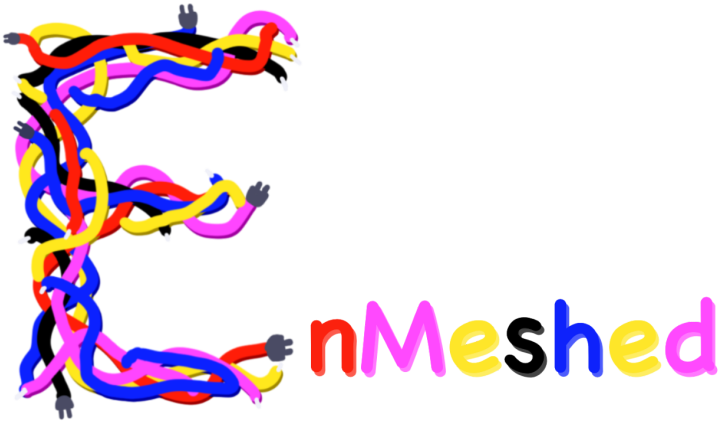
EnMeshed

Book of specifications



A Project by:

Louis VOLARD
Remi TROTEL
Sarah TSANGOU
Arthur GAILLEUL

Table of Contents

1. Introduction

1.1. The idea

1.1.1. Origin

This project first started from a simple statement : we like *among us* and we like *hellapagos*. The idea of mixing them into a new game quickly emerged amongst our group.

We picked this name after a heated debate that lasted for an entire night. We finally settled down on “EnMeshed” because it brought together everything we wanted. It carried at the same time the idea that players were stuck in this ship and needed to escape, but also with “Meshed” the fact that players will have to cooperate to win. Additionally, the entangled wires visible on the game logo represent the “entanglement” of players inside the vessel. Every player’s action in the game can and will have consequences on the others.

1.1.2. The main focus of the game

Once we had found the basic idea for our game, we quickly defined a few guidelines that we considered essential, basing our reasoning on observation about the games that we like. A few features stood out, including being **easy to understand** for beginners, being **quick** and **dynamic**, and granting players a **large freedom of possible actions and strategies**. We will thus try to follow these guidelines throughout the development of our game.

1.1.3. Inspirations

As stated before, our main inspirations were the two games *Hellapagos* and *Among Us*.

In *Among Us,* we really liked the simplicity of the game, allowing new players to quickly fall in love with it while maintaining a large amount of gameplay available which keeps players interested for a while. Furthermore, we liked the visual aspects of *Among Us* such as its 2D Gameplay which will be one of the core aspects of our game.

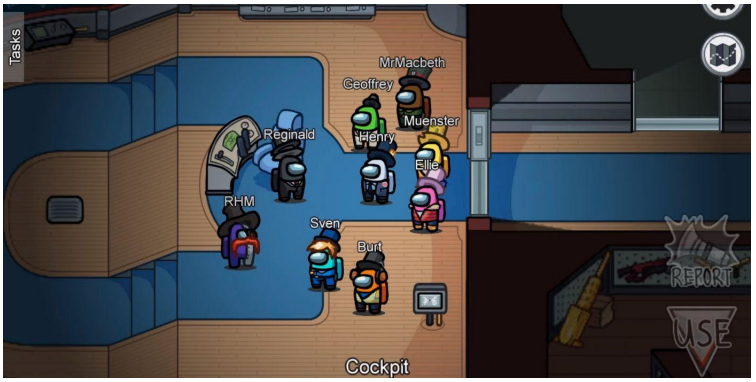


Figure 1: Screenshot from the Among Us game

The second major inspiration for our game was the cardboard game “Hellapagos” (or “Galerapagos” in french). This game about survival on a stranded island after a shipwreck was our main inspiration for the gameplay we wanted. We particularly like the fact that this is a cooperative game, until one player decides it is not anymore. Thus a lot of the rules and gameplay of our game is inspired from this cardboard game. We also used *Hellapagos* to imagine our item mechanics.

