

UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

Analyse et développement d'applications en Réalité Virtuelle pour la sensibilisation à la qualité de l'air

Rapport de stage ST52 – P2024

DREANO Hermance

Département Informatique
Filière MV - Mondes Virtuels
Filière IA – Intelligence Artificielle

ATMO GRAND EST

5 rue de Madrid
67 300 Schiltigheim
<https://www.atmo-grandest.eu>

Tuteur en entreprise
GECHTER Franck

Suiveur UTBM
DRIDI Mahjoub

Remerciements

Je tiens tout d'abord à remercier **Franck Gechter**, mon tuteur de stage, pour l'opportunité de ce projet et pour ses conseils.

Je remercie également **Alexandre Desbos**, qui m'a offert son soutien et ses compétences, en tant que co-développeur sur ce projet. Merci à lui, **Thibaut** et **William**, mes collègues de bureau, pour la bonne humeur qu'ils apportaient chaque jour durant 6 mois.

Enfin, je remercie tout Atmo Grand Est pour leur accueil chaleureux, et en particulier **Raphaèle Deprost** et **Mélanie Trommenschlager** pour leur aide et implication dans nos activités de développement.

Table des matières

1.	Introduction.....	5
2.	Présentation de l'entreprise et de l'environnement de travail	6
2.1	Un projet européen : LIFE V-air	6
2.2	Un partenaire-employeur et un partenaire d'accueil	7
2.3	Les autres partenaires et les dynamiques	8
2.4	Organisation et suivi du projet	9
2.4.1	Réunions inter-AASQUA	9
2.4.2	Les autres actions de communication	10
2.4.3	Environnement de travail	11
3	État de l'art du projet	13
3.1	Un Serious Game dans sa version fonctionnelle	13
3.1.1	Univers et règles du jeu.....	13
3.1.2	Architecture du projet.....	16
3.2	Les prémisses de l'Escape Game.....	16
4.	Travail réalisé	17
4.1	Planning des tâches	17
4.2	La conception de l'escape game : Sensibilisation à la qualité de l'air pour les citoyens.....	19
4.2.1	Les fondations du jeu	19
4.2.2	La salle laboratoire.....	21
4.2.3	La salle “Origine de la pollution”	21
4.3	Le serious game : la machine énergie	24
4.3.1	Les blueprint dans unreal engine	24
4.3.2	Énergie : Fonctionnement de la machine avant modifications	25
4.3.3	La refonte de la machine	26
4.4	La création de l'environnement 3D	30
4.4.1	Processus de modélisation et texturage	30
4.4.2	Cas particulier : la végétation	36
4.5	L'ajout d'effets visuel	38
4.5.1	La création de maillages et textures pour un système de particules	39

4.5.2	Shader et animations	41
5.	Suite du projet.....	43
5.1	Serious Game	43
5.2	Escape Game	43
5.3	Documentation.....	44
6.	Conclusion.....	45
7.	Bibliographie	47
8.	Annexes.....	48

1. Introduction

Afin de conclure ma cinquième et dernière année à l'UTBM en branche informatique, j'ai effectué un stage d'une durée de 23 semaines au sein d'ATMO Grand Est, du 26 février au 2 août 2024. Après avoir suivi la filière Mondes Virtuels durant mon cursus, j'ai souhaité orienter mon projet professionnel dans ce domaine. C'est dans ce cadre que j'ai eu l'opportunité de participer au développement de jeux en réalité virtuelle pour un projet européen nommé LIFE V-air.

Aujourd'hui, la réalité virtuelle ne se limite pas uniquement aux jeux vidéo : elle se présente également comme un outil pour l'éducation ou encore l'apprentissage de gestes techniques dans un cadre industriel. On assiste également à l'essor des « serious game », ou « jeux sérieux » en français. Ces termes désignent un jeu mélangeant divertissement et contenu sérieux, pédagogique. Ces nouveaux outils ont donc pour objectif de transmettre des connaissances en adoptant les codes des jeux vidéo, tant au niveau des interactions dans un monde virtuel que des règles.

Durant ce stage, j'ai eu pour mission de concevoir et développer deux serious game en réalité virtuelle, destinés à sensibiliser les utilisateurs sur des enjeux environnementaux, notamment sur la qualité de l'air. Ce sont des enjeux émergents qui sont maintenant au cœur des préoccupations de nombreux secteurs. Le premier jeu est spécifiquement conçu pour des décideurs politiques, tandis que le second vise un public plus large, les citoyens, afin de leur faire prendre conscience de ces questions environnementales qui nous concernent tous. Le développement, ayant débuté en février 2023, s'est fait sur Unreal Engine. Ce logiciel est un moteur de jeu reconnu dans l'industrie du jeu vidéo comme référence. Ce travail et celui de conception se sont faits en collaboration avec d'autres partenaires français et européens, amenant une dimension internationale au projet.

Ce rapport retrace le déroulement de ce stage et les activités auxquelles j'ai participé, en commençant par une présentation détaillée du projet, des objectifs pédagogiques visés, des différents partenaires impliqués et de l'organisation globale du travail. Ensuite, après un état de l'art du projet, je décrirai les différentes étapes de création des jeux. Cela inclut la conception des expériences immersives, les spécificités de la réalité virtuelle dans ce domaine et les choix techniques et esthétiques réalisés sous Unreal Engine. Dans chacune de ces parties, nous retrouverons quelles sont les stratégies d'optimisation mises en place pour garantir des performances optimales, soumises aux limitations des casques de réalité virtuelle.

2. Présentation de l'entreprise et de l'environnement de travail

2.1 Un projet européen : LIFE V-air

LIFE V-air est un projet regroupant plusieurs partenaires européens, visant à sensibiliser aux enjeux de la qualité de l'air. Ce projet fait partie du programme LIFE de l'Union européenne qui soutient des projets liés à la préservation de l'environnement. Il a vu le jour après un constat : des mesures pour éviter une trop grande pollution de l'air sont mises en place, mais à cause d'un manque de sensibilisation, celles-ci ne sont pas ou mal appliquées.

LIFE V-air a pour but de sensibiliser environ 1 400 décideurs politiques et 12 000 citoyens entre 2024, début du déploiement, et 2026. L'objectif est d'engager le dialogue avec les publics à travers deux jeux en réalité virtuelle (VR).

Pour atteindre ces objectifs, deux jeux distincts sont donc en cours de développement et de déploiement. Le premier, un serious game, est destiné aux décideurs politiques. Ce jeu se joue à deux : une personne avec le casque de VR et une autre sans casque, mais avec un manuel technique imprimé. Le joueur en VR est transporté dans une station de contrôle de la qualité de l'air dans le futur où l'environnement est gravement dégradé par la pollution de l'air et le changement climatique. Les deux joueurs coopèrent pour mettre en œuvre, dans la station, des actions adaptées à leurs territoires afin de prévenir cette situation. Le second jeu, une escape game, également en VR, est destiné au grand public. Les citoyens sont également projetés dans une partie de la station. Ils évoluent dans plusieurs pièces afin de résoudre des énigmes. Les joueurs doivent récupérer une clef pour améliorer la qualité de l'air de chacune des pièces de vie et pour retrouver un environnement sain au plus vite, avant d'être intoxiqué.

Pour le serious game, les décideurs politiques ont la possibilité de choisir le territoire dans lequel ils veulent jouer. Ils peuvent donc retrouver des données locales sur la qualité de l'air et le climat, fournies par les Associations Agréées de Surveillance de la Qualité de l'Air (AASQA) et des partenaires européens.

Ces jeux sont prévus pour être d'une durée relativement courte, mais permettant de créer un dialogue avec les joueurs. Pour les décideurs politiques, il facilitera les discussions avec les AASQA, permettant la mise en place de conseils et de plans d'amélioration de la qualité de l'air dans les zones concernées. Les citoyens seront eux sensibilisés et découvriront des moyens de réduire leur impact sur la qualité de l'air intérieur, à travers la déconstruction d'idées reçues.

2.2 Un partenaire-employeur et un partenaire d'accueil

Parmi tous les partenaires de LIFE V-air, on retrouve notamment l'Université de Bourgogne-Franche-Comté (UBFC) et ATMO GRAND EST. L'UBFC était mon employeur tandis qu'Atmo Grand Est m'a accueilli dans ses locaux durant les six mois de stage. Ces deux partenaires jouent un rôle différent dans le projet :

L'UBFC : L'Université de Bourgogne-Franche-Comté, dont l'UTBM fait partie et est l'entité représentant l'UBFC dans le projet. En tant que partenaire officiel du projet LIFE V-aiR, l'UTBM est responsable de la production des deux jeux en réalité virtuelle (design, développements et tests).

Atmo Grand Est : Atmo Grand Est est une AASQA, du groupe Atmo France, présente à Strasbourg, Metz, Nancy et Reims. Cette organisation est impliquée dans la surveillance de la qualité de l'air, la communication, et dans l'accompagnement des acteurs régionaux dans la mise en place d'actions.

Les missions principales d'Atmo Grand Est sont :

- Surveiller la qualité de l'air et assurer la conformité aux réglementations en place.
- Accompagner les initiatives locales.
- Communiquer pour informer et sensibiliser les citoyens, les médias et les autorités.
- Évaluer les plans, programmes et actions de dépollution.

En tant que stagiaire chargée du développement des jeux, l'UTBM m'a employé sous la supervision de Monsieur Franck Gechter. Cependant, mon lieu de travail était le site d'Atmo Grand Est à Schiltigheim, près de Strasbourg. Sur le site d'Atmo Grand Est, j'étais encadrée par Alexandre DESBOS, chargé de développer les deux jeux.

Le fait de réunir le « pôle développement » sur un site unique a grandement facilité les tâches de conception et développement, car Atmo Grand Est, en tant que partenaire porteur du projet, est un acteur central et nécessitait donc une communication régulière pour assurer la cohérence du développement des jeux avec les attentes des partenaires et le cahier des charges. Cela a également permis d'organiser fréquemment des phases de test et de recueillir des retours, que ce soit lors d'évènements ou directement parmi les employés.

Au sein d'Atmo Grand Est, les interactions se faisaient principalement et quotidiennement avec l'unité internationale (INT), dédiée à la collaboration transfrontalière avec l'Allemagne, ainsi qu'à des projets européens et internationaux plus larges.

2.3 Les autres partenaires et les dynamiques

Aux côtés de l'UTBM et d'Atmo Grand Est, nous retrouvons d'autres partenaires français et européens :

- Plusieurs AASQA : ATMO Bourgogne Franche-Comté, ATMO Hauts-de-France, ATMO Nouvelle Aquitaine, ATMO Normandie, MADININAIR (Martinique) ;
- Une autre université : l'Université de Malaga en Espagne ;
- L'agence roumaine de l'énergie : ABMEE ;
- Des partenaires co-financeurs et associés représentant des élus : Région Grand Est, La Rochelle, Fédération ATMO France...

Le projet est structuré en huit « Work Packages » (WP), chacun définissant une méthodologie globale spécifique et dirigé par l'un des partenaires :

1. **Management du projet :**
Ce WP est dirigé par Atmo Grand Est. Il englobe toutes les activités liées à la gestion globale du projet : la planification, le suivi des progrès, la coordination entre les différents partenaires, la gestion des ressources...
2. **Préparation des données d'entrée pour les jeux :**
Sous la responsabilité d'Atmo Grand Est, ce WP consiste à collecter et préparer les données réelles des territoires, nécessaires pour le développement des jeux.
3. **Écriture et conception des jeux :**
Piloté par Atmo Hauts-de-France, ce WP implique la création des scénarios et des concepts des jeux.
4. **Production des jeux :**
L'UBFC-UTBM a la charge de ce WP. Il couvre la phase de développement des jeux, allant du design graphique et technique à la programmation, en passant par les phases de test pour garantir la qualité et la jouabilité des jeux.
5. **Déploiement auprès des décideurs et des citoyens :**
Atmo Normandie pilote ce WP, qui se concentre sur la diffusion des jeux auprès des publics cibles.
6. **Suivi et évaluation du projet :**
Ce WP, également piloté par Atmo Grand Est, vise à mesurer l'impact et l'efficacité du projet. Les activités incluent la collecte de feedback, l'analyse des données d'utilisation et l'évaluation des résultats par rapport aux objectifs initiaux.
7. **RéPLICABILITÉ ET TRANSFÉRABILITÉ DES JEUX :**
Sous la responsabilité d'Atmo Grand Est, ce WP se concentre sur la documentation et la mise en place des conditions nécessaires pour que les jeux puissent être adaptés dans d'autres contextes ou régions.
8. **Communication, diffusion et mise en réseau :**
Piloté par MADININAIR, ce WP inclut toutes les activités de communication et de diffusion des résultats du projet.

2.4 Organisation et suivi du projet

Bien que chaque partenaire du projet soit associé à un ou plusieurs WP, les échanges entre les différents acteurs sont très réguliers. Il est courant de collaborer sur un même sujet, surtout au niveau du développement. La collaboration se fait également dans le sens inverse : en ayant plus d'expérience avec le monde de la réalité virtuelle et du développement de jeu, nous pouvions aider les partenaires, par exemple au niveau des stratégies de communication. Par conséquent, il fallait s'assurer de maintenir un contact régulier avec les partenaires.

2.4.1 Réunions inter-AASQA

- Les comités exécutifs mensuels

Pour soutenir cette communication et coordination, chaque mois, le comité exécutif se réunit, rassemblant des représentants de tous les partenaires du projet. Lors de ces réunions, les avancements de chaque Work Package sont présentés : réalisations, évolutions, futur travail, prochaines dates d'échéance... Ces réunions permettent de maintenir une vue d'ensemble cohérente et de s'assurer que tous les partenaires sont alignés et informés des progrès et des problèmes du projet. Du côté du WP-4, l'avancée du travail sur le serious game et l'escape game été présentée : nouvelles mises à jour, avancées, contraintes, plans futurs...

- Les réunions avec partenaires - groupes de travail

En plus d'une division large par WP, dirigé par un partenaire désigné, des groupes de travail sont créés à l'intérieur de ces divisions. Des réunions étaient donc organisées, afin de travailler sur des tâches précises. J'ai principalement pris part aux réunions de deux groupes de travail :

W - T.4.2 : Développement de 2 versions fonctionnelles : Le groupe W – T.4.2, le projet prévoit le développement de deux versions fonctionnelles distinctes (serious game et escape game). Chaque version sera conçue pour répondre aux besoins principaux (énigmes jouables, scénario pour les partenaires disponibles...). Le développement de ces versions inclut une phase de test pour identifier et corriger les éventuels bugs, ainsi que pour valider les choix de design et d'interaction. Pour ce groupe, nous travaillons principalement avec les responsables du projet, membres de l'unité INT, qui conduisent la plupart des tests et nous font des retours des citoyens et décideurs politiques, afin d'améliorer l'accessibilité et l'expérience utilisateur.

WP – T.8.1 : Communication auprès du public : Le groupe WP – T.8.1 se concentre sur la communication auprès du public. Pour cela, nous nous réunissions avec plusieurs partenaires (principalement l'ABMEE, MADININAIR, l'Université de Malaga et Atmo Normandie). Nous travaillons sur plusieurs points :

- Une vidéo d'introduction du serious game, en début de partie ;
- La répartition des informations sur la qualité de l'air à mettre dans le document technique et leurs formatages pour les énigmes ;
- Le style graphique global du document technique (couleurs, police...).

- Les réunions hebdomadaires

Toutes les semaines ou toutes les deux semaines, une réunion est tenue entre développeurs afin de discuter de mes avancements. Ces réunions permettent de définir les prochaines tâches mais d'avoir également un retour sur le travail que j'ai effectué depuis la dernière réunion. Les missions effectuées seront détaillées en 3. Etat de l'art.

2.4.2 Les autres actions de communication

En plus de ces réunions récurrentes, nous avions des événements exceptionnels, que ce soit directement en lien avec l'organisation du projet ou lors d'actions de déploiement.

- **Un déploiement à l'Eurométropole de Strasbourg en avril**

Pour toute la durée du projet, LIFE V-air doit sensibiliser 1 400 décideurs politiques et 12 000 citoyens. La majorité des campagnes de sensibilisation se fera une fois les jeux terminés, quand ils seront disponibles dans leur version complète en ligne, et ce jusqu'en 2026. Mais le décompte se fait déjà depuis le début de la période de développement. Des événements divers sont organisés, comme des sensibilisations en lycée ou collèges, des salons comme la fête de la Science à Belfort/Montbéliard, ou des événements privés dans des entreprises.

Courant avril, j'ai eu l'occasion de participer à un déploiement à l'Eurométropole de Strasbourg (EMS). Les employés pouvaient participer à plusieurs ateliers de sensibilisation aux questions environnementales, et notamment tester le serious game. Selon leurs fonctions dans l'EMS, les joueurs étaient donc décomptés comme décideur ou comme citoyen.

Durant la journée, nous devions donc faire jouer en binôme les employés qui se présentaient, en leur donnant des indications si nécessaire mais, pour le développement, il était tout de même important de les laisser trouver la solution aux énigmes par eux-mêmes. De cette manière, nous pouvions réellement voir les qualités et défauts de ce qui était déjà développé : manipulations en VR difficiles, énigmes simples, document technique non-clair...

En plus de ces retours, chacun des participants devait remplir un questionnaire concernant les points suivants : le joueur et sa fonction, sa satisfaction sur le côté visuel du jeu et sa satisfaction sur le côté éducatif. Les réponses étaient ensuite traitées par le WP-6 : suivi et évaluation du projet.

