universidade de aveiro

SendASnack

Android Project

Pedro Monteiro 97484

Eduardo Fernandes 98512

Introdução à Computação Móvel

June 23, 2022

Contents

1 Introduction

According to Introduction to Mobile Computing' curricular plan, this report is the result of the execution of the second project, which we will carry out in parallel with the Software Testing and Quality course. The main focus is to create a rider's application and the name we chose was **SendASnack**.

2 Motivation

After some meetings, we found it appropriate and challenging to make an application linked to the group project of the Software Testing and Quality course. While in TQS we have costumers and restaurants owners sides, here we aim to develop an application focused on the rider, where he can access customer orders and make deliveries.

3 Solution

We started by thinking about our solution, with the whole team, including the TQS team members. After some research and sharing ideas we designed a prototype in figma and decided that the application will consist of:

- orders page: where the rider, after logging in into the application, can see the current orders and pick one to deliver;
- deliveries page: where the rider has access to a map where the delivery address is marked. This map was built using the Google API;
- profile page: contains information about the rider, such as name, email, phone number and address.

These pages were made using fragments and RecyclerView's. With fragments we can represent a reusable portion of ours UI app. We can handle its own input events and define and mangage its own layout.
In the orders page we used RecyclerView's to display the order's informaiton on the screen. This is very helpful because you just need to supply the data and define how each item looks, and the RecyclerView library dynamically creates the elements when they're needed.
In the deliveries page we tried to show the pick up and delivery location per order, but this just not went well. We had some problems initializing the fragment with the necessary data, and due to the lack of time, we decided to just implement a map with some other location displayed.
In the profile page we just have some personal information about the person that is logged in the application such as name, email and location.

3.1 Features

The main features implemented in our application are:

- Map: allows to view the place of delivery of the products;
- Cloud Firestore: used to store important information online for the correct use of the system;
- Firebase Authentication: this Firebase module was used to perform the authentication of the system.
- ROOMDatabase: used to store information offline;

4 Architecture

There are a ton of moves toward utilizing the Architecture Components and executing the suggested engineering. The main thing is to make a psychological model of what is happening, understanding the way in which the pieces fit together and how the information streams.
We choose to implement some components that were already mentioned during this class. One of these components is the ROOMDatabase. This was used to store locally some important information such as orders and their products.

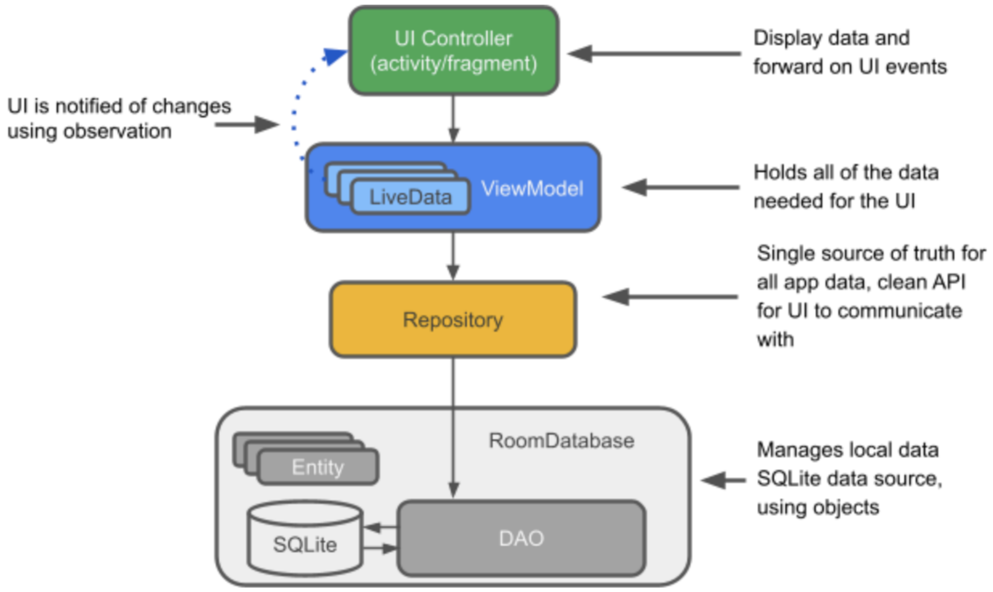

UI Controller (activity/fragment)
Display data and forward on UI events
UI is notified of changes using observation
LiveData
ViewModel
Holds all of the data needed for the UI
Repository
Single source of truth for all app data, clean API for UI to communicate with
RoomDatabase
Entity
SQLite
DAO
Manages local data SQLite data source, using objects

