

# Aegina : rapport de soutenance n° 1

Florian AMSALLEM (amsall\_f) , Théo ISSARNI (issarn\_t),  
Julien MOUNIER (mounie\_a), Romain MOUNIER (mounie\_r)

AIM<sup>2</sup>



# Table des matières

<b>Introduction</b>	<b>4</b>
<b>1 Univers du jeu</b>	<b>4</b>
1.1 Aegina . . . . .	4
1.2 Ile . . . . .	5
1.3 Les minerais . . . . .	5
1.3.1 Mithril . . . . .	5
1.3.2 Floatium . . . . .	5
1.3.3 sunkium . . . . .	6
<b>2 Réalisation</b>	<b>8</b>
2.1 Pré-janvier . . . . .	8
2.1.1 Cycle jour/nuit (Julien) . . . . .	8
2.1.2 Inventaire (Romain et Florian) . . . . .	8
2.1.3 Animation 3D (Théo) . . . . .	8
2.1.4 Déplacement du personnage et de sa camera (Théo et Julien) . . . . .	9
2.1.5 Models 3D (Julien) . . . . .	9
2.2 Janvier . . . . .	9
2.2.1 Barre de selection (Florian) . . . . .	9
2.2.2 Menus (Romain et Florian) . . . . .	9
2.2.3 Class <i>Item</i> (Romain) . . . . .	9
2.2.4 Multi-joueurs (Florian) . . . . .	10
2.2.5 Son (Théo et Romain) . . . . .	10
2.3 Février . . . . .	10
2.3.1 Génération aléatoire (Florian et Julien) . . . . .	10
2.3.2 Drop d'Item (Florian et Julien) . . . . .	11
2.3.3 Bibliothèque (Florian, Romain et Théo) . . . . .	11
2.3.4 Histoire (Romain et Théo) . . . . .	11
2.4 Mars . . . . .	11
2.4.1 Interaction avec l'environnement (Florian et Julien) . . . . .	11
2.4.2 Cristal (Romain et Julien) . . . . .	11
2.4.3 Chat (Florian) . . . . .	11
2.4.4 Musique (Théo) . . . . .	11
2.4.5 Survie (Théo) . . . . .	11
2.4.6 Alpha (Florian) . . . . .	11

<b>3</b>	<b>Prévision</b>	<b>12</b>
3.1	Les quatre grand axe . . . . .	12
3.1.1	Creation d'objet . . . . .	12
3.1.2	Ajout des créatures . . . . .	12
3.1.3	Ajout des succès . . . . .	12
3.1.4	PVP et Conquête . . . . .	12
3.2	Les améliorations . . . . .	12
3.2.1	Survie . . . . .	12
3.2.2	Génération aléatoire et sauvegarde . . . . .	12
3.2.3	Histoire . . . . .	12
3.2.4	Environnement sonore . . . . .	12
3.2.5	Personnage . . . . .	12
	<b>Conclusion</b>	<b>13</b>

# Introduction

# 1 Univers du jeu

## 1.1 Aegina

Le monde d'Aegina est constituée d'une quasi infinité d'île volante se trouvant au dessus d'un vide béant appelé les *abysses*. Se monde contrairement au notre est instable, il ne partage pas nos lois de l'équilibre et de nombreux phénomènes étranges s'y déroule sans que l'on puisse les expliquer la présence des *abysses* n'étant qu'un de ses phénomènes.

La principale source de l'instabilité de ce monde sont les cristaux que l'on peut trouver sur certaines îles. Ceux-ci sont capable des déchirer les trames de l'espace et du temps permettant de réaliser des miracles comme le retour dans le temps ou la téléportation. Malgré toute ces instabilité un écosystème ses tout de même développée dans ce monde. Des animaux comme ceux que l'on connait tel que des lapins, papillons et moutons peuple les îles d'Aegina. Mais d'autres espèces plus agressifs et plus exotiques sont apparue suite à l'instabilité du monde comme les slimes. Un instinct naturel pousse toute ces créatures agressifs à s'attaqué aux cristaux si ceux-ci sont activé pour les empêché de rendre le monde plus instable qu'il ne l'est actuellement.

