

Projet Analyse et Conception Logiciel

M1 - Info



Dungeon Crypt

BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

Table des matières :

1) Vision du produit	4
2) BackLogs	5
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
Sprint 3 : Review	25
Sprint 3 : Rétrospective	26
Diagramme de classe (actualisé) :	27
Liste des fonctionnalités	28
Salles	28
1) Créer la salle de départ du jeu et son affichage (Version terminal)	28
2) Créer un générateur de salle simple	28
3) Créer une sortie vers la salle suivante	28
4) Certaines cases du labyrinthe sont spéciales :	28
• Créer des trésors :	28
• Créer des pièges :	28
5) Améliorer la génération de salle	28
6) Gestion des étages / difficulté	29
Héros	29
7) Créer, placer, déplacer le héros	29
8) Le héros peut infliger des dégâts aux monstres	29
9) L'arme du joueur possède un CD (temps de recharge)	29
10) Le héros peut utiliser des bonus	29
11) Le joueur peut avoir un bouclier	29
12) Recul lorsque le joueur subit des dégâts	29
Monstres	30
13) Créer un monstre simple, se déplace aléatoirement	30
14) Les monstres infligent des dégâts aux joueurs	30
15) Créer différent type de monstre au comportement différents	30
• Squelette	30
• Zombie	30
• Fantôme	30
• Boss	30
Collisions	30

BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

16) Gérer les collisions entre le héros, les monstres et les murs	30
17) Recul lorsque les monstres subissent des dégâts	30
Interface Graphique & divers	31
18) Adapter le projet a l'utilisation de LibGDX	31
19) Utiliser Gradle pour l'installation et le lancement du jeu	31
20) Afficher les éléments d'une salle graphiquement (écran)	31
21) Actualiser l'affichage des entités et modifier leur déplacement	31
22) Ajouter un menu principale	31
23) Afficher et gérer le score	31
24) Afficher et gérer la barre de point de vie	31
25) Gérer la fin du jeu (mort du héros + écran Game Over)	32
26) Pouvoir sauvegarder / Charger une partie	32
27) Ajouter du son / bruitage / musique	32
28) Ajouter la compatibilité d'un controller externe (manette)	32
29) Ajouter un menu pause	32
Correction de bug	32
30) Problème lié au comportement de la porte	32
31) Problème de redimensionnement de caméra	32
Idées	34

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.

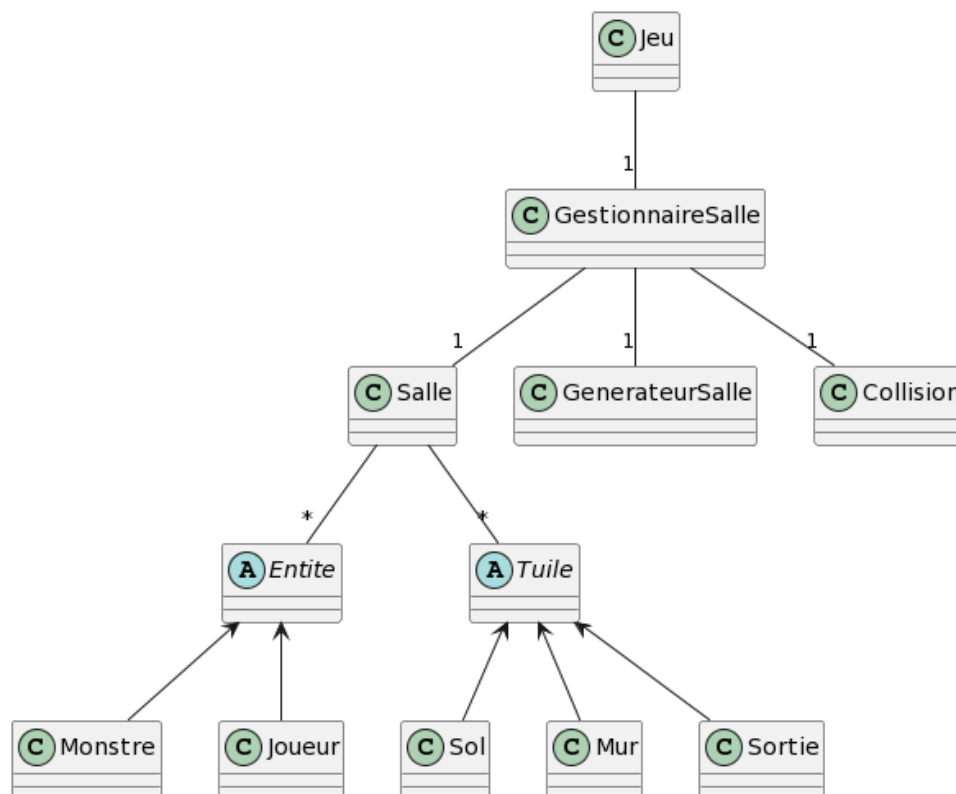
Des points lui sont attribués pour chaque monstre qu'il tue, certains équipements qu'il récupère et chaque étage qu'il descend.

