Design of an Android Application using a Tool for Decision-making Support

Jan Bohacik

***Abstract*—**These days people around the world often need data about the current weather and weather forecast for the following days. They would like to access this data easily and anywhere. These requirements can be solved with the development of an Android application running on a smartphone. In this paper, a developed Android application for the current weather and weather forecast and its testing are described. This application establishes communication with a data server for the use of the XML data from the OpenWeatherMap API service, parses the data and analyses it. It shows the current weather and a 5-day forecast for a chosen location with updates every 3 hours. The XML data obtained from the data server is parsed either with SAX or with DOM. Experimental results with both of these algorithms suggest that the DOM algorithm is a lot faster than the SAX algorithm.

***Keywords*—**Android application, decision-making support, OpenWeatherMap, XML parsing.

I. Introduction

Weather has something to do with the atmosphere at some particular time and place and it is experienced by people through their senses [9]. It effects people significantly. For example, people usually choose their clothes on the basis of weather. They may also change how they spend their time in relation to weather. Freezing can harm potato crops and cause a price rise. Globally, the weather may be good at some places and bad in other places at the same time, and it is described by various measures such as air temperature, air pressure, wind speed and wind direction, humidity, cloud amount and rainfall, and so on [7]. Air temperature quantifies the energy of motion of the gas molecules of the air and it is mostly influenced by the radiation from the sun and the radiation from Earth. Air pressure represents the weight per unit of area of a column of air reaching to the top of the atmosphere of Earth. Wind is related to the movements of air from the areas with high air pressure to the areas with low air pressure in some direction and at some speed. Humidity measures how much water vapor is in the air. Clouds are small ice crystals developed from water vapor during saturation. Rainfall is the amount of falling droplets which have condensed from atmospheric water vapor. Since the influence of weather on people is significant, it is important to know the current weather and weather forecast. The current weather is observed from various sources such as land-based observation stations, radars, aircrafts, balloons, ships, satellites and radio sounds [8]. Weather forecast reflects the expected change in the current state of the atmosphere and it is based on the use of the observations from those sources in miscellaneous weather forecast models.

For the purposes of decision support, it is important to have the current weather and weather forecast available all the time. This can be achieved with mobile devices such as tablets and smartphones since they are light-weight enough to be taken anywhere and at any time. The most popular operating system in these devices is Android [2] and so it is sensible to start developing the availability of the current weather and weather forecast for Android at first. Android is an operating system based on the Linux kernel with good functionality for especially touchscreen mobile devices [10]. It contains middleware, libraries, APIS and an application Java-compatible framework on top of the Linux kernel. It is developed by a consortium known as Open Handset Alliance which is composed of many companies including Google and Samsung. A developed Android application for the current weather and weather forecast and its testing are described. In general, Android applications are usually written with the use of the Android software

J. Bohacik, Faculty of Management Science and Informatics, University of Zilina, Zilina, Slovakia (e-mail: jan.bohacik@fri.uniza.sk).

development kit and Java which can be combined with C or C++ [6]. The Go programming language and the Kotlin programming language are also supported broadly.

The paper is organized in the following way. The user requirements for the Android application giving the current weather and weather forecast and its development are analyzed in detail in Section II. In Section III, the testing of the developed application and obtained experimental results related to parsing are described. The paper is concluded in Section IV.

