

TOWARDS AN E-LEARNING 3.0 CONTEXT-AWARE NOTIFICATION SYSTEM

Jianghua Hui¹ and Pedro Isaias^{2,3}

¹ ITEE (School of Information Technology and Electrical Engineering),

² ITaLI (Institute for Teaching and Learning Innovation),

³ UQ Business School,
The University of Queensland, St Lucia, Queensland, Australia

ABSTRACT

At present, E-Learning 3.0 is becoming more and more dynamic as it is a rapidly growing learning strategy. This paper focuses on designing and developing an E-Learning 3.0 Context-Aware Notification System for The University of Queensland Blackboard in order to improve students' university activities management and enhance their learning experiences. The conceptual and technical knowledge of Web and E-Learning was studied to form the prototype foundation. A number of design and development stages with various techniques were used to build the Front-end and Back-end of the prototype. The final E-Learning 3.0 Context-Aware Notification System prototype was made up of four components, which were Notification Data Extraction, Prototype Website, RESTful API and Prototype Database. This research project offers conceptual and technological contributions towards E-Learning 3.0 in both of the Front-end and the Back-end perspectives. Prospective works will be focusing on completing the rest of the proposed functions.

KEYWORDS

E-Learning 3.0, Web 3.0, Context Aware Notification System

1. INTRODUCTION

E-Learning is one of the most rapid growing industries in the world now and it will keep continue to grow fast. The market has increasing by 900% since 2000. The total European population that used the self-direct learning increased from 25% in 2007 to 32% in 2013. Besides, nearly 4.6 million students are undertaking one or more courses online in the United State, which is 46% of the total college students in France. The total number of college students who use E-Learning will rise to 50% by 2019 (Laskaris 2015).

The Blackboard Learning System is the current E-Learning system used by The University of Queensland (UQ). It provides a standard course management system for the classroom as well as an online educational assistance for the UQ students and staff (Bradford et al. 2007). In spite of this, different students have various needs and requirements in terms of study load management based on their enrolments. Therefore, the research question of this study is "How to Develop an E-Learning 3.0 Context Aware Notification System for UQ Blackboard?" As a result, this research aims to develop an E-Learning 3.0 Context Aware Notification System (CANS) that can help the students to manage their educational progress more effectively. It can add value by providing more comprehensive learning information through personalised notifications for each individual student. This notification system will also offer students convenience by providing them with integrated learning information, and will reduce the time spent on information searching and browsing.

This paper will first emphasise the literature relevant to the E-Learning 3.0 as well as the CANS. Then the methodology section will describe the design and building process for this CANS prototype. This is followed by the prototype result and outcome of the CANS prototype. After that, there will be discussions to analyse the CANS prototype as well as recommendations for future work.

1.1 Research Question

The research question is stated as follow: “How to Develop an E-Learning 3.0 CANS for UQ Blackboard?”

1.2 Research Aims

As a result, the project aims to develop an E-Learning 3.0 CANS that can help the students to manage their educational progress more effectively.

2. LITERATURE REVIEW

2.1 Web and E-Learning Concepts

According to Miranda, Isaias and Costa (2014), Web originated from Web 1.0 as Read Only Web. There was no direct communication between the users and the web owners. Then the web evolved into Web 2.0. as the Dynamic Web, which allows users to read, write, collaborate, social interaction and information sharing (Hussain 2013; Miranda, Isaias & Costa 2014). Web 3.0 will be the next generation web, which is also called the Semantic Web, which can understand the meaning of the content and it will predominantly rely on databases (Miranda, Isaias & Costa 2014).

Dominic et al. (2014) suggest that E-Learning 1.0 was created in parallel with Web 1.0. E-Learning 1.0 was a one direction system that provided interactive functions to connect to the students and instructors (Miranda, Isaias & Costa 2014). Then E-Learning evolved into the 2.0 version, where it became a platform that supported collaborations and interactions (Hussain 2013). Based on Hussain’s (2013) study, it is found that Web concepts and technologies have significant contributions to that of the E-Learning. The E-Learning 3.0 is evolving in parallel with Web 3.0, and it is built based on the Web 3.0 principles.

