

Software Application for Disabled Students

Laszlo Csiki

Department of Engineering and Computer Science,
“Vasile Goldis” Western University of Arad, Romania
csikilacko95@gmail.com

Andrei Gabor

Department of Engineering and Computer Science,
“Vasile Goldis” Western University of Arad, Romania
94 Bulevardul Revoluției, Arad, Tel. +40 257 211 884
andrei.gabor@uvvg.ro

Marius Popescu

Department of Engineering and Computer Science,
“Vasile Goldis” Western University of Arad, Romania
94 Bulevardul Revoluției, Arad, Tel. +40 257 211 884
mpopescu@uvvg.ro

Antoanela Naaji

Department of Engineering and Computer Science,
“Vasile Goldis” Western University of Arad, Romania
94 Bulevardul Revoluției, Arad, Tel. +40 257 211 884
anaaji@uvvg.ro

Abstract

Modern universities across the world rely increasingly on information and communication technologies, both in their educational and research processes, and the management of academic databases. The android app “Virtual University” is a virtual platform designed especially to be used by students and is aimed at accessing, in real time, information connected with the subjects being studied (grades, announcements, notification), financial situation, as well as connecting with the secretariat of the university. The application allows each student to check their financial situation and to pay their educational fees and dues online. The app is meant as an expansion of existing functionalities within universities, and not as a replacement for them. Using the application may lead to an improvement in the quality of life, particularly for people with disabilities.

Keywords: Virtual University, mobile application, Android app, disability

1. Introduction

Recent discoveries in information technology have led to the development of applications that could replicate real-life situations as closely as possible. The development of wireless and sensor technologies has provided a different way to establish interactions with resources within virtual universities (Hwang, Tsai, Yang, 2008).

Mobile terminals are the most commonly used devices in daily life, and according to recent studies, over 50% of Internet searches are made from a tablet or mobile phone. Users tend to look for a mobile application for anything, being almost dependent on their usefulness in everyday life.

Given this fact, creating an educational mobile application could be a very important factor. Through the mobile application designed by us, students can interact with various items of information on the status and coverage of courses (grades, scheduling of courses and labs, announcements of last-moment changes), as well as interact with the secretariat of the university in real time. The application works with a MySQL server which is integrated into the LMS system of the virtual university.

One of the features of the application is also the possibility to send questions and answers through a text message centre between participants. Additionally, the application concept places special emphasis on increasing communication between users.

Each student may consider the comfort and flexibility of online services provided by virtual universities. People with disabilities comprise a fairly high percentage from the population of Romania, amounting to a number of 784,527 according to a press release by the National Authority for People with Disabilities within the Ministry of Labour and Social Justice, and increasing each year according to the chart in Figure 1 (The Romanian Ministry of Labor and Social Justice, Statistic Bulletin, 2017). The number of severely disabled people accounts for 36.91% of the total (compared to 36.07% on 31 March 2016), those moderately disabled for 51.66% (compared to 52.24% on 31 March 2016), and those with mild disabilities for 11.43% (compared to 11.69% on 31 March 2016). There are several reasons for this constant increase, including population ageing and technological progress that can easily identify disability among people.

