Descent - survival horror game

A cooperation between students of MA Computer games Art and Design and MSc Computer Games Entertainment of Goldsmiths College.

Team consisted of 2 part-timers, 1 PhD and 3 working students!

Download: https://drive.google.com/drive/folders/0B-0eZh2vG7WOczcwZExKVWxUQkU

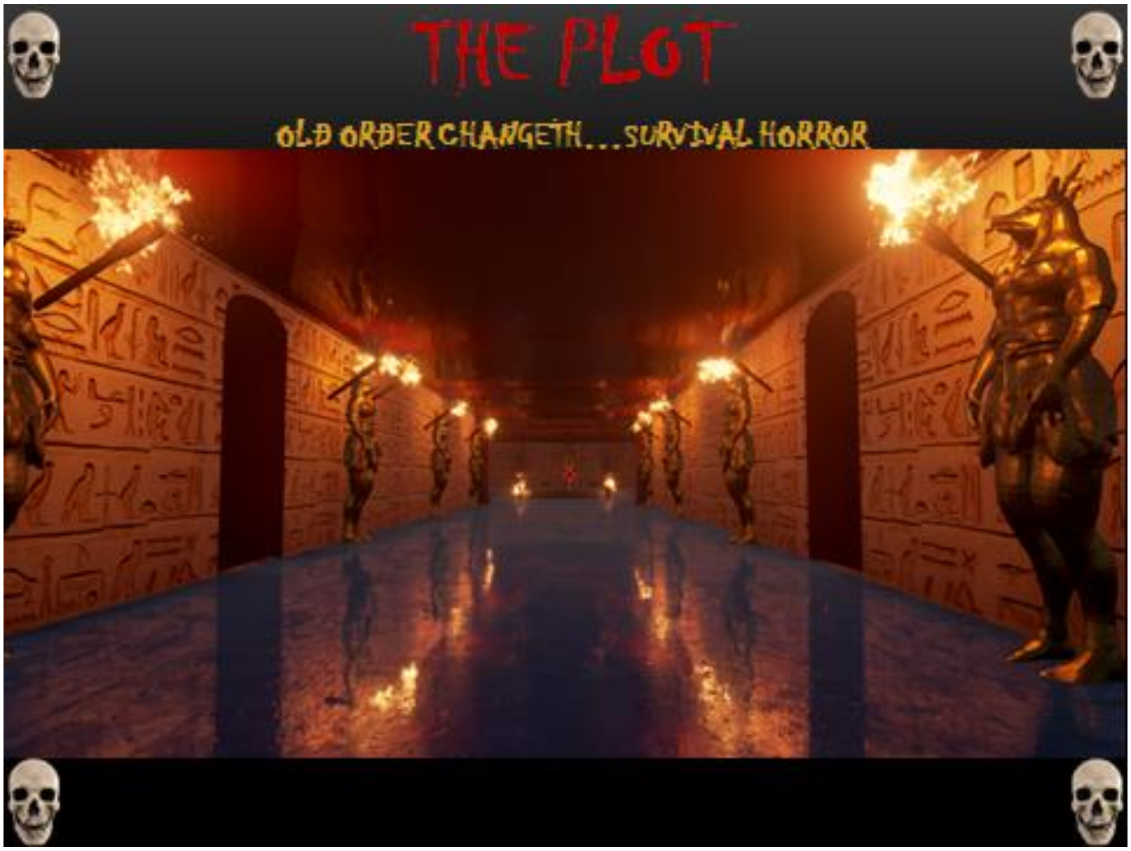
Video: https://www.youtube.com/watch?v=H7Oh1xJ3M38&feature=youtu.be

Download: http://tinyurl.com/z6nlkxv

Video: http://tinyurl.com/gm2hscm

Github Main Repository: https://github.com/parmandorc/Descent

`



The plot

We present a survival horror game that takes place in Egyptian theme. Player wakes up in unknown place and slowly discovers what is happening in the world around. We leave player with few clues on what to do in order to solve riddles, but leaving game to realistic feeling of unknown. Slowly Player learns, that the Pyramid is in fact temple haunted by Monster, who was once a Pharaoh.

During his life Pharaoh claimed that he deserves this temple and he should be called God of Afterlife saying he has power over the God, Anubis. He forced and convinced some Egyptians to worship him.

So, what will happen if you try to change old order of praising the Gods by mortals by ego driven Pharaoh?

Anubis killed infidels and cursed Pharaoh – he was now ment to roam the temple to witness and remember who is real God.

`

Now Anubis descended to his temple on earth. He has to gather items to prepare for ritual he always does – weighting feather against heart of Pharaoh. To do it, player needs to find Pharaoh tomb and other items in order to solve the puzzles/ riddles.

Technical details

Team used following software in order to finish the task:

-Unreal 4.14

-Zbrush

-Photoshop

-Paint Shop Pro

-Substance Painter

-UVLayout

-Fruity Loops 12

-Maya 2016-2017

-Git Hub

-XCode

-Visual Studio

-Leader did commits in Git Hub for whole team due to their technical difficulties

-Whole artists team worked on few blueprints, although they were eventually unused.

`

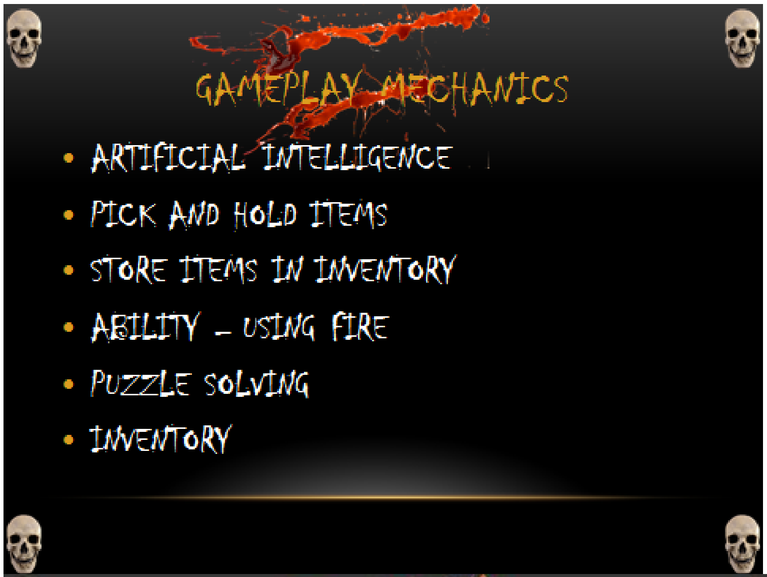
Game Controls

I – inventory open/close. Click on item in list to bring it back to world or see description.

E- pick item and store in inventory

Right Mouse Click – interact with world, hold item in hand

Left Mouse Click - Fireball



GAMEPLAY MECHANICS

- ARTIFICIAL INTELLIGENCE
- PICK AND HOLD ITEMS
- STORE ITEMS IN INVENTORY
- ABILITY – USING FIRE
- PUZZLE SOLVING
- INVENTORY

THE TEAM

Team leader Artist	Artist	Artist	Artist	Programmer	Programmer	Programmer
Olivia Grambo	Rui Fan	Tara Collingwoode-Williams	Faris Omar	Pablo Rodriguez	Alejandro Garzón	Louis Bennette

ARTISTS ROLES: MODELLING, TEXTURING, LEVEL DESIGN

PROGRAMMERS ROLES: PROGRAMMING, MIDDLEWARE, HELP

`

Inspiration

We took inspiration for game controls and gameplay from Amnesia games. Team agreed, that controls in Amnesia were interesting and familiar to most PC gamers.

Of course more obvious inspiration came from Ancient Egypt and mythology, which all team enjoyed a lot and agreed to. With a little twist we tried to stick to some reality.

Music and sounds

I decided that the background melody should be ambient only. After incorporating a melody it did not fit the mood and feel. I wanted a chilly ambient that would add up to the scariness of the game. Although I created a screeching melody in the middle of music, I believe it was a maximum that could be done without spoiling the mood.

I believed that not adding any sound for opponent makes it much more realistic. In reality we are never warned if there is a danger creeping upon us.

`

Future

We agreed in 4 people that were most of all engaged in the project, that we will continue our work on Descent, hopefully publishing it in future. We will continue to fix existing known bugs and polish the game as it is, only than adding new features and introducing new ideas or those we didn't have time to implement. Team of 4 is me Oliwia Grambo, Ray Fan , Pablo Rodriguez and Alejandro Garzon. We found working between us most efficient with outstanding level of understanding and together we love the vision and are happy to continue and do our best to make it.

Personal Comments

MA team

Oliwia Grambo – Team leader

I had the pleasure to become a leader of this project and a vision holder of the project plot and game design. I got a lot of valuable experience. Our team was very specific, having 2 part-timers and 3 people who are working , all living in opposite sides of London made our communication very restricted. We started off our actual work 3 weeks behind all other groups.

Very often communication was hard. I decided we would be communicating over whatsapp and skype, as we would save much time on travelling such long distances.

Being a team leader taught me how to solve unexpected problems and try to find other solutions to problems, especially from a design point. Many of our initial ideas could not be implemented so I was the one to make a quick decision on what to do – either to remove it from project, simplify it or change to something else. Along with my other responsibilities I had a lot of work to do, but as a leader I was doing my best to keep the pace, motivate my team and supervise their work, actively assigning new tasks or re-arranging existing ones if they found them too difficult to do – especially if I saw that they would not do it on time. I still believe that remote communication was a better option for this team than meeting up. I have learned how to manage 7 person team and assign tasks according to their abilities. I encouraged them to share their concerns and ask for help if needed. Even if I did not know how to help them I tried to look up information in internet to give them, hence speeding up the process of creating game. Often I had to contain myself and show a lot of understanding and patience. Taking into account personal circumstances it was absolutely crucial to manage workload effectively , otherwise we would fail the assignment.

At the very end of the project some of my team mates came up with new ideas, that I had to explain could not be implemented not only due to lack of time but due to no logical implementation within

`

project. I believe I am very lucky that my ideas were fully accepted and implemented by team. Almost all initial ideas and plans were completed.

The best feedback I received is fact that at least 3 people intend to work on this game further and developing it in order to sell it. I am also happy to meet the expectations of my team. I believe my team did brilliant job against the odds and were very cooperative.

