

Projet Unity

Projet Unity	1
1. Présentation générale du problème	2
1.1 Projet.....	2
1.2 Contexte.....	2
1.3 Énoncé du besoin (finalités du produit pour le futur utilisateur tel que prévu par le demandeur) 1.4 Environnement du produit recherché	4
1.4 Environnement du produit recherché.....	4
2. Expression fonctionnelle du besoin	5
2.1 Fonctions de service et de contrainte	5
2.2 Critères d'appréciation (en soulignant ceux qui sont déterminants pour l'évaluation des réponses).....	5
2.3 Niveaux des critères d'appréciation et ce qui les caractérise	6
3. Cadre de réponse.....	6
3.1 Pour chaque fonction.....	6
3.2 Pour l'ensemble du produit	7

1. Présentation générale du problème

1.1 Projet

1.1.1 Finalités

Les objectifs de départ ont pour la plupart été réalisés tels que la partie programmation, la partie algorithme Unity, les graphiques et les fonctionnalités principales implémentées.

Reste seule incomplète, la partie level design qui s'avère plus contraignante au niveau du temps.

1.1.2 Espérance de retour sur investissement

L'espérance de retour sur investissement de ce projet était avant tout de prendre en main le logiciel Unity en espérant le publier sur une plateforme de jeux. Mais avant tout le principal était de monter en compétence au niveau de la programmation objet, relationnel et logiciel.

Ce projet a également pour but d'être présenté pour l'épreuve E4 du BTS SIO.

1.2 Contexte

1.2.1 Situation du projet par rapport aux autres projets de l'entreprise

Néant.

1.2.2 Études déjà effectuées

Des bases de programmation objet (C#), la maîtrise de logiciel de traitement (Ps), des bases algorithmiques, des notions mathématiques.

1.2.3 Études menées sur des sujets voisins

Différentes études ont été menées sur des sujets voisins tels que l'étude du marché qui nous a permis d'innover certaines fonctionnalités mais aussi de prendre exemple sur différents moyens et techniques de programmations.

1.2.4 Suites prévues

De nombreuses améliorations sont à venir, notamment l'enregistrement des statistiques via une base de données, notamment la sauvegarde du contenu acquis en jeu, la suite des niveaux prévue initialement, le menu, des fonctionnalités supplémentaires implémentées suite à des mises à jour futures, l'ajout de musiques de fond.

1.2.5 Nature des prestations demandées

La nature exigée pour ce projet était de créer un jeu sous Unity. Nous avons pour cela carte blanche au niveau du thème. La seule exigence était d'avoir un projet jouable et pour lequel nous avons réalisé une partie de programmation.

1.2.6 Parties concernées par le déroulement du projet et ses résultats (demandeurs, utilisateurs)

Avant-projet :

Les parties concernées étaient les demandeurs (nous-même et notre professeur) avec des idées de fonctionnalités.

Projet :

Les parties concernées étaient les utilisateurs (nous-même et des bêta-testeurs) qui s'occupaient de noter différents bugs trouvés à l'aide de méthodes de débogage .

1.2.7 Caractère confidentiel s'il y a lieu

Néant

1.3 Énoncé du besoin (finalités du produit pour le futur utilisateur tel que prévu par le demandeur) 1.4 Environnement du produit recherché

Le besoin était la réalisation d'un jeu sur Unity.

1.4 Environnement du produit recherché

1.4.1 Listes exhaustives des éléments (personnes, équipements, matières...) et contraintes (environnement)

Ce projet composé de trois personnes, a utilisé les différents environnements suivants :

-Visual Studio (C#)

-Unity

-Photoshop

1.4.2 Caractéristiques pour chaque élément de l'environnement

-Visual Studio (C#) :

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles.

-Unity :

Le logiciel a la particularité d'utiliser du code (C#) sur la plateforme « .NET » avec l'implémentation Mono. Son éditeur était auparavant basé sur MonoDevelop, via MonoDevelop-Unity, mais à partir de la version 2018.1, il est basé sur Visual Studio Community.

-Photoshop :

Édité par Adobe, il est principalement utilisé pour le traitement des photographies numériques, mais sert également à la création ex nihilo d'images. Il travaille essentiellement sur images matricielles car les images sont constituées d'une grille de points appelés pixels. L'intérêt de ces images est de reproduire des graduations subtiles de couleurs.

2. Expression fonctionnelle du besoin

2.1 Fonctions de service et de contrainte

2.1.1 Fonctions de service principales (qui sont la raison d'être du produit)

Un jeu de plateforme en 2D avec un personnage animé, des équipements interchangeables, différents ennemis ayant des niveaux de difficultés adaptés à chaque niveau.

2.1.2 Fonctions de service complémentaires (qui améliorent, facilitent ou complètent le service rendu)

Un système de mise en vente (shop) et une interface intuitive.

2.1.3 Contraintes (limitations à la liberté du concepteur-réalisateur)

Néant

2.2 Critères d'appréciation (en soulignant ceux qui sont déterminants pour l'évaluation des réponses)

Jeu simple à prendre en main, gratuit contenant des améliorations et une certaine satisfaction à évoluer via les différents niveaux de difficulté.

2.3 Niveaux des critères d'appréciation et ce qui les caractérise

2.3.1 Niveaux dont l'obtention est imposée

Néant

2.3.2 Niveaux souhaités mais révisables

Néant

3. Cadre de réponse

3.1 Pour chaque fonction

3.1.1 Solution proposée

Développement d'une solution applicative.

3.1.2 Niveau atteint pour chaque critère d'appréciation de cette fonction et modalités de contrôle

Le niveau de jouabilité est atteint mais reste à implémenter des niveaux supplémentaires (partie graphique).

3.1.3 Part du prix attribué à chaque fonction

Pour chaque fonction de la solution, un coût en temps uniquement était pris en compte.

3.2 Pour l'ensemble du produit

3.2.1 Prix de la réalisation de la version de base

Seul un coût en temps puisque tout le projet a été créé à partir de sources gratuites.

3.2.2 Options et variantes proposées non retenues au cahier des charges

La réalisation d'un tout autre type de jeu, que ce soit au niveau du thème, de la dimensions utilisée (3D), ou de la variété.

3.2.3 Mesures prises pour respecter les contraintes et leurs conséquences économiques

Néant.

3.2.4 Outils d'installation, de maintenance à prévoir

Néant.

3.2.5 Décomposition en modules, sous-ensembles

Un seul module : l'application.

3.2.6 Prévisions de fiabilité

Une certaine fiabilité est garantie notamment grâce aux bonnes pratiques de programmation et aux corrections de bugs.

3.2.7 Perspectives d'évolution technologique

Mise en place d'une sauvegarde automatique via MySQL.