

RAPPORT DE SOUTENANCE 1

Fragments of Ember

par Emberfall



Thomas BOBEE
Lucas ETIENNE
Hugo GUYENNET
Alexandre d'AVEZAC de CASTERA

Table des matières

1	Introduction	2
1.1	Objectif et intérêts pour ce rapport	2
1.2	Présentation du rapport	2
1.3	Rappel des différentes parties et avancement	2
2	Interfaces et mécaniques de jeu	4
2.1	Mécaniques de jeu	4
2.1.1	Déplacements	4
2.1.2	Interactions	7
2.1.3	Pouvoirs	8
2.2	Interfaces	9
2.2.1	Menus en jeu	9
2.2.2	Menu écran titre	10
3	Univers du jeu	11
3.1	Cartes	11
3.2	Objectifs du jeu	12
3.3	Assets	13
3.3.1	Modèles 3D	13
3.3.2	Son	14
3.3.3	Difficulté rencontrée	14
4	Multijoueur	15
4.1	Réseau	15
4.2	Mode de jeu	15
5	Le site web	16
5.1	État actuel	16
6	Conclusion	17
6.1	Avancement du projet	17
6.2	Plan pour le futur	18

1 Introduction

Le projet Fragments of Ember est un jeu vidéo d’aventure et de puzzle développé par notre équipe, Emberfall , dans le but de créer une expérience immersive et réfléchie pour les joueurs. Ce jeu s’inscrit dans un univers riche où chaque salle représente un défi unique, alliant exploration et résolution de mystères. L’objectif principal de ce projet est de concevoir un gameplay original qui stimule à la fois la réflexion et l’interaction avec un environnement captivant.

1.1 Objectif et intérêts pour ce rapport

Dans les sections suivantes, nous détaillerons la conception du gameplay, l’architecture technique, l’interface utilisateur et l’aspect narratif de Fragments of Ember, ainsi que l’approche artistique qui soutient l’ambiance mystérieuse et envoûtante du jeu. Enfin, nous discuterons des perspectives d’évolution du projet et des projets futurs qui pourraient enrichir l’expérience de jeu.

1.2 Présentation du rapport

Ce rapport de soutenance présente les étapes clés de la conception et du développement de Fragments of Ember, en détaillant la vision créative de l’équipe, le processus de développement technique, ainsi que les choix artistiques qui ont façonné l’univers du jeu. À travers ce document, nous mettrons en lumière les défis rencontrés et les solutions mises en place pour offrir un jeu fluide, innovant et captivant. Nous aborderons également la manière dont nous avons intégré les retours des tests et les améliorations continues, afin de garantir une expérience de jeu optimale.

1.3 Rappel des différentes parties et avancement

TABLE 1 – Avancement des tâches par soutenance

Tâches	Soutenance 1	Soutenance 2	Soutenance finale
Scénario	100%	100%	100%
Multijoueur	0%	50%	100%
Réseau	0%	50%	100%
Son	0%	25%	100%
Modèles 3D	10%	70%	100%
Animations	0%	50%	100%
Site Web	65%	85%	100%
Gameplay	5%	85%	100%
Ajout de l’IA	0%	50%	100%
Menu	10%	100%	100%
Documentation	40%	70%	100%

Tâches	Alexandre	Thomas	Hugo	Lucas
Gameplay				
- Mouvements		R		S
- Interaction		R		S
- Pouvoirs		R		S
Graphisme				
- Design	R		S	
- Environnements et niveaux	R		S	
- Animations	S		R	
Intelligence artificielle				
- Comportement des ennemis			S	R
- Interactions avec le joueur			S	R
Multijoueurs				
- Mécanique coopérative	R			S
- Synchronisation des joueurs	R			S
- Réseau	R			S
Site Web				
- Conception		S		R
- Documentation		R	S	
Sons				
- Musiques		S	R	
- Effets sonores		S	R	
Scénario				
- Lore	S			R
- Développement des chapitres		S	R	
- Dialogue	R		S	

