

CONCEPTION

Projet ACL / Groupe sboAa

Ce document de conception contient :

- *La liste de toutes les fonctionnalités implémentées*
- *La conception : tous les diagrammes de classes et de séquence réalisés à chaque sprint et également les diagrammes finaux.*
- *L'organisation de projet de notre équipe (distribution des tâches pour chaque sprint).*

Fonctionnalités

1. **Navigation dans le Labyrinthe** : Capacité pour le joueur de se déplacer dans un labyrinthe complexe, en utilisant les touches directionnelles (haut, bas, gauche, droite).
2. **Système de Combat** : Interaction avec les monstres par un système d'attaque en utilisant la touche ESPACE
3. **Découverte des items** : Mécanique de jeu permettant de trouver et de collecter des items tels que la clé et les cœurs. Ces derniers rechargent la jauge de vie du joueur lorsqu'elle est en dessous de 50% de sa capacité maximale.
4. **Progression et Niveaux** : Différents niveaux de difficulté avec une progression adaptative pour le joueur. A chaque déverrouillage de la porte grâce à la clé récoltée, le joueur passe automatiquement au niveau suivant.
5. **Interface Utilisateur** : Affichage des informations essentielles comme la santé du joueur et le nombre de monstres encore vivants

Sprint1

Backlog Sprint 1

Mise en Place de l'Environnement de Développement

La première étape cruciale de notre projet a été l'installation et la configuration de l'environnement de développement. Cette phase a inclus :

- **IDE** : Choix de l'Environnement de Développement Intégré (IDE) Eclipse qui est adapté au langage de programmation Java avec lequel nous allons développer le jeu.
- **Création d'un Dépôt Git** : Établissement d'un dépôt Git pour une gestion de version efficace et collaborative, marquée par un premier commit initial avec les noms de tous les membres du groupe.

Génération du Labyrinthe

Une partie fondamentale du jeu est la conception et la création du labyrinthe, ce qui implique :

- **Conception du Labyrinthe de Base** : Création d'un prototype simple de labyrinthe, intégrant des obstacles prédéfinis pour tester la logique de navigation.

- **Structure de Données pour le Labyrinthe** : Définition et implémentation d'une structure de données optimale pour représenter de manière efficace le labyrinthe dans notre code. Nous avons choisi la génération basée sur un fichier.txt

Review

Durant la review de ce premier sprint, nous avons réalisé que notre avancement n'avait pas été à la hauteur de nos attentes.

Cette expérience a été un point tournant. Nous avons pris le temps de réévaluer notre approche, de clarifier nos objectifs et de renforcer notre compréhension du projet. En conséquence, dans le sprint 2, nous nous sommes concentrés sur une progression plus structurée et ciblée, ce qui a entraîné une amélioration significative tant en termes de développement technique que de collaboration d'équipe.

Ce processus d'apprentissage et d'adaptation a été essentiel pour les succès ultérieurs de notre projet, démontrant notre capacité à surmonter les obstacles et à tirer des leçons de nos expériences.

Sprint2

Backlog

Objectifs

Notre groupe n'ayant pas progressé durant le sprint 1, nous avons défini les tâches essentielles pour progresser :

- créer les classes Main, GamePanel, Entity (Aurélien, Mohamed)
- créer la première version de l'interface (Hajar)
- Initialiser le plateau (labyrinthe, héros placé aléatoirement, génération de monstres aléatoirement) (Omar)



