

Application Development for Mobile Devices Project 2021

POCKET DR

team: **Alexandra Fernandes,** xferna17
André Raro, xraroa00
Bruno Lopes, xlopes08
Mariana Ribeiro, xribeir06

Project goal

This project is a part of the TAMa's course at VUT-FIT. It was proposed to us the construction of a mobile application that could be useful and have into account the several aspects that should be considered in such a task, such as the main use-cases, user-experience, overall design of the application, etc. Keeping this in mind, we tried to make an application that helps people in need.

The goal of our app is to help people keep track of medicine intake, being one of two main actors: either a caretaker or a dependant. The first one is in charge of someone and can track their personal medicine intake and the medicine taking of one or several dependants, to see if everything is okay. As for the second one, they use the app to receive the medicine taking reminders and also inform their caretakers that everything is in order.

Used Technologies

- For the development of this project we based it in java and Kotlin, which is a programming language based in java and also compatible with it.
- We chose Android Studio as our IDE (Integrated Development Environment). A great feature that this IDE has is that it includes an Android Emulator integrated, using Android Virtual Devices (AVD) provides an easy way to test our app.
- For the authentication and database, we used Firebase and a few of their services namely, Firestore Database to store our data, Firebase Cloud Functions to deploy functions that deal and update this data accordingly to actions and Firebase Authentication to provide authentication through Google sign-in. Further in the development, we realized we did not need Firebase Cloud Functions.
- To create our logo we used a platform called Canva.

Used Resources

The resources used were all from the internet, mainly videos, written tutorials and documentation that helped us with the base of the application.

Specifically:

- a few of the Kotlin video tutorials where: Build a Complete Android App with Firebase - Full Course with Kotlin) , Build Alarm App in Android With Kotlin and Android Push Notification Using Firebase Cloud Messaging in Kotlin — GeeksforGeeks that helped us.
- We also consulted a lot of documentation from the Firebase Documentation in order to get all of the synchronization and database working.
- To create our logo we used a platform called Canva.
- Finally the documentation from android developer's community, which can be found in this link

Most Important Achieved Results

- First of all the synchronism that was achieved between databases and app fully in real time (figures 1 and 2). Every change on the database is automatically updated on the RecyclerView of any item, being it medicine or users/dependants.
- Secondly, the relationship between caretaker and their dependants (also shown in figures 1 and 2) that in the beginning was hard, due to the need to do sequential operations that have an asynchronous nature.
- Lastly, the push-notifications that are sent as reminders and the overall flow and aesthetic of our app.



Figure 1: Database



Figure 2: Corresponding real-time view of 1 on the app
The account signed in at the moment is Ne Raro

Controlling of Created Program

Firstly, it is necessary to login using Google Sign In in this case, it is only necessary to press the button and a Pop-up dialog is open to insert the email of the account and password. The app automatically detects if the account is of a caretaker or a dependant.

After this, the first thing the user sees is the list of the meds they need to take, in case it is a caretaker they can one of two pages: personal or dependants, with their correspondent meds' list. Here there is a button where it is possible to add a new medicine, each medicine needs a few data that is displayed in figure 3

On the left, there is a navigation bar with different buttons as seen in figure 4. A caretaker can see their dependants (in the correspondent activity it is possible to see a list - displayed with a RecyclerView - of their dependants and there is also a button to add new dependants). On the other side, a dependant can only see their personal meds.

Each user then receives notifications on their phones as reminders to take their medicines, as shown on figure 5.