- **Un audit en mai**

J'ai également été chargée de préparer et faire la présentation du Work Package 4, celui du développement, lors du dernier audit de LIFE V-air durant le mois de juin. Cette présentation était particulièrement importante car le projet accusait un retard considérable par rapport au planning initial, érigé en 2022.

En effet, une version du serious game dans son état final aurait dû être livrée en fin d'année 2023. Pour l'escape game, une version fonctionnelle du jeu aurait dû être également livrée en décembre 2023, tandis que la version finale aurait dû être disponible en ligne en juin 2024. Cependant, ces dates n'ont pas été respectées car bien trop optimistes. Pendant 1 an,

le développeur était seul sur un projet où il devait maîtriser toutes les compétences pour créer un jeu : conception, développement, environnement 3D... Les délais imaginés au lancement du projet n'étaient pas réalistes de la réelle production d'un jeu, considérant la complexité demandée par les partenaires et la taille de l'équipe.

Durant l'audit, mon rôle a donc été de mettre en avant les étapes déjà réalisées pour chacun des jeux, d'expliquer les raisons de ce retard tout en montrant que la suite du développement était contrôlée, afin de rassurer l'expert sur le bon fonctionnement du projet. Cela s'est bien passé, avec des retours positifs de l'expert, qui a précisé que, malgré le retard, le développement semblait bien être sous contrôle et avançait vers de bons horizons.

2.4.3 Environnement de travail

- Environnement matériel

Le projet Life V-Air est développé sous Mac OS. Pour ce projet, les casques de réalité virtuelle Occulus Quest 2 et Meta Quest 3 ont été choisis. Ces casques disposent de leur propre système d'exploitation, sont légers et autonomes (il n'y a pas besoin de PC pour le faire fonctionner). Ils proposent donc plusieurs avantages dans le cadre d'un projet comme Life, où les déploiements sur évènements sont courants et où de nombreuses personnes jouent en même temps avec chacune leur casque :

- **Coût abordable** : Environ 300 euros pour le Quest 2 et 500 euros pour le Quest 3, achetés reconditionnés s'il y a la possibilité.
- **Autonomie** : Facile à installer pour des sessions de test dans des salons ou des lieux sans Internet ni prises de courant.
- **Bon rapport qualité/prix** : Offre de bonnes performances pour leur prix.
- **Retour haptique**

Le Quest 3, sorti en 2023, est plus puissant et possède une meilleure qualité graphique par rapport au Quest 2, tout en gardant les avantages d'autonomie et en ayant sensiblement le même poids (entre 510 et 520 grammes pour les deux casques). Cependant, la polyvalence de ces casques a un inconvénient. En étant des casques autonomes, leurs processeurs graphiques limitent la qualité visuelle des rendus. Durant le développement, nous avons dû parfois surmonter ces contraintes matérielles et adapter nos scènes 3D, notamment causées par les limites de puissance du Quest 2.

- Environnement logiciel

- Unreal Engine et IDE

Unreal Engine est un moteur de jeu développé par Epic Games, l'une des références en termes de logiciel dans l'industrie du jeu vidéo. Durant le stage, Unreal Engine a été l'outil principal pour le développement du jeu. Il permet de développer le gameplay, ou jouabilité, et la logique du jeu avec des Blueprint ou en C++. Pour développer les jeux avec les spécificités de la réalité virtuelle, nous utilisons le plugin OpenXR dans Unreal Engine. Ce plugin est spécialement conçu pour faciliter le développement en offrant des fonctions, objets et autres paramètres pour la réalité virtuelle.

En termes d'IDE, d'hébergement et gestion du code source, nous utilisons **JetBrains Rider** ainsi que **Github**. Rider, associé à Unreal Engine, facilite grandement la gestion des scripts et des assets, et possède des fonctionnalités de versionning intégrées qui simplifient le suivi des modifications et la résolution des conflits.

- **Blender**

Blender est un logiciel open-source de modélisation 3D, d'animation, de rendu, et de création 3D. Blender a été utilisé principalement pour la modélisation des éléments 3D du jeu (décor, machines...). L'optimisation des modèles 3D est importante pour assurer des performances élevées dans le jeu, vis-à-vis de la puissance de chacun des casques.

- **Substance Painter**

Substance Painter est un logiciel utilisé pour la création de textures pour objets 3D. Il supporte la création de textures PBR (Physically Based Rendering), ce qui permet d'avoir des textures simulant les interactions de la lumière sur l'objet 3D. Ce logiciel est également considéré comme une référence pour la création de textures.

- **Figma**

Figma est un outil de design d'interface utilisateur (UI) et d'expérience utilisateur (UX). Durant ce stage, nous utilisions Figma principalement pour la conception car de nombreux outils de collaboration sont à disposition : diagrammes de classe, use-case, tableau blanc... Certaines des textures plus simples (non-PBR) du jeu ont également été créées sur Figma, avec l'outil de conception d'interfaces.

- **Notion**

Notion est un outil de prise de notes, gestion de projet, que nous utilisons dans le projet pour plusieurs aspects :

- **Documentation et partage de ressources** : on retrouve sur l'espace notion les différents documents utilisés pour le développement du jeu, comme les documents de game design (GDD ou Game Design Document) du serious game, celui de l'escape game... Mais sont rédigés et stockés également les guides d'utilisation de différents outils, plutôt destinés aux autres partenaires.
- **Suivi des tâches et gestion de projet** : Notion offre des outils de gestion qui nous permet de suivre l'avancement des différentes tâches, de les assigner, de leur donner une date de complétion à respecter. Nous appliquons à notre échelle la méthodologie de gestion de projet Kanban ("à faire", "en cours", "terminé" pour chaque tâche), notamment lorsque nous travaillons sur le même sujet. Nous avons également une page qui permet aux partenaires qui testent les jeux de nous faire part de suggestions ou de bugs, à la manière d'un ticket informatique.

3 État de l'art du projet

3.1 Un Serious Game dans sa version fonctionnelle

En février, le serious game était déjà bien avancé, étant en développement depuis un an. La conception du jeu et des énigmes avait été réalisée par l'entreprise Serious Frames. Le document de conception (ou Game Design Document ; GDD) donnait les directives à suivre pour le développement sur plusieurs aspects : l'univers, les règles du jeu, l'environnement du joueur, le style graphique, et le gameplay.

3.1.1 Univers et règles du jeu

Le GDD avait évolué depuis le début du projet, intégrant les retours des utilisateurs et les nouvelles idées développées au cours de l'année. Dans sa version fonctionnelle, soit la période à laquelle j'ai commencé à prendre part au développement, les éléments clefs étaient les suivants :

- **Univers** : Le joueur VR incarne un technicien d'une station météo volante fictive de ATMO, lui permettant d'influencer et d'affecter tous les paramètres de la région survolée afin d'améliorer et d'impacter directement la qualité de l'air. À l'intérieur de la station, il se retrouve dans le dôme (figure 1) : environnement dans lequel se trouve plusieurs machines : machine résidentielle, machine agriculture, machine industrie, machine mobilité et machine énergie. Le décor mélange des objets aux inspirations naturelles et d'autres plus science-fiction.



Figure 1: Dôme de la station

➤ **Règles du jeu et Gameplay :** La pollution sur terre s'étant grandement accentuée, les deux techniciens doivent collaborer pour réparer les problèmes survenus sur chaque machine, et faire descendre le niveau de pollution. A l'aide de son partenaire, qui a le document technique entre les mains, le technicien VR doit donc résoudre les énigmes proposées pour chacun des secteurs.

- **Machine “Résidentiel”** : Le joueur se retrouve face à une maquette divisée en 3 secteurs : résidences, espaces verts, structures et services. Pour chacun des secteurs, 3 actions à appliquer lui sont proposées et il doit choisir la plus adaptée à la problématique présentée.
- **Machine “Agriculture”** : Sur une table de laboratoire, le joueur doit analyser plusieurs échantillons de productions de fermes du territoire et doit les classer comme respectueuses de l'environnement en fonction des pratiques de la ferme associée.
- **Machine “Industries”** : Le joueur utilise des modules représentant des actions industrielles pour réduire les émissions du territoire, tout en respectant un seuil défini.
- **Machine “Mobilité”** : Pour cette machine, le joueur se trouve devant un plan où différents chemins sont dessinés entre différentes zones. En fonction des moyens disponibles, le joueur doit choisir le moyen de transport le plus adapté entre ces deux zones.
- **Machine “Energie”** : Le joueur dispose d'une table où plusieurs moyens de productions d'énergie sont associés à un potentiomètre. Il doit trouver le mix énergétique qui produit assez pour le territoire tout en respectant la surface disponible et sans dépasser un seuil d'émissions polluantes.

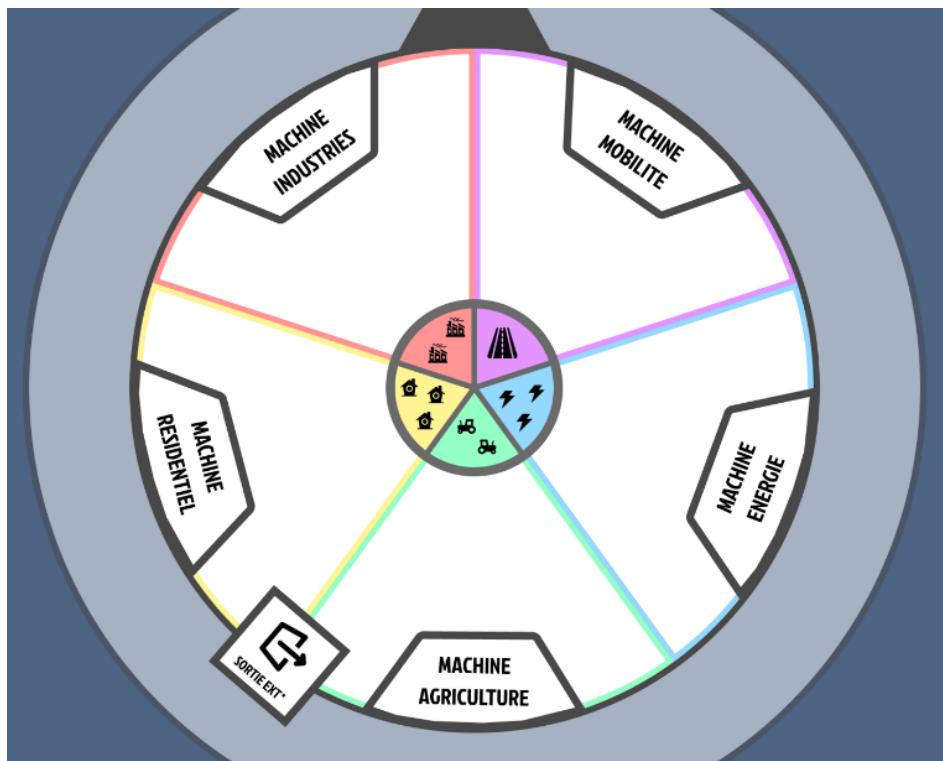


Figure 2: Plan de la station et des différents secteurs

Toutes les machines étaient donc jouables et de nombreux tests avaient été effectués. Certaines machines fonctionnaient déjà bien en production, comme la résidentielle ou la mobilité, mais d'autres demandaient de nouvelles itérations pour qu'elles soient bien adaptées à la population qui était visée par chaque partenaire (âge, lieu de vie...). Pour la machine agriculture, il était nécessaire de reprendre le modèle 3D et de l'adapter, tout en intégrant le système de scénario. Cela permet à chaque partenaire d'avoir une expérience de jeu adaptée à son territoire en choisissant les types d'échantillon de production. Pour la machine énergie, beaucoup étaient à reprendre : le modèle n'était pas satisfaisant car il manquait de personnalité, résoudre l'énigme était trop difficile pour les joueurs et non-intuitive. De plus, il y avait du travail à réaliser du côté de l'architecture du code.

- [La salle de tutoriel](#)

En plus des machines à résoudre, qui constituent la grande partie du jeu, une salle de tutoriel a été rajouté quelque temps avant mon arrivée (figure 5). En effet, le besoin s'était fait ressentir sur les salons, car des personnes non-initierées à la VR ou aux jeux vidéo avaient du mal à progresser dans le monde virtuel. Le but de cette salle est donc de découvrir les différentes manipulations en VR. Pour les personnes habituées, cette étape passe rapidement, tandis que pour les autres, elle permet de s'acclimater à un environnement virtuel et aux différentes manipulations. Voici les actions à maîtriser :

- Appuyer sur un bouton
- Scanner un objet
- Attraper un objet
- Pousser un levier

Les joueurs ne sont pas catapultés dans le dôme principal avant d'avoir réussi chacune de ces manipulations au moins une fois.



Figure 3: Salle de tutoriel

3.1.2 Architecture du projet

Le projet est open source et disponible pour tous sur GitHub. Il est structuré avec plusieurs branches pour gérer les différentes étapes du développement. La branche principale, nommée *SeriousGameBeta*, est utilisée pour la version la plus stable du projet. Elle contient les dernières modifications validées et testées qui sont prêtes à être testées par les partenaires. Autrement, chaque fonctionnalité majeure dispose de sa propre branche, comme *DecoUpdate* pour la décoration ou *EnergyMachineRework* pour la machine à énergie.

En termes de structure des fichiers, le projet est organisé en différentes catégories :

- **Assets** : Regroupent les images, modèles 3D, textures et autres ressources dans une sous-arborescence avec un fichier pour chacune des machines ou élément spécifique (tableau de bord, salle de tutoriel...).
- **Core** : Regroupe les fichiers essentiels au gameplay et à la logique du jeu. Cette section est également subdivisée par machine ou élément pour garder la même organisation.
- **FX** : Inclut les effets spéciaux visuels tels que les particules, les animations et autres éléments visuels.
- **Maps** : Les **maps** dans Unreal Engine sont des niveaux ou des scènes où sont placés les objets et éléments interactifs du jeu. Le projet comprend trois maps principales : les menus du début, la salle de tutoriel, et le dôme de la station.
- **UI** : Contient les éléments de l'interface utilisateur, tels que les boutons, les panneaux d'information et les autres composants interactifs.
- **Localization** : Regroupe les fichiers nécessaires à la traduction et à l'adaptation du jeu dans différentes langues. En effet, nous devons traduire le jeu dans les langues des partenaires, soit allemand, roumain, et espagnol mais également en anglais.
-

3.2 Les prémisses de l'Escape Game

En février, aucun document de game design formel n'avait été élaboré pour l'Escape Game. Cependant, quelques consignes simples avaient été définies pour guider la création du jeu. L'objectif était de concevoir un jeu d'une durée de 20 minutes, accessible au grand public. Il devait pouvoir être joué à domicile ou de manière autonome sur salon, avec un casque Oculus 2 ou 3 mais sans document technique ou deuxième joueur. L'idée de cette deuxième version est de toucher un plus large public : des citoyens lambdas et non uniquement des décideurs politiques.

Le thème central de l'Escape Game est l'environnement, avec pour but de sensibiliser les joueurs aux enjeux écologiques tout en offrant une expérience ludique et engageante. Ces directives servaient de base pour le document de conception, qui devait détailler les éléments spécifiques du jeu tels que les scénarios, les énigmes, les mécanismes de jeu, et les interfaces utilisateur. L'objectif est de créer une expérience éducative, intégrant des éléments interactifs qui encourageraient les joueurs à réfléchir pour résoudre les énigmes liées à différentes thématiques environnementales.

4. Travail réalisé

Durant ce stage, j'ai été impliquée dans divers aspects de la création d'un jeu vidéo, me permettant de développer un profil polyvalent. J'ai participé à la conception d'un jeu complet et à la refonte d'énigmes déjà existantes, à la création d'environnements 3D et au développement de la logique de jeu. Mon travail comprenait également la vulgarisation de concepts techniques lors de réunions que j'animaïs, souvent dans un contexte multilingue. J'ai aussi contribué à la communication du projet, en participant aux diverses réunions de groupes de travail, comme sur la création des pages du document technique ou d'une vidéo d'introduction au jeu.

Mon rôle principal était donc de développer le jeu, tant sur le plan de la logique que de l'environnement, tout en donnant de mes connaissances sur les sujets connexes au développement mais tout aussi important pour le projet. Étant donné du nombre de tâches différentes auxquelles j'ai pu prendre part au cours de ces six mois, il serait difficile d'en faire un résumé exhaustif. Seuls les principaux sujets sur lesquels j'ai travaillé seront abordés. Ce sont en particulier les sujets qui m'ont permis de faire progresser fortement mes connaissances sur la durée du stage.

4.1 Planning des tâches

- Planning prévisionnel

En plus des comptes-rendus hebdomadaires, dont découlaient des mini-planning, et en fonction des demandes des partenaires, un planning mois par mois a été dressé et était aménagé en fonction de mon avancement. La figure 4 ci-dessous résume les grandes catégories de tâches que j'avais à traiter, de février à juillet.

Activité	Février	Mars	Avril	Mai	Juin	Juillet
Prise en main des outils						
Machine Agriculture : refonte visuelle et ajout des scénarios						
Machine Energie : refonte visuelle et de la logique de jeu						
Création de l'environnement						
Design de l'escape Game						
Développement de l'escape Game						

Figure 4:Planning prévisionnel des tâches mois par mois

- Planning réel

Cependant, le développement n'était pas tout lisse : il y avait parfois de grosses difficultés comme des changements soudains de dates d'échéances ou d'autres imprévus. Je devais parfois travailler sur plusieurs catégories de tâches à la fois pour pouvoir régler les problèmes dans les temps. Le planning initialement prévu a donc été modifié au cours du stage. Le tableau 5 ci-dessous est donc représentatif du travail réalisé mois par mois.