TABLE 2 – Répartition des tâches par membre de l'équipe

2 Interfaces et mécaniques de jeu

2.1 Mécaniques de jeu

2.1.1 Déplacements

Tous les objets Unity possèdent un composant appelé Transform, qui sauvegarde la position et la rotation de l'objet dans l'espace. Pour gérer les déplacements du ou des joueurs, il est nécessaire d'ajouter un composant supplémentaire à l'objet : le Rigidbody. Ce composant permet d'intégrer l'objet au moteur de physique de Unity et offre de nouvelles méthodes et propriétés, telles que velocity, qui gère la vitesse de l'objet. Ainsi, en modifiant la vitesse d'un objet en réponse à une entrée utilisateur (par exemple, l'appui sur une touche), il est possible de générer un mouvement.

```
var Forward:bool = Input.GetKey(ForwardKey);
var Backward:bool = Input.GetKey(BackwardKey);
var Right:bool = Input.GetKey(RightKey);
var Left:bool = Input.GetKey(LeftKey);

Vector3 spdX = Vector3.zero, spdY = Vector3.zero;

if (Forward) {
    spdX += new Vector3(Mathf.Sin(angle), 0, Mathf.Cos(angle));
}

if (Backward) {
    spdX += new Vector3(-Mathf.Sin(angle), 0, -Mathf.Cos(angle));
}

if (Left) {
    spdY = new Vector3(-Mathf.Cos(angle), 0, Mathf.Sin(angle));
}

if (Right) {
    spdY = new Vector3(Mathf.Cos(angle), 0, -Mathf.Sin(angle));
}

jump();
```

FIGURE 1 – Code extrait du script du joueur, permettant le déplacement de celui-ci

Les vecteurs gérant la direction du joueur utilisent l'angle du joueur. En effet, si l'on tourne le personnage vers la droite, le joueur souhaite avancer tout droit par rapport au personnage. Grâce à la trigonométrie, nous pouvons déterminer les formules présentées ci-dessus.

Le saut est un peu plus complexe. En effet, le personnage ne peut pas sauter dans les airs. Il est donc nécessaire de vérifier que le joueur touche le sol avant de lui permettre de sauter. Pour cela, nous pouvons utiliser le ray casting, qui consiste à envoyer un rayon dans une direction donnée afin de détecter l'objet touché par ce rayon. Unity met à notre disposition la méthode `Physics.Raycast` pour effectuer cette opération.

```
private bool IsGrounded() {  
    debug  
    return Physics.Raycast( origin: rb.position, direction: Vector3.down, out _, maxDistance: .5f);  
}
```

FIGURE 2 – Code extrait

La fonction ci-dessus permet d'obtenir un booléen indiquant si le rayon a intercepté un objet situé à moins de 0,5 unité de distance du point de départ.

Pour empêcher les objets de se traverser, il est nécessaire d'ajouter un nouveau composant : le Collider. Ce dernier peut prendre plusieurs formes et permet de gérer les interactions entre objets en évitant leur enchevêtrement.

