Hands-on experience of developing WebVR applications in classroom: From learner's perspectives

Vinh T. Nguyen
Computer Science Department, Texas Tech University, 2500 Broadway
Lubbock, Texas 79409, United States
vinh.nguyen@ttu.edu
http://myweb.ttu.edu/vinhtngu

Rebecca Hite
Department of Curriculum and Instruction, Texas Tech University, 2500 Broadway
Lubbock, Texas 79409, United States
rebecca.hite@ttu.edu
http://www.depts.ttu.edu/education/our − people/Faculty/rebecca_hite.php

Tommy Dang
Computer Science Department, Texas Tech University, 2500 Broadway
Lubbock, Texas 79409, United States
tommy.dang@ttu.edu
http://myweb.ttu.edu/tnhondan

Virtual Reality (VR) content advancement tools are in persistent use by both eager specialists and software development companies to create new and energizing VR settings. Thus, software engineering understudies may benefit by partaking in this improvement, for learning indispensable programming abilities, as well as to build up their innovativeness through a coordinated effort. Web-based VR (WebVR) has risen as an autonomous stage structure that licenses students to make rich and intuitive VR applications. However, the achievement of WebVR depends on understudies technological acceptance, the intersectionality of a users apparent utility of and convenience with said innovation. So as to decide the viability of the rising WebVR tool for software engineering understudies of differed encounter levels, this paper exhibits a contextual investigation of 38 undergraduate students entrusted with building up their fantasy house utilizing WebVR and compares WebVR with other VR technology. Results demonstrated that students showed technological acceptance by not just learning and executing WebVR in a brief span (one month), but at the same time were equipped for exhibiting inventiveness, critical thinking aptitudes and foundation knowledge for further study. Results just as suggestions, lessons learned and further research are addressed.

Keywords: Web-based virtual reality, technology acceptance model, course design, student creativity, A-frame

1. Introduction

Virtual Reality (VR) and Augmented Reality (AR) have turned out to be the two universally realized advancements utilized over an assortment of spaces, from exploration [1], medication [2, 3], instruction business [4], craft, design [5] to the military [6]. The Camry Web Configurator[a] has won the Visionary Award at the Summit Awards for Emerging Media due to the flexible car configuration using WebVR technology. From these examples, users can access VR/AR from a "*user*" perspective rather than a "*designer/creator*". There are also several studies approaching VR/AR from a design point of view, but mainly focused on developing 3D models without mentioning "sharing" and "easy of use". In the recent online discussion with faculty from Earth and Environmental Sciences fields in Lab Innovations with Technology at The California State University, instructors have pointed out that the biggest difficulty in conveying knowledge in the new age of technology is the lack of skilled IT personnel to create 3D contents in teaching and sharing information online. This need is also necessary for their students. They all have ideas, contents but do not know how to build systems, especially in limited time, hardware affordance and knowledge of programming.

The above problem raises the questions: Is there any VR/AR development framework/library that meets the above criteria? (ease of use, sharing, and not too dependent on hardware). If the library exists, how easy is it to be used and how it meets users' expectation? What are the pros and cons of using libraries? To solve a more complex problem, is the library prioritized for use? Answering these questions play an important role in contributing to improving the quality of teaching and learning for teachers and learners, especially from the perspective of a new designer/creator as pointed out by Psotka [7] "Teachers and trainers need to be exposed to VR in multiple ways so that they can begin preparing themselves and their institutions for future changes" and "... to give educators ..a better view of the strengths and weaknesses of these environments; ... documenting effectiveness and formative evaluations."

Same idea, already paraphrased revised and extend needed: From a learning point of view, coupling the affordances of VR advances and student agency and possession through their production of classroom exercises [8] could likewise be transformative in K-12 education teaching and learning. In this way, there is a transparent needs to determine aspects of frameworks that not exclusively empowers users to make their own VR contents anyway allows them to aptitude and offer their work with various learners over the world. Such a framework should address users' issues of utility and simple use, to encourage their acknowledgement according to the Technological Acceptance Model (TAM) [9]. At the point when convenience and utility work inside a framework, that structure could likewise be worthy to a broad audience of researchers and learners, independent of their experience and skill such as engineers, developers, designers, instructors, markers, artists, and specialists.

Same idea, already paraphrased revised and extend needed: From a teaching point of view, there is likewise a need guideline for choosing a reasonable framework for students to utilize while making VR contents in the classroom. Previous stud-

[a]https://customise.toyota.com.au/camry/

ies [10–12] concentrated on the utilization of 3D professional software, such as CAD drawing and Unity 3D, that require prior knowledge and some mastery of programming and design. Along these lines, the main part of educative user programming is with advanced engineering students. Yet, research suggests younger learners benefit from experiences with software development and programming in enhancing their psychological understandings of substance and inventive skills [13, 14].

In recent years, WebVR technology is developing rapidly due to the popularity of smart handsets such as mobile phones or tablet with built-in internet browser. A series of libraries/frameworks for WebVR development have appeared such as React VR, Three.js, Vizor, Babylon.js, PlayCanVas, etc to help users create 3D application to their interest. These libraries/frameworks are constantly updated to patch or add new features, the addition of these features requires users to understand how to use them based on available platforms. This will lead to certain difficulties for users when the system becomes cumbersome and difficult to control, especially for new users. To alleviate this problem, programmers have created more libraries (i.e., A-frame) based on the existing framework but are easier to use, shorten programming time and easier to learn for new learners.

In this revised and extended version of [15], we seek to address the previous research questions by having a complete VR course. The extended version of this paper will add 2 more use cases. The research content of each use case is presented briefly as follows:

- **Use Case 1**: Build a dream house using WebVR framework/library: This use case seeks to address the questions: What are the WebVR framework/library that students are interested in? Does the chosen framework meet students' expectation? What are the challenges faced during the development? Is there any suggestion for further development?.
- **Use Case 2**: Create an application with a freedom of choosing any VR/AR framework/library in a given topic: This use case seeks to address the questions: What is the VR/AR framework/library that the majority of students used for a more advanced requirement? What is the role of WebVR knowledge in the second Use Case. If students did not use WebVR library to solve the requirement, then what are the reasons?.
- **Use Case 3**: Create an application with a freedom of choosing any VR/AR framework/library in *any* topic: Research questions for this use case include: What topic are students interested in? What did VR/AR libraries they use when experienced a variety of frameworks?.

Unlike other studies, the main contribution of this paper is experience, VR development feedback from the perspective of new users who explore, learn and develop by a three-types study [16] (self-study, independent from others and with support from instructors) rather than following a traditional teacher's instruction. We believe this synergy will be an important document for learners and teachers who love VR/AR in selecting, designing and developing coursework materials on appropriate libraries.

The following paper is organized as follows: summarizing related research in Section 2; presenting the study design in Section 3; Section ?? analyzes the results from students' perspectives; lessons learned and challenges are described in Section 5.5; recommendations are provided in Section ?? and finally, conclusion and future research direction are concluded in Section 6.