2.2 Web 3.0 and E-Learning 3.0 Concepts

Amarin N.Z. (2015) summarises four Web 3.0 technologies. The first technology is the Semantic Web, which means that the computer can understand the meaning of contents. The second one is Openness, and it indicates the openness among the Application Program Interfaces (APIs), data format, protocol and Interoperability between devices and platforms. Another one is the Global Repository of Data, where the data can be accessed through different applications and platforms in Web 3.0. The Distributed and Cloud Computing is also one of Web 3.0 technologies, being a service rather than a product.

E-Learning 3.0 technologies are predicted to be different from the previous E-Learning generation (Hussain 2013). This author also suggests that E-Learning 3.0 will contain five main technologies, namely Personal Learning Environments (PLE), Mashups, Social Semantic Web, Second Life and Personal Avatars.

Based on Dominic et al. (2014), it is found that E-Learning 3.0 is made up of a number of distinct technologies, which are identified as PLE, Mashups, Social Semantic Web, Personal Agents, Big Data, Linked Data, 3D Visualisation, Virtual Reality, Smart Ubiquitous Device, Augmented Reality, AI, Distributed Computing, Cloud Computing and Global Database.

2.3 Related Work

Three existing notification systems designed and built with Web and E-Learning principles and technologies are analysed and evaluated as case studies for the proposed CANS. The first case study develops a context-aware notification application named AlterMe and it is based on the Semantic concepts and personalisation. AlterMe is developed by using ontology languages. Web Ontology Language (OWL) is used for reasoning of the decision making, and the final decision is achieved by Reasoner with Semantic Web Rule Language. There are three modelling techniques used including Modelling people and course, Modelling alerts and Modelling teams and categories (Leonidis et al., 2009).

From the study of Haron et al. (2010), a Radio Frequency Identification (RFID)-based context-aware and personalised notification system is developed for university students. The context-aware indicates that the program or the application can recognise and respond according to users' requirements. In addition, it can change based on users' behaviours. The article identifies four types of context for the project, and three out of four types are selected to form the notification context.

In Amelung's study (2007), the notification system is based on the learner's social context and individual's preferences. The Context-Aware Activity Notification System (CAANS) provides activities' notifications relevant to the users based on the social context and personal interest. The paper then defines the social context and discusses the social context hierarchy. After that, the authors identify the principles of the CAANS framework, where those principles form the foundation of the CAANS.

3. PROTOTYPE DESIGN AND DEVELOPMENT

There are four major parts in the prototype design and development section. The first part is the prototype conceptual approach, which is applying the suitable conceptual approaches for the E-Learning 3.0 CANS prototype. The second part is the prototype UI design, where it contains three phases of the UI design. This is followed by the prototype structural design part, where the prototype structure is separated into three components. The last part of this section emphasises building the prototype in the context of coding.

3.1 Prototype Conceptual Approach

The proposed notification system is designed and built based on the concepts and technologies of Web 3.0 and E-Learning 3.0. Those concepts and technologies are studied in order to gain a profound understanding of the knowledge, where those Web 3.0 and E-Learning 3.0 practices are discussed and analysed in a collective way. Furthermore, three different notification systems are analysed and evaluated as case studies for the proposed notification system. The reason for those three systems being selected is because they are developed by implementing relevant frameworks and technologies. This CANS also needs to be consistent with its relevant principles, while various technologies will be applied for the application to ensure it can achieve the desired functionalities.

3.2 Prototype User Interface Design

The notification UI design of the E-Learning 3.0 CANS prototype is in the second part of this section. The prototype website UI went through three design phases. In the first phase, there is a single UI called Notification Display Board (NDB) which is designed with functions: enable create, update, delete, edit and save actions on the five information segments for every notification at the first phase of the UI design.

