EVOLVING THE VERY ACT OF READING WITH SOCIO-COLLABORATIVE DIMENSIONS - AR-ANNOTATIONS OVER PHYSICAL BOOKS

Iulia-Cristina STĂNICĂ [1, 2], Alin MOLDOVEANU[1], Hesham Adnan ALABBASI[3], Florica MOLDOVEANU[1], Alexandru GRĂDINARU[1], Maria DASCĂLU[2], Anca MORAR[1], Victor ASAVEI[1], Mireille RĂDOI[4]

Revitalizing reading and modernizing books have become important issues today, when technology is everywhere around us. New means are needed to make reading and physical books attractive to all people, regardless of age, hobbies or occupation. Our current paper presents an innovative project called Lib2Life which creates a mobile application that can improve books by adding annotations and social capabilities to them.

Keywords: annotation, digitization, digital reading

1. Introduction

The current technological era we live in introduced digital components in almost every field. Social media, web pages and online courses have started to be preferred in terms of means of information, study and communication. As physical books start to lose their charm, new means are needed in order to revitalize them and make them popular especially among young people. E-books were created as a digital reading alternative, but studies show that they did not reach a high popularity [1]. Emerging technologies, including virtual, augmented or mixed reality are proved to be effective tools for revitalizing cultural environments, such as libraries, patrimony buildings or universities [2] [3].

By combining physical books with emerging technologies (augmented reality, internet of things, optical character recognition, social media), we can bring a series of advantages: increase interest, raise attractiveness, encourage

[1] Faculty of Automatic Control and Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: iulia.stanica@upb.ro, alin.moldoveanu@cs.pub.ro, florica.moldoveanu@cs.pub.ro, alex.gradinaru@cs.pub.ro, anca.morar@cs.pub.ro, victor.asavei@cs.pub.ro

[2] Department of Engineering in Foreign Language, University POLITEHNICA of Bucharest, Romania, e-mail: maria.dascalu@upb.ro

[3] Computer Science Department, College of Education, Mustansiriyah University, Iraq, email: hesham.alabbasi@uomustansiriyah.edu.iq

[4] Central University Library of Bucharest “Carol I”, Romania, e-mail: mireille.radoi@gmail.com

collaboration. A digitized physical books can close the gap between ages (young people who prefer digital content and elder people who prefer physical books).

Our paper presents Lib2Life, a prototype of a digital social-collaborative annotation system for physical books, working on smartphone devices, as a means of revitalizing and augmenting books and libraries [4] [5]. In the next chapters, we are going to present digital reading and collaborative editing as information and communication methods, then we are going to present the Lib2Life application, including the user functionalities and development process. Finally, we will draw the conclusions of our study and detail the future improvements of the application.

2. Digital reading & Collaborative editing

Digitization of information is essential in a great variety of fields, from library documents preservation to data extraction from medical records [6]. Digital reading refers to the process of extracting meaning from any text which is found in a digital format (web pages, digital documents, ebooks and so on). Digital reading has some differences when compared to classical reading of physical books, in terms of process, skills required and consequences on the reader's cognition. It allows a more selective, non-linear process of assimilating the information [7]. Furthermore, digital reading is usually part of a research task, as it involves browsing through digital content to find the appropriate information [8]. Digital reading might also encourage criticism (in an era when anyone can publish anything online, one must be selective) and social collaboration, as digital content can be easily shared on social media with friends or colleagues. In this technological era, emerging technologies, such as virtual reality (VR) or augmented reality (AR) are used for making the reading process more attractive or efficient [9]. AR represents a technology which combines both real world and virtual elements, allowing the user to interact with the virtual ones (images, graphical elements, 3D objects or animations etc.) [10]. Studies show that including augmented reality elements in the reading process can lead to an improved level of comprehension of the reading content, increased satisfaction and desire to learn more about different topics [11]. This can be very useful when we think about both written courses, literature books, magazines or scientific work.