What I made in project:

- Game concept
- Whole project in Unreal – setup of all items, landscape and some textures, foliage
- Look and feel – game design
- Puzzle ideas – slight consultation with team at the beginning
- Anubis human form – with help from Rui
- 29-piece modular kit – jointly with Rui
- Skull model
- Torch model
- Papyrus model and story
- Music
- Storyline
- Some of textures
- Enemy idea – initial model
- Uploading to Git all of our work
- Assigning tasks to each artist individually and general to programmers
- Placing riddles
- Supervising work
- Blueprint for door opening (unused)
- Decision maker (on everything)
- Decided on Scale
- Making all documents (except for personal thoughts-copied)
- 2d cover art/logo
- UV Mapping
- Normal Maps

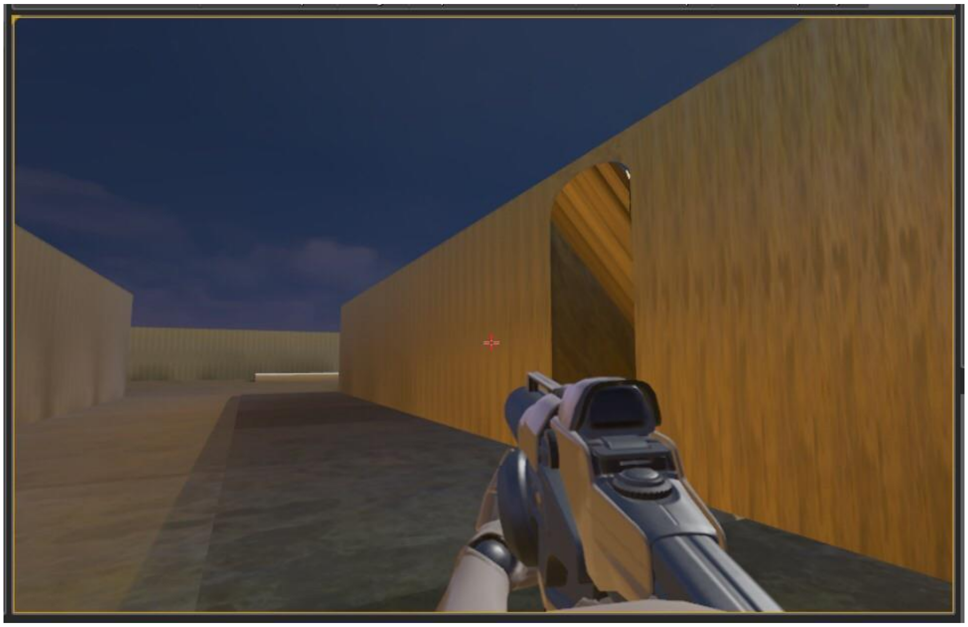
Skills learned and used:

- Zbrush – at the beginning of this project I had never used it before. It was quite an experience
- Unreal – same as above. I learned everything myself about Unreal and its requirements – for example how imported models looks like in Unreal

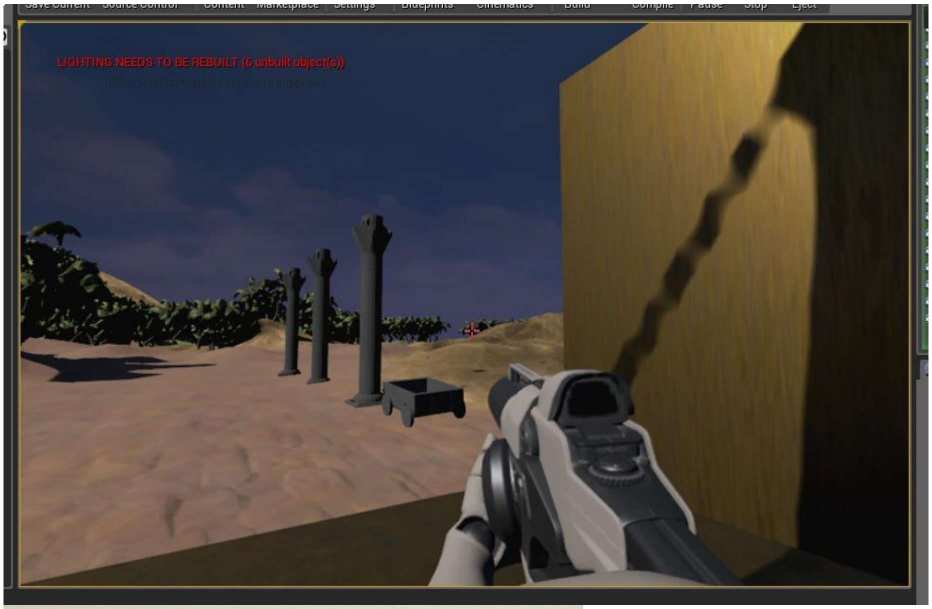
`

- Photoshop, Corel Paint Shop Pro
- Fruity Loops – I had only limited experience before the project

Early Stages, sketches, some models I done

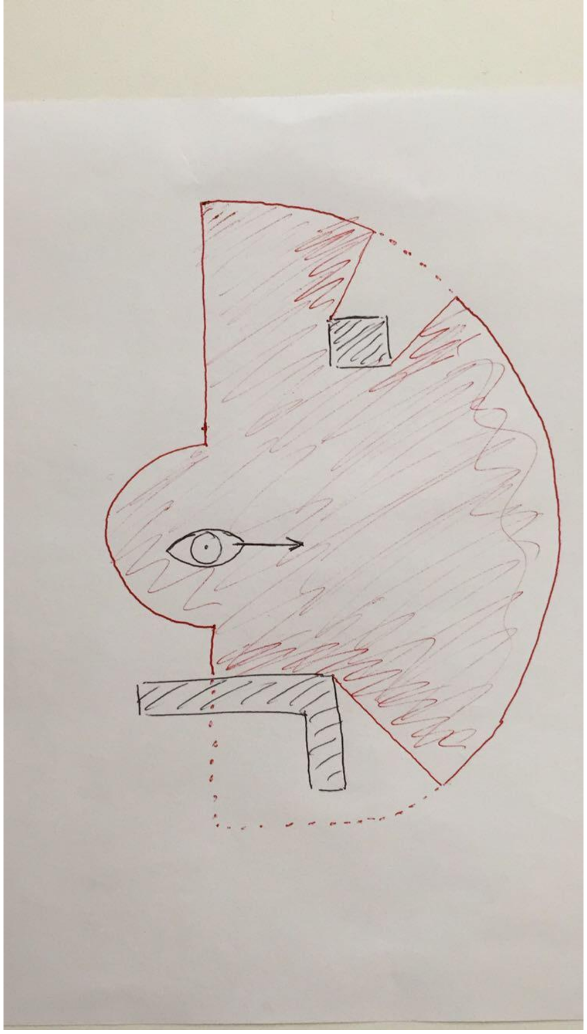


`

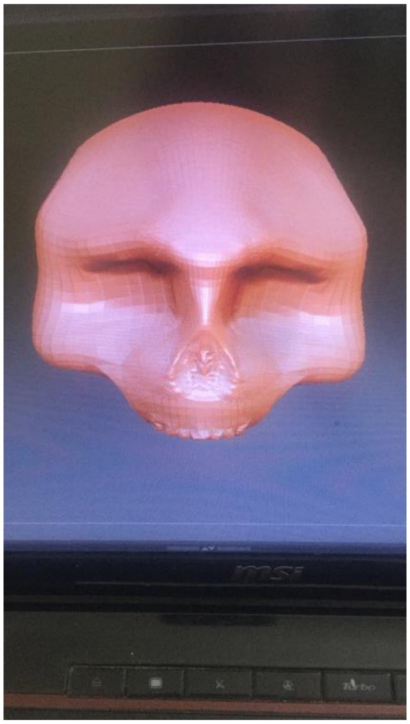


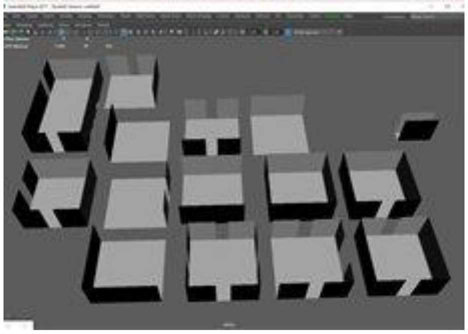
LIGHTING NEEDS TO BE REBUILT (5 unbuilt object(s))

