Rapid Mobile Augmented Reality Prototyping with Unity, Vuforia, and 3D Modeling

Conference Paper · November 2018

DOI: 10.21125/iceri.2018.0873

3 authors:

Tuomo Muilu
Haaga-Helia ammattikorkeakoulu

Nguyen T. Nguyen
Haaga-Helia ammattikorkeakoulu

Amir Dirin
Haaga-Helia ammattikorkeakoulu

RAPID MOBILE AUGMENTED REALITY PROTOTYPING IN EDUCATION CONTEXT

Tuomo Muilu[1], Nguyen Nguyen[1], Amir Dirin[2]

[1,2]Business Information Technology, Haaga-Helia University of Applied Science, Helsinki Finland
[1] {tuomo.muilu@myy.haaga-helia.fi}
[1] {nguyen.nguyen3@myy.haaga-helia.fi}
[2]amir.dirin@haaga-helia.fi

Abstract

Mobile Augmented Reality (MAR) is getting popular among students. In this study we developed a MAR application to help the freshmen to learn about the university premises already before and during school starts. The Augmented Reality (AR) elements have been a trendy and user-engaging mobile technology in many famous applications such as Pokémon Go, Snapchat and Instagram. The MAR application concept is designed by applying mLUX framework, which is reported in different conference paper. In this study our focus is to report how we rapidly came up with a functional MAR application within a short period. For the rapid prototyping we apply Unity, Vuforia, and 3D modelling for augmented reality. After the application is implemented we conducted a functional, usability, and user experience test. The functional test indicates that the application implementation is based on initial specifications. Furthermore, the usability test indicates that the application is easy to use and efficient**.** These international applications have proven that users engage more with their surrounding via the applications by interacting and blending with the AR characters on their camera screen. Not only they come back to use the application but their experience with the AR elements becomes an emotional connection they have with the application as well.

Key Words: Mobile Augmented Reality, Unity, Usability evaluation

1 INTRODUCTION

In the Methodology section, we discuss our technical implementation of our concept. Using the Unity game engine with Vuforia AR SDK and IBM Watson SDK allows us to quickly create AR solutions. The application design is aimed to give Haaga-Helia's University of Applied Science, BITE programme's freshmen a new learning experience to their new university through AR technology. We built a framework and script in Unity and filled out the application's content, such as the AR character, speech-dialogue, and real-life school campus's 360-degree spheres. We created our own 3D models that were based on real-life teachers in the programme's campus and make their dialogue as close to the real-life character's personalities as possible. The digital solution was built for Android phones as well as web browser. Its content was created to showcase the school's facilities via interaction with the environment and performing a series of gamified user tasks, such as finding all the items, unlocking new maps, and playing mini games. In the Design section, we discuss our initial concept design for the application and the improvement we have made, including the AR elements. The application's beta version was uploaded to Google Play and deployed with a WEBGL version. The name of the application can be found on Google Play as "HH Freshman" with a logo of Haaga-Helia. The WEBGL version can be found at *https://goo.gl/*gk3qYN. Within the project's timeline, we have created a completed Unity application that serves as a demo for BITe's programme. We then conducted a new Usability Test to evaluate our usability and design as well as the application's content and features. The purpose of Usability Testing and Evaluation is to make sure our goal and our application serve the users what they really want and need. In the Conclusion section, we discuss about our conclusion on the project's work and its future, the application's implementation and its future improvement, and our further researching intention as well as an expanding point of view about mobile solutions for higher education.

2 METHODOLGY

The method that we have applied in this study comprise of two different parts

1. The application concept development: At this phase we applied mLUX framework for mobile learning application development [1]. mLUX framework is an iterative steps which is based on User Centered Design (UCD) principle. mLUX ensures the mobile application usability and user experience.

