

# QUIZ PANIC **EVOLVED**



Matthieu PETIT

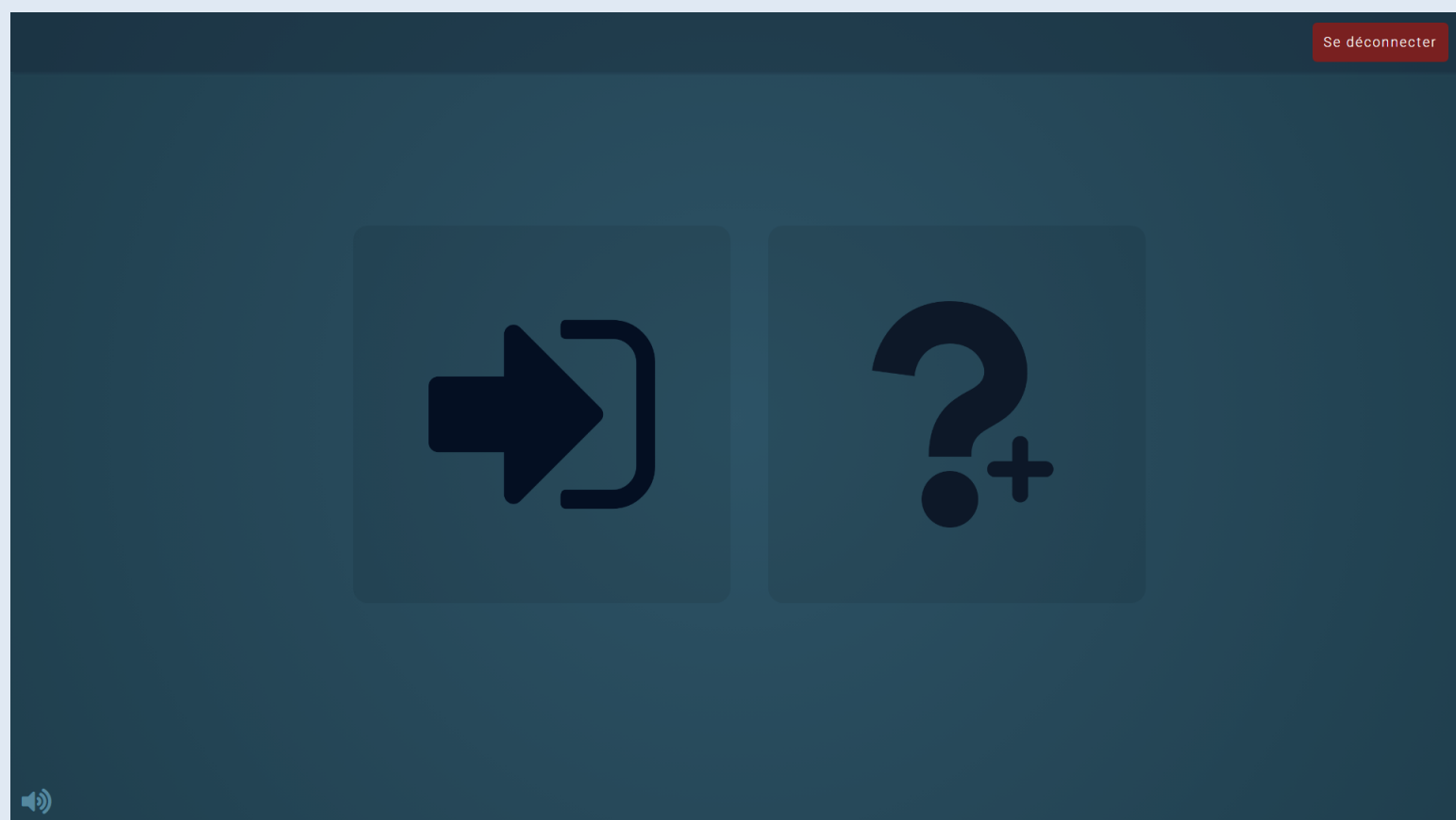


## I. Connexion

Sur la page d'accueil, l'utilisateur peut choisir de s'identifier ou de s'inscrire en cliquant sur le bouton en haut à droite.

