

-

# Group 61: iCreate - Generative Design in Virtual Reality Technology Review and Implementation Plan

Rhea Mae V. Edwards  
Software Developer

CS 461  
Fall 2017



## Abstract

The purpose of this document is to represent further research on different aspects related to the creation of the senior capstone project generative design in virtual reality. Topics that are discussed within this paper are types of headsets, a variety of programming languages used, and a handful of software that can be used along with a virtual reality program. Each of these topics described consists of three different options and how they are common and also differ from one another. Out of the three options, one of the options has been chosen to be the initial choice of being implemented for the generative design project. Overall, this document identifies components of the problem the project consists of, identifies technologies for building solutions, researches alternatives, analyzes trade-offs using criteria, and works toward persuading its reader of the best choice based on its criteria.

# TABLE OF CONTENTS

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Headsets</b>	<b>4</b>
2.1	Overview . . . . .	4
2.2	Criteria . . . . .	4
2.3	Potential Choices . . . . .	4
2.3.1	HTC Vive . . . . .	4
2.3.2	Oculus Rift . . . . .	4
2.3.3	PlayStation VR . . . . .	4
2.4	Discussion . . . . .	4
2.5	Conclusion . . . . .	4
<b>3</b>	<b>Programming Languages</b>	<b>5</b>
3.1	Overview . . . . .	5
3.2	Criteria . . . . .	5
3.3	Potential Choices . . . . .	5
3.3.1	C# . . . . .	5
3.3.2	C/C++ . . . . .	5
3.3.3	Java . . . . .	5
3.4	Discussion . . . . .	5
3.5	Conclusion . . . . .	5
<b>4</b>	<b>Connective Software</b>	<b>6</b>
4.1	Overview . . . . .	6
4.2	Criteria . . . . .	6
4.3	Potential Choices . . . . .	6
4.3.1	Dynamo . . . . .	6
4.3.2	Grasshopper - Iris VR . . . . .	6
4.3.3	Leap Motion . . . . .	6
4.4	Discussion . . . . .	6
4.5	Conclusion . . . . .	6
	<b>References</b>	<b>7</b>

## 1 INTRODUCTION

There are many tools that our team will have to use in order to fulfill all of the requirements for our senior capstone generative design virtual reality program project. A few of the tools that we will have to implement are a headset, a programming language to write in, and the possibility of including some kind of other software that can be used with our finalized virtual reality program in order to intensify such a virtual user experience.

The options that we have chosen to research for headsets are the following:

- 1) HTC Vive
- 2) Oculus Rift
- 3) PlayStation VR

The options that we have chosen to research for programming languages are the following:

- 1) C#
- 2) C/C++
- 3) Java

The options that we have chosen to research for connective software are the following:

- 1) Dynamo
- 2) Grasshopper - Iris VR
- 3) Leap Motion

This document will further discuss the options stated above, and which ones out of each of the topics we have chosen to initial use in our project implementation. Each section will describe each of the options individually, and then compare each of them in regards to our project requirements and preferred implementation choices.

## 2 HEADSETS

### 2.1 Overview

In order for our virtual reality program to be useful and actually be implemented for its full purpose, there will need to be a headset for us to download our program onto. A headset is the main piece of hardware that our team will need in order for us to properly test and use our virtual program with. Choosing a headset that will fulfill all of our needs and requirement is important, and highly sufficient enough for us for our project.

### 2.2 Criteria

Our chosen headset will need to be able to run out virtual reality program, and not necessarily for example augmented reality. Also, a headset that will have particular specifications that will closely relate our generative design concept of our program, will be an even better option for us to choose to use.

