

Projet PHP



Boris CERATI

Année scolaire 2021 / 2022

Pensez à la planète avant d'imprimer ce document

Jeu

Introduction

Le but de ce projet est le développement d'un jeu de plateau multijoueurs dans lequel le joueur répondra à des questions afin d'avancer en cas de bonne réponse ou de reculer en cas de mauvaise réponse.

Au moment de jouer, le joueur pourra choisir la difficulté de sa question.

- niveau 1 : vert ;
- niveau 2 : jaune ;
- niveau 3 : bleu ;
- niveau 4 : orange ;
- niveau 5 : rouge ;
- niveau 6 : noir.

Lorsque le niveau est choisi, une question apparaît à l'écran dont la difficulté dépend de la couleur choisie. Vert étant la plus simple, noir la plus compliquée.

L'utilisateur devra répondre oralement à la question. Si la réponse est bonne, il avance d'autant de case que la difficulté (1 case pour vert, 6 case pour noir).

Le premier joueur à arriver sur la case finale remporte la partie. Les autres joueurs continuent jusqu'à ce qu'il y ait un perdant de la partie. Le nombre de cases est fixé à 48.

Le jeu pourra être joué de deux à six joueurs. Un compte sur la plateforme est nécessaire pour démarrer une partie. Au moment de jouer, tous les joueurs sont sur leurs propres terminaux (smartphone, tablette ou pc).

Le premier joueur initie la partie en renseignant les pseudos des autres joueurs. Une fois le formulaire de démarrage de partie envoyé, un QR code est généré pour chaque joueur. S'ils sont dans la même pièce, ils pourront scanner le QR Code afin d'arriver sur la page de la partie. Dans le cas contraire, le joueur maître pourra leur envoyer le lien en cliquant sur un bouton "Envoyer le lien par mail". Vu que le joueur est déjà inscrit sur la plateforme, le lien lui sera envoyé par mail.

Organisation

Le projet se fera en deux étapes :

- **Étape 1** : la gestion des utilisateurs, questions et réponses ;
- **Étape 2** : développement de la partie jeu en PHP/JavaScript.

Dans le cadre du cours de PHP, programmation serveur, **seule l'étape 1 sera développée**.

Ce projet est à développer par groupes de deux étudiants et le rendu final du projet devra être effectué au plus tard le **vendredi 24/12** sous la forme d'un fichier ZIP (**sans le dossier *vendor***) déposé sur l'ENT.

Technologies utilisées

Pour ce projet la technologie PHP est obligatoire, sans utilisation de framework *fullstack* existant.

Les technologies suivantes devront être utilisées :

- **PHP** : pour le développement de tout le projet et l'interface avec la base de données MySQL. La version de PHP à utiliser sera la version 8.0 ;
- **JavaScript** : le JavaScript peut être utilisé sur l'interface du backend afin d'améliorer l'expérience utilisateur
- **Composer** : pour la gestion de vos dépendances PHP ;
- **POO** : toute la partie backend devra être orientée objet au maximum ;
- **MySQL** : la base de données sera gérée à l'aide du SGBD MySQL.

Il n'y a aucune restriction sur les bibliothèques utilisées, vous pourrez utiliser par exemple :

- **Fastroute** : pour la gestion de votre routing ;
- **Bootstrap** : pour la partie interface graphique ;
- **Doctrine ou PDO** : pour l'interface avec la base de données ;
- **Twig** : comme moteur de templates ;
- etc.

N'hésitez pas à utiliser au maximum les bibliothèques open source de la communauté.

Consignes

Entités à créer

Le projet nécessitera plusieurs entités :

- **User** : pour la gestion des utilisateurs (inscription/connexion)
- **Question** : pour l'affiche des questions ;
- **Réponse** : pour le potentiel affichage des réponses possibles. Si la question ne possède qu'une réponse, la réponse ne devra pas être affichée à l'utilisateur et il devra la deviner.

Voici les informations minimales nécessaires des entités

User

- id : int
- username : string (unique en base de données)
- password : string
- email : string (unique en base de données)
- roles : string[] (contient les rôles de l'utilisateur (ROLE_ADMIN et/ou ROLE_USER))
- firstName : string
- lastName : string
- createdAt : Datetime (date d'inscription/première connexion)

Question

- id : int
- label : string
- level : int (entre 1 et 6, représente la difficulté)
- answers : Answer[]

Answer

- id : int
- label : string
- question : Question
- valid : bool (true si la réponse à la question est la bonne)

Étapes conseillées pour le développement

Étape 1 : mise en place du projet

Dans cette étape, il s'agirait d'instaurer les bases du projet. C'est-à-dire :

- structure des dossiers ;
- mise en place de la logique du routing ;
- mise en place de la connexion à la base de données ;
- mise en place de composer ;
- création des entités ;
- etc.

Étape 2 : inscription des utilisateurs

Dans cette étape, il s'agirait de développer l'inscription d'un utilisateur. Une fois inscrit l'utilisateur est automatiquement connecté et redirigé vers la bonne page (cf étape 4)

Étape 3 : connexion des utilisateurs

Dans cette étape, il s'agirait de développer la connexion d'un utilisateur qui s'est précédemment inscrit sur la plateforme. L'utilisateur devra renseigner son "username" ainsi que son mot de passe pour se connecter.

Étape 4 : accès à l'administration

Une fois l'utilisateur connecté il faut le rediriger vers une page :

- / : si l'utilisateur ne possède pas le rôle ROLE_ADMIN on le redirige vers la page d'accueil afin qu'il puisse jouer
- /admin : si l'utilisateur possède le rôle ROLE_ADMIN on le redirige vers la page d'administration où nous affichons deux blocs :
 - un bloc affichant le nombre d'utilisateurs avec un lien menant à la page de gestion des utilisateurs ;
 - un bloc affichant le nombre de questions avec un lien menant à la page de gestion des questions.

Étape 5 : accès à la liste des utilisateurs

Créer la page "/users". Cette page affichera un tableau avec la liste de tous les utilisateurs et deux boutons permettant de modifier et supprimer l'utilisateur.

Étape 6 : accès à la gestion des utilisateurs

Créer les deux actions permettant de gérer les utilisateurs :

- **modification d'un utilisateur** : permet de modifier le nom, prénom, email et rôles de l'utilisateur ;
- **suppression d'un utilisateur** : dans le tableau de gestion des utilisateurs, un lien permettant la suppression d'un utilisateur doit être créé.

Étape 7 : accès à la liste des questions

Créer la page “/questions” affichant notamment un tableau avec la liste de toutes les questions ainsi que deux boutons permettant de modifier et supprimer la question.

Étape 8 : accès à la gestion des questions

Créer les deux actions permettant de gérer les questions :

- **modification d'une question** : permet de modifier toutes les informations relatives à la question y compris la ou les réponses possibles ;
- **suppression d'une question** : dans le tableau de gestion des questions, un lien permettant la suppression d'une question doit être créé.