ASD Tower Defense

Version 2.0 – En réseau

Rapport Intermédiaire

2010

GEN

HEIG-VD

01/01/2010

Aurélien Da Campo

Lazhar Farjallah

Pierre-Dominique Putallaz

Romain Poulain

Table des matières

[2. Introduction 2](#_Toc260747356)

[3. Règle du jeu 2](#_Toc260747357)

[4. Etapes de lancement du jeu 3](#_Toc260747358)

[5. Cas d’utilisation 3](#_Toc260747359)

[6. Protocoles d’échange 3](#_Toc260747360)

[Serveur d’enregistrement 3](#_Toc260747361)

[Serveur de jeu 3](#_Toc260747362)

[7. Model de domaine 3](#_Toc260747363)

[8. Charte graphique 4](#_Toc260747364)

[9. Rôle des participant au sein du groupe 5](#_Toc260747365)

[10. Itérations 6](#_Toc260747366)

[Itération 1 – Serveur d’enregistrement + Interface graphique 6](#_Toc260747367)

[Itération 2 – Serveur de Jeu + Architecture 7](#_Toc260747368)

[Itération 3 –Intégration du serveur de jeu + Interface du Jeu en réseau 7](#_Toc260747369)

[Itération 4 – Lifting de la GUI + Game Design + Amélioration Mode Solo 8](#_Toc260747370)

[Itération 5 – Serveur Web (facultatif) 8](#_Toc260747371)

# Introduction

Ce projet prend place durant notre 4e semestre aux seins de la Haute Ecole d’Ingénierie et de Gestion du canton de Vaud (heig-vd). Notre cours de Génie logiciel (GEN) nous propose de mettre en pratique les notions théoriques acquises en créant une application de type Client/ Serveur.

Nous avons tout de suite pensée à l’amélioration d’un jeu que nous avions créé durant nos cours d’Algorithmes et Structures de Données (ASD2) suivi durant notre 3e semestre. En effet, nous avions réalisé un jeu et ce travail est pour nous l’opportunité d’étendre ce logiciel en lui fournissant des fonctionnalités réseau.

La gestion de ce projet se basera sur la méthode UP.

# Règle du jeu

# Etapes de lancement du jeu

# Cas d’utilisation

# Protocoles d’échange

## Serveur d’enregistrement

TEXTE DE PRESENTATION

*Vous découvrirez le protocole en annexe.*

## Serveur de jeu

TEXTE DE PRESENTATION

*Vous découvrirez le protocole en annexe.*

# Model de domaine

# Charte graphique



# Rôle des participant au sein du groupe

# Itérations

## Itération 1 – Serveur d’enregistrement + Interface graphique

**Durée : 1 semaine – 30 avril 2010 au 7 mai 2010**

**Implémenter complètement la partie Serveur d’enregistrement ainsi que son interface graphique.**

Le serveur d’enregistrement permet d’enregistrer les serveurs de jeu sur un serveur central afin de fournir la liste de ces serveurs aux clients. Ces derniers peuvent ensuite choisir la partie qu’ils veulent rejoindre.

**Pourquoi cette itération prend-t-elle place ici ?**

C’est une petite partie fournissant une bonne introduction à la notion de communication CLT-SRV notamment pour les points suivants :

- Création de notre premier protocole réseau (fixation des standards)

- Première communication CLT-SRV en Java (Création des classes de base)

- Intégration du tout dans une interface graphique cohérente. (Schéma d’interface)

**Résultat attendu :** le client peut enregistrer ses parties sur le serveur d’enregistrement et voir la liste de toutes les parties en attente de joueur(s) depuis une interface graphique.

|  |  |  |
| --- | --- | --- |
| **Fonctionnalités attendues** | | |
| **Ok** | **Responsable(s)** | **Fonctionnalité** |
| 🗆 | Lazhar | Etablir une connexion client / serveur avec échange de message |
| 🗆 | Lazhar | Enregistrer une partie sur le serveur d’enregistrement. |
| 🗆 | Lazhar | Voir les parties inscrites sur le serveur d’enregistrement |
| 🗆 | Lazhar | « Désenregistrer » d’une partie sur le serveur d’enregistrement. |
| 🗆 | Lazhar | Mettre à jour les informations d’une partie. |
| 🗆 | Aurélien | Interface graphique pour l’enregistrement de la partie |
| 🗆 | Aurélien | Interface graphique pour voir les parties inscrites |

## Itération 2 – Serveur de Jeu + Architecture

**Durée :** 1 semaine - 7 mai 2010 au 14 mai 2010

**Pourquoi cette itération prend-t-elle place ici ?**

On prépare tous les éléments pour les fusionner ensuite (avec adaptations quasis certaines)

**Création de l’application client / serveur pour le jeu. En parallèle nous commencerons la restructuration de l’architecture pour correspondre à un jeu multi-joueurs.**

Il s’agit de mettre place (sans interface) une communication entre un joueur et le serveur de Jeu.

