

NEW YORK INSTITUTE OF TECHNOLOGY

Final Report

Project Title: City Scope

CSCI-455: Senior Project

Spring 2023

Prof. Maherukh Akhtar

Team 2 – Overwatch

Jeffrey Aguilar - 1265967 jaguil06@nyit.edu

Alexander Pachnicki - 1230402 apachnic@nyit.edu

Jhoselyn Moro-Romero - 1263800 jmororom@nyit.edu

O'Sean Blagrove - 1187413 oblagrov@nyit.edu

TABLE OF CONTENTS

1.	Abstract	6
1.1.	Introduction	6
1.2.	Existing Systems	7
1.3.	Proposed System	10
1.4.	Software Engineering Model	11
1.5.	Purpose	11
1.6.	Project Objective/Goals	12
1.7.	Technologies and Tools	12
1.8.	Users	13
1.9.	Timeline	15
2.	Diagrams	20
2.1.	Sequence Diagram	20
2.2.	Context Diagram	21
2.3.	Data Flow Diagram	22
3.	Database Design	23
3.1.	Table Schema	23
3.2.	Entity-Relationship Diagram	24
4.	Functionality and Implementation	25
4.1.	Features	25
4.2.	Security	26
5.	Testing	26

6.	Conclusion	29
6.1.	Results	29
6.2.	Limitations	33
6.3.	Future Enhancements/Recommendations	35
6.4.	Team Members/Roles	36
6.5.	Learning Outcomes	36
7.	References	38

TABLE OF FIGURES

Figure 1. Examples of Yelp's UI	7
Figure 2. Examples of Visit A City's UI	8
Figure 3. Examples of Like A Local's UI	9
Figure 4. Burndown Chart #3	16
Figure 5. Burndown Chart #4	16
Figure 6. Burndown Chart #5	17
Figure 7. Burndown Chart #6	18
Figure 8. Burndown Chart #7	18
Figure 9. Burndown Chart #8	19
Figure 10. Burndown Chart #9	20
Figure 11. Sequence Diagram #1	20
Figure 12. Sequence Diagram #2	20
Figure 13. Context Level Diagram	21
Figure 14. Data Flow Diagram	22
Figure 15. Table Schema	23
Figure 16. Entity-Relationship Diagram	24
Figure 17. Initial Home Screen	27
Figure 18. Initial Sign-Up Screen	28
Figure 19. Initial Login Screen	28
Figure 20. The Map Tab	30
Figure 21. The UI when a user searches for a nearby location	31
Figure 22. The UI when a user taps on a card to reveal additional info	31

Figure 23. The UI when a user taps the card on the left to reveal reviews + photos 32

Figure 24. The Library Tab 33

ABSTRACT

We have designed a hybrid app using Flutter/Dart that provides services and recommendations for the best places to go and things to do around a user's location. Our application is designed to give users the best ability to explore and view the cities that they travel to. We believe that the key for users to have the best experience traveling and exploring is simplicity and maintaining and perfecting the key elements of what is important for the user. In this case, the most important elements of our app are the location services (e.g live map and location tracking) and the ability to view and rate local businesses. The app also recommends the best local places for the user to visit based on the ratings given by users and previous data taken from the Yelp API.

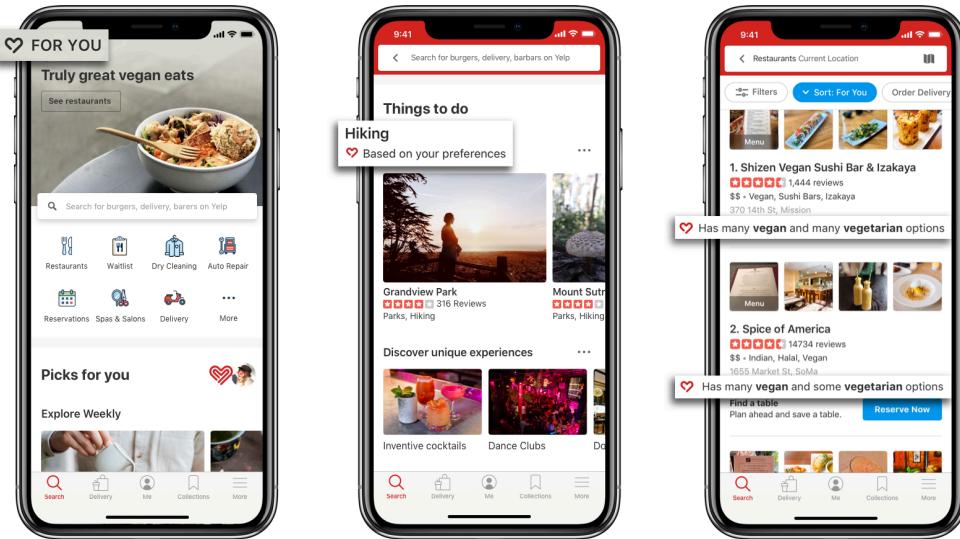
1.1 - INTRODUCTION

Our team convened and discussed what our skill sets were in order to see what tools/technologies we were all experienced and familiar with. This is important for the development of the app because it's important that we all come to an agreement of what tools/technologies we will all feel comfortable working with in order to make the development of the app as seamless as possible. The team agreed on using Flutter/Dart to develop the UI of the app and using Google's Firebase for the backend. Next, we planned out further details on what is necessary in order to develop this app (e.g system diagrams such as context level diagrams, data flow diagrams, database modeling, etc). Once everything was planned out and analyzed, we were able to move forward with database design and the construction of the app's UI. Our team was using the iterative model during the development of this app. We applied the Scrum technique in

order to help communicate/collaborate with each other and better structure and manage the workload. Finally, we were also in contact with our advisor Professor Akhtar via weekly Scrum meetings in order to review our progress as development of the app was progressing.

1.2 - EXISTING SYSTEMS

Yelp



Star rating as of August 12, 2019

(Fig. 1. Examples of Yelp's UI - Blog, Yelp – Official, and Akhil Kuduvalli Ramesh. “You Can Now Align the Yelp App to Your Lifestyle, Diet, Interests & Other Preferences.” Yelp, 1 Sep. 2021, Digital Image)

Yelp is an app that is essentially an online directory for users to discover and review local businesses. This includes businesses such as bars, restaurants, cafes, hairdressers, gas stations, etc. Businesses are listed and sorted by business type and results are filtered by location, price

range, and other features such as outdoor seating, delivery, or the ability to accept reservations. Users are encouraged to leave ratings, written reviews, and even take pictures of their experience of the places they visit. Our app will work similarly to Yelp except we will be showing a live map at all times to the user. This will allow the user to discover new locations to visit at all times when they open the app in real time.

