Final Report

Semester 2 Project

The PROgrammers

RUNNING FOR THE WILD

Léo VERNIQUET

Manon LOVELAND

Constance CARLE

Julien CLAUSS

27-05-2020

Contents

1. Introduction

Our aim for our S2 project was to create a game that consisted in escaping a maze. Said like that, it sounds like an easy task to accomplish. However, it was much harder than what it seemed. This "escape the maze" game is a game which mixes both a logical side and an adventurous side whilst being a multiplayer game allowing a total of 4 players to play simultaneously in the maze.

We are the PROgrammers and are very proud to present to you our game, Running For the Wild, which is fully functional and allows players to escape the maze and solve different puzzles and enigmas in order to escape. This game was a complicated task that we had set ourselves to achieve and we have managed to finish designing, coding and creating this game from beginning to end, and are proud to call it our very first game.

Working on this project allowed us to discover both the good and the bad parts when creating a game from scratch as a group. It was a very good first experience for us and showed us how to work as a group in order to create a fun and exciting game to play with your friends, as well as allow us to discover the world of game design, creating games...

2. The Game

2.1 Detailed explanation of the game

Our game is quite simple to understand but involves a certain level of skill to play. The objective of our game is to escape the maze, by completing different levels as we go on, a total of 3 different levels. Each level will have a task, for instance small logic puzzles that the player will have to accomplish in order to escape the maze. And each level will ask the player to think differently and use a different logical perspective to win and escape.

The first two levels are equipped with an enigma the players will have to solve. One will be an individual enigma that each player will have to solve in order to escape, while the other will seek for coordination and help from the other players to escape. These two levels show that sometimes to win you can do it on your own, but sometimes you need the help from others, and in this case from the other players.

However, the last level is a little different. There will not be an enigma to solve. The goal of this level is to be the last one standing and the only one to win and escape the maze and win the game.

The game will be shown from a portrait perspective in 2D. However, the entire screen

will be blacked out, all except a small perimeter around the player, which allows him to discover parts of the maze only when he is near them. Indeed, this bubbled sized light emission will follow the player as he moves around the maze, and reveal the various ways he or she can take to find the exit. He will have to jump and find a way out of the treacherous maze, which is generated with dead ends, obstacles... The player is able to play against other players, attack them in order to steal items... but also avoid them in order to not get attacked.

The player will be able to move using the control pad on a computer, and jump onto different platforms of the maze using the space-bar. In addition to these controls, the player will be able to attack the other players within the game, to steal their possessions for instance. However, the aim is also for the player to avoid the other players in order to not get attacked.

The player will take the form of a young zombie. The player dies and wakes up as a zombie in a maze and thus has to find the exit of the maze.

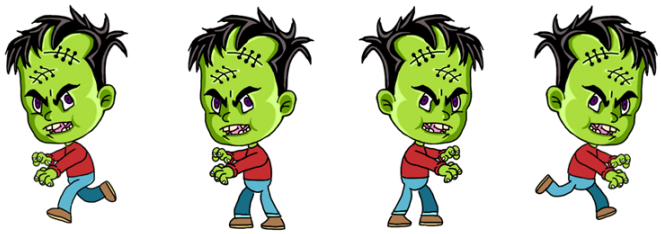


Figure 2.1: Choosing the level Menu

Our game was inspired by our love for escape games where the players have 1 hour to find a solution to escape the room they are trapped in. In addition, we were also inspired by the game Super Meat Boy, a platform game which has multiple aspects such as gravity and

physics which could be very interesting for our game. As well as this game, there is also the famous game Spy Mouse, where the mouse has to get the cheese and avoid the cats in order to find the exit.

Figure 2.2: Inspiration for our game

To code our game, we used a Unity software, coding in the language C sharp using Rider as our coding platform.

Before exiting the maze, the player must complete a fun logical puzzle. These were created by us and will force the player to travel around the maze to find the various pieces needed to complete the puzzle.

The complexity of the game resides in the fact that the player wants to be the first to exit the game, therefore before the other players. Lastly, this game will involve a certain amount of skills to acquire throughout the game.

The player, once on the Main Menu screen will be able to click the play button and create a name for his player and name the room/maze he is going to play in. On the side, he

will also be able to see if there are any other players connecting to his room/maze. In addition, on the main menu, the player can click on the bottom left hand-side corner where the name of the game is and that will directly bring him to our website. Following this, the player will be able to select which level he wants to play, between level 1, 2 or 3.

Figure 2.3: Image of the Main Menu of our game

Figure 2.4: Image of the Create Room Menu of our game

Figure 2.5: Choosing the level Menu

2.1.1 Level 1

In the next three subsections, we are going to explain in detail each level and how to play this level and it's complexity compared to the other levels.

The first level set in the maze is what we could call an individual level. This means that each player will have the task to accomplish the level and solve the enigma and lastly, escape the maze. In each level, there can be a total of 4 players playing at the same time. Each player will start in a corner of the maze and move around the maze to try to find the exit. However, before exiting the maze, the players must accomplish a simple logical puzzle.

Figure 2.6: Bucket and Tap Level1

When cruising around the maze, the players will probably fall upon a tap and a bucket, both hidden somewhere within the maze. The goal is for them to fetch a bucket at the area the bucket is hidden and try to find the tap in order to fill the bucket up. Once it is filled up, the player will have to find the door, all whilst still holding his bucket. Once he/she reaches the exit door, the player will be confronted with a big reservoir. Hence, he/she will have to fill up the reservoir with the bucket full of water. Once this is accomplished, the doors will open and allow the player to walk through them and bring him back to the main menu.

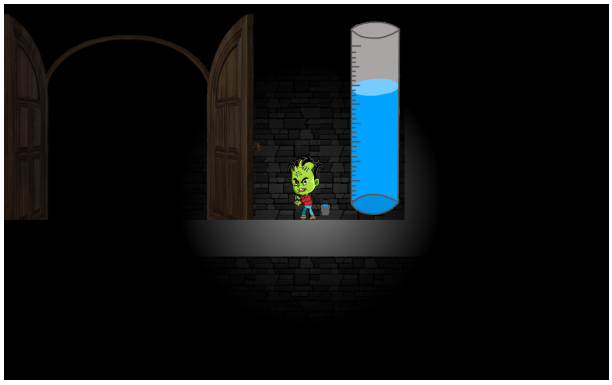


Figure 2.7: Container full and doors open Level1

2.1.2 Level 2

In the second level, another skill is asked of the player. And this is teamwork and coordination. This level represents in a way what we had to go through in order to work as a team to create this game.

The goal of this level, is yet again to exit the maze. However, this time the players will have to work together to win and not try on their own. If they do, they will quickly find out that they cannot escape without the help of the other players.

The players will yet again each start in a corner of the maze. They will have to find two levers hidden within the maze. One for each pair of players for example. The two levers will have to be activated at the same time, which means both players will have to work together to do so. Once both levers are activated, the doors to exit the maze will open. However, the players will only have a short amount of time to exit through the doors before they shut again. If so, they will have to start the whole process again, of finding the levers and quickly rushing to the door.

Figure 2.8: Level2 game

2.1.3 Level 3

The third and last level of Running for the Wild is a little different from the two previous levels. In this level, the players will have to exit the maze, just like the other levels, however this time they will have to fight the other players.

Each player will start next to the exit of the maze, however they will have to go through a long maze to actually be able to access the exit of the maze. Once they find themselves confronted with the door to exit the maze, the players will quickly realise that they cannot exit, that the door will not open.