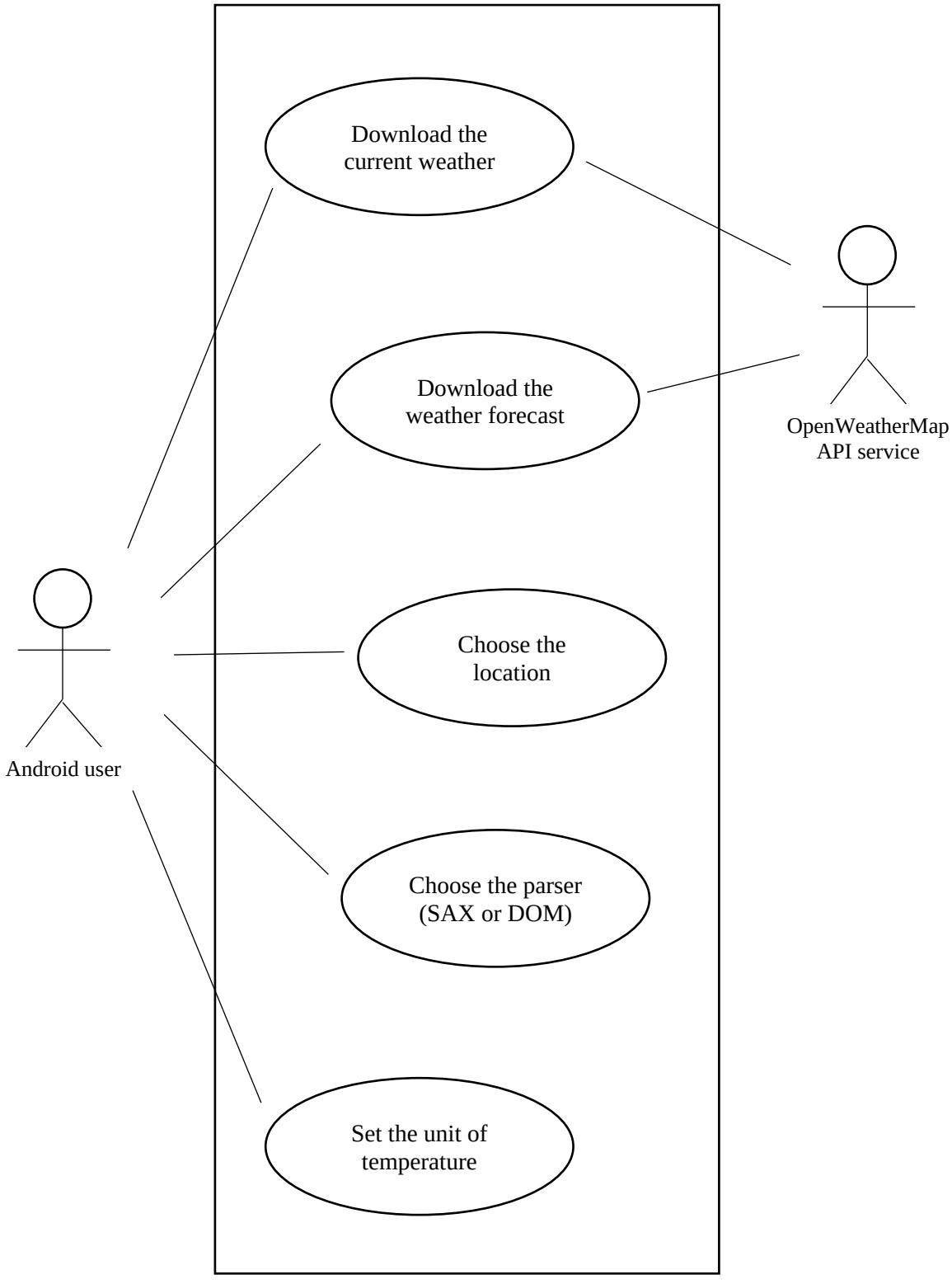

Fig. 1 Use case diagram

II. User requirements for the Android application and its development

It is important to know the representative user of the Android application for the current weather and weather forecast. The application should easily provide relevant information about the current weather and the weather forecast for the following days with minimal processor load of an employed mobile device. And so, the user should be able to download weather information for a particular location. In addition, the user should be able to use the application in a place without the internet on the basis of previously downloaded weather information. Various users may be used to various units of temperature and so the user should be able to choose if the Celsius scale or the Fahrenheit scale are displayed. The use case diagram of the application is shown in Fig. 1 where the utilization of OpenWeatherMap service is also shown.

