Ministry of Education, Culture and
Research Technical University of
Moldova
Faculty of Computers, Informatics and Microelectronics

Mobile Development

Telemedicine

Author:
Cojocari Daniela
St. gr. FAF -161

Supervisor:
Antohi Ion

1 Purposes of the laboratory work

- Taking in consideration requirements and given design, develop the Telemedicine mobile app;
- Gain knowledge about the work with drawables, layouts and other resources;
- Implement user authentication;
- Use a storage method for the information used in the application;

2 Laboratory Work Requirements

- **Frontend** Using the provided design implement the UI of all the Views bellow:
  - Splash Screen;
  - Welcome Screen;
  - Login/Signup Screen;
  - Home Screen;
  - Approved Request Screen;
  - Doctor List;
  - Doctor Contacts;

  The UI should be adapted for min 3 screen sizes(mdpi, hdpi, xhdpi). Present in emulator or on a real device.
- **Backend** Implement the functional part of the app according to the API structure from below:
  - Authentication - Login;
  - Registration - Register users;
  - Get Profile - Extract profile information;
  - Get Doctors List - View the list of Doctors;
  - Get Doctor info - View information about a Doctor;
  - User Request - Add an appointment for a user;

3 Laboratory work implementation

3.1 Frontend

Frontend of Mobile app was done by steps in order to create the Views with the UI and the transitions of the views:

- First, was selection of the needed layers from provided PSD files in GIMP as layers. Needed ones were exported as PNG files. Those next were imported in Android Studio through Batch Drawable Importer, automatically was provided 5 different densities i.e. hdpi, mdpi, xhdpi, xxhdpi and xxxhdpi.
- To use reasonable and easier some images, because was needed to use some images in one object was created some additional drawable gile, like optional one that encapsulated more images at a time. As an example is the confirm button, which has a layer-list as beow:

```
<layer-list xmlns:android="http://schemas.android.com/apk/res/android"
                        android:opacity="opaque">
    <!-- The background -->
    <item android:drawable="@drawable/btn_2" />

    <!-- The text -->
    <item>
        <bitmap android:src="@drawable/confirm"
                        android:gravity="center" />
    </item>
</layer-list>
```

- To have the appropriate themes which were close to design, was implemented 3 additional themes in styles.xml:
  - TelemedicineTheme - With the background green and with the pattern applied;
  - TelemedicineTheme.Launch - For the Splash Screen;
  - TelemedicineThemeMenu - For the views with white background;
- Next were created and implemented the layout files for each view needed in the requirements;
- Connection or switches between views and the transitions in each activity class, is done through corresponding functions, which has Intents to launch another activity:

```
Intent intent = new Intent(this, ClassName.class
); startActivity(intent);
```

- The functions are called with the help of Click Listeners provided for each button;

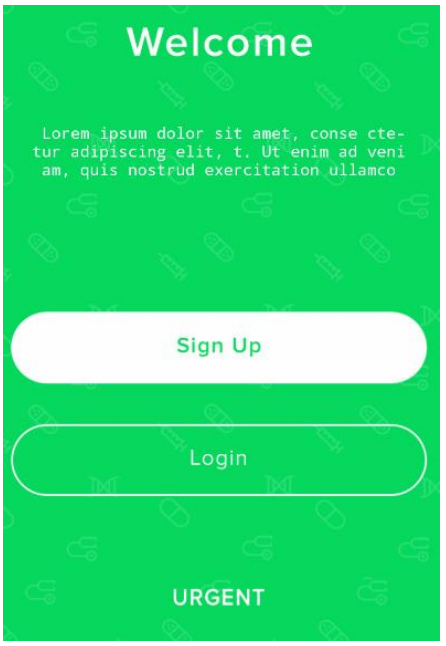
Picture No.1 – “Splash Screen”

In picture no. 1 is the splash scree. It appears once the app is launched. To appear one more time is needed to fully close the app and launch it again. It has only the App Name: “Telemedicine”.

