Environment and Development of the Weather Monitoring Application in Kosovo

Milazim Shabani[1], Naim Baftiu[2], Egzon Baftiu[3] and Betim Maloku[4]
milazim.shabani@kolegjibiznesi.com, naim.baftiu@uni-prizren.com,
egzonbaftiu5@gmail.com, betim.maloku@uni-prizren.com
[1]College "Biznesi", Department of Emergencies, Pristina, Kosovo; [2,4]University "Ukshin Hoti" Prizren, Kosovo,
[3]solution25, Pristina. Kosovo

Abstract

The environment in Kosovo is a topic of concern for the citizens and the state because of the temperatures that affect the health of the citizens and the climate around the world. Kosovo's climate is related to its geographical position. Stretching in the middle latitude, Kosovo's climate depends on the amount of heat coming from the Sun, the proximity of the Adriatic Sea, the Vardar valley, the openness to the north. In order to better understand the climatic features of Kosovo, one must know the elements of the climate such as: sunshine, temperature, precipitation, atmospheric pressure, winds. The Meteorological Institute of Kosovo is responsible for measuring temperatures in Kosovo since 2014 and until now 12 meteorological stations have been operationalized with automatic measurement and real-time data transfer to the central system for data collection and archiving. The hydrometeorological institute lacks an application for measuring temperatures in all the countries of Kosovo.

Software applications are generally built to suit the requirements of different governments and clients in order to enable easier management of the jobs they operate on. One of the forms of application development is the development of mobile applications based on android. The purpose of the work is to create a mobile application based on the Android operating system that aims to display information about the weather, this type of application is necessary and important for users who want to see the temperature in different places in Kosovo, but also the world. This type of application offers many options such as maximum temperature, minimum temperature, humidity, and air pressure. The built application will have real and accurate data; this will be done by comparing the results with other similar applications. Such an application is necessary for everyone, especially for those people whose daily work is dependent on the weather or even for those who decide to spend their vacations, such as summer or winter. In this paper, comparisons are also made within android applications for tablets, televisions and smart watches.

Keywords:
Environment, Temperature, Mobile Applications, Retrofit, Json.

1. Introduction

In the historical past IHMK- (Hydrometeorological Institute of Kosovo) had the meteorological network which covered the entire territory of Kosovo and met all the criteria according to the monitoring objective and according to the OBM regulation [1-3]. This network was completely destroyed during the last war and since 2000 its re-functionalization has started. Initially, 4 meteorological stations with classical measurements were operationalized, Prishtina, Pejë, Ferizaj, Mitrovica, and then from 2014 and until now 12 meteorological stations with automatic measurement and real-time data transfer to the central system for collection and archiving have been operationalized. of data. Currently, there is a good coverage with the meteorological monitoring network and meets the criteria according to the monitoring objective and according to the OBM regulation, which has meteorological, climatological, agrometeorological and precipitation stations. IHMK made the measurements of Insolation (h), Relative Humidity-Rh (%), Daily rainfall (mm), Wind speed and direction (m/s), Snow layer (cm), Air pressure (hPa), Temperature¬Tmax (°C), Temperature¬Tmes (°C), Temperature¬Tmin (°C), Cloudiness (0¬-10) [4].

Android mobile apps can be developed in two languages Java and Kotlin. Although Java derives most of its syntax from the C and C++ programming languages, compared to these two languages, Java has a simpler object model and less support for direct work with system resources. This scientific paper develops an application in the Java programming language Android mobile applications compared to IOS mobile applications at the level of user growth for the business world has found application or greater use of applications on the Android platform. For the reason that applications on the android platform are much more user-friendly and usable. The purpose of the application is to display real-time temperature data in all countries of the world considering communication with the server in London where there is an opensource API for application developers without payment. We will see the way of developing the application as the use of technology, tools for creation in the next steps that are presented in this scientific paper [5-6].

The objectives of the scientific paper are:

- To present a mobile application on the android platform.
- To contain accurate weather data.
- To implement the solution of a certain problem.

Manuscript received August 5, 2022
Manuscript revised August 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.8.46

- Developing the application correctly and realistically.
- Give weather data in all countries of the world.
- To elaborate the whole process from data acquisition to project implementation.
- Using the API to build our application.
- To give conclusions about the result and who are the interested parties.

