DEPARTMENT OF COMPUTER SCIENCE

UNIVERSITY OF TEXAS AT AUSTIN

Project Writeup

WhereToGo - Real-time Taxi Density Analyzer

Author: oc5477 - Osman Calisir
Submission Date: 18-04-2024

I confirm that this project writeup is my own work and I have documented all sources and material used.

Starnberg, Germany, 18-04-2024 oc5477 - Osman Calisir

Acknowledgments

I would like to extend my sincere thanks and appreciation to the University of Texas at Austin Computer Science Department for their invaluable support and guidance throughout my academic journey in this module. The exceptional faculty members and staff have consistently demonstrated their commitment to excellence, providing us with the knowledge and skills necessary to succeed in this module. I am grateful for the opportunities and resources that the department has provided, and I look forward to applying the skills I have acquired to make a positive impact in the world.

Contents

1 Project Overview

1.1 Team Members

For this project, I, as Osman Calisir worked on this project alone.

1.2 Project Title

WhereToGo: Navigating Urban Mobility with Real-Time Taxi Density Mapping

1.3 Brief Description

WhereToGo is designed to provide real-time visibility into taxi availability. Utilizing advanced mapping technologies and real-time data analytics integrates with Google Maps API that displays heatmaps of taxi densities.

1.4 Origin of the Project idea

I personally worked on this project with a team in a Makeathon that is organised by the Technical University of München in 2021. The Machine Learning and Data Processing codes are located here publicly: https://github.com/Wissenschaften/Makeathon

2 User Interface Design

2.1 Description of the UI layout and design choices

The user interface (UI) of WhereToGo is designed with the aim of providing an intuitive experience to users seeking real-time taxi availability information. The design philosophy centers around simplicity.

2.2 Main Map View:

- Central Focus: The primary interface is dominated by a dynamic map powered by the Google Maps API. This map provides a real-time graphical representation of taxi density using heatmaps. Users can pan and zoom effortlessly to view different areas of the city.
- Heatmap Integration: Different color gradients signify the density of available taxis, ranging from sparse (cool colors) to dense (warm colors), making it easy to assess taxi availability at a glance.

2.3 Search Functionality:

- Top Bar Placement: A search bar is positioned at the top of the screen, enabling users to enter specific addresses or landmarks. This feature allows users to quickly navigate to areas of interest without manually searching the map.

2.4 Design Choices:

- Responsive Design: The UI is fully responsive, ensuring consistent functionality and aesthetic appeal across various devices and screen sizes.
- Show Optimized Data Toggle: You can reach this toggle in the Map Screen on the top right corner. With this toggle, you will have a mutable shared state between the fragments. There are two data files from Firebase storage. One is the general file that includes New York taxi locations, and the other is includes the optimized one. With this toggle, we can change the data file where defined in DataService.kt under services. Please check Figure 2.7 and 2.9 below.

