

# Projet Analyse et Conception Logiciel

## M1 - Info



*Dungeon Crypt*

BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

## Table des matières :

<b>1) Vision du produit</b>	<b>4</b>
<b>2) BackLogs</b>	<b>5</b>
Sprint 0 : Lancement	5
Sprint 0 : Review	6
Sprint 0 : Rétrospective	6
Sprint 1: Lancement	7
Sprint 1 : Review	11
Sprint 1 : Rétrospective	13
Sprint 2 : Lancement	15
Sprint 2 : Review	18
Sprint 2 : Rétrospective	19
Sprint 3 : Lancement	21
Liste des fonctionnalités	22
Salles	22
1) Créer la salle de départ du jeu et son affichage (Version terminal)	22
2) Créer un générateur de salle simple	22
3) Créer une sortie vers la salle suivante	22
4) Certaines cases du labyrinthe sont spéciales :	22
• Créer des trésors :	22
• Créer des pièges :	22
5) Améliorer la génération de salle	22
Héros	23
6) Créer, placer, déplacer le héros	23
7) Le héros peut infliger des dégâts aux monstres	23
8) Le héros peut utiliser des bonus	23
9) Recul lorsque le joueur subit des dégâts	23
Monstres	23
10) Créer un monstre simple, se déplace aléatoirement	23
11) Les monstres infligent des dégâts aux joueurs	23
12) Créer différent type de monstre au comportement différents	23
• Squelette	23
• Zombie	23
• Fantôme	23
• Boss	23
Collisions	24
13) Gérer les collisions entre le héros, les monstres et les murs	24
14) Recul lorsque les monstres subissent des dégâts	24
Interface Graphique & divers	24
15) Adapter le projet a l'utilisation de LibGDX	24
16) Utiliser Gradle pour l'installation et le lancement du jeu	24
17) Afficher les éléments d'une salle graphiquement (écran)	24

BOHAN Thomas      JACQUES Louis      SERRATORE Alexandre      SUCK Hugo

18) Actualiser l’affichage des entités et modifier leur déplacement	24
19) Ajouter un menu principale	24
20) Afficher et gérer le score	25
21) Afficher et gérer la barre de point de vie	25
22) Gérer la fin du jeu (mort du héros + écran Game Over)	25
23) Pouvoir sauvegarder / Charger une partie	25
24) Ajouter du son / bruitage / musique	25
Correction de bug	25
25) Problème de redimensionnement de caméra	25
Idées	25

# 1) Vision du produit

Dungeon Crawler. Thème Halloween/Horreur.

Un personnage traverse différentes salles avec des monstres à combattre et récupère de l'équipement. Le joueur peut consommer certains objets pour lui attribuer des avantages (exemple : gain de vie).

Lorsque le joueur atteint la sortie finale, il descend d'un étage et les monstres présents sont plus difficiles à combattre.

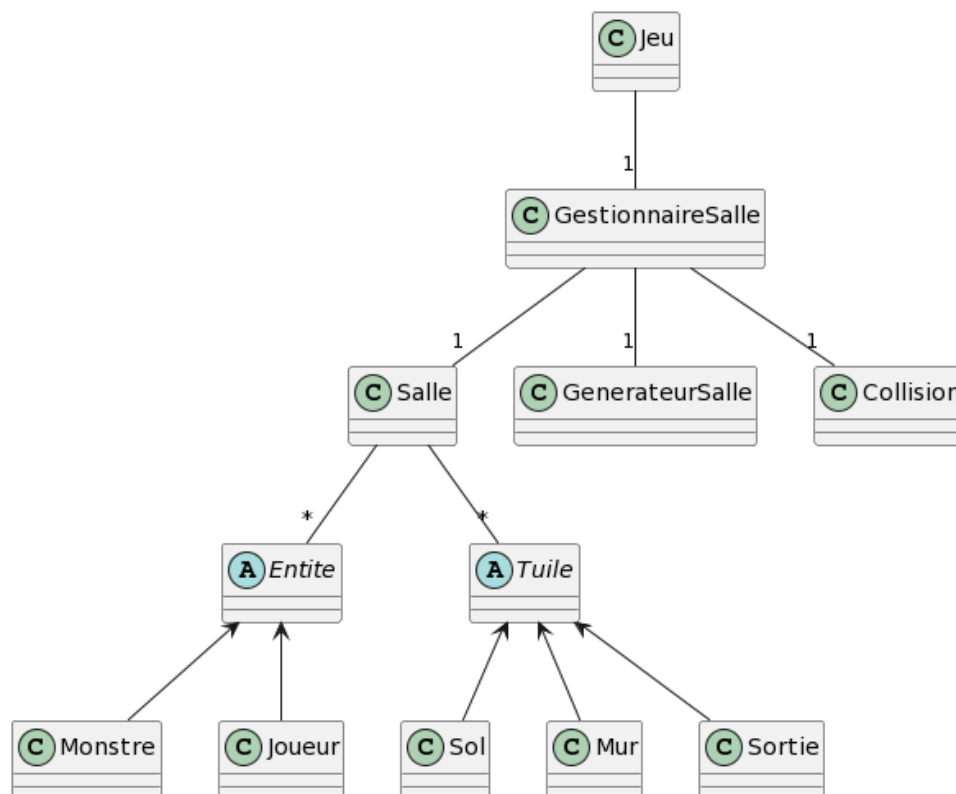
Des points lui sont attribués pour chaque monstre qu'il tue, certains équipements qu'il récupère et chaque étage qu'il descend.

## 2) BackLogs

### Sprint 0 : Lancement

Liste des tâches	Responsable
<a href="#">Créer la salle de départ du jeu et son affichage (Version terminal)</a>	Hugo
<a href="#">Créer un générateur de salle simple (Version terminal)</a>	Hugo
<a href="#">Créer, placer, déplacer le héros</a>	Thomas
<a href="#">Créer un monstre simple, se déplace aléatoirement</a>	Thomas
<a href="#">Gérer les collisions entre le héros, les monstres et les murs</a>	Alexandre
<a href="#">Créer une sortie vers la salle suivante</a>	Louis

### Diagramme de classe



BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

## Sprint 0 : Review

Liste des tâches	Validé	Commentaires
<a href="#">Créer la salle de départ du jeu et son affichage (Version terminal)</a>	Validé	Réalisé sans soucis
<a href="#">Créer un générateur de salle simple (Version terminal)</a>	Validé	Implémenté pour faciliter l'ajout de patterns et de forme de génération
<a href="#">Créer, placer, déplacer le héros</a>	Validé	Réalisé sans soucis
<a href="#">Créer un monstre simple, se déplace aléatoirement</a>	Validé	Les monstres n'ont aucun intérêt pour l'instant.
<a href="#">Gérer les collisions entre le héros, les monstres et les murs</a>	Validé	Collisions simples pour l'affichage via terminal. A modifier dans le futur. Suppression de la classe Collision.
<a href="#">Créer une sortie vers la salle suivante</a>	Validé	Réalisé sans soucis

## Sprint 0 : Rétrospective

Ce qui s'est bien passé (ou pas) et les décisions pour le sprint suivant :

- Réalisation d'une architecture solide du projet pour faciliter l'implémentation graphique.
- Bonne organisation et communication via le git et trello. Chaque personne notifiait les autres membres du groupe de l'avancée du travail. Chaque membre était constamment au courant de quelle fonctionnalité est terminée et quelle fonctionnalité est en cours et par qui.
- Le travail a été commencé et terminé tôt, ce qui a permis de corriger les éventuels bugs rapidement.

