**Mobile Devices Homework**

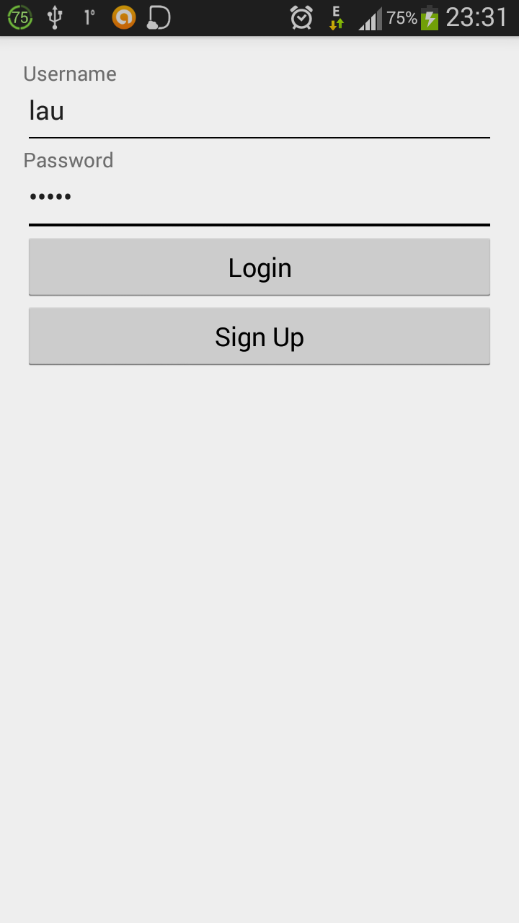
1. Teamwork

This homework was done by Codreanu Florin Laurentiu and Pirjol Adina-Florina, from group 1231E. The implementation of the application was done for Android mobile devices, with minimum OS of 4.1. The project was uploaded on a Git repository for better teamwork and an efficient sharing of ideas.

1. Specifications

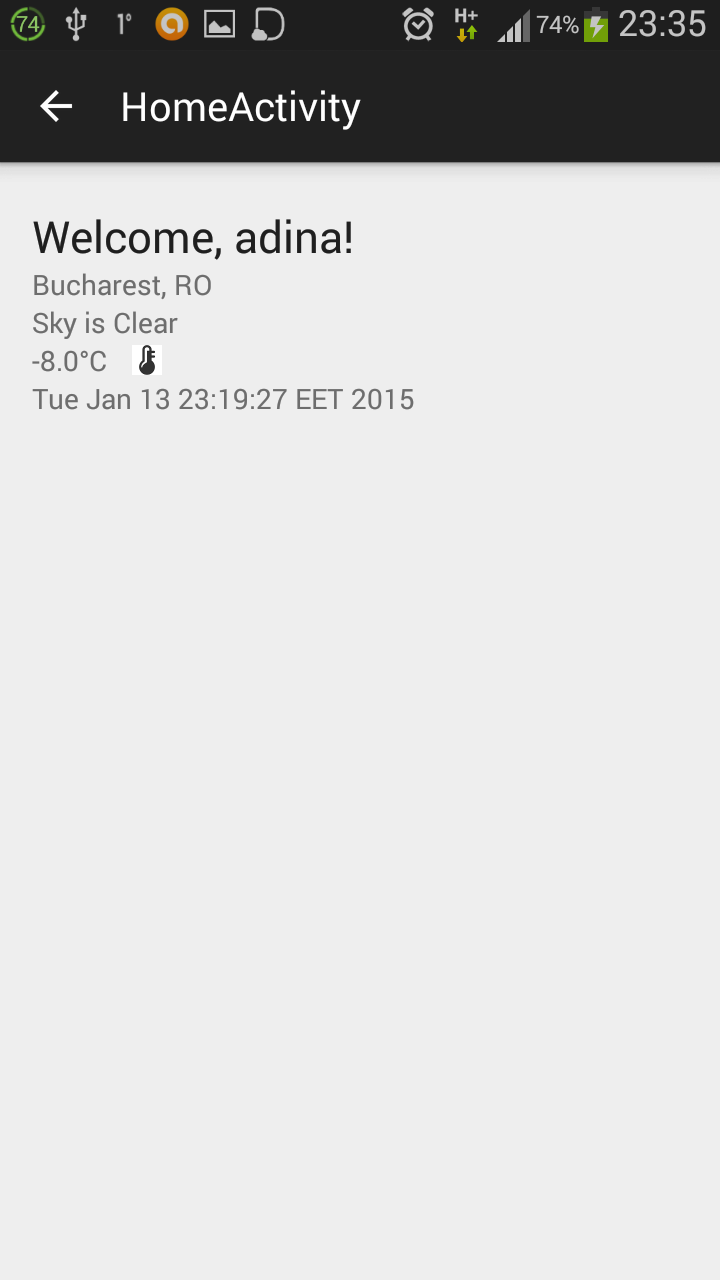
* User can Sign In/Sign Up in the application
* As an authenticated user, user can see his homepage
* User can see a weather widget on his homepage
* On his homepage, user can change his password

1. Functions and implementation

The user login was done using the Parse.com tool. Parse provides prowerful backend tools and services for developers. The Parse library was created and instantiated in the onCreate() method of the LoginActivity. In order to use Parse, we firstly needed to identify ourselves using the clientId and applicationId we received with the creation of our Parse account.

The login activity uses the Parse service. This way, the user credentials data for anyone that wants to login is stored safely online. This, however, requires constant internet connection; otherwise, data might be lost and, of course, the user cannot login. The first time, we used a “UserCredentials” class for storing the credentials on the cloud, which is similar with a table with two columns for username and password. Then, we decided that it is better to use objects of type ParseObject, already defined by the Parse library in order to create the login. We added validations for creating new users/extracting existing users.

Once the user logs in successfully, he/she can see his homepage. In the top right corner, there is an welcome TextView of the format “Welcome, {currentUser}!” for ensuring and remembering to the user that he/she logged in with the right username.



Then, the is some weather information displayed, using the OpenWeatherMap api, available for free use online. The connection to the weather api is done by establishing a http connection via a HttpURLConnection object in the com.app.md\_hw.OpenWeatherMap.OpenWeatherMapClient.getWeatherMessage, from where a JSON response is read regarding weather information in “Bucharest,RO”. The response received is converted into a JSONObject, and the JSON is parsed accordingly and the needed information is extracted in the com.app.md\_hw.OpenWeatherMap.OpenWeatherMapClient.parseWeatherMessage() method, which is then stored in an object of type Weather. The Weather class stores information regarding the location, date, temperature, weather conditions and description from the place/time the homepage was accessed.

Back on the homepage, this information is processed and displayed using the local WeatherAsyncTask class, of type AsyncClass, that allows background operations to be performed and then displayed using the UI. Our class has three generic types, <String, Void, Weather>, where String is the return type of the data extracted from the local api using a location parameter, void is the progress result that does not affect the homepage layout, and Weather is the result returned by the onPostExecute overridden function, that sets the TextViews and the ImageView on the home\_activity layout.