Université de Paris

RAPPORT DU PROJET POO
Pet Rescue Saga

Ismail Badaoui : 21964109
Skander Jeddi : 21957008

05/01/2021
POO – MATH-INFO1 – S3
UNIVERSITÉ DE PARIS

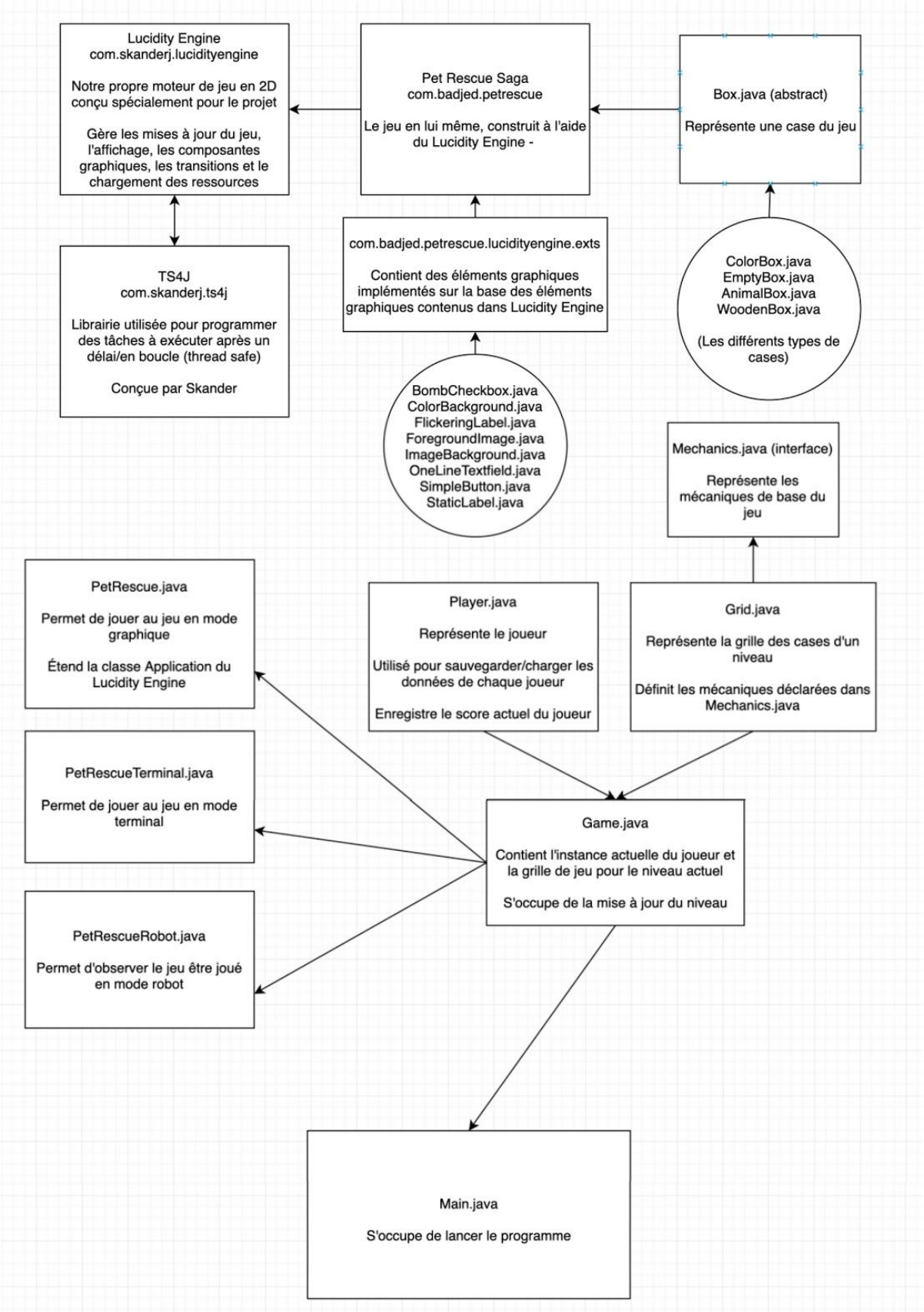
SOMMAIRE

INTRODUCTION

Le projet dont nous sommes en charge consiste à réaliser un jeu (PET RESCUE saga) en **Java**. Ce jeu sera simplifié par rapport aux règles du jeu normal, car il doit répondre aux contraintes du sujet. En effet, le but est de réaliser un moteur permettant de jouer à Pet Rescue avec la bonne utilisation des connaissances vues en cours et les principes fondamentaux de la programmation objet, c'est à dire l'**héritage** entre les classes, la gestion des **interfaces**, le **polymorphisme**, les **Threads**, etc. Nous présentons alors dans ce rapport notre démarche en décrivant les principaux axes de développement avec **Diagramme** à l'appui en p.3,

