

Génie Logiciel

Mini-projet Bomberman

Rapport intermédiaire

Groupe

Julien Leroy

Loïc Serafin

Adriano Ruberto

Matthieu Villard

Table des matières

- [1. Fonctionnement général de l'applcatif](#)
 - [1.1. Objectifs de base](#)
 - [1.2. Utilisation de l'applcatif](#)
 - [1.3. Règle du jeu](#)
 - [1.4. Contraintes](#)
- [2. Responsabilités client/serveur](#)
 - [2.1. Client](#)
 - [2.2. Serveur](#)
- [3. Cas d'utilisation](#)
 - [3.1. Diagramme général de contexte](#)
 - [3.2. Description des acteurs](#)
 - [3.3. Scénario « principaux »](#)
- [4. Protocole d'échange client/serveur](#)
 - [4.1. Amorce de connexion](#)
 - [4.2. Création de compte](#)
 - [4.3. Authentification](#)
 - [4.4. Déconnexion](#)
 - [4.4. Créer une salle](#)
 - [4.5. Rejoindre une salle](#)
 - [4.6. Quitter une salle](#)
 - [4.7. Mise à jour des salles](#)
 - [4.8. Mise à jour de la salle](#)
 - [4.9. Lancement d'une partie](#)
 - [4.10. Déplacement, Poser une bombe](#)
 - [4.11. Fin de partie](#)
 - [4.12. Statistiques](#)
- [5. Ébauche du modèle de domaine](#)
 - [5.1. Modèle pour le serveur](#)
 - [5.2. Modèle pour le client](#)
- [6. Base de données](#)
 - [6.1. Objectifs](#)
 - [6.2. Modèle conceptuel](#)
- [7. Rôle des participants](#)
- [8. Plan d'itérations](#)
 - [8.1. Itération 1 - 18.04](#)
 - [8.2. Itération 2 - 25.04](#)
 - [8.3. Itération 3 - 02.05](#)
 - [8.4. Itération 4 - 09.05](#)
 - [8.5. Itération 5 - 16.05](#)
 - [8.6. Itération 6 - 23.05](#)
 - [8.7. Itération 7 - 30.05](#)
 - [8.8. Itération 8 - 06.06](#)

1. Fonctionnement général de l'applcatif

1.1. Objectifs de base

Dans le cadre du cours Génie Logiciel, il nous a été demandé de concevoir et réaliser une application client/serveur. Nous avons alors décidé de réaliser le jeu "Bomberman" en Java.

1.2. Utilisation de l'applcatif

Le client est d'abord invité à se connecter. S'il n'a pas de compte, il a la possibilité d'en créer un. Une fois connecté, il arrive sur le salon avec toutes les parties créées. Il peut ensuite rejoindre une salle non en cours ou en créer une.

Lors de la création d'une partie, le joueur est invité à nommer la salle, d'y associer un mot de passe s'il le souhaite et de mettre le nombre minimum de joueur [2-4] avant de commencer. Il se connecte ensuite à cette salle et attend que d'autres joueurs le rejoignent. Dès que le nombre minimum de joueurs ou plus sont connectés dans la salle, un compte à rebours démarre à la fin duquel la partie commence. Il n'est pas possible pour un joueur de rejoindre une partie en cours. Pour créer une partie, un joueur dispose de cartes prédéfinies et n'a donc pas la possibilité d'en créer de nouvelles.

Lorsqu'une partie commence, chaque joueur est placé dans un coin de la carte. C'est pourquoi le nombre de joueurs est limité à quatre. Une déconnexion du joueur implique la mort de ce dernier (et donc le dernier au classement).

Le client peut regarder ses statistiques de jeu, telles que son temps de jeu, son nombre de parties gagnées ou perdues, le nombre de personnes qu'il a tué, son meilleur temps de survie avec la moyenne et le nombre total de bombes posées.

Le client a la possibilité de modifier ses informations telles que son pseudonyme et son mot de passe.

L'administrateur a, en plus de tout ce que peut faire un client, le droit de tricher en éjectant un joueur d'une partie ou de le bannir complètement, de se rajouter des power-ups ou de carrément gagner la partie.