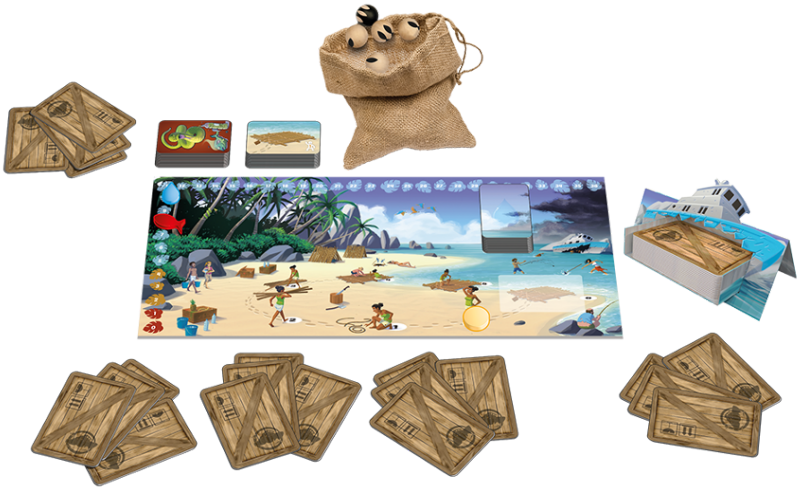


Figure 2 : “Galerapagos” Cardboard game

1.1.4. Basic Technical Aspects

The project is meant to be a 2D game, we will mainly develop it using Unity’s tools. Every asset we will use are either going to be freely available assets or self created ones.

The game will run on Microsoft Windows OS for the moment while making sure we leave open the possibility of deploying it to other operating systems and devices for potential future development on our free time.

1.2. SyntaxError

1.2.1. Origin

The name of the group was picked pretty unanimously. It originally comes from a joke, where Rémi named an Exception in his code “SyntaxError”, to confuse those who were going to test it (because he loves messing with the ACDCs). We believe this name would represent our group well as SyntaxError is a reference to coding, which we love, and also to mistakes which we will inevitably make as beginners and students.

1.2.2. Sarah TSANGOU

I have always been intrigued by computer science, especially when it was related to video games. Today I want to explore the programming fields. With EnMeshed I will be enjoying the process of developing a video-game. I am sure that this project and the SyntaxError Team will make me grow.

1.2.3. Rémi TROTEL

I first learned to code back in primary school, before even learning what a division was. Since then, I have always been passionate about it, and spent a lot of time coding a large variety of projects. I have to admit, most of them are useless, but for me, it was primarily a way to learn and have fun. I really look forward to this project, as a way to challenge and improve my coding skills, but also to discover all other aspects of game development.

1.2.4. Arthur GAILLEUL

At a young age I was always intrigued by everything related to computers and eventually to programming. I've also been attracted to video games as a toddler so this project will make me experience what it is to create a video game. I still have a lot to learn and through this project I believe that I will be able to mature as a programmer and at the same time enjoy, have fun.

1.2.5. Louis VOLARD

Coding has been very important for me for the past few years. Having used Unity numerous times in the past and with my natural interest in Videos Games, I was very interested in making a video game in Unity for our S2 project. I am particularly interested in working on the multiplayer aspect of the game and hope to learn new ways to collaborate on somewhat large coding projects.

I have been chosen as Group Leader for my skills in English which will be particularly useful during the group presentations.

2. EnMeshed

2.1. Game Rules

2.1.1. Synopsis

Space exploration : a dream for many humans that suddenly became a nightmare. Hit by an asteroid field, your ship is in critical condition. Many components are out of order, and the ship is slowly drifting towards the nearest planet. Even worse, the escape pods have been badly hit too. With only a few minutes before the impact, will you and your crew be able to repair the ship and regain control ? Or will you try to repair an escape pod and bail out of the ship, leaving your fellow crew members behind ?

2.1.2. Main Objective(s) of the players

To win the game, you must survive. It can be done either by bailing out of the ship before the impact thanks to an escape pod, or by avoiding the impact by repairing the vital components of the ship as a group. That also means that either the entire crew can win, or only a few, or even none at all if all crew members are still in the ship at the time of the crash.

The entire crew can win by repairing the spaceship. To do so, they must repair every vital component of the ship. This task is long and will need a lot of resources, and one player can't do it alone. The crew will need to work as a team, sharing tasks and helping each other.

