Valentin Destrebecq

Cahier des charges Station 88

Description du document

|  |  |
| --- | --- |
| **Titre** | Cahier des charges Station 88 |
| **Date** | 01/10/2018 |
| **Auteur(e)s** | Valentin Destrebecq |
| **Sujet** | Création du cahier des charges de Station 88 |
| **Email** | [valentin.destrebecqepitech.eu](mailto:playficient_2019@labeip.epitech.eu) |
| **Mots clés** | Cahier des charges, projet libre, station88 |
| **Version du modèle** | 1.0 |

Tableau des révisions

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Auteurs** | **Section** | **Commentaire** |
| 01/10/2018 | Valentin Destrebecq | Toutes | Création et détail de toutes les sections |

**Sommaire**

[I. Description générale du projet 1](#_Toc526185596)

[A. Définition du terme Metroidvania 1](#_Toc526185597)

[B. Expérience de jeu voulue 2](#_Toc526185598)

[II. Technologies utilisées 2](#_Toc526185599)

[III. Scope/Domaine du projet 3](#_Toc526185600)

[IV. Description des différentes parties du projet 3](#_Toc526185601)

[V. Description des tâches fonctionnelles 4](#_Toc526185602)

# Description générale du projet

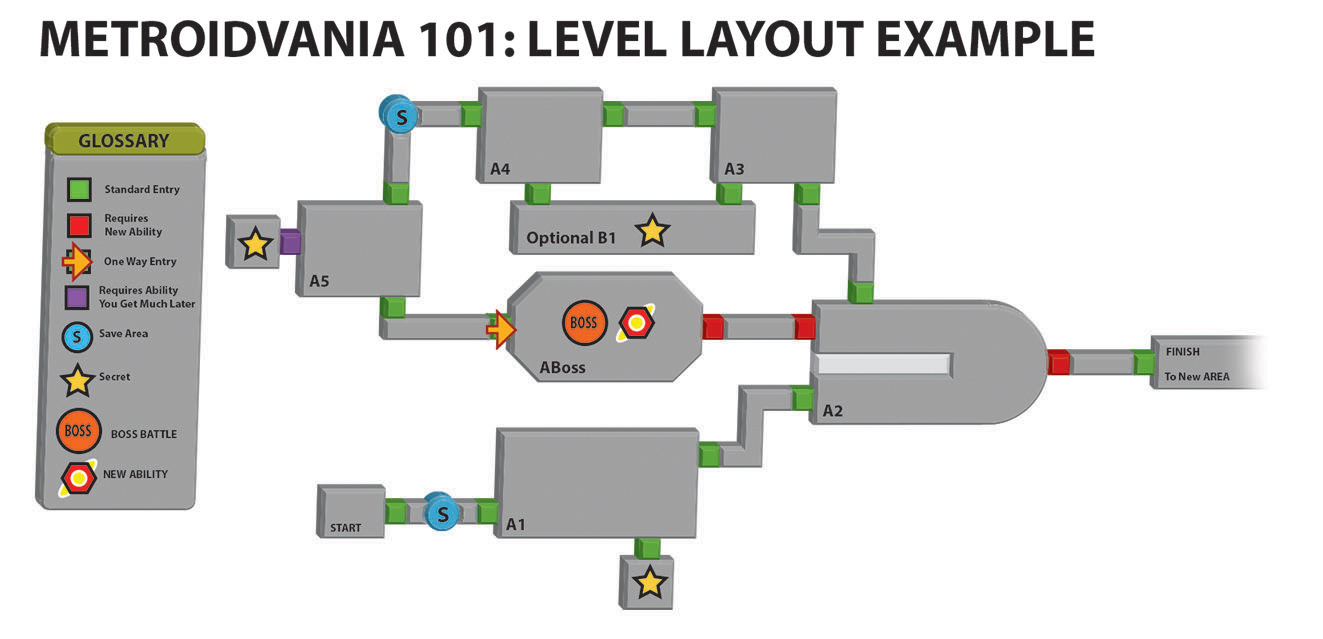
Station 88 est un MetroidVania se déroulant dans une station spatiale abandonnée.

