



TEST FULLSTACK

Nous sommes au Ceasar Palace casino à Las Vegas, assis à une table spéciale.

But du jeu : piocher le plus de paires.

Nombre de joueurs : 2

Matériel : 5 paires de cartes random

Exemple :

10 de ♠

Valets de ♠

Reine de ♠

Roi de ♠

As de ♠

10 de ♥

Valets de ♥

Reine de ♥

Roi de ♥

As de ♥

●Règles du jeu :

Il s'agit de former des paires de cartes selon le principe suivant :

Sur une table, il y aura les 10 cartes disposer aléatoirement faces cachées. Il faudra piocher 2 cartes aux choix sur la table en essayant de former une paire.

- Compte des cartes:

La partie se joue en 10points

Chaque paire piocher par le joueur vaut 2 points.

exemple:

Piocher reine de ♥ et 10 de ♠ , aucun point ne sera compté.

Piocher reine de ♥ et reine de ♥, 2 points accordés au joueur.

Un minimum de 3 paires piochées par un joueur veut dire que la partie est remporté par ce dernier.

- Indications:

Il y aura, en début de partie, un 3..2..1..go

La partie se joue en 1min avec deux décomptes timers généraux pour chacun des deux joueurs.

Un tour dure 5 seconde par joueur. Ce dernier est synchronisé au timer général du joueur ayant le tour.

Lorsqu'un joueur dépasse ses 5 secondes, il passe son tour au second joueur.



Lorsque le joueur pioche les deux cartes et qu'elles ne forment pas la paire, les cartes sont retourner sur la tables et mélangées random.

Puis le tour et le décompte timer du second joueur s'active.

Le reste des secondes après la mauvaise pioche sont cumulables au timer générale du joueur ayant le tour.

A chaque pioche des deux cartes par un joueur, ces dernières se mélangent random en faces cachées sur la table.

Si le joueur ayant le tour pioche une paire, il peut continuer le jeu pendant les secondes restante.(Décompte de 5sec)

A la fin du décompte timer général d'un joueur, la partie est remportée par l'opposant.

Cas d'exception: le compte des cartes.

(Si un joueur détient plus de cartes que son opposant et que son timer général se termine, il remporte tout de même la partie)

Un chat devra être intégrer pour permettre un échange écrit entre les deux joueurs.

Il faudra aussi stocker le nombres de points pour chaque joueur, ainsi que leur date time.

Un compteur de victoires / défaites devra être aussi affiché et stocké.

Des animations devront être faites pour développer l'entièreté de ce mini jeu.

●**Technique:**

-Setting up a MERN stack pour le développement.

-Utiliser Redux et SASS.

-Vous avez le choix: material-ui , styled-components ou autre.

-Containerizing l'application en utilisant Docker.

-Le Code source doit être pushé sur GitHub attaché avec un Readme et une vidéo de démonstration du fonctionnel

- Votre code sera évalué sur la base de : la structure du code, les meilleures pratiques de programmation, la lisibilité.

- Mettez l'accent sur la qualité du code. Ne piratez pas quelque chose rapidement, prenez votre temps et créez quelque chose de propre.

- Nous sommes à la recherche de quelque chose de réactif et créatif.