



NOM DU PROJET : Tic Tac Toe

> PRÉSENTATION GÉNÉRALE

VISION GÉNÉRALE

Notre objectif est de créer un jeu qui soit semblable aux règles du vrai Tic Tac Toe, avec des modes de jeu: joueur contre joueur et joueur contre machine. Dans ce dernier cas, on veut mettre des difficultés, ainsi que la facile (ou "aléatoire") et la difficile (ou "impossible"). L'esthétique nous est très importante aussi, alors qu'on essaye de le rendre le plus beau visuellement possible.

> ORGANISATION DU TRAVAIL

EQUIPE

- Guilherme ZELAZNY
- Pedro SANTOANTONIO
- Daniel SALLENT

RÔLES

Daniel s'est occupé du programme en mettant les idées du groupe dans des lignes de programmation. Il a été extrêmement important pour réaliser le programme avec tk.Tk, et de la révision du Compte Rendu:

- La démarche et la mise du programme dans le croisillon du jeu ;
- Tout le programme avec des révisions constantes et création de celui-ci;
- Révision du Compte Rendu.

Pedro s'est occupé de l'organisation, des éditions et corrections du programme original que Daniel le faisait en même temps en partition d'écran à Discord. Révision du programme après l'avoir fini en additionnant des commentaires, et du Compte Rendu :

- Organiser les horaires pour les mises en commun et pour ne pas entraîner le projet à se perdre à cause des manques de communication;
- Révisions constantes peu à peu que le programme évolue et des commentaires;
- Création du Compte Rendu.

Guilherme a lu le document écrit par Daniel et Pedro, et avec les idées de base il a beaucoup aidé au programme et au Compte Rendu. Il a été aussi important à la révision finale du projet.

COMMUNICATION

On s'est beaucoup parlé par un groupe WhatsApp. Tout au début du projet, la communication était très difficile. Pedro a dû s'arranger à cause d'un voyage marqué au dernier moment vers la France tout seul, et cela a beaucoup impacté le commencement du programme. Comme Pedro était censé organiser le groupe, cela a été un problème pendant quelques jours. Alors, pour se mettre en commun et pouvoir se communiquer autrement que par des messages, nous nous sommes arrangés avec des rendez-vous marqués par des horaires semanales à la plateforme Discord. Quelques fois aléatoirement, lorsqu'un intégrant du groupe en a eu une idée intéressante ou une évolution importante au sein du projet, on s'est réunis sur Discord aussi. Guilherme nous a beaucoup manqué dans la communication.

> LES ÉTAPES DU PROJET

PROJET INITIAL ET ÉVOLUTION DU PROJET

L'idée initiale n'était pas beaucoup plus différente de celle présentée à la fin du projet. On imaginait créer un jeu plus tourné vers des modes de jeu différents et de modalités aussi qui seraient distinctes au sein du simple nom du jeu Tic Tac Toe. On a bien réussi à faire nos idées de joueur contre joueur et celle contre machine, mais notre idée d'augmenter le projet pour créer de nouveaux modes et difficultés a dû être abandonnée à cause du manque de temps et de la disponibilité de quelques intégrants à des occasions isolées (events, restaurants, voyages...). La création du mode joueur contre joueur a été la première à être créée et n'a pas posé autant de problèmes. Au contraire de ce qu'on imaginait, contre la machine et ses difficultés nous a posé beaucoup plus de problèmes. À la fin du projet avec nos modes de jeu et difficultés complètes, nous étions satisfaits du résultat, même si quelques idées ont été abandonnées au cours du projet.

ADAPTATION

Nos adaptations ont été au sein des idées abandonnées et d'autres additionnées. On a eu l'idée d'insérer des images PNG des symboles "X" et "O", mais à la fin cela nous a été inutile, car on pouvait très bien mettre seulement un texte avec les lettres. Ils nous a manqué un peu de temps pour pouvoir adapter quelques idées comme créer quelques nouvelles difficultés et modes de jeu, mais en discutant entre le groupe, on a conclu que transborder de difficultés et de modes de jeu nous paraîtrait plus original, mais très loin de l'objectif primaire qu'on voulait atteindre. L'idée de créer un mode "impossible" a été très bien reçue et on nous a adaptés à trouver des tactiques pour additionner au jeu, en fonction que gagner de la machine soit, en effet, impossible.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ

PROBLÈMES ET SOLUTIONS

P.1 - Créer le tableau originel de TicTacToe de façon que toutes les cases soient touchables, fonctionnelles et singulières.

S.1 - On a eu l'idée d'utiliser nos connaissances pour utiliser le tkInter. Au début toutes les cases étaient touchables et fonctionnelles, mais elles se mêlaient entre soi: lorsqu'on cliquait dans une seule case, on avait une réponse pour des multiples cases appuyées d'une fois. Après quelques tests, on a trouvé des problèmes aux premiers "def" du programme: on voulait trop résumer les lignes, et on finissait par se perdre à la compréhension, alors que les résultats étaient confus. Après avoir tout changé de façon qu'on n'essaye pas de résumer celui-ci, on a réussi à résoudre ce problème.

P.2 - Utiliser "Assert".

S.2 - De même que dans P.1, cela nous a apporté beaucoup de problèmes car on finissait par se perdre dans les révisions en mêlant TkInter et les fonctions avec Assert.

P.3 - Les difficultés étaient difficiles à créer.

S.3 -Après quelques tests et recherches sur internet, on a réussi à faire en sorte que la CPU nous réponde parfaitement dans le tableau. Pourtant, il y avait encore quelques problèmes pour chaque difficulté.

P.3(1) Difficulté Facile:

S.3(1) - On ne savait pas par où commencer, car on voulait que cela nous réponde aléatoirement, mais on ne savait pas comment le faire. A la fin, cela ne nous a pas autant été difficile. A la fin, c'était même amusant car nous mêmes on essayait forcément de faire la machine gagner, de si difficile que c'était de perdre.

