

# Group 47 - Senior Capstone 2019

## Immersive Web VR/AR - Problem Statement

Jonathan Jones

### **Abstract**

The Immersive Web is on the front lines of up and coming cutting edge technology. WebXR, building on its predecessor WebVR, has finally added support for VR controllers on the web. WebXR prides itself on delivering a VR experience on a website without requiring users to download client-side software. This technology is new and is lacking in demonstrations of its abilities. Group 47's job is to research and develop an application that showcases the capabilities of the WebXR API. Some proposed projects include a physics simulation, data visualization, event seating, and simple video games. Each project is aimed at providing an interactive VR experience that does not require client-side software. The project will be considered complete when it has a working prototype, satisfactory UX, maintained compatibility, is interactive and remains within the VR frame budget of 11 milliseconds.

## I. DEFINITION AND DESCRIPTION OF PROBLEM

The immersive web is an up and coming revolution on the world wide web. Just as the transition from terminal based computers to user interfaced experiences, there are major changes incoming. Virtual Reality is slowly becoming common in our society, yet it has been limited by the lack of supportive technology for its use on the web. For the last few years, web developers have bathed in the delight of the WebVR API. This open-source software has allowed for users to enjoy a VR experience without having to download a client-side program. WebVR is compatible with lots of different VR devices and browsers but has struggled in its absence of VR controller support. Users have been unable to interact with the virtual environment they have been presented with. This has been a barrier to most developers who want to create immersive VR experiences for their users.

Recently, the immersive web community has developed WebXR. This is an upgraded version of WebVR that includes VR controller and AR support. With this new, cutting-edge technology, an entirely new world of immersive web programming has been opened up.

The problem here is that WebXR is relatively new and there are no real applications out there that showcase its abilities. Our client, Alexis Menard at Intel, wants us to create an application that is able to demonstrate what WebXR can do. There are thousands of possibilities to explore.

We, Group 47, have the honor of being pioneers in this field as there are little to no applications that showcase the abilities of WebXR. Our central purpose is to essentially develop a leading example for WebXR applications. It is imperative that we make an enjoyable VR experience that enhances the web user experience.

## II. PROPOSED SOLUTION

Our group will work to develop an application that is able to demonstrate the WebXR abilities. Specifically, we want to make use of VR controllers in our web application. Since WebVR is already well known and doesn't make use of controllers, our job is to create an experience that you are able to interact with using WebXR.

Many ideas have been thrown around already. These include a physics simulation where users can explore the laws of physics on objects in the scene such as spheres. Users would be able to scale objects, throw them around, create tracks for testing physics and change the global physics values in the scene.

Another proposed application was an event seating viewer. Users, while purchasing seating tickets for a sporting event or live show are often unsure about the view that their seats have. Users could place themselves into a virtual reality environment where they could view a 3D model of the stadium/theatre from their chosen seat before purchase.

Also discussed were Data Visualization applications. Some scatterplots and graphs are incredibly difficult to read when they incorporate gigantic amounts of data into a small 2D space. These graphs are often hard to visualize. VR makes data visualization easier because you are able to inspect the data in three dimensions. The additional dimension removes the clutter and restrictions of a 2D space, enhancing the user experience. Interaction with the scene can come in the form of scaling and rearranging the data.

Several games were proposed such as Battleship and Tower Defense. Everyone likes a good video game. So naturally a good solution for this problem is to provide some sort of VR gaming experience on the web.

Most of the proposed solutions were defined by how they are able to deliver an experience that the user cannot normally get without many additional steps. Most important is how each project is able to deliver some impact to those who experience them. The goal of this project is to deliver an experience that garners interest in the WebXR platform and its capabilities.

## III. PERFORMANCE METRICS

Completion of the project rests upon a few metrics. The most obvious of course is that of a working prototype of our WebXR application. There needs to be some sort of playable experience in VR for the users that visit the 01.org website where our application will be hosted.

Equally as important is that the UX is satisfactory and comfortable. A great VR application is one that is visually appealing and easy to use. Measurement of this metric should come from periodic user testing and feedback. Should most of the feedback be positive, a satisfactory UX job has been completed.

Parallel to the UX is the requirement that the frame budget is met. For VR, there is about 11 milliseconds of time for each frame to render. Should the content in the application exceed 11 milliseconds of load times, users will experience frame drops and motion sickness. The frame budget should never be exceeded.

The project should not lose any compatibility that the WebXR API already has. The goal of this demonstration is to show what WebXR can do; this includes its wide compatibility with web browsers and VR devices. Should the group introduce any code that removes compatibility, the demonstration has failed.

The experience must be interactive. Anyone can make a WebVR application that has no actual scene interaction. The client wants to be able to showcase an immersive web, this requires VR controller integration and user interaction.

Finally, should any of the proposed solutions implement a multiplayer experience, an obvious metric is that multiplayer connections work.