





# Projets 2025

I.	Système de quiz en ligne avec suivi des résultats 	2
II.	Jeu de Memory 	3
III.	Jeu de course en vue de dessus (niveau supérieur) 	4
IV.	Jeu de Simon 	5

## I. Système de quiz en ligne avec suivi des résultats

### **Description :**

L'application permet à un administrateur de créer des quiz sur divers thèmes, et aux utilisateurs de s'inscrire pour les passer, en obtenant un score en temps réel et en se positionnant dans un classement général.

### **Fonctionnalités Front-End :**

- **Création et gestion des quiz :**
  - Interface d'administration pour ajouter, modifier et supprimer des quiz et leurs questions.
  - Possibilité de définir différents types de questions (QCM, vrai/faux, questions ouvertes avec réponses préformatées).
- **Interface utilisateur pour le quiz :**
  - Affichage dynamique des questions avec une barre de progression indiquant le nombre de questions restantes.
  - Validation instantanée des réponses avec feedback visuel (bon/mauvais, explications facultatives) 👍👎.
  - Chronomètre pour imposer une contrainte de temps sur la réponse et augmenter le challenge ⌚.
- **Gestion de l'inscription et de l'authentification :**
  - Formulaires clairs pour l'inscription et la connexion avec des vérifications côté client (JavaScript).
  - Affichage personnalisé du profil utilisateur et historique des quiz passés.

### **Intégration Back-End :**






- Utilisation de PHP pour traiter les formulaires d'inscription, la validation des réponses et le calcul des scores.
- Stockage des données (quiz, questions, réponses, scores, historique) dans une base de données MySQL.
- Création d'un module d'administration sécurisé pour la gestion des contenus et des classements 🔒.

## II. Jeu de Memory


### Description :

Un jeu de mémoire où le joueur doit retrouver des paires de cartes identiques dans une grille, avec un système de score basé sur le temps et le nombre de coups.

### Fonctionnalités Front-End :

- **Génération dynamique et affichage des cartes :**
  - Algorithme de mélange aléatoire pour disposer les cartes différemment à chaque partie .
  - Animation fluide (effet de flip) lors du retournement des cartes, réalisée via CSS et JavaScript.
- **Interface de jeu :**
  - Mise en place d'un chronomètre et d'un compteur de coups pour évaluer la performance du joueur  .
  - Indicateurs visuels pour signaler une paire trouvée et progression dans le jeu.
  - Option de choix du niveau de difficulté (nombre de cartes et temps imparti) pour varier les challenges.
- **Feedback utilisateur :**
  - Messages de félicitations, encouragements ou suggestions en cas d'erreur .
  - Bouton de redémarrage pour une nouvelle partie .

### Intégration Back-End :

- PHP pour gérer l'enregistrement et la récupération des meilleurs scores et de l'historique des parties.
- Base de données MySQL pour stocker les statistiques de chaque joueur, permettant ainsi d'afficher un classement en ligne et des statistiques de progression .

### III. Jeu de course en vue de dessus (niveau supérieur)



#### Description :

Concevoir un jeu de course dynamique où le joueur pilote un véhicule sur des circuits en vue de dessus, avec des obstacles à éviter et un système de chronométrage pour mesurer les performances.

#### Fonctionnalités Front-End :

- **Contrôle du véhicule :**
  - Gestion des déplacements (accélération, freinage, direction) à l'aide du clavier, avec une réactivité en temps réel en JavaScript.
  - Détection des collisions et gestion des interactions avec les obstacles 🚧.
- **Conception des circuits :**
  - Création de circuits variés, soit préétablis, soit générés aléatoirement, offrant des défis uniques à chaque partie.
  - Utilisation de l'élément HTML5 Canvas pour dessiner les pistes et animer le mouvement du véhicule.
- **Interface et effets visuels :**
  - Affichage d'un chronomètre précis pour mesurer le temps mis sur le circuit 🕒.
  - Effets visuels (animations de vitesse, collisions, trajectoires) et sonores (bruit du moteur, musique de fond) pour enrichir l'expérience immersive 🎵.
- **Gestion de la difficulté :**
  - Progression graduelle de la difficulté (augmentation de la vitesse, obstacles plus complexes) au fur et à mesure des niveaux 📈.

#### Intégration Back-End :





- Utilisation de PHP pour enregistrer les temps de course et gérer un classement en ligne.
- Stockage des scores et des statistiques de chaque course dans une base de données MySQL, avec une interface pour consulter l'historique des performances.

## IV. Jeu de Simon


### 1. **Description :**

Créer un jeu de mémoire séquentielle dans lequel le joueur doit reproduire une séquence de lumières et de sons qui s'allonge progressivement, testant ainsi sa capacité de mémorisation et sa rapidité.

### **Fonctionnalités Front-End :**

- **Génération de la séquence :**
  - Mise en place d'un algorithme JavaScript pour générer aléatoirement une séquence de couleurs et de sons à reproduire.
  - Affichage séquentiel de la séquence avec des délais prédéfinis entre chaque signal .
- **Interface interactive :**
  - Boutons lumineux correspondant aux différentes couleurs, réagissant aux clics du joueur avec des animations et effets sonores synchronisés .
  - Indicateur visuel du niveau en cours et du score basé sur la longueur de la séquence réussie.
- **Gestion du temps et des erreurs :**
  - Implémentation d'un timer pour mesurer le temps de réaction du joueur lors de la saisie de la séquence.
  - Affichage d'un message d'erreur dès qu'une séquence incorrecte est saisie, avec la possibilité de recommencer  .
- **Progression et feedback :**
  - Augmentation progressive de la difficulté en allongeant la séquence à chaque tour réussi.
  - Bouton de démarrage et de réinitialisation pour permettre une nouvelle partie

### **Intégration Back-End :**

- Utilisation de PHP pour enregistrer les scores et le niveau atteint par chaque joueur.
- Stockage des données dans une base de données MySQL afin d'afficher un classement global et de suivre l'évolution des performances individuelles au fil du temps .