Problèmes :

BOHAN Thomas

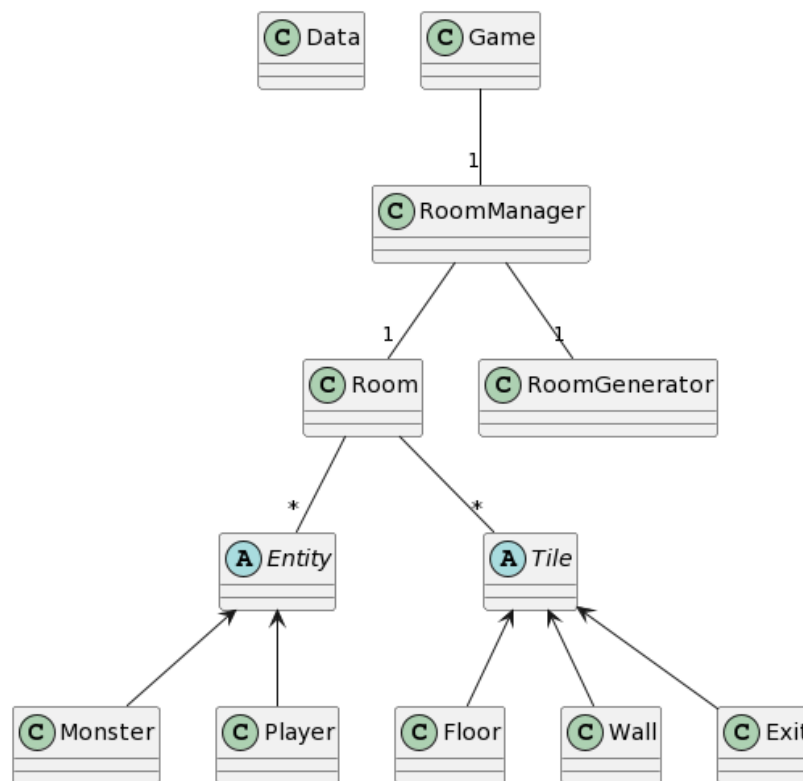
JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

- Il y a cependant eu une mauvaise utilisation du système de branche sur git.  
**Décision prise :** Créer et utiliser des branches pour chaque fonctionnalité lors du prochain sprint et utiliser les pull request afin de vérifier le code des autres intervenants avant la merge avec le reste du projet.
- Un problème a eu lieu au niveau de la compilation du projet pour générer le fichier JAR. Des problèmes causés par un souci dans l'ordonnancement du projet et une non détection du dossier de ressources. Pour ce cas exceptionnel, nous avons généré le fichier JAR via IntelliJ.  
**Décision prise :** Pour le sprint suivant nous aurons les lignes de commandes claires et fonctionnelles pour générer et lancer le projet depuis les fichiers sources. Ceci sera effectué en début de sprint 1.

Diagramme de classe (actualisé) :



## Sprint 1: Lancement

Liste des tâches	Responsable
<a href="#">Adapter le projet a l'utilisation de LibGDX</a>	Tous
<a href="#">Utiliser Gradle pour l'installation et le lancement du jeu</a>	Alexandre
<a href="#">Afficher les éléments d'une salle graphiquement (écran)</a>	Thomas
<a href="#">Afficher et gérer la barre de point de vie</a>	Louis
<a href="#">Les monstres infligent des dégâts aux joueurs</a>	Alexandre
<a href="#">Gérer les collisions entre le héros, les monstres et les murs</a> (Adapté à la librairie utilisée)	Alexandre
<a href="#">Le héros peut infliger des dégâts aux monstres</a>	Louis
<a href="#">Actualiser l'affichage des entités et modifier leur déplacement</a>	Thomas
<a href="#">Créer différent type de monstre au comportement différents</a> (Zombie, Fantômes)	Fantôme : Louis Zombie: Hugo
<a href="#">Améliorer la génération de salle</a>	Hugo
<a href="#">Certaines cases du labyrinthe sont spéciales :</a>	Hugo

Tâches Bonus	
<a href="#">Afficher les éléments d'une salle graphiquement (Ajout des sprites)</a>	Thomas
<a href="#">Ajouter un menu principale</a>	Alexandre
<a href="#">Afficher et gérer le score</a>	Thomas
<a href="#">Gérer la fin du jeu (mort du héros + écran Game Over)</a>	Alexandre

