BABEŞ-BOLYAI UNIVERSITY CLUJ-NAPOCA

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

COMPUTER SCIENCE

DIPLOMA THESIS

CENTRALIZATION OF EDUCATIONAL DATA

Supervisor
Associate Professor Dr.
Sanda Avram

Author
Paul-Valentin Haraga

2020

BABES-BOLYAI UNIVERSITÄT CLUJ-NAPOCA
FAKULTÄT FÜR MATHEMATIK UND INFORMATIK
INFORMATIK IN DEUTSCHER SPRACHE

BACHELORARBEIT

ZENTRALISIERUNG VON BILDUNGSDATEN

Betreuer
Außerordentlicher Professor
Dr. Sanda Avram

Eingerichtet von
Paul-Valentin Haraga

2020

UNIVERSITATEA BABES-BOLYAI CLUJ-NAPOCA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
SPECIALIZAREA INFORMATICĂ ÎN LIMBA GERMANĂ

LUCRARE DE LICENTA

CENTRALIZAREA INFORMATIEI EDUCATIONALE

Coordonator
Conferentiar Universitar Dr. Sanda Avram

Absolvent
Paul-Valentin Haraga

2020

Abstract

Throughout the development of a project, the developers will learn and evolve. This project was no different, I learned to work as a team, to work with different tools, to resolve problems that arise and to bring new ideas to the project.

The thesis is structured in three parts: the introduction, which contains the chapters: Abstract, Introduction, Motivation and The Problem. In this part I welcome the reader to a short description of what I try to achieve through the development of the application, presenting the found problem and my motivation to develop it.

The second part of the thesis consists of the chapters: Related Works, Solution, Technologies and UBBApp. This chapter is a technical one: that presents other platforms that inspired me throughout the development, describing the technologies that I used to develop the application and the application itself, how it works. I think this chapter being the most important of the thesis, because it can be used as a guide for the application that I have developed.

The final part of the thesis is formed from the chapters: Difficulties during the application development, Future development, Conclusion and Bibliography. Here I try to explain what was difficult for me to do during the development and where the application could be improved. And as a closure for the thesis I give my final thoughts on the impact that this application could have.

Content

1. Introduction

With the evolution of the computers there is a trend to move the physical information, like documents, forms etc. to the digital world, therefore the digitalization of information is still in a development phase. A branch of society that needs to be digitalized is the education data, so that the data can be accessible for the persons that work and learn in this system.

But just the digitalization can sometimes not be enough, especially for complex systems like education, so this system must be centralized, so that the information can be easily added, displayed and maintained.

Based on this idea the UBBApp was developed, to digitalize and centralize the information in the University Babes-Bolyai, so that the students can find all the needed information about their faculty in one place, for example the schedule, the grades and many others. If needed the students can find the information of the professors to contact them too. The professors can through this application visualize and edit the discipline data assigned to them.

The application is composed out of two main platforms. The first component is the android platform which is dedicated to the students, so that they can see the information of their professors and their disciplines, as well as to see the evolution of them during the semester, like grades and attendances. The second component of this application is the web platform dedicated to the professors, through which they can edit the student information.

For the development of the application a series of technologies was used to make it more efficient and easier to develop. Such technologies are: Firebase, Android Studio, XAMPP and JavaScript. Firebase has been used for the database, Android Studio has been used as development tool and to test of the android application, and both XAMPP with JavaScript have been used to create the web application.

2. Motivation

The motivation behind this application is to centralize all the information about students in one application, so that it can be easily found and used. My observation during my period of study at the Faculty of Mathematics and Informatics is that they digitalized the information, but through different methods, such as professors webpages and official faculty webpage that offers the schedule, professors data and news, but never centralized the whole information in one application so that it can be easily found and read. An example that I have experienced is that some information, like courses delayed or canceled was announced during classes, and this can lead to information being missed.

Because of these information gaps, I concluded that an application such as one developed for this thesis, can help the student through different solutions. These solutions cover the basic needs of the student, such as schedule visualization, homework visualization and grades and attendance visualization, just to name a couple of them. Through these the student can keep up to date with the development during the semester and how the classes are being held.

To be sure that we cover as many of the needs of the student as possible, the application has references to the official sites of the faculty, so that if needed, the user can access and look for more information or explanations.

3. The problem

The University Babes-Bolyai is a very famous educational institute in Romania, containing a large number of faculties. For each one of this faculties there needs to be a constant channel through which the students, together with the professors and the administration, can keep in touch and informed. At first the already existing webpages seemed enough, but over the years, I learned that there were some minuses to these, such as the need to check a lot of websites to get the homework or to see the attendance or the grades. This may not seem to be a problem, but to create a stable platform that can cover these shortcomings, so that the student can focus on his or her development.

Another problem from my perspective is that the information that is online offered by the faculties is limited just to the faculties. There should be promoted other information too, like public transport schedule, sport activities around the city, information where can be found medical assistance, like dentists. These I think are not essential, but they can improve the overall experience of the students, bringing them closer to the faculty and the city where they study.

4. Related Work

4.1. AcademicInfo

An online platform designated for grades, student contracts, taxes and administrative tools for the students to be in contact with the faculty is AcademicInfo, see Figure 4.1.1.

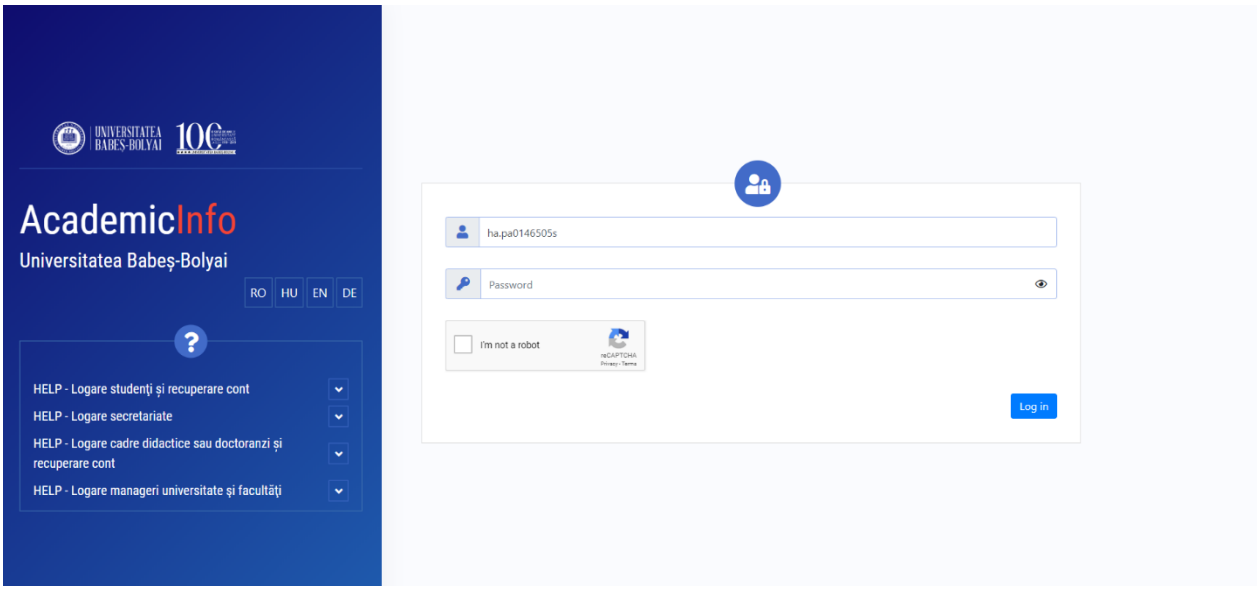

Figure 4.1.1: Preview of the AcademicInfo Application [1]

The advantages of this platform are that it is user friendly and has a lot of features, like evaluating professors, creating studying contracts, checking the final mark of a class and paying the required taxes. Another great feature of it is the possibility of using in a variety of languages, such as Romanian, English, German or Hungarian. This is a great administrative tool to have, but here it reaches its limit, it has no other features that can help the user, so the user must use other platforms to get more information.

The disadvantage of the platform is that there is no support for the user, even though the user interface is friendly and can be easily navigated, the user still needs another way of getting in contact with the personnel to get more answers.

4.2. Moodle

One of the most used open-source platform of the type Learning Management Systems, which is free and improved constantly, is Moodle, see Figure 4.2.1.

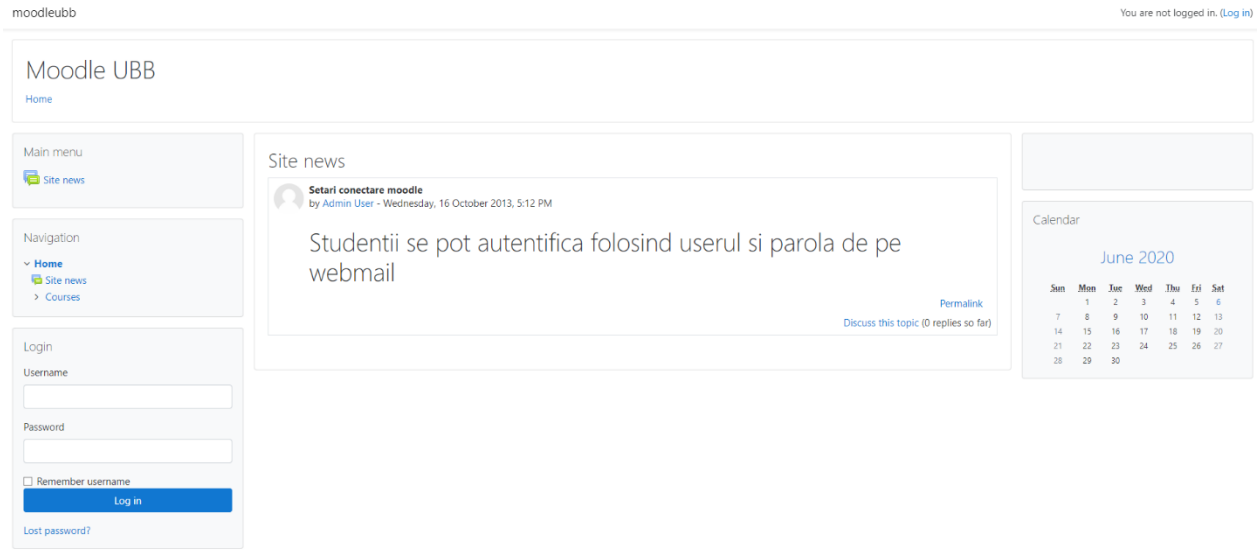

Figure 4.2.1: Preview of the Moodle Application [2]

This platform has a lot of advantages, that facilitate the online teaching and learning. It provides a lot of tools to help the teachers and participants to interact and to improve the overall user experience. Such tools are the chat, the forum, online examination, the possibility to manage training resources, like audio or video files or PDF, presentations. Another great feature of Moodle is the level of customization that it brings. It can be modeled by the user so that it can fit the desired needs and format.