Visit A City

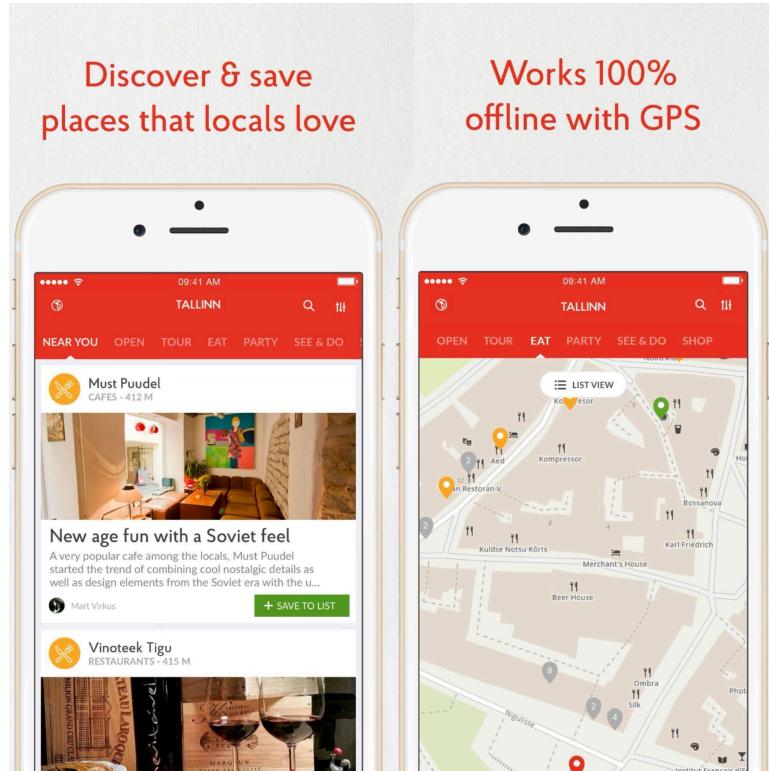


(Fig. 2. Examples of Visit A City's UI - Roberts, Dawna M. "The 14 Best Trip Planner Apps for Your Perfect Vacation in 2020." Compare and Choose The Best One, Compare and Choose The Best One, 11 Feb. 2021, Digital Image)

Visit A City is an app that works as an online travel guide. With this app, you are able to create a customized itinerary and book tours and activities for users, discover famous landmarks and local attractions, and are presented with detailed information and reviews in order to find the best

experience for the user. This app seems more catered towards tourists who are traveling rather than presenting locals new experiences found near them.

Like A Local



(Fig. 3. Examples of Like A Local’s UI - Rizzo, Cailey. “5 Apps For Exploring a New City Like A Local.” Mashable, Mashable, 29 Oct. 2021, Digital Image)

Like A Local is a travel app that presents users with city guides to different cities around the world such as New York City, Amsterdam, Athens, Berlin, Buenos Aires, and etc. The app presents users with local reviews of the city, a Q&A section between tourists and locals, and the ability to book tours and save places to visit for the future. The app also has a map of the city that works in real time that displays places that are open/accessible wherever and whenever a user is browsing. However, an issue this app has is that many of the locations recommended by the app are mainly directed towards tourists and fail to truly recommend local hidden gems.

1.3 - PROPOSED SYSTEM

The app will open to a login/sign-up screen. Depending on what the user selects, different features will be presented to the user. If the user is a first-time user, then they will select the sign-up button which will present them with a sign up screen. The sign up screen will ask for the user's name, password, and email. Once a user has a registered account, they will be directed to the app's home page. The home page will automatically take in the user's location and present them with a Google Maps view of their present location.

The app will automatically show the user the best local spots and things to do nearby. Users can also look up locations near a certain area and will be presented the best local spots and things to do nearby the area the user has inputted. Users will be able to leave ratings and reviews on the locations they have visited and be able to indicate which were their favorites. Users will also be presented with reviews from other users who have visited the local spots that were recommended to them.

We also plan to implement the Yelp API in order to get initial reviews into our database. With enough ratings and reviews, we could look into implementing a recommendation system in our app that can recommend users the best places to visit near them. Since a potential recommendation system would work off of other user's ratings, we could construct a collaborative filtering system that will provide users recommendations that were rated highly by similar users.

1.4 - SOFTWARE ENGINEERING MODEL

The software engineering model we plan to use to complete our project in the most efficient way possible will be the **iterative** software engineering model. This model suits our project perfectly because we can develop a basic program which accomplishes the goals and meets the requirements our project needs in order to function.. After the first iteration/version of the app, we can develop a newer version of the application that will address issues found in the previous version and be able to focus on designing, testing and implementation, rather than starting over from the beginning. This is the main reason we are choosing the iterative software model. Overall, this model is ideal for us since we can make a rough version, which simply accomplishes the goals of our application, and then design and focus on polishing that version even more and repeating those steps until we are satisfied with the final version.

1.5 - PURPOSE

We want users to be able to explore their local area in order to promote not only local businesses and shops, but also promote users to go out and discover a sense of adventure in knowing that there are many places in their local area that they might have not known existed if not for this app.

1.6 - PROJECT OBJECTIVE/GOALS

- Login/Sign Up Screen (w/ Forgot Password Feature)
- Home Page w/ Google Maps API
- Live Location Tracking
- Search Bar
- Recommended Tab (Will Show Ratings and Reviews)
- Favorites Tab
- History Tab
- Profile Page

1.7 - TECHNOLOGIES AND TOOLS

The tools and technologies we used include the following:

1. **Flutter/Dart** - An open source software for User Interface development, framework.
2. **Android Emulator** - The Android Emulator simulates Android devices on your computer so that you can test your application on a variety of devices and Android API levels without needing to have each physical device.
3. **iOS Simulator** - An application provided by Apple to test, build, and run apps on a simulated or real device. Simulated devices help debug apps on a variety of hardware developers may not have immediate access to. The tradeoff is that simulated devices run within the Simulator app on Mac and don't replicate the performance or features of an actual device.

4. **Google Firebase** - A toolset that gives a large portion of services such as analytics, authentication, databases, configuration, file storage, push messaging, and etc. These services are hosted in the cloud.
5. **Google API** - Allows the program to access interfaces to Google Cloud Platform services. For this project, we mainly integrated Google Maps API, Directions API, and the Places API.
6. **Yelp API** - Allows the program to get the best local content and user reviews from millions of businesses around the world provided by Yelp.
7. **Discord** - A communication app to communicate with team members and coordinate ideas
8. **Google Docs / Slides** - A web based service to allow all team members to edit and work on documents and presentations.
9. **Github** - code hosting platform so our group can share code and project progression

1.8 - USERS

The desired user base for our application will be teens to adults who want to discover and visit new local locations such as restaurants, cafes, shops, etc. Through the use of this application, users will be able to venture out into their local area with friends, family or by themselves and be able to find new ways to spend their time. Users will also be able to leave ratings and reviews about the places they have visited. This will help our system recommend users the best locations to visit.

Manager: While using this app, managers will be able to analyze the application's systems and determine which software applications could improve efficiency. Managers will also be able to:

- Make recommendations on whether to upgrade the existing systems or install new ones.
- Monitor the roll-out of new software applications to ensure there are no problems.
- Troubleshoot and resolve any problems within the app.
- Create and oversee protocols and procedures for the use of any new software applications.
- Create, execute, and maintain company databases.
- Maintain up to date knowledge of the latest software developments.

Admin: While using this app, admins will be able to view and manage user permissions in the application. Admins will also be able to:

- Manage profile editor
- Manage profile mappings
- View users and groups

1.9 - TIMELINE

The project was developed using Scrum methodology. Our team met to discuss our progress every Tuesday at 2:30pm during the project's development.

SPRINT 1

(2/1/23 - 2/5/23)

Planning and Communication - Discussion and how we plan to handle the development of our app (what tools/technologies we will use and familiarize ourselves with everyone's knowledge and skills). We will also begin documentation.

