

Raport de projet Jeu de stratégie au tour par tour

Thomas Doguet 22205490, Romain Moalic, Etienne Bossu, Alexandre Fautpoint

5 décembre 2024

Table des matières

1	Introduction	2
1.1	Contexte	2
1.2	Résumé du projet	2
2	Conception	3
2.1	Architecture du projet	3
2.2	Diagrammes UML	3
3	Organisation du projet	8
3.1	Structure des répertoires	8
3.2	Rôles et répartition des tâches	9
4	Difficultés rencontrées	10
4.1	Problèmes techniques	10
5	Exécution du projet	11
5.1	Utilisation de <code>build.xml</code>	11
5.2	Prérequis	11
6	Conclusion	12
6.1	Résumé du projet	12
A	Annexe	13
A.1	annexes	13

Chapitre 1

Introduction

1.1 Contexte

Le contexte du projet est dans le cadre de la matière "Méthode de Conception" de notre premier semestre de L3 informatique, il consistait a mettre en place un Jeu de Stratégie au tour par tour en mettant en place les concepts vus en cours et durant les travaux pratiques de ce cours.

1.2 Résumé du projet

Le projet consistait de mettre en place un projet en MVC qui se Joue possiblement a deux, qui contient trois vues possibles, la vue du joueur 1 (contenant son inventaire, sa vie, les objets qu'il a caché sur la grille), la vue du joueur 2 (contenant les mêmes choses que pour le joueur 1) et une troisième vue, plus générale, qui contient les information sur les deux joueurs, un jeu qui est jouable grâce aux touches du clavier.

Chapitre 2

Conception

2.1 Architecture du projet

Nous avons donc mi en place une structure en MVC (modèle Vue Contrôleur) :

- **Model** : gestion des données.
 - Modélisation des Joueurs (classe Player)
 - Modélisation de la grille et des Cellules (classes Grid et Cell)
 - Modélisation des armes (Interface Item, implémentée par la classe Abstraite Objet étendue par les classes Bomb, Mine et Gun.
 - Création des Proxys (pour l'inventaire, les Bombes et les Mines(pour les bombes et les mines, un proxy est nécessaire que lorsqu'elles ont été posées).
 - Voir diagramme Uml du modele
- **View** : affichage des données.
 - Affichage de la vue "Générale" dans le Fichier View.
 - Affichage de la vue du Joueur dans le Fichier ViewPlayer.
 - Affichage de la vue Pour le paramétrage des joueurs avant le début de la partie dans le fichier PlayerSelectionView.Voir le diagramme UML de la Vue.
- **Controller** : interaction entre l'utilisateur et les données. Voir le diagramme UML du Controller.

2.2 Diagrammes UML

Incluez vos diagrammes UML ici. Par exemple, utilisez une image exportée de vos diagrammes :