1.3. Règle du jeu

Le "Bomberman" se joue généralement à 4 joueurs dans une arène fermée. Le but étant d'être le dernier survivant d'une lutte sans merci. Pour éliminer ses opposants, les joueurs ont la possibilité de se déplacer ainsi que de poser des bombes, qui exploseront après un certain délai. Ces bombes laissent une déflagration qui se répand dans quatre directions en ligne droite.

Les bombes peuvent, en plus d'exploser les joueurs, casser des éléments destructibles de l'arène afin de faire tomber des bonus. Ces bonus sont récupérables par tous les joueurs présents encore dans l'arène et peuvent donner des avantages tels que poser plus de bombes, avoir des bombes avec une déflagration plus grande ou encore augmenter la vitesse de déplacement du joueur.

La partie se termine quand il ne reste plus qu'un joueur ou que tout le monde est mort. Des statistiques sont alors affichées sur le nombre de personnes que chaque joueur a tuées ainsi que leurs temps de survie. Ces données sont rajoutées aux statistiques globales des joueurs, où ils peuvent voir leurs performances de toutes leurs parties jouées.

1.4. Contraintes

Avoir un temps de réponse client/serveur le plus bas possible.

2. Responsabilités client/serveur

2.1. Client

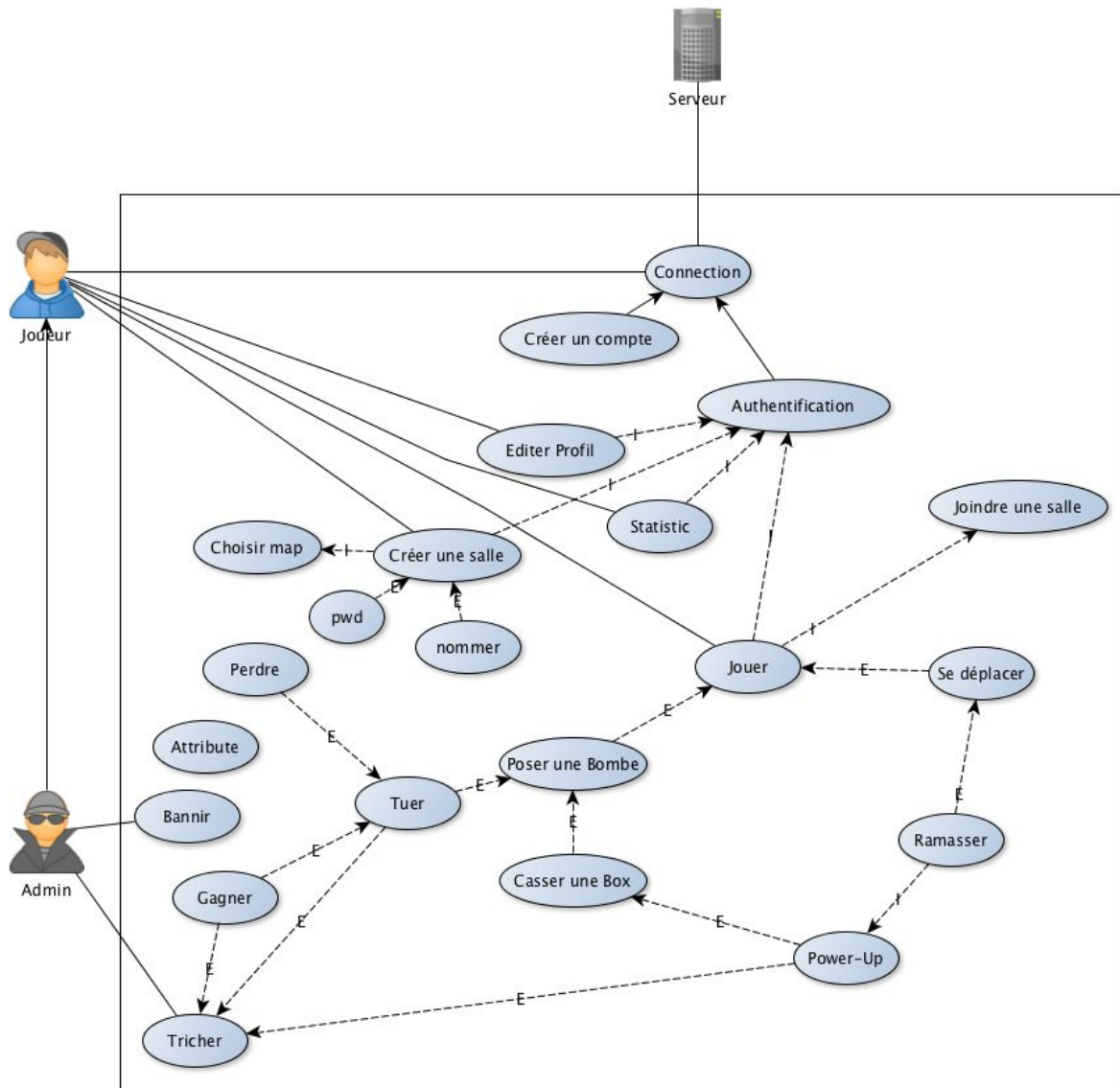
Le client est utilisé par les joueurs. Il est responsable d'afficher le jeu et de communiquer avec le serveur qui gère les parties. Il offre une page de connexion, de création de compte et d'affichage de statistiques de parties également.