In order to actually be able to exit the maze and finish our game Running for the Wild, the players will have to fight the other players and end up as the last one standing. To do so, the winner will have to kill the other players. Once this is accomplished, the winner will be able to exit the maze as the doors will open and he will become the ultimate winner of Running for the Wild.

To kill the other players, he/she will have to push them and make them fall to the ground. Once the player falls to the ground, the player will explode and break into a thousand pieces.

3. The different parts of the game

Throughout this entire project, we had assigned a numerous amount of tasks that we would have to accomplish before the end of the project. The following table shows you what deadlines we had put for the project in accordance to each defence, as well as how we managed to finish each one. As you can see, we had said we were going to finish every task for this last defence, and we are proud to say that we have a functioning game which is fun to play, alone as well as with friends.

Different parts of the project (in percent)			
Tasks	Defence 1	Defence 2	Last Defence
Character Movements	30	70-75	100
Multiplayer	30	60	100
Level generation	30	60	100
Game mechanics	30	60	100
Music and Sounds	20	50	100
Level design	30	60-70	100
LateX reports	30	60	100
Creation of enigmas	30	60	100
Animation and Appearance	20	50	100
Website	30	60	100

4. Evolution

4.1 Task Division for the Last Defence

Task Divison for the Final Defence				
Tasks	Constance	Léo	Julien	Manon
Multiplayer		X		
Character Movements		X		X
Level generation	X			
Game mechanics	X	X		X
Music and Sounds	X			
Level design				X
Objects				X
Character frames				X
LateX reports				X
Creation of enigmas	X	X	X	X
Animation and Appearance		X		X
Website	X			
Trailer	X			

4.2 Our aim for the last Defence

For this last and final Defence, our aim was to finish all the different tasks we had set ourselves at the beginning of this project. This next table shows how we had decided to distribute the various tasks that were left for this project. As you can see with the previous table, we had new tasks that came to the surface. In addition, we can also see that, compared to the table above, one person didn't contribute as much as the others in this project.

Aim for last Defence				
Tasks	Constance	Léo	Julien	Manon
Character Movements	done	done	done	done
Level generation	X	X	X	X
Game mechanics		X	X	
Music and Sounds	done	done	done	done
Level design	X			X
Character frames	done	done	done	done
LateX reports				X
Creation of enigmas	X	X	X	X
Animation and Appearance		X		X
Website	X			

5. Personal Reports

5.1 Léo Verniquet

For this last defence, I am going to explain everything I had to do for our project : Running For The Wild. My main missions were the following :
-Implement access to network and multiplayer with photon
-Implement character movements

But I also spent a lot of time on secondary missions such as implementing some animations, thinking about enigmas and looking for music in order to help my teammates in their missions.

Of course, I was not qualified in the first place to carry out these missions so this project allowed me to learn how to look for the information I was looking for: Not only did I watch tutorials and read articles on the Internet, but I also asked for help from other students that helped me find solutions to tricky situations I was getting in by my incompetence. I am now way more comfortable with Unity than before, I still have a lot to learn, but let's get back to the missions.

I decided to implement the multiplayer first as it was recommended to me by other students. I used photon but as it was hard for me, having no photon background, I watched hours and hours of tutorials in order to end up with a multiplayer that fits our project. It is based on the fact that the first player in the room is the host of this room, so we don't have to rent a server in a remote part of the world to host our network.

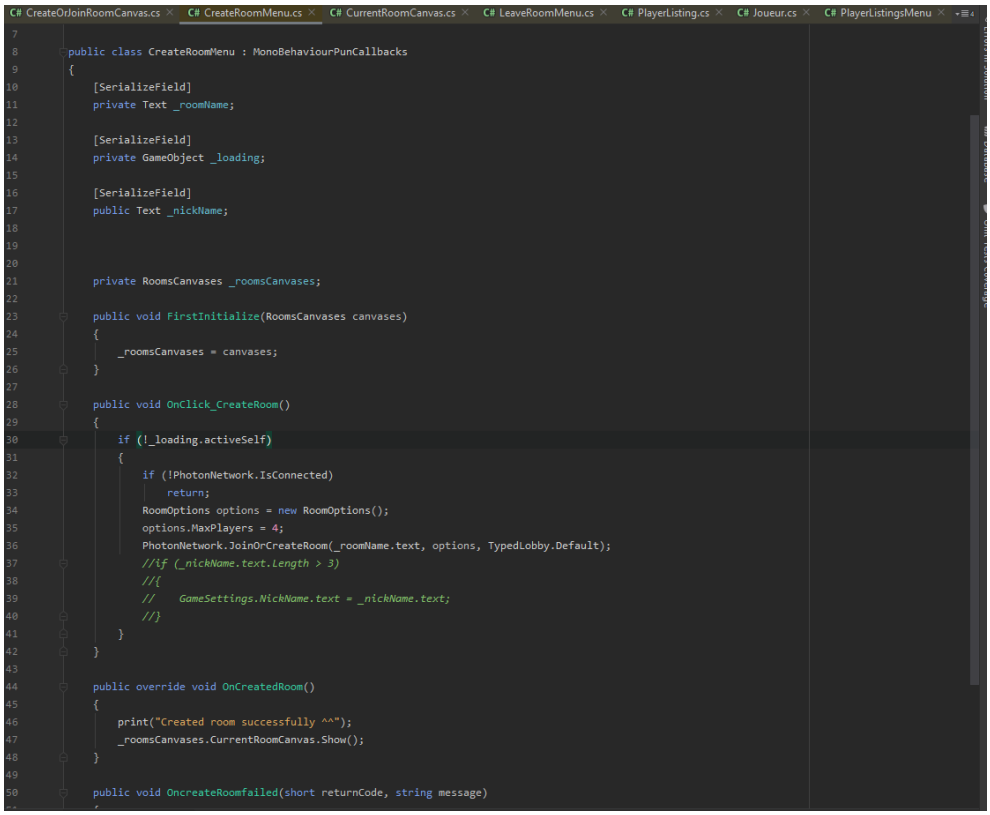


Figure 5.1: Photo Photon Script

I started by creating a script that allows the player to connect to the Internet via a main menu that I coded but Manon decorated it. Then, the player is able to either connect to another player by joining their room or to create their own room so that other players can join. I added a loading GameObject because I noticed that doing as the tutorial said was not the best option : in fact , if the player tries to create a room or connect while trying to connect to the internet via the photon script, it can create problems and the player may have