2.5 Screenshots of Key Screens

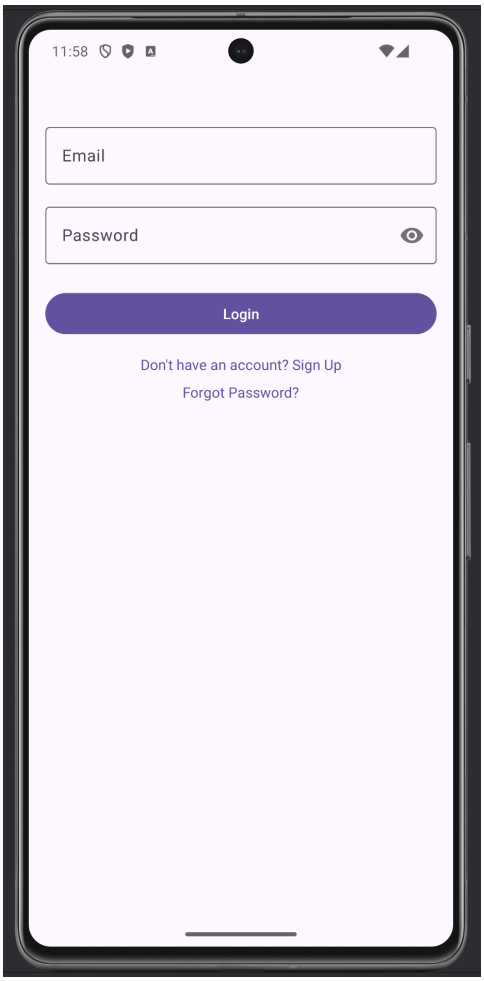

Figure 2.1: Login Screen - Horizontal

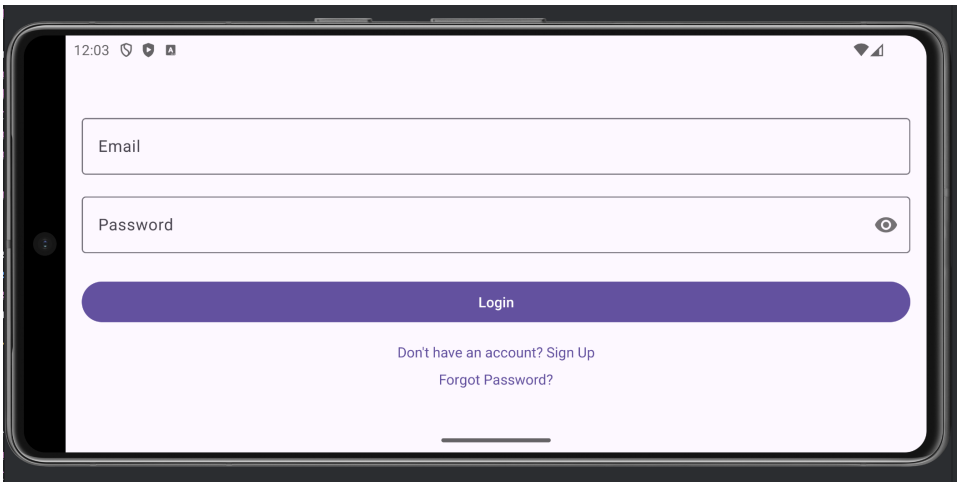

Figure 2.2: Login Screen - Vertical

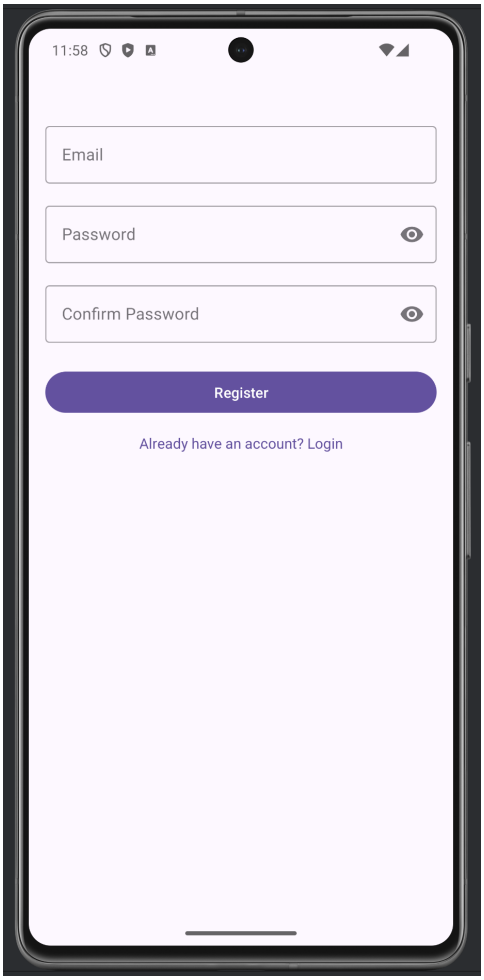

Figure 2.3: Register Screen - Horizontal

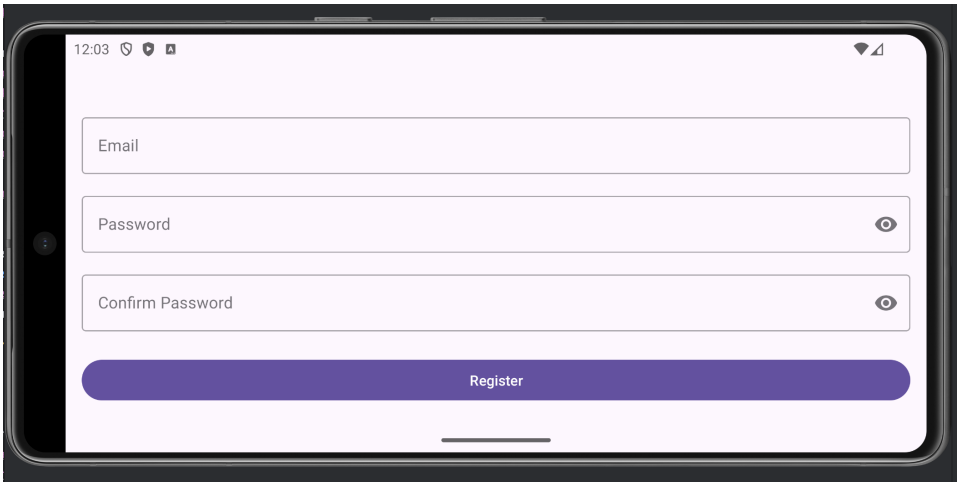

Figure 2.4: Register Screen - Vertical

Figure 2.5: Forgot Password Screen - Horizontal

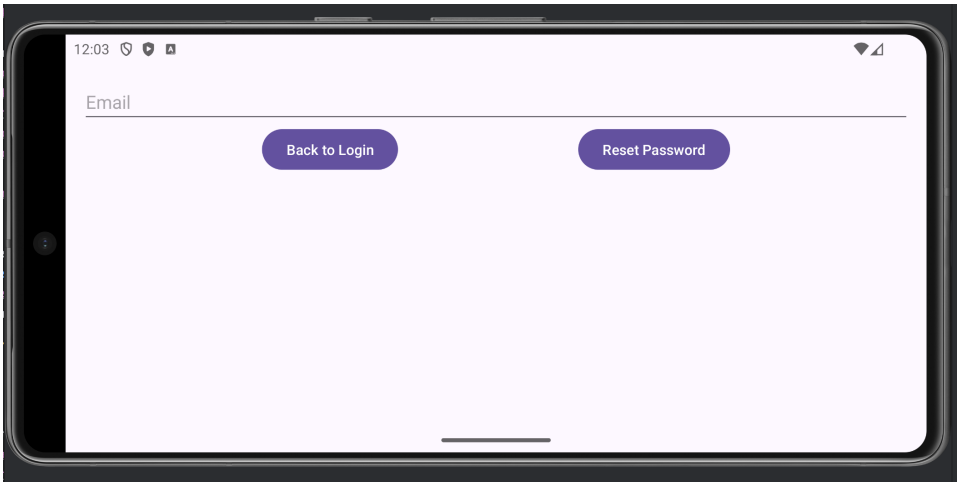

Figure 2.6: Forgot Password Screen - Vertical

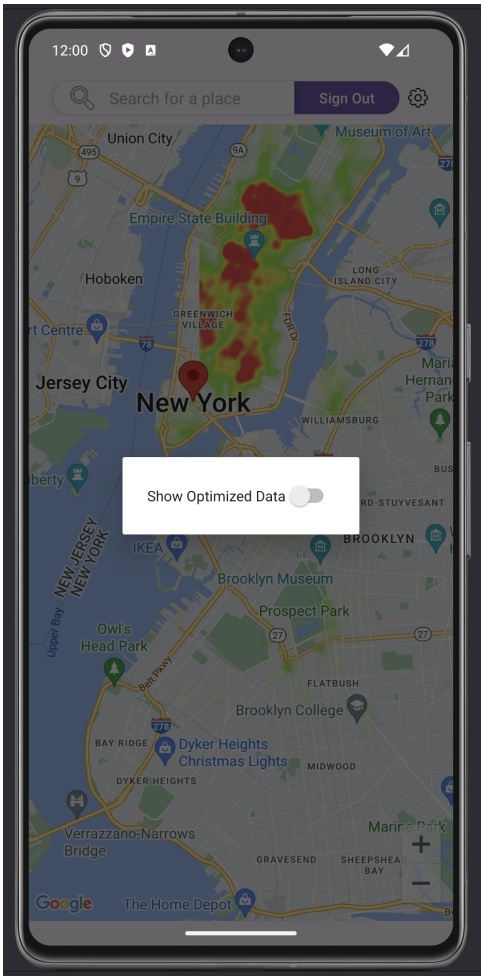

Figure 2.7: Map Screen with no optimization - Horizontal

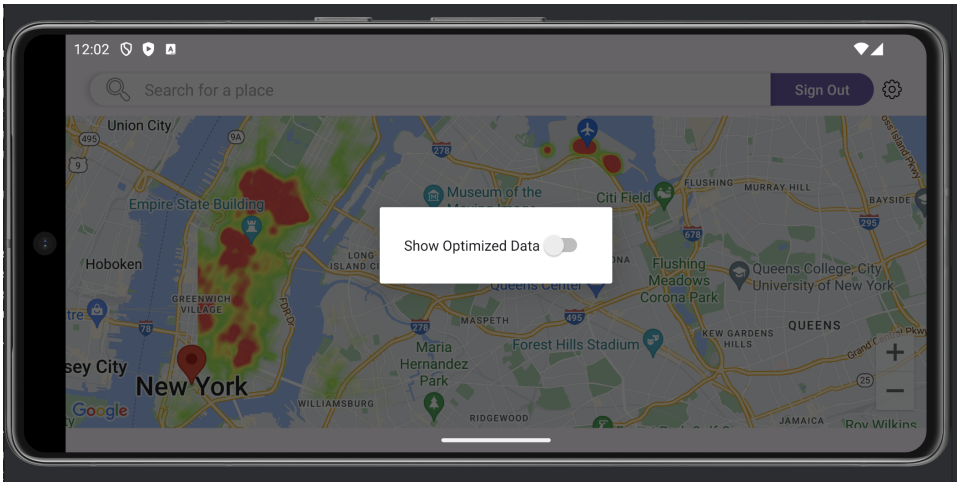

Figure 2.8: Map Screen with no optimization - Vertical

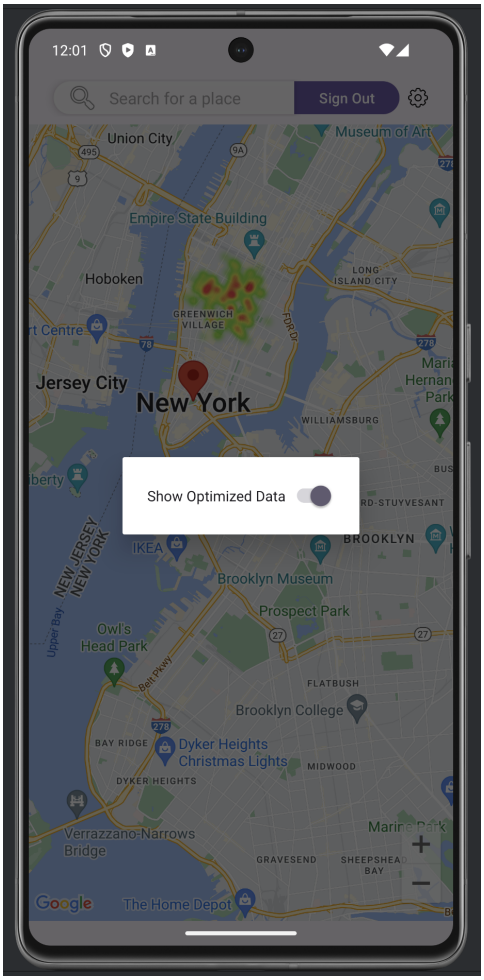

Figure 2.9: Map Screen with optimization - Horizontal

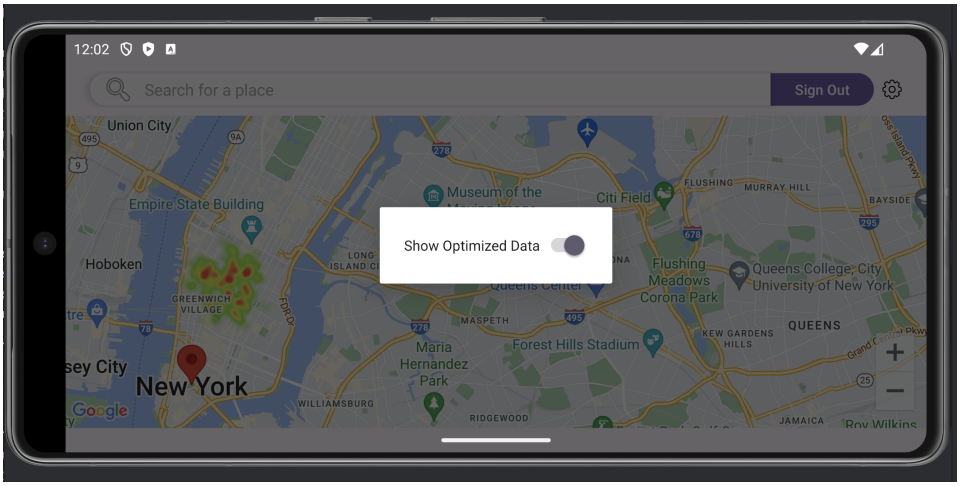