2.2. Serveur

Le serveur est responsable de gérer la validité de l'authentification des utilisateurs, de stocker les statistiques globales et particulières à chaque joueur et de maintenir un nombre de salles disponibles pour lancer des parties. Il gère également le déroulement de toutes les parties en cours. Il reçoit toutes les entrées émises par les joueurs et vérifie la validité de celles-ci. Il prévient ainsi des actions non permises par les joueurs.

3. Cas d'utilisation

3.1. Diagramme général de contexte



3.2. Description des acteurs

- Joueur : Acteur principal dans : connexion, éditer son profil, visualiser les statistiques, créer une salle, jouer
- Serveur : Acteur secondaire dans connexion, créer une salle, jouer.
- Administrateur: Acteur principal dans bannir et tricher.

3.3. Scénario « principaux »

- Le joueur peut se connecter, si il a déjà un compte d'utilisateur, il peut s'authentifier, si non il peut en créer un. Pour se connecter il a besoin d'un mot de passe et d'un pseudo.
- Dès le moment ou il est connecté il peut éditer son compte, voir les statistiques de jeu, créer une salle, rejoindre une salle déjà créée par un autre joueur.
- Quand le joueur est dans le jeu, il peut se déplacer dans les quatre directions, et en se déplaçant sur les power-up, il les ramasse. Le joueur peut poser des bombes, qui en explosant détruisent des box, des power-up ou des joueurs.

4. Protocole d'échange client/serveur

4.1. Amorce de connexion

Au lancement de l'application client, une connexion via un socket est ouverte avec le serveur sur le port 3737.

4.2. Création de compte

Le client envoie une demande de création de compte au serveur.

Le serveur renvoie une nouvelle clé publique de cryptage au client.

L'utilisateur fournit ses données personnelles dont son mot de passe qui sera d'abord haché puis crypté avec la clé reçue, avant d'envoyer ces données au serveur.

Le serveur décrypte le mot de passe et stocke toutes les informations dans sa base de données.

4.3. Authentification

Le client envoie une demande d'authentification au serveur.

Le serveur renvoie une nouvelle clé publique de cryptage au client.

L'utilisateur fournit ses informations de connexion, puis hache et crypte son mot de passe avant de les envoyer au serveur.

Le serveur décrypte le mot de passe et compare le hash avec celui correspondant dans la base de données. Si le mot de passe correspond, envoie au client une liste de salles disponibles pour pouvoir jouer ainsi qu'une nouvelle clé de cryptage. Sinon, envoie un message d'erreurs et invite à retenter la connexion.

4.4. Déconnexion

Le client envoie une demande de déconnexion au serveur.

Le serveur ferme les threads et sockets correspondants.

4.4. Créer une salle

Si dans la liste des salles disponibles envoyées par le serveur, certaines sont marquées comme vides, alors le client a la possibilité de créer une salle.