Then, the original NDB page is replaced by a new webpage called Student Notification DashBoard (SND) in the second stage of the UI design to enhance the user experience of the prototype type. The SND is changed to be the main page of the CANS prototype. The SND page contains three parts, which include the Header section at the top, the Event Modification section on the left and the Event Display on the right. The Event Modification section enables the users to modify the notifications in the six different notification precincts in the Event Modification section. Those notifications precincts are personal, assignment, school, faculty, university and club.

More transformations are made in the third phase of the UI design. The layout of the SND webpage is changed into left and right. A fixed vertical navigation bar is on the left with seven tabs, where those first six tabs are linked to six corresponding notification webpages. Those six webpages correspond to the six notification precincts. Every notification row in those six pages provides edit, save and delete functions for notification, and the seventh tab on the navigation bar allows users to create the new notification on the related notification page (Figure 1).

Student Notification DashBoard	Date	Time	Event Content	Location	Type	Action
Personal Notification	01/03/2018	10:00	Employment on Campus	Terrace Room	Career	More...
Assessment Notification						Delete
School Notification	02/05/2018	20:00	Assessment due	On line	DECO7861	More...
Faculty Notification						Delete
University Notification	20/05/2018	11:00	Meeting	21 Holt Room	Personal	More...
Club Notification						Delete
New Notification	24/05/2018	12:00	University Conference	81-119	University	More...

Figure 1. Student Notification Dashboard

3.3 Prototype Module Structure

The proposed CANS prototype structure is made up of two parts, which are the Front-end and the Back-end. The Front-end is made up of the Notification Data Extraction (NDE) and CANS website. The client side displays all the notification information and operates the simple functionalities of the notification system. The Back-end of the CANS prototype is designed to provide the RESTful API and Database functions. Therefore, it is necessary for the structural design of the prototype to be sophisticated enough to meet best practice standards and requirements in order to satisfy the proposal. The structure of the E-Learning 3.0 CANS prototype is designed at the different phases of the prototype structural design.

Notification Data Extraction (NDE) Module is proposed to extract notification from relevant webpages of the UQ website and it needs to sort personalised notification information into six notification categories. Then the information is stored in the Back-end Database and can be displayed on the prototype webpages. The NDE module resolves tasks on the notification data extraction and formatting.

Prototype Website Module of E-Learning 3.0 CANS is designed with three sets of functional webpages, where each set of the webpages is designed with different UI illustrations and functions to achieve their distinct purposes.

The third section of the E-Learning 3.0 CANS prototype structure is the Representational State Transfer Application Program Interface (RESTful API) and Database. RESTful API operates within the Model-View-Control (MVC) architecture. The Model (M) can pull data from the Database without knowing the complexity of the Database, and it also provides an abstraction layer with the Database. The Controller (C) manipulates the information flow between the model and the view. It controls what data is retrieved from the Database and what data is sent to the view by programmed logic. It also receives information from the client and applies the logic by changing the view, or modifying the data through the view, or both (The Django Book, n.d.).

The last component of the CANS prototype is the Back-end Database. The Student Table has relations with the Faculty, School, Club and Course Table, which in turn, all have relationships with the Event Table. This Back-end database relation design can link all the notification data together, in order to realize the Linked Data concept.

3.4 Prototype Code

The last part of the E-Learning 3.0 CANS prototype development is building the prototype by coding. The NDB page and the initial SND page are both written by using HTML and CSS. In the later version of the prototype, all of the webpages are built using ReactJs and CSS. Then, the Notification Data Extraction module is written by using Python. The Web API is coded by using C# and the Back-end database if coded by using SQL.

4. PROTOTYPE OUTCOMES

The end product of the E-Learning 3.0 CANS prototype consists of four components each of which manages particular tasks based on their functionalities.

The final NDE program is written using Python language. It can achieve extraction on five categories of notifications from predefined webpages of the UQ website, and the prototype can save those notifications as Excel files and store them on the desktop computer. Those notification categories include the users' assignment, exam, school, faculty and university. Those assessment notifications are from four random Master of Information Technology courses.

