Ailouros - First report

Maxence Bornecque, Maïeul Boyer, Clovis Lefebvre, Bamlak Gurara

March 7, 2022

Contents

Abstract

This report provides a brief overview of what has been accomplished since the submission of the book of specifications, which parts are ahead of plan and which are behind schedule. Furthermore, difficulties faced and solutions adopted have been discussed. Finally, it describes what has to be done for the next presentation.

To keep things simple, Ailouros is a game in which we play as cats fighting for food and life. The players are organized into two teams of one to five cats. While fighting the enemies, each team will be tasked with hunting preys on the map, and getting them back to the team's base. The game is won by the first team to gather the target amount of preys.

This game is currently being developed by Team Purr-fect in C# on Unity, with the support of additional software such as Blender for 3D assets, and Garage Band / Audacity for audio assets.

For this initial stage of the project, we intended to create the game's essentials, which included constructing a functional multiplayer mode, a 3D character with the ability to move in all directions, a user interface that shows the current life status of the player and a level design with some sound effects. We also completed the game's main mechanics, and the website has been updated since we handed in the book of specifications. Basically, we concentrated on the most significant elements.

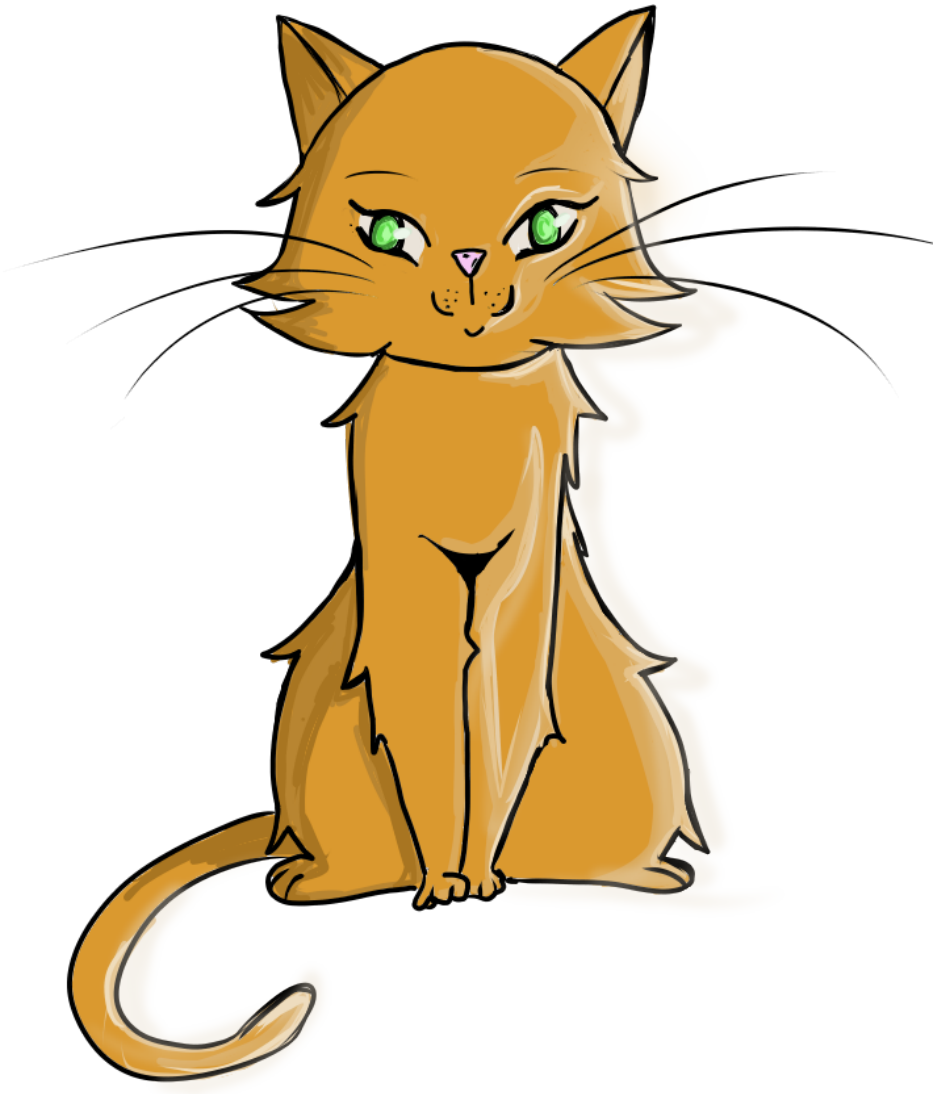

Figure 1: Ailouros logo

Chapter 1

General progression

This section of the report details how far we've progressed toward completing our project since the submission of the book of specifications. It determines which portions are on time, and which are ahead of schedule.

Task	Expected	Completed
UI	30%	30%
Character movements	50%	60%
Combat system	50%	50%
Multiplayer	60%	60%
AI	20%	20%
Level design	40%	40%
Particle Effects	5%	5%
3D modeling	15%	33%
3D animation	10%	10%
Music	15%	33%
Sound Effects	5%	5%
Website	60%	60%

We are a bit ahead of schedule in three parts: character movements, 3D modelling and music. Indeed, the movements are nearly complete, and the cats only miss a way to crouch. On the modelling part, we have most 3D models, and only miss good preys and foxes. Finally, we already have a menu music, and only miss an in-game music and a combat music.

