NFC-based Pervasive Learning Service for Children

Rosen Ivanov

Abstract: *In this paper we present the design, development and implementation of the pervasive learning system for children. The required hardware is a Near Field Communication (NFC)-enabled mobile phone with built-in accelerometer. Trainers (teachers or parents) describe the objects from the learning environment through their voice and associate this voice-based description to a specific object by using the radio frequency identification (RFID) tags. They are able to sync their audio recordings, to share and re-associate with their own RFID tags (clone) audio recordings of other trainers through Google App Engine cloud environment. Children can later use the service to scan surrounding augmented objects and verbalize their identity and characteristics. The service allows the development of a variety of learning scenarios, such as getting started with the colors, letters and numbers, shape recognition, recognition of objects from learning environment, learning foreign languages, and many others. Interaction between children and objects from learning environment should be simple and intuitive. To meet this goal for invisible computing, we simulate human-human interaction than use well known human-computer interaction models. We use a tangible user interface (TUI) based on following interaction techniques: touch, gesture recognition, natural voice-based output. Preliminary results show that the service can be used easily by young children, thanks to its tangible interface that is simple, easy to use, useful, accessible and invisible to technology.*

Key words: *children's learning, u-learning, tangible user interfaces, NFC-enabled phones.*

INTRODUCTION

Today, trends in IT technologies impact in almost all areas of public life, including the education of children [1]. Solutions that give e-learning systems are not directly applicable to the children's learning. In such systems, the user interface should be simple, transparent to the technology, intuitive, accessible, and where is possible must simulate the natural way of communication between people. Children usually interact with the physical world by touching, moving, and looking at surrounding objects. Before beginning with the development of children's learning system, we need to get an answer to the following question: Can the children use electronic devices that simulate the above techniques for interacting with the real world? The majority of electronic learning systems for kids like the Fisher Price iXL, VTech's MobiGo, LeapFrog's Leapster and others are based on customized electronics hardware and software. The user interface is simple - the number of buttons is minimal and the main way to interact is by touching the screen of the device. Such electronic systems can be used successfully by children aged 3-8.

Research suggests that tangible user interfaces (TUIs) were associated with improved usability [2]. They are often considered to be more fun to use than traditional kind of interfaces [3, 4]. Up-to-date mobile technologies enable the creation of TUIs through which children have the opportunity to acquire knowledge interacting with real objects rather than their text, image or video description. When these interfaces are implemented, mobile learning [5] can be done anytime and anywhere, not just in classrooms [6]. Compared with desktop computer e-learning, mobile learning (m-learning) is implemented with mobile devices such as smartphones and tablets. M-learning systems enable the learning at anytime and anywhere, but they cannot take into account context information from surrounding learning environment. M-learning systems are inaccessible and not sufficiently useful for children. In contrast, the ubiquitous learning (u-learning) systems allow children to obtain learning information from the learning environment through variety type of sensors [7, 8].

CompSysTech'13, June 28-29, 2013, Ruse, Bulgaria.

Sensors are major part of u-learning systems [9] as they allow designing tangible interfaces – a bridge between physical world and users. Radio frequency identification (RFID) tags and quick response (QR) codes are often used to realize context-aware services and pervasive computing environments [10].

Pervasive computing integrates technologies such as mobile devices, wireless communications and embedded sensors into a personal environment. Pervasive learning benefits when using TUIs are the follows: (1) There are close link between physical objects and cognition; (2) They are more intuitive and suitable for users with little technical experience, particularly for children; (3) Coupling physical activity with learning my increase reflection in young children and children with special needs; and (4) The opportunity for self-study and teaching at home.

SYSTEM ARCHITECTURE

We use things-to-human interaction to recognize and describe objects from learning environment. The proposed children's learning system has got a greet potential for children because it is simple, easy to use, useful, accessible and invisible to technology. We use smartphones with built-in NFC reader and accelerometer as an interface to physical objects from learning environment.

The basic requirements, related to the design and implementation of the proposed system are follows:

1. Human-centred design. The architecture of the system and the user interface must be conformable to the expierence of the teachers and children. We use human-centred design, based on the standard ISO 9241-210:2010. Final version of the user interface is obtained from an adaptive iterative human-centred process.

2. Invisibility to the technology. Users must be able to work with the system, without the need to have additional technical expertise and long time training.