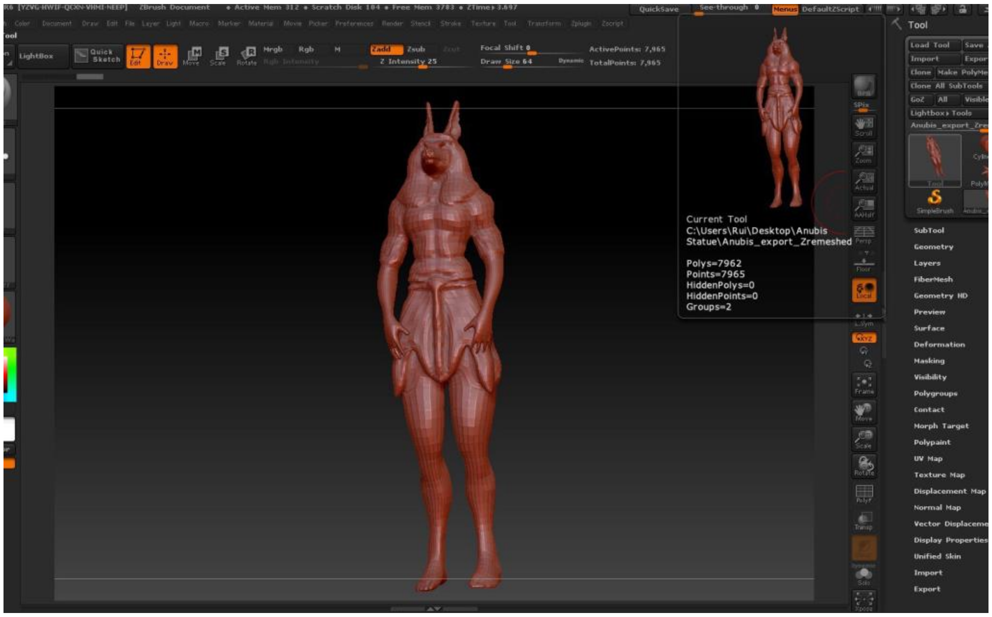
DisableAllScreenMessages to suppress



`

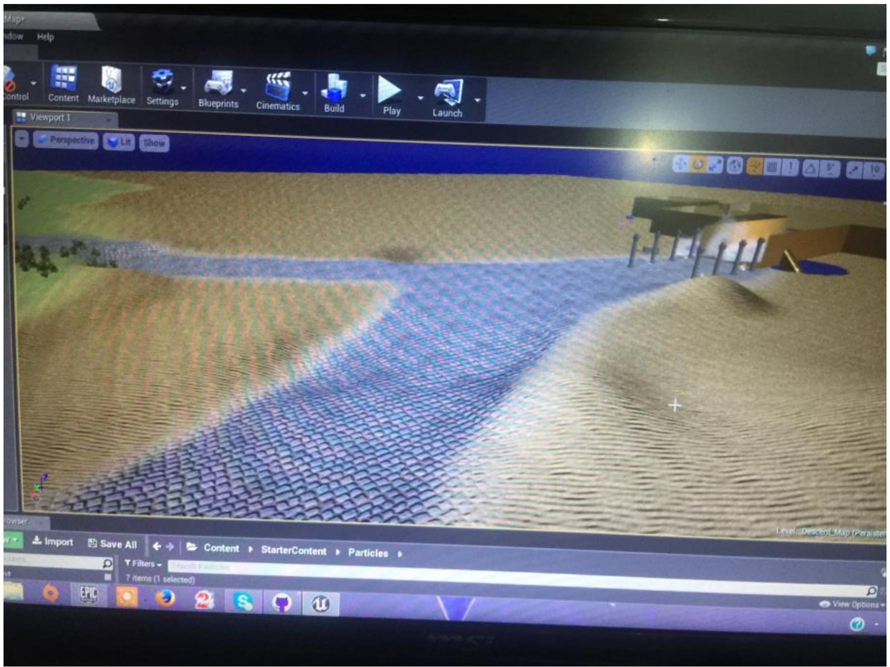


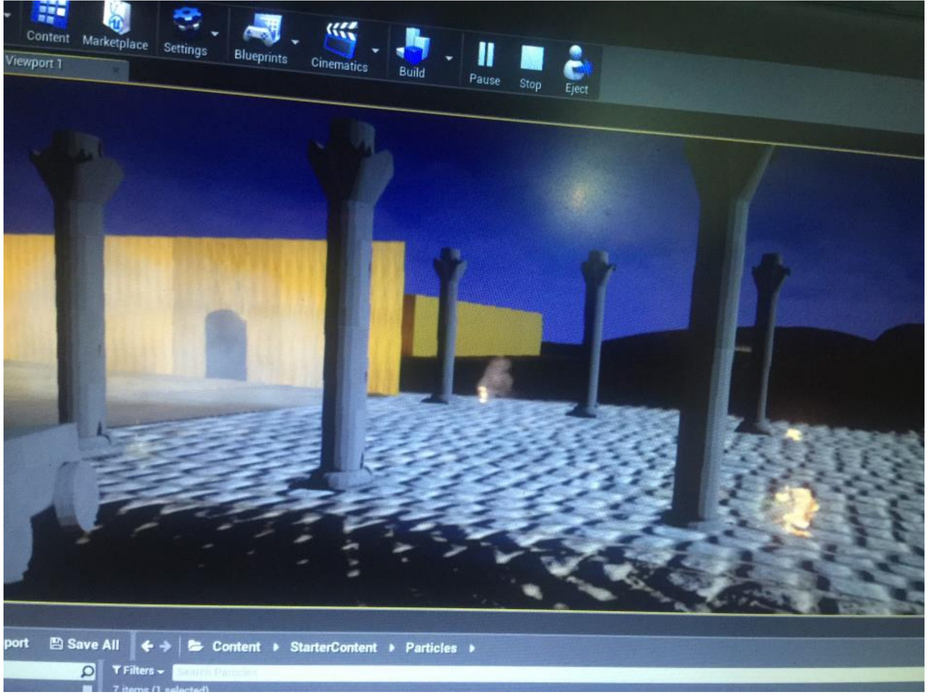




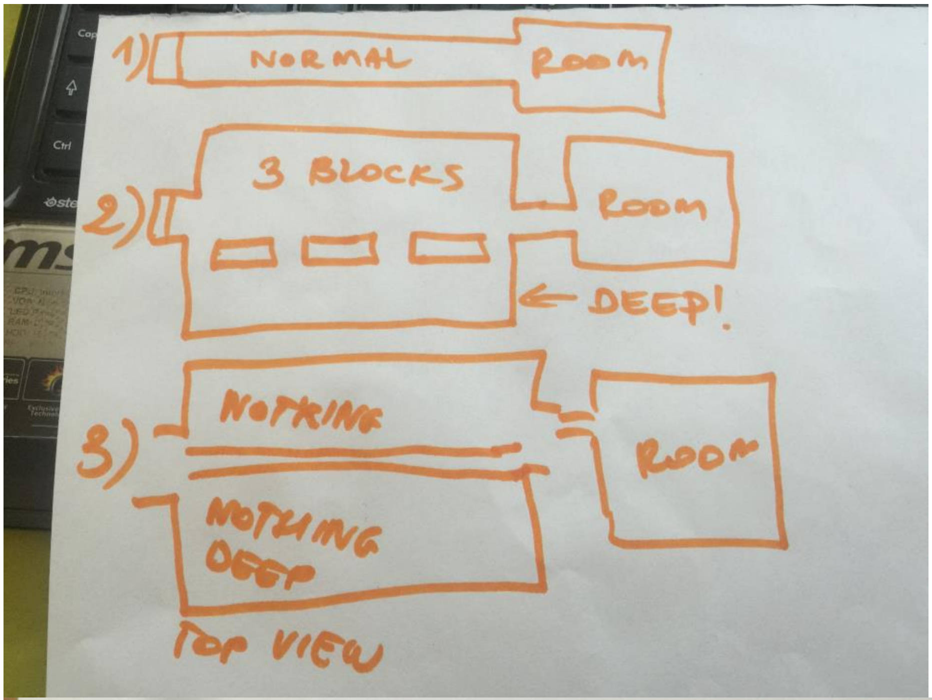
```
Current Tool
C:\Users\Rui\Desktop\Anubis
Statue\Anubis_export_Zremeshed

Polys=7962
Points=7965
HiddenPolys=0
HiddenPoints=0
Groups=2
```

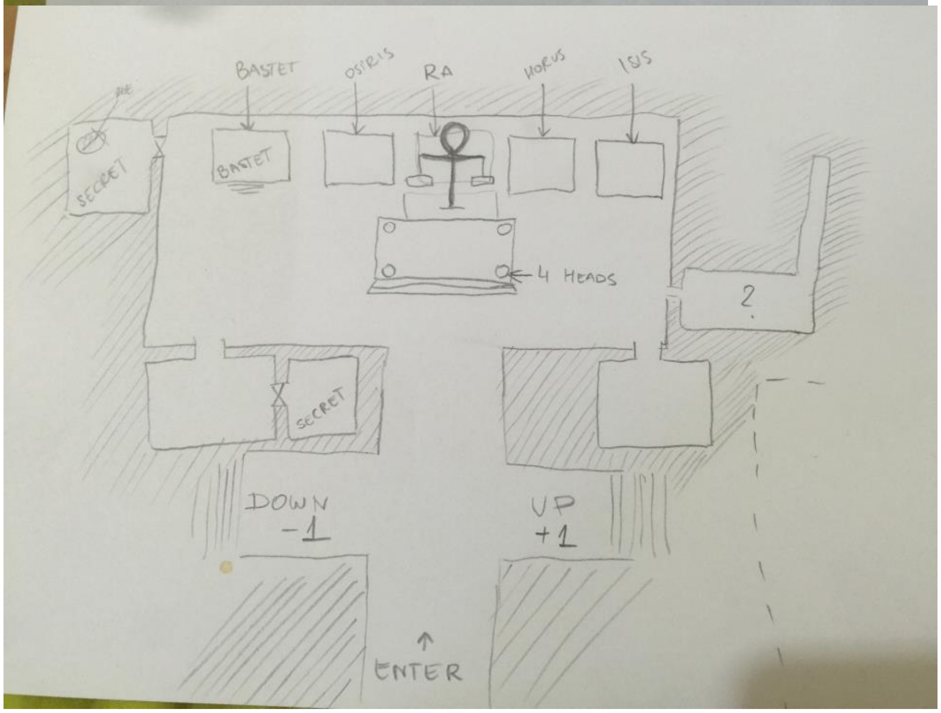
`


Help
Content
Marketplace
Settings
Blueprints
Cinematics
Build
Play
Launch
Viewport 1
Perspective
Lit
Show
Import
Save All
Content
StarterContent
Particles
Filters
7 items (1 selected)
View Options

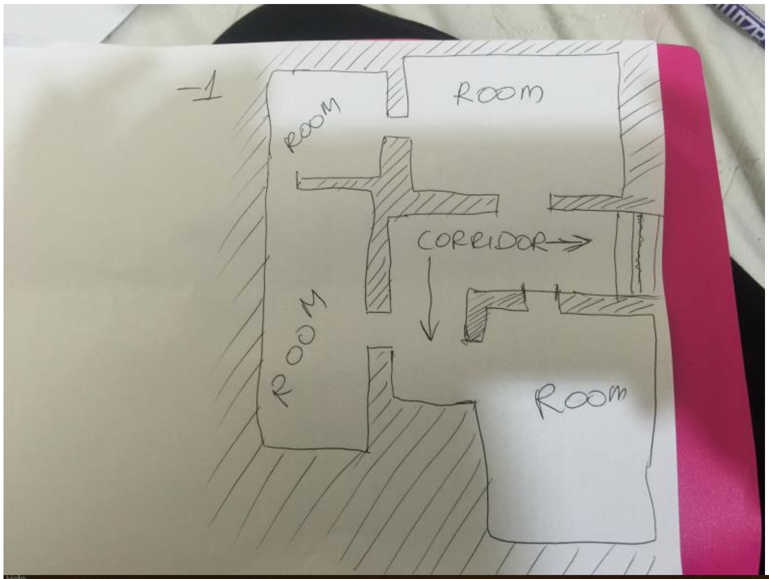

Content
Marketplace
Settings
Blueprints
Cinematics
Build
Pause
Stop
Eject
Viewport 1
port
Save All
Content
StarterContent
Particles
Filters
7 items (1 selected)