Figure 1: Simple architecture scheme

One thing that is not present in the figure is the Cloud Firestore where we keep our information stored in a online database.
This module communicate with the ViewModel and Repository, providing them with information that, more specifically the Repository with help of DAO, stores locally in the SQLite database.

5 Overal Assessment

After completing the application, we thought we did a good job and learned how to work with Android Studio, although, due to lack of time the final product was not what we were expecting, not having completed all the features we wanted.

5.1 Achieved objectives

As main objective we wanted to develop a functional and ready-to-use application. We think we have achieved this goal, having an application that allows drivers to select an order to pick and deliver it.

5.2 Issues Found

The main problem we ran into was the lack of time due to the high load of projects, so we were not able to implement features that we wanted to implement with more time.

Besides that we found it difficult to connect our application to the API built in TQS, so we choose to use firebase.

6 Tutorial

Starting SendASnack application, the login page will appear, asking for the username and password of the rider:

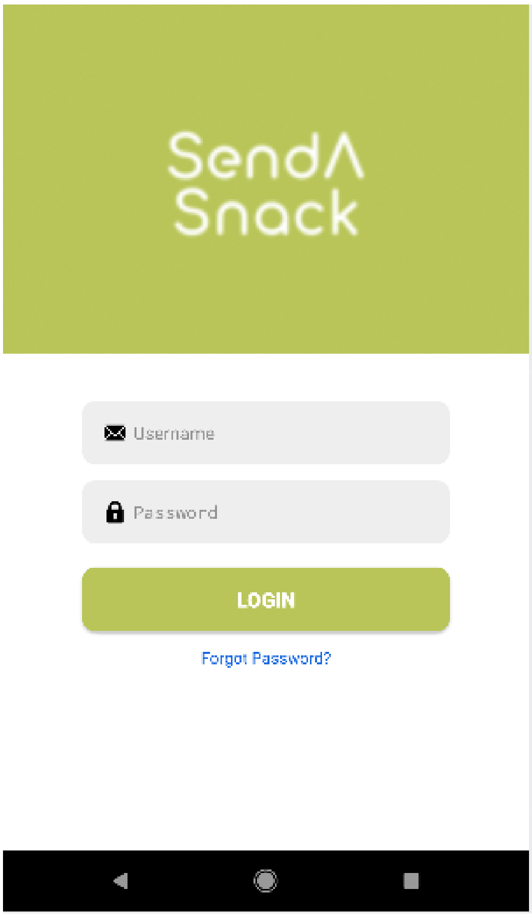
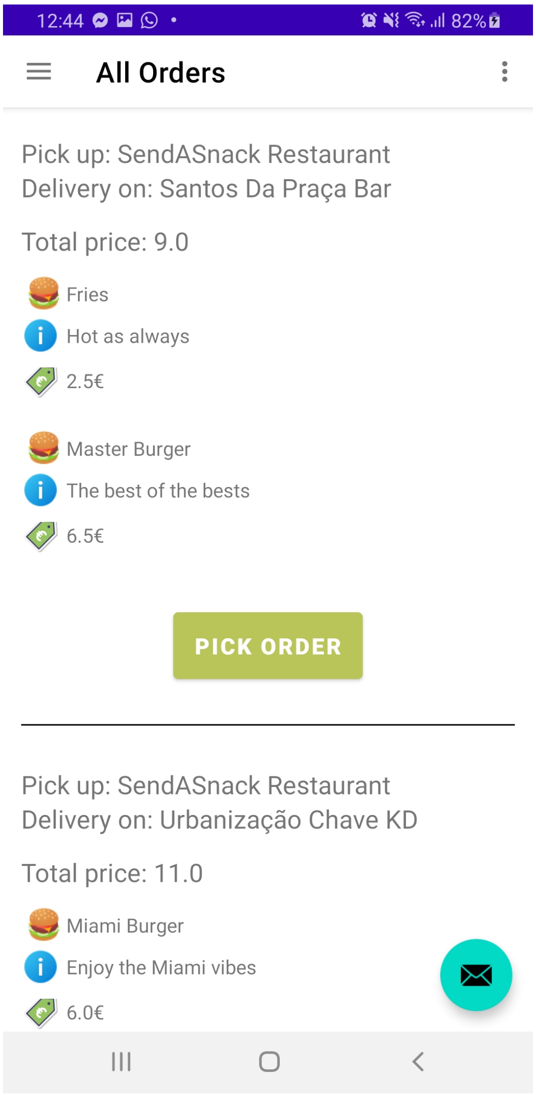


