**VR Project 4 – Puzzler**

This project is built based on the Udacity’s instructions, with the help of my friends on user testing inputs. I built this project for Android, the apk could be found in the GitHub link below.

The reason to build this project is to learn the VR design process taught by Udacity course 4, from start of design to the release of a production. As a software engineer in the software industry, this is a very useful skill for me to jump out of the engineer role, but to think as a designer. Therefore, I feel more confident to design features and GUIs of a project, and build successful production in the future. Another unique skill is specific to VR design, which requires me learning VR concepts first, which are totally new to me.

This project totally took me about 12 hours, including watching the course videos and hands on building in Unity, and getting feedbacks from my test users as well. Setting up test interviews, watching users play the game, and collecting feedbacks, and making changes according to feedbacks definitely took a lot of time.

**GitHub**

<https://github.com/martinsuchen35/NanoDegree-VR-P4-Puzzler>

**Final Results**

**Video of Game Play**



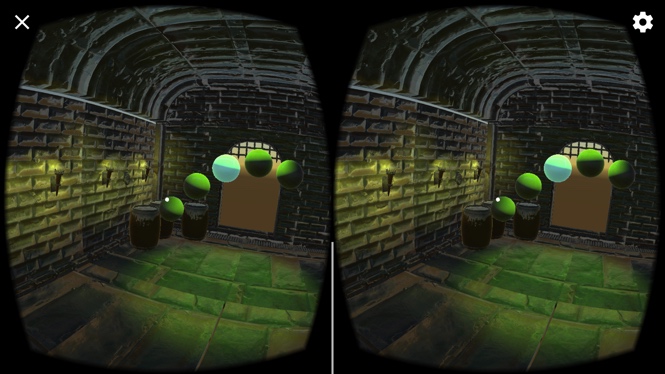
**Breakdown of the Final Piece**

The starting scene of this game is a welcome canvas.

Technically this is a vertical placed Canvas, which includes Texts “Welcome to Puzzler” and “Press start to begin” and a “Start” button. The button is listening to event “PointerClick” and once the event occurs, it triggers the method GameLogic.startPuzzle, which starts navigating the view to behind this welcome canvas, which shows the dungeon.



The animation will take the player navigate through the door of the dungeon, and enter the inner space. There is background music playing to create the gloomy atmosphere, together with the low intensity lights on one side of the wall. Five crystal balls are floating in the middle of the room, and the part I like most is there’s a green spotlight shed on the five crystal balls from above. Think about the witch from Oz, yeah, that’s the atmosphere I want to build.



Once player has been led to the position, the balls will be lit in a random order. That random order is the key that player must remember. The sequence has 5 items. Let’s label the balls from left to right as A, B, C, D, E, then for example, a random sequence could be BBCAD. The player needs to remember the sequence, and once the sequence has finished, player needs to click on the balls in the same order. If there’s any error with the sequence player is tapping, an error sound will be played, and player will immediately be disabled to continue clicking, and the balls will start being lit again, to remind player the correct sequence. This process keep going until user successfully reproduce the correct sequence. When user succeeds, a success sound will play as well.



Then, the game will lead the user to exit the dungeon through the back door. And show a canvas saying “Congratulations”. This concludes the game logic. Player could restart the game by simply clicking “Restart” button, and player will be led to the welcome canvas again.

I found the storytelling for user tests is very important. And here’s the version that I think attracts most of my testers:

“Once you start the game, you will be led into a dungeon, with 5 magic crystal balls floating in front. The OZ witch has encoded a sequence that the 5 crystal balls will be lit on in order. Your goal of this game is to escape the dungeon, and the only way to escape is to remember the sequence of the balls lit on and reproduce the order by clicking crystal balls. The sorcerer has some mercy that you can have unlimited tries, and will be shown the sequence again every time you fail.”

Once I told testers this story, they immediately know how to play the game.

**Statement of Purpose**

On one side, this project is for me to exercise the design process and user testing, together with programming with Unity as well. On the other side, this game is designed for people who are not familiar with the VR experience and who are interested in getting a feeling of it. Here I list persona of the 3 test users:

**Persona 1**

Name: Xiaoxi Guo

Age: 28

Occupation: Student

VR Experience: None

Quote: “I would love to have a taste of VR!”

