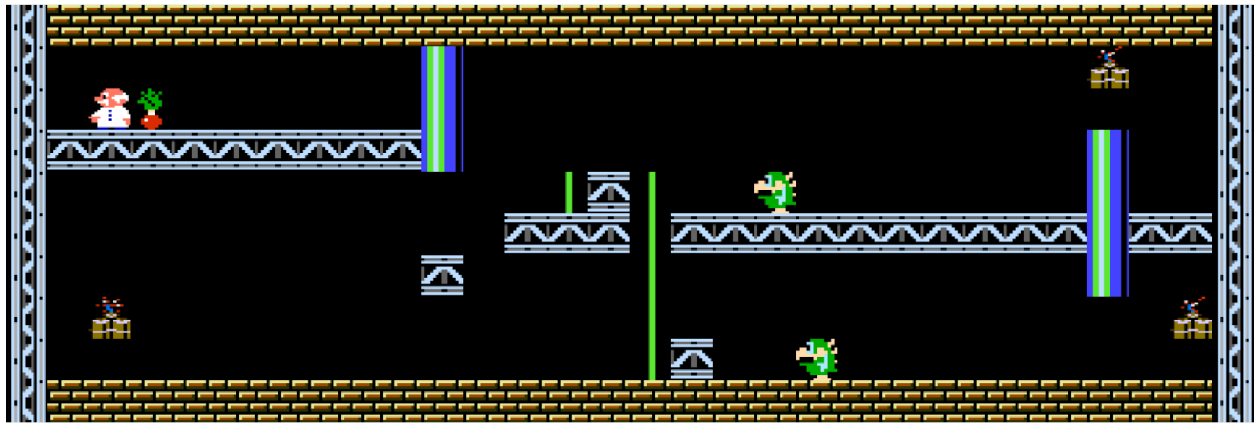


# Rapport Projet LifaPOO

Gyromite - Automne 2022

Nicolas Patino – Juan C. Lenis



## Liste des fonctionnalités :

### - Fonctions de base :

1. Gravite : Les Entités dynamiques tombent si elles sont suspendues dans le vide. Temps : 0min (déjà implémentée dans le code fourni).
2. Gestion des collisions : Les Entités Dynamiques (DynaEnt) ont des traitements différents en dépendant du type de l'Entité avec laquelle elles font collision (Docteur/Smick => Game Over ; Docteur/Corde => Docteur peut monter la corde ; etc...). Temps : +/- 2h.
3. Capacité de monter/descendre corde : Les DynaEnt sont capables de monter ou descendre en utilisant les cordes. Cette fonctionnalité dépend aussi de l'implémentation de SuperEntite (cf. plus bas). Temps : 30min + Temps(SuperEntite).
4. Capacité à prendre les Bombes : En cas de collision avec le docteur et une bombe, on enlève la Bombe de la grille et on modifie le compteur de bombes en conséquence (condition de victoire). Temps : 30min.
5. Piliers : Le joueur peut déplacer les piliers (les descendre ou monter) en appuyant sur une touche (si appuyée : piliers descendent, sinon : ils montent). Les piliers peuvent aussi écraser et déplacer les DynaEnt sous les bonnes conditions (DynaEnt sur pilier quand il remonte est déplacé vers le haut ; DynaEnt sous un pilier quand il tombe est écrasé s'il y a un support sous elle). Temps : 3h.
6. IA Smicks : Les Smicks se déplacent tout seuls et sont capables d'utiliser les cordes. Temps : 3h.
7. SuperEntite : Pour permettre la superposition de deux Entités dans la grille du jeu on a implémenté la classe SuperEntite (fille