Chapter 2

Obstacles

One member of our team (Maïeul) can't work on the project for now because of medical issues, so his substitute (Maxence) continued his tasks. Maïeul had already researched how to handle the multiplayer part and helped Maxence put up a working prototype.

Speaking of the multiplayer part, we had to handle some incompatibility bugs between Windows and Linux, and sometimes the coordinates didn't synchronize correctly between two computers. One player would sometimes fall and respawn infinitely (see figure 2).

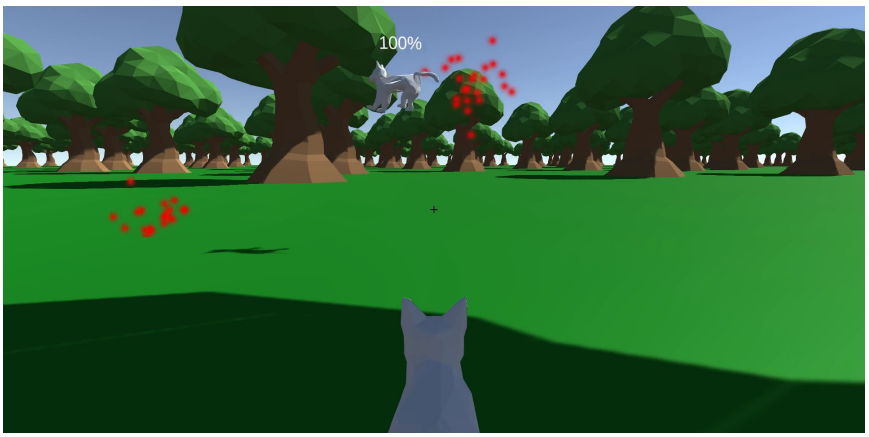

Figure 2.1

Moreover, we had a lot of problems implementing the combat system. Indeed, while the position of a player was automatically synchronized, the dealing of damages to another player was harder. We had to go through the documentation of Mirror to figure out how to synchronize the damages between clients, and with the UI.

Chapter 3

Task distribution

Maïeul is not available for now because of medical issues. Thus, Maxence took on his role as a substitute to Maïeul. Luckily, Maïeul had already researched the multiplayer part and helped Maxence find the resources and libraries to use.

Maxence is the group leader. He distributed the work and kept track of the work done, in progress, and to do. He coded the multiplayer part, the character movements and the combat system. He also started the UI, the AI and the website. This seems like a lot of work, but he simply walked in the steps of Maïeul for his substitute's tasks.

Clovis is a creative person and already had experiences with Blender. That's why he is in charge of the 3D modelling, 3D animation and level editing as Blender is one of his strengths. He helped for the AI part as Maïeul wasn't available.

Bamlak is mainly in charge of the aesthetics of the game. Which includes the music, sound effect and particle effect.

The task distribution didn't change.

Task	Main	Substitute
UI	Maxence	Bamlak
Character movements	Maxence	Clovis
Combat system	Maxence	Clovis
Multiplayer	Maïeul	Maxence
AI (preys and foxes)	Maïeul	Maxence
Level design	Clovis	Bamlak
Particle Effects	Bamlak	Clovis
3D modeling	Clovis	Maïeul
3D animation	Clovis	Maïeul
Music	Bamlak	Clovis
Sound Effects	Bamlak	Clovis
Website	Maxence	Maïeul

Chapter 4

Achievements

This section is about our achievements in each part of the game.

4.1 Multiplayer

Maxence handled the multiplayer part.

It was the longest part but it was also straightforward. We used the Mirror library, and followed an official tutorial, the Quick Start Guide.

While following the tutorial, we encountered a lot of problems, as we didn't blindly follow the path given. We had our own goals for Ailouros, and we had to do a lot of stuff differently, while learning from the ideas of the tutorial. As stated in the obstacles part, we went through the whole documentation of the Mirror library to be able to achieve what we want.

4.2 AI

Maxence and Clovis worked together on the AI.

For now, we made a program for the preys. Later, we are going to improve it and design an AI for the foxes.

```
using System.Collections;
using System.Collections.Generic;
using System.Numerics;
using UnityEngine;
using Vector3 = UnityEngine.Vector3;

public class PreyController : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {
    }

    // Update is called once per frame
    void Update()
    {
        var allPlayers = GameObject.FindGameObjectsWithTag("Player");
        //Debug.Log($"players: {allPlayers.Length}");
        var move = new Vector3(0, 0, 0);

        foreach (var player in allPlayers)
        {
            var heading = this.transform.position - player.transform.position;
            heading.y = 0;
            if (heading.sqrMagnitude <= 2.0 * 2.0)
            {
                move += heading.normalized;
            }
        }

        move *= Time.deltaTime;
        transform.Translate(move);
    }
}
```

Figure 4.1

To code this, we started with the idea that when a cat gets close to a prey, the latter tries to flee from the cat by going in the opposite direction. If there are multiple cats, we sum the forces before applying them.

4.3 3D modeling & animation

Clovis handled the modelling and the animation.