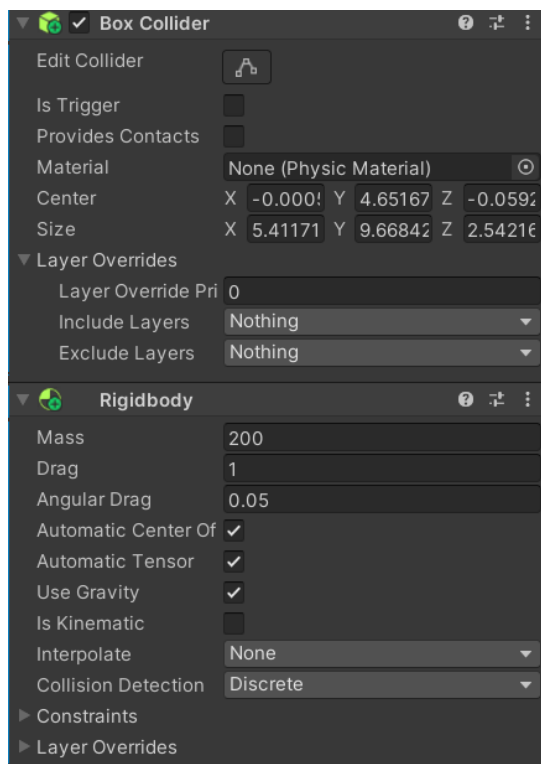


FIGURE 3 – Code extrait

Le joueur doit également avoir la possibilité de tourner sa caméra afin d'observer le monde qui l'entoure. Pour cela, nous utilisons la méthode `Input.GetAxis`, qui renvoie la position relative de la souris par rapport à sa position précédente. De plus, la caméra doit rester attachée au joueur, ce qui est assuré par la dernière ligne de code. Celle-ci permet de déplacer la caméra grâce à la propriété `follow`, qui, dans ce cas, est définie sur le joueur.

```
if (!GlobalSceneScript.paused) {  
    var mouseX :float = Input.GetAxis("Mouse X");  
    var mouseY :float = Input.GetAxis("Mouse Y");  
  
    var (angleX :float , angleY :float , angleZ :float ) =  
        (rb.rotation.eulerAngles.x, rb.rotation.eulerAngles.y, rb.rotation.eulerAngles.z);  
  
    var newAngleX :float = angleX + mouseY * sensivity;  
    var newAngleY :float = angleY + mouseX * sensivity;  
  
    rb.rotation =  
        Quaternion.Euler(newAngleX, newAngleY, angleZ);  
  
    transform.position = Follow.transform.position;  
}  
}
```

FIGURE 4 – Code extrait

2.1.2 Interactions

Abordons maintenant les interactions. Comme dans tout jeu d'énigmes, le joueur a la capacité d'influencer le monde qui l'entoure. Pour cela, nous réutilisons `Physics.Raycast` afin de détecter l'objet avec lequel le joueur interagit. Ensuite, un tri est effectué pour déterminer quel script doit être exécuté suite à cette interaction, si une exécution est nécessaire.


```
if (Input.GetKeyDown(InteractKey)) {  
    RaycastHit hit;  
  
    if (Physics.Raycast(origin: Tete.transform.position, direction: Tete.transform.forward, out hit, maxDistance: 4,  InteractLayer)) {  
  
        if (GlobalScript.debugMode)  
            Debug.Log(hit.transform.name);  
  
        var split :string[] = hit.transform.name.Split(' ');  
        switch (split[0]) {  
            case "InteractableBook":  
                var bookNumber :int = int.Parse(split[1]) - 1;  
  
                ListScriptsBooks[bookNumber].Interact();  
                break;  
        }  
    }  
}
```

FIGURE 5 – Code extrait

Dans le cas des livres dits interactifs, nous devons les incliner afin de composer un code qui servira à résoudre différentes énigmes. Les angles dans Unity étant représentés par des Quaternions, leurs valeurs varient entre -1 et 1. Le script suivant permet d'incliner un livre dans la position souhaitée :

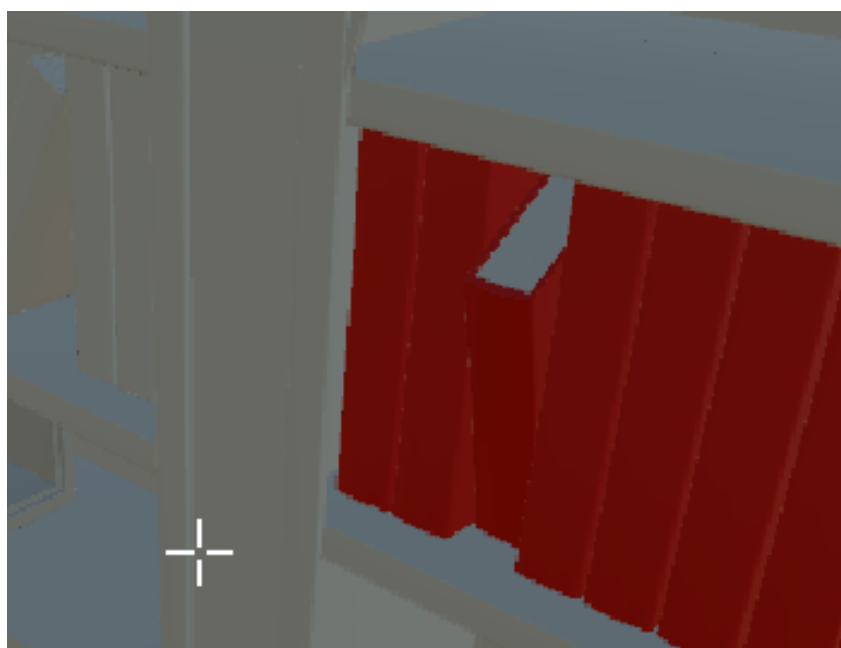


FIGURE 6 – Livre actionné


```

var angle :float = transform.rotation.z;

if (activated && angle < -0.5648) {
    transform.Rotate(new Vector3(0, 0, 1));
}
else if (!activated && angle > -0.7071) {
    transform.Rotate(new Vector3(0, 0, -1));
}

