# UNIVERSITY CANTEEN FOOD ORDERING APPLICATION

CLOUD COMPUTING AND APPLICATIONS

MINI PROJECT 2 - FINAL REPORT

GROUP 04

## ABSTRACT

This project was introduced as a “Aduwa” mobile application that facilitate to ordering food in university canteen to reduce food wastage and provide continuous food delivering to university canteen users. Through this app university student can order foods. Canteen owners can update food list according to availability. Canteen users can search food items also they can see nearby canteen using map in the university. This app suggest food according to user previous food preferences. And also suggests canteen according to food availability.



## ACKNOWLEDGEMENT

A special thank you to Mr. Ishara Dissanayake for helping us, guiding us through the project, explaining cloud services, showing how to carry out each task, giving better suggestions to improve our application and successfully finish this "Aduwa" app.

# TABLE OF CONTENTS

# TABLE OF FIGURES

# 1. INTRODUCTION

## 1.1 BACKGROUND

In today's day and age, one of the major concerns faced by university students revolves around accessing their breakfast and lunch options. Unfortunately, university canteens often struggle to meet the demands of the student population due to limited food availability and a lack of variety. Consequently, when there is a large influx of students, some find themselves unable to obtain a meal. Moreover, the canteen staff lacks a clear understanding of the quantity of food required, exacerbating the problem further. Additionally, different canteens within the university offer varying food selections, leaving students unsure of where to go for their lunch, resulting in unnecessary time wastage as they search from place to place.

## 1.2 MOTIVATION

Using this app university canteen can handle students' orders in a proper manner. At university, there are huge queues in front of the university canteens. Thus this app provides the facility to see the availability of food in a particular canteen they wish to buy foods. Suggesting foods according to previous food preferences can easily identify where the food is available. Also, this app provides easy management for canteen owners. Canteen owners add available food to the list. They can get orders from the canteen users. Then owners can easily supply food as required. Also, owners can get a better understanding of the probabilities of ordering particular food according to the ordering history.

## 1.3 RELATED WORKS

There are various applications existing for ordering food. Also, there are some food-ordering apps developed for canteens in colleges and universities. The existing systems only get ordering and doing billings. Those existing systems provide the canteen menu and customize orders according to user preferences and display order status as well. Apart from that our ‘Aduwa’ app provides a facility to suggest a nearby canteen in the university where the food availability is according to user food preference. Users can select the canteen and place orders.

# 2. PROBLEM OVERVIEW

## 2.1 PROBLEM STATEMENT

- University students face challenges in accessing breakfast and lunch options due to limited availability and variety in canteens, particularly during high-demand periods.
- The lack of clear food quantity and diverse offerings create confusion and waste time.

## 2.2 FEATURES & FUNCTIONALITY

- Users of this mobile application - University students & canteen owners.
- Canteen owners should be registered into the app before login.
- Owners can update the available food list.
- Student login can be verified by the student university email address.
- Students can order food according to the available food list.
- Suggest the canteen locations according to the user's previous food preferences.

## 2.3 OBJECTIVES

- To identify food requirements of university students and supply foods according to requirements.
- Minimize food wastage by preparing excess foods.
- Saving students time by reducing long queues in the canteen.
- To develop this app for all university students around Sri Lanka to use in any university.

# 3. REQUIREMENT ANALYSIS

## 3.1 FEASIBILITY STUDY

The requirement analysis phase is focused on conducting the feasibility of the available technologies to be used in the project. This project does not need any external hardware components other than a smartphone camera. This is an android mobile application. The app will include features such as user registration and authentication, a comprehensive menu and item listings, an intuitive ordering system, real-time updates on availability and menu changes, secure payment integration, order tracking, a loyalty program, user reviews and ratings, notification and alerts, accessibility considerations, analytics and reporting capabilities, and seamless integration with the canteen's backend systems.