BOHAN Thomas

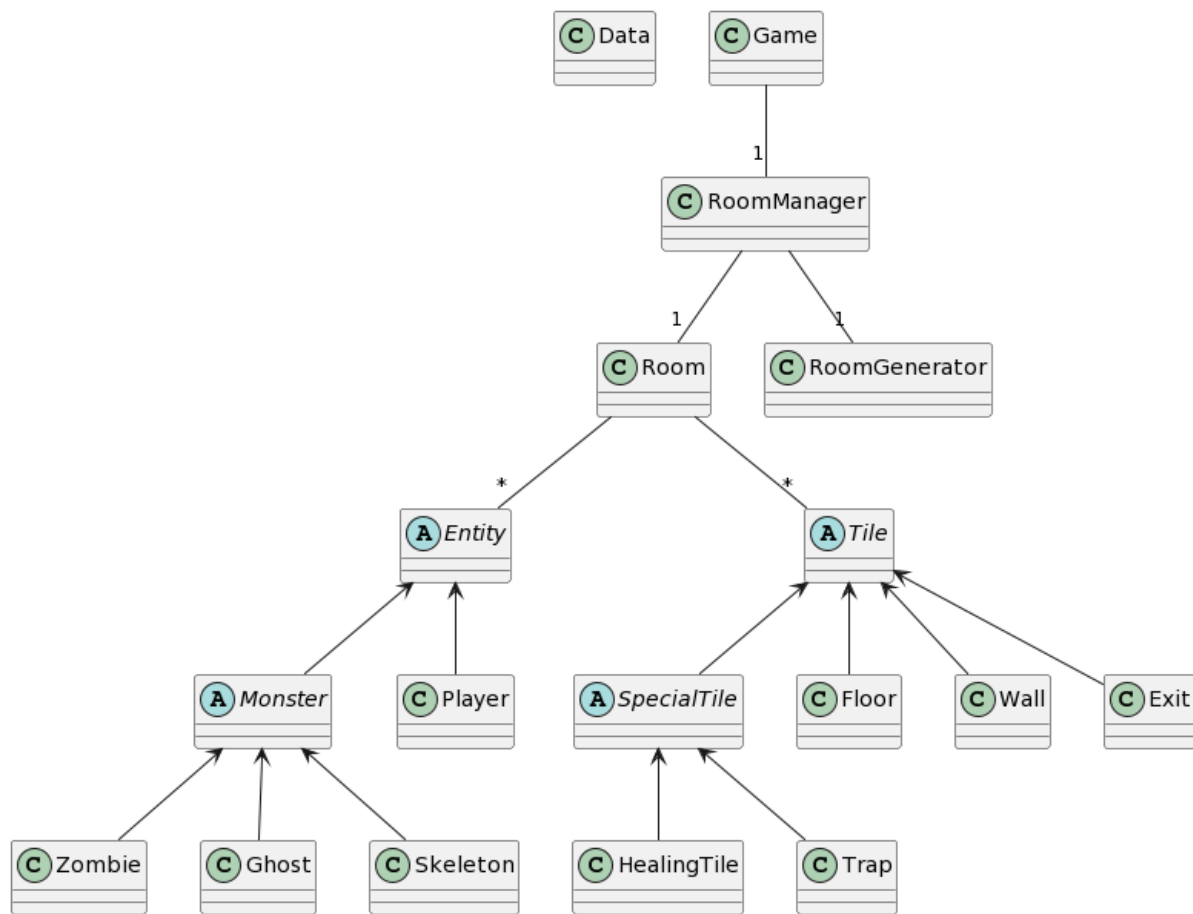
JACQUES Louis

SERRATORE Alexandre

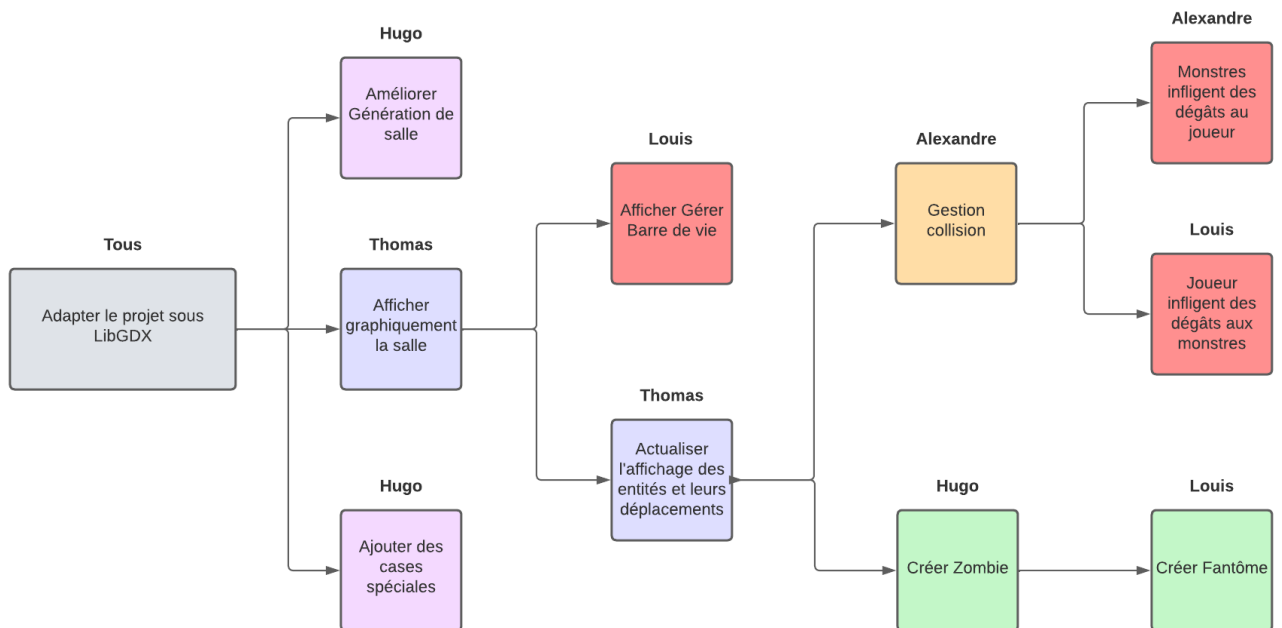
SUCK Hugo



Diagramme de classe :



## Diagramme d'ordonnancement des tâches :



## Sprint 1 : Review

Liste des tâches	Validé	Commentaires
<a href="#">Adapter le projet a l'utilisation de LibGDX</a>	Validé	Réalisé sans trop de soucis.
<a href="#">Utiliser Gradle pour l'installation et le lancement du jeu</a>	Validé	Création d'un script pour faciliter l'installation et le lancement du jeu.
<a href="#">Afficher les éléments d'une salle graphiquement (écran)</a>	Validé	Réalisé sans soucis.
<a href="#">Afficher et gérer la barre de point de vie</a>	Validé	Réalisé sans soucis.
<a href="#">Les monstres infligent des dégâts aux joueurs</a>	Validé	Réalisé sans soucis.
<a href="#">Gérer les collisions entre le héros, les monstres et les murs</a> (Adapté à la librairie utilisée)	Validé	Problème d'accès aux bodies des entités, réécriture de certains constructeurs pour faciliter la tâche.
<a href="#">Le héros peut infliger des dégâts aux monstres</a>	En cours	Problème lié au body de l'arme qui permet d'infliger des dégâts.
<a href="#">Actualiser l'affichage des entités et modifier leur déplacement</a>	Validé	Petit blocage lié à un oubli sur le type d'entité d'un body qui n'était pas dynamique.
<a href="#">Créer différent type de monstre au comportement différents</a> (Zombie, Fantômes)	En cours	Implémentation plus longue que prévu pour l'algorithme de pathfinding du zombie. Fantôme implémenté sans soucis.
<a href="#">Améliorer la génération de salle</a>	Validé	Réalisé sans soucis et facilement évolutif (ajout ou suppression de pattern/génération).
<a href="#">Certaines cases du labyrinthe sont spéciales :</a>	Validé	Création d'une classe abstraite pour les cases spéciales pour simplifier l'évolutivité.

Tâches Bonus		Commentaires
<a href="#">Afficher les éléments d'une salle graphiquement (Ajout des sprites)</a>	Validé	Problème de caméra rencontré. Nous avons dû revoir totalement la gestion de la caméra.
<a href="#">Ajouter un menu principale</a>	Validé	Réalisé sans soucis.
<a href="#">Afficher et gérer le score</a>	Validé	Réalisé sans soucis.
<a href="#">Gérer la fin du jeu (mort du héros + écran Game Over)</a>	Validé	Réalisé sans soucis.

## Sprint 1 : Rétrospective

### Résolution des problèmes du Sprint 0 :

- Mauvaise utilisation des branches GIT :  
**L'utilisation des branches sur Git** a été mise en place, ce qui a facilité le travail en parallèle de plusieurs tâches. Des revues de code ont pu être effectuées avant les merge afin d'éviter d'éventuel problème.
- Problème de la génération du fichier JAR :  
**Création d'un script BASH** pour générer automatiquement le fichier JAR. Une seule commande suffit désormais pour générer le fichier exécutable JAR et de lancer l'archive.

### Ce qui s'est bien passé (ou pas) et les décisions pour le sprint suivant :

- **L'implémentation de la bibliothèque graphique** s'est faite sans trop de problème. Nous avons dû adapter quelque petit détail à la configuration gradle fournie avec Libgdx, mais le reste a été grandement facilité de part l'architecture créée lors du Sprint 0.
- Des problèmes d'accès à certaines fonctions/variables ont été rencontrés lors de **la gestion des collisions**. L'écouteur de Libgdx ne permettait pas de remonter jusqu'au **body** et à l'**objet** lié aux **fixtures**. Une légère réécriture de certains constructeurs a dû être effectuée.
- **Le diagramme d'ordonnancement des tâches** à faciliter l'organisation de l'équipe. Nous utiliserons de nouveau ce genre de diagramme pour les sprints suivants.
- **Le comportement des différents types de monstres** n'est pas terminé :
  - Le fantôme fonctionne (traverse les murs en direction du joueur)
  - Le zombie ne possède pas son comportement de déplacement (l'implémentation du pathfinding est plus longue que prévu).