2) BackLogs

Sprint 0 : Lancement

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

Diagramme de classe



BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

Sprint 0 : Review

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

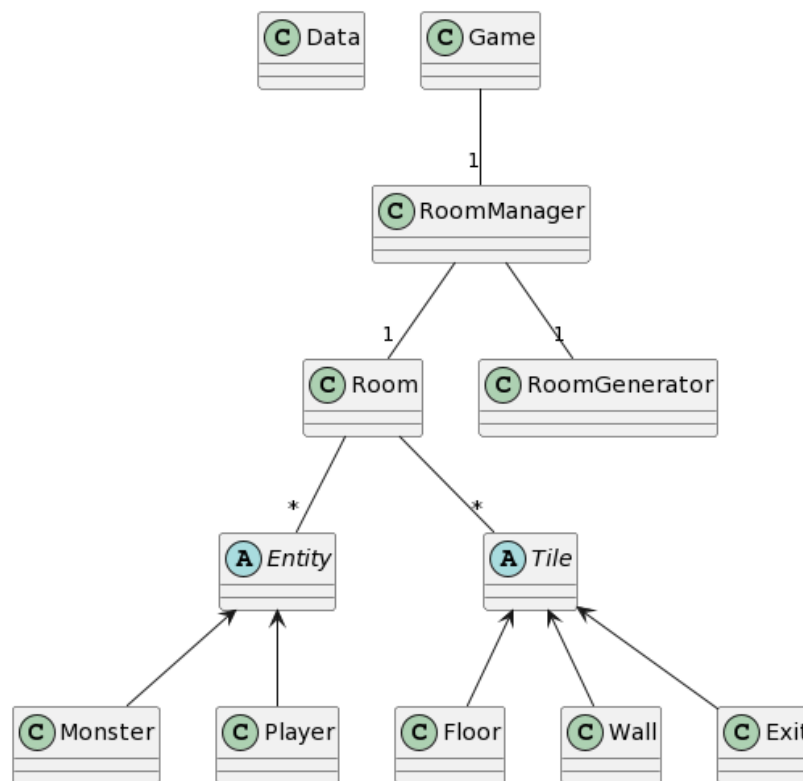
JACQUES Louis

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

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

BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

Diagramme de classe :

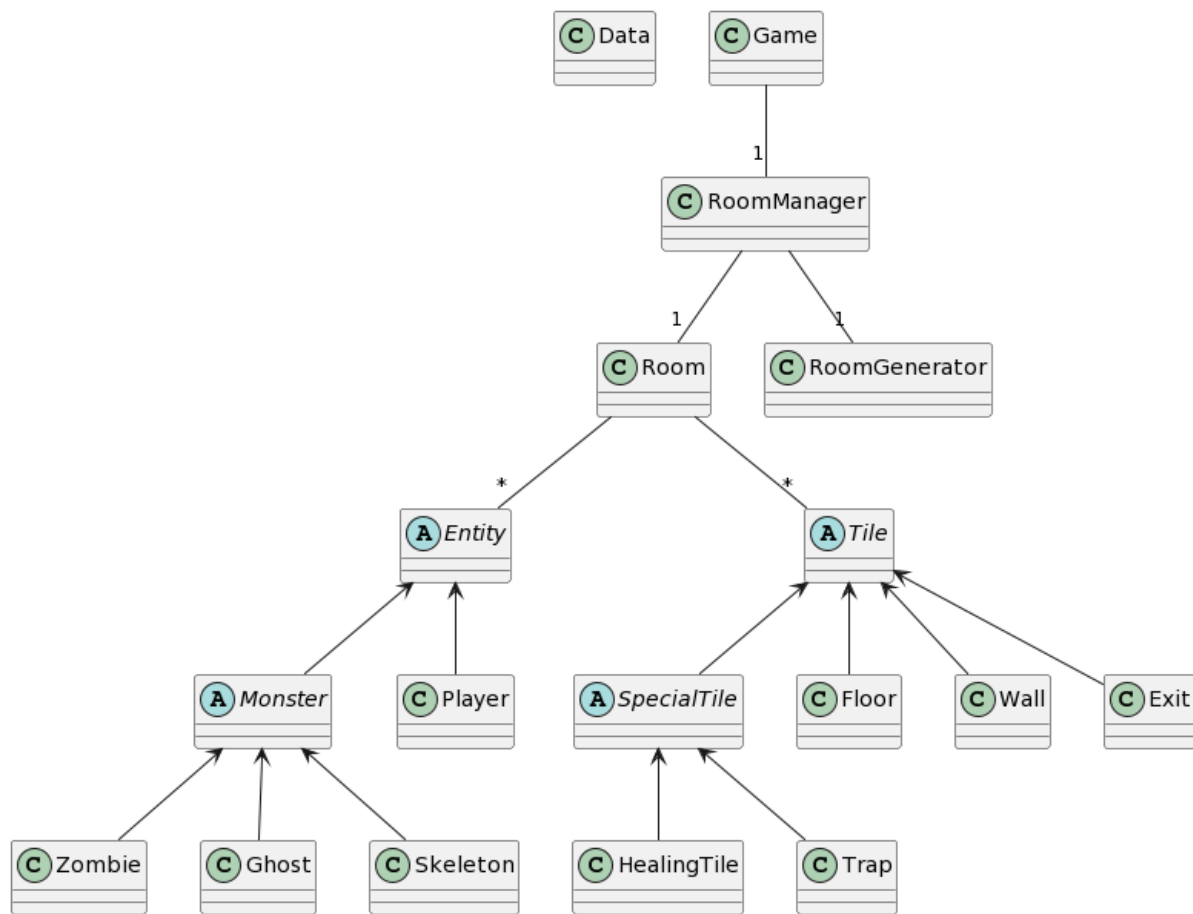
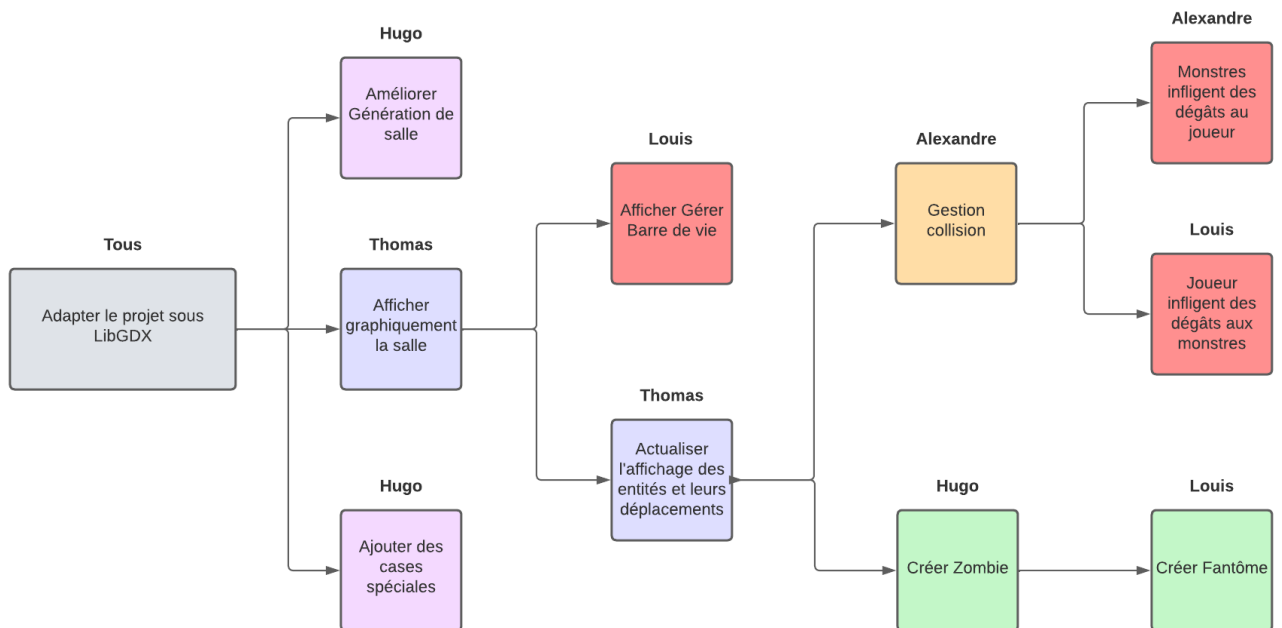