# 4. DESIGN AND IMPLEMENTATION

## 4.1 ARCHITECTURE

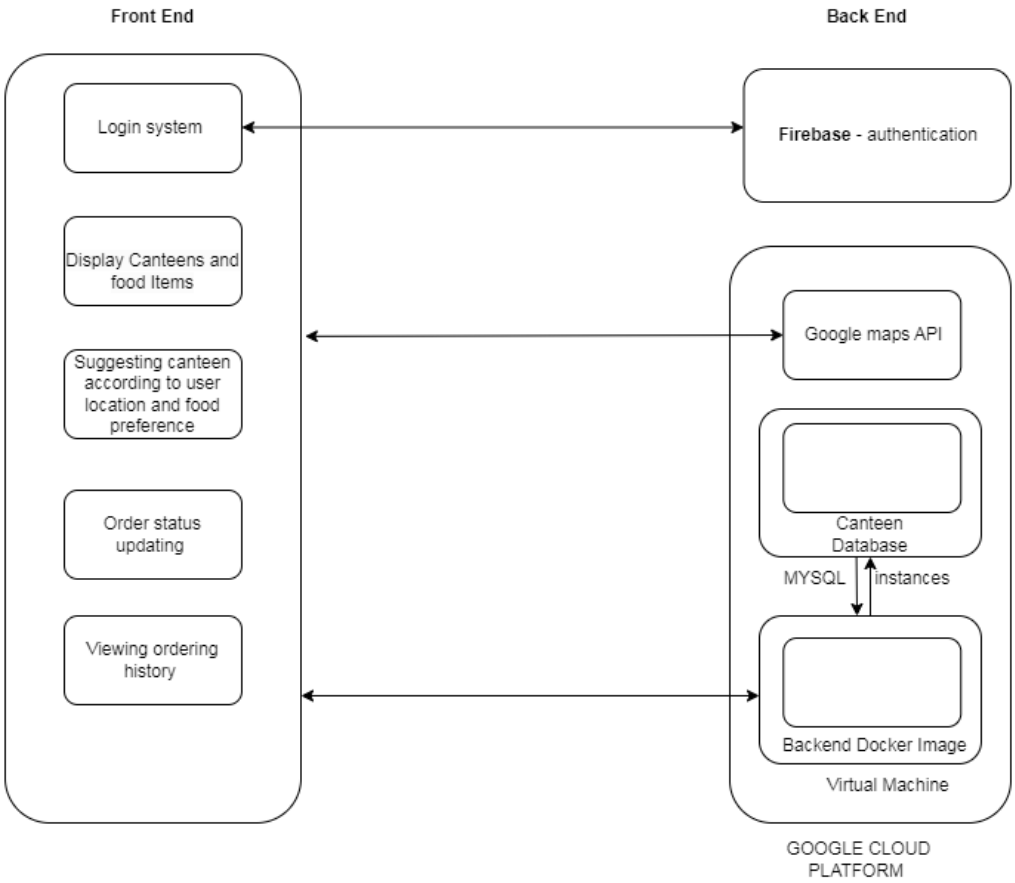


Figure 1:Architecture Diagram

## 4.2 DEVELOPMENT SUPPORTIVE

### 4.2.1 LANGUAGES

- Dart Language (Flutter)
- Python
- SQL

### 4.2.2 TOOLS

- Android Studio
- Visual Studio Code

### 4.2.3 CLOUD SERVICE PROVIDER

In our ADUWA app, cloud computing plays a major role, enabling a range of services and functionalities. This app is a PaaS model but it uses IaaS and some sub-categorical services like BaaS and DBaaS also. Here we use Google Cloud Platform and Firebase as cloud service providers. How cloud services are utilized, along with their respective types are listed below.

- Google Cloud Platform

**Google Maps API (Mapping Service)**

- Service Type: Platform as a Service (PaaS)
- We use Google Maps API, a PaaS, to get the google maps, provide location services (current location and search location) which helps to do the suggestion part for the user to find nearest canteens.
    - Geocoding API: To converts coordinates into easily understandable addresses(texts).
    - Places API: To search places. This enhances user-friendliness with autocomplete suggestions during search.

**Google Cloud Hosting (App Deployment)**

- Service Type: Infrastructure as a Service (IaaS)
- The app is hosted on Google Cloud using virtual machines. This allows users to access the app over the internet.

### 4.2.4 AUTHENTICATION

- Firebase

**Firebase Authentication (Authentication Service)**

- Service Type: Backend as a Service (BaaS)
- Firebase Authentication is a Google service which provide backend development facilities for offering and manages user authentication. It empowers users to sign in via email, social media, or phone numbers. Firebase handles authentication securely, including safeguarding user credentials.
- For authenticating users when they are login, we use firebase authentication.

### 4.2.5 DATABASE & STORAGE

- MySQL in Visual Studio Code to local host

  First we implement database in MySQL to store and retrieve data. We locally host the app and at final stage we connect app to cloud platform because otherwise we have to pay for the cloud platform if we exceed the free limit.

- Google cloud SQL

**Google Cloud SQL (Database Service)**

- Service Type: Database as a Service (DBaaS)
- Google Cloud SQL offers a fully managed relational database service.
- We use Google Cloud SQL to store menu items, ordering details etc.

## 4.3 DEVELOPMENT METHODOLOGY

There are few major steps of the development of the ‘Aduwa’ app.

**Frontend Development with Dart (Flutter)**

Flutter Project Setup

- Use Visual Studio Code to create a new Flutter project.
- Set up the project structure, including screens for user registration, order placement, and view the ordering history.

UI/UX Design

- Design the user interface for the app, including screens, buttons, forms, and navigation.
- Integration with Firebase Authentication and implement Firebase Authentication in the Flutter app to allow users to sign in and register securely. Integrate Google Maps API to get the Google Maps API to display maps, enable location services like getting current location to suggest nearby canteens and search locations to set the university

**Backend Development with Python**

Backend Setup with Flask

- Use Python and the Flask framework to set up the backend server.

MySQL Database Integration

- Connect the Flask app to a locally hosted MySQL database in VSCode.
- Create tables for storing orders details, canteen details, and User details.
- Implement API Endpoints to handle order placement, and retrieval of menu information.
- Connect Backend to Flutter App using HTTP requests (GET, POST) in Dart and exchange data with the Flutter app.

**Containerizing the MySQL Database with Docker**

Docker Image Creation:

- To ensure consistency and portability, we converted our Python backend into a Docker image.
- Docker allows we to package all the necessary components and dependencies of our app in one container, making it easier to deploy and manage.

Docker Image Push:

- We pushed the Docker image to our chosen cloud environment. This step is vital for deploying our app and its backend in a consistent manner across different environments and platforms.

**Implement Google Cloud MySQL**

Set Up Google Cloud MySQL

- Create a MySQL instance on Google Cloud Platform.
- To ensure security, and scalability, we utilized GCP to host a MySQL instance and database.
- GCP provides a cloud-based environment with extensive features, making it a reliable choice for hosting our database.

Database Migration

- Migrate the data from the Docker container to the Google Cloud MySQL instance.

**Firebase Authentication**

Firebase Project Setup

- Create a new Firebase project and configure it with the Flutter app.