Le jeu commence par une interface simple (**terminal**) dans laquelle le joueur est invité à choisir entre jouer en **terminal**, mode robot, ou en interface graphique. Il est invité ensuite de créer un pseudonyme ou choisir son pseudonyme s'il est ancien joueur, après il choisit le niveau souhaité ou bien le mode de jeu infini. Ensuite la partie commence, sur l'image en p.8 on observe un plateau du jeu qu'on a développé. On y distingue différents blocs de couleurs, et parfois à leur sommet des animaux y sont perchés. L'objectif est de les amener à les faire disparaître "sauver". Pour y arriver le joueur va sélectionner l'une des cases de couleurs et, dès lors que le nombre de cases contiguës de même couleur est suffisant, celles-ci disparaîtront. Le reste du plateau se réorganise pour proposer une nouvelle configuration (les blocs se déplacent de haut en bas et de droite à gauche), et le jeu continue ainsi de suite. Pour passer au niveau suivant il faut impérativement gagner le niveau antérieur, en vous donnant des **POWER UPS** (en forme de bombe détruisant les blocs 3x3) à la fin de chaque niveau gagné.

DIAGRAMME DES CLASSES



FEATURES DU PROJET

1. **Animaux**

   Nouveaux animaux de compagnie: sauvez des animaux de toutes formes et tailles, vous rencontrerez toutes sortes de créatures au cours de votre voyage, des Pandas des chats aux petits cochons!

2. **Mode infinie**

   nouveau mode de jeu qui vous permet de jouer jusqu'à l'infinie et d'avoir de meilleurs scores ainsi que gagner des "Bombes" bonus.

3. **Mode terminal**

   Mode spécial pour les développeurs où vous pouvez jouer en terminal sans avoir besoin d'interface graphique.

4. **Mode robot**

   Vous voulez vous initier et apprendre à jouer comme un pro, le mode robot vous aidera à apprendre les meilleurs coups et stratégies possibles seulement en le regardant.

5. **Animation**

   l'animation du jeu a été principalement travaillée par des sons, l'interface débute par un son celtique qui définit bien le monde du jeu, un effet sonore spécial pour les coups (explosion des bombes et explosions des blocs). Et un son caché que vous allez découvrir en jouant en mode Horreur. On vous laisse découvrir.

6. Image

Création du logo inspiré du jeu original ainsi que l'icône accompagnée.





Création d'images d'animaux choisis en plus des blocs colorés et de la bombe, travailler à l'aide d'un outil d'édition d'image **PIXLR**. Les arrières plans sont des images à libre droit qui suivent le thème du jeu.

7. Scène de crédit

Une scène ou on vous remercie d'avoir essayer notre jeu et on vous incite à trouver les deux **easter eggs** cachés dans le jeu.

8. Easter egg

Un jeu vidéo sans easter egg sera une sacré déception pour **Warren Robinett.** (Référence à Marvel et au mode HORREUR).

9. Power Ups

POWER UPS vous seront attribuées (en forme de bombe détruisant les blocs 3x3) à la fin de chaque niveau gagné. cela faciliterait le jeu pour les niveaux supérieurs

10. Statistiques

Une fois vous avez un pseudonyme et que vous avez joué, vos meilleurs scores dans chaque niveau seront visible dans la rubrique statistique

11. Les niveaux

Plus on avance dans le jeu de 1 à 6 plus le niveau de complexité se renforce. Le 1er niveau est basique et simple pour se familiariser avec l'interface, le 2eme niveau on y retrouve des blocs de bois difficile à casser, 3eme niveau on a que 3 coups à faire pour sauver les animaux ainsi que des blocs en bois et ainsi de suite.

12. Save & Load

Notre programme offre les fonctionnalités suivantes :

Gestion du profil : si le joueur a déjà joué sur le jeu, il choisit son nom et l'application se charge de lui créer une nouvelle partie, sinon il choisit un nom et le joueur va être enregistré sur la base de données.

Traçabilité : l'ensemble des joueurs ainsi que leurs niveaux atteint et le meilleur score atteint dans chaque niveau.