Activité	Février	Mars	Avril	Mai	Juin	Juillet
Prise en main des outils						
Machine Agriculture : refonte visuelle et ajout des scénarios						
Machine Energie : refonte visuelle et de la logique de jeu						
Création de l'environnement						
Design de l'escape Game						
Développement de l'escape Game						

Figure 5: Planning réel des tâches mois par mois

4.2 La conception de l'escape game : Sensibilisation à la qualité de l'air pour les citoyens

4.2.1 Les fondations du jeu

Une partie importante de mon travail a été dédiée à la conception des jeux, et plus particulièrement à celle de l'escape game. En effet, en amont de la production des jeux, une équipe se doit d'écrire un document de Game Design, qui sert de guide pour le développement. Il contient toutes les informations nécessaires pour différents aspects : mécaniques de gameplay, univers, scénario, niveaux, aspects visuels et sonores, contraintes techniques... Cette phase de conception est essentielle pour structurer le travail et ne pas se lancer aveuglément dans le développement. Afin de pouvoir compléter la rédaction de ce document, nous avons tenu plusieurs réunions avec les partenaires et déterminé les caractéristiques principales en fonction des besoins et des attentes.

L'objectif principal de l'escape game est de sensibiliser le grand public aux problématiques de qualité de l'air. Le jeu doit couvrir les différents polluants que le joueur peut rencontrer dans la vie réelle, leurs impacts sur la santé et l'environnement, ainsi que les sources de cette pollution. Nous souhaitons particulièrement focaliser l'attention des utilisateurs sur la manière dont ces polluants les affectent directement et quelles actions quotidiennes, qui peuvent sembler anodines, peuvent émettre des polluants tels que l'oxyde d'azote (NOx), les particules fines (PM) et les composés organiques volatils (COV), ou des gaz à effet de serre comme le CO2.

Le jeu est destiné au grand public, comprenant divers types de personnes qui ne sont pas nécessairement familières avec les jeux vidéo ou la technologie en général. Grâce aux retours d'expérience des précédents déploiements du serious game, nous avons pu remarquer que des actions en VR qui nous paraissent simples, en tant que développeurs, sont loin d'être intuitives pour beaucoup de joueurs. Pour garantir que ces joueurs puissent interagir facilement avec le jeu et en tirer un maximum d'apprentissage, nous avons donc choisi de nous concentrer sur des interactions très basiques : attraper un objet, appuyer sur un bouton, se déplacer en marchant ou avec la téléportation activée, si le joueur se sent suffisamment à l'aise. En plus de cette contrainte, nous devions concevoir le jeu autour de plusieurs autres contraintes ayant été décidées durant des réunions avec les partenaires :

- Une session de jeu ne doit pas durer plus qu'une vingtaine de minutes ;
- Les utilisateurs doivent pouvoir jouer de manière autonome ;
- Pour qu'une partie sans téléportation se déroule bien, la zone de navigation doit être de 2m x 3m ;
- Le joueur doit pouvoir recevoir des indices, comme dans une escape game réelle ;
- Le jeu doit offrir de la "rejouabilité".

Pour la question de la rejouabilité, nous avons décidé de diviser une session en plusieurs pièces à traverser pour atteindre l'objectif, qui est de dépolluer l'intérieur de la station. Le joueur devra en premier finir la pièce sur les impacts des polluants pour accéder à la pièce suivante. Le nombre minimum de pièces est de 2, comprenant la pièce d'introduction. En fonction des paramètres du jeu sélectionnés dans le menu principal, le nombre de pièces peut être augmenté ou diminué. Cette expérience modulaire permet également de choisir la longueur de la session de jeu et nous permet d'avoir un jeu adapté au déploiement lors d'évènements. En effet, chez lui, un utilisateur pourra choisir le nombre de pièces, et par conséquent la durée de sa session de jeu. Cependant, lors de présentations du projet sur des

salons ou dans un cadre privé, il est préférable de garder une expérience de jeu relativement courte, soit environ 20 minutes. Pour adapter au mieux l'expérience lors des évènements organisés par les partenaires, ces derniers pourront choisir le mode « partenaire », qui permet donc de modeler l'expérience en fonction du public en face.

Chaque pièce est donc conçue avec l'idée que le joueur passe environ dix minutes à l'intérieur. Chacune est composée d'énigmes à résoudre qui sont basées sur des idées reçues que des citoyens peuvent avoir. Notre première idée était de traiter un polluant par salle : salle NOX, salle PM... Finalement, choisir de structurer les pièces par thème d'idées reçues plutôt que par polluant nous a paru plus pertinent. Premièrement, cela permet de mettre en lumière les liens entre les polluants et les sources et d'éviter la redondance. C'est également plus réaliste et concret des problématiques de l'environnement. Nous avons donc dressé une liste de ces idées reçues puis regroupées, ce qui nous a permis de dégager des thèmes et donc des pièces associées aux thèmes :

Type de pièce	But de la pièce
“Laboratoire”	Introduction aux polluants et leurs impacts sur la santé
“Origine de la pollution”	Démontrer que chaque secteur (agriculture, industrie, chauffage...) contribue à la pollution en émettant des polluants spécifiques.
“Pollution de l'air intérieur”	Idées reçues sur les encens, bougies, huiles essentielles et la ventilation

Figure 6: Type de pièce et sujet traité

Un autre point sur lequel il fallait trouver une solution est la question de la taille des pièces. En effet, lors d'évènements, il est difficile de délimiter une zone de jeu plus grande que 2 mètres par 3 mètres. De plus, se téléporter ou se déplacer avec un objet dans les mains était peu intuitif pour beaucoup de joueurs. Nous voulions éviter que le joueur soit forcé d'utiliser la téléportation afin de simplifier les mécaniques de jeu. Cette contrainte nous a donc conforté dans notre choix de création de multiples petites pièces car cela nous permettait de créer de petites zones de jeu avec tous les éléments interactifs suffisamment proches entre eux pour pouvoir se passer de la téléportation. Ce jeu se passant dans le même univers que le serious game, soit dans une station volante, nous avons donc décidé que les pièces dans lesquelles l'utilisateur évoluerait seraient les pièces de vie des techniciens de la station.

Le **level design**, ou conception de niveaux, est une étape importante de la création d'un jeu, car il consiste à concevoir l'agencement des niveaux de manière à offrir une expérience claire pour le joueur. Cela inclut par exemple la disposition des éléments dans le jeu, mais aussi la manière et l'ordre dans lequel les joueurs interagissent avec ces éléments pour progresser. Dans notre projet, la conception des différentes pièces et leur disposition relève donc du game design mais également du level design, où nous cherchons à guider le joueur intuitivement pour qu'il apprenne et évolue dans l'espace par lui-même. Dans les 3 salles que nous avons conçues, nous avons appliqué les principes de conception de niveaux, comme le fait de faire ressentir à un joueur sa progression dans le niveau.

4.2.2 La salle laboratoire

La première pièce dans laquelle le joueur évoluera est donc la pièce laboratoire. Cette salle a été conçue pour adresser des idées reçues courantes concernant la pollution de l'air. L'objectif est de corriger des idées erronées telles que : le CO₂ est un polluant atmosphérique, tous les polluants ont des effets similaires sur la santé. Cette pièce introduit le joueur aux différents types de polluants présents dans le jeu et démontre leurs impacts spécifiques sur l'environnement et la santé humaine.

Un point clef du level design est de créer le premier niveau comme un tutoriel, afin de familiariser l'utilisateurs aux concepts et manipulations spécifiques au jeu. Cette salle sert donc de tutoriel pédagogique pour initier les joueurs aux différents polluants que l'on peut retrouver et l'impact sur notre santé, mais également de tutoriel pour les manipulations en VR. Ces dernières seront limitées au déplacement dans une salle virtuelle, à la prise en main d'objets interactifs et à leur insertion dans des zones prévues à cet effet.



Figure 7: pièce de laboratoire en cours de développement

4.2.3 La salle “Origine de la pollution”

Pour décrire le processus complet de conception d'une salle, nous allons nous concentrer sur la pièce "**Origine de la pollution**", car moins axée sur le tutoriel (elle offre une expérience de jeu plus poussée) et que j'ai pu prendre la direction de la conception.

Parmi les partenaires ayant testé le jeu, il est apparu que la pièce qui leur semblait à développer en priorité était celle qui traitait des questions des origines naturelles ou anthropiques de la pollution. Lors de précédentes discussions, nous en avions tiré les idées reçues suivantes, sur lesquels nous devions ensuite nous baser afin de concevoir une expérience à la fois éducative et ludique :

- La pollution de l'air est entièrement causée par les voitures.
- La pollution n'est due qu'aux activités humaines.
- La pollution de l'air est surtout due aux industries.
- La pollution ne provient que des combustibles fossiles.
- Brûler des déchets verts ne pollue pas.

- Objectifs Pédagogiques de la Salle

Le but de cette pièce est de faire comprendre aux joueurs que la pollution de l'air est causée à la fois par des activités humaines et des phénomènes naturels. Un autre objectif est d'aider le joueur à réaliser l'impact de différents secteurs ou moyens de transport par rapport à d'autres sources de pollution. En effet, chacun de ces secteurs est loin d'avoir le même poids et ils n'influencent pas de la même manière la concentration de chaque polluant.

- Comment la Pièce a-t-elle été Conçue ?

L'idée était de faire en sorte que le joueur puisse comprendre intuitivement où commencer et comment progresser dans la pièce, tout en étant confronté aux idées reçues sur la pollution. Plutôt que de simplement présenter ces idées de manière directe, ce qui n'aurait pas incité les joueurs à réfléchir, je voulais intégrer des puzzles qui introduisent implicitement les réponses aux idées reçues. C'est pourquoi les puzzles conçus présentent plutôt subtilement des concepts et permettent de vérifier la compréhension du joueur sur ces concepts à la fin de la pièce, où il devra répondre à des questions qui permettent de démontrer les idées reçues. Les puzzles et les éléments interactifs ont été conçus non seulement pour divertir, mais surtout pour renforcer le côté éducatif de l'escape game.

- Mécanique de jeu et conception des puzzles

L'un des puzzles principaux de la salle est un Open Path Puzzle¹, basé sur une balance à deux plateaux, où le joueur doit comparer des secteurs de pollution (comme l'industrie, les voitures, avions, les déchets verts, etc.) représentés sous forme de pièces de puzzle, et trouver l'équilibre grâce la balance en fonction de chacun des polluants (COV, NOx, PM). Il pourra mettre par exemple 4 pièces de puzzle représentant le secteur aérien et chercher combien de pièces de puzzle représentant le secteur automobile il doit mettre afin d'équilibrer la balance et ainsi de comparer les émissions des deux secteurs pour un polluant donné. Ce puzzle vise à montrer que différents secteurs ont des impacts variés sur les émissions de divers polluants et qu'il n'y a pas un unique secteur responsable de toute la pollution de l'air.

Une fois ce puzzle résolu, le joueur doit transférer les informations obtenues à une télévision cassée, en reconstituant des relations d'équilibre à l'aide de pièces manquantes. La télévision affichera ensuite une série de questions basées sur les idées reçues que le joueur vient de déconstruire, ce qui permet de vérifier qu'il a bien compris les concepts abordés.

Enfin, si le joueur répond correctement, comme pour chacune des pièces de la station, il obtient le filtre qu'il doit insérer dans l'un des tuyaux de la salle, où la pollution se trouve, pour "dépolluer" et finaliser la salle.

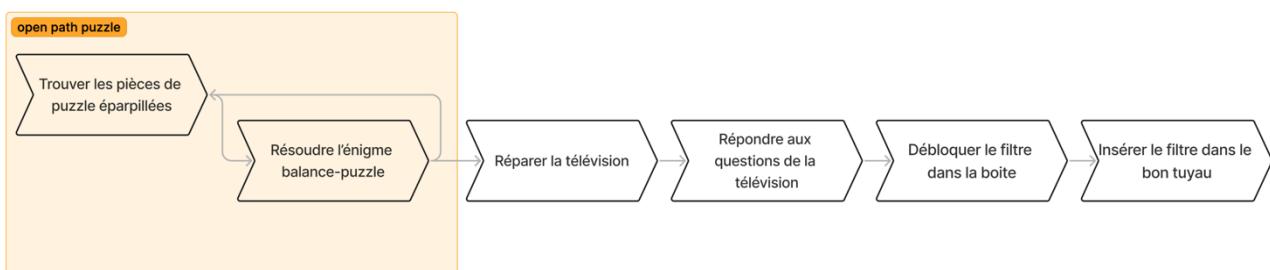


Figure 8: Flow chart des puzzles de la pièce

¹ Puzzles qui peuvent se faire dans n'importe quel ordre mais, pour passer au puzzle suivant, tous doivent être complétés.

- Comment concevoir l'espace de jeu ?

Pour que le joueur comprenne intuitivement ce qu'il doit faire, l'espace de jeu a été conçu pour guider ses actions sans nécessiter des instructions explicites. La perception est l'un des principes fondamentaux pour la conception d'expériences en VR. La vue semble être le premier sens à exploiter pour guider le joueur, suivie de l'ouïe et du toucher. Plutôt que d'utiliser des messages s'affichant dans le casque, nous avons donc opté par un jeu de lumière, de sons et retours haptiques afin de guider le joueur dans l'espace.

Ainsi, dès l'entrée dans la salle, une lumière ciblée éclaire la télévision et le boîtier, accompagnée d'un son émanant de l'appareil, attirant naturellement l'attention du joueur. À mesure qu'il progresse, de nouvelles lumières s'éclaireront pour le guider vers des éléments clefs, comme la table avec la balance. Afin d'avoir un retour sur ses actions, il est également prévu de rajouter des vibrations haptiques pour chaque interaction importante, comme le placement correct d'une pièce de puzzle, ou à l'approche d'un élément interactif.

Si des informations supplémentaires sont nécessaires, elles seront intégrées sous forme d'interfaces diégétiques. Cela signifie que les informations apparaîtront directement dans l'univers du jeu, par exemple sur des écrans ou des objets présents dans la salle, plutôt que sous forme d'instructions flottantes. Le personnage qu'incarne le joueur peut autant voir les informations que le joueur lui-même. Cette approche permet au joueur de naviguer et d'interagir dans l'univers virtuel en restant immergé dans l'expérience, sans être distrait par des éléments extérieurs.

Sur la figure 9 ci-dessous, nous pouvons voir la disposition des meubles des éléments interactifs. Ces derniers sont placés de manière à ce que le joueur puisse découvrir l'espace qui lui est ouvert, sans pour autant devoir faire beaucoup de trajets entre les manipulations.

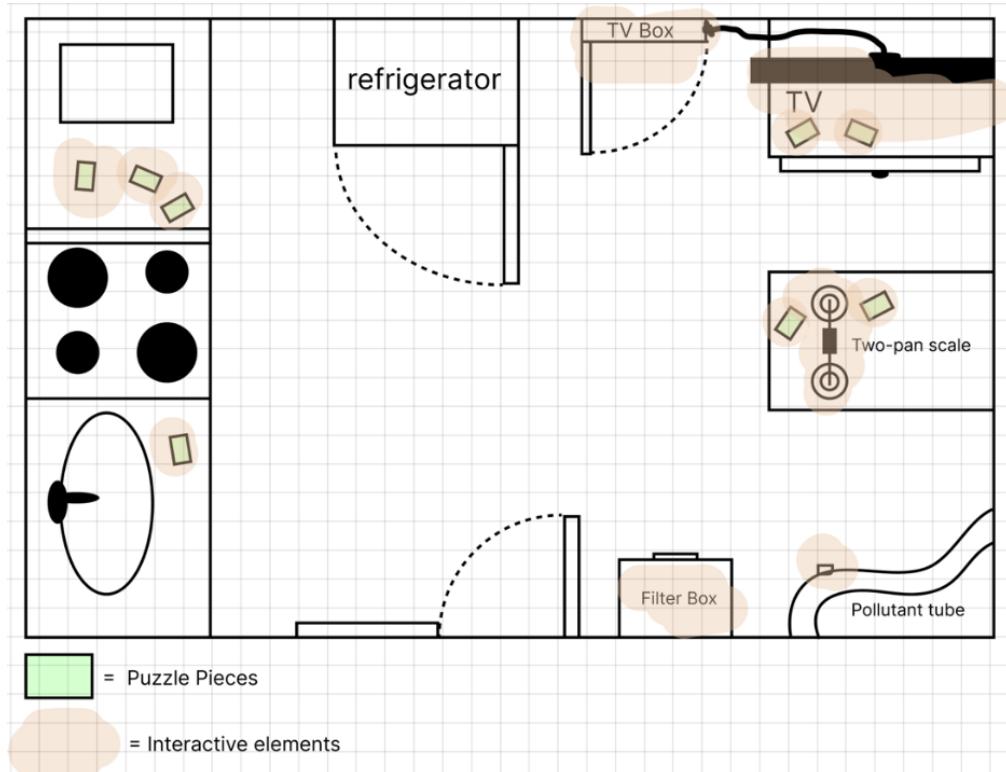


Figure 9: Plan de la pièce "Origine de la Pollution"

4.3 Le serious game : la machine énergie

Comme précisé auparavant et contrairement à l'escape game, la conception du serious game avait déjà été faite ainsi que l'implémentation d'une version fonctionnelle de toutes les machines. Les premières étapes du stage concernaient la reprise de deux machines (agriculture et énergie) au niveau de la conception en blueprint et de leurs modèles.

