Windows XPerience
By Junior Garcia and Juhee Noh

1. *Project Description:* describe the space, the "storyness" it conveys, and the inter-relation between modes of interaction and the logic of the world.

As a kid, I was always fascinated by computers. How could they achieve tasks that are so inherently complex in a matter of seconds and with seamlessly little effort? Now that I am studying Computer Science, I know that every single computer-based task takes hours of algorithm design and coding implementation that breaks down the difficulty for the end-users. As such, I wanted to convey the difficulty of a computer-based task as simple as submitting an assignment through gmail by taking full advantages of the features lended by Virtual Reality through the Unity Game Engine. In addition, I wanted to explore the idea of space, and how space is defined in a medium as nascent as Virtual Reality where the artists have a full range of control of the experience they want their audience to feel, see, listen, and fully engage in. As such, me and my partner Juhee decided to bring back a vintage piece of technology in Virtual Reality: the Windows XP computer. We want the users to feel enthralled by the space inside of a Windows XP computer, to enjoy the iconic green mountains laying against the beautiful blue sky. This scenery will be juxtaposed by the clock-ticking task of submitting an assignment via email. The user has to enter the computer, physically juggle through a bunch of documents that come from the folder icon to look for the assignment, enter google chrome, and submit the assignment. However, the former task is not as easy as expected as the internet connection is lost and our player must face the jumping dinosaur from Google Chrome. To submit the assignment, the user must kill the dinosaur with a giant cactus. This allows for the mailbox to appear, where the user can submit the assignment and win the game.

Our project is a combination of a linear-storyline game with the abstractive story telling of an iconic space. We hope that combining both elements will provide users with an experience that is both sensory satisfying and competitively fulfilling.

Originally, the theme we chose for this project is "close the door". However, after implementing the project, we believe the scope of our theme expanded. The theme is touched upon when the internet connection is lost and the user is stuck in the Chrome environment with the dinosaur, with no escape other than to garner the courage to fight the dinosaur or succumb to their inevitable death. However, given the exploratory nature of our ideation process , the user can interpret the storyline the way they best see fit. Is the door really being closed after entering inside a computer or are we opening the door to a world where humans can live in a digital environment? Is the dinosaur a friend or a foe, acting out of spite towards an intruder or merely following the strings of ones and zeros that tell it to jump on top of every obstacle that lies in front of it? This are the questions we hope our users take with them, and we expect that our projects leaves users with more questions than answers.

2. *Process and Implementation:* discuss how you built the space and design choices you or your group made. What did the brainstorming process involve? How did you go about defining the logic of this world? What were the steps to implementing the user's role? Share images or sketches of visual inspiration or reference, including the storyboard/layout.

Idea Development:

Initially, our team was just excited to play with the idea of entering inside the Windows XP Environment. However, given the limitless provided by this idea, we relied heavily on several playtest sessions and feedback from different individuals in order to narrow the scope of our project to the point where it was playable, achievable, and entertaining enough.
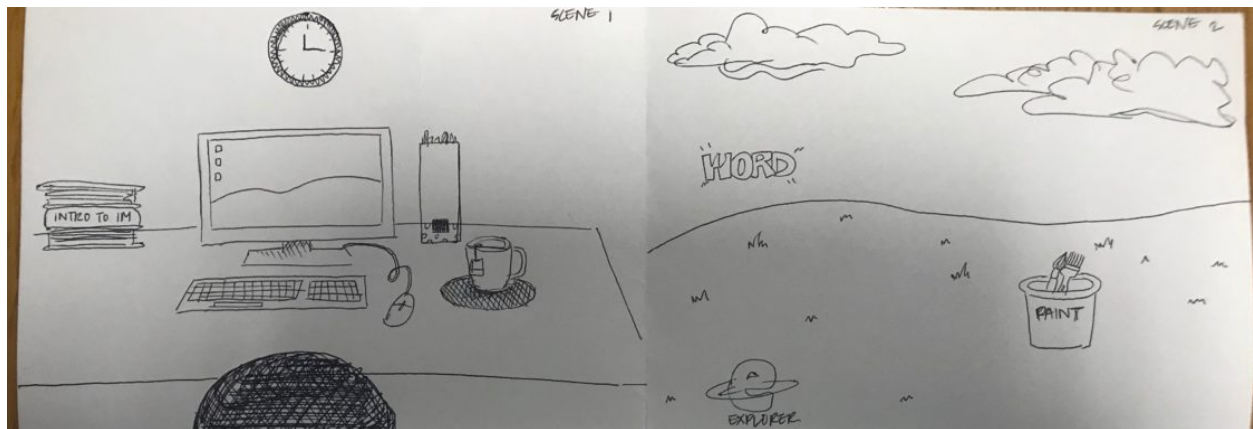