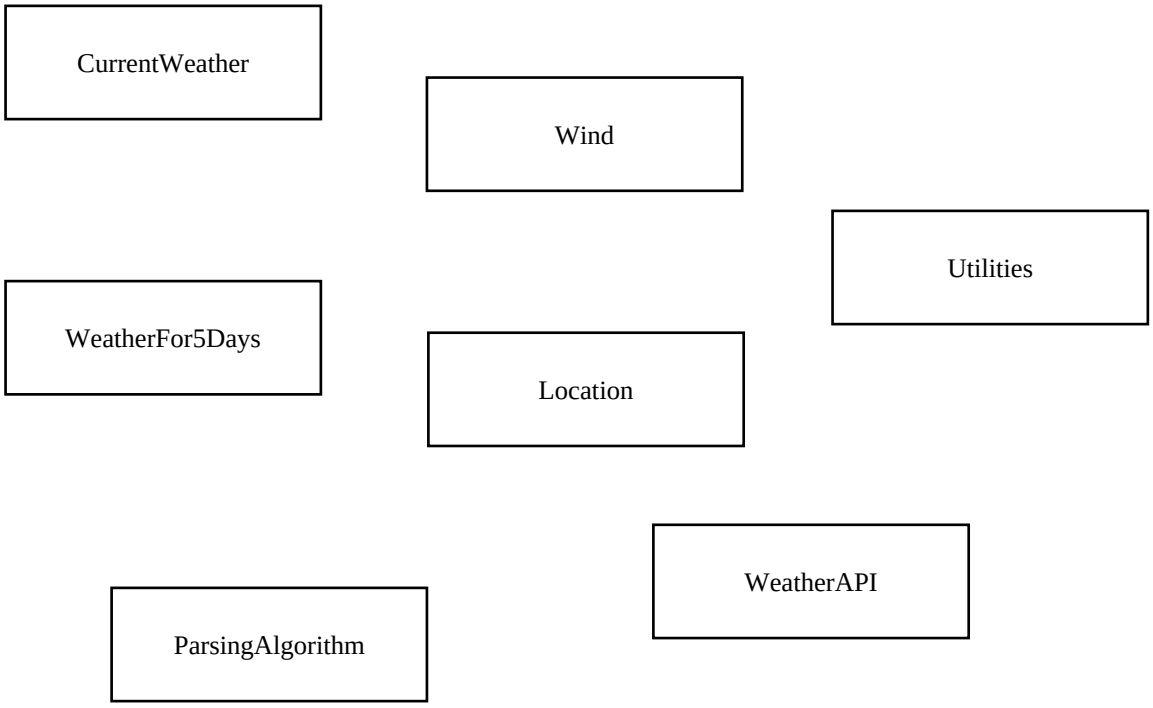

Fig. 2 Classes

The classes which are used in the implementation of the Android application for the current weather and weather forecast are shown in Fig. 2. Information about the weather is stored in CurrentWeather, WeatherForecastFor5Days and Wind. WeatherAPI provides the API which is used for collecting the information on the weather. Class ParsingAlgorithm allows the processing of the information on the weather with a parser. Class Utilities contains utilities for gaining information about a chosen location, storing downloaded information, and so on.

There are several Android versions used on the market at the same time since new Android versions regularly bring changes and improvements [6]. Therefore, the Android application needs to be configured for a target minimal API level for which it works without issues. When a recent API level is targeted, the latest features of the Android operating system are available. Lower API levels allow the application to be used on more mobile devices on the market. API level 15 is chosen for the Android application for the current weather and weather forecast. The application is developed in Android Studio which is an official IDE for the Android operating system and which is built on IntelliJ IDEA from JetBrains [6]. Android Studio can be used for the development of Andorid applications on ChromeOS, Linux (with GNOME or KDE desktop), macOS (version 10.10 or higher) and Windows (version 7 or higher). It needs at least 4 GB of RAM and 2 GB of available disk space. Android Studio 2.0 is used for the development.

Fig. 3 Main window [3]

The main window of the application contains information about the current weather as it can be seen in Fig. 3. It provides the location, temperature, wind speed, cloudiness, humidity and air pressure. It also has a button allowing to show the weather forecast for the following 5 days. In the top-left corner, the menu for accessing all other things of the application can be found.