On pourrait donc penser que sans personnes pour activer les cristaux et avec une telle armée de créature près à défendre ce monde celui-ci aurait trouver une certaine forme d'équilibre. Mais c'était sans compté la présence d'envahisseurs, d'être extérieur à ce monde qui utilise celui-ci à des fins barbares, les êtres humains. Bien que ne vivant pas directement dans ce monde et n'ayant aucune information sur celui-ci les êtres humains on trouvé un moyen de « téléporter » leurs déchets dans un néant qui n'est autre que Aegina. Lorsque les déchets sont seulement des restes et que ceux-ci tombent directement dans les *abysses* cela ne rend pas Aegina plus instable, au contraire cela peut créer un apport de ressource à ce monde si ils ne tombent pas directement dans les *abysses*.

Mais certaines autorités humaines utilise se moyen pour se débarrasser de déchets plus embêtant : d'autres humains. Malheureusement une fois dans le monde d'Aegina ceux-ci tente désespérément de survivre et ils font alors appelle aux cristaux et à la pouvoir quasi divin leur permettant d'échapper à la faim et à la mort. certains voyagent alors d'île en île créant quelques infrastructure telle que des ponts pour s'aider, laissant leur trace dans ce monde mais l'amenant petit à petit à sa fin sans le savoir. Heureusement jusqu'a présent ils ont tous finit par périr sous l'assaut des créatures d'Aegina ou par suicide ne pouvant rentrer chez eux mais il arrivera bien un jour ou l'un d'eux emportera ce monde avec lui dans sa chute.

## 1.2 Ille

Ille est le personnage principale de notre jeux. Il a l'apparence d'un bucheron roux et barbus. En vérité Ille n'est pas son vrai nom mais en arrivant dans le monde d'Aegina Ille a perdu la memoire et ne s'en souviens donc plus.

## 1.3 Les mineraïs

### 1.3.1 Mithril

Le mithril dans le monde d'Aegina est un minéral rose un peu plus solide que le fer. Assez rare sur la surface des îles on en trouve toujours en petite quantité à l'intérieur de celle-ci puisque le mithril est un des deux minéraux nécessaires à la survie d'une île. Le mithril a l'étrange particularité de réagir avec les phénomènes étranges d'Aegina parfois les amplifiant d'autres fois les stabilisant. Il est impossible de savoir comment celui-ci va réagir avant d'avoir testé.

A l'état naturel il catalyse ces instabilités et les relâche sous formes de vagues d'énergie rendant le minéral chaud au touché. Une fois épurée le mithril continue de catalyser les instabilités mais peut relâcher son énergie sous de nombreuses autres formes que la chaleur comme le ferait un cristal mais à bien plus faible échelle.

D'ailleurs l'énergie catalysée et stockée par le mithril semble être la même que celle se trouvant dans les cristaux car les deux matériaux peuvent échanger cette énergie entre eux. Ainsi quelques chercheurs hérétiques qui se sont trouvés sur Aegina ont supposé que les cristaux étaient formés d'un alliage composé en majorité de mithril. Ces chercheurs ont trouvé la mort quelques jours plus tard tués par des sangliers après avoir déchargé leur cristal à l'aide de mithril.

### 1.3.2 Floatium

Le floatium est un minéral quasi transparent que l'on peut trouver dans Aegina. Les humains qui arrivent dans Aegina le confondent souvent avec du quartz lorsqu'ils en voient dans la roche mais celui-ci est bien plus transparent et s'apparenterait plus à du verre ou de la glace. La vraie particularité du floatium se remarque une fois le minerai épuré de toute impureté. La première constatation est que celui-ci est bien plus solide que ce à quoi on peut penser, sa dureté égale celle du fer, mais surtout il est impossible de le faire tomber par terre. En effet une fois celui-ci lâché il flotte.