```

FIGURE 7 – Code extrait pour les livres

2.1.3 Pouvoirs

Les différents pouvoirs pouvant être obtenus à fur et à mesure des différents chapitres qui seront indispensables pour accomplir certaines énigmes. Le joueur avancera dans une salle suivante pour avoir la possibilité de trouver des indices pour revenir dans la salle précédente pour tenter de débloquent le pouvoir. Pour l’instant nous travaillons sur un pouvoir permettant de modifier la taille du personnage. Ce pouvoir permettra au joueur de se faufiler dans des petits espaces précédemment inaccessibles.

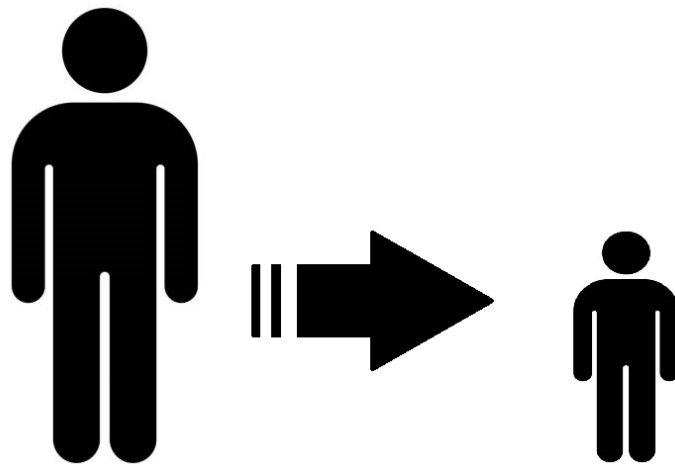


FIGURE 8 – Exemple de pouvoir

2.2 Interfaces

2.2.1 Menus en jeu

Comme illustré ci-dessous, le menu pause est fonctionnel et intègre les boutons "Resume", "Setting" et "Exit", répondant ainsi à l'ensemble des objectifs fixés pour ce menu. La section "Setting" offrira la possibilité d'ajuster le volume général du jeu ainsi que celui des musiques. Cependant, cette fonctionnalité n'est pas encore opérationnelle à ce stade.



FIGURE 9 – Exemple de menu pause en jeu

2.2.2 Menu écran titre

L'écran titre, interface principale du jeu, est actuellement fonctionnel. Il donne accès aux options pour personnaliser les paramètres, tels que le volume (et, à venir, les commandes), ainsi qu'au lancement du jeu, qui propose deux modes : solo et multijoueur. Une option "Quit" permet également de quitter le jeu.

Son design, en harmonie avec l'univers médiéval du jeu, intègre des textures, des couleurs et des typographies spécifiques à l'ambiance globale.



FIGURE 10 – Exemple de l'écran titre.

Le menu multijoueur, encore en développement, est une fonctionnalité clé. Il permettra de gérer les interactions en ligne via un lobby, espace intermédiaire où les joueurs pourront créer ou rejoindre des parties, configurer des paramètres comme la salle de départ, le nombre de participants et les règles, ou communiquer avant la partie grâce à un chat.

L'interface visera une ergonomie intuitive et une immersion cohérente avec l'univers du jeu. Elle devra également répondre à des contraintes techniques telles que la stabilité réseau, la gestion des latences et la synchronisation des actions. Des éléments visuels et sonores viendront renforcer l'ambiance et l'expérience utilisateur.



FIGURE 11 – Menu play.