a/ niveau 1

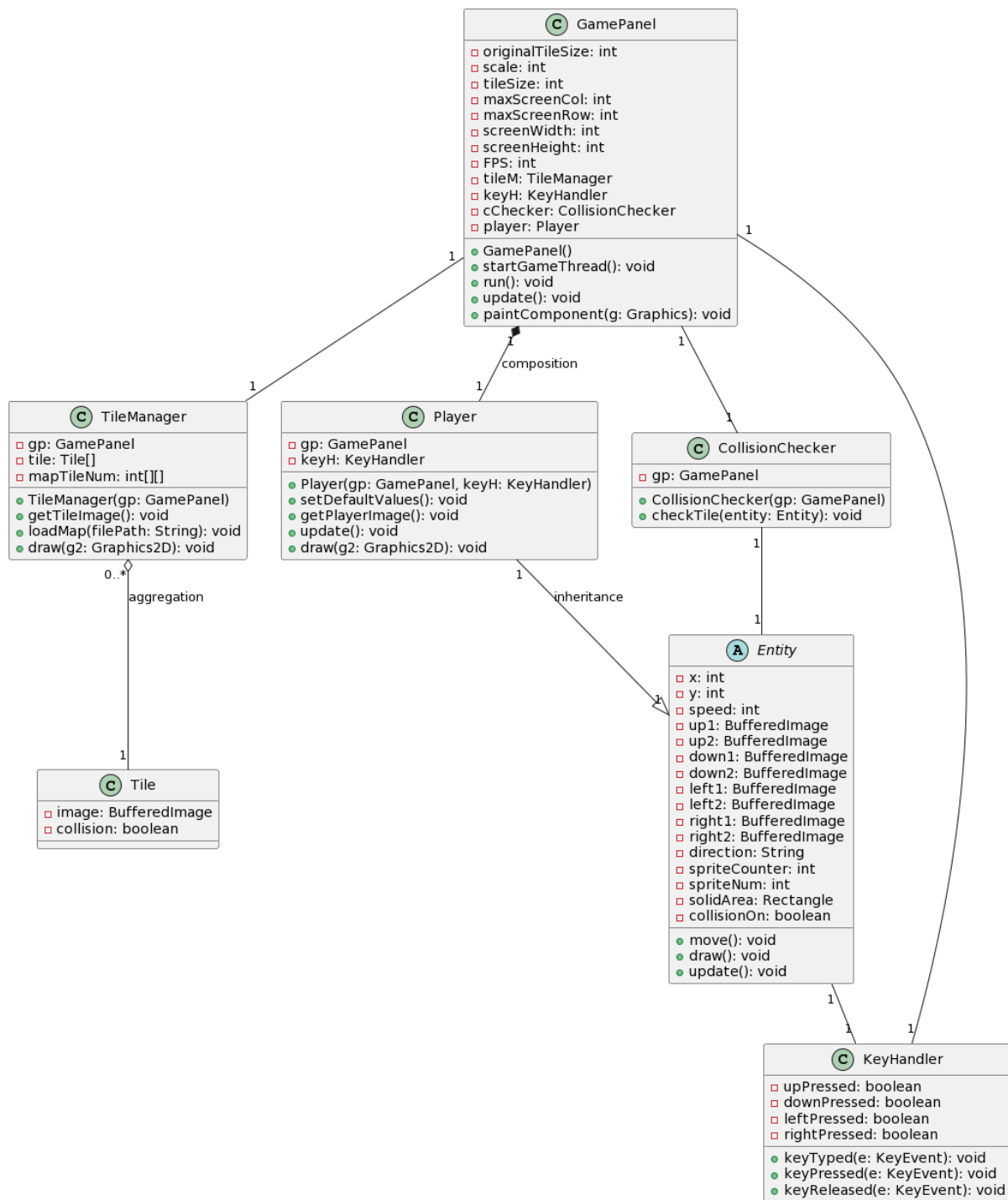


b/ niveau 2

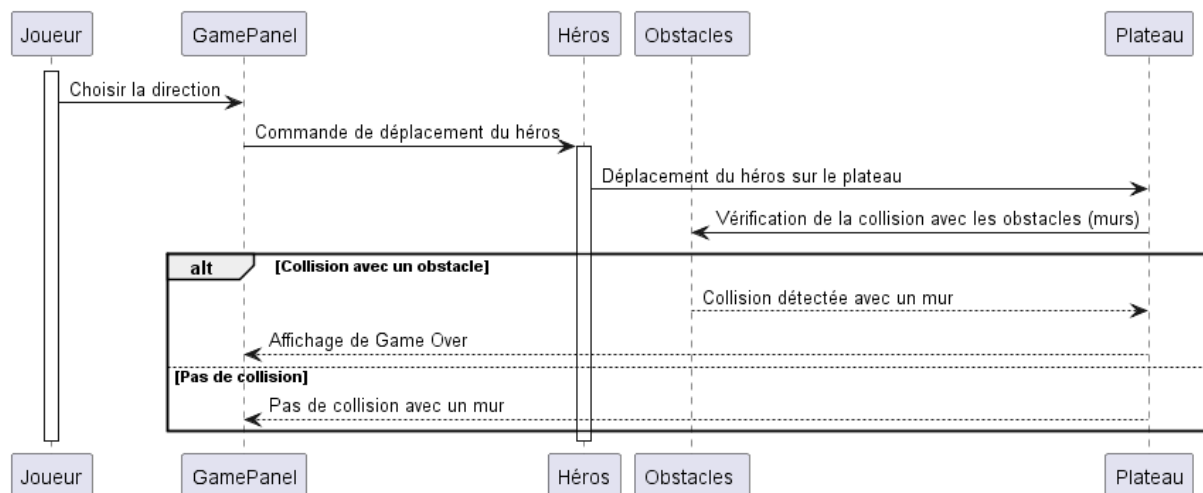


c/ niveau 3

Diagrammes



a/ Sprint 2 Diagramme de classe



b/ Sprint 2 Diagramme de séquence

Review

On a bien avancé sur le jeu, il nous manque le monstre et les items. Il faut aussi réfléchir à comment implémenter les différents niveaux de jeu et créer différents map pour chaque niveau.

La génération de la map grâce au fichier texte est réussi et le héro se déplace fluidement avec les flèches de l'ordinateur.

Retour suite au sprint par Mme Phuc Ngo :

- Mieux définir les objectifs entre sprints.
- Se répartir les taches de manière efficace.
- Développer les reviews
- Revoir le diagramme de séquence