4.3. MySouthampton App

An application that is designed to help students and staff is the MySouthampton App, which can be used to guide the student through the studying years, which can be seen below in Figure 4.3.1. The guide offers the possibility to view a map of the campus, a timetable for the buses, a section of news, in which the students can keep up to date and more feature to improve the overall experience of the students with the faculty.

Advantages of this kind of applications are that there exists a platform that has gathered and centralized a lot of information, that the students can use. For example the MySouthampton

application offers the timetable of the buses, instead of searching it online and wasting time, or the map for the campus which will help the student find his or her way around, instead of using another application for maps and search for the needed place. Such features are exactly what make the integration of the students easier with the faculty.

Figure 4.3.1: Preview of the MySouthampton Application [3]

5. The solution

The solution that this application offers is a centralized platform through which the student can communicate with professors and vice-versa. This solution can be used for any faculty in the University Babes-Bolyai, the application not being personalized or constrained to just one faculty of the university.

By connecting to the internet and using this application students will get access to a variety of services. One such feature is that they will be able to visualize the faculty schedule, which can be composed out of even and uneven teaching weeks. In addition, they would be able to send an email to a chosen professor, get their materials, for example: courses and homeworks, and for homeworks the student will be shown for what week the homework is due, the discipline the homework is for and the message, if there is one, that the professor left for the students. One of the services that the user will be able to use is to see their attendances and grades to a chosen discipline. Beside all of these receive new from the professors and check the official faculty, professor websites to be precise and university websites

6. Technologies

6.1. Android Studio

One of the technologies used for the development of the application is: Android Studio, which consists of the Android Studio application [4], which is the official integrated development environment (IDE) for the development of the Android applications, supported by Google; through the Android Studio application the user has the power to develop applications for any device, that could be smartphone, smartwatches, television and many others devices [5].

Another feature of the application is that it is based on the IntelliJ IDEA [6], which is developed by JetBrains. This environment has a lot of features to offer for example: smart completion inspections and quick-fixes, inline debugger, version control and terminal, just to name a few [7], which enables the developer to create more and easier.

What makes the Android Studio a useful tool to develop Android applications is the built-in virtual machines, which give the developer the power to test how the application reacts and runs on devices. The application offers a variety of choices when it comes to devices and the system that they run on. Using this the user can add, remove and duplicate these machines, using the Android Virtual Device Manager, as can be seen in Figure 6.1.1.

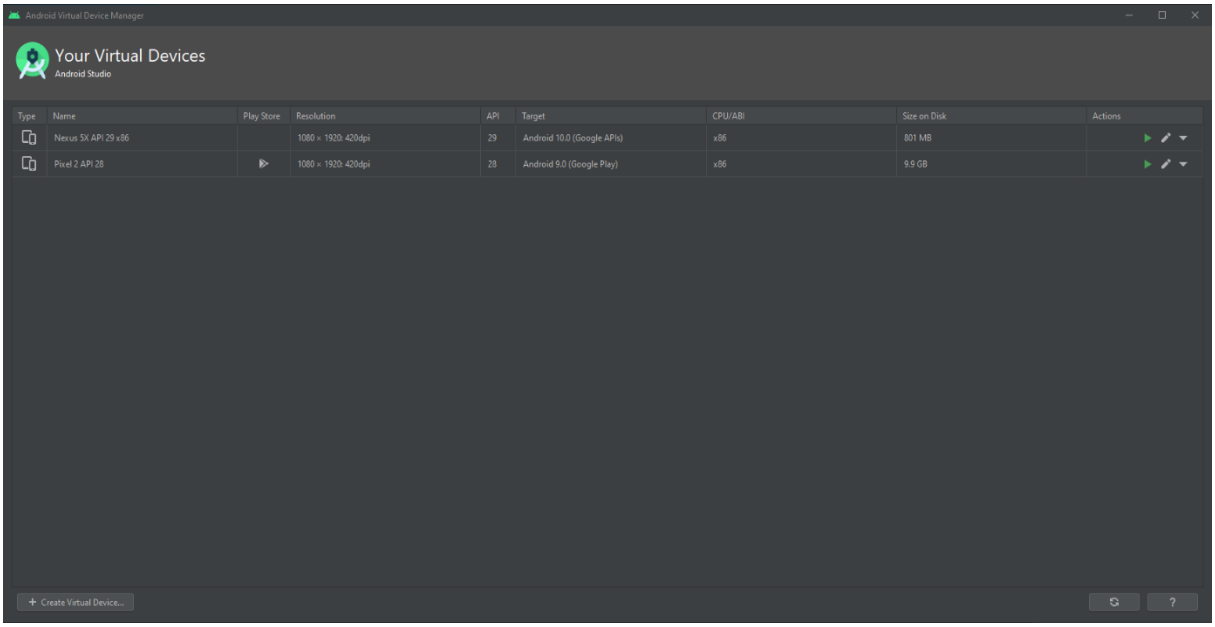

Figure 6.1.1: Preview of the Android Studio virtual devices

A great feature for Android Studio is that it can be integrated to function with git projects. The user can use the buttons of the console of the application to make commits, updates, revert and see the history of the project. This helps develop an application more easily and efficiently because it gives the user the possibility to work with a team in the development of the applications.

Another reason why Android Studio is so helpful and good to use is the built-in tools that helps the user develop an application. For example Android Studio has Firebase Assistant through which the user can easily integrate any feature of the Firebase application with his own application, as can be seen below in Figure 6.1.2. The only requirement to use Firebase Assistant is to connect with the Google account and it is ready to be used. To integrate for example the Realtime Database the user will need just to click a couple of buttons and the needed dependencies will be added. Beside this the Assistant offers guides on how the tool works, as can be seen in Figure 6.1.2.

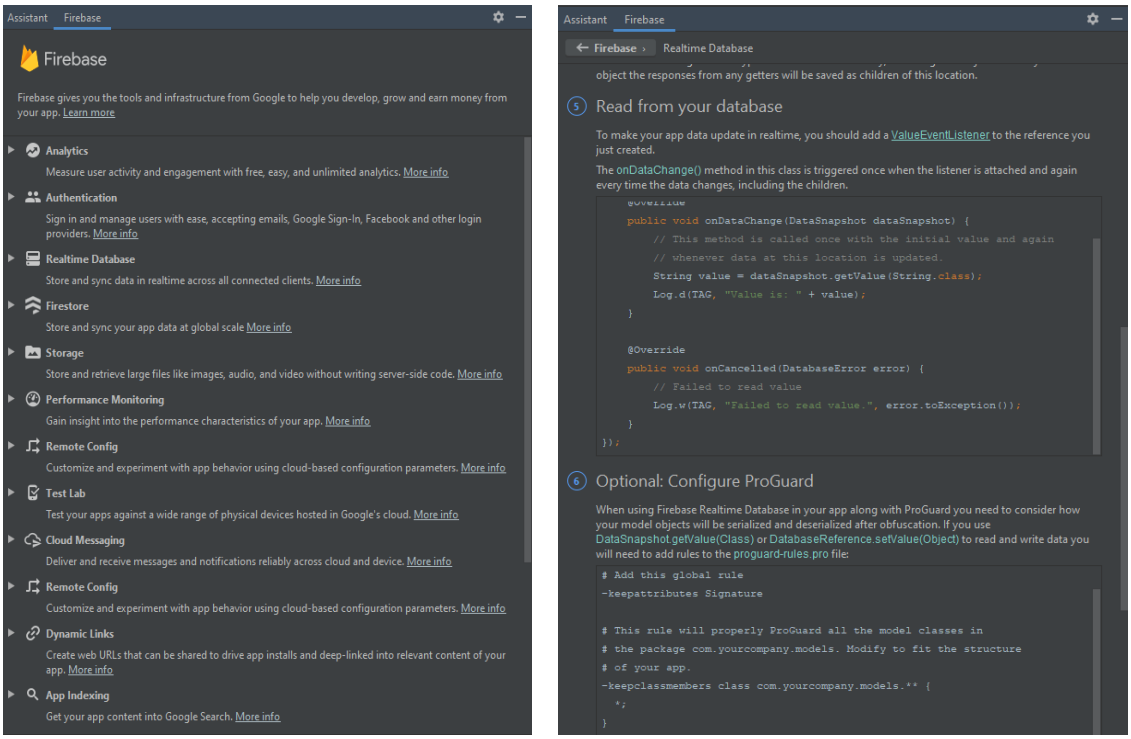

Figure 6.1.2: Preview of the Android Studio Firebase Assistant (left side being the general window, right side being the Realtime Database window)

6.2. Firebase

Firebase is an web application developed by Google. This application has a variation of support for the development platforms, these include: Apple iOS, Android and we. The purpose of Firebase is to give the users the possibility to develop applications more easily and to give them new tools to do so. Such tools are: Authentication is a tool that enables the applications that use it to create a simple and secure log in. It has several authentication options, rather than just the simple username/email and password, such as: Google, Facebook, phone etc. [8]; Realtime Database, which is an online database, that stores the data in JSON format, and it is synchronized in realtime to every connected clients [9]; Storage is a tool to store objects on the cloud like images, videos, documents and many more [10];

6.2.1. Authentication

Firebase Authentication is a tool of the Firebase web application that helps a user to develop a log in and sign in. The compatibility of the authentication is the same as the Firebase application described earlier. The tool is easy to integrate with the working environment and easy to work with it.

