EL GAMOUNE Elyes

LEPICHOUX Zoé

Rapport du projet « Qui est ce ? »



Ce rapport présente le développement du jeu "Qui est-ce". Nous avons pour objectif de développer une application qui permettra à l'utilisateur de jouer à "Qui est-ce" en choisissant un personnage mystère et en posant des questions pour déterminer l'identité de ce personnage. Dans ce rapport, nous détaillerons les différentes étapes de développement de cette application, depuis l'analyse jusqu'à la réalisation et les tests effectués.

Table des matières

[**Introduction** 3](#_Toc135666089)

[**Analyse** 3](#_Toc135666090)

[**Réalisation** 3](#_Toc135666091)

[**Tests** 4](#_Toc135666092)

[**Perspectives d'amélioration** 4](#_Toc135666093)

[**Conclusion** 4](#_Toc135666094)

**Introduction**

Le jeu "Qui est-ce" est un jeu de société classique qui a captivé des générations de joueurs. Son principe est simple : deviner le personnage mystère choisi par l'adversaire en posant des questions ciblées sur ses caractéristiques physiques. Dans le cadre de ce projet, nous avons pour ambition de développer une application informatique qui permettra aux joueurs de profiter de l'expérience "Qui est-ce" directement sur leur ordinateur, en utilisant une intelligence artificielle comme adversaire.

La problématique qui se pose est la suivante : comment créer une application interactive et divertissante basée sur le jeu "Qui est-ce", en exploitant les capacités de l'intelligence artificielle pour proposer un adversaire virtuel capable de poser des questions pertinentes et de fournir des réponses réfléchies ?

Pour atteindre cet objectif, le développement de l'application se déroulera en plusieurs étapes clés. Tout d'abord, il sera nécessaire de concevoir une interface conviviale et attrayante, permettant à l'utilisateur d'interagir facilement avec le jeu. Cela inclura des éléments tels que la présentation d’une liste des personnages, les boutons qui affichent les questions qui seront posées sur les caractéristiques et les affichages des résultats.

Ensuite, il faudra intégrer une intelligence artificielle qui agira comme l'adversaire virtuel. Cette IA devra être capable de poser des questions pertinentes en fonction des caractéristiques des personnages et des réponses précédentes de l'utilisateur. Elle devra également être capable d'analyser les informations fournies par l'utilisateur pour affiner ses hypothèses et choisir les questions les plus efficaces pour arriver à une réponse.

Enfin, des mécanismes de gestion du jeu seront mis en place, tels que la détection de la victoire ou de la défaite, la possibilité de rejouer une partie.

**Analyse**

Dans cette première étape du développement de l'application, nous avons analysé en détail les différentes fonctionnalités nécessaires et identifié les bibliothèques requises pour les implémenter avec succès.

La première fonctionnalité clé de l'application est la capacité de charger les personnages à partir d'un fichier JSON. Pour réaliser cette fonctionnalité, nous avons utilisé la bibliothèque JSON, qui offre des méthodes pour lire et écrire des données au format JSON. Nous avons pu extraire les informations relatives aux différents personnages à partir du fichier JSON et les stocker dans une structure de données appropriée pour une utilisation ultérieure.

La deuxième fonctionnalité essentielle consiste à choisir un personnage mystère de manière aléatoire parmi les personnages chargés à partir du fichier JSON. Pour ce faire, nous avons utilisé la bibliothèque random, qui fournit des fonctions permettant de générer des nombres aléatoires. Grâce à cela, nous avons pu sélectionner aléatoirement un personnage mystère à chaque nouvelle partie, ce qui ajoute de la variété et de l'excitation à l'expérience de jeu.

La troisième fonctionnalité implique la possibilité de poser des questions à l'utilisateur concernant les caractéristiques physiques du personnage mystère. Pour mettre en œuvre cette fonctionnalité, nous avons créé une fonction dédiée qui affiche une boîte de dialogue à l'utilisateur avec une question spécifique. Nous avons utilisé la bibliothèque tkinter pour créer cette interface conviviale qui permet à l'utilisateur de répondre à la question posée.

La quatrième fonctionnalité clé est la capacité de trouver le personnage mystère en fonction des réponses fournies par l'utilisateur. Pour cela, nous avons développé une fonction qui parcourt la liste des personnages possibles et met à jour cette liste en fonction des réponses de l'utilisateur. Cela implique de filtrer les personnages en fonction des caractéristiques physiques mentionnées par l'utilisateur et de maintenir une liste mise à jour des personnages correspondant aux réponses fournies.