2. We applied qualitative data gathering presentation in application usability assessment.

2.1 Unity

Unity was chosen as the main implementation tool for this project, since it supports both two-dimensional and three-dimensional as well as augmented and virtual reality application development. Furthermore, the ability to build onto any platform from the same C++ codebase allows for prototyping of different operating system. For this project, an Android OS and a WEBGL version for desktop browsers was created. Unity's built-in UI tools allow for quick prototyping, since the Unity Editor is a visually-based interface with drag and drop functionality and its component add-on system allows for objects to receive properties of UI elements such as buttons without the need to code from scratch. Unity also allows unlimited customization of the Editor and provides an asset store with both Unity and community-made free and paid extensions, add-ons and resources. The assets used for this project are IBM Watson SDK for Unity by IBM, Lean Touch by Carlos Wilkes and an InsideOut shader found on Nicholas Dingle's YouTube video. The IBM Watson SDK allows developers to implement Watson's APIs into their application.[2] We used Watson's Speech-to-Text API in Freshman. Lean Touch allows users to move the player view with fingers or a mouse and includes common gestures found in modern applications.[3] The InsideOut shader allows placing a panorama picture to the inside of a 3D sphere, making a 360-degree experience.[4] Using ready-made assets found on the Unity Asset store greatly reduces the time required to develop applications.

2.2 Vuforia

Vuforia is a computer vision platform used for AR development and is integrated into newer versions of Unity. There are other AR platforms such as ARKit and ARCore, however, they are only for iOS and Android, respectively. Vuforia works with both iOS and Android, and to some extent, can be used with ARKit and ARCore as needed, therefore, ideal for rapid development as it uses the same code for both operating systems.

3 DESIGN

The initial project began in October 2017 when we chose to develop and create an innovative and digital solution for Haaga-Helia's Business Information Technology (BITe) programme. The objective of the project was to deliver a visualization of our concept with a demo of a hi-fi prototype. After identifying our users' needs, we proceeded to create our prototypes and conducted usability tests for the first prototypes to survey the feasibility and the feedback of test users. From the data we collected, our conclusion was that it was a viable conceptual solution and we continued to develop the prototypes into actual application since March 2018, with the approval from BITe programme's head director.

Figure 1. Screenshots of the initial AxureRP8 prototype

The initial prototype as shown in Figure 1, was made in Axure RP8 and was lacking AR elements. However, our feasibility testing on the prototype itself showed that the concept is more than welcome by users both experiencesnew and new.experienced users. The idea of interacting with the school's real environment and learning new things by playing is what makemade the concept becomes feasible and a unique solution in the higher education context. The user-interface and design were based on Haaga-Helia's colour palette with an intuitive and consistent usage of icons, symbols, and texts. We have made many changes related to the application's content, but the most significant change is the adding of an AR character as the Pasila Campus' guiding person.

4 IMPLEMENTATION

We conducted a survey asking students of BITe programme about their experience with the orientation week they had and the problems that occurred during their first semester. By surveying our potential users, we wanted to know what we could offer as a solution that benefits the students and the school. After our concept was finalized through a series of prototypes and feasibility testing, we proceeded to research for the right tools to implement the concept into a functional application. The development of the application was then tested for its usability. The usability testing process was to ensure that our application is in touch with the needs of our users.

To prepare for the creation of the application, developer accounts for Vuforia, IBM Cloud and QR Code Generator must be made to access their services. All three services have free accounts with the option to upgrade for extra features for a fee. The free accounts are enough to start developing. For Vuforia to place an AR object to the world, it needs a target. In the Vuforia Developer Portal, a development key and a target need to be created. For the HH Freshman application, only image targets were used to project the AR model. We used the QR code generated from the QR Code Generator website and added design around it, giving Vuforia more features to track for a more stable AR projection and used it as our image target. The image was then printed to be used for testing. Using the IBM Cloud account, we can create a project to access the Speech-To-Text service and receive the credentials required to use the API. Using the Speech-To-Text API, we can turn the users'users' voice commands into a string of text which can be used to command the AR character to do certain functions.

To create the AR character, we decided to use Adobe Fuse, as it had quick customization options as well as a direct upload to Mixamo, a webtool to rig and animate the character using their premade animations, compatible with Unity, saving many hours of work. We selected waving and talking animations for the character. Alternatively, a 3D model can be created from scratch or a community-made asset downloaded from Unity's Asset Store. The AR character was modeled based on BITe programme's academic advisor. For the AR character, we also recorded voice lines from the academic advisor to use as responses for the user. For the 360-degree tour, we used an iPhone SE on a tripod with the Google Street View application to photograph around the campus. For the panorama pictures to scale properly, they need to be around 2:1 ratio to width and height.