Figure 2.10: Map Screen with optimization - Vertical

3 Backend Architecture

3.1 Overview of Backend Services

The "WhereToGo" application leverages powerful backend services to provide real-time taxi density data and geographical information, primarily using Google Maps API and Firebase. These services are integral to the app's operations, enabling efficient data handling, user authentication, and interactive mapping functionalities.

3.2 Google Maps API:

Google Maps API is a core component of "WhereToGo," utilized to render detailed and interactive maps on the user interface. This API allows the application to display current geographical locations with high accuracy and visual appeal. Key features integrated from Google Maps include:

- Maps and Geolocation: Provides the canvas for displaying the user's current location and surrounding area, enabling users to navigate the map freely with gestures.
- Heatmaps Layer: An overlay on the map that visualizes taxi density using color gradients, where data points are aggregated into colors ranging from green (low density) to red (high density). This feature is crucial for users to identify areas with high or low taxi availability.
- Geocoding and Places API: Used to convert addresses into geographic coordinates and vice versa. It enhances the search functionality by allowing users to search for locations based on names or addresses, making the app more user-friendly.

3.3 Firebase:

Firebase provides a suite of cloud services that support the backend infrastructure of "WhereToGo," offering solutions for database management, authentication, and hosting. The implementation of Firebase in the app includes:

- Firebase Authentication: Manages user registration and login processes. It supports various authentication methods, including email and password-based authentication, resetting password, ensuring secure and reliable access to the app's features.
- Realtime Database: Stores and synchronizes data in real-time. In "WhereToGo," this database is used to store and retrieve the locations and densities of taxis. Whenever data changes, the app receives updates in real-time, which are then reflected on the heatmap.
- Cloud Functions: Serverless framework that lets us run backend code in response to events triggered by Firebase features and HTTPS requests. Used in "WhereToGo" to calculate and update taxi densities based on data from taxi operators.