La connexion se fait en AJAX, tout comme la vérification de l'existence du pseudo lors de l'inscription.

## II. Accueil



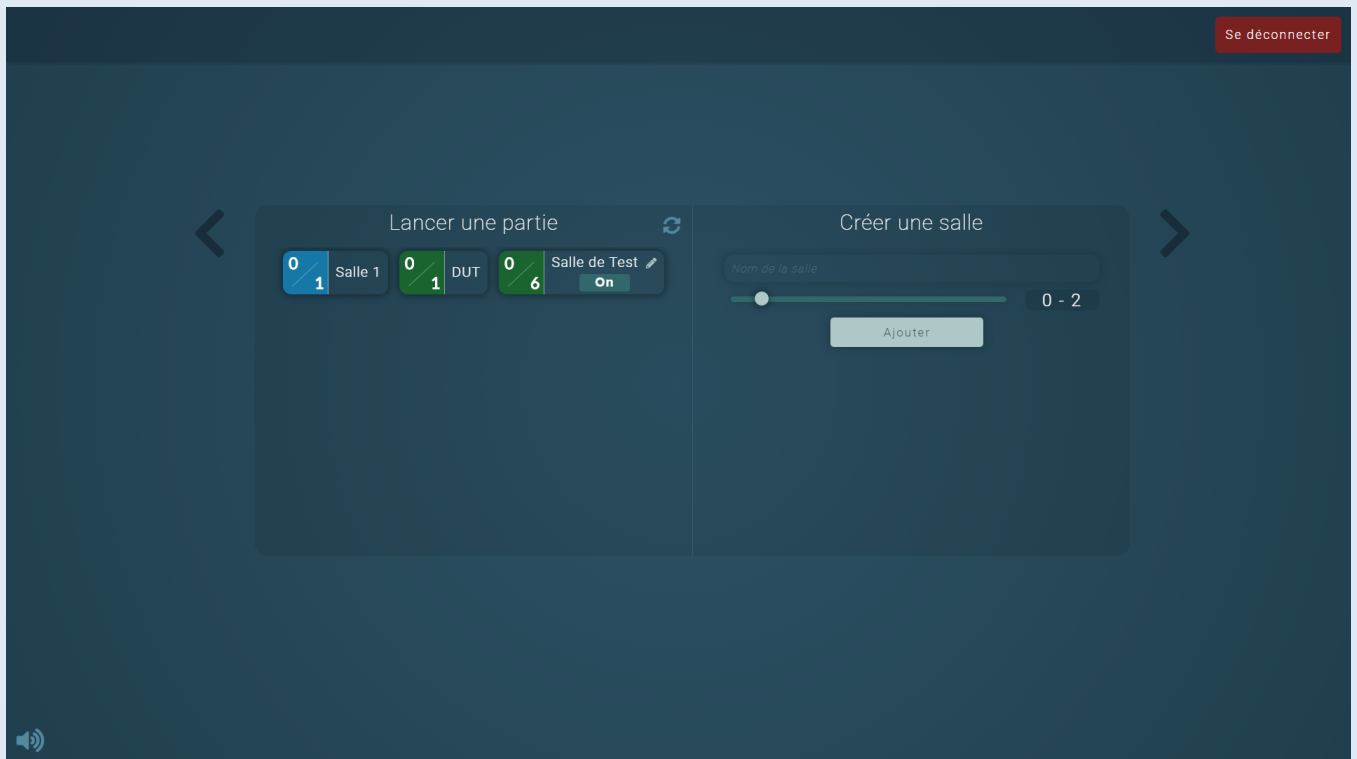
Une fois connecté, on accède à l'écran d'accueil où l'on peut choisir de gérer / rejoindre une salle, ou de gérer / ajouter une question.