**Décision prise :** La tâche sera donc **prolongée sur le prochain sprint**.

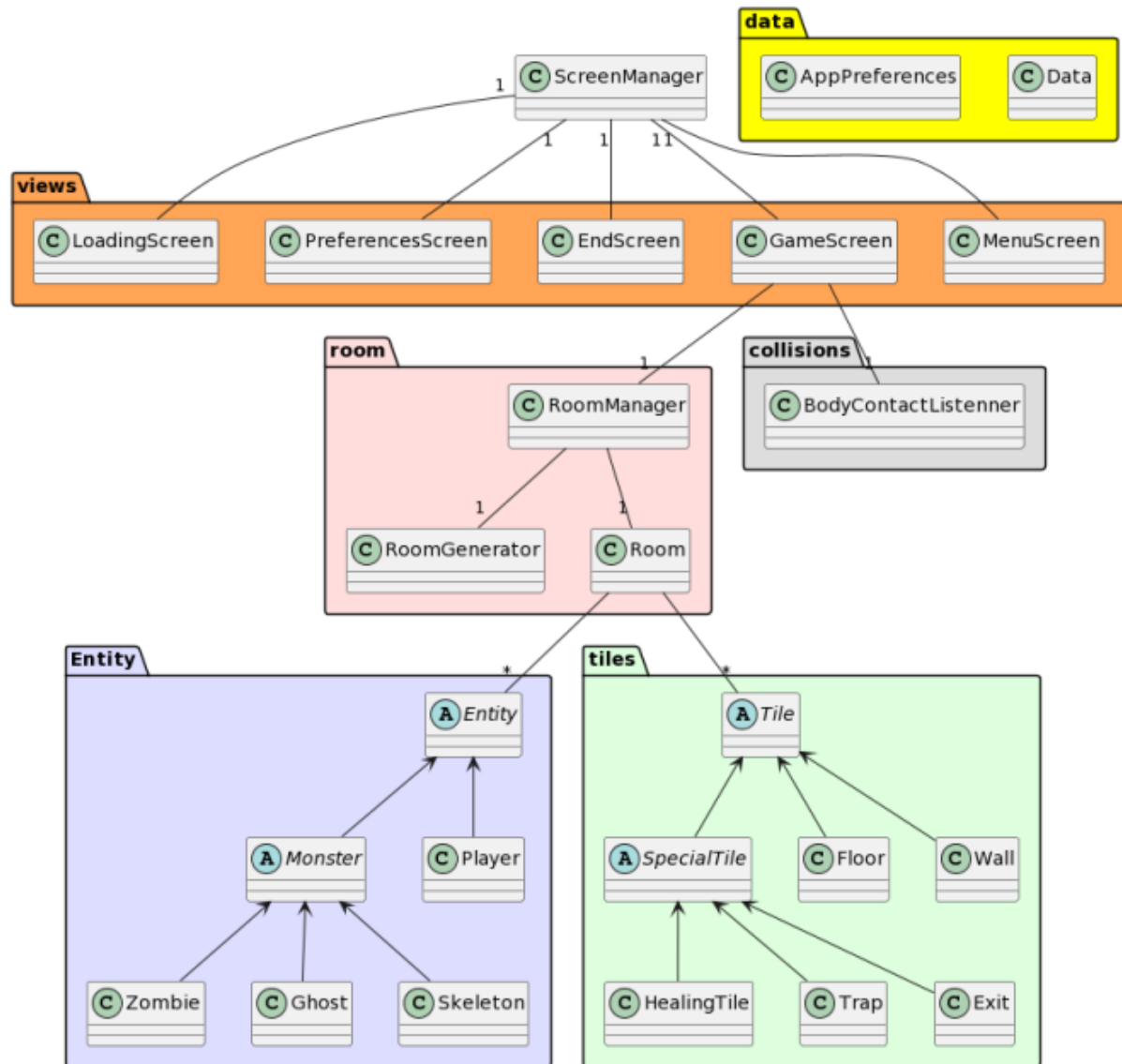
- **La création de l'arme du joueur** lui permettant d'infliger des dégâts n'est pas terminée. Un problème sur la position du body de l'objet a été rencontré. Le sprite quant à lui est placé au bon endroit.

**Décision prise :** La tâche sera donc **prolongée sur le prochain sprint**.

- **Le redimensionnement et la position de la caméra** posent des problèmes en fonction de la taille de la fenêtre.

**Décision prise :** Le bug sera corrigé sur le **prochain sprint**.

Diagramme de classe (actualisé) :



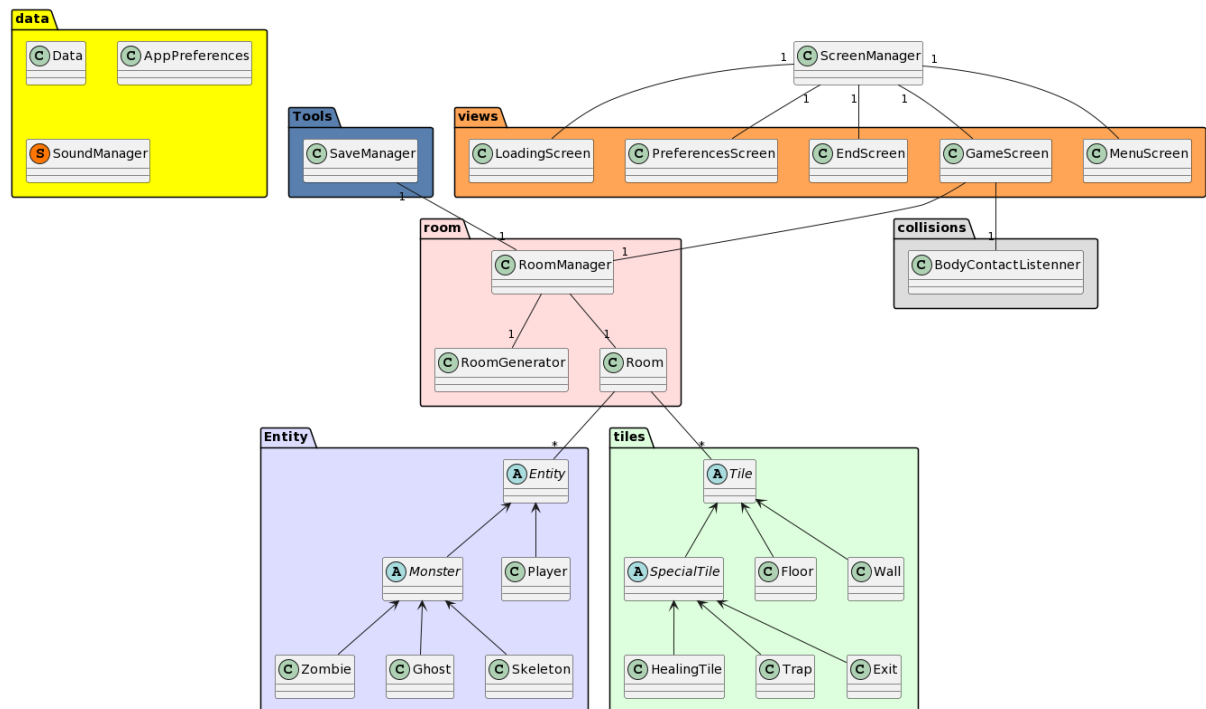
## Sprint 2 : Lancement

Liste des tâches prolongées du Sprint 1	Responsable
<a href="#">Créer différent type de monstre au comportement différents</a> (Zombie)	Hugo
<a href="#">Le héros peut infliger des dégâts aux monstres</a>	Louis (Créer et afficher l'arme) Alexandre (Gérer la collision entre l'arme et les monstres)