2. Android applications

Android mobile apps are based on Linux operating system and Kernel where they are designed to run on an android device or emulator. The term also refers to an APK file that stands for Android Package. This file is a Zip file that contains the application code, resources, and information and meta information. Android apps can be written in Kotlin, Java and C++ and run inside the virtual machine. Apps are normally distributed through app markets such as Google PlayStore, so it is possible to enable installation from the APK file or via USB connection in the device settings. To install the APK it must have a unique package name to store meta information. After installation this name is registered in the package manager. For apps listed on Google Play, the package name can be seen in the URL [7-9].

2.1 Advantages of mobile applications

Mobile apps are the best solution that requires regular development and use in the industry, a mobile app allows users to set preferences, create accounts and keep vital information at hand. On a business basis, a mobile application is a very good opportunity to target an audience and therefore build marketing campaigns for different users. Another integrated capability of mobile applications is the capability of notifications to users who have installed the application on their system. In turn giving publishers the ability to send messages directly to users. The ease of sending the notification is one of the main reasons why many businesses want to have a mobile app in the first place, because it makes managing work easier for both its staff and business management. Another advantage of mobile use is the possibility of using applications even when offline. The reason is that applications are installed on a mobile device, they will continue to provide content and features without an Internet connection. The application stores data locally and then deploys it after a connection is made. to expand the market. Tomorrow we are not limited to work only in the local market, our opportunities extend to the global market [10].

Businesses are always looking for experienced android developers. Flexibility of work and schedule, we can work with application development only with half-time work, full-time, in the office, freelancer, other opportunities that do not condition us in time and residence. Competitive salary, we can provide a good salary even with only a few hours of work per day. The expansion of the mobile device market, mobile devices are becoming more and more popular and important in our social and business lives. The empowerment of these devices also represents an increase in the demands of software developers in the industry. For the opposite of the good side of mobile applications, we also have the negative side of mobile applications, so below we will clarify and see the disadvantages of mobile applications.

2.2 Disadvantages of mobile applications

Disadvantages of mobile applications are: Insufficient transport layer protection, weak authorization and authentication, client-side injection, wrong handling of audiences, security reviews, data leakage in the malware side channel. We should require TLS/SSL encryption with strong algorithms within communications. The unique bug is the app's unscripted links to a third-party app development company. These vulnerabilities are installed on the server side. The best practices to follow are the same as with online applications. Especially for application development device identifiers should be ignored as devices can be removed and tampered with. In downloading mobile applications, a large number of downloads can be from children looking at the picture or icon of the mobile application looking for games. In fact looking at an average adults download less than one or more apps per month than children. Update and maintenance is not only investments in the first development of a website or native mobile applications. Both should also be updated regularly and as needed whenever there are any product or service changes. Account for Android and iOS updates and this can be for three different developers who have to be loaded with three sets of updates, not only that but businesses that create mobile apps take time to re-approve updated apps and that consumers to benefit from an update, they must either download software updates or re-download an entire application. Next on our list of disadvantages of mobile apps is the way they are shared and distributed. When it comes to native apps, your marketing efforts tap into multiple niches which probably include: your website, your app on google play, as well as the app store, and maybe even more if your app is listed on other app stores outside of the ones mentioned above. This means that you will have to develop a very efficient strategy for each separate channel. In comparison progressive web applications can be shared with a single url or a QR code. One of the biggest disadvantages can be the cost of a native mobile application that costs thousands of dollars [11]. Mobile app stores also receive a percentage commission on sales of up to 30%, then there are marketing costs as well. Whereas native apps are completely dependent on marketing efforts and customer feedback.

