

Documentation Projet Geoquiz :

1) Présentation rapide

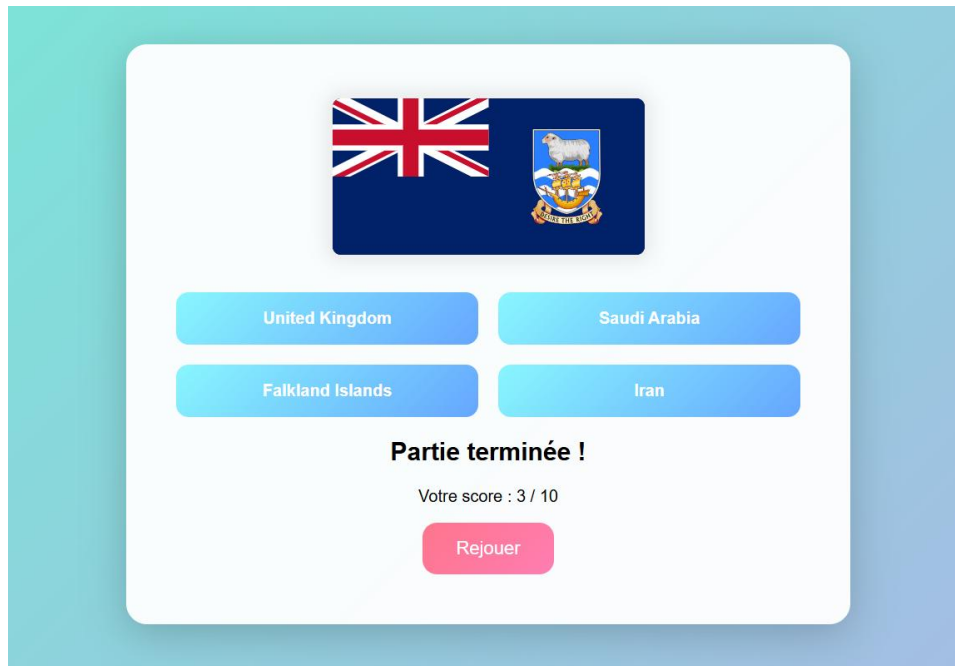
- Nom du projet : GeoQuiz
- Objectif : Tester les connaissances géographiques des utilisateurs via un quiz interactif de 10 manches portant sur les drapeaux du monde.
- Contexte : Mini-projet de développement JavaScript / Application de quiz dynamique.

2) Fonctionnalités

L'application propose une expérience de jeu complète et automatisée :

- Génération dynamique : À chaque manche, un drapeau est tiré au sort parmi plus de 200 pays.
- Système de QCM : Génération de 4 propositions (la bonne réponse + 3 pays aléatoires) pour chaque question.
- Feedback instantané : Validation visuelle de la réponse (Bonne/Mauvaise) avec un message explicatif.
- Gestion de session : Déroulement sur 10 manches avec calcul du score final et possibilité de rejouer.
- Anti-triche : Désactivation des boutons après le premier clic pour éviter les réponses multiples.





3) Partie technique simplifiée

- Langages utilisés : HTML5, CSS3, JavaScript (ES6+).
- API externe : Utilisation de RestCountries API pour récupérer les données mondiales (noms des pays et URLs des drapeaux en SVG).
- Mécanique JS : * Utilisation de fetch et async/await pour la communication avec l'API.
 - Utilisation de Set() pour garantir que les 4 options de réponse soient toutes uniques.
 - Utilisation de setTimeout() pour créer une transition fluide entre les manches.

4) Difficultés / Ce que j'ai appris

Ce projet a été l'occasion de résoudre des problèmes de logique de programmation intéressants :

- Consommation de données asynchrones : J'ai appris à charger toutes les données de l'API dès le départ (fetchCountries) pour que le jeu soit instantané et fluide une fois lancé.

- Algorithme de sélection aléatoire : J'ai dû concevoir une logique pour choisir une réponse correcte et s'assurer que les trois autres choix ("distracteurs") soient différents de la réponse et uniques entre eux.
- Gestion de l'état du jeu : J'ai appris à gérer le passage des manches, l'incrémentation du score et l'affichage conditionnel des éléments (masquer le jeu pour afficher le score final).
- Expérience Utilisateur (UX) : J'ai implémenté des délais de transition et des retours visuels pour que le joueur comprenne ses erreurs avant de passer automatiquement à la suite.