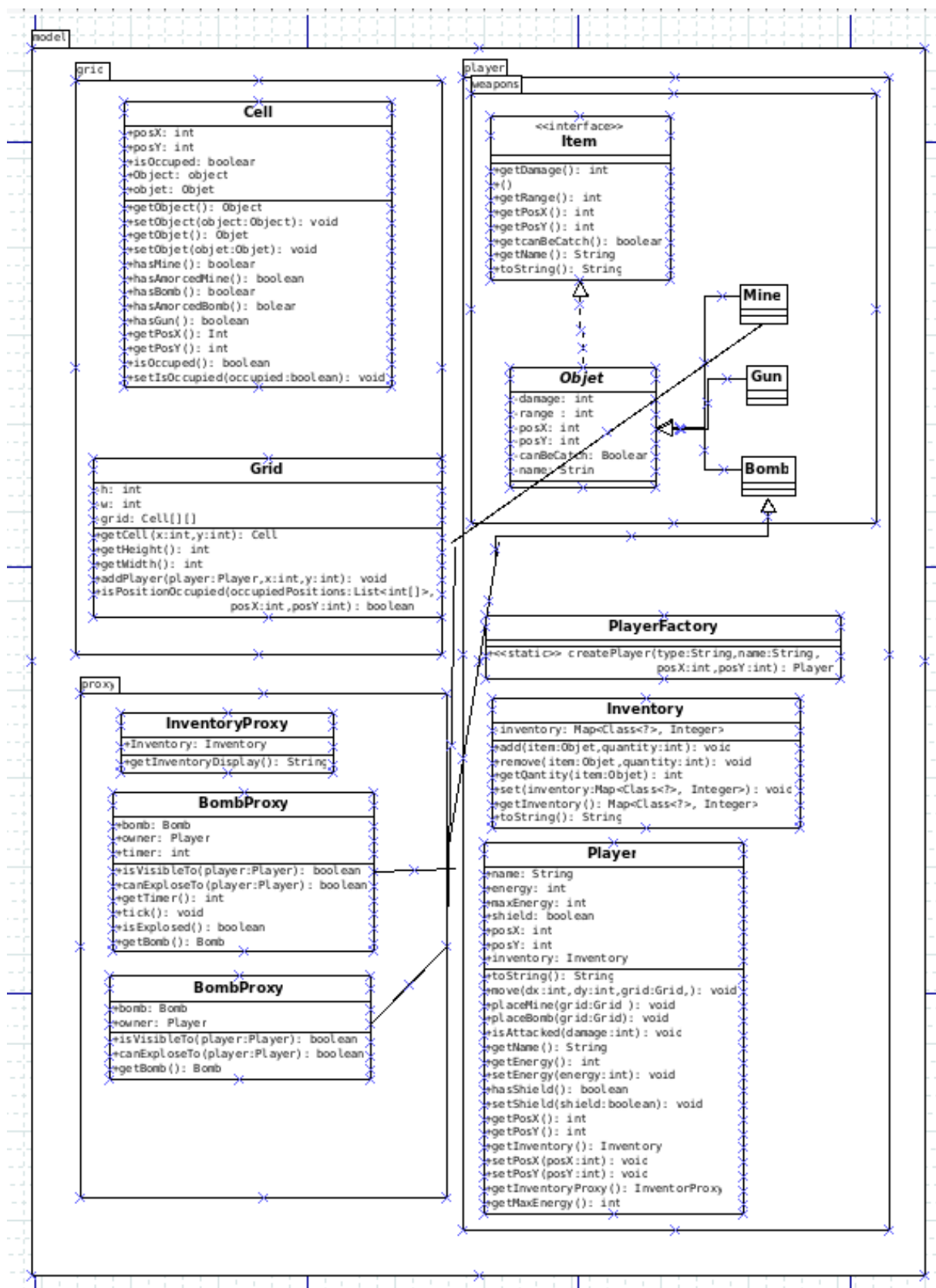


FIGURE 2.1 – Diagramme UML du modèle

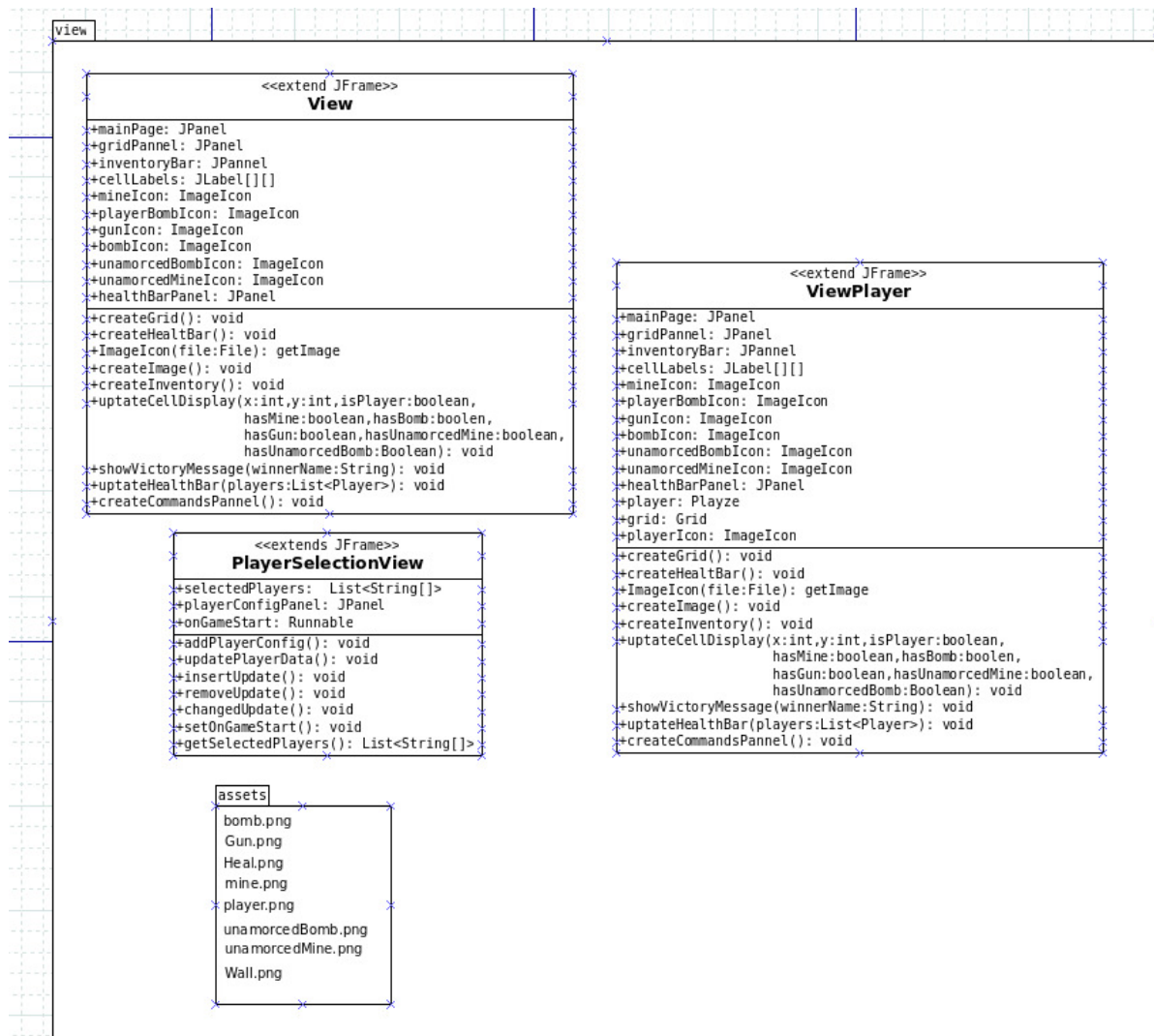


FIGURE 2.2 – Diagramme UML de la vue

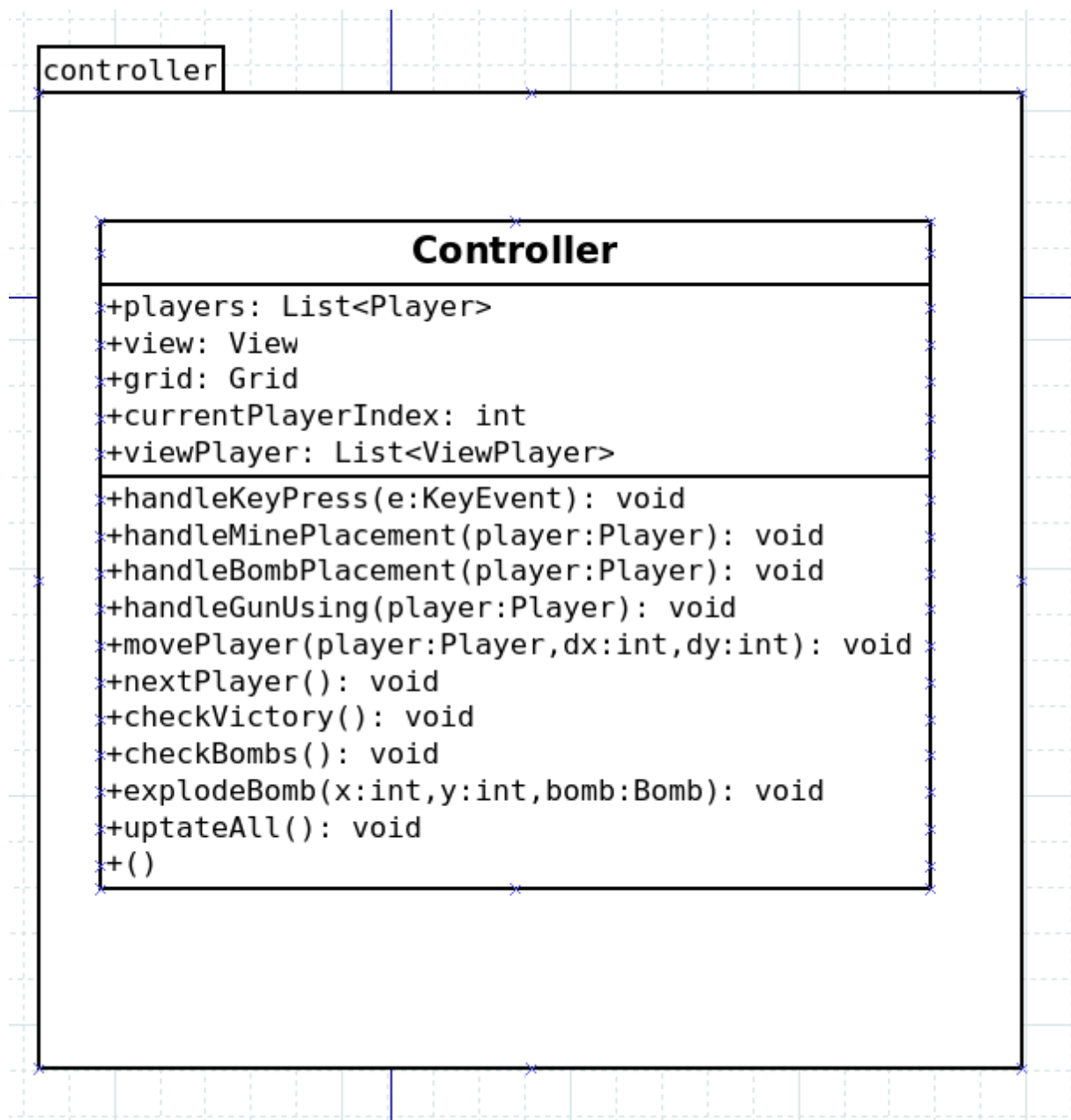


FIGURE 2.3 – Diagramme UML du controller

Chapitre 3

Organisation du projet

3.1 Structure des répertoires

```
src
├── controller
│   ├── Controller.java
│   └── Game.java
├── model
│   ├── grid
│   │   ├── Cell.java
│   │   └── Grid.java
│   ├── player
│   │   ├── Inventory.java
│   │   ├── PlayerFactory.java
│   │   └── Player.java
│   ├── weapons
│   │   ├── Bomb.java
│   │   ├── Gun.java
│   │   ├── Item.java
│   │   ├── Mine.java
│   │   └── Objet.java
│   └── proxy
│       ├── BombProxy.java
│       ├── InventoryProxy.java
│       └── MineProxy.java
└── view
    ├── assets
    │   ├── bomb.png
    │   ├── Gun.png
    │   ├── mine.png
    │   ├── player.png
    │   ├── unamorcedBomb.png