Details: Xiaoxi is a business major student, and she likes to get familiar with VR applications, to grow a sense in the technical field, as she plans to join tech companies as product manager later.

**Persona 2**

Name: Jiangsha Meng

Age: 31

Occupation: Web Developer

VR Experience: Very little

Quote: “This feeling is amazing, I can really feel the depth of field!”

Details: Jiangsha is a web developer talented with Ruby on Rails, and he’s curious about how VR could affect the web world. So this puzzler is a perfect introduction for him.

**Persona 3**

Name: Changqing Wang

Age: 30

Occupation: Test Infrastructure Developer

VR Experience: Some

Quote: “I like this puzzler app, which has user interactions and scene cuts, 5 stars! I look forward to seeing more stages in puzzler. Some other feedbacks: 1. Could not figure out a way to turn off or leave the game, the little cross on the top left corner does not seem clickable. 2. Would love to see some moving objects in the app with even more realistic experience. 3. I remember some VR apps allow popping up a menu if long pressing the screen, you may consider similar implementation.”

Details: Changqing is a software engineer focusing on test infrastructure. He would like to check out the project that I’m building with Udacity courses, so he’s another perfect audience for this puzzler game.

**User Tests**

**Test 1**

**Feedback:** Xiaoxi thought the balls should have highlight when reticle hover over and a sound should play when player clicks the crystal balls.

**Details:** The app previously didn’t have the visual effect for “Pointer Enter”, “Pointer Exit”, and “Pointer Click”, and there wasn’t a sound when the ball is clicked. I think Xiaoxi’s feedback is very good since highlighting the balls when user looks at it and clicks it could help the control a lot, and it’s also a very obvious indicator for user to know whether the operation is really made or not.

**Test 2**

**Feedback:** Xiaoxi felt the light is too dim.

**Details**: The app previously only had 1 light on the wall, and it’s quite dark when player enters the room. I think it makes sense, because I also need to let players to see the barrels and other details in the room. If the light is too dim, bunch of details of the app could be ignored, making it not so realistic. So I added 5 lights on the wall, and also adjusted the light from above the crystal balls.

**Test 3**

**Feedback:** Jiangsha felt the same as Xiaoxi about the dim light, and also suggested the spotlight for the balls could be different.

