

Document

INFO-F307 - Génie logiciel et gestion de projets

Introduction

Ce document détaille tous les choix de conception et d'utilisation d'outils pour mener à bien le projet demandé par le client. Lors de cette première itération, 5 histoires ont été implémentées comme prévu lors de la réunion avec le client. Il y est aussi indiqué les quelques designs patterns utilisés pour coder efficacement en orienté objet. Pour finir, les difficultés rencontrées sont également mentionnées.

Développement

Histoire 1

Toutes les fonctionnalités des menus de connexion, d'enregistrement et de changement de mot de passe ont été implémentés comme convenu avec le client. C'est aussi dans cette histoire que nous avons décidé de configurer une base de données en utilisant sqllite. La base de données en elle-même est représentée par un modèle.

Difficultés rencontrées

- Installation et configuration de sqllite sur IntelliJ. Difficulté résolue en y accordant assez de recherches.
- Apprentissage de JavaFx et mise en place sous forme de MVC.
- Nous avons résolu cela en faisant passer le Main a chaque controller.

Choix de conception

- Un pattern MVC implémenté dès le Main. Utilisation de SceneBuilder afin de faciliter la création des menus et de les rendre plus esthétiques.

Histoire 2

L'implémentation de la modification, suppression et création de paquet dans la database par la GUI de création de paquets avec redirection vers les menus de création de carte/modification et le mode étude.

Choix de conception

- Nous avons utilisé un MVC hérité de notre classe abstraite (dont tous les menus héritent) pour pouvoir l'intégrer dans les menu
- Modification d'un paquet à la fois par listes de cartes et tags

Histoire 3

L'implémentation des menus des modifications, suppressions et créations de cartes ainsi que les nouvelles interactions avec la database et avec la GUI des paquets ont été faites.

Choix de conception

- Utilisation du design pattern implémenté dans l'histoire 1 pour faire en sorte que le "pointeur" appartenant au MVC main pointe vers les bon menus.
- Création d'une table DeckCard dans la base de données permettant de retrouver quel paquet appartient à quelle carte.

Histoire 4

Le mode d'entraînement a été réalisé. Il permet de s'auto-évaluer sur un jeu de cartes sélectionné. Il utilise l'algorithme de sélection de cartes implémenté pour l'histoire 5.

Choix de conceptions

- Nous avons choisi d'enregistrer les scores des cartes au fur et à mesure de l'auto-évaluation de l'utilisateur afin d'éviter la perte de progression en cas de problème du système.

Histoire 5

L'algorithme de sélection des cartes a été réalisé.

Choix de conception

- La classe qui s'occupe de la sélection est instanciée au début de la partie avec le paquet de cartes.
- Un cooldown a été mis en place pour éviter qu'une même carte soit piochée plusieurs fois de suite.
- Les cartes qui n'ont pas été montrées depuis très longtemps, même si leur score est bon, peuvent être piochées.

Pour conclure l'itération 1, nous avons implémenté toutes les fonctionnalités des 5 histoires.