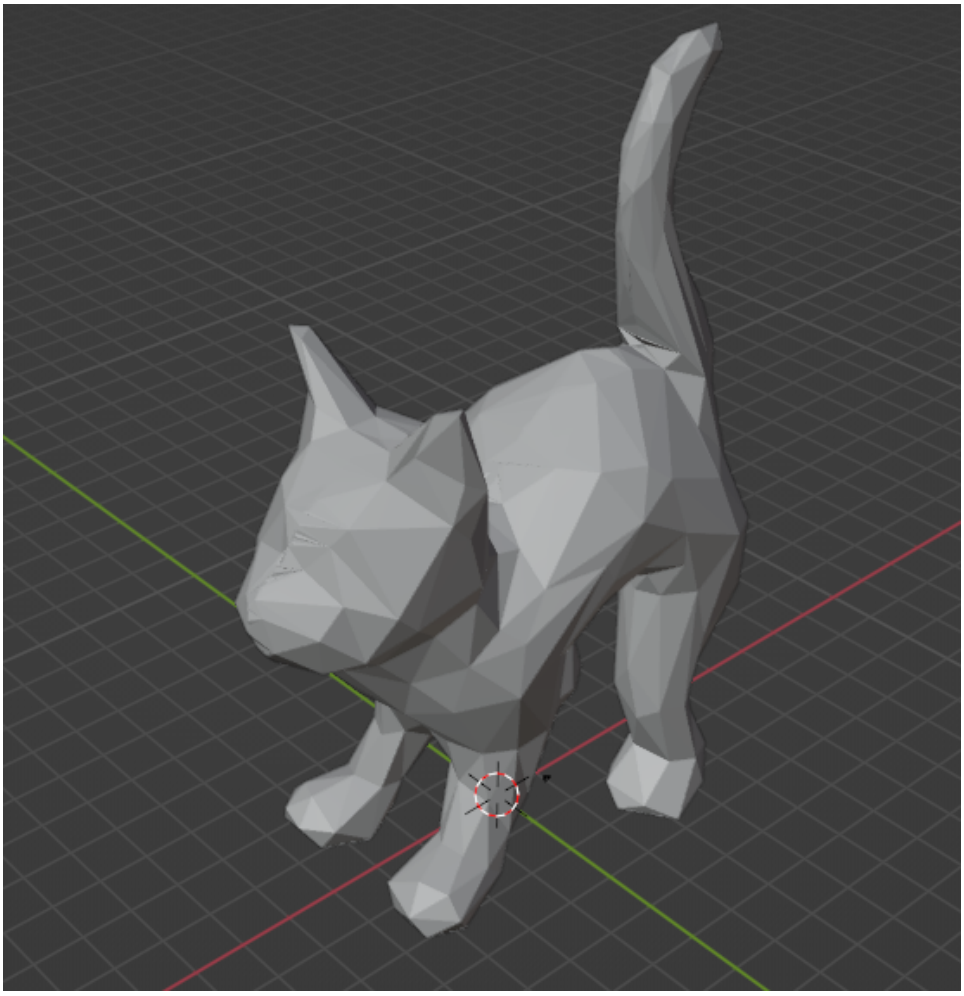

Figure 4.2: the original cat model

When we modelled the cat model that the player controls, we used a low-poly cat model from andresrojas on Thingiverse released under a Creative Commons license.

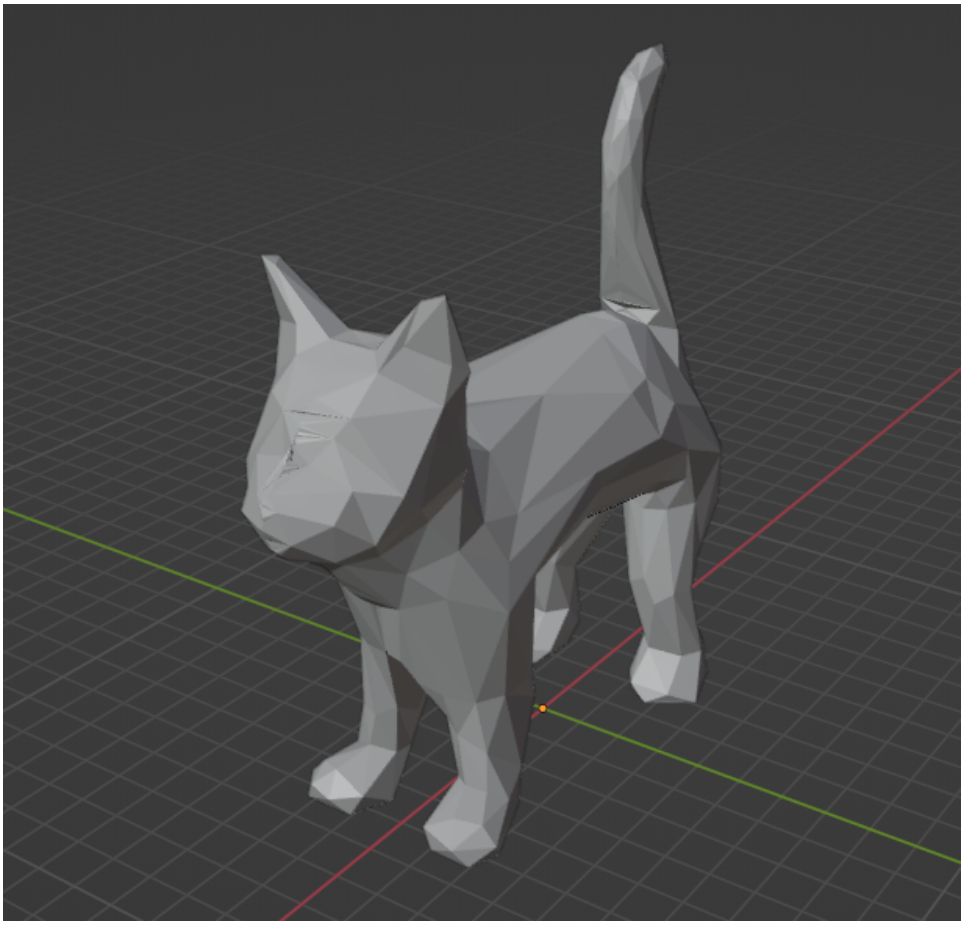

Figure 4.3: the edited cat model

Then we edited it so that we get a model easier to animate.

