NearBy

CSEE 5590 - Web & Mobile Programming

MERN STACK WEB APPLICATION

INCREMENT 3 REPORT

Team 2- The Pioneers

Vineeth Reddy Sheri

Yousef Almutairi

Kalyani Nikure

Piyush Narhire

Monday, November 15, 2021

Contents

OUR STORY

Our focus is to help people with the best way to enjoy travel adventures by finding the *NearBy* attractions for them. We design and implement a creative solution to the everyday problems of users who love to travel and explore places.

Our intent is to develop a MERN stack web application for finding places nearby and presenting them to its users. Our application aims to satisfy the needs of those who are looking for a simple solution for searching and saving the places they wish to visit. Any newcomer to the place would find this application useful for exploring nearby locations and managing in the profile these places for future reference.

We would like to stay connected with our users on social media (Facebook, Instagram, Twitter) to share new updates and understand more about their feedback and user experience. We are ecstatic to have this rare opportunity to meet and share experiences with like-minded people interested in exploring this world adventure with The Pioneers Team.

Features offered -

- Single Page Application (SPA) with responsive web design
- A variety of places are presented to the user as a result of their search query
- Advanced search options that include subcategories and a mile range
- Provides detailed information about the locations searched
- Next Venues and Similar Places suggestions with complete detailed information on each venue
- The Favorites tab that allows registered users to save liked places
- Video tutorials to help users learn how to use the application
- Quality of the overall user experience

THE DATA

Data management is a very crucial part of any web application. We want to protect user's information and also offer them the best information of their interest on the UI.

Ideally, we'd like to save the login information of registered users as well as their favorite places in a NoSQL-like MongoDB database. This is an open-source database that stores data in JSON like format.

All the information presented to users on the venues is fetched from the Foursquare API calls through our application.

The usage details of the database and Foursquare API is discussed in below table:

DB/API ELEMENT	Parameters	Use in Application

Collection - Users	Collection fields:
	{
	_id,
	firstname,
	lastname,
	emailAddress,
	password,
	city,
	state,
	zipcode,
	verficationCode,
	createdAt,
	updatedAt
	}
		We use users collection to store users' information provided by them with each user registration.
		
		This collection is also referred at the time of user login authentication. Once the user provides credentials on the User Interface, they are validated against the information in this collection. If these details match, user is redirected to the further session.

Collection - Favorites	Collection Fields:
	{
	_id,
	emailAddress,
	favourites: {Array:Venue Ids},
	createdAt,
	updatedAt
	}
		This Collection holds the information of favorited places by any registered user on the platform.
		
		The Array of Venue ID's is stored uniquely mapped with the username (email) of any particular user along with timestamp information.

Collection - Userverifications	Collection Fields
	{
	_id,
	emailAddress,
	verficationCode,
	createdAt
	}
		This collection comes into play when any user clicks on Forgot Password link and provides email address to send instructions.
		
		The collection then stores the verification code which is also sent to user for authentication and resetting the password. Later, the application verifies this code from the database to authorize reset password activity.
		
		Since the code is only valid for 4 minutes, the record gets removed after its expiry.
		
		The record expiry is mentioned in the collection schema to ensure automatic deletion of the record.