The Authentication is very malleable and can be integrated so that the user can use different accounts to log in. Some examples are: Google accounts, Facebook accounts, phone number and email, just to name a few. The implementation of the sign up is as easy as the log in, here the developer having the possibility again to let the user use different accounts to sign up [11]. Beside these the Authentication offers the possibility to reset the password, recover email or receive SMS verification, using a template that can easily be designed in the Firebase console, see Figure 6.2.1.1.

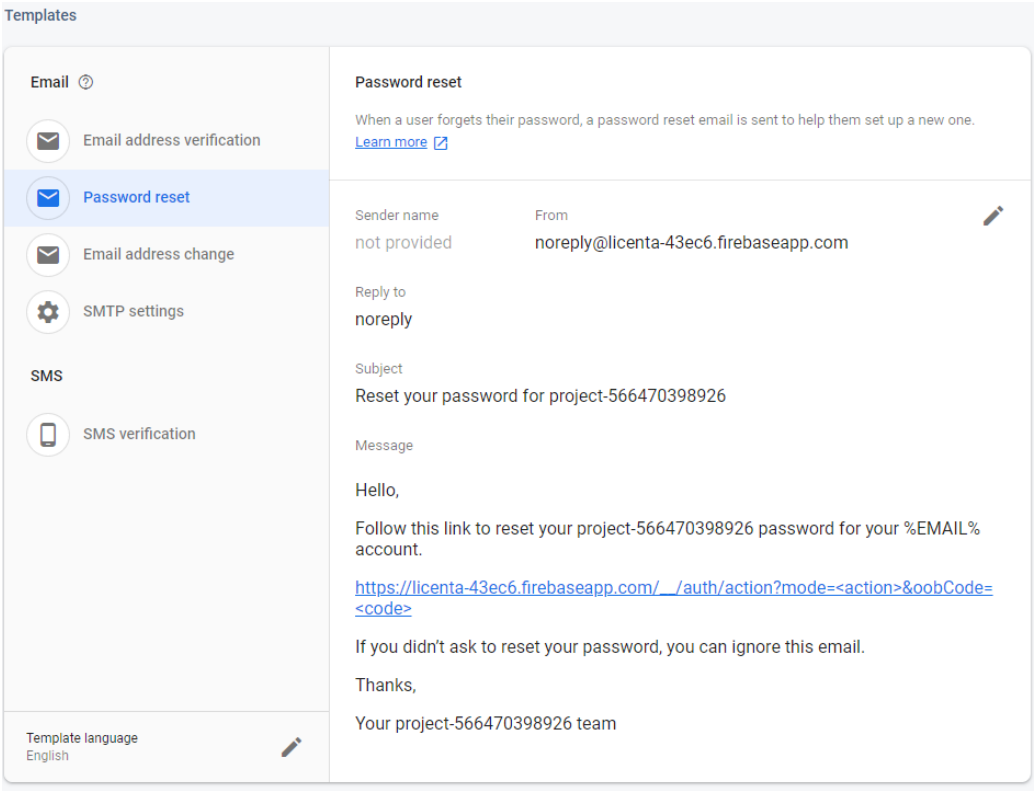

Figure 6.2.1.1: Preview of the Firebase Authentication templates

Another useful tool that Authentication offers is the ability of the developer to view the users that have registered and the usage of the phone verification, which can give insight into how the users behave with the application.

6.2.2. Realtime Database

Firebase Realtime Database is another tool provided by Firebase web application that gives the developer the possibility to work with online databases. The database is free and offers a lot of tools to help develop an application. Even though most of databases are based on a relational logic, Firebase Realtime Database data is stored as JSON format and it's a NoSQL database. The data being structured as a JSON tree, as seen in the Figure 6.2.2.1 below, the root of the tree being the "licenta-43ec6". Here the tables of a relational database are just nodes in the

tree, that the user can access using the name of the node, or they can be iterated over until the desired node is found. The same principle applies for the nodes of a node, for example for the "discipline" node, seen in Figure 6.2.2.1 below, we can iterate the children and stop only when a condition is true, like the "day" property from the child is equal to a value.

Another useful information about the Firebase Realtime Database is that the data can be nested, it allows up to 32 levels deep, but still it is not recommended to nest that much [12]. An example of such nested structure is that the "professor" object is stored as a property for the "discipline" node, for the example seen in Figure 6.2.2.1, and this brings the advantage that when the user gets the "discipline" node, all the information is available, but the data from the "professor" node can repeat across multiple "discipline" instances.

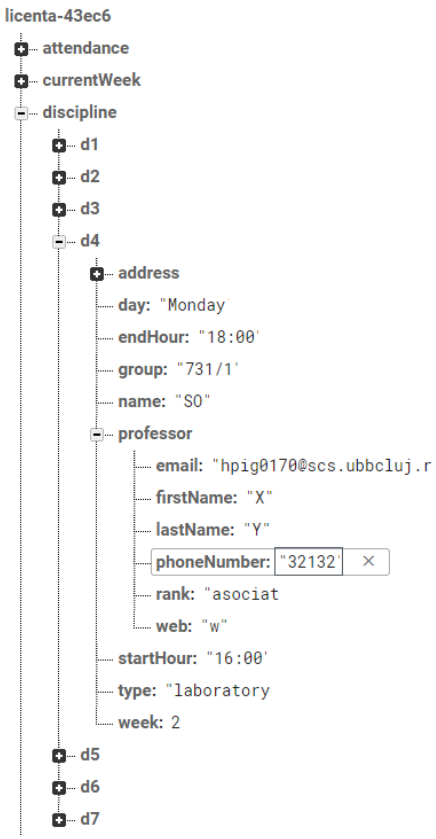

Figure 6.2.2.1: Preview of the Firebase Realtime Database structure

As it was the case for the Authentication, the Realtime Database from Firebase offers tools for the visualization of the usage of data, such as number of connections, the size of the downloads and so on [13].

6.2.3. Storage

The Firebase Storage is a tool provided by Firebase to give the user the possibility to store files, being secure because of the Google security, regardless of network quality. Through this tool the developer will have the ability to upload and/or to download content.

The storage is structured as the file system on your local hard disk. That means reaching or saving a file will require a path to that location. The Storage console can be used to edit the name of the files, create folders or create access tokens that can be used for security, as can be seen in the Figure 6.2.3.1 below.

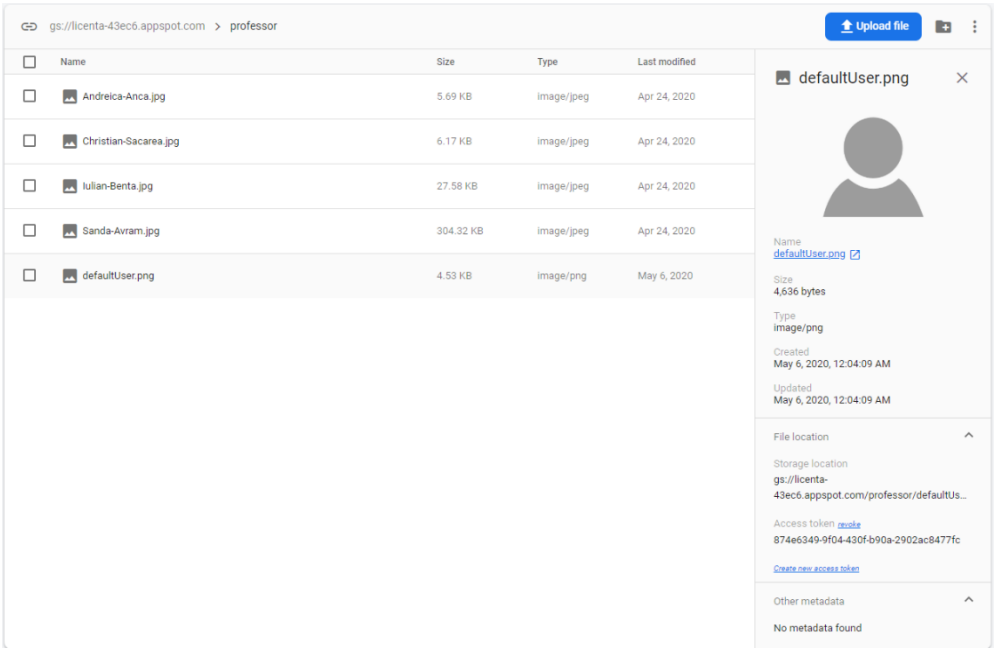

Figure 6.2.3.1: Preview of the Firebase Storage console

As for the metrics, the Storage has this tool, to help and create an image about the usage of data, such as number of request, number of objects stored and how much data is stored.

6.2.4. Rules

For each tool that Firebase provides, there is the possibility of creating rules on how the data can be accesed. The Firebase Rules are using a JavaScript like syntax and can be of four types: ".read", ".write", ".validate" and ".indexOn", these being used to determine who has read and write access to the database and storage. The ".validate" is a great tool because the developer can enforce validation of the input before the data will be added, this being a safety feature that can ensure the running of the application in a safe way.