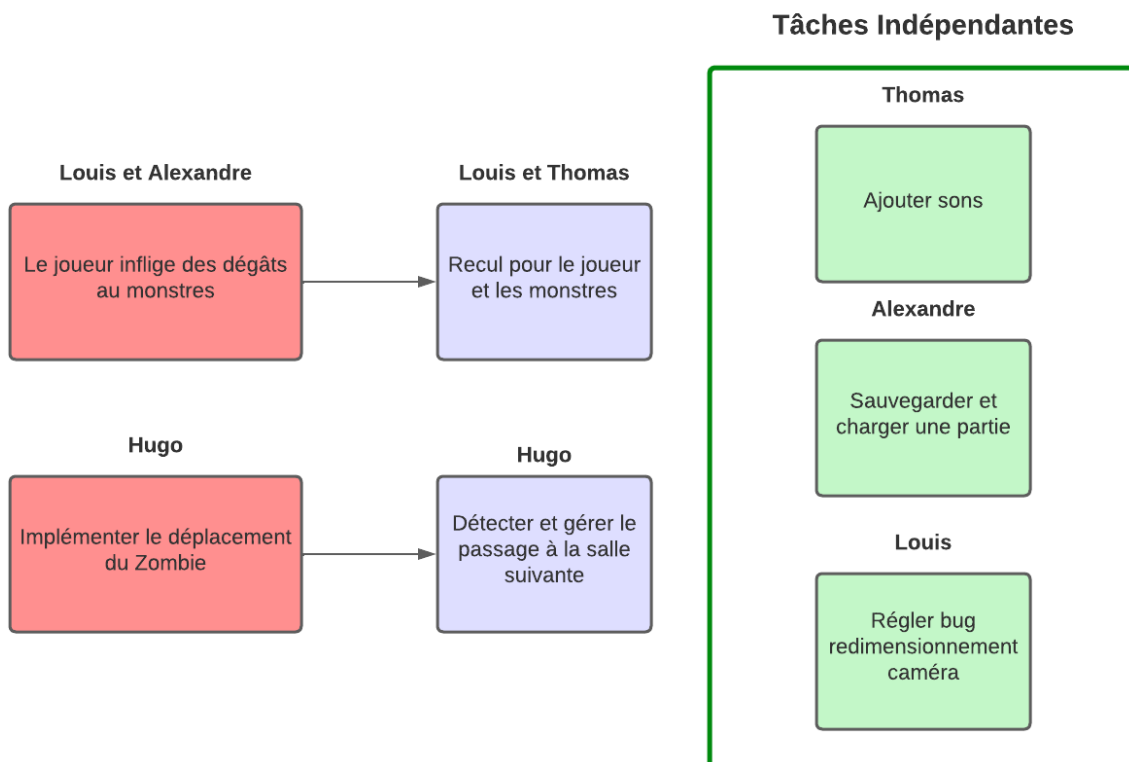
Liste des tâches	Responsable
<a href="#">Problème de redimensionnement de caméra</a>	Louis
<a href="#">Ajouter du son / bruitage / musique</a>	Thomas
<a href="#">Pouvoir sauvegarder / Charger une partie</a>	Alexandre
<a href="#">Recul lorsque le joueur subit des dégâts</a>	Thomas / Louis
<a href="#">Recul lorsque les monstres subissent des dégâts</a>	
<a href="#">Créer une sortie vers la salle suivante</a>	Hugo

Tâches Bonus	Responsable
Ajouter des tuiles	
Ajouter des monstres	

## Diagramme de classe :



## Diagramme d'ordonnancement des tâches :



BOHAN Thomas

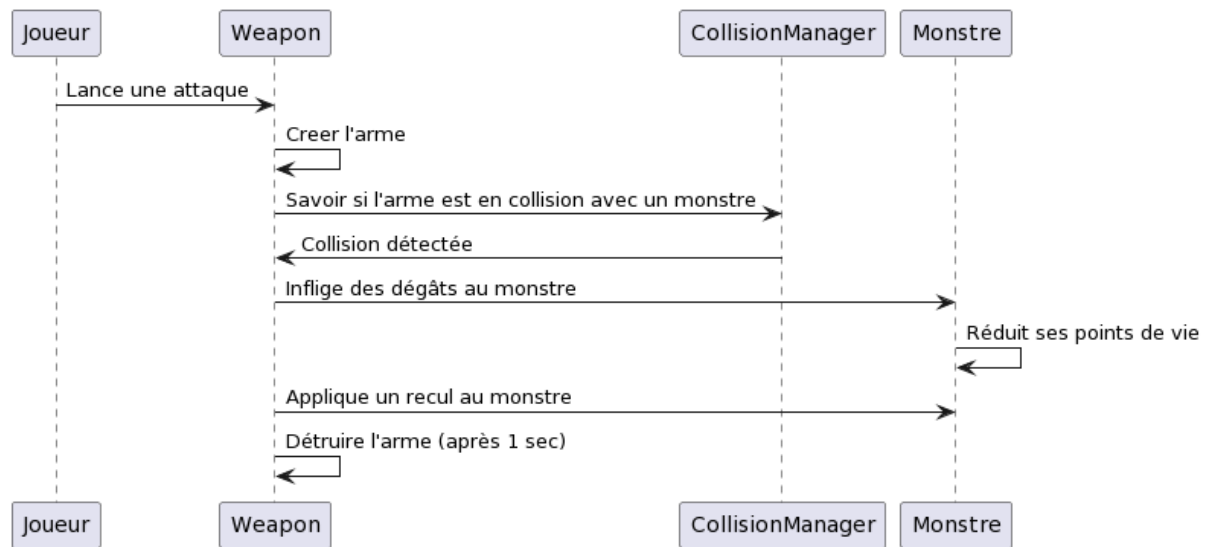
JACQUES Louis

SERRATORE Alexandre

SUCK Hugo



## Diagramme de séquence : Infliger des dégâts aux monstres



## Sprint 2 : Review

Liste des tâches	Validé	Commentaires
<a href="#">Créer différent type de monstre au comportement différents</a> (Zombie)	Validé	Beaucoup de difficulté pour implémenter A*. Des problèmes liés à la représentation du système de coordonnées pour le pathfinding ont été rencontrés.
<a href="#">Le héros peut infliger des dégâts aux monstres</a>	Validé	Ajout de l'incrémentation du score lorsqu'un monstre est tué. Le monstre meurt lorsqu'il n'a plus de PV.
<a href="#">Problème de redimensionnement de caméra</a>	En cours	Source du problème non identifié pour l'instant. Un fix partiel a été mis en place.
<a href="#">Ajouter du son / bruitage / musique</a>	Validé	Réalisé sans soucis.
<a href="#">Pouvoir sauvegarder / Charger une partie</a>	Validé	Difficultés de gestion des erreurs en cas de fichier non existant.
<a href="#">Recul lorsque le joueur subit des dégâts</a>	Validé	La seule difficulté a été de trouver la direction de recul qui devait être appliquée.
<a href="#">Recul lorsque les monstres subissent des dégâts</a>	Validé	
<a href="#">Créer une sortie vers la salle suivante</a>	Validé	Réalisé sans soucis.

## Sprint 2 : Rétrospective

### Résolution des problèmes du Sprint 1 :

- **Le comportement de déplacement du Zombie :**  
Après plusieurs heures de recherche, le Zombie possède désormais son comportement de déplacement. C'est-à-dire, il se dirige vers le joueur en prenant le chemin le plus court (et sans avoir d'obstacles devant lui). Pour réaliser l'implémentation, nous avons récupéré l'algorithme de pathfinding A\* sur internet. Ensuite nous l'avons modifié pour correspondre à notre système de position qui était différent. Pour finir, de nombreux ajustements ont été nécessaires pour rendre le déplacement fluide et agréable à regarder. Actuellement, le Zombie se déplace vers le joueur uniquement s'il est à une certaine distance de lui. Il s'agit d'un choix de l'équipe pour ne pas rendre le jeu trop difficile. Tel que c'est actuellement implémenté, le Zombie peut très bien trouver un chemin de longue distance et le parcourir (D'un bout à l'autre d'une salle).
- **La création de l'arme du joueur :**  
La création et la destruction de l'arme ont été implémentées. En appuyant sur la touche Espace, l'arme apparaît devant le joueur (en fonction de sa direction de déplacement) et disparaît peu de temps après. Si le joueur se déplace avant la disparition de l'arme, celle-ci est détruite. Si un monstre rentre en contact avec l'arme, celui-ci prend des dégâts et donc voit ses points de vie descendre.