Integrate Firebase Authentication

- Use Flutter's Firebase authentication plugin to integrate Firebase authentication into the app.

**Hosting the App on Google Cloud Platform**

Deploy Backend on Google Cloud

- Use Google Cloud Platform to host the Python backend.

Deploy Flutter App

- Set up Google Cloud App Engine to host the Flutter app.

Configure Firebase Hosting

- Use Firebase Hosting to deploy and serve the app's frontend.

These steps will be explained under the Development and Implementation procedure.

# 5. DEVELOPMENT AND INTERFACES

## 5.1 DEVELOPMENT

### 5.1.1 CLOUD USAGE (BACKEND)

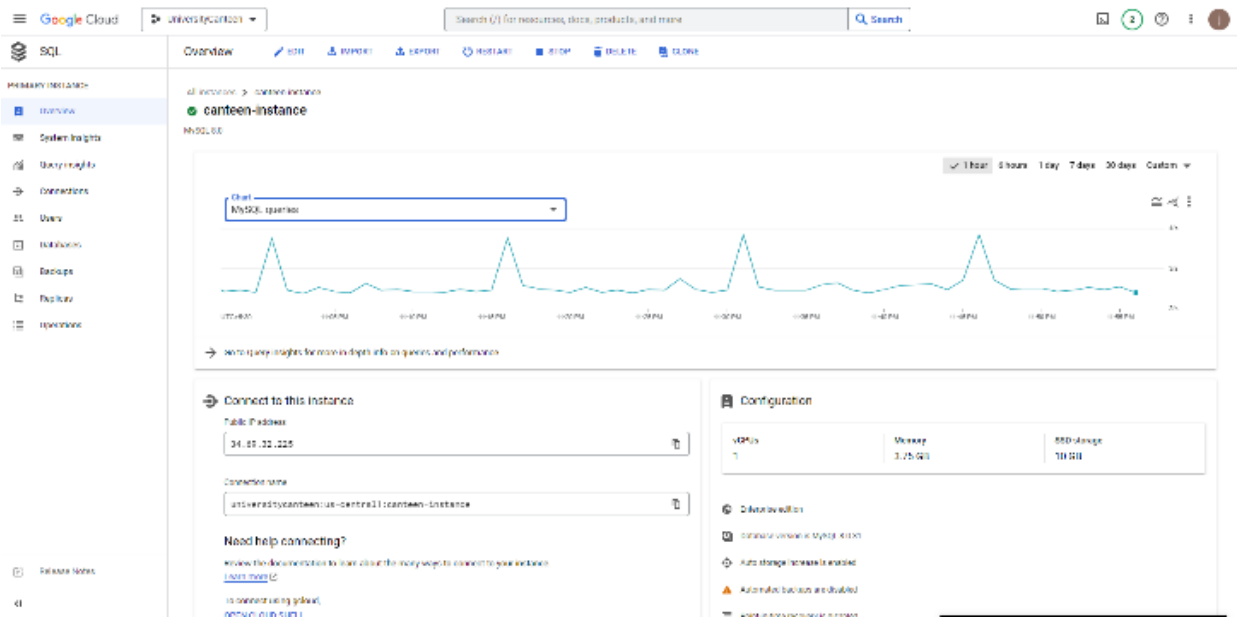


Figure 2:MYSQL Instance

Utilizing a MySQL instance on Google Cloud Platform makes database maintenance easier, improves dependability, and offers a safe, scalable solution for the data storage requirements of our

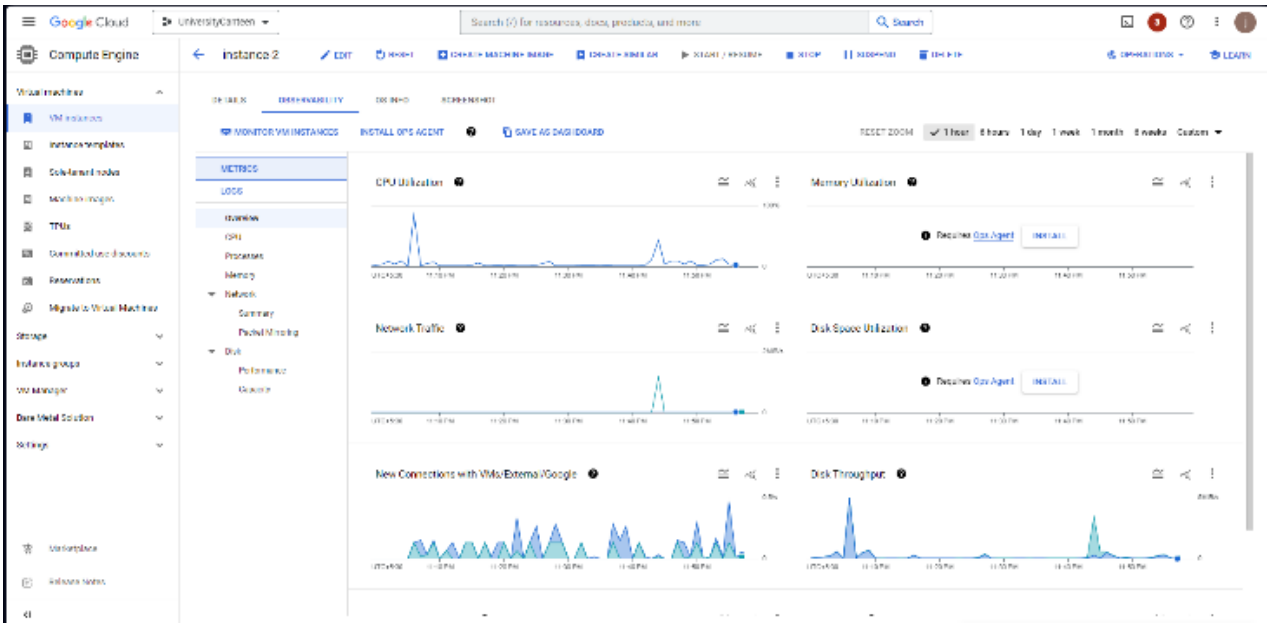


Figure 3:Virtual Machine Instance

They offer the resource allocation, scalability, and cost-effectiveness that come with cloud computing along with the control and customization of physical servers.

