

Rapport de Projet League Of Stones

Licence MIASHS
Parcours Informatique

Phuong NGUYEN , Duy-Anh NGUYEN, Thomas ABADIE , Ahmed NSIBI

url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web

SOMMAIRE

I – Présentation Générale

- 1.** Introduction
- 2.** Contexte
- 3.** Problématique
- 4.** Description de la plateforme
- 5.** Objectifs du projet

II – Conduite du Projet

- 1.** Ressources
- 2.** Partage des tâches

III – Analyse

- 1.** Cahier des charges
- 2.** Choix retenus

IV – Réalisation du jeux

- 1.** Conception du jeu
- 2.** Prototypage
- 3.** Développement du jeu
- 4.** Tests et itérations

V – Difficultés rencontrées

- 1.** Évolutions du cahier des charges
- 2.** Difficultés techniques

VI – Conclusion

- 1.** Tests et mise en place
- 2.** Évolutions du projet
- 3.** Apports personnels

VII – Annexes

I/ Présentation générale

1. Introduction

Ce projet devait être réalisé en groupe. Notre groupe était constitué de 4 personnes issues de différentes formations, afin que chacune d'entre elles puisse apporter son savoir-faire dans le domaine qui lui avait été confié :

- Duy Anh NGUYEN : Gestion de projet et Développer du Back-End
- Phuong NGUYEN : Designer et Développer du Front-End
- Thomas ABADIE : Responsable qualité et Développer du Back-End
- Ahmed NSIBI : Architecte

Dans ce rapport, nous présenterons en détail les différentes étapes de développement du jeu, en commençant par la création de la partie utilisateur, suivie de la mise en relation des joueurs et enfin la gestion d'un match. Nous aborderons également les responsabilités spécifiques de chaque membre de l'équipe de développement, telles que la gestion de projet, la qualité du code, le design, l'architecture et le développement de la version mobile.

Le rapport comprendra également une analyse approfondie des fonctionnalités implémentées, des choix technologiques et des défis rencontrés tout au long du projet. De plus, nous fournirons le code source de l'application développée, ainsi que les différentes sections rédigées par chaque responsable dans le cadre de leur contribution au projet.

Nous espérons que ce rapport sera une représentation complète et détaillée de notre travail, mettant en évidence nos compétences en développement web, notre collaboration en équipe et notre capacité à relever les défis techniques.

2. Contexte

Le projet "League of Stones" a été réalisé dans le cadre d'un projet universitaire à l'Université Toulouse 2 - Jean Jaurès. L'objectif était de développer une application web basée sur un mashup des jeux vidéo Hearthstone et League of Legends.

Hearthstone, développé par Blizzard™, est un jeu de cartes en ligne qui met en scène des affrontements entre joueurs utilisant des decks de cartes. League of Legends, développé par Riot Games™, est un jeu en ligne très populaire qui propose des affrontements entre équipes de champions.

Le concept du projet "League of Stones" consistait à fusionner ces deux jeux pour créer un jeu de cartes stratégique. Les données ouvertes du jeu League of Legends ont été utilisées pour créer les cartes représentant les champions. Les joueurs devaient constituer un deck de 20 cartes et tenter de réduire les points de vie de l'adversaire à zéro en utilisant des attaques et des défenses stratégiques.

Le développement de l'application a été réalisé en utilisant des requêtes AJAX pour interroger les Web Services disponibles. Le framework React a été utilisé pour créer une interface utilisateur réactive et conviviale. L'objectif était de développer une application web fonctionnelle et responsive, accessible à la fois sur ordinateur et sur smartphone.

