



Cours	: GEN
Professeur	: Eric Lefrançois
Assistant	: Christophe Greppin
Date	: 6 mai 2010

Groupe

Jonas Berdoz (JBZ)
Fabrizio Beretta Piccoli (FBP)
Valentin Delaye (VDE)
Michaël Sandoz (MSZ)

Table des matières

1	Introduction	1
2	Règles du jeu	1
3	Objectifs de base	2
3.1	<i>Objectifs optionnels</i>	2
4	Partage des responsabilités Serveur/Client	2
4.1	<i>Serveur</i>	2
4.2	<i>Client</i>	3
5	Rôle des participants	3
6	Annexes	4

1 Introduction

Dans le cadre du cours de Génie Logiciel (GEN), nous devons réaliser un projet d'une application client-serveur de préférence sur un jeu.

Une fois notre groupe formé, nous avons choisi de réaliser un jeu de tir en réseau en 2D.

Plusieurs joueurs se retrouvent sur une carte et doivent s'éliminer. Les joueurs peuvent se connecter à un serveur et participer à une partie.

Par ce projet, nous désirons mettre en place les méthodes vues au cours pour la gestion de projet et aussi acquérir de nouvelles capacités dans le langage Java.

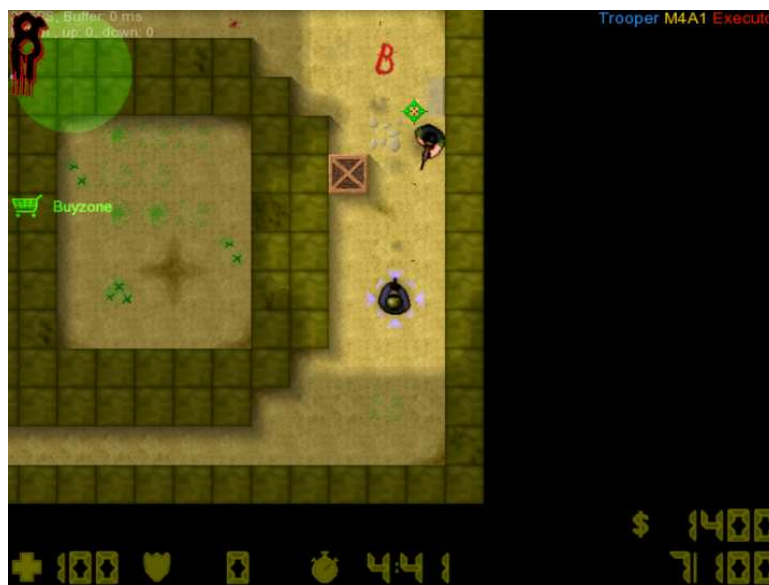


Figure 1 : Counter Strike 2D
Exemple d'un jeu de combat en réseau

2 Règles du jeu

Un joueur dispose de 100 points de vie au début du jeu et d'une arme de base.

Les armes ont une portée de dégât fixe, par exemple l'arme de base peut blesser de 40 points de vie.

Si deux joueurs sont dans la même équipe ils ne peuvent pas se blesser.

Quand un joueur n'a plus de points de vie, il meurt et doit attendre la fin de la partie pour pouvoir recommencer à jouer.

Lorsqu'une partie est terminée, le serveur peut décider de soit recommencer la partie avec les mêmes paramètres soit de quitter et ainsi détruire celle-ci.

3 Objectifs de base

Voici une liste des options que nous désirons fonctionnelles pour la présentation du projet.

- Une seule arme utilisable
- Une seule carte possible pour jouer

3.1 Objectifs optionnels

Voici une liste des options que nous aimerions pouvoir implémenter mais qui ne seront certainement pas opérationnelles pour la présentation.

- Plusieurs armes possibles
- Plusieurs cartes possibles pour jouer une partie
- Possibilité de récupérer des objets sur la carte pour se soigner ou augmenter la puissance de tir.

4 Partage des responsabilités Serveur/Client

Il y aura deux applications pour jouer à notre jeu, une application serveur et une cliente.