SPRINT 2

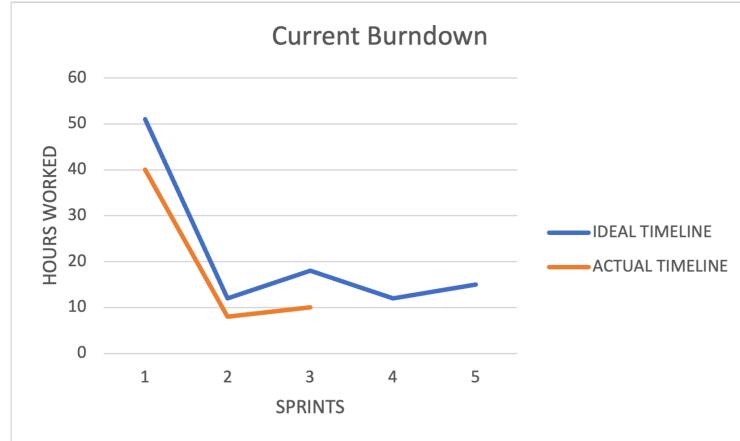
(2/6/23 - 2/20/23)

Analysis and Database Design - All team members have downloaded Flutter and are comfortable with the language and coding environment. Visualization of app design is created along with initial ERD.

SPRINT 3

(2/22/23 - 3/8/23)

Code - Team members will now begin coding and team members will begin working on each individually assigned role whether it be frontend, backend, or database management. Initial UI design for pages (login, sign-up, home page, etc.) and established navigation routes.

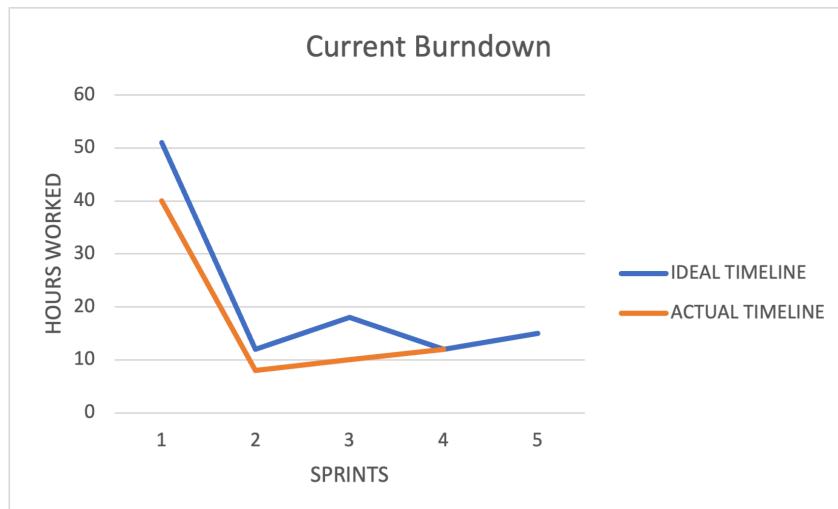


(Fig. 4. Burndown Chart #3)

SPRINT 4

(3/9/23 - 3/12/23)

Code - Added Google Maps API w/ functioning search bar, a sliding widget overlaying the map screen (the sliding widget will ideally display the closest locations for a user to visit), updated UI (layout and color scheme), and configured Firebase to work on iOS devices.

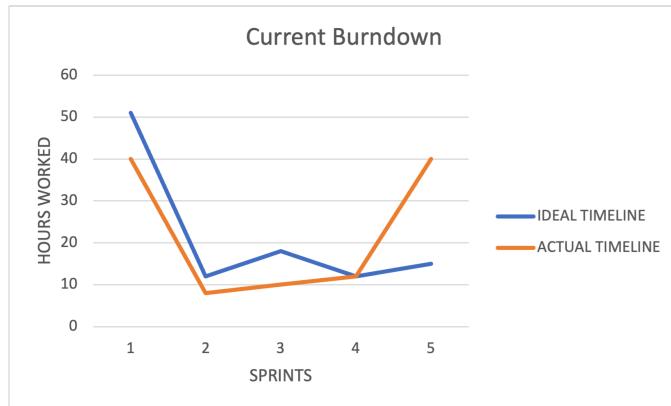


(Fig. 5. Burndown Chart #2)

SPRINT 5

(3/13/23 - 3/17/23)

Testing & Documentation - Initial testing of the app will begin. Debugging and resolving any issues that occurred will be looked into. We will also discuss what else we can add moving forward.

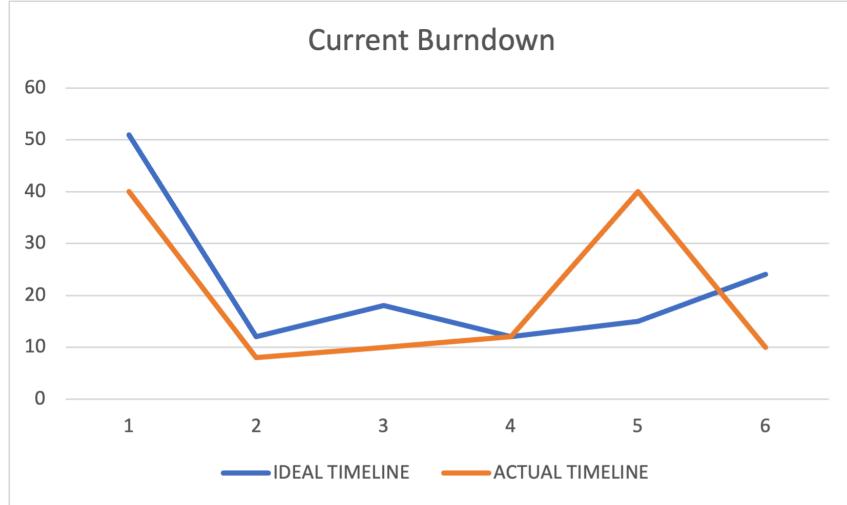


(Fig. 6. Burndown Chart #5)

SPRINT 6

(3/18/23 - 4/5/23)

Code - Continued development by working on implementing live location features, configured Firebase for Android devices and successfully connected Firebase to the frontend, added initial code for searching businesses using the Yelp API.

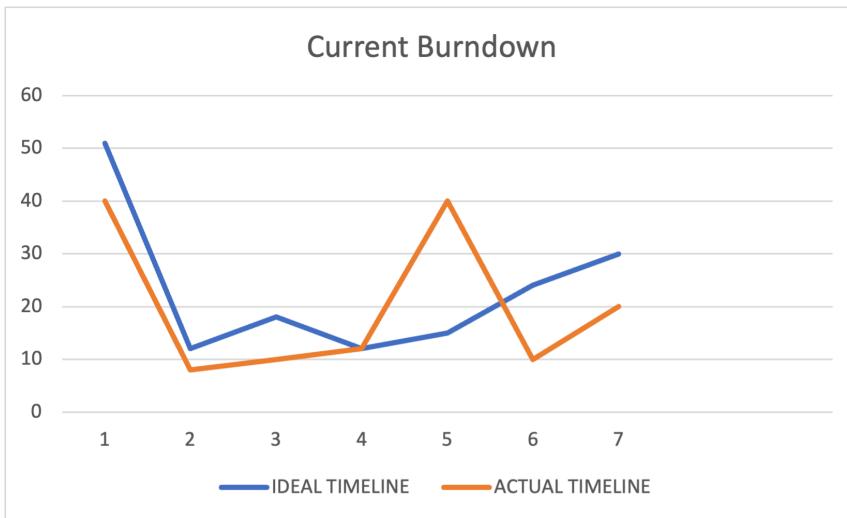


(Fig. 7. Burndown Chart #6)

SPRINT 7

(4/6/23 - 4/25/23)

Code - Added Directions API and Google Places API and successfully implemented a find directions feature. Implemented a feature where a user taps on the map in order to display a circle to indicate the radius of the location they want to explore.