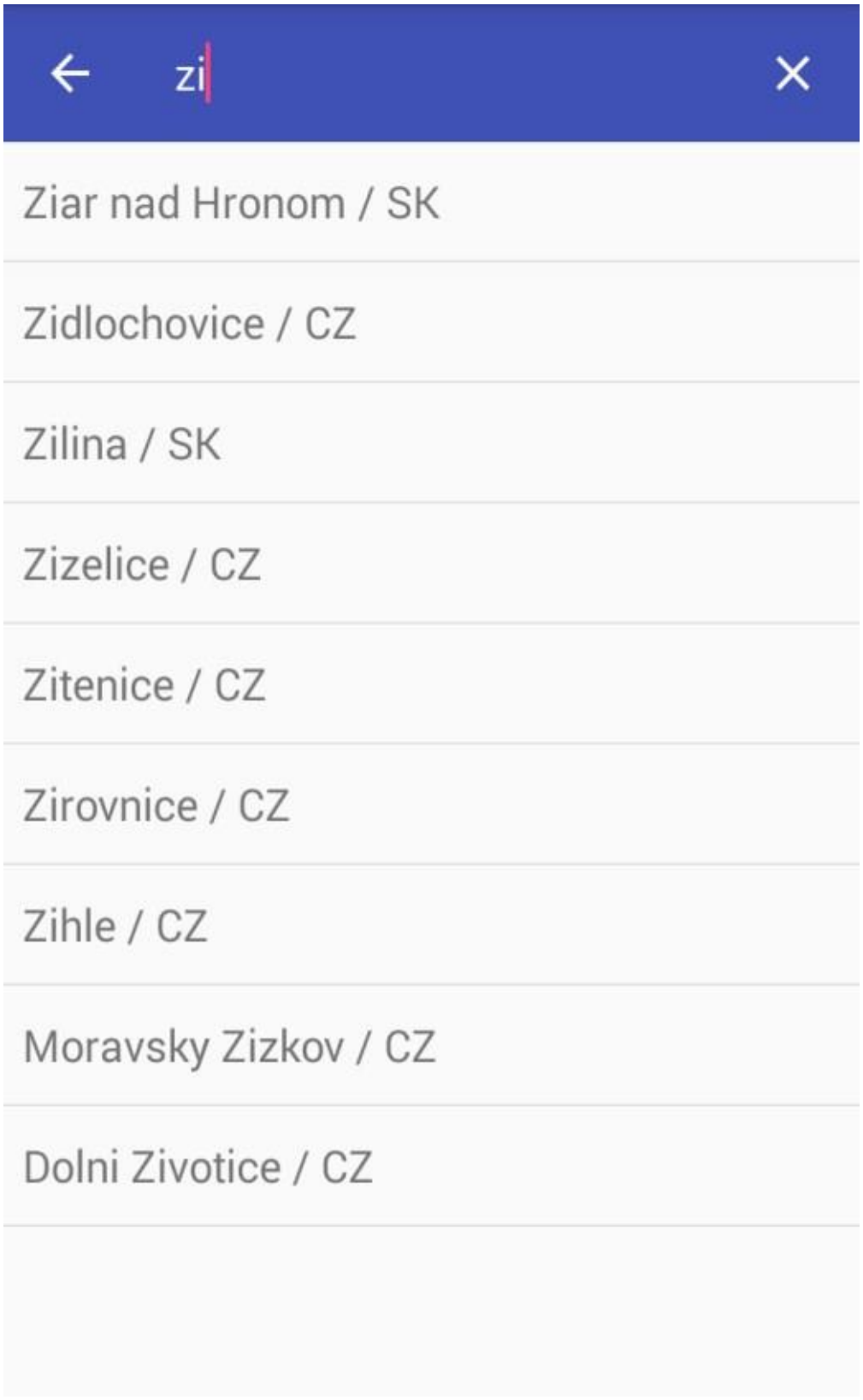

Fig. 4 Choice of the location [3]

The user can change the location for the current weather and weather forecast as it can be seen in Fig. 4. This can be activated by the user with choosing the plus sign in the top-right corner. It shows many locations with available information about the weather. These locations can be searched and this search can be sped up with writing a part of the preferred location. The written part causes that only the places with this part are displayed in the application.