Diagramme d'ordonnancement des tâches :



Sprint 1 : Review

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

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

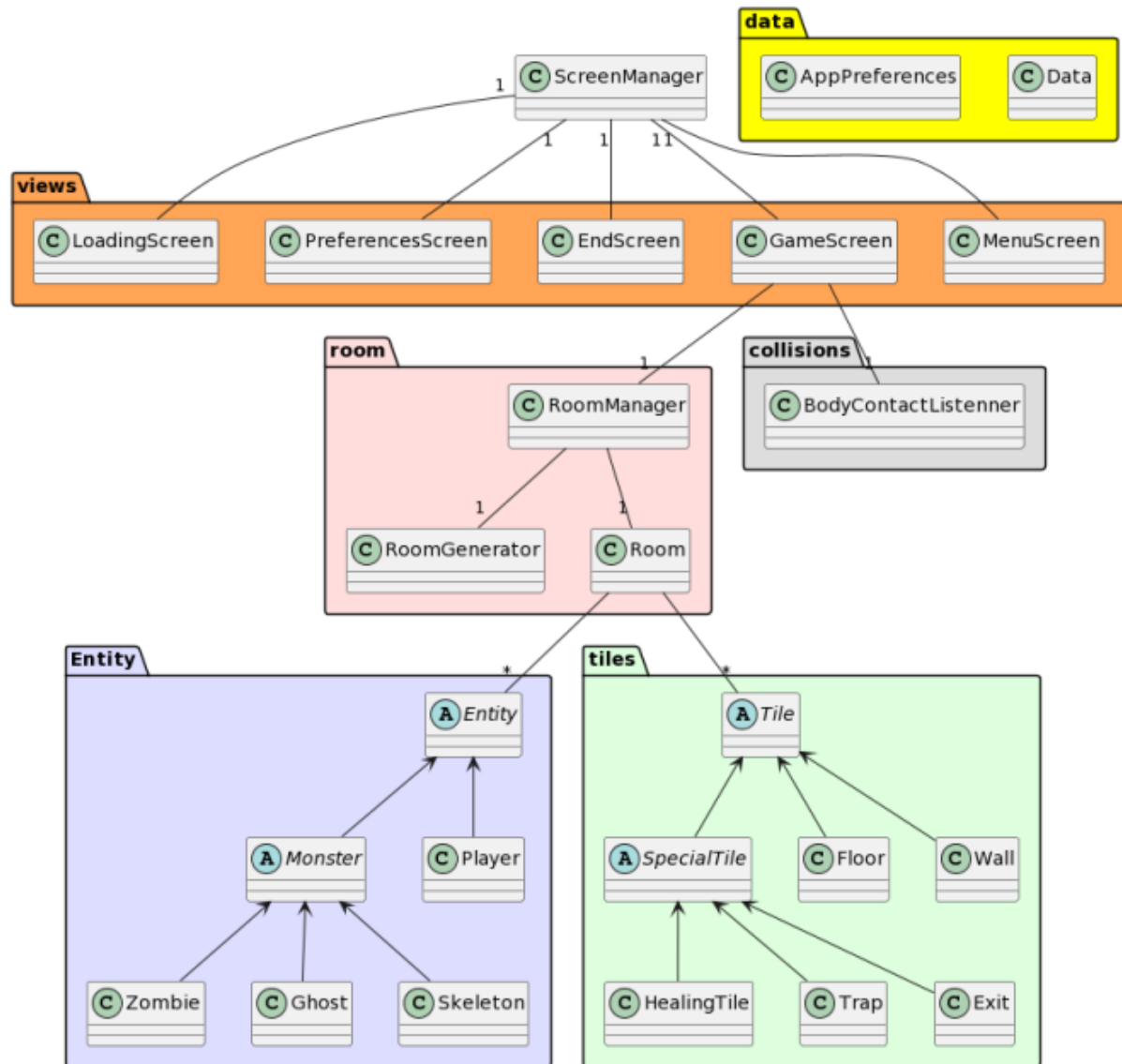
- **La création de l'arme du joueur** lui permettant d'infliger des dégâts n'est pas terminée. Un problème sur la position du body de l'objet a été rencontré. Le sprite quant à lui est placé au bon endroit.

