

SEG 2105: Introduction to Software Engineering

Semester Project Report: Mealer App

Group 27 members:

Mohamed Shakir, 300245079

Lama Koleib, 300259290

Jacob Zhang, 300231094

Carl Li, 300235679

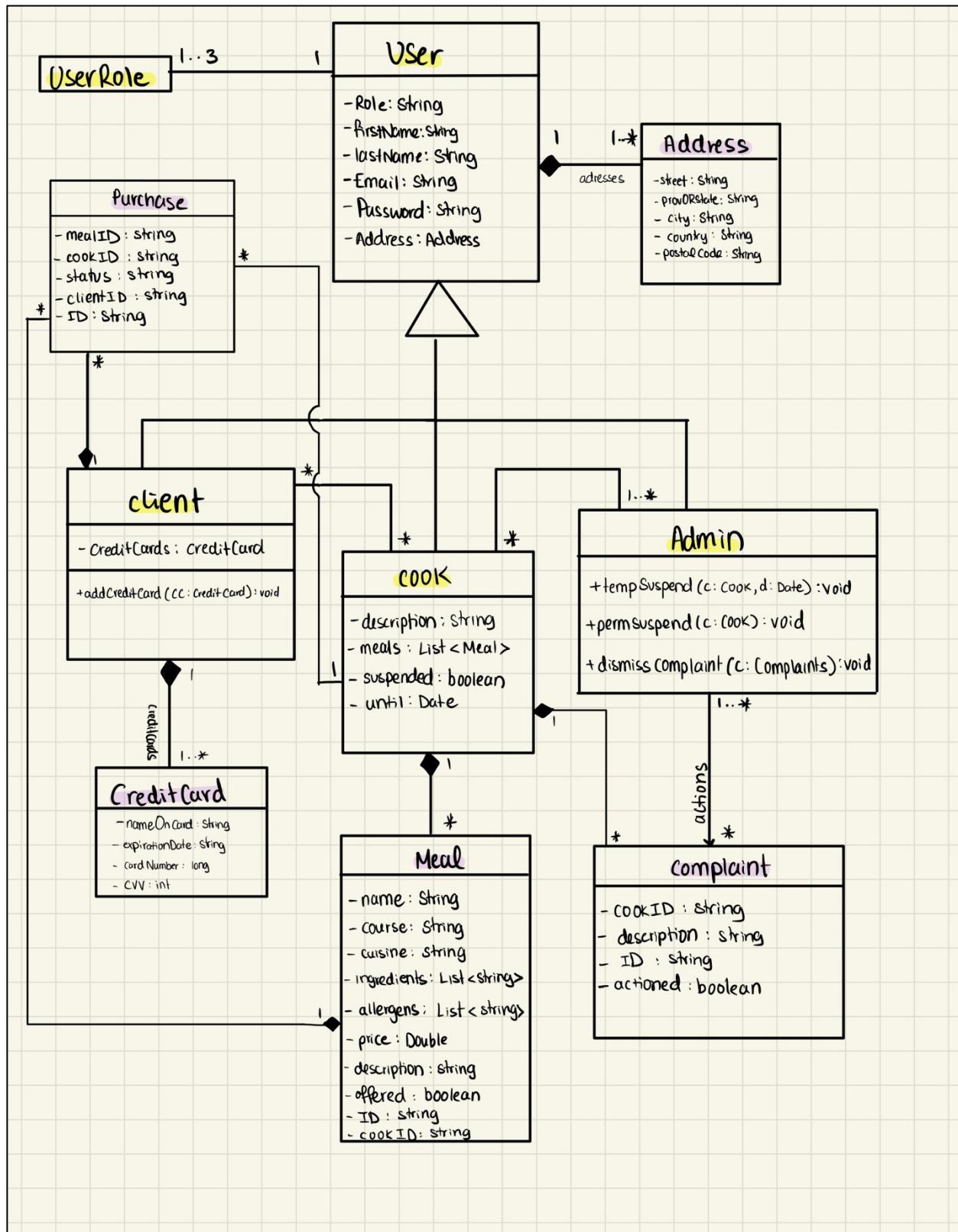
Yassine Sami, 300146704

Report submitted: December 7th 2022

Introduction

Mealer is an Ottawa-based meal sharing application where clients can purchase home-made meals from local cooks. A user on Mealer can create an account as a cook or client, or both, and can use the app as either. The application is created with simple flow and usability and is designed for all types of users with the best user experience in mind. A user opens the app and is directed to the login page where he/she could access a sign up as well. The user can then register as a cook or client. Both the cook and client are able to search for and order meals through the home page from the cuisine catalog, meal type catalog, or search through all the meals in the app. The cook has a “Manage Menu” feature in his/her profile and is able to add or edit meals that he/she offers. The app has an admin who can order meals just like clients, but can also view complaints submitted by clients and action them by either dismissing the complaints or suspending the associated cook.

UML class diagram



Group member	Contribution
Mohamed Shakir	<ul style="list-style-type: none"> - Integrated Firebase Auth and Firebase FireStore - Created CI/CD pipeline using CircleCI <ul style="list-style-type: none"> - Implemented Firebase Cloud Messaging for notifications - Logic for user logging on/off, and fetching user information from database - Validated all input fields and created Espresso and Instrumentation tests for testing all field validation and button functionality - Designed database and relationships between collections/documents - Implemented logic for fetching stored complaints from the database - Implemented logic for actioning complaints - Created logic for checking if Cook is suspended and redirecting him to a suspension method, as well as added logic to automatically detect and update status changes for cook suspension (i.e. suspension expiring) - Implemented meal management functionality using the View-Holder design pattern