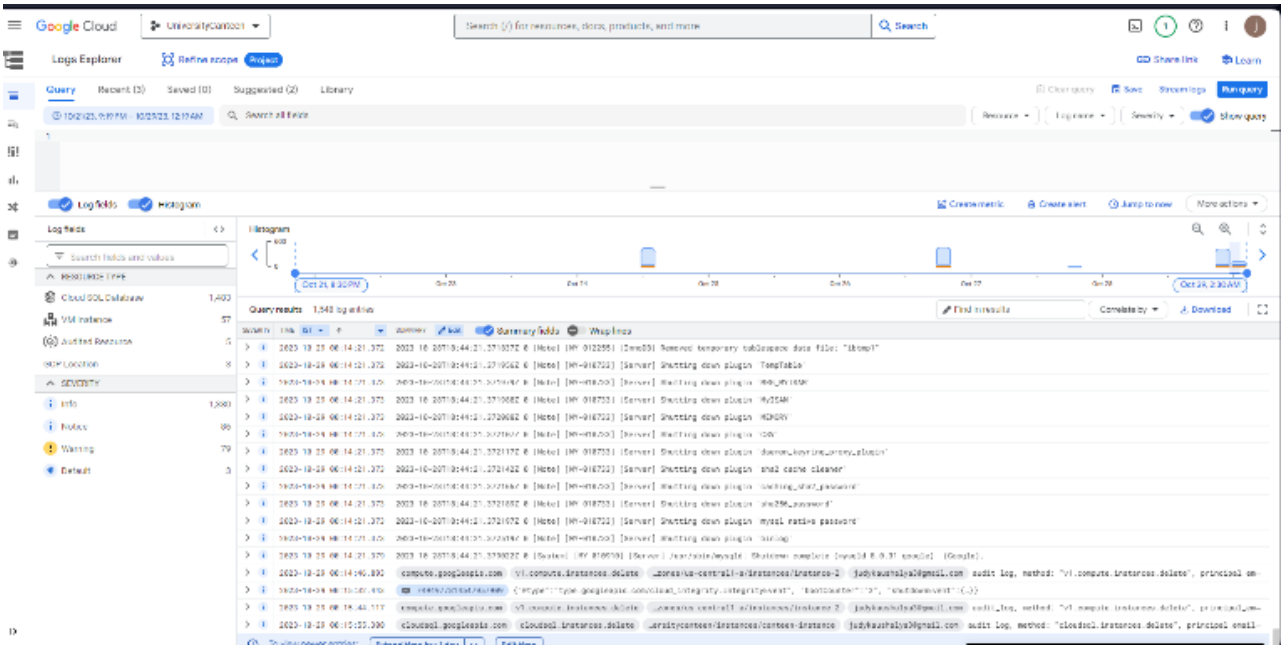


Figure 4:GCP Log

GCP distinguishes itself by providing a wide range of tools and services in the field of artificial intelligence and machine learning.

