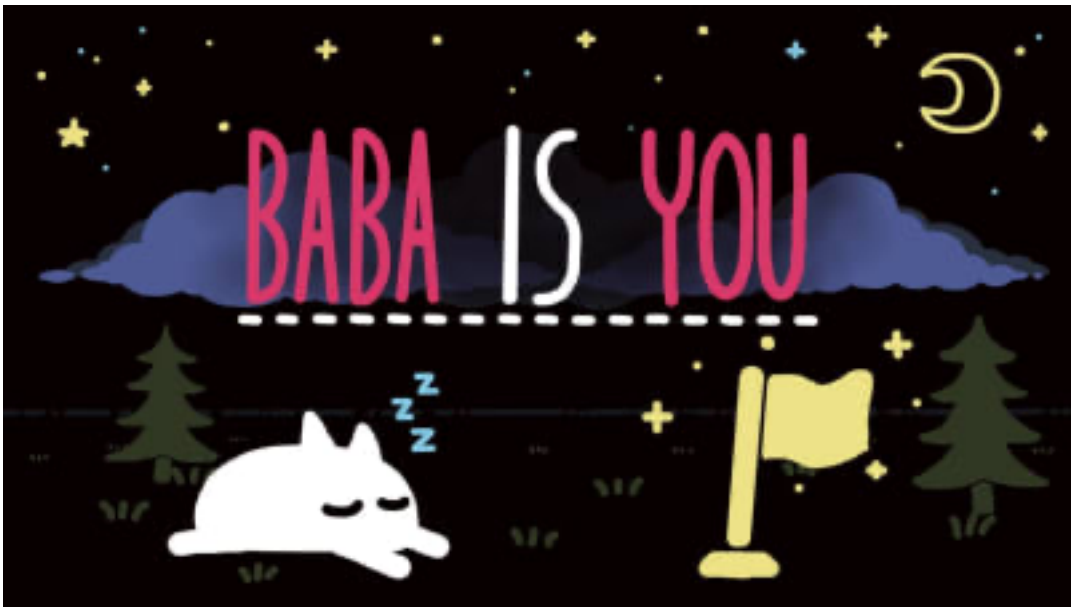

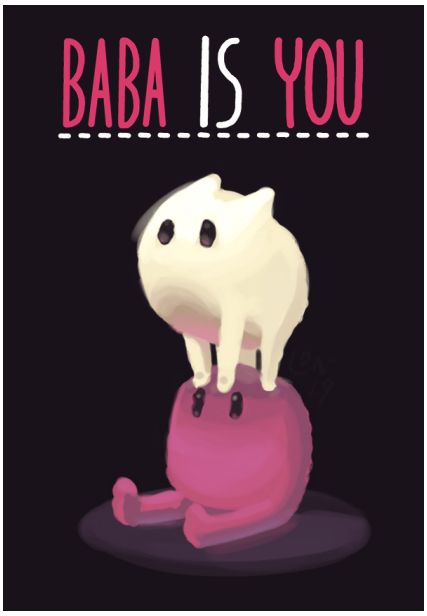

BabaIsYou

C++

58112 |BENMASSAOUD El Mamoune
58132 |EL HARROUTI IMAD

GBA| DEV4 | 21/04/2023

Sommaire :

Introduction :

Le projet Baba Is You est un jeu vidéo de type puzzle créé par Hempuli Oy. Il a été développé en C++ avec la bibliothèque graphique SFML. Le but de ce projet était de recréer ce jeu en utilisant une architecture MVC avec un design pattern Observateur/Observé, ainsi que les fonctionnalités d'Undo/Redo pour permettre au joueur de revenir en arrière ou de répéter une action.

Baba Is You est un jeu de réflexion conçu pour exercer la logique et la créativité du joueur. Le but de ce rapport est de présenter l'architecture du code ainsi que les principales fonctionnalités du jeu.

