Capstone_Stage1 (Frontend Web)

- Description
- Intended User
- Features
- User Interface Mocks
  - Screen 1
  - Screen 2
  - Screen 3
  - Screen 4

- Key Considerations
  - How will your website handle data persistence?
  - Describe any corner cases in the UX.
  - Describe any third party libraries you'll be using and share your reasoning for including them.
  - Describe how you will implement a JS Framework.
- Next Steps: Required Tasks
  - Task 1: Project Setup
  - Task 2: Planning
  - Task 3: Implement UI for Each web page and frame
  - Task 4: Implement google maps
  - Task 5: Implement Database

GitHub Username: khargeprachi

Home Switch

Description

Home Switch is a temporary exchange of accommodation by two home owners for a mutually agreed period of time. This website allows people to find such places within their region of interest. This form of vacation has various benefits such as no accommodation costs, free usage of cars, local lifestyle experience, and convenience of a homely environment, new acquaintances, and many more. This application also helps users look for tourist spots near a selected household.

Intended User

This website is intended for a variety of users ranging from single travel enthusiasts or a group of friends to entire families who wish to plan an excursion. The users can be located anywhere in the country and should be willing to trade accommodation.

Features

1. It allows the users to save a list of locations they are interested to visit.
2. It allows users to upload photographs of their own home for others to view and help them select according to their preferences.
3. The user can view all the home switch enabled houses, within a given neighborhood marked on an interactive map.
4. The users can also look for tourist spots nearby and plan their holiday accordingly.
5. Users can communicate using a chat window.

User Interface Mocks

Screen 1: Search page

This is the homepage of our application. The map will be centered to the current location by default. The user can login and create its own profile. User can enter a desired location in the search query along with the begin and end dates of their holiday. The results of the query are highlighted on the map in the form of markers. It is made sure that only those home switch enabled households are highlighted which are available during the entered duration.
The corresponding info windows are displayed on clicking any marker. These info windows showcase the photos of the household uploaded by their owners, along with the availability of vehicles, number of days and the desired holiday dates of the selected household. Tourist spots nearby can also be searched.

Screen 2: Your Profile

The user profile contains details about the user such as name, address and landmarks. User can add images as they wish along with their availability, vehicles and any other notes or remarks. The status can always be toggled between active and disabled.

Screen 3: Your Wish list

The user can add houses into its wish list directly from the info window on the map. The enlisted locations are also highlighted on the map in the same way.

Screen 4: Your Messages

Users can contact any house owner directly through the map using a chat window. In this page, the location of the house owner who is being contacted currently is highlighted on the map.

Key Considerations

How will your website handle data persistence?
Firebase Realtime Database is used to store data. This data includes the user profiles, login credentials, messages received and wish list locations.
Describe any corner cases in the UX.
The user returns to the Search Page in the website if they hit the logo on any page. The map will be centered on the current location by default. The user can navigate between the wish list, profile and messages pages through the navbar on the top.
Describe any third party libraries you'll be using.
We use Google maps API to show the home switch enabled locations and tourist spots. React library will be used to create the project due to its usability, reusability of components and great developer tools.

Next Steps: Required Tasks

Task 1: Project Setup

1. Get a Google Maps Javascript API key by registering and creating a project.
2. Register and create a project for using Firebase Realtime Database.
3. Install create-react-app and create a react project. Run 'npm run build' to execute in the production environment and 'npm start' to execute in the development environment.

Task 2: Planning

1. Finalize all the functionalities to be added.
2. Plan the entire process methodically and figure out which functionalities can be grouped together into a single component.
3. Decide the flow of the application once all components are decided.

Task 3: Implement UI for Each Web Page and Frame

1. Create UI for map component and link it to the main App component.
2. Create UI for the list component.
3. Create UI for the login functionality, profile page and chat windows.

Task 4: Implement Google Maps

1. Make sure that maps are asynchronously loaded and errors are handled properly.
2. Implement markers for every home switch household and highlight them on the map.

3 Implement info window for every marker which displays all the required information.

4. Places google map library is used to find tourist spots nearby.

Task 5: Implement Database

1. Store the user details, wish lists, messages on the cloud database (Firebase Realtime Database) and retrieve accordingly.
2. Make sure the data is updated on any change.