4.1 Steps for Creating the Freshman application

For the Freshman application, two scenes need to be created in Unity. One for the AR scene with an interactive AR character responding to voice commands and one scene to handle the 360-degree campus tour. Changing scenes prevents a single level becoming too big to affect the performance of the application as well as clearing unnecessary scripts running in the background.

4.1.1 AR Scene

The application license key needs to be added from the Vuforia Developer Portal and the image target database downloaded and activated in Unity's Vuforia configuration. The IBM Watson SDK can be imported at this point and the credentials for the Speech-To-Text API found on IBM Cloud needs be added to the ExampleStreaming script found with the Speech-To-Text example scene found in the SDK. We add a Vuforia image target to the scene and use the database of our downloaded and activated QR code. As a child of the image target, we add our 3D model of the academic advisor. We then modify the ExampleStreaming script to send the string received from the Speech-To-Text API to the script handling animations for the 3D avatar, bringing her to life. With this, whenever a user says something, the character picks up on the keywords and gives an appropriate response from the collection of voice lines and animations. A virtual button to change scenes as well as UI elements are added. Figure 2 shows the AR scene in play mode where the 3D avatar is created on top of the QR code and is waving and speaking in response to a “Hello”. When developing to mobile operating systems, audio clips should be added to separate empty objects with an Audio Source component instead of all added to one object, otherwise when building the application, the audio will not work.

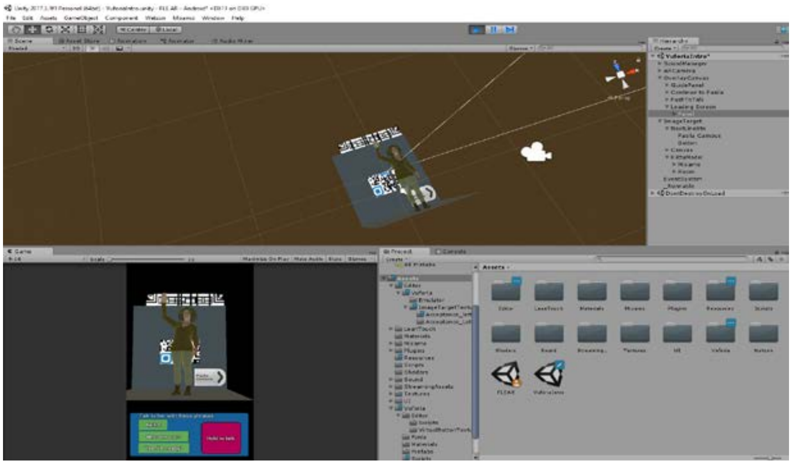

Figure 2: Final AR Scene in Unity

4.1.2 360-degree Campus Tour Scene

The next scene is the 360-degree campus tour. For this scene, a regular main camera is sufficient. To prevent Vuforia from running in the background, it should be disabled by adding a Vuforia Behaviour script to the main camera of the new scene and disabling the script. The 360-degree photos taken with the Google Street View app need to be placed on 3D spheres with the InsideOut shader. For each panorama picture, a sphere needs to be created. The Lean Touch asset is imported and used as the main tool to handle the camera. Travel between spheres is created using a script that moves the camera to the center of the next sphere when clicking on a interactable object. Icons and UI elements were created in Adobe Photoshop and added as interactable items inside the sphere that give valuable information and moves the user between spheres. Furthermore, game elements such as finding all the interactable items to complete a mission and unlocking new features of the app like quick navigation are added with scripts.

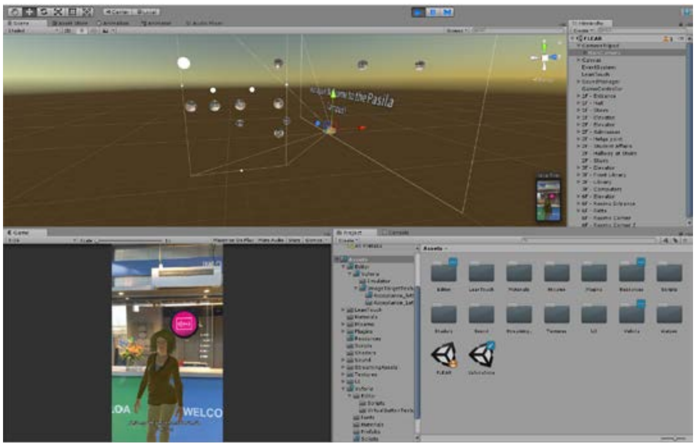

Figure 3: Final 360-degree scene in Unity

In Figure 3, the scene view shows multiple spheres on different levels on the y-axis, representing the different floors of the campus building. This is done because the camera retains its rotation when moving between spheres, so it is simpler to have them on the same plain and rotate the sphere’s, so they match the direction where the user moves than individually calculate and script in which rotation the camera should be when moving to a specific sphere.