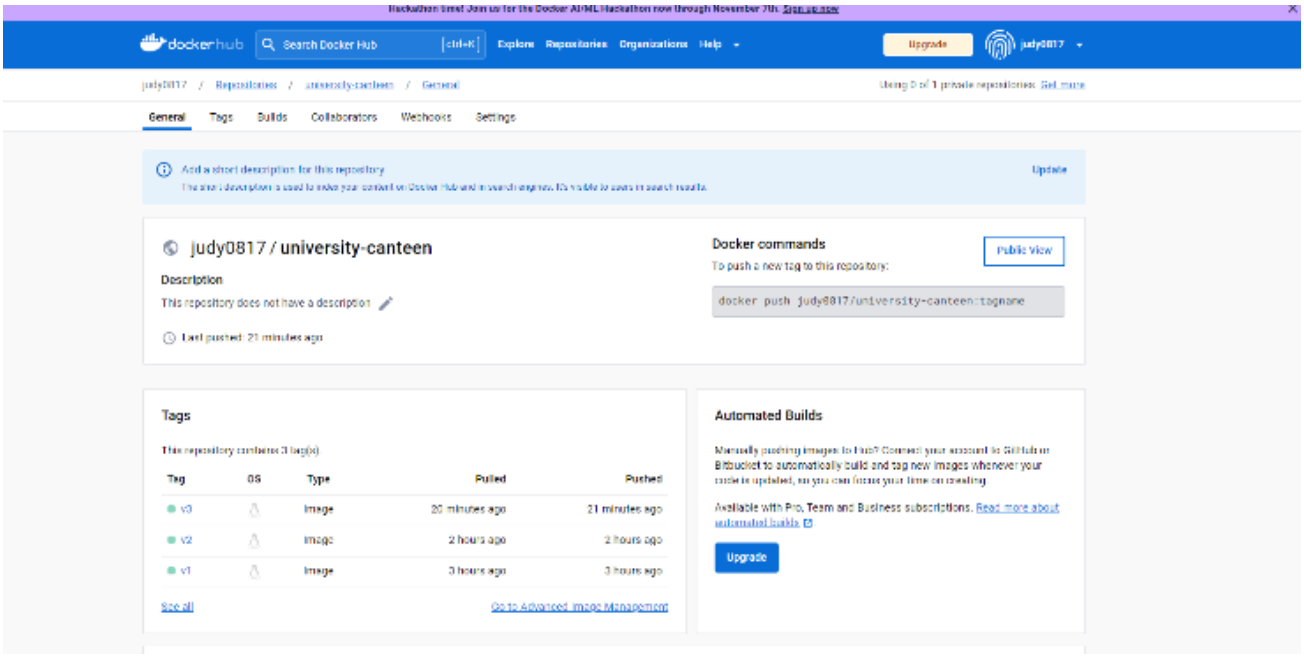


Figure 5:Docker repository

Its containerization technology is a useful tool for operations teams as well as developers, having completely changed the way applications are produced, deployed, and managed.

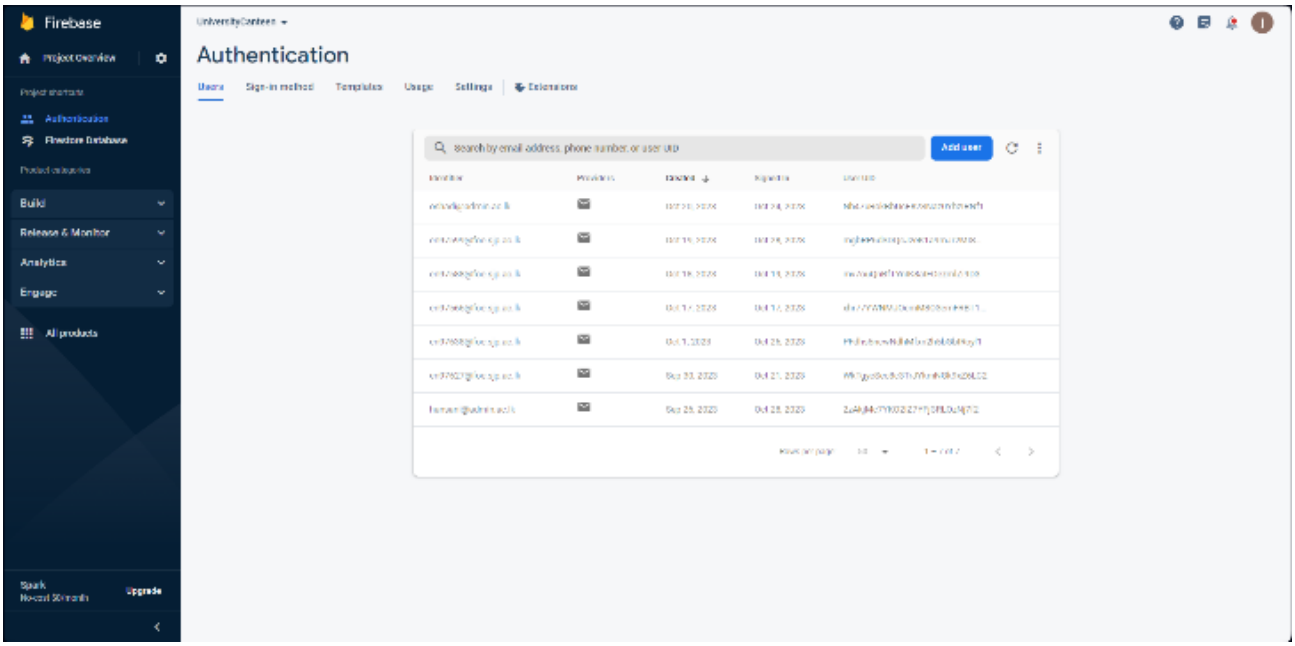


Figure 6:Firebase Authentication

Developers looking for a complete and user-friendly authentication solution will find Firebase Authentication to be highly recommended due to its ability to streamline user identity management, improve security, and accommodate various authentication methods.