**Résultat :** Un protocole de communication mis en place pour l’échange de message ente le client et le serveur de jeu. Concernant la restructuration, on attend un mode 1 joueur avec exactement les mêmes fonctionnalités mais avec une architecture beaucoup plus propre.

|  |  |  |
| --- | --- | --- |
| **Fonctionnalités Attendues** | | |
| **Ok** | **Responsable(s)** | **Fonctionnalité** |
| 🗆 | Pierre-Dominique & Romain | Implémentation de tous les messages fournis par le protocole. (liste trop exhaustive pour les citer tous, *se référer au protocole en annexe*) |
| 🗆 | Aurélien | Le mode 1 joueur fonctionne correctement et comme avant. Présentation du schéma de classe ou de domaine. |

## Itération 3 –Intégration du serveur de jeu + Interface du Jeu en réseau

**Durée :** 2 semaines - 14 mai 2010 au 28 mai 2010

**Intégration du serveur dans l’architecture et le jeu fonctionne.**

Le serveur de jeu devra être intégré à l’architecture de l’application (au noyau du jeu). Le client et le serveur pourront alors interagir avec le modèle (point de vue MVC) du jeu. L’interface du jeu permettra d’illustrer ces changements.

**Pourquoi cette itération prend-t-elle place ici ?**

Il est temps de faire fusionner tous les éléments et en faire un programme plus cohérent.

**Résultat :** A la fin de cette itération, le jeu doit fonctionner et tous les messages transitant entre le client et le serveur doivent être correctement traités pas l’entité réceptrice.

|  |  |  |
| --- | --- | --- |
| **Fonctionnalités Attendues** | | |
| **Ok** | **Responsable(s)** | **Fonctionnalité** |
| 🗆 | Aurélien | L’interface permet de solliciter des actions du jeu. |
| 🗆 | Aurélien | Le client peut se connecter à une partie de jeu. |
| 🗆 | Pierre-Dominique & Romain | Le serveur et le client interprètent les messages réseau et modifie correctement le model. Ceci est visible grâce au changement du terrain de jeu. |

## Itération 4 – Lifting de la GUI + Game Design + Amélioration Mode Solo

**Durée :** 1 semaine - 28 mai 2010 au 4 juin 2010

**Revoir le design et faire de notre logiciel un « vrai » jeu vidéo**

Actuellement, au niveau de l’interface graphique, notre jeu ressemble plus un à logiciel applicatif qu’a un jeu vidéo. Nous aimerions dans cette itération rendre notre jeu plus attractif en créant une interface plus agréable. Il serait aussi intéressant de concevoir nos propre ressources (images / sons / etc.) car actuellement, une bonne partie de nos images sont reprises d’autres jeux.

Une bonne chose serait de revoir également toutes les valeurs liées au jeu pour le rendre plus « jouable » (*level design*). Cette partie peut paraître bénigne mais elle est cruciale et très complexe pour ce genre de jeu car il y a énormément d’éléments qui influencent la durée de vie du joueur.

Nous voulons également implémenter un système de progression dans le mode solo pour que le joueur ressente l’envie de finir complètement le jeu.

**Résultat :** Un programme plus esthétique, plus jouable et avec un système de progression.

|  |  |  |
| --- | --- | --- |
| **Fonctionnalités Attendues** | | |
| **Ok** | **Responsable(s)** | **Fonctionnalité** |
| 🗆 | Lazhar | Lifting de l’interface, celle-ci ressemble plus à un jeu. |
| 🗆 | Pierre-Dominique & Romain | Adaptation des valeurs, jeu plus agréable. |
| 🗆 | Aurélien | Système de progression mise en place. |

## Itération 5 – Serveur Web (facultatif)

**Durée :** moins d’une semaine - 4 juin 2010 au 9 juin 2010

**Serveur web de stockage des meilleurs scores**

Il s’agit de mettre en place un serveur de web fournissant un service web de sauvegarde et récupération des meilleurs scores pour les différents terrains de jeu. Le but étant de motiver le joueur à s’améliorer (il doit avoir envie de rejouer le plus souvent possible).

**Résultat :** Un système permettant de sauver et voir les meilleurs scores de tous les joueurs (du monde).

|  |  |  |
| --- | --- | --- |
| **Fonctionnalités Attendues** | | |
| **Ok** | **Responsable(s)** | **Fonctionnalité** |
| 🗆 | Romain | Mise en place du serveur web |
| 🗆 | Aurélien | Intégration du service dans l’application |