

# ***SIU-CAVE***

## **Cave Automatic Virtual Environment**

### **Project Plan**

Version 1.0 **(DRAFT)**

Prepared for

**Dr. Christos Mousas**

By

**JBU**

on

February 16th, 2018

## Table of Contents

|                                    |   |
|------------------------------------|---|
| 1. Introduction_____               | 3 |
| 2. Statement of Work_____          | 4 |
| 3. Work Requirements_____          | 5 |
| 4. Schedule/Milestones_____        | 6 |
| 5. Acceptance Criteria_____        | 7 |
| 6. Resource Requirements_____      | 7 |
| 7. Risks_____                      | 8 |
| 8. Roles and Responsibilities_____ | 9 |

# 1. Introduction

The purpose of this document is to present a detailed description of the plan that will be followed throughout the next several months in the design and implementation of the CAVE. In this document, the following sections are included:

- A *statement of work*, which describes the end goal of the project, along with what is and what is not included in the scope of the project.
- *Work requirements*, which will provide specific deliverables and tasks that need to be completed in order to finish the project successfully.
- A *schedule/milestones* that will provide a detailed schedule and project timeline to ensure successful completion of the project at the end of the semester.
- *Acceptance criteria*, which details the interaction between the client and the project team to ensure that both parties understand how the acceptance of deliverables will work.
- A list of *resource requirements* that details the required resources needed to complete the project.
- A description of the *risks* involved in working on the project and how to limit these risks.
- A well defined set of *roles and responsibilities* that each team member will have during the course of the project.

Dr. Christos Mousas has requested the design and implementation of a functional CAVE prototype at Southern Illinois University. CAVE stands for Cave Automatic Virtual Environment and is a type of virtual reality that allows for an immersive experience without limiting one's spatial awareness. It involves the use of projectors and three to six screens in a cubed-sized room along with some type of motion tracking system in order to make the user experience a sense of immersion like in virtual reality. Unlike other types of virtual reality, the CAVE allows for multiple people to share the virtual reality experience at the same time. CAVE systems are a unique experience, and they come with a very high price. The goal of this project is to create a prototype simulation of a CAVE system that is relatively cheap for a proof of concept. This will allow for this project to be expanded in the future with more expensive equipment in order to get an even better experience with more possible applications. CAVEs are very different from other types of virtual reality, which allows for the CAVE to have many interesting applications across many different fields.

## 2. Statement of Work

The project that Dr. Mousas has requested from us is a prototype simulation of the CAVE. In order to complete the task given to us, we will develop a functional CAVE prototype along with a demo application to demonstrate how it works.

The final project will consist of three screens, one in the front and two on the sides. Each of these screens will have a projector that will display the correct output. Each of the projectors will be hooked up to a computer, and each computer will be running an instance of Unity. We will be using an add-on for Unity called MiddleVR which is designed for multi-screen setups such as CAVE systems, as well as tracking inputs. In order to better track the user's movement and get the sense of localization, we will be using Microsoft's Kinect v1 as well as a Nintendo Wiimote for user interaction.

Our demo application will focus on three things; localization, interaction, and using the virtual environment. Since this is a prototype, we need to show the benefits of the CAVE system as well as its unique features. We believe that these three factors will best represent a demo application within our CAVE system. An example would be a simple racquetball game that centers around the player, and the cave screens represent the walls of the court. The player would have to move to location of the ball in the CAVE to serve it back, utilizing the motion of the Wiimote or the Kinect's gesture system. We will be using Unity to design and code the environment for the CAVE system as well as MiddleVR and VRPN to provide a unified interface for input devices and motion sensors. MiddleVR is designed to increase the portability of the VR application from one VR system to another and provide intuitive interface to control detailed settings of the VR environment, and VRPN helps track the details of input devices such as the Wiimote or other controllers in our scene.

### 3. Work Requirements

#### Kickoff:

- Gather research on the CAVE and the required materials to build it
- Further research and decide on the software we will be using and implementing
- Finalize and order a list of the required materials along with the total price in order to build the CAVE

#### Design Phase:

- Get familiar with the Unity MiddleVR package, VRPN and any other softwares
- Design an application utilizing all of the desired CAVE features, including localization and interactivity within a virtual environment

#### Build Phase:

- Set up the projector screens and the projectors in the room
- Connect one computer to each projector
- Set up the projectors and get the projectors to display correctly at a good resolution without any shadows on all three screens
- Run instances of Unity and the MiddleVR package on all three computers, with one acting as the master node and the other two acting as the slave nodes.
- Get all the screens and sensors calibrated and synced up properly
- Set up the Xbox Kinect and Wii Remote in order to capture the user's motion and position

