## ligne horizontale

0



Projet Python - Quizz jeu vidéo - “Game over”

20.09.2023

**─**

Bancelin Andreia / Leforestier Antoine / Devleeschauwer Lilian / Mareschal Mathis

CESI Nancy

3 Rue du bois de la Champelle

54500 Vandoeuvre-lès-Nancy

# Vue d'ensemble

# Analyse du cahier des charges

Différence entre “ Vous devez définir le temps minimum à respecter par question” et

“Vous devez définir le temps minimum à respecter de temps réponse par question”. Et sens de cette phrase? Que signifie un temps minimum de réponse ??? Est-ce un temps maximum avant que la réponse soit invalidée ?

Le projet consiste à programmer un quizz sous python, sur le thème des jeux vidéo.

## Organisation du projet : Step by step

* Identification des contraintes et des objectifs primaires et secondaires. Tri par priorité de ces dits objectifs.
* Découpage en tâches et en sous-tâches à effectuer.
* Mise en place d’un moyen de suivi de notre avancement et répartition des tâches.
* Etude fonctionnelle et technique : quelle fonction pour faire quoi ?
* Architecture du logiciel et structuration de la base de données.
* Développement logiciel.

# Objectifs

1. Créer un logiciel fonctionnel et user friendly, correspondant aux contraintes techniques demandées par le client.

* Base de données contenant toutes les questions possibles existantes.
* Ajout de données temporaires compilant les réponses choisies par l’utilisateur.
* Maintenabilité/ évolutivité. Possibilité de simplement maintenir et faire évoluer le soft.
* Sélectionner en Bdd des questions différentes pour chaque partie, aléatoirement.

1. Y ajouter des caractéristiques spécifiques supplémentaires qui donnent une plus-value au logiciel, sans entraver pour autant son fonctionnement initial ni complexifier inutilement son utilisation.
2. Créer un manuel d’utilisation - “règles du jeu” - qui permettra aux utilisateurs de s’y référer en cas de difficulté.

# Caractéristiques

## Règles basiques

* Minimum 10 questions avec chacune un nombre de points et un temps imparti.
* Scores au fur et à mesure et score final donc stockage des résultats et incrémentation.
* Temps réalisé par question et au total en fin de partie.
  + ET temps à respecter avant échec de la question.
  + ET temps enregistré pour chaque question avec la date de la réponse

## Améliorations choisies

* Interface graphique
* Système multijoueur
* Niveaux de difficulté
* Question bonus (en cas de win streak)

## Améliorations supplémentaires

* Menu des high scores
* Win streak (nombre de points qui augmente par question si on fait des séries de bonnes réponses d’affilée - si 5 bonnes réponses d’affilée passer à une question bonus plus dure qui donne masse de points)
* [Versions anglaise et espagnole du jeu]

## Logiciel : kesako ?

~~Logiciel avec une interface graphique, avec un menu principal (possibilité de quitter à ce moment-là ou bien de “commencer”), puis entrer le nom de joueurs sur une prochaine fenêtre (jusqu’à 4 joueurs), puis choisir le niveau de difficulté (facile, moyen, difficile, expert)~~

~~[Éventuellement choisir le thème du quizz (jeux d’aventure, de combat, party game, mmo, etc), puis la décennie de sortie du jeu (80’s, 90’s, 2000, 2010…).]~~

~~Puis menu récapitulatif avec la possibilité de revenir en arrière pour changer un paramètre ou bien démarrer le jeu.~~

~~Puis le jeu se lance “Le jeu commence”, puis bouton “Démarrer la partie”. Ensuite on accède aux questions du quizz, qui se succèdent les unes après les autres, une fois la réponse sélectionnée et validée par le joueur (choix entre réponses A B C D, sélectionnables par le biais de boutons) ou si le temps imparti est écoulé.~~

* ~~Questions correspondant bien entendu au thème et à la difficulté choisis.~~
* ~~Les questions ne sont pas toujours les mêmes.~~

~~Ainsi : base de données qui contient un nombre de questions supérieur aux questions qui vont être posées.~~