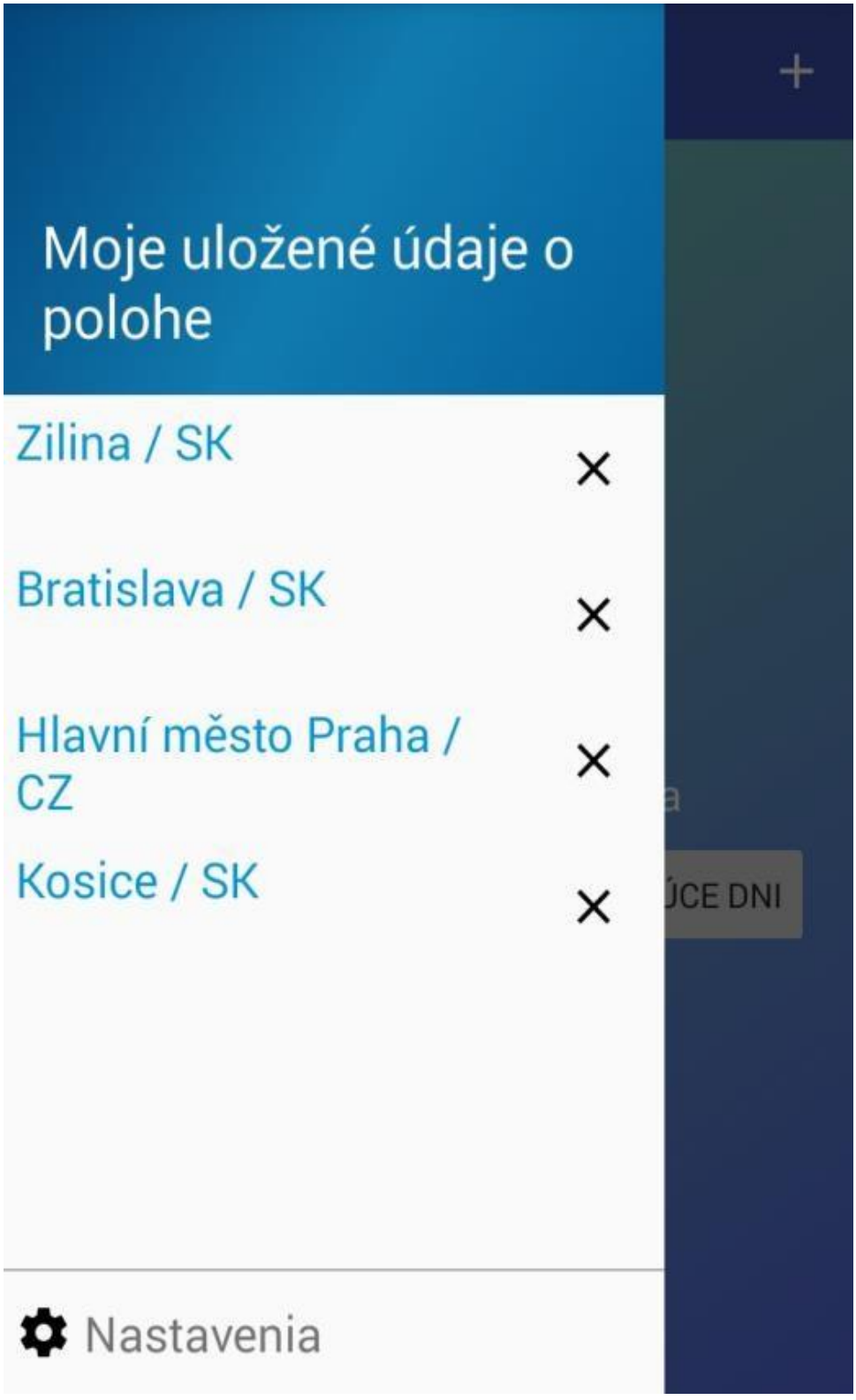

Fig. 5 Menu of the application [3]

The menu of the application is shown in Fig. 5. It contains a list of previously chosen locations. Any of the shown locations in the list can be chosen again and a location from the list can be removed with the use of the X mark. The menu also contains the Settings button which displays the settings of the application. These settings are shown in Fig. 6.