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
Créer différent type de monstre au comportement différents (Zombie)	Hugo
Le héros peut infliger des dégâts aux monstres	Louis (Créer et afficher l'arme) Alexandre (Gérer la collision entre l'arme et les monstres)

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

Tâches Bonus	Responsable
Ajouter des tuiles	
Ajouter des monstres	

Diagramme de classe :

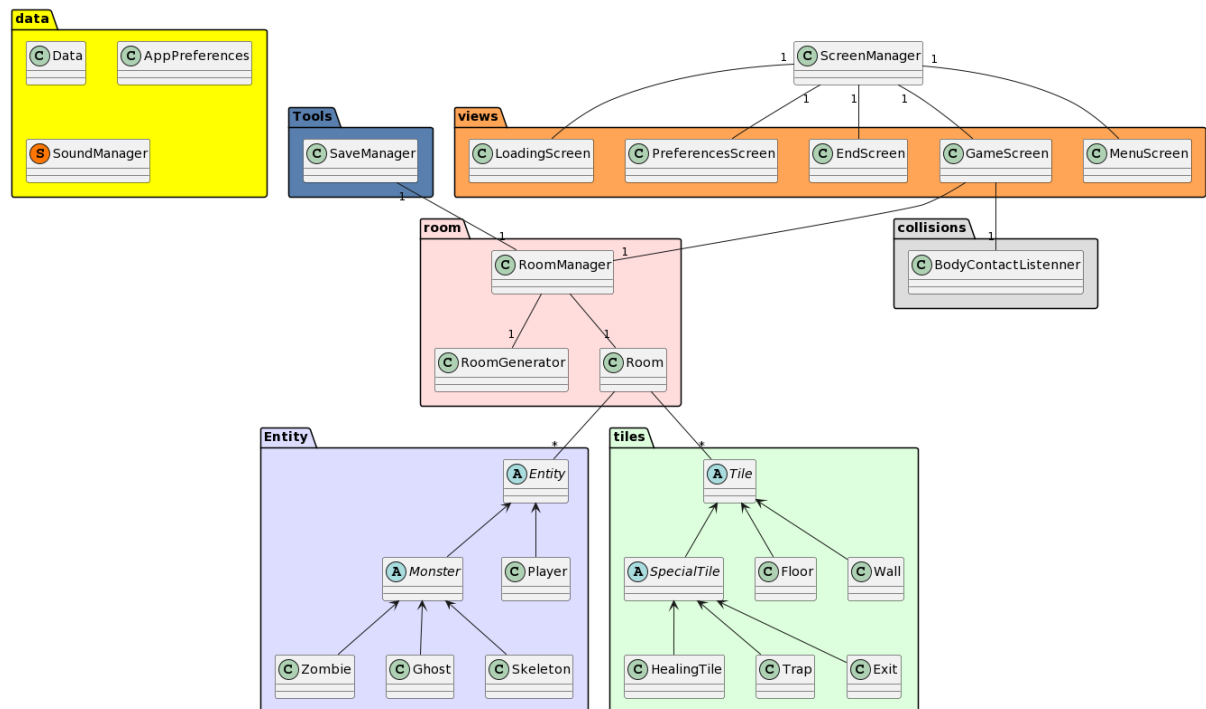
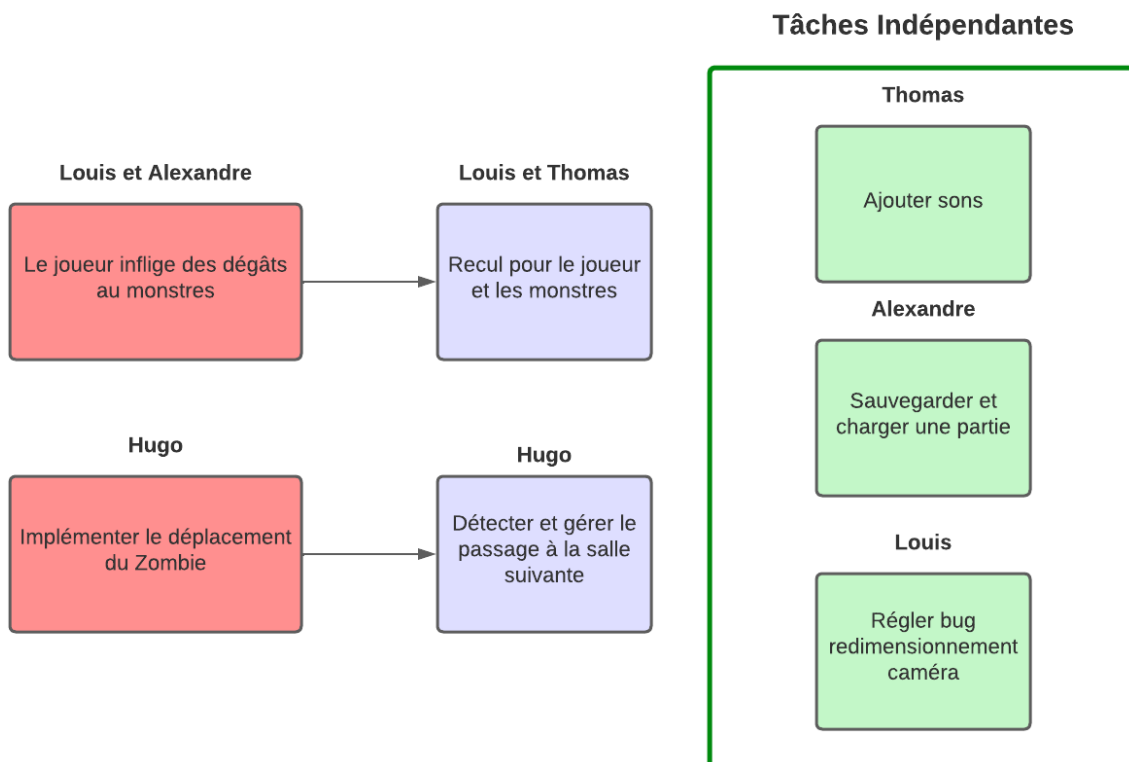


Diagramme d'ordonnancement des tâches :