The second component is the CANS prototype website (Figure 1), where all the webpages are built by using ReactJs and CSS. The first six webpages are the six respective notification pages on the left. The vertical navigation bar with six notification options is on the left, and those options are linked to the six corresponding notification webpages. The New Notification option is connected to create new a notification page. Every row of notification in the notification webpages directs to the More and Delete webpages. The More page links to full notification view and it allows Edit, Save and Back, while Delete connects to the delete notification page that enables delete.

The third component of the CANS prototype is the Web API, which is written by using C#. Web API is built to with data manipulation methods to enable the webpages function to modify the data stored in Back-end database.

The last module for the E-Learning 3.0 CANS prototype is the final Database module (Figure 2) and it is set up by using the MySQL Workbench. The Database is built with the different notification tables where they are interconnected. The connections form relationships among the linked tables; hence all the different notification data are linked and accessible. There are seven notification tables in total, each having at least two relationships with other tables.

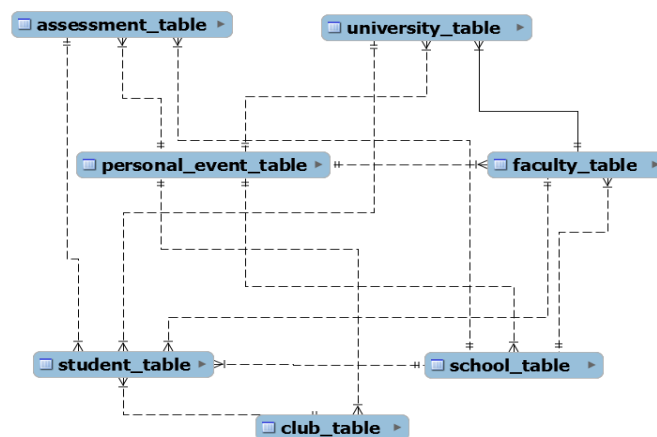


Figure 2. Database Structure and Relationship

5. PROTOTYPE EVALUATION APPROACH

For this project, the focus group is conducted at the Institute for Teaching and Learning Innovation (ITaLI) that is located at the UQ St Lucia Campus. Six students who are currently undertaking at least one course during the semester are investigated on the CANS prototype user experiences through focus group interview to evaluate the prototype. The project researcher carries out the focus group activity and collects feedback of the prototype from six UQ students between 3pm and 4pm. The response recording process is achieved by writing down participants' answers on paper as they speak. Taking notes as respondents speak allows the researcher to record complete and accurate feedback of the participants.

There are five questions for the focus group interview, and those questions are selected based on focusing on user experience of the E-Learning 3.0 CANS prototype on various aspects. Before the prototype user experience testing commences, the researcher provides a project introduction which allows the participant to have an overall understanding of this research. After that, the researcher presents a walkthrough based on three tasks on the prototype with the students, where each task focuses on a specific area in order to cover all aspects of the prototype. Afterwards, the researcher asks the students five questions in order to learn their user experiences on the prototype.

5.1 Prototype Walkthrough Tasks

Task 1: Check all types of notifications

This walkthrough task's goal is for the participants to determine whether the different notifications in the CANS website pages are accessible, perceivable and understandable for them.

Task 2: Check the complete notifications

This walkthrough task's goal is to show the participants the method of checking the complete details of the notifications and how to use the More button.

Task 3: Create, edit, save and delete a notification

The third walkthrough the on CANS website is to use the Create, Read, Update and Delete (CRUD) functions to assess whether the participants could easily find those functionalities and the effectiveness of those functions.

5.2 Focus Group Interview Evaluation

During the focus group session, the researcher asks the six students five open-ended questions relevant to the aspects of the CANS prototype after the walk through is completed:

1. Does the prototype do what it is supposed to do?
2. Do you think the prototype's design matches its purpose, and does the UI look good?
3. What prototype features are missing?
4. Does anything in the prototype seem out of place or unnecessary?
5. When you are exploring the product, does the prototype confuse you?