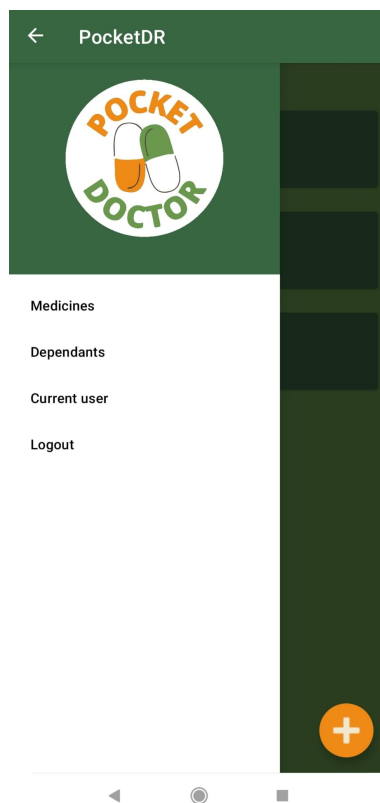


Figure 3: Personal meds screen

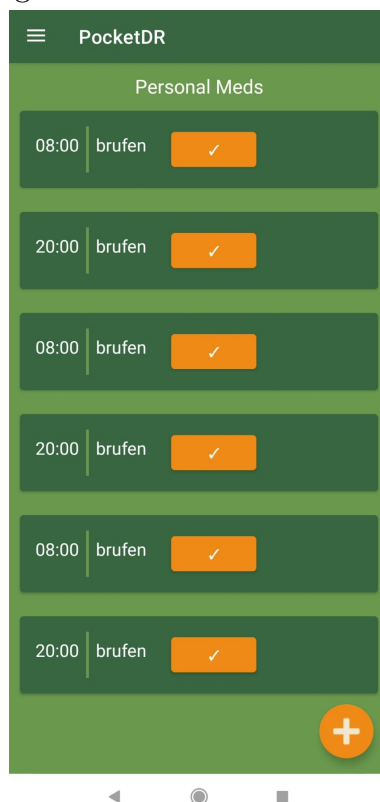


Figure 4: Caretaker's navigation bar

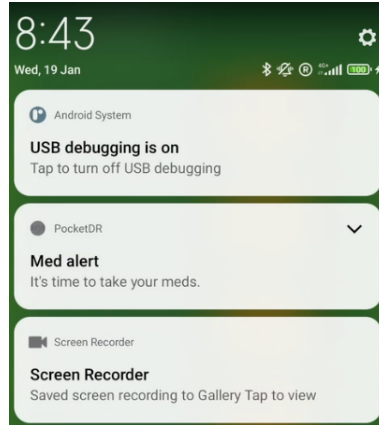


Figure 5: Notification sent as a reminder

Experience with the Selected Platform

None of team members had programmed a mobile application before, so we had to learn how to use all of the resources from scratch (learning Kotlin language, how to use Firebase and all of the database management) . This was a fulfilling experience, since it made us search for every solution and do a lot of research until we could understand how to use each platform.

Android Studio and Firebase are actually very user-friendly platforms, even though it took some time to be familiarized with these platforms, all of us learnt a lot.

On the other hand, Kotlin can have some troubles when mixing versions and with imports, which lead to some tricky aspects when building the app. Firebase also has a down side, when using the cloud functions it is necessary to pay to have the full access to their feature, like deploying the database management functions (there functions also had to be written in JavaScript, which we had to learn a little bit also).

Distribution of the Work in the Team

Alexandra Fernandes: 25%

André Raro: 25%

Bruno Lopes: 25%

Mariana Ribeiro: 25%

Everyone was equally committed in this work. We jointly brainstormed and discussed the idea in order to find the key features that we wanted to implement. Overall, we worked together during the development time, but being more specific Mariana and André mostly focused on front-end, creating all the necessary layouts files for the Android Activities, taking care of user input errors and ensuring a clean and consistent aesthetic. While Alexandra and Bruno mostly focused on the back-end, making sure the synchronization between databases and app were functional and the data-fetching was smooth and correct.

What Was the Biggest Challenge

The biggest complication was to define the goal of our application that means besides choosing the theme and what the app will do, we needed to determine what was possible to do and what

would be useful for our users, having into account the daily context of our soon-to-be clients. In other words, defining the main use-cases and features according to them, and also taking into account the time available and our knowledge.

So the development of the app was not the hard part, but defining and choosing what to use, was definitely the hardest part. After starting, assuring synchronization seemed to be the greatest challenge. Working with authentication, database and app synchronization was hardworking and it was still not perfect.

Experience Gained From the Project

The development of the application was an introduction to the development of mobile applications for us, we had never worked with emulators or mobile applications before and learning in a self taught way was a preparation for the reality of the work world.

Being more specific, we got a better perception the impact of the user's experience has on an application development process, which is in fact huge. The way of thinking is totally different, because it has in mind not only the functionality but also the experience the user is going to have, which lead to the fine-tuning of ideas and then a better development. Besides this we also gained a lot of knowledge about real-time databases, programming in Kotlin and Java, amongst some other technical skills.

