

Ibragimov U.M., Kozhabekova A.E., Dildabaeva M.S.

M.Auezov South Kazakhstan State University, Shymkent, Kazakhstan

ON ONE APPROACH FOR THE DEVELOPMENT OF A JAVA LANGUAGE APPLICATION FOR A TERMINOLOGICAL DICTIONARY

Introduction. The last few years in Kazakhstan there has been an increasing spread of devices based on mobile platforms. From the point of view of analysts, the iPhone and Android platforms will develop in the next few years. Mobile application development in many aspects is similar to PC application development. At the same time, the development of mobile applications poses new challenges for programmers. Such tasks, in particular, include the development of a terminological dictionary.

In the work, we consider a certain class of applications for mobile devices based on the Android OS platform. The main limitations of this class are as follows: Applications are developed using the principle of separating the program code of the application logic from the program code of the application user interface (for example, using the “Model” - “Presentation” technology: Each application is developed for a specific, narrow range of tasks, due to which the number of application representations is a finite number of the order of 1000. For any application, there is a finite set of user interface elements that allows you to Nost set views of the application (its UI).

Program implementation. Applications for the Android platform are collected in bytecode for execution by the Dalvik virtual machine. During the assembly process, the bytecode of the Java virtual machine is reassembled into the bytecode of the Dalvik virtual machine with a special utility that is part of the application development toolkit [1,2]. The application consists of two parts, this is a server and client application.

Client part. We will begin to describe the client part in order, so the first thing we will consider is the creation of the menu, to be more precise - the implementation

of the folding / unfolding submenu. This fragment is very simple and requires little explanation, so we will describe only the logic of the code.

```
<Button
    android:id="@+id/alphabet"
    android:layout_width="@dimen/b_size"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="@string/b_alphabet"
    android:visibility="gone"
    android:onClick="Alphabet"/>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

<Button
    android:id="@+id/english1"
    android:layout_width="@dimen/b_size2"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="@string/b_english"
    android:visibility="gone"
    android:onClick="TestingWords"/>

</LinearLayout>
```

As you can see from the xml-file code snippet, initially all the submenu items are hidden, then, when you click on the corresponding button, they become visible, and the other items that were active are minimized, one of the functions is presented below, the rest are practically no different.

Next, you need to consider adding the word to the database on the server and on the smartphone. All words that the user added to himself are stored in the TreeMap <String, TransRus> class, the key for this class is the English word, and the

transcription and translation are stored in the TransRus class. You also need to add a TextView, which stores the value of the first letter of the word, different from the first letter of the next word.

Server application. The server application is multithreaded, uses the MySQL database and transmits data using the TCP protocol. The ServerOneJabber class is responsible for accepting / sending data. It inherits the Thread class.

```
public ServerOneJabber(Socket s) throws IOException, SQLException {  
    socket = s;  
    in = new BufferedReader(new InputStreamReader(socket.getInputStream()));  
    out = new PrintWriter(new BufferedWriter(new OutputStreamWriter (socket.  
getOutputStream())), true);  
    sqldb = new SqlDB(); start(); }  
}
```

The constructor is nothing complicated, you just need to pay attention to the sqldb object, which is responsible for communication with the database.

Application Description. As can be seen from Figure 1, when you click on the "Check words" or "Words" button, a submenu appears, so to speak. They were marked with the special "+" symbol, and when opened, the symbol is replaced by "-". When you click on one of the items in the expanded menu, it collapses to its original position. Before you change and delete words, you need to add them, therefore, we must consider this mechanism.

The program also has functions for changing and deleting words. Everything is simple here: after clicking on the desired menu item, a window opens in front of us where the words are arranged in alphabetical order. The windows for deleting and changing words are practically no different, just the icon on the button with the word, therefore, they used one xml file to create them. If the user changes the word, then the xml-file for adding words is used to change it, only with different functionality, all three fields are filled with old values.