Le client entre le nom de la salle, le nombre de joueurs minimum et un mot de passe si désiré. Si il y a un mot de passe, il est crypté avec la clé reçue après l'authentification avant qu'il soit envoyé avec le nom de la salle au serveur.

Le serveur enregistre la demande et met à jour sa liste de salles disponibles.

4.5. Rejoindre une salle

Si dans la liste des salles disponibles envoyées par le serveur, certaines sont marquées comme ouvertes, alors le client a la possibilité de la rejoindre.

Le client envoie alors une demande de connexion à une salle en fournissant l'ID de la salle, et le cas échéant, le mot de passe crypté avec la dernière clé de cryptage reçue.

En cas de réussite, le serveur met à jour le contenu de la salle.

4.6. Quitter une salle

Le client envoie une demande de déconnexion d'une salle.

Le serveur met à jour la salle correspondante et renvoie la liste des salles disponibles.

4.7. Mise à jour des salles

Le serveur envoie de manière périodique à tous les clients connectés qui ne sont ni dans une salle, ni dans une partie en cours la liste des salles disponibles avec les différents changements qu'il y a eu.

Le client a la possibilité de mettre à jour manuellement cette liste en faisant une demande au serveur. Le serveur enverra alors de la même manière cette liste.

4.8. Mise à jour de la salle

En cas de changement d'état d'une salle, le serveur envoie le nouvel état de la salle à toutes les personnes présentes dans celle-ci.

4.9. Lancement d'une partie

Les clients présents dans une salle ont la possibilité de se mettre "prêts" pour lancer la partie. Dans ce cas, le client envoie au serveur cette information.

Le serveur met à jour l'état de la salle.

Si tous les joueurs sont prêts, alors le serveur prépare la partie.

Le serveur envoie aux joueurs la carte choisie et l'état de tous les joueurs.

4.10. Déplacement, Poser une bombe

Le client envoie au serveur toutes les actions effectuées. Les touches W, A, S, D correspondent au déplacement, et la touche Espace correspond à poser une bombe à l'emplacement du joueur.

Pour chacune des interactions, un message est envoyé au serveur avec l'action effectuée. Le serveur met à jour l'état de la partie et renvoie à tous les joueurs la liste de modifications de l'état de la partie.

4.11. Fin de partie

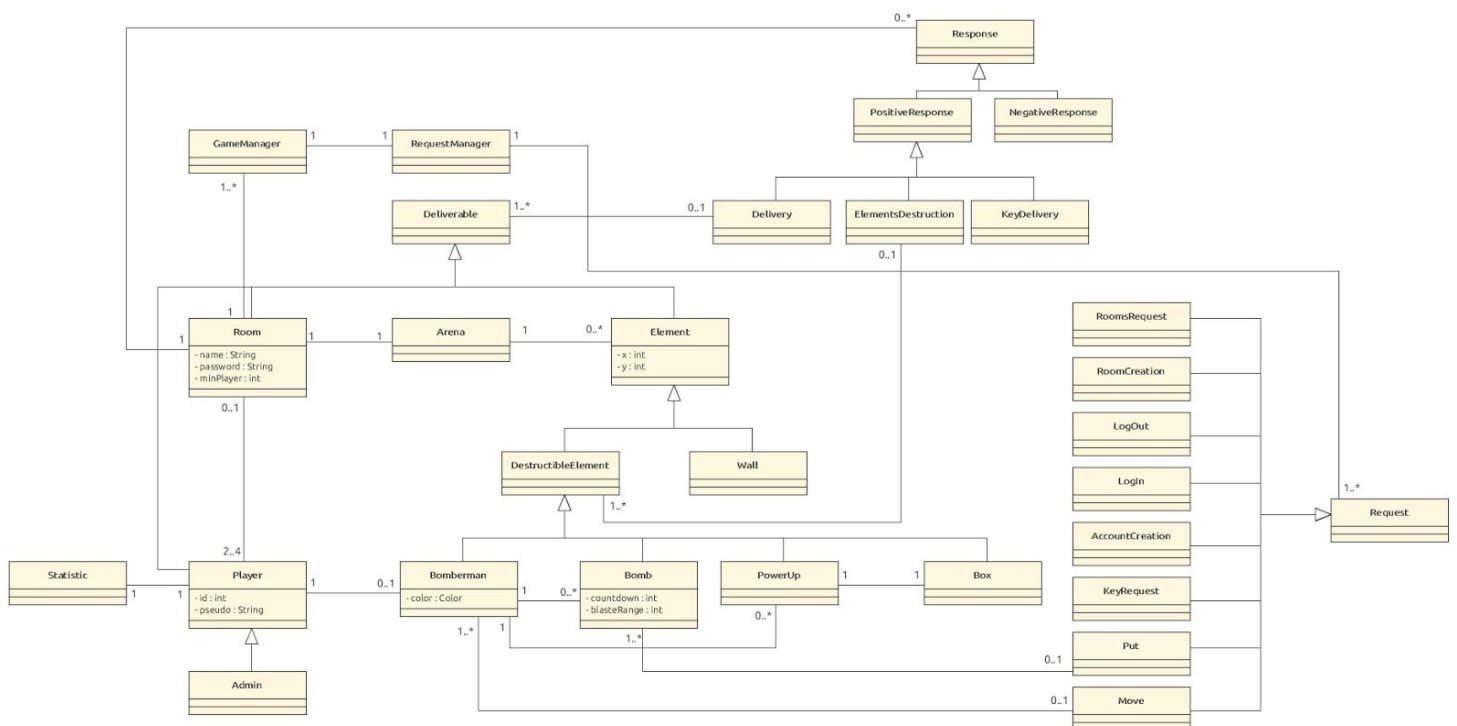
S'il ne reste plus qu'un ou plus aucun joueur, le serveur arrête la partie et envoie aux joueurs les statistiques de la partie.