3. Problématique

La réalisation du projet "League of Stones" a soulevé plusieurs questions clés :

- 1.**Comment fusionner efficacement les données de League of Legends et les mécaniques de jeu de Hearthstone ?
- 2.**Comment concevoir une interface utilisateur réactive pour une expérience de jeu optimale sur différentes plateformes ?
- 3.**Comment gérer la mise en relation des joueurs et les parties de manière fiable et sécurisée ?
- 4.**Comment assurer la qualité du code et maintenir une architecture évolutive ?
- 5.**Comment surmonter les défis techniques liés à l'intégration de technologies telles que React, AJAX et les Web Services ?

4. Description de la plateforme

La plateforme "League of Stones" est une application web réactive développée avec React. Elle permet aux joueurs de s'affronter dans un jeu de cartes stratégique utilisant des champions de League of Legends. La plateforme propose un système de matchmaking pour trouver des adversaires de niveau similaire et garantit une expérience de jeu optimale sur différents appareils.

5. Objectifs du projet

Le projet "League of Stones" avait pour objectifs principaux :

- 1.** Développer une application web fonctionnelle et réactive permettant aux joueurs de s'affronter dans un jeu de cartes stratégique basé sur les champions de League of Legends.
- 2.** Intégrer de manière efficace les données ouvertes du jeu League of Legends dans le système de jeu de Hearthstone pour créer une expérience de jeu cohérente et immersive.
- 3.** Concevoir une interface utilisateur conviviale et adaptative, offrant une expérience de jeu fluide et intuitive sur différentes plateformes, notamment sur ordinateur et sur smartphone.
- 4.** Mettre en place un système de matchmaking fiable et équilibré, permettant aux joueurs de trouver des adversaires de niveau similaire et de participer à des parties compétitives.
- 5.** Assurer la qualité du code développé, en suivant les meilleures pratiques de développement web et en maintenant une architecture modulaire et évolutive.
- 6.** Surmonter les défis techniques liés à l'intégration des technologies telles que React, AJAX et les Web Services, en garantissant la performance, la sécurité et la stabilité de l'application.

II/ Conduite du projet

1. Ressources

- Duy Anh NGUYEN : Gestion de projet
- Phuong NGUYEN : Designer
- Thomas ABADIE : Responsable qualité
- Ahmed NSIBI : Architecte

2. Partage des tâches

1. Développement de l'application :

Ahmed NSIBI était responsable du développement de la logique de jeu, de la gestion des cartes et des mécanismes de jeu.

Duy Anh NGUYEN était chargé du développement de l'interface utilisateur, en utilisant le framework React pour créer les composants et les interactions.

2. Conception et design :

Phuong NGUYEN était responsable de la conception de l'interface utilisateur, en créant des maquettes et des prototypes pour définir l'apparence et l'expérience utilisateur.

3. Gestion de projet :

Duy Anh NGUYEN était responsable de la gestion de projet, en planifiant les tâches, en suivant les délais et en assurant la coordination de l'équipe.

4. Assurance qualité :

Thomas ABADIE était responsable de l'assurance qualité, en effectuant des tests réguliers, en vérifiant la conformité du code aux normes de qualité et en assurant la stabilité de l'application.

III/ Analyse

1. Cahier des charges

Objectifs

- Créer une expérience de jeu immersive et divertissante pour les joueurs.
- Intégrer les données de League of Legends dans le système de jeu de cartes.
- Développer une interface utilisateur réactive et conviviale.
- Mettre en place un système de matchmaking équilibré.
- Assurer la qualité du code et maintenir une architecture évolutive.

Fonctionnalités

1. Jeu de cartes stratégique

- Les joueurs peuvent sélectionner et jouer des cartes représentant les champions de League of Legends.
- Les cartes ont des statistiques, des compétences et des capacités spécifiques.
- Les joueurs utilisent leurs cartes pour attaquer, défendre et gagner des points de victoire.
- Le jeu se déroule en temps réel avec des tours alternés entre les joueurs.

2. Intégration des données de League of Legends

- Les données des champions de League of Legends sont utilisées pour créer les cartes virtuelles.
- Les statistiques et les compétences des champions sont reflétées dans le jeu de cartes.