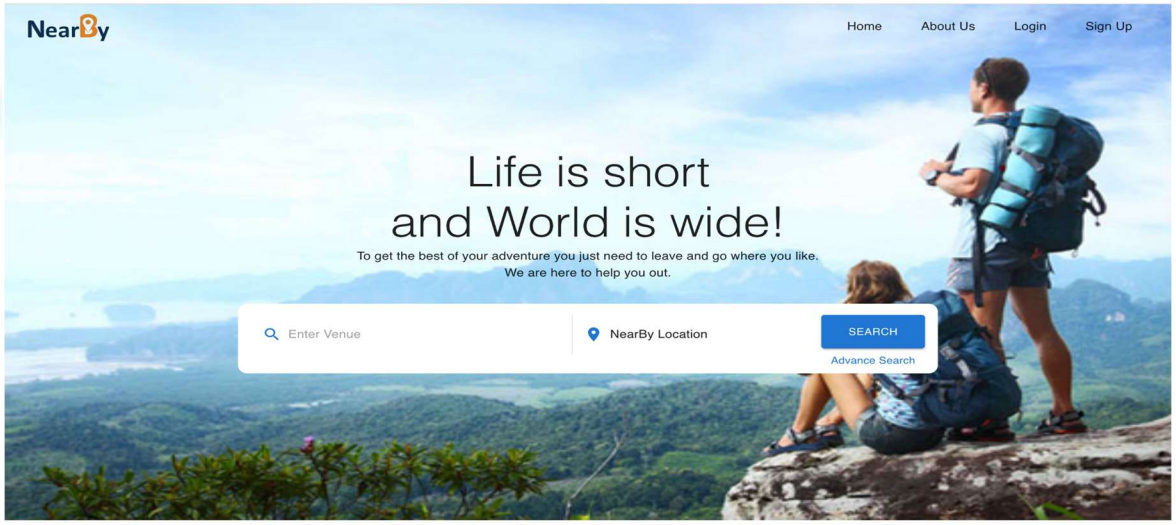
API- Foursquare	API Endpoints used https://api.foursquare.com/v2/venues/VENUE_ID https://api.foursquare.com/v2/venues/categories https://api.foursquare.com/v2/search/autocomplete https://api.foursquare.com/v2/venues/explore	We utilized various endpoints offered through Foursquare API to provide information about places on the user interface. Endpoint /venues/VENUE_ID provides information of a specific venue. Endpoint /venues/categories gets the list of categories and being populated in the search dropdown. Endpoint /venues/autocomplete will return all the possible categories and names of the venue based on your "search" string. Every time you enter a letter in the text field, the autocomplete API will be called and displayed on the screen for the user to select. If the search string itself is empty will display the general categories such as top picks, food, etc. Endpoint /venues/explore is called when the user selects any of the categories from the search (for example

		food category). The API will return all the venues that are related to food. This API will be called only when the user selects a category or search string. But not when the user selects the Venue directly in the search option.
	https://api.foursquare.com/v2/venues/VENUE_ID/nextvenues	Endpoint /venues/VENUE_ID/nextvenues provides a list of Next Venues related to selected Venue ID.
	https://api.foursquare.com/v2/venues/VENUE_ID/similar	Endpoint /venues/VENUE_ID/similar helps to get the list of similar location in relation to selected venue.

THE APPLICATION

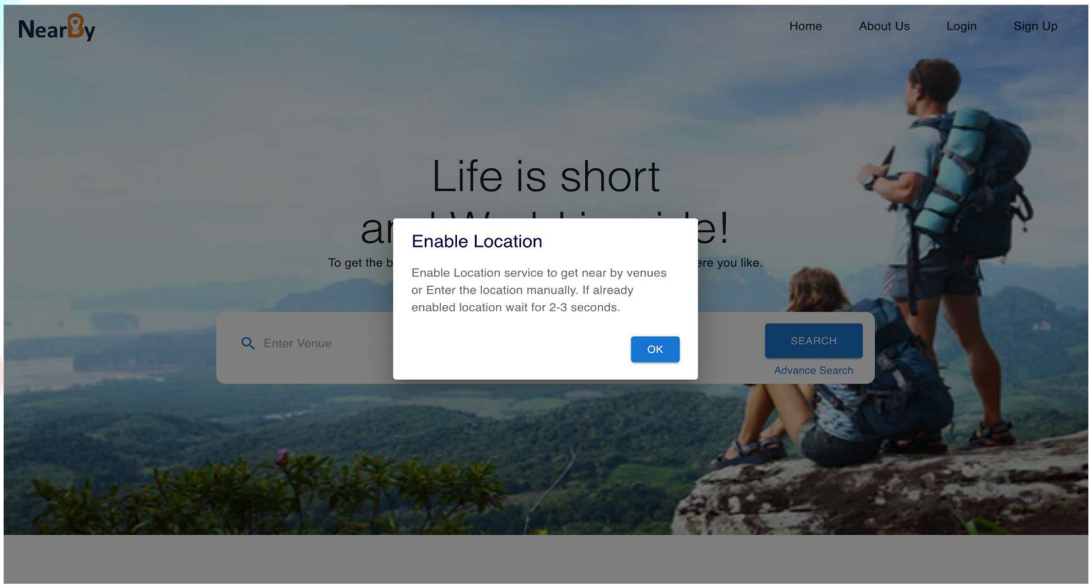
1. Home Page

When a user clicks on the web link to our application, he or she will be taken to the home page where the application is going to ask for your location. The screen below shows our default home page's navigation bar, which includes Home, About Us, Login, and Sign Up. Users can enter specific criteria and then specify a location to find the desired venue, and the results will appear on the screen. The user can keep up with us by using the icons in the footer screen to follow us on social media.