2. Related Work

The VR concept was not new and it first was introduced in the early nineteenth century by a French general painter Louis-Francois Lejeune [17], the teaching of VR concepts in classroom was conducted by Bell [18] over implementing VR applications. This was impacted by the absence of VR equipment (costly at the time) and thus traditional desktop computers were utilized for practice. This paper does not mean to review all current VR work, rather concentrating exclusively on instructive methodologies.

In education, the implementation of VR/AR as a structural guideline was introduced in [12] to leverage existing libraries and frameworks. Miyata et al. [19] focused more on personal skills and group work performance by engaging students in creating VR application cooperatively. Throughout the course, the study showed that students' creativity and knowledge were gradually increased. Another similar study to Miyata et al. work was performed by Häfner et al. [20] where students were engaged in developing a 3-year industrial VR project with an additional finding in task specification which played the vital role for the success of the final VR product. A more in-depth study was conducted by Takala et al. [21] where the authors shared their teaching experience of VR development courses over a span of five years.

Stansfield [22] presented a VR course that combined both a traditional teaching lecture-based approach and a hands-on experience section. The author argued that the VR course would provide more experience for a capstone project regarding multidisciplinary fields, communication skills, research skills and presentation skills needed onto the success after graduation.

Several efforts have been made to bring VR into classroom, from concepts [18], content [23], and hands-on learning experiences [19, 24] to integrate VR contents into a computer graphics course [25]. The challenges and issues found in most researches are the accessibility of HMDs, a standard hardware platform for VR, just as the multifaceted requirements for programming knowledge and specialized libraries for developing VR content. Numerous reasons cause these issues, similar to those evaluated here, because of the expense of the VR headset for testing; the compatibility of the application on the deployed gadgets, and the expert abilities required for making 3D, VR content.

This paper endeavors to mitigate on these current issues when helping designers and learners in creating VR content in an academic setting. In particular, this paper furnishes an understanding into these issues with a novel instructive methodology which will be introduced in Section 3.2.

3. Methodology

This study seeks to answer the four main research questions:

- **R1**: What is the the typical VR/AR development framework/library that allows novice users to stimulate their interest by creating and sharing VR/AR content with ease?.
- **R2**: Given an answer from the previous question, does this library/framework prioritized to use to solve a more complex problem?
- **R3**: Upon releasing from any constraint, which library/framework does students use to develop a VR/AR application of their interest?
- **R4**: What are pros and cons of WebVR compared to app-based VR/AR from learners' perspective?

3.1. *Goal and objectives*

Based on the previous research questions, the goal of this study was to evaluate the feasibility of learning, developing a WebVR application and then pinpoint the most suitable framework/library for new eager users in inter-disciplinary fields as well as compare the pros and cons of WebVR to other app-based VR/AR applications. To meet this goal, our research approach held three objectives:

- to help students actively engage in the learning process when developing AR/VR applications;
- to understand students preferences on current typical VR/AR libraries;
- and to collect and analyze students' experiences when developing a VR/AR application .

3.2. *Study design*

In order to meet these objectives, this study was designed such that students can complete their projects within a given time frame to meet their performance objective. This was a challenging task since the majority of students did not have a basic background on 3D computer graphic and design. Learning curve among students is another critical issue because not all of them had the same learning speed, especially when they came from multiple fields and different graduate levels. To address the mentioned problem, an instructing method should be cautiously structure so that it can enhance learning performance and mitigate the learning curve among learners. In our approach when structuring the course, student engagement and inspiration are the most basic elements adding to their learning performance. Of the accessible teaching methods [26] (e.g., inquiry-based, situated-based, task-based, project-based, studio-based), studio-based learning is the most reasonable methodology that meets our teaching strategy. In this study design, learners are strongly engaged in three-projects development. We investigate each project through a use case, result for each use case will partially answer the research questions, and when they are combined the entire questions will be addressed. The entire course activity is shown in Fig. 1. Detail on each activity in each activity is presented in the following sub-section.

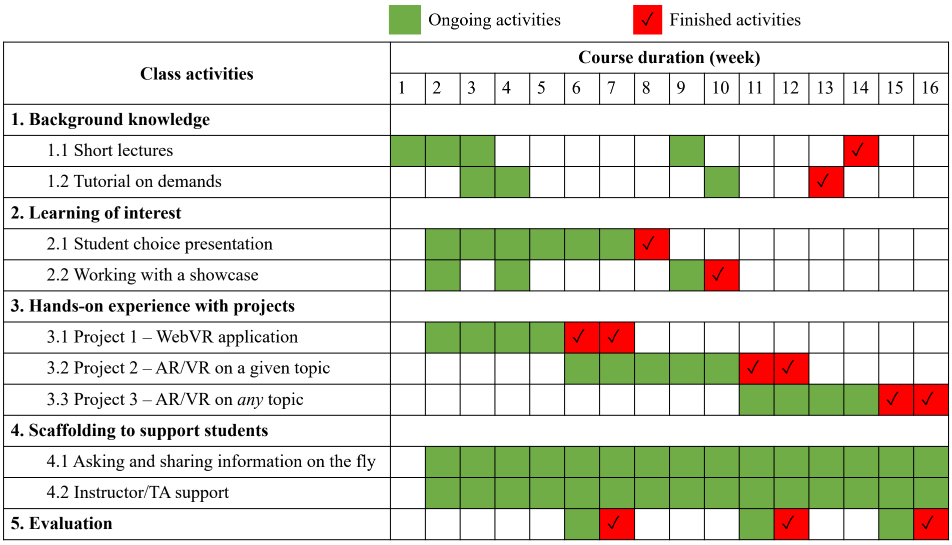

Fig. 1. The study design of 16-week long activities.

3.2.1. *Background knowledge*

Short lectures/tutorials are given toward the start of the course to provide an overview of VR/AR, current available VR/AR development tools (e.g., ThreeJS, Unity3D, Vuforia) [27,28], and principles of visual design. Tutorials are given when needed or on demand (e.g., read, process and display data or showcase of an application).

3.2.2. *Learning of interest*

A 5-minute in-class talk (student choice presentation) was provided by each student on a given topic of student interest. Each student was required to present a 5-minute in-class talk on a given VR topic. The topics for presentation were suggested by the instructor or by student's choice. The reason for this short talk was for all students to independently acquaint themselves with VR-based new advances and imparting this new information to the peers. The discussed content should include the following but not limited to information such as an overview of a given hardware/software/library, pros and cons, and its impact on the emotional sickness.

Along with the exploration of new emerging VR/AR topics, learners have an opportunity to experience and immerse themselves in a virtual environment by playing with a small VR/AR showcase (a virtual reality room). This sample project allows users to build a basic understanding of HTML and JavaScript from a scratch.