3. Availability. The system must be available to users from any place at any time.

4. Usability. The system must be as simple and natural as possible.

6. Usefulness. The system should lead to improving the level of knowledge of children.

The proposed solution for children learning is a cloud-based mobile service (see Figure 1). The system's main components are the following: (1) Client-side mobile app that runs on the smartphone with built-in NFC reader and accelerometer; (2) Passive NFC tags are used to identify physical objects, and (3) An optional for the solution cloud-located Software-As-A-Service (SaaS) that allows sync, sharing and cloning of user's recordings. The software on the client side is responsible for: application registration, object scanning, audio recording, deleting audio recordings, sync, sharing and cloning of user's recordings. Access to the cloud computing environment is obtained through the radio network of the mobile operator (bearers UMTS, HSPA or LTE) or through the available Wi-Fi access points. Client-side software is a signed MIDlet. To build the client-side application we use Oracle Java ME as a mobile platform, due to its platform independent architecture and because it provides all the required APIs: access to NFC reader (JSR-256), access to accelerometer (JSR-257), audio capture (JSR-135), access to flash drives (JSR-75), access to WEB services through HTTP and HTTPS (JSR-118), and push notifications (JSR-118).

The modules that implement the functionality of the client-side application are following: (1) NFC module is responsible for communication with RFID/NFC tags and RFID smart cards; (2) Accelerometer module (ACC) implements the processing of raw data from a built-in electronic accelerometer; (3) Mike module digitizes the signal from the micro-phone and records audio samples in the application's local cache as an audio file; (4) Text-To-Speech (TTS) module is responsible for the implementation of the voice-based user interface; (5) Cloud module is used to build a HTTP communication channel between the software on the client side and the cloud-based WEB services; and (6) File manager (FM) module is responsible for access to the phone's flash memory (internal flash drive or an external flash card). This memory is used as a local cache for users' audio recordings. The FM module implements and accesses application's persistent memory called the Record Management Store (RMS). We use RMS to hold and restore the state of the application.