Auto-evaluation

Overall, we think our project is at 50%, it needed more time to fine-tune it since we took a lot of time thinking about what we wanted to do and how to do it. It is not a final and completely robust application, since one of the greatest challenge was synchronization and a lot of errors and bugs rose from this important necessary characteristic of our app.

There where a lot of problems involving adding another account with authentication through Google Sign In , one specific bug that we could not solve entirely was when adding a dependant, the first thing that is done is sign out from the caretaker's account and then open the sign in popup dialog, if the operation is not complete there is no account signed in after this and the app crashes or when more than one dependant is added the method `GetLastSignedInAccount` of the Google API does not fetch the correct account (this data is what is used to do the sign in operation).

Technical Design (80%):

The solution that was thought was good, even though not perfectly implemented. The design of the database was well thought and it was consistent with the problem considered. The tools selected also seemed appropriate having into account the need of real-time data accessibility.

Programming (40%):

The quality and readability of the code is not great, but we tried to cover a lot of user-input errors that could happen, although not everything was covered. Since it was a project that involved some complicated aspects, the code is messy because a lot of things were tried, due to no previous knowledge of these matters in the field of mobile applications.

Usability of the Created Solution (90%):

The app is easy to use and intuitive. We tried to keep a uniform aesthetic and smooth experience. Besides this we were careful with the user input, so it only takes it when it makes sense, for instance it is not possible to add a end date for a medicine without setting a begin date before.

Use of Resources (60%):

A lot of software and documentation was consulted in an attempt to no "re-invent the wheel" was made. But even with all that research there were some solutions that took a lot of effort, like coming out with a way of using Google Sign In twice when adding a dependant (this was done by doing a sequence of asynchronous actions - signout, signin into to the new account and the signout and signing back in to the caretaker account -, this had to be done because all of these actions are asynchronous and would cause conflicts).

Team Cooperation (50%):

Some parts were harder than others and in the end we did not achieve all of what we planned, it is possible to see that some things were better implemented and others seemed a bit rushed and not clean, concerning the readability of the code. The core idea and functionalities are still present.

Chances of Publishing the App (10%):

Right now, the app is not very robust, the core features are implemented but not fail-proof, hence not being ready for publishing.

Overall Impression (90%): It was a great experience learning how to create an app from scratch. The selected goal had a useful purpose. We think the selected goal represented a great amount of work, that is why we could not actually finish with a completely fail-proof version.

Five Main Questions of TAM

What drives this sector of IT?

The constant growth smartphones' usage is inevitably the biggest driver, with the increasing number of people using them, greater is the variety of people and the needs that accompany them. So fulfilling the needs of people or just the improvement of the technology that already exist is what drives the IT sector.

What it will be like in five years?

It is impossible to know with certainty, the market will most likely be hugely bigger because for every day that passes, greater is the number of users and more technologies are invented, such as Virtual and Augmented Reality devices that accompany the development of ordinary apps.

What slows it down, speeds it up?

This area is very volatile, the entire world affects it, we believe that what could slow down this sector probably is the fear of being overtaken by technology. But on another note, what speeds it up is the never ending inspiration and curiosity of developers.

What ideas are dead (though they appeared great once)?

The idea of chatting through pictures, initially introduced by SnapChat for instance. After some years, this way of communicating on its own has decreased amongst users, because more complete apps have emerged since then. Nowadays, both Facebook and Instagram have the same capabilities as SnapChat and come along with a great diversity of filters to make this way of communicating more fun. Also WhatsApp has recently added a temporary photo option, that makes the picture available for 24 hours and can only be viewed once.

Where do new ideas come from?

New ideas normally came to fulfill a need, if something is missing that is just an ground-breaking idea waiting to become reality.

Recommendation for Assigning Future Projects

Overall, we enjoyed developing the project, but we were lacking a base. Being this our first contact with mobile devices applications, the introduction of the project the way it was, made it harder, due to all of the research we needed to do.

Recommendation for Future Students

For the course of TAM, simplicity is the key, the choice of a simple yet useful goal is the difficult part.

When choosing a goal, if you make the error of choosing something very complicated, it will not be possible to finish in time and it will seem that more and more doubts and difficulties will appear as the app is made. So our advice is to go for something simple that could be useful in your daily life. Really think before doing, because planning and definition are never a waste of time, which does not mean you cannot change your mind during the development process.