3. Interface utilisateur réactive

url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web

- L'interface utilisateur s'adapte de manière fluide aux différents types d'appareils (ordinateurs, tablettes, smartphones).
- Les éléments de l'interface sont intuitifs et faciles à utiliser.
- Les animations et les effets visuels améliorent l'expérience de jeu.

4. Système de matchmaking

- Les joueurs sont jumelés avec des adversaires de niveau similaire.
- Le système garantit des parties équilibrées et compétitives.
- Les joueurs peuvent rechercher des parties aléatoires ou inviter des amis à jouer.

Contraintes techniques

- Utilisation du framework React pour le développement de l'application.
- Intégration de technologies telles que AJAX, Web Services pour récupérer les données et assurer la communication avec le serveur.
- Sécurité des données et protection de la vie privée des utilisateurs.
- Performance optimale de l'application pour une expérience de jeu fluide.

2. Choix retenus

• Choix stratégiques organisationnels :

Nous nous sommes dans les séances de projet dans la salle **GS146**. Ensuite nous avons travaillé aussi chacun chez soi en partageant les tâches où chaque membre du projet exposait ses avancées et ses problèmes rencontrés.

Nous avons utilisé la méthode du télétravail durant toute la durée du projet en parallèle de nos rendez-vous physiques.

Nous nous envoyons régulièrement des mails pour demander de l'aide ou des

conseils aux différents membres du groupe. Nous établissons des discussions de groupes via un logiciel de messagerie instantanée.

- **Choix techniques :**



Concernant le framework, nous avons dû développer le jeu en **React** ; c'est un framework que nous maîtrisons en partie et que certains d'entre nous manipulent depuis déjà quelque temps.

React est largement utilisé pour la création d'interfaces utilisateur interactives et réactives, ce qui est essentiel pour offrir une expérience de jeu fluide et immersive.



Pour accompagner le **React**, nous avons opté pour des fonctions **JavaScript**, et notamment la combinaison de technologies baptisée **AJAX**.
AJAX (Asynchronous JavaScript And XML (« XML et JavaScript asynchrones ») est une solution informatique libre combinant plusieurs technologies comme le **HTML**, le **CSS** et le **JavaScript**, sont 3 éléments que nous allons couramment manipuler.

IV/ Réalisation

1. Analyse des besoins

Fonctionnalités du jeu :

- Les règles du jeu
- Types de cartes
- Fonctionnalités supplémentaires

Interface utilisateur :

- Les interfaces nécessaires : L'écran d'accueil, le plateau de jeu, les menus, les écrans de gestion du profil et les écrans de création de deck.
- Les éléments d'interface utilisateur : les boutons, les icônes, les barres de progression, les cartes visuelles.

Authentification et gestion des utilisateurs :

- Les fonctionnalités d'authentification utilisateur : la création de compte, la connexion et la déconnexion.

Matchmaking et gestion des parties :

- Les mécanismes de matchmaking
- Les fonctionnalités de création et de gestion des parties : la recherche d'adversaires, le suivi des résultats et la mise à jour des classements.

Gestion des cartes et des decks :

- Les fonctionnalités de gestion des cartes :
- Les contraintes de construction de decks : le nombre maximum de cartes autorisées, les restrictions de type.

2. Planification du projet

Le tableau suivant illustre le planning des différentes phases de réalisations de notre projet.

