The Sixth International Conference on e-Learning (eLearning-2015), 24-25 September 2015, Belgrade, Serbia

E-LEARNING TOOL FOR MATHEMATICS – THE MATHIST

MARKO PAVLOVIĆ
Student at the University of Niš, Faculty of Electronic Engineering, marko@themathist.com

MILENA MIŠIĆ
Student at the University of Niš, Faculty of Electronic Engineering, milena@themathist.com

FILIP PAVLOVIĆ
Student at the University of Niš, Faculty of Electronic Engineering, filip@themathist.com

Abstract: *As interest in e-learning technologies grows and as there are more and more students joining the online classrooms, there is a growing need for specialized tools, which can improve the learning experience. One of the key aspects of successful e-learning is good communication, and while it is well established in various areas, some, such as mathematics, still experience problems. In this paper, we present "The Mathist" (www.themathist.com), which we have developed during our studies. It enables efficient writing of mathematical formulas and notes in an online environment, without the knowledge of particular language or special notation, such as TeX/LaTeX. By storing notes in the cloud, it also enables sharing work and collaboration among students and professors.*

Keywords: *E-learning, mathematics education*

1. INTRODUCTION

While a lot of interactive material is available nowadays for helping people with learning mathematics (such as video lectures, variety of games, etc), a problem still exists when one tries to write mathematics on a smart device. That can be considered a major drawback in varieties of related software. Some solutions are available but not convenient for all types of users, for example, students, especially younger ones. The time required to learn the syntax of LaTeX, Mathematica or any similar solution, or the time required to type a simple formula in systems without adequate syntax for mathematics, such as MS Word, present challenges in e-learning that are difficult to overcome.

Nowadays, LaTeX, as one of most influential typesetting systems, is a standard for scientific documents (1). While it is very powerful, it requires its user to possess at least basic knowledge of the syntax. MathML, being "low-level specification on mathematical and scientific content on the Web" (2), also requires a customized syntax relying on XML to be learned first. Even then, it is suitable only to those who create websites and is not convenient for a student to use it in class, for instance. An example of high quality software working online is MathJax with partners such as American Mathematical Society, IEEE, Elsevier, to mention but a few. It is capable of rendering typeset mathematic formulas in a browser, but still, the input has to be in a form of TeX, MathML or ASCIImath (3).

What a good software for writing mathematics should provide, can be defined as follows: it should enable intuitive WYSIWYG writing and it should cover all symbols and types of formulas in common use today. These two conditions may seem to exclude each other. Nevertheless, they do represent completely valid goals. Most of the systems previously mentioned focus on the second: you can write anything and apply any formatting, you can use it further to do calculations, plot graphs or perform some kind of analysis. On the other hand, the first goal is also very significant and that is what inspired us to develop The Mathist as a software which aims to achieve speed in writing in order to make it useful for classrooms along with keeping it flexible and complete enough to be really useful.

2. KEY FEATURES

The Mathist is a web application available at www.themathist.com. Since "the most important advantage of e-learning when compared to other traditional methods is that it significantly decreases the costs" as stated in (4), The Mathist is free of charge. As such, it can be conveniently accessed by anyone interested in mathematics or related fields, from anywhere in the world and at all times.

The Mathist works in all major browsers, regardless of the operating system or platform. It relies on Google Drive for storing created documents and requires a Google account for that purpose.

The key advantages of The Mathist are its customized keyboard and the auto complete feature.

Customized keyboard

The Mathist provides a customized keyboard modelled on top of the standard QWERTY layout. This reduces the time needed for adaptation as much as possible. In addition to standard QWERTY part of it, the keyboard has two additional rows on top with the most commonly used mathematical symbols: brackets, fractions, root, percentage, plus, minus, numbers, etc. The figure 1 shows the keyboard.

The four arrows at the bottom right corner are used for navigation through the elements of a formula. The button at the bottom left corner is used for changing the language of the keyboard: a user can choose between the standard Latin and Greek keyboard, chosen for its frequent use, and their uppercase and lowercase variants.

Long press on certain buttons shows additional options. For instance, when the user presses the brackets button, different types of brackets are shown.

This on-screen keyboard shows when the user chooses to type a formula by clicking the *f(x)* button placed in the bottom right corner of a note. If the user, on the other hand, chooses to type some plain text (an explanation, for example) by clicking the *Text* button, a standard, platform specific keyboard will be used. For example, on a desktop computer, the user would use the physical keyboard and on a touch screen device, a standard operating system keyboard will be shown, and the customized keyboard, The Mathist provides, will be hidden. This means the plain text can be entered in the native language of the user.