### 2.3 Potential Choices

#### 2.3.1 *HTC Vive*

The HTC Vive provides a decent space for its users to move within a room or in other words "play area". This headset allows a maximum of 15 feet for its user to move from its base-stations while in use. Such a play area does not have to be a perfect square to work effectively either. These boundaries built for the HTC Vive have been made to be fairly flexible. The HTC Vive also can detect objects and can possible even replace any obstacles within its play area, preventing any overlap or object confusion within its visuals. This specification also helps with the safety and real-life physical movements. [1]

A downside with the HTC Vive though, involves this idea of its use of effectively tracking its user. According to the Gadget review of 2015, "Here's EXACTLY How The HTC Vive Works," there is a certain way the headset should be worn, begin "mounted above head height, [...] faced down at a 30-45 degree angle." [1] Such accuracy can be very particular in its use for a user to wear.

#### 2.3.2 *Oculus Rift*

One thing about the Oculus Rift, this headset has Windows 10 cross-device compatibility, where streaming through the device is a possibility when connected to a Xbox One. The Oculus Rift has similar head motion controls with its user when moving through a play area, being about a 3-foot by 3-foot square in any given space. This headset can also connect to a computer for further use allowing future installs and upgrades for its software. [2]

Also, the game engines Unity and Unreal Engine are supported by the Oculus Rift.

#### 2.3.3 *PlayStation VR*

The PlayStation VR is a fairly new headset which has not been out on the market for barely even a year. The PlayStation VR is highly compatible with game consoles, begin a true console-based user experience. This headset also provides "HDR (high dynamic range) - compatible TVs" meaning its support with HDR - compatible games. Also for a consumer's perspective on the headset, the price one pays for the device is the best value provide for how much it costs, which is high consumer's benefit. [3]

### 2.4 Discussion

### 2.5 Conclusion

We chose the HTC Vive headset setup because its play area specifics and available resources we have through the internet and personal user experiences in using the device. With our virtual reality program in relation with generative design, using the HTC Vive provides a great amount of physical space for a user to move and design their elements within the program, and hardware specifics that allow a user to get a maximum virtual reality experience.

### 3 PROGRAMMING LANGUAGES

#### 3.1 Overview

Before writing any code, we need to figure out what programming language we will want to write in. Basis of the basics.

#### 3.2 Criteria

Implementing a programming language that has been studied to be ideal for writing virtual reality programs in would be our team's main requirement in choosing a programming language. Another idea that will play in which language we choose, is that the programming language is compatible with the game engine and with the platform we plan to use for our project also. A language that has been found by other developers to be the one they preferred the most with also be another positive for us, along with the idea that such a language would be an ideal language for us, as new virtual reality programmers, to write in. A programming language that is simple enough for us to pick up and learn, with many available resources for us to use would be exceptional.

#### 3.3 Potential Choices

##### 3.3.1 C#

A note from C# developers is their love with the language of C# "for begin pleasant to use and well-designed". C# was originally designed by Microsoft for developing apps. The language is also highly recommended to be used for creating games through the Unity game engine being easy to start with, which is relevant to our team's over project, in relation to writing a virtual reality program. C# is a high level language, where there is a focus more on programming than the little details of the language itself, which can be slightly annoy for using other programming languages out there. [4]

On the scalability side, C# is easy to maintain and fast for being a statically typed language. Also, there are many resources and examples written in C# for developers to reference and use. [4]

##### 3.3.2 C/C++

C++ is a language that allows its programmer to have a lot of power and control over its computers resources through the act of coding. With sufficient knowledge in using the language, its programs can be created cheaply with high speeds and ability to run compare to other programming languages. C++ also allows its programmers "to develop game engines, games, and desktop apps. Many AAA title video games are built with C++." [5]

A downside of coding in C++, is its beginner friendliness. C++ is a lower level language that can be very complex, especially in regards to brand-new programmers coding a virtual reality program such as every member in our team. Such a complex with the language, also makes it a little difficult to maintain in general.