3.2.3. *Hands-on experience with projects*

Working with projects is the main component of the course activity whereas students are engaged in working with three projects. The first project topic is to build a fantasy or a 'dream' house on web-based VR framework derived the idea of the sample project. The purpose of this project was to help students understand the theoretical of 3D modeling. In the second project, a topic is given (i.e., water management) in which learners start going deeper beyond a simple 3D model by exploring a suitable library/framework that can help them solve the problem. When students gathered enough knowledge on VR/AR, the third project enables them to develop a VR/AR application on any topic, fostering student thinking and creativity.

3.2.4. *Scaffolding to support students*

The instructor and the teaching assistant(TA) are working as mentors to support students in learning. The communication channels between mentors and learners can be established by a means of face-to-face talk, online discussion forum (i.e., Piazza) or by email. Studio-based learning approach focuses more on self-teaching and learning and hence we consider these channels play a crucial role in our course.

3.2.5. *Evaluation*

In this revised version, we conducted three additional assessments (two evaluations for Project 2, 3 respectively, and one for the whole course) for gathering students' feedback of their difficulties and interest. Data collection was performed via an online survey tool (i.e. Google Form), and all data was consolidated and saved in the Excel format. Each assessment was conducted right after one project was delivered to avoid being forgotten as it may happen at the end of the course. In the first project, the survey questions most likely try to explore students choice on available library/framework for a Web-based VR application and how they approach to solve a simple problem or Project 1, in which we name it 'learning from an individual element'. In Project 2, a peer-review survey is conducted for each individual project by each student. Because this study tries to look VR/AR from students perspectives, so getting their opinion for all part of the project is essential. The survey for Project 3 is similar to one in Project 2. Evaluation survey for the whole course was conducted at the final day of the course where overall information is collected. The face validity of the questionnaire was evaluated and revised by the instructor and the TA, to ensure that all questions were easy to understand and inclusive of students' current knowledge of VR development. Questions are based upon perceived utility and ease of use of using technology, which are the major constructs of the validated Technological Acceptance Model (TAM) theory [9]. Question reliability was performed by both the instructor and the TA to maintain all responses were within the range of possible answers.

Ethical concerns in survey data collection: Before taking the survey, students were informed about the research and assured that their private information would remain confidential in both data collection and analysis. After students submitted their surveys, the data were copied to Excel for analysis. Data was de-identified and encoded by converting categorical text data to number as well as shortening open-ended answers.

3.3. *Participants*

Participants in this study are students enrolled in the virtual reality course at the Computer Science department of a university. There is a total of 38 students (as shown in Table 3.3). The average general programming experience is 3.5 years. This computer science course was comprised of university students who held general programming skills as an expectation for enrolling in the class. This course also encouraged students in other majors other than computer science, so sampled students represented inter-disciplinary subjects.

Table 1. Participant distribution by *gender* vs *graduate level*

	Undergraduate	Master	PhD
Male	28	2	2
Female	5	1	0
Sum	33	3	0

4. Use Case Study

4.1. *Use Case 1: Developing a WebVR application*

4.1.1. *Project description and deliveries*

In order to answer the first research question (R1), students were engaged on the first project whose purpose is to have students has the sense of virtual reality by exploring current WebVR library and creating a simple web-based VR application. The project's topic was to build a VR "dream" house in which a user can experience and interact with 3D objects in the scene. Students have the freedom to develop the application in any programming language, library or software. The project's requirement is the ability to navigate and perform simple interactions with several object models by using both computers and hand-held devices (tablet or smartphone) via an internet browser. The result of their performance would be a useful guide for new learners, instructors, or researchers conducting similar studies.

Project delivery Completed VR applications were presented in the classroom. Students presented their VR dream house projects using a desktop computer connecting to a projector and WebVR. Interactions in the scene were performed by using a mouse and keyboard. Students could also view the VR house with their laptops, tablets, smartphones, or immerse into the virtual world with an HMD (i.e. Google Cardboard headset). When users were immersed in VR environment, moving around was controlled by pressing the button at the top right of the Google Cardboard, whereas other interactions (e.g. turning on/off lights or triggering blinds) were triggered by gazing at the objects within a certain amount of time (i.e., 3 seconds).

4.1.2. *Project assessment*

The project was developed in one month and students presented their work in the classroom. A report of how they carried out their tasks, created models or sourced models (obtained from the open sources) was also submitted for evaluation. To evaluate students' projects, three criteria/requirements were used, where each level gradually increased in difficulty. The purpose of categorizing the requirements into three levels was to evaluate the completeness of the tasks within a short time. Overall, the levels are summarized as follows:

- **Level 1:** Students are able to create a house model from scratch or customize an existing model; make a 3D simple object or use at least 10 free 3D models, and light up the house.
- **Level 2:** Students are able to import more 5, 3D models to the scene without affecting the performance of the application, make some simple adjustments such as to control the light of the house (for example turn on/off, increase/decrease the intensity of the light, or switch to different color themes), and navigate around the house.
- **Level 3:** Students are able to interact with a certain object (e.g. drag/move/scale, run animation, or close/open doors), or display dynamic/movable objects in the house.

4.1.3. *Survey*

To gather data for answering the research question, a post-activity survey was conducted to ask students their opinions of creating a WebVR application. The questions were informed by the two constructs of the Technology Acceptance Model:

perceived ease of use and perceived usefulness. The survey consisted of 10 items which is shown in Fig. 2:

Question number	Question	Question type	Research question(s)
Q1	Which library/framework did you use to do your project? (e.g., Three.js, Unity, Babylon…).	Open-ended	R1
Q2	How did you find your library/framework?	Single choice	R1
Q3	To what extend did you find your library to use on the scale 1 (easy) – 10 (very difficult)?	Likert scale	R2
Q4	Did you find your chosen library meet your expectation to finish the project on the scale from 1 (No) to 10 (fully met the expectation)?	Likert scale	R2
Q5	To what minimum required level of programming (you think) to learn and use your chosen library?	Single choice	R2
Q6	How did you get the models for your project?	Multiple choice	R1, R2
Q7	How did you learn about the library that you used for project?	Multiple choice	R2
Q8	How many days that you spent for your project (including time to study your library and programming language)?	Open-ended	R2, R3
Q9	How did you rate the difficulty of the project on the scale 1-10?	Likert scale	R2, R3, R4
Q10	How did you rate the usefulness (for your learning of building WebVR) of project on the scale 1 – 10?	Likert scale	R2

Fig. 2. Research questions for the survey.

4.1.4. *Project results and reports*

Along with presenting the projects in the classroom, students are required to submit a report describing their WebVR applications, including sources of references (for downloaded assets), challenges, lessons learned.

At the end of the one month of project implementation period, 37 total houses were created. Fig. 3 provides a snapshot of 35 WebVR applications created by students. We show an enlarged example of a WebVR house on the moon in Fig. 4(a), a relaxing dream house in Fig. 4(b), and a bar dream house in Fig. 4(c).

The performance results (grades) of each student is illustrated in Figure 5; the majority of students finished their projects were awarded a Level 2 and 3. Further analysis on the result of grades is provided in the next section.