On peut aussi faire clic droit pour naviguer dans le menu.

L'icone de son en bas à gauche sert à désactiver tous les sons de notifications.

Lorsque l'utilisateur est inactif depuis trop longtemps, un pop-up apparaît pendant 1 min pour lui rappeler qu'il sera déconnecté automatiquement.

# III. Salles



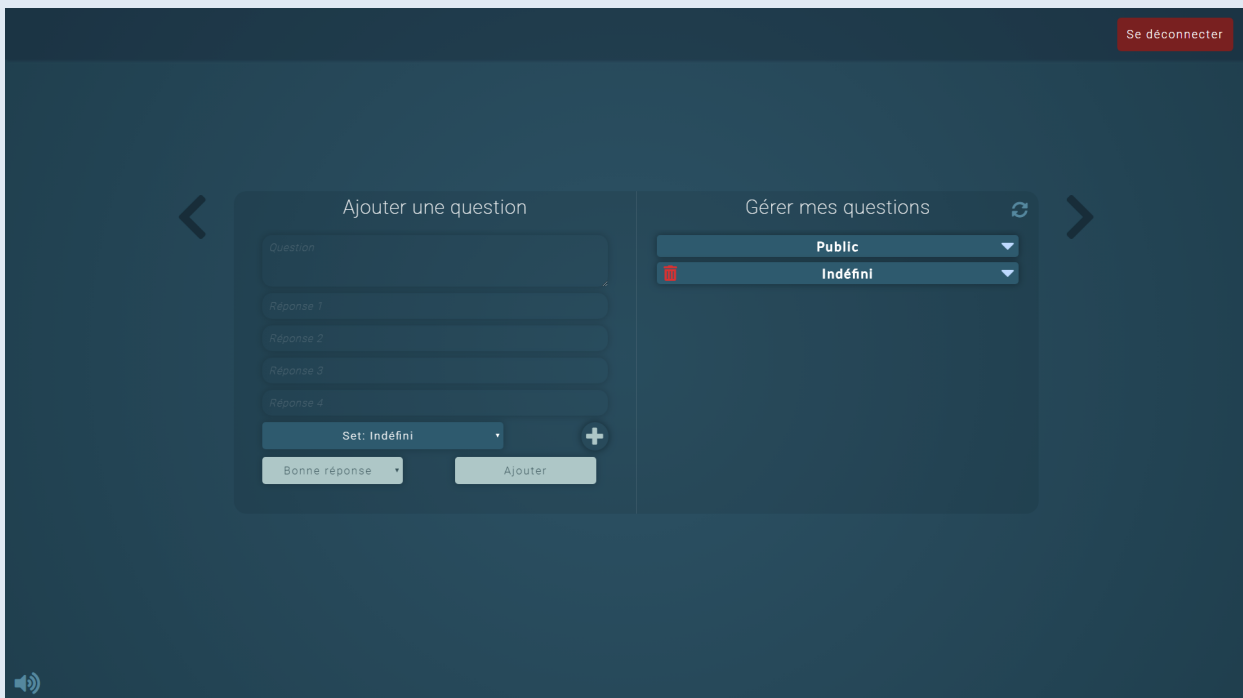
La liste des salles ainsi que le formulaire de création de salle sont disponibles.

Lorsque la salle appartient au joueur, il a la possibilité de la modifier, la supprimer et de la désactiver, ce qui entraîne le kick de tous les utilisateurs présents dans la salle, et elle ne sera plus visible par les autres joueurs.

Sur la gauche s'affiche les salles disponibles, avec le nombre de joueurs. Si un admin est présent dans la salle, elle est indiquée en bleu sur l'écran.

Lorsque l'user est le premier à rejoindre une salle, il peut choisir le nombre de questions. Et lorsque la salle lui appartient, il a la possibilité de se connecter en tant qu'admin et peut ainsi choisir le nombre de questions, le temps imparti pour répondre, ainsi que le set de questions (par défaut le set Public).

