- Firebase’s feature Admob would be implemented to generate profits to keep the website running. Firebase analytics which measure app usage would help with organizing advertising strategies.
- Attach departure times to saved trips so show the change in route details depending on time of day
- Speed up load times
- Implement a better algorithm to find songs for the playlist generator
- The brand’s name would be simplified such that it is easier to remember such as ‘GeoPlan’.