2.3 The weather app

Mobile applications are becoming among the most necessary applications in the business world, based on the idea we have, we will design an application for displaying the temperatures of different countries in the world. We also have these applications in different technologies, such as on the web, on the desktop, etc. But our goal is to create it in a mobile application that I think is more usable and has ease of use for the customer. We have thought that the development of this application is important and helps us if we are in an unknown place and we want to see the weather, what temperature we have or we are planning something that we definitely need the time to be as we want and through the application we can to see the exact measurement of temperatures as well as the maximum temperatures that will reach during the day as well as the minimum ones Initially researching on the Internet and looking for an open api (free) [12] that we can use we came across the openweather website where during familiarization with web page and extracting the requirements, we have come to the conclusion that we can develop an application where the data will be shown to us such as: maximum temperature, minimum temperature, humidity, air pressure, and the possibility of viewing the location with the help of GPS. For the development of this application we will use the editor Android Studio, which is the only editor we have for the development of mobile applications on android. About the use of technologies, we will talk in more detail in the result section about what we used to develop the application from start to finish. I think this application can also be useful for some customers whose profession is closely related to time or temperatures. For example, workers who work on the coast, or workers who are in ski resorts during the winter season, and I think it also applies to farmers who work the land. Then we will see the progress of the development of the project in short points, where we will talk about the results in more detail.

3. Problem Analysis

As in any software system, we also have problem statements that we have developed in our application. During the development of this project, passing through many stages, we had various problems. As a start, thinking about the creation of the application, the idea of how we should do it, what technology should be used, what are the tools for using the application, etc. By researching and analyzing we have come to the conclusion that for the development of the application our requirements were to find a free api that provides data free of charge, during the research we have seen several open apis that provide data about the weather in real time through the open weather website. When we entered the web page that offers an open api for the exchange of information between the client and the server, we defined what we can design and how our application will look. When analyzing API information for communication, we are forced to create an api key where it is unique and you are identified with that key, where no other key can be similar, nor can anyone else use that key. The next processing of the problem statement in our application is if we have not imported the technologies to gradle. This means that we must necessarily implement it in the gradle part, such as the implementation of picasso, the implementation of retrofit, the implementation of gson conversion, as well as this application for communication with the server requires internet and the internet permission must be made that allows us to use it every time of the application. Another statement of the problem is if we request a request from the server, but we communicate with it incorrectly. The requirement for communication with the client are two and they are created id and the other is the city where we want to see it. With these two we can communicate with the server to get the data input of that country. While the problem that we can define is that if we do not have one of these two in the link, we cannot communicate with the server because our api cannot communicate with the server if you have not met the condition required for communication, which are two parameters city id and name [13].

3.1 Problem selection methodology

In order to be informed and understand more about the topic and similar applications and their technologies, we have researched scientific articles and various materials using books as well as information from the Internet to learn more about these applications. mobile, in our methodology we have included several methods such as: By browsing the literature we have analyzed various similar applications that are used in the world, the technology that these systems use. Analyzing the data, we have researched various materials about the operation of other applications similar to the application that we will develop, we have also analyzed how these applications bring benefits and help users such as students, people who travel, are on the way or work etc. App comparison, a method used to understand the advantages and disadvantages of this application on different devices. Application development, after analyzing similar applications and the functions of these most used applications, we have come to the conclusion that we can develop an application for mobile devices that suits us and fulfills our requirements [14].

3.2 Project development and analysis

The application that we will build will be in the mobile application using android studio which allows us to develop our application. After we have analyzed and planned whether it can be realized and there are expected results and who are the interested parties, I decided to create the application for displaying the weather. Initially, for the development of this application, I needed an API that contains real statistical data in all countries of the world, and I researched this data from the Internet on the openweather.org website where I created my account and after I have created a personal key and through that key we have received information from different countries of the world that we have called for many countries where our key has contacted the server in London. After we have received the information to us as a client, it has returned us an information in Json file where for us we have classified it and divided it into attributes where in that information we had: latitude longitude coordinates, city name, maximum temperature, minimum temperature, air pressure, an icon in which if the weather was sunny we had the icon of the sun or vice versa with other atmospheric conditions depending on the weather such as snow, rain, wind, etc. After we got the API and thought how our application will look like, we started with the button part, text view, where we used them and connected them to each other with constraint layout after we connected them and we have looking at our view of how it will be, we have started with the logical part of the code where the communication with the server starts from base_url together with our id which we have generated for receiving data, where we are obliged to write the name of the place where we are we want to see the temperature and through onClick of the button where it accepts input, it gets the name of the city, if it is not wrong, the name of the city calls the function where we declared the base_url, also our id, and shows us the data in real time, how they are on the server , if you do not find the country, it will show us an error indicating that the name of the city or country is not correct, it must be fixed [15].