Enfin, la cinquième fonctionnalité concerne la gestion de l'interface graphique de l'application en utilisant la bibliothèque tkinter. Nous avons créé une fenêtre principale qui englobe tous les éléments de l'interface graphique, tels que les boutons pour poser des questions et pour afficher la liste des personnages. Nous avons utilisé les fonctionnalités de la bibliothèque tkinter pour organiser ces éléments de manière attrayante et intuitive, offrant ainsi aux utilisateurs une expérience de jeu fluide et agréable.

Concernant le fichier Json, celui-ci sert de base de données pour contenir tous les personnages. De ce fait, il aurait fallu écrire à la main 60 personnages avec leurs 10 caractéristiques. Vous conviendrez que c’est très fastidieux. Pour pallier ce problème, nous avons écris une fonction dans un fichier à part pour générer aléatoirement 60 personnages, répartir aléatoirement les 10 critères pour chaque personnage avec la bibliothèque Random et génère pour chaque personnage un nom/prenom avec la bibliothèque Faker.

En combinant ces différentes fonctionnalités et en exploitant les bibliothèques appropriées, nous avons pu développer une première version de l'application "Qui est-ce" avec une interface graphique interactive, la capacité de charger les personnages depuis un fichier JSON, de poser des questions à l'utilisateur, de trouver le personnage mystère et de fournir une expérience de jeu engageante.

**Réalisation**

**Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Description générée automatiquement**

Voici l’interface du jeu. Nous pouvons voir plusieurs boutons que le joueur peut utiliser.

Les boutons de couleurs sont cliquables et affiches la question en fonction du critère affiché sur le bouton.

Le bouton « afficher la liste des personnages » affiche tous les personnages présents et devinable.

Le bouton « jouer » permet de lancer le jeu. Comme énoncé dans la partie Analyse, un personnage mystère est généré aléatoirement et l’ia commence à poser des questions puis nous devons choisir oui ou non pour que l’ia affine ses recherches pour deviner le personnage

mystère comme suit : Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Dans cette deuxième étape du développement de l'application, nous avons approfondi la mise en œuvre des différentes fonctionnalités en utilisant les bibliothèques nécessaires pour assurer le bon fonctionnement technique de l'application.

Tout d'abord, pour charger les personnages à partir du fichier JSON, nous avons utilisé la bibliothèque json. Nous avons ouvert le fichier JSON contenant les données des personnages à l'aide de la fonction open() et l'avons chargé à l'aide de la fonction json.load(). Cela nous a permis d'obtenir un objet Python contenant la liste des personnages avec toutes leurs caractéristiques.

Pour choisir aléatoirement un personnage mystère parmi les personnages chargés, nous avons utilisé la bibliothèque random. En utilisant la fonction random.choice(), nous avons sélectionné un élément aléatoire de la liste des personnages, qui deviendra le personnage mystère pour la partie en cours. Cette approche garantit une sélection aléatoire et équitable du personnage mystère à chaque fois que le jeu est joué.

En ce qui concerne la fonctionnalité de poser des questions à l'utilisateur, nous avons développé une fonction spécifique qui utilise la bibliothèque tkinter. Nous avons créé une boîte de dialogue modale à l'aide de la fonction messagebox.askyesno() qui affiche une question spécifique à l'utilisateur avec les caractéristiques du personnage. L'utilisateur peut répondre en sélectionnant "oui" ou "non". La fonction renvoie ensuite la réponse de l'utilisateur, qui sera utilisée pour mettre à jour la liste des personnages possibles.

La fonctionnalité de trouver le personnage mystère en fonction des réponses de l'utilisateur repose également sur l'utilisation des bibliothèques random et tkinter. Nous avons mis en place une boucle qui se répète tant qu'il reste plus d'un personnage possible. À chaque itération, nous choisissons aléatoirement une caractéristique à poser à l'utilisateur en utilisant random.choice(). Ensuite, nous posons la question correspondante à l'utilisateur à l'aide de la fonction de question précédemment mentionnée. En fonction de la réponse de l'utilisateur, nous mettons à jour la liste des personnages possibles en utilisant une compréhension de liste pour filtrer les personnages en fonction de la caractéristique et de la réponse. La boucle continue jusqu'à ce qu'il ne reste plus qu'un seul personnage possible.

