ENIGMA

Escape game 2D



**Cahier des charges**

Créez vos propres escape game, et jouez s’y seuls ou à plusieurs dans ENIGMA !

Sommaire :

1. Présentation du projet et de l’équipe 3

2. Spécifications fonctionnelles 4

a. Créateur d’escape game 4

b. Lanceur d’escape game ?

3. Spécifications techniques 7

a. Langage et Framework 7

b. Architecture 7

c. Modèle de données 7

d. Conventions 8

4. Charte Graphique 8

5. Planning 8

6. Annexes 9

a. Bilans des réunions 9

b. Échanges avec le tuteur ?

7. Remerciements et crédits 11

1. Présentation du projet et de l’équipe

Dans le cadre du projet tuteuré de 2e année en informatique, nous avons choisi de réaliser un jeu, du style escape game, intitulé "Enigma". L’enseignant [Luc Hernandez](mailto:luc.hernandez@u-pec.fr) est le tuteur. (Merci !)



Annexe 1 : Game design Document (GDD) décrivant les caractéristiques d’Enigma

**Livrable** : Le produit attendu à la fin de ces 5 mois est une application qui permette de créer des escapes games, et d’y jouer.

**L’équipe** :

* Jorys-Micke ALAIS : programmeur
* Louka DOZ : programmeur, chef de projet, Gérant du sprint board
* Quentin RAMSAMY-AGEORGES : programmeur, Scrum master
* Loïc SENECAT : programmeur

Postes à pourvoir : responsable des réunions (actuellement Quentin)

1. Spécifications fonctionnelles

Cette catégorie n’est en aucun cas finie, toutes les spécifications ne sont décidées que pour la partie "créateur de l’escape game".

Les spécifications ont une priorité selon le fonctionnement MoSCoW *(ordre de priorités (Must Should Could Would) d’une tache, de M les taches à haute priorité jusqu’à W les taches optionnelles)*.

* 1. Créateur de l’escape game

|  |  |
| --- | --- |
| Priorité | Description |
| 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 de l’escape game. |
| 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 |

Les étapes seront approfondies lors de leur sélection par l’équipe de développement.

**L’objectif est de rendre réalisable le scénario suivant** :

Monsieur Admin lance l’application et clique sur éditeur de niveau. La liste de ses cartes est vide et il clique sur « nouvelle niveau ».

L’écran de création s’affiche avec la catégorie « salles » sélectionnée. Il sélectionne la première salle de la catégorie « type 1 » et la place au centre de sa map. Il place à gauche et à droite de cette salle, une salle  « type 2 ». Il supprime le mur entre la salle de type 1 et sa voisine de droite.

Il se déplace dans la catégorie « objets » et ajoute sur la map une porte entre la salle de type 1 et sa voisine de gauche.

Il clique sur la catégorie personnages, et déplace le personnage n°1 de type femme dans la salle à gauche de type 1. Il lui ajoute une action de la catégorie action : dialogue (« vous m’avez trouvée ! ») et une action qui la fait disparaitre après avoir prononcé son dialogue.

Il sauvegarde la map.

Il clique sur la porte et ajoute une action « clef » qui oblige le joueur à avoir un objet clef pour ouvrir la porte.

Il sélectionne la catégorie objets, et place deux interrupteurs dans la salle de type1, un dans chacun de ses sommets supérieurs. Il ajoute un panneau entre les deux interrupteurs et donne au panneau le contenu suivant « le bon interrupteur est celui de droite - Bill Gates ».

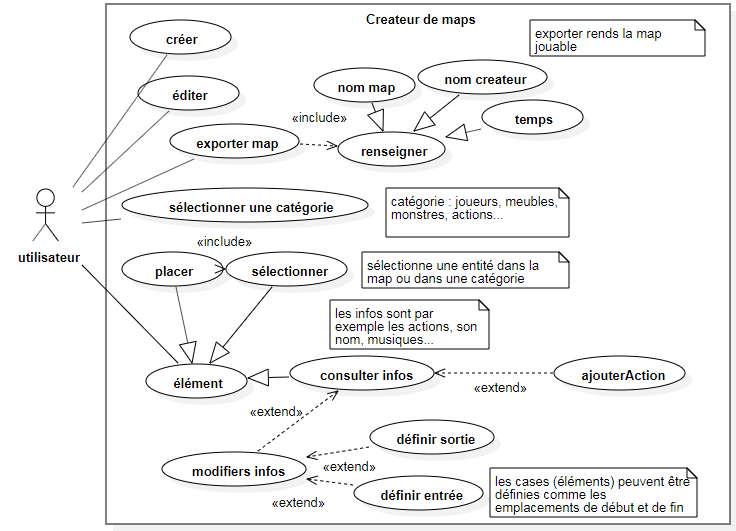
Il met un coffre dans la salle à droite de celle de type1 et une clef à l’intérieur. Il lui ajoute une action qui ne l’ouvre que si l’interrupteur gauche a été activé sinon il affiche un message « un coffre fermé ».