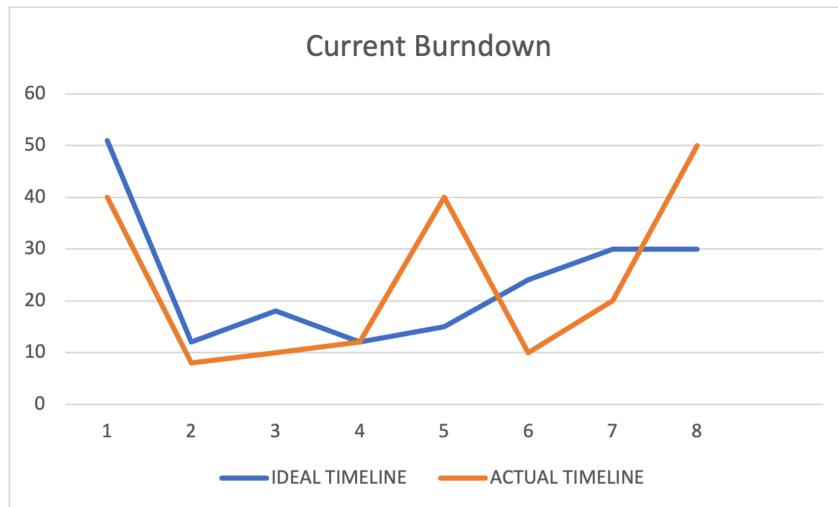


(Fig. 8. Burndown Chart #7)

SPRINT 8

(4/26/23 - 5/3/23)

Testing, Documentation, and Code - Second round of testing with an audience in order to generate feedback on what needs improvement. Using this feedback, we began working to complete our app's purpose. Began working on the final presentation and technical report.

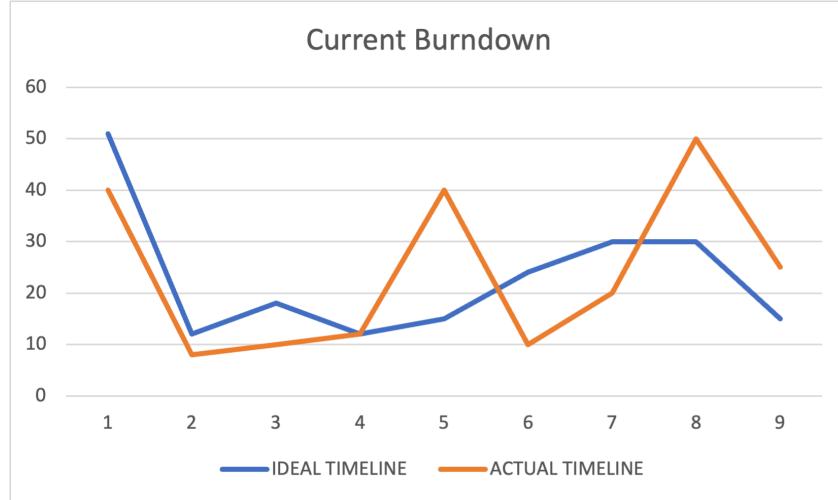


(Fig. 9. Burndown Chart #8)

SPRINT 9

(5/4/23 - 5/9/23)

Deployment and Evaluation - Final version of the project will be complete and ready to be displayed. We start our final preparations for the project and practice getting ready for the final presentation.



(Fig. 10. Burndown Chart #9)

DIAGRAMS

2.1 - SEQUENCE DIAGRAM

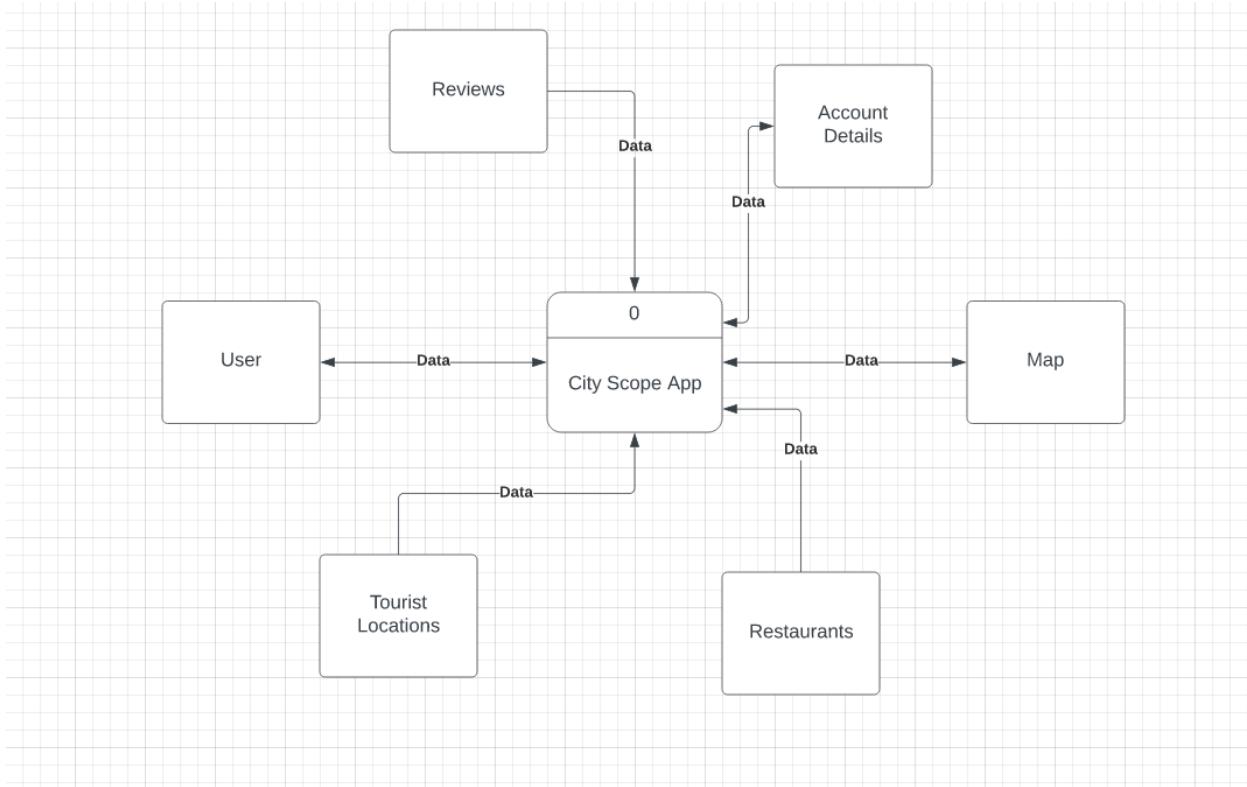
<https://imgur.com/a/ptc9sFn>

(Fig. 11. Sequence Diagram #1)

