
# Computação Móvel

ParkBuddy - Android Studio Project

Work made by:
Afonso Teixeira 93170
Manuel Couto 93285

# 1 Authors

Manuel Couto 93285

Afonso Teixeira 93170

You can find the repository of our work here https://github.com/79AFonso/ParkBuddy_AS

# 2 Motivation

Who has never scratched his head thinking where he did his parking?
The motivation for this project came through experiencing this problem ourselves. So basically the problem is when you park your car and then you can't seem to remember where you parked it. We also realized one thing that would improve our parking experience was not only being able to pay online for the parking ticket but also keep track of the amount the ticket will cost . So we thought of an idea that we would like to use, in order to never forget where we parked it, since there's not a well known app trying to solve this issue. And the ones that exist, seem to have some sort of feature missing that we thought was crucial. Adding to this idea we also implement a system that allows us to keep track of the amount of money the parking ticket will be and allows us to pay for it using only the app.

# 3 Project Description

This project consists of creating an app that allows the user to store the GPS location of several cars. So when you go out and park your car somewhere you can take a picture of the parking spot and also save the GPS location of your car. The app tries to resolve a problem that is quite common, but it can be used for different purposes, for example when you want to give your car location to a friend of yours. You can share the car location through scanning a simple QR Code. The app also has parking payment features like keeping track of the amount of the fee or actually paying for private parking.

# 4 Features

These are the features that are implemented in the app

- Biometric sensor to do authentication;
- QR Code to share car location;
- QR Code Scanner;
- Camera to take photos of your parking;
- List of your parked cars;
- Removing cars from that list;
- Information about your parked car;
- Map so you can get an idea where you are and where you need to go;
- Markers that give your location and the location of your car;
- Walking Path between the user and the car;
- Pedometer keeping track of the steps you've made;

- Track the amount we will pay in a paid parking
- Changing password of account
- Changing email of account
- Pedometer and timer working then app is on pause
- Proximity sensor, changing the brightness of the app

# 5 Architecture/Technical options

In this project, we utilized Firebase, a mobile and web application development platform developed by Google, to take advantage of several of its services including the real-time database, user authentication, hosting, and storage. In particular, we used the real-time database to store information about vehicles such as the model, license plate, and coordinates, and the user authentication and storage services to manage user accounts and store images.

We also utilized Glide, a library that allows developers to easily load and display images from various sources in their Android app.

To implement a parking timer that would continue running even when the app was closed, we used a foreground service. Foreground services are given higher priority by the Android system and are less likely to be terminated because they are considered essential for the user's current activity. In order to create a foreground service, a notification must be displayed to the user to alert them that the service is running in the foreground.

We used the Google Maps API to create the map feature in our app.

Additionally, we designed a logo with various versions including a color version, a black and white version, and a version without a background. We also incorporated the same shade of

red from the logo throughout the app to improve the overall aesthetic and consistency of the user interface.

# 6 Overall assessment

All objectives of ours were achieved. We even got to add more features that were not thought of at the start of the project. We were able to use 6 technical features in total:

- Biometric sensor, for authentication purposes;
- User authentication;
- Camera, to take pictures of your parked car;
- QR Code Scanner, to share the location of your car;
- GPS, to save the location of your car and to know where you are;
- Pedometer, to keep track of the amount of footsteps taken after parking the car, so the user had an idea of the distance they were from the car;
- Timer, to calculate the payment fee that works even if the app is closed.
- Payment simulation page;
- Map with walking path;

Given the time constraints, we had to prioritize certain tasks and were unable to complete others. **For future work**, we encourage you to integrate a reliable payment method, such as Google Pay or PayPal, into our platform. We also aim to complete the update feature for the list of parked cars and make the steps page more user-friendly by making it autonomous (i.e., removing the need for buttons). Currently, users can update vehicle data by performing a long click on a card, but this feature has not yet been connected to the Firebase database.