Ferrailleur à la recherche de matériaux, vous vous retrouvez bloqués sur la station et chercherez à en sortir tout en comprenant les raisons de son abandon.

## Définition du terme Metroidvania

Le Metroidvania est un sous-genre du jeu vidéo d'action-aventure qui emprunte fortement au système de jeu de la série Metroid et à celle de la série Castlevania, plus précisément à partir du jeu Castlevania : Symphony of the Night et ses suites. Le terme est un mot-valise des deux noms de la série.

Un jeu Metroidvania est le plus souvent un jeu de plates-formes doté d'une seule et grande carte interconnectée avec des pièces ou des sections simples. Toutes les zones de cette carte ne sont pas accessibles dès le début du jeu, le joueur doit obtenir des items, des armes, des clefs ou une amélioration du personnage pour franchir l'obstacle ou la difficulté qui bloque son avancement. Souvent, les items sont protégés par des boss, permettant de scénariser le défi contre eux tout au long du jeu. Les cartes sont non-linéaires, et obligent souvent le joueur à parcourir plusieurs fois la carte pour terminer le jeu. Les monstres les plus faibles réapparaissent quand le joueur repasse dans des zones où il est déjà passé, lui permettant de récupérer des points de vie, des munitions ou des points d'expérience.

Les plus gros jeux comportent généralement des points de sauvegarde ainsi que la possibilité de téléporter rapidement le personnage dans des zones éloignées de la carte, supprimant les longs et fastidieux déplacements en fin de partie. L'obtention de nouvelles capacités permet également d'ouvrir des raccourcis qui réduisent les temps de déplacements et de découvrir des secrets qui améliorent les capacités du personnage. En tant que tel, le genre se concentre sur l'exploration d'un monde avec une grande carte et l'amélioration des capacités du personnage.

*Fig. 1 : Example de section de niveau d’un Metroidvania*

## Expérience de jeu voulue

Le joueur, suite à l’activation d’un menu de démarrage, pourra accéder à son personnage, fraîchement atterri sur la station 88, dans un affichage 2D (similaire à du 16 bit). Dans la première pièce du jeu, il découvre comment manipuler son personnage : déplacements sur les côtés, saut avec nuance qui lui permet d’escalader une caisse barrant le chemin vers ce qui semble être la sortie. Dans la seconde pièce, il trouve une arme afin de se défendre à distance.

A partir de ce moment, le jeu s’ouvre et offre au joueur des possibilités d’explorations : il a accès à une zone sécurisée où il pourra sauvegarder (et plus tard accéder aux marchands et personnages non-joueurs (PNJ)), mais surtout à des zones plus hostiles lui permettant d’aller plus loin dans la station, de s’améliorer, et de trouver un moyen d’en sortir. Chaque zone est construite de façon similaire à la figure 1 (voir ci-dessus), évidemment agrandie et développée afin de permettre une meilleure expérience de jeu.

# Technologies utilisées

Station 88 est un jeu vidéo réalisé en C# sur le moteur Unity 3D. Il s’agit d’un moteur 3D gratuit, permettant la réalisation de jeux de types et de scopes assez différents, 2D comme 3D.

Pour la gestion du code, l’IDE Visual Studio, intégré en termes de compatibilité avec Unity, sera utilisé. Le code sera versionné grâce à git et sera évidemment disponible sur le dépôt github du projet, en conformité avec les consignes du projet libre.

Pour la création des assets visuels du projet, Aseprite sera utilisé. Il s’agit d’un éditeur de pixel-art payant très pratique et souple, permettant également la création rapide d’animations.

Les sons et musiques utilisés dans le jeu seront pris sur des banques de son gratuites en ligne, et au besoin remixés via Audacity (éditeur sonore).