The principle of sharing, commenting and improving content can also be seen in the collaborative editing process which is gaining more and more ground nowadays. Real-time collaborative editing is widely used, as multiple users can contribute simultaneously on a shared document. Google Docs is probably the most famous example, as it is used in both industry and research fields, allowing users to write their contribution to the same document and reviewing in real time the work of their peers. Conversations started as comment threads (comments and their replies) allow exchanging ideas, reviewing and discussing suggestions related to a certain topic or part of the content. The writing process becomes thus more interactive, collaborative and social, as changes and discussions are taking place in real time. Small scale scenarios of collaborative editing have been extended and they involve now even entire communities of users [12]. Collaborative editing, if included in the reading process of physical books, could have a huge impact and increase their popularity and attractiveness.

All these aspects are reflected in the young people's reading preferences, who sometimes perceive classical reading as outdated and unnecessary. Unfortunately, they do not take into consideration the cognitive impact of digital reading, which can affect concentration, recall rate or comprehension of the text's meaning [13]. This is why we need to include the social-collaborative advantages of digital reading and the cognitive advantages of classical reading in the same place. This is also the main working principle of our project, Lib2Life, which tries to revitalize reading and libraries by creating a mobile application which augments physical books by adding interactive digital elements to them. The social-collaborative component is essential, as all the content created by users is saved in a database and can be shared with close friends or with all readers. Collaborative comment threads are improving the attractiveness and can make a physical book interactive and pleasant. By including millions of books from the partner libraries in the database, and by also allowing users to add their own preferences, a large data set of knowledge is created and shared.

3. Lib2Life mobile application

The application is developed for Android mobile devices and future releases are intended to extend it also to iOS. The goal is to allow users add social-collaborative annotations while reading physical books. By scanning a certain text fragment or an image from the book, the user can add digital annotations which can be further shared with other users, creating thus an entertaining, collaborative reading process. Every person who is passionate about reading, every student who wants to read their colleagues' notes or share their own notes can benefit from the social-collaborative Lib2Life application. Humanities fields, where analysis and reflection on the text's meaning is essential, can find the application useful, as different perspectives can be shared and discussed. Medicine students who want to add notes on their medical atlases to help them study, or researchers who want to discuss their scientific findings - all of them can use Lib2life as a social-collaborative and communication platform.

Currently, the Lib2Life application in conceived and optimized for smartphones, in order to be able to reach a wider public. In the future, according to the evolution of technology and augmented reality devices, we intend to adapt it and make it function in a more natural way, using smart glasses.

In the following subchapters, we will describe the applications' functionalities, the proposed architecture and the used technologies.

3.1. Functionalities

The Lib2Life application includes essential functionalities needed in a social-collaborative, AR-based application:

- account and profile management
    - create an account
    - configure profile
    - configure group of friends
    - login with email & password or with social media account
- books management
    - save preferred books in a list
    - search for a book in the database by scanning the ISBN or introducing it manually
    - add a new book in the database
- defining annotations
    - scan the book page for adding / visualizing annotations
    - add a new social-collaborative annotation in the database (with various types and various levels of privacy)
- visualizing annotations from a page
    - visualize highlighted text fragments for a scanned page
    - visualize number of annotations and comments per highlighted text
    - visualize annotations of a highlighted text fragment based on different levels of visibility
    - visualize details of an annotation
    - add a comment
    - rate an annotation
- annotations overview and search
    - visualize a summary of annotations for a book, sorted by page order, number of views or rating
    - visualize a summary of all the annotations of a user
    - visualize a summary of all the annotations for a topic
    - visualize a summary of all the annotations containing a specified text
- social media
    - share annotations on social media

In order to use all the capabilities of the Lib2Life mobile application, the new users must create an account, by using either their email address or a social media account (Facebook, Twitter or Google). The use of a social media account is essential for collaborative aspects, as users can share content from the application, invite friends from social media or create groups with social media friends who already use the application.