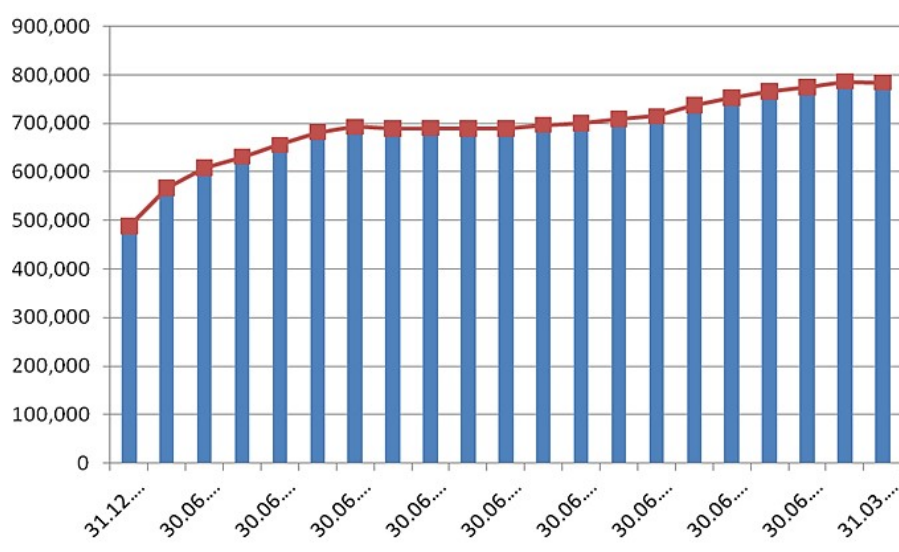


Figure 1. Evolution of the number of disabled people, December 2006 – March 2016 (The Romanian Ministry of Labor and Social Justice, Statistic Bulletin, 2017)

For starters, students can manage their issues regarding courses, scheduling, grades, relationship with the faculty office. Issues concerning the special needs of disabled people such as special monitors, mice, or keyboards can also be solved by online application.

By virtual connection to resources provided by our application, expenses on transportation and the companions of students with special needs can be reduced. Our application provides flexibility for students to express themselves and interact with their classmates and professors. A rapid and efficient interaction provides enough free time for disabled students to complete their tasks.

Android is an operating system for mobile telecommunication terminals based on a Linux kernel, developed by Google and, subsequently, by the Open Handset Alliance consortium.

Android allows developers to write codes managed in Java language, controlling the device through Java libraries developed by Google (Hwang et. al., 2008; Zhang et. al., 2013). The “Virtual University” (VU) app described in this paper can be used to access information on subjects and students (such as grades, announcements) and to communicate with the secretariat of the institution, through text messages.

Likewise, it can check financial data, and if necessary make the online payment of academic fees or dues. Android Studio was used to develop the “Virtual University” application, with frequent

recourse to the following elements: `listview`, `button`, `textview`, `edittext`, `spinner` and `http connector`.

Android Studio is a new IDE (*Integrated Drive Electronics*), proposed by Google, which allows developing mobile applications and provides many advanced facilities in installing the work environment, performance, code completion and refactoring. In addition to these, we used the MySQL database and PHP scripts (Aziz, Mitchell, 2007), which is an open source language, suitable for the development of web applications and which can be integrated into HTML.

The next part of the paper is organized as follows: section 2 which describes the structure of the application, section 3 which describes the application from technical point of view and conclusions containing also few ideas about future work.

2. Structure of the Android app

An Android app has a lot of components. Each component is a point through which the system can enter the application, which does not mean that an Android app can begin anywhere. Some components depend on others, but each component is a separate, unique entity.

There are four types of components in an Android application. Each has a different purpose and life cycle, and they are created and destroyed differently. These four components are: `activities`, `services`, `content providers` and `broadcast receivers`.

The *Activity component* consists of a single screen with graphic elements. For example, an email application can have one activity to show the inbox and another activity to compose an email. These two activities work together to form a pleasant experience for the user, although they are independent from one another. In terms of programming, these are classes expanding the `Activity` class. Unlike other systems in which one program contains several windows displayed simultaneously, in Android the screen is occupied by only one window at any time.

Moreover, the windows are connected through the simple fact that one window gives rise to several windows. The window appearing on the screen when the application is started is considered to be the main window, from which all other windows start.

To create an activity, a class will be created expanding the `Activity` class. Upon start-up, the program will generate it and events will be used to interact with the window. Some of the most important events are: `onCreate` and `onPause`.

The Virtual University application used `Activity` only for the `Main Activity`, which manages `Navigation Drawer`, and for the `Login Activity`.

The application is based on `Fragments`, which are classes amounting to an independent component, executed in the context of an activity, to facilitate the creation of interfaces that can adapt to different types of screens. Thus, a fragment can be seen as a portion of interface that can be reused within several activities or for a different positioning of the same interface – if other types of screens are used (Figure 2).