4.2 Usability Test Plan

The plan was to test two different types of users, group A represents experienced users that come from BITe's programme, and group B represents new users that are individuals outside Haaga-Helia's UAS. There were 3 test participants for each group. The test location took place in Haaga-Helia's Pasila Campus from 24th to 26th of April. Each participant has given their permission to be recorded and studied during the testing session. There were case scenario and instructional document as well as the facilitators' help during the test session for the participants. A pre-test interview was conducted to collect their general information, such as gender, age, occupation, smartphone system, and their experience with Orientation week in their college. The facilitators also recorded videos and phones' screens as well as wrote memos during the participants' test. The test facilitators provided brief information of the test session's purpose and the application's general information. Users read their case scenarios and the instructional document. After that, users were asked to complete a series of mini tasks using the application as well as to express their thoughts out loud during the session. The session ended with users filling out a statement questionnaire in which they choose to agree or disagree with statements about the testing application.

4.2.1 Test-user Profile

Group A's profile consist of 3 participants, new users without prior knowledge about Haaga-Helia's BITe programme and have not previously visited the Pasila campus.

Table 1. Group A's user profile

User	Gender	Age	Smartphone OS	Experience with Orientation Week (Y/N, any technology used?)
1	M	28	Android	Y, social aspect was fun because you get to meet new people, but the information was lacking, and it took me 2 months before I stopped getting lost around the campus.
2	M	25	Android	N
3	M	26	iOS	N

Group B's profile: 3 participants, experience users that are BITe programme's students and have experienced the orientation week.

Table 2. Group B's user profile

User	Gender	Age	Smartphone OS	Experience with Orientation Week (1-5)
1	F	21	Android	4/5, social aspect is fun but too much information, could not remember all the information.
2	F	23	iOS	3/5, information sessions were boring, partly due to the mandatory participation.
3	M	26	iOS	5/5, but could be biased because compared to Australia's Orientation, here was much better.

4.2.2 Case Scenario and Instruction Document

The user's case scenario is a part of the testing session to give a storyline for users that they can imagine and put themselves into the context they are in.

“You are a new Haaga-Helia's Business Information Technology programme's student and you have received an acceptance letter from Haaga-Helia which contains this file called “Freshman App’s Guide”. You open the file. You read and follow the instruction in the file to download an app called HH Freshman. The app is part of your orientation and is a new way that Haaga-Helia adapted this year.”

The instruction document is called “Freshman App’s Guide” that includes a note about the application’s AR features, which is only available for Android phones. The help document also contains the step-by-step instruction for new users of what they need to do. The instruction document is needed because there are two types of users. One is the Android phone user and the other is the web-browser user. Some of the notable instructions in the document are the QR code for Android phone users and the website link for web-browser users.

4.2.3 User tasks

For Android phone testing, the test participants are asked to perform small tasks as below:

Task 1: Read the scenario + the Instructional file

Task 2: Scan the QR code in the file and download the app

Task 3: Open the App, scan the QR code to talk with Riitta

Task 4: Talk with Riitta and follow her guide to Pasila Campus

Task 5: Find all the voice memos and animation around Pasila Campus

Sub task 5.1: Floor 1: Find all voice memos and animation

Sub task 5.2: Floor 2: Find all voice memos and animation

Sub task 5.3: Floor 3: Find all voice memos and animation

Sub task 5: Floor 6: Find all voice memos and animation, give feedback

Task 6: Find the completion code

For Web Browser testing, the test participants are asked to perform task 1, task 5 and its sub tasks, and task 6 from the Android’s test tasks. The user tasks for both user groups are created based on the application’s user flow, which starts when user read their scenario and the instructional file. It is important that the case scenario and user tasks are as close to the real-life situation as possible in order to measure the usability level.