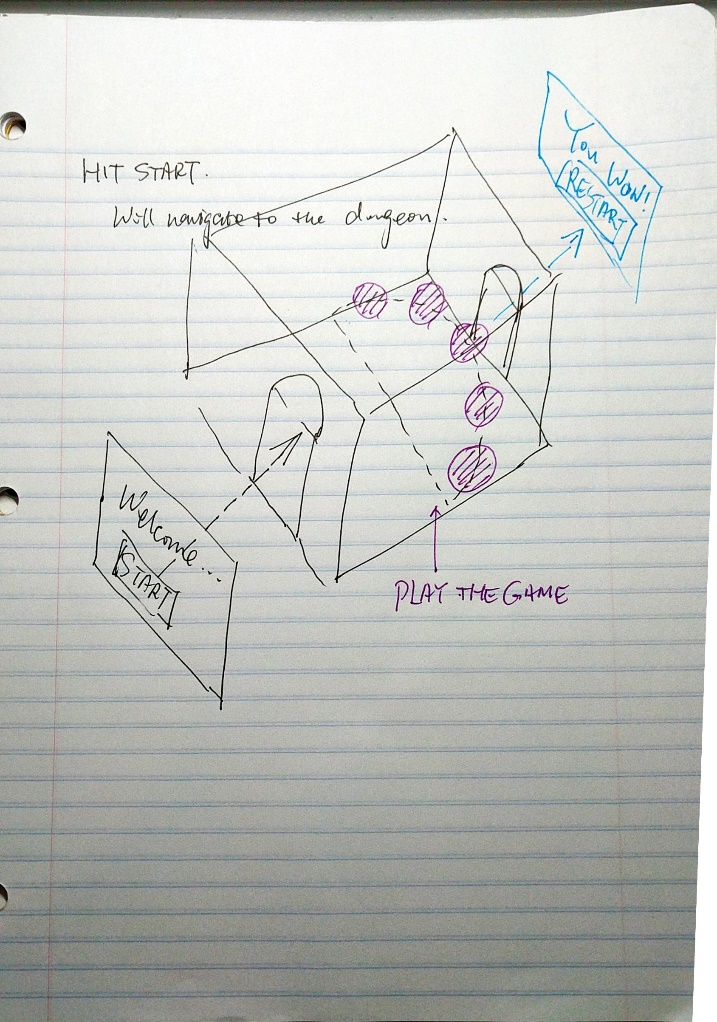
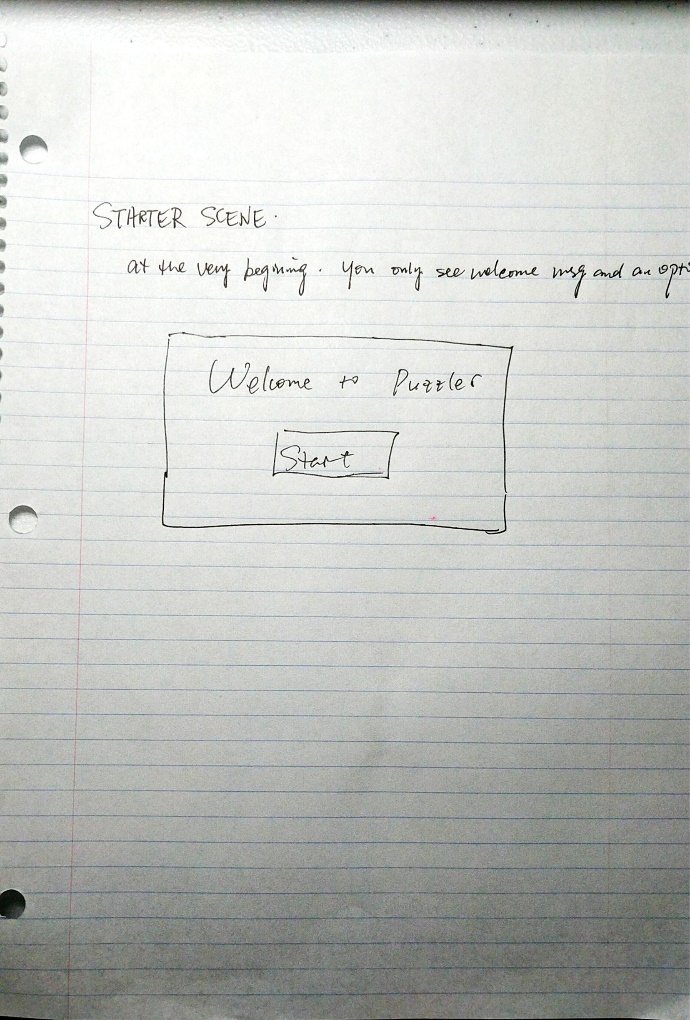
**Details**: For the spotlight, previously I chose a yellow light, which is same as the lights on the wall. Jiangsha’s suggestion is very good about atmosphere. So I tried several colors and finalized it with the green one, the idea comes from the witch in Oz.

**Test 4**

**Feedback:** Changqing suggested adding more moving objects in the room.

**Details**: I think this is a valid point to make app more fun. However, I chose not to do it, because I want the user to focus on the balls and the lit on sequence, which is the core of this Puzzler game. If it’s the apartment project, I definitely will take his advice to add some animated stuffs in the room. Like what MagicLeap’s concept video shows, I could add some floating objects in the room, for example a swimming turtle, imagine the MV of Coldplay’s Up&Up.

**Design Sketch**



**Conclusion**

**What I’ve learned:** I got more familiar with building projects in Unity, including laying out objects, writing scripts, and wiring them together. Got better idea on how Unity handles the object references, which is a fresh experience for me, as I’m a software engineer with experience in web, Android and iOS, all are about coding with 2D GUIs. This is the first time I write codes to control 3D objects, there’re similarities and differences. I’m very happy that I learned this course. Another thing that I learned is the design and user test process, which helps me to think of features from different perspectives.

**What could be improved**: The logic of the game could even be made more complicated, and there could be more reminders as the player goes. For example, when player enters the room, I could show a popup in the room to explain what player needs to do, instead of me telling them the stories, UI reminders could help a lot. There could also be a timer to finish the ball clicking, if time expires, player fails and needs to start over. Some of Changqing’s suggestions are also very good, for example, long press interactions.