P.3(2) Difficulté Difficile:

S.3(2) - On voulait créer quelque chose d'extrêmement difficile ou même impossible à gagner. On a fait en sorte que la machine essaye toujours de bloquer le joueur, ou bien de tirer profit de la situation et de gagner du joueur. Elle est dans une grande partie formée de "if" et "else".

P.4 - On ne savait pas comment faire que les cases soient d'une autre couleur et qu'on puisse les différencier du reste de l'onglet; aussi colorer seulement les cases du jeu.

S.4 - Après quelques recherches on a choisi alors de changer la couleur vers "lightblue", mais cela nous a pris beaucoup de temps pour faire en sorte que seulement les cases choisies soient de cette couleur, car dans des diverses circonstances tout l'onglet avait ses boutons bleus.

P.5 - Python Tutor nous était inutile, donc on avait à tout relire le programme avec debug ou bien sans aide.

S.5 - On a commencé à commenter dans les parties plus importantes pour ne pas se perdre, car le programme était déjà peu à peu en train de devenir extrêmement long.

P.6 - Les boutons de réinitialier les compteurs étaient confus et nous rapportait quelques problèmes.

S.6 - On a eu l'idée de créer un manuel et de tester diverses fois jusqu'aux boutons aient les résultats qu'on s'attendait à avoir..

P.7 - La taille du programme était difficile à lire et à comprendre.

S.7 - Pendant quelques jours on s'est priorisés à diminuer la taille du programme, car on avait fait en sorte que cela soit plus grand que 1500 lignes de programmation. On réussit alors à l'optimiser à plus ou moins 1000 lignes.

P.8 - Situation finale du jeu.

S.8 - Comme on peut le voir dans le programme, chaque fois que quelqu'un gagne, on a pas le temps de voir la situation finale du jeu. On a essayé avec le module time, plus précisément avec la fonction sleep() pour que le programme attende quelques secondes avant de réinitialiser la partie. À la suite de nombreux tests, on s'est rendu compte que cela était inutile, on est pas très sûrs du pourquoi mais notre hypothèse est que : avant que le programme

modifie la fenêtre de tkinter, il faut que tous les procédés, calculs, etc... soient finies. Ceci est dû à que même en mettant la fonction sleep dans la ligne suivante à celle où le programme est censé placer le X, le programme attendait quelques secondes, sans placer le X puis il réinitialise tout sans nous laisser voir le X et donc la situation finale. On ne peut pas laisser sans réinitialiser car on aurait l'option de cliquer sur d'autres boutons et donc le programme continuerait jusqu'à ce qu'il n'y ait plus de place même si quelqu'un avait déjà gagné. En revanche, quand il y a une égalité, il est possible de voir la situation finale car il n'y a pas de risque de mal fonctionnement du programme, pour rejouer il va falloir cliquer sur le bouton `restart_partie`.

AUTRE

P.Organisation 1 - Manque de communication et disponibilité au début des vacances.

S.Organisation 1 - A cause des problèmes personnels de chacun et de ses voyages individuels, on a pas réussi à se rejoindre pendant au moins une semaine. C'est aussi important de dire que, à cause d'autant d'informations des voyages et d'abondantes idées en relation au projet, nos connaissances étaient toutes mêlées, et le début du projet a été un peu problématique puisqu'on ne se rappelait pas de notre cours à sa totalité. Grâce à Jupyter, après quelques jours de travail, on est parvenu à bien s'en mettre d'accord avec Python.

P.Organisation 2 - Affaire Guilherme: il n'était pas disponible à faire des devoirs pendant deux semaines.

S.Organisation 2 - Comme Pedro et Daniel voulaient commencer tôt le projet, dès qu'ils ont eu le temps pour commencer, ils l'ont fait sans Guilherme. Parallèlement à tout ça, Pedro a beaucoup essayé d'appeler Guilherme pour participer, mais les différences d'horaire et l'indisponibilité lui ont fait faire un document temporaire en écrivant tout ce qui se passait pour que, après ces deux semaines, Guilherme puisse le lire et se mettre en commun avec nous. Dès que les 14 jours sont passés, Guilherme est réapparu et nous a bien mieux aidé.

P.Organisation 3 - Fuseau horaire.

S.Organisation 3 - Pedro (France) et Daniel (Espagne) étaient en Europe et Guilherme (Brésil) avec 5 heures de différence. Les intégrants d'Europe alors ont décidé de travailler pendant la nuit pour que celui de l'Amérique du Sud puisse nous aider.

TESTS

Pendant tout le projet, on s'est basé dans des tests pour pouvoir savoir si nos difficultés étaient conformes à leurs noms et si les modes ne comportaient aucune erreur. On a dû jouer le jeu des milliers de fois pour tester toutes les possibles erreurs. On a fait beaucoup de tests aussi pour voir si, esthétiquement, le jeu nous plaisait visuellement. On a essayé aussi quelques fois d'utiliser "assert", mais cela était trop technique, et nos problèmes étaient aux règles du jeu en soi. On a testé le jeu quelques fois aussi et avec ceux-ci, on a corrigé quelques choses dans le programme.

> OUVERTURE

PERSPECTIVES D'EVOLUTION

Nous sommes sûrs que ce projet pourrait bien être beaucoup plus plaisant à voir et à jouer. Nous croyons que la principale idée qui nous peut être bien intéressante à évoluer futuramente c'est de faire que le jeu joue machine contre machine, une idée abandonnée au long du projet, ainsi que le tableau de ranking des joueurs et ses victoires. Notre programme était aussi extrêmement long, et on imagine que diminuer cela serait bien mieux.