Semaine	Taches	Description
S1	Analyse des besoins	<ul style="list-style-type: none"> • Réaliser une analyse détaillée des besoins du projet, en se basant sur les fonctionnalités et les objectifs spécifiés. • Identifier les contraintes techniques et les exigences particulières. • Définir les spécifications fonctionnelles et techniques.
S2-S3	Conception et prototypage	<ul style="list-style-type: none"> • Réalise la conception de l'interface utilisateur, en créant des maquettes et des prototypes. • Travaille sur l'architecture de l'application, en définissant la structure globale du jeu et des mécanismes.
S2-s4	Développement	<ul style="list-style-type: none"> • Développer de la logique de jeu, de la gestion des cartes et des mécanismes de jeu. • Se concentre sur le développement de l'interface utilisateur, en utilisant le framework React pour créer les composants et les interactions.
S4	Assurance qualité et tests	<ul style="list-style-type: none"> • Effectue des tests réguliers pour s'assurer de la conformité du code aux normes de qualité. • Vérification de la stabilité, de la performance et de la sécurité de l'application. • Correction des bugs et des problèmes identifiés.
S5	Livraison et déploiement	<ul style="list-style-type: none"> • Préparation de la version finale de l'application.

3. Conception des maquettes

3.1 Page de Connexion :

- La page de connexion est conçue pour être simple et épurée, permettant aux utilisateurs de se connecter rapidement au jeu.
- Les champs requis sont l'adresse e-mail et le mot de passe, avec des boutons clairement indiqués pour soumettre les informations ou récupérer un mot de passe oublié.
- Un lien vers la page d'inscription est également présent pour les nouveaux utilisateurs.

3.2 Page d'Inscription :

- La page d'inscription invite les nouveaux joueurs à rejoindre le jeu en créant un compte.
- Les champs à remplir comprennent l'adresse e-mail, le nom d'utilisateur, et le mot de passe, avec des instructions claires sur les exigences pour chaque champ.
- Un bouton de soumission permet de créer le compte et de rediriger l'utilisateur vers la page de connexion ou directement dans le jeu.
- Ces pages sont conçues pour être intuitives et fournir une transition fluide entre la connexion, l'inscription, et l'accès au jeu. Des éléments visuels cohérents avec le thème du jeu sont utilisés pour maintenir l'engagement des utilisateurs.

Page de connexion

A hand-drawn sketch of a login page. At the top left is a logo consisting of two arrows pointing towards each other. To its right is a button labeled "S'inscrire". Below the logo is a large rectangular input field divided into three horizontal sections: the top section is labeled "EMAIL", the middle section is labeled "MOT DE PASSE", and the bottom section contains the text "SE CONNECTER".

Page d'inscription

A hand-drawn sketch of a sign-up page. At the top left is a logo consisting of two arrows pointing towards each other. To its right is a button labeled "S'inscrire". Below the logo are four input fields arranged vertically: the first is labeled "PSEUDO", the second is labeled "EMAIL", the third is labeled "MOT DE PASSE", and the fourth is labeled "CONFIRMER LE MDP". At the bottom right is a button labeled "VALIDER".

Connexion :

- Une fois le compte créé, l'utilisateur peut se connecter en utilisant son adresse e-mail et son mot de passe.
 - Le Web Service pour la connexion est : /login.
 - Il retourne les informations suivantes :
 - ID de l'utilisateur
 - E-mail
 - Nom d'utilisateur
 - Token d'identification de la session

Création de compte :

- L'utilisateur doit d'abord créer un compte en fournissant son adresse e-mail, un nom d'utilisateur, et un mot de passe.
 - Le Web Service pour créer un compte est le suivant : PUT /user.
 - Ce Web Service retourne l'ID de l'utilisateur qui a été créé.

3.4 Page d'Accueil :

- La page d'accueil sert de point d'entrée dans le jeu, offrant une navigation simple vers les fonctionnalités principales : le matchmaking et la galerie de cartes.
- Un en-tête distinctif présente le titre du jeu avec un design qui évoque l'univers fantastique de “League of Legends” et la mécanique de jeu de “Hearthstone”.
- Deux boutons principaux, “Matchmaking” et “Galerie”, sont mis en évidence pour diriger les joueurs vers les sections respectives.