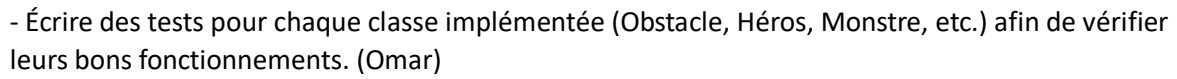
Sprint3

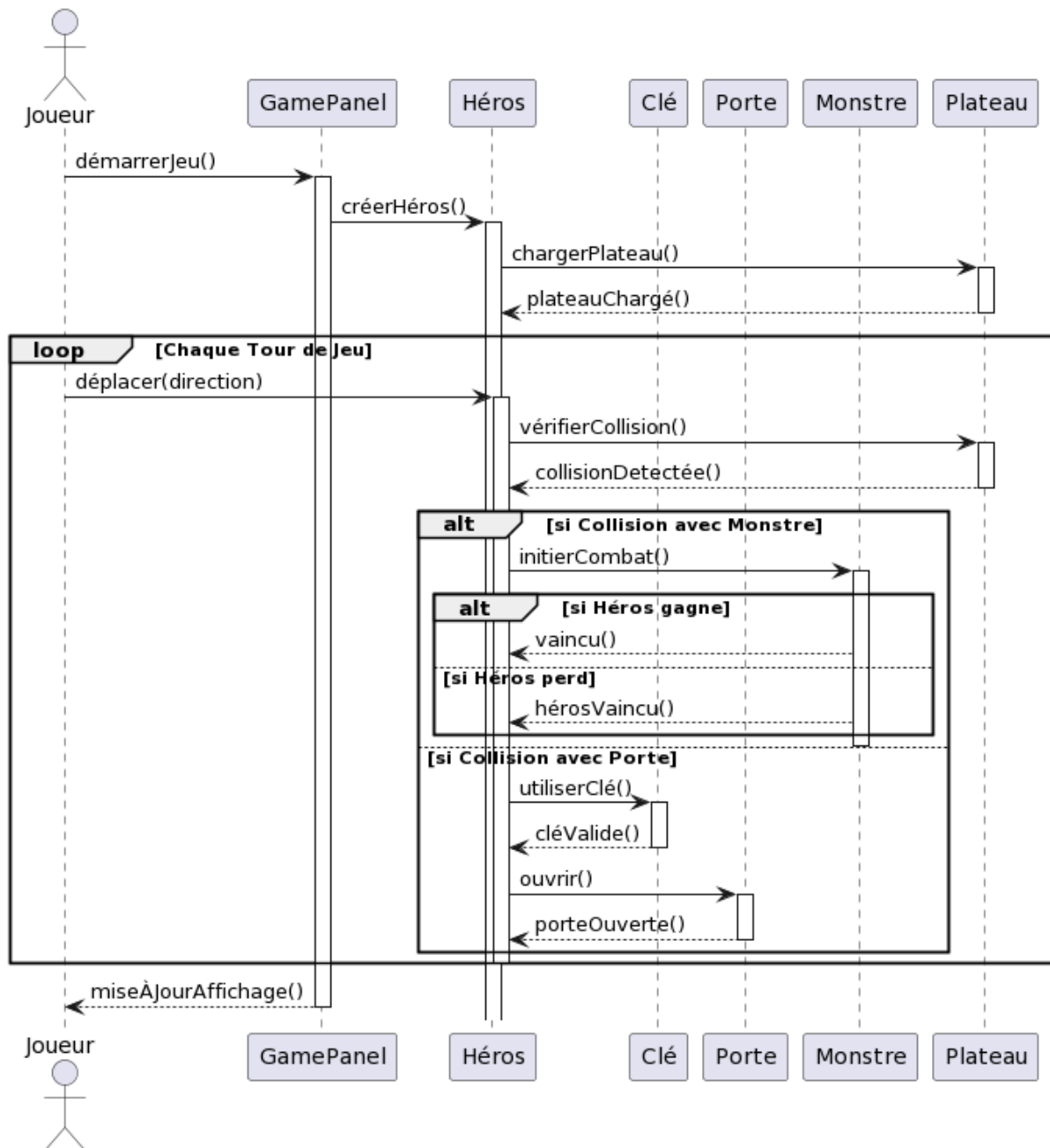
Backlog

Fonctionnalités du jeu

- Gérer les collisions avec les murs et les obstacles. (Aurelien)
- Changer/améliorer l'élément graphique du héros et ajout des nouvelles maps de niveaux supérieurs. (Mohamed)
- Rajouter de nouveaux éléments graphiques de pour représenter les Trésors/récompenses dans l'interface. (Mohamed)
- Rajouter les points vies du Player. (Hajar)

Mécanismes de jeu





d/ Sprint 3 Diagramme de séquence

Review

Ce qui a été réalisé :

Nous avons rencontré des problèmes lors de

- Implémentation de la fonctionnalité permettant au joueur de se déplacer dans le labyrinthe en utilisant les touches directionnelles.
- Gestion des collisions avec les murs et les obstacles.
- Ajout de nouveaux éléments graphiques pour représenter les trésors/récompenses dans l'interface.

- Intégration des points de vie du joueur.

Ce qui n'a pas été réalisé :

- Tests JUnit pour toutes les classes.