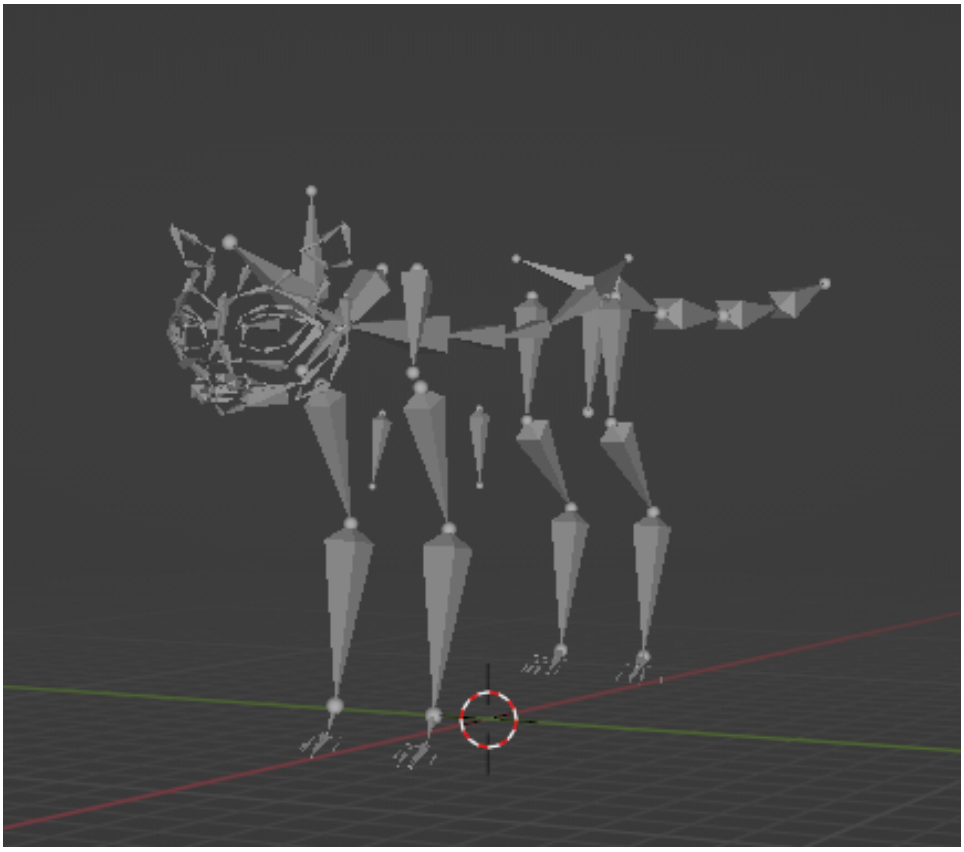

Figure 4.4

After that, on Blender, there is an add-on called "Rigify" that allows you to get pre-made skeletons.