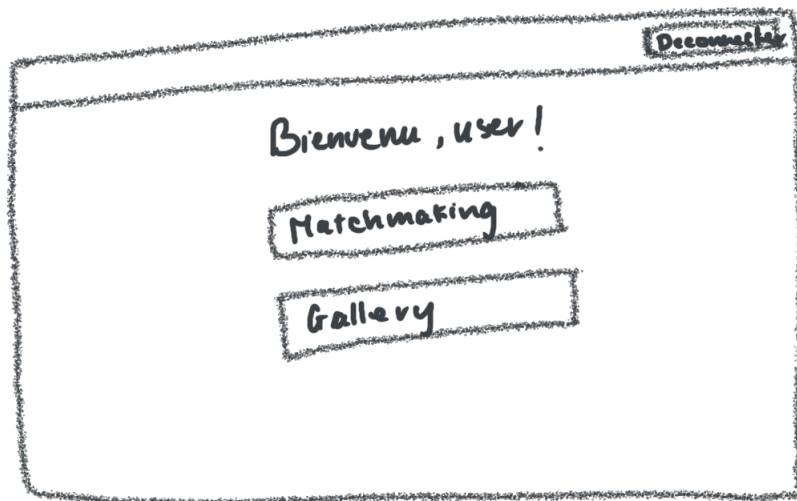
3.5 Page de Matchmaking :

- Cette page permet aux joueurs de se connecter avec d'autres compétiteurs en ligne. Elle affiche une liste d'adversaires potentiels et offre la possibilité de lancer un match.
- Trois boutons interactifs, “Valider le Match”, “Accepter le Match” et “Annuler le Match”, permettent une gestion fluide des invitations et des confirmations de match.

3.6 Page de Galerie :

- La galerie présente l'ensemble des cartes disponibles dans le jeu, permettant aux joueurs de composer leur deck de 20 cartes.
- Un système de sélection visuelle permet aux joueurs de choisir facilement leurs cartes, avec un compteur indiquant le nombre de cartes sélectionnées.

Page d'accueil



- Matchmaking : Lorsque l'utilisateur clique sur ce bouton, il est redirigé vers la page de matchmaking.
- Gallery : Lorsque l'utilisateur clique sur ce bouton, il est redirigé vers la galerie.

Two hand-drawn sketches illustrating the Matchmaking and Gallery features.

The Matchmaking page shows two user profiles: "User 1" and "User 2" facing each other with a "VS" between them. Below the profiles are three buttons: "Valider", "Accepter", and "Annuler".

The Gallery page displays a grid of 20 small square icons representing cards. Below the grid, the text "Mon deck" and "Prendre 20 heros pour attaquer" is written.