Likewise, when we are shown the data of a country through the text that is weather in London or weather in Kosovo, with a click of that text view that we access, the location is shown to us through GPS that goes from one activity to the activity of Google Maps and can see for example london in map or where we have set as user addMarked which is an object of MarkerOptions.

Android Studio- In this editor called Android Studio we will create our application first before entering the development phase of the project we have several steps in which we must select what we want in the development of the application we must we must have the solution of the version, which version I want to build the project, also the configuration and implementation of what we have to use must be done in the manifest so that we do not have any problems during the development process in our application, then we must to choose whether we want to develop an application on mobile, TV, tablet or where we have decided, we have to choose the activity that we want to develop, we have two activities that we will use and that will be connected to each other, such as Google maps activity with empty activity, this google maps activity aims to show the location of the country, for example if we want to see a country through gps latitude dh longitude that through x,y coordinates we find the required place, we also had the empty activity where we will plan how our application will look, we will give it a new look of what we like, after we have chosen it we have to start the activity with the layout part where we are dealing with constraint layouts.

Android Manifest - Android Manifest is an xml file that has to do with the versions, the encoding, the package of our application, what we have created, it tells us which is the main class where our application is called, how it will appear, has a label, a theme, an icon, etc. It also tells us where we have activities, which means the connection of a class with another class in android we call classes activities. I think that a statement of the problem is that in the android manifest the Internet permission should be applied, in case we do not apply the Internet permission, we cannot communicate with the server. Internet permission must definitely be applied to receive information on the server. At the same time, the permission to access fine location must be applied for the development of our application. In this xml file we also have meta data that serves us to display the location of the country we are looking for and we have our key implemented here that we generated from the developer console through our email

Gradle Implementation- If we don't apply our tools like: Gson, Retrofitin, RetrofitConverter, Picasso, we can't develop our application that we thought. Because for the development of our application we need to use these above, each of them has its own actions that must be implemented to use their resources, otherwise our application does not know how to act otherwise

Application Programming Interface - API can take several forms, the publication of which enables the construction of applications, provisions, websites and other tools, based on the information and services of the agency, so that the audience gets what it needs. from different countries. The APIs are not experimental. More than half of the traffic of large companies, such as Twitter and eBay, goes through APIs. The API helps extend agency information to other users. The API provides real-time access to data, so an application can automatically access the latest information without having to download each update from the agency's website. API and api data makes it easier to use the app. It is an increasingly important tool for publishing materials. However, with a data package available via API, it is important that the entire package can be accessed using bulk download. The tools used in this application are: Web, Java, Json, Retrofit, API key.

3.3 Application development and class sharing

The process of developing the application is quite complicated, but with the help of the json formatter we have come to the end of solving our algorithm on how we will develop the application divided into classes. After we have created the application development logic, we will analyze which tools are needed and should be used in our application, so that during the application's journey we do not have major problems, as it is known that problems are inevitable in every project. Thinking also about testing our application, we thought that as a test piece we take by giving incorrect input, it means the name of the city incorrectly for reasons of testing whether it is at the level or not. If we do not encounter any problems with the emulator during the execution of the code, then our application is run. And when we are ready to write the name of the city, if we write the name of the city correctly as it is, nothing presents us with a problem and the correct information is displayed, but if we write the name of the city incorrectly or write a place that does not exist with intentional errors, then a message is generated that we have the wrong name of the given city, for example, if we take it and write Kosovo with two more letters, this does not show us the data because our server has no idea where this is located place and reports a text that the requested name is not found.

MainActivity- MainActivity is a class that is executed first in our application and has the onCreate method through which the entire application is called. This class has the text view attributes where we have used them to refer to the minimum temperature, maximum temperature, current temperature, image view where for in our image view is the icon of the sun, rain, etc., we also have the buttons, the progress bar that serves as long
d show us a circle, until the moment when it displays the data and performs the communication and more then it disappears. This class also has its own methods that we have referred to the API, getWeatherFun (String city, String appid) which has these two parameters, it also has the method getWeatherDataLocation (String latitude, String longitude) where through this method the other google activity is called maps that receives GPS data.