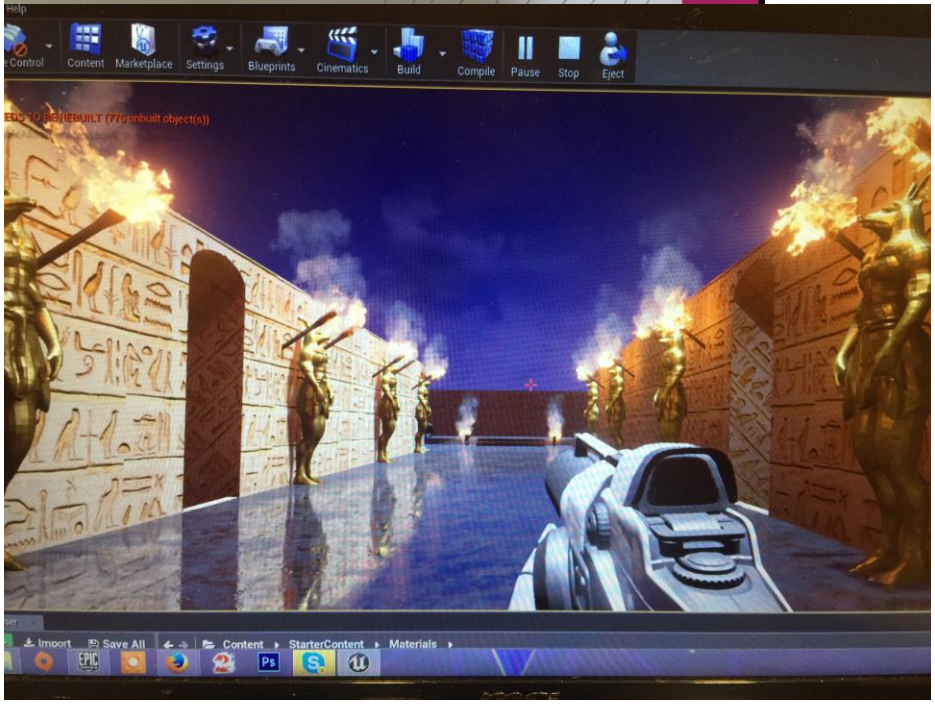
`


1)
NORMAL
ROOM
2)
3 BLOCKS
ROOM
← DEEP!
3)
NOTHING
ROOM
NOTHING
DEEP
TOP VIEW


BASTET
OSIRIS
RA
HORUS
ISIS
SECRET
BASTET
4 HEADS
2.
SECRET
DOWN
-1
UP
+1
ENTER

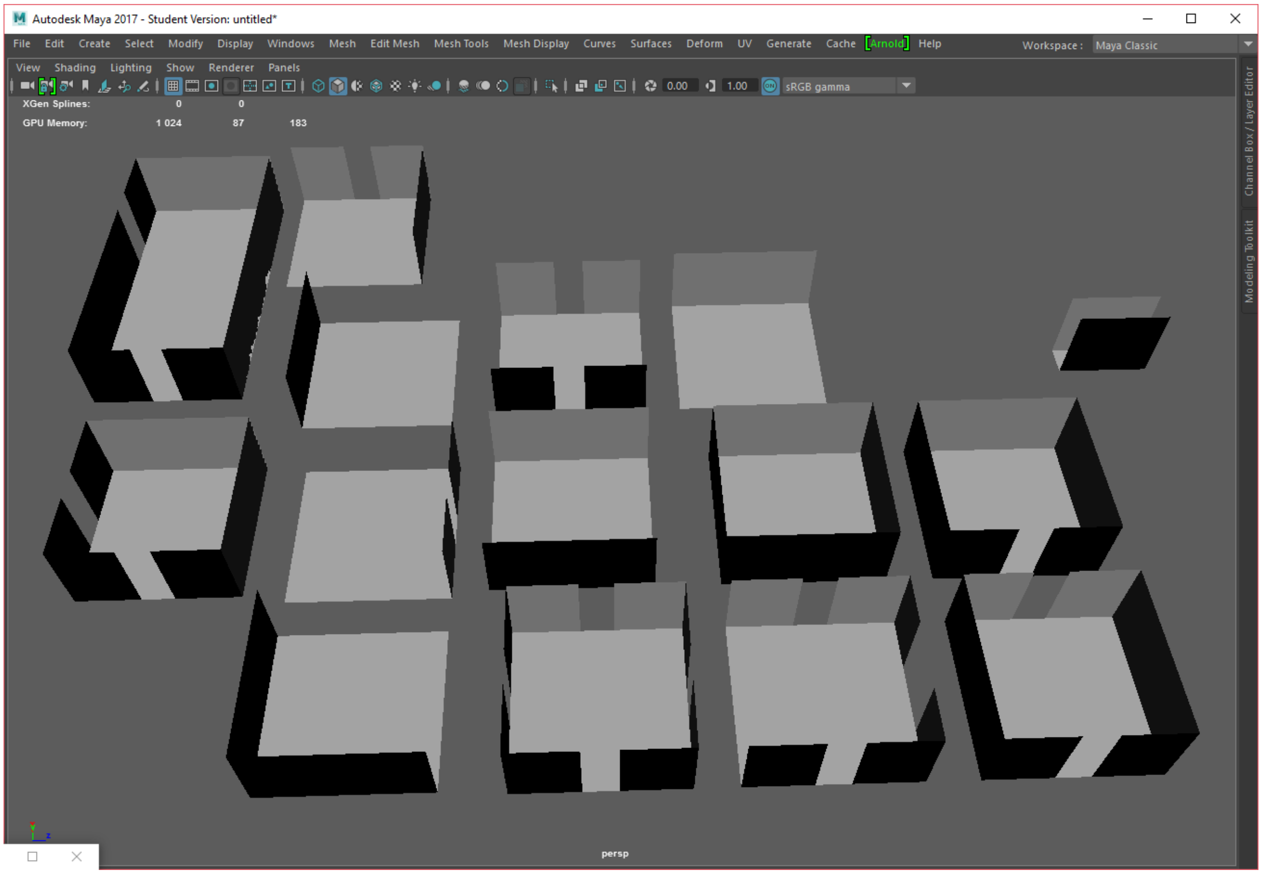
`





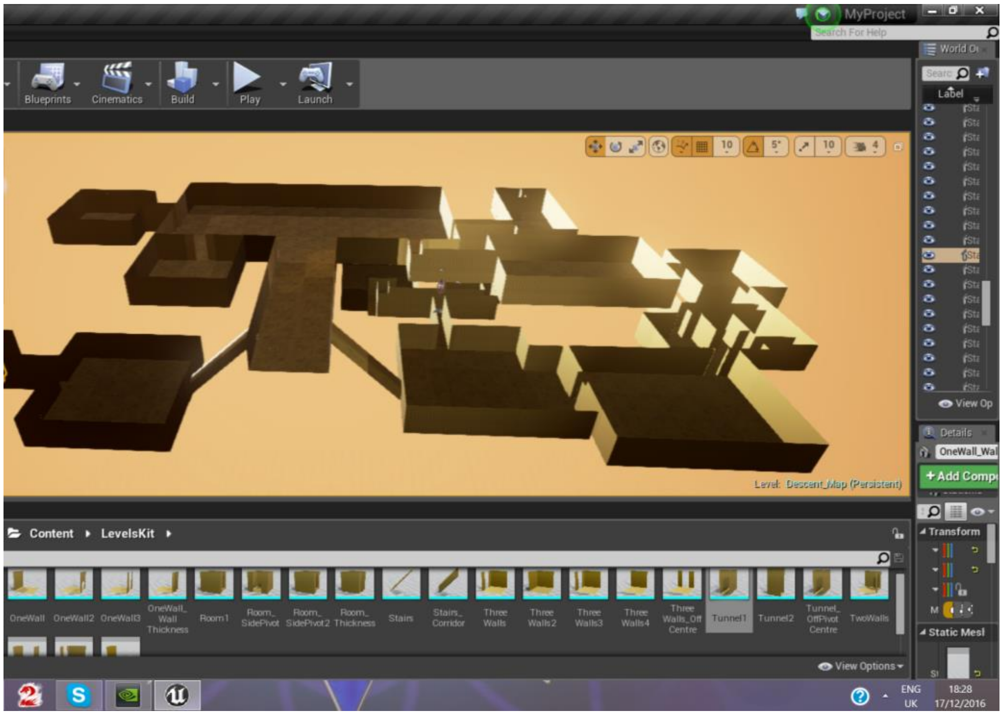
EDS TO BE REBUILT (775 unbuilt object(s))

`


Autodesk Maya 2017 - Student Version: untitled*

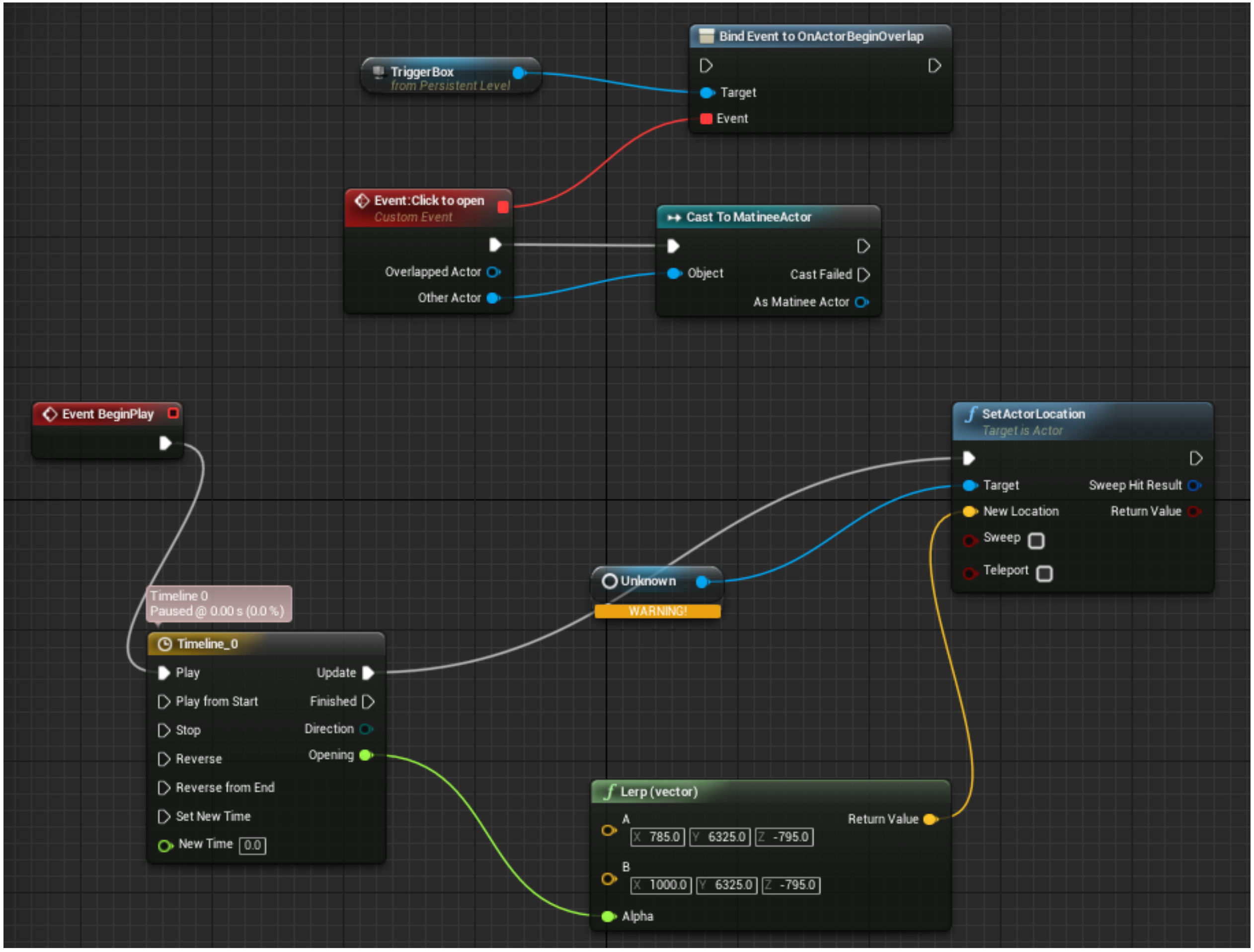
Workspace : Maya Classic

```
XGen Splines:        0        0
GPU Memory:      1 024       87       183
```


Level: Descent_Map (Persistent)

Content ▸ LevelsKit ▸

`



My blueprint for door opening (unused in the end)

`

DOCUMENTS OF OTHER TEAM MEMBERS:

Rui Fan

Rui Fan – MA team

Assets Created:

- Ra statue (undamaged and damaged with respective damaged body parts)

Ra was sculpted in detail in Zbrush, then retopologised into a lower poly mesh. The details were originally projected using Zbrush itself, however xNormal provided a much better alternative with editable cage for the projection radius, thus the normal map was baked in the latter program. The broken version of Ra was created by breaking the geometry and filling any holes created, then exporting the body, arm and leg as separate objects.

- 29-piece Modular Kit, with texturing done jointly by Oliwia

All modular kits were created in Maya, with some being identical replicas but with different pivot points to create a variety of possible combinations. The most challenging part is surprisingly texturing, because we used custom texture maps with the goal of making them seamless. This proved to be difficult since the material we used had shadows, which is unavoidable in real life photography. Upon reflection perhaps it would be better to use different methods (e.g. different filters in Photoshop) to remove unwanted shadow gradient.

- Anubis animal form statue

Anubis was sculpted in detail in Zbrush and retopologised into lower poly mesh. Normals were projected within Zbrush itself, while the maps were created using Quixel.

- Set dressings – Feather

The feather was a combination of two concave/convex planes with alphas. The texture for this was surprisingly big (10MB), but it is likely due to the extra alpha channel included in the image.

- Eye Model UV & texturing, initial model and texture done by Oliwia

Eye Model from Oliwia was created in Maya, while the UV was done in UVLayout and the original exturing was done in photoshop (by Oliwia) and later touched up by me using Substance Painter to get rid of unwanted seams where the UV breaks off the model's geometry

- Helped with model & UV of Anubis human form statue

I help retopologise the Anubis model into a lower poly version, and completed the UV in UVLayout.

- Sarcophagus (model and UV)

`

The model was created using a combination of Zbrush and Maya, where the human features were exported from Zbrush and the rest (lid and box, staffs, headdress) created in Maya.

- Heart

Sculpted in Zbrush with projected Normal maps also from the same program.

Technical skills acquired:

- Zbrush skills, particularly retopology and Zremesh guidelines, as well as using multiple subtools
- Using Xnormal instead of Zbrush for high detail normal projections
- UV in Maya, functions such as symmetry and conform
- Basic use of Substance Painter for texturing

Thoughts on working in a team

- I was working alongside the team leader, who is also the vision holder for the game design. The team leader is very patient and helpful. Overall the team worked great and overcame many obstacles in order to produce a good quality game with good mechanics and AI programming. I do feel that the team could benefit from weekly team communication via Skype to update each other on our progress towards the game as individuals.

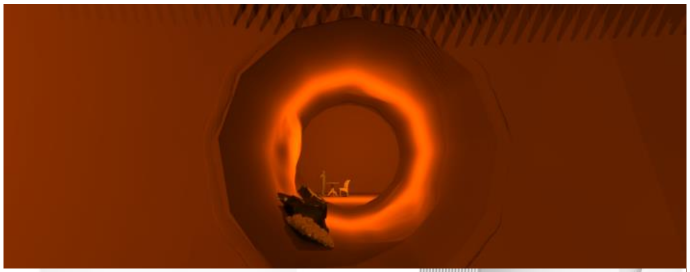
Faris Omar

For my first task was to create a cave that was initially meant to be implemented in the game level. The task was to create a cave with three rooms inside of it. The picture shown below is the initial cave design.