<https://imgur.com/a/chWYLGR>

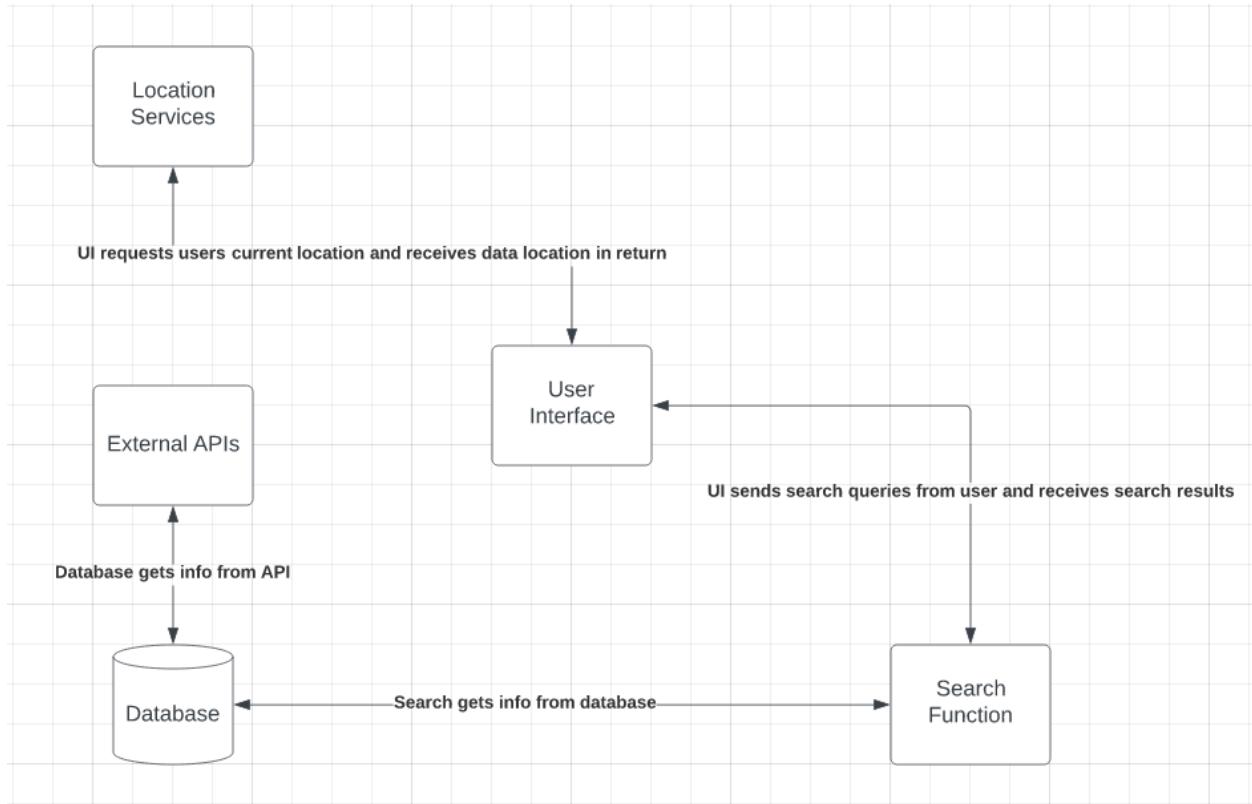
(Fig. 12. Sequence Diagram #2)

2.2 - CONTEXT LEVEL DIAGRAM



(Fig. 13. Context Level Diagram)

2.3 - DATA FLOW DIAGRAM



(Fig. 14. Level 0 Data Flow Diagram)

DATABASE DESIGN

3.1 - TABLE SCHEMA

Restaurants table:

Column Name	Data Type	Description
<code>id</code>	INTEGER	Unique identifier for the restaurant
<code>name</code>	VARCHAR(255)	Name of the restaurant
<code>address</code>	VARCHAR(255)	Street address of the restaurant
<code>city</code>	VARCHAR(255)	City where the restaurant is located
<code>state</code>	VARCHAR(255)	State where the restaurant is located
<code>zipcode</code>	VARCHAR(10)	Postal code of the restaurant
<code>phone</code>	VARCHAR(20)	Phone number of the restaurant
<code>website</code>	VARCHAR(255)	Website of the restaurant
<code>rating</code>	DECIMAL(2,1)	Average rating for the restaurant
<code>pricerange</code>	VARCHAR(10)	Price range of the restaurant (\$, \$\$, \$\$\$)
<code>cuisine</code>	VARCHAR(255)	Type of food served at the restaurant
<code>description</code>	TEXT	Description of the restaurant

Activities table:

Column Name	Data Type	Description
<code>id</code>	INTEGER	Unique identifier for the activity
<code>name</code>	VARCHAR(255)	Name of the activity
<code>address</code>	VARCHAR(255)	Street address of the activity
<code>city</code>	VARCHAR(255)	City where the activity is located
<code>state</code>	VARCHAR(255)	State where the activity is located
<code>zipcode</code>	VARCHAR(10)	Postal code of the activity
<code>phone</code>	VARCHAR(20)	Phone number of the activity
<code>website</code>	VARCHAR(255)	Website of the activity

Places table:

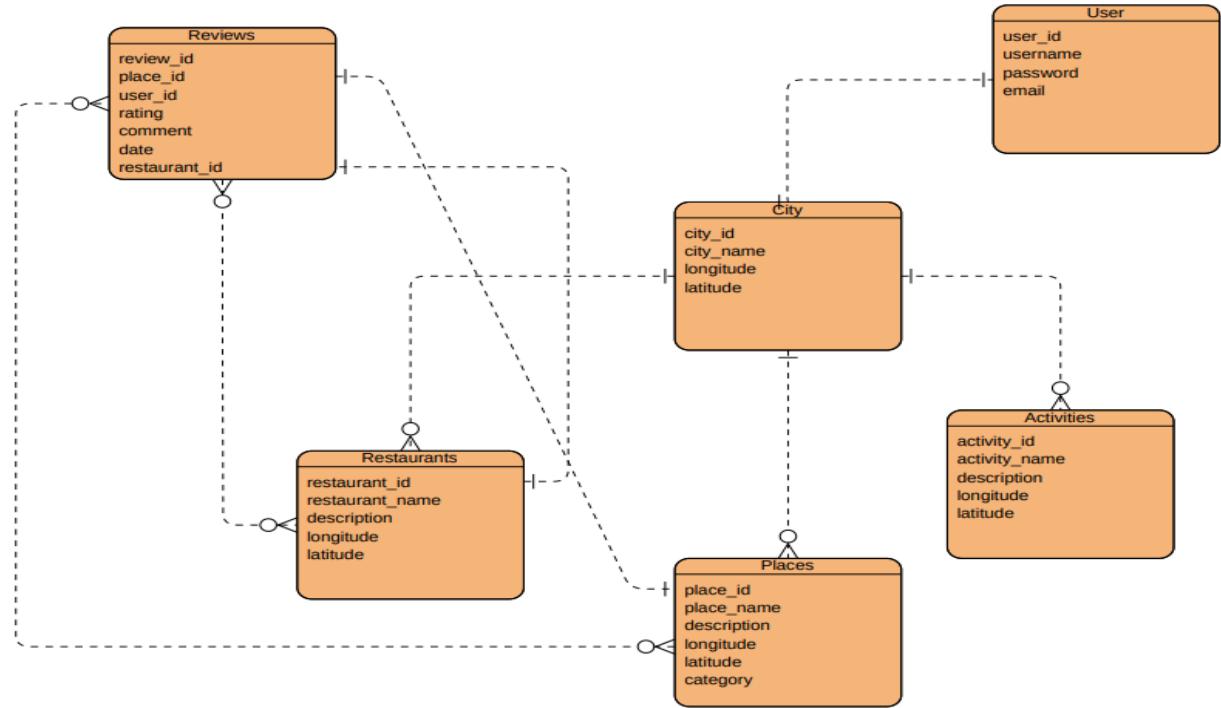
Column Name	Data Type	Description
<code>id</code>	INTEGER	Unique identifier for the place
<code>name</code>	VARCHAR(255)	Name of the place
<code>address</code>	VARCHAR(255)	Street address of the place
<code>city</code>	VARCHAR(255)	City where the place is located
<code>state</code>	VARCHAR(255)	State where the place is located
<code>zipcode</code>	VARCHAR(10)	Postal code of the place
<code>phone</code>	VARCHAR(20)	Phone number of the place
<code>website</code>	VARCHAR(255)	Website of the place
<code>rating</code>	DECIMAL(2,1)	Average rating for the place
<code>type</code>	VARCHAR(255)	Type of place (historical site, landmark, scenic view)
<code>description</code>	TEXT	Description of the place