Figure 2: Login Page

To log in, the application makes a GET request to Firebase, checking if the entered credentials exist.
After logging in, rider can see the home page, where all pending orders are, and where can see the pickup and delivery addresses, the product order specification and the total price.



Pick up: SendASnack Restaurant
Delivery on: Santos Da Praça Bar

Total price: 9.0

🍔 Fries
ℹ Hot as always
2.5€

🍔 Master Burger
ℹ The best of the bests
6.5€

Pick up: SendASnack Restaurant
Delivery on: Urbanização Chave KD

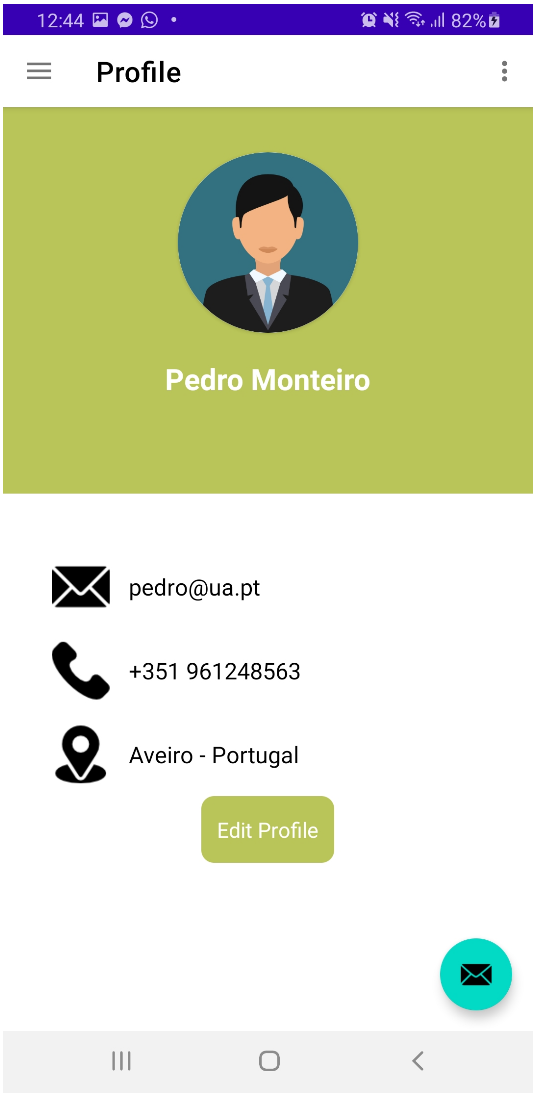
Total price: 11.0

🍔 Miami Burger
ℹ Enjoy the Miami vibes
6.0€

Figure 3: Orders Page

On this page, the rider has the opportunity to accept to make a delivery, by clicking on the ”Pick Order” button. After that, he can see the delivery address on the map page.

There is also a simple profile page where the rider can see their information.



Pedro Monteiro

pedro@ua.pt

+351 961248563

Aveiro - Portugal

Figure 4: Profile Page

7 References

[1] “Android Firebase” https://firebase.google.com/docs/android/setup?hl=pt

[2] “Recycler View” https://developer.android.com/guide/topics/ui/layout/recyclerview

[3] “Android Fragments” https://www.javatpoint.com/android-fragments