A player can also ensure his victory by repairing a personal escape pod. Escape pods will be easier to repair, but they need the same materials as the vitals components of the ship, meaning that a player trying to escape alone will slow down the progress of the entire crew to repair the ship.

And what if every player tries to escape alone ? We solved this problem by reducing the number of escape pods available. The exact number will depend on the number of players in the game, but there will never be enough escape pods for the entire crew to escape (this is represented in game by the fact that some escape pods are too damaged to be repaired at all).

2.1.3. Typical Playthrough

- **Collecting resources**

The collection of resources is the first thing that players will have to do. Indeed, no matter what strategy you are planning on using to get out alive, you will need quite a large quantity of them. This will lead players to scatter around the map to collect as much as they can. Time being limited, players need to be as efficient as possible to find these resources. Because of the need for multiple types of resources, they also need to coordinate, so that they can collect everything they need before the time runs out.

- **Maintaining Oxygen Level**

In order to survive, players will need to breathe oxygen. The ship starts with a level of oxygen dependent on the amount of players present. That level will drop through time according once again to the amount of players in the ship. Players will be able to generate oxygen via an action at the oxygen generator (c.f. 2.2.3 Actions). Incidents such as hull breaches may deplete the oxygen levels faster.

- **Repairing the ship**

  Using the previously collected resources and tools, the players will be able to try and fix the ship. Multiple components, such as the engines, are vital so the crew needs to repair them if they want to win as a crew, and some, like hull breaches, are not, meaning that the crew can ignore them and still win. However, these damages will still have a negative impact, and therefore, the crew will have to decide whether to spend resources to repair it, or to endure the effects of the damage. For example, hull breaches will drain the oxygen at an accelerated rate, forcing the crew to replenish the oxygen more often until they fix it.

- **Between cooperation and betrayals**

  We expect the players to begin the game with a cooperative approach, meaning that (almost) everyone tries their best to repair the ship. Players choosing to play alone at the very beginning will often result in the crew trying to get rid of them. Additionally, a single player will never be able to collect alone all the resources necessary to repair the ship. Thus playing cooperatively, at least during the early stages of the game, is necessary.

  However, as the game goes on, if the repairs are not quick enough, some players will start to think about their own survival. This is encouraged by the fact that they will find some items that can help them escape alone. Players will have to decide when to switch from helping the crew to helping themselves. Switching too early means that you are slowing the crew while the ship could still be saved and exposing yourself to a potential expulsion from the rest of the crew, and switching too late will mean that there will be no more escape pods for you. Players may also want to pretend they are working cooperatively while they are in fact working towards their own success. Every strategy is good, as long as it allows the player to survive.

- **Death of a player**

  There are multiple different deaths.

  The first, and the most notable, is when a player is still present inside the ship if/when it crashes on the planet.

  The second, is if a player no longer has enough oxygen to breath which leads him to a foreseeable death.

  Even if we want to discourage players from killing each other too often, it will still be an important part of the game. Whether you are trying to steal something from the crew to help yourself or trying to stop an other crewmate from doing the same, murders will be an important part of the different strategies that players can adopt.

  When players are killed before the end of the game, they will become spectators and will be able to cruise the ship, looking and listening at what the alive players are doing.

- **End of a Game**

  A Game ends when every player has either avoided the crash or died.

2.2. Main Gameplay Elements

2.2.1. Players

Each player embodies a crew member. They will be able to move around the ship while carrying a few items on them. Each player's inventory will be fairly restricted, forcing them to work together in order to carry out the necessary tasks.

To encourage teamwork, we want players to be able to interact as freely as possible. They will be able to give each other items so that they can find their own way of working together.

That also opens new opportunities for players that want to play alone, as they can try to steal items or resources from the different storages.

This freedom will allow players to interact in a lot of different ways, meaning that each game will be unique as to what players decide to do.

2.2.2. Rooms

The spaceship will consist of a set of rooms. Each room will have a specific function. Crew members will have to navigate between them in order to perform various necessary tasks.

The Dormitories are the spawning rooms of players, they will have a set of items to start the game in and an individual storage. Each player has a personal room and can use the storage they are given however they please.