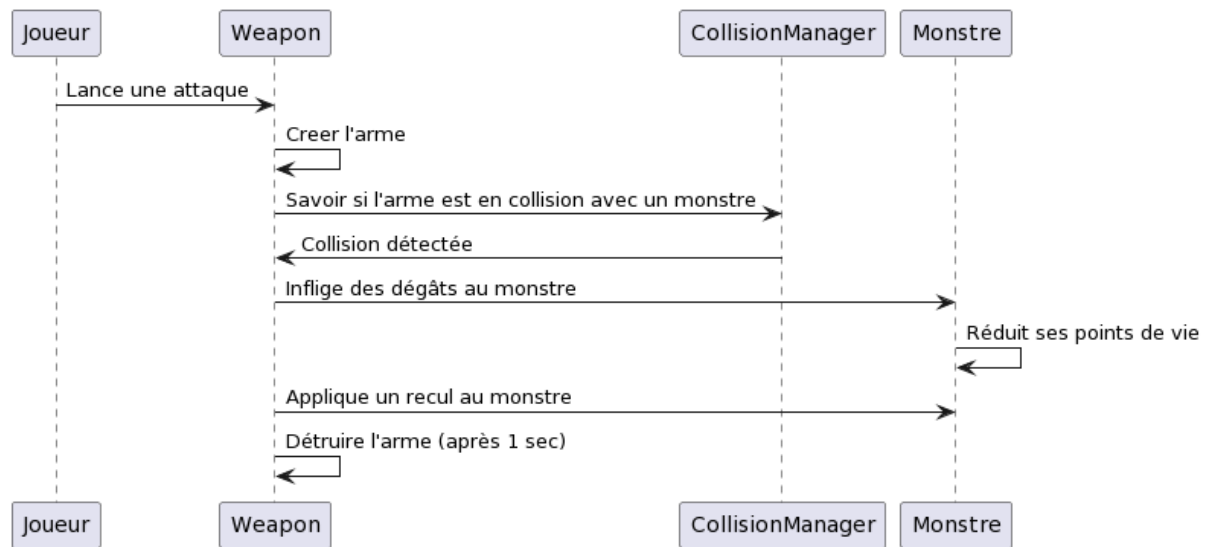
BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

Diagramme de séquence : Infliger des dégâts aux monstres



Sprint 2 : Review

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

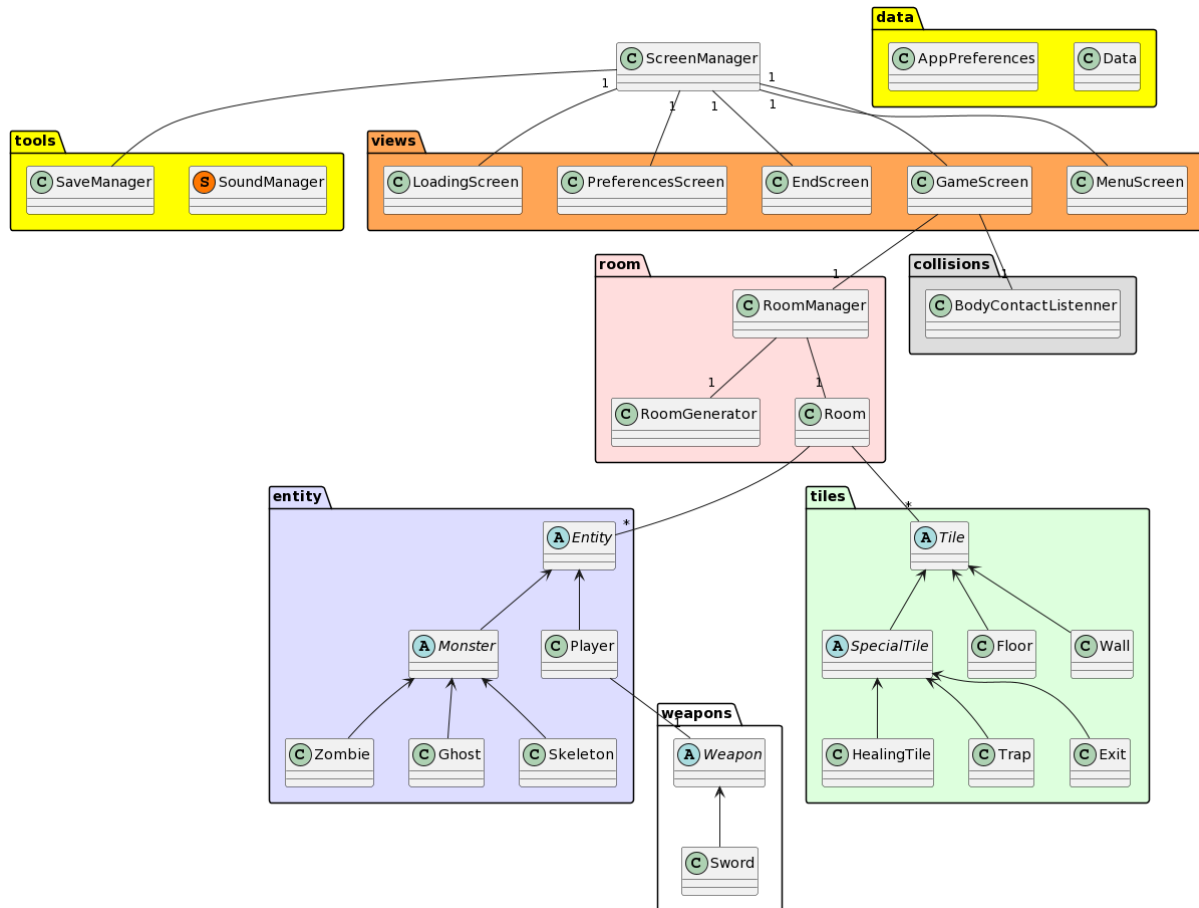
- Les **branches GIT** sont toujours **bien utilisées** par l'ensemble de l'équipe.
- Le **diagramme d'ordonnancement** ainsi que le **diagramme de séquence** nous ont permis de **mieux travailler** et de **mieux visualiser** le résultat que nous souhaitons.
- La **gestion du temps** s'est faite **sans 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
Problème de redimensionnement de caméra	Alexandre

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

Diagramme de classe :