To make it an even better tool, the developers of the Firebase application, made sure to create a console through which the user can test the rule, this console being named "Rules Playground". The Playground has a lot of ways in which the user can play with the rules, for example as can be seen in the Figure 6.2.4.1. There are simulation type, which can be set to "read", "set" and "update", to test how data is read or written in the database. The Playground has also implemented the Authenticated mode in which the user can test the rules as an authenticated user that tries to read or write the data.

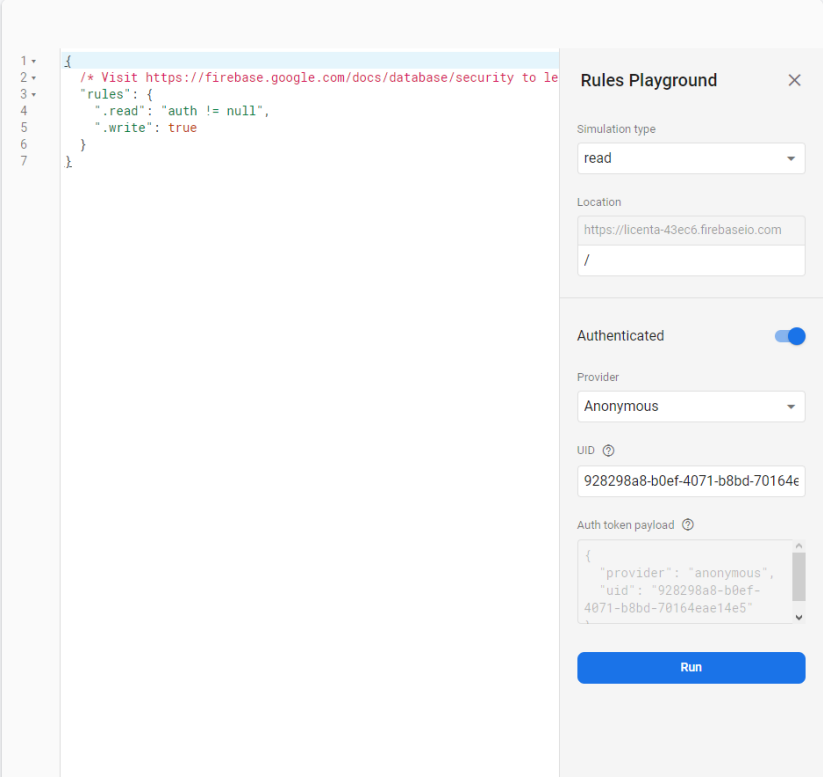

Figure 6.2.4.1: preview of the Firebase Rules on the Realtime Database [14]

6.3. JavaScript

JavaScript (JS) is a programming language that is used together with Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS) to develop web applications. But the difference between JavaScript and HTML, CSS is that JS enables the web pages to be interactive and it can be used also as a channel between the client-side application and the server. JavaScript has the power to create HTML elements and add to them CSS styles, which makes it great for creating responsive web pages, this giving it the possibility to create whole pages using JS only.

The way JavaScript works is the browsers now have a dedicated JavaScript engine to execute the code. To create such a file, that can be executed, the user needs a text editor which can be used to write the code and save it with the extension ".js" and the browser in which to run it. This makes JavaScript a programming language easy to use with no further tools needed to develop an application. But using it with a normal text editor can be difficult, so there have been created such editors to help develop JS programs even more easily.

6.3.1 WebStorm

One such editor is the WebStorm, which is JavaScript integrated development environment (IDE). It is developed by JetBrains, which uses the same design for the editor and makes it feel normal and easy to use, especially if the user used other products from JetBrains, such as Android Studio, PhpStorm, IntelliJ and many others, as can be seen below in Figure 6.3.1.

The advantage of WebStorm is that it is a modern editor which covers the basic needs from an editor, such as intelligent code completion, error detection and refactoring, just to name a few. Beside these it offers a debugger, so that the user can debug the application, through breakpoints as well as offering the possibility to see the variables at those points [15].

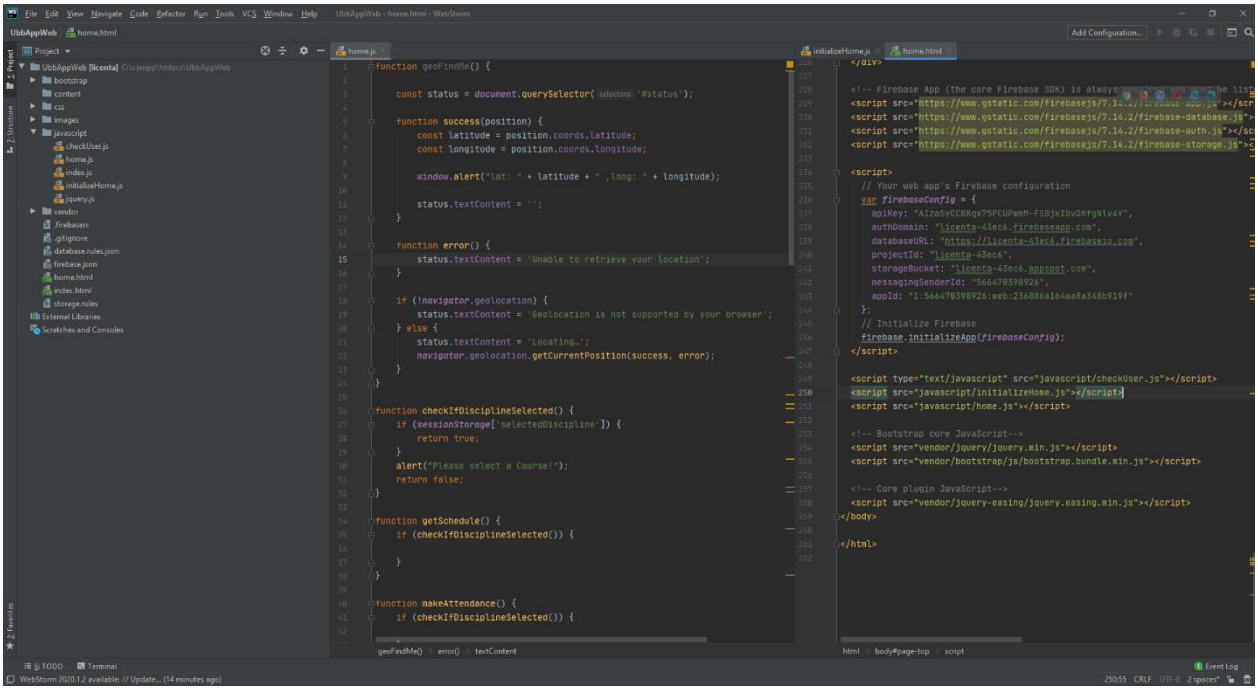

Figure 6.3.1: Preview of the WebStorm editor

6.4. XAMPP

XAMMP is a program that allows the developers of webpages to create a local server on their machines so that they can develop these pages without the need of a server and test them before deploying them on a live server. The transition of a webpage developed using XAMPP can be easily transitioned to a live server, because of the stark similarity between the XAMPP local server and the live servers.

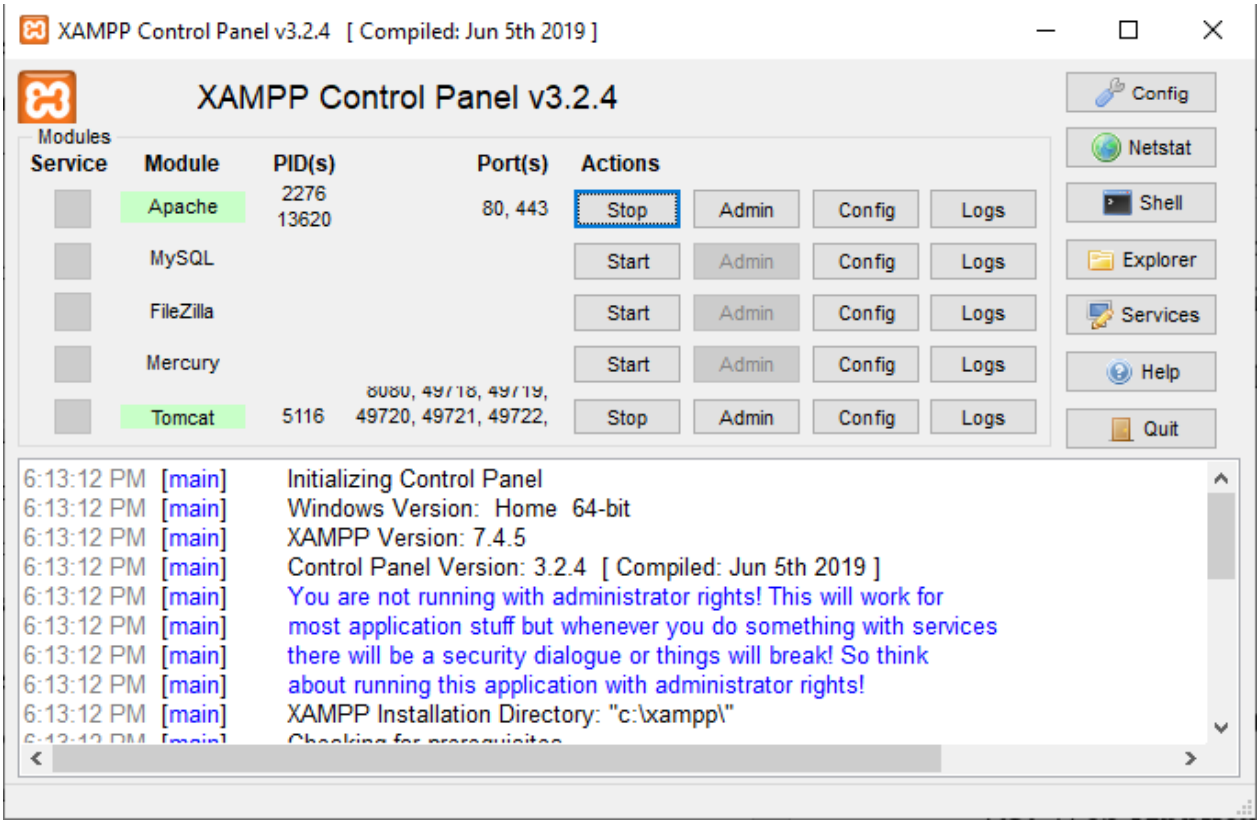

Figure 6.4.1: Preview of the XAMPP Control Panel

The program has a control panel that can be used to start or stop the services that are needed for the development of the application, as can be seen above in the Figure 6.4.1. Beside stopping and starting these services through the control panel the user can configure the settings of these services, such as changing ports and see the logs from these services, if problems arise.