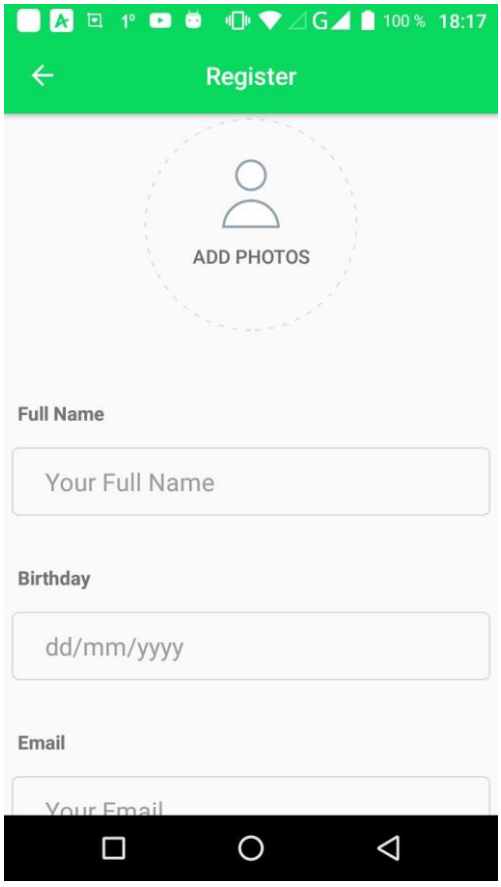
It has a transition to picture no.2, which is the second view.

It has a welcome message, and 3 buttons, sign up which will switch user to sign up page, login – to login and urgent – to urgent medical request.

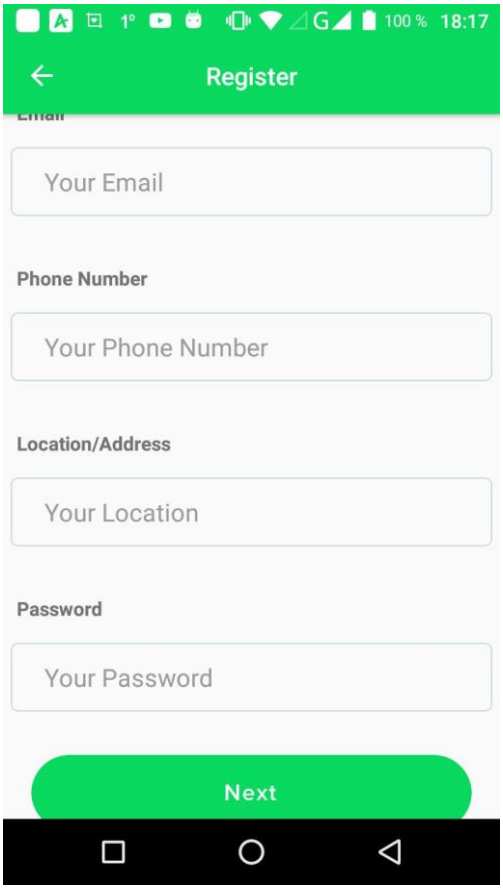
After Sign up button is pressed, user goes to SignUp screen. It has some Fields to complete as in picture no3. And picture no.4, after which user is automatically logged in. In case he already has an account he goes to Login Screen, picture no 5. Logging is done by inserting email and password.



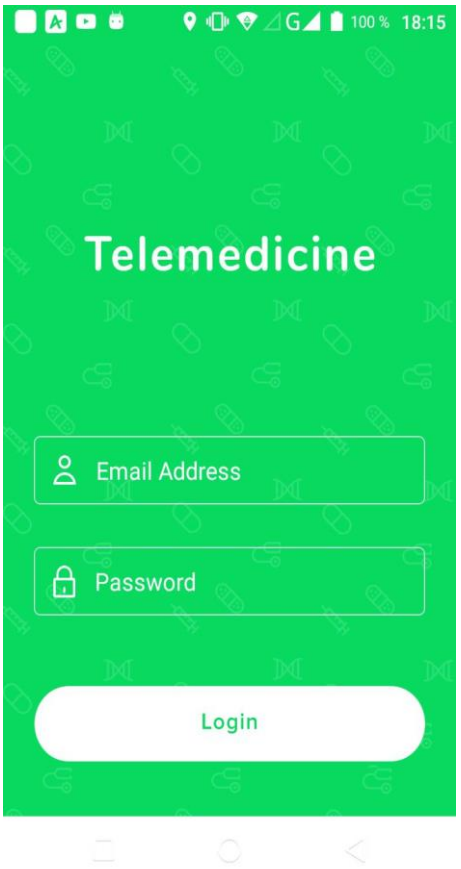
Picture No.2 – “Welcome Screen”



Picture No.3 – “Signup Sreen”