	<ul style="list-style-type: none"> - Created search functionality and querying logic - Created rating system - Implemented purchase functionality and dynamic status changes/updates - Implemented complaint creation process - Developed push notification system so cooks, clients and Administrators are notified of relevant data changes (i.e. complaint created, purchase actioned etc.) - Created custom Notification Adapter to dynamically render various notification types for each user type, with room for expansion
Lama Koleib	<ul style="list-style-type: none"> - Designed all of the UI/UX of the app (all fragments, activities, navigation, dialogs, and cards) <ul style="list-style-type: none"> - Created logo and theme for app - Implemented designs made on Adobe Illustrator/Photoshop in Android Studio - Implemented cuisine and meal type cards on home page - Implemented bottom navigation bar for switching between fragments - Assisted with creating user flow for app - Created UML diagrams for all deliverables

Jacob Zhang	<ul style="list-style-type: none"> - Helped in the Creation/Implementation of the Cook class - Helped in the Creation and functionality of the Suspension system for Admin and Cook interaction
Carl Li	<ul style="list-style-type: none"> - Helped in creating the order notification and meal creating for Cook class - Helped in Purchase initial - Helped in creating Complaint inbox on Administrator Class
Yassine Sami	<ul style="list-style-type: none"> - Initial work on Meal class - Initial work on Complaint class - Initial work on Complaint Adapter & Domain - Assisted in Admin complaint handling & Cook suspension screen functionality, implemented DatePicker for suspension duration selection