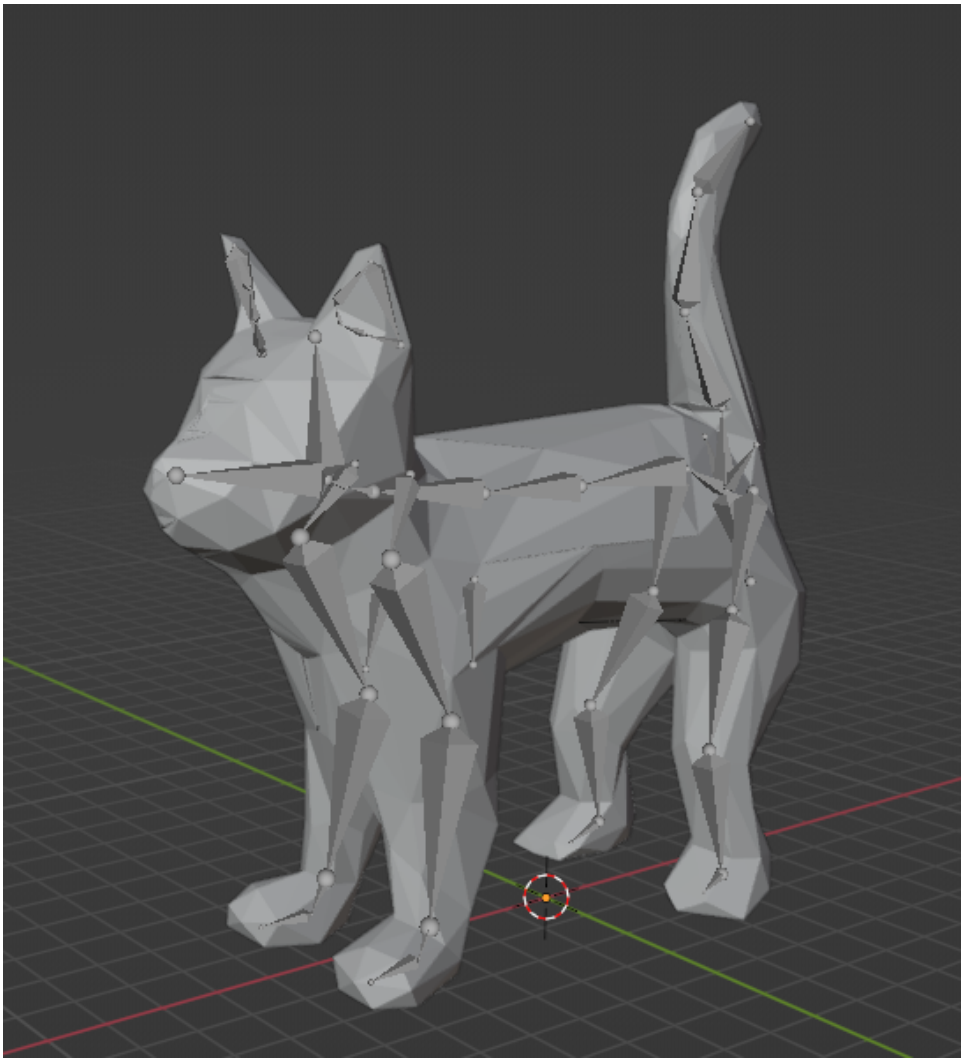

Figure 4.5

Then, by improving it and adapt it on our cat, we got this skeleton that allows us to animate our cat the way that we want it to be.

Figure 4.6

About the trees, for the first defence, we added only one type of tree that we found on TurboSquid. However, we are going to add more types of trees later.

4.4 Particle effect

Bamlak handled the particle effect. We started working on particle effects using Particle Systems. Particle effects are made directly from Unity and are especially useful to represent the game environment. A red "pop" will be displayed to announce a player's death.

Figure 4.7

4.5 Music

Bamlak was in charge of the music.

To make the music of the game, we made it in Garage Band. So, at the beginning, we thought about a music for the main menu that was calm and exclusively made with an instrument called the 7th player synth.

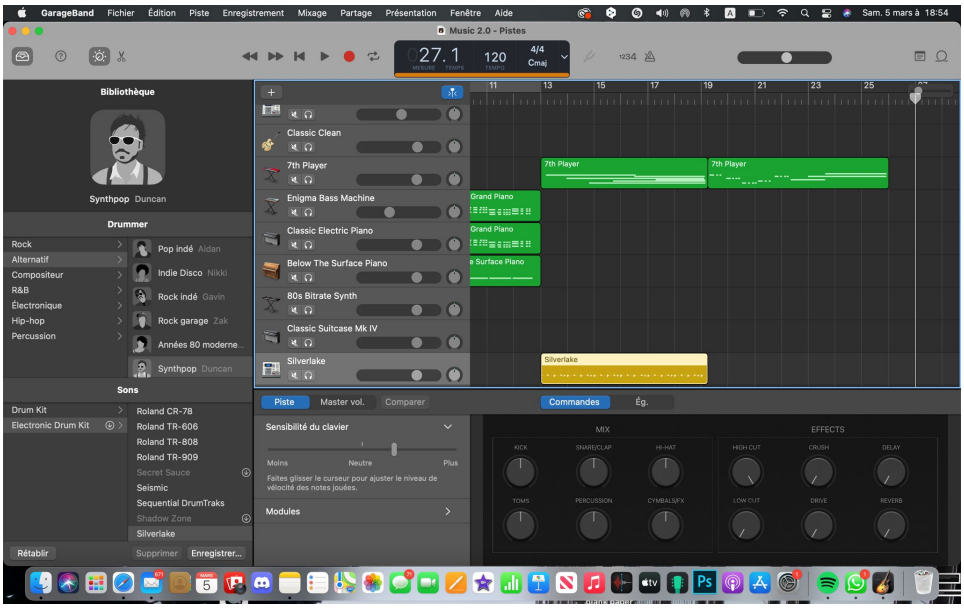

Figure 4.8

But finally, we decided to make a music that was happier and more welcoming to the player when they come into our game. That's why we finally used a music made of a classic clean accompagnied by a Big Room.

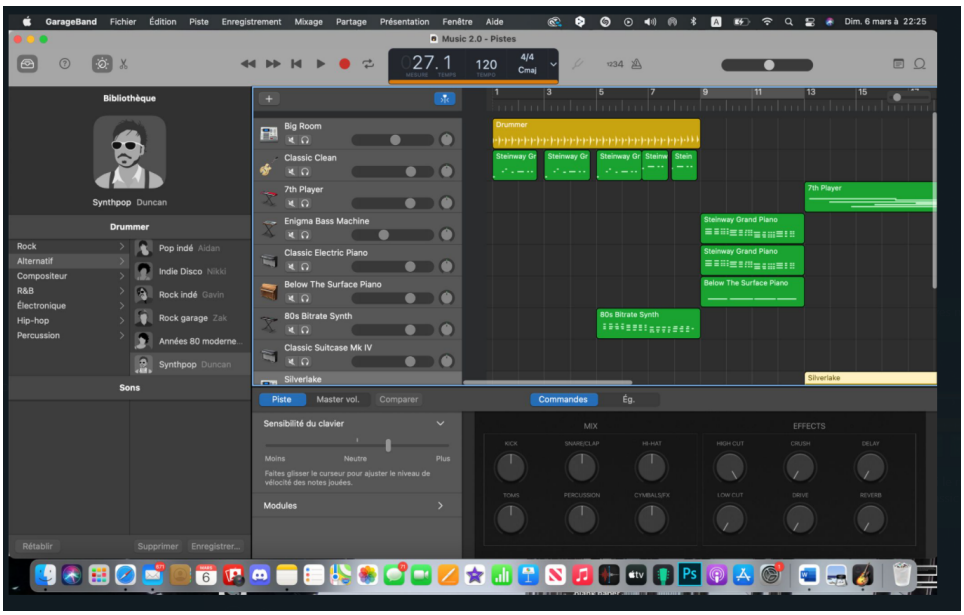

Figure 4.9

4.6 Website

A website was created by Maxence to keep track of the project's development.

