

## **Rapport Final SAE : 3.02 développer une application communicante**



Participants : Alexis Lemonnier, Axel Plault, Maël Bodin, Armand Zireg

# **Sommaire**

- I. Introduction
- II. Rappel du cahier des charges
- III. Comparaison avec les tâches réalisées
- IV. Planning prévisionnel
- V. Planning réel
- VI. Principaux algorithmes
- VII. Conclusion

## I. Introduction :

Le jeudi 6 avril 2023, l'IUT organise la journée "Rendez-vous savant" qui aura pour objectif de faire découvrir aux lycéens l'IUT ainsi que les différentes propositions qu'il propose. Dans ce cadre-là, des élèves de MP nous ont demandé de créer un quiz qui aura pour objectif de départager les élèves sur une série de 20 questions.

## II. Rappel du cahier des charges

Informations :

- Case Nom
- Case Prénom
- Case Classe

QCM :

- 4 propositions de réponses
- 1 seule de juste
- 20 questions au total

L'annonce des résultats se fait seulement quand tout le monde a terminé.

Pendant les 20 premières minutes, classements visibles uniquement pour les organisateurs.

- Classement individuel

Score final individuel = Somme des points accumulés après chaque question

Score + (Nom, Prénom, Classe)

- Classement par classe

Moyenne de toutes les notes d'une même classe

Pour téléphone seulement : IOS et Android

- Code pour entrer dans la séance ou QR code

Si possible insérer des images pour les questions

Prévoir pour 120 personnes

Règle générale du quiz :

Le joueur doit répondre à 20 questions. Il bénéficie de 30 secondes pour répondre à chaque question.

Les questions sont affichées de façon aléatoire.

Le vainqueur est celui qui obtient le plus de points.

Chaque bonne réponse rapporte des points. Ce nombre de points diminue en fonction du temps de réponse. Pour éviter les égalités, le temps de réponse vaut des points.

Le jeu quiz démarre en même temps pour tous les joueurs. Dès l'instant où un joueur a répondu à une question (réponse correcte ou fausse), il passe à la question suivante.

La bonne réponse pour chaque question est affichée pendant 3 secondes dès l'instant où le joueur a répondu à la question.

La durée maximale du jeu est de 11 minutes (10 minutes de jeu + 1 minute consacrée à l'affichage des bonnes réponses).

### III. Réalisation du cahier des charges

Le site est exclusivement adapté au format téléphone et pour qu'il s'adapte aux différents formats existants.

Quand les joueurs arrivent sur le site, ils doivent d'abord entrer leurs Nom et Prénom, ainsi que leurs numéros de groupe, ces données qui seront ensuite sauvegardées dans une base de données et où un ID leur sera attribué.

Pour que le quizz démarre en même temps pour tous les joueurs nous avons mis en place une page d'accueil répertoriant les règles du quiz et où ils devront attendre le signal de l'organisateur avant de lancer la partie.

Reprenant les fichiers transmis, nous avons repris les 20 questions avec les 4 choix de réponse, la bonne réponse, ainsi que les images que nous avons enregistrées dans une base de données.

Les questions sont tirées de manières aléatoires et de manière qu'elle tombe qu'une seule fois. A la fin des 20 questions le joueur sera automatiquement redirigé vers la page de classement.

Chaque joueur aura une durée de 30 secondes pour répondre à chacune des questions, une barre de temps lui indiquera le temps qu'il lui reste. Si le joueur appuie sur la bonne réponse, le nombre de points correspondant lui seront ajoutés (départ à 100 points puis il perd 1 point toutes les 0.3 secondes). Si le joueur choisit la mauvaise réponse, aucun point ne lui sera ajouté, dans les deux cas le joueur devra attendre la fin du timer pour avoir d'afficher la bonne réponse pendant secondes et passer à la question suivante. Si a la fin de ses 30 secondes le joueur

n'a pas répondu, la bonne réponse lui sera affichée et aucun point ne lui sera ajouté. De plus, la bonne réponse sera affichée durant 3 secondes.

Pour la partie classement nous avons créé une page avec deux boutons permettant d'afficher soit le classement par groupe soit le classement général. Dans le classement par groupe on retrouve le numéro du groupe ainsi que son score et dans le classement général on retrouve les noms, prénoms, numéros de groupe ainsi que le score de tous les participants.