5 RESULTS

5.1 Usability Test Evaluation

5.1.1 Case scenario evaluation

Users took approximately about 1.5 minutes to read through the case scenario and the instructional document.

5.1.2 Time-task measurement

Group A’s Time-task measurement is with 1 Android user and 2 Web browser users. In general, the completion rate of the test participants is above 80%. Among the total 6 user tasks, test participants on an average completed 5 tasks. The tasks that have the highest rate of completion are task 1: ”Read Scenario and instructional file”; sub task 5.3: ”3rd Floor”; sub task 5.4: ”6th Floor”; and task 6: ”Find the completion code”. There are tasks that have significant low completion rate which is sub task 5.1: ”1st Floor” and sub task 5.2: ”2nd Floor”. The reason is that when users first entered a new environment that they can move around in 360-degree, users get excited as well as slightly confused about the new place they are unknown of. It took all users the longest time on sub task 5.1 and sub task 5.2 because they needed to learn about their surrounding and familiarize themselves with what is happening. User 2 and User 3 had a hard time completing these tasks because they feel dizzy from looking around. As the results, User 2 only completed 25% of sub task 5.1 and 35% of sub task 5.2. Another reason for that is the animated ring that is used as a moving-forward function. User 2 said that the opacity and colour of the animate ring should be more significant with a word like “Go here!” or symbol like an arrow to indicate that user can explore further in this part. For User 3, he did not complete sub task 5.2 at all because of the design of a shortcut map button. The shortcut map button is placed at the far-right bottom of the screen; therefore, User 3 did not notice the option to go from one floor to another floor via this function. He managed to use the elevator board to move around, however, the elevator board design has small numbers on it. On average, the total time to finish the tasks are 9 minutes. User 1 is the only user out of three users that completed 100% of every task. However, given the case that he was testing the Android version, which was more intuitive to use his fingers on touch screen, it was easier for him to navigate throughout the application.

Table 3. Group A's Time-task measurement

Task\User	**U1 (with Android)**		**U2**		**U3**	
	Time	Completion	Time	Completion	Time	Completion
Task 1	0,85	100 %	1	100 %	2	100 %
Task 2	1	100 %	**The user was testing the web browser version**			
Task 3	0,15	100 %	**The user was testing the web browser version**			
Task 4	1,42	100 %	**The user was testing the web browser version**			
Task 5						
Task 5.1	2,17	100 %	1	25 %	3,5	100 %
Task 5.2	2,05	100 %	0,5	35 %	0	0 %
Task 5.3	0,95	100 %	1	100 %	3	100 %
Task 5.4	0,8	100 %	1,5	100 %	2,3	100 %
Task 6	0,67	100 %	0,2	100 %	0,75	100 %
Sum	**10,06**	**100 %**	**5,2**	**77 %**	**11,55**	**83 %**
The average time-task measurement and completion rate: **9 minutes with a completion rate of 87 %**						

Group B's Time-task measurement is with all 3 users as Android users. Group B's test participants are students from BITe, therefore, they are somewhat familiar with the environment. With that condition, the average completion rate is much higher than Group A, at over 98%. The only task that test users could not complete entirely is Task 4:"Talk with Riitta and follow her guide to Pasila Campus". The Task required users to speak with Riitta - the AR character, and listen to her guide that would lead the user to Pasila Campus. However, due to the inconsistency of the technical implementation at the time, User 1 and User 2 had to ask for help from us to talk with the character as well as to enter the Pasila Campus. The instructional dialogue that we used for Riitta was also partly making the task harder, because it was misleading, User 1 and User 2 did not know how to enter Pasila Campus. Only when they asked for help, they were able to proceed. Among the total of 9 tasks, including the sub-tasks, test participants from Group B on average completed 8 tasks. There was the same comment from User 2 and User 3 that after looking around in a 360-degree view environment while sitting down, they started to feel slight dizzy. The case with shortcut map button was much better than Group A, as soon as users learned about the shortcut map, they proceeded to use it to move around faster. There was longer completion times in Group B compared to Group A. Because the users were excited and as they were encouraged to say out loud, they took on a slower space to check on their campus. There is similarity in the length to complete sub task 5.1 and sub task 5.2 in both groups, which can be their reaction to an entirely new interactive world as well as the familiarization process to a responsive environment. In sub task 5.1, users were introduced with pieces of information as well as unlocked items while they wandered around the floor, thus built up their excitement on what is next.

