

1 Introduction

L'objectif est de créer une interface graphique pour un jeu de votre choix (validé par votre professeur), tout en suivant le schéma de conception *Modèle-Vue-Contrôleur*. Pour rappel, un bon moyen d'éprouver ce schéma est de réaliser plusieurs Vues différentes (texte et graphique, par exemple) sans modifier le Modèle.

Plus précisément, ce projet composé de deux applications. On prend comme exemple l'extension de votre projet du Semestre 5, à savoir un jeu d'exploration. Ces applications seront :

1. un éditeur de niveaux pour placer les différents éléments du jeu ;
2. l'application finale s'appuyant sur cet éditeur pour jouer seul ou à plusieurs.

1.1 Gestion de projet

Ce travail sera réalisé par groupes de 3 ou 4 étudiants. Vous devrez exploiter les compétences acquises au cours de l'année en gestion de projet. Le code devra reposer sur un système de versionnement (GIT par exemple). Il vous appartient de gérer le planning de suivi avec le client (Xavier Skapin). Toute modification envisagée par rapport au cahier des charges doit être validée par ce client.

1.2 Livrables

Chaque sous-projet donnera lieu à divers livrables. Certains livrables, listés ci-dessous, seront les mêmes pour chaque sous-projet. Des livrables supplémentaires peuvent être demandés selon les sous-projets.

Les livrables "communs" sont les suivants.

1. Un rapport de projet détaillant :
 - vos choix de conception, les difficultés rencontrées et leurs solutions ;
 - votre gestion de projet, en particulier le planning et **la répartition des tâches entre les membres du groupe**.
2. Un document technique décrivant la conception de votre architecture logicielle, pour permettre à de futurs développeurs (ou vous-mêmes) de reprendre ce travail plus tard. Toutes les compétences relatives à la POO (UML, MVC, *etc.*) sont à exploiter ;
3. Le code lui-même, stocké dans un dépôt de type `Git`. À partir de ce dépôt, il sera possible de récupérer votre projet et de le compiler. La compilation devra être indépendante d'un IDE (*i.e.* sera réalisable en ligne de commande). Le choix des outils de compilation (MAVEN, GRADLE, *etc.*) est laissé libre. Dans tous les cas, un `README.txt` décrivant les étapes à suivre pour la compilation et l'exécution ;
4. la documentation générée à partir des fichiers source ;
5. Un manuel utilisateur pour exécuter votre logiciel.

1.3 Suivi de projet

- des séances sur machines seront consacrées au suivi de projet ;
- des réunions supplémentaires avec le client pourront être organisées ;
- **2 mai 2025** : remise finale de l'ensemble du projet avec tous les documents.

Des séances de suivis de projet auront lieu au cours du semestre et des réunions avec le client pourront être mises en place à la demande.

1.4 Notation

La notation reposera sur les critères suivants :

- les réunions de suivi ;
- les livrables de chaque sous-projet.

2 Éditeur de niveaux

On conserve comme exemple le jeu d'exploration. Chaque niveau correspond à une zone à explorer.

2.1 Livrables spécifiques

L'interface contiendra un menu avec **au moins** les fonctionnalités suivantes :

- la création d'un nouveau niveau avec des dimensions (hauteur et largeur) sélectionnées par l'utilisateur ;
- la sauvegarde d'un niveau. Le format des fichiers de sauvegarde est laissé à votre appréciation ; on utilisera de préférence un format lisible pour l'être humain, JSON par exemple ;
- le chargement d'un niveau sauvegardé.

Un niveau peut être considéré comme une grille vide par défaut. Il s'agit ensuite de remplir cette grille avec différents items. À vous de déterminer leur type : objet, personnage amical, ennemi, etc.

3 Jeu version 2D

Il s'agit maintenant de créer la version du jeu s'appuyant sur l'éditeur de niveaux précédent.

3.1 Configuration générale

Le jeu commence par le réglage des paramètres

Ces paramètres déterminent notamment :

- le mode de jeu : **Single player** pour un seul joueur ou **Multiplayer** pour 2 à 4 joueurs ;
- la configuration (contrôle des joueurs, résolution de l'écran, langue...) ;
- la liste des meilleurs scores ;
- les auteurs du jeu ;
- un bouton pour quitter le jeu.

3.2 Livrables spécifiques pour le menu de configuration générale

Mode mono-joueur Créer des profils de joueurs (par exemple : aventurier, assassin, sniper, voleur...) pour déterminer des caractéristiques différentes.

Mode multi-joueur (optionnel) Le mode **Local** correspond au jeu sur un seul ordinateur. Vous pouvez également créer une version multi-joueur en LAN local si vous le souhaitez.

Configuration Il faut créer un menu permettant de préciser :

- pour chaque joueur, les touches du clavier associées ;
- la résolution de l'écran (en proposer aux moins deux) ;
- la langue utilisée (en proposer aux moins deux).

Hall of Fame Créer une fenêtre pour afficher les meilleurs scores réalisés.

3.3 Déroulement du jeu

Après avoir sélectionné un ou plusieurs joueurs, la partie peut démarrer.

Afficher dans l'interface les caractéristiques du joueur en fonction de son avancement : points de vie, armes, objets divers, *etc.*

Définir une touche du clavier pour mettre le jeu en pause (et accessoirement de couper/réactiver la musique de fond, ou de retourner au menu principal).

3.3.1 Fin du jeu

À vous de déterminer la fin du jeu. Cela peut correspondre à un niveau précis à atteindre, un objet à obtenir, ou encore des niveaux infinis et le jeu cesse quand le nombre de points de vie du joueur tombe à zéro.

À plusieurs joueurs, le jeu continue tant qu'il reste au moins un héros vivant. Si un autre joueur est mort, on peut le ressusciter en pressant la touche du clavier associée ; il retrouve alors sa position de fin de partie (ou une position aléatoire si la carte a changé) et le niveau et le nombre de points de vie de début de jeu.

À la fin de la partie, le menu "*Game over*" apparaît ainsi que le score des joueurs. Les meilleurs scores sont affichés dans le "*Hall of Fame*".