Screenshots

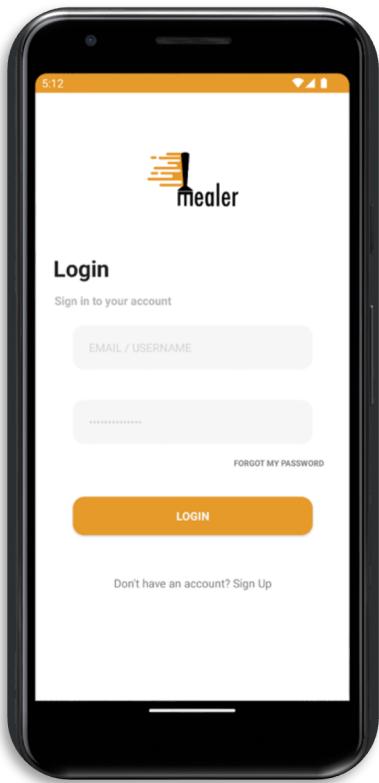


Figure 1.0: Login page

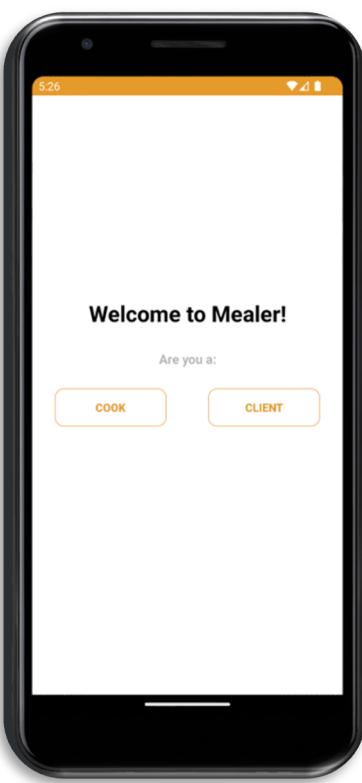


Figure 1.1: Select user type for sign up

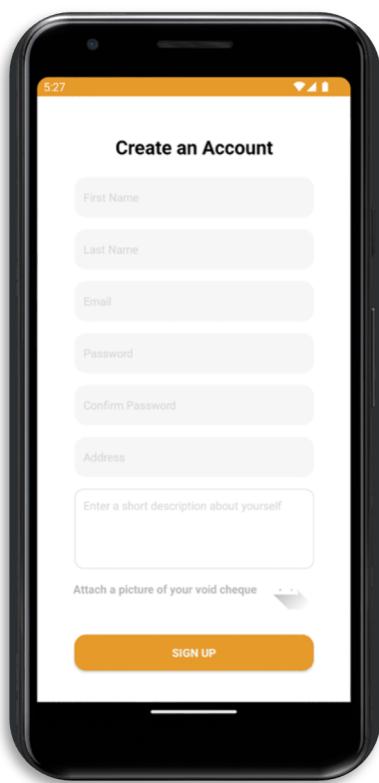


Figure 1.2: Sign up for Cook

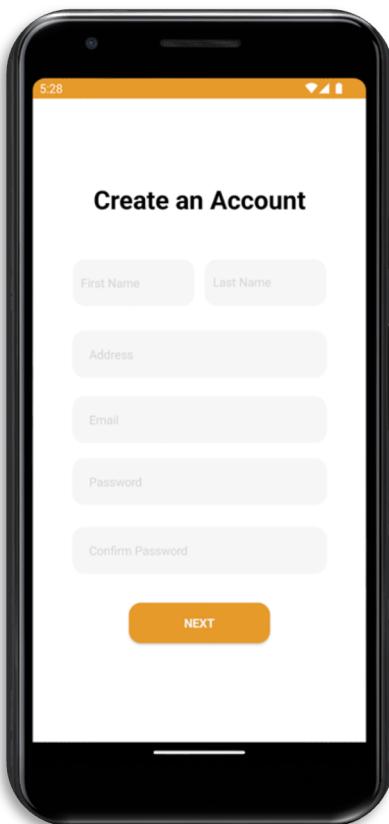


Figure 1.3: Sign up for Client

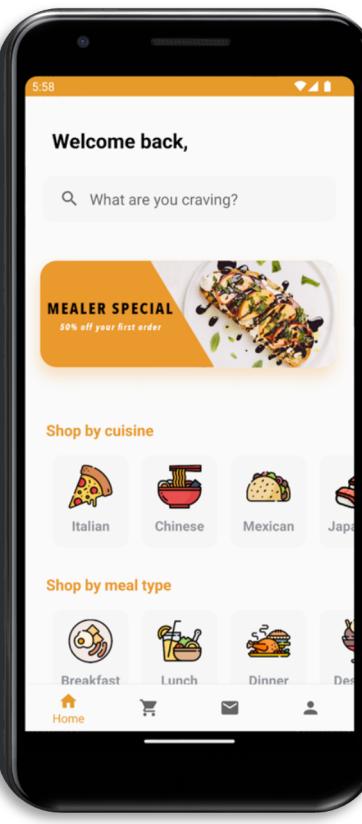


