

Création du jeu Wall is You en Python

Table des matières :

Introduction.....2

Partie 1: Organisation du Projet.....3

 1. Répartition des tâches dans le groupe.....3

 2. Pourcentage de répartition du travail.....4

 3. Modalités d’organisation.....4

 4. Difficultés rencontrées.....4

Partie 2 : Guide d’utilisation du programme.....4

 1. Lancement.....4

 2.Utilisation des menus.....5

 3. Utilisation du jeu.....5

Partie 3 : Avancement du projet.....5

 1. État d’avancement du projet.....5

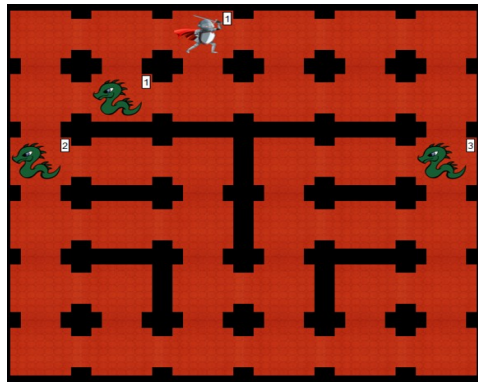
 2. Améliorations éventuelles.....6

 3. Informations sur le code.....6

Conclusion.....6

Introduction :

Le jeu Wall is You est un « remake » classique d'un jeu où le joueur incarne un chevalier qui doit nettoyer un donjon infesté de monstres. Cependant dans ce jeu, les rôles sont inversés. Le joueur joue ici le rôle d'un donjon bienveillant qui aide le chevalier à éradiquer les créatures sinistres du donjon avec pour but le nettoyage du donjon. C'est un jeu d'adresse se jouant à 1 joueur qui a la possibilité de pivoter les salles dans le sens horaire afin de permettre la création de l'intention qui est un chemin tracé par le code du jeu qui dirige le personnage vers le monstre le plus fort. Mais pour gagner la partie, le joueur doit faire pivoter les salles du donjon de sorte que le chemin tracé soit vers le monstre le moins fort avec un niveau équivalent au chevalier. Alors le monstre est tué et le personnage gagne un niveau sinon il meurt et la partie est perdue.



Partie en cours (Map test)

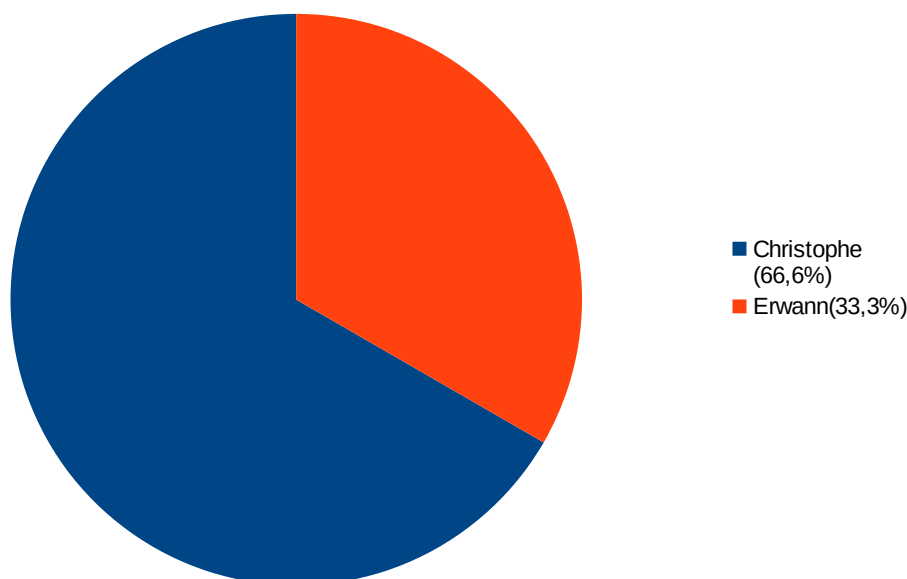
L'objectif est alors d'implémenter une version en python (langage de programmation) de ce jeu en créant sa structure de base (programme, variables, fonctions...), l'interface du jeu (présentation, ergonomie, esthétique, affichage...) et son paramétrage (donjons, donjons variants...). Ici ce rapport exposera dans une 1ère partie l'organisation du projet, ensuite dans une seconde un petit guide du jeu et d'utilisation du programme et enfin l'avancement du projet.