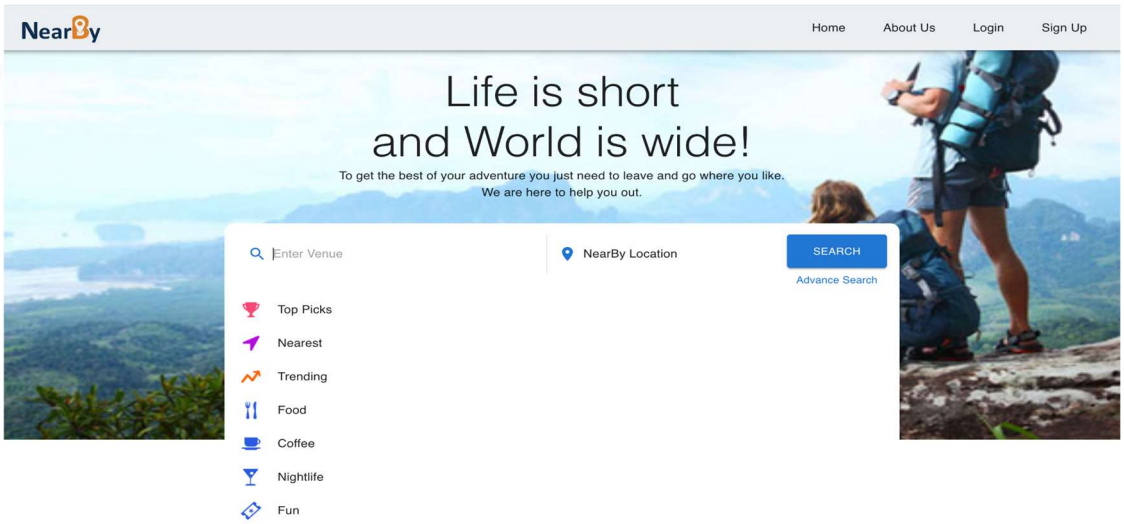


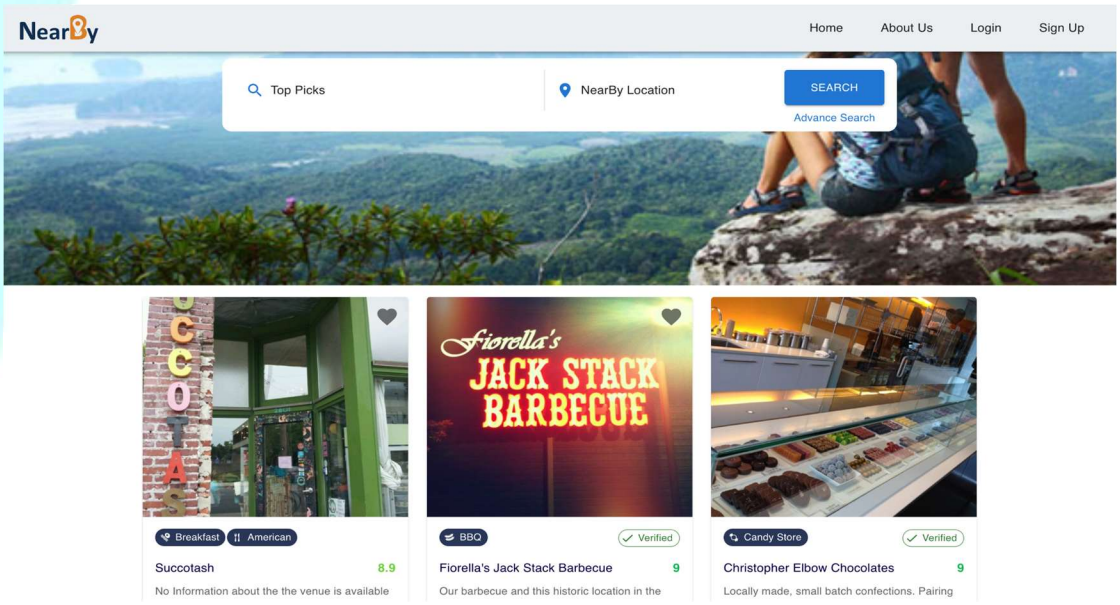
2. Search Functionality

If the user tries to enter the venue without giving a location, a 'Enable Location' dialog will appear; if the user does not wish to allow the current location, he/she must enter the required location and continue the search. In addition, when a user authorizes permission to access their location, the NearBy location tab takes the user's current location in order to provide results in the user's general neighborhood.