Ce minéral est encore un mystère d'Aegina car personne ne sait pourquoi il flotte. Il n'est pas plus léger que l'air puisqu'il reste à une distance fixe du sol mais il ne semble pas s'agir

non plus de force magnétique. De toute façon peu d'expérience on eu lieu sur le floatium car ,dans Aegina, en plus d'être un minerai très rare, les personnes capable de faire des expérimentations scientifique dessus le sont encore plus.

La seule information sur est que le floatum est, avec le mithril, l'un des deux composants permettant aux îles de voler. Cependant les cristaux et le floatium peuvent réagir entre eux , le cristal semble pouvoir enlever ces capacités au floatium tandis que le floatium est responsable de la rotation et de la lévitation du cristal, mais tout cela reste de la théorie.

Néanmoins il est possible de conjecturer que si un cristal à une trop grande influence sur une île il pourrait provoquer sa chute. Mais les seuls personne connaissant la vérité sur cette hypothèse sont surement mort.

### 1.3.3 sunkium

Le sunkium est un minéral d'une couleur mauve très sombre. C'est un minerai légendaire qui renferme de nombreuse qualité car il est le minerai le plus rare, le plus lourd et le plus sombre, le plus solide, le plus cool, le plus classe, le plus *dark* et le plus unique d'Aegina. Pourtant malgré sa solidité il ne pas si difficile à extraire des roches car il semble repousser les autres minerai lors de sa formation ce qui permet de le trouver dans des alvéoles formé par la roche. Cependant une fois extrait son poids reste tout de même un problème pour le transporter.

Mais ce qui fait de ce minerai le minerai le plus cool est qu'il peut être utiliser très simplement pour la création d'épée ou d'armure. En effet ce minerai nécessite la même température que le fer pour fondre et une fois mélanger avec du mithril il chauffe et refroidie à une vitesse exceptionnelle. De plus la couleur sombre des armures formé avec le sunkium apporte un coté cool et *dark* à toute personne les utilisant. Mais malheureusement cet impression ne reste que quelques secondes car rare sont ceux capables de se mouvoir avec une telle armure, le plupart se retrouvent coincé dans celle-ci incapable de bouger ou de l'enlever amenant à la mort la plus stupide d'Aegina.

Le sunkium, comme le mithril et le floatium, réagit avec les cristaux mais sa réaction est peut être la plus impressionnante des trois, après tout c'est le minerai le plus cool. Le sunkium change de taille et de masse sous l'influence du cristal mais pas comme tout le monde s'y attendrait car plus le sunkium devient petit plus il devient lourd. Il se met alors à absorber de la lumière assombrissant les alentours. Le cristal quand à lui peut absorber le sunkium mais personne ne sait réellement ce que ça lui apporte.

Certaine théorie stipulerait que les pouvoir du cristal viendrait de minuscules morceau de sunkium qui déformerait l'espace temps ce qui permettrait d'expliquer tout les pouvoirs

du cristal. Mais cette théorie s'appuyant sur d'autres théories qui n'ont pas été prouvées sa véracité est plus que discutable. Sans la présence des cristaux le sunkium serait aussi le minéral le plus mystérieux d'Aegina.



## 2 Réalisation

### 2.1 Pré-janvier

#### 2.1.1 Cycle jour/nuit (Julien)

Dans un premier temps, on a du déterminer la durée d'une journée que l'on a subdiviser en 30 phases. Ces phases servent à changer d'image de skybox. Seulement cette méthode à causé quelque problèmes. Entre autre, le changement de skybox était trop brusque et les différentes skybox n'étaient pas homogènes. C'est pourquoi nous avons du trouver une autre méthode : La *directional light*.