4.2. *Use Case 2: Developing a VR/AR application on a given topic*

4.2.1. *Project description and deliveries*

Unlike Project1 which focused on web-based VR application, in Project 2 students have the freedom to choose any types of application type and hardware platform (e.g., the application type can be web VR/AR, iOS VR/AR, Android VR/AR,

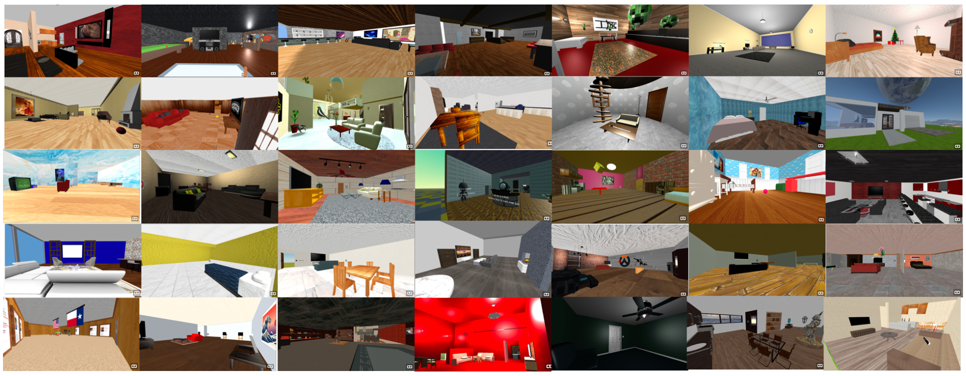

Fig. 3. A collage of WebVR applications created by sampled students.

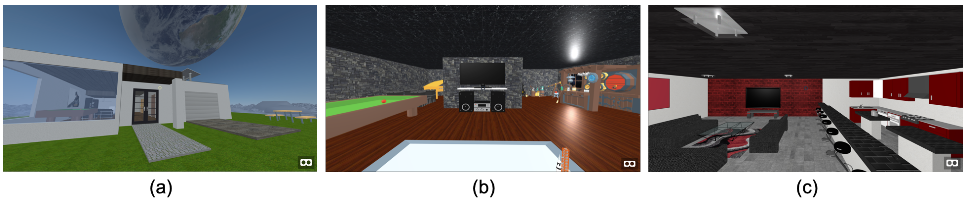

Fig. 4. Sampled Students' WebVR project examples: (a) A moon dream house (b) A relaxing dream house (c) A bar dream house

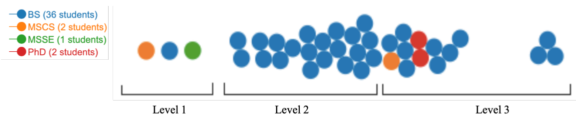

Fig. 5. Students' grade distribution of the WebVR dream house project: Level 1 is equivalent to a C while Level 3 is equivalent to a A.

or Oculus app). This project requires to create a VR/AR application that focuses on a given topic (i.e., water education). Students can select the topic suggestions provided on the classs website or come up with their ideas

Project delivery The VR/AR application will be presented in the classroom by students' device or Oculus Rift (upon requested), the running application will be shown on the screen with a projector. Along with the delivery of the developed applications, students are required to submit a report that addresses all the issues, challenges faced during the project developments. These reports will be used as part of data analysis.

Fig. 6. Survey responses from students in Project 1 from question 1 to 10.

4.2.2. *Project assessment*

In the second project, students' performance was evaluated by both instructor/TA and their peers via a Google Form. Peer evaluation was conducted via a Google Form which is shown in Fig. 7

Number	Question	Question type
Q1	How do you rate the difficulty (technical challenging) of this project on the scale from 1 (easy) to 10 (very difficult)?	Likert scale
Q2	How do you rate the usefulness of this project on educating water/environment on the scale from 1 (useless) to 10 (very useful)?	Likert scale
Q3	How do you rate the interestingness of this project on the scale from 1 (boring, don't want to try) to 10 (very much like to play with this VR/AR game)?	Likert scale
Q4	How do you rate the Innovation of this project idea on the scale from 1 (not at all) to 10 (very creative)?	Likert scale
Q5	How do you rate the visual design of this project on the scale from 1 (not appealing at all) to 10 (very visual appealing)?	Likert scale
Q6	How do you rate the sound effect of this project on the scale from 1 (distractive, or no sound track at all) to 10 (very effective)?	Likert scale
Q7	How do you rate the usability of this project on the scale from 1 (no guidance or very difficult to use) to 10 (very intuitive to use)?	Likert scale
Q8	How do you rate the effort of this group on the scale from 1 (no effort) to 10 (best effort)?	Likert scale
Q9	How do you rate the chemistry of this group on the scale from 1 (no teamwork at all) to 10 (best teamwork)?	Likert scale
Q10	How is your vote (score) for this group on the scale from 1 (bad) to 10 (very good)?	Likert scale
Q11	Your questions/comments for this group	Open-ended

Fig. 7. Survey questions for peer evaluation in Project 2.

4.2.3. *Project results*

In the second project, a total of 12 VR/AR applications is out due to the group work formation. Fig. 8 provides a snapshot of visual design layouts for each group's project.

4.3. *Use Case 3: Developing a VR/AR application on any topic*

4.3.1. *Project description and deliveries*

In this final project, students are released from any restriction on topic and tool. The purpose of this project had two folds: first, we want to know which framework/library do they end up with and second what research direction they are preferred.

4.3.2. *Project assessment*

Similar to Project 2, student's performance was conducted via an online survey tool. The question set was reused from Fig. 7 except for question number 2 as the requirement for topic was released.

4.3.3. *Project results*

There is a total of 14 group's VR/AR applications in the third projects. This number was increased due to the reformation of group and interest. A snapshot of visual design layout is shown in Fig. 9.