This website is used as a way to explain what the game is, who is behind the project and other relevant information such as providing the book of specifications of the game.

It is available at the following address: `https://purr-fect.github.io/`

It is made of simple HTML combined with a lightweight CSS template: Simple.css. Thanks to it, our website is beautiful and is both available with a dark and a light theme, and it works seamlessly on both mobile and desktop platforms.

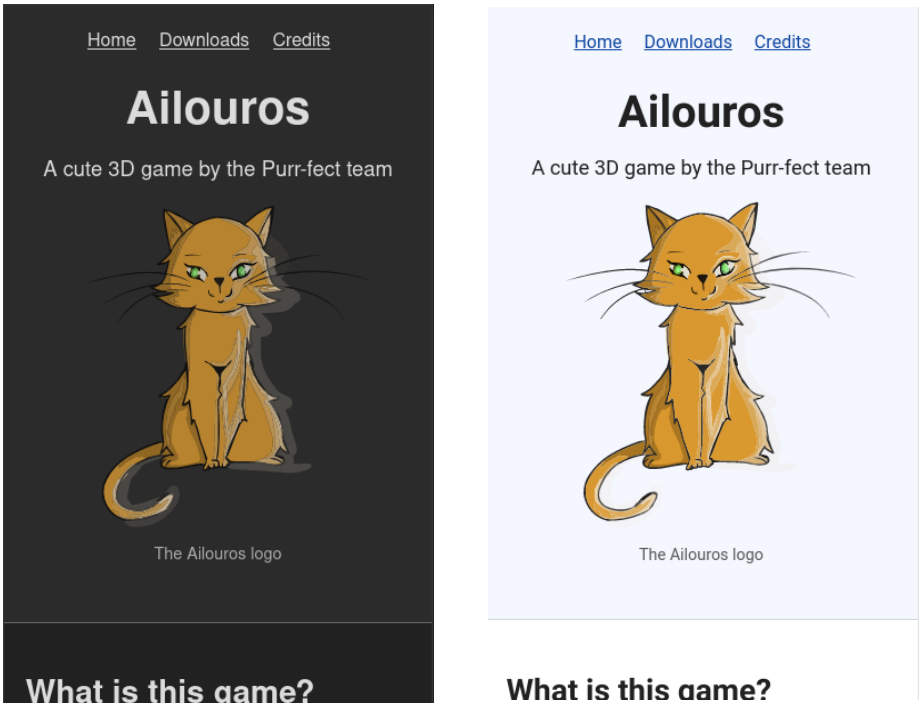

Figure 4.10: The website on mobile

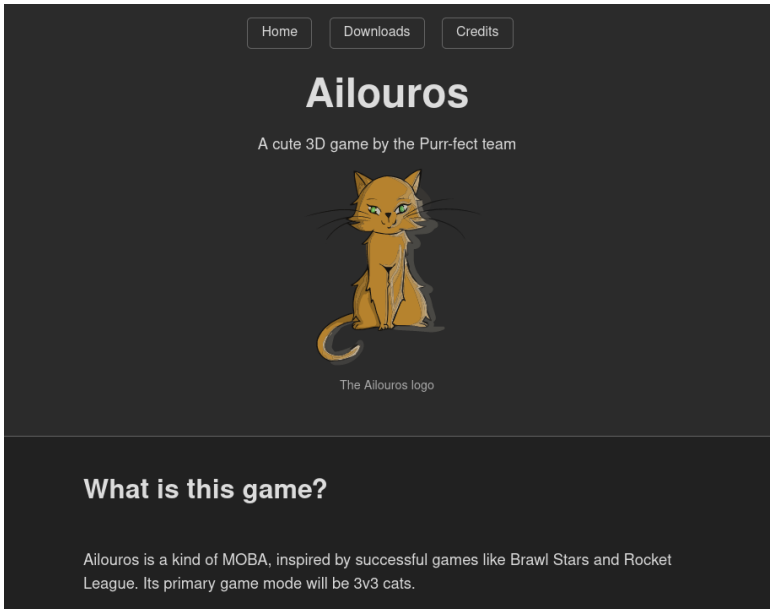

Figure 4.11: The website on desktop (dark)