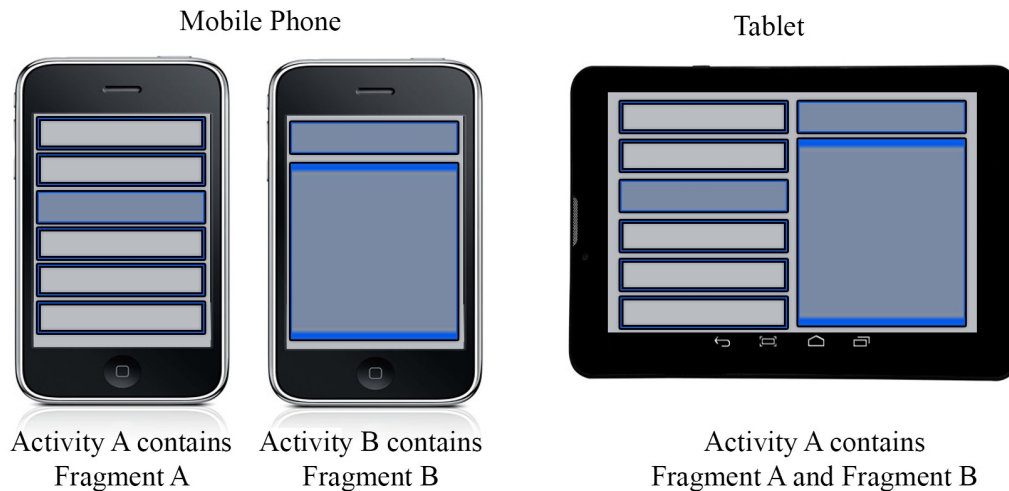


Figure 2. Android fragments

There are nine fragments in the application: 1. Student Data; 2. Subjects; 3. Grades; 4. Financial; 5. Pay; 6. Announcements; 7. My messages; 8. Write message; 9. Read message.

The `backend` part is done in PHP language. A script was created for each connection to the database, which runs a MySQL interrogation and returns a `json array`. The application downloads the returned `array` and afterwards reads the data.

Services are components running in the background for operations that run more. Services do not have visual representation. For example, a service can play a song from a *playlist* while the user is in another application, or can synch the email while the user is watching a video. Another component can start a service, such as an activity, or can be connected to the service to interact.

Content providers work with data which are shared and which can be in system files, in an SQL database, on the web, etc. Through `content providers` an application can interrogate, or even modify that data (if it has the required permissions).

Contacts in the phonebook can be accessed through the e-mail application using `content providers`. For example, the NotePad application uses `content providers` to save notes from there. *Broadcast receivers* are the component responding to announcements/system alarms: discharged battery, download completed, they do not have graphic interface but can create notifications.

The application is free and very useful in different moments in students' lives and provides interaction with various items of personal and group information. The size of the application file is 2.8 MB and can be downloaded from the link: <http://informatica-uvvg.esy.es/vu/app-Virtual.apk/>. Minimum requirements for systems on which the application can run are: resolution 480x800 pixels, API level 19, Android system and internet connection (3G/WIFI). Recommended system requirements are: resolution 720*1280, API level 23, Android system and internet connection (3G/WIFI) (Ma et. al., 2013; Shewale et. al., 2014).

3. Description of the application

3.1. Implementing the application menu

Starting the application displays the logging window (Figure 3), where each student can log in to the platform using personal data. After filling out the boxes, access to the platform is granted by tapping the "Login" button. After logging in the users can view news.

On the `Application Bar` there are two icons, the first is Menu (Figure 4), and the second is Messages with which "My Messages" are accessed. The menu comprises the following sections:

Student Data: section providing information regarding the students of the university, for example, name, surname, birth date, country, county, residence, sex, citizenship, nationality, personal identification code (CNP), major. The application connects the database using a PHP script, which requests the updated data from the database, which it downloads using HTTP URL Connector. Then, the program fills in the empty boxes (TextView) with downloaded data.

Subjects: displays the subjects taught in the education institution. There are 18 empty boxes (TextView), and in the background the HTTP Connector downloads the required data from the database and fills in the empty boxes.

Grades: information on students' grades, by declaring a List View, where each line contains 4 additional rows and 2 columns, with row 1 reserved for the title of the subject, and the second, third and fourth row contain boxes with partial grades, therefore, the second column contains only grades.