Prototype for feedback generation:
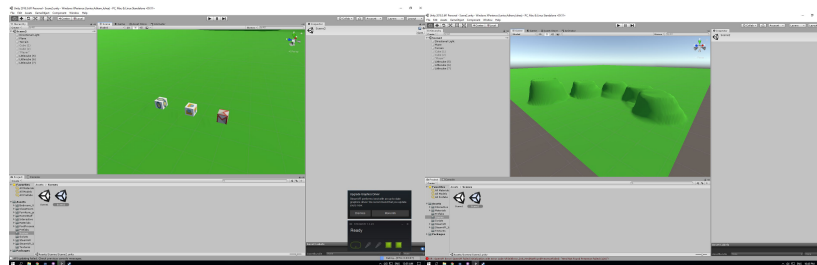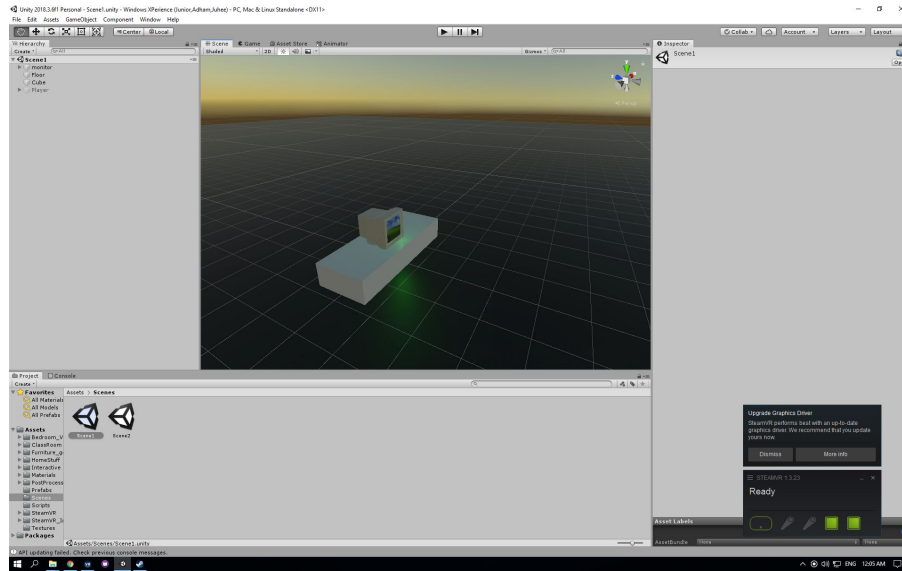
Our initial prototype we relied for user-feedback and playtesting was composed of two scenes. One scene was just a monitor and that was interactable and the second scene was just a green terrain with three computer icons.

We partook in a total of 4 playtest sessions and one idea-critique session with our Professor Sarah Fay Krom. After the playtest sessions, we decided to use the task of submitting the assignment in order to give the users some form of direction in this highly exploitable world. The storyline after the playtest sessions was the following:
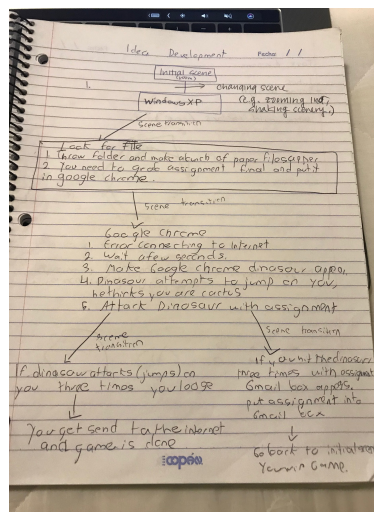
However, after talking with Sarah, we fixed several logical mishaps that were affecting our storyline. The first one was that the Google chrome Dinosaur signified lack of connection and it didn't make senses that the user would get stuck in the internet after loosing against the dinosaur. The second one was that initially we wanted to kill the dinosaur with the assignment itself. Sarah recommended that we instead kill the dinosaur with the cactus as this is what happens in the real-life version of the game. We incorporated this two pieces of feedback into our ultimate version of our storyline:

Storyline

1. User is in a room, the user clicks on computer
2. The user is inside the Windows XP computer
3. The user must get the assignment from the folder file
4. The user must go to google chrome to submit the assignment
5. The user is inside google chrome. The dinosaur approaches the user and attempts to jump on him/her. The user must grab the cactus to hit the dinosaur three times and kill it.
   a. If the user gets attacked by the dinosaur three times, then he is sent back to the initial room scene. Go back to step 1
   b. If the user attacks the dinosaur three times with the dinosaur, then the dinosaur disappears. Go to step 6
6. Gmail icon appears and the user must put the assignment in the icon. Game ends

Technical Implementation
Scene Design:
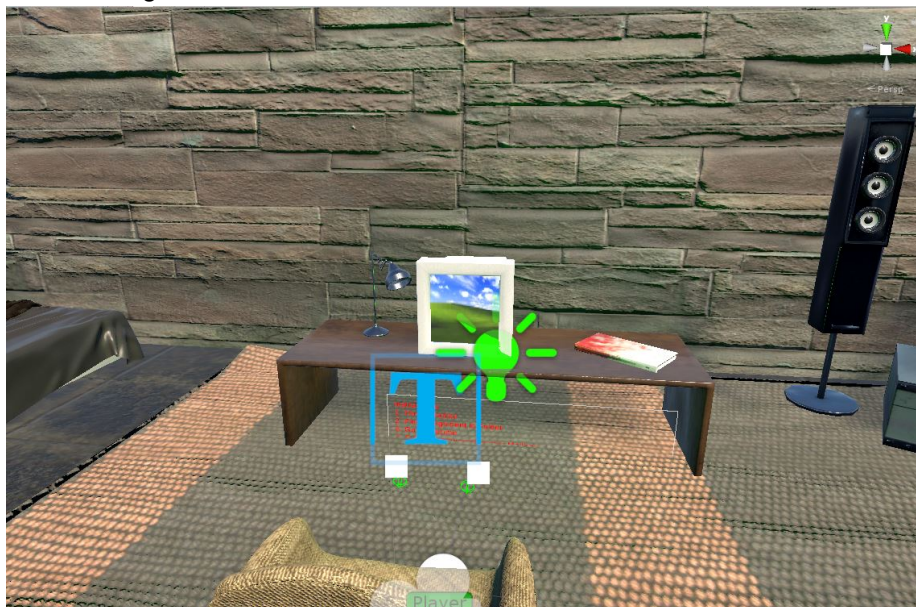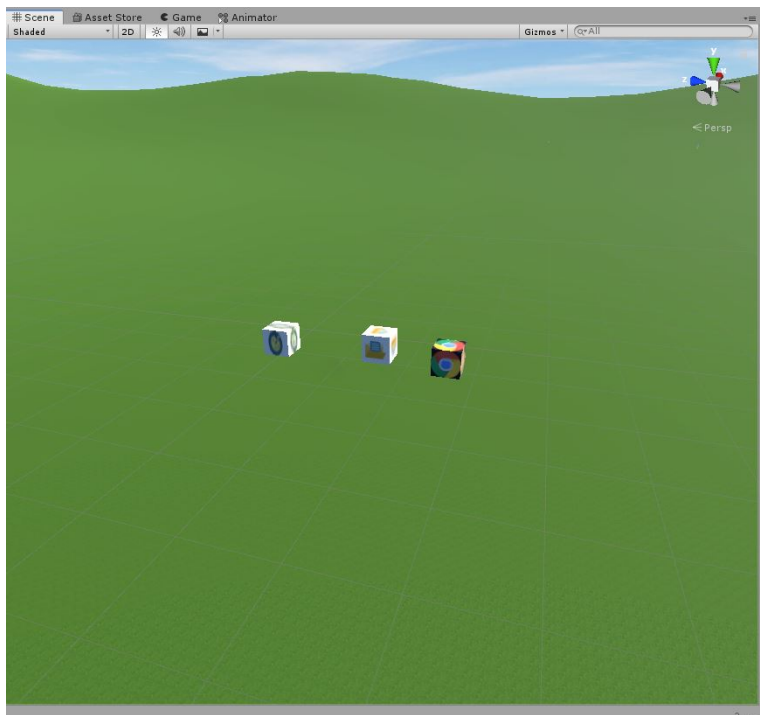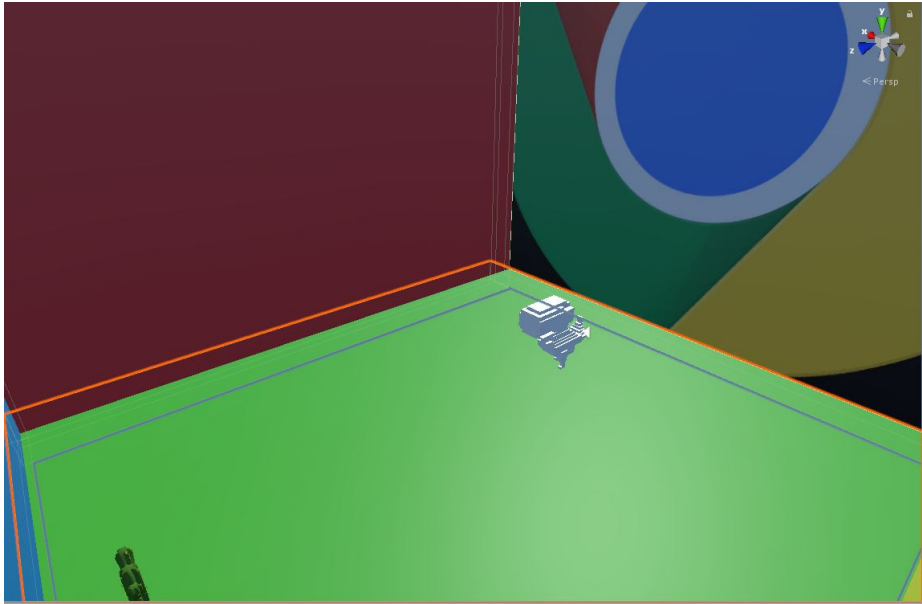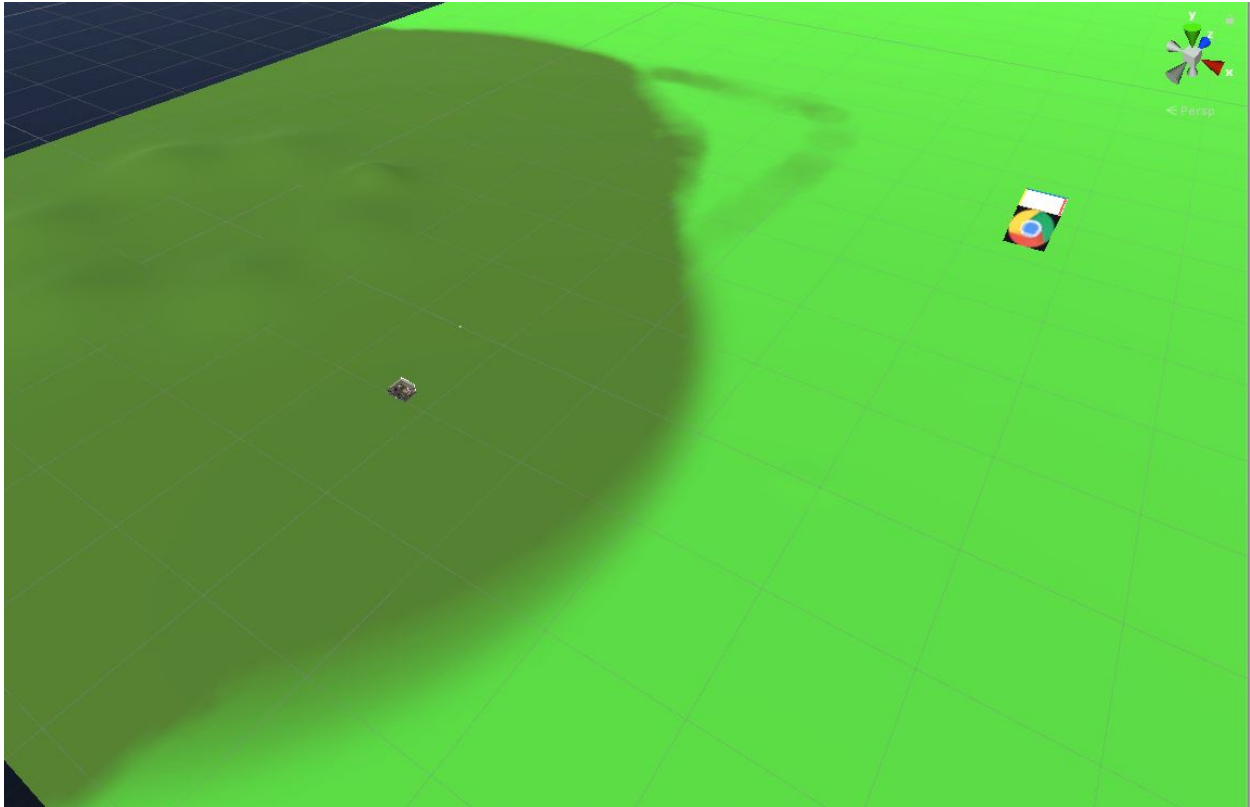
All three scenes were placed in the world and the transform.position of the player was changed in order to bypass the issues provided by the Scene Manager in Unity. The initial scene were the monitor was located in was designed with an early 2000s look in order to match the old-look of the monitor. The 2nd scene was designed to emulate the windows xp background, with mountains created by modifying the height of the terrain and the skybox changed with one found in the asset store that we believed best resembled the cloudiness and blueness of the Windows XP background. The third scene was a combination of a combat studio and the style of the Google Chrome's design. .We wanted to give the world an appearance similar to that of a combat studio because the user would be confronted, 1v1, with the dinosaur. As such, we went with a confined box environment, which creates the solitary, depredatory aura that we want the user to feel. We also decorated the box with Google Chrome's iconic green, yellow, red and blue colours on the sides of the cube, with one side being the google chrome icon itself in order to let the user know where they are at all times. The entirety of the project was accompanied by the Windows XP startup music in order to fully encapsulate the user with the illusion of being trapped in a Windows XP computer.