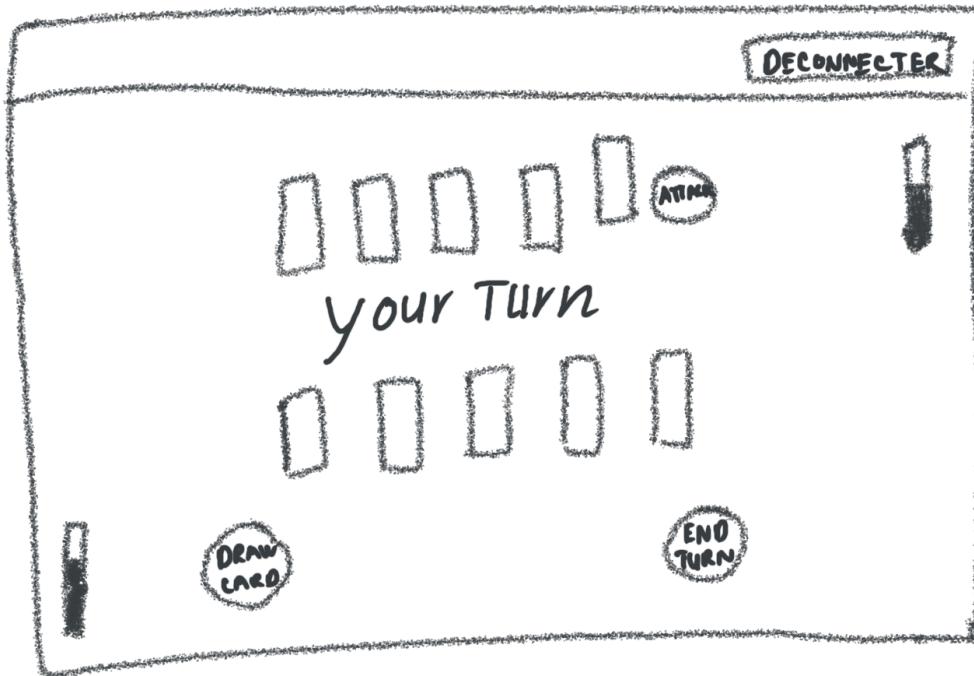
- Lorsque l'utilisateur accède à la page de matchmaking, il voit deux utilisateurs disponibles pour jouer.
- Leurs noms d'utilisateur ou autres informations pertinentes peuvent être affichés.
- L'utilisateur peut cliquer sur le bouton "Valider le match" pour indiquer qu'il souhaite jouer contre l'un des utilisateurs affichés.
- Si l'utilisateur reçoit une demande de match, il peut cliquer sur le bouton "Accepter le match" pour confirmer qu'il souhaite jouer contre l'autre utilisateur.
- Une fois que les deux utilisateurs ont accepté, le match commence.
- À tout moment avant que le match ne soit confirmé, l'utilisateur peut annuler la demande de match en cliquant sur le bouton "Annuler le match".

Page de gallery :

- Affichez toutes les cartes disponibles dans le jeu.
- Vous pouvez les afficher sous forme de vignettes ou de miniatures.
- L'utilisateur peut cliquer sur une carte pour la sélectionner.
- Ajoutez un compteur pour indiquer combien de cartes l'utilisateur a déjà sélectionnées.
- Initialement, ce compteur est à 0.
- Lorsque l'utilisateur a choisi 20 cartes, activez un bouton de validation.
- Ce bouton lui permettra de valider sa sélection.

3.7 Page de Match :

- La page de match est le cœur du jeu, où les joueurs s'affrontent en temps réel.
- Des boutons tels que "DrawCard", "EndTurn" et "Attaquer" sont disposés de manière ergonomique pour faciliter les actions des joueurs à chaque tour.



Accepter Match :

- 1 Piocher une carte (Draw Card) :
 - Si c'est votre tour, cliquez sur le bouton "Piocher une carte".
 - Cela ajoutera une carte de votre deck à votre main.
 - Vous ne pouvez piocher qu'une seule carte par tour.
- 2 Poser une carte champion (Play Card) :
 - Si vous avez des cartes dans votre main, vous pouvez les poser sur votre board (plateau).
 - Cliquez sur la carte que vous souhaitez jouer.
 - Cette carte devient active et peut attaquer lors de votre prochain tour.
- 3 Attaquer (Attack) :
 - Si vous avez des cartes actives sur votre board, vous pouvez les utiliser pour attaquer.
 - Cliquez sur la carte que vous souhaitez utiliser pour attaquer.
 - Ensuite, sélectionnez la carte adverse que vous souhaitez attaquer.
 - Les points de vie de l'adversaire seront réduits en fonction de la différence entre l'attaque de votre carte et la défense de la carte adverse.
- 4 Fin du tour (End Turn) :
 - Si vous avez terminé vos actions pour ce tour, cliquez sur le bouton "End Turn".
 - Cela passera le tour à l'adversaire.

4. Prototypage

Définition des Objectifs :

- Déterminez les objectifs du prototype, tels que tester une mécanique de jeu spécifique, l'interface utilisateur, ou l'équilibrage des cartes.

Création de Versions Simplifiées :

- Développez des versions simplifiées du jeu qui se concentrent sur les aspects les plus importants à tester.
- Utilisez des outils de prototypage rapides comme des cartes en papier ou des logiciels de prototypage numérique.