Figure 1.4: Landing page, Home

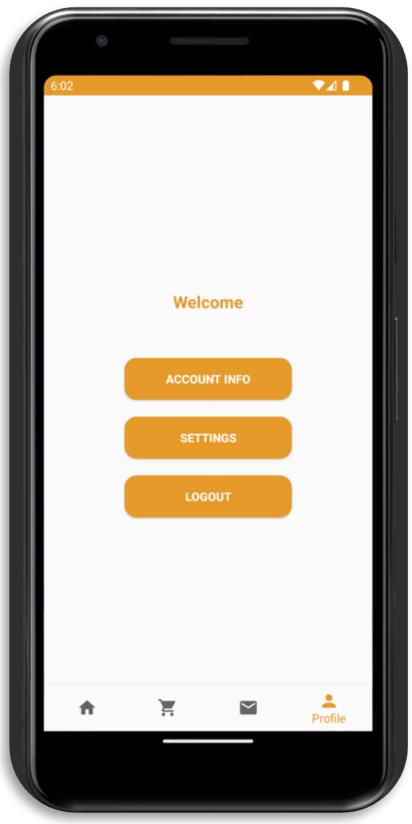


Figure 1.5: Profile: Admin / Client

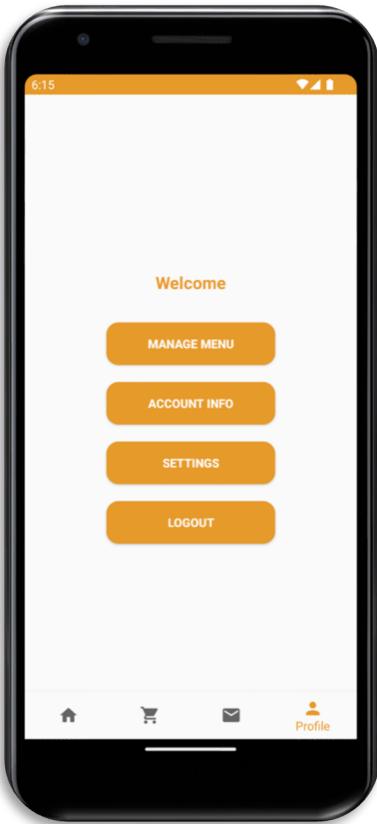


Figure 1.6: Profile: Cook

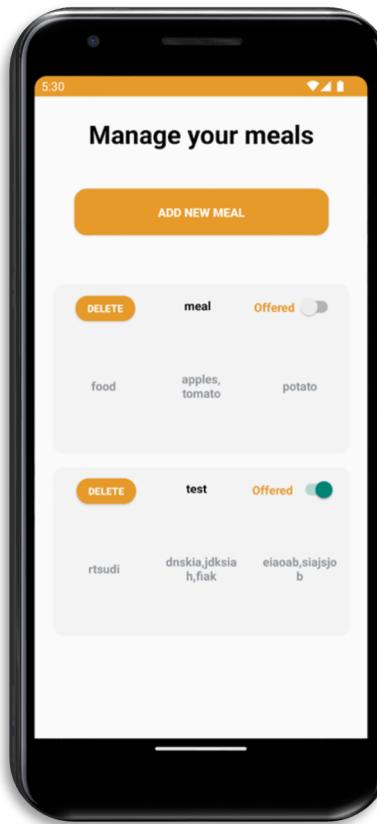


Figure 1.7: Manage your meals

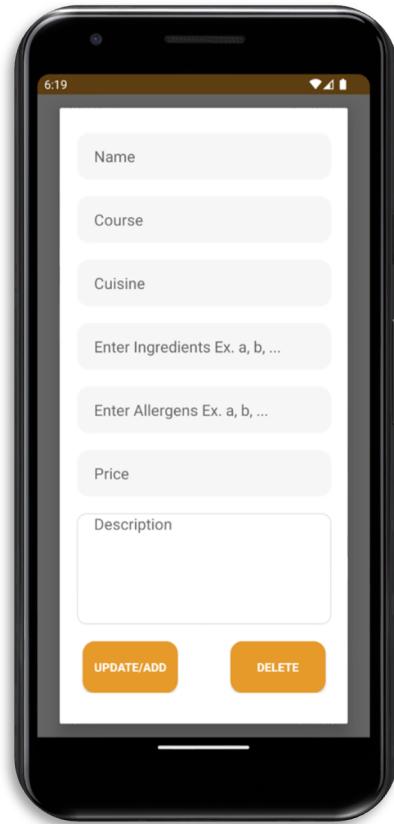


Figure 1.8: Add or update your meal