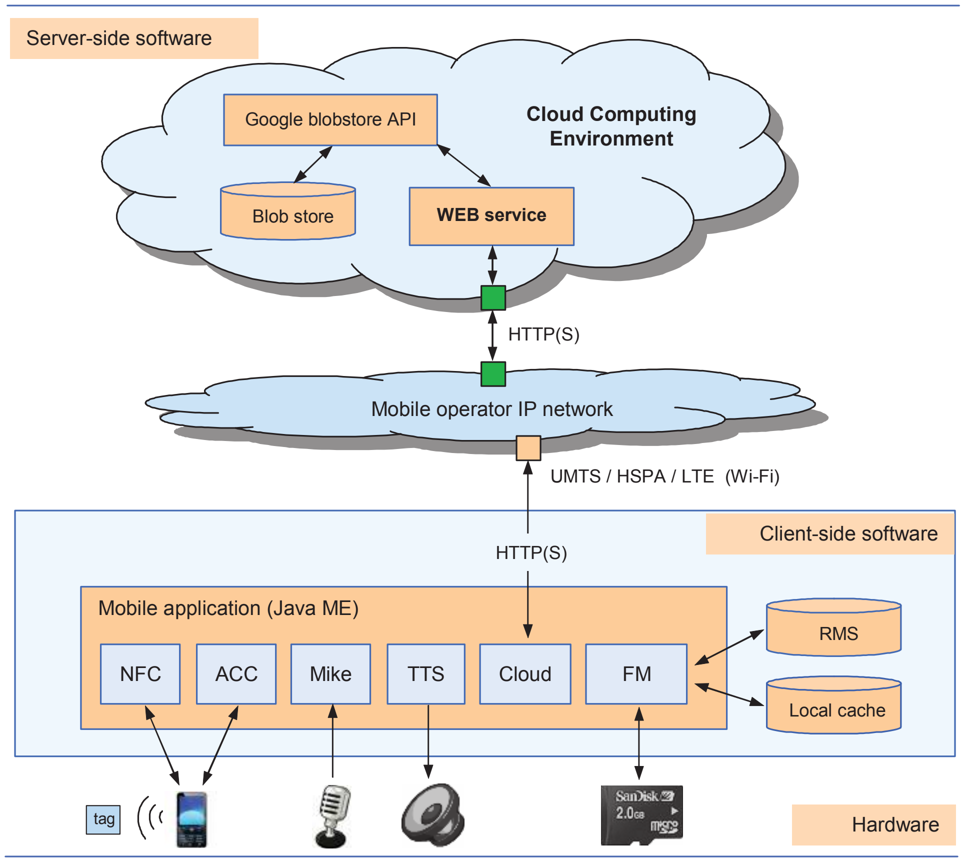

Fig. 1. General system architecture overview

We use Google App Engine (GAE) as a cloud-computing environment, due to its robustness, Java Enterprise Edition support and platform independent architecture. Blobstore service is used for the realization of sync, share and clone modes. We use Google Blobstore Java API to manage, upload, and download users' audio recordings. The main reasons for using this solution are the following: (1) simplified server-side program code and service deployment; (2) cost effective solution; (3) universal access to server-side software over standard POST HTTP requests; (4) data in the cloud can never be compromised; and (5) platform independent solution. The modes supported are as follows: registration, object scanning, audio recording, deleting audio recordings, sync, sharing, and cloning user's data. Switching from one mode to another is implemented using a shake- and tilt-based user interface (see Figure 2).

Mode "Registration"

When the application is launched after installation, mode "Registration" is activated. We use a UID code of the user's RFID ID smart card as an application identification code (AppID). Hashed value of AppID is stored on the RMS. After every launch of the application, the "Object scanning" mode is activated instead of "Registration".

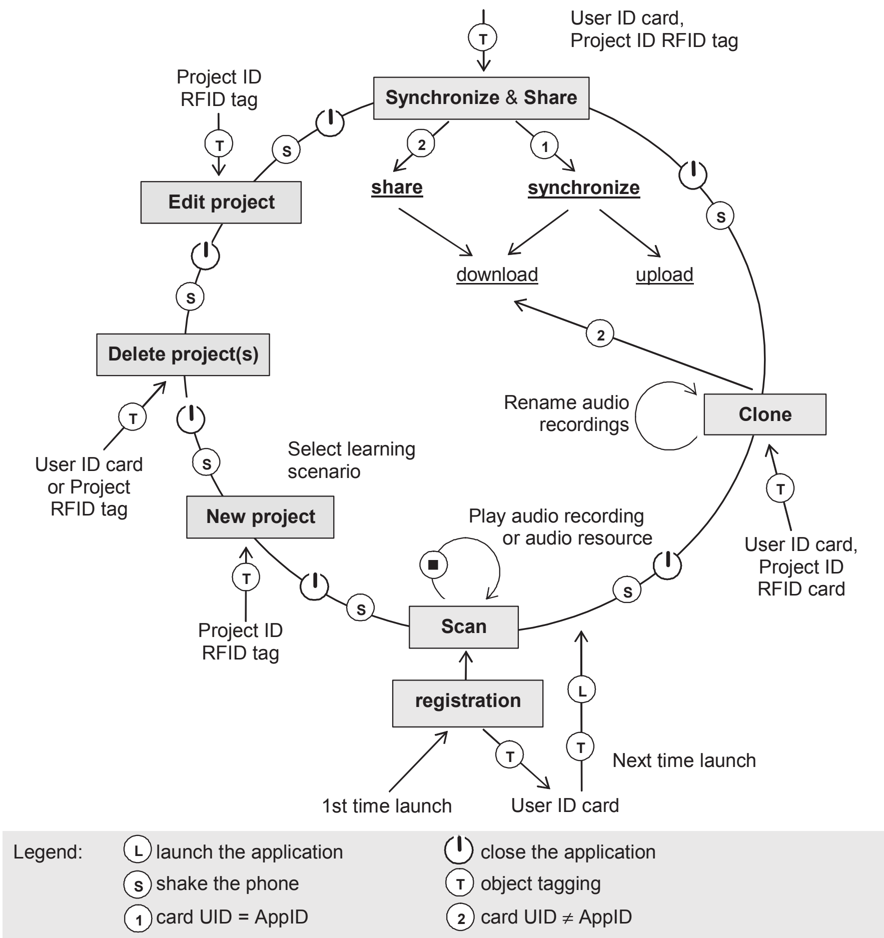

Fig. 2. Client-side mobile application life cycle

Mode "New project"

The creation of a new project goes through the following steps:

1. Create a Project ID RFID tag. This special tag that identifies the project is created after touching it to RFID reader of the mobile phone. Through this tag, it is possible to automatically start the application and activate the corresponding project.

2. Make voice-based description of the project.

3. Select learning scenario. The service supports the following learning scenarios: a) voice-based object description; b) "question-answer"-based scenario; c) find similar objects from a group of objects; and d) find the sequence of correct answers. For the last two learning scenarios are calculated score, which is based on the number of mistakes and time required.

