Sarah Kiel, Austin Lordi, Anthony Blakley, Bradford DeMassa,Tyler Hogue

Due May 11

Tyler’s Writeup:

For the first semester of the project, I was in communication with the leaders of the project to get more information about the project and to schedule times for meeting to focus in for next semester. Personally, I initially read through Unity tutorials in their official documentation to get a basic understanding of the interface and some of the terminology used to describe objects in a “game”. The first major personal project I pursued was building a basic basketball game, putting together concepts I learned in the documentation. To start, I did outside research to get a better understanding of Unity. Referring to the scripting API a lot, I gained much basic knowledge to create scripts that allow different GameObjects to interact with each other. The most difficult part was ensuring the camera rotation and player movement, which was optional, functioned properly. The problem was the 3rd assets I downloaded (name to be inserted) had a bit of a learning curve and didn’t seem to support player movement too well, so I opted to build a custom camera. This camera only operated by moving the player’s vision around. In addition, I was able to accomplish what I couldn’t before – namely the character movement. It was accomplished through the use of the player’s Transform component, which included the object’s position in the world. All camera and player movement were accomplished through a YouTube video I watched that outlined the scripting and relationships between GameObjects. The last major step in learning VR I completed was adding a 360 photosphere to the world of the basketball game. It was an attempt to make the game a little more entertaining. It required inverting the normals, which reversed the textures laid on the sphere object to point towards its center.

The next major project I attempted was the VRTK (VR Toolkit). It was something to learn how VR works. Beginning with the tutorials and a sample project prebuilt by the toolkit’s developers. A problem, though, needs to be addressed. VR interactions don’t seem to work. Could be something about working on a laptop, or the Linux Unity version I’m using, or I am missing dependencies, or I don’t know the controls well enough. In any case, not much progress was made. So, to make up for this, I decided to install and learn ProBuilder (weeks later) for some productive work. PolyBrush and PolyGrids cannot be completed because they are deprecated, according to the Unity Asset Store.

Bradford’s Writeup:

Personally, I followed the Unity tutorials set up by our project instructors. Following the order of the tutorials I had set up the unity environment on my computer as well as getting familiar with the development environment. This was a little challenging figuring out that Unity Hub was different from the desktop development environment. Unity Hub was where I could import assets as well as access my projects I was working on, which brought me into the development environment. Once I figured this out I started to work with GameObjects and how to position and shape those objects. GameObjects are crucial to understand as these are objects in the world that can be interacted with. One example of this work was shown in our second group presentation. I took on the low poly-res challenge which was to create shapes, color them and create a small environment. I decided to create a large tree with a hill and some boulders in the ground. To be able to even create shapes that resembled what I wanted was challenging. Instead of using the standard Unity shapes in the environment I used ProBuilder. ProBuilder is an extension to create uniquely shaped GameObjects so that it is easier to create what you want instead of trying to work with a square or rectangle and start from more organic shapes. This tool also gives a much easier interface to learn how to also appropriately color shapes as well. This made design much quicker and was a useful tool to learn.

Moving after the second presentation I started working on some more advanced features once I understood ProBuilder a little bit better. I started working on the FlappyBird tutorial at the end of the Unity tutorials. While modifying the game I learned some stuff about how to interact with the games physics engine and change characteristics of the objects in the game. I was able to go into the C# scripting of the main object and changed its mass and velocity. I did this so when a key was hit, it would respond appropriately to the press of the spacebar instead of a mouse key. This is done through C# scripting with triggers. Triggers can be made to handle and update when they are activated. For example, in the FlappyBird game the way the scoring was carried out was through a scoring zone. When the bird flew through the scoring zone it would update the score counter. The other zones were built on the pipes. If the bird hit this zone it would update the game so that it would end with the Lose screen. This was also done on all the game buttons, so that they would function accordingly. I also learned how to update the objects depending on when a specific key was hit. I was able to change sprites and made mine and an underwater, submarine themed game.

Sarah’s writeup:

This semester, the most significant thing I accomplished towards my capstone project was completing the Unity learning modules given to us by our project mentors. I really enjoyed learning how to build and create with the software that reminded me of my days as a mechanical engineering major using Solidworks for CAD designs. It helped that I had already worked with a similar technology in the past. After setting up the program on my computer, I started by making game objects and editing a simple prefab game. After that, I moved onto making a photosphere and working with immersive environments. These were closer to what we would be doing with our project and were very important to learn. Finally, I learned how to use ProGrids, ProBuilder, and PolyBrush in my projects to enhance and build my own objects to interact with in Unity. The games were especially fun to create and useful when it came to understanding how to make things interactive.

In particular, the module I enjoyed the most was probably learning about all the asset packages you can use. I really enjoyed designing the objects and figuring out all the measurements and whatnot. Finding new asset packages was interesting. It’s just like looking for JavaScript libraries to use in web development projects. I enjoyed ProBuilder due to its CAD-like components. These packages are extremely helpful when it comes to making the game elements that you see and interact with. This will be an important skill to have when deciding on mini-projects this fall. Our project mentors have talked about having interactive objects that trigger events in the VR experience. I would love to work on something like a checkpoint activity since I enjoy making interactive objects as well as have an interest in environmental science.