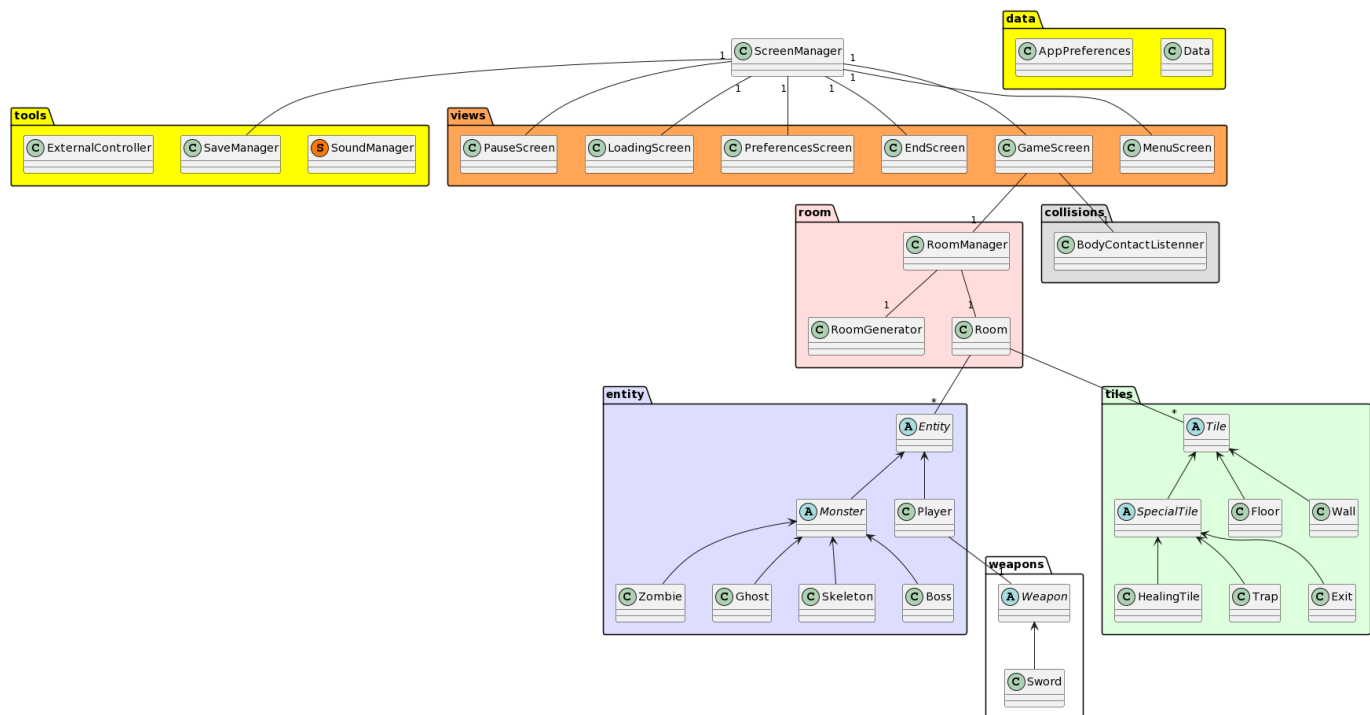
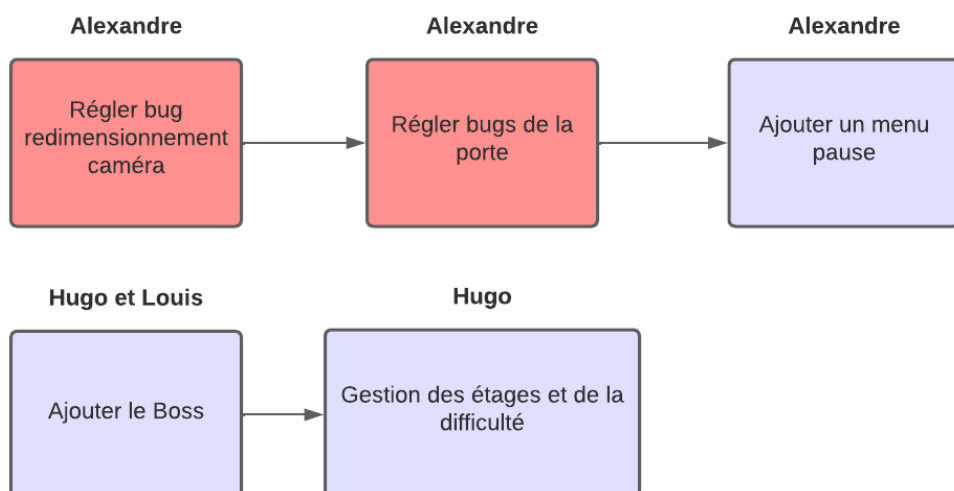
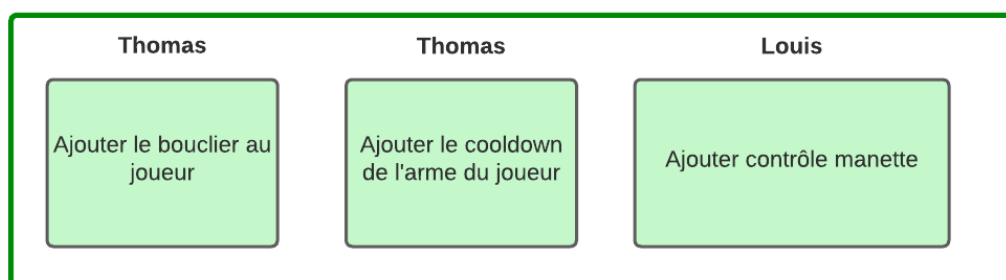


Diagramme d'ordonnancement des tâches :