### Ce qui s'est bien passé (ou pas) et les décisions pour le sprint suivant :

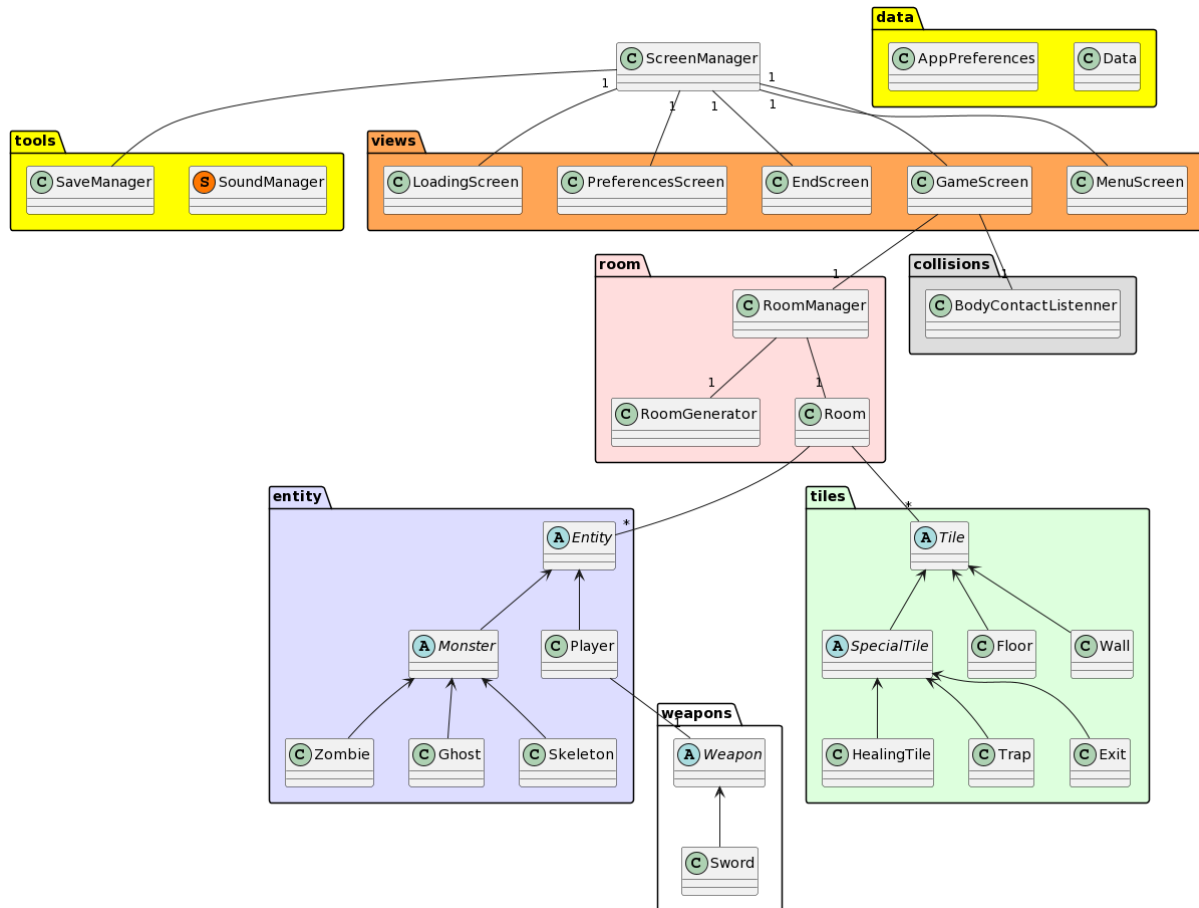
- Les **branches GIT** sont toujours **bien utilisées** par l'ensemble de l'équipe.
- Le **diagramme d'ordonnancement** ainsi que le **diagramme de séquence** nous ont permis de **mieux travailler** et de **mieux visualiser** le résultat que nous souhaitons.
- La **gestion du temps** s'est faite **sans mal** malgré les projets et devoirs des autres matières.
- **Il y a quelques légers bugs liés à la porte :**
  - Si le joueur touche la porte et qu'il reste des monstres en vie, la collision ne sera pas détectée lorsque le joueur voudra passer à la salle suivante en touchant la porte.
  - Si le jeu se lance sur une salle vide, la salle ne veut pas ouvrir la porte, malgré le fait qu'il n'y a pas de monstre en vie. Il est alors impossible de passer à la salle suivante.

**Décision prise :** Les bugs seront **corrigés dans le prochain sprint.**

- **Le redimensionnement de la caméra** posent des problèmes en fonction de la taille de la fenêtre. Nous pensons que la gestion des différentes fenêtres est responsable du problème.

**Décision prise :** Le bug sera **corrigé dans le prochain sprint**.

Diagramme de classe (actualisé) :