# Scope/Domaine du projet

Station 88 a pour but d’être un faire-valoir pour une embauche potentielle dans le milieu du jeu vidéo, et spécifiquement le développement gameplay.

De fait, le jeu n’a aucun but de ventes ou de visibilité (même si un suivi et une parution publics seront effectués). Il ne rentre donc pas en concurrence avec d’autres jeux.

Afin de satisfaire ces besoins, le projet sera une démo montrable lors d’un entretient, un salon ou un évènement. Il s’agira d’une bêta, dans laquelle on trouvera toutes les fonctionnalités d’un jeu fini, mais pas le contenu, qui prendrait évidemment bien plus de temps qu’une année à être produit.

# Description des différentes parties du projet

Le projet est découpé en plusieurs parties :

1. Le gameplay. Puisqu’il s’agit du point prioritaire du jeu, cette partie est la plus fournie de toutes. Cette partie regroupe tout ce qui est lié au gameplay du jeu, notamment les déplacements du joueur, ses actions et possibilités.
2. L’environnement et les ennemis. Dans cette partie sont répertoriés les éléments extérieurs à l’avatar du joueur, et avec lesquels il peut interagir. C’est notamment le cas de certains morceaux du décor (portes, caisses destructibles, échelles, etc.) ou encore des ennemis.
3. L’interface. Cette partie regroupe tout ce qui est lié à l’interface visible pendant le jeu (HUD) ainsi que les divers menus accessibles et stoppant son déroulement.
4. La musique et les effets sonores. Cette partie répertorie les tâches liées à l’intégration des sons et musiques dans le projet.