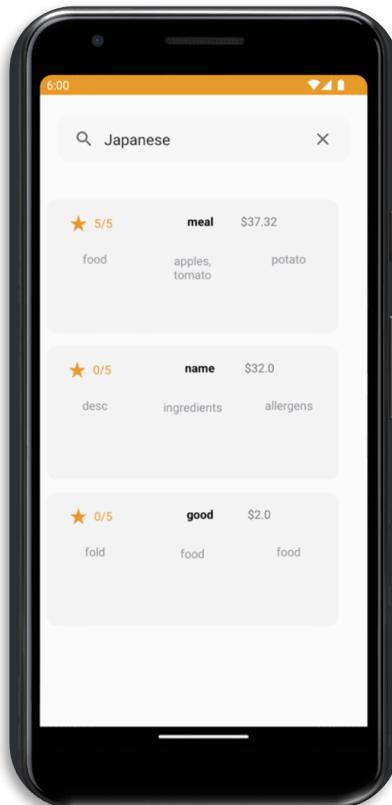


Figure 1.9: Search for meals using search filters

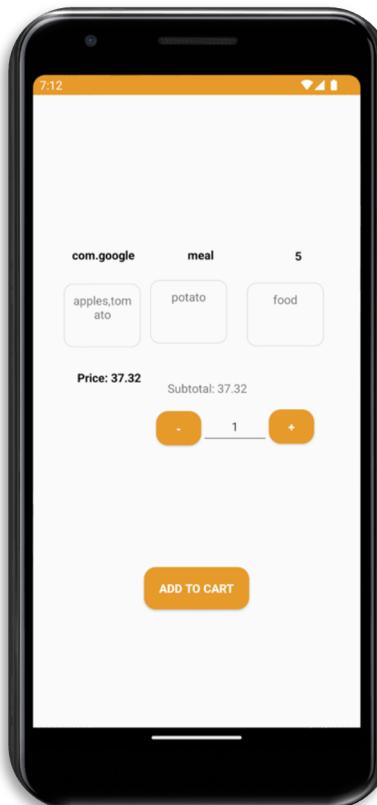


Figure 1.10: Meal details



Figure 1.11: Meal added in cart

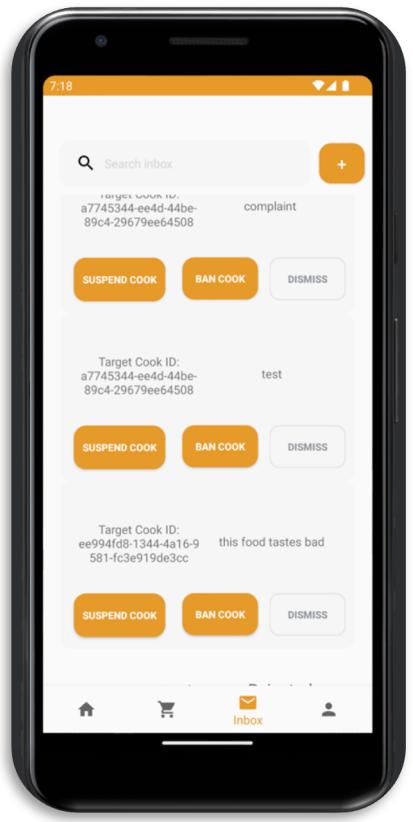
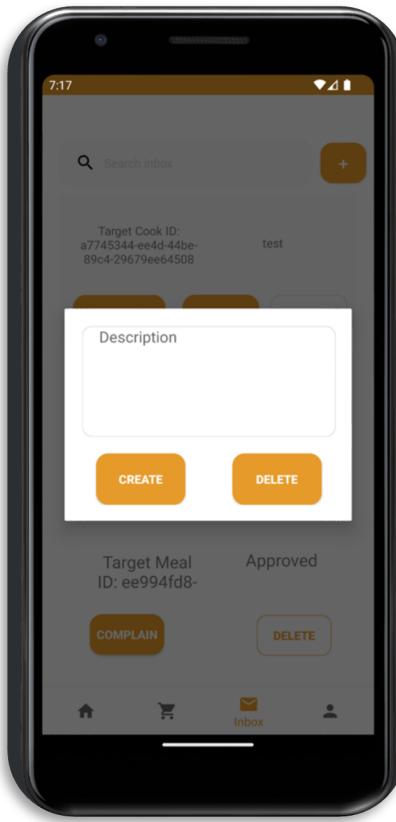
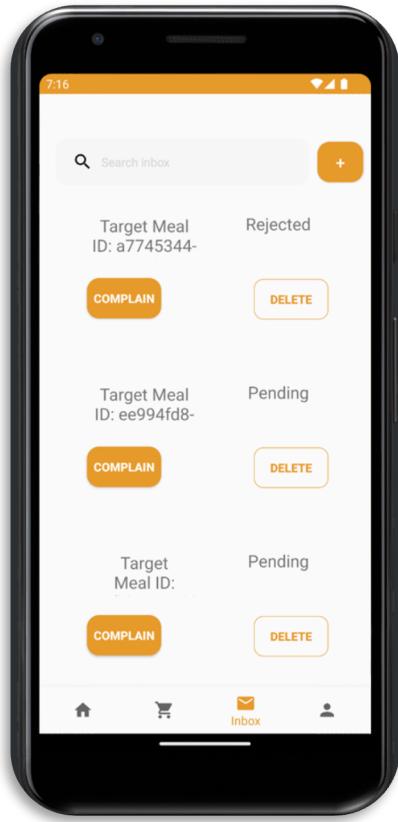