Text Predictions: On Accessibility: Investigate

Focus
List
Grid
Search
Help

(Fig. 15. Table Schema)

3.2 - ENTITY-RELATIONSHIP DIAGRAM



(Fig. 16. Entity-Relationship Diagram)

FUNCTIONALITY AND IMPLEMENTATION

4.1 - FEATURES

- Login, Sign-up, and Forgot Password Screen
- A library tab that displays a user's username and allows users to upload profile pictures. The library tab also includes buttons that would lead them to an edit account screen, a past reviews screen, a favorites screen, and the sign out button.
- A map screen that shows the user's location with a functioning search bar and find directions button. When a user taps on the map, a circle that indicates how wide the user would like to search for locations appears. The circle's radius is adjustable so users can search for locations 100 meters to 1000 meters.
- When the user indicates how wide the search radius should be, they click the search button on the radius slider and are shown cards at the bottom of the screen that indicate nearby locations as well as their corresponding pins. Users can also press the refresh button to load more locations to be displayed as each search only pulls a certain amount of results depending on how many locations are near the user.
- The cards at the bottom of the screen displays the name of the location, the rating, whether it's operational or not, and a corresponding picture pulled from Google.
- When a user presses the card, another card appears on the left side of the screen that displays the address, contact information, and photo of the location. When a user presses that card again, the card flips to reveal a list of reviews and more photos of the location.
- A functioning home page where you can switch between the map and library tab

4.2 - SECURITY

Our app uses FireBase to store the data for our consumers. Most, if not all, Firebase services encrypt data using “HTTPS and logically isolate customer data.” This means that the data directly related to a customer is often encrypted in the same algorithm, but using a different key. This way, even if data was somehow intercepted, the actual data cannot be directly read, preserving the integrity of the data. Firebase access is thoroughly monitored, where users that access the Firebase database must have 2-factor authentication and have Google accounts. Firebase also allows for rule-based data access limitations, meaning it is entirely possible to set up access controls such that employees may only access sections of the database relevant to their field of work.

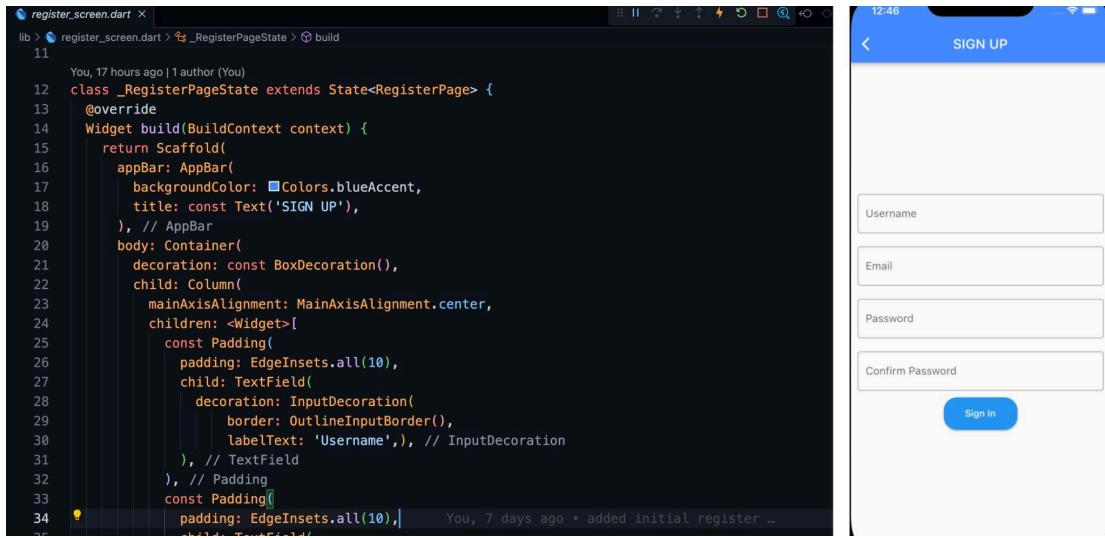
TESTING

Throughout this project’s development, our main method of testing was unit testing. Since Flutter offers a “hot reload” feature which allows users to test changes to their applications in real time, we were able to test our app’s functionality each time we made a change. We also performed black box testing during our app’s development. For example, we had to test whether or not a user would be able to log in if they successfully registered an account. Due to our various phases of testing, we were able to recognize what went wrong and what needed to be changed.



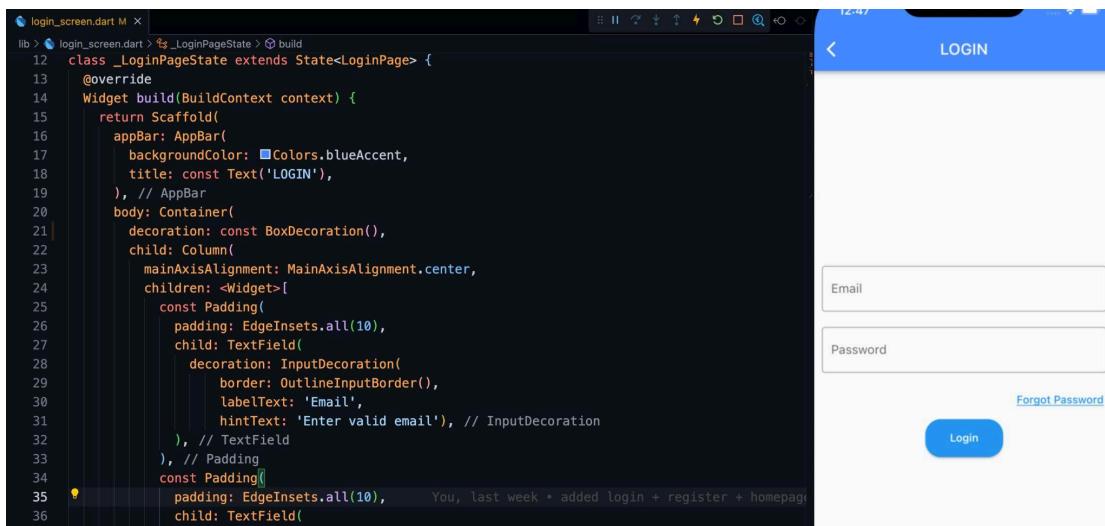
(Fig. 17. Initial Home Screen)

When initially designing the app's home screen, we had two buttons that led users either to a login or sign-up screen. We had to ensure that the buttons were working and able to send a user to the correct screen. While testing the buttons, sometimes the button wouldn't send us to the next screen so we had to play around with the Navigator class in Flutter. However, during the development process, we decided to forgo this idea for a home screen as it was redundant. Instead, we decided to let the app open into the login screen with text buttons highlighting if they want to sign-up or if they forgot their password.