~~Par exemple, pour 20 questions dans une partie on pourrait choisir parmi une quarantaine de questions. Les questions d’une partie sont choisies aléatoirement au sein d’une bibliothèque (composée de sous rubriques en fonction des paramètres choisis).~~

~~Représentation visuelle :~~

**~~Bibliothèque~~**

*~~Aventure~~*

~~Facile~~

~~Moyen -> Les questions seraient prises ici si l’utilisateur sélectionne aventure et niveau moyen. Sur les 40 questions possibles, 20 seront choisies pour cette partie, aléatoirement.~~

~~Difficile~~

~~…~~

~~Une fois la question utilisée, elle se supprime temporairement de la base de données afin qu’elle ne revienne plus à nouveau au cours de la même partie en cours.~~

~~A chaque fois que la réponse a été choisie par le joueur, une nouvelle fenêtre apparaît pour annoncer si la réponse choisie était la bonne réponse : True or False. Sur la même fenêtre un bouton “joueur suivant” apparaît, et une fois qu’on clique dessus on passe joueur suivant et la prochaine question, encore et encore jusqu’à la fin du jeu.~~

~~A chaque nouvelle réponse on a une ligne (en haut ou en bas de la fenêtre) qui rappelle le nombre de points actuel de chaque joueur.~~

~~Questions au tour par tour : chaque question appelle un joueur différent, pas dans le même ordre, pas forcément plusieurs fois de suite. “Question 1 : Lilian, en quelle année est sorti tel jeu ?”, “Question 2 : Antoine, de quoi parle tel jeu ?”, etc~~

~~Une fois les 20 questions terminées, une fenêtre annonce “Fin de la partie”, et un bouton propose de passer aux résultats “Voir les scores”.~~

~~La fenêtre qui s’affiche une fois qu’on a appuyé sur ce bouton montre ainsi le classement des joueurs et le nombre de points au total de chaque joueur. Possibilité de voir les détails (?), puis bouton “Suivant”.~~

~~Fenêtre donnant le nom de l’utilisateur qui a gagné puis annonçant donc qu’il est le vainqueur “Ibrahim, tu as gagné !”. En bas de cette fenêtre deux propositions : relancer une nouvelle partie ou bien quitter le jeu. Si les joueurs choisissent de quitter le jeu, le programme se termine. Sinon, retour sur le menu des paramètres de la partie (donc le menu avec le niveau, le thème, etc), et on reprend, etc…~~

**A TOUT MOMENT** on peut quitter.

-> enregistrer ou non les scores déjà effectué avant la fermeture ?

## Analyse fonctionnelle

//TODO

Ici mettre la bête à corne

Cas d’utilisation de base

Cas d’utilisation détails

1. Partie base de données :

* Prendre toutes les questions dans la BDD et les stocker dans une variable type ‘liste’. De même avec les réponses dans une autre variable.
* En fonction du choix des joueurs, prendre dans la variable liste les questions correspondant à la catégorie demandée.
* Déterminer si la réponse donnée par le joueur correspond à la réponse attendue.

1. Partie temporalité :

* Programmer un chronomètre
* Historiciser les parties (tel joueur a joué tel jour).

1. Partie scores :

* Stocker :
  + les points de chaque joueur. Afficher au fur et à mesure de la progression les points.
  + le nombre de bonnes réponses de chaque joueur (pour la fin de la partie).
  + le nombre de bonnes réponses à la suite de chaque joueur.
* Condition qui permet d’accéder à la question bonus si 5 bonnes réponses d’affilée (fenêtre supplémentaire).
* A la fin de la partie afficher le score détaillé.
* Également entrer le nom de la personne qui a gagné en fonction du nombre de points max “... a gagné”,et qui est de le deuxième, troisième et quatrième.
* Avoir un classement “High Scores” avec les meilleurs scores ?

1. Partie ordre :

* Chaque joueur répond l’un après l’autre aux questions toujours dans le même ordre.
* Fonction pour choisir de façon aléatoire les questions qui sortent.

# Grandes étapes

## Analyse des contraintes techniques et des attentes du client

Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici.

## Architecture logicielle / structure de la base de données

Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici.

## Fonctions utilisées

Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici Insérez votre texte ici.