Pour que seuls les administrateurs aient accès aux résultats nous avons mis en place une page administrateur permettant d'accéder aux résultats pendant la partie.

#### IV. Planning previsionnel

séance 1 :

- Analyse du cahier des charges
- Liste des tâches à faire
- Répartition des tâches

séance 2 :

- Construction de la BDD
- Construction des pages index.php et accueil.php avec leurs CSS

séance 3 :

- Écriture des requêtes sql
- Suite construction des pages index.php et accueil.php avec leurs CSS

séance 4 :

- Ajout des requêtes sql sur les pages index.php et accueil.php
- Début construction des pages questions.php et classement.php avec les CSS

séance 5 :

- ID aléatoire
- Suite construction des pages questions.php
- Minuteur

séance 6 :

- ID aléatoire
- Vérification bonne réponse
- Pages classement groupe

séance 7 :

- Calcul du score
- Requêtes dans classement groupe

séance 8 :

- Derniers ajustements

## V. Planning réel

### séance 1 :

- Analyse du cahier des charges
- Liste des tâches à faire
- Répartition des tâches
- Charte graphique

### séance 2 :

- Construction de la BDD + écriture des requêtes sql
- Construction des pages index.php et question.php avec leurs CSS

### séance 3 :

- Construction des pages index.php et question.php avec leurs CSS
- Écriture du code PHP pour la page index.php et pour la page question.php pour permettre d'afficher la question en fonction d'un ID

### séance 4 :

- Code pour la page question.php pour permettre d'afficher la question en fonction d'un ID
- Construction des pages accueil.php et question.php avec leurs CSS

### séance 5 :

- Recherche comment avoir un ID aléatoire
- Construction des pages classement.php et question.php avec leurs CSS

### séance 6 :

- Test ID aléatoire
- Construction des pages classement.php et question.php avec leurs CSS
- Code PHP pour la page classement.php

### séance 7 :

- Vérification bon fonctionnement ID aléatoire
- Construction des pages classement.php et admin.php avec leurs CSS

### séance 8 :

- Construction de la page admin.php avec son CSS
- Sphinx

### En dehors des séances :

- Sphinx
- HTML/CSS
- Minuteur
- Vérification de la réponse
- Envoi score entre JS et php

### Répartition des tâches:

<https://github.com/users/fruixy/projects/3/views/5?groupedBy%5BcolumnId%5D=Assignees>

## VI. Principaux algorithmes

Le script `rng.php` permet de récupérer toutes les questions qui n'ont pas encore été tirées avant d'en sélectionner une aléatoirement et de l'ajouter dans la table intermédiaire avec l'ID du joueur correspondant.

Le fichier `calcule_points.js` décrémente de 1 point une variable toutes les 0.3 secondes.

`choix.js` permet de récupérer la réponse du joueur et de faire disparaître les boutons lorsque le joueur appuie sur un bouton ou que les secondes sont écoulées.

`comparaison_envoie.js` vérifie si le joueur a sélectionné la bonne réponse et transmet les points au script : `ajout_score.php`.

Le code `ajout_score.php` récupère le score du joueur envoyé par `comparaison_envoie.js` et le transmet à la base de données.

`jquery.min.js` est une bibliothèque nécessaire pour envoyer les valeurs de js vers php.

## VII. Conclusion

Pour ce projet nous avons développé un site web avec plusieurs langages : HTML/CSS/JS et PHP. Nous avons réussi à réaliser les tâches demandées mais nous avons rencontré de nombreuses difficultés : d'abord pour réussir à récupérer la réponse du joueur, vérifier, si c'est la bonne réponse, calculer le score et le transmettre à la BDD, pour cela il a fallu apprendre à coder en js. On a aussi rencontré des difficultés avec le nombre aléatoire. Le cahier des charges a été réalisé au complet dans la mesure du possible. Le cahier des charges était flou et contradictoire, au vu de notre niveau certaines tâches étaient impossible à réaliser comme les 20 minutes d'attente pour l'accès au classement à partir du début de l'épreuve. Pour plus d'information consulter le `README.rst` disponible sur la page d'accueil de notre github.

[https://github.com/fruixy/quiz\\_savant](https://github.com/fruixy/quiz_savant)