4.3.1 Les blueprint dans unreal engine

Blueprint est un système de visual scripting intégré dans Unreal Engine, permettant aux développeurs de créer une logique de jeu sans avoir besoin de coder en C++. Ils sont conçus pour être intuitifs et accessibles, même pour ceux qui n'ont pas de connaissances en programmation. Ce système offre des avantages mais également des inconvénients, qui font que son apprentissage peut être long :

- **Moins Performant que du code natif en C++** : Bien que les Blueprints soient puissants, ils sont moins performants que le code natif en C++. Cependant, nous n'avions pas besoin de réaliser des calculs complexes ou des rendus très poussés pour notre projet. Pour des productions plus importantes, le C++ peut être plus utilisé, mais dans notre cas, le gameplay et une grande partie de la logique du jeu pouvaient être développés grâce aux Blueprints.
- **Visual Scripting** : L'un des grands avantages des Blueprints est qu'ils permettent une visualisation claire de la logique du jeu. On peut voir l'ensemble du flux de travail, ce qui facilite la compréhension et le prototypage rapide. Cela dit, il peut être difficile à prendre en main pour beaucoup, car il faut connaître les noeuds à utiliser et la logique peut différer de la programmation traditionnelle. Cela implique une courbe d'apprentissage assez importante, d'autant plus que le système est très complet et il y a toujours de nouvelles fonctionnalités à découvrir. J'ai en effet eu du mal à me faire à cette logique et offre de possibilités durant la première moitié du sage. Les solutions d'implémentations ne me venaient pas naturellement et j'avais du mal à être autonome. Cependant, une fois les difficultés surpassées, on peut se rendre compte de la force d'un langage visuel.

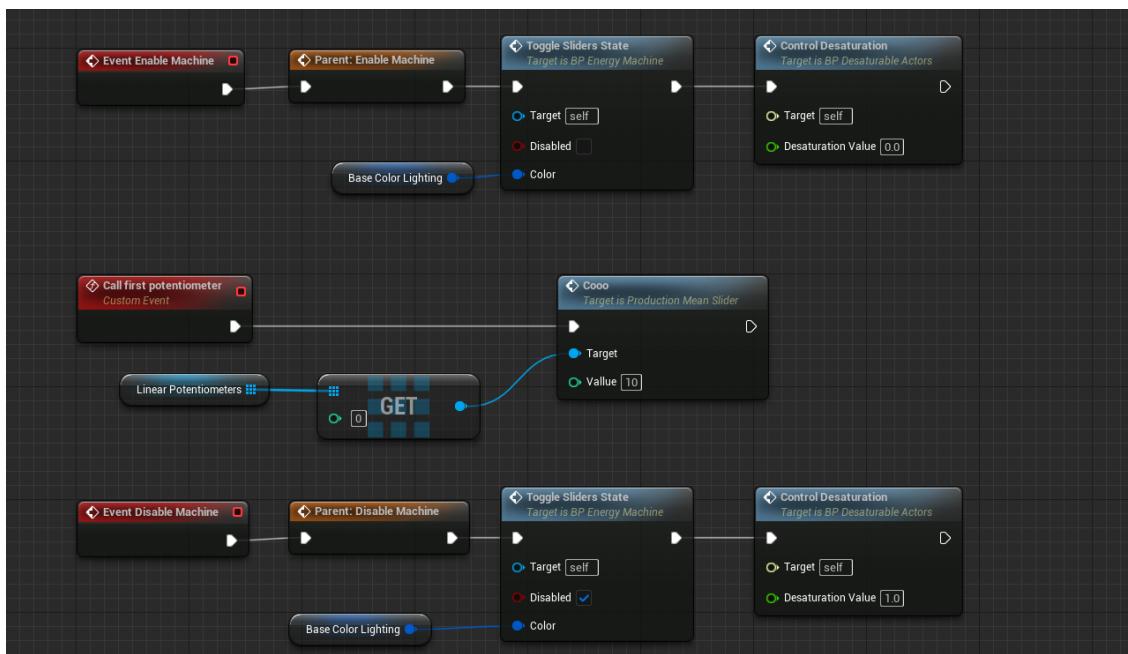


Figure 10: Évènement dans le Blueprint Machine énergie

4.3.2 Énergie : Fonctionnement de la machine avant modifications

Avant les modifications, la machine énergie était déjà à sa troisième version (figure 11). Dans cette dernière, le joueur avait devant lui une table avec 12 curseurs, chacun associé à un moyen de production d'énergie, tels que les éoliennes, la biomasse, ou le biogaz. L'objectif était d'atteindre une demande énergétique spécifique, par exemple 120 GWhs. Pour atteindre cette demande, le joueur ajustait chaque curseur afin de réguler la production d'énergie associée. Par exemple, en réglant le curseur de l'énergie éolienne sur 12, il produisait 12 GWhs.

Cependant, le joueur devait respecter des contraintes. S'il dépassait un seuil d'émissions polluantes, la machine échouait. De plus, il devait respecter la surface disponible sur le territoire choisi, pour les installations énergétiques. Si cette surface était dépassée, la machine ne pouvait pas être validée. Une fois la demande énergétique atteinte tout en respectant les contraintes d'émissions et de surface, les joueurs pouvaient valider la machine.

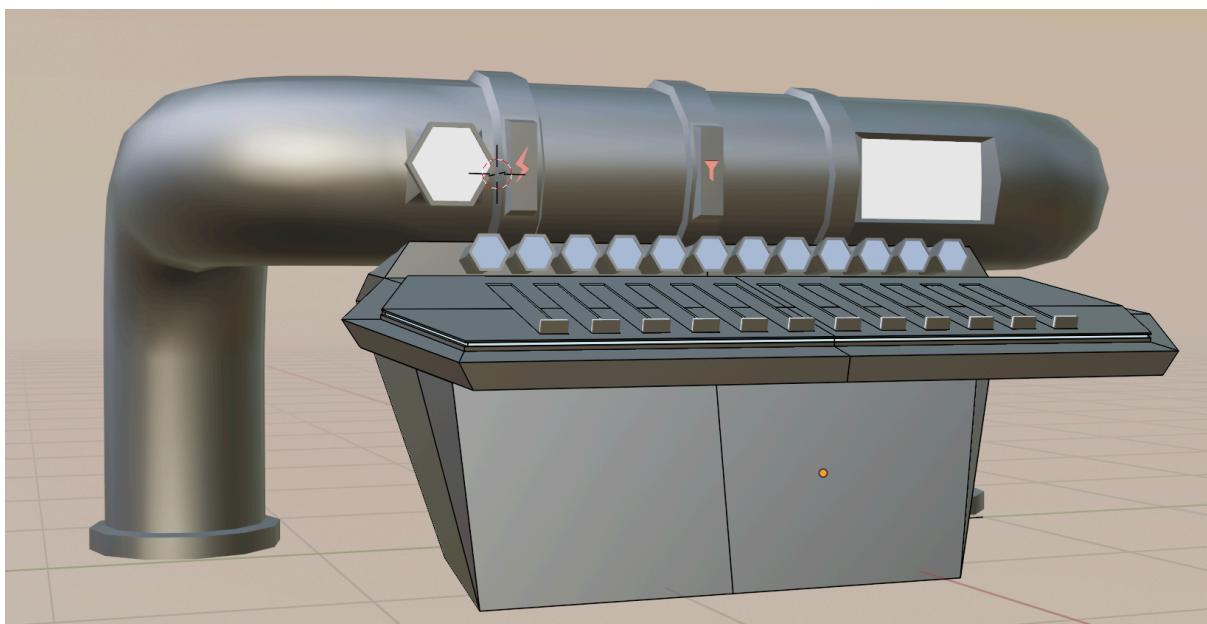


Figure 11: Troisième version de la machine énergie

En ce qui concerne la répartition des informations nécessaires à la résolution de l'énigme entre le joueur en VR et celui avec le document, le document technique fournissait les données telles que la surface occupée par chaque moyen de production ainsi que les émissions associées.

Après de nombreux retours utilisateurs et mes propres observations, plusieurs problèmes sont apparus :

- Échelle des curseurs non-uniforme : Chaque curseur avait une échelle différente, ce qui n'était pas clairement indiqué. Par exemple, pousser un curseur à fond pouvait produire 15 GWhs, tandis qu'un autre ne produisait que 8 GWhs. La valeur finale s'affichait mais les joueurs ne comprenaient pas toujours les différences d'échelle.
- Absence d'indications sur les émissions : Aucune indication claire sur les émissions polluantes à ne pas dépasser n'était fournie, obligeant les joueurs à deviner et tâtonner pour chaque curseur.

- Répartition inégale des informations : Le joueur avec le document technique disposait de toutes les informations de surface et d'émissions et décidait seul des ajustements nécessaires, tandis que le joueur en VR se contentait d'exécuter les instructions, limitant l'interaction et la coopération entre les deux.

Globalement, la machine n'était pas claire et il était compliqué pour les joueurs de trouver un équilibre énergétique correct, ce qui amenait plus de frustration que d'apprentissage.

4.3.3 La refonte de la machine

En reprenant cette machine, j'ai eu plusieurs points à mettre en place :

- Recréer le modèle de la machine pour une meilleure intégration dans l'environnement ;
- Après la re-conception de l'énigme, modifier la logique de jeu dans les Blueprint ;
- Ajouter le système de scénario qui permet aux partenaires d'avoir des données adaptées à leur territoire ;
- Recréer les pages du document technique, pour une meilleure répartition des données de jeu entre le joueur au document et celui au casque VR.

Refonte 3D

Au niveau de la refonte 3D de la machine énergie, il y avait peu à faire. Les retours négatifs concernaient surtout sa taille. En jeu, elle paraissait très grosse à la base et était moins harmonieuse que les autres machines. La figure 12 ci-dessous permet de voir les modifications apportées à la base ainsi qu'au tuyau, qui se branche cette fois-ci directement dans le sol de la station.

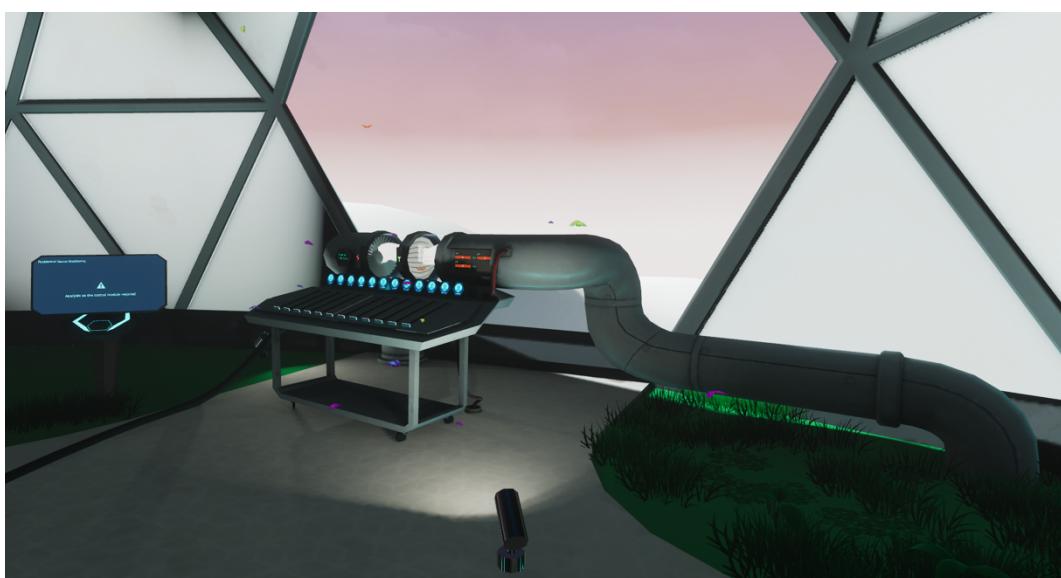


Figure 12: Version 4 de la machine énergie

Nouvelle logique de jeu et simplification des blueprints

Avec l'ancienne manière de résoudre l'énigme de la machine énergie, le joueur devait faire un mix énergétique mais, comme précisé précédemment, beaucoup d'utilisateurs avaient du mal à réussir ce mix car les consignes n'étaient pas assez précises. Il a été décidé, en accord avec les partenaires, de n'avoir qu'une seule solution correcte à chaque partie.

La nouvelle logique de l'énigme permet de simplifier les choix des joueurs, en proposant une solution unique concernant la valeur à associer à chaque moyen de production. Pour ce faire, les joueurs doivent résoudre des énigmes donnant un nombre unique (correspondants à la quantité d'énergie produite), compris entre 0 et 15 GWhs, pour les douze moyens de production. Les paramètres à prendre en compte pour ces derniers sont les suivants : acceptabilité des habitants du territoire, la surface disponible sur le territoire, l'impact sur le climat du moyen de production, celui sur l'air ainsi que son prix. Ces paramètres ont maintenant chacun une valeur fixe. Pour trouver la quantité d'énergie à fournir pour un type de production, les joueurs doivent résoudre un calcul formé à base des paramètres, leur donnant le total à reporter sur le curseur.

Cette nouvelle approche a grandement modifié la logique déjà implémentée dans le blueprint. Pour chacun des curseurs, il n'y a qu'un seul nombre qui peut être la solution, contrairement à la version précédente où n'importe quel mix pouvait valider la machine. Il suffit donc maintenant de regarder si tous les curseurs sont correctes et que les conditions limitantes sont valides (surface, acceptabilité...) pour réussir ou non l'énigme.

De plus, il y a une meilleure répartition des données entre les deux joueurs : dans l'équation donnée dans le document technique, certaines valeurs des paramètres sont manquantes, c'est au joueur en VR de les trouver grâce à un système de scan d'objet implémenté, qui permet d'obtenir des informations concernant l'objet scanné (entre autres un curseur).

La nouvelle structure d'un moyen de production est la suivante :

Structure	
Tooltip	
▶ Production goal	<input type="text"/> Integer ▾ <input type="text"/> ▾ <input type="button"/>
▶ isEnabled	<input type="text"/> Boolean ▾ <input type="text"/> ▾ <input type="button"/>
▶ isRenewableEnergy	<input type="text"/> Boolean ▾ <input type="text"/> ▾ <input type="button"/>
▶ Scan Value	<input type="text"/> Integer ▾ <input type="text"/> ▾ <input type="button"/>
▶ Scan Name	<input type="text"/> Text ▾ <input type="text"/> ▾ <input type="button"/>
▶ Symbol	<input type="text"/> Texture 2D ▾ <input type="text"/> ▾ <input type="button"/>
▶ RequiredSurface	<input type="text"/> Integer ▾ <input type="text"/> ▾ <input type="button"/>
▶ Public Acceptability	<input type="text"/> Integer ▾ <input type="text"/> ▾ <input type="button"/>

Figure 13: Structure d'un moyen de production

Implémentation du système de scénario

Comme mentionné dans une partie précédente, un système de scénario est mis en place pour chacune des machines. En effet, en fonction des territoires, les données ou exemples utilisés n'étaient pas forcément pertinent. Pour la machine agriculture, les types de cultures proposés au début (tomate, fraise, carotte et élevage de bœuf) était cohérent pour la France métropolitaine (en sa globalité et non par région ou département), mais moins pour la Roumanie et pas du tout pour la Martinique. Pour l'énergie, douze moyens de productions

étaient proposés au départ, sans distinction entre les territoires. Pourtant, peu de territoires peuvent se permettre d'avoir des installations hydroélectriques ou d'éoliennes en mer.

Il a donc fallu proposer une solution aux partenaires. Pour cela, nous avons implémenté des systèmes de scénario pour chaque machine. Pour cette machine énergie, la liste de tous les moyens de production sont disponibles. Un partenaire doit choisir dans cette liste les moyens de production d'énergie qu'il souhaite proposer aux joueurs, ainsi que la demande à atteindre et tout paramètre limitant (surface à atteindre, prix maximum pour l'installation...). C'est ensuite à nous d'adapter les valeurs données afin que les calculs à faire par les joueurs soient toujours justes. La figure 14 ci-dessous, montre la table de données d'un scénario, ici celui utilisé comme test. Tous les moyens de production étaient présents.

LittleCity	
LittleCity	
Energy Demand	110
Emission Threshold	0.0
Surface Available	50
Public Acceptability Total	80
Available production mean	12 Map elements
Roof Photovoltaic Panels	8 members
Wood energy	8 members
Hydroelectric power plant	8 members
Fossil fuels	8 members
Biomass	8 members
Geothermal energy	8 members
Biogas	8 members
Nuclear Power Plant	8 members
Micro Hydroelectric power plants	8 members
Wind Turbines	8 members
medium-depth geothermal power plant	8 members
Soil Photovoltaic Panels	8 members
% of renewable energy Total	25

Figure 14: Table de données d'un scénario de la machine énergie

Cette figure 15 montre quant à elle la structure utilisée pour construire notre datatable. Le type de la ligne « **Available production mean** » est une map avec en clef la liste des types de production et en valeurs la structure d'un scénario.

Structure		Default Values
Structure		
Tooltip		
Energy Demand	Integer	110
Emission Threshold	Float	0.0
Surface Available	Integer	50
Public Acceptability Total	Integer	80
Available production mean	E Energy Prod. S Energy Prod.	Map
% of renewable energy Total	Integer	25

Figure 15: Structure d'un scénario

L'héritage de classes Blueprint

Cette simplification de la logique se fait également par l'utilisation des évènements du blueprint parent, celui qui définit le fonctionnement commun à toutes les machines. Le blueprint de la machine énergie datait encore d'avant cet ajout ; il utilisait sa propre logique. Mais, en déplaçant cette logique vers le système d'évènements du blueprint parent, on a un code plus général qui respecte mieux les principes d'héritage et de polymorphisme.