Partie 1 : Organisation du projet

1. Répartition des tâches dans le groupe

Afin de bien entamer le projet, le travail à été divisé en 3 parties : Création des donjons et affichage personnages, interface graphique du jeu (présentation, fenêtres graphiques) et logique du jeu (intention...). Erwann Thomas s'est occupé de l'intention de l'aventurier (Chevalier) et des commentaires du programme puis Christophe Taratibu de la création des donjons, affichage des personnages, logique du jeu (rencontre chevalier-monstres, application du chemin du chevalier, clic de souris pour le pivotement des salles du donjons et le menu du jeu).

2. Pourcentage de la répartition du travail



3. Modalités d'organisation, fréquences de rencontres

La majorité du travail a été fait en cours et ces derniers ont servi à mettre en commun le travail fait à la maison. Les rencontres ont été régulières et des sessions sur Discord ont été menées pour structurer la suite du projet.

4. Difficultés rencontrées

Les problèmes encourus lors du codage du jeu ont été moyennes. Des soucis au niveaux de la fermeture de la fenêtre graphique fltk dues à l'ouverture à l'extérieur de la fonction et fermeture à l'intérieure de cette dernière et des problèmes du codage de l'attribution des tags des figures et images lors de la création graphique du donjon. L'implémentation du donjon graphique a été compliquée pour ce qui est du pivotement de salles et de leur connexion (permet de savoir si les salles sont ouvertes ou fermées). Aussi, les cas d'arrêt lors de la victoire où de la défaite du joueur ne sont pas gérés (après victoire ou défaite du joueur le jeu continue de tourner si on clique sur espace ou sur le clic gauche de la souris).

Partie 2 : Guide d'utilisation du programme

1. Lancement

Le jeu se lance grâce à une fonction et commence par demander au joueur s'il veut jouer. Ensuite il peut choisir 5 différents donjons et celui qui est choisi s'affiche à la suite grâce à une fonction d'ouverture de fichier qui découpe les données nécessaires à la création du donjon. Par la suite la fonction `init_dungeon` initialise le donjon et le joueur peut débiter.

2. Utilisation des menus

Comme expliqué dans le lancement le menu se divise en 2. Cela a été fait de sorte qu'une personne puisse interagir avec la fenêtre graphique fltk, La première partie, le joueur peut choisir en appuyant avec la souris sur les boutons jouer qui active la session ou quitter qui quitte la session. Dans la seconde partie, le joueur choisit entre 5 images (qui sont des boutons) de donjons.

3. Utilisation du jeu

Afin de pouvoir de pouvoir exécuter correctement une partie de Wall is You, le joueur fait pivoter les salles en utilisant le clic gauche de la souris avec pour but de terrasser les monstres du donjon. Lorsque un chemin est crée, le joueur à le choix de terminer son tour en appuyant sur la touche espace permettant ainsi au chevalier de bouger suivant ce chemin, ou de bloquer ce chemin si le monstre a un niveau supérieur au personnage.

Partie 3 : Avancement du projet

1. État d'avancement du projet

Le projet d'informatique est complété. Les tâches obligatoires ont été faites avec l'affichage du donjon et personnages, de l'intention du chevalier avec son déplacement, leurs déplacement et l'affichage de l'interface graphique pour bien présenter le jeu.

2 . Améliorations éventuelles

Afin de rendre le jeu plus beau, des fonds d'écran ont été créés pour le menu pour le rendre plus coloré.

3. Informations sur le code :

Le code du jeu du Wall is You est divisé en 4 fichiers wallisyou.py, game_menu.py, graphic_game.py, et logic_game.py. Le premier regroupe les fonctions qui permettent de lancer correctement le jeu et gère aussi la phase de jeu. Le second contient une classe bouton qui permet de les créer , de les cliquer dessus en utilisant les évènements de souris de fltk et des fonctions qui permettent d'agencer les menus lors de l'exécution du code. Le troisième quant à lui contient des fonctions permettant de gérer la partie graphique du jeu comme l'affichage des salles et des personnages. Le dernier des fonctions qui gèrent la logique interne du jeu et qui gère les conditions d'arrêt du jeu. Si les monstres sont éradiqués la partie est gagnée, si le chevalier meurt la partie est finie ou sinon la partie continue.

Conclusion :

Ce projet était une chose très intéressante à travailler . Il a permis de mettre en valeur nos talents dans tout ce qui est relatif au codage, de la logique du jeu, de la présentation du plateau et des graphismes du jeu. Le gain de liberté offerte par ce travail était très passionnant . Il a octroyé plus de possibilités dans l'organisation de l'équipe afin de répondre au mieux aux attentes.