Test du Prototype :

- Organisez des sessions de test avec des joueurs cibles pour recueillir des données sur l'expérience de jeu.
- Observez les interactions des joueurs avec le jeu et notez les points de friction.

Analyse des Résultats :

- Analysez les données recueillies lors des tests pour identifier les aspects du jeu qui fonctionnent bien et ceux qui nécessitent des améliorations.

Iteration :

- Modifiez le prototype en fonction des retours et testez à nouveau.
- Répétez ce processus jusqu'à ce que les objectifs du prototype soient atteints.

Documentation :

- Documentez chaque étape du processus de prototypage, y compris les décisions prises et les leçons apprises.

5. Développement du Jeu

1. Gestion Utilisateur

- Création de compte : Développement du Web Service PUT /user pour l'inscription des utilisateurs.
- Connexion et déconnexion : Implémentation des Web Services /login et /logout pour la gestion des sessions.
- Suppression de compte : Mise en place du Web Service /users/unsubscribe pour permettre aux utilisateurs de supprimer leur compte.

2. Matchmaking

- Participation : Création du Web Service /matchmaking/participate pour ajouter un utilisateur à la liste des joueurs désirant jouer.
- Liste des joueurs : Développement du Web Service /matchmaking/getAll pour récupérer la liste des joueurs disponibles.
- Gestion des requêtes : Implémentation des Web Services /matchmaking/request et /matchmaking/acceptRequest pour gérer les invitations de match.

3. Gestion d'un Match

- Récupération des informations du match : Utilisation du Web Service /match/getMatch pour suivre l'évolution du match.
- Constitution du deck : Mise en place du Web Service /match/initDeck pour que les joueurs choisissent leurs cartes.
- Actions de jeu : Développement des Web Services /match/pickCard, /match/playCard, /match/attack, et /match/attackPlayer pour les différentes actions durant un match.
- Fin du tour et du match : Création des Web Services /match/endTurn et /match/finishMatch pour terminer un tour ou un match.

4. Tests et Itérations

- Description des méthodes de test utilisées pour valider les fonctionnalités et l'interface utilisateur.
- Analyse des résultats des tests et des ajustements effectués en réponse aux retours des utilisateurs.
- Conception Graphique et Interface Utilisateur
- Explication du processus de création des maquettes et du design final.
- Discussion sur la réactivité de l'interface et son adaptation aux différents appareils.

V/ Difficultés rencontrées

Dans notre groupe de quatre membres, nous avons rencontré une difficulté concernant la répartition équitable du travail. Un des membres contribuait moins que les autres, ce qui a créé un déséquilibre dans la charge de travail. Pour résoudre ce problème, nous avons collectivement décidé de discuter et d'établir un planning de travail à distance, permettant à chacun de visualiser les tâches assignées et les échéances. Cette approche a également facilité la présentation du travail de chacun, assurant ainsi une meilleure transparence et responsabilisation au sein de l'équipe. Grâce à cette stratégie, nous avons pu améliorer la collaboration et l'efficacité de notre groupe, tout en veillant à ce que le travail soit réparti de manière plus équitable.

Difficultés techniques

Plusieurs difficultés techniques ont ralenti le bon déroulement du projet, nous amenant à penser autrement la façon de développer.

Tout d'abord, nous avons rencontré des difficultés techniques liées à l'absence de requêtes API nécessaires pour le développement de notre jeu "League of Stones". En particulier, nous avons ressenti le besoin

url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web

d'améliorer notre système de notification pour informer les utilisateurs lorsqu'ils sont invités à un match. De plus, nous avons identifié le besoin d'une API permettant d'annuler une invitation à un match. Pour surmonter ces obstacles, nous avons dû concevoir et implémenter des solutions personnalisées. Nous devons développer des fonctionnalités supplémentaires pour notre backend afin de gérer efficacement les notifications et les annulations d'invitation, garantissant ainsi une expérience utilisateur fluide et réactive. Ces améliorations ont nécessité une collaboration étroite entre les membres de l'équipe et une répartition équitable des tâches pour répondre efficacement à ces besoins techniques.