Il faut toutefois faire des ajouts dans le blueprint énergie, afin d'avoir les spécificités de la machine dans la logique. Par exemple :

- **Enable Machine (activation de la machine lors du début de la partie)** : Activer les curseurs en illuminant uniquement ceux disponibles dans le scénario joué et afficher les valeurs spécifiques (total de production).
- **Disable Machine (désactivation de la machine)** : empêcher le joueur de bouger les curseurs, éteindre les lumières.
- **Machine Succes (validation d'une machine)** : Illuminer tous les curseurs disponibles dans le scénario en vert.

4.4 La création de l'environnement 3D

Une bonne part du développement d'un jeu passe également par la conception et la création l'environnement 3D. En début de stage, le serious game avait déjà bien avancé dans ce domaine : la station entière (coque, dôme, hélice) était modélisée et texturée, la plupart des machines avaient un modèle définitif et on retrouvait également le poste central qui sert de décors mais permet aussi d'afficher les informations réelles des territoires.

Le travail restant se concentrat donc sur la refonte des machines Agriculture et Energie, ainsi que sur l'ajout de détails dans le décor pour rendre la grande salle du dôme plus vivante et immersive. Ajouter des éléments de décors qui ne sont pas directement liés au gameplay mais qui ont comme seul but de rendre l'espace plus agréable est essentiel pour créer un environnement engageant pour le joueur. Cela permet de raconter également une histoire visuelle, en plus de celle que le joueur découvre en jouant.

Concernant l'escape game, nous sommes donc partis de zéro. Grâce au travail de game design, nous avons décidé que les salles dans lesquelles le joueur évoluerait seraient directement intégrées dans la station, sous le dôme. Garder le même univers pour les deux jeux nous permet de réutiliser certains modèles 3D et textures créés pour le serious game, ce qui facilite la modélisation et le texturage du reste de l'intérieur des salles. Le style graphique étant déjà établi, la direction à suivre était donc claire.

Les salles étant les lieux de vie de la station, il était donc facile d'imaginer un lieu par salle, et donc de pouvoir rapidement modéliser les parties les plus importantes. Par exemple, pour la cuisine, nous pouvions déjà modéliser, une table, des chaises... sans passer un temps énorme sur la conception pure de la salle. Cela nous permet d'avoir une base de l'environnement rapidement pour tester les éléments de gameplay développés.

4.4.1 Processus de modélisation et texturage

Pour la création 3D, je suivais un processus précis. Ce workflow², allant de Blender pour la création des modèles, à Substance Painter pour la création des textures, jusqu'à l'intégration finale dans Unreal Engine, permet d'avoir des maillages et textures optimisées afin d'alléger la taille de la scène 3D. Pour comprendre ce processus, nous allons ici suivre la création de la machine énergie, dont je devais faire la refonte (comme expliqué dans la partie précédente).

Modélisation dans Blender

La première étape du processus de création des maillages est la modélisation 3D. Ici, nous devons nous assurer que le modèle est optimisé, car les performances des casques VR ne sont pas très bonnes. Pour le Quest 2, il est conseillé d'avoir entre 750000 et 1 million de triangles, tandis que pour le Quest 3, c'est entre 1.3 million et 1.8 million. Pour le confort des utilisateurs, il est souhaitable de garder une haute fréquence d'images par secondes. Nous devons donc minimiser le nombre de polygones (ou triangles lorsqu'importés dans Unreal

² « Workflow » ou Flux opérationnel : une suite de tâches qui doivent être réalisées dans un ordre spécifique.

Engine) sans sacrifier la qualité visuelle. Pour ce faire, deux versions de chaque modèle sont créées :

-High Poly : Un modèle « high-poly » est un modèle avec un nombre élevé de polygones qui permet d'ajouter des détails au modèle. Ce dernier n'est pas destiné à être utilisé directement dans le jeu, mais il sert de base pour générer des textures détaillées par la suite.

-Low Poly : Version simplifiée du modèle high poly, avec un nombre réduit de polygones. Ce modèle est celui qui sera importé dans Unreal Engine.

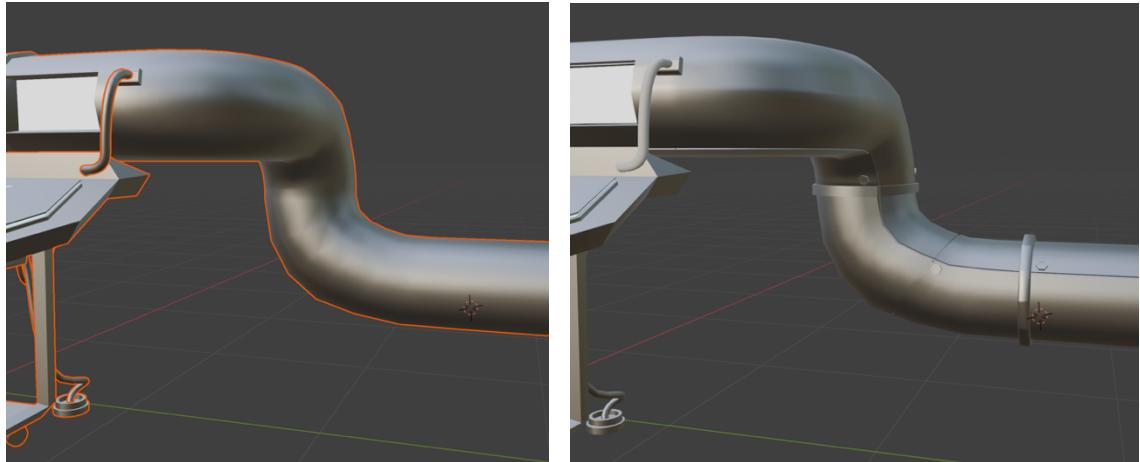


Figure 16 : modèle low-poly (à gauche) et high-poly (à droite) de la machine énergie

Il est également nécessaire d'éviter les n-gons (polygones à plus de quatre côtés), car ils peuvent causer des problèmes lors de l'exportation et de l'importation dans Unreal Engine, lors de la triangulation (passer des polygones aux triangles). Une fois importés dans le moteur de jeu, ils ne comportent pas bien et peuvent également causer des artefacts visuels avec les textures. Sur la figure 17 ci-dessous, on voit qu'une partie de l'ancienne machine énergie était composée de faces à 28 côtés. Sur la figure 18, les faces ont été modifiées pour obtenir uniquement des rectangles ou des triangles.

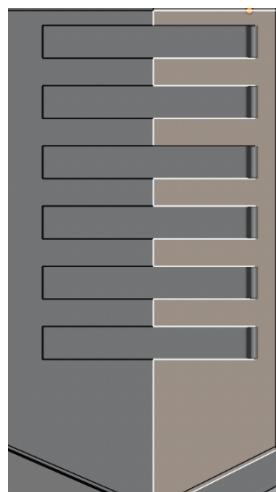


Figure 17 : Face à 28 côtés sur l'ancienne machine énergie

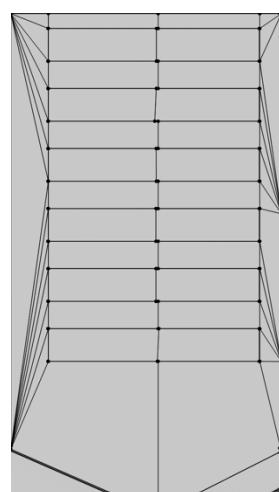


Figure 18: Quadrilatères et triangles sur la nouvelle machine énergie

En termes de topologie, il est aussi important de garder ce qu'on appelle une topologie en boucle, où les arêtes et les faces suivent les contours naturels du modèle. C'est essentiel pour des déformations lisses lors de l'animation. Il faut aussi garder une densité de polygones uniforme sur toute la surface du modèle et les garder carrés pour éviter les distorsions lors de l'application des textures et pour garantir une bonne subdivision si nécessaire.

De plus, il faut préparer des UV maps compactes pour chaque modèle. Une UV map (figure 19) est une projection 2D de la surface 3D du modèle. Cela nous donne une image 2D. Une UV map bien agencée permet de minimiser les déformations des textures et de stocker le maximum d'information dans le minimum de mémoire. Par exemple, des parties du modèle qui nécessitent des détails fins, comme le visage d'un personnage, doivent avoir plus d'espace UV alloué que des parties moins importantes, comme le dos.

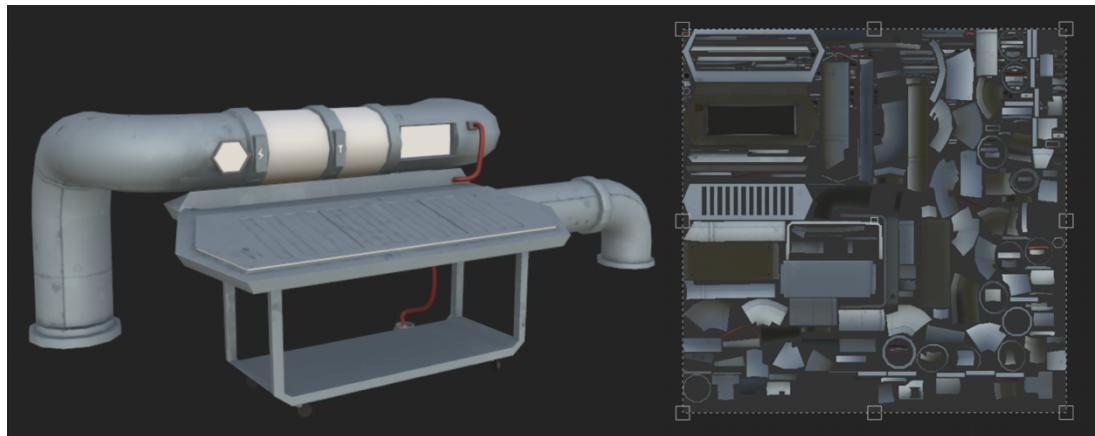


Figure 17 : UV map de la machine énergie

Dans certains cas, il est même préférable de regrouper plusieurs objets sur une seule texture, car cela est moins coûteux. Par exemple, pour des objets décoratifs, il n'est pas nécessaire d'avoir beaucoup de résolution car les détails ne seront pas visibles en jeux. On en regroupe donc plusieurs sur une seule image.

Texturage dans Substance Painter

Une fois les modèles et les UV maps prêts, le fichier du modèle est importé dans Substance Painter pour le texturage. Ce fichier contient les informations géométriques et l'UV map du modèle low-poly tandis que le modèle high poly est utilisé pour "**bake**" les textures. Cela signifie que l'on enregistre les informations du maillage 3D en un fichier de texture. Le baking permet donc de transférer les détails du modèle high poly sur le modèle low poly, ce qui donne l'illusion d'un modèle plus détaillé sans ajouter de polygones supplémentaires. Ce processus génère plusieurs types de maps PBR, qui permettent d'obtenir des matériaux réalistes et dont les ombres sont cohérentes avec les lumières :

- **Base Color** : Représente simplement la couleur de base du matériau.

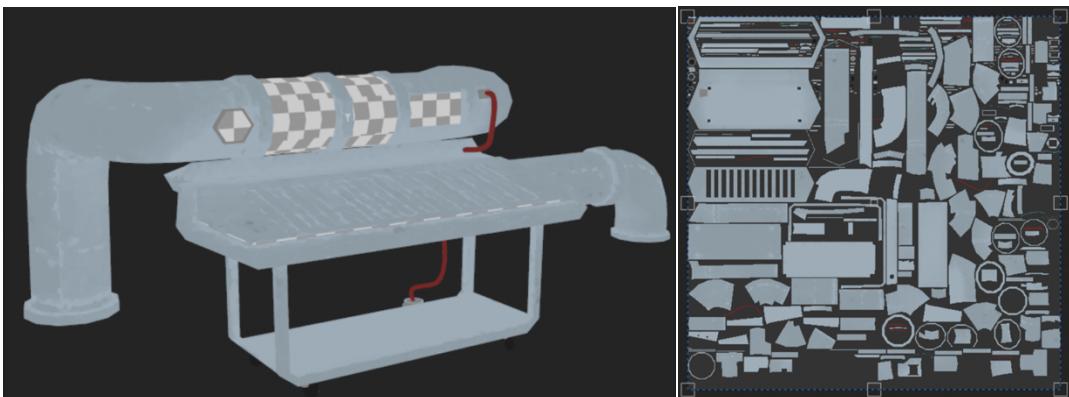


Figure 18 : Base color Map appliquée au modèle de la machine énergie

- **Normal Map** : Simule des détails de surface en créant des variations de lumière et d'ombre, ce qui donne l'illusion de complexité sans ajouter de géométrie supplémentaire.

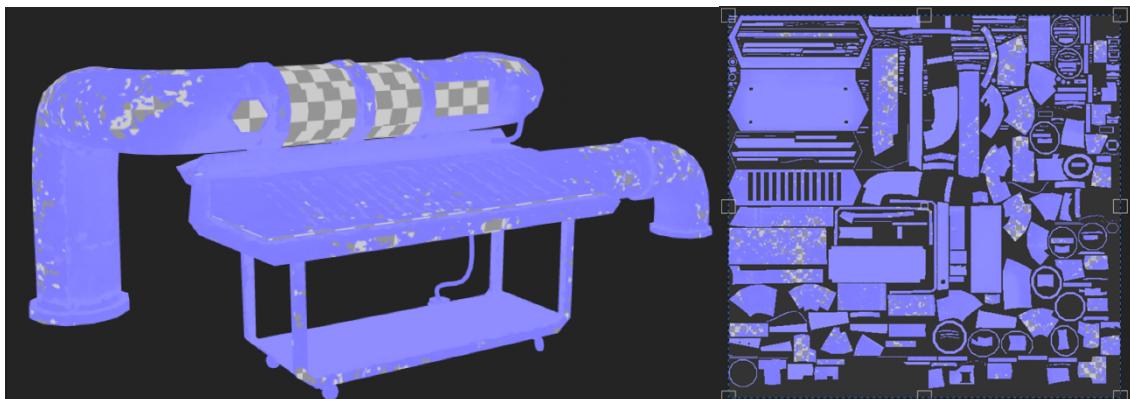


Figure 21 : Normal map appliquée au modèle de la machine énergie

- **Ambient Occlusion (AO)** : Ajoute des ombres douces dans les zones de contact et les recoins, ce qui améliore la perception de la profondeur et des détails. Lorsqu'un modèle n'a pas de version high-poly, il faut tout de même bake le modèle low-poly sur lui-même, car cela permet d'obtenir ces détails d'ambiant occlusion.

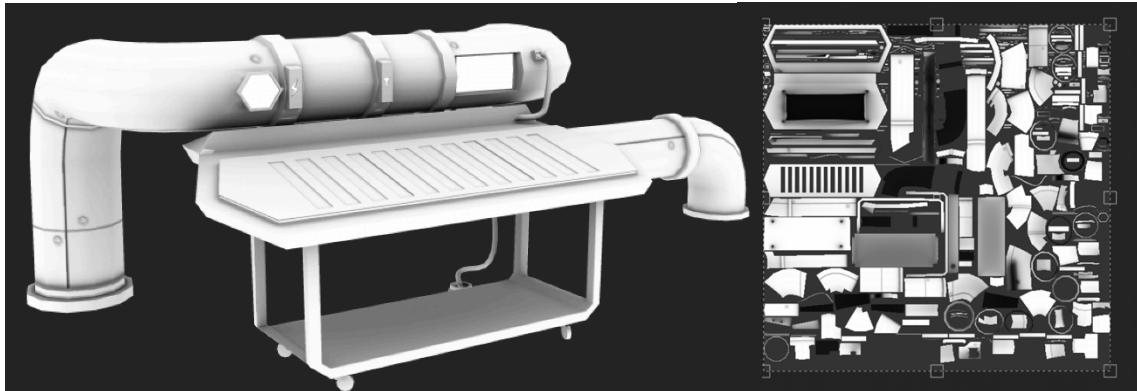


Figure 22 : AO map appliquée au modèle de la machine énergie

- **Roughness** : Détermine la rugosité de la surface, influençant la manière dont la lumière se reflète sur le matériau.
- **Metallic** : Définit quelles parties du modèle sont métalliques et lesquelles ne le sont pas, influençant également comment la lumière se reflète.

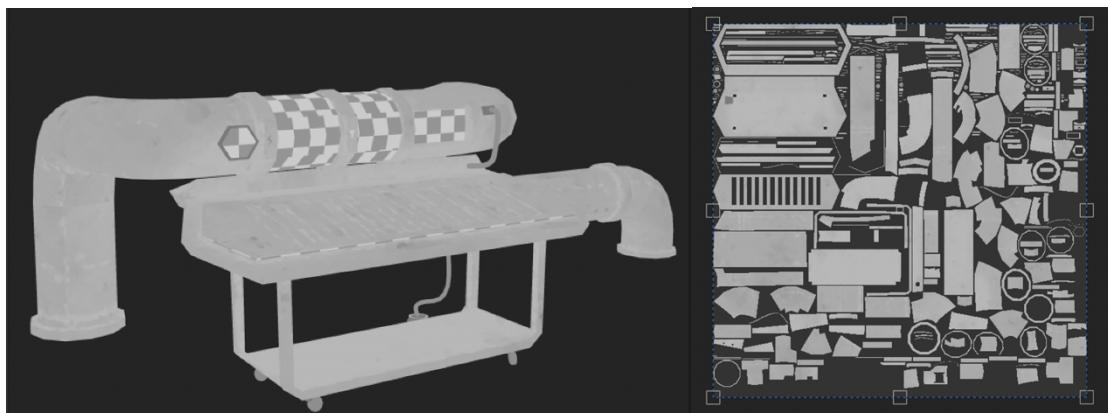


Figure 23 : Metallic & Roughness map appliquée au modèle de la machine énergie

Une fois la texture exportée, on ne se retrouve qu'avec trois images différentes : la base color map, la normal map et une image regroupant les informations de la metallic map, de la roughness map et de l'AO map.