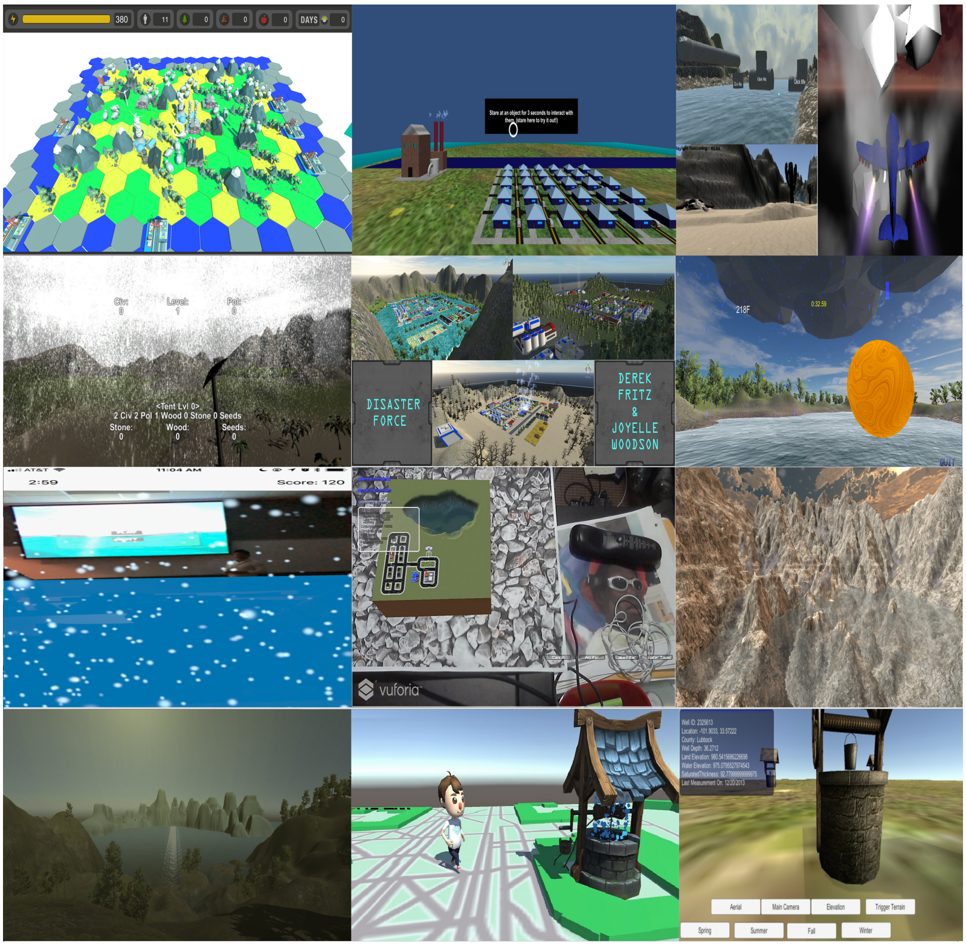

Fig. 8. A collage of VR/AR applications created by students.

5. Results and Discussion

A large portion of this section contributed to the *research question 1* as the main stream of this paper whereas research question 2, 3, 4 positioned as a supporting point for the *research question 1.*

5.1. *Research question 1: What is the the typical VR/AR development framework/library that allows novice users to stimulate their interest by creating and sharing VR/AR content with ease?*

To give answer for this research question, we analyzed Use Case 1 in more detail using descriptive and correlation analysis to garner a deeper understanding of stu-

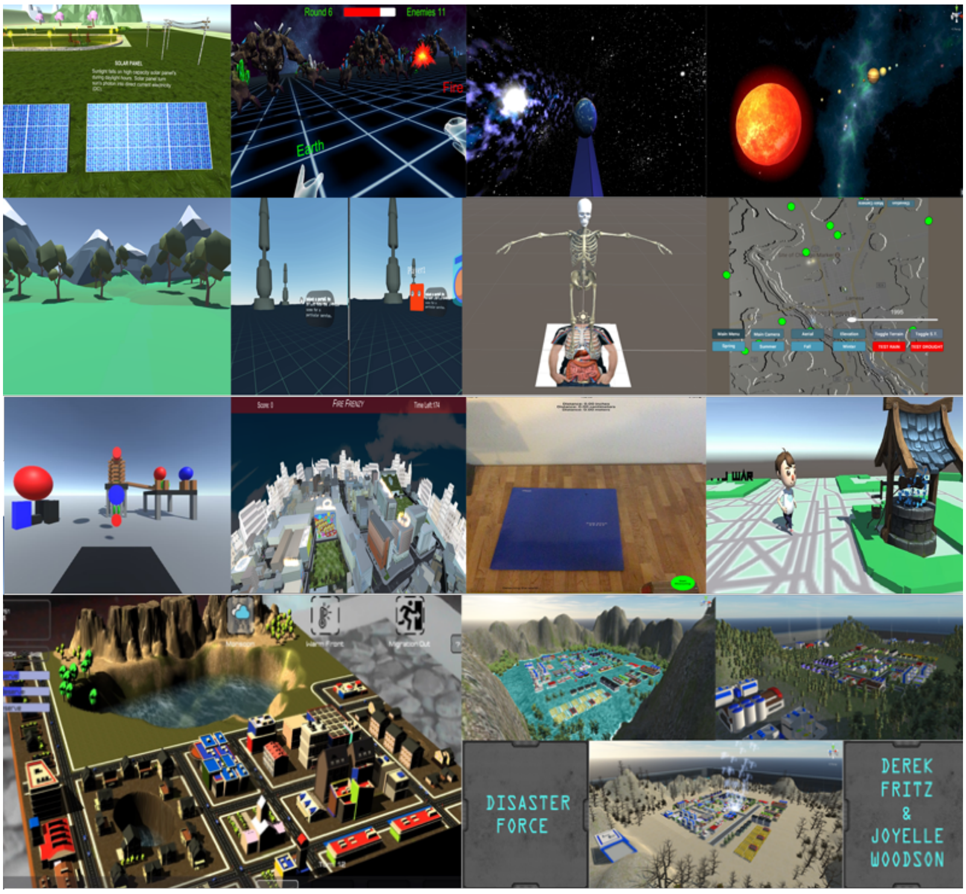

Fig. 9. A collage of VR/AR applications created by students in Project 3.

dents' reported experiences from the TAM-based post-project survey. In all, there were 36 completed survey responses.

Descriptive Analysis. As depicted in Fig. 6(1), majority (92%) of students chose *A-frame* as their WebVR library to complete their first project; two students used both *A-frame* and *Three.js*, whereas only one student used *Unity3D*. Reasons included suggestions provided by this instructor Fig. 6(2), accounting for 77.78% and searching on the internet (13.89% – due to student absence or unfocused in class). Although *A-Frame* claims to be one of the easiest libraries to create WebVR content for any user [27], 9% of students reported it difficult to use (only 3 students rate *A-Frame* difficulty from 8 to 10). Anecdotally, a part of this problem may have stemmed from the fact that some students had no experience with web development (when asked in the classroom and short TA interviews). In general, the chosen library (largely *A-Frame*) was considered by students to be of moderate difficulty regarding ease of use ($mean$ = 5.09). Overall, students' expectation of the chosen library was highly acceptable in terms of utility (overall mean score is greater than 7.0) as in Fig. 6(4). In terms of programming skills required to learn the library, a majority of students (97%) felt that they do not need an advanced level to complete their project, only 3% reported that they needed to learn more coding. This may be explained by the students' reports that, students spent most of their time in the project customizing objects (e.g. shape, size, color, position, rotation) rather than creating new objects. These results suggest that students with a moderate level of programming skills could creatively design and execute (accept the use of) a WebVR project within a short time.