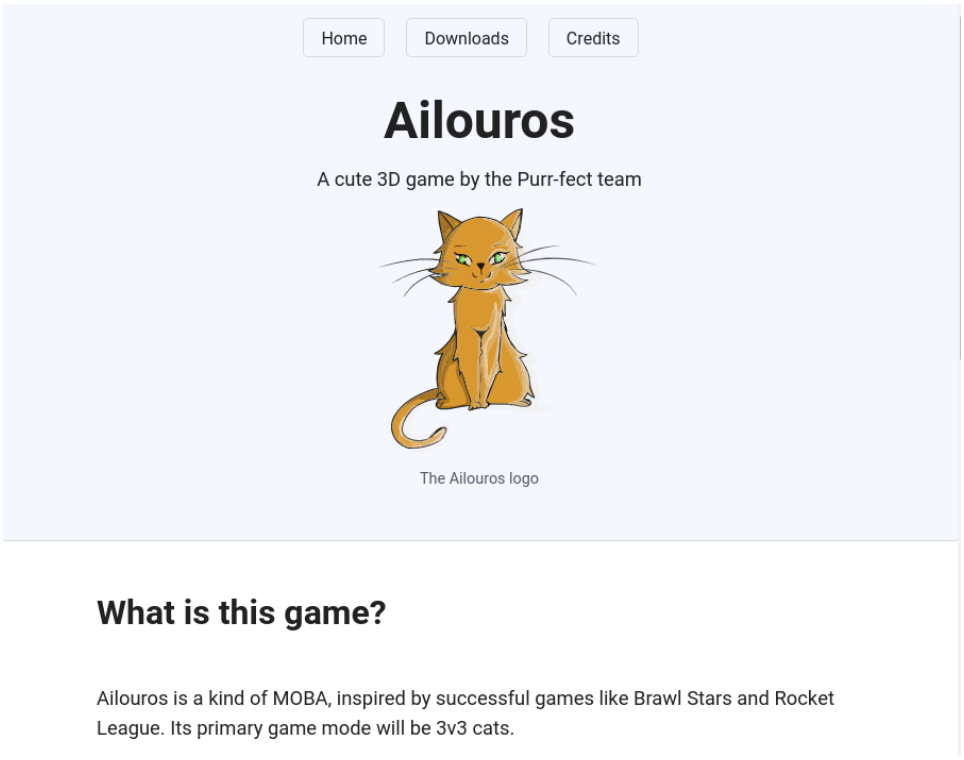

Figure 4.12: The website on desktop (light)

In terms of features, the website is made of 3 pages: a home page, a downloads page, and a credits page.

The project is presented on the home page like a FAQ. The team members and the assets used are listed on the credits page. Finally, the game, the specifications, and the report are all downloadable.

4.7 Level design

Clovis handled the level design.

For this part, we first thought about making a map like the following.