Importation dans Unreal Engine

La dernière étape du workflow consiste à importer les modèles texturés dans Unreal Engine. Les différentes maps PBR générées dans Substance Painter sont importées et appliquées aux modèles low poly, importés en tant que fichier FBX. Lors de l'export du FBX depuis Blender, il faut s'assurer que les repères X, Y, et Z sont correctement configurés. En effet, Blender et Unreal Engine n'utilisent pas les mêmes axes pour représenter les coordonnées spatiales. Blender utilise un système où l'axe Z est l'axe vertical (haut/bas), tandis qu'Unreal Engine utilise l'axe Y pour cela.

Il est donc important de réduire individuellement la taille des fichiers des maillages en réduisant le nombre de polygones mais nous n'avons pas besoin de faire des modèles complètement minimalistes. En effet, l'optimisation des performances dans Unreal Engine repose en grande partie sur l'utilisation des Levels of Detail (LODs). Les LODs sont des versions simplifiées d'un modèle 3D, utilisées pour réduire la charge de rendu lorsque l'objet est vu de loin. En fonction de la taille de l'objet à l'écran, Unreal Engine peut générer automatiquement ces modèles simplifiés. Il nous est également possible de créer manuellement ces modèles.

Un objet qui occupe une petite portion de l'écran peut donc être rendu avec un modèle de faible résolution sans que cela soit réellement perceptible pour le joueur. En revanche, un objet proche et en plein écran nécessitera un modèle plus détaillé, même si simplifié, pour conserver sa qualité visuelle et que le joueur puisse toujours le reconnaître. En ajustant ces paramètres, la charge sur le processeur graphique est considérablement réduite. Les LODs donc particulièrement utilisés sur les machines car les modèles sont assez gros et leurs géométries sont plus complexes.

4.4.2 Cas particulier : la végétation

D'une certaine manière, la technique de regrouper les informations de textures de plusieurs objets sur une seule image est utilisée pour la végétation en jeu. Dans la station, la verdure est très présente : herbes et fleurs sur les parcelles de terre au sol, arbres, décos de machines, plantes suspendues, ect. De plus, certaines de ces plantes sont animées. Comme mentionné, lorsqu'un joueur valide l'une des machines du dôme, la végétation morte autour disparaît progressivement tandis qu'une nouvelle végétation pousse. Ces animations provoquent une surcharge soudaine sur le processeur graphique. Cette charge cause la chute du nombre d'images par secondes, causant un inconfort visuel et même des nausées à cause de la latence.

Pour limiter au maximum cette chute de performance, les visuels et textures des plantes sont regroupées sur une seule texture, comme illustré sur la figure 25, mais également en une texture « mask » (figure 24), qui, une fois importée dans Unreal Engine et associée à un matériau, permet de délimiter les contours en rendant la zone noire, transparente.

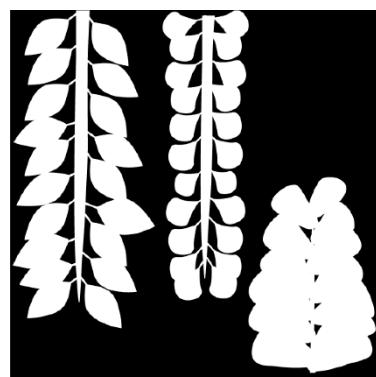


Figure 24 : Mask des plantes

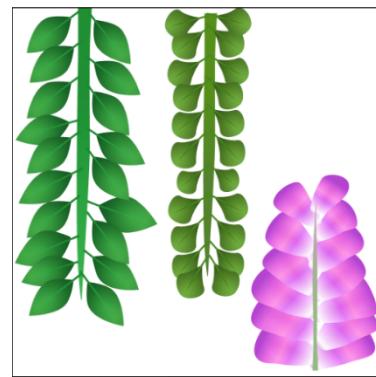


Figure 25 : Texture 2D des plantes

Le modèle 3D d'une plante n'est pas réellement modélisé mais il sert principalement de plan sur lequel appliquer les textures. Chaque plan peut être subdivisé ou non, si la végétation a une forme plus complexe à suivre qu'un rectangle. Lors de la création de l'UV Map, chacun des sommets du maillage doivent être placés manuellement afin de suivre les contours de notre texture, plutôt qu'en optimisant la place que chaque objet prend sur l'image.

La figure 26 montre l'exemple de l'UV Map d'un plan dans blender, avec l'image de texture en fond. Comme dit précédemment, le patron du plan est placé uniquement sur la végétation voulue.

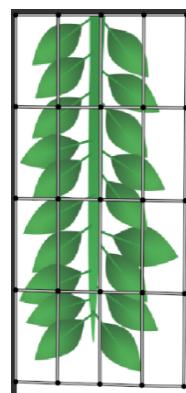


Figure 26 : UV Map de d'une branche sur la texture

Quant à la figure 27, elle permet de voir à quoi ressemble la plante grimpante entière dans Blender. Des plans ont donc simplement été placés et, grâce à l'UV map manuelle, ce n'est pas toute la texture avec les différentes plantes qui s'affiche mais uniquement la partie voulue. Lors du placement des sommets, il est préférable de minimiser la part de blanc sur la texture, soit de noir sur le mask et de transparence en jeu, car cette dernière augmente le coût de calcul des lumières en jeu.

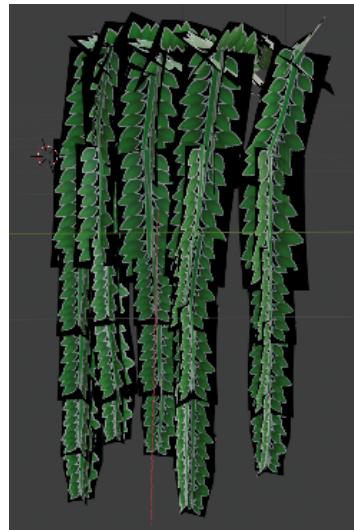


Figure 27 : Modèle de la plante grimpante vu dans blender

Une fois dans Unreal Engine, il faut créer un matériau qui traduit les informations des mask et des textures afin d'obtenir des plantes délimitées correctement et très peu couteuses (figure 28).

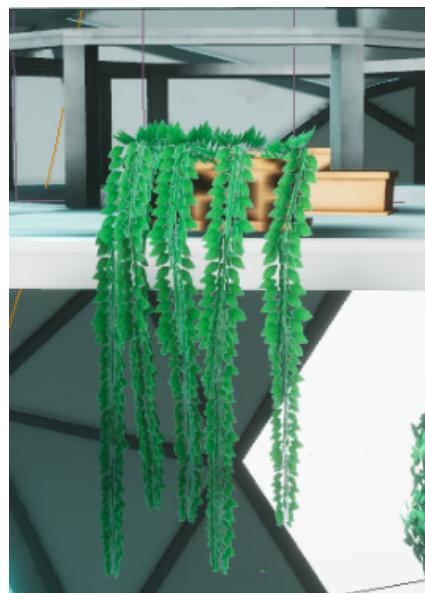


Figure 28 : Modèle de la plante grimpante vu dans Unreal Engine

4.5 L'ajout d'effets visuel

Afin d'enrichir l'environnement et l'immersion du joueur, le système Niagara d'Unreal Engine est utilisé pour créer des VFX ("Visual Effects" ou effets visuels). Niagara est conçu pour créer des particules, des simulations et autres effets en temps réel. En réalité virtuelle, ces retours sont particulièrement importants car ils renforcent la perception de l'environnement et facilitent l'interaction avec celui-ci.

À cet effet, j'ai pu créer quelques systèmes de particules, comme des papillons, ajoutant non seulement du dynamisme à la scène, mais aussi un repère visuel pour l'utilisateur, qui permet d'agrémenter la scène lorsque qu'on résout une machine. Ces particules s'ajoutent aux retours visuels déjà présents dans le jeu : lors de la résolution d'une énigme, l'environnement autour, qui était alors mort et terne, redevient vert. Les plantes mortes repoussent, les arbres retrouvent leurs couleurs et le ciel également.

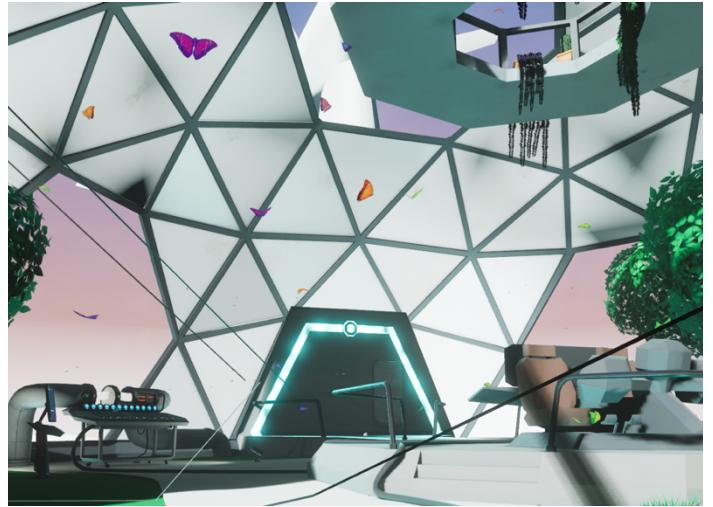


Figure 29 : Papillons dans le dôme

Tout d'abord, un système Niagara comporte quatre éléments essentiels (figure 30): les émetteurs, les systèmes, les modules et les paramètres. Les émetteurs génèrent des particules, les systèmes combinent plusieurs émetteurs, les modules définissent les comportements des particules et les paramètres permettent de contrôler plus précisément les propriétés et l'apparence des particules, comme leur taille, couleur, ou durée de vie.

Pour tenter de simuler les mouvements d'une nuée de papillons, nous avons rajouté plusieurs modules à l'émetteur. On retrouve par exemple le module **Curl Noise Force**, un champ de force qui permet de rajouter des mouvements de turbulence grâce à un champ de bruit. On a finalement un très grand contrôle sur le comportement au niveau des particules individuelles et du système.

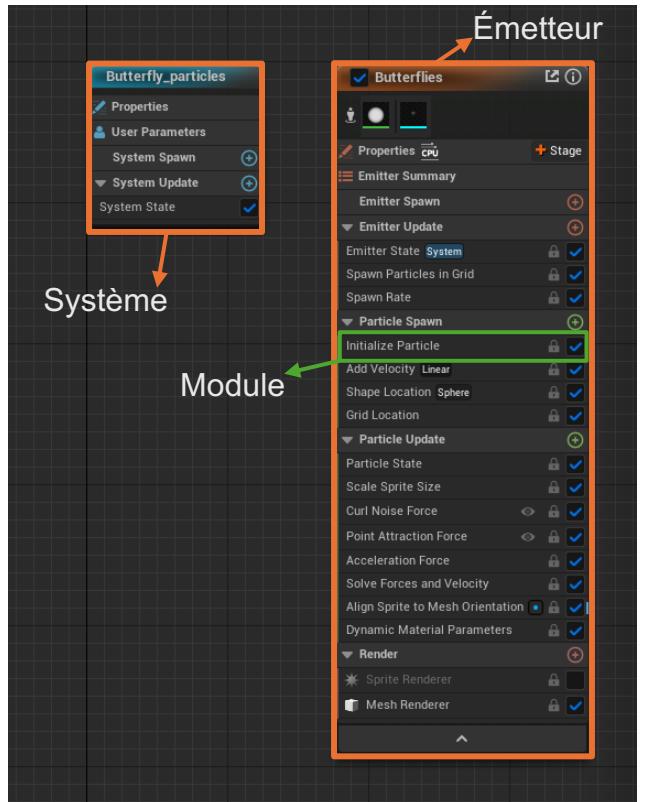


Figure 30 : Système de particules Niagara

Cependant, concernant les papillons, simplement ajouter un système de particule à la scène 3D ne suffit pas. En effet, le processus de création comporte plusieurs étapes, chacune utilisant des techniques particulières pour l'optimisation.

4.5.1 La création de maillages et textures pour un système de particules

Le workflow de la création du maillage est sensiblement le même que celui présenté précédemment : nous créons un modèle avec le moins de polygones possibles et nous préparons l'UV map afin de créer une texture par la suite. Cependant, pour un unique papillon, il n'était pas nécessaire de créer un véritable modèle en 3D. En effet, une fois intégré dans le jeu, le modèle sera petit, inséré dans une nuée d'autres papillons et ces derniers seront en constant mouvement. Un modèle détaillé et avec une épaisseur n'est pas utile car les subtilités de la géométrie ne pourraient pas être perçues par l'utilisateur : le maillage est simplement créé à partir d'un plan subdivisé dont les sommets ont été déplacés pour suivre les contours d'un papillon, comme l'on peut voir sur la figure 31 ci-dessous.

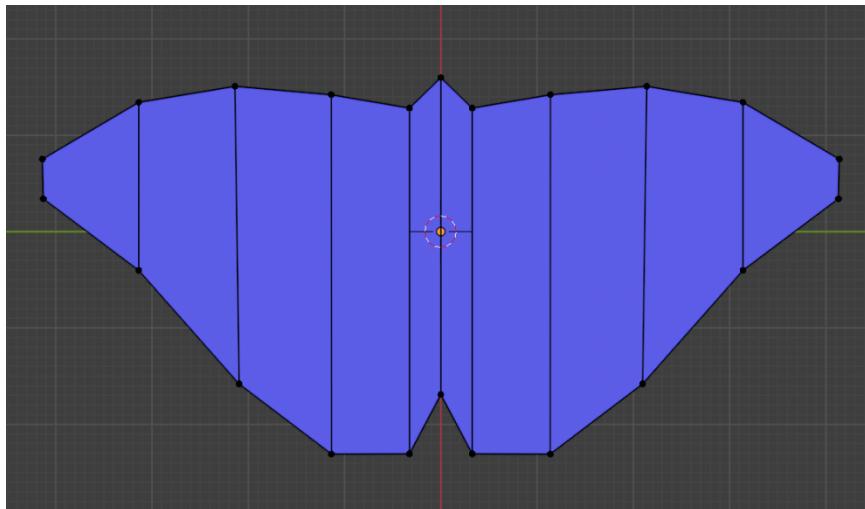


Figure 31 : Maillage du papillon

Pour la création de la texture 2D, nous avons adopté une méthode d'optimisation appelée **channel packing**. Cette technique consiste à tirer parti des trois canaux de couleur (R, G, B) d'une texture pour y stocker différentes informations nécessaires au rendu en temps réel. Plutôt que de créer plusieurs textures séparées pour chaque type de donnée, les informations sont combinées dans une seule texture, chaque canal étant dédié à un type de donnée spécifique. Une image est donc préparée pour chacun des trois canaux d'une texture. Les trois images représentent les informations suivantes :

- **Le canal vert (G)** contient une texture en noir et blanc. Cette texture sert principalement à définir les détails de la surface, comme la luminance ou des variations de teinte.
- **Le canal bleu (B)** contient un dégradé de couleurs. Ce dégradé est essentiel pour les animations des sommets (Vertex Animations), car il permet de contrôler les déplacements des points du maillage sans avoir besoin de squelette. Ce dégradé sera

donc utilisé pour animer le mouvement du battement d'ailes d'un papillon, en manipulant directement la position des sommets.

- **Le canal rouge (R)** est réservé à un **mask**. Ce masque détermine quelles parties de la texture sont affectées par certains effets ou animations.

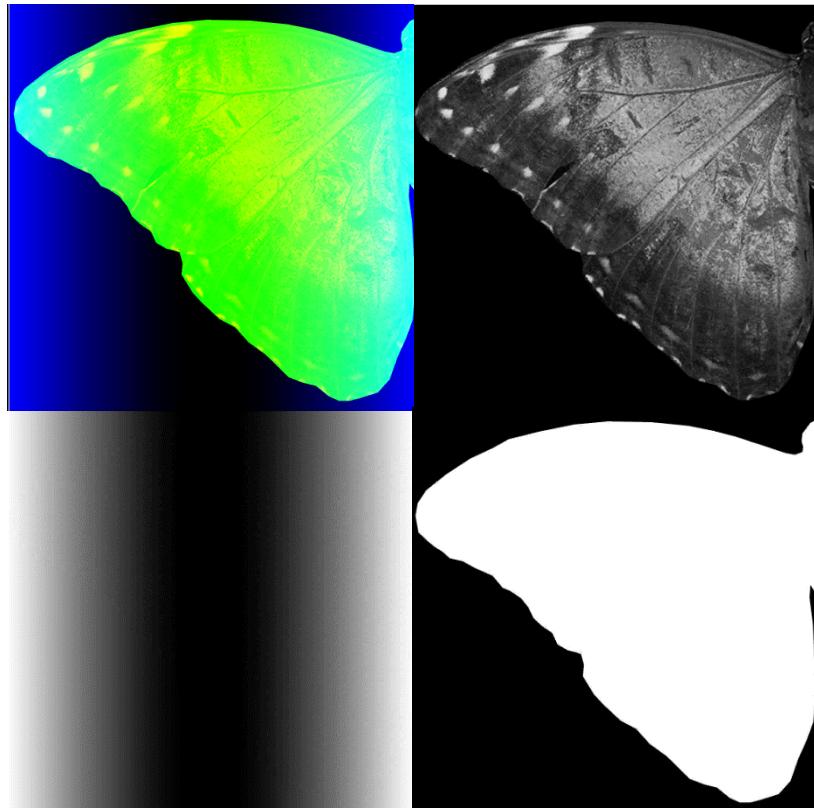


Figure 32 : De 1 à 4 - texture, canal G, canal B, canal R

Cette approche permet de réduire le nombre de textures nécessaires, ce qui diminue la charge de mémoire et améliore les performances du jeu, en particulier dans un système générant beaucoup de particules à la fois, et où les ressources graphiques doivent être gérées efficacement.