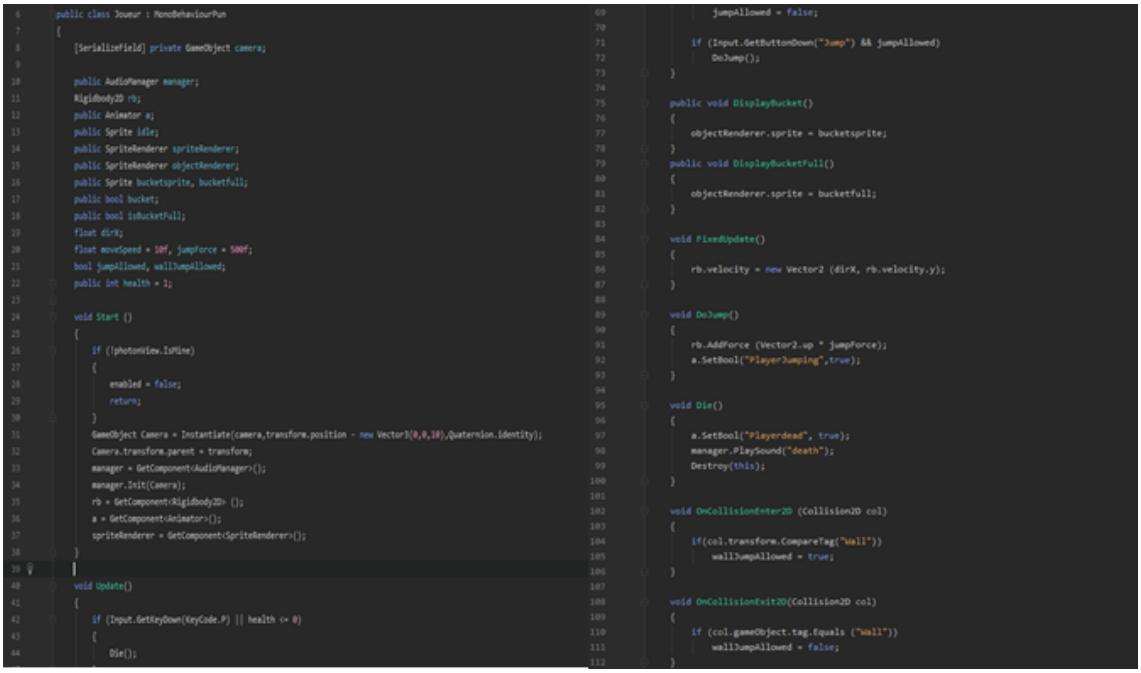
to restart the game. That is why, when trying to connect, the player can notice a text that says “loading” on the top right of the screen, which allows the player to wait a bit before trying to push buttons.

I implemented a feature that allows players to enter a name so that they can be different from other players in the room. Moreover, if a player does not set their name, it will be set to a default name with the from “Player[random number]” so that it can still be different from other players.

Figure 5.2: Photo Loading page and name

With the room system done, I started working on player’s movement because we needed that in order to play our game. Basic movements were very simple to implement as there were multiple tutorials for this part, where I just had to implement walking left and right, but also jumping.

But then, I started to tackle some unique features: wall jumping and wall climbing. It took me a lot of time and a lot of tries to find a good tutorial on wall jumping. In fact, I spent a lot of time trying to implement this wall jump script and I had to rewrite my basic movements too. I finally implemented that to ease the movements of the player in the maze, to have a more reactive game and more fun. It is important to notice that the gravity feels funny on purpose. Moreover, the player can propel itself with slopes which can be a lot of fun.



```
public class Joueur : MonoBehaviourPun
{
    [SerializeField] private GameObject camera;

    public AudioManager manager;
    Rigidbody2D rb;
    public Animator a;
    public Sprite idle;
    public SpriteRenderer spriteRenderer;
    public SpriteRenderer objectRenderer;
    public Sprite bucketsprite, bucketfull;
    public bool bucket;
    public bool isBucketFull;
    float dirX;
    float moveSpeed = 10f, jumpForce = 500f;
    bool jumpAllowed, wallJumpAllowed;
    public int health = 1;

    void Start ()
    {
        if (!photonView.IsMine)
        {
            enabled = false;
            return;
        }
        GameObject Camera = Instantiate(camera,transform.position - new Vector3(0,0,10),Quaternion.Identity);
        Camera.transform.parent = transform;
        manager = GetComponent<AudioManager>();
        manager.Init(Camera);
        rb = GetComponent<Rigidbody2D> ();
        a = GetComponent<Animator>();
        spriteRenderer = GetComponent<SpriteRenderer>();
    }

    void Update()
    {
        if (Input.GetKeyDown(KeyCode.P) || health <= 0)
        {
            Die();
            jumpAllowed = false;

        if (Input.GetButtonDown("Jump") && jumpAllowed)
            DoJump();
    }

    public void DisplayBucket()
    {
        objectRenderer.sprite = bucketsprite;
    }
    public void DisplayBucketFull()
    {
        objectRenderer.sprite = bucketfull;
    }

    void FixedUpdate()
    {
        rb.velocity = new Vector2 (dirX, rb.velocity.y);
    }

    void DoJump()
    {
        rb.AddForce (Vector2.up * jumpForce);
        a.SetBool("PlayerJumping",true);
    }

    void Die()
    {
        a.SetBool("Playerdead", true);
        manager.PlaySound("death");
        Destroy(this);
    }

    void OnCollisionEnter2D (Collision2D col)
    {
        if(col.transform.CompareTag("Wall"))
            wallJumpAllowed = true;
    }

    void OnCollisionExit2D(Collision2D col)
    {
        if (col.gameObject.tag.Equals ("Wall"))
            wallJumpAllowed = false;
    }
```

Figure 5.3: Script for the player

When this was implemented, I still had to finish the multiplayer scripts as the players must spawn in the labyrinth in order to play the game. This part might have been the hardest one as I needed the help of another student to explain my mistakes, as I could not find the solution on the Internet.

This was one of the hard parts of this last period of time as there were many camera problems, but also spawning problems. A friend explained me the PhotonView method that allows to spawn a player prefab by player and allows to limit the camera conflicts, but I finally decided to create the camera in the player script to avoid any camera conflict, so that each player creates his own camera.

I finally managed to implement the player spawns. After that hardship, I decided to implement the melee attack which allows the players to fight and kill each other.

In order to do that, when the attack button is pressed ('A' key), the player deals 1 damage in a zone in front of him. If any other player is in this zone, it will then take 1 damage, but every player is spawned with 1 health point, that means that a player is killed by one punch.

What I discovered with this project is the naming problem of the scripts, which is a very important part of the project so that my group does not get lost while looking for a part to modify, for example to implement sounds. I never worked on a project with so many scripts before, so I could not realize the importance of using folders and clear naming process.

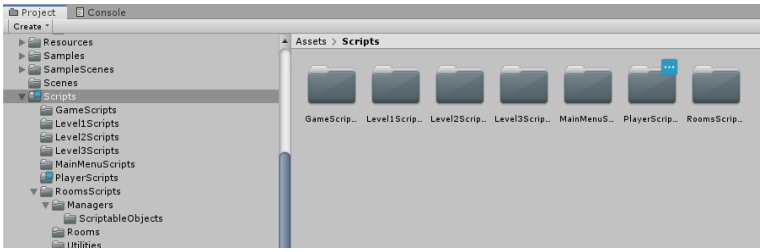


Figure 5.4: The various folders

At the end of the project, when modifying textures so that they can be affected by lighting, I had to fix our 3 levels a first time because of the change of size of the floor, wall and slope prefabs. Then I changed again a part in these prefabs and had to fix the levels again. That is why my group helped me fixing these levels, which was the most annoying part of this last time period.

For this project, I proposed the idea of the current main menu music which is the death music in TowerFall Ascension. As our project won't be commercialized, I decided not to ask for the rights to use this music in our game, but if I needed I could as I can contact the family of Alec Holowka who is the composer of this music. By finding this music, I could find a lot of music written by Alec Holowka which could fit in video games and that are very expressive. The music we are using lets the listener feel not only sadness but also a kind of suspense that allows one to hope an happy ending, such as leaving this maze.