Students also reported that the most time consuming part of the project was regarding content creation. Creating 3D models can be a painful task, especially to users who are new to designing 3D models. Consequently, 84.21% of students decided to use (download) existing models from the internet as depicted in Fig. 6(6). In evaluating students' projects, researchers found that students often made 3D models from the basic shapes (e.g., a cube, a sphere and plane as indicated in the report); and these low poly models increased the performance (rendering and interactivity) of the application significantly compared to those downloaded from the internet. Only a few students challenged themselves by using third party software to create models (7 out of 38). Although, an sample project was provided to students at the beginning of their project to provide context to the learning activity, students' feedback in Fig. 6 (7) showed that they actively referenced tutorial documents on the chosen library website (73.68%). Overall, half of the class (50&) is able to finish the project in one week as shown in Fig. 6 (8). This estimation would have been more precise if we had measured students hourly as some students spent whole class days working on their project, while others allocated project time on multiple class days. We found that this duration was reasonable for the majority of the students who wanted to experience VR before getting bored.

Fig. 6 (9) illustrates students' feedback in terms of the difficulty of the project. Students felt that this project was neither too easy nor too difficult ($mean = 5.25$ and $Std.Deviation = 1.57$). This is what we expected when introducing a WebVR project for students from various disciplines. On the other hand, the usefulness of the Project toward learning VR did not yield expected results ($mean = 7.08$ and $Std.Deviation = 2.28$) as seen in Fig. 6 (10) since it only focused on WebVR application. Yet, this result may be a good indicator to help instructors for preparing similar classes using a student learning-centered model [29].

Correlation analysis. To further investigate the use of the chosen library, we conducted a correlation analysis between the chosen library with the project expectation, the SPSS software was used for data analysis, the Pearson correlation is used as a metric for evaluation. Fig. 10 shows the correlation values among questions Q3, Q4, Q9, Q10. It can be seen that there is a moderate negative correlation (r =-.401, p = 0.015) between Q4 (library expectation) and Q9 (project difficulty), indicating a relationship between high library expectation and lower difficulty of the project (and vice versa). While exploring the correlation between the ease of use of the chosen library with library expectation to finish the project, the Pearson correlation (r =-.441) was only moderately negative and significant only at 0.01 confidence level. A weak correlation can also be found between library ease of use and perceived project difficulty (r = .620 and p =.000).

		Q3	Q4	Q9	Q10
Library ease of use (Q3)	Pearson Correlation	1	-.441	.620	.06
	Sig. (2-tailed)		.007	.000	.713
Library expectation (Q4)	Pearson Correlation	-.441	1	-.401	.272
	Sig. (2-tailed)	.007		.015	.108
Project difficulty (Q9)	Pearson Correlation	.620	-.401	1	-.109
	Sig. (2-tailed)	.000	.015		.526
Project Usefulness (Q10)	Pearson Correlation	.064	.272	-.109	1
	Sig. (2-tailed)	.713	.108	.526	

Correlation is significant at the 0.01 level (2-tailed)
Correlation is significant at the 0.05 level (2-tailed)

Fig. 10. Pearson correlation test scores produced by SPSS software [15].

5.2. *Research question 2: Given an answer from the previous question, does this library/framework prioritized to use to solve a more complex problem?*

In Project 2 (summarized in Table 5.2), a large portion of applications was developed in Unity3D (11 out of 12 apps), only one application was carried out in Xcode (an integrated development environment (IDE) for MacOS user). None of the application was created using A-Frame as introduced and practiced in Project 1. This was due to several reasons: (1) first and foremost important point is the curiosity of having experience on the new hardware device (i.e., Oculus Rift). However not all student's personal laptop was able to meet the minimal hardware requirement (lack of graphic card), (2) some certain components to fulfil project's requirement were not available in the current WebVR library, students came up with "how to use" components in Unity3D rather than creating their component in JavaScript, (3) new IDE tool (Unity3D and Xcode) provided a more intuitive way to generate and mange programming codes, (4) the formation of teamwork which directed members of the group unified a common framework/library, and finally the fear of being alone lets students go with "crowd effect" – that is, students can consult their peers for an advice. This could be the reason why other development tools were not taking into consideration (e.g., Unreal Engine).

Table 2. A summary of VR/AR application types and deployed hardware

	Tool/library	Application type	Hardware	Count
Project 2				**12**
	Unity3D	VR	Oculus Rift	3
	Unity3D	VR	Desktop/Laptop	6
	Unity3D	AR	Smartphones	2
	Xcode	AR	Smartphone	1
Project 3				**14**
	Unity3D	VR	Oculus Rift	5
	Unity3D	VR	Desktop/Laptop	2
	Unity3D	VR	Smartphones	2
	Unity3D	AR	Smartphones	3
	Xcode	AR	Smartphone	1
	A-Frame	VR	Smartphone, Computers	1

5.3. *Research question 3: Upon releasing from any constraint, which library/framework does students use to develop a VR/AR application of their interest?*

Unlike Project 2 where half of the VR/AR applications was deployed and tested on student's personal laptop. In the last project, this number was reduced to only 2 whereas Oculus Rift and personal smartphone were utilized suddenly. Only one group continued to use the first learn library (A-Frame). Final survey response when students had a freedom of developing their interest reported that the majority of students wanted to explore and experience some "fun stuff" on their devices (smart phone) whereas some groups continued developing and improving the existing second project because they wanted to "learn more about it". Reason for developing A-Frame in the third project was "easy testing" on student's devices.

5.4. *Research question 3: What are pros and cons of WebVR compared to other app-based VR/AR from learners' perspective?*

This section does not try to address all WebVR issues compared to other related app-based VR/AR applications. Instead, it summarized all problems faced by students during the development of the three projects which were given in the students' reports, questions posed in classroom and survey responses at the end of the semester.

5.4.1. *WebVR application development*

Pros

- *Fast development*: With a little knowledge on HTML and JavaScript, users are able to create and enjoy immersing into VR environment with ease even with new users that said "Web VR is not that difficult".

- *Responsive community*: The young community provides a responsive feedback to users once a question was asked; specifically via Slack, Github, Twitter and Stack Overflow.
- *Compatible devices*: Since WebVR runs on an internet browser that support WebGL, it is compatible with most modern smartphones.
- *Open-source*: Written in vanilla JavaScript and HTML, the majority of WebVR library provides an open-source code for enthusiasts.
- *Easy sharing*: The VR contents can be easily shared with multiple users by providing a link to the webpage. In addition, users are able to enjoy being with each other in the same VR environment without purchasing additional specialized hardware (e.g., social VR room as presented by one group in Project 3).
- *Easy integrate available VR contents*: There are a lot of available components that allow users to bring VR contents and 3D models into live with ease in a single line of code.
- *Available components*: Although still in development, the continuously developing components allows creators to speed up their work (e.g., Aframe-extras and Networked-aframe)

Cons

- *Performance*: One of the downsides of WebVR is performance due to its lower fps (frame-rate per second) compared to native VR/AR applications. The fps sometimes drops in a non-native support WebVR browser (i.e., 25 - 30 fps).
- *Loading time*: Since WebVR uses internet connection to load VR contents on the fly and upon requested, loading time is depended on the network traffic and bandwidth. Hence more VR contents would take longer time to retrieve.
- *Coding control*: It is challenging for users to manage and control a certain part of the application when its size continues growing (e.g., more codes and components), especially with asynchronous issues when users do not know which part of the programming runs first, leading to difficulty in debugging.
- *No standardized WebVR*: Because WebVR is still in the early development stage, there are a lot of libraries/frameworks being developed. Thus, it has not been standardized yet.
- *Unstable libraries and components*: The developing libraries/components keeps updating every two to five months to patch or add more functions. In addition, some components may not work with the legacy version leading to unstable system or vice versa.
- *Available components*: Compared to other VR/AR development tools, available components of WebVR are still behind. This was one of the reason why none of students used WebVR for a more complex problem.