L'application serveur permettra de gérer des parties et l'application cliente de jouer sur une partie créée par le serveur.

Un compte admin permettra de gérer les comptes des joueurs qui se connectent au jeu ainsi que leurs scores. Il pourra donc modifier les enregistrements de la base de données.

Lorsqu'on se connecte avec un compte admin ces options apparaissent en plus des options de bases du client.

Les comptes et les scores sont gérés dans la base de données du serveur.

Nous décrivons ici les parties spécifiques au serveur et au client.

4.1 Serveur

Le serveur centralise toutes les parties. Chaque joueur (client) connecté au serveur peut choisir de créer une nouvelle partie.

Le nombre de joueurs variera entre 2 et 8. Le serveur choisit l'un des 2 modes suivants :

- Deux équipes (rouge, bleu) : lorsqu'un autre joueur se connecte, il peut choisir dans quelle équipe il veut jouer. Il peut être forcé à jouer dans une équipe si l'autre est pleine
- Chacun pour soi : il n'y a pas d'équipe et le but est d'être le dernier survivant.

Dans chacun de ces modes le joueur qui a créé la partie décide quand celle-ci démarre avec au minimum 2 joueurs.

Une option (cachée pour les utilisateurs qui ne sont pas administrateurs) permet à un administrateur de modifier ou supprimer des comptes.

4.2 Client

Le client sera l'application qui permettra à un utilisateur de jouer. Il devra tout d'abord se connecter à un serveur, créer un compte pour pouvoir ensuite jouer au moyen de ce compte qui sert d'identification. Ce compte permettra également de garder un historique des scores du joueur sur le serveur.

Ensuite le jeu démarrera et chaque joueur sera positionné sur une carte. Son but est de tuer les membres de l'équipe adverse. Il pourra se déplacer au moyen des touches directionnelles et changer son angle de tir en bougeant sa souris. L'utilisateur pourra tirer en cliquant sur la souris (click gauche).

Lorsque le client est tué, il doit attendre la fin de la partie pour pouvoir rejouer sur ce serveur ou il peut se déconnecter et soit créer une nouvelle partie soit se connecter à un autre serveur.

Si un client se déconnecte pendant une partie, il est considéré comme mort et son score est enregistré dans la base de données du serveur.

Si le serveur se déconnecte pendant une partie, un message apparaît à l'écran des clients et ceux-ci se retrouve au début du programme où ils ont le choix de créer une partie ou de rejoindre une partie. Les scores des clients sont également inscrits dans la base de données du serveur.

5 Rôle des participants

Rôle	Participants	Description
Responsables des conventions	Fabrizio Beretta Piccoli	<ul style="list-style-type: none"> Faire appliquer les normes de codage
Représentants des utilisateurs	Eric Lefrancois Christophe Greppin	<ul style="list-style-type: none"> Client final
Chef de projet	Valentin Delaye	<ul style="list-style-type: none"> Suivre l'avancement et avoir une vue globale sur le projet. Vérification des itérations
Analyste	Jonas Berdoz Fabrizio Beretta Piccoli Valentin Delaye Michaël Sandoz	<ul style="list-style-type: none"> Analyser les besoins Collecte des demandes de changement
Architecte	Jonas Berdoz Fabrizio Beretta Piccoli Valentin Delaye Michaël Sandoz	<ul style="list-style-type: none"> Concevoir l'architecture des différents modules
Programmeur	Jonas Berdoz Fabrizio Beretta Piccoli Valentin Delaye Michaël Sandoz	<ul style="list-style-type: none"> Conception des différents artefacts. Tests unitaire Codage
Responsable des tests fonctionnel	Michaël Sandoz	<ul style="list-style-type: none"> Intégration continue des composants Tests fonctionnels
Responsable de configuration	Jonas Berdoz	<ul style="list-style-type: none"> Gestion des releases Responsable SVN

6 Annexes

- Plan d'itération
- Planification
- Cas d'utilisation
- Scénario
- Interfaces utilisateurs
- Protocole réseau client-serveur
- Modèle de domaine