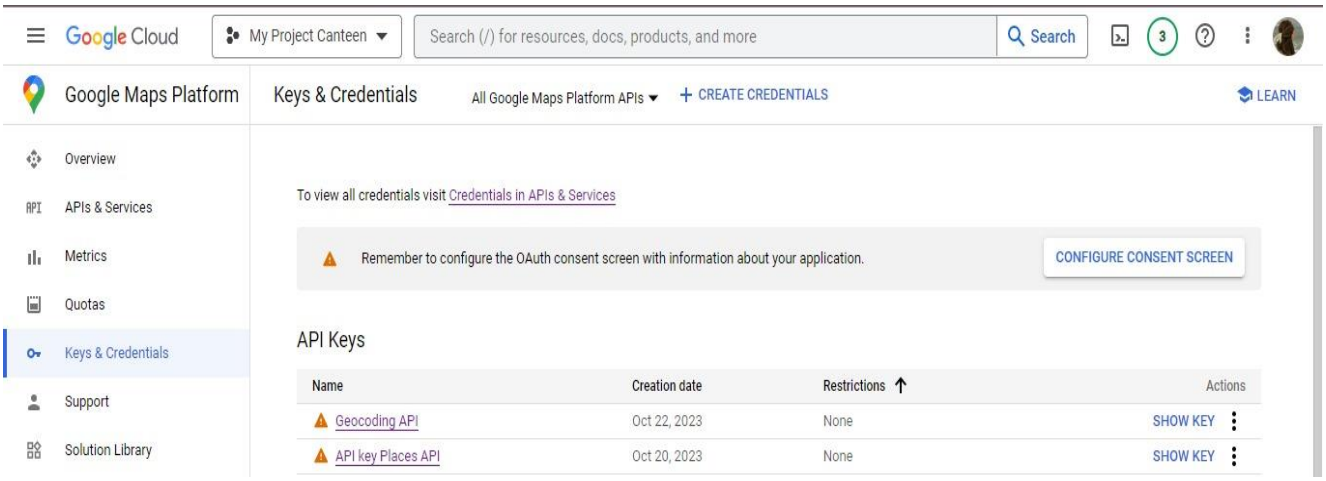


Figure 7:Google map in GCP

The Google Cloud Platform (GCP) Maps API is a suite of tools and services that enable developers to integrate mapping features into their applications. It provides high-quality, up-to-date maps. In here we use 2 APIs, Geocoding API and Places API.

## 5.2 INTERFACES

### 5.2.1 FRONT-END

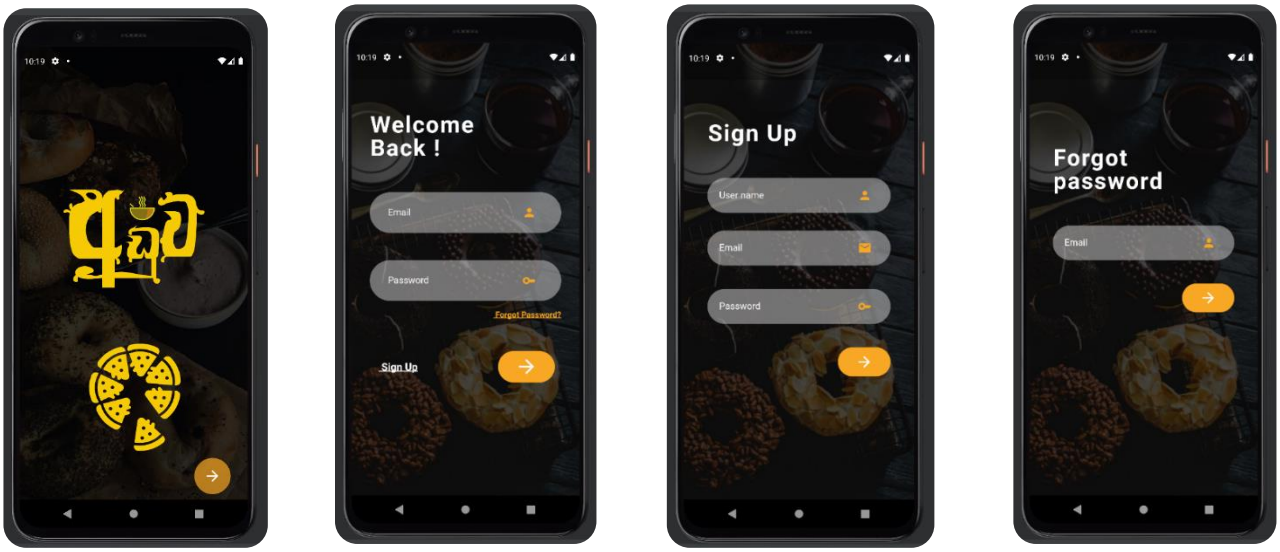


Figure 8:Loading screen and SignIn SignUp screens

Users have two user levels.

- Canteen owners can log into their accounts using xxxx@admin.ac.lk this email.
- Students can log into their accounts using enxxxxx@foe.sjp.ac.lk this email.

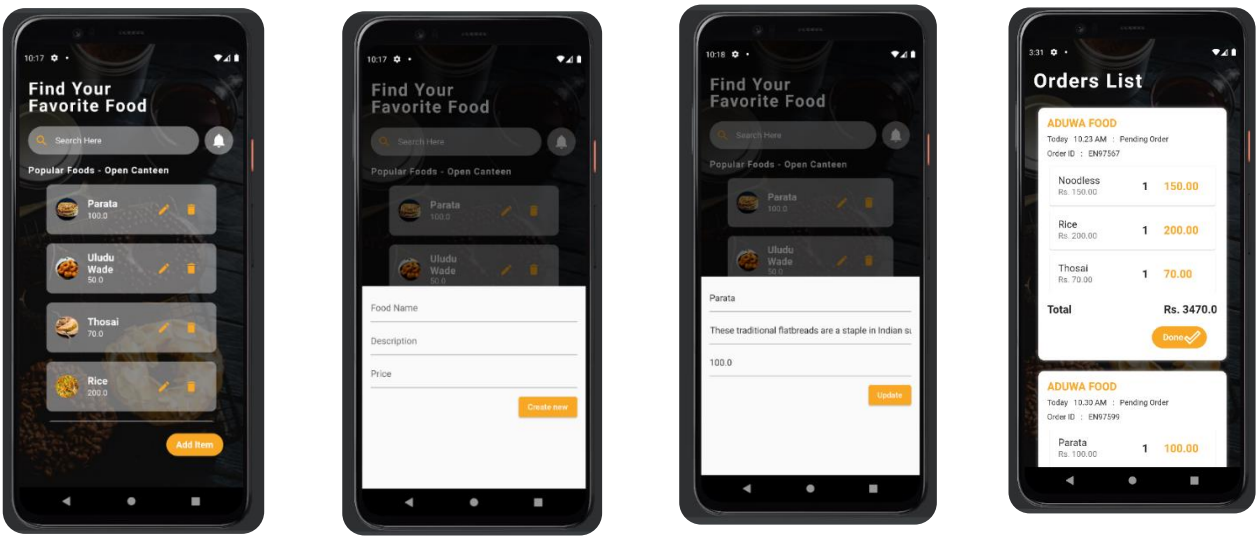


Figure 9:Canteen owner Home screen and Features

Using this interfaces canteen owner can

- Insert food items
- Update food list
- Delete unnecessary food items.
- View Order List