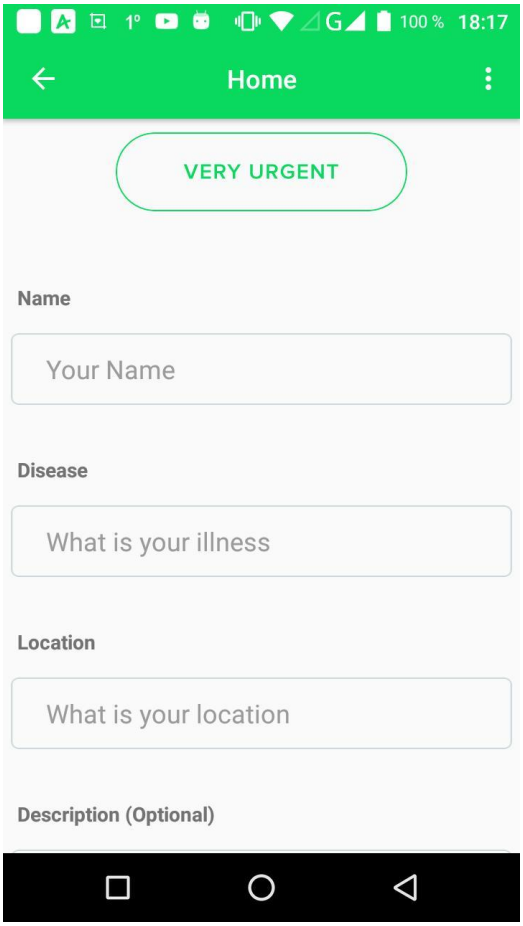


Picture No.4 – “Signup Sreen”

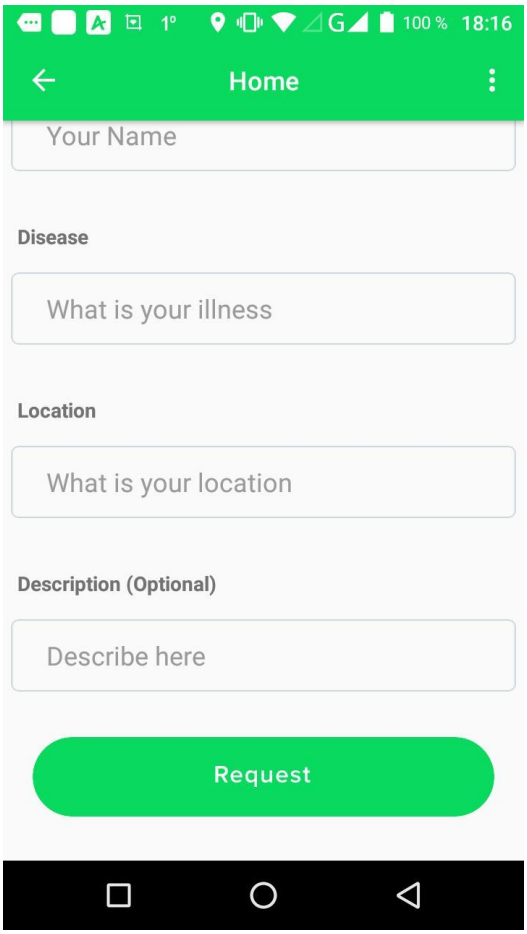


Picture No.5 – “Login Sreen”

After logging user has the possibility to complete the call of urgent case.
As in pictures no. 6 and 7, he has some fields to complete:

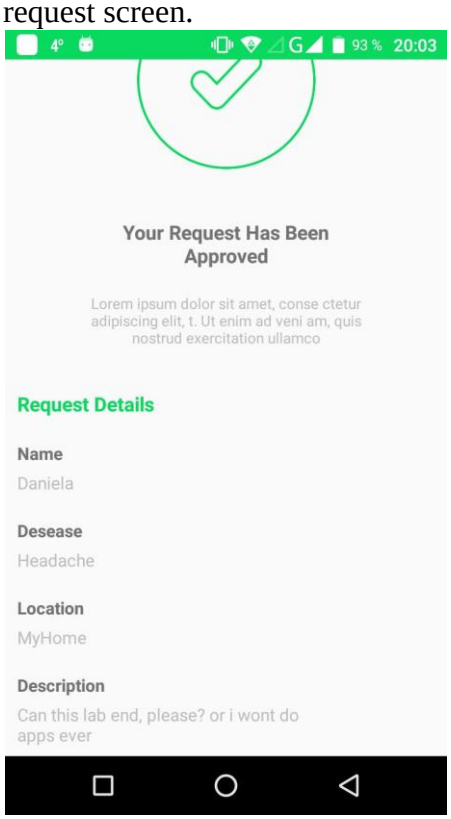


Picture No. 6 – “Home screen”



Picture No. 7 – “Home screen”

After completing he has a approved request screen.