Each logged in user can add preferred books in their application (Figure 1 - a), by either scanning the ISBN or manually searching the book in the database based on their title - author - edition combination and / or isbn. The database contains 4 million titles of books included in the archives of the libraries which are partners in the Lib2life project. If the wanted book is not found, the user can add it manually so that it will be saved in the database and become available for everyone. The user can take a photo of a book's page using their phone camera and highlight the region of interest (containing either a few words, a text paragraph or an image). Then they can add digital annotations in terms of text, image, video or link with preview, which are saved in the database (Figure 1 - b).

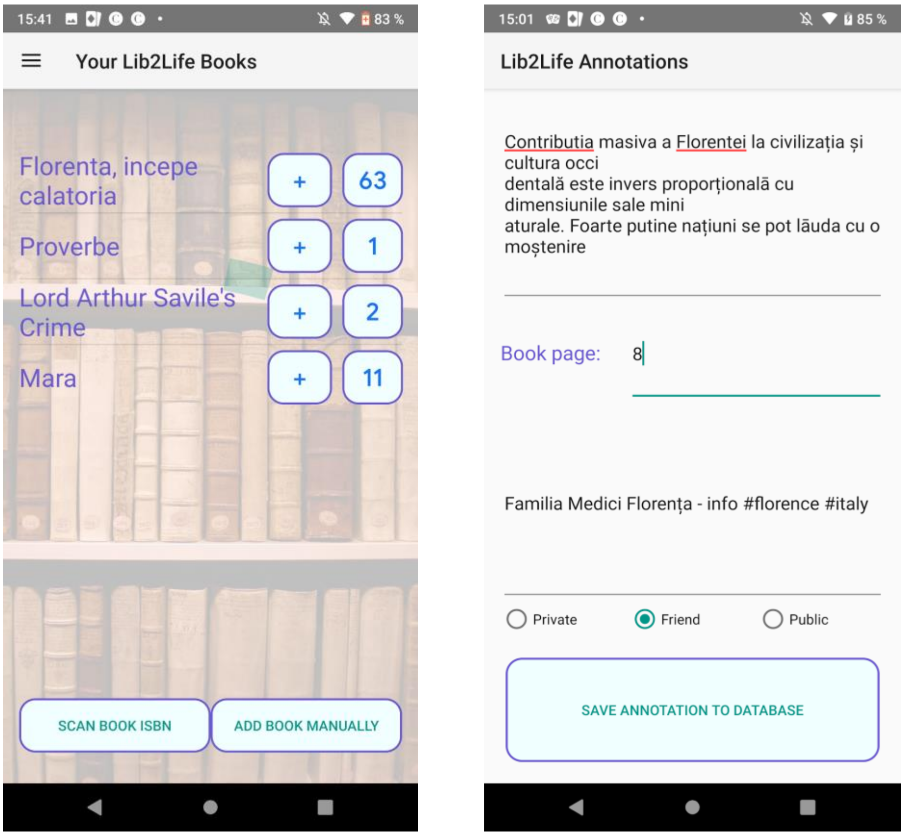

Fig. 1. Lib2Life Mobile Application a - List of saved books, b - Adding a new annotation with topics

Each annotation can have one or more topics associated to it, in a Twitter manner (using a hashtag symbol - #) and one or more users mentioned (using at symbol - @). If the user puts the special symbol (# and @ respectively) and types 3 characters, the application will display in a dropdown list the possible suggestions of topics / user mentions that the user can add to the annotation. This is based on the topics and the users which already exist in the database. A topic is saved in the database if at least one annotation containing this particular topic exists. Furthermore, annotations have three levels of accessibility: private (visible only to the current user), friend (visible to a group of friends) and public (visible to all users), which restrain their availability.