Each one of us (the 3 who worked) found an enigma each; I found one which can be interesting: in the maze, there are two levers and the door. The door can only be opened if both the levers are activated, so the players have to activate the levers, but also be careful about the door, as one can just stand next to the door as the others activate the levers. This enigma can be interesting when the players know the labyrinth.

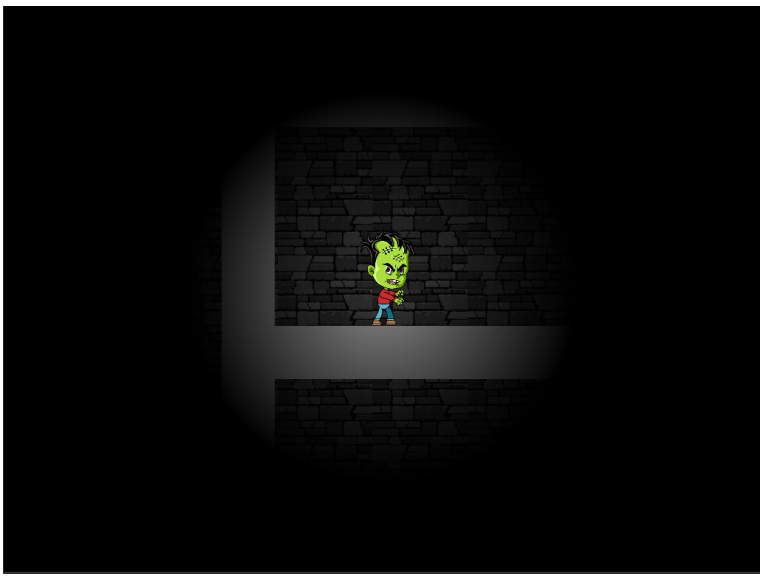


Figure 5.5: Picture of the player in the Maze

In fact, as the whole labyrinth is in the dark, the players usually don't know the labyrinth and have to play to discover it or even make a map of it.

To sum up my part of the work in this project I would say that I mostly worked on the skeleton of the game so that my group can add their part in order to end up with a complete game. Of course, I was not alone working on this skeleton as I was helped by my teammates when I asked for it.

5.2 Manon Loveland

For this last and final defence, we had a lot to finish in order for the game to be playable and completely finished and ready to play. We already had a lot of things that had been accomplished for the first and second defence, but numerous tasks were left for this final one. We had a character, the main menu, a website... but we were still missing the actual levels of the game.

Therefore I set off to draw the mazes we would need for the levels by hand. I started off by drawing big squares and rectangles to try and determine which shape we would want the maze to take and started filling them in with numerous lines and walls and slopes in order to create a maze. However, I quickly realised that the maze I had started drawing would be too big for one level and way too complicated for the players to accomplish. That said, also because the whole maze would be in the dark which would mean that the players can not only not see where the exit is, but also nothing further than a small bubble light that follows them around the maze.

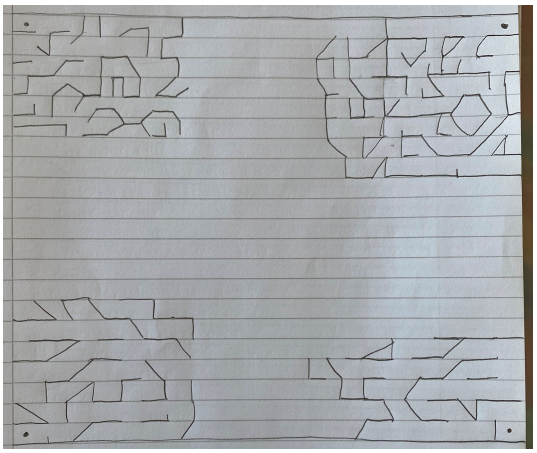


Figure 5.6: The first sketch for the levels

So once I decided to scratch those first sketches for the maze, I moved onto creating three smaller rectangles. Each rectangle represented one of the levels. I used colours in order to place where the various players would start in the maze, where the exit of the maze would be located and where the objects used for the enigmas in the levels would be placed.

In total we can have a maximum of 4 players playing simultaneously, which means I had to make sure the maze was doable and solvable from the four different perspectives and starting points. In addition, I also had to make sure each player could access the various objects placed within the maze and make sure all the players had an equal chance of finding them and finding the door. This meant that I could not place walls and floors and slopes just how I liked, but had to make sure that the player would have the same chance as player number 2, 3 and 4 to solve the maze.

Therefore, I started my drawing for the first maze by hand. I had a couple of difficulties at some points, for instance when I realised that the maze was going to be too tight for the players to run through, even if we shrunk the players, it would still have been complicated so I had to start over again. It took me a couple tries before I got the maze exactly perfect. It was solvable by each player, the objects would be easy for the players to access and it was an equal playing field for each player. This is what the first level looks like on paper before implementing within Unity:

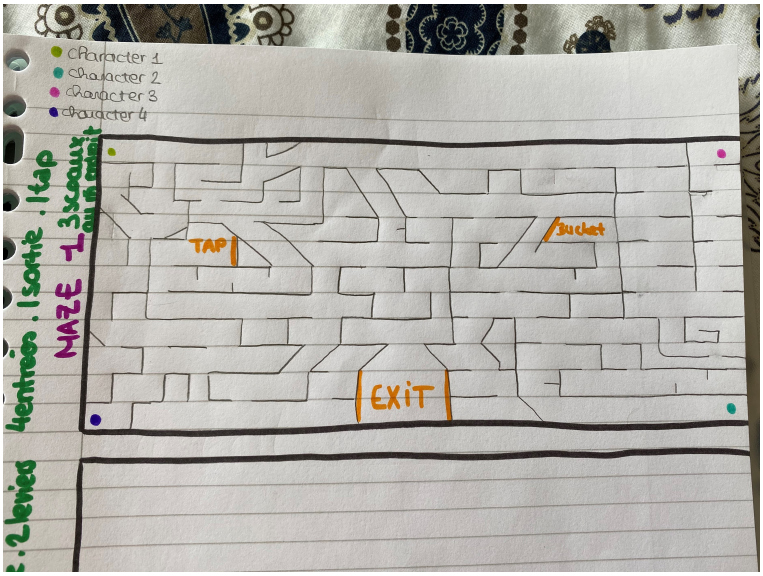


Figure 5.7: Level 1 final sketch

On the side of each level, I had to write what we needed in this particular level. For instance for level 1, we needed to place four players, an exit, a tap and a bucket generator for the players.

After finishing to draw the first maze, I started drawing the second maze. I faced the exact same problems with this second maze, even though I had already faced them once for the first maze, I still managed to make the same mistakes. Thus, I drew and redrew this second maze for level 2 until it was perfectly drawn. This meant I had to make sure the maze was at the same level of complexity for each of the players as they all start each in one of the corners of the maze. I had to put the right amount of dead ends, obstacles... to balance the maze out and make it equal for each player. This maze needed four places for the players to start, two levers for the players to switch in order for the door to open, and a door for the exit. This is what the final sketch for level 2 ended up looking like:

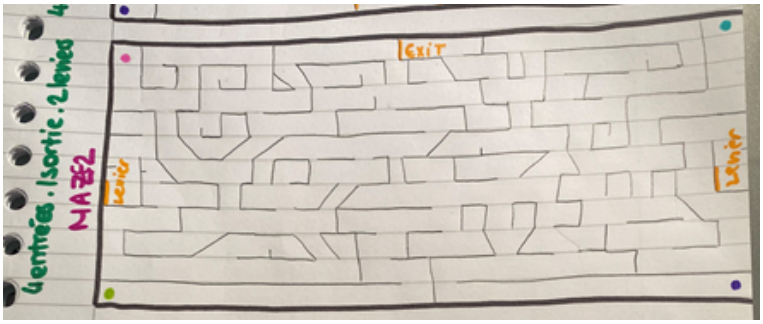


Figure 5.8: Level 2 final sketch

Finally I had to tackle the sketch for the third and last level of Running for the Wild. I wanted to make it a little different as this is the last maze. For the two previous ones, the players all started in a corner of the maze, all very far from each other. However, I wanted to change it up a little for this last level. Therefore, I decided to create a big square in the middle of the big rectangle where I was drawing the maze. This big square would be the exit of the maze, where the door would be. And just outside from this big square, on the other side of the wall, each player would start in one of the corners of this square.