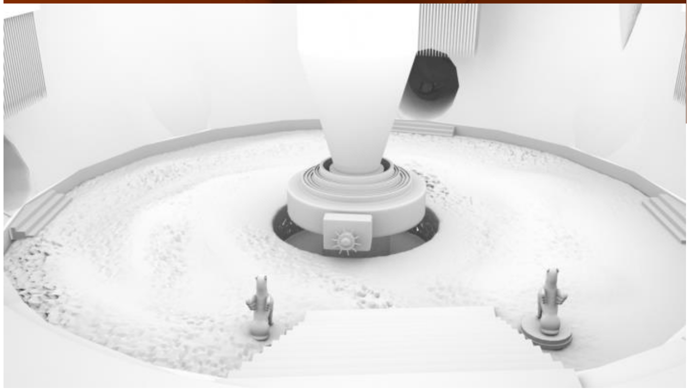
a) Entrance Of The Cave done with Zbrush.

b) Further into the entrance

Allows user to view the pillar.



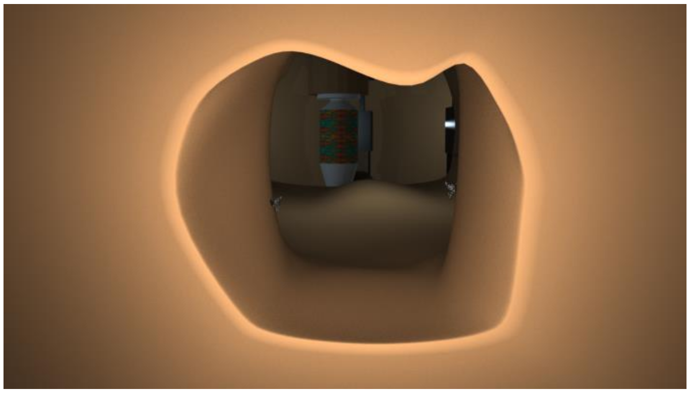
c) One of the sub rooms in the cave design.



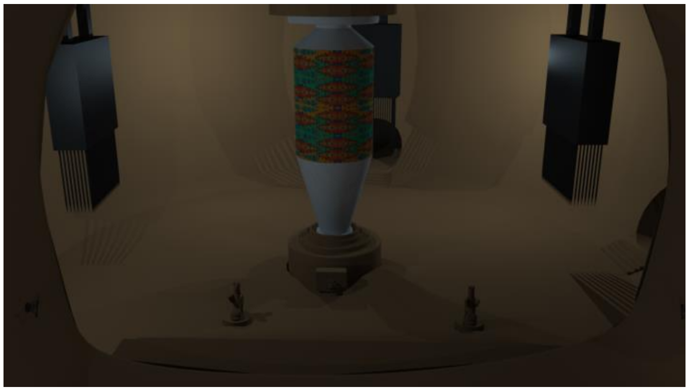
d) The middle area of the cave where the entrances to all three sub rooms are.

The entrance of the cave was done through Zbrush in order to achieve a more realistic and uneven cave look. The stones and pebbles on the floor and in the sub-rooms' hallways were also done in Zbrush. At the entrance of every sub-room, there are spears hanging above them. This meets the gameplay requirements for puzzles. The pillar was done in Maya 2016 by reforming a tube shape into a pillar-like form. The block that is connected to the pillar was also done through Maya 2016.

After having further discussions with the group leader, the cave had to be altered as it did not meet the game requirement as the topology count was too high. I then proceeded into making a simpler cave with less topology. Below are some pictures of the reinvented cave.



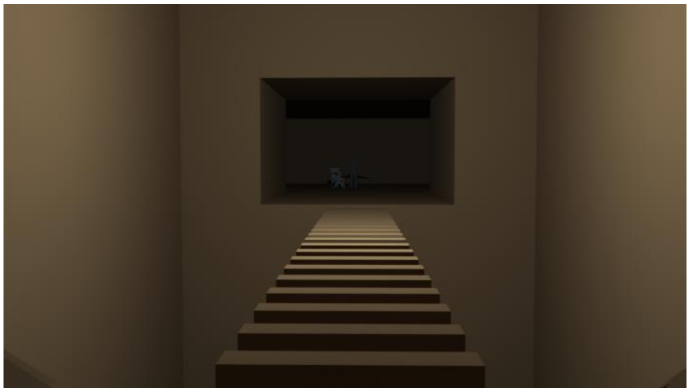
a) Reinvented cave entrance



b) Reinvented main hallway (without stones on floor.)

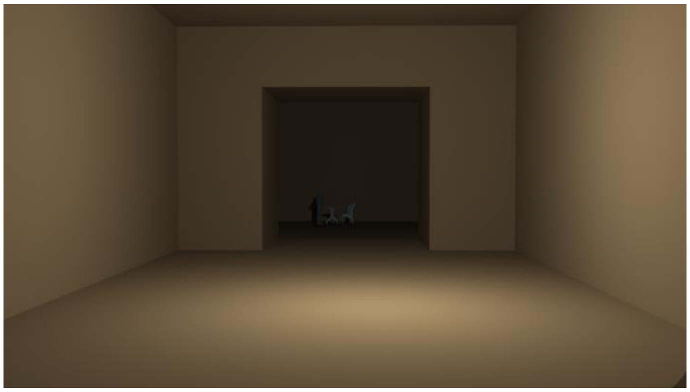
The major thing that has changed in the reinvented cave was to remove everything that had been done in zbrush as i found that the cave entrance was the most expensive in terms of topology. This new reinvented cave was done without the use of zbrush but only by using Maya 2016. Entrances and the hallways of the sub rooms have also been done in Maya instead of Zbrush.

The next task that was given to me was to set up the paths that lead up to the room. Each room has different ways of reaching the room. Refer to the pictures and brief explanation below.



a) First rooms shows that users have to jump on those steps to reach the end of the room.

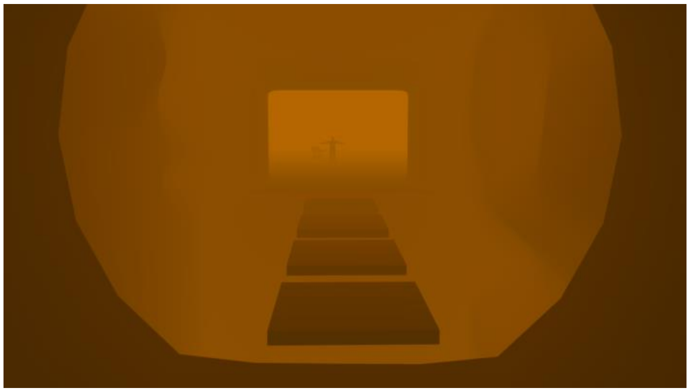
b) Second room shows that users have to walk the straight pathway in order to reach the end of the room.



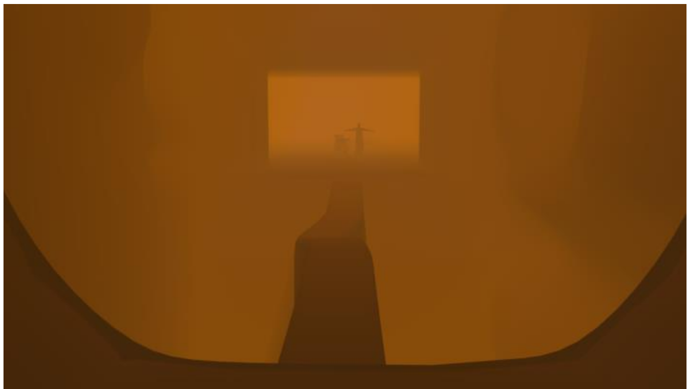
c) There were no changes yet to this room at the time as we had yet to discuss on what to do for this room.

Based on the first room the user is supposed to get to the end of the room while hopping on the steps. If the user falls, then the user will fail its mission. For the second cave, the user is supposed to walk on the pathway to reach the end of the room but at the same time will face heavy wind from the sides of the walls so the user has to maintain balance on the path giving it a sense of challenge.

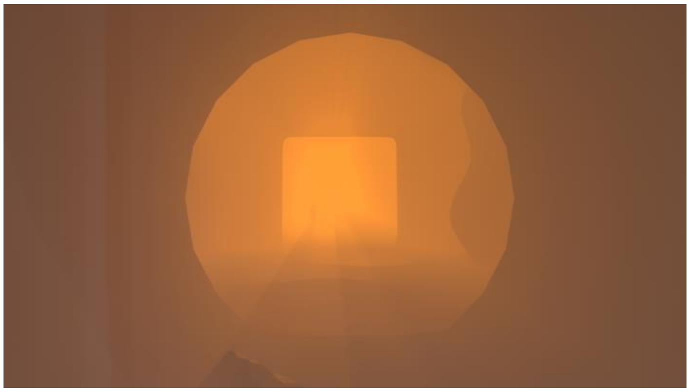
After having a discussion with the project leader, i was given the task to change certain parts of the rooms. Refer below for the changes made.



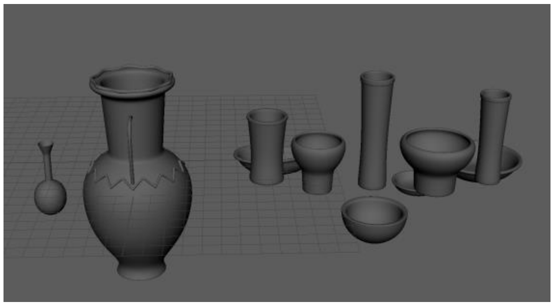
a) The change that has been made to the first room is to decrease the steps as having too many steps would make it less challenging for the user. Also the decision to make the steps like a floating block is because users now have a time limit for how long they can step on the step before it falls.



b) As for the 2nd room, the only major change here is to make the path narrower to make it more challenging for the user.

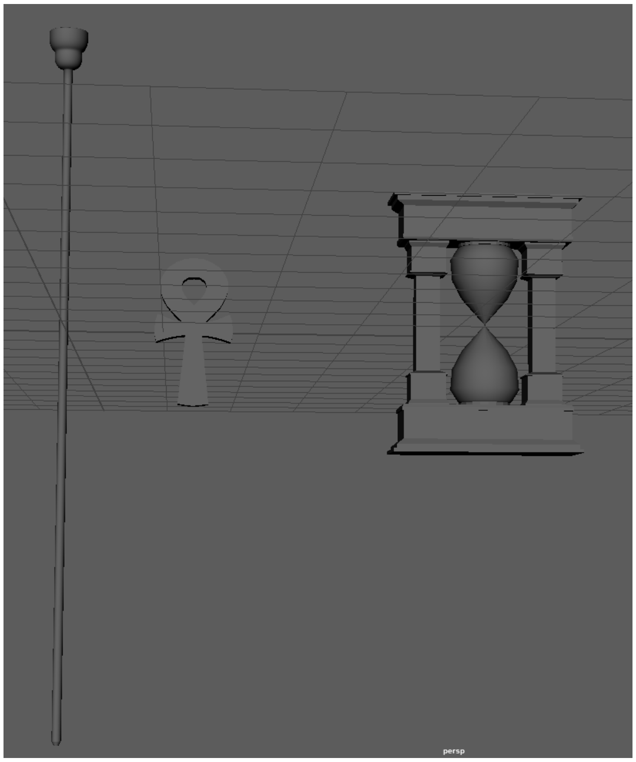


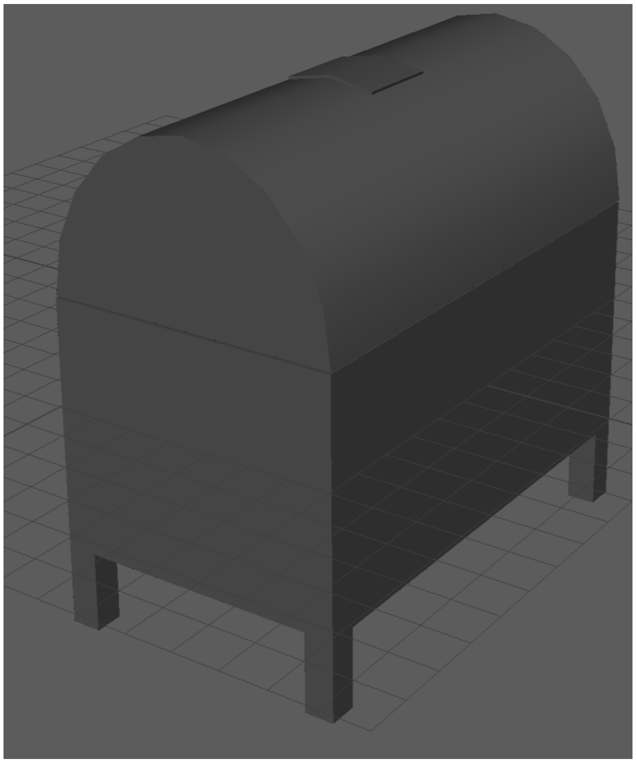
c) As for the third room, the decision was to make the path as a zig-zag pattern. If users cross the zig-zag path then the user will fail its mission.