The added annotations can be further visualized based on various criteria. The user selects the corresponding option for viewing annotation and scans the specific page of a previously selected book. Based on text matching, the application will detect the AR elements in terms of annotations anchors (all fragments which have annotations present on that specific page, highlighted with various colors, based on the level of accessibility: green - public annotations, blue - friend annotations, purple - private annotations) (Figure 2 - a). Augmented Reality is often based on so-called markers (visual elements which are tracked by the camera or smart glasses) and the AR content is rendered in the real world based on the marker's content. Our application does not need special markers, as every text fragment detected by OCR and containing annotations on it will become the marker for our annotations. Each highlighted fragment displays the total number of annotations contained and the total number of comments for the annotations associated on that particular fragment. Clicking on each particular fragment will display its list of annotations. The user can then select a particular annotation from the list in order to see details about it: comments section and rating. By choosing the reply option, the user can add a comment to the current annotation (a simple text, a text with topics or user mentions, an image or a link with preview). A share option is also available, to distribute the annotation by email or on any social media used by the user. A 5-star rating system permits the classification of the annotations based on their popularity (Figure 2 - b). Topics used in annotations are hyperlinked and can bring the user on a page displaying all annotations containing that particular topic (Figure 2 - c). Furthermore, clicking on mentions will open the user's profile page. This can be useful if we want to visualize the annotations added by a certain user (if they have a public privacy or one of type “friend” for users present in our configured group of friends. These friend groups can be created wither based on friends from social media (if the user created their Lib2Life account using social media) or by searching people in the Lib2Life application based on their name.

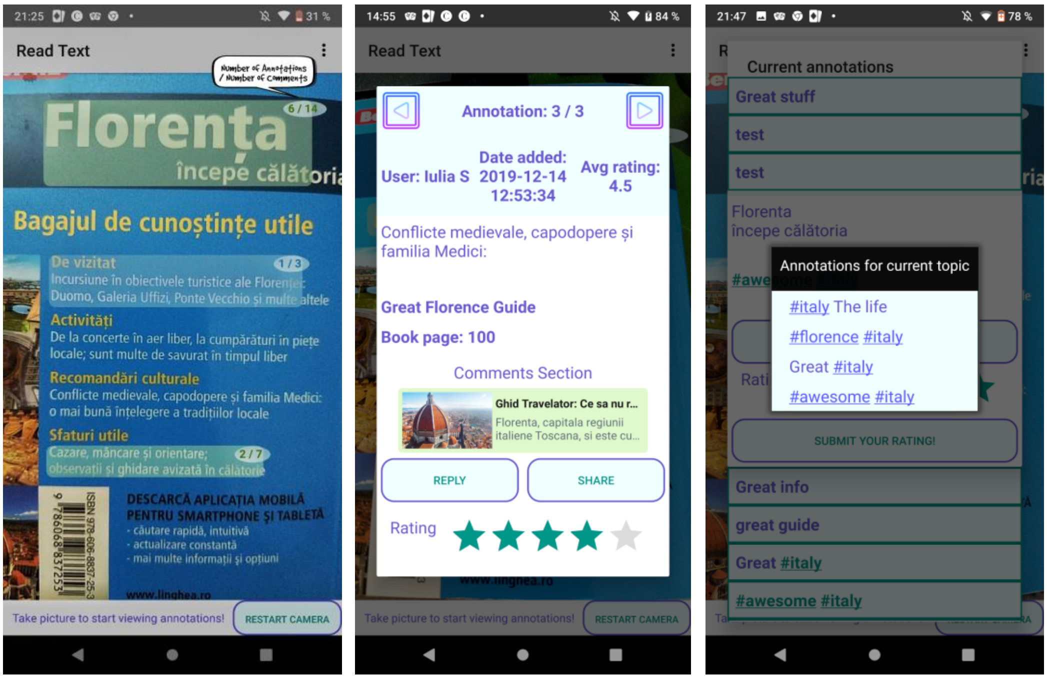

Fig. 2. Visualizing annotations a - Text highlights with number of annotations and comments, b - Annotation details and comments section, c - Annotations for a specific topic

3.2. Architecture

The architecture used is a multi-tier one (separating the interface, business logic and data) thus ensuring a low coupling of the systems' components. As the system has a modular architectural structure, the mobile application can be easily connected to other components (mostly the server including the database) or can be extended further with other functionalities. For each functionality described above, the user information from the interface (GUI) is taken and verified and / or processed by the controller in order to build the appropriate model which is sent to or received from the server. The communication with the server is done through a REST (representational state transfer) service, for creating, updating or deleting information from the database. The data exchange is done through JSON files.