ApiInterface- This interface contains two methods, these two methods are used for communication with the server and receiving data, but they have their own differences. One method is used to obtain the data of max temperature, minimum temperature, air pressure, etc. where you need a private key and a base_url location where you want to check the temperature. While the other method also serves to obtain data, this method shows the place, the geographical position, meaning latitude and longitude. The method contains latitude, longitude as well as our application id which we have developed. This interface will be used in the Main Activity class where it will be replaced by Json in Retrofit [16].

MapsActivity- This Maps Activity class contains two attributes for latitude and longitude, it is understood that these two attributes are double, as well as this class has generated the onCreate method that the application is called through it, but it also has an intent that means this activity is derived from another intent that has a latitude and a longitude in this case we have used it in the Main Activity class. It also has its own method called onMapReady () where the display of our country data is done by this method because it has the addMarker method of the MarkerOptions object and we get the country. We also use other classes such as: Coordinates, Weather Response, Weather, Generation of keys.

3.4 Application risk management

Each project also has risks from the unexpected that may arise during its implementation, and each project may have different risks. For a successful project it is important to consider all possible risks that we may face along the way, so that when they occur they are not unexpected. In addition, strategies for avoiding them should be written based on possible risks. Therefore, even in our case, the risks that could lead to stagnation or failure of the project were investigated. Taking as a basis the nature of the project, the professional preparation of the developer, as well as from the initial research about it, the following possible risks have been identified: Error in the calculation of the implementation time, Lack of knowledge in certain technologies, Failure to find potential testers.

Application security

Applications based on Android Mobile technology have many advantages, such as ease of access anywhere, creating the possibility of use on different devices, building interfaces that offer an admirable experience to users, etc. But this property comes at the cost of exposure to security risks. Moreover, with the achievement of new technology standards, the range of cyber attacks has also increased. So security is an important factor for any software and should be given special treatment. Therefore, even during the development of the temperature acquisition project, several steps have been taken to guarantee the correct security of the system. Security along with privacy are two important elements in an application in order to maintain the integrity of the application user and not to misuse the code. This application that will provide elements of temperatures through the key, this key is personal and must be maintained that it identifies the developer of the application and is only for one user [17].

3.5 The code used to communicate with the server

In this part of the code that I have written below, we communicate with the server by accepting two parameters, one is our generated id and the other is the city where we want to see the temperature. However, to call these two, we have implemented them with an interface where the class method is overwritten and we develop the code. Next, the information we receive from the server returns the answer in kelvin, not in degrees Celsius, and we have to convert it to degrees Celsius by subtracting minus 273.15.

```
public void getWeatherDataFun(final String city,final String appid) {
    ApInterface mApiInterface = this.getInterface();

    final Call<WeatherResponse> mService = mApiInterface.getWeatherData(city, appid);
    progressBar.setVisibility(View.VISIBLE);
    Log.i("Request ", mService.request() + "");
    mService.enqueue(new Callback<WeatherResponse>() {
        @Override
        public void onResponse(Call<WeatherResponse> call, Response<WeatherResponse>
response) {
            progressBar.setVisibility(View.INVISIBLE);
            WeatherResponse = response.body();

            if (weatherResponse != null) {

                Weather weatherInfo = weatherResponse.getWeather()[0];
                mCity.setText("Weather in " + weatherResponse.getName());
                cityName = weatherResponse.getName();
                String iconID = weatherInfo.getIcon();
                String URL = "https://openweathermap.org/img/w/" + iconID + ".png";
                Picasso.get().load(URL).into(weathericon);

                Main mainInformation = weatherResponse.getMain();
                double kelvinTemp = mainInformation.getTemp();
                int celsiusTemp = (int) (kelvinTemp - 273.15);

                Coordinates = weatherResponse.getCoord();
                latitude = coordinates.getLat();
                longitude = coordinates.getLon();

                mTemperature.setText(celsiusTemp + "°C");
                mdesription.setText(weatherInfo.getMain() + ", " +
weatherInfo.getDescription());

                double kelvinTempMin = mainInformation.getTemp_min();
                int celsiusTempMin = (int) (kelvinTempMin - 273.15);

                double kelvinTempMax = mainInformation.getTemp_max();
                int celsiusTempMax = (int) (kelvinTempMax - 273.15);

                mpressure.setText(String.format("%.2f", mainInformation.getPressure()) +
" hpa");
                mtemp_min.setText(celsiusTempMin + "°C");
                mtemp_max.setText(celsiusTempMax + "°C");

            } else {
                Toast.makeText(MainActivity.this, "Could not locate city ! " +
city_name.getText(), Toast.LENGTH_LONG).show();
```

