



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : Projet Ascartes

> PRÉSENTATION GÉNÉRALE :

• *Idée et objectifs*

En utilisant le cours et en nous appuyant sur les connaissances acquises en classe, les élèves de la spécialité NSI au lycée Blaise Pascal devaient préparer un projet sous forme de jeu sur Python en utilisant la programmation orientée objet, ainsi qu'un diaporama expliquant ce projet. Nous avons 1 mois pour imaginer, produire et finaliser ce projet. De ce fait, notre professeur nous avait proposer deux types de jeu : un jeu de cartes ou un jeu de combat. Notre manuel de NSI contenant déjà quelques pistes concernant les jeux de cartes, je me suis donc intéressé à ce jeu.

• *Origines et intérêts du projet*

Nous avons choisi de programmer en Python le jeu de cartes "Ascartes" car ce jeu paraissait assez simple à jouer (et donc à coder !!) et car nous y jouons assez régulièrement. Ce jeu est une variante du très célèbre jeu « le 8 Américain ». Coder ce jeu n'a finalement pas été si simple du fait des nombreuses règles s'appliquant à ce jeu et de la nouveauté de ce modèle de programmation. Il fallait adapter sa pensée à la programmation orientée objet.

> ORGANISATION DU TRAVAIL :

• *Présentation de l'équipe (prénom de chaque membre et rôle dans le projet)*

HAKOMANI Angel est le développeur, chef de projet.

TAOFIFENUA Levi, coordinateur et correcteur du code.

TEVANU Rachel, co-développeuse, innovatrice et rédactrice.

• *Répartition des tâches*

HAKOMANI Angel était chargé de programmer le jeu en Python.

TAOFIFENUA Levi devait surveiller l'avancée du projet, faire les jeux de tests et corriger les erreurs de programmation.

TEVANU Rachel devait aider à la programmation du jeu, chercher de nouvelles idées pour le jeu et rendre compte par écrit l'ensemble du projet.

• *Organisation du travail (répartition par petits groupes, fréquence de réunions, travail en dehors de l'établissement scolaire, outils/logiciels utilisés pour la communication et le partage du code, etc.)*

Une semaine avant les vacances scolaires d'avril, nous nous étions réunis 3 fois dans la semaine pour imaginer notre projet et pour se répartir les tâches. Durant les vacances, tous les 3 jours, Angel devait programmer le jeu sur l'EDI nommé PyScripter, aidé par Rachel en appel vidéo sur Messenger avec lui. Ensuite, un appel de groupe nous permettait de récapituler nos actions produites durant la journée. Puis, le fichier était transféré à Levy par Gmail. Ce dernier commençait à lancer les jeux de tests et corriger les erreurs qui auraient pu être faites par Angel. De plus, Rachel prenait en note toutes nos actions et nos démarches qu'elle mettait sur un fichier Word.

LES ÉTAPES DU PROJET :

- *Présenter les différentes étapes du projet (de l'idée jusqu'à la finalisation du projet)*

[Angel] Je suis parti du principe du jeu de cartes « Ascartes », variante du jeu « 8 Américain », et je comptais le programmer en Python en privilégiant des règles simples pour une meilleure compréhension du jeu. Mon livre de NSI me proposait 2 classes pouvant me servir de base pour mon jeu, une classe Carte et une classe Jeu de Cartes, que j'ai donc utilisé. J'ai ensuite créé une classe Joueur que Levy a amélioré par la suite afin de compléter les 2 autres classes.

[Levy] J'ai proposé de mettre une condition d'arrêt dès le début du jeu si on lançait par erreur ce dernier. J'ai aussi eu l'idée de mettre un tuto au début du jeu, que l'on pouvait passer si l'on connaissait les règles du jeu Ascartes.

[Rachel] J'ai fait des recherches pour compléter le tuto du début. De plus, notre professeur de NSI nous a vivement conseillé d'utiliser la fonction sleep de la bibliothèque « time » afin de permettre au joueur de voir le jeu défiler à l'écran. J'ai donc soumis mon idée au groupe.