This was one of the most complicated levels to draw because I had only drawn one entrance to the big square in the middle. However, the same rules still applied for this last level just like the two previous ones. The players should all have an equal chance of arriving at the door, the complexity should be the same for each player. After trial and error, I finally managed to create the perfect maze for the last level. In this last level, there was no need for any objects because there aren't any enigmas in level 3. The "enigma" is to kill every other player and be the last one standing. Once this is accomplished the door will open for the winner and the game Running for the Wild will be over. There can only be one ultimate winner. Therefore, I only needed to place a door and create a space big enough for the players to fight. This is what the sketch for the last level looked like once it was perfect:

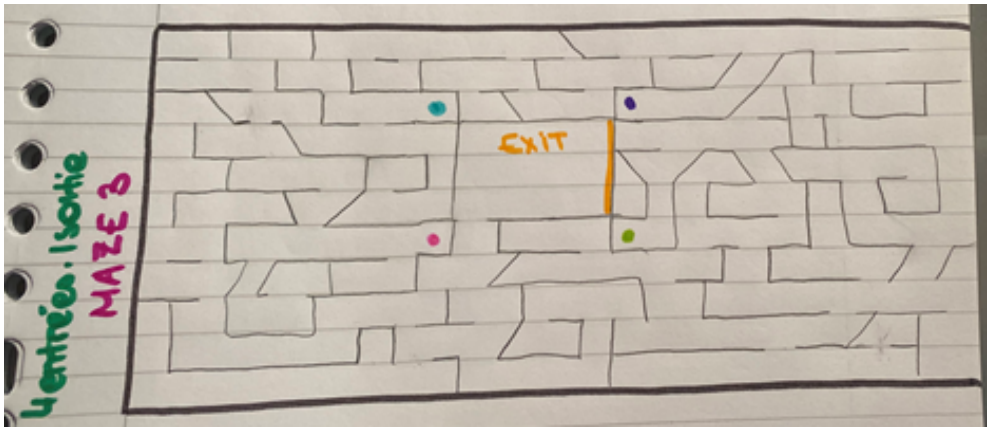


Figure 5.9: Level 3 final sketch

Once I had finished drawing the mazes by hand, I sent the pictures to Constance so she could implement them within Unity. I then moved onto putting the various objects in Unity. I had already drawn a tap and a bucket. However, I quickly realised that it would be complicated to implement the bucket because I had drawn it in 3D using Blender. Thus I decided to redraw the bucket, this time in 2D to be able to easily implement in Unity. To do so, I used Artstudio as my drawing program. I decided to do a very simple drawing. I drew the bucket as empty, so the players can carry it and then fill it up at the tap. Thus to show the difference, I also drew the same bucket full of water.

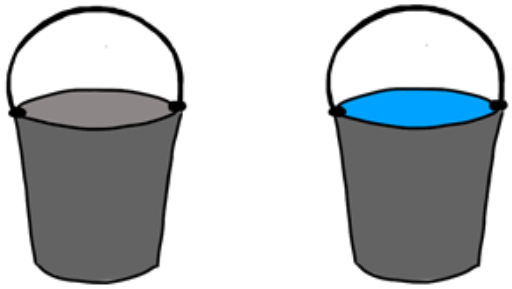


Figure 5.10: Pictures of the new bucket (empty and full)

In order for the enigma to work in the first level, there were a couple more things I needed to draw. As said previously, the first enigma consists of finding a bucket, then finding the tap and filling up the bucket and then finding the door. Next to this door, there is a reservoir which needs to be filled up with water for the doors to open. Therefore, using Artstudio, I drew a reservoir, a container which would fill up with water when the player walks in front of it. The container was drawn and coloured in grey and then blue to create the effect of water being inside the container.

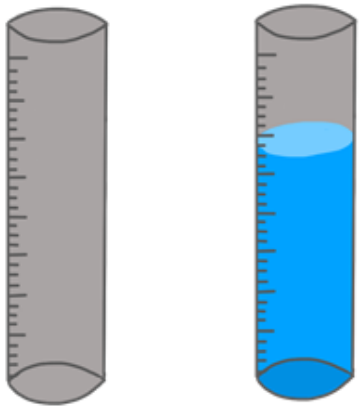


Figure 5.11: Pictures of the container (empty and full)

For the first enigma, all the objects were drawn, so I then moved onto drawing the object for the second enigma, hence level 2. For this enigma, we needed levers to be placed in the maze, a total of two. Both levers would have to be activated at the same time for the door of the maze to open and allow the players to exit the level. As usual, I used Artstudio to draw the levers, drawing an outline of the levers before moving onto colours. The lever had to be drawn in an up position to show that the lever was activated. Therefore there were two types of levers that had to be drawn. This is what they ended up looking like in the end:

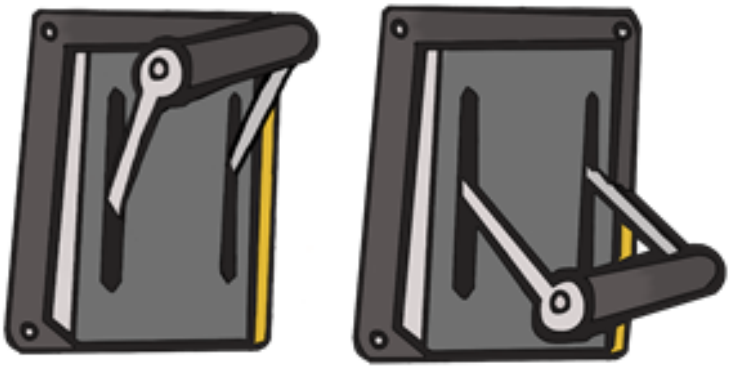


Figure 5.12: Pictures of the lever (up and down)

The last object I had to draw, which is maybe one of the most important objects, was the door which enables the players to exit the maze. This door is present in each level so it had to be perfect. I used ArtStudio to design the door, working with an outline found online. I then added a little detail to the door to go in the atmosphere of the game. The door had to be drawn as if it was closed and as if it was open. So I had to draw the frame of the doors open.

Figure 5.13: Pictures of the door (open and closed)

Once all the objects were drawn, I had to implement the objects within Unity. To do so, I had to write a specific script in C for each object that needed an interaction with the player. When placing the various objects in the maze, I decided to change their size and put them a little bigger than what they would look like in real life, which would allow the player to find them a little easier when walking around the maze. I started off by placing the tap and the bucket which are needed for the enigma in level 1. I placed the bucket in the maze, and made it as a bucket generator in some ways. The player finds the bucket and once he walks in front of it, he finds himself with a tiny bucket in his hands. Therefore, we are able to see the player run around in the maze with a bucket in his hands. To do so, in the script, I had to make sure that the player was not already holding a bucket in his hands. If so, he would not be able to pick up a bucket.