3 Univers du jeu

3.1 Cartes

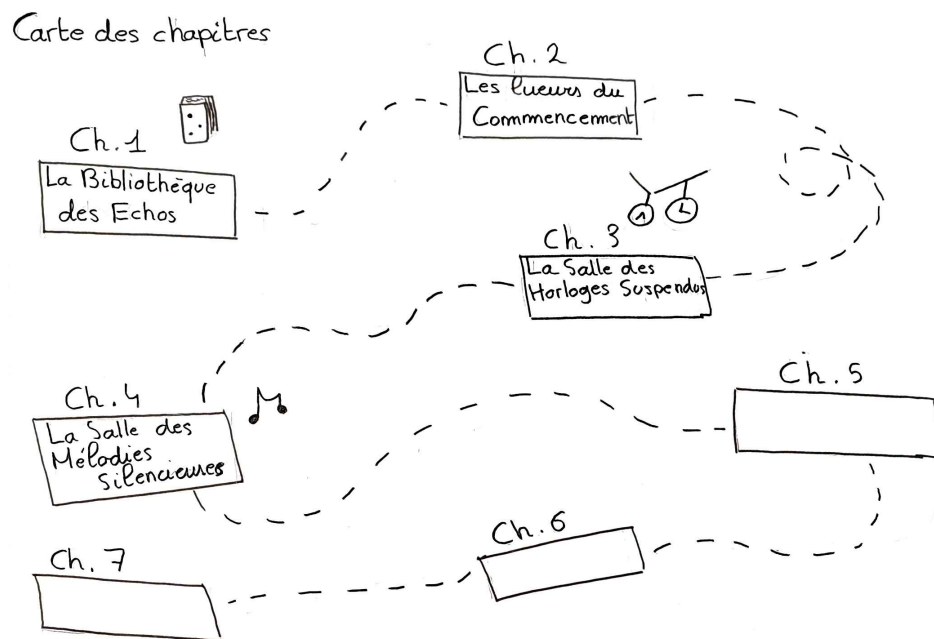


FIGURE 12 – Carte du jeu

Le jeu sera composé de 7 chapitres, chacun relié entre eux.

3.2 Objectifs du jeu

Dans *Fragments of Ember*, le joueur incarne un héros plongé dans un monde mystérieux composé de salles énigmatiques. L'objectif principal est de restaurer l'équilibre d'Ember en collectant et en assemblant les Fragments, des artefacts essentiels dispersés dans chaque salle. Chaque environnement propose un défi unique, mêlant puzzles logiques, mécaniques, environnementaux et narratifs. Le joueur doit explorer ces espaces clos, observer et écouter attentivement les indices laissés par un monde vivant qui réagit à ses actions.

Le jeu est structuré en 7 chapitres, chacun représenté par une salle distincte. Chaque chapitre se termine par la résolution de l'énigme principale de la salle, permettant d'accéder au chapitre suivant. Certaines salles renferment également des pouvoirs cachés. Les 7 salles forment une suite logique et chronologique, mais le joueur a la possibilité de revenir dans une salle précédente pour tenter d'obtenir ces pouvoirs cachés, en s'appuyant sur des indices découverts ultérieurement.

L'histoire se dévoile progressivement à travers des interactions avec des gardiens éthérés et des inscriptions cryptiques, qui révèlent la mission du héros et son lien profond avec Ember. La réflexion et l'observation sont essentielles, car les énigmes reposent sur des éléments variés tels que des motifs, des sons et des lumières. En relevant les défis propres à chaque salle, le joueur répare les liens brisés entre les différentes parties du royaume.

Au fil de sa progression, le joueur découvre également des fragments du passé d'Ember, enrichissant sa compréhension de cet univers complexe. Restaurer l'équilibre final nécessite de surmonter les obstacles, de rassembler tous les Fragments et de lever le voile sur les mystères entourant le héros et son rôle dans ce monde fascinant.

3.3 Assets

3.3.1 Modèles 3D

En termes de modèles 3D, nous disposons actuellement de plusieurs éléments pour la première salle. Parmi eux, une bibliothèque, des objets de décoration tels qu'un personnage et des torches. Ces choix s'intègrent dans le style médiéval que nous avons adopté, lequel se reflète également dans le design de l'interface utilisateur présenté précédemment.