4.5.2 Shader et animations

L'étape suivante était donc d'utiliser cette texture dans notre shader conçu pour les papillons. Un shader est un programme exécuté sur le GPU pour déterminer l'apparence des pixels et des sommets sur l'écran (absorption et diffusion de la lumière, ombres, position, rotation...). Ici, ce shader déforme les sommets du maillage des ailes et ajuste la couleur des papillons en temps réel, durant toute la durée de vie de la particule. De la même manière qu'un Blueprint, un shader peut se créer uniquement en utilisant des nœuds. Ci-dessous, la figure 33 correspond au matériau, et donc au shader, des papillons.

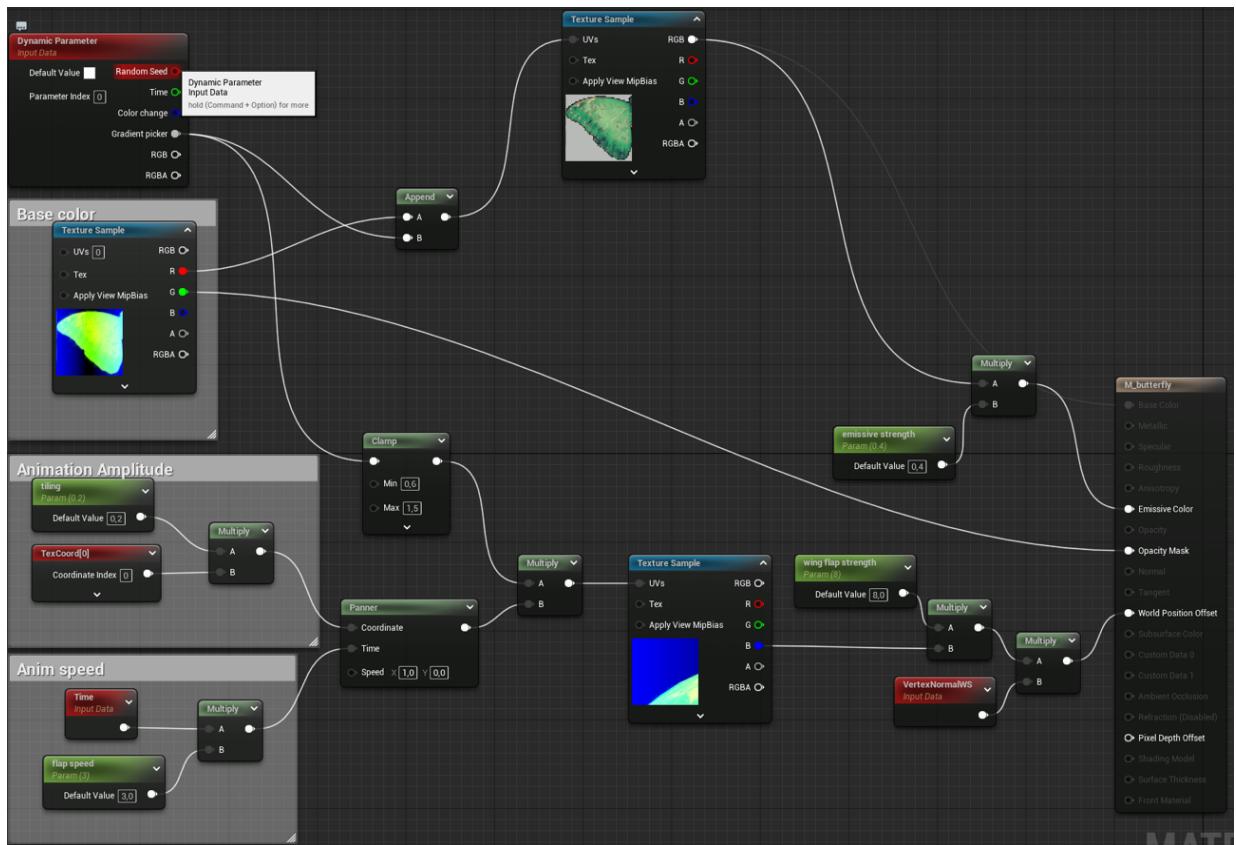


Figure 33 : Shader du matériau des papillons

L'animation des papillons sans squelette est possible grâce à l'utilisation d'un nœud **Panner**, qui anime la texture appliquée aux papillons en la faisant défiler le long des coordonnées UV. Ce défilement horizontal, contrôlé par le Panner, permet de régler à la fois la vitesse et l'amplitude du battement des ailes, en utilisant les informations contenues dans le canal B de la texture.

Pour la déformation des sommets, le dégradé animé par le Panner est multiplié par la normale des sommets dans l'espace monde (VertexNormalWS). Cela détermine la direction et l'intensité du mouvement à appliquer à chaque sommet du maillage du papillon. Le résultat est ensuite connecté au World Position Offset ce qui modifie dynamiquement la position des sommets.

En parallèle, le shader gère également la coloration des papillons en utilisant une image contenant plusieurs dégradés de couleurs. Chaque ligne de l'image représente un dégradé différent, permettant de varier les couleurs appliquées aux papillons dynamiquement. Pour cela nous utilisons un Dynamic Parameter, paramètre défini dans le système émetteur de particules. Ce paramètre, modifié aléatoirement chaque fois qu'une particule est générée dans le système, permet de sélectionner une position Y différente sur l'image de dégradé de couleurs.

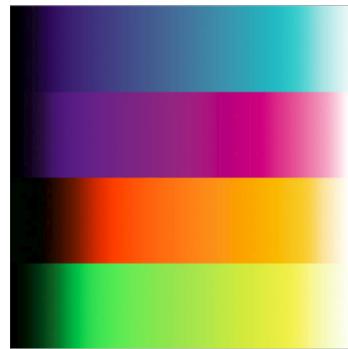


Figure 34 : Texture dégradée des papillons

Ainsi, avec un seul shader et deux textures, nous pouvons créer des papillons animés et aux couleurs différentes en utilisant moins de ressources sur le processeur graphique. Afin de créer les nuées de papillons, certains modules ont été rajoutés à l'émetteur dans Niagara.

5. Suite du projet

A compter d'aujourd'hui, il reste encore un an de développement avant la fin du projet. Sur chacun des aspects, il reste donc beaucoup de travail à fournir pour terminer les jeux dans le temps et garantir aux futurs utilisateurs l'expérience la plus agréable possible. Pour de futurs développeurs, de par la nature open source du projet, il est également important de fournir un projet avec les finitions propres et les documents nécessaires pour sa continuation.

5.1 Serious Game

Le Serious Game approche maintenant de sa forme finale, mais il reste encore plusieurs tâches importantes pour garantir que le jeu plaise aux futurs joueurs, au niveau du gameplay et de l'environnement, et que les parties puissent se dérouler sans bugs.

Pour cette tâche-ci, bien que la majeure partie du développement soit terminée, il peut rester quelques bugs. Pour les trouver, il n'y a pas le choix : nous devons faire tester au plus de monde possible, afin de récupérer les retours d'expériences des utilisateurs. En faisant jouer des citoyens ou décideurs, nous pourrions donc déceler certains bugs mais pour être sûr de parcourir tout le jeu, nous devons faire « exprès » de casser le jeu, pour être sûr de recréer toutes les situations possibles. Par exemple, un testeur devra faire exprès de pousser plusieurs potentiomètres de la machine énergie en même temps, pour s'assurer que rien ne se passe, ou essayer de se téléporter dans des zones qui normalement ne s'y prêtent pas. De plus, analyser ces retours permet de faire les ajustements nécessaires pour améliorer les mécaniques de jeu et répondre aux attentes des joueurs.

Ensuite, la décoration de l'environnement n'est pas encore terminée. L'esthétique du jeu est importante. Nous l'avons remarqué durant les tests, les joueurs sont conquis rien qu'en rentrant dans le dôme, car ils n'ont pas l'habitude de la VR. Rendre l'environnement encore plus cohérent et joli permet l'amélioration de l'immersion des utilisateurs. Ces ajouts incluront des objets décoratifs et des effets visuels.

Parallèlement, nous devrions se pencher sur les effets sonores (SFX). Pour l'instant, il y en a quelques-uns dans le jeu mais la musique principale n'est pas encore mise. En effet, nous devons ajouter une musique d'ambiance, qui évoluera en même temps que l'utilisateur. Pour chaque machine validée, un instrument se rajoutera. Lorsque le joueur aura validé toutes les machines, il pourra redécouvrir la station et sa verdure mais également entendre la bande sonore complète.

5.2 Escape Game

Pour l'Escape Game, le développement progresse vers une version fonctionnelle des deux premières pièces : le laboratoire et la cuisine. Actuellement, la conception des mécaniques de jeu de ces deux salles est terminée et nous commençons maintenant l'environnement 3D. Après la complétion du développement de ces deux pièces, qui devrait être fini durant le mois de septembre, il faudra donc commencer la conception les pièces suivantes, toujours basées sur les thèmes des idées reçues à intégrer dans une pièce de vie de la station. Ce processus de conception va prendre du temps, car nous devons régulièrement échanger sur ce sujet avec les partenaires, afin d'avoir une pièce d'escape game qui correspond à tous avant de commencer le développement pur.

5.3 Documentation

LIFE V-air était open source, il est essentiel de créer une documentation pour le serious game ainsi que l'escape game, afin que des personnes extérieures puissent à leur tour utiliser le projet pour créer. Le développement a commencé il y a maintenant un an et demi et le projet est devenu assez vaste. Pour les deux jeux, on retrouve énormément d'éléments, de blueprint, de fonctions à l'intérieur de ces blueprints... Pour quelqu'un qui ne connaît pas bien Unreal Engine ou le projet, on peut rapidement se sentir perdu. C'était d'ailleurs mon cas au début : le projet était déjà bien avancé et il m'a fallu du temps pour me repérer et comprendre certains choix d'architecture. L'idée serait donc de mettre en place un wiki pour le projet, qui répertorierait les différents blueprints et autres éléments, avec des descriptions techniques des systèmes de jeu et des mécaniques de gameplay

6. Conclusion

Ce stage a été une expérience extrêmement enrichissante, tant sur le plan professionnel que personnel. Au fil des six mois passés au sein d'ATMO Grand Est, j'ai eu l'opportunité de m'immerger dans deux projets, le serious game et l'escape game, qui ont grandement évolué depuis février. L'environnement du serious game est devenu bien plus accueillant et immersif, et les retours concernant des erreurs en jeu, sur le document technique ou autres points négatifs se font plus rares. Les partenaires sont satisfaits, la plupart des joueurs arrivent aujourd'hui à finir les machines et à apprivoiser les commandes en réalité virtuelle. Le jeu se rapproche aujourd'hui de sa version finale, même si des améliorations de l'univers visuel sont toujours envisageables. Travailler sur l'escape game m'a également permis d'approfondir ma compréhension du développement de jeux, en particulier en ce qui concerne le level design et le game design. Ce projet continue d'évoluer, et j'aurais l'occasion de consacrer un mois supplémentaire en septembre pour parfaire cette expérience en intégrant les retours des partenaires pour davantage le design. Le développement dans Unreal Engine a débuté en juillet, il sera important de réaliser des tests utilisateurs rapidement pour être sûr que la direction de développement est la bonne.

J'ai également ressenti une réelle différence dans mes compétences et mon habilité à travailler sur de gros projets en entreprise, entre février et juillet. En effet, bien que les projets étudiants permettent d'avoir un premier regard en matière de moteurs de jeux et de logiciels de modélisation 3D, participer à un projet concret avec des partenaires extérieurs m'a permis de prendre pleinement conscience de l'ampleur du travail requis. Cette expérience m'a sorti de ma zone de confort en prenant part à la médiation, la communication, ou encore l'optimisation des performances d'un jeu. J'ai également pu développer en autonomie mes compétences en conception d'expériences virtuelles. J'ai ainsi pu développer un profil polyvalent en touchant à divers aspects du développement de jeux vidéo et de la gestion de projet.

Au-delà des compétences techniques, ce stage m'a également permis d'affiner mes envies professionnelles. J'ai découvert que la conception, le game design et le level design sont les domaines qui m'intéressent le plus, et je suis désormais déterminé à orienter mon projet professionnel dans cette direction.

Il n'y a pas eu de nombreuses contraintes durant le stage, en grande partie grâce à notre petite équipe de deux personnes, ce qui nous a permis de répartir les différentes tâches de manière flexible. Nous nous divisions chacun les tâches en fonction des envies mais également des besoins. En effet, en ne prenant qu'en compte mes envies, j'aurais probablement passé plus de temps sur la création de l'environnement 3D ou la conception. Cependant, après un premier travail sur la machine Agriculture, j'ai réalisé qu'il me restait encore des lacunes dans le développement en Blueprint et sur Unreal Engine en général. J'ai donc poursuivi avec le développement de la machine énergétique, ce qui m'a permis de renforcer mes compétences sur cet outil clé de l'industrie du jeu vidéo.

La seule contrainte majeure était la coordination avec les nombreux partenaires impliqués. Aucun d'entre eux n'étaient spécialisés en développement de jeux vidéo et, avec Alexandre, nous étions les seuls à être à plein temps sur le projet. Il y avait des semaines où nous ne recevions aucun retour de la part des partenaires. Certains étaient tout de même assez présents et réactifs, mais de manière générale, chaque étape de conception ou développement nécessitait l'approbation de plusieurs d'entre eux avant de pouvoir passer à l'étape suivante. Cela entraînait souvent des ralentissements, car il n'était pas rare d'attendre jusqu'à deux semaines après l'envoi des documents pour obtenir les validations nécessaires, ou à défaut, organiser des réunions pour traiter du problème. Nous étions donc dépendants

du temps que prenaient les différents partenaires pour examiner et approuver les éléments soumis.

Pour finir, ce stage a été une très bonne opportunité de renforcer mes compétences, d'affiner mes aspirations professionnelles, et de me préparer à de futures responsabilités dans le domaine du développement vidéoludique, ou plus largement de l'informatique graphique.