La *directional light* est un gameobject propose par unity. Son intérêt majeur est que l'orientation de cette lumière influence la skybox du monde. De plus nous pouvons choisir la couleur de cette lumière. Pour ce faire nous avons utilise une fonction qui converti la longueur d'onde des rayons lumineux en couleur RGB et nous avons représente le spectre solaire a l'aide d'une fonction mathématique.

$$255((3.25t - 4)x^2 + (-4t + 4)x)$$

#### 2.1.2 Inventaire (Romain et Florian)

Nous avons créé un inventaire, celui ci est constitue de 24 cases sous forme de 4 lignes et 6 colonnes. Chaque case peut contenir un element, qui peut être déplacé grace a un drag and drop. De plus le joueur peut avoir une description de chacun des elements en passant sa souris sur celui ci. Cette inventaire s'ouvre lorsque l'on appui sur la touche *inventaire* qui est de base la touche i et peut ensuite être ferme avec la touche *inventaire* ou la touche escape. Lorsque l'inventaire est ouvert le joueur et la camera ne peuvent plus bouger

#### 2.1.3 Animation 3D (Théo)

Ille est un personnage que nous avons télécharger depuis l'asset store qui possédais de base deux animations, "marcher" et "rester immobile" cependant ces deux seules animation ne nous suffisais pas. A partir de ce moment nous avons decide de faire les animations des nos personnage nous meme. Le premier problème que nous avons rencontre : notre personnage n'était pas accessible depuis notre version de blender, trop récente pour le modele 3D que nous avons. Il nous a donc fallut trouver une ancienne version de blender compatible avec notre model 3D. Depuis nous essayons d'améliorer nos animation dans le but d'être le plus réaliste possible.

### 2.1.4 Déplacement du personnage et de sa camera (Théo et Julien)

Nous avons permis à notre personnage de se déplacer en fonction de ce que l'utilisateur souhaite faire, avancer (avec la touche "w" ou "up"), reculer (avec la touche "s" ou "down"), s'orienter vers la droite (avec la touche "q" ou "right") et s'orienter vers la gauche (avec la touche "d" ou "left"). Notre personnage peut courir si ce dernier se déplace et que l'utilisateur appuie sur la touche "shift", il peut aussi sauter si l'utilisateur appuie sur la barre d'espace. A partir de ce moment notre personnage était devenue opérationnel mais à quoi cela servirait il si nous ne pouvions le voir? Nous avons alors décider de créer une camera qui suivrais notre personnage où qu'il aille. Cette caméra n'est pas une simple caméra fixe, elle possède des caractéristiques variées grâce auxquelles une grande liberté est accordé au joueur. A la troisième personne le joueur peut vois son personnage de cotés, de devant ou encore de derrière tout cela en étant en étant proche ou éloigné de son personnage. A la première personne notre camera permet au joueur de voir exactement ce que son personnage vois.

### 2.1.5 Models 3D (Julien)

## 2.2 Janvier

### 2.2.1 Barre de selection (Florian)

Nous avons ajoute une barre de selection synchronise avec la 4<sup>eme</sup> ligne de notre inventaire. De plus le joueur peut sélectionner une case de cette barre et utiliser l'objet s'y trouvant. Cette barre apparait meme quand le joueur n'est pas dans son inventaire

### 2.2.2 Menus (Romain et Florian)

Nous avons cree des menus permettant de modifier la langue et le volume du son du jeu. Ces menus s'ouvre avec la touche escape quand aucune autre interface n'est ouvert et ce ferme avec la touche escape ou avec le bouton *continuer*. Dans le menu le joueur et la camera du joueur ne peuvent pas bouger comme lorsque l'inventaire et ouvert.

### 2.2.3 Class *Item* (Romain)

Dans l'inventaire nous voulons parfois des objets en grandes quantité nous avons alors cree des classes *Item* et *Itemstack*

## 2.2.4 Multi-joueurs (Florian)

Le multi-joueurs est une part importante du projet car celle ci permettra la cooperation des joueurs pour progresser dans le jeu. Nous avons dans un premier temps utilise les fonctionnalitees de *Unet* pour pouvoir creer un serveur (ou un host dans notre cas), mais aussi pouvoir rejoindre un serveur grace a son adresse ip.

