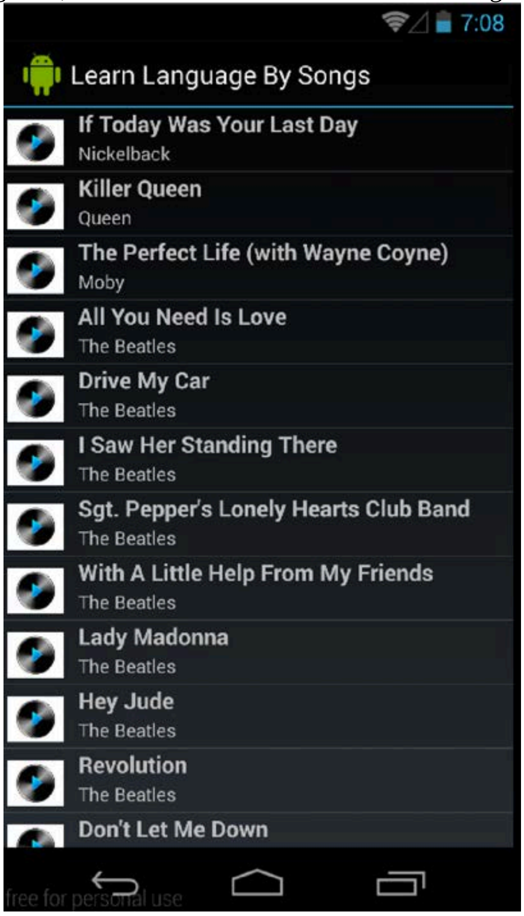
FRONT-END DEVELOPMENT OF THE APPLICATION "LEARN LANGUAGE BY SONGS"

Leontev P.N., Sannikov M.A.
Scientific supervisor: Zvigintsev I.L., teaching assistant
Language supervisor: Pichugova I.L., senior teacher
Tomsk Polytechnic University, 30, Lenin Avenue, Tomsk, 634050, Russia
E-mail: leontev.petr@gmail.com

The knowledge of foreign languages is very popular today. People want to learn languages faster, but it is too hard to find the most convenient and simple method for that. The article is focused on the development of the application Learn Language By Songs (LLBS) which allows studying foreign languages by using songs in a particular language. Users can choose any composition, which is available on their devices, or simply enter the desired name, and then the application will load both lyrics and audio-file via the Internet.

In the launch window, which is represented in Figure 1, user can select one of the available songs.



7:08

Learn Language By Songs

If Today Was Your Last Day
Nickelback

Killer Queen
Queen

The Perfect Life (with Wayne Coyne)
Moby

All You Need Is Love
The Beatles

Drive My Car
The Beatles

I Saw Her Standing There
The Beatles

Sgt. Pepper's Lonely Hearts Club Band
The Beatles

With A Little Help From My Friends
The Beatles

Lady Madonna
The Beatles

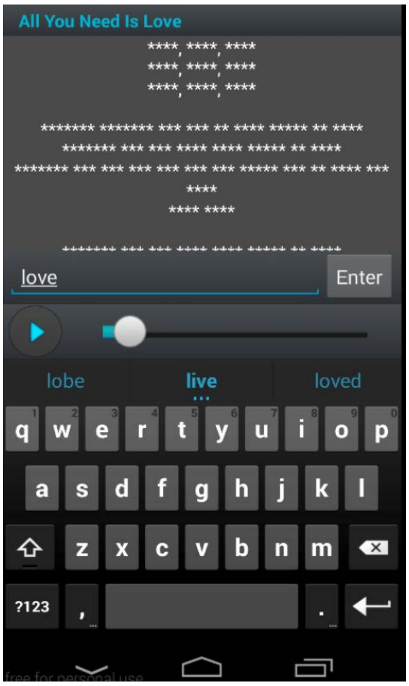
Hey Jude
The Beatles

Revolution
The Beatles

Don't Let Me Down

Fig. 1. Application launch window

After that the main window (Figure 2) appears. It contains song title at the top, song lyrics in the middle of the window and playback control panel and input field at the bottom. All characters of the song text are replaced with asterisks (*). Then we can start to listen to the song and enter words.



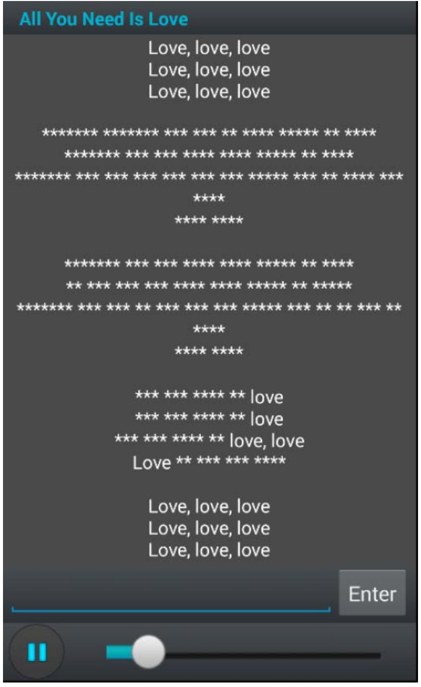
All You Need Is Love

```
****, ****, ****
****, ****, ****
****, ****, ****

******* ******* *** *** ** **** ***** ** ****
******* *** *** **** **** ***** ** ****
******* *** *** *** *** *** *** ***** *** ** **** ***
****
**** ****

******* *** *** **** **** ***** ** ****
```

love

Fig. 2. Application main window

If the user enters the word contained in the song lyrics, all its occurrences will be instantly displayed (Figure 3).