Il place ensuite la sortie derrière le personnage femme et l’entrée au centre de la salle de type1.

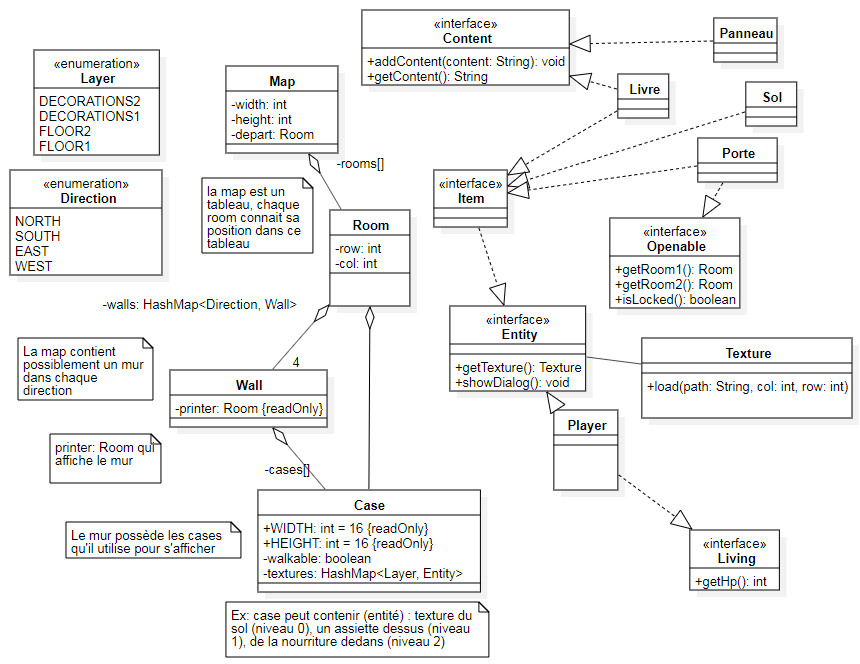
Il exporte la map en lui donnant le nom « map alpha », le créateur « admin » et sa durée « 5 min ».



Annexe 2 : Idée du rendu graphique du scénario



Annexe 3 : Diagramme des cas d’usages de l’utilisateur avec le créateur



Annexe 4 : Diagramme de classes de la structure, seulement quelques méthodes

1. Spécifications techniques
   1. Langage et Framework

Le langage utilisé sera le JAVA 10, avec sa bibliothèque graphique swing pour l’application de création d’escape game et le framework LIBGDX pour la partie jeu.

L’application sera codée en suivant le modèle MVC et en appliquant la méthodologie de développement AGILE-SCRUM.

Le logiciel sera disponible sur PC, toutes versions, tous OS.

* 1. Architecture

L’application sera codée dans le logiciel Intellij IDEA.

Elle sera sauvegardée sur git hub : <https://github.com/LoukaDOZ/Enigma>

* LoukaDOZ
* QuentinRa
* SenecatLoic
* Niskey77

Aux niveaux de l’organisation des dossiers :

* Enigma : racine du projet
  + assets : toutes les ressources, triés par catégories (hud, utils, …)
  + libs : les bibliothèques (libgdx, annotations)
  + src : contient les sources du projet
  + documents : les documents du projets (rapport…)
    - editor : les sources de l’éditeur
    - game : les sources du jeu

Le site trello sera utilisé pour le scrum board : <https://trello.com/b/pTGVfrp6/enigma>

* 1. Modèle de données

Les données seront sauvegardées dans un/plusieurs fichiers ou dans une base de données telle que SQLite. *(en réflexion)*

* 1. Conventions

On a convenu de conventions telles que :

* le code est rédigé en anglais
* la documentation est en français
* Utiliser les annotations (@NotNull, @Nullable, @MagicConstant)
* regarder trello avant de coder
* *(d’autres à venir)*