Nous avons dans un second temps synchroniser la position, la rotation et les actions des joueurs. Pour cela on a d'abord utilise le composant d'unity *NetworkTransform*, mais celui-ci ne permettait pas la synchronisation fluide des personnages. Du coup nous avons coder nous meme un script permettant la synchronisation des personnages. De plus pour rendre les déplacements le plus fluide possible nous utilisons le principe d'interpolation.

Pour finir nous avons synchronise les elements, les sons et les caracteristique du monde, que ce soit les iles, les biomes, les bruits de pas des joueurs, l'etat de la journee (jour/nuit) ou bien les arbres.

## 2.2.5 Son (Théo et Romain)

Afin de ne pas avoir un jeu fade, nous avons introduit quelque sons. Ces sons nous permette d'avoir une base que l'on pourras enrichir au fur et à mesure que notre jeu avancera. Nous avons commencer par ajouter un son pour l'ouverture et la fermeture de notre inventaire, mais aussi lorsque nous interagissons avec notre menu. Par la suite nous avons ajouté des bruit de pas afin d'entendre notre personnage marcher ou courir, par manque de réalisme nous avons synchronisé ces sons afin que les autres joueurs jouant avec nous les entendent aussi.

## 2.3 Février

### 2.3.1 Génération aléatoire (Florian et Julien)

Comme nous l'avions specifie dans notre cahier des charges, le monde du jeu est genere proceduralement. Pour cette premiere soutenance, nous avons implemente la generation aleatoire d'un chunk, et la liaison des chunks entre eux. Nous avons donc au lancement d'une partie un monde compose de quatre chunks chaccun relies par des ponts.

Nous avons creer 2 formats de chunk differents possedant chaccun une ile. Sur ces iles sont poses des ancrs qui correspondent a des emplacements d'objets. Cela permet lors de la generation du monde de choisir des 3 biomes aleatoirement (foret, desert ou neige), puis de creer sur les ancrs un objet coherent avec le biome. Par exemple dans un biome desert nous pouvons faire apparaitre un cactus alors que dans le biome neige cela est impossible.

Nous avons donc tous les elements pour creer des chunks, il ne restera plus qu'a faire apparaitre les chunks proceduralement avec un agencement correcte. Puis il sera aussi necessaire de sauvegarder le monde avec les modifications que les joueurs y ont apportees.

### **2.3.2 Drop d'Item (Florain et Julien)**

### **2.3.3 Bibliothèque (Florian, Romain et Théo)**

### **2.3.4 Histoire (Romain et Théo)**

## **2.4 Mars**

### **2.4.1 Intéraktion avec l'environnement (Florian et Julien)**

### **2.4.2 Cristal (Romain et Julien)**

### **2.4.3 Chat (Florian)**

### **2.4.4 Musique (Théo)**

Afin de compléter l'univers que nous essayons de créer à travers notre jeu, nous avons ajouté des musiques de fond. Nous avons ajouté pour l'instant quatre musique à notre jeu, la première est une musique qui se lance dès que l'on arrive sur le menu de notre jeu, il s'agit du seul endroit où vous pourrez entendre cette musique qui est consacré à notre menu. Trois autres musique on part la suite étaient ajouté, avec trois thèmes différents qui sont : la foret, le désert et la neige. Pour le moment le choix de la musique écouté se fait aléatoirement.

### **2.4.5 Survie (Théo)**

### **2.4.6 Alpha (Florian)**

## 3 Prévvision

### 3.1 Les quatre grand axe

#### 3.1.1 Creation d'objet

#### 3.1.2 Ajout des créatures

#### 3.1.3 Ajout des succès

#### 3.1.4 PVP et Conquête

### 3.2 Les améliorations

#### 3.2.1 Survie

#### 3.2.2 Génération aléatoire et sauvegarde

#### 3.2.3 Histoire

#### 3.2.4 Environnement sonore

#### 3.2.5 Personnage

## Conclusion