Table 4. Group B's Time-task measurement

Task\User	**U1**		**U2**		**U3**	
	Time	Completion	Time	Completion	Time	Completion
Task 1	0,75	100 %	0,75	100 %	0,5	100 %
Task 2	2	100 %	2	100 %	0,5	100 %
Task 3	1	100 %	0,75	100 %	0,75	100 %
Task 4	1,50	80 %	1,50	90 %	1	100 %
Task 5						
Task 5.1	3	100 %	2,5	100 %	3	80 %
Task 5.2	2	100 %	1,5	100 %	3	100 %
Task 5.3	2	100 %	1	100 %	2,75	100 %
Task 5.4	2	100 %	1,25	100 %	1,75	100 %
Task 6	0,25	100 %	0,5	100 %	0,75	100 %
Sum	**14,50**	**98 %**	**11,75**	**99 %**	**14**	**98 %**
The average time-task measurement and completion rate **13 minutes with a completion rate of 98%**						

5.1.3 Usability Heuristic Evaluation

Heuristic Evaluation is the usability evaluation that test participants are given at the end of the test session to assess the application's performance.[5] The Heuristic Evaluation includes 10 usability heuristic statements about the application and users can give their agreement level about these statements. The scale of agreement level is from 1 to 5 with 1 (disagree), 2 (somewhat disagree), 3 (neutral/no comment), 4 (somewhat agree), and 5 (agree). In Group A's Evaluation, there are 2 out of 3 users that have high agreeing levels with how the application is performing, with 4.4 from User 1 and 3.7 from User 2. The highest agreeing level are about statements of the match between system and the real world; the consistency and standards; the aesthetic and minimalist design, the help for users' recognition, diagnose, and recovery from errors; and the help and documentation. In the table 5, the green colour cells are the indicated of positivity from the users regarding the system's usability.

Group B's Heuristic Evaluation is with 3 Android users. The Heuristic Evaluation was given out at the end of the test session. Test participants were asked to give their opinion about 10 statements relating to the application's usability. There were higher agreeing levels toward the system's performance. The green colour cells in the table indicated that 8 out of 10 heuristic statements are agreeable. All the users' average agreeing level are well above with the lowest level from User 3 as 3.9. However, there are also statements about the user control and freedom as well as the aesthetic and minimalist design that need to be assessed internally. User 3 suggested that we should bring more informative boxes to help users get to know the campus even better. The user's control and freedom can also be improved by creating more indicative navigation symbols like the elevator panel.

Table 5. Usability Heuristic Evaluation

Heuristic\User	**Group A**			**Average Agreeing Level**		**Group B**			**Average Agreeing Level**
	U1	**U2**	**U3**			**U1**	**U2**	**U3**	
Visibility of system status	4	4	2	3,3		5	4	4	4,33
Match between system and the real world	5	3	3	3,7		5	5	5	5,00
User control and freedom	3	4	3	3,3		5	4	2	3,67
Consistency and standards	5	4	4	4,3		4	5	4	4,33
Error prevention	5	3	3	3,7		5	4	3	4,00
Recognition rather than recall	4	3	2	3,0		5	5	5	5,00
Flexibility and efficiency of use	3	3	3	3,0		5	5	5	5,00
Aesthetic and minimalist design	5	5	4	4,7		4	4	3	3,67
Help users recognize, diagnose, and recover errors	5	4	2	3,7		5	5	3	4,33
Help and documentation	5	4	3	4,0		4	5	5	4,67

5.1.4 Open feedback from Test Participants

The test participants were asked to express their thoughts and feelings regarding their experience as well as their overall feedback about the application's concept. All users had given feedback about their experience in the application as "It was Awesome/Amazing!" (User 1 and User 2 from Group B), "I'm glad I get a chance to know about the campus before I arrive" (User 3 from Group B and User 1 from Group A), "It was okay, I now know something I didn't know before" (User 2 and 3 from group A). These feedbacks are from all test users:

"It is a helpful and realistic concept. I can imagine people use this to learn about their school not only when they are freshman but as a come-back application that they can remind themselves of useful information". "It's good to know something about the school and it shows the real environment of a school. It is easy to use when user moves the mouse or finger, it feels intuitive and the environment responds to the users' actions. However, the image of the avatar is a bit scary and not friendly for people who do not know who Riitta is." "It was good that I know something about the school and it shows the real environment of a school. However, the map button need to be changed because I didn't notice it at all and tend to forget that it is there for me. The animated ring/rolling sign is confusing to me because there is no indication that I can move forward or go there with that ring. The elevator part is also confusing because some of the buttons are not working and make me think it was broken." "I like it because of the game aspect and it is more exciting than the orientation week I had. I'd prefer this game than having to go to the orientation week only. I really like the AR character that I can talk to. I have never used and seen the kind of app like this so it impresses me." "It was so cool! I like Riitta the most! It is really useful for the freshmen for the library and the student affair. I wasn't able to find the student affair when I was a freshman. I'd prefer to do this virtual tour, it was so much fun. It was kind of dizzy moving around but it is a common problem using 360-view. Everything looks real to me and in a good scale." "Nice concept and will be benefiting to people. It is very intuitive that you can use your fingers. There is a nice animated ring that is really nice design. The elevator buttons should be more indicative for freshman. The freshman would feel closer to the campus when they use the app, they will think that their future school is cool and get excited."

6 DISCUSSIONS

The web version indeed has a setback compared to its Android version, because users use their mouse to move around and the wider computer screen can easily create a motion sickness from interacting with 360-degree view while sitting down. Moreover, the Android version gave users an experience of interacting with an AR character through their camera, thus increased their excitement, positive emotions and engaged them to their storyline. However, the Android version has a drawback on the completion of the 360-degree virtual tour. When users unlock the new map upon tour completion, the next map is only available for web browser. Android users will have to switch their devices to play the next map's games. On the other hand, web browser users will be transferred directly to a new window that allow them to play the next map's games immediately. This drawback will be modified and changed so that both user groups are satisfied. Another setback in Android version is the dialogue of the AR character during the camera interaction. The Speech-To-Text technology can create both excitement if the AR character responds successfully and frustration if the character stays silent. Adding small improvements to the responsiveness and interaction of the AR character prevents users from being discouraged. The future development for the application can be the improvement of the user-interface aspect as well as the extension of the application's content to other degree programs.

7 CONCLUSION

Using Unity to develop an augmented reality application is relatively easy with no prior knowledge of the Unity. Vuforia allows a very streamlined way of developing AR content. The Watson SDK for Unity is daunting at first, however, reading the documentation carefully will clear up misunderstandings. There is also documentation and many YouTube tutorials for creating various AR content using Vuforia, ARCore and ARKit, therefore, it's feasible for educational facilities to implement their own solutions in their educational content without the need to invest in AR professionals.

The application's concept created excitement throughout users' experience during the test session, therefore, users feel that they learn about their school without force. The results suggest that augmented reality in education context is approachable and welcomed by students. Integrating modern technology into daily school life piques curiosity among the students and make learning a subject more fun. The nature of self-learning and exploration is encouraged within a safe and responsive environment where they have full control of their actions. The results indicate that the development of this application has proven that the digital solution is serving the users' needs.

ACKNOWLEDGEMENTS

We wish to acknowledge the help of BITe's programme's director board, Mr. Jari Hyrkäs for approving the project's development and created opportunities for us to complete our research. We are also thankful of Ms. Riitta Blomster for her contribution on the content of the application as well as the AR modelling process.

REFERENCES

[1] A. Dirin and M. Nieminen, "mLUX: Usability and User Experience Development Framework for M-Learning," *Int. J. Interact. Mob. Technol.*, vol. 9, no. 3, pp. 37–51, 2015.

[2] IBM Watson APIs. 2018. Unity SDK to use IBM Watson services. https://github.com/watson-developer-cloud/unity-sdk

[3] Wilkes, Carlos. 2018. Lean Touch. https://assetstore.unity.com/packages/tools/input-management/lean-touch-30111

[4] Dingle, Nicholas. 2017. [Tutorial] Unity3d – 360 Virtual Tour. https://www.youtube.com/watch?v=2q9wc9Y35wg Last accessed: 8.5.2018

[5] Nielsen, Jacob. 1995. 10 Usability Heuristics for User Interface Design. https://www.nngroup.com/articles/ten-usability-heuristics/