6. FOCUS GROUP RESULTS

6.1 Question 1: Does the Prototype do what it is supposed to do?

For this first question, there were three participants with similar answers. They considered that the E-Learning 3.0 CANS prototype was not effective enough in terms of notifying the UQ students on their related activities, as there should be a snapshot of the notification at the Student Dashboard of the UQ Blackboard, where the snapshot user interface shows important notification information. One of these students said that this extra UI can encourage students to check the NDB and be more organised. There was one interviewee concerned that the students could be overwhelmed by the notification. The participant also doubted the logic of the Database in terms of notification editing and updating. The notifications can be the notification category from Assessment to University. Other students thought that the prototype was functioning as they expected.

6.2 Question 2: Do you think the Prototype's Design Matches its Purpose, and does the UI Look Good?

The participants mainly had feedback with opposing opinions on the prototype user interface design of the E-Learning 3.0 CANS prototype. There were two students who thought that the SND user interface was not well designed, and both of those students suggested the notifications should be displayed in calendar view. One of the two participants recommended that the calendar view should be exchangeable with the current

notification list view. The other student opposed the notification list view design and the student thought the list was too long to scroll down and view.

From the previous question, there were students who provided the suggestions on the user interface design of notification snapshot mentioned above. Those focus group interviewees expressed that there should be a banner-like area in the Student Dashboard of the UQ Blackboard with notification reminders, where it could display prioritised notifications with the number of new notifications. One of the students suggested that there needed to be a home page where all the different categories of notifications should be in one list and ordered by date. The other participant also underlined the importance of the notification snapshot banner in terms of its colour and location choice. The person indicated the colour must be bright and distinctive, and the location must be easy to find.

Although there are many UI issues raised by the participants, they all had positive feedback on the notification information sorting. Participants agreed that the prototype organised the different event information so that they were neat, easy to read and understand. Two students responded that the prototype notifications were comprehensive and they did not need to obtain the information elsewhere, which was convenient for them.

6.3 Question 3: What Prototype Features are Missing?

The feedback for this question mainly focuses on the functional elements. There was one participant who wanted to view the notifications by date or by type. This respondent also recommended that the webpage links should also be provided for all the notifications in the school and faculty categories for accessibility. The interviewee also suggested choosing different colour backgrounds for the various types of notifications in the notification list, so that the notifications could be better distinguished from each other.

Another student also suggested notification filter by the notification type. This student recommended that the updates and edits on the notifications should be highlighted. The fourth participant would like to allow notifications prioritisation. An interviewee suggested that the prototype needs to prevent the users making mistakes with clear displays. This student also recommended adding a notification type for assessment grades.

6.4 Question 4: Does anything in the Prototype seem out of Place or Unnecessary?

This question prompted different responses. There was one student who thought it is not necessary for the faculty notification to be editable. Another student advised that the notifications from university are not very essential. One of the participants said the notifications for university and faculty could be combined into one.

6.5 Question 5: When you are Exploring the Product, does the Prototype confuse you?

For this question, all the focus group participants had similar responses that the E-Learning 3.0 CANS prototype was not confusing to them. The participants thought that the prototype organises the notification information so that it is neat and easy to interpret. In addition, they also suggested that the functions were not confusing to use.

7. DISCUSSION

The E-Learning 3.0 CANS prototype only achieves some of the proposed functions. First of all, ReactJS is effectively used because the CANS website is heavily data dependent. In addition, the prototype can provide notification based on users' learning information from different sources. Apart from that, it can notify university events to the users. In addition, the CANS prototype allows users to add new personal notifications. Furthermore, the CANS prototype organises and demonstrates the notifications in a systematic way.

In spite of this, many of the proposed functions cannot be implemented due to time and knowledge constraints. Those include: show empty time slots to an individual student, auto scheduling with personalised learning activities, auto learning information recognition and sorting and allow users to organise the notification based on their preferences. The four components of the prototype are not combined, and only the dummy notification data is used for the website.

Those functions cannot be implemented, because there is a large amount of new knowledge needed to accomplish them. The researcher needs a significant amount of time to learn and understand the techniques covered in the four components. It is outside the researcher's ability to learn and understand the knowledge mentioned above in the given time frame.