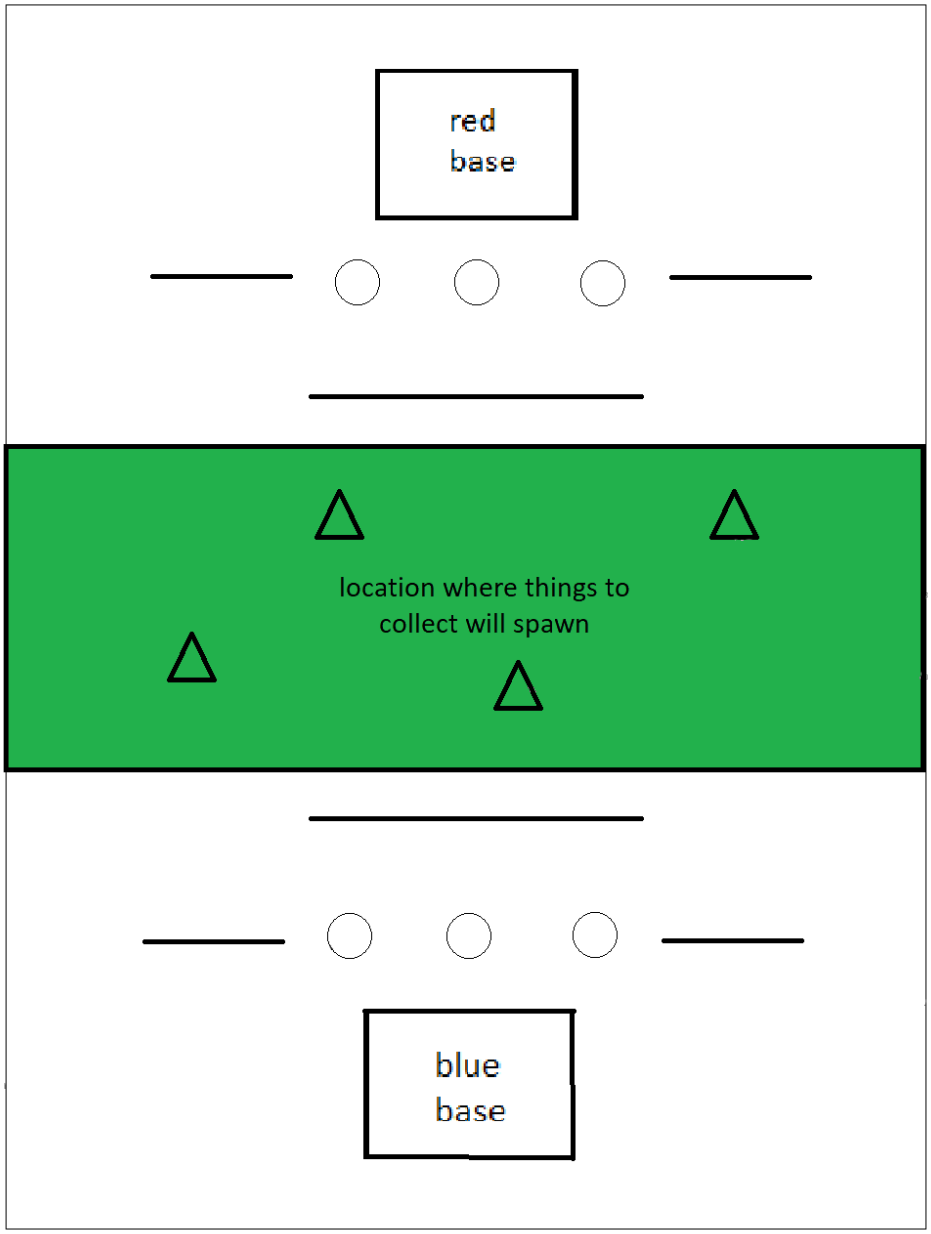

Figure 4.13

Then, Clovis edited the map by replacing the rectangle representing the forest with a circle so that we have a more interesting map where players can't hide in a corner. Moreover, we removed obstacles near the 2 bases because we think that the forest is big enough to slow down players.

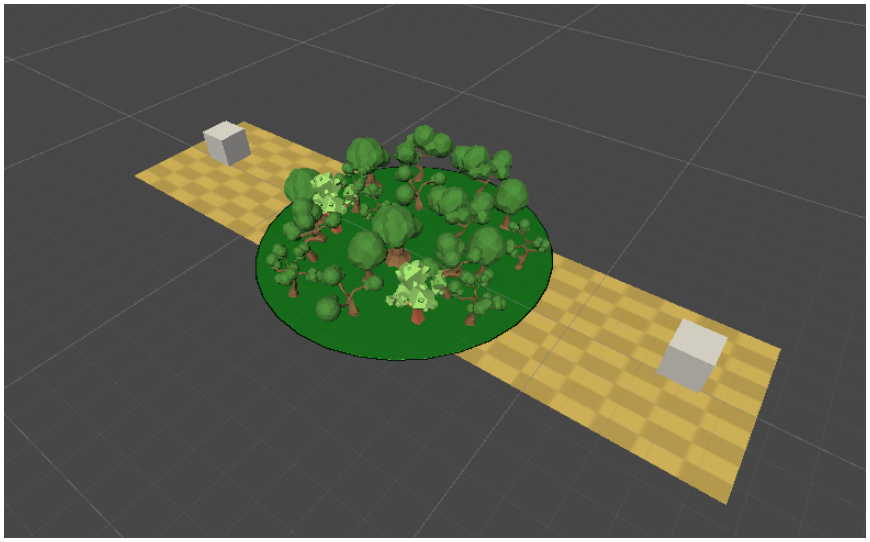

Figure 4.14

Finally, the last version of the map we kept is the following.

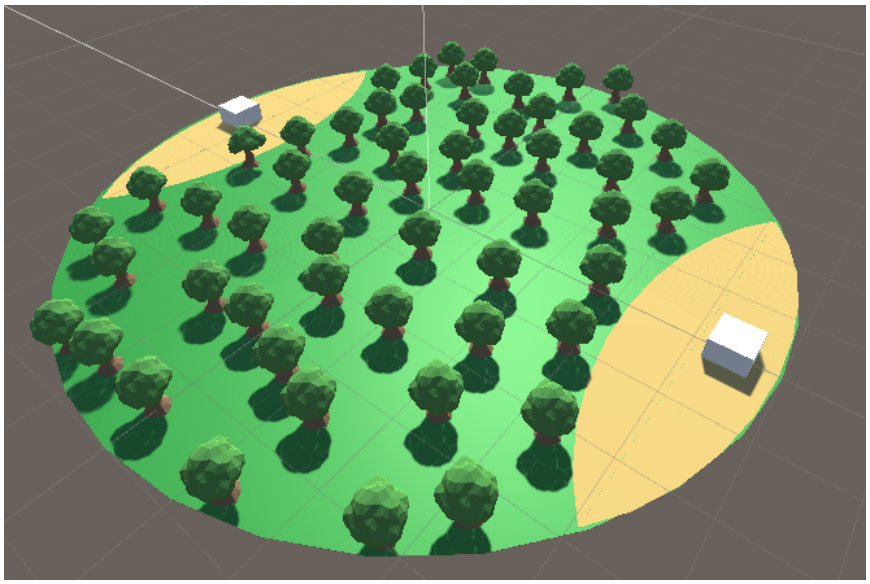

Figure 4.15

So, instead of having different shapes between the forest and the cats' bases, Clovis put all of them on the same circle and reduces the bases areas so that cats have more spaces to fight and collect preys. Moreover, it helped us to add more trees and have the possibility to spawn more preys for a better gameplay.

Chapter 5

Expected progression for the second defense

This section of the report details the expected sequence of work to be accomplished by the second defense. We didn't think having a second column for the last Defense is useful, as everything should be done at 100%.

Task	2nd Defense
UI	60%
Character movements	75%
Combat system	75%
Multiplayer	90%
AI	60%
Level design	60%
Particle Effects	40%
3D modeling	55%
3D animation	45%
Music	65%
Sound Effects	65%
Website	90%

Chapter 6

Conclusion

To sum it up, despite certain issues that have been successfully overcome, the team is now on track with all tasks. The website is up and running and the most crucial parts of the game have already been developed.

We have already built a strong foundation for this game with all the necessary mechanisms and features. Moreover, the coming features will be mostly adding fun to the game, and will be quickly rewarding to us as developers. Thus we are confident that we will be a bit ahead of our schedule for our next defense.