Scripts
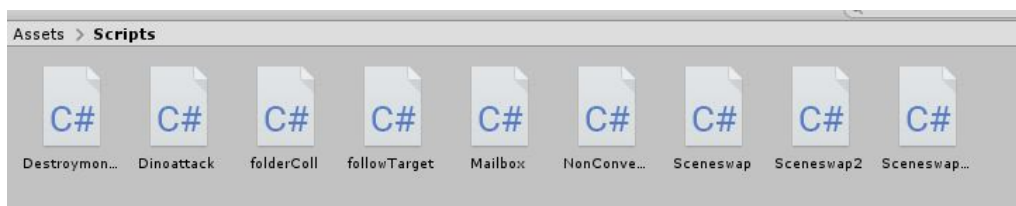
1. Destroy Monitor: This script is attached to monitor in initial scene to ensure it is destroyed after the player is moved to the next scene.
2. Mailbox Script: Script attached to the mailbox that ensures the game is restarted after the user wins the game by submitting the assignment to the mailbox
3. Follow Target: Controls the behaviour of the dinosaur and ensures that the dinosaur follows the user by using the Vector3.Movetowards function. A force is added to the rigid body to simulate the jumping. And counters (namely "lives" and "hits") are used to know the times the dinosaur is being hit.

Important code design in this script;

- Vector3.Movetoward: This function simplified my code immensely. It boils down a lot of the code that I did before to make the dinosaur follow the target as it literally makes the object follow the object with an incremental change in position after every frame and it also changes the rotation of the object so that it looks at the target.
- Rigidbody.addforce: Before, I was trying to literally hardcode the position the dinosaur had to be when it jumped, which seemed unnatural as the dinosaur would disappear and then appear somewhere in the air and then fall down. By setting the ForceMode to Impulse and determining the position in space you want the dinosaur to be after the jump, this line of code creates a natural jump of the object to the desired height.
- Quaternion. Euler: Something that was happening after the force was added to the rigidbody was that the dinosaur would move amongst its own axis in space. In order to stop this, I froze the rotation of the Game object using the quaternion.euler function in the Update function so that it remains in place within its own axis after every jump.
- Lives and hit counter: The lives and hit counters are an essential variable in the dinosaur that determines a lot of functionalities in my project. Whenever the number of hits is equal to the number of lives assigned to the dinosaur in the inspector, the mailbox appears. The mailbox then detects a collision between itself and the assignment to determine the win condition. This can be further appreciated in the mailbox script I attached to the mailbox.
- Coroutine and Delay function: I also learned about coroutines and implemented them in the script as well. I used them to cause a delay between every jump as the jumps were happening very frequently frame after frame. I also added the coroutine on the collision detection so that the number of hits between the dinosaur and the player was reduced as a lot of collisions were happening frame after frame too.

4. Foldercoll: Script that instantiates papers and the assignment when the folder icon touches the ground.
5. Sceneswap 2_1,scenesawp 2, and sceneswap: Changes the transform.position of the user whenever the user throws the monitor (scene 1 to scene 2) , the user puts the assignments in chrome(from scene 2 to scene 3). In scene swap 2 and 2_1, 2 detects a collision between assignment and chrome icon and sets the assignment collide boolean to true, which is then referenced be Sceneswap 2_1 to change the transform of the user to the next scene.
6. Dino attack; Script that keeps track of the number of times the dinosaur attacks the player. If the lives are exhausted, the user loose and the game is restarted.

3. *Reflection/Evaluation:* This should discuss your expectations and goals in the context of the what you felt was achieved with the finished piece.

All in all, the project succeeded in raising questions on the possibilities allowed by VR by allowing us, the designers of this world, to exploit the untethered territory of vintage technology by bringing it to life. We believe that this project was able to achieve its goal from an ideological perspective as it proved to spark questions and positive comments on every user we player tested it with. Indeed, most users started

questioning the reason why they had to fight with the dinosaur. Others saw the dinosaur as an immediate threat as evidenced by their screams after they first had the majestic creature in sight. Others would just spend a couple of seconds enjoying the scenery of the WIndows XP backgroundHowever, there are some technical bugs that need to be fixed:

● In chrome world, the assignment goes far away

In the hastiness of fighting the dinosaur with the cactus, some players would throw the assignment far away and wouldn't be able to catch it after beating the dinosaur so they could win the game. Maybe adding teleportation would have solved this problem, in order to give the user more freedom of movement. Also, teleportation would have made the final fight scene against the dinosaur more exciting as the user could escape the dinosaur and fight it at the same time. This problem could have also been solved by adding box colliders to prevent the assignment from going beyond reach.

● The initial scene is too big

Scaling our scene was a big issue. When we made the scenes, they were done at a very small size, which also affected the gravity of our objects. Resizing all of the game objects to an appropriate Unity-unit size would have made the experience more enjoyable.

● Make a more robust, complete UI system

Definitely, the UI we implemented in our project wasn't as complete as it could have been as we still needed to vocally guide users through our game despite the fact that we added the UI canvas with the instructions. Maybe incorporating audio feedback in every scene would have helped the users know what to do at every step in a more succinct manner.

In retrospect, this project was a nice way of culminating a semester long endeavor of exploring the possibilities brought forth by Virtual Reality and it empowered me to be an artist,designer, technological philosopher, and software engineer all at once.