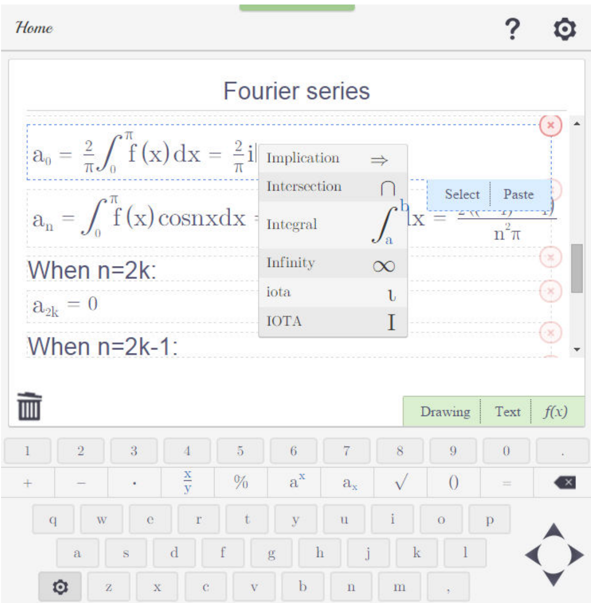

Figure 1: The Mathist editor with customized keyboard and list of suggestion

Auto complete feature

Although the customized keyboard provides some additional options and enhances the typing experience, it can be, by no means, considered sufficient. This is because there are many types of mathematical expressions that can be written, and the keyboard can only display so many buttons, before it becomes difficult to use. That is why we created the auto complete feature, which is used for writing the mathematical expressions and symbols, that are not assigned a specific button on the customized keyboard. This feature is triggered when the user starts typing a formula, and has many similarities with the auto complete feature found in traditional code editors. It works by displaying a list of suggestions, based on the letters the user is currently typing. The time between the last typed letter and appearance of suggestions is calculated from user's average writing speed. Thus the feature is not intrusive but helpful.

Each mathematical expression or symbol in The Mathist has a unique name assigned to it, which aligns with the names commonly used in mathematics, for example: limit, integral, fraction, root, pi, product, etc. When the user starts typing, the whole registry of mathematical expressions and symbols is searched for matching names, and if there are any, they are listed in the auto complete box, which appears beneath the cursor. The user can then choose one of the suggested options, or may proceed with typing if none of the options is suitable. Figure 1 illustrates this using integral as an example.

3. A TOOL FOR COLLABORATION

The Mathist requires a Google account in order to be used efficiently. This is because all notes made with The Mathist are stored on the user's Google Drive under the extension *tmn (The Mathist Note)*. Using standard Google Drive features, notes can easily be shared with or edited by people of user's choice. Since the data is stored in the cloud, notes can be accessed from any device and from anywhere in the world.

Collaboration and interaction between students and professors is the essence of successful e-learning. A wide range of devices with Internet access available today contribute to this goal: "schools worldwide have started to adopt policies that allow students to bring their own mobile devices to school, a trend known as Bring your Own Device, or embark on initiatives that equip students with mobile devices as a way of expanding equitable access to educational opportunities" (5).

Through The Mathist, students can share their solutions for particular problems, teachers can check homework, provide examples or give complete lectures accessible publicly or by a group of people. Sharing notes can be managed in such a way that the public can only see them, but not edit, or editing might be allowed.

In addition to writing and sharing notes, the user can print notes directly from the browser, using predefined options. The printed version of the document will only contain the note body itself, without other elements that appear in the window (keyboard, menu, etc).

4. WOLFRAM|ALPHA AND THE MATHIST

Wolfram|Alpha introduces searching for information by doing "dynamic computations based on vast collections of built-in data, algorithms and methods" as stated in (6). The results provided contain not only simple facts about the world, but also conclusions that follow, such as solutions to mathematical problems.

The Mathist uses Wolfram|Alpha's public API to provide additional information, perform calculations, plot graphs and solve problems per user's request.

When the user enters some mathematical formula and presses the Select button, a light bulb appears. Clicking on the light bulb activates the Wolfram|Alpha feature, and the selected formula is translated to a syntax that Wolfram|Alpha can understand. That translation is sent as a search query to Wolfram|Alpha and the results are then received and adapted for the note. A dialog box opens offering the user to choose between possible types of received results: alternative forms of the formula, graphs or solutions of integrals, limits or equations, etc. The user can choose one or multiple options and as each is chosen, it appears in the note. An example is shown in figure 2.