# 7 Contribution assessment

The work was done by Afonso Teixeira and Manuel Couto, both did the same amount of work. Both gathered a lot of information and in order not to have irregularities, we had meetings where we grouped up and decided what we wanted to implement. Then, we separated the functions each one had to do, so no one would end up doing more work than the other. So the percentage of work each one  has done is 50/50

Afonso’s part:

- Profile page;
- Map and GPS;
- Biometric;
- Pedometer.

Manuel’s part:

- QR pages;
- Path;
- Pedometer;
- Payment pages;
- Park list page.

The UI was distributed evenly.

# 8 User Manual

The usage of the app is pretty intuitive and easy to use, nonetheless, we are going to do a manual for better understanding of how to perform each feature.

- We did a personalized launch screen and personalized icons/logos for our app.

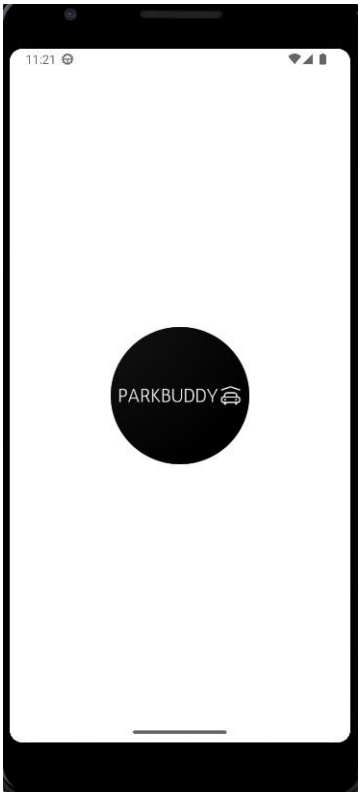
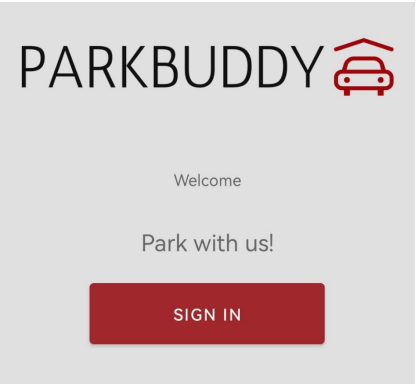



Fig1 - App launch and apk icon.

- This is the Login page, here you can authenticate your account by pressing sign in.

Once pressed the sign in , you will have to authenticate your device through the fingerprint (biometric sensor)



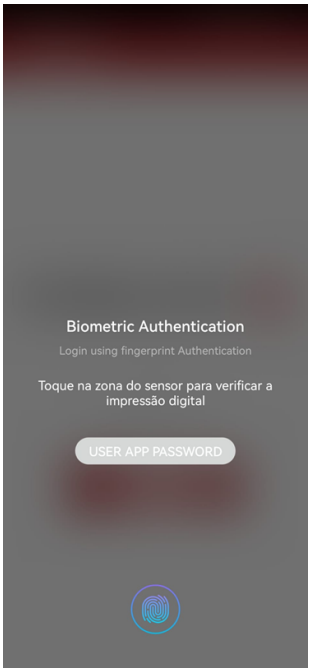
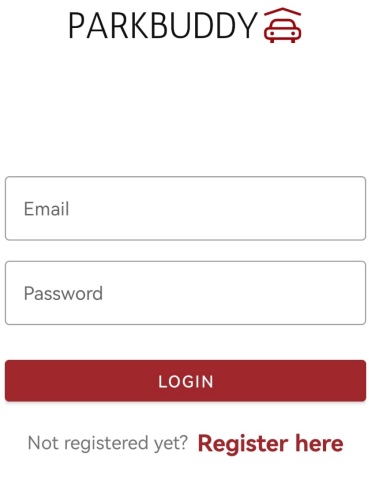




Fig2 - sign In

Fig3 - Biometric sensors

After passing through the biometrics you will have to enter your account details.