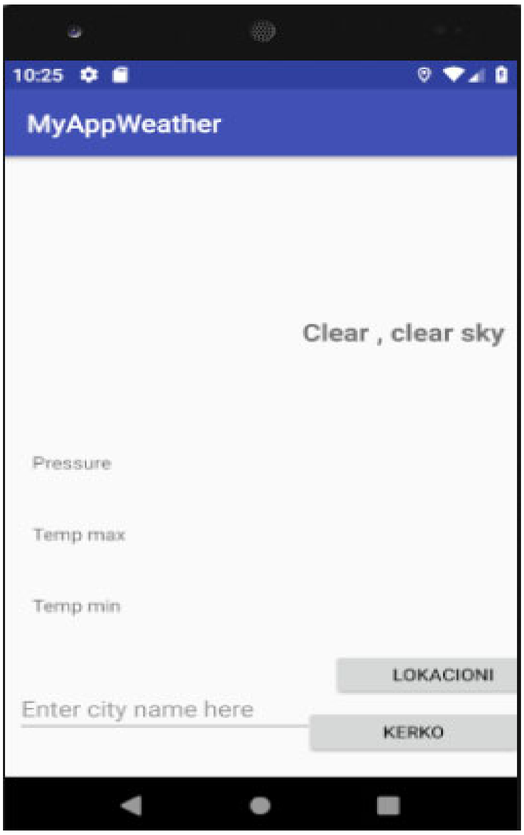


Figure 5. Code-based layout presentation.

3.6 Implementation of Picasso

Picasso is an image library for android apps designed to maintain and take care of image processing uploads. Simplifies the process of displaying images from external countries. Images also add much-needed context and visual flair to Android apps. Picasso is implemented with just one line of code. Where it should be implemented in the application manifest as gradle so that it can be imported into our classes for use. In our application we have used Picasso to display weather icons in case we have rainy or sunny weather in the requested location Picasso displays the appropriate icon depending on the temperature. For the implementation of Picasso in our application, we have defined it below.

```
implementation
'com.squareup.picasso:picasso:2.71828'
```

Figure 6. Representation of the implementation of Picasso in the layout.

3.7 Retrofit usage code

The code below is for getting data through the api. The api gives us a response in gson format and for us that part of the code should be readable, but we definitely need to convert gson to retrofit using the retrofit class where it has its own implemented builder methods and accepts a base url where through we can convert this base url using addConvertFactory and it accepts a gson as a parameter.

```
public ApInterface getInterface(){
    Gson gson = new GsonBuilder().
setLenient().create();

    Retrofit retrofit = new Retrofit.
Builder().baseUrl(BASE_URL).
addConverterFactory(GsonConverterFactory.
create(gson)).build();

    ApInterface = retrofit.create
(ApInterface.class);
    return  apInterface;
}
```

3.8 Code for receiving country input

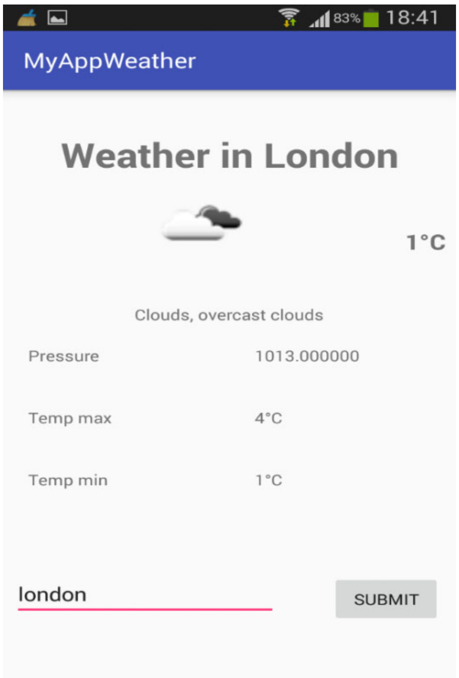
The code developed below serves us to accept a text that text for us will be the required place that we will search to show us the temperature data in real time. If we write the wrong place of the city or state in the given input, our application will announce a message where it will show that it did not find that place because it was not possible to reach the server communication with our incorrectly given request.

```
button_submit.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String cityName =
city_name.getText
().toString();
        if(cityName.isEmpty()){
            Toast.makeText(MainActivity.
this,"Please fill a city name before
submiting",Toast.LENGTH_LONG).show();
        }
        else {

getWeatherDataFun(cityName,APP_ID);
        }
    }
});
```