After meeting with Prof. Bodzin and Junior, our project mentors, at the end of the semester, we decided on our approach to the fall semester. The project we are working on is in the fine-tuning stage and in order to make the most of our time, we will be taking on smaller mini-projects. Each will be given to one or more people in our group based on interest and skill. I feel like this is the best way to give everyone in our group a chance to shine and show off their achievements while contributing to the project as a group. Using this divide and conquer method, our team will get far more work done on the project and will leave a lasting mark.

Austin’s Writeup:

The biggest things I completed this semester were the Unity VR tutorials on coursesite. I feel that going through these tutorials has taught me a lot about working with Unity and has given me a lot of basic background experience that I can use when contributing to the project next fall. I have pretty much completed every module that is found on coursesite. These modules include the basic installation and setup of Unity, learning about the basic unity interface, creating, managing and using gameobjects, importing, using, and managing assets, creating a basketball game, creating a polysphere, using progrids, probuilder, and polybrush, learning a bit about writing scripts in C# and adding these scripts to certain objects in Unity projects, experience with the VR toolkit, and I also did the final extra practice tutorial which involved creating a complete Flappy Bird like game.

I thought creating this game was very useful because it involved building a project from scratch and implemented almost all of the techniques learned in previous tutorials. This included importing various sprites to use in the game. To get these sprites you need to either make them yourself using a picture editing software(i used paint) or downloaded free to use ones online. Creating this game also involved adding different triggers, properties, and flags to the game objects, such as a score zone to represent scoring a point, a dead zone to represent when you lose, as well as adding basic gravity and physics mechanics to the avatar. It also involved moving these objects on screen to simulate infinite movement. In addition, this project taught about good organization within projects, as we needed to separate and organize the different screens, such as the main menu, the scoreboard, the reset screen, and the main game screen.

The next steps for us as a group is to figure out how to divide up the work. We will be having another meeting on May 22nd where we will go over that. Based on the skills I have and what I have learned while going through the Unity tutorials, I think I would prefer to do something more backend and coding oriented, as opposed to doing something with the design of the game. In the meantime, I have found a bunch of interesting Unity tutorials on youtube and I plan on doing some messing around on Unity over the summer, to keep improving my skills with the interface and to make sure I do not forget everything that I learned once fall gets here.

Anthony’s Writeup:

As previously noted, we were tasked with completing multiple modules that would aid us in learning unity. These modules ranged from learning how to simply import assets into a 3D environment, to adding scripts that could manipulate our game objects. While actually learning Unity was interesting, it wasn’t my main goal. Our project advisor informed us about several tasks that we could focus on as an overall goal for the project. These tasks would, then, be completed after we learned the basics of Unity. One of these tasks was improving the image quality in the phone version of the VR game. Because I had experience with 3D modeling software, I decided to also focus on lighting, resolution, and image quality alongside the modules. Those were the things I planned to improve in the following semester.

I believe that I made great progress in understanding my task and getting closer to solving it. One of the technologies that I implemented was a unity plugin called OpenStreetMaps. The OpenStreetMaps plugin uses real-world data, like Google Maps, that allows me to create a 3D version of an environment. The benefit of this is that unlike static images, 3D environments across multiple hardware are adjustable. The current state of the project uses photos to navigate the Lehigh Watershed area. Taking resolution and quality into our own hands, I thought, could be a solution. When I actually created the lighting needed for my 3D environment, various problems had arisen. The main problem was that I didn’t have a good understanding of how lighting-assets worked in Unity. While the tutorial touches on this topic, it didn’t give all the information I desired. As a result, I decided to use a software called Cinema 4D. With this software, I was able to import my environment and add lightning to a basic object. I believed this could be used as a proof of concept when determining how to fix the issue on image quality.

After our first meeting with our project advisors finished, I learned that they had used high-resolution cameras to take new photos. This technique would be used to improve image quality. This of course means that my suggested approach wouldn’t be used. Nonetheless, we were given other tasks that we could all work on. A potential one for me is moving the code from one of the VR-headset’s software to another. With this in mind, my goal for the summer is to understand the major programming differences between the two headset-software.

Goals:

Our goal for this upcoming semester is to contribute to the various tasks we were given via our group meeting. These potential tasks include importing high resolution images into the photospheres and working on writing port-related code. The purpose of the new code would be to enable us to use Oculus as opposed to the HTC Vive headset. In our upcoming meeting (May 22), we plan to discuss, in detail, what goals need to be accomplished for each task. Additionally, we plan to have weekly checkups with our project advisors to ensure that our work is on par with what’s expected of us.

For the hi-res images, we will also be improving on the story-telling of the VR tour. Accomplished using historical imagery stored in the Google Drive we also use to import the photospheres.