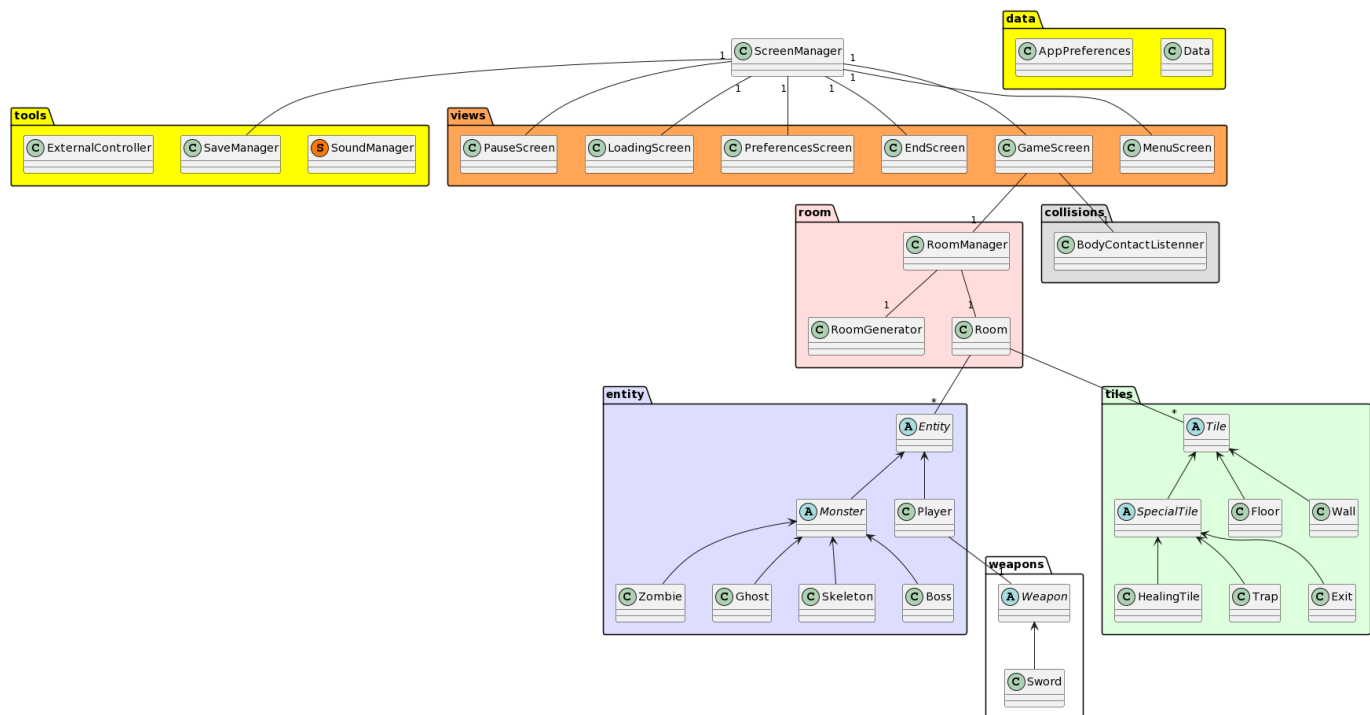


## Sprint 3 : Lancement

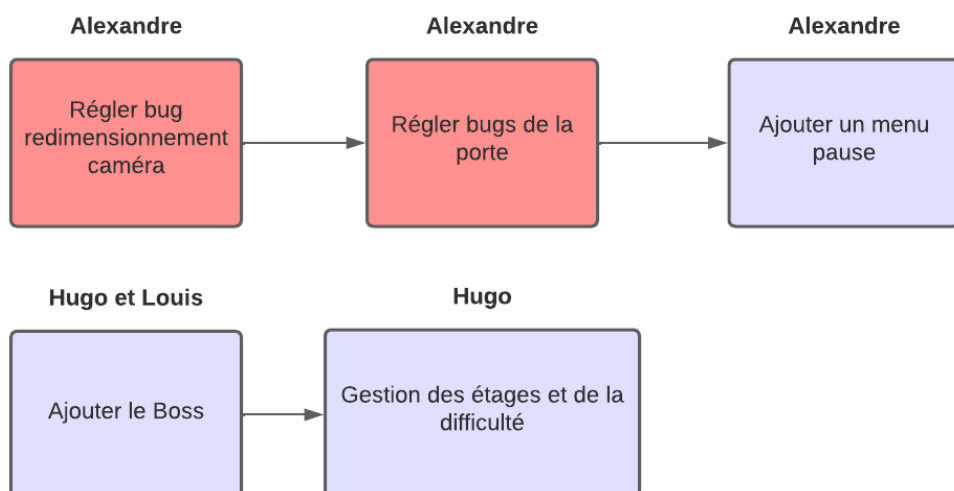
Liste des tâches prolongées du Sprint 2	Responsable
<a href="#">Problème de redimensionnement de caméra</a>	Alexandre

Liste des tâches	Responsable
<a href="#">Problème lié au comportement de la porte</a>	Alexandre
<a href="#">Ajouter un menu pause</a>	Alexandre
<a href="#">Ajouter la compatibilité d'un controller externe (manette)</a>	Louis
<a href="#">Créer différent type de monstre au comportement différents</a> (Boss tous les X salles)	Hugo Louis
<a href="#">Gestion des étages / difficulté</a>	Hugo
<a href="#">Le joueur peut avoir un bouclier</a>	Thomas
<a href="#">L'arme du joueur possède un CD (temps de recharge)</a>	Thomas

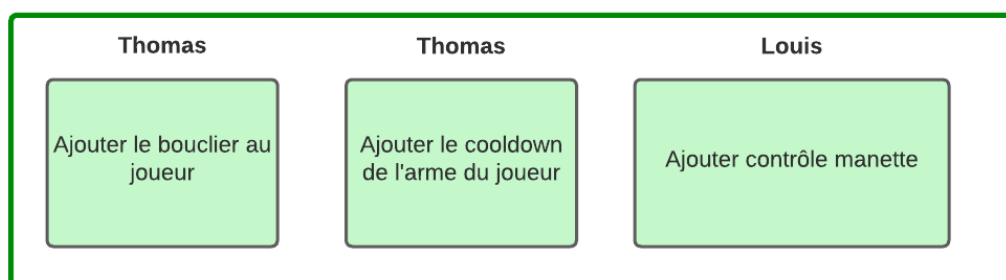
Diagramme de classe :