4. Make audio recordings. This step allows users to record a voice message that describes the selected tag. The user must touch an RFID or NFC tag with his or her mobile phone, and this will be associated with the selected object. The application gives an audio announcement for the start of voice recording. To start and stop an audio recording, the user must shake the phone. The maximum recording time is limited to 120 seconds. The name of the audio file is equal to the tag UID.

5. Save project or cancel. For each project, the application creates a Project Description File that contains project's data in XML format.

Mode "Delete project(s)"

In this mode, the user can delete one or all projects. If a project ID tag is detected, only data that is associated with it is deleted. If the software detects an user ID card, mode "delete all projects" is activated. In this case, the client-side application deletes all project's data in the local cache. The status of the corresponding project(s) from the blobstore is set to non-shareable and non-clonable. If within 24 hours the user does not recover his or her project's data through sync mode, they are deleted from the blobstore.

Mode "Edit project"

In this mode it is possible to change the voice-based project description and selected audio recordings. The mode is activated with a project ID tag.

Mode "Synchronization and Share"

The proposed service supports bi-directional synchronization. In "Synchronization" mode the user can backup all their projects on blobstore. The reverse process is possible as well – project's data from blobstore can be stored on the phone's flash drive or card. The sync mode is activated with a user ID card. The application automatically decides whether to activate the upload, download, or upload and download of project's data. The mode "Sharing" is activated if the ID card belonging to the user whose projects are desired is used. In this case, it is assumed that the user agrees to share your data.

Mode "Clone"

When a user wants to create training material based on project of another user but with its own RFID tags, it is necessary to activate the mode "Clone." In this mode, after downloading the audio recordings with each one of them must associate a new RFID tag. For this purpose, after the user hears the audio recording he or she touches RFID tag to the phone's NFC reader. Actually, the client-side application only changes the names of the recordings with the UID of the new RFID tags.

Mode "Object scanning"

In this mode, when the user touches a RFID or NFC tag associated with an object, he or she hears the corresponding audio recording.

USER INTERFACE

The service has two types of users: 1) trainers, and 2) students. Trainers can be qualified teachers or children's parents. They get full access to the service, if they launch the client-side application with their User ID smart cards. The students can only scan objects from learning environment. The functionality of the service is accessed through a tangible user interface of type "one touch, one shake and tilt." Tangible techniques can benefit children's learning from many ways: (1) requires little time to learn how to use interface. Children can concentrate on learning task rather than how to use service or application, (2) TUI can provide children with social experience. The children are more productive when they learn in groups and (3) Services with a tangible interface can easily be adapted for use by children with special educational needs or emotional difficulties. The proposed client-side mobile application uses a multi-modal user interface which does not require the location of items on a screen: object touching, shaking and tilting the mobile phone, and voice-based output - notification to users about changes in the status of the application, incoming events, or any errors are implemented through prerecorded audio and voice messages.

Touch-based interaction

When the user touches a tag with his or her mobile phone, the following events can be occur: (1) Automatic launch of the application; (2) Play audio file, associated with the tagged object; (3) Record audio-based object description file; (4) Delete project(s); and (5) Sync, share or clone user's data.

Voice-based interaction

The application does not use the mobile phone screen to select or display any information. The sensitivity of the touchscreen is blocked. This gives freedom to children to hold the phone in the most convenient way for them. To save battery power, the screen is black.

Shake- and tilt-based interaction

Navigation in the "menu" of the application shall be implemented through analysis of data from a built-in accelerometer. Each raw value represents G-force values along the x-, y-, and z-axes, relative to the ground. In order to detect mobile phone shaking and tilt, we analyzed raw accelerometer readings and their filtered values. In Nokia phones, when using Java ME technology, the sampling rate of accelerometer data is 40Hz. This sampling frequency is sufficient to create not only shake- and tilt-based user interfaces, but also simplified user interfaces based on gesture recognition. Shaking the phone is used to change the current mode or to confirm events such as the following: start and stop of audio recording, "Cancel", "Request to exit", etc. When the user places the phone approximately horizontal with the screen to ground for a period of time over 3 seconds, this activates mode "Cancel" or "Request to exit", depending on the current state of the application. The user has 3 seconds to confirm activation of the "Request to exit" mode by shaking the phone.