En ce qui concerne l'interface graphique de l'application, nous avons utilisé la bibliothèque tkinter. Nous avons créé une fenêtre principale à l'aide de la classe Tk() et avons ajouté des éléments graphiques tels que des cadres, des libellés et des boutons à l'aide des classes fournies par tkinter. Nous avons également configuré les propriétés visuelles des éléments, comme la police, la taille, les couleurs, etc. Les boutons étaient liés à des fonctions spécifiques, ce qui permettait de déclencher des actions lorsque l'utilisateur les cliquait.

En ce qui concerne le fichier JSON, il a été utilisé comme une base de données pour stocker tous les personnages du jeu. Plutôt que d'écrire manuellement les 60 personnages avec leurs 10 caractères, nous avons opté pour une approche technique plus efficace. Nous avons créé une fonction distincte dans un fichier séparé pour générer aléatoirement les 60 personnages avec leurs caractéristiques. Pour ce faire, nous avons utilisé la bibliothèque random pour répartir aléatoirement les 10 critères pour chaque personnage.

Cette fonction de génération aléatoire des personnages utilise également la bibliothèque Faker pour générer aléatoirement un nom et un prénom pour chaque personnage. Cela nous permet d'obtenir une base de données de personnages variés et fictifs pour le jeu.

Lorsque l'application est exécutée, elle charge automatiquement les personnages à partir du fichier JSON généré aléatoirement. Cela garantit que chaque session de jeu est basée sur un ensemble de personnages uniques et différents.

**Tests**

Dans cette dernière étape de développement, nous avons consacré du temps à tester minutieusement l'application afin de nous assurer de son bon fonctionnement et de sa qualité. Les tests ont été effectués pour toutes les fonctions de l'application, à la fois de manière individuelle et en interaction les unes avec les autres, afin de garantir leur cohérence et leur bon fonctionnement global.

Nous avons également testé l'interface graphique de l'application pour s'assurer de sa fluidité et de sa facilité d'utilisation.

**Perspectives d'amélioration**

Nous avons réussi à développer une application fonctionnelle qui permet de jouer à "Qui est-ce" sur un ordinateur. Cependant, des perspectives d'amélioration pourraient être envisagées, telles que l'ajout de nouveaux personnages et de nouvelles caractéristiques.

Nous voulions proposer dans l’affichage de la liste des personnages un affichage d’image ou de dessins générer par la bibliothèque pygame en fonction des critères des personnages. Nous avions commencé à le faire mais nous n’avions pas assez de connaissance pour le faire pour 60 personnages. Puis les dessins étaient approximatifs.

Des perspectives d’améliorations ont déjà été réalisé. En effet, tout le programme que nous avons conçu est issus de nos connaissances personnelles (Elyes ayant fait un bts informatique a déjà pu pratiquer certaines bibliothèques). Ensuite nous avons utilisé le site « blerow.blogspot.com » rubrique python jeu « qui est ce » pour avoir une idée possible de l’algorithme à utiliser (car connaitre les règles du jeu c’est bien, les appliquer en algorithme c’est mieux). Pour finir, nous avons utilisé la doc pour la bibliothèque faker avec le site « docstring.fr ». Nous n’avions pas envie de nous attarder sur une rédaction manuelle éreintante, donc nous voulions trouver une solution automatique pour générer tous les personnages mais nous n’avions aucune connaissance d’une telle bibliothèque.

A la fin de la réalisation et concaténation de nos recherches et d’utilisation des ressources précédemment énoncées, nous avons passé tout le code sur chatGpt pour corriger les éventuelles erreurs et avoir une cohérence sur tout le code « patchwork » que nous avions tapé. De toute manière, chatgpt n’aurait pas pu générer tout ce code-là d’une traite car il lui manque trop de précision, de consignes etc.

**Conclusion**

En conclusion, le développement de l'application "Qui est-ce" a été un succès. Nous avons suivi les différentes étapes de développement, depuis l'analyse jusqu'aux tests, pour garantir un bon fonctionnement de l'application. Des perspectives d'amélioration pourraient être envisagées pour améliorer l'application et offrir une expérience de jeu encore meilleure.

Tout fonctionne mais nous aurions pu apporter plus de visibilité au joueur concernant les personnages avec des images pour qu’il sache qui faire deviner à l’ia.