Tâches Indépendantes



BOHAN Thomas

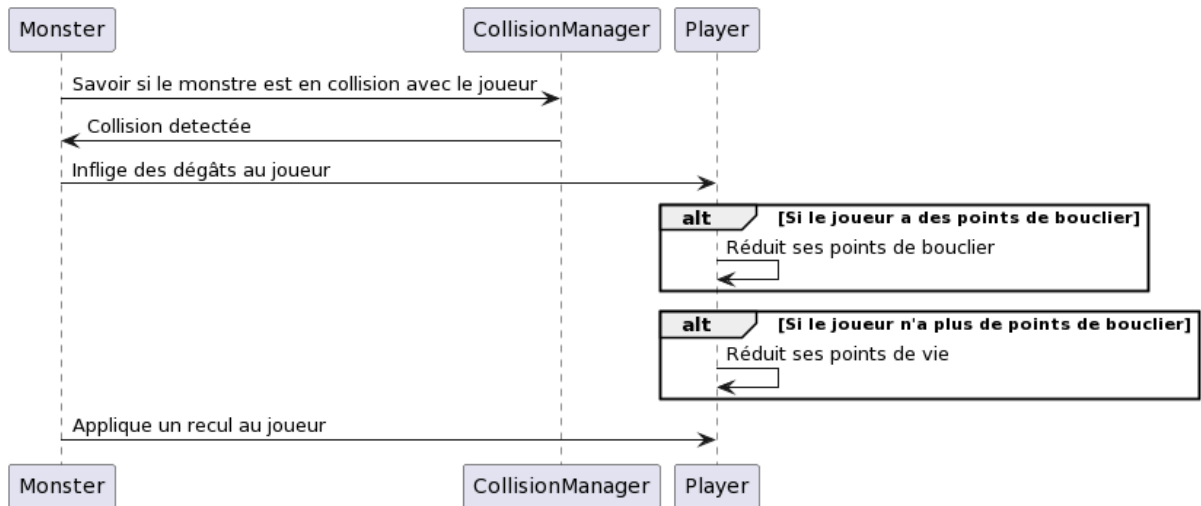
JACQUES Louis

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

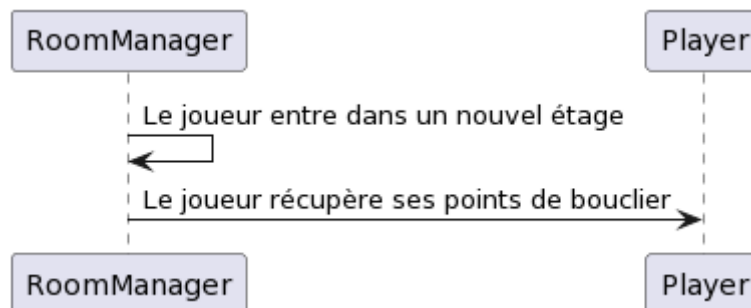
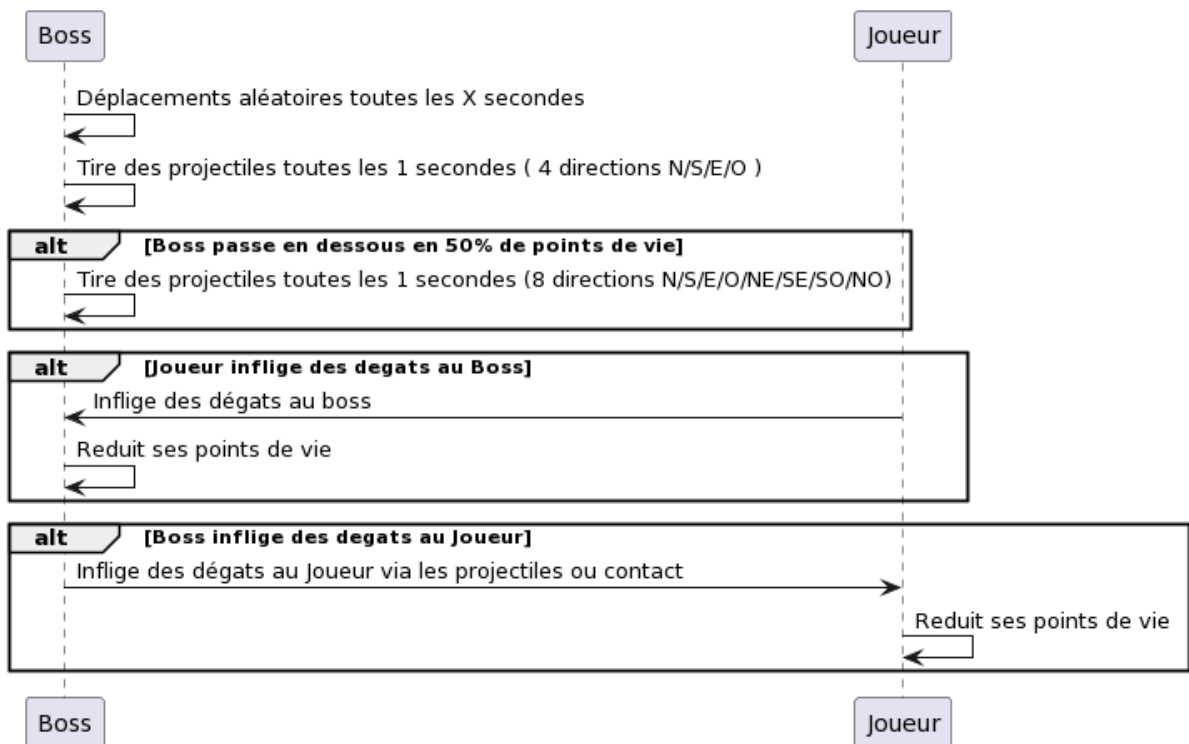


Diagramme de séquence du Boss :



Sprint 3 : Review

Liste des tâches	Validé	Commentaires
Problème de redimensionnement de caméra	Validé	Problème lié au positionnement des sprites identifié comme une des sources du problème. Les deux ont été corrigés.
Problème lié au comportement de la porte	Validé	Réalisé sans soucis.
Ajouter un menu pause	Validé	Réalisé sans soucis.
Ajouter la compatibilité d'un controller externe (manette)	Validé	Réalisé sans soucis.
Créer différent type de monstre au comportement différents (Boss tous les X salles)	Validé	Réalisé sans soucis. Le comportement du boss correspond à nos souhaits.
Gestion des étages / difficulté	Validé	Réalisé sans soucis.
Le joueur peut avoir un bouclier	Validé	Réalisé sans soucis.
L'arme du joueur possède un CD (temps de recharge)	Validé	Réalisé sans soucis.