(Fig. 18. Initial Sign-Up Screen)

When initially designing the app's sign up screen, we had to ensure that the information provided by users were correctly stored in Firebase in order for users to be officially registered in the system. While testing, we ran into an issue where users were unable to sign up because the method we used to store a user's ID would only work a small percentage of time. This was strange since the initial testing for the sign-up process worked just fine. Unfortunately, we weren't able to find a solution to this problem so we ended up removing the method that stores a user's ID into Firebase.



(Fig. 19. Initial Login Screen)

When designing the app's login screen, we had to test if information inputted by users allowed them to sign in if their account was registered. During testing, the login process was able to work. However, a problem that we encountered was that users weren't immediately brought into the home screen and required the app to restart. Only then were users immediately brought into the home screen when the app was opened again. After fixing these initial issues, we were able to continue with the development of the project and move forward starting with a change of the UI, adding the rest of the features, and looking more into the overall app structure.

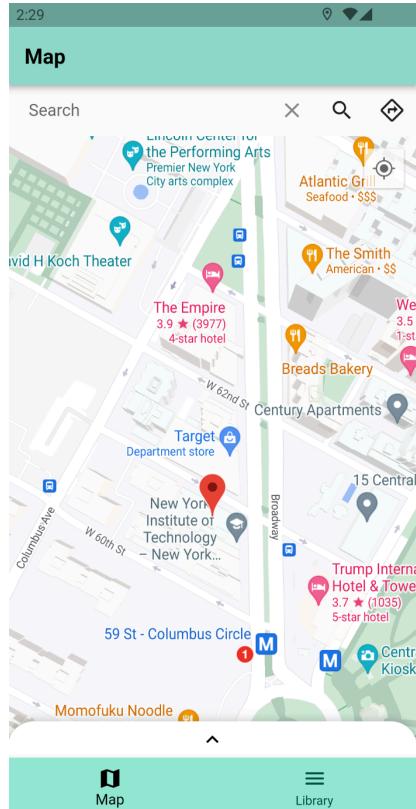
CONCLUSION

6.1 - RESULTS

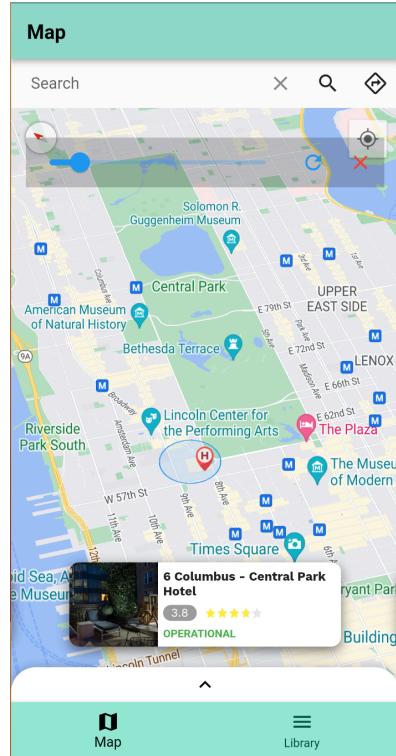
Our initial goal was to develop an app that provides services and recommendations for the best places to go and things to do around a user's location. We also wanted the app to allow users to rate and be able to leave written reviews on the places they visited. Additionally, the app would recommend the best local places for the user to visit based on the ratings given by users and previous data taken from the Yelp API.

After about 3 months of developing this app, we were able to develop it to a point where it successfully met its intended purpose. When a user signs in or is already signed in, they are first met with a Google map. When a user taps on the map, a circle appears that indicates how wide of an area the user would like to search for locations to visit. When users begin to search for locations near them, a horizontal list of cards appears that detail a location's information, rating, reviews, and a photo of the location. When the user clicks on a card, another card appears

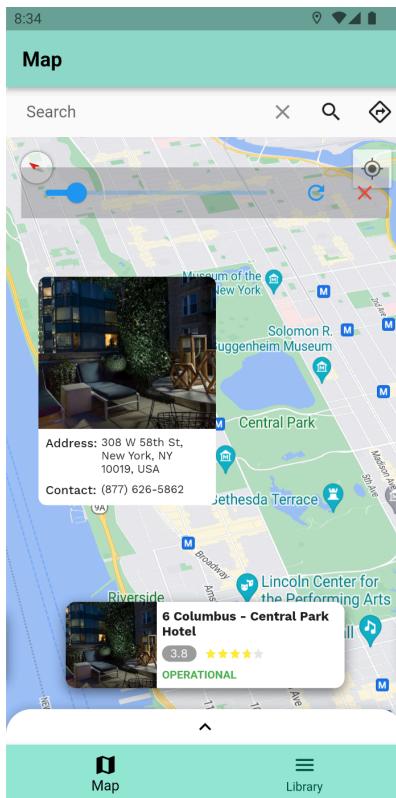
on the left side of the screen. That new card will display an image of the location, address, and contact information. If a user taps on that card, it will flip to reveal reviews of the location provided by users and more photos of the location.



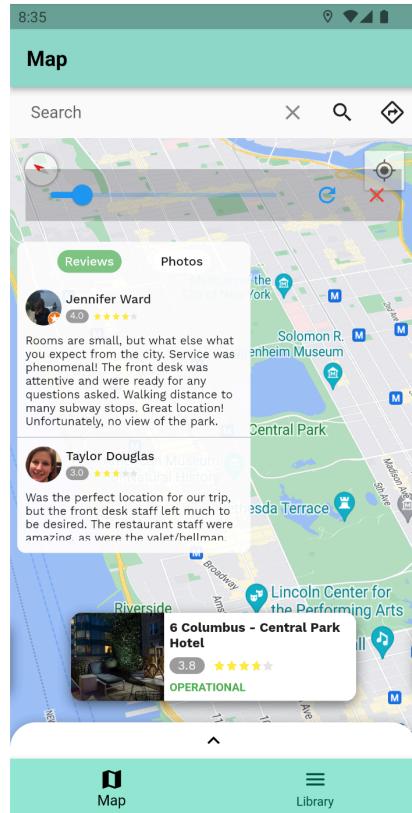
(Fig. 20. The Map Tab)



(Fig. 21. The UI when a user searches for a nearby location)

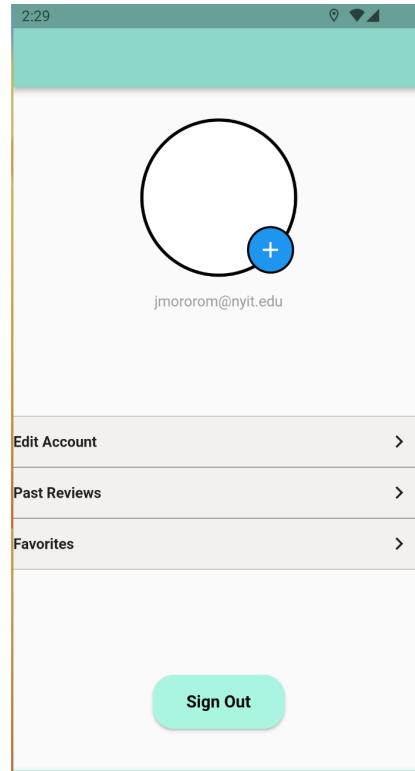


(Fig. 22. The UI when a user taps on a card to reveal additional info)



(Fig. 23. The UI when a user taps the card on the left to reveal reviews + photos)

Users also have access to the library tab where they can view buttons such as edit account information, view a list of their favorite locations, view past reviews, and add a profile picture. Unfortunately, due to time constraints and issues during our project's development, we were unable to actually provide those features to our "final" version of the app. We were also able to incorporate the login, sign-up, and forgot password screens. However, there are issues with these features.