3.9 Application Functionality

In our created application, the way the application works is very easy, it is easy for the reason that at the moment when we put as input a name of the country where we want to see it and our communication is regular with the server it does not present obstacles and shows data correctly without a doubt that we do not have any problems or errors. At the moment when our application is run and executed, the emulator appears where the main static method onCreate () is called and what is inside the method everything is executed, if we do not have any problems with communication with the server and our server responds to us according to the information it has about the country then our layout will look like the picture below. Showing the maximum temperature, the minimum temperature, the icon of whether it is raining with snow or sunny, as well as showing the air pressure and humidity.



Weather in London

1°C

Clouds, overcast clouds

Pressure 1013.000000

Temp max 4°C

Temp min 1°C

london

Figure 8. Presentation of the requested country

In case in our application, we do not put the name of the required input or place in the right way. Our application will not provide information about that country, only that a text will appear showing that the requested name is not found in the server room, asking for country information. Therefore, we must be careful not to make spelling mistakes in the requested place.

4. Discussions and Conclusions

With this application that I have developed, the conclusion of this is that we have realized a reliable and also requested application for customers or users of any application for measuring the temperature in the current country or even in different countries. Correctness and Performance of this application are two basic elements where their priority is at a high level. These apps can be pre-installed on phones during production platforms, or delivered as web apps using server-side or client-side processing to provide an app-like experience within a web browser. Application software developers also have to consider a long range of screen sizes, device specifications and configurations due to strong competition in mobile software and variations within each of the platforms. The development of mobile applications has been steadily increasing in revenue and sites created. A 2013 analyst report estimates that there are 529,000 direct jobs in the app economy within the EU, followed by 28 members including the United Kingdom, 60 percent of which are mobile app developers.

As part of the development process, mobile UI design is also essential in creating mobile apps. The mobile user interface considers constraints, contexts, input screen, and mobility as blueprints for design. The user is often the focus of interaction with their device, and the interface includes both hardware and software components. User input allows users to manipulate a system, and device output allows the system to show the effects of user manipulation. I think my application is an application that has a correctness and good performance, also useful for people who are interested in any application to look at the weather, for the reason that we always want to know the temperature of our country or any other place where we plan to watch it. Also, for the execution of this application, internet permission is needed because we do not have the possibility to communicate through the api with the server. Therefore, first we need to connect to the Internet and then use our application for our needs.

References

[1] A. Developer https://developer.android.com

[2] S. Amol dhe K. J. Delaney, "Introduction of Mobile Application ", Chicago, 2007.

[3] A. Application, «Android Developer, » Available: https://developer.android.com/reference/android/app/Application.

[4] A. Studio, «Android Studio, » Available: https://developer.android.com/studio .

[5] B. Ryan, "Google is working on a mobile OS, and it's due out shortly", 2007.

[6] M. Dan, Android Developer Blog, 2008.

[7] Q. Pearrce James, "Google Strong Mobile Related Patent Portfolio", 2007.
[8] L. Sean, "Mobile App usage grows 35%, TV and web not so much, 2012.
[9] C. Eric, "Android Market Update ", 2009.
[10] O. Weather, «Open Weather, Available: https://openweather.org/
[11] J. Formatter, «Json Formatter, Available: https://jsonformatter.org/ .
[12] L. Layout, «Android Devs, Available: https://developer.android.com/guide/topics/ui/layout/linear[13] R. L. f. Calls, «Retrofit Library, Available: https://square.github.io/retrofit/ .
[14] G. Convertor, «Gson Convertor, Available: https://github.com/google/gson .
[15] L. L. a. Mascots, Linux Online, 2010.
[16] A. D. GitBook, «Introduction of Android Developer, » 2020.
[17] Të dhëna nga Instituti Hidrometolorogjik i Kosovës-IHK