Retour suite au sprint par Mme Phuc Ngo :

Revoir les diagrammes de classe et de séquence : mettre tous les attributs de chaque classe et mettre les mêmes noms de fonctions que dans le programme

Sprint4

Backlog

Répartition

- Rajouter le monstre et gérer son déplacement aléatoire.(Mohamed)
- Mettre la collision entre le joueur et le monstre pour un gameplay plus fluide et logique. (Aurélien)
- Ajouter un système de points de vie pour le monstre.(Omar)
- Gérer le déroulement d'un combat entre le joueur et le monstre, incluant la perte de points de vie et la mort du monstre en cas de défaite. (Mohamed / Hajar)

Intégration du Monstre Aléatoire

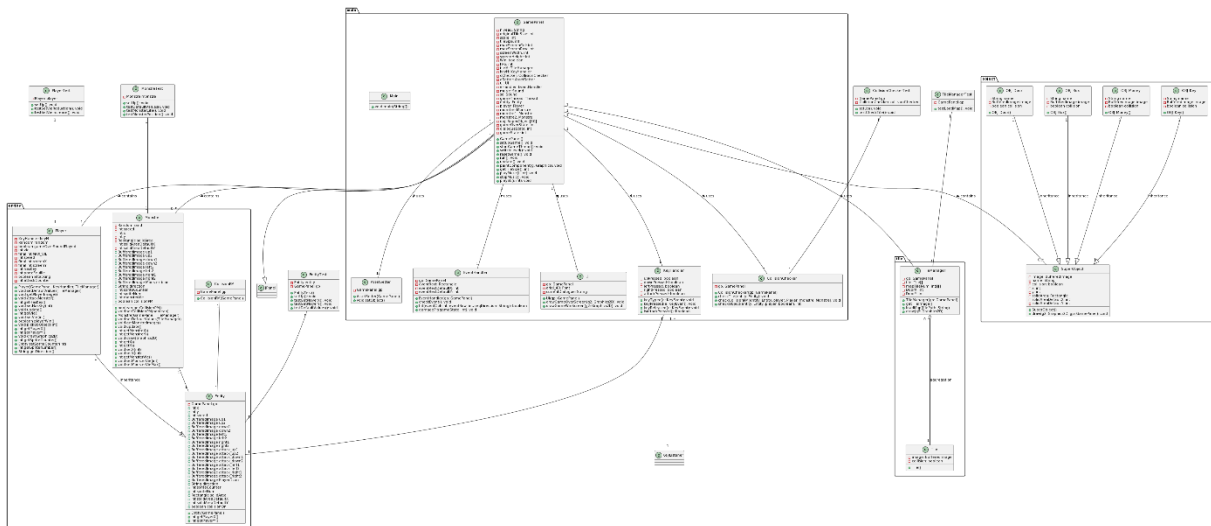
Pour ajouter un élément de défi, nous avons intégré un monstre se déplaçant aléatoirement dans le labyrinthe :

- **Création des Monstres** : Développement d'un monstre avec un comportement de déplacement aléatoire au sein du labyrinthe qui diminuent la jauge de vie du joueur lorsqu'il y a collision.
- **Détection de Collision** : Implémentation d'un mécanisme pour identifier les rencontres entre le héros et le monstre, essentiel pour la dynamique de jeu.

Tests avec JUnit

- Écrire des tests unitaires pour chaque classe implémentée jusqu'à présent afin de garantir leur bon fonctionnement et leur intégrité. (Toute l'équipe)

Diagrammes



e/ Sprint 4 Diagramme de classe final

Review

On a pu régler le bug de frontière mis en évidence par notre chargé de TD durant le TP Sprint 3. Nous avons également amélioré les fonctionnalités du joueur en ajoutant un système d'attaque. De plus, le nombre de monstres a augmenté.

Une touche musicale a également été ajoutée avec la musique du jeu. En effet, des sons ont été assignés lorsque le joueur collecte les items ou encore déverrouille la porte.

Conclusion

Pour conclure, nous avons énormément appris de ce projet que ce soit au niveau du développement ou au niveau du travail d'équipe. Nous avons programmé notre premier jeu vidéo en apprenant de nous-même et en partant de zéro. La répartition équitable du travail et l'avancement constant n'était pas une chose aisée mais nous avons su tirer les leçons et nous amélioré au fur et à mesure des différents sprints.

Le résultat final est à la hauteur des efforts fournis et nous sommes impatients d'en apprendre davantage lors de prochains projets.