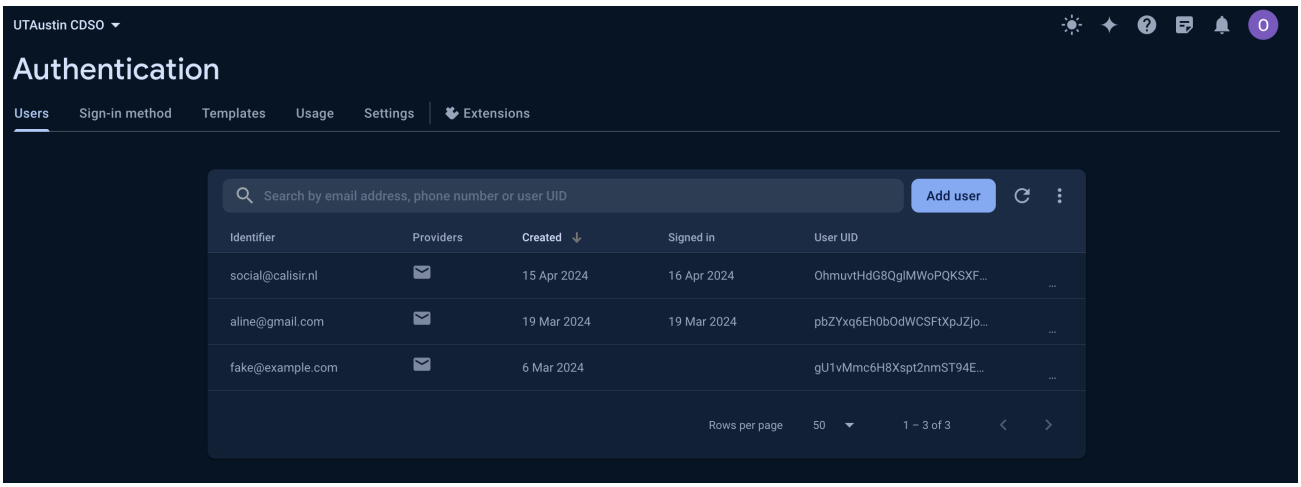

Figure 3.1: Firebase Authentication Screen

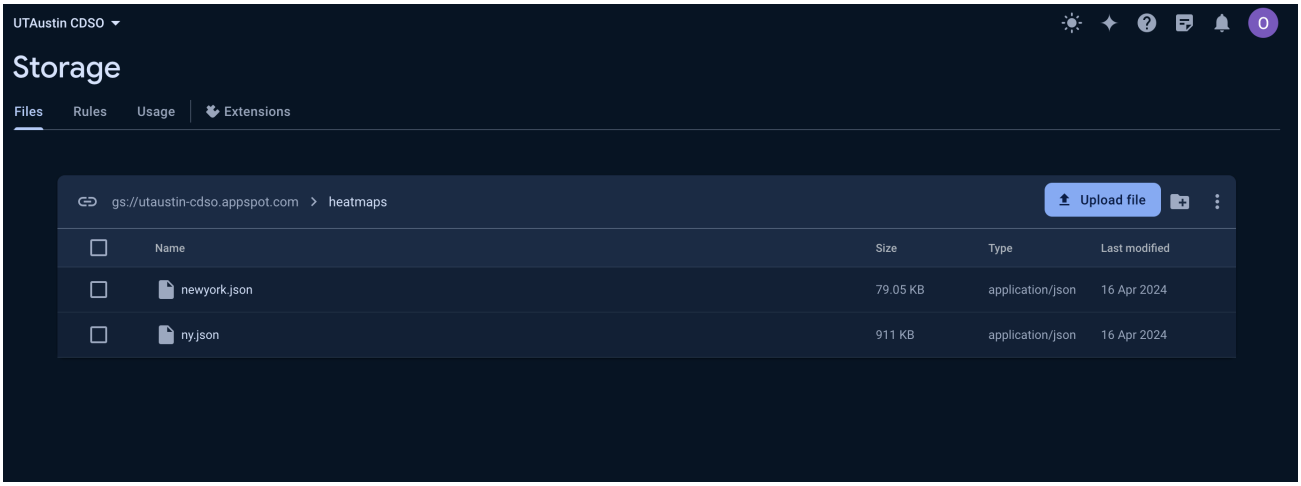

Figure 3.2: Firebase Storage Screen

3.4 Integration and Data Flow:

The integration of Google Maps API and Firebase is designed to ensure seamless data flow and synchronization across the application:

- Data Collection: Taxi location data is collected and sent to the Firebase Realtime Database by taxi operators' devices.
- Data Processing: Cloud Functions process this data to calculate density, updating the database with new density information.
- Data Visualization: The updated density data triggers an update in the app, fetched via Firebase, and visualized on the Google Maps interface using heatmaps.

3.5 List of mutable shared states

Here, in the project, the following states are shared as mutable:

- isOptimized: This state is for changing the data file from Firebase Storage (Please see the Figure 3.2). With that, the app will switch the data between optimized(newyork.json) and non-optimized(ny.json) ones.
- Firebase Authentication states: These states are for checking the authentication situation. With that, user can register, login, and request a forgot password link (to test this, you have to use an accessible email address)

4 Learnings

4.1 Key Learnings

With this project, I learn how to combine the knowledge that I have from the courses and learn how to manage them all to create a single yet powerful application.

4.2 Difficult Challenges Faced and Resolved

Developing the "WhereToGo" application posed several technical and operational challenges that required innovative solutions and strategic problem-solving. Below are some of the most significant challenges faced during the project and the methods used to overcome them:

- Real-Time Data Synchronization: Although its mentioned that this is a real-time application, that refers that the application is getting data from the Firebase Storage. Fetching this data and mapping it without causing delays or inconsistencies proved difficult.
- User Interface Responsiveness: Maintaining a smooth and responsive user interface while rendering complex heatmap overlays and handling user interactions was challenging. Initial versions of the app experienced lag and slow map updates, particularly on lower-end devices. To improve UI responsiveness, I optimized the heatmap rendering logic by reducing the complexity and frequency of updates. We implemented a caching mechanism that minimized redundant computations and network requests. Additionally, the use of asynchronous programming ensured that the UI thread was not blocked by long-running operations, keeping the app responsive at all times.
- While developing the application, I have used a mutable shared state that was really hard to maintain. In the first implementation, the data was updating but not coming in the real time and I needed to refresh the application in the Android Studio which was not looking good. Later, I resolved it with DataRepository object as hub.

5 Conclusion

5.1 Report of the count of lines of code in the project

With this project, I learn how to combine the knowledge that I have from the courses and learn how to manage them all to create a single yet powerful application.

- XML: 19 Files, 42 Blank, 30 Comment, 602 Code
- Kotlin: 9 Files, 87 Blank, 1 Comment, 464 Code