3.2 Backend

To implement the required functionalities, was used an additional tool: Firebase. Firebase is a web platform which provides developers with tools like Authentication, Database, Hosting etc. to ease the development process for different platforms like Android, IOS, Unity etc.

Same, taking by steps first was created the project in the Firebase console. Only after creation it can be accessed. To ease the usage of the Firebase, there is a manager UI from which is possible to manage the tools which is used in the app, or can be used in the app. In our case in authentication section in which can be visualized the users information which were introduced through app. Also, Firebase allows to enter manually the user and its details. Plus the console provide statistics.

Project implemented functionalities:

- Authentication/Registration - Login, Registration and User Storage is done through Authentication tool from Firebase. To do it, from the start is necessarily to specify and determine the sigh method as email/password, phone, google etc.

  In current project is used, in classes is used a FirebaseAuth(mAuth) and a FirebaseAuth.AuthStateListener(mAuthListen).

  Taking by steps, first is assigning the listener to the mAuth:

```
mAuth.addAuthStateListener(mAuthListner
```

Next is needed to initialize the Firebase app and get the instance to it:

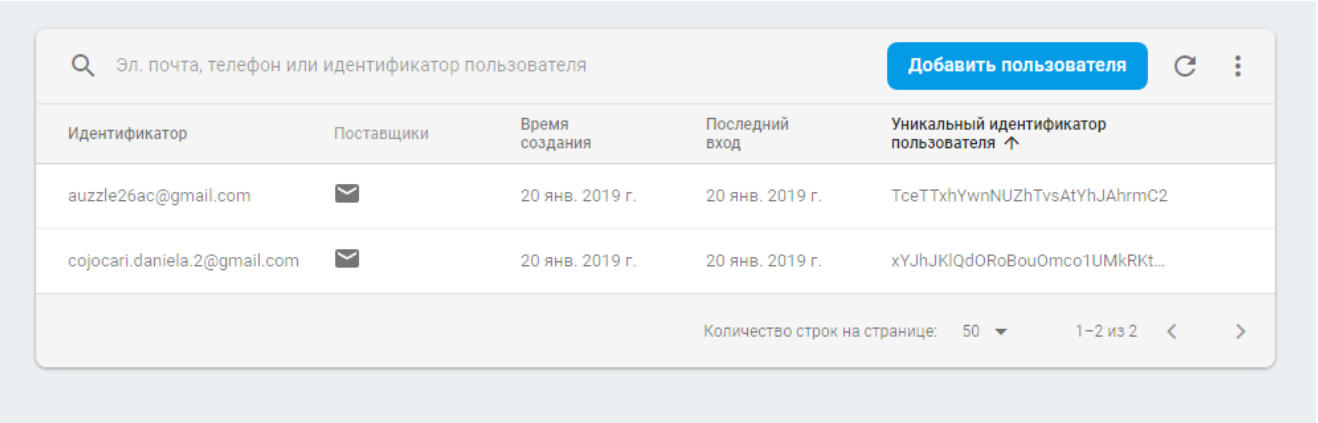
```
FirebaseApp.initializeApp(this);

// initialize FirebaseAuth instanc
e mAuth = FirebaseAuth.getInstance
();
```

After it can be used the API provided by FirebaseAuth, for example:

```
// check the current user
if (mAuth.getCurrentUser() != null) {
        startActivity(new Intent(LoginActivity.this,
                                                    HomeAc
        finish();
}
```

In the end we have some registered users so we can manage them and their actions in Firebase console as in picture:



Conclusion

First of all, the app is not complete because of misunderstanding of doctor list and its criteria of usage. Per stuff done, can be mentioned that there are a big variety of apps that use Android, so Android must be done in different resolutions to be visually acceptable by users. In order to this were introduced different pixel densities. There are measured either by an abbreviation e.g. tvdpi, mdpi, hdpi etc. or can be a number. Some examples can be seen in Android Studio when we select a device for a preview.

Another user attraction factor is UI of the app, which can be made more user-friendly with the help of the design. Besides the options we have in Android Studio or other IDE to style our drawables we can use different software like Photoshop, Illustrator, GIMP etc. to create images, icons and all we need.

One important factor to mention is the functionality which was done easily and workable with Firebase ,a realtime database and backend as a service.