8. CONCLUSION

In this paper, there are various approaches and techniques used in order to design and to build an E-Learning 3.0 CANS for UQ Blackboard, as the current UQ Blackboard notification system is not effective. The actual prototype does not achieve all the purposed functions, but the initial idea is to develop the system with the Semantic, Context Aware and Linked Data techniques in order to assist the students to manage their study related activities better and improve their study experiences. The current E-Learning 3.0 CANS prototype provides a foundation for developing more advanced functionalities in order to meet these aims.

REFERENCES

- Amarin, N.Z. 2015, 'Web 3.0 and its Reflections on The Future of E-Learning', *Academic Journal of Science*, [online]. Vol. 4, no. 2, pp. 115–122. Available at: <https://www.zuj.edu.jo/wp-content/uploads/2014/05/web3.0.pdf>. [Accessed 10 Dec. 2018].
- Amelung, C. 2007, 'Using Social Context and E-Learner Identity as a Framework for an E-Learning Notification System', *International Journal on E-Learning*, [online]. Vol. 6, no. 4, pp. 501-517. Available at: <http://www.learntechlib.org.ezproxy.library.uq.edu.au/p/21786/>. [Accessed 31 Dec. 2018].
- Bradford, P., Porciello, M., Balkon, N., & Backus, D 2007, 'The Blackboard Learning System', *The Journal of Educational Technology Systems*, [online]. Vol. 35, pp.301-314. Available at: <http://uupinfo.org/research/working/bradford.pdf>. [Accessed 25 Dec. 2018].
- Dominic, M., Francis, S., & Pilomenraj, A. 2014, 'E-Learning in Web 3.0', *I.J. Modern Education and Computer Science*, [online]. Vol. 6, no. 2, pp. 8-14. Available at: <http://www.mecspress.org/ijmecs/ijmecs-v6-n2/IJMECS-V6-N2-2.pdf>. [Accessed 4 Dec. 2018]
- Haron, N. S., Saleem, N. S., Hasan, M. H., Ariffin, M. M., & Aziz, I. A 2010, 'A RFID-based Campus Context-Aware Notification System', *Journal of Computing*, [online]. Vol. 2, no. 3, pp. 122-129. Available at: <https://arxiv.org/ftp/arxiv/papers/1003/1003.4080.pdf>. [Accessed 5 Dec. 2018]
- Hussain, F 2013, 'E-Learning 3.0 = E-Learning 2.0 + Web 3.0?', *Journal of Research & Method in Education*, [online]. Vol. 3, no. 3, pp. 39-47. Available at: <http://www.iosrjournals.org/iosr-jrme/papers/Vol-3%20Issue-3/H0333947.pdf?id=7373>. [Accessed 17 Dec. 2018]
- Laskaris, J 2015, '20 Interesting Facts About eLearning [Infographic]', TalentLMS. Retrieved from <https://www.talentlms.com/blog/20-facts-elearning-infographic/>. [Accessed 24 Dec. 2018]
- Leonidis, A., Baryannis, G., Fafoutis, X., Korozi, M., Gazoni, N., Dimitriou, M., Koutsogiannaki, M. Boutsika, A., Papadakis, M., Papagiannakis, H., Tesseris, G., Voskakis, E., Bikakis, A. & Antoniou, G 2009, 'AlertMe: A Semantics-based Context-Aware Notification System', *33rd Annual IEEE International Computer Software and Applications Conference*, [online]. WA, USA: IEEE. Available at: http://eis.bristol.ac.uk/~xf14883/files/conf/2009_secasa_alertMe.pdf. [Accessed 29 Dec. 2018]
- Miranda, P., Isaias, P., & Costa, C. J 2014, 'E-Learning and Web Generations: Towards Web 3.0 and E-Learning 3.0', *International Proceedings of Economics Development and Research*, [online]. Vol. 81, pp. 92-103. Available at: <http://www.ipedr.com/vol81/015-ICERI2014-R00037.pdf>. [Accessed 6 Dec. 2018].
- The Django Book n.d., 'Django's Structure – A Heretic's Eye View'. The Django Book [online] Available at: <https://djangobook.com/mdj2-django-structure/> [Accessed 7 Dec. 2018].