    │   └── unamorcedMine.png
    ├── PlayerSelectionView.java
    ├── View.java
    └── ViewPlayer.java
```

└─ build.xml

3.2 Rôles et répartition des tâches

- Romain a axé principalement sur la création des différentes vues, l'affichage agréable, et les proxys
- Alexandre a d'abord créé l'architecture en MVC, le patterns factories et a et a mis en place les murs et les soins.
- Thomas a mi en place le contrôleur, les actions possibles et la gestion des armes (les ramasser, les utiliser), il a appliqué le principe des armes pour la vue.
- Etienne a mi en place "l'architecture" pour les armes, il a crée les mouvements et les attaques au corps a corps

Chapitre 4

Difficultés rencontrées

4.1 Problèmes techniques

Nous avons rencontrés plusieurs problèmes durant notre Travail.

- Nous avons eu des difficultés a mettre en place le proxy dans un premier temps.
- Dans un second temps nous avons eu des difficultés dans la mise en place des différentes vues.

Chapitre 5

Exécution du projet

5.1 Utilisation de build.xml

Expliquez comment exécuter le projet avec Ant. Par exemple :

1. Nettoyer le projet :

```
ant clean
```

2. Compiler le projet :

```
ant compile
```

3. Lancer l'application :

```
ant run
```

5.2 Prérequis

Listez les prérequis pour exécuter le projet :

- Java JDK (version 8 ou ultérieure).
- Apache Ant.
- Système d'exploitation compatible (Linux, Windows, MacOS).

lstlisting

Chapitre 6

Conclusion

6.1 Résumé du projet

Notre objectif initial était de mettre en place un jeu de stratégie Tour par Tour en MVC en utilisant les principes Vus en cours, ce que l'on a réussi à mettre en place avec succès

Annexe A

Annexe

A.1 annexes

Nous avons utilisés différentes images pour la mise en place de la vue

Liste des images

- bomb.png
- Gun.png
- mine.png
- player.png
- unamorcedBomb.png
- unamorcedMine.png