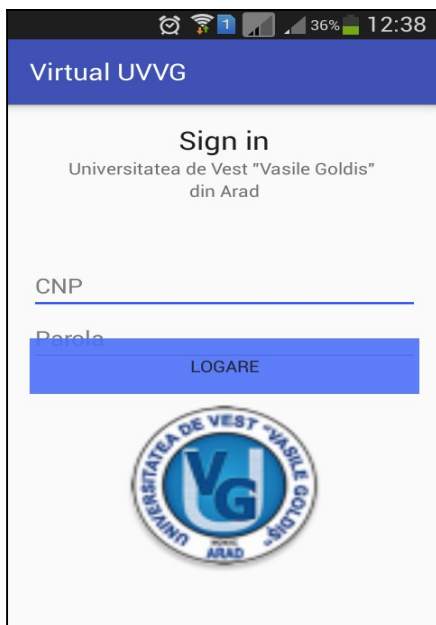


Figure 3. Login screen

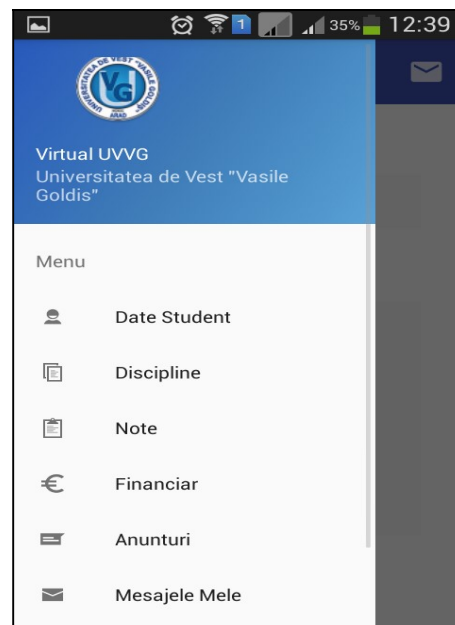


Figure 4. Menu of the Android app

Financial: section dedicated to checking financial data and online payments. Thus, financial data can be viewed, if the student is enrolled in paid education, then the downloaded data contain the total price/year, paid/year and the rest. The rest contains unpaid money. If the user wished to make a payment, they can pay online by tapping the “Pay online” button. Tapping the button will open a window where the payment can be made. Here the user must input the following data: credit card type, card number, expiration date and security code if available. After filling in the boxes, the payment is finalised by tapping the “Pay” button. If everything is in order, there will be a Toast displaying the message “Payment completed”.

Announcements: news is viewed after logging in to the platform, where announcements can be read. An announcement element contains three lines: line 1 displays the title, line 2 comprises the date of issue, and the third line contains the message/announcement. The application downloads the required data upon each login and thus updated announcements can be viewed.

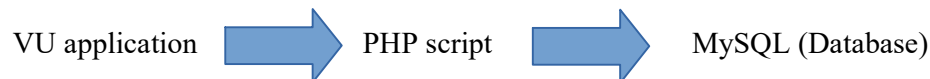
My messages: section dedicated to communications with the secretariat or between students, of the CNP of the recipient is not known. In this section a List View containing messages was declared. One line in the list contains 4 elements: element one is an image, element two contains the CNP of the sender (if a message is received from the secretariat, only the name of the secretariat is seen, not the CNP), element 3 contains the title of the message and element four contains the

message, where a maximum number of 20-30 characters are visible. An FAB (*Float Action Button*) is implemented in this section and can be used to write a new message. Only received messages appear, namely the inbox folder. Unfortunately we can only see the inbox, as the application described in the paper is under testing. Here the user has to tap the message they wish to view, which opens a new window where the content of the message can be viewed. If the user wants to reply to the message, then they have to tap the “Reply” button, opening a window where the boxes `Title`, `Recipient` and `Message` must be filled in. A title will be written under the “Title” section; under “Recipient” the user does not have to select anything as the selection is automatic. The message is written and then sent using the “Send” button, and, if necessary, it can be deleted. In the “new message” section the user must fill in the sections `Title`, `message`, and under `Recipient` a `Spinner` has been implemented, where the following destinations are chosen: 1. Secretariat FSEII (Secretary of the Faculty of Economical Sciences, Computer Science and Engineering); 2. Secretariat FSJ (Secretary of the Faculty of Law); 3. Secretariat FSOEUFS (Secretary of the Faculty of Socio-Humanistic Sciences and Physical Education and Sports); 4. Secretariat Medicina (Secretary of the Faculty of Medicine); 5. Secretariat Stomatologie (Secretary of the Faculty of Dentistry); 6. Secretariat Farmacie (Secretary of the Faculty of Pharmacy); 7. Other – if this option is selected, then the CNP of the recipient will have to be filled in.