The Meeting Room is the room in which crew members can gather to perform votes in order to eliminate other players if they wish to do so.

The Action Rooms are the rooms in which players execute tasks. For example, the engine room in which repairs are performed on the ship's engines, or the oxygen generator room are both action rooms.

The Collectable Rooms are the rooms in which items can be found by the player. For example, the storage room is a collectable room.

An interactive interface should be used by the player in order to perform actions in the two later types of rooms.

2.2.3. Actions

Players will have to perform different actions all around the ship. To do a task players will be given a simple mini game similar to "among us" tasks in order to finish it. Players will encounter them in a number of places, in order to collect resources and items, generate oxygen, repair components and some others.

2.2.4. Items

Through the course of the game, players will be able to collect items. While some are necessary to win, others are optional, and only give small bonuses. The latter are primarily intended as a way to introduce new opportunities to the player as to how to approach the game. Some of them are going to offer more opportunities as a crew, while others can give an advantage specifically to the player. Items will have a rarity as some items will be more common (like a screw) than others and others will be especially rare

(like a gun). It may happen that the player can encounter a useless item that neither gives him an edge or penalizes him (like a tissue). This will aim to mitigate the overcollection of rare items and reinforce the luck factor.

2.3. Multiplayer

2.3.1. Photon Networking Package

Because we only have a few months to develop the game, we excluded the possibility of coding an entire network manager from scratch. Thus we decided to look for a premade multiplayer manager plugin which we could use and adapt to our game. After an extensive search, we decided to choose Photon over other services such as Mirror or UNET. We realized Photon was at the same time very complete, with its different plugins, allowing us to implement every functionality we had in mind, and also easy to use and implement, allowing us to spend more time on other features we want to implement. Photon also provides a lot of its services for free, allowing us to use it during this project.

2.3.2. Creating/Joining Games

Our game is first and foremost a social game. Thus creating a solid multiplayer system allowing players to host and join games without any bugs or data loss will be a priority during the development of our game. Consequently, this will be one of the first, if not the first step to create our game. We chose peer to peer networking, allowing players to play together without the need for third party servers. To do so, We chose to use Photon PUN 2 Network Manager Plugin to set the groundwork for our networking system. PUN 2 is optimized for peer to peer hosting and allows server host migration (useful in case the server host disconnects) which makes it a very good tool for us to use.

Players will have three options when launching our game. The first will be to create a server for others to join. Server hosters will have some limited control and options to modify the server to their liking. The second option will be to search for an existing game which the player will then be able to join. Finally, if the player does not have a specific server to join, he will be able to enter a matchmaking algorithm, assigning him to any random open server. This feature will be particularly useful for players who do not have friends to play with or that aren’t currently playing.

2.3.3. Voip

As stated previously, we want our game to be interactive and dynamic thus we needed to find a way to allow communications between players that are not necessarily in the same room while maintaining the dynamic aspect of the game. For that reason, we ruled out the option of having a typed chat similar to the one in *Among Us* and decided to go for a voice chat. We will use Photon's Voip Plugin which will handle communications between players who will be able to coordinate or maybe form alliances, depending on the strategy they adopted.

2.4. Artificial Intelligence

2.4.1. Robot

The ship comes equipped with a robot originally designed to carry out trivial tasks for the crew. However, it apparently was not designed to withstand the shock of an asteroid field. The crew will hence have the possibility to spend some resources to repair it, so that the robot can help them repair the ship. Once repaired, the players will be able to assign it to simple tasks, allowing the crew to gain some efficiency. To preserve the game balance, the robot will have a limited set of tasks available.

The robot will therefore need to be able to navigate the ship by itself, but also to decide what to do next. Indeed, even if the task assigned to the robot is decided by a player, it will need to figure out the different steps to complete this task alone.

3. Project Realization

3.1. Tasks

3.1.1. Task Description

In order to distribute the work that has to be done between all the members, we have decided to separate all individual tasks and group them in "Task Groups" or "Fields". There are 9 task groups each containing individual tasks that need to be done. Almost every member is the lead "worker" on 2 task groups and assistant work on at least two others. Distribution of the tasks has been done in a way to take account of each member's individual skills, knowledge, and motivation on said task.