It was the same sort of idea for the tap. I had to make sure that the player was indeed holding a bucket in between his hands but also had to make sure that this bucket was not already full.If it was, the player would not be allowed to fill the bucket up even more.

Figure 5.14: Player with the bucket in his hands

```
private void OnTriggerEnter2D(Collider2D other)
{
    if (other.CompareTag("Player"))
    {
        var pl = other.GetComponent<Joueur>();
        if (pl.isBucketFull)
        {
            spriteRenderer.sprite = reservoirfull;
            open.Open();
        }
    }
}
```

```
private void OnTriggerEnter2D(Collider2D other)
{
    if (other.CompareTag("Player"))
    {
        var p = other.GetComponent<Joueur>();

        if (p.bucket == true && !p.isBucketFull)
        {
            p.isBucketFull = true;
            p.DisplayBucketFull();
            p.manager.PlaySound( key: "robinet");
        }
    }
}
```

```
Event function
private void OnTriggerEnter2D(Collider2D other)
{
    if (other.CompareTag("Player"))
    {
        var p = other.GetComponent<Joueur>();
        if (p.bucket == true)
        {
            return;
        }
        p.bucket = true;
        p.DisplayBucket();
    }
```

Figure 5.15: Script for the bucket,tap and container

The same had to be done for the reservoir. I had to make sure the container was not already full before filling it up with water. And make sure the player was actually holding a bucket full of water. Once the container is full, I had to create a sort of mechanism that would allow the doors to open for that player. Therefore, when the container is full with water, the doors of the maze will open up for that player to exit.

The same sort of manoeuvre was done for the other levels. In level 2, both levers have to be activated for the door to fling open for the players to exit the maze. Hence, to leave the players enough time to understand what they need to do... I put a timer of 5 minutes. This timer keeps the door open for a certain amount of time to allow the players to get to the door on time before it shuts. If the timer runs out, the door will shut and they will have to start all over again and activate the levers.

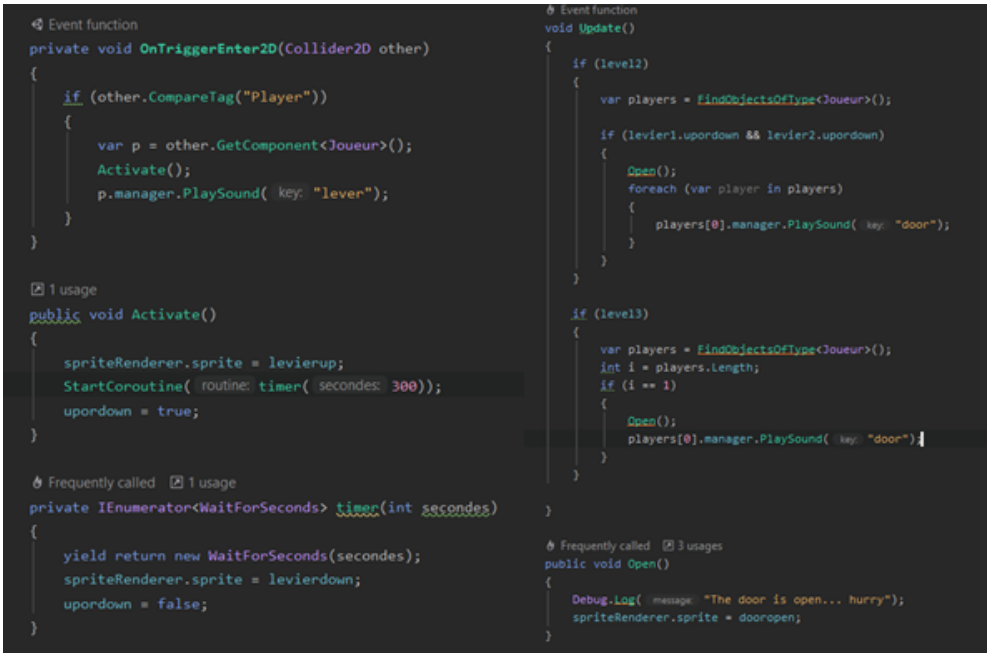


Figure 5.16: Script for the levers and doors

Finally for the last level, we just have to make sure that there is only one player left in the maze. Once the last player standing has killed all the other players, the doors will open and he will have won the game Running for the Wild. Hence, the player will have the honor of having won our great game.

Lastly, I also worked on the whole LateX report, as well as creating the video for the last defence of this project.

Working on this project has been really fun. I managed to learn a lot on the subject as well as really understand how games are created. It was extremely fun designing the player, the objects and the levels as those are the things that make the game personal. For instance, our little zombie character is unique, he is in our game and one of a kind. He belongs in Running for the Wild. It was also interesting drawing the levels as I had always wondered how difficult it would be to create a maze and this allowed me to finally have the answer to that question: it is indeed hard to make sure the maze is playable from everywhere! But I learned quite a lot as well from drawing and implementing the objects as I realised how players can interact with them, use them... Implementing the objects was new for me as it took a lot of trial and error before knowing for sure that it worked. Lastly, working on the LateX report as well as doing the editing for the video showed me that a project is not only creating a game, coding and drawing, but there is also a more formal part to it. Having to write the report helps to explain the game, as well as understand it, but mainly helps with the administrative part of the game. And working on the video editing helped me develop a new skill which I'm very pleased about.

Overall, working on this project in a group has been a very good experience. Even though we did not have the easiest project in mind, or some difficulties in our group, we are very proud of our game Running for the Wild.

5.3 Constance Carlé

For this third and final defence, I worked on various tasks on the game while keeping the website up to date. Overall, I was mainly in charge of implementing the sounds in the game. As I explained during the second defence, there are different sounds like the victory and defeat themes, the maze theme, the sound of jumps and all the different noises. These sounds were implemented in different ways. Another essential part of my work was to implement the three levels the players would play on. Each level was created one by one and piece by piece.

For the creation of the level, Manon sent me the drawing that I was supposed to implement in the game. So, I created different objects: floors, walls and slopes. Each has different properties, for example, walls allow us to wall jump. I duplicated those objects in order to have enough pieces to create the different mazes. Then, I assembled piece by piece each maze. I needed to assemble them correctly in order to not create unwanted micro edges that would stop the player on a supposed line.

Figure 5.17: Example of an edge in the game

Once the levels were finished Manon implemented different elements like the player's character, the different doors or the enigmas. Meanwhile, Leo put some texture to the maze. We first thought of a texture made of rocks but this didn't fit very well the floors and walls of the maze, since these are long rectangles, it was stretching the texture. This is why we opted for a sober grey texture. This texture fits the maze very well.