[Angel] Pour compléter l'idée de Rachel, j'ai créé une variable vitesse_défilement pour pouvoir demander à l'utilisateur à quelle vitesse voulait-il que le jeu défile à l'écran. Cela permettait d'inclure plus de personnes dans ce jeu, tel que les enfants qui ne lisent pas assez vite.

[Rachel] Dès lors, nous avons commencé puis finalisé le jeu de test sur l'EDI PyScripter, afin que le Joueur 1 et le Joueur 2 puissent jouer à tour de rôle leurs cartes selon les règles du jeu. De plus, nous avons rajouté certaines cartes spéciales, qui avantage le joueur qui la pose. Cependant, certaines cartes spéciales tel que le 10 ont été enlevées car elle alourdissait le programme.

➤ FONCTIONNEMENT ET OPÉRATIONNALITÉ :

• **Avancement du projet (ce qui est terminé, en cours de réalisation, reste à faire)**

➤ Ce qui est terminé :

L'utilisateur peut observer 2 joueurs s'affronter durant une partie d'Ascartes. Il peut également nommer ces joueurs et choisir la vitesse à laquelle le jeu défile à l'écran. Ces derniers jouent leurs cartes selon les règles du jeu « Ascartes », expliquées au début du programme par un tutoriel que l'on peut passer. Le joueur qui n'a plus de cartes au tour suivant gagne la partie.

➤ Reste à faire :

Certains cas de figure n'ont pas été traités à temps :

- Si les 2 joueurs ont fini de jouer toutes leurs cartes au même tour,
- Si la pioche est vide
- Pouvoir jouer plusieurs cartes de même figure ou même valeur en même temps (familles)
- Jouer un As et changer la couleur du talon par la couleur que le joueur a le plus dans sa main.

• **Approches mises en œuvre pour vérifier l'absence de bugs et s'assurer de la facilité d'utilisation du projet**

Lancer plusieurs fois le jeu de test à chaque nouvelle mise à jour du programme nous a permis de trouver les bugs présents et de les corriger. Pour s'assurer que le projet soit facile à utiliser, nous avons demandé à des enfants/pré-ados de notre entourage de tester le programme et de nous donner leurs avis.

• **Difficultés rencontrées et solutions apportées**

- Les joueurs ne pouvaient pas rejouer après avoir joué la carte « Valet », nous avons donc utilisé des variables précédemment créées nommées `Joueur1ajouéunecarte` et `Joueur2ajouéunecarte` qui permette de rester dans la boucle si l'on joue le « Valet ».
- La condition `while not` qui permet d'arrêter le jeu dès qu'un des 2 joueurs n'a plus de cartes ne fonctionnait pas pour les 2 joueurs. Nous avons donc forcé la sortie du programme avec la fonction « `break` ».

> OUVERTURE :

• *Idées d'améliorations (nouvelles fonctionnalités)*

- Coder le programme pour que l'utilisateur soit capable de jouer lui-même contre l'ordinateur.
- Apporter de l'esthétique au projet, développer l'aspect visuel du jeu en affichant des représentations de vraies cartes dans le jeu.

• *Stratégie de diffusion pour toucher un large public (faites preuve d'originalité !)*

- Diffuser le jeu pour qu'il soit jouable en ligne sur un site déjà créé.
- Proposer à des adolescents de s'initier à l'informatique au collège/lycée en leur présentant ce jeu.

• *Analyse critique du résultat (si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?)*

- Nous créerions un jeu plus interactif et jouable contre l'ordinateur, ainsi qu'une interface visuelle créé avec Pygame.

DOCUMENTATION

- *Spécifications fonctionnelles (guide d'utilisation, déroulé des étapes d'exécution, description des fonctionnalités et des paramètres)*
- *Spécifications techniques (architecture, langages et bibliothèques utilisés, matériel, choix techniques, format de stockage des données, etc)*
- *Illustrations, captures d'écran, etc*