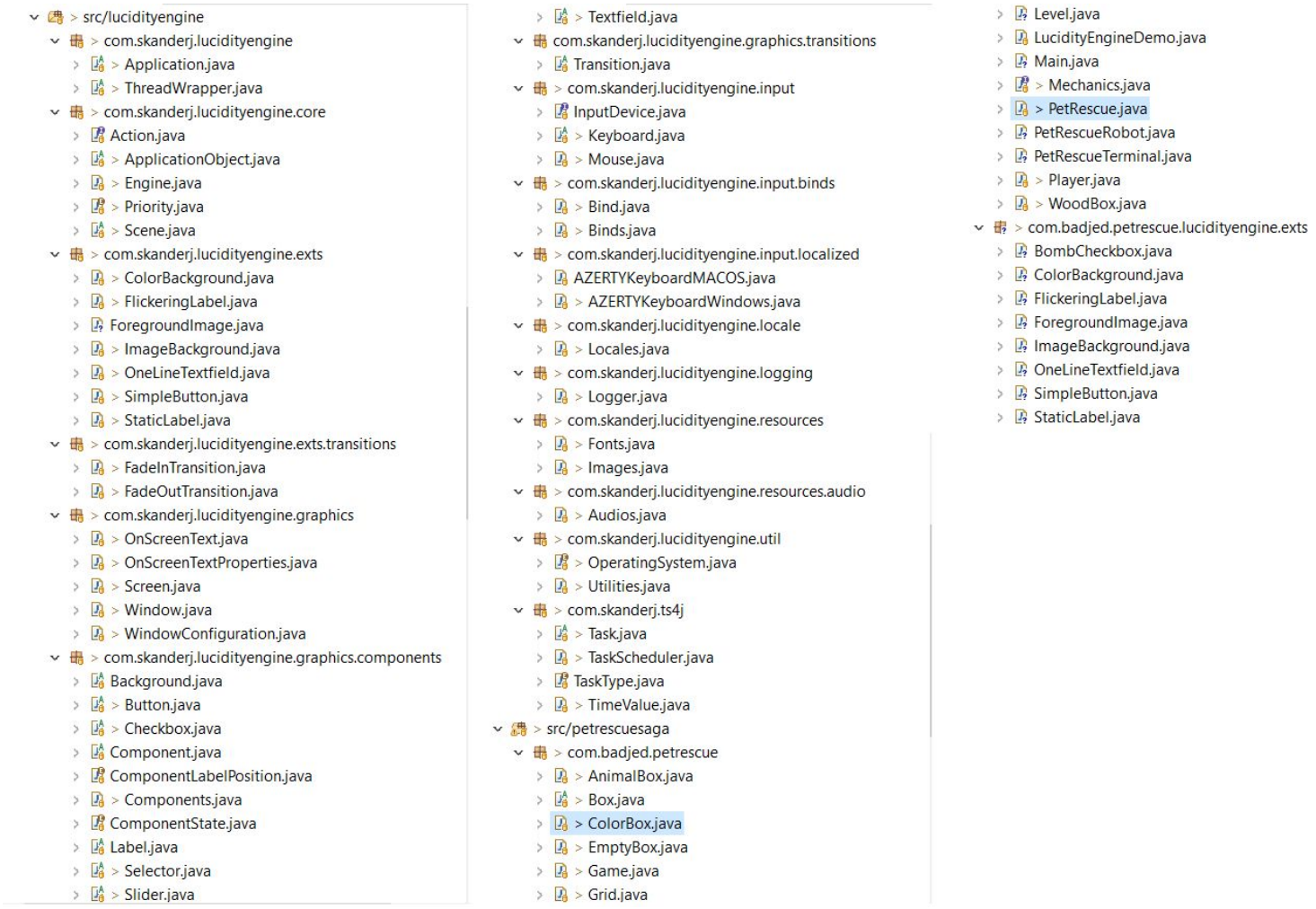
ANALYSE DU TRAVAIL

Un des objectifs principaux de ce projet a été de nous habituer au travail en binôme et aux notions de projet. Pour mener à bien ce travail on était obligé d'organiser notre temps, établir les tâches à faire pour chacun ou encore nous obliger à respecter l'énoncé du projet. Nous avons dans un premier temps travaillé ensemble sur les premières classes à créer afin d'établir un socle commun.

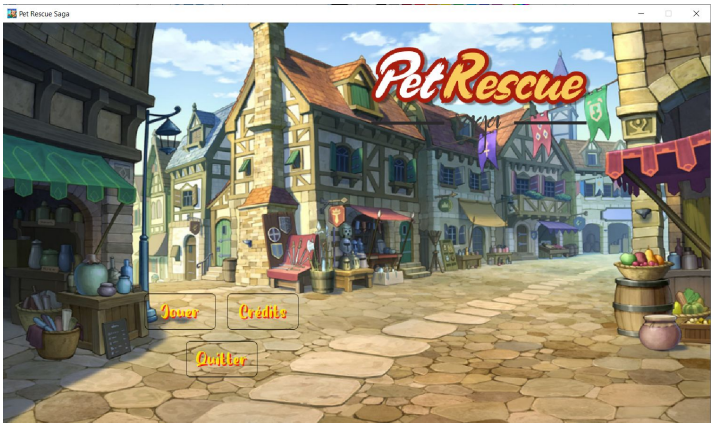
Concernant la plateforme de développement on a utilisé **Eclipse** et **INTELLIJ IDEA** comme principaux **IDE**, vu qu'ils facilitent le travail et la gestion des fichiers ainsi que pour les **build.**