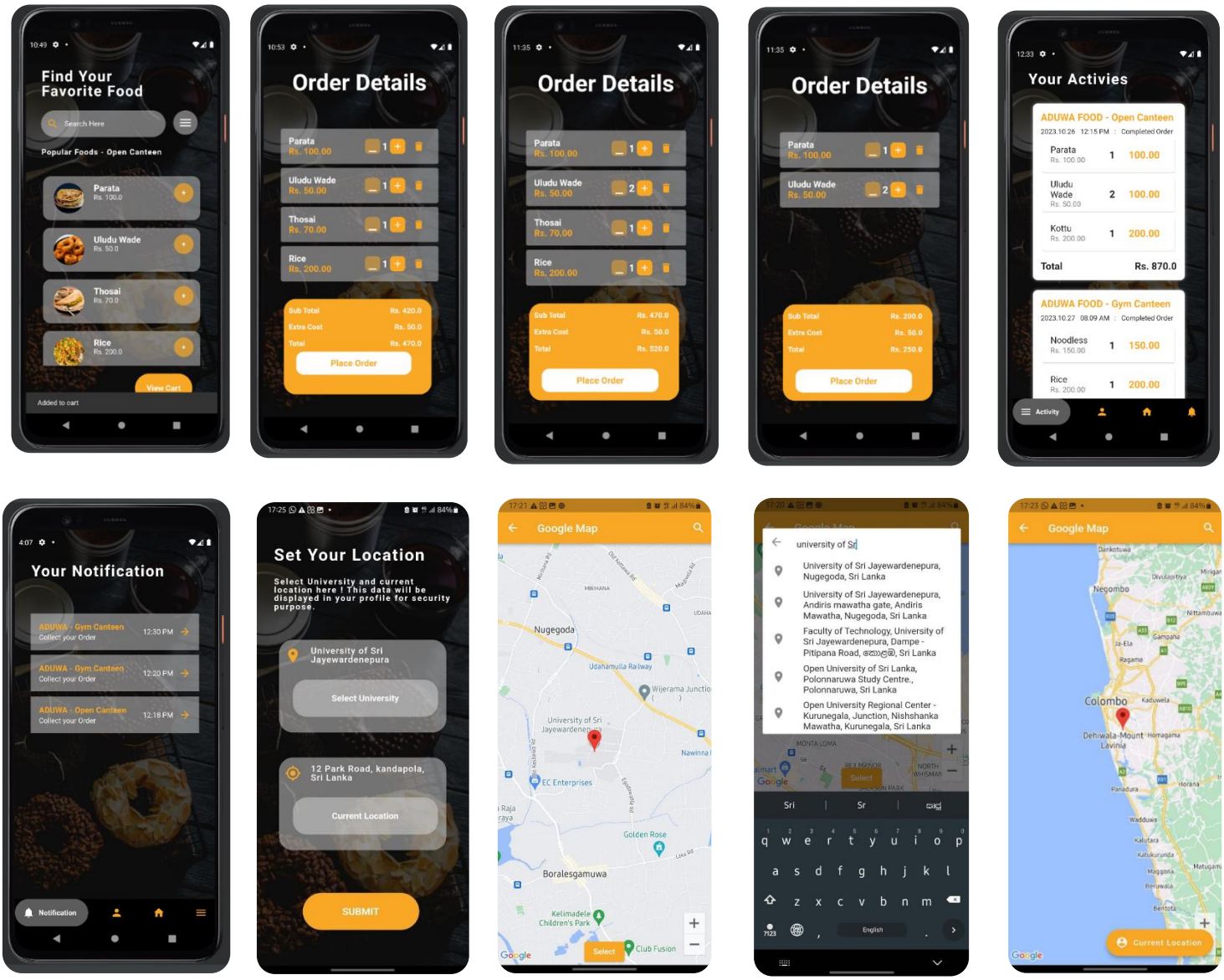


Figure 10:User Home screen and Features

Using this interfaces Students can

- Add necessary foods to their cart.
- View cart details and total cost of all current food items in cart.
- Update their food quantity and view updated total cost of all food items.
- Delete unnecessary foods from their cart list and finally they can place the order.
- View their history cart details with date and time.
- View their notification about their food orders.
- Select location and submit. Then student can view university canteen list.

## 5.3 PRACTICAL IMPLEMENTATION

Our project, Aduwa, addresses critical issues within our faculty, notably the canteen problem. Part of this initiative involves the adaptation and implementation of a specialized version of the Aduwa app tailored to streamline faculty canteen procedures. The app aims to tackle challenges such as queue management, menu accessibility, and order quantity. By providing user-friendly interfaces, the app is designed to integrate seamlessly into the existing procedural framework. Our objective is to enhance the canteen experience, fostering increased efficiency and improved services for faculty members and students.