The use of this program is the possibility of simulating a server and run server-side programs such as webpages that have a PHP file to communicate with a database. This creates a great environment to create webpages that use PHP.

6.5. SourceTree

SourceTree is a program designed to create a simple interface through which the user can easily visualize and manage their Git repositories, as can be seen below in Figure 6.5.1.

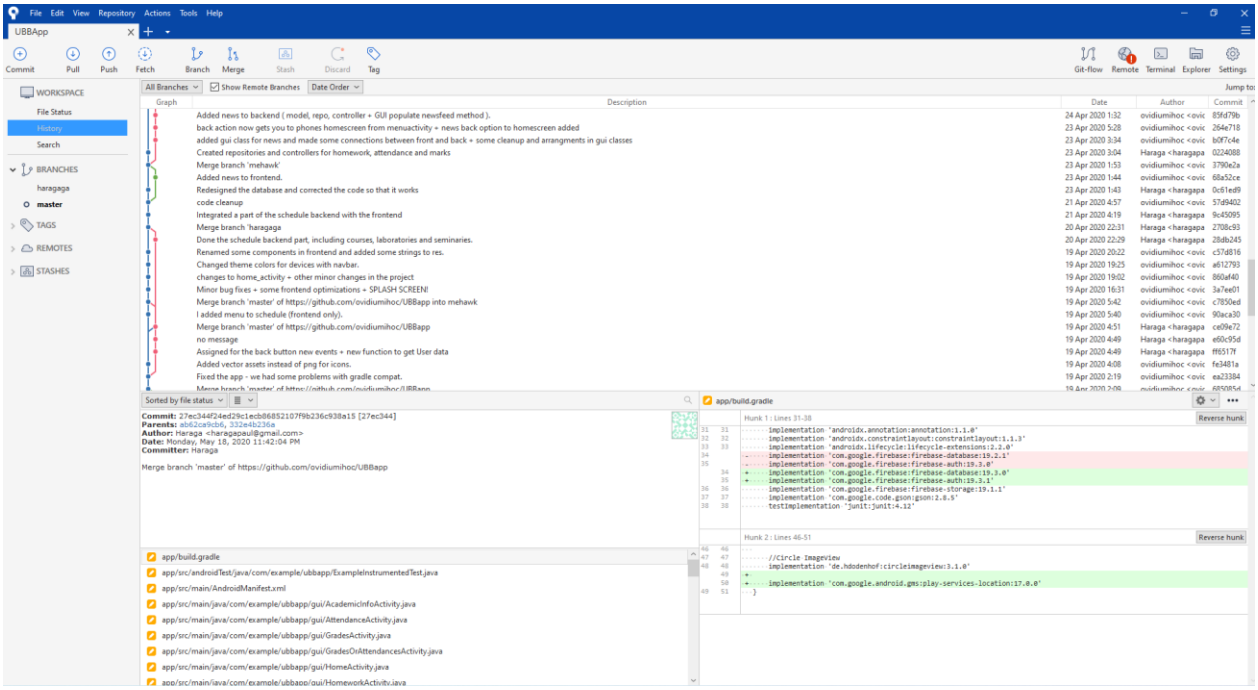

Figure 6.5.1: Preview of the SourceTree interface

Such a program as SourceTree is great to be used when working on big projects in which several people can work at once. The advantage of this approach is that the developers of the project can work on the same project, but on different parts of it and after each of the developers has completed their implementation of the code, they can merge all their work in one branch. Another advantage is that the user can roll back to a previous version of the code.

7. UBBApp

Seeing that there are other platforms that can be used as a friendly and helpful environment, the idea to create an application for the University Babes-Boylai through which the students can get the helpful information was born. Because of the complexity of the project, the team was composed of two developers to make sure we can create a stable, friendly and useful application.

Throughout the development of the project the team used a bunch of different tools to be able to work. One of the most used tools was SourceTree, which enabled us to work with the Git repository and be able to work on the same project at the same time.

7.1. Database

First step of this project was to find a suitable database for this task and we chose to use Google's Firebase application, which contains the Authentication, Realtime Database and Storage. Firebase database being a cloud-hosted database made it the perfect database for us to use and be able to sync all clients of the application.

Because Firebase database is structured as a JSON tree and that it is not based on a relational logic, the structuring of the database for the project was harder. After reading the documentation offered by Firebase for the database, we arrived at the structure that can be seen below in Figure 7.1.1.

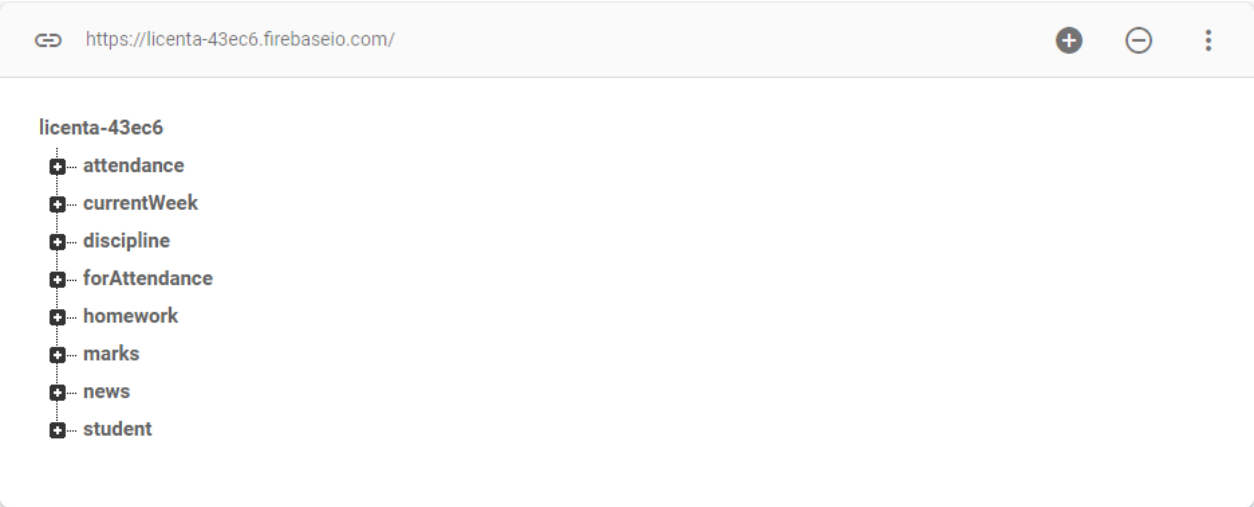

Figure 7.1.1: Preview of the structure of the database of the project

Now a short legend of the nodes, or tables as would have been called if a relational logical database would have been used:

- **Attendance:** every child of this node is an instance of a student's attendance; this instance stores the email of the student, so that we can identify his instance, and a node for every course the student is taking. Every node of the course contains two nodes that will act like a list in storing the attendance record of the student to that course, the value "1" being used when the student attended, value "0" if it not attended and the value "-1" if the attendance didn't take place. The exact structure can be seen in Figure 7.1.2.

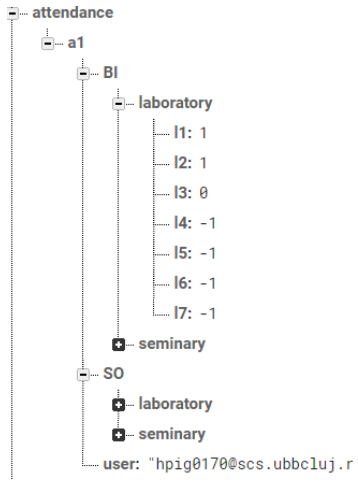

Figure 7.1.2: Preview of the "attendance" node

- **currentWeek:** this node is used to keep track of the current week of the semester, because the schedule can differ between even and uneven weeks. It has only one child called "value" where the week number is stored.
- **Discipline:** the node is the most important node of the database, holding the information about the courses, the course's professor information, the address where the course takes place, the day of the week, the interval in which it takes place, the name and the week in which takes place, this can be seen in Figure 7.1.3.

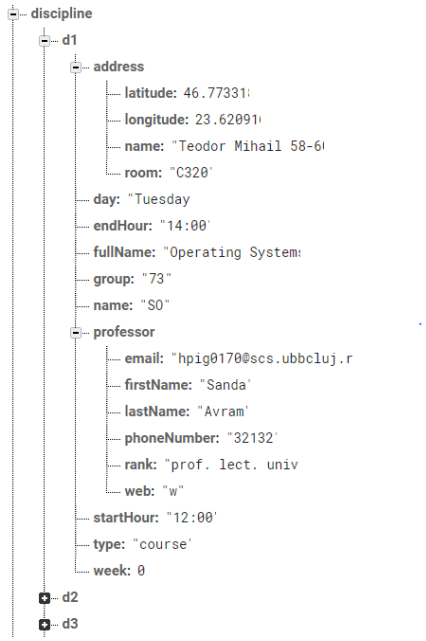

Figure 7.1.3: Preview of the "discipline" node

- **forAttendance:** having the idea to create a method through which a professor could create a session which would be used to create a list of the students which attends the class, the node "forAttendance" came to our minds. This node is used when the professors uses the web interface to create this attendance session, in which the student input will be checked against the data from database. If correct the students will be added on a list in this node called "students", which will be used to create a table so that the professor can check for fraud. When the session is done, or the user logged out, the students from this list will be written in the database as attended.
- **Homework:** this node of the tree is used to store the details of the homework: the deadline of the homework, the class for which the homework is, a download link which can be used to download the task, a text if further information needs to be added and the year of study for which this homework is.
- **Marks:** the node "marks" works exactly the same way as the "attendance" node, the only difference between them being that this node stores the grades of the student for each course, this can be seen below in Figure 7.1.4.