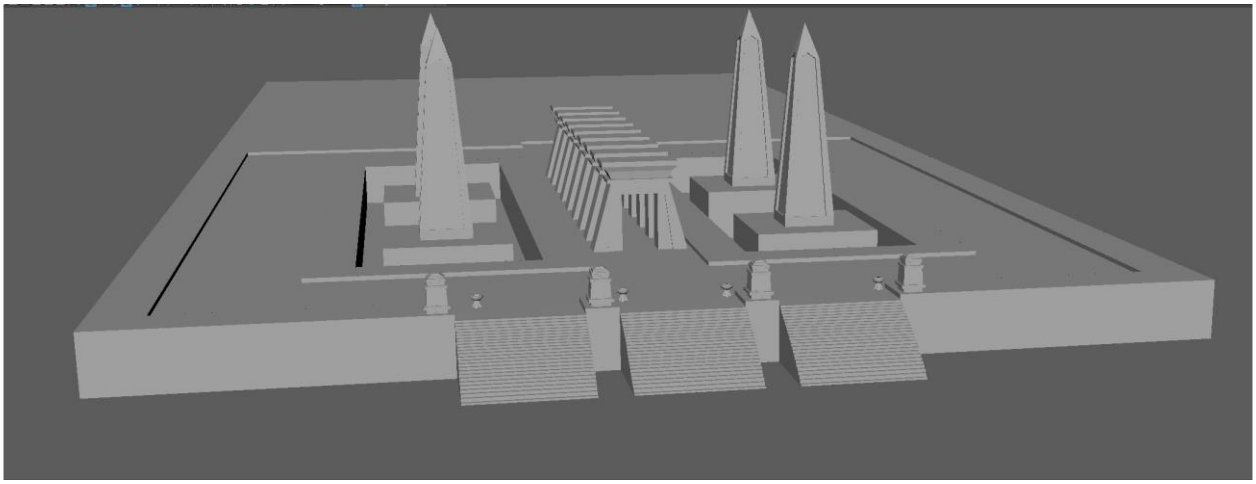


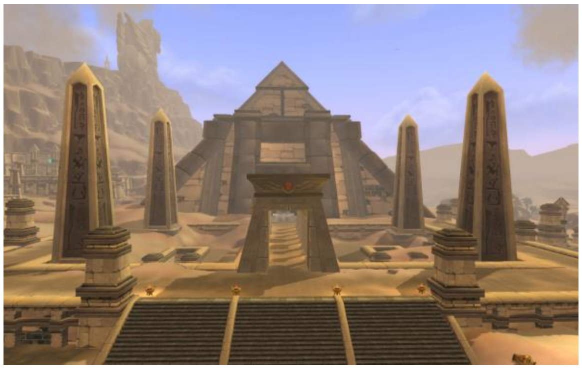
The whole group was given the tasks to create as many decorations to fill up the game levels. What i have contributed were vases , cups , plates , the cross that Anubis holds , and many more decorations. Below are pictures of decorations that has been done.

`









The last model was inspired by this picture.

I did this model to be used at the front of the cave. It wasn't that difficult of a task as the shapes in the picture were very simple to model. It was eventually used as a separate instance on map.

`

Challenges, problems and reflections.

The most challenging aspect of this project was modeling itself. Having no background on modeling made it a challenge for me to keep up with. This caused many time delays in the group time frame as i would have to achieve my tasks by the given time. However i have managed to meet some of the tasks and able to hand in my part of the project in time but most of the times i had to delay the team due to not being able to complete the models in time.

Other than that, the biggest problem that happened during the group project was the cave that i made. The cave was supposed to be another chapter of the game but as the cave was producing too many bugs and instability issues, we had decided to remove the cave from the game plan as it was taking up too much of the development time. The group was then focused on making the other game level to produce a better level than the cave.

However, doing the project i realised just how much i thoroughly enjoy creating model content especially if its being made for games. The learning process was challenging but i enjoyed everything that i have learned so far about modeling and creating content for game level designs. I was also fortunate enough to have a group leader that kept the team together and up to date and also helped me in the process of creating content for the game level design. Overall i am happy with how the project went as it allowed me to experience and improve upon my team working skills and also improve and challenge myself in relation to my game level design work.

Tara Collingwoode-Williams

I'm a PhD Student studying Embodiment in Virtual Reality. I come from a core computer science background, studying Creative Computing Bsc at Goldsmiths University of London. During this Introduction to Modelling and Animation Module, I learnt how to use the Maya and Blender interface, but also how to take advantage of the different workflows in each of the programs for achieving different tasks. Following getting used to the interface, we began exploring the best practices for modelling objects/assets for a game. This is taking into consideration cost (or polygon count) and also creating a model that is created efficiently. We worked on expand from simply shapes to more complex ones, and using the 'bevel' or 'add new edge loop' tool to curve edges and add detail. At this point I had decided to do most of my modelling within Maya due to my slight familiarity with its interface and transform tools.

Another level of asset creation was the addition of texturing. In this we looked at UV mapping; using the unwrapping workspace to create a texture map that can then be textured in Maya itself or in more detail in an external software like Photoshop. I struggled slightly in getting used to working with seams and creating the detailed texture however it was interesting to see what different effects that could be created, especially when adding normal maps to add depth and context to the texture material. To create normal maps, I used a software called 'CrazyBump.'

`

Later in class we moved to unreal engine and explored blueprints and creating toolkits. As a unity developer it was interesting and challenging to switch over to a different workspace and I still need time to get used to it. For our group project it was decided to go with an Ancient Egypt theme. It began as a shooter but then developed into a puzzle adventure with aspects of horror. We sat and discussed what puzzles we will design and add to the project and also the levels which would be involved.

We split the levels into three. I designed a level using Maya. I used a set of previously created model corridors and walls from my colleague (Rui) to customise a floor. Here I had to exercise the 'extrude' tool and also the 'add new edge' tool. I also worked on the game concept, asset modelling and texturing, the main document, blueprints and supporting code for an event for a trap.

In terms of assets, I created and textured (some of which weren't implemented in real game)

- A cart
- A vase
- Cylinder Column
- Squared Column
- Spikes
- Level

These models were made in Maya. The textures were found on the internet.

Sounds:

- Fire throw(ablaze/fireball)
- Ghostly sound
- Door opening gradually
- Door opening slowly
- Torch fire

Sounds were acquired on freesounds.org.

Additionally, I added to the main document, providing background context on the game influence, a run through of the game mechanics and long/short term objectives of the game as well as explain how we plan to market the game to the chosen target audience. I also added info on the character Anubis within the game.

Within Unreal 4 engine, tried to explore different methods of activating or setting up the Spikes within the corridors of the level. I first used the (legacy) 'matinee timeline' tool to create an animation timeline in which for 5 seconds the spikes will rise and sink into the ground in a loop. In this way the player will have to time their movements in order to get past. Once I placed the key-frames, I entered the level blueprint and synced the matinee reference to the play function.

I additionally wanted to try this effect using C++, I created a reusable class (adapted from a tutorial) in which I set up an animation curve for 5 seconds. I then set up functions which would either move the spike up by a certain amount when overlaying box collisions trigger the animation or rotate a door by a certain degree from a certain angle when collision occurs. I then made a blueprint class from this initial Actor class which inherited the properties needed to change mesh and transform and I was able to initialise a mesh object which can be added to the scene with the animation attached.

Overall I enjoyed the project and would like to work further to develop it into a publishable product. As a student only taking one module of this masters course I did find it slightly difficult to access module

`

content/info. I also found it hard to contribute to other areas of the submission outside modelling due to this. However, I have learnt a lot of skills which I will continue to develop after the completion of this module.

MSc Team

Alejandro Garzon

Main contributions:

- Pick up, drop and throw objects controls

This feature was the base for picking up/dropping/throwing any object that was simulating physics.

It was mainly based on doing a LineTraceByChannel from the player camera position and adding the proper logic when pressing/releasing mouse buttons.

It was also the start for the Pick up to Inventory feature.

How to throw rays and evaluate the hit result has been the main outcome from this feature.

- Blocking door trap

This trap was intented to block a door by falling rocks from the ceiling.

It was approached by adding a box trigger that when the player overlapped it, an event was disaptched to spawn rocks simulating physics.

The rock spawners were placed in the environment. They were coded with some editable properties for the artits so they could change the spawning cadence, number of rocks to spawn and spawning start delay.

A considerable amount of time was needed for tweeking the spawners in order to achieve a good result.

However, taking into account that the result wasn't determinitisc and sometimes not very ideal, it suggests the a different approach might be needed that doesn't rely on spawning rocks.

For example, moving a group of static meshes(without physics) could have been much easier to implement and maybe with a better good looking result.

All in all, how to spawn dynamic actors and using the event dispatcher has been the main outcome from this feature.

- Skulls puzzle

This puzzle consisted on collecting skulls from the environment with the purpose to place them in a balance. Each time an skull was placed,a door was opened slightly.
The puzzle was set up editable so artist could add the number of skulls needed. The skulls were using the inventory item template so they could be kept in the inventory.
Whenever an skull was colliding with a box placed in the balance, the actual skull was destroyed and an invisible static one that was already placed on the balance was made visible.

`

Reviewed features:

- Eye AI: Simple behaviour tree and eye patrolling in the environment using waypoints.
- Pick up to inventory + UI: Usefull so I could applied the item template to the skulls in order to collect them.
- 9 tile puzzle & Altar puzzle: Good to see how this puzzle and Altar puzzle was using cameras that probably could be applied to the skulls puzzle too.
- Day night/switch puzzle : Usefull how it work as I needed to place some skulls only during the day.

What didn't work and needed to improve:

- Lack of main features spec definition from a design point of view:

It took around two weeks to start coding something game specific, because the spec weren't clear at all, not defined puzzle, etc.

In part, programmers fault expecting artist to work on the design of the game too. Probably programmers should have got more input

on the design of the game. Also, as a team, we should have done an accesible document about the different puzzle and mechanics of the game.

- Scrum board

I personally set up a board in github so we could add all the tickets (stories and bugs). The main goal was to track progress and have visibility about the state of the project. Programmers start using it but not the artist. It was hard to check how the art was going for example.