Instead, if you press “Register Here” you go to the register page



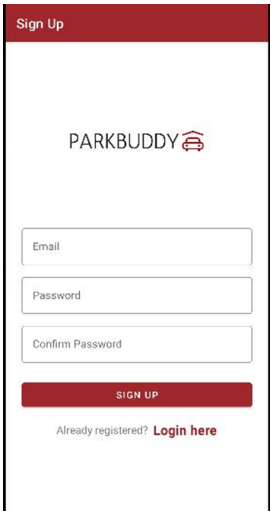




Fig4 - Login Page

Fig5 - Register Page

- This is the main page, where you can navigate to the several screens just by clicking the button of the screen you want to access

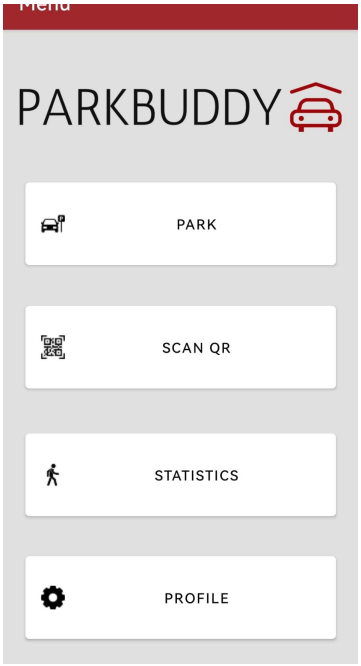
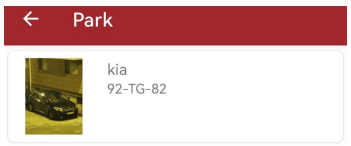



Fig6 - Menu Page

Park will take you to the park page (Fig3), ScanQR will take you to your camera so you can scan the QR code, statistics will take you to a footstep counter page (Fig and Profile will take you to the profile (Fig7)).

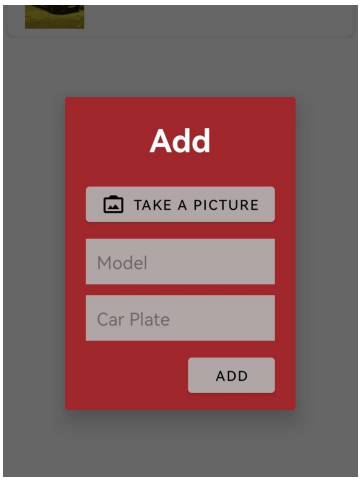
- This is the park page, where you can add parked cars to your list, by clicking the "+" button on the bottom right corner and remove from the list by simply sliding the car you want to remove.




You can go to the screen of each parked car just by pressing the card of the car you want to open.
You can delete parked cars just by swiping.

Fig7 - Park Page

- This is the popup you get when you press the “+” button on the Park page, here you can  add a car to your list.




Your localization is saved automatically, you just need to provide the car name and the car plate that you want to associate and of course take a picture of the local.

Fig 8- Add car

- This is the info page of each parked car. It contains the photo you took, you can share the QR code so other people can know where your car is, you can check the map and even pay for your parking.

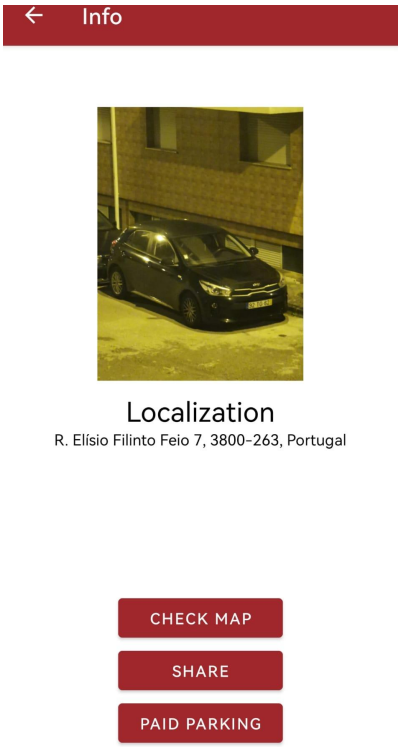



Fig 9- Info page

- This is the map where you have two markers the red one being the location of your car and the blue one being your location, when you press the walking icon you will

get the walking path between the two markers, if you press the other button which have a icon of a marker it will change the map camera between your position and the car position and zoom a little to make it easier to find.