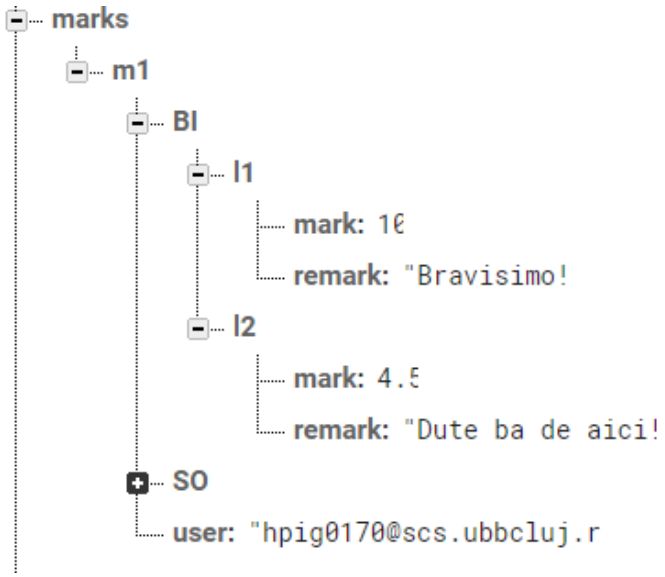

Figure 7.1.4: Preview of the "marks" node

- **News:** the "news" node is used to store the information of the news that can be displayed in the application as cards. The information hold is: the author of the news, the content, title and the targeted students, it can be a news for all students from **Student:** this node is used to store the information about the student, this being: first name, last name, email and the year of study, group and semigroup.

7.2. Android Application

The Android application is strictly for student uses, such as schedule, grades, attendances and homework visualization. Through it the students can also send emails to their professors, mark him/herself as attendant and go to the faculty official sites. Down below in Figure 7.2.1 can be seen the flow of the application, the entry for the application being in the "LogInActivity" and after logging in successfully the user is taken to the "HomeActivity", and the successful log in is saved so that the user is not needed to log in again.

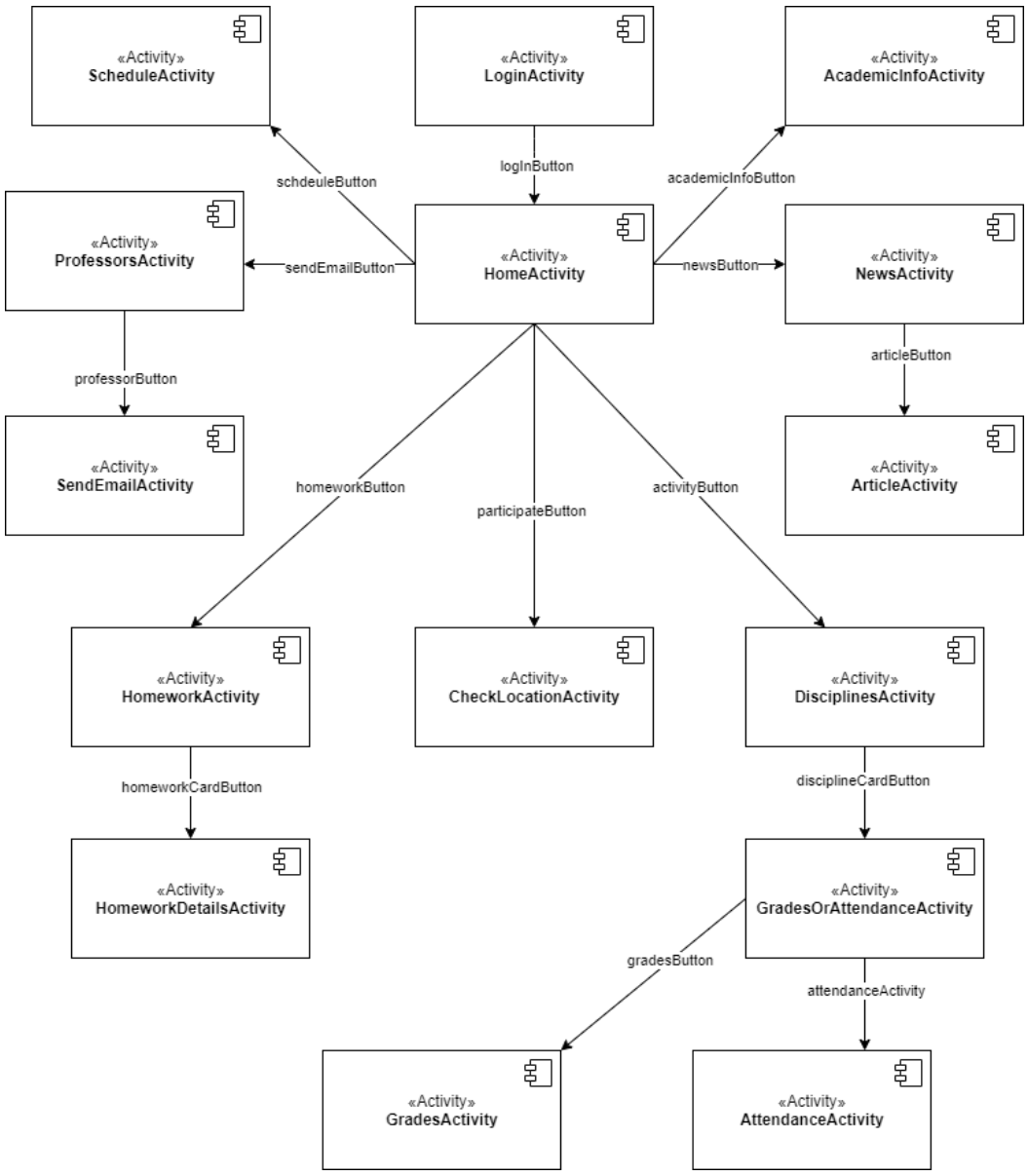

Figure 7.2.1: Preview of the Android application activity flow

One of the features that I have implemented is the “ScheduleActivity”, it being the activity where the schedule will be displayed. After the users clicks the schedule button he/she will be taken to a new window, where the current day of the week will be selected and the classes for that day will be displayed in chronological order according to starting hour, as can be seen in Figure 7.2.2.

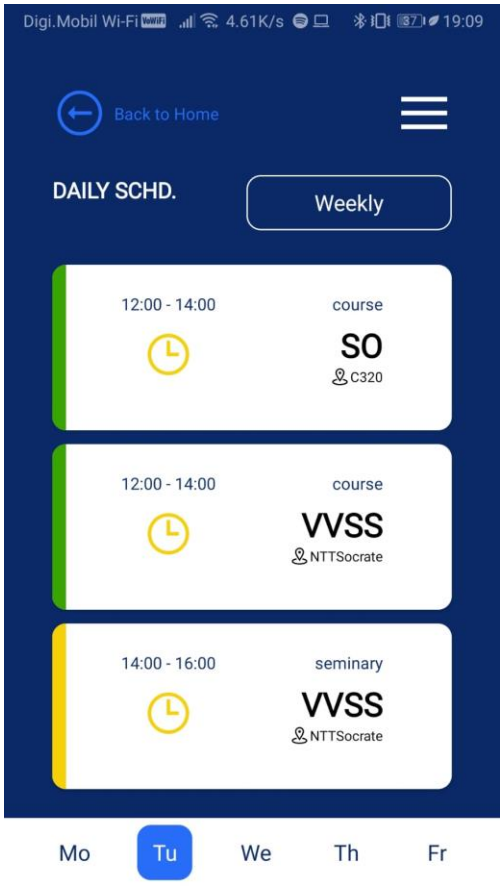

Figure 7.2.2: Preview of the" SchdeuleActivity" with Daily option on

The student can navigate through the days of the week, to see what classes are coming up, by clicking the buttons at the bottom of the window, currently in Figure 7.2.2 the "Tu" button is toggled so that means the Tuesday classes are shown. Beside the information about the classes the user has, the card in which the information is displayed is also clickable, the functionality of the card being the option to open an application for mapping, like Google Maps, and get direction for the location where the class will be held.

This feature is the "Daily" schedule, but the user has the option to see the "Weekly" schedule, which will display the number of classes for each day of the week, this can be seen below in Figure 7.2.3.