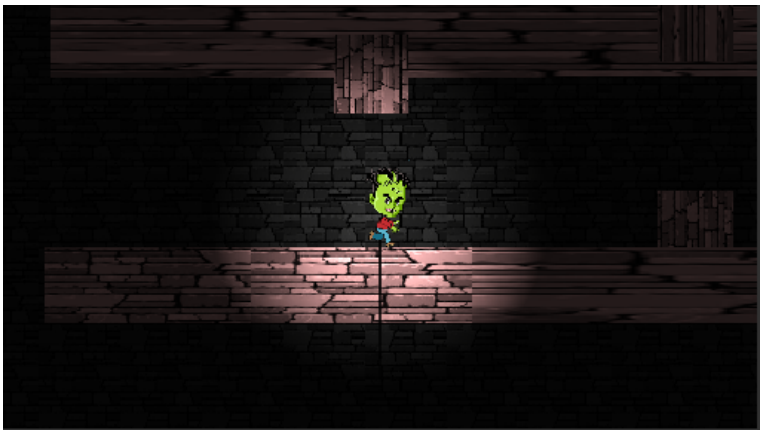


Figure 5.18: Old textures in the maze

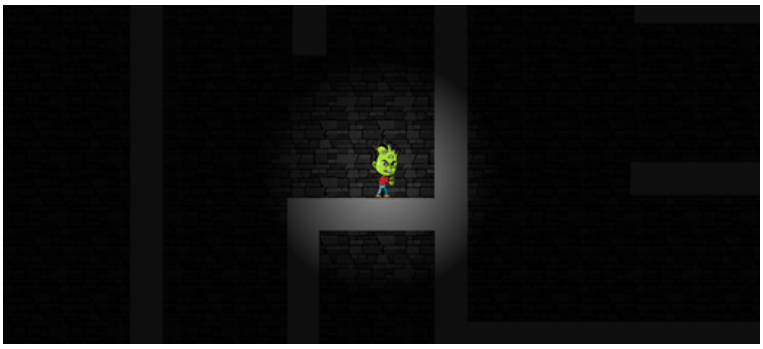


Figure 5.19: New textures in the maze

Unfortunately, the creation of the multiplayer aspect has made unwanted changes in the levels and it was up to us to fix the problems. In fact, some platforms were moved and needed to be rearranged. This is why we decided that Manon would fix the first maze, Léo would take care of the second maze and I would correct the third level.

My next task was to implement the different sounds. For this, I needed to take into account the various goals of these sounds. Indeed, the background music that plays while we are in a maze and that need to loop will not be implemented the same way as a door opening once. I first created a directory called "Resources" in which I transferred the music to Unity. I then started to implement the maze theme via C script of the different levels. Then I made the music loop. I did the same thing for the menu music.