#### Implementation Phase:

- Code a VR application that demonstrates how the CAVE works using Unity
- This application will show off the three main features of the cave: localization, interaction, and the multi-screen setup.
- The application that we will code would be a game, such as racquetball, that would demonstrate localization by the Kinect tracking the player's movements within the game following the movement of the ball.
- It would demonstrate interaction through the use of a Wii Remote when the player goes to hit the ball.
- It would make use of the multi-screen setup because each screen would act as a wall, and the player would be able to hit the ball off of any of the three walls
- Make sure the Kinect and Wiimote are synced up with the user movement

#### Training Phase:

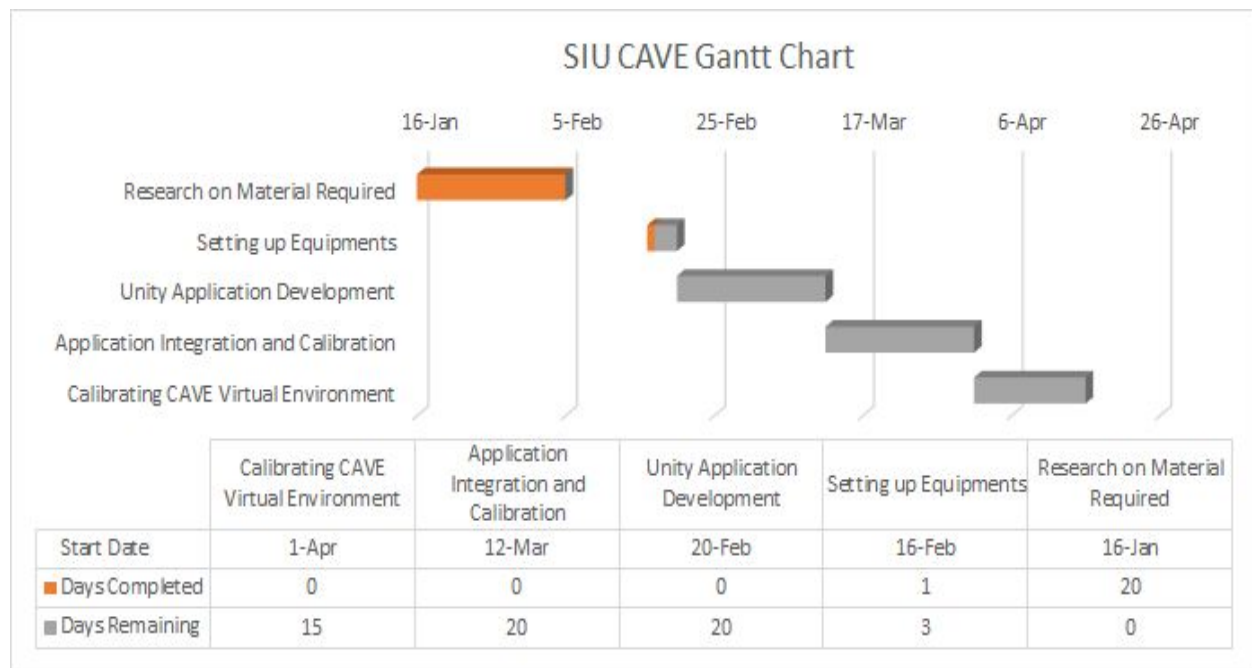
- Make sure all the screens are synced up and are showing the proper display
- Calibrate the Kinect tracker with the displays and make sure the calibration is correct and everything is synced up
- Test our application for bugs and make any corrections as necessary

#### Handoff/Closure:

- Make sure everything works properly

- Get approval from Dr. Mousas to make sure it meets his expectations
- Demonstrate the final product

## 4. Schedule/Milestones



By February 5th, all required parts and materials will be ordered for the CAVE prototype. This includes projectors, the Xbox Kinect tracker, Wii Remote, Ethernet switches for the PCs, and projector mounts. It should take about 1-2 weeks maximum to get everything shipped to the school. During this time, we will get the room prepared for all the equipment and plan out how we are going to mount the projectors and the projector screens so when the equipment arrives, we will be ready to immediately start setting it up. Also, during this time, we will have Unity and the MiddleVR add-on package installed, and we will start to get familiar with the software. Once the equipment arrives, we will begin to set everything up which should only take a few days maximum. Then, around the middle of February depending on when the equipment gets ordered and shipped, we will begin to code the application for the CAVE in Unity. This will require approximately a month in order to code the entire application and work out all the bugs. Once the application is coded and ready to go, we then have to properly calibrate the screens and the motion tracking with our application in order to ensure the desired output. This will require approximately 2-3 weeks. This will involve continuous tweaking of our application along

with the calibration of the displays and motion tracking. Once this is complete, our CAVE prototype will be ready for demonstration.