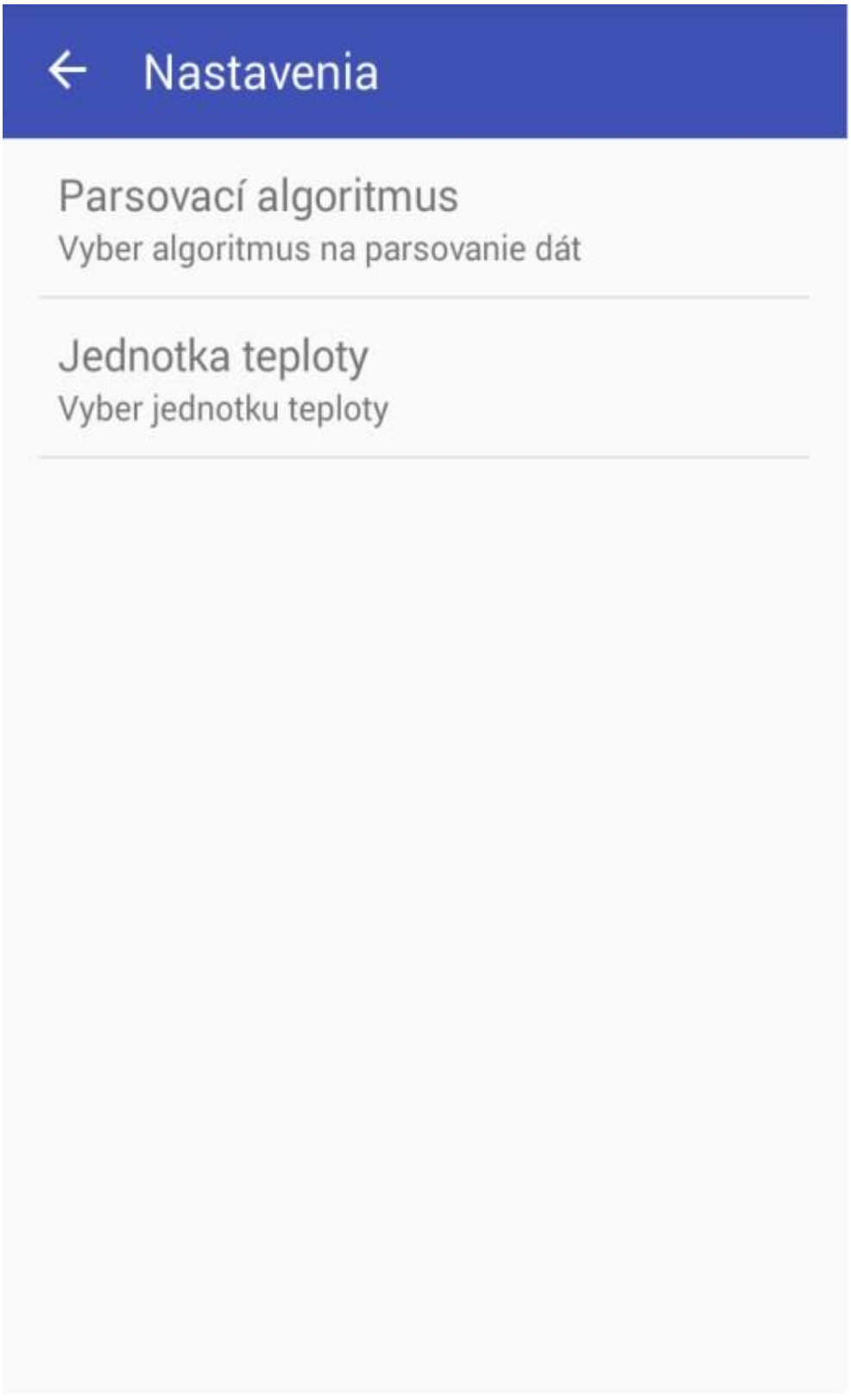

Fig. 6 Available settings of the application [3]

As it is shown in Fig. 6, the user can choose the algorithm used for parsing the downloaded information about the weather (SAX or DOM) and the unit of temperature (Celsius or Fahrenheit). The downloaded weather information from the OpenWeatherMap API service is in the XML file format. Parsing an XML file is necessary for the accession of its content and structure and it may decrease its processing performance [4]. When the SAX algorithm is used for parsing, the whole document is read and some particular event is executed without placing the whole document into the memory [12]. It does not allow any random access to the XML document since applications using it receive notifications in sequential order from the top to the bottom. The DOM algorithm employs a tree which is created on the basis of the XML file and which is placed in the memory of the computer [11]. Thanks to the tree in the memory, the contents and the structure of the XML file can be examined with a random access.

III. TESTING AND RESULTS

Since there are many versions of the Android operating system used on the market at the same time and these Android versions also run on devices with various hardware specifications, testing applications created for Android may prove to be complicated. However, developers of these applications can use an emulator of a device and a particular version of its operating system for testing if they do not have some device. Android Emulator employed in the development of the application for the current weather and weather forecast was configured with an Android Virtual Device which contained a hardware profile, storage area, system image, and so on [5]. During the development, the application was tested on a virtual device emulating Samsung Galaxy Nexus I9250 with Android 4.4.2, ARM processor armeabi-v7a and 1 GB of RAM. The developed application was tested on the following two real mobile devices: 1) Nvidia Shield K1 with Android 6.0, 2.2 GHz quad-core processor and 2 GB of RAM; and 2) Sony Xperia Z5 with Android 6.0, 2 GHz octa-core processor and 3 GB of RAM. The application worked without issues.