Level Design:

- Creating/Generating Maps/levels
- Implementing assets created by other task groups on the map
- Compiling everything everyone has done into one playable game
- testing playability of the map and implemented assets

Graphics:

- Creating a logo/visual identity for the game/group
- Creating UI assets for menus and the website
- Creating graphic assets for any ingame element that may require it
- Creating character models, graphics, and animations (w/ “character mov/attributes”)

Gameplay Mechanics:

- Creating/developing all scripts/game objects relating to the playability of the game: objectives, missions, interactables, …
- In charge of the progression of the player in the game

Character Movements and Attributes:

- Creating Player models/animations (w/ “graphics”)
- Creating/developing player movement scripts (w/ “multiplayer”)
- Developing the character inventory system
- Developing the different characters present in the game and their attributes

Multiplayer:

- Setting up the multiplayer environment handling server hosting and joining
- Adjusting scripts and assets made by other members in order to ensure they are compatible and optimized for the multiplayer environment
- Ensuring the correct synchronization and the lack of data loss between different clients
- Synchronizing player movements (w/ “player mov/attributes”)
- Setting up Voice Chat

Artificial Intelligence:

- Path Finding for the robot to move around on the map
- Decision Making for the robot to decide the steps to fulfill it’s given task

User Interface:

- Designing all menus and user interfaces
- Implementing UI graphics created by “Graphics”
- Coding/setting up the menus in order for them to be interactive

Sound:

- Creating/implementing soundtrack for main menu
- Creating/implementing specific sounds for specific actions: walking, pushing buttons, door opening, firing a gun, …

Website:

- Designing and creating a website that presents the project and allows users to download the game and all documentation relative to the project
- Updating the website throughout the development of the project to ensure it contains all the latest news

3.1.2. Task Distribution

Task/member	Louis	Remi	Sarah	Arthur
Level Design			Lead	Assistant
Graphics			Assistant	Lead
Gameplay Mechanics	Assistant	Lead		
Character movements/ attributes		Assistant	Lead	
Multiplayer	Lead		Assistant	
AI		Lead	Assistant	
User Interface		Assistant		Lead
Sound	Lead			Assistant
Website	Assistant			Lead

3.1.3. Expected task progression and schedule of work

Task/member	1st Presentation	2nd Presentation	Final Presentation
Level Design	30%	45%	100%
Graphics	25%	50%	100%
Gameplay Mechanics	50%	85%	100%
Character movements/ attributes	65%	90%	100%
Multiplayer	50%	75%	100%
AI	0%	50%	100%
User Interface	40%	75%	100%
Sound	10%	50%	100%
Website	50%	75%	100%

3.2. Tools Used

3.3. Financial Aspects

3.3.1. Current Economic Model

For us, this project is meant to teach us the basics of development, and has no real commercial goal. We all quickly agreed that developing any sort of business plan for the current state of the project seemed out of place, leading us to decide to make our game free to play and without any in-game payment. Furthermore, not planning on receiving any financial revenue from the development of this game, we chose only free to use software and plugins to create it. Currently, the only paid program we use is Procreate which Arthur bought on his own before the start of this project

3.3.2. Potential Economic model

With all that being said, we still considered what could have been or maybe could one day become of this project if our main goal became to be profitable. We first of all excluded the possibility of making the game a pay to win which felt contrary to the values which are dear to us and which led us to make this game in the first place. We oriented ourselves towards the possibility of making a paid version of the game, potentially for a few euros. We wouldn't necessarily need much income to become profitable. Indeed, we are not planning on hosting any servers for the game as it is a peer to peer networking system thus the only planned expenses currently are those of paid plans for photon or other software which would become necessary once launching the game to an extended audience.

4. Conclusion

EnMeshed is a project that aims at creating a fun game that everyone can play with their friends. Between cooperation and betrayal, it's up to the players to decide. Even if simplicity is a core component of the game, it doesn't mean that the project will be simplistic. With many game mechanics, this project will require a lot of work, but we believe that the idea is worth it, and that we can create a game enjoyable for everyone.