- GitHub link: https:  https://github.com/HasarindaAmala/Aduwa_App.git

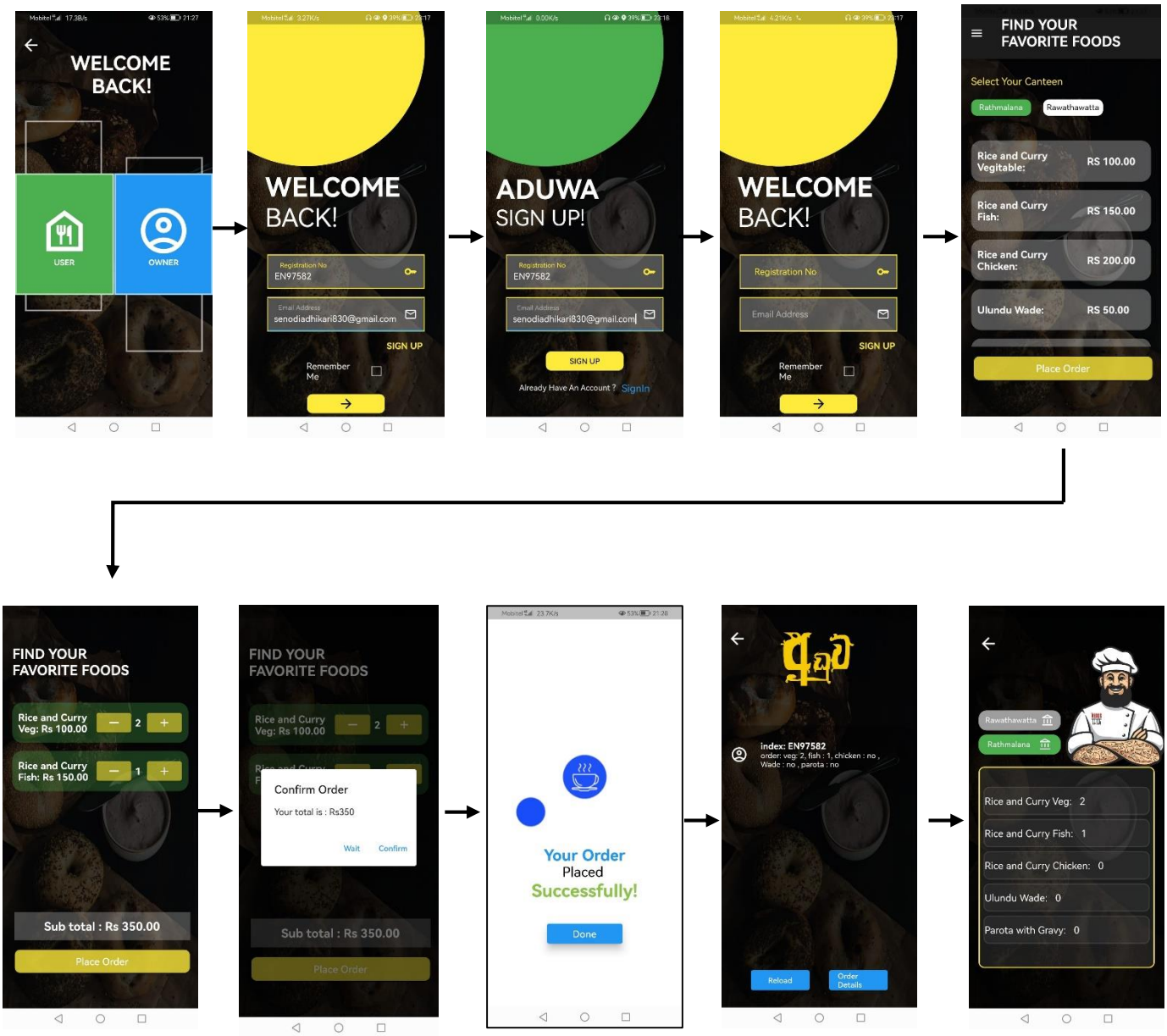


Figure 11:Practical Implementation

# 6. DISCUSSION

## 6.1 TESTING

The unit testing is being carried out under distinct functional units. To make sure everything works as it should, every module and component is put through separate testing. To ensure their accuracy, this entails testing methods, classes, and functions.After the final system has been integrated, Integration testing is done. It is being tested to ensure overall functionality. It ensures the interfaces and interactions between different parts in the app. It guarantees that the various components of the application function as a unit.Before making the programme available for wider testing, the main goals are to find any usability problems, get input, and improve its functionality. For that we do alpha testing using dummy users and dummy data.A chosen user is receiving the actual device with the app installed for beta testing, and the beta round is being completed once they receive a thorough in-depth description of the software system.

Our Aduwa app pass all the testing phases successfully and it works in actual devices in a great manner.

## 6.2 FUTURE IMPROVEMENTS

- Currently this app is only developed to work on University of Sri Jayewardenepura. Can expand the application to work on any university in Sri Lanka.
- Can add order processing tracking system.
- Can add comment section to get user feedbacks to the canteens.
- Can add online payment method to the app.

# 7. CONCLUSION

In this project, we used Python (Flask) for the backend and Dart (Flutter) for the frontend to create a mobile app for ordering foods in university canteens. For cloud storage, database administration, and authentication, Firebase and Google Cloud services were used. Thorough testing was used to guarantee the app's operation and user experience, including unit, integration, alpha, and beta testing. With room for future growth and innovation, the project successfully integrates modern technology to provide a better food ordering experience to university students.

# 8. APPENDIX

## 8.1 REFERENCES


- Firebase documentation (no date) Google. Available at: https://firebase.google.com/docs?gad=1 (Accessed: 14 July 2023).
- Documentation :  android developers (no date) Android Developers. Available at: https://developer.android.com/docs (Accessed: 14 July 2023).
- Google cloud documentation  |  documentation. Available at: https://cloud.google.com/docs (Accessed: 14 July 2023).


.