Clearly, a communication problem on emphasize the usefulness of the scrum board.

- Work integration process

At the end of the project, many features needed to come and merge into the map. The map was a binary file that didn't merge well and always was creating conflicts. We end up many times redoing some work because of this issue. At the end of the project, having 3 programmers was more a trouble than a solution, where some programmers were idle and waiting for others to finish in order to avoid conflicts. Definitely, something to improve and take into account in the future.

What went well:

- Feature based branches

We split all the features between programmers, created different branches for them so we could work separately. We focus first on making it work, later merge and start integrating the feature into the map (quite problematic at the end), fix bugs and polish the feature.

- Different communication channels (Whatsapp group, Skype programming meeting)

Definitely they help with any unclear design issue or programming problem

Team member recommendation: Pablo Armando Rodriguez Codes

`

Reason: He dealt with different puzzles that added nice camera transitions plus the AI of the game. Also, he gave nice review comments on some of the pull requests. Finally, he took the responsibility the final days to make sure that all the features were coming and merging properly.

Full name: Pablo Armando Rodríguez Codes

Role: AI/gameplay programmer

AI

As the AI programmer in the team, one of the main tasks has been the development of the AI system of the game. Because of the limited time of the project, it was difficult for artists to create humanoid-type models for the enemies in the game. Therefore, we settled on one only enemy that would have the shape of an eye, which would chase the player to try to kill them.

The AI of the eye uses a behaviour tree that controls two basic states of the AI. The lowest priority state is dedicated to the patrolling behaviour. In this state, the eye will patrol cyclically through a set of waypoints.

The highest priority state controls the chasing behaviour, and could be defined as the following:

The eye will chase the player whenever it has vision of them.
The eye will see the player if they are at a certain distance from the eye and inside a certain angle range in front of the eye.
The eye will see the player if they are at a certain distance (smaller than the previous one), regardless of what the relative position is (even if they are behind the eye, if close enough).
Tests are performed with a simple distance and angle check.

The eye will not see the player if there are obstacles that block vision of them.
This test is performed with a line trace over the Visibility channel that goes from the eye to the player.

When the eye loses vision (described in the previous terms) of the player, they will go to their last known location. Once there, the eye will wait for a certain amount of time, and if they haven't found the player, the patrolling behaviour will be resumed. If at any point, the eye regains vision of the player, it will start chasing them directly.

Even if the behaviour tree is rather simplistic, it has served to gain familiarity with several features of Unreal's AI systems, such as the actual behaviour trees, including blackboard assets, tasks, decorators and services. However, a lot of work has been devoted to the navigation system (NavMesh), and tweaking the different settings so that the navigation is performed correctly through the whole pyramid, being narrow corridors and specially stairs (due to their high slope) the bigger challenge.

Gameplay

However time-consuming and challenging the development of the AI has been, a big amount of work has also been done in general gameplay programming, specifically related to puzzles in the game. The

`

puzzles I have developed entirely have been the 9-tile puzzle, the day/night switch and the main altar puzzle (heart & feather).

All of these have required a lot of experimentation with collision settings, camera animations, object management, and even some level streaming (used for day and night scenarios).

Team

One of the most challenging parts of the project has been the communication between the different parts of the team, especially between artists and programmers. Here it would have been nice to have had a full introduction to the git workflow in the Introduction to Programming course, which would have solved a lot of these difficulties. However, some documents (like the Readme.md of the project) have been created to try mitigate this problem by giving everyone a step-by-step description on how to clone the project and make changes to it.

The workflow has been using branches for all changes and pull request reviews from at least one other programmer. However, another big difficulty has been the resolution of conflicts between branches due to the binary files in Unreal, but with a little practice (and the ability to copy-paste most of the changes from asset to asset) this problem has been reduced over time.

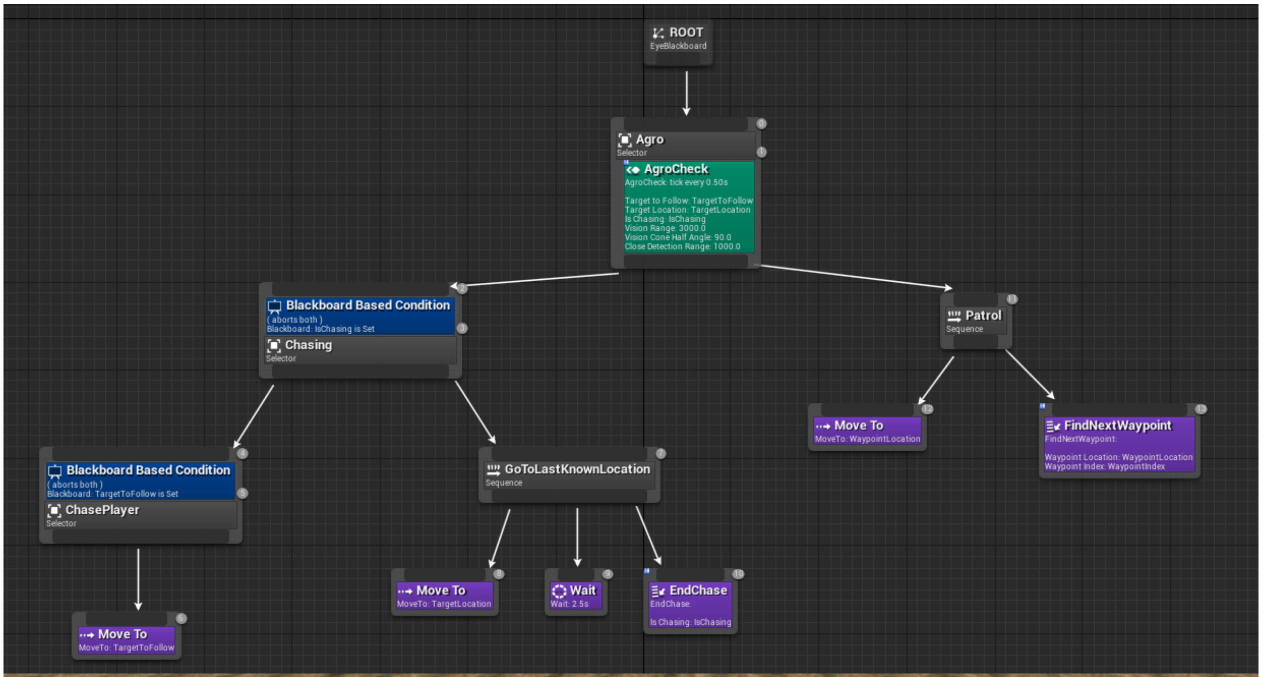
Future

Part of the group has already declared their intention of keeping working on the project after the deadline, which is really exciting, and would allow for building and even better game that would look great in all of our portfolios.

Part of the changes that I see I would make in the future stages of the game is doing some research on Unreal's Environment Query System and how it could be introduced to the game's AI system. The EQS allows for complex tests on different variables on the environment, to create very complex behaviours.

However, probably the most immediate change would be porting all of the work done to C++ code, since everything was done in Blueprint due to the limited time.

`



AgroCheck: tick every 0.50s

Target to Follow: TargetToFollow
Target Location: TargetLocation
Is Chasing: IsChasing
Vision Range: 3000.0
Vision Cone Half Angle: 90.0
Close Detection Range: 1000.0

Blackboard: IsChasing is Set

MoveTo: WaypointLocation

FindNextWaypoint:

Waypoint Location: WaypointLocation
Waypoint Index: WaypointIndex

Blackboard: TargetToFollow is Set

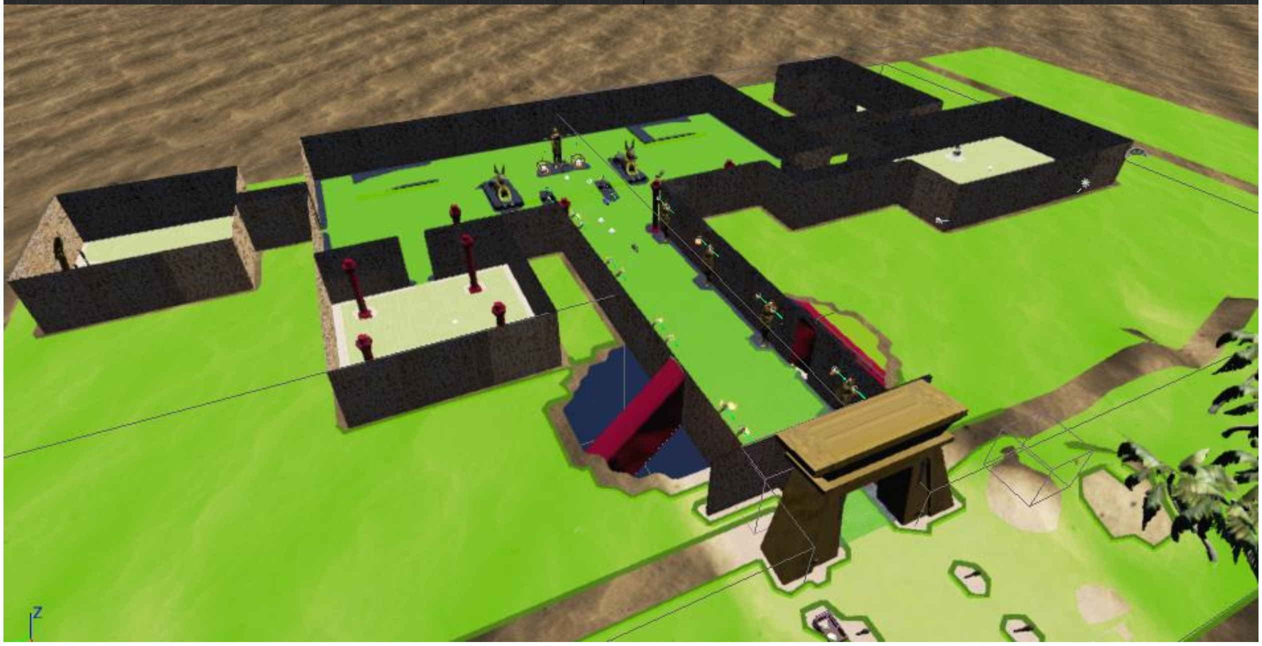
MoveTo: TargetToFollow

MoveTo: TargetLocation

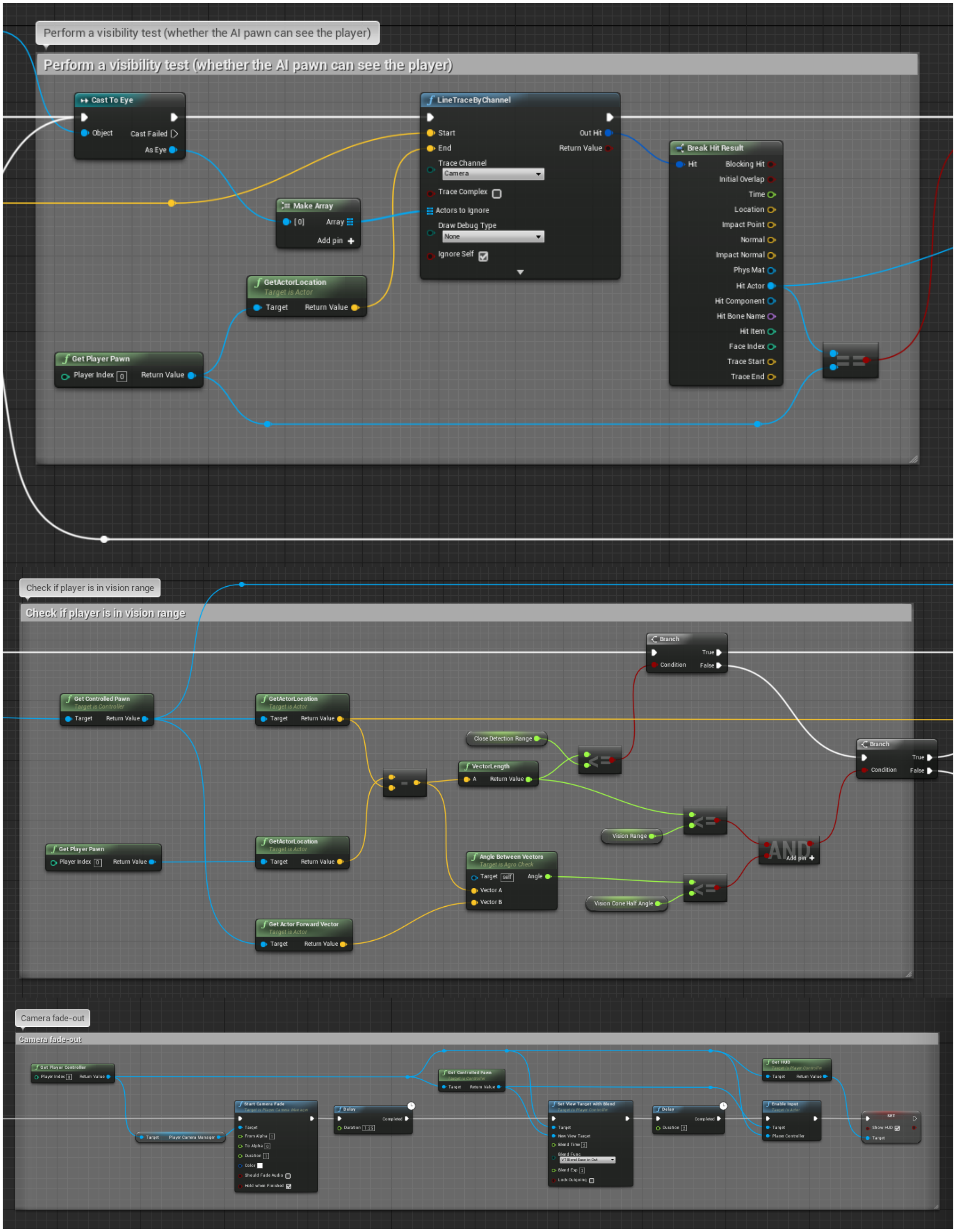
Wait: 2.5s

EndChase:

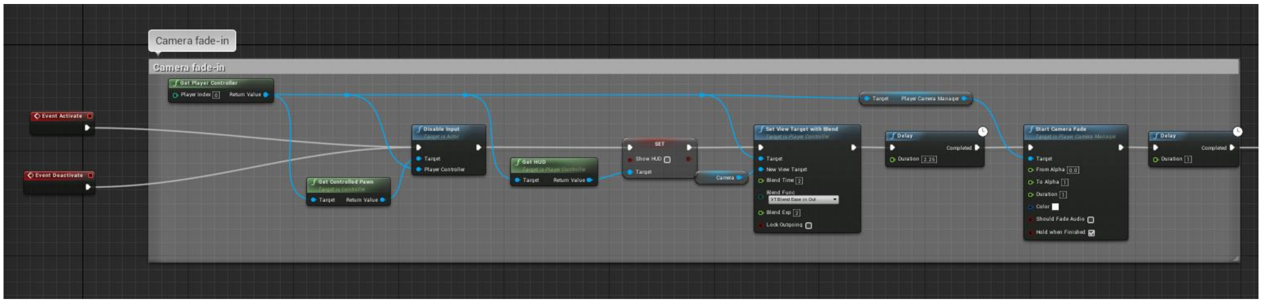
Is Chasing: IsChasing

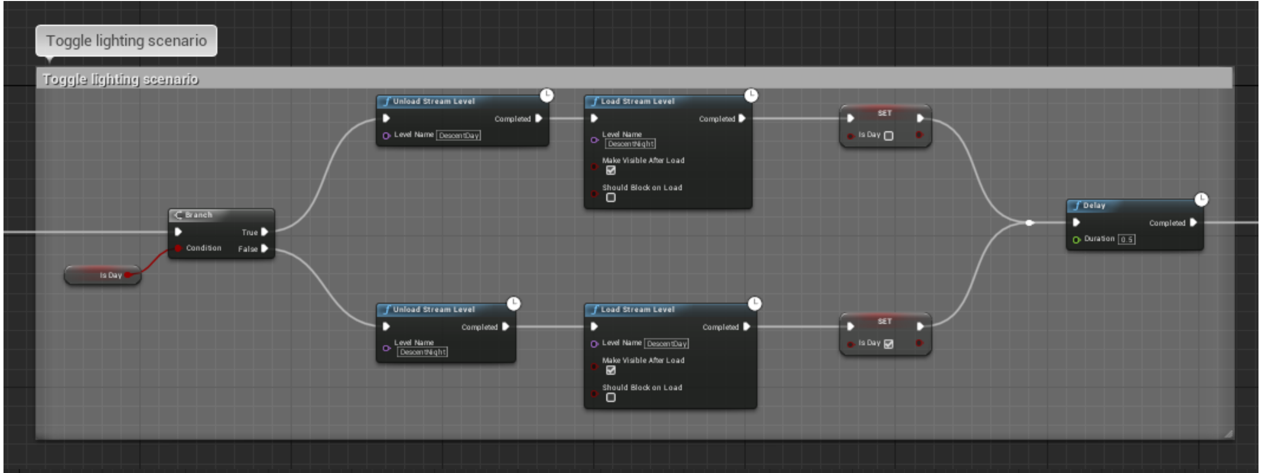


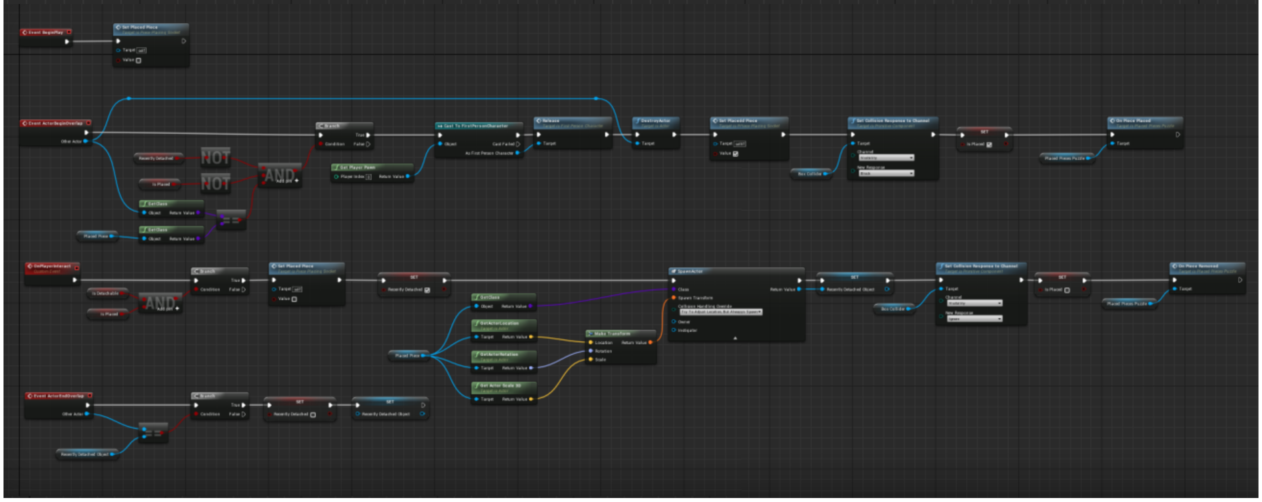
`


Perform a visibility test (whether the AI pawn can see the player)
Perform a visibility test (whether the AI pawn can see the player)
Cast To Eye
Object
Cast Failed
As Eye
LineTraceByChannel
Start
End
Out Hit
Return Value
Trace Channel
Camera
Trace Complex □
Actors to Ignore
Draw Debug Type
None
Ignore Self ☑
Make Array
[0]
Array
Add pin +
Break Hit Result
Hit
Blocking Hit
Initial Overlap
Time
Location
Impact Point
Normal
Impact Normal
Phys Mat
Hit Actor
Hit Component
Hit Bone Name
Hit Item
Face Index
Trace Start
Trace End
GetActorLocation
Target is Actor
Target
Return Value
Get Player Pawn
Player Index 0
Return Value
==
Check if player is in vision range
Check if player is in vision range
Get Controlled Pawn
Target is Controller
Target
Return Value
GetActorLocation
Target is Actor
Target
Return Value
Branch
Condition
True
False
Close Detection Range
<=
-
VectorLength
A
Return Value
Branch
Condition
True
False
<=
Vision Range
AND
Add pin +
Get Player Pawn
Player Index 0
Return Value
GetActorLocation
Target is Actor
Target
Return Value
Angle Between Vectors
Target is Agro Check
Target self
Angle
Vector A
Vector B
<=
Vision Cone Half Angle
Get Actor Forward Vector
Target is Actor
Target
Return Value
Camera fade-out
Camera fade-out
Get Player Controller
Player Index 0
Return Value
Get Controlled Pawn
Target is Controller
Target
Return Value
Get HUD
Target is Player Controller
Target
Return Value
Target
Player Camera Manager
Start Camera Fade
Target is Player Camera Manager
Target
From Alpha 1
To Alpha 0
Duration 1
Color
Should Fade Audio □
Hold when Finished ☑
Delay
Completed
Duration 1.25
Set View Target with Blend
Target is Player Controller
Target
New View Target
Blend Time 2
Blend Func
VTBlend Ease in Out
Blend Exp 2
Lock Outgoing □
Delay
Completed
Duration 2
Enable Input
Target is Actor
Target
Player Controller
SET
Show HUD ☑
Target

`


Camera fade-in
Camera fade-in
Get Player Controller
Player Index 0
Return Value
Event Activate
Event Deactivate
Get Controlled Pawn
Target is Controller
Target
Return Value
Disable Input
Target is Actor
Target
Player Controller
Get HUD
Target is Player Controller
Target
Return Value
SET
Show HUD
Target
Camera
Target
Player Camera Manager
Set View Target with Blend
Target is Player Controller
Target
New View Target
Blend Time 2
Blend Func
VTBlend Ease in Out
Blend Exp 2
Lock Outgoing
Delay
Duration 2.25
Completed
Start Camera Fade
Target is Player Camera Manager
Target
From Alpha 0.0
To Alpha 1
Duration 1
Color
Should Fade Audio
Hold when Finished
Delay
Duration 1
Completed


Toggle lighting scenario
Toggle lighting scenario
Branch
Condition
True
False
Is Day
Unload Stream Level
Level Name DescentDay
Completed
Load Stream Level
Level Name
DescentNight
Make Visible After Load
Should Block on Load
Completed
SET
Is Day
Delay
Duration 0.5
Completed
Unload Stream Level
Level Name
DescentNight
Completed
Load Stream Level
Level Name DescentDay
Make Visible After Load
Should Block on Load
Completed
SET
Is Day


Event BeginPlay
Set Placed Piece
Event ActorBeginOverlap
Other Actor
NOT
NOT
AND
Branch
Condition
True
False
Cast To FirstPersonCharacter
Get Player Pawn
Release
DestroyActor
Set Placed Piece
Set Collision Response to Channel
SET
Is Placed
On Piece Placed
OnPlayerInteract
Branch
AND
Set Placed Piece
SET
SpawnActor
Make Transform
SET
Set Collision Response to Channel
SET
On Piece Removed
Event ActorEndOverlap
Other Actor
Branch
SET
SET

`