5.4.2. *App-based VR/AR application development*

Pros

- *Well developed IDEs*: These existing development tools have appeared a long time in the market so the IDEs are more stable and well developed.
- *Good and stable performance*: The unexpected fps dropping in WebVR was hardly found in app-base VR/AR applications. Users can have smooth experience in VR due to the availability of graphic card.
- *Rich community*: Unlike young community in WebVR, the long established community can provide and share a lot of information for new learners.
- *Available VR contents/components*: There are plenty of free VR contents can be found on the Assets stores with rating information from reviewers. Users are able to shorten development time by using free developed components with ease.

Cons

- *Specialized hardware*: The first issue was the lack of HMDs headset for testing VR experience. The second problem came from the development devices. In our study, none of the participant had laptop that can meet the minimum requirement of Oculus Rift SDK installation. Students had to create a VR/AR for testing on desktop/laptop first then go to the instructor's lab to test the final version as said '*no one on the team had a supported GPU and travelling to the lab to debug, attempt to set up VR, and debug again wasn't viable, this was not possible*'
- *Sharing experience*: It is difficult to share the same experience with multiple users at the same time due to the cost of HMDs as well as deployed machines that need graphic card.
- *Huge space for development*: Compared to WebVR, app-based VR/AR applications take a lot more space that make it difficult to share the source code to others for testing and demo, especially via an online repository which does not allow to upload files with more than a certain size (e.g., Github restricts file to 100Mb).
- *API incompatibility*: One of the biggest issues when dealing with app-based VR/AR development is the compatible API with legacy system and on the deployed devices. The first problem was due to the different software versions on each group's members. When they share the application for testing and demo, one component may not work well on the other computers. The other issue related the Operating System (OS) on the target device in which the VR/AR applications halt, quit or black screen when the OS was upgraded.

5.5. *Lessons learned and discussion*

The results of the three projects provides unique insights in using WebVR compared to other app-based VR/AR approaches for individual and collaborative development of VR content among learners of various programming skill levels. Although *Three.js* library was introduced first and highly recommended to students for its utility, or ability to understand the basic idea of how WebGL works, only two students used this library for their projects due to the complexity of the code, time to set up the basic VR environment, and interactions. An important reason for this was likely due to the effort (knowledge) needed by the user to maintain and control the program from running correctly when a new component was imported. The second library (*A-frame*), on the other hand, was more attractive to students because of the ease of use, namely its simplicity and easy VR setup compared to *Three.js*. In addition, instead of spending time on programming efforts, students focused on logical design and performance of the application. For example, while working with the high fidelity 3D models (obj or fbx), the size of these models are big, so it heavily affects the performance of the application. Students overcame these problems by (1) creating a similar low poly model, and (2) converting these heavy models to another format (gltf - GL Transmission Format [30]). Therefore, students' perceptions of ease of use became paramount with equivalent platforms in utility, mediating their technological acceptance and overall use.

Of note, there were many helpful tricks made by students, one trick was with 3D object interactions. While some libraries did not allow direct interaction with 3D models (meaning only interact with some primitives such as cube, rectangle, cylinder, torus with pre-defined Box Collider), one student created an invisible big cube that cover the whole 3D models in the same position, functioning as a Box Collider, although was is not. From the user's perspective, it appeared to interact with the 3D model, but in fact, only interacts with the invisible cube. Another interesting trick was in creating scroll blinds on house windows, an animation made by merely creating two cubes on top of the windows and moving to the bottom windows, once the user clicked on the switch. When the cubes moved down, they blocked the light from entering into the house, creating an illusion of window blinds.

Although all students had some prior experience with programming, not all of them had troubleshooting skills, or ability to find out the reasons why application encountered an error. Such errors are unavoidable, especially when the size of the project gets bigger with complex VR applications. For example, it is easy to fix a mistake if there is an indicator of an error, however, what happens if the application shows no error? Several students faced this problem and one common strategy was to remove/comment parts of the application (*leave-one-out* approach) until it runs successfully. Then, students added back the codes incrementally to locate errors.

Besides the advantages of WebVR framework, there are some difficulties in developing WebVR applications. The first challenge is how to handle complex 3D models. Although these models can be converted to another format to reduce their size, they lost their fidelity, which reduces the apparent realism unique to WebVR environments. The second challenge was to reduce latency when users change their orientations suddenly which often causes dizziness or nausea, otherwise known as VR sickness. These challenges are common, yet important in maintaining virtual presence in 3D, VR environments [31].

The advantages of WebVR in terms of "ease of use" and "online sharing" would have not become apparent without considering Use case 2 and Use case 3 whereas students faced a lot of troubles in "*Collaborative design and issues with version* *control for certain aspects of the project*" or "*Dealing with library/hardware incompatibilities and fixing problems with the tools that we are using.*". This issue was not the case for WebVR as it required only a text editor for the development.

Recommendations Based on this research, we present four recommendations for educators, instructors, and course designers when teaching WebVR in the classroom:

- Before project development, short presentations for WebVR applications should be introduced by students. First, it engages students in the process by researching contents and securing materials for self-guided learning. Second, it motivates and inspires other students by listening to peers showcasing the diversity of extant VR assets. And third, it provides a snapshot of this knowledge within a various content interests as indicated in the course feedback "*I didn't realize all of the different applications for VR, but learned a lot from the student choice presentations.*". Each presentation should only take around 5 to 7 minutes as longer presentations may lead to information overload.
- To increase motivation for students in doing a project, a basic, sample project should be provided. It can work as a "skeleton" project as a baseline from where students can start, especially for those who are new to programming to "*Learning how to use it in a very short time*". By having a sample project, students can see an exemplar of the WebVR project expectation.
- For agile project development, use of an online discussion forum and TA moderation can play an essential role in helping students throughout the development process. Any question comes out during project development can be posted, receiving feedback quickly without having to wait until next class. Interesting findings can also be shared among students, providing an opportunity to co-construct meaning of the process.
- Regarding technological approaches, *A-frame* is a good starting point for undergraduate students and new learners. From the survey responses and project deliveries, students preferred *A-frame* as a skeleton structure and gradually develop the project based upon utility and ease of use, as per the TAM. Complex codes can be embedded later on, through careful scaffolding of information and guiding students through advanced programming.