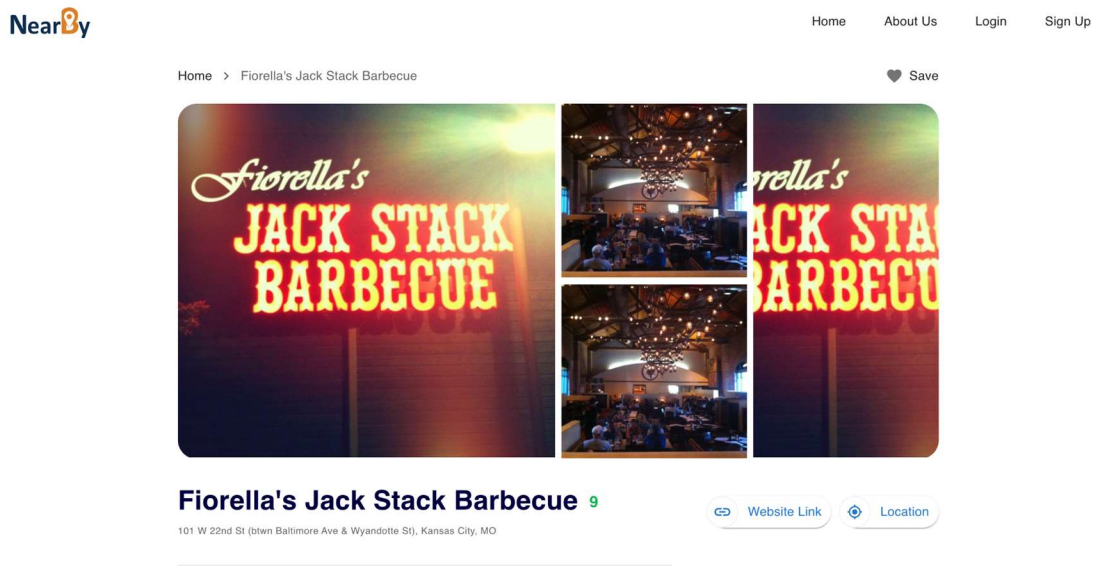
If the location is provided, the user can search by entering a venue name or selecting a category from the drop-down options. Advanced search allows users to narrow down their search by selecting a category, sub-category, and range of miles. For example, if the category is food, the sub-category is Mexican restaurant, and the range is 15 miles, the app will display the results accordingly.

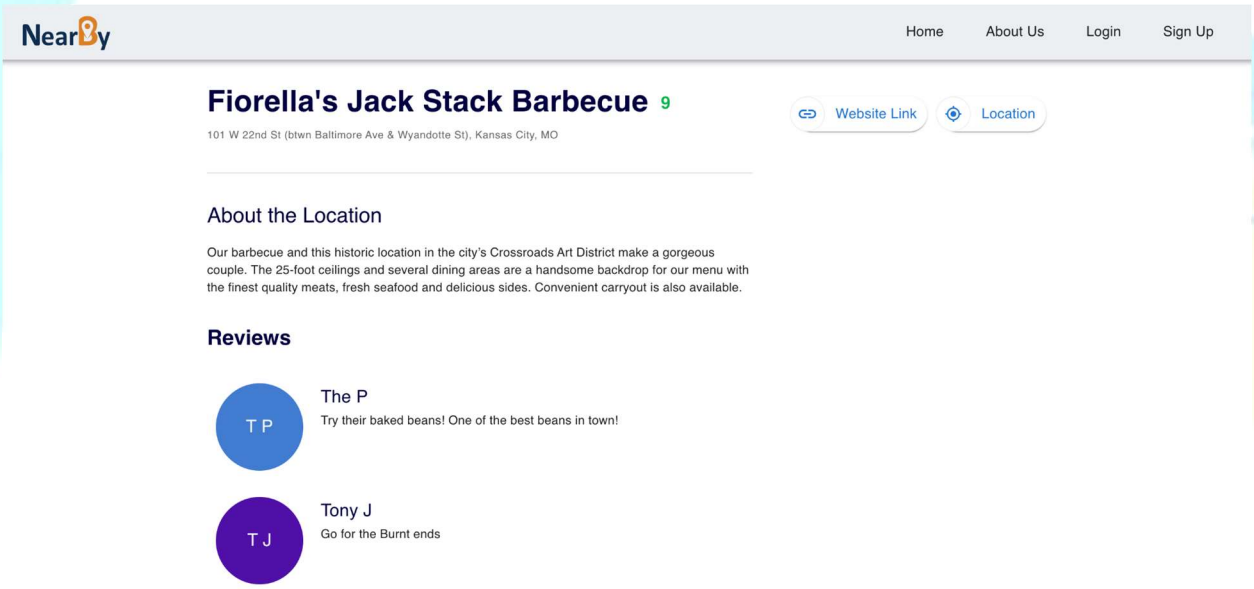




3. Venue Details Page

On the Venue Details Page, the user will be able to read more about the venue's details such as: a summary of the venue, a link to the venue's website, directions to the location (Google Map), recent user reviews, next venues, and similar locations. We're using the Foursquare API to get information about venue lists that match the search criteria of our users. Foursquare provides information such as the venue's name, address, popularity, images, tips, category, and a summary. Some of the endpoints used are venue/venue_id, /categories, /autocomplete, /explore, /nextvenues, and /similar.