# Description des tâches fonctionnelles

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Station 88 - Projet Libre - Tableur des tâches | | | | |
|  |  |  |  |  |
| Nom | Prénom | Partie du projet | Tâche fonctionnelle | Charge de travail (jour/homme) |
| Destrebecq | Valentin | 1. Gameplay | Système de déplacement : le joueur peut se déplacer de gauche à droite, collisionner avec les murs et obstacles | 3 |
| Destrebecq | Valentin | 1. Gameplay | Système de saut (nuancé) : le joueur peut effectuer un saut. Selon le temps de pression du bouton, le saut est plus ou moins grand | 2 |
| Destrebecq | Valentin | 1. Gameplay | Attaque à distance : le joueur peut générer des projectiles qui infligent des dégâts aux ennemis) | 1 |
| Destrebecq | Valentin | 1. Gameplay | Attaque de mêlée : le joueur peut utiliser une arme type épée lui permettant d'infliger des dégâts si les ennemis sont à portée | 1 |
| Destrebecq | Valentin | 1. Gameplay | Système de combo : le joueur peut enchainer plusieurs coups, créant un combo infligeant un bonus de dégâts | 2 |
| Destrebecq | Valentin | 1. Gameplay | Création du double saut : le joueur peut effectuer un saut supplémentaire dans les airs une fois le 1er fini | 0,5 |
| Destrebecq | Valentin | 1. Gameplay | Création du dash/esquive : le joueur peut utiliser un dash, lui permettant de se déplacer rapidement. Le dash permet notamment d'agrandir la distance parcourue via un saut | 1,5 |
| Destrebecq | Valentin | 1. Gameplay | Création du système d'inventaire : le joueur peut posséder un certain nombre d'objets ramassés lors de sa progression | 1 |
| Destrebecq | Valentin | 1. Gameplay | Création du système de statistiques : le joueur possède des statistiques (attaque, défense, vitesse) pouvant être améliorées | 0,5 |
| Destrebecq | Valentin | 1. Gameplay | Création du système de prise et gestion des dégâts : le joueur possède un nombre fini de points de vie, et subit un game over s’ils tombent à zéro | 1 |
| Destrebecq | Valentin | 1. Gameplay | Ajout du système de modifications de tir : Le joueur peut débloquer des portes spécifiques au type de tir en question (tir explosif, tir électrique). Le type de tir a aussi une influence sur les dégâts occasionnés (plus grands si utilisé sur le bon type d'ennemi) | 2 |
| Destrebecq | Valentin | 1. Gameplay | Système de "ledge grabbing" : le joueur peut s'agripper sur les rebords des plateformes | 2 |
| Destrebecq | Valentin | 1. Gameplay | Création du système de sauvegarde : le joueur peut sauvegarder dans des salles définies | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Système de portes : le joueur peut ouvrir les portes en tirant dessus, passer une porte charge la pièce suivante et y place le joueur | 1 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Eléments destructibles : certains éléments du décor (de taille variable) sont destructibles (caisses, débris, etc.) | 2 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Points de sauvegarde. Activable par le joueur via pression d'un bouton. Permet d'accéder à un menu de sauvegarde | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Système de dialogue : le joueur peut dialoguer avec les personnages non-joueurs via un choix de dialogue (gain d'informations) | 1,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Ennemi : tireur. L'ennemi possède une IA simple. Il peut créer des projectiles devant lui à vue du joueur ou par cycles. | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Ennemi : mêlée. L'ennemi possède une IA simple. Il se déplace à vue du joueur et tente de frapper à portée | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Ennemi : saut. Ennemi de mêlée pouvant également sauter lorsqu'il est proche du joueur pour le surprendre | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Ennemi : rampant. Ennemi se déplaçant sur les plateformes/les murs sans prendre en compte le joueur. Il lui inflige des dégâts uniquement au contact | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Ennemi : boss. Possède une IA complexe. Le boss pourra se déplacer vers le joueur et l'attaquer si celui-ci est loin, renvoyer des projectiles. Le boss peut enclencher des attaques de zones (signalées par un signal visuel et sonore). Le boss possède deux phases, un dialogue étant lancé entre deux. | 2 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Plateforme mouvante : plateforme suivant un trajet prédéfini (rail) | 0,5 |
| Destrebecq | Valentin | 2. Environnement et Ennemis | Echelles : certains murs possèdent un système d'échelle, permettant au joueur de s'agripper et ainsi de monter | 1 |
| Destrebecq | Valentin | 3. Interface | Informations "ingame" (barre de vie, sélection du type de tir) | 1 |
| Destrebecq | Valentin | 3. Interface | Minimap (prévisualisation de la carte pendant le jeu) | 0,5 |
| Destrebecq | Valentin | 3. Interface | Carte : affichage de la position du joueur, des liens entre les salles, de leur type (normale, avec secret, sauvegarde, avec amélioration, avec boss). Accessible via un bouton | 1,5 |
| Destrebecq | Valentin | 3. Interface | Statistiques : le joueur peut consulter ses différentes statistiques via un menu (attaque, défense, vitesse, portée) | 0,5 |
| Destrebecq | Valentin | 3. Interface | Inventaire : le joueur peut consulter et modifier son équipement en fonction des objets qu'il possède via un menu | 1 |
| Destrebecq | Valentin | 3. Interface | Menu principal : le joueur peut lancer une partie, quitter le jeu et accéder aux options | 0,5 |
| Destrebecq | Valentin | 3. Interface | Menu options : le joueur peut régler certains paramètres, comme les touches, le volume des effets sonores, de la musique | 1 |
| Destrebecq | Valentin | 3. Interface | Menu accessibilité : le joueur peut régler la difficulté du jeu (entre normal et accessible, qui divise les dégâts pris), le contraste entre les différentes couches d'éléments 2d, ainsi que les couleurs de l'interface. | 1,5 |
| Destrebecq | Valentin | 4. Musique et effets sonores | Effets sonores : les tirs, morts des monstres et autres éléments comportent un effet sonore dédié | 0,5 |
| Destrebecq | Valentin | 4. Musique et effets sonores | Musique : chaque zone comporte une musique dédiée. C'est également le cas du boss. | 0,5 |

# Annexes

## Mock-up de décor

Ci-joint le mock-up d’un décor créé sous Aseprite, témoignant de la direction prise au niveau de l’ambiance du jeu.