6. Conclusion and Future Work

This paper presents the feasibility of adapting the current WebVR preferred by a majority of students and compares its usability to app-based applications, specifically the giant Unity3D. While WebVR is still in the early stages of development, its usability and ease of use are undeniable because it supports a wide range of hardware devices and is independent of any operating system, facilitating technological acceptance among even the most novice of users. Access to hardware is an ongoing challenge for classrooms across the world (i.e ineffective or obsolete hardware, restrictions to third-party software, and so on), therefore WebVR has broad appeal due to the ubiquitous presence of web-based technologies. By partaking in the three projects, through simple, complicated to of interest, students' skills were gradually increased. Although the scope of the study was huge (no restriction on library and tool or even topic), we consider this contribution would provide an important research documents for future eager learners in Virtual Reality. The main drawback of this study was to consider alternative emerging libraries/tools (e.g., Babylonjs or Unreal Engine) due to their unused from students .In the future work, we continue our research by exploring in more detail in Use case 2 and Use case 3 in terms of learning new framework/library.

References

[1] J. A. Barceló, M. Forte and D. H. Sanders, *Virtual reality in archaeology* (Archaeo-Press Oxford, 2000).

[2] B. Atzori, H. G. Hoffman, L. Vagnoli, A. Messeri and R. L. Grotto, Virtual reality as distraction technique for pain management in children and adolescents, in *Advanced Methodologies and Technologies in Medicine and Healthcare*, (IGI Global, 2019), pp. 483–494.

[3] M. Rus-Calafell, P. Garety, E. Sason, T. J. Craig and L. R. Valmaggia, Virtual reality in the assessment and treatment of psychosis: a systematic review of its utility, acceptability and effectiveness, *Psychological medicine* **48**(3) 362–391 (2018).

[4] Y. C. Huang, K. F. Backman, S. J. Backman and L. L. Chang, Exploring the implications of virtual reality technology in tourism marketing: An integrated research framework, *International Journal of Tourism Research* **18**(2) 116–128 (2016).

[5] B. L. Ludlow, Virtual reality: Emerging applications and future directions, *Rural Special Education Quarterly* **34**(3) 3–10 (2015).

[6] N. Robitaille, P. L. Jackson, L. J. Hébert, C. Mercier, L. J. Bouyer, S. Fecteau, C. L. Richards and B. J. McFadyen, A virtual reality avatar interaction (vrai) platform to assess residual executive dysfunction in active military personnel with previous mild traumatic brain injury: proof of concept, *Disability and Rehabilitation: Assistive Technology* **12**(7) 758–764 (2017).

[7] J. Psotka, Immersive training systems: Virtual reality and education and training, *Instructional science* **23**(5-6) 405–431 (1995).

[8] C. Deed, P. Cox, J. Dorman, D. Edwards, C. Farrelly, M. Keeffe, V. Lovejoy, L. Mow, P. Sellings, V. Prain *et al.*, Personalised learning in the open classroom: The mutuality of teacher and student agency, *International Journal of Pedagogies and Learning* **9**(1) 66–75 (2014).

[9] F. D. Davis, R. P. Bagozzi and P. R. Warshaw, User acceptance of computer technology: a comparison of two theoretical models, *Management science* **35**(8) 982–1003 (1989).

[10] F. M. Dinis, A. S. Guimarães, B. R. Carvalho and J. P. P. Martins, Virtual and augmented reality game-based applications to civil engineering education, in *Global Engineering Education Conference (EDUCON), 2017 IEEE* IEEE2017, pp. 1683–1688.

[11] A. Z. Sampaio, M. M. Ferreira, D. P. Rosário and O. P. Martins, 3d and vr models in civil engineering education: Construction, rehabilitation and maintenance, *Automation in Construction* **19**(7) 819–828 (2010).

[12] V. T. Nguyen and T. Dang, Setting up virtual reality and augmented reality learning environment in unity, in *Mixed and Augmented Reality (ISMAR-Adjunct), 2017 IEEE*

International Symposium on IEEE2017, pp. 315–320.

[13] F. Kalelioğlu, A new way of teaching programming skills to k-12 students: Code. org, *Computers in Human Behavior* **52** 200–210 (2015).

[14] E. Wang, Teaching freshmen design, creativity and programming with legos and labview, in *Frontiers in Education Conference, 2001. 31st Annual* **3**, IEEE2001, pp. F3G–11.

[15] V. T. Nguyen, R. Hite and T. Dang, Web-based virtual reality development in classroom: From learner's perspectives, in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)* Dec 2018, pp. 11–18.

[16] L. S. Vygotsky, *Mind in society: The development of higher psychological processes* (Harvard university press, 1980).

[17] C. Duffy, *Borodino and the War of 1812* (Seeley Service & Company Limited, 1972).

[18] D. Bell, Teaching virtual reality, *ACM SIGCSE Bulletin* **28**(2) 56–61 (1996).

[19] An educational framework for creating vr application through groupwork, *Computers & Graphics* **34**(6) 811–819 (2010).

[20] P. Häfner, V. Häfner and J. Ovtcharova, Teaching methodology for virtual reality practical course in engineering education, *Procedia Computer Science* **25** 251–260 (2013).

[21] T. M. Takala, L. Malmi, R. Pugliese and T. Takala, Empowering students to create better virtual reality applications: A longitudinal study of a vr capstone course, *Informatics in Education* **15**(2) p. 287 (2016).

[22] S. Stansfield, An introductory vr course for undergraduates incorporating foundation, experience and capstone, *SIGCSE Bull.* **37** 197–200 (February 2005).

[23] J. Zara, Virtual reality coursea natural enrichment of computer graphics classes, in *Computer Graphics Forum* **25**(1), Wiley Online Library2006, pp. 105–112.

[24] D. Kamińska, T. Sapiński, N. Aitken, A. Della Rocca, M. Barańska and R. Wietsma, Virtual reality as a new trend in mechanical and electrical engineering education, *Open Physics* **15**(1) 936–941.

[25] D. C. Cliburn, J. R. Miller and M. E. Doherty, The design and evaluation of online lesson units for teaching virtual reality to undergraduates, in *Frontiers in Education Conference (FIE), 2010 IEEE* IEEE2010, pp. F3F–1.

[26] A. S. Carter and C. D. Hundhausen, A review of studio-based learning in computer science, *Journal of Computing Sciences in Colleges* **27**(1) 105–111 (2011).

[27] Mozilla, A-frame school https://aframe.io/aframe-school/.

[28] R. Cabello, Javascript 3d library https://threejs.org/.

[29] J. G. O'Brien, B. J. Millis and M. W. Cohen, *The course syllabus: A learning-centered approach* (John Wiley & Sons, 2009).

[30] Khronos, Gl transmission format https://www.khronos.org/gltf/.

[31] C. G. . J. M. G. JHite, R., Hardware affordances and challenges to produce presence and learning in k-20 science virtual reality environments.