Sprint 3 : Rétrospective

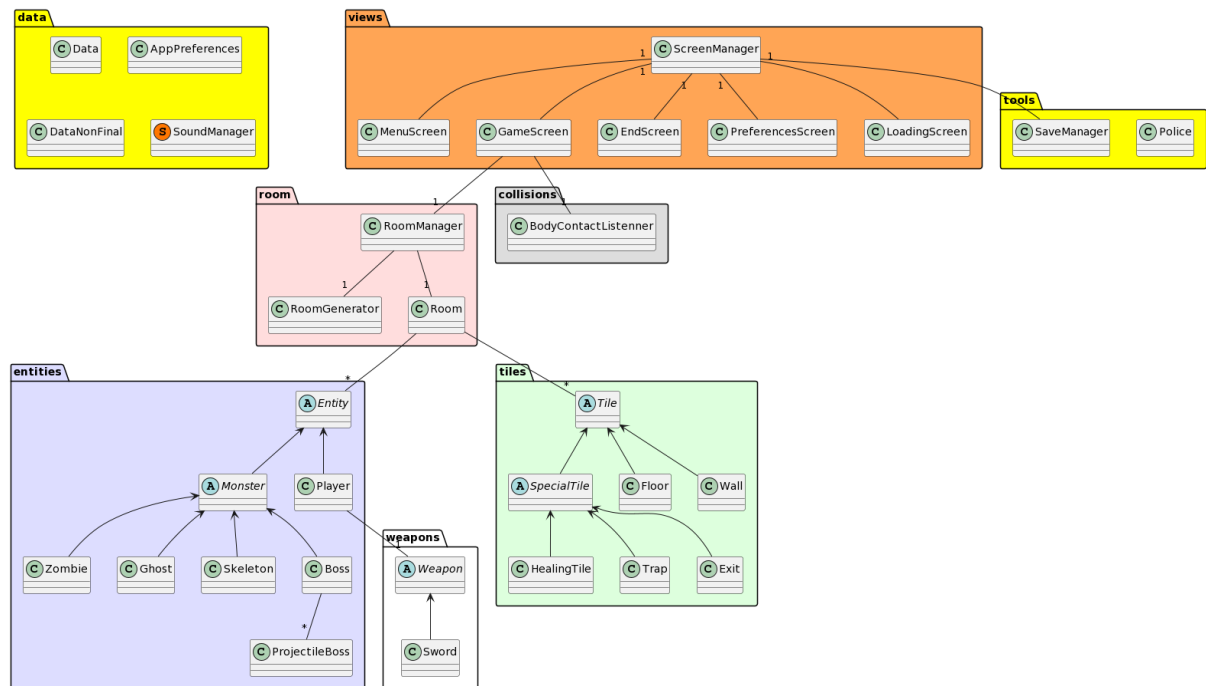
Résolution des problèmes du Sprint 2 :

- L'origine du problème était également lié à **la création et la position des sprites** en eux-mêmes. Le problème des sprites a donc été résolu en même temps que les soucis liés au **redimensionnement** de la caméra. **Dorénavant l'utilisateur peut redimensionner la fenêtre** comme bon lui semble, **le jeu occupera tout l'espace disponible sans distorsion**.
- **Le comportement de la porte** a également été résolu. Interagir avec la porte fermée n'altère plus le comportement de la porte une fois celle-ci ouverte.

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 les **diagrammes de séquence** nous ont permis de **mieux travailler** et de **mieux visualiser** le résultat que nous souhaitions.
- Malheureusement, par un manque de temps et des problèmes liés à JUnit et LibGDX, nous n'avons pas pu réaliser des tests unitaires. Cependant, nous avons effectué de nombreux tests en jouant directement. Après avoir implémenté une nouvelle fonctionnalité, une petite séance de test était effectuée. Pour les éventuels **prochains sprints**, nous réaliserons des **tests unitaires** avec JUnit.
- Nous souhaitons terminer le projet **sans problèmes ou bugs majeurs**, ce que nous avons **réussi**. Si nous devons continuer avec des nouveaux sprints, nous avons en tête de rajouter un système d'inventaire, des objets utilisables, des marchands, des nouvelles armes, des nouveaux monstres et des nouvelles tuiles. Une fonctionnalité qui nous plairait serait d'implémenter un multijoueur en écran scindé.
- Le dernier sprint **s'est très bien déroulé**. Nous avons actuellement un jeu qui nous plait et qui correspond à notre vision originale.

Diagramme de classe (actualisé) :



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

Si le joueur subit des dégâts, ce dernier est légèrement repoussé dans la direction opposée.

Monstres

13) Créer un monstre simple, se déplace aléatoirement

Dans un premier temps nous créerons un monstre basique qui se déplace de manière aléatoire.

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 tous les 5 étages. Se déplace aléatoirement dans une salle vide, puis tire des projectiles. Lorsqu'il a plus de 50% de sa vie, il tire des projectiles en vertical et en horizontal. Si sa vie est inférieure à 50%, il tire en plus des projectiles en diagonal.

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

Lorsqu'un monstre subit des dégâts, ce dernier est légèrement repoussé.

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

Le score s'incrémente lorsque le joueur tue des monstres/ramasse des objets. Son affichage doit être fait en haut à droite de la fenêtre.

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

Ajouter une musique pour l'écran titre, le jeu et le game over.

Ajouter des bruitages pour les différentes actions (attaque, dégât, mort).

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

Si le joueur touche la porte AVANT qu'elle soit ouverte, cette dernière ne fonctionne plus.

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.

BOHAN Thomas

JACQUES Louis

SERRATORE Alexandre

SUCK Hugo

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)