**Diagramme d'ordonnancement des tâches :**



## Tâches Indépendantes



BOHAN Thomas

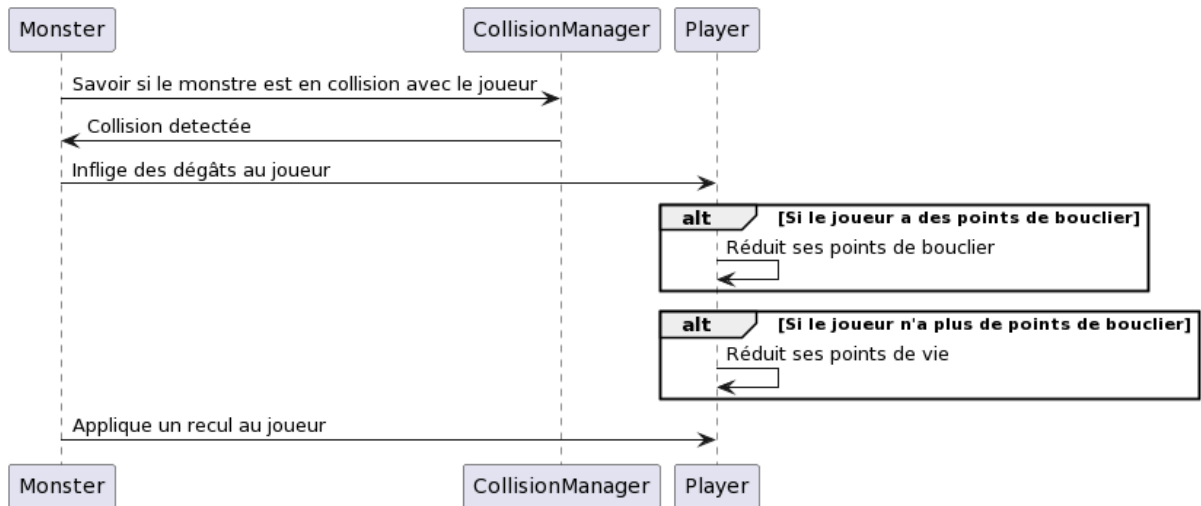
JACQUES Louis

SERRATORE Alexandre

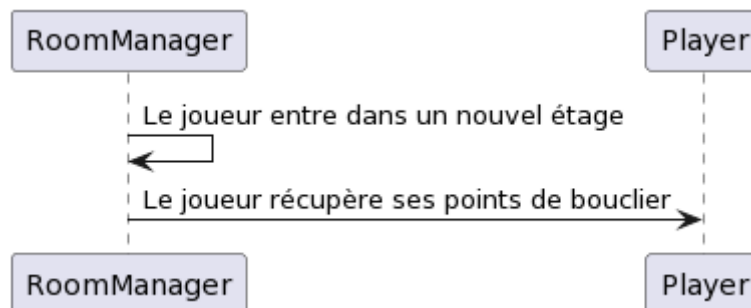
SUCK Hugo

## Diagramme de séquence Bouclier :

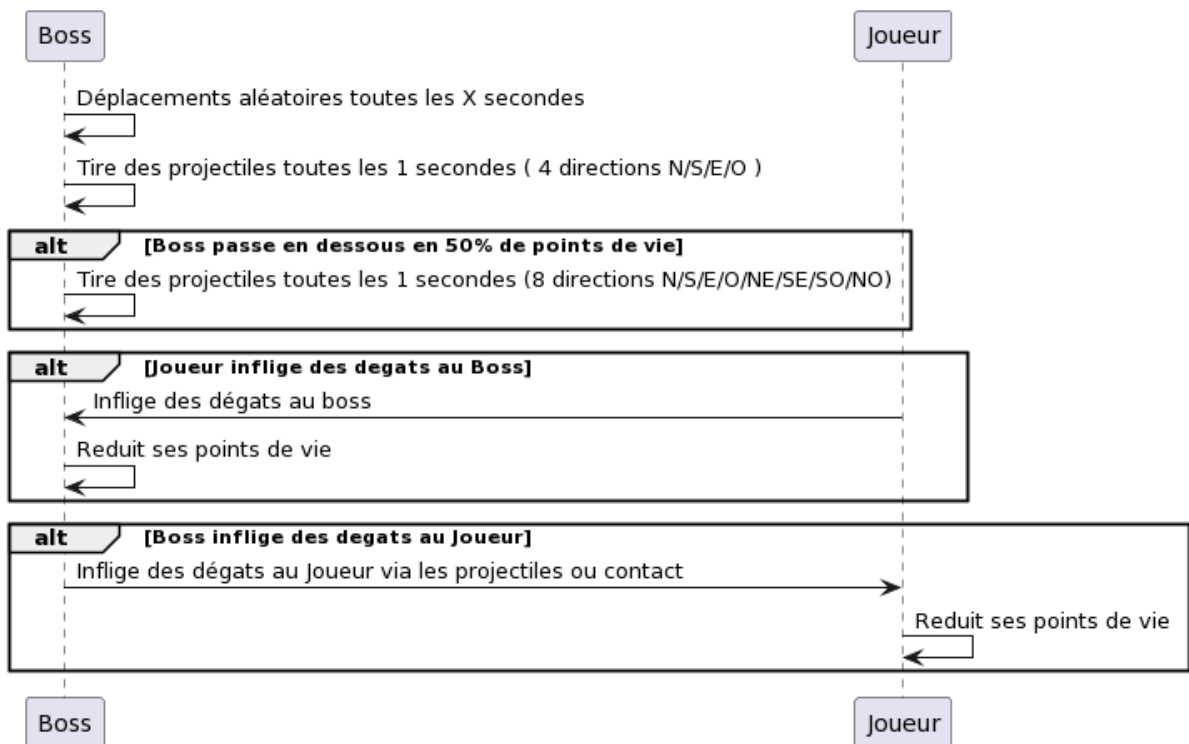
### Cas 1 : Le joueur subit des dégâts



### Cas 2 : Le joueur regagne ses points de bouclier en entrant dans une nouvelle salle



### Diagramme de séquence du Boss :





# Liste des fonctionnalités

## Salles

### 1) Créer la salle de départ du jeu et son affichage (Version terminal)

Créer une salle à génération fixe. Une salle possède des casses de sols et des cases de murs.

### 2) Créer un générateur de salle simple

Créer des salles à génération variable. Se baser sur des patterns fixes écrits sur un fichier texte externe.

### 3) Créer une sortie vers la salle suivante

Lorsque le héros atteint la sortie de la salle, il est téléporté dans la salle suivante. La sortie est disponible lorsque tous les monstres ont été tués.

### 4) Certaines cases du labyrinthe sont spéciales :

#### • Créer des trésors :

Si le héros arrive sur la case, il gagne un bonus.

#### • Créer des pièges :

Si le héros arrive sur la case, il subit des dégâts.

### 5) Améliorer la génération de salle

- Ajouter des paternes
- Ajouter des générations
- Améliorer les générations
- [Placement de la sortie modulable]

## 6) Gestion des étages / difficulté

Après chaque boss, le joueur change d'étage. La difficulté augmente au fur et à mesure que le joueur passe des étages.

## Héros

### 7) Créer, placer, déplacer le héros

Le héros se déplace en fonction des entrées clavier. Il ne peut pas traverser les murs. Le héros possède des points de vie.

### 8) Le héros peut infliger des dégâts aux monstres

- Ajouter une barre de vie aux monstres
- Afficher une épée lorsque le joueur frappe le monstre

### 9) L'arme du joueur possède un CD (temps de recharge)

Le joueur doit attendre un certain temps entre deux coups d'épée.

### 10) Le héros peut utiliser des bonus

Consommer des potions, armes, bouclier, etc

### 11) Le joueur peut avoir un bouclier

Le joueur possède une jauge de bouclier. Les dégâts infligés au joueur sont subis en premier par la jauge de bouclier.

Le bouclier se recharge entièrement à chaque changement de salle.

### 12) Recul lorsque le joueur subit des dégâts

## Monstres

- 13) Créer un monstre simple, se déplace aléatoirement
- 14) Les monstres infligent des dégâts aux joueurs

Les dégâts sont infligés lors du contact avec le joueur. Lorsque le joueur est touché, il ne peut plus être touché pendant un court instant (iFrame)

- 15) Créer différent type de monstre au comportement différents

- Squelette
  - Se déplace de manière aléatoire
- Zombie
  - Se déplace vers le joueur (Algo A\* pour le pathfinding)
- Fantôme
  - Se déplace à travers les murs vers le joueur (Algo A\* pour le pathfinding)
- Boss
  - Apparaît tout les X étages

## Collisions

- 16) Gérer les collisions entre le héros, les monstres et les murs

Utiliser les body de libgdx pour gérer les collisions.

- 17) Recul lorsque les monstres subissent des dégâts

## Interface Graphique & divers

- 18) Adapter le projet a l'utilisation de LibGDX

Passage au mode graphique avec la librairie LibGDX.

- 19) Utiliser Gradle pour l'installation et le lancement du jeu

Créer un script pour faciliter l'installation et le lancement du jeu (Mode Desktop).

## 20) Afficher les éléments d'une salle graphiquement (écran)

- Placer les body (mode debug)
- Ajouter les textures sur les body

## 21) Actualiser l'affichage des entités et modifier leur déplacement

Avoir un rendu en temps réel et non plus action par action.  
Gérer les déplacements via libgdx.

## 22) Ajouter un menu principale

Avant d'entrer en jeu, un menu principale s'affiche avec:

- Nouvelle partie
- Charger une partie
- Options
- Quitter

## 23) Afficher et gérer le score

## 24) Afficher et gérer la barre de point de vie

Le joueur possède une barre de PV visible sur l'écran. Lorsqu'il subit des dégâts ou lorsqu'il est soigné, la barre de PV s'actualise.

## 25) Gérer la fin du jeu (mort du héros + écran Game Over)

Si les PV du joueur atteignent 0, un écran de Game Over s'affichera.

## 26) Pouvoir sauvegarder / Charger une partie

Le jeu possède UNE sauvegarde automatique qui est écrasée à chaque changement de salle et des sauvegardes manuelles.

Il est possible de sauvegarder manuellement lorsque tous les monstres de la salle sont morts. Le joueur choisit alors le nom de sa sauvegarde.

27) Ajouter du son / bruitage / musique

28) Ajouter la compatibilité d'un controller externe (manette)

Implémentation avec une manette de Playstation 4.

(voir: <https://gamefromscratch.com/libgdx-tutorial-part-14-gamepad-support/> )

29) Ajouter un menu pause

Le joueur peut mettre en pause sa partie en appuyant sur la touche Echap.

Depuis le menu pause, le joueur peut quitter la partie, sauvegarder si aucun monstre n'est vivant et modifier les options.

## Correction de bug

30) Problème lié au comportement de la porte

31) Problème de redimensionnement de caméra

Le redimensionnement de la caméra pose problème lorsque l'on modifie la taille de la fenêtre dans les écrans hors jeu. L'écran de jeu est complètement déréglé.

## Idées

*Ajout de boss*

*Ajout de monstres*

*Ajout de tuiles (trou, lave, eau, buff)/*

*Ajout d'arme/équipement (arme à distance, )*

*Ajout d'un bouclier/armure*

*Système d'inventaire avec objets (potions, )*

*Utiliser une manette*

*Améliorer les statistiques du joueur (vitesse de déplacement, vie max, dégâts, etc)*