Figure 5.20: Example of script used for the sound

For the rest of the sounds. I created a dictionary that contains every name of every sound. Then, I implemented every sound into the right function: the "death" sound into the "Die" function, etc. For some sounds I encountered some difficulties like the "click" sound (that happens when we click on an icon on the menu screen). This sound was particularly annoying to implement because clicking on an icon would play the sound and send us to another page of the menu (the page where we create a room for example) but the fact that we move to another page would cut the "click" sound. This is why I created a "Don't destroy on load" function that allows us to keep the sound playing even if we already switched to another page.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class SoundEffect : MonoBehaviour
{
    void Awake()
    {
        var source = GetComponent<AudioSource>();
        DontDestroyOnLoad(gameObject);
        Destroy(gameObject, 3);
    }
}
```

Figure 5.21: Example of a "Don't destroy on load " code

Of course, I still needed to keep the website up to date. This is why I took some screenshots of the game in order to put them in the gallery of the website. The images on the website are important because it gives a first impression of the game. This is what will attract new players to the game.

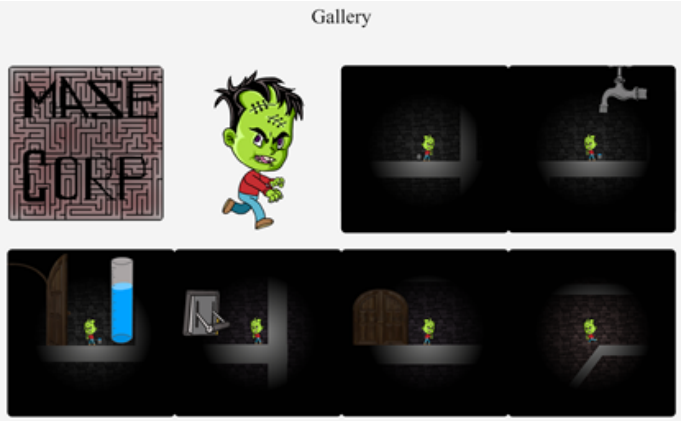


Figure 5.22: Picture of the website's gallery

I also added a table showing what we managed to do in comparison to the last de-

fence. We can see that we corrected some tables. In fact, while preparing our third defence we realised that some tasks that we thought we had finished before were not really finished. For example, the line with “Music and sounds” was filled with “Done” because we had found all the music and sounds. So, we marked that task as finished, but it was finished only for the second defence since we had to implement those music and sounds. Also, some lines have been added like “objects” which is the creation and implementation of the various objects in the maze. These objects are the one we use to solve enigmas. For example, this includes the doors we are supposed to open, the bucket that let us transport some water or the lever. This is why we changed the tables a little bit.

What we did up to the third defence

Tasks :	Constance :	Léo :	Julien :	Manon :
Multiplayer		X		
Character mouvements		X		X
Level generation	X			
Game Mechanics	X	X		X
Music and Sound	X			
Level design				X
Objects				X
Character frames				X
LaTeX reports				X
Creation of enigmas	X	X	X	X
Animation and Appearance		X		X
Website	X			
Manuals				
Trailer	X			

Figure 5.23: Picture of the website’s table

We then decided to create a trailer for our game. This is why we recorded some parts of the game. Manon was in charge of recording her gameplay and she recorded a full game. It

was important to record essential points of the games like the enigmas and the multiplayer for example. I then made a video about it adding some text and music. The trailer was supposed to show the enigmatic aspect of the game. Once finished, the trailer was uploaded to YouTube and the "Trailer" button on the website was changed in order to redirect us to the video.

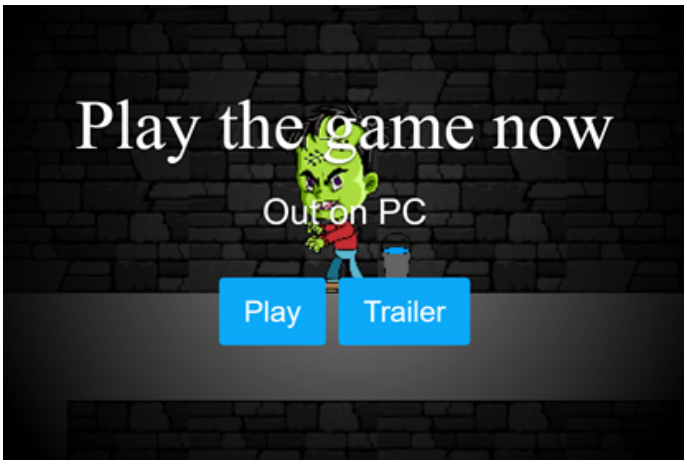


Figure 5.24: Button for the Trailer

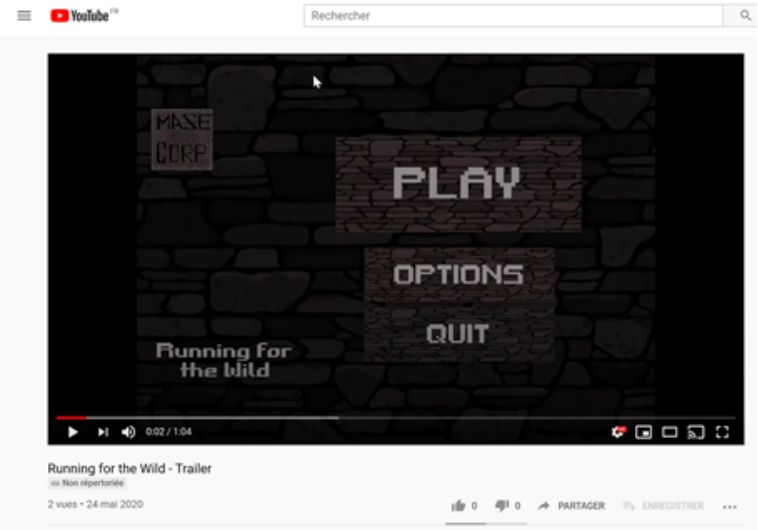


Figure 5.25: The trailer on Youtube

Overall, the making of this game made me discover and learn different things. First, the creation of the website has made me discover and learn the Html and CSS languages. To put the website on the internet I had to use a new tool GitHub which was very useful and I might use it again in the future for future projects. But I spent most of my time on Unity. This application is very complete and can be very efficient if well used. A considerable number of basic functions are already implemented in Unity and this makes the program more user friendly. Overall, this application is very complete and does not over-complicate things. For the making of the trailer I had to learn how to edit a video and I had to use a video editor called Movavi. This project allowed me to teach me the use of multiple tools but it also made me realise the importance of teamwork. To be efficient we need an organized team where each member can communicate and help each other. This is why this project was a great experience for me.

5.4 Julien Clauss

We did not receive any text/video or work from him.

6. Conclusion

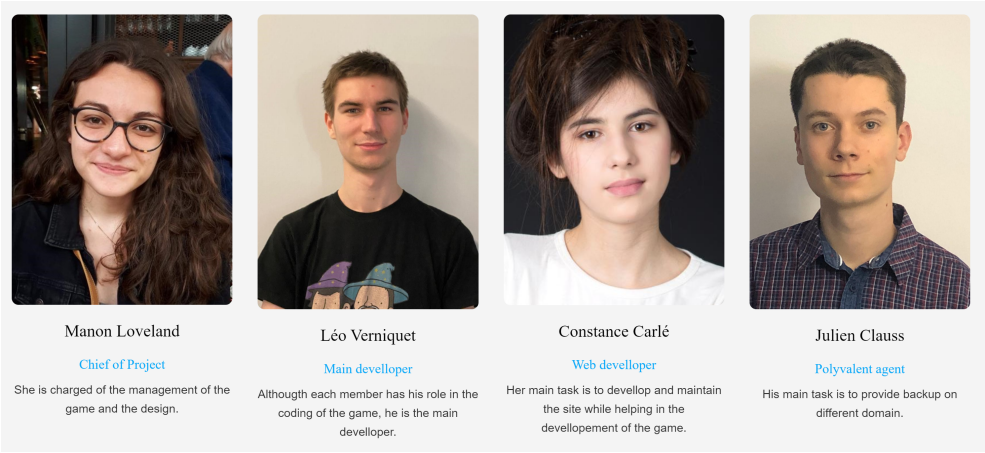
In conclusion for this last and final report, we have managed to completely finish our game Running for the Wild. We are very proud of this game as it is an adventure game where the player must run around in a maze trying to find the exit. But it's much more than just running around a maze, the player must also solve various puzzles and enigmas to escape which adds an element of logic to the game.

We drew characters, objects, the maze, coded the multiplayer, the character movements... The character figures and actions have been implemented within the game and the various designs are in action and are slowly taking their part in the game. In addition, we have sounds for most of the actions and scenarios in our game such as the music in the background, the sound of the character walking and running and even various victory and descriptive sounds.

We wanted our game to be enjoyable and fun for the player. Not only fun to play but we also had fun creating this game. This game is interesting in terms of originality for instance, but also be easy for the player to start playing and acquire the basics of the game, thus the player will not need any previous experience before playing this game. In addition, to make this game original, and not like so many other maze games available on the web, we incorporated enigmas for the player to solve whilst playing, which will allow the player to think before acting, and think rather than just jump and run. To accentuate this perspective, the player will be against other players, which means a competitive side of him or her will show.

Creating this game has been very fun and has allowed us to individually grow and learn new and exciting aspects of creating a video game. We had fun creating it even though it was tough at some points. It also taught us how to work as a team but also find solutions quickly when a problem arises.

In conclusion, creating Running for the Wild has been a great experience for all of us and has taught us the principles of creating games, working as a team but also all the other small tasks that go hand in hand with creating games such as writing reports, filming videos, designs, sounds... We hope you like the game as much as we do and hope you find it fun to play.



The PROgrammers (Manon Loveland, Léo Verniquet, Constance Carlé and Julien Clauss)

Bibliography

Sources

Youtube Videos for creating a bucket:

-https://www.youtube.com/watch?v=gdR0q2zj-Uc

-https://www.youtube.com/watch?v=fx33sPEAZEklist=PLa1F2ddGya_$UvuAqHAksYnB0qL9yWDO6index = 31$

$-\ https://www.youtube.com/watch?v = BZp8jbOGvGs$

$-\ Blender$

$Youtubevideofordrawingatap:$

$-\ https://www.youtube.com/watch?v = YrvTD0cCD74$

$-\ https://www.youtube.com/watch?v = WejiXdc223o$

$Youtubevideosfordrawingacharacter:$

$-\ https://www.youtube.com/watch?v = 61qUZgdH - aI$

$-\ https://www.youtube.com/watch?v = XxNUIRVOmMw$

$-\ ArtStudio$

$Photonmultiplayer:$

$-https://www.youtube.com/watch?v = Lq4nw2CxIM8list = PLWeGoBm1YHVgXmitft-0jkvcTVhAtL9vGindex = 5$

$-https://www.youtube.com/watch?v = nn4I18fdDhMlist = PLkx8oFug638oMagBH2qj1fXOkvBr6nhztindex = 19$

2d movements

-https://www.youtube.com/watch?v=dwcT-Dch0bAlist=PLPV2KyIb3jR6TFcFuzI2bB7T MNIIBpKMQindex=2

-https://www.youtube.com/watch?v=2MImqNvgrNg

Music :

-https://www.youtube.com/watch?v=1a-u4Vz9q8o (1:49:22)

Youtube video for creating a website:

-https://www.youtube.com/watch?v=KsCUcBK9q14

-https://www.youtube.com/watch?v=f6tXN9hZvew

-https://www.youtube.com/watch?v=yFI7P24iE4U

-https://www.youtube.com/watch?v=ppauAg1JKNgt=97s

-https://www.youtube.com/watch?v=i4n6HVtbBX8t=142s

How to create a button:

-https://fr.w3docs.com/snippets/html/comment-creer-un-bouton-html-agissant-comme-un-lien.html

How to create a table:

-https://openclassrooms.com/fr/courses/1603881-apprenez-a-creer-votre-site-web-avec-html5-et-css3/1606851-ajoutez-des-tableaux

Player :

-https://developer.mozilla.org/fr/docs/Web/HTML/Element/center

-https://www.youtube.com/watch?v=1QfxdUpVh5I

-https://answers.unity.com/questions/278883/how-do-i-make-my-player-die.html

-https://www.youtube.com/watch?v=nkgGyO9VG54