1. Charte Graphique (en cours de rédaction)

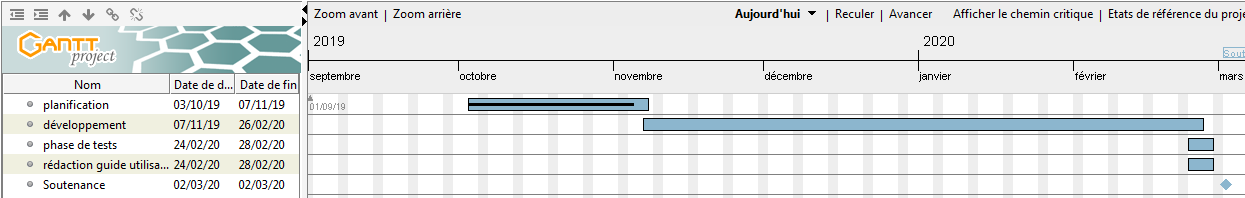
Police d’écriture ?

Choix de l’interface :

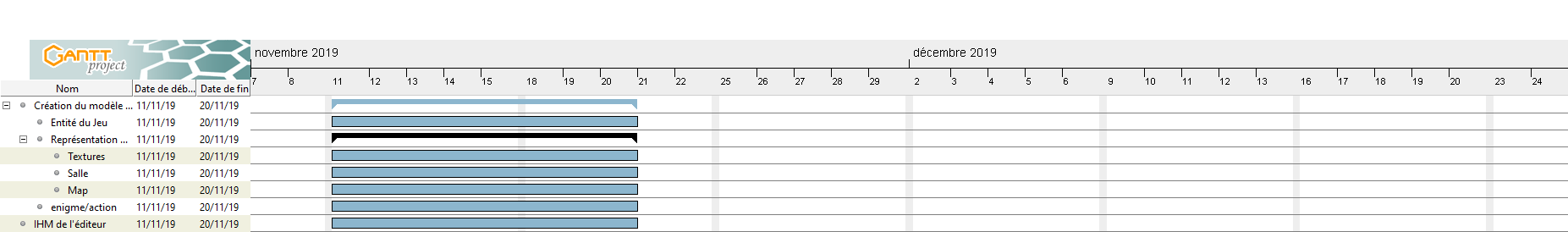
* Buttons ? (survol, clic)
* ….

1. Planning

Le rythme des sprints a été déterminé : 2 semaines.



Voici le diagramme de GANTT du projet entier, on va créer un deuxième diagramme de GANTT pour la partie développement, complété au fur et à mesure de l’avancement.



1. Annexes
   1. **BILAN DES REUNIONS**

**Réunion du jeudi 17 octobre 2019 (de 13h40 à 17h40) à Fontainebleau**

L’équipe traite la partie création d’escape game, une liste des taches a été écrite.

La bibliothèque sera composée de :

* Personnages : 3 personnages non-joueurs et 2 monstres
* Pièces : 4 salles
* Objets : un livre, une clef
* Meubles : Armoire/bibliothèque, un panneau, un interrupteur et une porte

Lors de la réunion, un DCU de la partie création a été conçu. Il a été décidé de la mise en œuvre du projet : nous réaliserons à 3, les fonctionnalités « must » de la création, puis laisserons Loic faire les « Should Could Would » pendant que nous nous occuperons de la partie jeu.

Un scénario a été réalisé et la décision à l’unanimité a été prise de ne pas faire de Diagramme séquence.

La réunion du jeudi 24 octobre est annulée.

1. Questions traités

Question : Présenter une salle puis une autre, … ou toute la carte d’un coup  
Décision : Présentation de toute la carte

Question : Doit-on afficher des salles préconçues (avec leurs murs) et les « coller les unes aux autres » ou ajouter la salle (sans les murs), le logiciel ajoute automatiquement les murs et si on met une salle à côté : le mur qui les sépare disparaît.  
Décision : Les salles sont préconçues et l’utilisateur peut seulement supprimer un mur, s’il est entre deux salles. (Donc toute la carte est entourée de murs).

Question : Comment le joueur ajoute des actions ? (catégorie actions ou bouton +/paramètres) ?  
Décision : Il peut ajouter des actions depuis catégorie (actions prédéfinies tel que « dialogue ») et avec un bouton +/paramètre sur chaque entité.   
Le bouton +/paramètres permet de consulter les actions mises, en créer, les modifier et les supprimer. (on ne peut créer que les actions faisables sur l’entité).