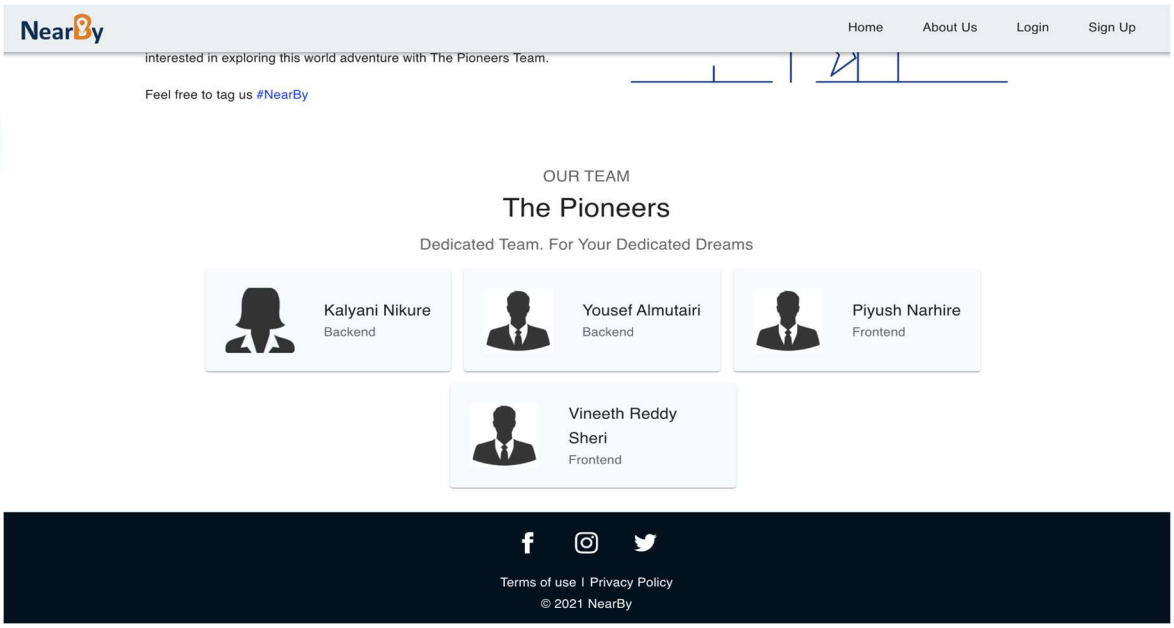




4. About Us

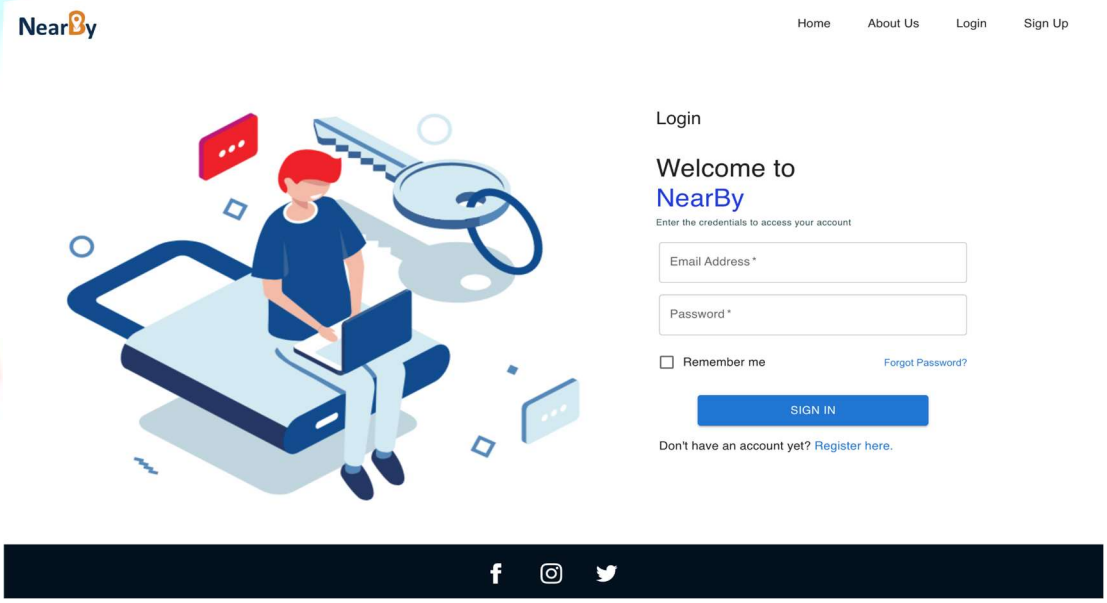
We are team "The Pioneers" studying at University of Missouri - Kansas City and we are working on our web application "Nearby". Our focus is to help people with the best way to enjoy travel adventures by finding the NearBy attractions for them. Our product aims to satisfy those who are looking for a simple solution for searching and saving the places they wish to visit.