## 5. Acceptance Criteria

There will be constant communication between our team and Dr. Mousas in order to ensure that everything is going smoothly. Through this communication, we will also know that the work we are doing is acceptable and that Dr. Mousas is satisfied with it. The end deliverable of our project is the final prototype of the CAVE that we have made as well as the demo application to go along with it. Both of these things are required in order for successful completion of this project. This deliverable will be accepted through the demonstration of the CAVE which will show that everything is working properly. This is the main way to show that the prototype works along with the application. Dr. Mousas is the one that is authorized to accept the work.

## 6. Resource Requirements

The equipment required to build our CAVE system is as follows:

| Equipment                                 | Quantity |
|---|----------|
| Computer (department PCs)                 | 3        |
| Projectors                                | 3        |
| Projection Screen Material                | 3        |
| Projector Frames                          | 3        |
| Microsoft Kinect v1                       | 1        |
| Microsoft Kinect SDK                      | 3        |
| MiddleVR package for Unity                | 3        |
| VRPN (Virtual Reality Peripheral Network) | 1        |
| Wii Remote                                | 1        |
| Projector Stands/Mounts                   | 3        |
| Ethernet switch                           | 1        |
| HDMI cables                               | 3        |

## 7. Risks

Building a CAVE system can have many obstacles that needs to be handled in a proper manner. Some of the risks include:

- I. Equipment needs to be ordered and shipped and there is a chance of this getting delayed which can hinder the project timeline. In order to mitigate this, we will order all the equipment as early as possible.
- II. We have to make sure the price for the equipment does not exceed the total budget allocated for the project. Constant communication needs to be maintained throughout the research of the equipment in order to stay within the budget.
- III. There is a potential risk that the equipment ordered will not be properly efficient for the space we have or that the products do not work as intended. This risk will be minimized by conducting the proper research and communicating with Dr. Mousas in order to ensure that all the equipment will work as intended.
- IV. The equipment needs to be calibrated and synced with each of the computers and some of the devices may cause some issues. Thorough research about the devices is vital to avoid such circumstances. Proper time management and allocation will be critical in order to ensure the proper calibration of the devices.
- V. The application needs to properly showcase the CAVE system, with proper localization and immersion or else there is no point in the CAVE. This will require a lot of personal testing within our CAVE, as well as constant tweaking to the application.
- VI. There are a lot of moving parts involved, so making sure hardware and software interaction works as intended is crucial. We will be using a handful of different pieces of equipment as well as software, so it is very possible to have conflicting interactions.



## 8. Roles and Responsibilities

| Role                    | Research on the Hardware and Software   |
|-------------------------|---|
| Responsibility          | The members are responsible for collecting information about the required equipments for the CAVE system, as well as the needed software. The devices include PCs, projectors, projection canvas and motion sensors. The main purpose of this is to find the right equipment at the right cost. |
| Designated Team Members | Josh Maier, Brady Sprinkle, Utsav Dhungel   |

| Role                    | Setting up the CAVE Environment  |
|-------------------------|--|
| Responsibility          | The members of this role are responsible for connecting the devices and equipments to create the CAVE System. The projectors need to project onto the screens and they need to be connected with the Computers all running an instance of Unity Cluster Rendering. The motion sensors need to be connected to the computers as well. |
| Designated Team Members | Josh Maier, Brady Sprinkle, Utsav Dhungel  |

| Role                    | Application Developer  |
|-------------------------|--|
| Responsibility          | The members of this role are responsible for designing a virtual environment in which the User will interact in the CAVE system using Unity's software along with the MiddleVR package. This role includes the design and coding of the application. |
| Designated Team Members | Josh Maier, Brady Sprinkle, Utsav Dhungel  |

| Role                    | Implementing Application  |
|-------------------------|---|
| Responsibility          | The members of this role are responsible for the calibration and synchronization of the hardware and software. This role includes working heavily in the MiddleVR package for Unity. Members in this role will be using the configuration file in MiddleVR to constantly adjust the settings for the Kinect sensor, Wii Remote, and screens in order to properly calibrate all the sensors and screens to ensure a smooth experience. |
| Designated Team Members | Josh Maier, Brady Sprinkle, Utsav Dhungel   |

**Date:** February 16th, 2018

**Approved by:**

**Approver Signature:**

**Mentor Name:**

**Mentor Signature:**