Afin de renforcer l'immersion et de souligner la progression narrative, nous nous efforcerons de varier les couleurs, les textures et les objets utilisés dans chaque salle. Cette approche permettra de créer une ambiance unique pour chaque chapitre, en harmonie avec les thématiques et les défis spécifiques qui y sont proposés. Par exemple, des teintes sombres et des matériaux bruts pourraient évoquer un chapitre axé sur l'oppression ou le mystère, tandis que des couleurs vives et des textures légères pourraient accompagner une atmosphère plus apaisante ou triomphale.

Le jeu de lumières jouera également un rôle central dans l'établissement de l'atmosphère de chaque salle. L'éclairage pourra être utilisé pour créer des contrastes marqués, renforcer les tensions ou guider subtilement le joueur vers des points d'intérêt. Par exemple, une lumière vacillante issue de torches pourrait ajouter un sentiment d'insécurité ou de mystère, tandis qu'un éclairage doux et uniforme pourrait apaiser le joueur dans des moments de répit. Dans certaines salles, des effets dynamiques comme des ombres en mouvement ou des changements d'intensité lumineuse seront également exploités pour renforcer l'immersion.

Les modèles 3D actuellement en cours de production incluent des éléments pour les chapitres 1 et 2, dont une première version des couleurs, textures et lumières est prévue pour mi-février. Ces premiers prototypes serviront de base pour développer et affiner les éléments des salles suivantes, en garantissant une cohérence visuelle tout en permettant des variations marquées d'un chapitre à l'autre.

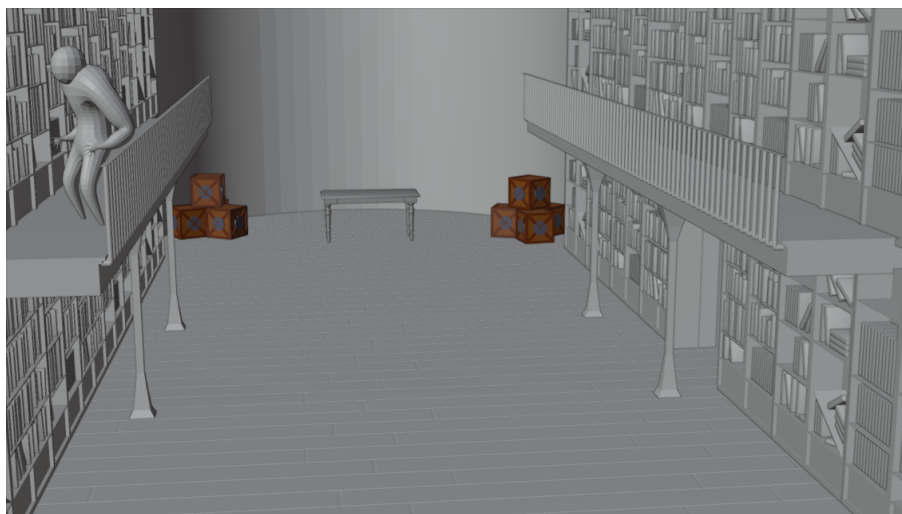


FIGURE 13 – Chapitre 1 - La bibliothèque des échos

3.3.2 Son

Dans le cadre de notre développement, nous avons principalement travaillé sur les effets sonores et les bruitages de l'environnement. À ce stade de notre avancement, nous avons intégré plusieurs effets sonores fondamentaux à notre jeu.

Par exemple, des bruitages de pas sur un plancher en bois ont été créés pour enrichir l'immersion dans les chapitres où le personnage évolue dans des espaces tels qu'une chambre ou une bibliothèque. Nous avons également développé des sons spécifiques pour accompagner les moments clés, notamment lorsque le joueur termine une quête. Ces sons varient donc en fonction de la réussite ou non de l'énigme présentée au joueur.

Prochainement nous allons développer les musiques de fond qui seront jouées tout au long du jeu et qui changera en fonction la scène dont le joueur est présent.

3.3.3 Difficulté rencontrée

Lors du développement du son dans notre jeu vidéo, nous avons rencontré plusieurs problèmes techniques et artistiques. Par exemple, lors de la composition des musiques jouées en fond sonore, un choix précis et cohérent d'instruments est absolument nécessaire. Ce choix n'est pas anodin, car il influence directement l'expérience immersive des joueurs.