Architecture MVC

L'architecture MVC (Modèle-Vue-Contrôleur) est un modèle de conception qui permet de séparer la logique de présentation de la logique métier. Dans ce projet, la classe `BabaIsYou` représente le modèle qui contient les éléments du jeu. La classe BabaIsYou`View` est responsable de l'affichage console, tandis que la classe `BabaIsYouController` est le contrôleur qui gère l'interaction entre le joueur et le modèle.

Design pattern Observateur/Observé

Le design pattern Observateur/Observé est un modèle de conception qui permet à un objet (l'Observé) de notifier automatiquement ses dépendances (les Observateurs) lorsqu'il subit un changement. Dans ce projet, la classe `BabaIsYou` est l'Observé qui notifie la vue lorsqu'un changement survient dans le modèle.

Les classes du projet

Board : pour représenter le plateau de jeu.
Tiles : représente une case du plateau de jeu
Element : pour représenter un élément sur le plateau.
Material : pour représenter le matériau d'un élément.
Word : pour représenter un mot dans une règle.
Subject : pour représenter le sujet d'une règle.
Operator : pour représenter l'opérateur d'une règle.
Complement : pour représenter le complément d'une règle.
Direction : pour représenter la direction de déplacement d'un élément.
Position : pour représenter la position d'un élément sur le plateau.
LevelLoader : pour charger un niveau à partir d'un fichier.
Rule : pour représenter une règle.
RuleManager : pour gérer les règles du jeu.
BabaIsYou : pour représenter le jeu lui-même.

La classe Board

La classe `Board` est une classe qui fait partie du modèle et qui représente le plateau de jeu. Elle contient des objets de la classe Tiles, qui représentent chacun une case du plateau. La classe Board fournit des méthodes pour accéder et modifier les éléments du plateau de jeu.

La classe BabaIsYou

La classe `BabaIsYou` est le modèle dans notre implémentation du jeu Baba Is You. Cette classe représente l'état du jeu, c'est-à-dire l'emplacement des éléments, les règles et les conditions actuelles. Elle contient également des méthodes pour accéder et modifier l'état du jeu, comme la méthode move() pour déplacer le joueur. Elle utilise également la classe `CommandManager` pour implémenter les fonctionnalités Undo/Redo et la classe `Observer` pour implémenter le design pattern Observateur/Observé.

Les classes complémentaires

Les classes complémentaires comprennent la classe `LevelLoader`, qui est responsable du chargement des niveaux à partir de fichiers texte, la classe, qui représente une position sur le plateau de jeu, la classe `Direction`, qui représente une direction de déplacement, la classe `Operator`, qui représente un opérateur dans une règle, la classe, qui représente un complément dans une règle, et la classe `RuleManager`, qui est responsable de trouver et d'ajouter des règles dans le modèle.

Fonctionnalités Undo/Redo

Les fonctionnalités Undo/Redo permettent au joueur de revenir en arrière ou de répéter une action. Dans ce projet, la classe est responsable de stocker l'historique des commandes effectuées par le joueur, ainsi que les états correspondants du modèle. La classe `CommandManager` est responsable d'annuler une commande, tandis que la classe `UndoCommand` est responsable de la répéter.

Conclusion

L'implémentation du projet Baba Is You a été un succès, avec toutes les fonctionnalités requises implémentées. Le modèle MVC a été bien appliqué, avec une séparation claire des responsabilités entre le modèle, la vue et le contrôleur. L'utilisation de l'observateur/observé a permis une communication efficace entre les différentes parties du jeu. L'ajout d'undo/redo a été une fonctionnalité supplémentaire importante qui a permis aux joueurs de revenir en arrière et de refaire des mouvements en cas d'erreur. En outre, la gestion des règles et des transformations a permis une grande flexibilité dans le gameplay, donnant aux joueurs la possibilité de changer les règles du jeu pour atteindre les objectifs. Dans l'ensemble, l'implémentation du projet a été réussie et a abouti à un jeu amusant et engageant.

En conclusion, le projet Baba Is You a été un défi intéressant et nous a permis de mettre en pratique plusieurs concepts de programmation avancée, tels que l'architecture MVC, le design pattern Observateur/Observé