This can be very useful as a helping option when studying and can be used for checking results or visualizing a solution through a graph. On the other hand, students might misuse it and instead of doing homework, they could just use Wolfram|Alpha to solve the all the given problems. That is why a teacher's' tool was introduced. The Mathist can highlight the parts of the note that have been generated by Wolfram|Alpha. This is beneficial in 2 ways, first any misuse of the Wolfram|Alpha tool will immediately be brought to the teacher's attention, and secondly, it provides the teacher with valuable information about which parts of the homework the student needed help with, and which parts of the assignment have proven to be too difficult for the particular student. This ensures The Mathist can be used for giving assignments and writing homework, since "assessment plays an important role in teaching activities and has significant effects on learning." (7).

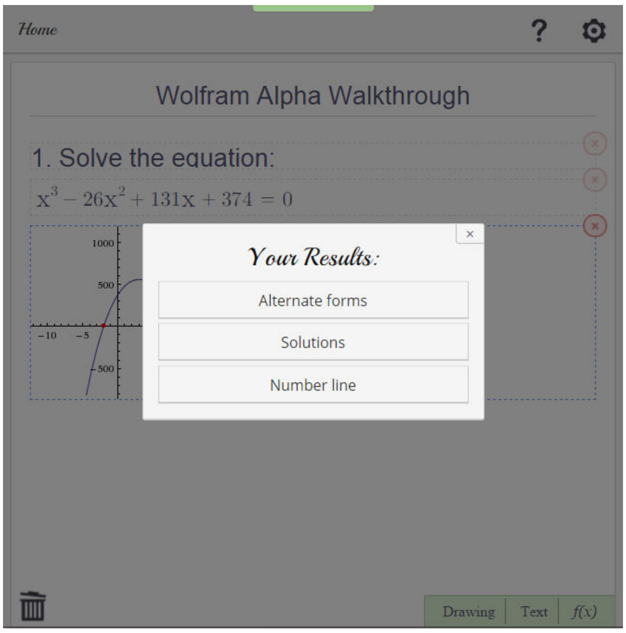

Figure 2: Editor with received Wolfram|Alpha results

5. CURRENT AND FUTURE USERS

At the moment of writing this paper, The Mathist has approximately 4400 registered users. They come from all over the world, but the majority is from the USA, as it is shown in figure 3.

A typical user is between 18 and 24 years old, English-speaking and fond of new technologies, also interested in science and mathematics.

The Mathist is also helpful to students with certain kinds of writing difficulties, such as spatial ordering or graphomotor problems. Although this aspect requires further research, The Mathist has received positive feedback from teachers who work with students with such difficulties.

Regarding the teachers and the usage of modern technologies, an interesting experience is described in (8): "the knowledge and readiness of teachers, who taught ICT subjects, of mobile learning were lower than teachers with mathematics, science, and other subjects."

"The success of mobile learning largely depends on the ability of teachers to maximize the educational advantages of mobile devices" (9). This is also applicable in a more general situation of e-learning and illustrates the importance of teachers' contribution in acceptance of any e-learning initiative.

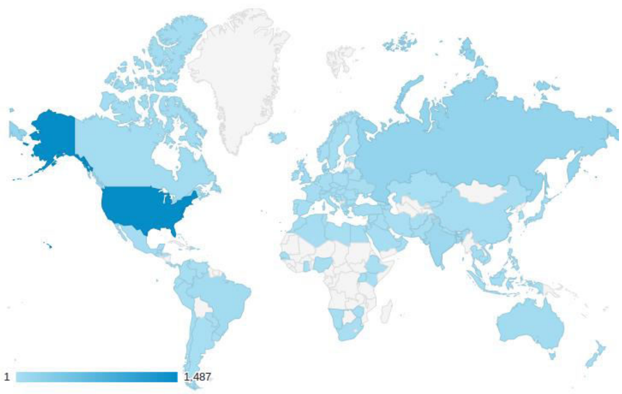

Figure 3: World map with currently registered users with numbers given by country

Users experiences

Ever since The Mathist was introduced to the public, experiences from users have had a great impact on further development. Mathematics and physics teachers from high schools mainly located in the US and the UK have provided us with valuable feedback. The most praised characteristic by those teachers is the speed of writing which The Mathist allows. They find it easy to use since it does not require pressing a lot of buttons while searching for adequate formula. It is not the case in other solutions and users found them quite time-consuming and distracting. Another feature which is considered a big plus is the integration with Google Drive, which is already transforming the way teachers can share lectures and assignments with students. That is, presumably, because of the Google for Education, a recent project launched by

Google which provides applications, books and devices for schools around the world (10).

