|  |
| --- |
| JOURNAL DE BORD |
| Xavier DEGRAEVE  Enzo DERO  **Max M’BEY**  **Felix Cloerec**  [Site web] |

Journal de bord de la réalisation et des tests du projet Mille Bornes

**Semaine du 25/10 :**

**Xavier :** J'ai fait le choix de commencer par créer les assets du jeu, pour pouvoir concevoir l'interface en sachant déjà à quoi elle ressemblerait. J'ai utilisé le logiciel gratuit piskel pour faire l'entièreté des visuels du jeu.

**Enzo :**

**Max :**

**Felix :**

**Semaine du 08/11 :**

**Xavier :** suite de la réalisation des visuels

**Enzo :**

**Max :**

**Felix :**

**Semaine du 15/11 :**

**Xavier :** fin de la réalisation des visuels

**Enzo :**

**Max :**

**Felix :**

**Semaine du 22/11 :**

**Xavier :** Une fois le circuit et les voitures dessinés et animés j'ai commencé par travailler sur la correction de certains bugs d'affichages.

J'ai géré le cas où on lance le jeu pour la première fois et lorsque la sauvegarde n'existe pas.

J'ai ensuite implémenté le système de tour dans le modèle et utilisé la fenêtre de log implémenté par Enzo.

Dans un premier temps, j'ai implémenté le système de tour dans une boucle while qui appelait les joueurs tant qu'il n'y avait pas de gagnants.

Cette implémentation m'a posé des problèmes, notamment de performances et entrainait beaucoup de problèmes avec le tour de l'utilisateur.

j'ai donc opté pour un simple système de boolean qui autorisé le joueur à jouer (à cliquer sur les boutons), et lorsque son tour prends fin, son boolean est mit à false et la méthode d'action des bots est appelée. La méthode d'action des bots appelle une méthode qui met le boolean du joueur à true.

Il y avait un problème concernant la sauvegarde qui prenait de plus en plus de place au fur et à mesure des parties (alors que la sauvegarde de partie n'enregistre que la dernière partie si elle a été laissée en suspens). Le problème venait du fait que la liste de joueurs n’était pas réinitialisée

Il y avait aussi un problème qui venait de la manière dont on initialisé les boutons. On ajoutait les actionListeners en boucle sans jamais les enlever, il y avait donc des soucis quand on cliquait sur les boutons, car les méthodes étaient appelés plusieurs fois au lieu d'une. J'ai réglé le souci en forçant les boutons à n'avoir qu'un seul ActionListener (supprimer tous les listeners avant de les rajouter lors de l'initialisation)

**Enzo :**

**Max :**

**Felix :**

**Semaine du 29/11 :**

**Xavier :** J'ai ensuite commencé à travailler sur les interactions entre le joueur et l'interface graphique, notamment lorsque le joueur joue une carte. Le contrôleur est devenu un peu complexe, j'aurais pu mieux gérer la distinction entre contrôleur et modèle pour faciliter la compréhension du code.

Lorsque les éléments les plus importants ont été implémenté sur le joueur, j'ai implémenté le système de jeu des CPU, basé sur un arbre de décision en utilisant les méthodes faites par Felix vérification() et jouerCarte() en les adaptant en fonction du type de joueur. L'architecture MVC a facilité le travail. A la fin, les bots pouvaient piocher, défausser, finir leur tour, et jouer en fonction de leur personnalité. Les attaques sont lancées sur le joueur le plus avancé (si l'attaque est possible). En cas de joueur rendu à la même borne, la cible est choisie au hasard.

Il fallait ensuite permettre au joueur de suivre la partie, j'ai donc implémenté un système complet qui informe des coups joués par le joueur et les bots. Le système de log permet aussi de savoir pourquoi une action du joueur n'est pas possible.

**Enzo :**

**Max :**

**Felix :**

**Semaine du 06/12 :**

**Xavier :** J'ai implémenté le système de partie, le joueur qui commence à jouer est choisi au hasard. J'ai aussi commencé à implémenter le système de sauvegarde des parties. J'ai fait en sorte que certains logs importants soient stockés dans nouveau un fichier .txt à chaque partie. Lorsqu'on reprend une partie, le fichier .txt associé à la manche que l'on reprend est chargé dans la console de log.

**Enzo :**

**Max :**

**Felix :**

**Semaine du 20/12 :**

**Xavier :** Le jeu était globalement opérationnel, j'ai donc commencé à ajouter des éléments visuels secondaires (des cartes personnalisées en pixel art, des couleurs différentes pour chaque voiture, une interface entièrement en pixel art pour plus de cohérence visuelle, des meilleures animations de voitures avec des ombres). J'ai pris soin que l'interface s'adapte à la taille à l'écran sur lequel on affiche le jeu.

J'ai continué l'implémentation du système de manche et de partie qu'avait commencé Felix. Il faut finir la manche à pile 700 bornes pour gagner une manche. Lorsqu'un joueur gagne le jeu s'arrête quelques secondes en annonçant le gagnant, et lance ensuite une autre manche en annonçant les points des joueurs. Si la partie est gagnée, le jeu annonce le gagnant de la partie et relance une partie. Les fichiers de sauvegardes de manche sont tous mit dans un nouveau fichier .txt.

J'ai ensuite résolu des bugs mineurs en faisant de nombreuses de parties tests

**Enzo :**

**Max :**

**Felix :**