1. Fonctionnalités & propositions

On peut éditer un escape game.

Clic sur une entité, on voit les anciennes actions ajoutées, on peut les modifier/supprimer.

Lors de l’ajout d’une salle, on demande si elle doit être cachée.

Une carte est un dossier avec un sous-dossier/fichier par niveau.

Une bibliothèque contient un seul livre ou plusieurs ?

**Réunion du lundi 04 novembre 2019 (de 08h40 à 11h15) à Fontainebleau**

L’équipe s’est chargée de la rédaction du cahier des charges et de l’élaboration du diagramme de classes du cœur du jeu. Le **rythme de sprint** a été décidé : **2 semaines**.

**Un rendez-vous avec le tuteur est pris le mardi 12 novembre 2019 à 17h45**.

Les objectifs du rendez-vous sont :

* lui présenter le cahier des charges, avoir un retour/conseils
* discuter de l’avancement, du rythme de sprint choisi

Il est prévu pour la prochaine réunion de traiter le sprint planning.

1. Questions traitées

Question : Comment est la map ? (grille ou plutôt un espace)  
Décision : C’est une grille, composés de cases et des salles prennent un certain nombre de ses cases

Question : Comment on détermine qui affiche un mur (ex : entre deux salles)  
Décision : On a choisi qu’un mur contient la salle qui va le dessiner, plutôt qu’un booléen que l’on devrait changer 60fois par seconde (60 mises à jour de l’affichage par seconde).

Question : Comment on représente une pièce ?  
Décision : C’est 2 parties, on dessine d’abord les cases (sol, objets, …), puis on dessine les murs.

Question : Profondeur ?  
Décision : Une case contient des niveaux, on va afficher par exemple, au niveau 0, le sol, au niveau 1, un tapis, au niveau 2, un joueur… Les niveaux sont rendus de 0 à nombre de niveaux.

Question : Contenu des cases ?  
Décision : On peut placer dans une case, une entité, dont on appellera sa méthode getTexture pour la dessiner.

Question : Comment se répartir les tâches ?  
Décision : …  
Décision le 08/11/19 : on met les tâches dans le sprint planning et chacun en prends une et la fait. Il faut le consulter avant de coder.

Question : Nombre de travail à faire par semaine/horaires ?  
Décision : …

1. Fonctionnalités & propositions

Utiliser les annotations (@Nullable, @NotNull, @MagicConstant…)

Patron de conception proxy pour éviter de charger inutilement les textures ou de les charger plusieurs fois (voir patron de conception wikilivres)

Utiliser un tableau Excel/site web pour le sprint backlog.

La personne qui écrit le code n’est pas celle qui le documente. Cela permettrait une meilleure maitrise du code car on devrait apprendre le code pour bien le documenter.

La méthode getTexture prend un enum plutôt que les coordonnées de la sous-texture.

1. Remerciements et crédits

Merci à Luc Hernandez pour avoir accepté d’être notre tuteur, ainsi qu’à toute l’équipe étudiante, pour leur suivit durant ces deux années.

Remerciement spécial à Florent MADELAINE, pour nous avoir conseillé sur la structure de notre jeu, et avoir conçu un TD en relation avec notre sujet.

Les ressources ont été, majoritairement, récupérés sur itch.io. Merci à tous les auteurs de les partager !

Ressources utiles :

* [itch.io](https://itch.io/game-assets) pour tous les graphismes
* [howtomakeanrpg](https://howtomakeanrpg.com/) (livre) pour ses explications sur la structure et de développement des jeux vidéo
* [Java Game Développement with LIBGDX](https://www.apress.com/gp/book/9781484233238) (livre) pour ses explications sur l’utilisation de la LibGdx.
* Discord : <https://discord.gg/zktyRaK> application de communication
* [LibGdx wiki](https://github.com/libgdx/libgdx/wiki): Le wiki official de la libgdx
* GitHub : <https://github.com/LoukaDOZ/Enigma>, plateforme GIT
* Trello : <https://trello.com/b/pTGVfrp6/enigma>, plateforme pour le sprint board
* [Tiled](https://www.mapeditor.org/) : créer les maps
* [Texture Packer](https://www.codeandweb.com/texturepacker): Plusieurs sous-textures dans un même fichier
* [Ezgif](https://ezgif.com/sprite-cutter) : Séparer une texture