Wolfram|Alpha feature is also very well received. Fast calculations and inclusion of different graphs are found very convenient, especially by physics teachers. On the other hand, the Wolfram|Alpha feature also presented a problem some teachers faced when students used it to do the homework for them. It was solved with the Wolfram|Alpha indicator which highlights results received thus pointing out which tasks students had problem with.

6. SYSTEM ARCHITECTURE

The front end of the application is written in HTML5, CSS3 and JavaScript. It also relies on the jQuery JavaScript library. Ruby on Rails framework and PostgreSQL are employed in the back end. The following paragraphs explain these choices in more detail.

Mathematical notation often consists of nested structures, which are not found in plain text writing. Apart from brackets, which are an obvious example, there are roots, fractions, integrals, limits, etc. Each of them can have other mathematical expressions in one of their sub fields. For example, the fraction notation is clearly defined by 2 separate fields called the numerator and the denominator, and a rule which says that the numerator is written above the denominator, separated by a horizontal line. The challenge is to efficiently model the nested structure and rules of mathematical notation, as in the example above, providing that the rules can be updated to accommodate the ever expanding list of supported mathematical expressions.

The solution was found in using the combination of HTML and CSS. Being derived from XML, HTML is perfectly suited for describing the hierarchical structures of mathematical expressions. CSS with its wide range of selectors is built for isolating parts of the HTML document and applying positioning and styling rules to it. If we describe the structure of the mathematical content with HTML, then CSS can be used to model the rules of mathematical notation. This ensures that the rules can be expanded or modified by changing only the CSS part of the application, while the HTML content can remain mostly the same.

JavaScript is used to enable interactive writing of mathematical notes, by implementing the logic, which processes individual events generated by the user's input. Structured as a modular application, The Mathist consists of around 25 individual JavaScript modules. Loose coupling is employed, and all the communication is handled by a central mediator component. It transmits messages between the modules taking into account that each may be subscribed to only a subset of all message types. When the user interacts with the application, a new message is created, which contains the relevant information about that interaction event. The message is then simultaneously broadcasted to all the subscribed modules. This ensures that the functionality of a particular module is strictly defined and encapsulated, while allowing it to freely subscribe to new message types in the future.

An example of this interaction is shown in the figure 4. When user chooses a file for opening (or decides to create a new one), the Files module creates a message which is sent to the Mediator. The Mediator module forwards this message and the Google Drive module responds to it by loading the file (or creating a new one). When it loads a new file, the Google Drive module sends a message to inform others that the file is loaded. This time, the Convertors module responds by sending the message that the note is ready. The Keyboard and Editor module react to this. When the user later presses the key, a message is sent by the Keyboard module through the Mediator and then received and responded to by the Editor module.

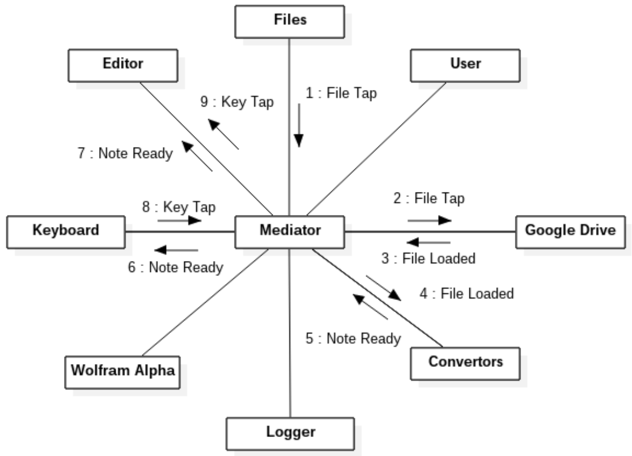

Figure 4: UML collaboration diagram showing interaction between JavaScript modules

A short overview of functionalities of some of the most important modules, such as the Editor or Wolfram|Alpha module, is given below.

