



> PRÉSENTATION GÉNÉRALE

VISION GÉNÉRALE

Notre première idée pour le pendu était de faire d'une façon que le projet analyse les lettres en fonction du mot aléatoire de la liste choisi (idée soutenue jusqu'à la fin du projet), et en échange rapporter un dessin du pendu progressif tout au long du jeu. A la moitié du projet, l'idée de créer un pendu écrit dans la console nous était plus facile, pourtant cela ne nous a pas satisfait: pour faire un dessin plus visible et bien encadré, on a décidé de changer les dessins de la console pour devenir des commandes turtle. A la fin du projet on a trouvé intéressant de créer des dialogues plus amicaux entre le joueur et le jeu, et d'ajouter des détails au turtle.

> ORGANISATION DU TRAVAIL

EQUIPE

- Pedro SANTOANTONIO
- Luca REINS
- Daniel SALIENT

RÔLES

Daniel s'est occupé principalement de turtle et de la révision du programme:

- La démarche et la mise dans le programme des dessins;
- Après la conclusion du programme, la révision ligne par ligne de celui-ci.

Pedro et Luca se sont occupés principalement du programme écrit et de l'organisation:

- Variables, construction du programme, forme;
- Documents, partage, programme... .

COMMUNICATION

Au long des deux semaines de vacances, on s'est communiqué par Discord. Dans les occasions où on était ensemble visant le programme, les trois intégrants du groupe partageaient son écran et on discutait les différentes alternatives d'idées. On a fait le projet dans sa plupart au début de la première semaine, et on a conclu le projet au sein de la seconde semaine. On s'est partagé les rôles, mais on était toujours ensemble pour écrire le programme, s'entraider et trouver les solutions ensemble. Quand chacun avait fini sa partie, on a envoyé à Daniel comme la plus grande partie était le dessin turtle, mais aussi pour après mettre les phrases types sur turtle et comme Daniel était déjà spécialisé sur cela, on l'a fait ensemble mais seulement sur l'ordinateur de Daniel.

> LES ÉTAPES DU PROJET

PROJET INITIAL ET ÉVOLUTION DU PROJET

Notre première idée pour le pendu était de faire d'une façon que le projet analyse les lettres en fonction du mot aléatoire de la liste choisi (idée soutenue jusqu'à la fin du projet), et en échange rapporter un dessin du pendu progressif tout au long du jeu. A la moitié du projet, l'idée de créer un pendu écrit dans la console nous était plus facile, pourtant cela ne nous a pas satisfait: pour faire un dessin plus visible et bien encadré, on a décidé de changer les dessins de la console pour devenir des commandes turtle. A la fin du projet on a trouvé intéressant de créer des dialogues plus amicaux entre le joueur et le jeu, et d'ajouter des détails au turtle.

ADAPTATION

Depuis le début du programme on a bien sûr fait quelques adaptations: notre première réalisation consistait sur un dessin, avec toutes nos phrases type (comme par exemple: "Oui, la lettre {proposition} se trouve dans ce mot") qui s'écrivait et dessinait dans la console. Cela ne nous satisfaisait pas, donc on a tout transféré sur turtle... Celle-ci fut notre principale adaptation. Pourtant au cours du programme

> FONCTIONNEMENT ET OPÉRATIONNALITÉ

PROBLÈMES ET SOLUTIONS

- 1 Premièrement, on a eu quelques difficultés pour commencer le projet, on ne savait pas où commencer à part la liste de mot.

Solution - Après avoir décidé plutôt rapidement qu'on commencerait par une liste de mot qu'on tirerait au sort, on a alors vu que les conditions (c'est-à-dire les "if" et "else") seraient les plus pratiques pour réaliser le projet.

- 2 Un de nos plus grands problèmes est que l'on avait besoin d'une fonction qui inclut quelque chose dans une liste ou quelque chose semblable, pourtant on ne savait pas comment le faire.

Solution - Pour ce problème, on s'est renseigné sur internet, où on a trouvé la fonction "if ... in ... :" ce qui nous a beaucoup aidé pour inclure à chaque fois ce dont on avait besoin. Mais aussi grâce à Daniel qui connaissait déjà la fonction "if ... not in ... :" mais ne savait pas si l'inverse marchait.

- 3 On a essayé de trouver une manière de faire en sorte que le programme reconnaisse quand on saisit plusieurs lettres, pour qu'il soit plus complet. Mais cela a été compliqué.

Solution - On a dû réviser toute la forme du programme pour inclure cette condition, comme on l'a placée au début du programme.

- 4 Le programme nous a fourni plusieurs petits problèmes lors du procès du jeu et les conditions if et else.

Solution - On a réussi à corriger les problèmes avec l'aide de python tutor.

- 5 L' output du programme, au début, ne se ressemblait pas exactement à ce qu'on imaginait comme on n'avait pas utilisé turtle, mais plutôt la propre console pour faire le dessin: on a en premier créé le dessin sur python, dans le programme (avec des "-", "/" et "0", pour former le dessin du pendu), mais comme on n'était pas satisfait du résultat on a voulu le faire sur turtle, ce qui nous a posé quelques problèmes: comme par exemple on a eu du mal à l'inclure dans le programme avec les conditions.

Solution - On a alors décidé de continuer avec turtle, même si on avait déjà la forme (comme on l'avait déjà fait mais avec des "-", "/" et "0"), on a alors tout supprimé et on a mis dans le "else" qui se trouvait le dessin, les coordonnées de turtle pour dessiner le pendu. Donc à chaque fois qu'il y a une lettre erronée, le pendu se complète au fur et à mesure, jusqu'à ce que les tentatives se soient écoulées, ou jusqu'à ce qu'on trouve le mot choisi.

- 6 Faire le dessin turtle: les distances nous causaient quelques problèmes, comme calculer le degré d'un angle pour savoir la distance d'un trait. Le pinceau passait par des endroits difficiles à gérer.

Solution - Concernant les distances on a testé jusqu'à trouver les distances qui étaient bien pour chaque rectangle ou cercle et surtout savoir tout le temps les coordonnées de où on voulait placer notre pinceau. On a fait plusieurs recherches sur internet sur plusieurs fonctionnalités de turtle pour faire un bon dessin.

TESTS

On a fait des tests sans arrêt pour le programme. À chaque fois qu'on finissait une toute petite partie, on testait pour voir si ça marchait. Notre plus grand exemple est le dessin sur turtle: pour chaque coordonné on testait pour voir si le dessin était bien celui qu'on imaginait, ou si le pinceau s'arrêtait au bon endroit, et même pour voir le dessin en soit, c'est-à-dire un bonhomme plus grand, des bras plus longs, des jambes plus grosses... Jusqu'à ce qu'on trouve la taille parfaite.

> OUVERTURE

PERSPECTIVES D'EVOLUTION

On a pensé peut-être de faire un fond coloré ou plus beau derrière le pendu, quelque chose de plus animé. On voulait créer une fonctionnalité qui donnerait l'option d'utiliser un conseil en donnant une lettre au hasard du mot rechercher, pourtant on n'a pas réussi. On pense qu'on peut faire un programme plus court avec moi de ligne...