# IV. Questions



Lorsque l'utilisateur créé une question, il peut choisir le set dans laquelle est elle ajoutée. Sinon, il peut en créer un nouveau. Il existe un set 'Public' qui est commun à tous les utilisateurs.

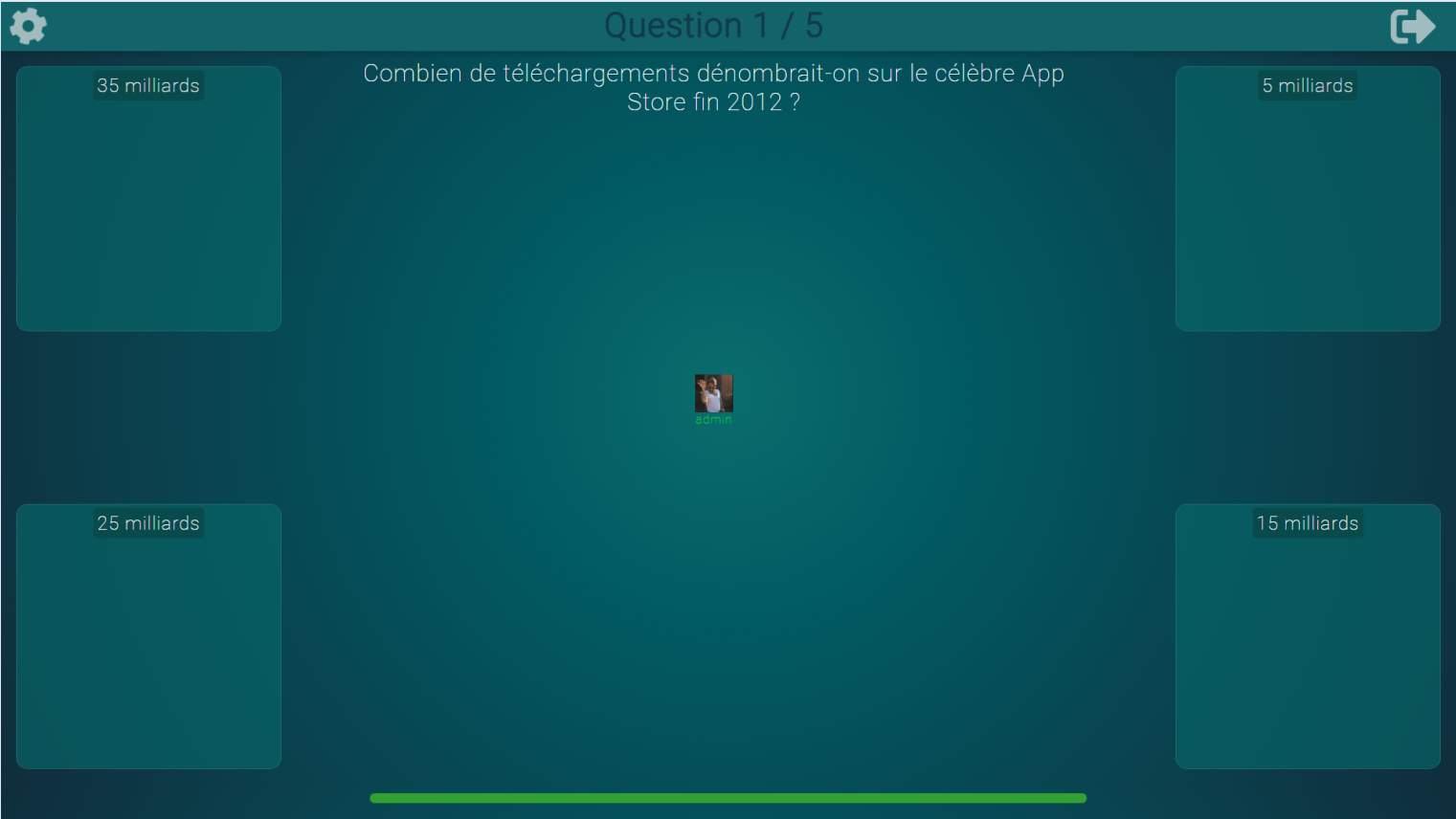
On peut supprimer un set, modifier / déplacer une question.