Table 1 Speed of the algorithms for parsing in milliseconds [1], [3]

Experiment	**The algorithm for parsing**	
	SAX	**DOM**
Number 1	50	403
Number 2	52	378
Number 3	63	493
Number 4	83	431
Number 5	65	473
Number 6	82	363
Number 7	92	325
Number 8	66	443
Number 9	72	387
Number 10	94	382

Number 11	63	333
Number 12	69	452
Number 13	107	414
Number 14	84	492
Number 15	87	496
Number 16	66	452
Number 17	45	498
Number 18	71	355
Number 19	68	431
Number 20	60	386
Minimum	45	325
Maximum	107	498
Average	71.95	419.35

The speed of the algorithms for parsing downloaded XML files from the OpenWeatherMap API service was tested as well. The obtained results are presented in Table 1. The XML files had been downloaded in advance so that the Internet connection did not have an impact on the results. The XML files had also been downloaded 20 times in various moments so that the experiments are more relevant statistically. As it can be seen in Table 1, the SAX algorithm with 71.95 milliseconds on average is faster than the DOM algorithm with 419.35 milliseconds. It is interesting to point out that even the maximal time of 107 milliseconds for the SAX algorithm is much lower than the minimal time of 325 milliseconds for the DOM algorithm.

IV. Conclusions

A developed Android application for the current weather and weather forecast and its testing were described. This application communicates with a data server so that the XML files from the OpenWeatherMap API service are downloaded into the application. The files are parsed with the SAX algorithm or the DOM algorithm and used in the application for displaying the weather. A screenshot of its main window, a screenshot for the choice of the location in the application, a screenshot of the menu, and a screenshot of available settings were discussed. In the experiments described in the paper, the SAX algorithm parsed the XML data for 71.95 milliseconds on average, which was much faster than the average speed of the DOM algorithm. The functionality of the application was tested with a virtual device emulating Samsung Galaxy Nexus I9250 and with real devices Nvidia Shield K1 and Sony Xperia Z5 successfully.

Acknowledgment

This publication was realized with the support of the “Operational Program Research and Innovation in the frame of the project: ICT for smart society, code ITMS2014 +: 313011T462, co-financed by the European Regional Development Fund”.

REFERENCES

[1] J. Bohacik, "A decision support tool for current weather and weather forecast using the Android system," *Journal of Information Technologies*, vol. 9, no. 2, pp. 1-7, 2016.

[2] M. Casserly, "iPhone vs Android market share," *Macworld*, retrieved from https://www.macworld.co.uk/feature/iphone/iphone-vs-android-market-share-3691861/ on 20 November 2019.

[3] K. Chudjak, *Implementation of a tool for decision support with the Android system*. Bachelor's thesis. Zilina, Slovakia: University of Zilina, 2016.

[4] V. M. Deshmukh, G. R. Bamnote, "An empirical study of XML parsers across applications," in *Proceedings of the International Conference on Computing Communication Control and Automation*, pp. 396-401, 2015.

[5] Google Developers, *Create and manage virtual devices*. Retrieved from https://developer.android.com/studio/run/managing-avds on 25 September 2019.

[6] Google Developers, *Documentation for app developers*. Retrieved from https://developer.android.com/ docs on 12 September 2019.

[7] NASA, "Weather," *World Book at NASA*, retrieved from https://web.archive.org/web/20101213184908/http://www.nasa.gov/worldbook/weather_worldbook.html on 19 November 2019.

[8] A. K. Pandey, C. P. Agrawal, M. Agrawal, "A hadoop based weather prediction model for classification of weather data," in *International Conference on Electrical, Computer and Communication Technologies*, 2017, INSPEC accession number 17393538.

[9] N. Peterson, K. Broad, "Climate and weather discourse in anthropology: from determinism to uncertain futures," *Anthropology and Climate Change: From Encounters to Actions* edited by S. Crate and M. Nuttall and published by Left Coast Press, pp. 70–86, 2009.

[10] G.-H. Shao, "Application development research based on Android platform," in *International Conference on Intelligent Computation Technology and Automation*, pp. 579-582, 2014.

[11] Tutorialspoint, Java DOM Parser - Overview, *Tutorialpoint*, retrieved from http://www.tutorialspoint.com/java_xml/java_dom_parser.htm on 16 November 2019.

[12] Tutorialspoint, Java SAX Parser - Overview, *Tutorialspoint*, retrieved from http://www.tutorialspoint.com/java_xml/java_sax_parser.htm on 17 November 2019.