The application includes several open source APIs for Android, which will be detailed in the following subchapter (including the creation of the social media account, OCR capabilities, creating external links etc.).

3.3. Technologies

The server for the Lib2Life application is developed in PHP using Laravel web framework and following the model-view-controller architectural pattern.

The mobile application can thus easily communicate with the database server using REST services [14]. REST architectural style includes a series of principles needed in order to manipulate Web resources in a client-server architecture, usually through the HTTP(s) communication protocol. All resources are identified using special uniform resource identifiers (URI) and can be further manipulated through four operations: POST (create), GET (read), PUT (update/replace) and DELETE (remove). In our particular case, all resources used in the mobile application (user register data, login data, book details, annotations, topics etc.) will be accessed (sent or received) using a standard format called JSON. JavaScript Object Notation (JSON) is a well used data-interchange format, which can be easily understood by both humans and machines [15]. The data interchange is allowed thanks to the use of asynchronous tasks in Android (AsyncTask) which assure the executions of instructions on a background thread, keeping the main thread available. If the main thread needs the result of the background thread, the user interface will be temporarily disabled and a progress bar will be shown. The progress of the background task can be shown using the onProgressUpdate() method, and the result will be returned by onPostExecute() [16]. For a better modularization of the application, increased readability of the code and the avoidance of deprecated functionalities, we plan to integrate Retrofit, an HTTP client API for Java and Android [17].

Each window of the application is an Android component of type Activity, the transitions between activities being made through buttons, listview elements or menus. Data transfer is accomplished through intents, also useful for selecting images from the phone's gallery or distributing annotations by email or social media. We also include in the application a local SQLite-based Room database, used to store the data required for a user registration session (for example, to store the user's favorite books on the device, allowing limited functionality even without an internet connection). Room uses Entity type elements to create data templates (books, annotations, etc.) and DAO (Data Access Objects) elements to retrieve entities from the database or send changes to the database. Room database is essential because it allowed us to implement and test functionalities before having them implemented on the server or before having to connect them to the server [18].

The authentication in the mobile application is done using OAuth2 - the standard authorization method for all web services, which is based on tokens. The interaction between the user and the application is permitted only after obtaining a so-called access token. An access token is a unique string obtained by each user after creating a new account in the application, which must be further used in order to access the application's resources. Each access token has a certain life span, after which it will expire, so the server will verify if the access token is still valid and correct at the moment of use. OAuth2 uses also a second type of tokens (refresh tokens) which can be used by the user to obtain new access tokens (after the current one has expired, for instance) [19]. Access tokens are encrypted and stored in our application's shared preferences, so that they allow the autologin of the user (so that users do not have to login each time they open the Lib2Life application on their personal mobile device). Furthermore, storing the token in the local file on the device will allow easy transmission of HTTP requests for accessing resources on the server, since the access token must be included in the JSON headers in order to have the right to perform any operation on the data stored on the server database.