(Fig. 24. The Library Tab)

6.2 - LIMITATIONS

While developing this app, we ran into multiple issues that limited our ability to develop features for the product we initially planned on creating. One issue that our team faced was that the majority of our group was not familiar with the Flutter/Dart framework and had to learn the language and the way the language is structured. Some of our team members were struggling with how Flutter uses widgets to structure an app's layout. Additionally, since some team members never worked with Flutter before, they had to research how to properly install Flutter/Dart & Android Studio into their systems. An issue some team members encountered was not correctly updating the PATH for the Flutter SDK. This resulted in team members being unable to work on the project as their IDE's could not find the Flutter SDK in their system.

A major issue that our team encountered was when implementing the live location feature. When implementing the live location feature, the app wouldn't display the user's live location and instead would either place a user's map marker in the middle of the ocean or in San Francisco. This was always the result even when we switched from Flutter's geolocator package to the location package. Even after spending 2-3 weeks on this issue with multiple attempts by multiple people, we were unable to resolve the issue and were forced to use a static point.

Another issue that our team encountered was connecting the backend of our application with the frontend. Our team used Firebase to host our app's database and authentication services. However, this was challenging for those who worked in the backend because Firebase does not have too many decent tutorials on how to implement it into Flutter applications. Specifically, Firebase has a lot of outdated learning materials that makes it difficult to connect the backend services with the frontend of our app. Fortunately, we were able to find a way to connect the frontend and backend of our app. However, we still run into issues such as the password reset feature not updating a user's password in the database and the app not allowing users to login to the app even though their account information can be found in the database.

6.3 - FUTURE ENHANCEMENTS/RECOMMENDATIONS

A suggestion that was given during this app's development that we found interesting was the idea of letting users be able to access the app's features without having to create an account as a prerequisite. In other words, similar to how other app's work, users would immediately be able to access the map feature and be able to view places near them as well as be able to search and get directions to locations. There would be a sign-up button located in the app bar where users can choose to create an account in order to gain access to features such as the favorites tab, be able to leave reviews, choose a profile picture, and etc.

Other features we would be interested in adding to the app would be as follows:

- Add edit account page
- Let users favorite a location
- Let users leave a review
- Add favorites page
- Add past reviews page
- Add direction information in the sliding panel
- A friends list so users can view their friend's profile, favorites, and reviews.
- A splash screen
- An app icon

6.4 - TEAM MEMBERS/ROLES

(Team Lead) Jeffrey Aguilar - Experienced with Java, Python, HTML, CSS, Flutter/Dart and Xcode/Swift. Knowledge of C++ and SQL.

Roles: Responsible for front-end development and contributed towards project's proposal, final presentation and technical report writing.

Alexander Pachnicki - Experienced with React and React Native, Java language and SQL.

Roles: Responsible for backend development, reports, logs, and contributed towards project's final presentation and technical report writing.

Jhoselyn Moro-Romero - Experienced with Java, Python, C++, SQL. Some experience with HTML. Have some experience with front and back end development.

Roles: Supporting frontend and backend development and testing features.

O'Sean Blagrove - Experienced with Java and some React Native, Node.js, Javascript and MySQL.

Roles: Backend Developer

6.5 - LEARNING OUTCOMES

(Team Lead) Jeffrey Aguilar - During the development of this project, I learned a lot about being a project's lead. I learned that being a project's lead results in a lot more work and responsibility on your shoulders. For example, I was the one who had to connect my team with our project's advisor, set up meetings, help and guide other team members when they ran into issues, and put more time and effort into the development process to the point where I was exhausted during our final presentation. Throughout the app's development, team members

would face issues with their IDE's, Flutter/Dart Workspace, and even unforeseen personal issues. Due to this fact, I also learned how to adapt when faced with unexpected situations that halt the project's development progress. Additionally, I learned how to work with APIs considering I had no prior experience with them. Finally, I was able to continue building my knowledge of Flutter/Dart and adding experience with this language and framework.

Alexander Pachnicki - My learning outcomes from this project included learning how to stay organized and work together as a team despite each of us having different schedules. I also learned more about database design and the different approaches you can have towards designing and implementing a database into an application. I also learned some front end experience with flutter and dart which I believe will be useful in my future.

Jhoselyn Moro-Romero - This project has taught me many things. Throughout the development of the project I faced many personal challenges. I had a few personal emergencies where I was unable to contribute to the project and that put me behind on the progress of the project. I learned to rely on my teammates and to ask for help, I found myself lost many times when it came to working with Flutter/Dart and Jeffrey was kind enough to teach and guide me through the processes. Trying to learn a completely new programming language was difficult especially when the documentation was extremely outdated and no longer relevant to the work that we were doing. This project is easily one of the most memorable experiences I have from my time as a college student. I had a team that was very willing to communicate with and help each other out. I gained very valuable experience working as a team on a big project while learning to juggle personal obstacles, school work, and managing my time.

O'Sean Blagrove - This experience has taught me a lot about collaborative work. It was the longest time I've spent on a single project but my teammates made the experience a lot easier

than it would have been if I were doing it myself. Picking up a new framework and language in just a few months and creating something worthwhile from that is something I am proud of. I struggled early on to find any resources to help me learn and develop quickly to match with my team members but I made this clear to them and they guided me through by providing the resources they were using themselves. They also voiced when they were having difficulties as well which led to us all helping out wherever we could. This is one of the most memorable part of any of my projects I have done in college. The similarities to working in the real world such as setting long term goals, meeting deadlines and working as part of a team are skills that I will take with me throughout the rest of my college career and into my professional one.

7 - REFERENCES

1. Blog, Yelp – Official, and Akhil Kuduvalli Ramesh. “You Can Now Align the Yelp App to Your Lifestyle, Diet, Interests & Other Preferences.” Yelp, 1 Sept. 2021, <https://blog.yelp.com/news/yelp-is-releasing-a-new-personalized-app-experience/>.
2. *Google Cloud Apis*, Google, <https://cloud.google.com/apis/docs/overview#:~:text=Google%20Cloud%20APIs%20are%20programmatic,data%20analysis%20to%20your%20applications>.
3. Nield, David. “The 9 Best Travel Apps for Exploring Cities.” Gizmodo, Gizmodo, 16 Sep. 2016, <https://gizmodo.com/the-9-best-travel-apps-for-exploring-cities-1786593373>.
Stephenson, Brad. “All about Yelp and How It Works.” Lifewire, Lifewire, 27 Mar. 2020, <https://www.lifewire.com/what-is-yelp-4685842>.

4. Roberts, Dawna M. "The 14 Best Trip Planner Apps for Your Perfect Vacation in 2020." Compare and Choose The Best One, Compare and Choose The Best One, 11 Feb. 2021, <https://launchberg.com/best-trip-planner-apps/>.
5. Rizzo, Cailey. "5 Apps for Exploring a New City like a Local." Mashable, Mashable, 29 Oct. 2021, <https://mashable.com/article/local-travel-apps>.
6. "Run Apps on the Android Emulator : Android Studio : Android Developers." *Android Developers*, <https://developer.android.com/studio/run/emulator>.
7. "Running Your App in Simulator or on a Device." *Apple Developer Documentation*, <https://developer.apple.com/documentation/xcode/running-your-app-in-simulator-or-on-a-device>.
8. "Privacy and Security in Firebase." *Google*, firebase.google.com/support/privacy.
[Privacy and Security in Firebase \(google.com\)](https://firebase.google.com/support/privacy)