Fig 10 - Map page

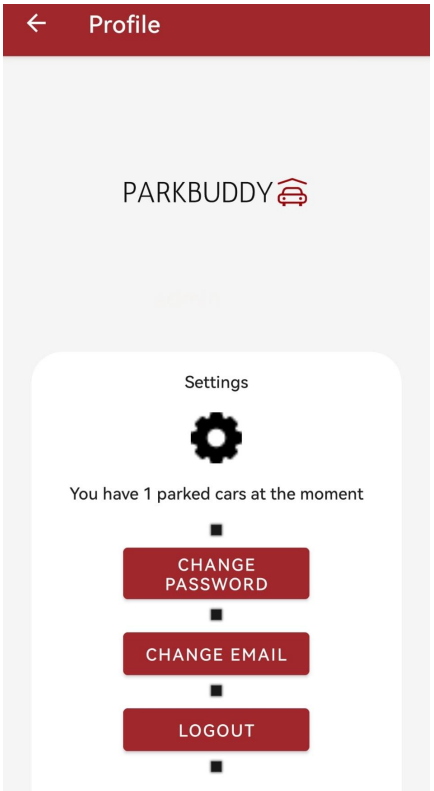

Fig 11 - Map Page after pressing walk path button


Fig 12 - Map Page View when marker button pressed


Fig 13 - Map Page showing marker “Im here” title


Fig 14- Map Page showing Vehicle Plate

- This is the profile/settings page, where you can logout or change either your email or your password.
    - In this page we have implemented the proximity sensor to reduce the brightness.



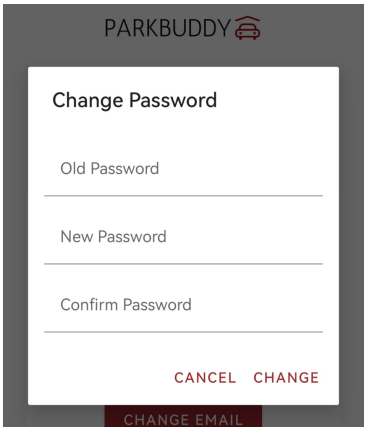




Fig 15 and 16 - Profile page and Change Password

- This is the Statistics Page where you can start a footstep counter to keep track of how many steps you've taken since parking the car.