Editor Module

The main goal of the editor module is to traverse the HTML element tree and modify it as the user is typing. It handles creation and removal of the individual symbols, as well as navigation through the formula by placing the caret. It reacts to messages which are created when the user presses a key, moves the caret, chooses an option in the auto complete list, etc. Most of the operations it performs rely on the jQuery library, which helps with manipulating the HTML content.

All symbols and formulas that can be written in The Mathist are represented by either a simple or a complex field. Simple fields are HTML elements that contain individual characters and symbols, such as letters or + and - signs, while complex fields represent parts of the formula that can contain sub elements, such as other simple and complex fields.

An example of a complex field is shown in Figure 5.

$$\sqrt[a]{b}$$

Figure 5: Root represent by a complex field

Complex field with sub elements which are also complex fields is shown in Figure 6.

$$\sqrt[a]{\lim_{x \to 0} \frac{\sin x}{x}}$$

Figure 6: Complex field containing other complex fields

The composite design pattern can be recognized here. However, slight modifications are introduced to model the various types of mathematical expressions in a better way. Namely, there are strictly defined areas in each complex field which can contain other elements..

The key responsibility of the complex field is to position the areas where the sub elements can be inserted. This is done by assigning each complex field type a specific CSS class. The class is then used to style both the complex field, and to position its children.

Wolfram|Alpha Module

This module is responsible for converting content from the internal HTML format used in The Mathist, as described above, into a format that the Wolfram|Alpha query engine can interpret and vice-versa.

Converting from HTML into Wolfram|Alpha compatible format is the easier task, since HTML perfectly describes the nested structures often found in mathematical notation. The result of this process is a plain text representation of the formula. The algorithm is a simple depth-first recursion, where each new level of recursion is separated with brackets. Additionally, brackets are used to separate each complex field on the same level of recursion.

The results provided by Wolfram|Alpha are given in plain text, which is not optimal for describing mathematical formulas. Because of this, for some specific mathematical expressions there may be more than one valid HTML equivalent. This was resolved by mimicking the way Wolfram|Alpha interprets the same input, to insure maximum compatibility. The general algorithm consists of several steps that are commonly found in code compilers. Tokenization comes first, followed by the intermediary code which overcomes the flaws of the plain text representation using simple heuristics. Finally, an abstract expression tree is created from the stream of corrected tokens. The abstract expression tree can be easily converted into the corresponding HTML representation.

Notes and Google Drive Modules

In order to be saved to Google Drive, The Mathist notes are first serialized by the notes module into the JSON format. Besides the note body, additional meta data is saved. It holds information such as the version of The Mathist application used for creating the note, which is important to enable the compatibility of the current version of application with the future ones.

The Google Drive module exposes a service to the notes module and stores the serialized notes as individual files, without interpreting their content. It reacts to messages such as: *load_files*, *open_file*, *save_file*, *confirm_delete_file*, which originate from the file chooser UI. The functionality of this module is based on asynchronous interaction with the Google Drive JavaScript API.

Separating the notes module from the file storage service, such as the Google Drive module, allows The Mathist to support different platforms and file storage services in the future.

The server and Ruby On Rails

While most of the functionality of The Mathist is implemented on the client side using the technologies and methods described above, some features were designed to work on the server side. For example, users' feedbacks, preferences and usage of Wolfram|Alpha credits are stored in the server side database. Also internal logs are kept on the server about errors and possible problems with the application.

The framework used to build the server application is Ruby on Rails, chosen for its ways of efficiently abstracting the handling of HTTP requests, interfacing with the database and for strictly conforming to the MVC design pattern. Built on top of the Active Record pattern for handling the object relational mapping and implemented in a dynamic, interpreted language such as Ruby, it presents a good choice for development of various server side applications.

Client Controller

The client controller renders the publicly available pages of the application, such as the index page as well as the app and beta pages. It handles all the requests from the client, once the user is logged in. For example, it will save the user's preferences, communicate with the Wolfram|Alpha API, collect feedback, save logs, etc.

Admin Controller

The Admin controller renders the pages for the administration of the website. It aggregates the most important metrics, such as the total number of users and the number of Wolfram|Alpha API credits available. Additionally, it handles the authentication and authorization of the administrator users.

7. FURTHER DEVELOPMENT

Since the simplicity is seen as one of its most important characteristics, it serves as a guide for further development. Having two types of input fields for formulas and text can be a bit confusing. Consequences are that the user must choose the appropriate, otherwise it will not satisfy user's needs. The fact that notes cannot be easily customized, and that there is no support for inline formulas, has proven to be a limitation for some users. Therefore, the editor should be improved to offer a more intuitive experience.