##### 3.3.3 Java

According to Codementor, "this general-purpose language [of Java] was designed to be easier to use than C++." Companies have used Java to develop desktop apps and website backend systems. Being more related with virtual reality programs, Java is highly used in developing Android apps. Java is beginner friendly and can be fairly easy to use to optimize performance within a written program, but can be a frustrating the use from the start, due to its great lengths in code when writing. [6]

#### 3.4 Discussion

#### 3.5 Conclusion

We chose to use C# as our main programming language because mainly with our initial chosen game engine, Unity, for our group to code in, C# is the more widely use language to code in. Many of the tutorials and examples when coding in Unity are in C# , along with C# begin the most often recommended programming language when creating games through the Unity game engine. [4]

## 4 CONNECTIVE SOFTWARE

### 4.1 Overview

Suggested by our client, was a handful of separate software and development tools that we as a team, can think of using with our program once we get our basic implementation working for our main project. The idea behind these additions, is to further our basic idea of a program and provide a more diverse user experience with our generative design virtual program. There are hopes also that adding a single or multiple additional software and tools, the these additions will provide a more realistic experience for the program's users overall.

### 4.2 Criteria

Most importantly, a software or tool that is compatible with our chosen headset and finalized written program will be a requirement in choosing a certain addition. Also, a software or tool that will actually enhance our generative designing experience the best in some way for our users, would be a priority addition for our project's implementation.

### 4.3 Potential Choices

#### 4.3.1 *Dynamo*

Dynamo is a software that is useful to implement in regards to "easing the modeling, visualization, and analysis tools" for most connected devices. In relation to our generative designing virtual reality project, Dynamo can help users "generate sophisticated designs from simple data, logic, and analysis" due to its structural work-flow in the software along with is such implementation. [7]

#### 4.3.2 *Grasshopper - Iris VR*

Grasshopper - Iris VR helps provide exported "surfaces, meshes, and breps" that can be found very useful in our generative design virtual reality program. This software can also help apply material and color to a structure or object in a given program. In regards in providing a greater designing ability with generative design virtual reality program for our users, this addition can be helpful in providing such capabilities. [8]

#### 4.3.3 *Leap Motion*

Through the cameras of a headset and infrared light, Leap Motion can help integrate a user's real hand motions within a virtual reality program. Leap Motions is rated to be fairly simple to use and implement by a developer, which is a great benefit for beginner programmers wanting to just simply create a highly realistic program through virtual reality in a short amount of given time. [9]

### 4.4 Discussion

### 4.5 Conclusion

We chose Leap Motion as our initial main software to connect our main virtual program to because it allows its users to use their bare hands when maneuvering through a virtual reality program. Allowing this type of availability for our user with generative designing will be a very convenient addition, and allow our users to experience a more realistic building of a virtual physical structural design with their own hands in virtual reality.

## REFERENCES

- [1] D. Peppiatt. (2015). *Here's EXACTLY How The HTC Vive Works* . [Online]. Available: <https://www.gadgetdaily.xyz/raspberry-pi-3-is-on-sale-now/>
- [2] M. Andronico, S. Smith. (2016). *What is the Oculus Rift?* . [Online]. Available: <https://www.tomsguide.com/us/what-is-oculus-rift,news-18026.html>
- [3] W. Fulton. (2017). *PlayStation VR (2017) review* . [Online]. Available: <https://www.digitaltrends.com/vr-headset-reviews/playstation-vr-2017-review/>
- [4] Codementor. (2016). *Why Learn C# ?* . [Online]. Available: <http://www.bestprogramminglanguagefor.me/why-learn-c-sharp>
- [5] Codementor. (2016). *Why Learn C++?* . [Online]. Available: <http://www.bestprogramminglanguagefor.me/why-learn-c-plus-plus>
- [6] Codementor. (2016). *Why Learn Java?* . [Online]. Available: <http://www.bestprogramminglanguagefor.me/why-learn-java>
- [7] J. Dellel. (2017). *Dynamo: Autodesk's answer to Computational Design* . [Online]. Available: <https://enterprisehub.autodesk.com/articles/dynamo-autodesk-s-answer-to-the-computational-design>
- [8] IrisVR. *Grasshopper* . [Online]. Available: <https://help.irisvr.com/hc/en-us/articles/220686068-Grasshopper>
- [9] A. Colgan. (2014). *How Does the Leap Motion Controller Work?* . [Online]. Available: <http://blog.leapmotion.com/hardware-to-software-how-does-the-leap-motion-controller-work/>