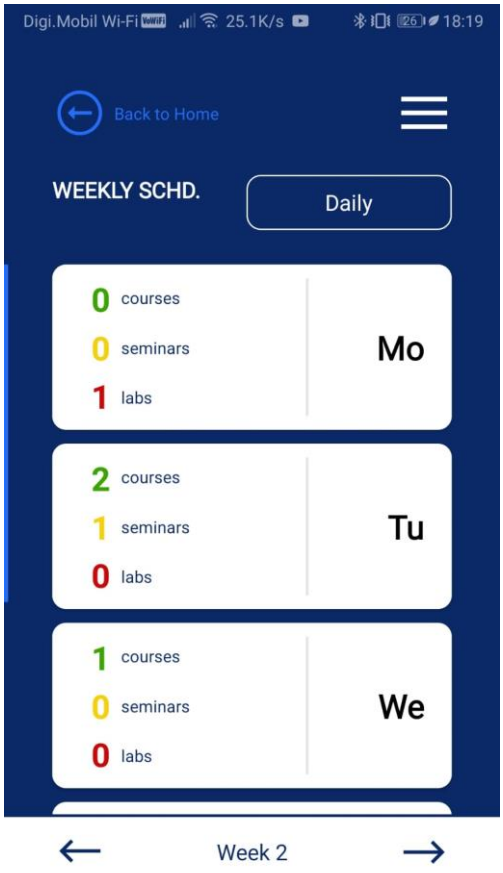

Figure 7.2.3: Preview of the "ScheduleActivity" with Weekly option on

Both options of the schedule, when used, are programmed to load the data from the database. In case changes appear, they take effect in real time, this being the one of the Firebase's Realtime Database feature, to be more specific, asynchronous call. This being said, for the development of the application we needed a callback interface so that we can respect the MVC (Model-View-Controller) model. The use of this interface was to return the found data from the database to the view, because being an asynchronous call the data will not be seen from outside it.

A second feature developed by me for this application consists of the "HomeworkActivity" and "HomeworkDetailsActivity", which were designed so that the students can see the assignments that will need to be turned in. In Figure 7.2.4 we can see that the current week of the semester is displayed, and under it are the assignments from this week on, including it, the due week for each of them, the name of the class that they are for and the type of class. The user can get to Figure 7.2.5 if the cards that holds the information about the homework, from Figure 7.2.4, is clicked. In the "HomeworkDetailsActivity" the user will be able to read further information about the homework and if there is any support material the user can download it and read it.

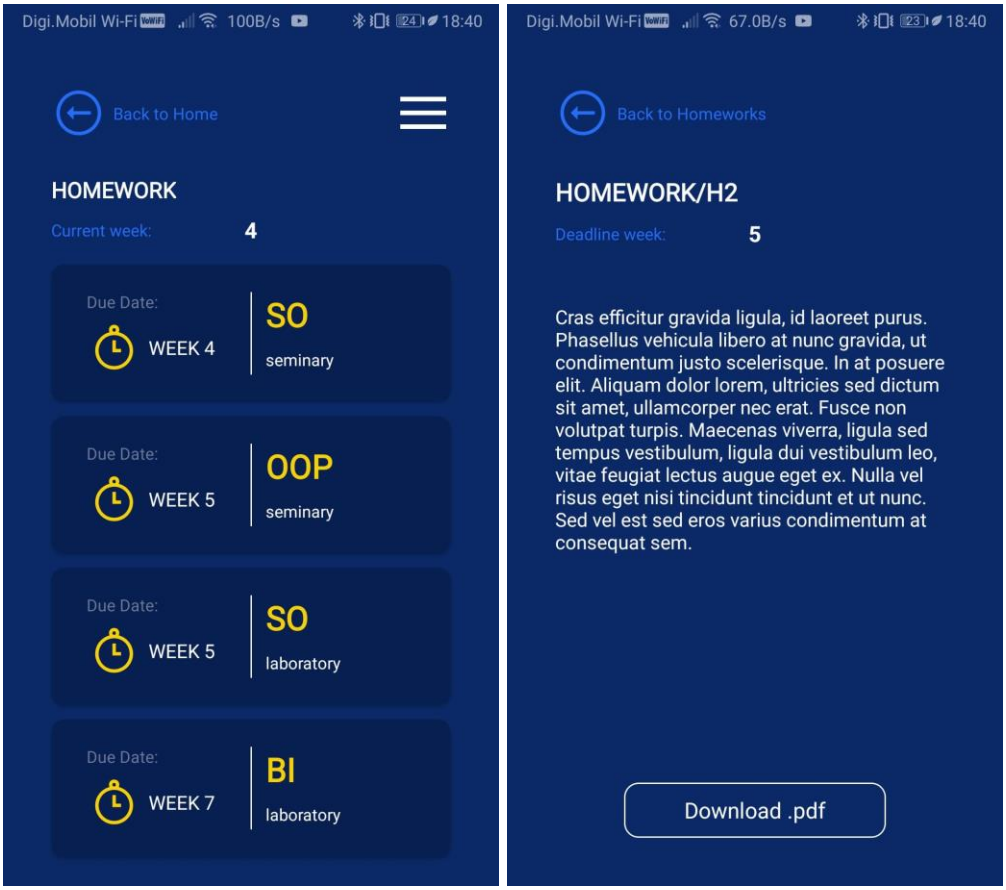

Figure 7.2.4, Figure 7.2.5: Preview of the "HomeworkActivity" (left) and "HomeworkDetailsActivity" (right)

The last feature developed for the Android application was the news section, where the students will be able to read news, for example about the changes that could happen in their schedule. This feature was designed so that there could be general news, like changes that happened in the university or faculty, and news that are specific for the specialization the student is, this can be seen in Figure 7.2.6.

As was the case for the other features, the cards here can be clicked and the user will be taken to the "ArticleActivity" where further information can be read about the article, such information being the title of the article, the author, the date and the article itself, as seen in Figure 7.2.7.

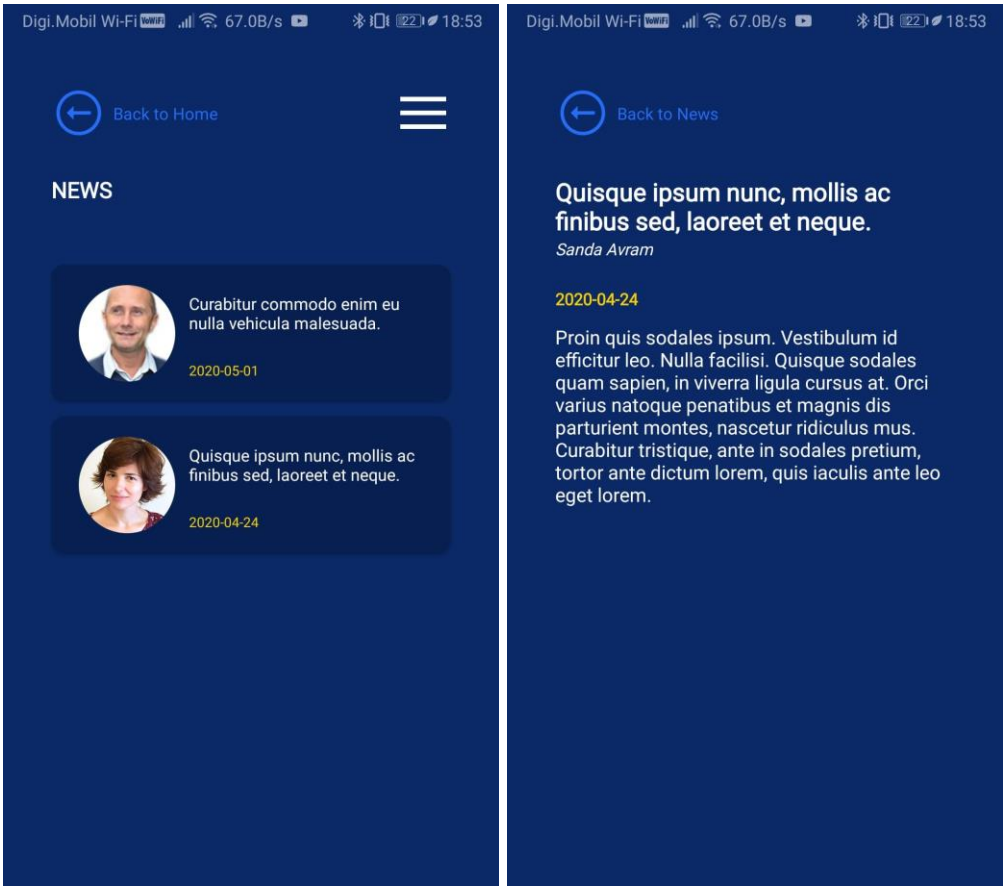

Figure 7.2.6, Figure 7.2.7: Preview of the "NewsActivity" (right) and "ArticleActivity" (left)

For the development of the "NewsActivity" we decided to show an image of the author too, and to implement this we used a framework to load the images from the database storage called *Glide* [16].

7.3. Web Application

Because of the need for a platform for professors too, to manage their class's grades, attendance and schedule, we decided to build a web application for them. The web application is designed as an administrative platform for the user, in this case the professor, to do her job using it. The decision of using a web platform was due to the compatibility with a large number of devices.

The first page the users will be prompted to is the log in page, that can be seen in Figure 7.3.1. Here the implementation of the log in was done using Firebase's Authentication, it being fairly easy and direct, using the Firebase's documentation [17].