This page helps users to know more about us. This tab helps users with a [User Guide Video] which tutors them with application usage with a demo of the overflow.

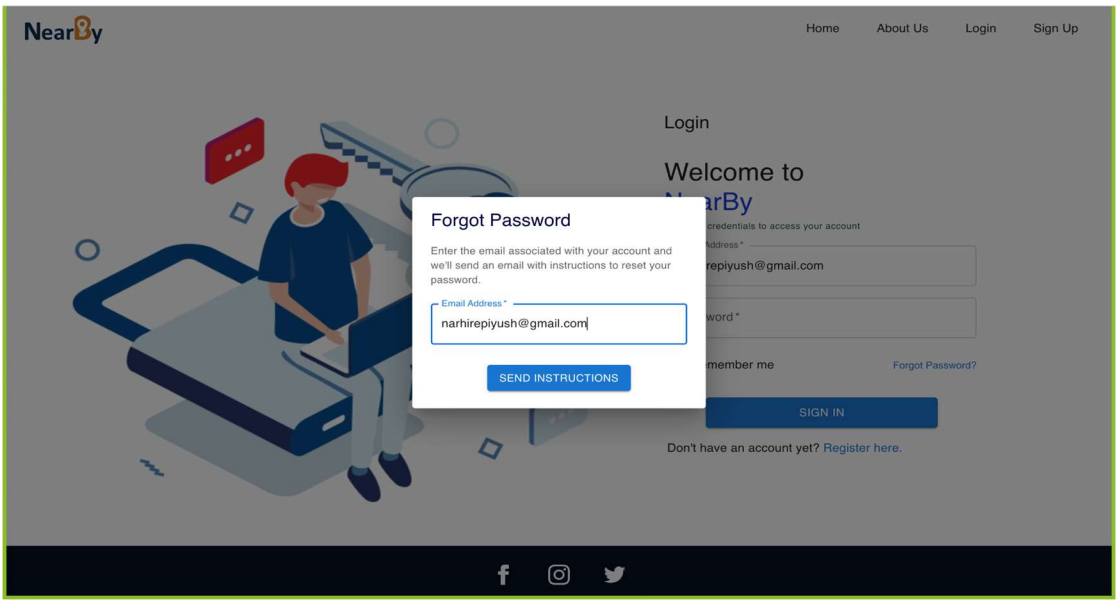


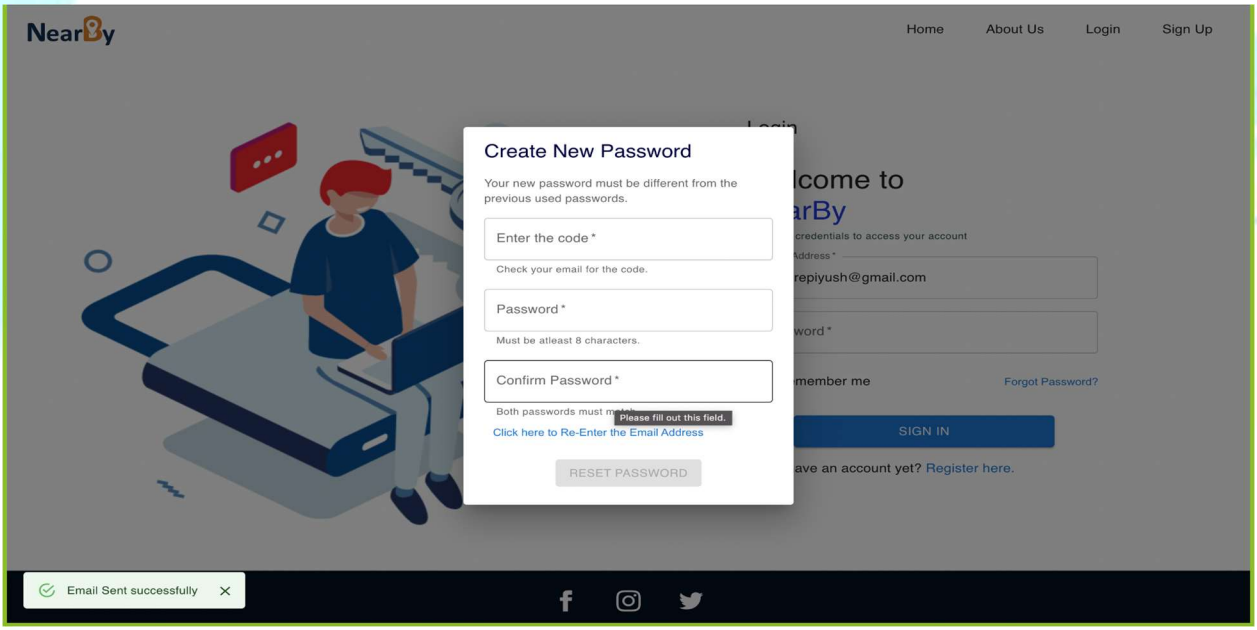
5. Login Page

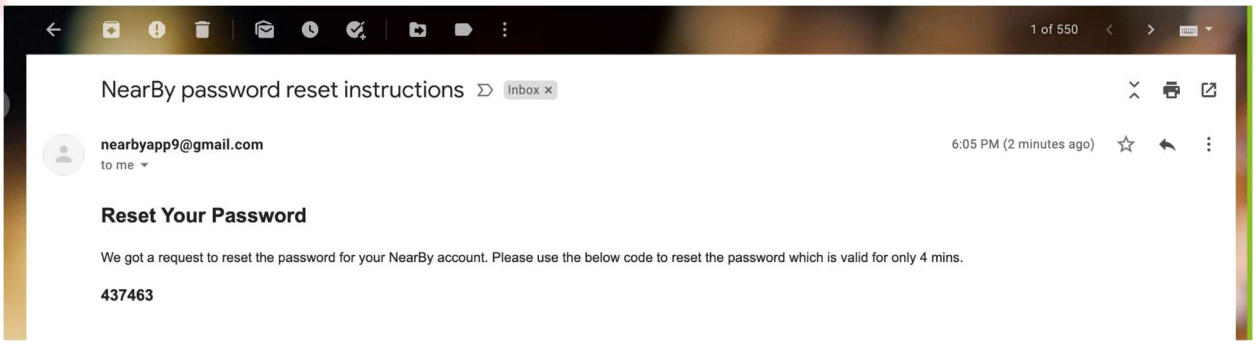
If the user already has an account with us, they can login to our application by entering their username (email) and password. If the user enters an incorrect username or password, an error message displays, stating that either the username or password is incorrect and that the user should try again. User information is crucial to us in order for us to keep users' data private and protected.



If a user forgets their password, we provide a forgot password feature. The user will receive an email with password reset instructions and a code for verification which allows them to change their password and try logging in again. The code will be valid for 4 minutes only. After logging in, customers can save their favorite venues so that they can visit them in the future by opening our app and accessing them.