A major step towards this goal is already made and is available in the beta version of the application. The whole note body, apart from the title, is a single area which supports seamless input of both plain text and complex mathematical expressions. By analyzing the statistical results available regarding users' response to this change, it can be concluded that it makes a very positive impact on the user experience. The percentage of users who decide to continue using the application has increased by approximately 60% if they tried the beta version.

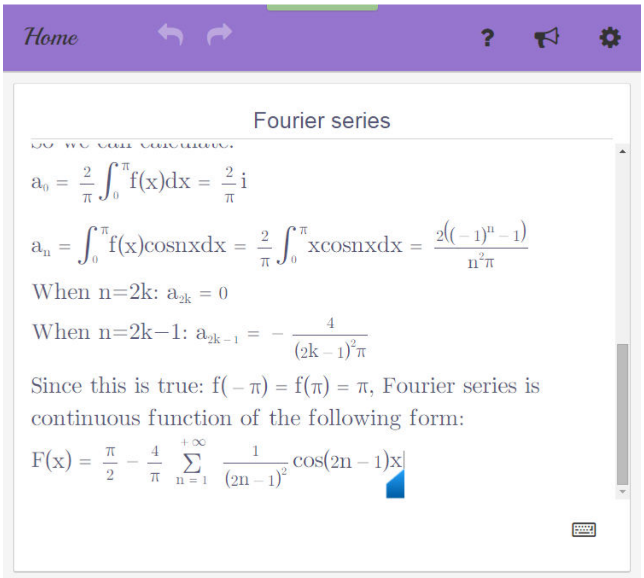

Figure 7: Editor in the beta version of the application

A problem proficient users may face is that although a lot of common formulas and types of mathematical expressions are available through the keyboard and through auto complete, the list might not be sufficient. Even though, more formulas will certainly be added through time, an alternative is also needed. The best solution found so far is to allow entering specialized formulas in the form of LaTeX, which would then get translated into HTML and inserted into the note. Most of the formulas that are in common use today are already supported, so a valid assumption is that a user who needs advanced mathematical formulas probably already has experience with other solutions, especially with LaTeX. That might also be a way to enhance application's functionality without making it overcomplicated for the average user. This is still being considered and no steps are currently taken to conduct this.

8. CONCLUSION

The vision of The Mathist we first had in mind while creating it, is to provide a platform for collaboration and writing of mathematical notes, without requiring any special skills or knowledge. Being a web application, it can be accessed by every student and professor allowing them to form virtual study environments in the cloud.

Further development is focused on making the application even more accessible to new users by reducing the initial learning curve. This is achieved by implementing well known user interface paradigms found in traditional text editors that users are already familiar with.

Also, as mentioned above, The Mathist is proving useful to students with certain kinds of disabilities that affect fine coordination, allowing them to use large touch screens as their primary input devices.

LITERATURE

1. *LaTeX* [Online] http://www.latex-project.org/.
2. W3C Math Working Group. *World Wide Web Consortium.* [Online] http://www.w3.org/Math/.
3. *MathJax.* [Online] 2015. https://www.mathjax.org/.
4. *The Effects of Problem-Based E-Learning on Prospective Teachers' Achievements and Attitudes towards Learning Mathematics.* Üzela, Devrim and Özdemira, Emine. 2012, Procedia - Social and Behavioral Sciences.
5. *Bridging the app gap: An examination of a professional development initiative on mobile learning in urban schools.* Mouza, Chrystalla and Barrett-Greenly, Tommi. s.l. : Elsevier, 2015, Computers & Education.
6. *WolframAlpha.* [Online] 2015. http://www.wolframalpha.com/.
7. *Implementation of Web-based dynamic assessment in facilitating junior high school students to learn mathematics.* Wang, Tzu-Hua. s.l. : Elsevier, 2011, Computers & Education.
8. *Teachers and mobile learning perception: towards a conceptual model of mobile learning for training.* Yusria, Iin Karmila, Goodwina, Robert and Mooneya, Carl. s.l. : Elsevier, 2015, Procedia - Social and Behavioral Sciences.
9. West, W. and Vosloo, S. *UNESCO policy guidelines for mobile learning.* Paris, France : UNESCO, 2013.
10. *Google for Education.* [Online] Google. https://www.google.com/edu/.