7. Bibliographie

Blueprint : Fonctionnement et directives de performance – Unreal Engine France. (s. d.). [En ligne]. Disponible sur: <https://unrealenginefrance.com/blueprint/blueprint-fonctionnement-et-directive-de-performance/>

Catak, Guven & MASALCI, Server & Şenyer, Seray. (2020). A Guideline Study For Designing Virtual Reality Games. AJIT-e: Online Academic Journal of Information Technology. [En ligne]. Disponible sur :
https://www.researchgate.net/publication/348195955_A_Guideline_Study_For_Designing_Virtual_Reality_Games

Chorianopoulos, Konstantinos & Giannakos, Michail & Chrisochoides, Nikos. (2014). Design Principles for Serious Games in Mathematics [En ligne]. Disponible sur :
https://www.researchgate.net/publication/266377068_Design_Principles_for_Serious_Games_in_Mathematics

David, D., Edwin, N., Arman, E., Hikari, N., Chandra, N., & Nadia, N. (2019). Development of Escape Room Game using VR Technology. *Procedia Computer Science*. [En ligne]. Disponible sur : <https://www.sciencedirect.com/science/article/pii/S1877050919311421>

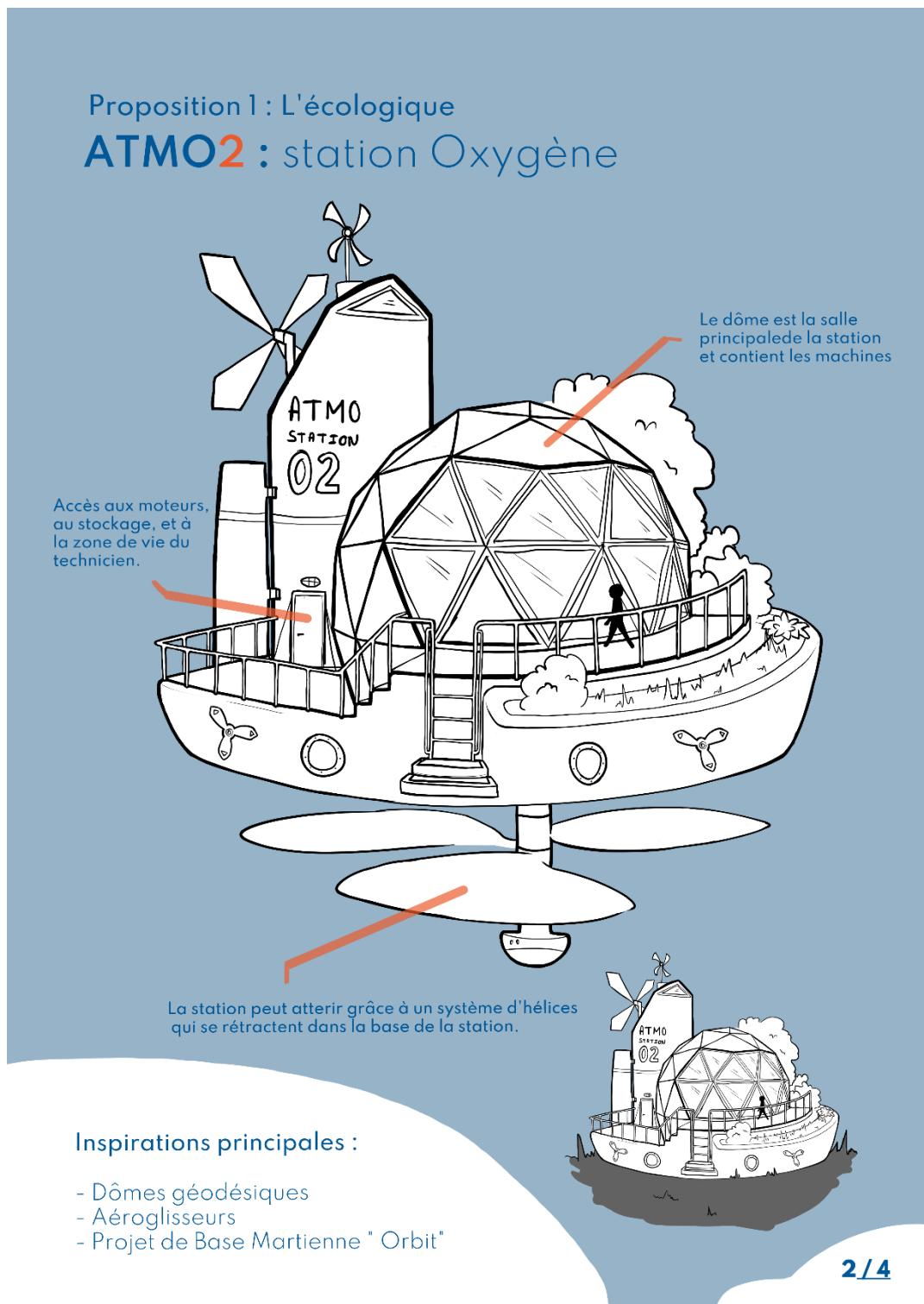
Quibel, O. (s. d.). *Serious Frames - Creation 3D transmedia visites virtuelles et serious games*. [En ligne]. Disponible sur: <https://www.seriousframes.com/>

Qui sommes-nous ? | ATMO GrandEst. (s. d.). | ATMO GrandEst [En ligne]. Disponible sur :
<https://www.atmo-grandest.eu/qui-sommes-nous>

Unreal Engine 5.2 Documentation. Epic Games. [En ligne]. Disponible sur:
<https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-5-4-documentation>

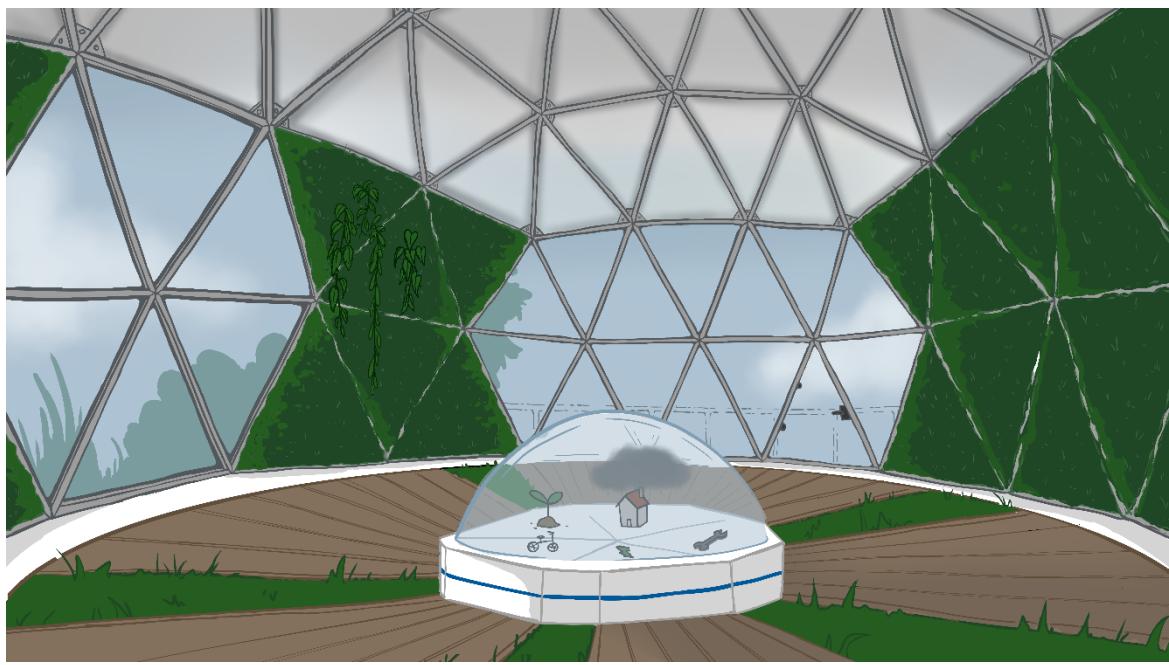
8. Annexes

1. Dessins de conception de la station, réalisés par Serious Frames.



Inspirations principales :

- Dômes géodésiques
- Aéroglisseurs
- Projet de Base Martienne "Orbit"



Intérieur de la station : le dôme

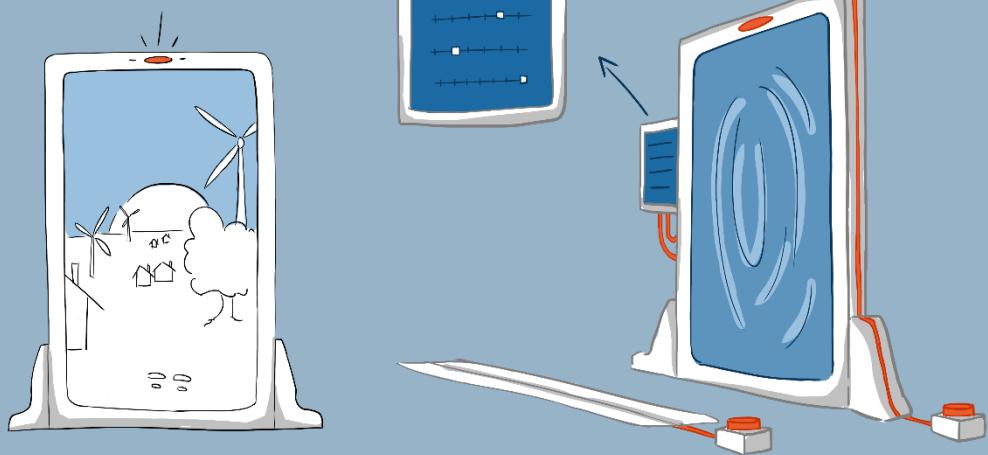
Machine "Résidentiel" ATMO2 :
Carte paysager

Cette machine prend la forme d'un plan paysager sous verre à la façon d'un architecte. Ce plan est posé sur un socle qui monte quand le technicien travaille dessus afin de mieux l'observer. Le technicien devra repérer les logos sur les éléments en volume de la carte. Le technicien devra ensuite faire des choix grâce aux boutons. Ses boutons sont activés, quant allumés.

A diagram of the ATMO2 Residential Machine. It shows two versions of a landscape plan under glass. The left version is a simple green field with small trees. The right version is more detailed, showing houses, trees, and a road. Above these plans are two circular callouts. The first callout contains a stylized green tree with a small number '3' next to it. The second callout contains a small house with a star and a crescent moon logo on its facade. An arrow points upwards from the base of the machine towards the plans.

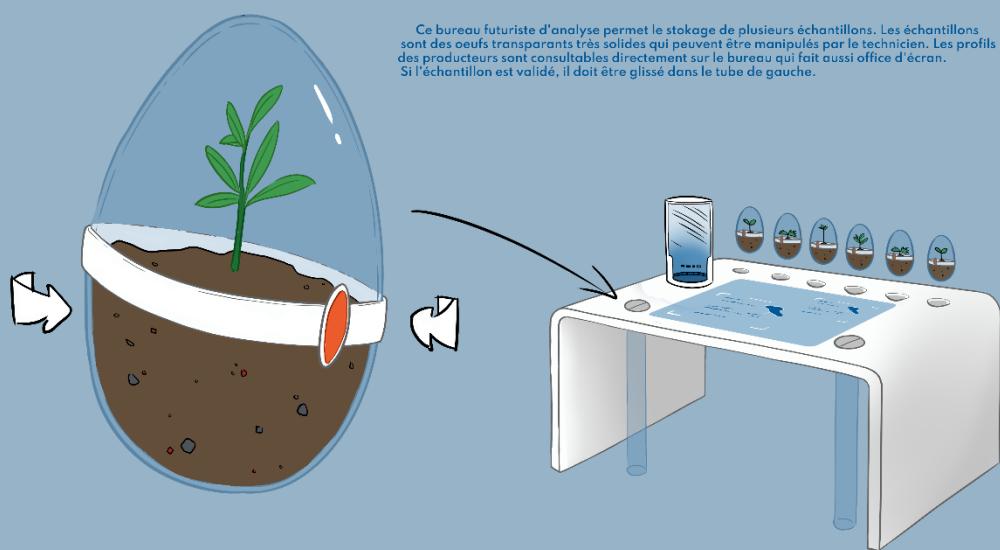
Machine "Energies" ATMO2 : Numérique

Cette porte sort du sol à la demande du technicien. Le panneau de contrôle numérique gère les éléments du paysage qui s'affichent dans la porte. Un bouton au sol permet de fermer ou d'ouvrir la porte pour changer les éléments.



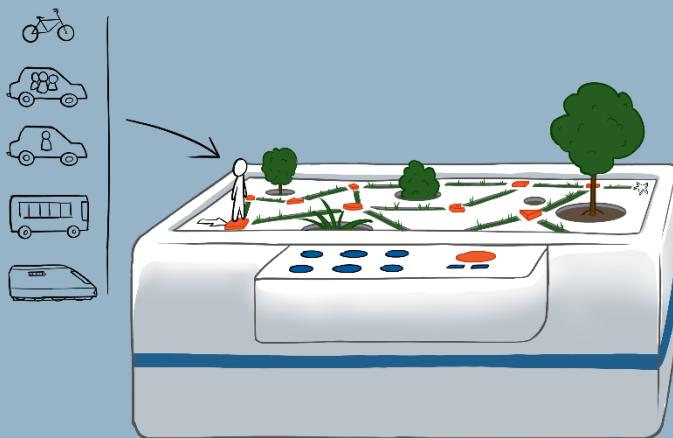
Machine "Agriculture" ATMO2 : Les oeufs de verre

Ce bureau futuriste d'analyse permet le stockage de plusieurs échantillons. Les échantillons sont des œufs transparents très solides qui peuvent être manipulés par le technicien. Les profils des producteurs sont consultables directement sur le bureau qui fait aussi office d'écran. Si l'échantillon est valide, il doit être glissé dans le tube de gauche.



Machine "Mobilité" ATMO2 : Le pot de fleur

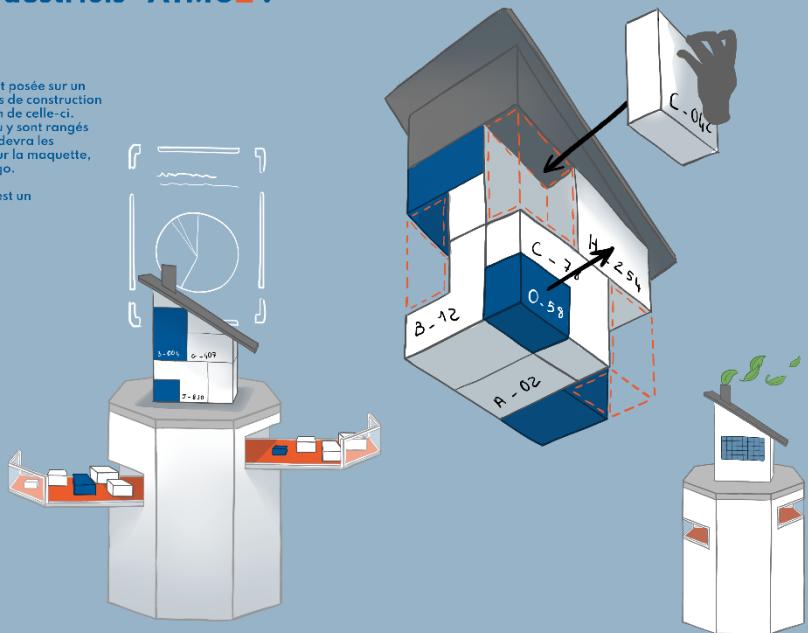
Sur la base d'un pot de fleur, a été installé cette machine. Le parcours d'un personnage d'un point A à un point B. Pour cela le technicien va devoir choisir un moyen de transport adapté, grâce aux boutons, de forme en forme et cela jusqu'à l'arrivé. Une fois choisi, le personnage se transforme en vélo (si c'est le moyen de locomotion choisi) et va jusqu'à la prochaine forme.



Machine "Industriels" ATMO2 : Lego

Une maquette d'usine est posée sur un socle qui contient des pièces de construction essentielles à l'amélioration de celle-ci. Ce socle ouvre des tiroirs où y sont rangés les éléments. Le technicien devra les chercher et les assembler sur la maquette, comme une construction lego.

Le panneau d'information est un hologramme.



2. Exemple d'une page du document technique : la machine Agriculture.

Module :Agriculture

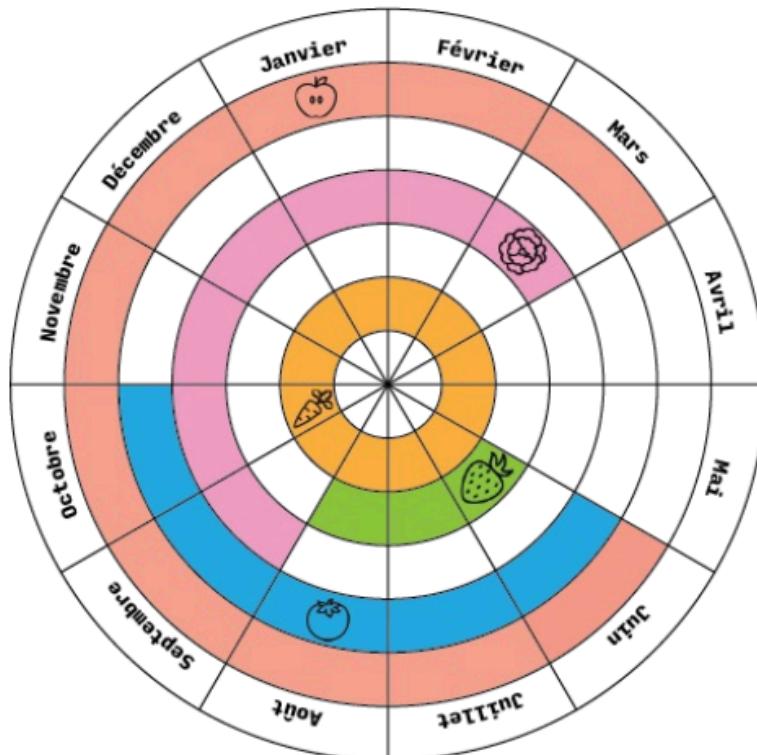
Fonctionnement du module:

Les œufs sur la table représentent différentes exploitations dont les fiches techniques sont affichées sur le tableau:

- Associez les œufs aux exploitations
- Associez les exploitations à leur fiche technique
- Pour chaque exploitation, vérifiez si elle est majoritairement vertueuse ou non (période récolte, état du sol, pratiques agricoles).
- Triez les œufs dans chaque tuyau

Périodes de récoltes :

Vérifiez que les cultures soient récoltées dans le respect des saisonnalités de chacune



DREANO Hermance

Rapport de stage ST52 - P2024

Sujet de stage et description du travail réalisé

Ce rapport résume le travail effectué durant mon stage de fin d'études en informatique, réalisé chez Atmo Grand Est, association en charge de la surveillance de la qualité de l'air dans la région Grand Est.

Le projet présenté est celui de LIFE V-air, un projet européen qui notamment pour but de créer deux jeux sérieux visant à sensibiliser la population aux questions environnementales comme la qualité de l'air. Le premier jeu, à destination des décideurs politiques, se déroule dans une station où les joueurs doivent résoudre des énigmes en collaboration, chacune se rapportant à un secteur différent, touchant la qualité de l'air. Le deuxième jeu, pour le grand public, se présente à la manière d'une escape game, où le joueur doit résoudre une série de puzzles basés sur des idées reçues concernant la pollution et la qualité de l'air, afin de sortir d'une salle dans le temps imparti.

Ces pages parcourent donc le travail de développement de l'environnement virtuel avec Blender et Substance Painter, de la logique de jeu sur Unreal Engine, ainsi que la conception des énigmes des deux jeux.

ATMO GRAND EST

5 rue de Madrid
67 300 Schiltigheim
<https://www.atmo-grandest.eu>