The Matlab-style code of the algorithm that recognize event "shaking the phone" is shown in Figure 3.

```
Initialization:
TS = 25;                    // sampling period in ms
ALPHA = 0.75;               // IIR filter coefficient
startFlag = false;
lastMagnitude = 0;
WINDOW_LENGTH = 350; // time window length in ms
SHAKE_THRESHOLD = 2*g*0.75;  // default value
windowLengthInSamples = WINDOW_LENGTH /TS;
counter = windowLengthInSamples;
lastXF = 0; lastYF = 0; lastZF = 0;

Input:
Raw data from sensor (accelerometer): data
Output:
Add shake event to event's queue, if any.

X = data(0); Y = data(1); Z = data(2);
XF = ALPHA *X+(1- ALPHA)*lastXF;
YF = ALPHA *Y+(1- ALPHA)*lastYF;
ZF = ALPHA *Z+(1- ALPHA)*lastZF;
lastXF = X; lastYF = Y; lastZF = Z;

magnitude = (abs(XF)+abs(YF)+abs(ZF))/3;

if (startFlag == true),
    counter = counter + 1;
    if (counter > windowLengthInSamples),
        counter = windowLengthInSamples;
        startFlag = false;
    end
end

if (lastMagnitude <= SHAKE_THRESHOLD & magnitude > SHAKE_THRESHOLD),
    if (counter == windowLengthInSamples),
        Events.add (Events.PHONE_WAS_SHAKEN, magnitude);
    end
    counter = 0;
    startFlag = true;
end
lastMagnitude = magnitude;
```

Fig. 3. Pseudo-code of algorithm for "shaking the phone" event detection

This code is executed when a new data from accelerometer are detected. First, the raw data from the accelerometer is filtered with a low-pass filter (lines 2-4). For each acceleration vector (XF, YF, ZF) an estimate for the acceleration magnitude was computed (line 6). We join the three axes into one single vector (magnitude). Thus, the algorithm goes independent to phone orientation. The event PHONE_WAS_SHAKEN is detected when there is a transition in the acceleration magnitude from the value below the threshold SHAKE_THRESHOLD to a value above this threshold (line 14). At this moment a time filter with duration of WINDOW_LENGTH is activated (lines 7-13). If during the next detection of event PHONE_WAS_SHAKEN time window has not expired, then the event is rejected. The value of SHAKE_THRESHOLD is calculated adaptively for each user, depending on the preferred strength of shaking.

PRELIMINARY TESTS AND RESULTS

The service is at the stage of preliminary testing. The current version of the user interface and supported training scenarios are chosen after analysis of the interview conducted with 20 kids' teachers and 20 parents of children aged 3-8. Two scenarios were tested with 10 children aged 3-8 years: 1) Description of the object with voice or sound, and 2) Questions-answers. For the realization of these scenarios we use 3D cubes [11]. On each side of the cube is glued RFID tag. We use the following types of tags (see Figure 4a): 1) Application start tag; 2) Tags, associated with a voice description of the objects, questions or answers; and 3) Tags, containing sound description of the objects. The cube on the left side of Figure 4a is used for the study of animals. When a child touches a picture with your mobile phone on which there are animal and question mark it hears the name of the animal and its brief description. On the opposite side of the cube has a picture with the same animal and musical note. When a child touches this picture with your mobile it hears real animal sound. The cube on the right side of Figure 4a implements a training scenario based on questions and answers. The child hears the question and if it is difficult to answer, it could hear the correct answer, which is associated with the image on the opposite side of the cube, for example: Q: How many trees you see in the picture? Count! A: There are ten trees: one, two, three... ten. The experiments show that even young children aged 3-4 do very well with the service (see Figure 4b). They easily learn how to touch the cube with the mobile phone and quickly recognize which image contains a description of the object and which contains question or answer.

