

Culture Quiz

Objectif du Projet

Développer une application web de quiz de culture générale nommée "Culture Quiz" en utilisant React. L'application doit être conçue en priorité pour les mobiles (mobile first). L'utilisateur pourra sélectionner une catégorie de questions et répondre à un quiz chronométré.

Fonctionnalités Requises

1. Page d'accueil :
 - Afficher le logo de l'application (génération par IA possible).
 - Afficher le titre "Culture Quiz".
2. Sélection de la catégorie :
 - Après avoir cliqué sur la page d'accueil, l'utilisateur doit pouvoir sélectionner une catégorie de quiz (exemples : Histoire, Cinéma, Sport, etc.).
3. Quiz :
 - Une fois la catégorie choisie, l'utilisateur commence le quiz.
 - Chaque question a un timer de 30 secondes affiché en haut.
 - Quatre propositions de réponses sont affichées pour chaque question parmi plusieurs réponses possibles (10 réponses par question mais 4 affichées dont la bonne réponse)
 - L'utilisateur doit cliquer sur une réponse avant la fin du timer.
 - Si l'utilisateur clique sur la bonne réponse, la réponse se colore en vert avant de passer à la question suivante.
 - Si l'utilisateur clique sur une mauvaise réponse, la réponse se colore en rouge avant de passer à la question suivante.
 - Si l'utilisateur ne clique sur aucune réponse, la question suivante s'affiche automatiquement après 30 secondes.
4. Résultats :
 - À la fin des 10 questions, le score du joueur est affiché.
5. API :
 - Les catégories, questions et réponses doivent être récupérées depuis une API que vous mettez en place avant le projet.

Barème:

- L'application est correctement réalisée en mobile first (**2pts**)
- La page d'accueil présente bien un logo et le titre de l'application (**1pt**)
- Les catégories sont correctement affichées et proviennent de l'API (**2pts**)
- La question et les propositions sont bien affichées et proviennent de l'API (**3pts**)
- Le timer est correctement implémenté et passe à la question suivante si pas de réponse (**2pts**)
- La coloration de la question avant le changement de page se fait correctement (**1pt**)
- Le score est correctement calculé et affiché une fois les 10 questions passées (**2pts**)
- Le code est facilement maintenable, les pages et composants utilisés sont pertinents (**2pts**)
- Le design est cohérent au sein de l'application (charte graphique, polices..) (**1pt**)
- Présentation orale (**4pts**)

Rendu:

Pour le rendu du projet il me faudra un lien github sur lequel retrouver votre projet avec également un export de votre base de données.

Il faudra aussi me dire comment vous avez réparti les missions de développement de ce projet au sein de votre groupe.

Vous pourrez alors me communiquer ceci par mail à l'adresse:

theau.debureau.pro@gmail.com

Lors de la dernière séance il y aura une présentation orale d'environ 20 mn de votre projet.

Dans celle-ci il vous faudra au minimum présenter les points suivants:

- explication de votre choix graphique
- les endpoints de votre api et pourquoi vous avez choisi tel langage pour la mettre en place
- le stockage utilisé pour les données de votre api et ses particularités
- l'architecture de votre application (pages, composants, routing..)
- une partie de votre code au choix
- les difficultés rencontrées et comment vous les avez surmontées

Suite à cela vous pourrez alors lancer votre application web afin de montrer le rendu final de ce que vous avez développé.