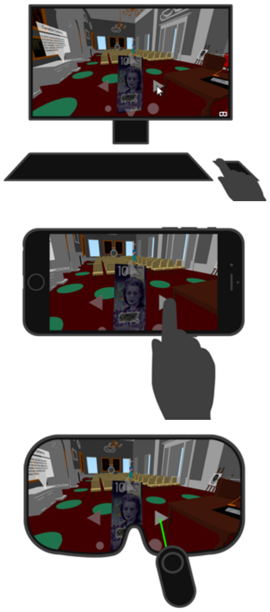
A2 - Multi-Platform Interactions (14%)

OVERVIEW

Create an application that allows for the selection, manipulation, and release of VR elements that emit sound across at two different platforms (mobile and desktop). An example to the right.



Requirements

- A-Frame component for instantiating virtual elements
- A-Frame component for destroying virtual elements
- A-Frame component that allows for selection, manipulation, and release (e.g. picking up and rotating)
- Add sound and/or music to scene for feedback and ambience.
- Code should work on both platforms chosen (though it might work slightly differently on each)
- use source-control in application development

Recommended:

- Explore environments that are relevant to the interaction(s).
- Explore interesting ways of manipulating objects (e.g. picking up, additional options pop up etc.).
- Explore how interactions needs not be the same on each platform

SUBMISSION

Each student will submit a "***lastnameFirstname_IMD3901.zip***" via CULearn by the deadline specified. This zip will contain your entire Node.js / A-Frame Project, and it **should work when I run "*npm init*", "*node app.js*", then access the content at "http://localhost:8080".**

The following components of this assignment are expected:

1. A readme.txt of (3) paragraphs describing:
   - Overview of what you did (i.e. what are the controls? Why this design?)
   - What was challenging.
   - What went well (i.e. how did you solve the above challenges?).
   - The URL to your GitHub repository. If it is private, please add me as a collaborator in your git project settings.
2. The downloaded Node.js/A-Frame project zipped project from your Github repository.
3. Between 10s - 30s .mp4 video showcasing the working project. Include sound.

NOTE: *Late policy defined by class policies on CULearn. Plagiarism policies will follow Algonquin College's regulations found here; http://www.algonquincollege.com/policies/policy/plagiarism/*

LEARNING OUTCOMES

This assignment aims to educate in the following ways:

- To plan and develop an application that uses Node.js and A-Frame WebVR.
- To assess and select input technologies suitable for a given project concept.
- To assess and explore how to use spatial sound in application design.
- To use source-control in application development.

EVALUATION RUBRIC

Outcomes	Exemplary	Proficient	Sufficient	Poor
Creativity (25%)	The overall experience is very unique, creative, and well-developed. The provided project constraints are used in an unexpected way to create a very original experience.	The overall experience is unique and creative. The end result is original and makes good use of the constraints imposed by the project.	The overall concept is creative, though takes an obvious everyday direction. Could use a little more thought to make it into something extra-ordinary.	creativity and originality. An execution with little thought.
Interaction (25%)	The interactions are very responsive. Their use is both original and intuitive so that little to no instructional training is required to use the experience.	The interactions are responsive. Their use is original and mostly intuitive; however, some instructional training is required to use the experience.	The interactions are mostly clear and responsive, however can be difficult to use and understand at times.	The computer interactions rarely work, and their influence on the on-screen result is not completely clear.
Technical Complexity (25%)	A high level of technical complexity is used to implement advanced features and/or creatively solve difficult technical problems that are well beyond the basic requirements. This is evident by looking at the source code.	An above average level of technical complexity is used to implement features and/or creatively solve difficult technical problems that are beyond the basic requirements. This is evident by looking at the source code.	An average level of technical complexity is evident in the project. The technical problems solved are the basic requirements of the project. The source code demonstrates an understanding of the basic required subject matter.	A poor level of technical complexity is evident in the project. The basic problems required by the project are not solved and/or the source code demonstrates a poor understanding of the basic required subject matter.
Overall Result (25%)	Great effort has been put into making the on-screen portion interesting. Elements are consistent with each other, polished, and feel as though they are a part of the same experience. All requirements met. Code is well formatted and has comments.	Good effort has been put into making the on-screen portion interesting. Some further attention to detail would add polish to the experience. All requirements met. Code is well formatted.	Some effort has been put into making the on-screen portion interesting. The result is somewhat consistent, though could use more attention to detail to render it to a completed state. Some requirements met. Code is readable.	The on-screen portion lacks serious effort to make it interesting and feels incomplete and/or unfinished. Few requirements met. Code is messy and unreadable.