All You Need Is Love

```
Love, love, love
Love, love, love
Love, love, love

******* ******* *** *** ** **** ***** ** ****
******* *** *** **** **** ***** ** ****
******* *** *** *** *** *** *** ***** *** ** **** ***
****
**** ****

******* *** *** **** **** ***** ** ****
** *** *** *** **** **** ***** ** *****
******* *** *** ** *** *** *** ***** *** ** ** *** **
****
**** ****

*** *** **** ** love
*** *** **** ** love
*** *** **** ** love, love
Love ** *** *** ****

Love, love, love
Love, love, love
Love, love, love
```

Fig. 3. Application main windows after entering the word «love»

The application is designed as client-server which interacts by using HTTP. It is represented in Figure 4.

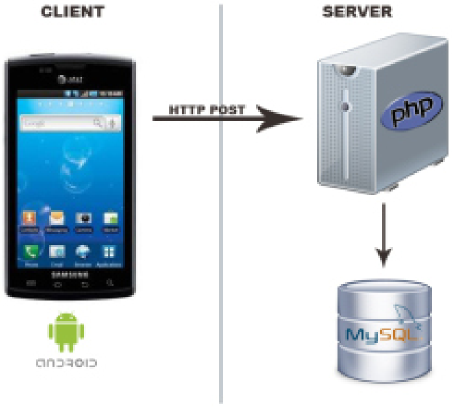


Fig. 4. Client-server model

It usually means a distributed application that partitions tasks between providers of a resource or service, called servers, and service requesters, called clients [2]. In our case, client or front-end is the application that works under popular mobile platforms, such as Android and iOS. The application is written in Java and ObjectiveC languages. Server or back-end has API written with PHP-framework Symphony. It is selected with the purpose to speed up the creation and maintenance of server and to replace repetitive coding tasks [3]. It is much easier to write the API-server once, and use it from different platforms, than implement similar functionality every time. The MySQL databases are used to cache song lyrics and for user authorization/registration.

This architecture is convenient and practical in the view of the fact that it is possible not only to reduce the size of the client, but also to use it only as a view showing the data which is necessary for user at a given moment. Furthermore, the size of the client and functions performed by it can be critical factors while installing it on a device with low-powered hardware.

To use all the possibilities provided by the server, client can make use of Application Programming Interface (API) which usually comes in the form of a library that includes specifications for routines, data structures, object classes, and variables and allows performing the required action quickly [4]. In more detail, to call any API method it is necessary to set not only its name but also mandatory parameters characterizing the object about which the client wants to get information from the server.

Server uses the caching technology which allows fast processing of incoming user requests in the case if they were received previously and so it seriously reduces the runtime of such operations. For example, taking into consideration compositions rating it is always possible to make the cache of song's lyrics which are the most requested by the clients.

Given the fact that almost all people have account in a social network, for instance, Vkontakte, Facebook, Twitter, etc., the application provides the possibility to interact with it. In particular, there are popular actions «tell friends» and «share results». Nowadays, it is relatively straightforward to implement such support since every known service whether it is social network or team blog has its own API whereby developers can extend the functionality of their own applications as necessary.

More generally, LLBS application is a kind of integration platform for the services and no matter what exactly the size of each platform is because the final result for user is always represented as a whole.

Currently, the application development for mobile operating systems such as Android and iOS is one of the most accessible directions due to the fact that there are a lot of devices on the market operating under the control of these systems [5]..

In the future we plan to develop our application in the following directions:

- personalized search – selection of the compositions based on user requirement (duration, language, music genre, etc.)
- learning to rank algorithms – songs sorted by sound complexity (i.e. susceptibility of the words by the person for whom the language used in the composition is not native);
- audio-files recognition – finding all available information about the composition based on special analysis of recorded part (use of time-frequency analysis techniques);
- free- and premium- versions of the application - providing different functionality depending on user requirements.

References

1. Android for developers [Electronic resource], Access mode: https://developer.android.com – free.
2. Client-server model [Electronic resource]. Access mode: http://en.wikipedia.org/wiki/Client-server_model – free.
3. High Performance PHP framework for web-development – Symphony [Electronic resource]. Access mode: http://symfony.com/ – free.
4. Application Programming Interface [Electronic resource]. Access mode: http://en.wikipedia.org/wiki/Application_programming_interface – free.
5. Mobile Development overview [Electronic resource]. Access mode: http://www.clarity-ventures.com/articles/article/676/mobile-development-overview-statistics-on-mobile-development-mobile-development – free.