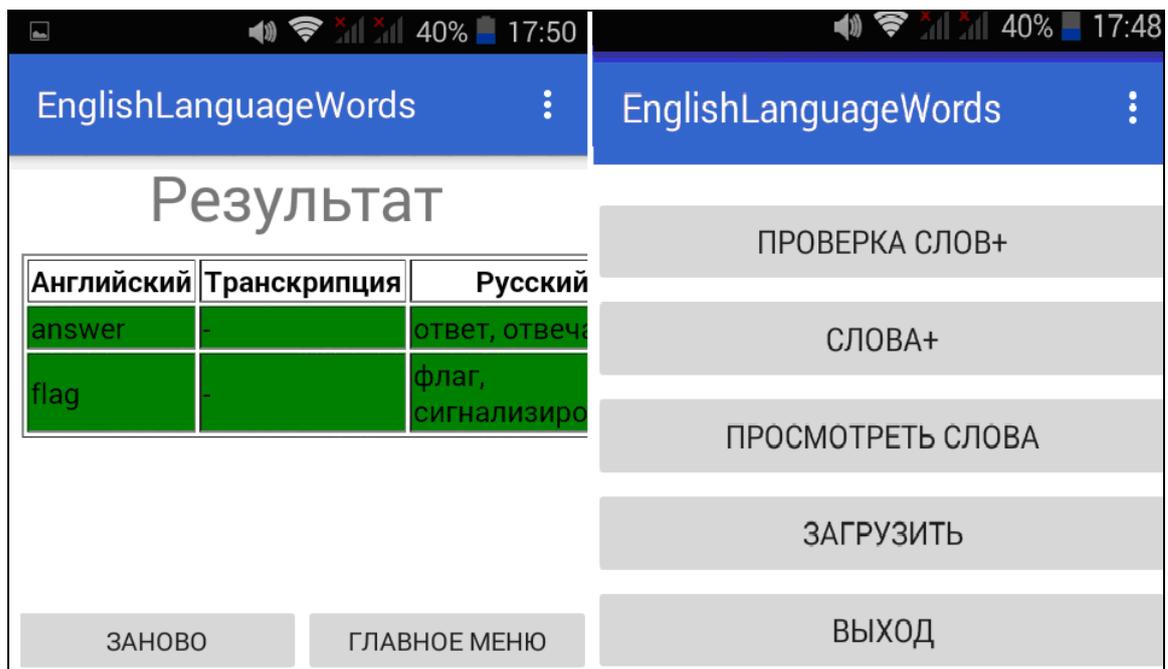


Figure 1 - General view of the application

After entering new values, when you click on the corresponding button, the word is changed and saved on the user's device without making changes to the database on the server. You can delete all words at once, or you can delete each word individually, but before each deletion, a confirmation dialog will pop up.

The created mobile application in Java on the Android OS platform was integrated with the database of the terminology dictionary on information and communication technologies in three (Kazakh, English, Russian) languages. A graphical user interface has been developed on the Android OS platform and an algorithm has been developed that allows us to calculate input terms based on a database of terminology dictionary. The mobile application is developed using the principle of separating the program code of the application logic from the program code of the application user interface. Qualitative and quantitative criteria have been developed by which the mobile application has been tested.

Conclusions. In this work, research and development of programming methods for mobile applications in the language, development of a graphical user interface on the Android OS platform, as well as the creation of a database of terminology dictionary on information and communication technologies were carried out. The

principles of object-oriented programming and elements of the algorithmic theory were also applied in the development of algorithm models.

This mobile application can be used by students of different settings and configurations. The use of affordable equipment to create a system allows you to use it even in low-cost installations, which significantly expands the circle of potential users of the system. In addition, the results can be useful to developers of virtual environment settings for creating their own terminological dictionaries and for their linguistic analysis for comparison.

References

1. <https://ru.wikipedia.org/wiki/Java>
2. Herbert Schildt. Java 8. Complete guide, 9th edition = Java 8. The Complete Reference, 9th Edition. -M.: « Williams », 2015. -1376 p.