4.12. Statistiques

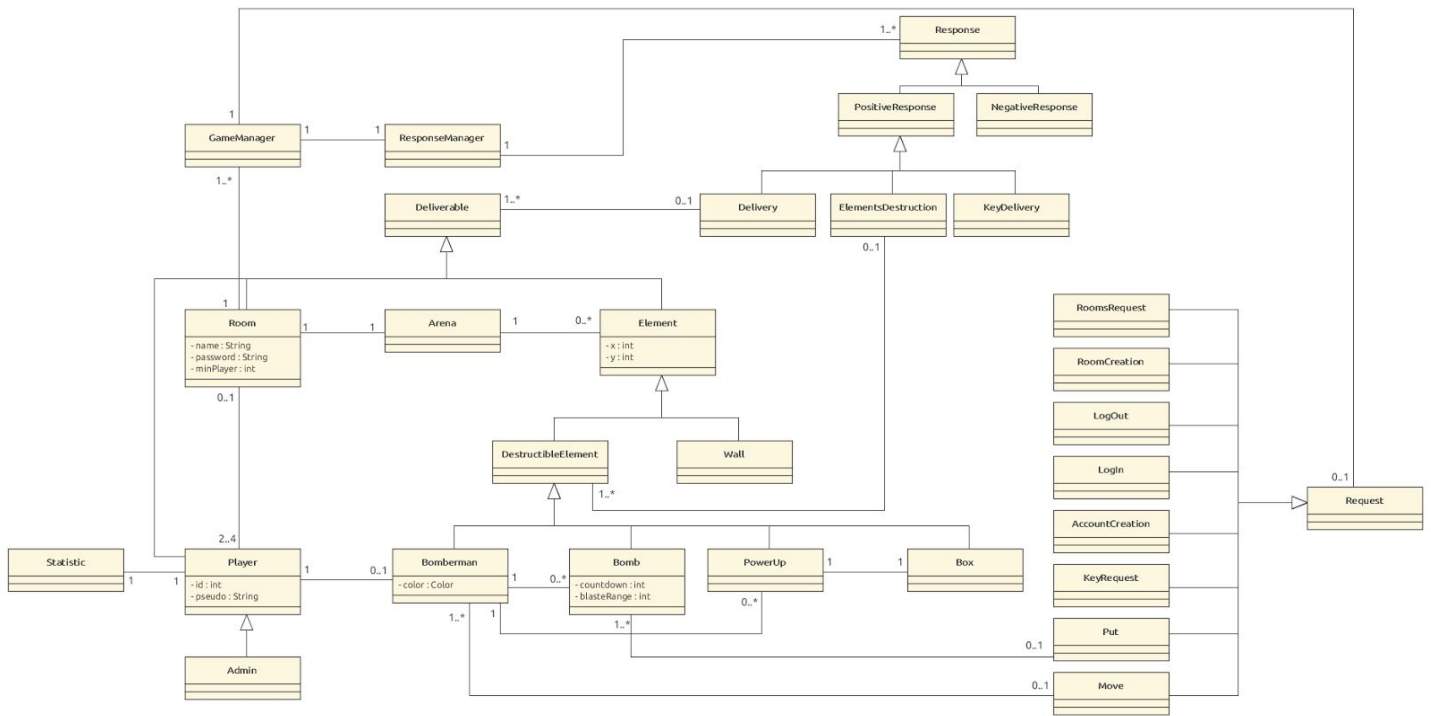
Le client peut demander de voir ses statistiques. Il envoie une requête au serveur qui lui renvoie directement ses statistiques.