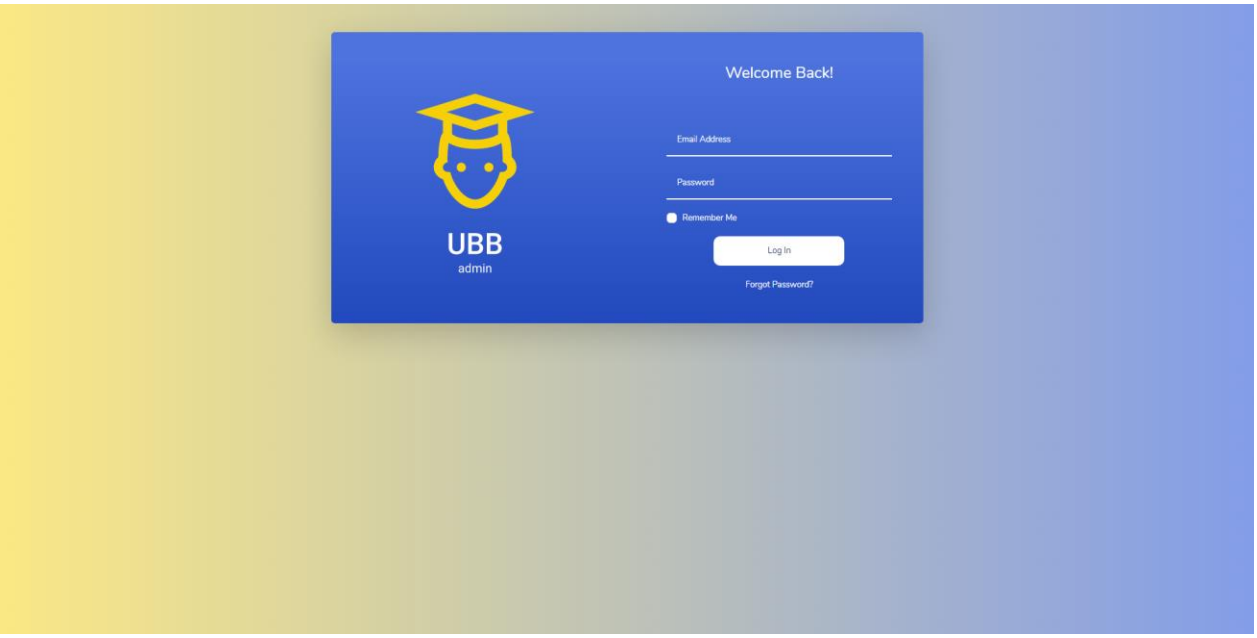

Figure 7.3.1: Preview of the log in screen for the web application

Because of the use of JavaScript to develop the application, it enabled us to make a responsive application. If the inputs for the log in are false, a failed to log in message will be displayed without the need to load the page.

After logging in successfully the user will be redirected to the dashboard, where all the needed tools will be. From the dashboard the user first needs to select one of the available classes from the section “My Courses”, see Figure 7.3.2. After selecting one of these, the tools from the “Utilities” section will be available, to make sure that the tools are not used before selecting a course, they are disabled until then. The user can edit the schedule for that class, make an attendance list, give a student a grade or give the assignment to the students, see Figure 7.3.3. Again because of the implementation of the web page using JavaScript we were able to make it responsive and interactive, the changes after selecting a course or a tool taking place right away, with no need for reloading the page.

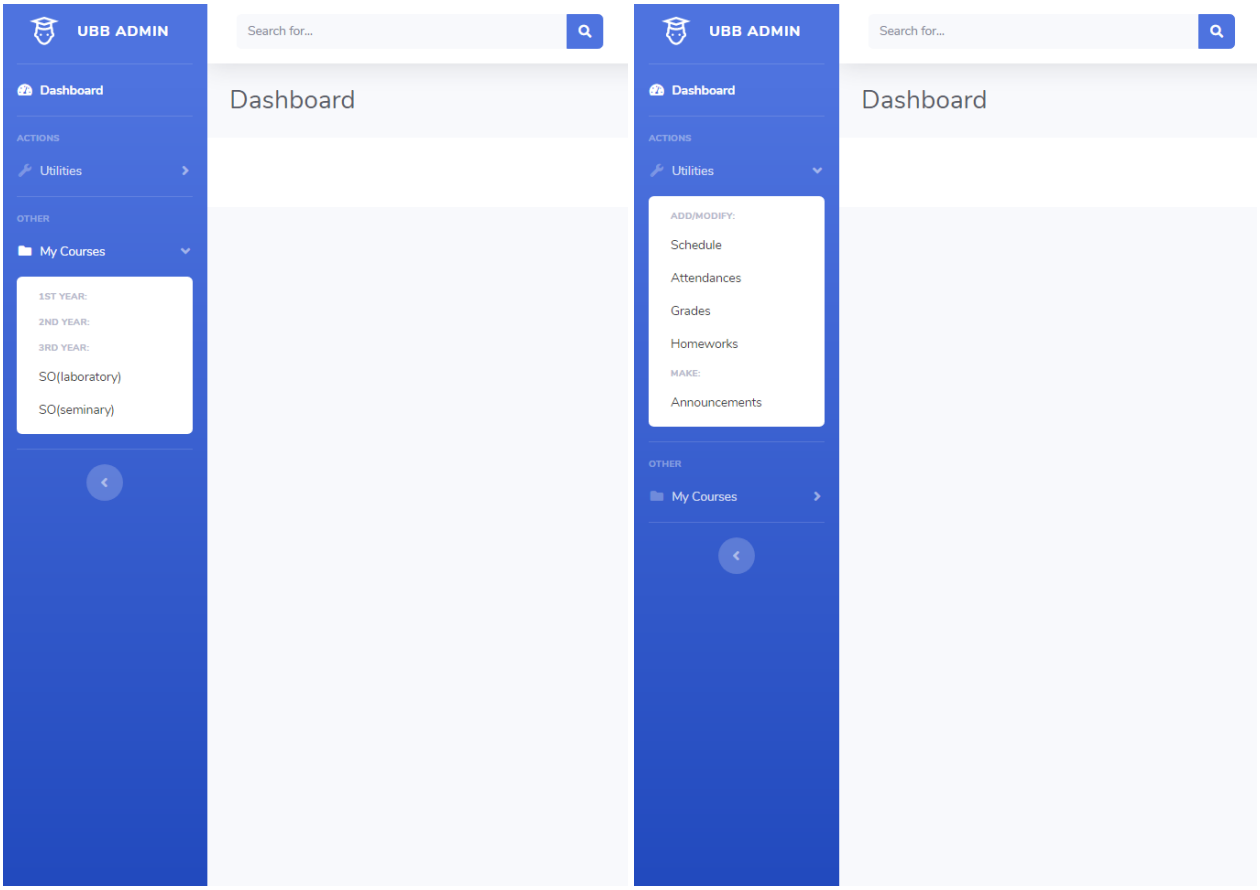

Figure 7.3.2, Figure 7.3.3: Preview of the dashboard with section "My Courses" open (left) and section "Utilities" open (right)

My involvement with the development of the web application was in the backend, making all the functionalities of the pages.

8. Difficulties during the application development

During any process of creating an application there are difficulties and for our application there is no exception. The problems that arose were: the challenge to work with Firebase database, trying to implement some design features and working as a team on this project.

One of the greatest difficulties that I have faced with the development of the application was to work with Firebase's asynchronous way of communication. This being my first time working with something like this. First problem with it was that, being asynchronous, the rest of the variables were not available, so that meant we had to create a callback interface to return my data from these calls.

Another challenge that I have faced was to implement some design features for this application, such features were to make the background of a button change color to create the effect of a pressed button and to clear and remodel the current window. These might not seem such a hard task to implement, but it's a time consuming task and at some point there might not even be a feasible solution.

With no experience of working as a team on a project like this, working with a teammate was quite a challenge. This being the first time we used a Git repository too, but this experience taught me a great deal. The difficulties of this were to constantly keep in touch with the partner, trying to understand other's code and working with Git.

9. Future developments

The shape of the application has changed a lot since the beginning, and through the development we found new ideas for how this application could evolve. The most interesting idea that we found during it was the need for animations for the user interface, this making the overall user experience more enjoyable. Beside animations I think there is room for more features that could be added to the application, to make it even more useful, such as to implement the schedule of the buses that pass through the proximity of the places where the classes take place, an in app alert to notify the students of schedule changes, a web version of the android application so that the students can have access to these features from more devices and so that users that have other operating systems can have access.

I think the first extension to this application should be to create the web version of the android application, because of the reach that this will bring. After this the best way to improve the platform is to make a survey and get information about what should be added to it, because the best input and ideas can only come from students that see the voids in the system.

10. Conclusion

Throughout the development of the application we have found new ideas of how this application can evolve into a stable and helpful application for the students in the university and we arrived at this version. I think that such an application would be a big step for educational systems in the future, because it would make the data of the students accessible to them and it would give the users the possibility for better organization.

I think that this application has a lot of room for improvement and could become a great platform to bring the information that usually is needed by students, instead of making them search for it.

11. Bibliography

[1] “AcademicInfo UBB”

< https://academicinfo.ubbcluj.ro/Info/ >

[2] “Moodle UBB”

< https://moodle.cs.ubbcluj.ro >

[3] “MySouthampton – Google Play”

<https://lh3.googleusercontent.com/xQV_ycLUeayR1fv1DRhSjQMJTiYOidnJAvFQtHtdrKxLpjQkgfLIfardssthoFJ5Ig=w720-h310-rw>

[4] “Android Studio – Introduction “

<https://developer.android.com/studio/intro>

[5] “Android Studio – Documentation (Devices)”

<https://developer.android.com/docs>

[6] “JetBrains – IntelliJ – Overview “

<https://www.jetbrains.com/help/idea/discover-intellij-idea.html>

[7] “JetBrains – IntelliJ – IDEA “

<https://www.jetbrains.com/idea/features/>

[8] “Firebase – Authentication – Documentation - Introduction“

<https://firebase.google.com/docs/auth>

[9] “Firebase – Realtime Database – Documentation - Introduction”

<https://firebase.google.com/docs/database>

[10] “Firebase – Storage – Documentation – Introduction”

<https://firebase.google.com/docs/storage>

[11] “Firebase – Authentication – Sign In “

<https://firebase.google.com/docs/auth/android/firebaseui>

[12] “Firebase – Realtime Database – Structure Data”

< https://firebase.google.com/docs/database/android/structure-data >

[13] “Firebase – Realtime Database – Monitor Usage”

< https://firebase.google.com/docs/database/usage/monitor-usage >

[14] “Firebase – Realtime Database – Rules “

< https://firebase.google.com/docs/database/security/securing-data >

[15] “WebStorm – Features”

< https://www.jetbrains.com/webstorm/features/ >

[16] “Bumptech – Glide”

< https://github.com/bumptech/glide >

[17] “Firebase – Authentication – Web”

< https://firebase.google.com/docs/auth/web/password-auth >