Création du moteur de jeu "**Lucidity Engine**" pour mettre en pratique nos connaissances vues au cours de ces 3 semestres. Pour structurer notre projet on a implémenté le patron de conception **MVC (Model View Controller)** cela nous permettra de séparer les interactions de l'utilisateur, l'accès aux données et l'affichage des informations.

on a travaillé ensemble tout au long de cette période, malgré quelques difficultés rencontrées car en cette période de crise sanitaire on était obligé de travailler à distances et communiquer via **discord**, et on a créer un **repository** sur **GitHub** pour pouvoir **Commit et Pull** les nouvelles mises à jour du code et de travailler en collaboration d'une manière efficace.



Résultat



Menu principal

Choix des niveaux, mode infini et voir les statistiques

Plateforme du premier niveau

DIFFICULTÉS TROUVÉES

1. Mauvaise connexion internet qui nous a empêché de bien communiquer.
2. Les bugs, compilations échouées, erreurs mystérieuses.
3. Les tests à distance.
4. Première fois qu'on crée des images et logo ainsi que l'icône de l'application.
5. Devoir choisir les sons pour le jeu qui vont avec le thème.
6. Choix de l'arrière-plan qui n'était pas aussi trivial qu'on pensait.

CONCLUSION

Ce projet s'est révélé enrichissant pour chacun d'entre nous dans la mesure où nous avons pu découvrir et réaliser un jeu et de la voir aboutir. Ce fut enrichissant d'un point de vue culturel, mais aussi immensément d'un point de vue technique. Ce projet nous a permis d'ancrer nos connaissances en **Java** (**héritage**, **encapsulation**, **polymorphisme**, **abstraction**) et cela a été l'occasion de nous familiariser avec la programmation graphique, le **MVC**, mais aussi de se construire un environnement de développement efficace et adapté à nos besoins.

Nous avons utilisé pour ce projet : Docs de google pour la rédaction de rapport du projet et PIXLR pour la manipulation des images.

RÉFÉRENCES

- **Musique:**

  . Black Wolf's Inn par Derek Fiechter

  . Hospital Of the Dead par Revolt Production Music

GLOSSAIRE

CLASSE ABSTRAITE: Est une classe dont l'implémentation n'est pas complète et qui n'est pas instanciable.

DIAGRAMME DES CLASSES: Il détaille le contenu de chaque classe mais aussi les relations qui peuvent exister entre les différentes classes.

EASTER EGG: est, en informatique ou dans les jeux vidéo, une fonction cachée au sein d'un programme.

ECLIPSE: Est un environnement intégré de développement (IDE)

ENCAPSULATION: Est une manière de définir une classe de telle sorte que ses attributs ne puissent pas être directement manipulés de l'extérieur de la classe, mais seulement indirectement par l'intermédiaire des méthodes.

FEATURES: en français « une fonctionnalité » est une action particulière qui amène une possibilité supplémentaire.

HERITAGE: Héritage est un mécanisme qui permet, lors de la déclaration d'une nouvelle classe, d'y inclure les caractéristiques d'une autre classe.

INTELLIJ IDEA: Est un environnement intégré de développement (IDE)

INTERFACE: Une interface est un ensemble de signatures de méthodes publiques d'un objet.

JAVA: Java est un langage de programmation orienté objet

MVC: Modèle-vue-contrôleur

POLYMORPHISME: Des fonctions et des méthodes de mêmes noms peuvent avoir des comportements différents ou effectuer des opérations sur des données de types différents.On distingue 2 types de polymorphisme, la surcharge et la redéfinition.

POO: Programmation orienté objet

POWER UPS: (en français « augmentation de puissance »), parfois bonus, est un objet qui permet en général d'améliorer son armement dans les jeux vidéo et notamment les shoot them up.

PIXLR: est un logiciel d'édition et de retouche photo gratuit, composé de deux outils distincts, accessible sur un navigateur web.

TERMINAL: Le terminal est un point d'accès de communication entre l'homme, un ordinateur central ou un réseau d'ordinateurs.

THREAD: Fil ou tâche est similaire à un processus car tous deux représentent l'exécution d'un ensemble d'instructions du langage machine d'un processeur.

WARREN ROBINETT: Joseph Warren Robinett, Jr., né le 25 décembre 1951 à Springfield, est un concepteur et programmeur de jeux vidéo américain.

LIENS

https://stackoverflow.com/

https://www.geeksforgeeks.org/

https://github.com/

https://fr.wikipedia.org/

https://fr.wikibooks.org/