Figure 1.12: User Inbox after ordering

Figure 1.13: Submit a complaint

Figure 1.14: Admin Inbox viewing complaints

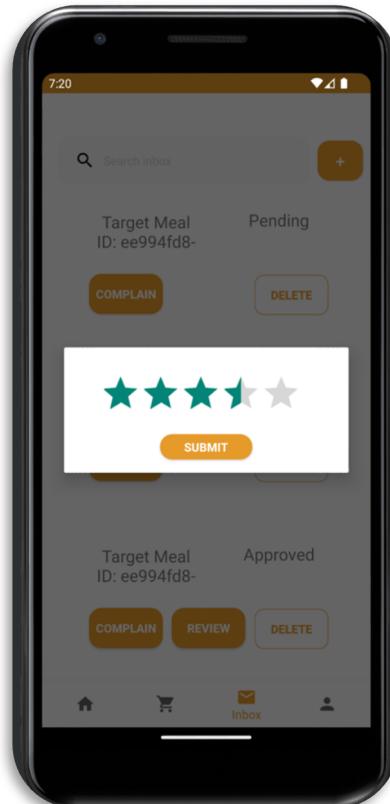


Figure 1.15: Suspending cook, date picker

Figure 1.16: Rating cook's meal

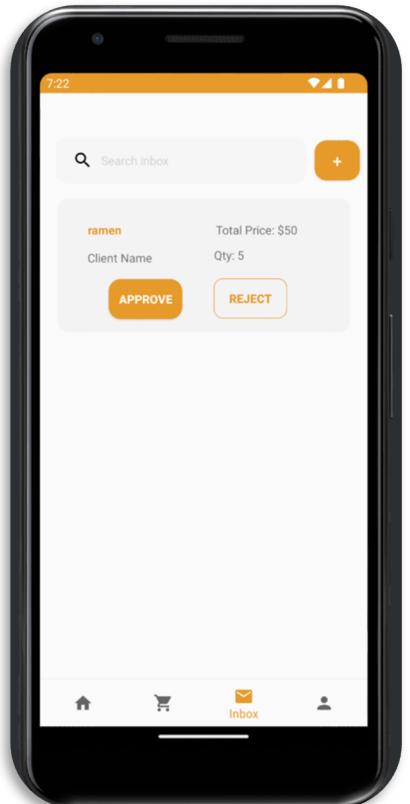


Figure 1.17: Cook's inbox after user orders



Figure 1.18: Cook's account information and rating

Lessons learned

We've learned many new things about dealing with databases and mobile app design, especially when trying to integrate design patterns that were introduced in class (like person-role, generalizations, etc.). The UML diagram helped us organize our ideas on how we wanted the app to function and we tried following design guidelines to design a modern looking android app which is simple, elegant, and user-friendly. Working on this project will definitely help us in future classes like Databases and User Interface Design, and even in our future jobs, if we encounter a similar project to work on.