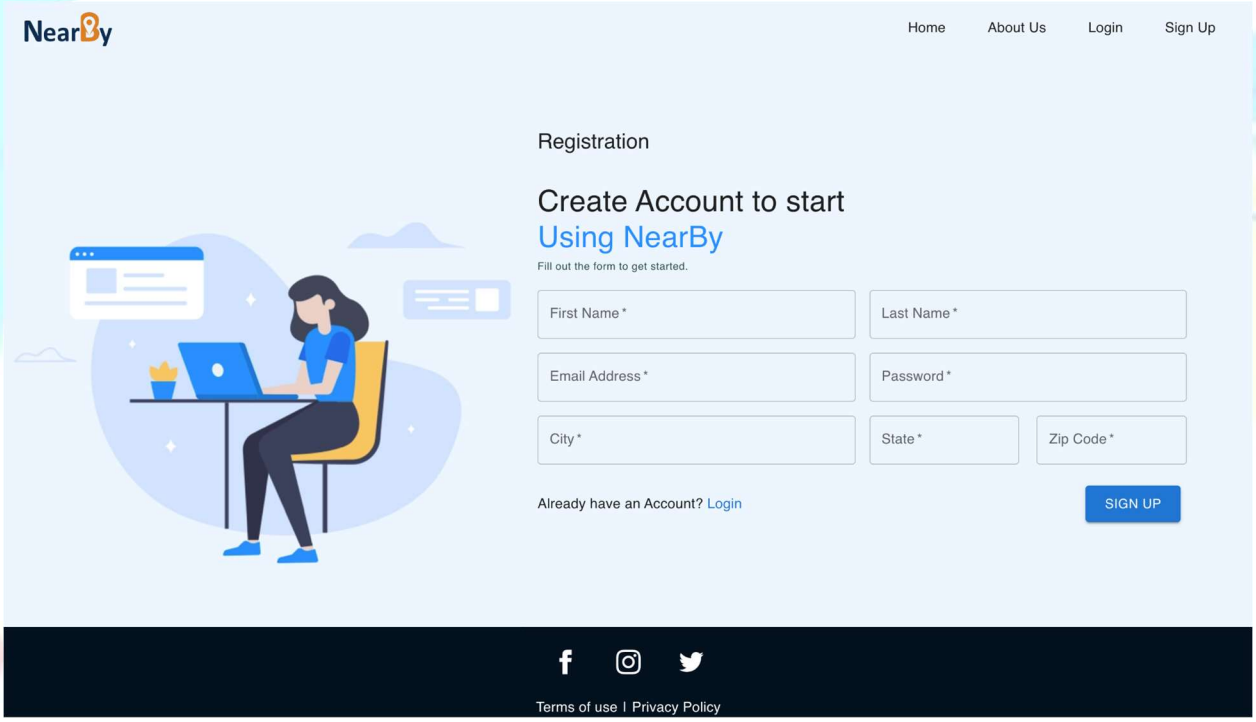






6. Sign Up Page

We've simplified the sign-up procedure so that users just must provide the bare minimum of information in order to register an account with us and take use of all of the services. Also, we ensure that the proper data is being recorded in the database by activating Client-Side Error Messages Handling. For example, the length of the password string does not satisfy the minimum criteria, the format of your email address is incorrect, or mandatory fields are missing. To favorite a location and save it for a later visit, users must first create an account on our web application.



MEMBER CONTRIBUTIONS

We have distributed our project activities among our team members based on Back-end and Front-end application development. The front-end development was proactively handled by Vineeth Reddy Sheri and Piyush Narhire to give a better user experience. Kalyani Nikure and Yousef Almutairi, on the other hand, worked on the back-end integration with the database and other tasks. Documentation for the project was done collaboratively by everyone.

CHALLENGES

As any other project, we faced multiple challenges in our project. One of the main challenges that we are facing every time is resolving the bugs and issues during the development process. These came as the static images are breaking while reloading in VenueDetails, Link Component Conflict between Material UI and React Router DOM due to the same naming convention, and UI scrolling issue, for instance, when the user scrolled down to the bottom of the homepage for example and want to go to about us page, it will start at the same point (bottom page)

In addition, FourSquare is providing 3 different packages for using their API. The free package that we are using for our project is limited to calling only 2 photos and 2 tips per venue which restrict us from providing more information.

THE GITHUB REPOSITORY

https://github.com/VINEETHREDDYSHERI/PlacesNearBy

CONCLUSION

This project aims to be part of people's travel adventures. We provide adequate and relevant information on the venues in order to provide travelers with the information they need. In order to get our users to the best places, we tried to match their preferences. Additionally, Next Venues suggests other potential destinations for users to visit. As well, they can select places that have similar specifications to those they are interested in from the similar location suggestions.

Moreover, since we place a high priority on the user experience, we strive to offer platforms that connect us to our users. For connecting with our users and sharing the latest updates and features, we have created social media accounts on Facebook, Twitter, and Instagram. This provides us with a unique opportunity to stay connected and also from a community of people who are interested in exploring the world together.

Although we tried to make the best of our efforts by creating this application for our users, we will be focusing our future efforts on addressing the challenges we have mentioned in the Challenges section and adding features that allow users to interact with the website at their convenience and get accurate information about places they would like to visit.

REFERENCES

1. https://docs.mongodb.com/manual/core/data-model-design/
2. https://developer.foursquare.com/docs/places-api/
3. https://spectralops.io/blog/yelp-api-guide/
4. https://mui.com
5. https://reactjs.org
6. https://nodemailer.com/usage/
7. https://reactrouter.com/docs/en/v6
8. https://expressjs.com/en/starter/basic-routing.html
9. https://developer.foursquare.com/docs/places-api/endpoints/