## V. Jouer



Voici l'écran de jeu. L'utilisateur peut voir le nombre de joueurs et leur avatar. En cliquant en haut à droite, l'utilisateur quitte la salle.

En haut à gauche, on peut accéder au menu des paramètres. On peut choisir notre avatar, et communiquer avec les autres utilisateurs de la salle. Ce chat est réinitialisé à chaque partie.



Une fois que la salle est remplie, la partie se lance. Les réponses puis la question s'affichent, on peut voir qui choisit quelle réponse. Les scores s'affichent entre chaque cycle.

Lorsque la partie se termine, on affiche les scores puis tout le monde est kické au bout de quelques secondes.

# Technique

La totalité des échanges avec le serveur sont effectués avec AJAX, j'ai aussi utilisé jQuery pour la quasi-totalité du javascript.

J'ai eu beaucoup de mal à trouver une technique efficace pour le déroulement d'une partie. Au début, tous les utilisateurs envoyaient une requête au serveur avec leur score, leur position etc.. chaque seconde, ce qui était ingérable côté serveur.

C'est maintenant le premier joueur de la salle qui 'héberge' la partie et s'occupe de générer aléatoirement un ensemble de questions, et c'est aussi ce joueur qui attribue les scores. Ainsi, si les joueurs ne sont pas synchronisés avec les autres joueurs, il ne peuvent pas tricher en étant en avance ou en retard. Si un joueur s'est connecté en administrateur, c'est lui qui héberge la partie.

Ce système n'est pas parfait, mais a l'air de fonctionner et je n'ai pas trouvé mieux.

Pendant tout le projet j'ai fait en sorte que le site soit fonctionnel sur dwarves aussi, mais cela a créé beaucoup de problèmes d'encodages pour la base de données et le JSON.

J'ai essayé de sécuriser un maximum le site contre le XSS, les injections, les permissions etc.. Mais le côté jeu n'est pas totalement protégé, les requêtes sont en AJAX et le joueur peut facilement en envoyer des fausses.

# Idées

La date du projet ayant été avancée, je n'ai malheureusement pas eu le temps de tout implémenter.

Par exemple:

- Ajouter un système de tokens pour la sécurité.
- Créer des questions spéciales et un système de classement/temps comme dans le vrai jeu.
- Améliorer la compatibilité pour mobile, faire une version lite du site, sans animations etc.
- Ajouter des fonctionnalités de personnalisation, couleur dans le chat, argent virtuel pour acheter des tenues ou des powers-up.
- Ajouter un système de timestamp pour kicker les joueurs qui ont fermé leur navigateur sans quitter la salle, ainsi le kick est automatique. (Mais surement lourd à gérer côté serveur..)
- Trouver un moyen de ne pas faire de requêtes chaque seconde; que tout les joueurs soient synchros tout le temps, que le serveur informe le joueur sans requête de sa part.

**\* Après quelques recherches le jour du rendu, je viens de découvrir Node.js qui permet apparemment de faire tout ça...**

CONCLUSION: J'ai vraiment appris beaucoup de choses avec ce projet, j'ai aussi été bloqué plusieurs fois sur des détails mais j'ai au final implémenté tout ce que je voulais; je suis donc satisfait dans l'ensemble, et heureusement car ça m'a coûté de précieuses heures de sommeil :(

DÉTAILS: Le développement s'est fait principalement sur Chrome, il peut donc y avoir quelques soucis au niveau du CSS sur Firefox ou autre. Le serveur dwarves plante quasi tout le temps à cause des trop nombreuses requêtes.