5. Ébauche du modèle de domaine

5.1. Modèle pour le serveur



5.2. Modèle pour le client

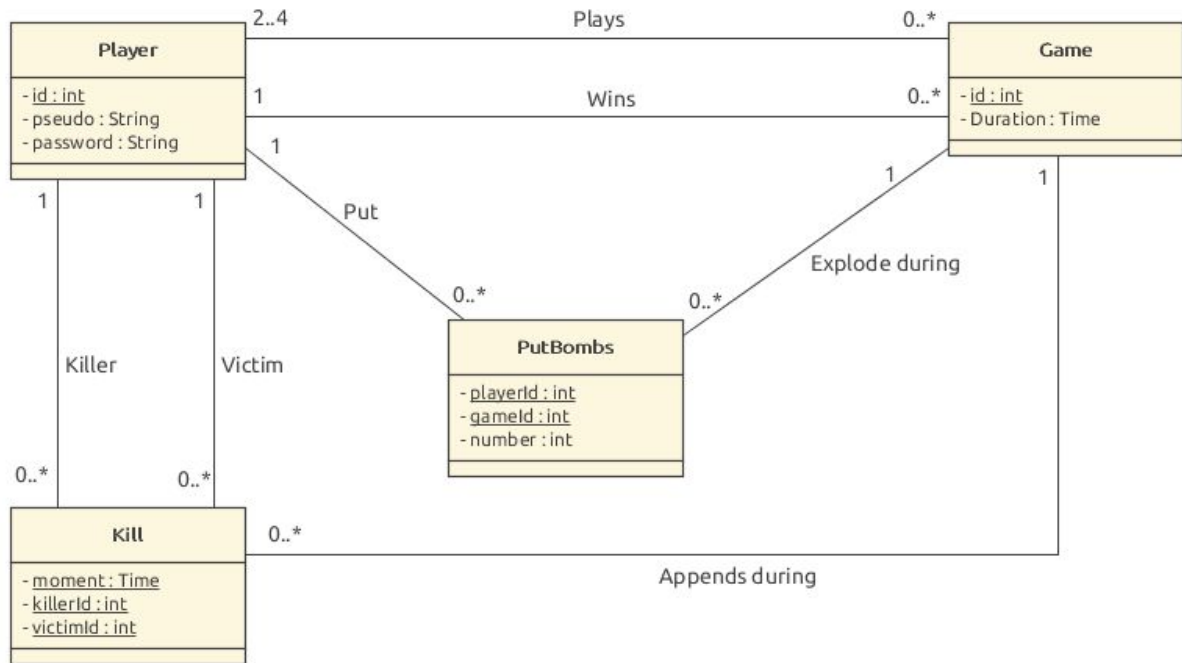


6. Base de données

6.1. Objectifs

L'objectif de la base de données est d'enregistrer les différentes statistiques de chaque partie ainsi que les données de chaque joueur. Ainsi, il devra être possible de stocker les joueurs avec leurs pseudonymes et leurs mots de passe. Il devra également être possible de mémoriser les parties auxquelles ils ont participé en indiquant si ils ont gagné ou perdu. De plus, pour chaque partie, nous devrions être capables de retrouver le nombre de bombes que chaque joueur a posées et déterminer qui a été tué durant cette partie, en spécifiant le joueur qui a infligé cette tragédie et à quelle moment fatidique.