Nous devons donc analyser attentivement chaque scène de notre jeu, en prenant en compte l'ambiance visuelle, narrative et émotionnelle, afin de trouver des instruments qui, auditivement, s'harmonisent parfaitement avec la mise en scène. Cette étape est cruciale pour garantir une immersion totale.

En effet, si elle est négligée, une musique qui ne correspond pas à l'atmosphère d'une scène peut non seulement perturber l'expérience des joueurs, mais aussi rendre certaines séquences inconfortables, voire déplaisantes. Cela pourrait avoir un impact négatif sur la qualité perçue du jeu, voire nuire à son succès global.

Pour éviter de tomber dans cet écueil, nous avons choisi d'investir un temps conséquent dans le processus de sélection des instruments et l'analyse minutieuse de chaque scène.

Ce travail préliminaire nous a permis de poser des bases solides pour la composition musicale. Grâce à cette méthodologie rigoureuse, nous sommes désormais en mesure de nous concentrer pleinement sur la création musicale elle-même, en utilisant les instruments que nous avons soigneusement sélectionnés.

Ce processus, bien qu'exigeant, nous assure que chaque note et chaque mélodie contribueront à renforcer l'immersion des joueurs dans l'univers de notre jeu, tout en rendant leur expérience mémorable et agréable.

4 Multijoueur

4.1 Réseau

Le multijoueur de Fragments of Ember sera disponible en mode local (LAN) et en ligne (Online).

En mode local, les joueurs pourront se connecter via un réseau local, ce qui est idéal pour jouer dans un même espace physique. Une fois les appareils connectés au même réseau, un joueur pourra héberger la partie et les autres pourront rejoindre facilement en utilisant une fonction de découverte automatique ou en entrant une adresse IP locale.

En mode online, les joueurs pourront se connecter à distance grâce à un serveur dédié. Ce serveur gèrera les connexions, synchronisera les données en temps réel et permettra de sauvegarder les progrès en coopération. Chaque joueur devra disposer d'un compte pour accéder à ce mode et rejoindre ou créer des parties en ligne.

L'intégration de ces fonctionnalités dans le jeu se fera par Unity Netcode.

4.2 Mode de jeu

Le multijoueur de Fragments of Ember repose sur la coopération. Les joueurs évoluent dans les mêmes salles, résolvant ensemble les énigmes.

Chaque joueur contribue activement, avec des actions simultanées ou complémentaires.

L'objectif est de favoriser le travail d'équipe pour une expérience immersive et collaborative.

5 Le site web

5.1 État actuel

Dans le cadre du projet Fragments of Ember, nous avons conçu et développé un site web dédié à la présentation de l'équipe et des objectifs du projet. L'objectif principal était de mettre en avant nos valeurs tout en créant une mise en page attrayante, moderne et fonctionnelle pour le site. Nous avons créé une structure responsive en utilisant HTML et CSS, permettant un affichage optimal sur tous les types d'appareils, notamment les ordinateurs, tablettes et mobiles. Cependant, des tests supplémentaires sont encore nécessaires pour vérifier la compatibilité avec tous les appareils. Une page d'accueil a été intégrée, résumant certaines idées clés et offrant des boutons pour poursuivre la lecture. De plus, une timeline a été ajoutée pour permettre de suivre l'avancement du projet et de le partager, avec la nécessité de la mettre à jour régulièrement. Nous avons également développé une grille de présentation pour organiser chaque membre de l'équipe dans des "cartes" distinctes, avec un design harmonieux et cohérent. Des styles visuels modernes, incluant des bordures, des ombres et des avatars arrondis, ont été appliqués pour améliorer l'expérience utilisateur. Des contenus textuels ont été intégrés, détaillant le rôle et les responsabilités de chaque membre de l'équipe, tout en respectant une identité visuelle professionnelle et immersive. Enfin, nous avons testé l'affichage du site sur plusieurs navigateurs pour garantir une compatibilité optimale.