The most important technology for the functioning of the application is related to the optical character recognition (OCR) process. The technique is used for recognizing text from scanned images and converting it into editable digital content [20]. In our case, we need OCR for recognizing the text which is annotated and the ISBN code for identifying the scanned books. Normally, before being used for identifying text, images must be preprocessed (through deskew – aligning the text, binarization – obtaining a black and white image, layout analysis, line and word detection, isolating characters). We wanted an API which does most of these features, plus recognizing the characters. Several methods for text recognition were investigated (including Tesseract, ABBYY Real-Time Recognition SDK and Yunmai OCR) and after performance analysis, we decided to use the Google Mobile Vision API for both character recognition and ISBN detection. This API is completely free to use and can be integrated in both Android and iOS applications, providing very accurate results if the camera of the phone can take pictures with a decent resolution and has a good autofocus system (features present on all mid-level mobile phones appeared in recent years). Of course, the better the resolution and autofocus, the higher the accuracy of OCR. After the tests performed (with Google Pixel 2 XL resolution 12.2 MP, f / 1.8), the results are so good that, most of the times, the detected text differs only by some diacritics from the original text (for instance, the text in Romanian “Florența, orașul plin de cultură” is recognized as “Florenta, orasul plin de cultura”). The Google Mobile Vision API uses a CameraSource object that we can use to configure the parameters needed for a good screen capture (good screen resolution and autofocus on; if only large blocks of text would be used, lower resolutions might also work, but we want to cover as many cases as possible, even if this will affect the image processing time by a few seconds). The text segmentation module divides it into blocks (paragraphs), lines and words, used by us for saving or detecting annotations. The processing of the detected content is done through the OcrDetectorProcessor module, which also includes a graphic component [21] (used by us for debugging - framing rectangles of the detected text and seeing the detected text placed over the real text, available even in real time, in order to evaluate the OCR performance). The components obtained from segmentation are essential, as they are used by us in the processes of adding and viewing annotations. Thus, to add an annotation, after making an image with the camera of the device, the user drags their finger over the desired text to make a selection rectangle (highlight). In the database, we save several components: the highlighted text, the collaborative content (annotation) and all its properties. We apply text matching based on the currently detected text by OCR and the text saved in the database by the user, more precisely we apply a modified "longest common subsequence" (LCS) algorithm [22], to determine the longest sequence in each pair of strings from the two groups of interest (text blocks segmented by the user-captured image on the one hand, and text saved in the current book database from previously added annotations). Unlike substrings, subsequences should not contain only consecutive characters, so that a wrong character of the detection algorithm (for example a diacritical sign, or characters with similar forms, eg "i", l, 1, etc.) will not have a strong impact on the algorithm (for example, if we used substrings instead of subsequences and wanted to detect the word BEFORE, we would have obtained that BEEF is closer to the correct word than BFFORE; in the algorithm used by us, BFFORE has a greater similarity, as only one character differs). Following the tests, we decided to take into consideration a similarity between the text in the database and the text blocks detected of at least 80% (for texts over 50 characters) and at least 90% (for texts shorter than 50 characters), in order to validate the comparison as being a match. The values of 80% and 90% were established experimentally, after performing tests with the Google Pixel 2 XL and with other devices with a similar camera. These values can also be modified in the future, based on the correlation with the performance of the device (cameras with a slower autofocus system will have a more predisposition to errors). Finally, all the annotations corresponding to the detected text are displayed to the user on the image made by them, and the initially selected text is highlighted with a color specific to the level of visibility. The same main modules are also used for ISBN detection, but a module is also added to read the barcode (thus allowing both the scanning of the ISBN as a numeric code and as a barcode).

The application is developed using Android Studio and Java programming language. It can thus be used on any Android device with the minimum version of the operating system (OS) being Android 5.0 (recommended version Android 9.0).

4. Conclusions

In conclusion, the initial version of the Lib2Life application for Android includes many functionalities essential for creating a digital annotation system for physical books. Users can create a personal account, add and visualize socio-collaborative annotations of different types, create threads of discussions. The results obtained through OCR using the mobile device camera have a good accuracy if the recommended hardware specifications are respected, thus ensuring a satisfying user experience. By finding the appropriate parameters for text matching, we created an application which can detect text with a high accuracy and revolutionize paper-based reading using augmented reality elements.

Currently, this first version of the application is in a testing phase, in order to improve the user experience (based on the opinion of all the people testing it, both technical and non-technical persons) and to improve the existing functionalities. After that, we will start developing secondary functionalities (filtering inappropriate content, adding more augmented reality elements) so that we can finally develop a complete AR system for evolving the very act of reading.

Acknowledgement

This work was supported by a grant of the Romanian Ministery of Research and Innovation, CCCDI - UEFISCDI, project number PN-III-P1-1.2-PCCDI-2017-0689 / „Lib2Life- Revitalizarea bibliotecilor si a patrimoniului cultural prin tehnologii avansate" / "Revitalizing Libraries and Cultural Heritage through Advanced Technologies", within PNCDI III

R E F E R E N C E S

[1] *The Digital Reader*, Trade Sales Up, eBook Sales Down Through September 2016, 2017

[2] *A. Moldoveanu, A. Gradinaru, O. Ferche, L. Ştefan,* The 3D UPB mixed reality campus: Challenges of mixing the real and the virtual, 18th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, 2014, pp. 538-543.

