Equipe : Louka DOZ, Quentin RAMSAMY-AGEORGES, Loïc SENECAT

Livrable : Une application qui permet de jouer à un escape game et qui permet d’en créer.

On considère comme états :

* <vide> : non commencé
* S : commencé
* P : en pause
* E : terminé
* D : Abandonné

MoSCoW est un ordre de priorités (Must Should Could Would) d’une tache, de M les taches à haute priorité jusqu’à W les taches optionnelles.

**Liste des taches**

|  |  |  |
| --- | --- | --- |
| MoSCoW | Description | États |
| M | Regrouper et pouvoir lister les catégories (Personnages, Décors, Actions, Salles) d’entités (pnj1, livre, panneau, interrupteur, porte, …) |  |
| M | Ajouter des actions à une entité (case de la map, joueur, panneau, …) |  |
| M | Définir le lieu de début/fin |  |
| M | Placer une entité sur la map |  |
| S | Version graphique (faire wireflow & wireframe), Glisser-déposer éléments sur la map, tri des éléments déposables par catégories… |  |
| C | Cacher les zones et définir un déclencheur pour les afficher |  |
| C | Faire pivoter les salles |  |
| C | Personnages non-joueurs peuvent vendre des objets (donc on peut définir quoi) |  |
| C | Définir les conditions de fins personnalisées. |  |
| W | Importer une carte en ligne pour l’éditer |  |
| W | Ajouter ses propres éléments (sprites, …) |  |
| W | Possibilité d’ajout de scénarios |  |
| W | Pouvoir lancer une simulation de l’escape game pendant la création |  |

Prochainement toutes les fonctionnalités du jeu.

Spécification technique :

Java 10 swing, Libgdx, modèle MVC, méthode scrum

IntelliJ, Github

Anglais code, Français pour la doc et commentaire

Plateforme : PC

Modèle de données : fichiers et/ou BD (SQLite)