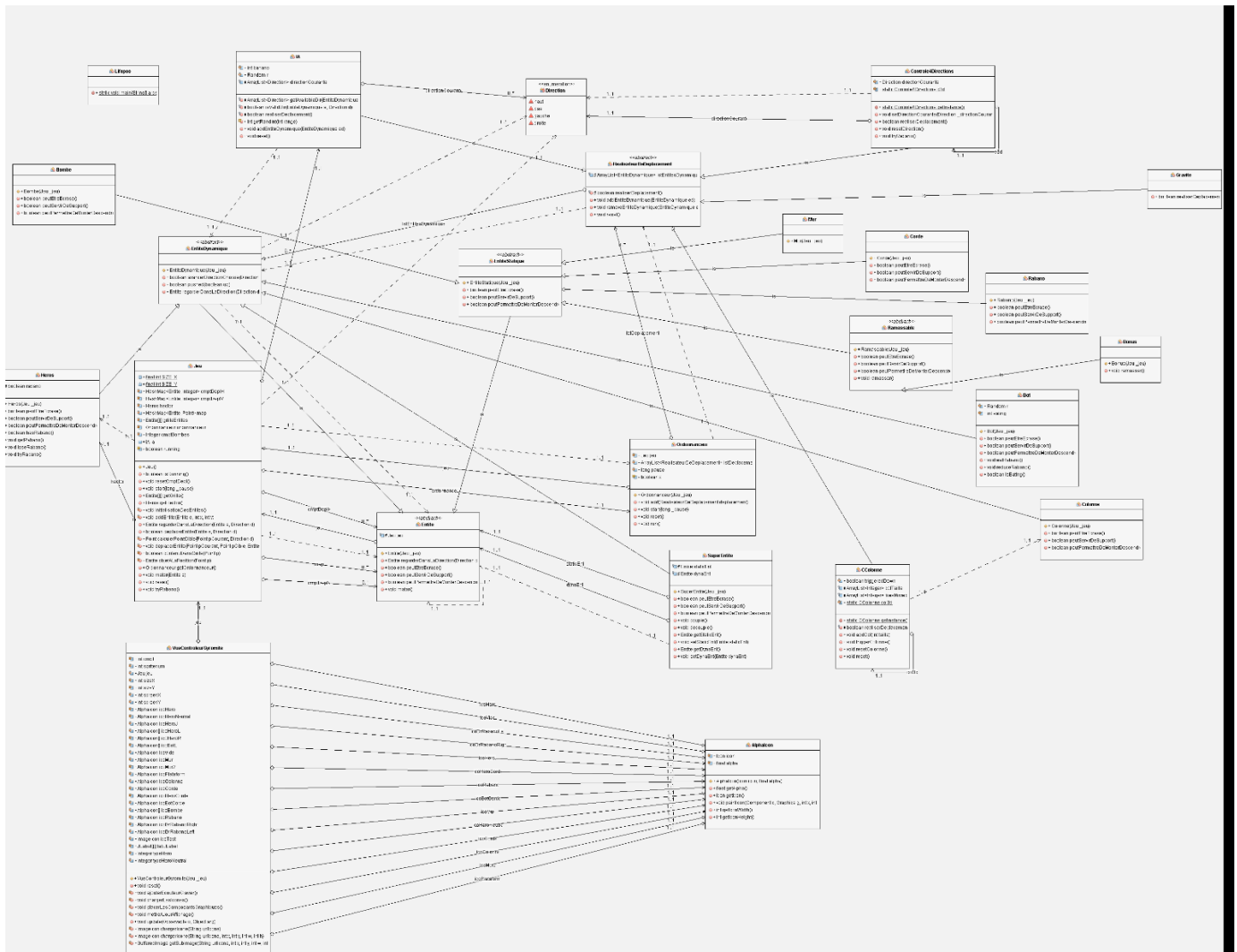
d'Entité) qui contient une référence à une Entité Statique (StaticEnt) et une à une DynaEnt. On applique différents traitements avec les collisions en dépendant des Entités qui le composent. Temps : 1h.

8. Game Over/Victoire : La partie « fini » sous les bonnes conditions. En réalité cela est juste affiché sur la console et on peut redémarrer une nouvelle partie en appuyant sur la touche correspondante. Temps : 15 min.

## - **Extensions :**

1. Radis : Les radis sont des Entités Statiques que le joueur peut ramasser et emporter avec lui (1 max.) et les déposer dans d'autres positions. Si un Smick collisionne avec un Radis il est distrait pendant quelques secondes. Temps : 40min.
2. Transparence : Pour la gestion de la transparence des sprites on a changé les images Icon par des Alphalcon, cette class permet de changer l'intensité de l'image par rapport a une valeur donner et de permet la superposition d'images dans l'affichage. Temps : 1h30.
3. Animations : Les animations fonctionne par rapport au temps, on a créé un compteur qui se réinitialise quand le nombre de images est fini et qui fonctionne depuis le début du jeu, ce compteur permet le changement d'image des entités dans l'affichage. Pour chaque animation les images sont stocker dans un tableau de Alphalcon.

Le diagramme est plus lisible dans UML.png



## Images du jeu