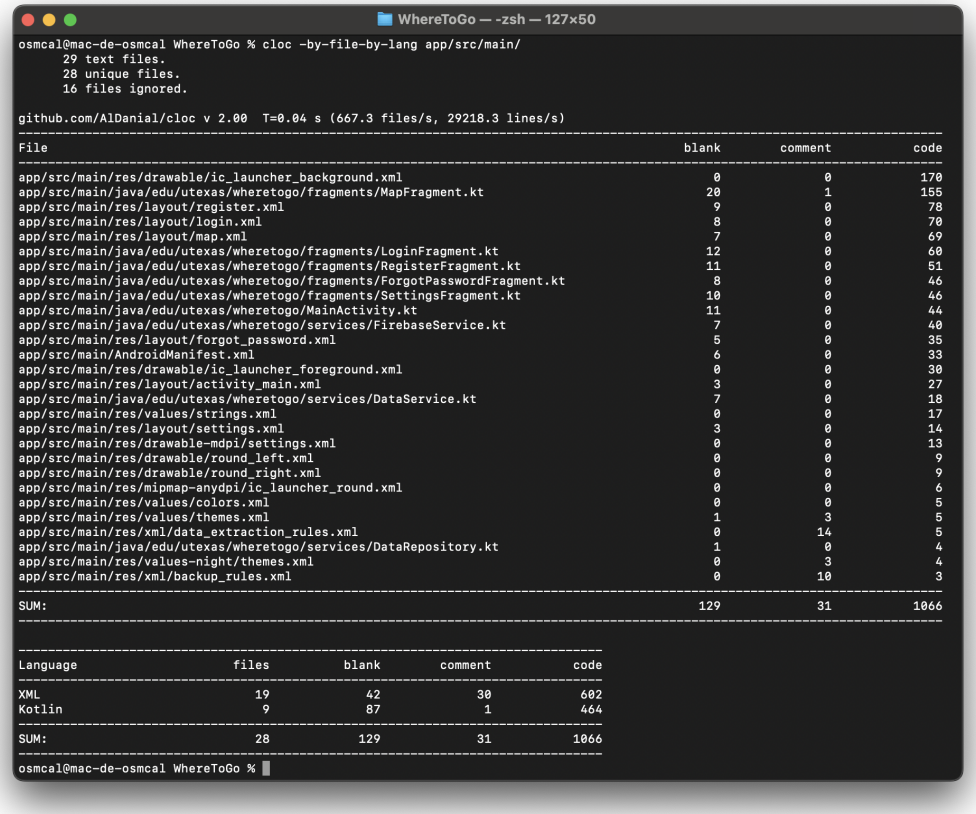

Figure 5.1: CLOC Report

5.2 Resources of the Project

- GitHub Link: https://github.com/ut-msco-s24/project-osmancalisir
- Video Link: https://youtu.be/BI9CByVUXrA

5.3 Researches made on this idea

- Medium page: https://medium.com/analytics-vidhya/new-york-yellow-taxi-demand-prediction-using-machine-learning-fc697d20ff86
- Nih.gov page: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8093244/

5.4 Code Frequency page

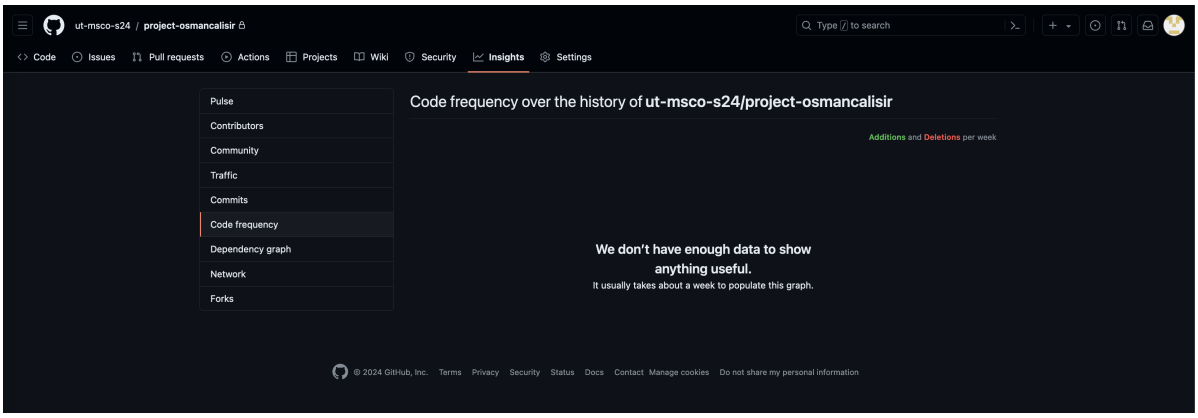

Figure 5.2: Code Frequency Screen Result