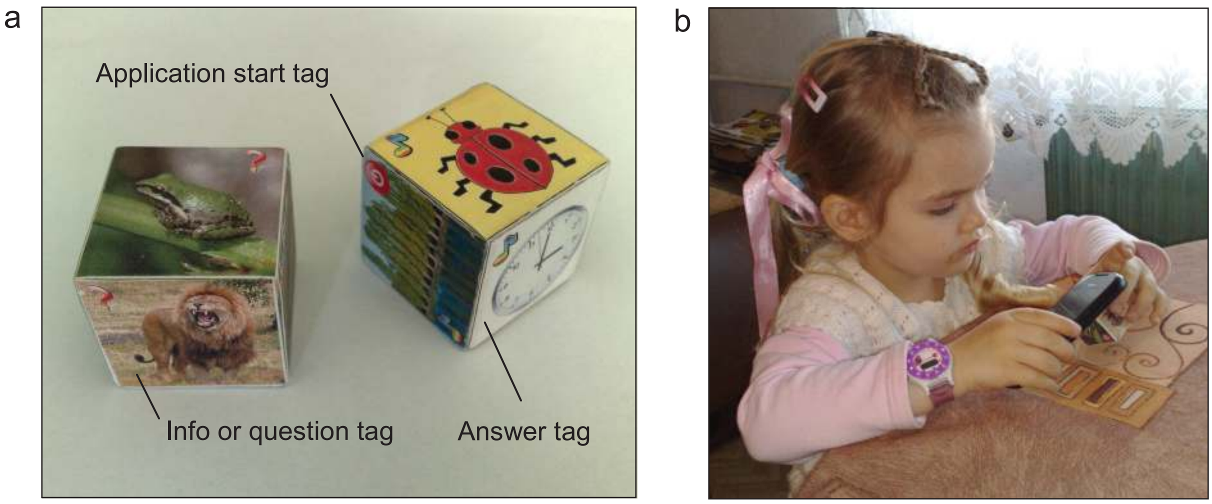

Fig. 4. Preliminary test of the service: a) interactive learning cubes, b) Little Nicole playing with the learning cube. Although it is only three and a half years old, she uses the service easily and with great interest.

CONCLUSIONS AND FUTURE WORK

This paper presents a novel mobile pervasive service for children's learning. The current version of this solution has been obtained after human-centred design. The service enables teachers to create training scenarios that allow children to learn in a funny and enjoyable way. The primary features of the proposed solution are following: First, high availability and easy of use of the system, thanks to the implemented human-centred design. Second, selected tangible user interface makes the users independent of the technology and ensures our aim for invisible computing. Third, the service is robust, secure, and available from anywhere at any time. Finally, children use without significant problems their mobile phones when they realize the communications of the type "human to human." For them this new way of learning through play is very interesting and amusing.

We plan to test the service with a large number of children of different age groups. The aim is to specify the final version of the user interface and the possible training scenarios.

REFERENCES

[1] Price, S. Tangibles: Technologies and interaction for learning, in Sage Handbook of Digital Technology Research, Chapter 20, 2013.

[2] Marshall, P., Y. Rogers, E. Hornecker, Are tangible interfaces really any better than other kinds of interfaces? In: CHI'07 workshop on Tangible User Interfaces in Context & Theory, 2007.

[3] Xu, D., Tangible User Interface for Children - An Overview, Proceedings of the SIXTH Conference in the Department of Computing, pp.579-584, 2005.

[4] Zaman, B. et al., Editorial: the evolving field of tangible interaction for children: the challenge of empirical validation, Journal Personal and Ubiquitous Computing, vol.16 (4), pp. 367-378, 2012.

[5] Lam, J., J. Yau, S. K. S. Cheung, A Review of Mobile Learning in the Mobile Age, Lecture Notes in Computer Science, Springer Berlin Heidelberg, vol. 6248, pp. 306-315, 2010.

[6] Tsung-Yu, L., T. Tan-Hsu, C. Yu-Ling, Outdoor Natural Science Learning with an RFID-Supported Immersive Ubiquitous Learning Environment, International Forum of Educational Technology & Society, vol.12 (4), pp.161-175, 2009.

[7] Pyykkönen, M., J. Riekki, I. Alakärppä, I. Sanchez, M. Cortes, S. Saukkonen, Designing Tangible User Interfaces for NFC Phones, Journal Advances in Human-Computer Interaction, Vol. 2012, pp. 1-12, 2012.

[8] Sánchez, I, M. Cortés, M. Cortés, J. Riekki, M. Oja, NFC-based interactive learning environments for children, Proceedings of the 10th International Conference on Interaction Design and Children, pp. 205-208, 2011.

[9] Mandula, K., S.R. Meda, D.K. Jain,, R. Kambham, Implementation of Ubiquitous Learning System Using Sensor Technologies, IEEE International Conference on Technology for Education (T4E), pp.142,148, 2011.

[10] Vazquez-Briseno M. et al., Using RFID/NFC and QR-Code in Mobile Phones to Link the Physical and the Digital World, Journal Interactive Multimedia, InTech, pp. 219-242, 2012.

[11] Terrenghi, L., M. Kranz, P. Holleis, A. Schmidt, A cube to learn: a tangible user interface for the design of a learning appliance. Journal Personal and Ubiquitous Computing, vol.10 (2-3), pp.153-158, 2006.

ABOUT THE AUTHOR

Assoc. Prof. Rosen Ivanov, PhD, Department of Computer Systems and Technologies, Technical University - Gabrovo, E-mail: rs-soft@ieee.org.