[3] *M. Anton, G. Nicolae, A. Moldoveanu, O. Bălan*, Virtual museums - technologies, opportunities and perspectives, Romanian Journal of Human - Computer Interaction, vol.11, no.2, 2018, pp.127-144

[4] *M. A. Băluțoiu, A. Grădinaru, A. Moldoveanu, F. Moldoveanu, A. Morar, A.-K. Nazare, M. Rădoi*, LibQuest - Revitalize Libraries and Reading Through Gamification, 15th International Scientific Conference on eLearning and Software for Education (eLSE) - New Technologies and Redesigning Learning Spaces, vol I, pp. 173-180

[5] *A. Moldoveanu, I. Stănică, M.-I. Dascălu, F. Moldoveanu*, A review of methods used for revitalizing books and reading experience, International Symposium on Fundamentals of Electrical Engineering, București, 2018

[6] *L.A. Tache, M. Dragoicea, E. S. Apostol, C. O. Truica.* Text Mining and Data Analysis of Medical Records, The 7th IEEE International Conference on E-Health and Bioengineering - EHB 2019 Grigore T. Popa University of Medicine and Pharmacy, Iasi, Romania, November 21-23 2019

[7] *C. E. Carter*, Mindscapes: Critical Reading Skills and Strategies, 2nd ed. Wadsworth, Cengage, 2014

[8] *J. Coiro,* Rethinking Online Reading Assessment in Educational leadership: journal of the Department of Supervision and Curriculum Development, N.E.A vol. 66, March 2009, pp. 59-63

[9] *I. Stănică, A. Moldoveanu, M.-I. Dascălu, F. Moldoveanu, M. Rădoi, I. Nemoianu*, Emergent Technologies to Enrich Reading Outcomes through Augmented Reality, Revue Roumaine des Sciences Techniques, Romanian Academy, vol. 64, iss. 1, pp. 95-100, 2019
[10] *R. Azuma,* A survey of augmented reality, Presence: Teleoperators and Virtual Environments, vol. 6, 1997, pp. 355-385
[11] *H. Bursali, R. M. Yilmaz,* Effect of augmented reality applications on secondary school students' reading comprehension and learning permanency, Computers in Human Behavior, vol. 95, 2019, pp. 126-135
[12] *Q.V. Dang, C.L. Ignat*, Performance of real-time collaborative editors at large scale: User perspective, 2016 IFIP Networking Conference (IFIP Networking) and Workshops, Vienna, 2016, pp. 548-553.
[13] *Sydney Morning Herald*, Internet Makes Us Stupid: Study, Australia, July 2014
[14] *The Java EE 6 Tutorial*, What Are RESTful Web Services?, Official Oracle Java Docs, 2013, https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html
[15] *D. Crockford*, How Javascript works?, Introducing JSON chapter, 2018
[16] *Android developers official documentation,* AsyncTask, https://developer.android.com/reference/android/os/AsyncTask
[17] *M. Pöhls*, Retrofit — OAuth on Android, Future Studio Tutorials, 2015
[18] *Android developers official documentation*, Room Persistence Library, Android Jetpack, https://developer.android.com/topic/libraries/architecture/room
[19] *S. Peyrott*, Refresh Tokens: When to Use Them and How They Interact with JWTs, Auth0 blog, 2015, https://auth0.com/blog/refresh-tokens-what-are-they-and-when-to-use-them/
[20] *R. Kasturi, L. O'Gorman, V. Govindaraju*, Document image analysis: A primer, Sadhana vol. 18, 2002.
[21] *Developers Google*, Text Recognition API Overview, Mobile Vision, https://developers.google.com/vision/android/text-overview
[22] *L. Bergroth, H. Hakonen, T. Raita* (2000). "A Survey of Longest Common Subsequence Algorithms". SPIRE. IEEE Computer Society. 00: 39–48