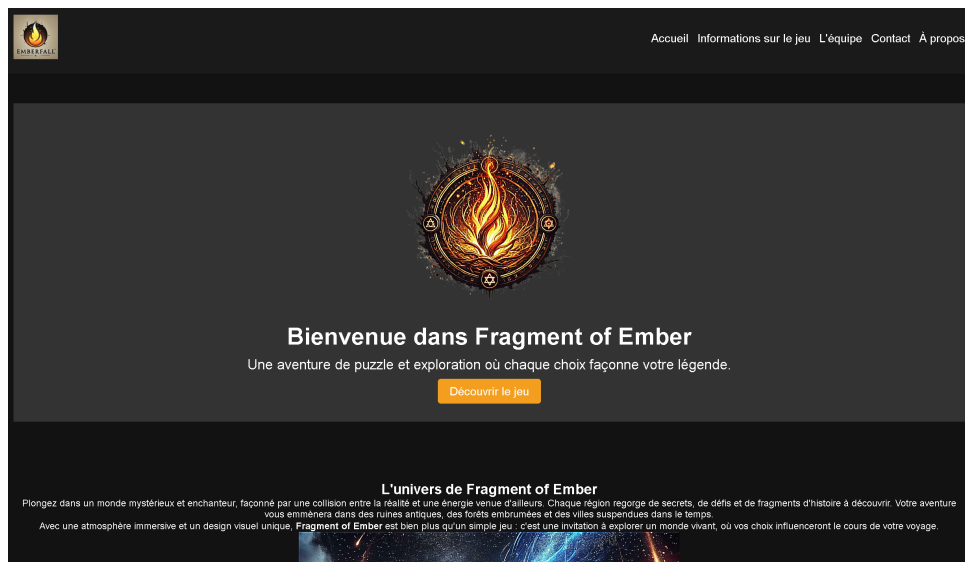


FIGURE 14 – Site web - Accueil

6 Conclusion

Le projet Fragments of Ember représente une fusion ambitieuse entre innovation, réflexion, et créativité. À travers ce premier rapport de soutenance, nous avons pu explorer les fondements de son développement, qu'il s'agisse de la conception des mécaniques de jeu, de la création d'un univers immersif, ou encore de la mise en place des aspects techniques et artistiques essentiels.

Les progrès réalisés témoignent de l'engagement et de la synergie de notre équipe. Chaque étape, qu'il s'agisse du développement du gameplay, des graphismes ou de l'intégration sonore, a été franchie avec soin, en veillant à respecter les délais et les objectifs initiaux. Les défis rencontrés ont été l'occasion de renforcer nos compétences et d'affiner notre vision collective du projet.

Ce premier jalon marque une avancée significative dans la concrétisation de notre jeu. Il ouvre également la voie à de nouvelles opportunités d'amélioration, avec la finalisation des éléments encore en cours.

En conclusion, Fragments of Ember n'est pas simplement un jeu, mais une expérience immersive et engageante, qui continue d'évoluer grâce à la passion et au dévouement. Nous sommes impatients de poursuivre cette aventure et de partager avec vous l'aboutissement de notre travail dans les prochaines étapes du projet.

6.1 Avancement du projet

TABLE 3 – Avancement des tâches par soutenance

Tâches	Soutenance 1	Soutenance 2	Soutenance finale
Scénario	40%	100%	100%
Multijoueur	0%	50%	100%
Réseau	0%	50%	100%
Son	5%	25%	100%
Modèles 3D	10%	70%	100%
Animations	0%	50%	100%
Site Web	65%	85%	100%
Gameplay	10%	85%	100%
Ajout de l'IA	0%	50%	100%
Menu	70%	100%	100%
Documentation	25%	70%	100%

	En retard
	En avance

6.2 Plan pour le futur

Des défis ont été rencontrés lors du développement du jeu, et ces expériences serviront de leçons pour améliorer nos processus à l'avenir. Par exemple, pour les modèles 3D, nous veillerons à appliquer les textures avant l'assemblage ou l'envoi. En programmation, nous adopterons une méthode de travail séparée, avec deux versions du jeu : une pour les tests et une autre stable pour la version finale. Concernant le multijoueur, nous commencerons directement à coder en mode multi afin d'éviter les problèmes de synchronisation.

Dans le cadre du développement web, l'alignement des éléments pour assurer un rendu harmonieux sur différents écrans ont été un défi, résolu grâce à l'utilisation des grilles CSS et de Flexbox après plusieurs itérations. La gestion des couleurs et de la typographie a nécessité des tests répétés pour concilier l'univers visuel de Fragments of Ember et une lisibilité optimale.