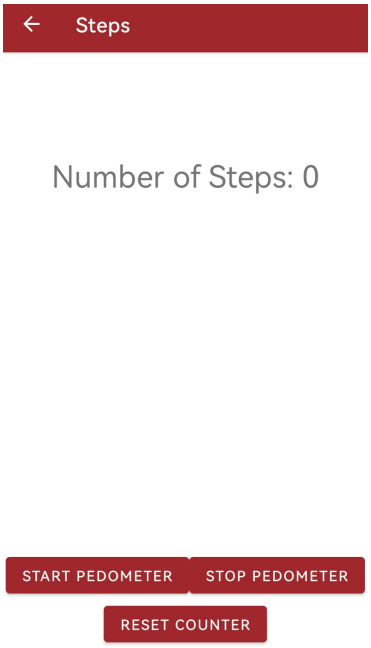
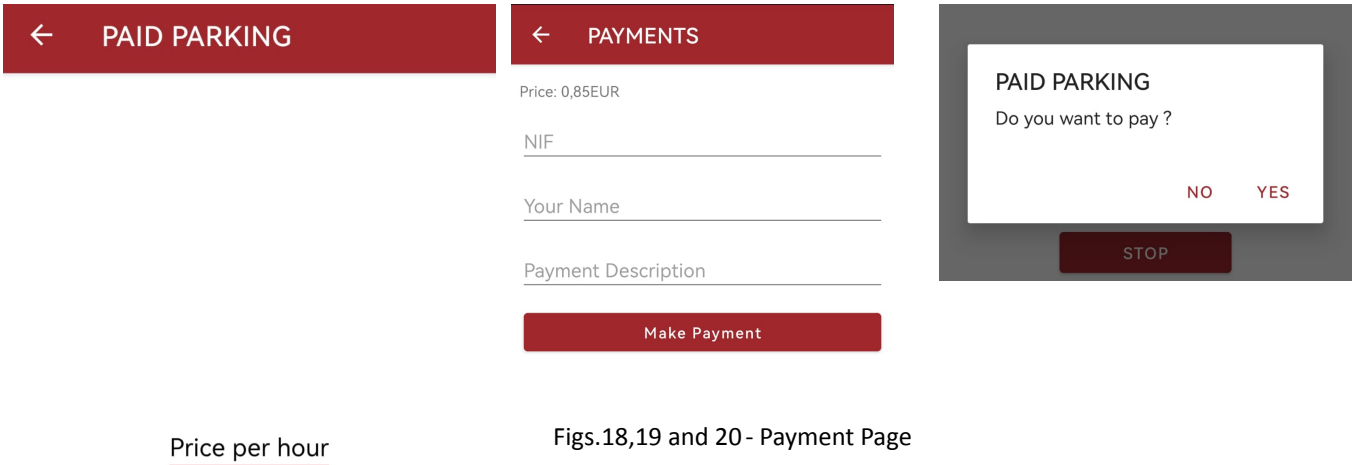



Fig.17 - Pedometer page

- This is the payment page. Here you can make an estimate of how much you will pay for parking , you need to start the timer when you park and when you're ready to pay just click stop and it will popup a question asking if you want to pay. If so you will be taken to another page to perform the payment.




Figs.18,19 and 20 - Payment Page

Fig.21- Timer Notification

- If we close the app after starting the parking timer the timer stills working and we receive a notification of the foreground service.

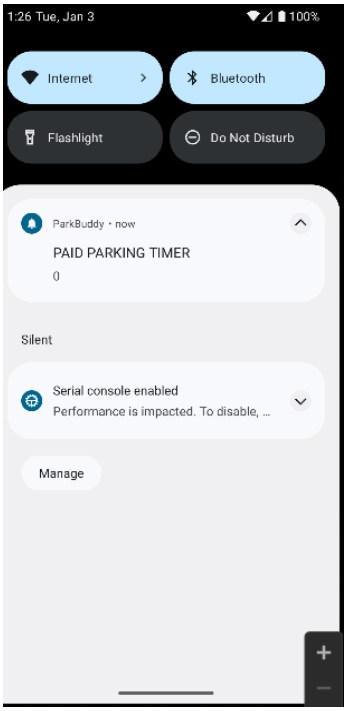
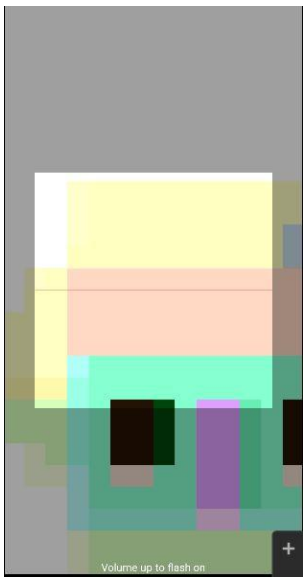



Fig.21- Scan Page

- This is the Scan Page, where we scan the Qr code from other users and we will be redirected to a map that will show where his car is parked.




- If we press the volume buttons we can turn on/off the flash and when the scan is completed we hear a beep sound.

# 9 Bibliography

https://developer.android.com/docs