6.2. Modèle conceptuel



7. Rôle des participants

Rôle	Responsabilités	Personne en charge
Représentants des utilisateurs	<ul style="list-style-type: none">• Collecte des besoins• Spécifications des tests de fonctionnalités• Explication des aspects métiers	Leroy
Chef de projet	<ul style="list-style-type: none">• Planification• Coordination avec les utilisateurs	Ruberto
Analyste	<ul style="list-style-type: none">• Spécifications• Collecte des demandes de changement	Serafin
Architecte, concepteur en chef	<ul style="list-style-type: none">• Conception de l'architecture du produit	Leroy
Programmeur	<ul style="list-style-type: none">• Participe à la conception du produit• Écrit les tests unitaires• Codage	Leroy Serafin Ruberto Villard
Responsable des tests	<ul style="list-style-type: none">• Participe à l'intégration continue des composants• Écrit les tests fonctionnels• Met en place l'architecture permettant de lancer régulièrement les tests fonctionnels	Serafin
Responsable de la configuration	<ul style="list-style-type: none">• Gestion de la base des artefacts du projet• Gestion des releases• Allocation des droits• Responsable de la configuration (logicielle & matérielle)• Intégration des changements	Villard

8. Plan d'itérations

Chaque itération sera documentée sur tous les cas d'utilisations implémentés.

8.1. Itération 1 - 18.04

Objectif		Rapport intermédiaire
Durée		1 semaine
Tâche	Leroy	Cas d'utilisation
	Serafin	Protocole client/serveur, Plan d'itérations, Responsabilité client/serveur
	Ruberto	Fonctionnement général, Plan d'itérations, Modèle pour le client
	Villard	Base de donnée, Modèle pour le serveur
Effort consacré en heures		

8.2. Itération 2 - 25.04

Objectif		Connexion, Base de données
Durée		1 semaine
Tâche	Leroy	Serveur
	Serafin	Conception serveur
	Ruberto	Connexion client au serveur
	Villard	Base de données
Effort consacré en heures		

8.3. Itération 3 - 02.05

Objectif		Déplacement
Durée		1 semaine
Tâche	Leroy	Conception interface utilisateur
	Serafin	Logique du jeu
	Ruberto	Logique du jeu

	Villard	Interface utilisateur
Effort consacré en heures		

8.4. Itération 4 - 09.05

Objectif		Créer salle, poser une bombe, casser une box
Durée		1 semaine
Tâche	Leroy	Interface salle
	Serafin	Salle serveur
	Ruberto	Poser une bombe, casser une box
	Villard	Salle client
Effort consacré en heures		

8.5. Itération 5 - 16.05

Objectif		Jouer, Rejoindre une salle
Durée		1 semaine
Tâche	Leroy	Finaliser interface
	Serafin	Gestion jeu côté serveur, dessins
	Ruberto	Conception des maps
	Villard	Gestion jeu côté client
Effort consacré en heures		

8.6. Itération 6 - 23.05

Objectif		Tuer, Perdre, Gagner, Bannir, Tricher
Durée		1 semaine
Tâche	Leroy	Interface gagner/perdre
	Serafin	Tuer
	Ruberto	Perdre, Gagner
	Villard	Bannir, Tricher
Effort consacré en heures		

8.7. Itération 7 - 30.05

Objectif		Power-up, ramasser
Durée		1 semaine
Tâche	Leroy	Conception power-up
	Serafin	Dessins power-up
	Ruberto	Implémentation des power-up
	Villard	Implémentation des power-up
Effort consacré en heures		

8.8. Itération 8 - 06.06

Objectif		Statistiques
Durée		1 semaine
Tâche	Leroy	Interface statistiques
	Serafin	Stockage des statistiques
	Ruberto	Finaliser la documentation
	Villard	Interfaçage des statistiques
Effort consacré en heures		