Exit: to exit the application / log out from the platform. After logout the application returns to the login window.

3.2. Connection of the application with the database

The application makes the connection to a MySQL database, where the data required for each section are downloaded.



The application creates a connection with a PHP script which implements a connection to a MySQL database:

```

<?php
require "conn.php";
$user_name=$_POST["user_name"];
$sql="select*from Financial where CNP like '$user_name'";
$result=mysqli_query($conn,$sql);
$response=array();
while($row=mysqli_fetch_array($result)){
    array_push($response,array("Total"=>$row[1],"Paid"=>$row[2],"Rest"=>$row[3]));}
echo json_encode(array("server_response"=>$response));
$conn->close;
?>
  
```

The php script requests the data, in this case `Total`, `Paid` and `Rest`, that is, the financial data in the `Financial` table, and after these it creates an array in which it will find the response.



The program will wait for the response, which is a JSON (*JavaScript Object Notation*) (Munch-Ellingsen, Karlsen, 2015) and will download on this JSON. The code runs in the background (`Background Task`), creating a connection to a php script, which requests POST data (for example: `user_name`). After receiving the data, the script creates a response in JSON format, and `HTTP URL Connector` awaits this response, which will be declared as a `string`.

@Override

```
protected String doInBackground(String... params){
    try {user_name=params[0];
        URL url=new URL(json_financial_url);
        HttpURLConnection httpURLConnection=(HttpURLConnection) url.openConnection();
        httpURLConnection.setRequestMethod("POST");
        httpURLConnection.setDoOutput(true);
        httpURLConnection.setDoInput(true);
        OutputStream outputStream = httpURLConnection.getOutputStream();
        BufferedWriter bufferedWriter=new BufferedWriter(new
            OutputStreamWriter(outputStream,"UTF-8"));
        String post_data=URLEncoder.encode("user_name","UTF-8")+"="+
            URLEncoder.encode(user_name,"UTF-8");
        bufferedWriter.write(post_data);
        bufferedWriter.flush();
        bufferedWriter.close();
        outputStream.close();
        InputStream inputStream=httpURLConnection.getInputStream();
        BufferedReader bufferedReader=new BufferedReader(new
            InputStreamReader(inputStream));
        StringBuilder stringBuilder=new StringBuilder();
        while ((JSON_STRING = bufferedReader.readLine())!=null){
            stringBuilder.append(JSON_STRING+"\n");
        }
        bufferedReader.close();
        inputStream.close();
        httpURLConnection.disconnect();
        return stringBuilder.toString().trim();
    } catch (MalformedURLException e){e.printStackTrace();}
    } catch (UnsupportedEncodingException e){e.printStackTrace();}
    } catch (ProtocolException e){e.printStackTrace();}
    } catch (IOException e){e.printStackTrace();}
    return null;}
}
```

4. Conclusions

It is a known fact that the number of disabled students enrolling in universities is increasing. It is also known that they do not have the same graduation rate as able-bodied students. Disabled students should benefit from auxiliary services in addition to those provided by online courses. The Interaction of disabled students with the virtual university must be considered as a web accessibility issue. We chose to address this using the proposed application, for the very reason of meeting and solving the challenging requirements of people with disabilities.

The “Virtual University” described in the paper is an original virtual platform, designed for mobile telecommunication terminals, which should be seen as a bridge between the higher education institution and the learner, making it easier for the secretariat of the university to track the behaviour and activity of the student, or to present their academic status in a customised way.

The application is simple, unpretentious, and user-friendly, but delivers satisfactory results in academic communication, and the logical structure of the application can be developed for any higher or secondary education institution. Additionally, it can expand functionality by adding modules in the form of a chat or a forum.

Among the benefits described above, the most important are, for the university secretariat, sending information to students in real time and tracking feedback, and for students, the possibility of using the simplest and most efficient way to communicate through mobile device and obtain/access information stored/requested by the university secretariat.

We may conclude that providing 3G and 4G Internet connections adds a host of facilities to the mobile e-learning environment. Some facilities were also presented in our work through the interaction between student and secretariat, viewing exam results and number of credits. In addition to other applications, our proposal offers students the possibility to check their financial data, and, if necessary, make online payments for fees or dues. We shall continue to support these directions in the future, in the area of information and communication services, between student and university.

References

- Aziz, A., Mitchell, S. (2007). An Introduction to JavaScript Object Notation (JSON) in JavaScript and .NET, Retrieved from <https://msdn.microsoft.com/en-us/library/bb299886.aspx>
- Hwang, G.J., Tsai, C.C., Yang, S.J.H. (2008). Criteria, strategies and research issues of context-aware ubiquitous learning. *Educational Technology & Society* 11.2, 81-91
- Ma, X., Zhao, Y., Zhang, L., Wang, H., Peng, L. (2013). When Mobile Terminals Meet the Cloud: Computation Offloading as the Bridge. 0890-8044/13/\$25.00, IEEE
- Munch-Ellingsen, A., Karlsen, R. (2015). Two-factor authentication for android host card emulated contactless cards. *Mobile and Secure Services (MOBISSECSERV)*, First Conference. Retrived from <http://ieeexplore.ieee.org/document/7072874/>
- Shewale, H., Patil, S., Deshmukh, V., Singh, P. (2014). Analysis of Android vulnerabilities and modern exploitation techniques. *Journal on Communication Technology*, Vol.5, 864- 867
- The Romanian Ministry of Labor and Social Justice, National Authority for People with Disabilities (2017). Statistic Bulletin, First Trimester, Retrieved from <http://anpd.gov.ro/web/transparenta/statistici/trimestriale/>
- Zhang, W., Wen, Y., Wu, J., Li, H. (2013). Toward a Unified Elastic Computing Platform for Smart phones with Cloud Support. 0890-8044/13/\$25.00, IEEE



Laszlo CSIKI (b. January 30, 1995) received his BSc in Informatics (2017), from “Vasile Goldis” Western University of Arad. Now he is MSc of computer science in Department of Engineering and Computer Science, Faculty of Economics, Engineering and Computer Science, “Vasile Goldis” Western University of Arad, Romania. His current research interests include different aspects of artificial intelligence applied in biological systems. He has (co-) authored 2 papers, more than 3 conferences participation.



Andrei GABOR (b. September 9, 1969) received his BSc in Electronics and Telecommunication (2000), PhD in Electronics and Telecommunication (2014) from “Politehnica” University of Timisoara, Romania.. Now he is lecturer of computer science in Department of Engineering and Computer Science, Faculty of Economics, Engineering and Computer Science, “Vasile Goldis” Western University of Arad. His current research interests include web technologies, e-learning technology, multimedia databases, computer networks. He has (co-) authored and (co-) edited 7 books and more than 30 papers.



Marius POPESCU (b. May 19, 1965) received his MSc in Automation and Computers (1990) and in Informatics (1995), PhD in Electrical Engineering (2002) from University of Craiova, Romania. Now she is professor of computer science in Department of Engineering and Computer Science, Faculty of Economics, Engineering and Computer Science, “Vasile Goldis” Western University of Arad, Romania. He has (co-) authored and (co-) edited 21 books and more than 200 papers. His research interests involve signal processing, modelling, optimization and simulation processes. His preoccupations in the above-mentioned fields are reflected in a number of over 20 projects, as manager or member, funded by national or international research authorities or county council.

Antoanela NAAJI (b. February, 1963) received her MSc in Automation and Computers (1987), PhD in Mechanical Engineering (2005) from “Politehnica” University of Timisoara, Romania. In 2009 she received a LL.M degree in Administrative Law and the Statute of Public Officers from “Vasile Goldis” Western University of Arad, Romania. Now she is associate professor of computer science in

Department of Engineering and Computer Science, Faculty of Economics, Engineering and Computer Science, “Vasile Goldis” Western University of Arad, Romania. Her current research interests include web technologies (e-learning), biomechanics (modeling of biological systems) and medical informatics. She has (co-) authored and (co-) edited 13 books and more than 90 papers. Her preoccupations in the above-mentioned fields are reflected in a number of over 20 projects, as manager or member, funded by EU, national or international research authorities or county council. She is a senior member of IEEE, member of EAEIE (European Association for Electrical Engineer and Information Engineering), AGIR (General Association of Romanian Engineers) and SRIM (Romanian Society for Medical Informatics).