Ensuite, nous n'avons pas encore mis en place une gestion sécurisée des sessions utilisateur, ce qui est crucial pour la protection de la confidentialité et la sécurité des données. De plus, le responsive design reste un défi, car nous devons nous assurer que notre jeu offre une expérience utilisateur cohérente sur une variété d'appareils.

Enfin, notre équipe travaille ensemble pour trouver des solutions efficaces et mettre en œuvre les meilleures pratiques pour surmonter ces obstacles techniques.

VI/ Conclusion

1. Tests et mise en place

Après la fin du développement, nous avons testé toutes les fonctionnalités pour s'assurer que tous les bugs avaient été éradiqués.

url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web

2. Si c'était à refaire

url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web

Avec l'expérience, acquise, tant au niveau de la réflexion que de la conception, s'il fallait refaire le projet, avec le gain de temps, on aurait pu mettre en place bon nombre d'améliorations.

Sinon concernant le déroulement de ce projet, au tout début nous travaillions beaucoup en télétravail, en mettant nos idées en commun une fois par

semaine, ce qui a été peu efficace au début, lors de la phase de réflexion, surtout que les besoins n'avaient pas été clairement définis, donc s'il fallait refaire, nous aurions travaillé physiquement en équipe, du moins pour la phase de réflexion.

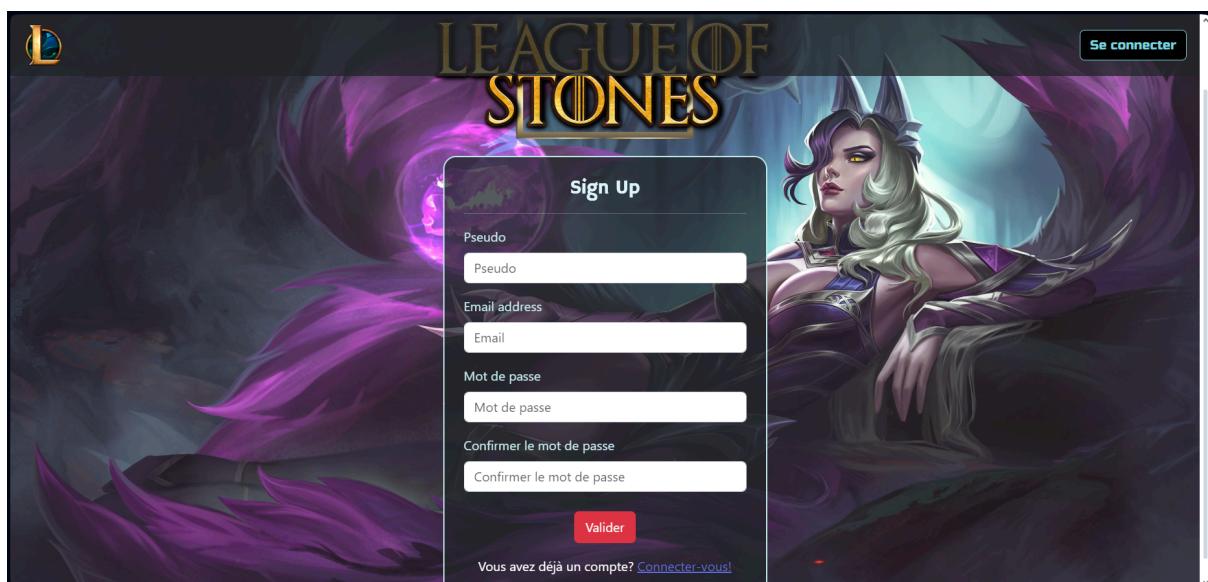
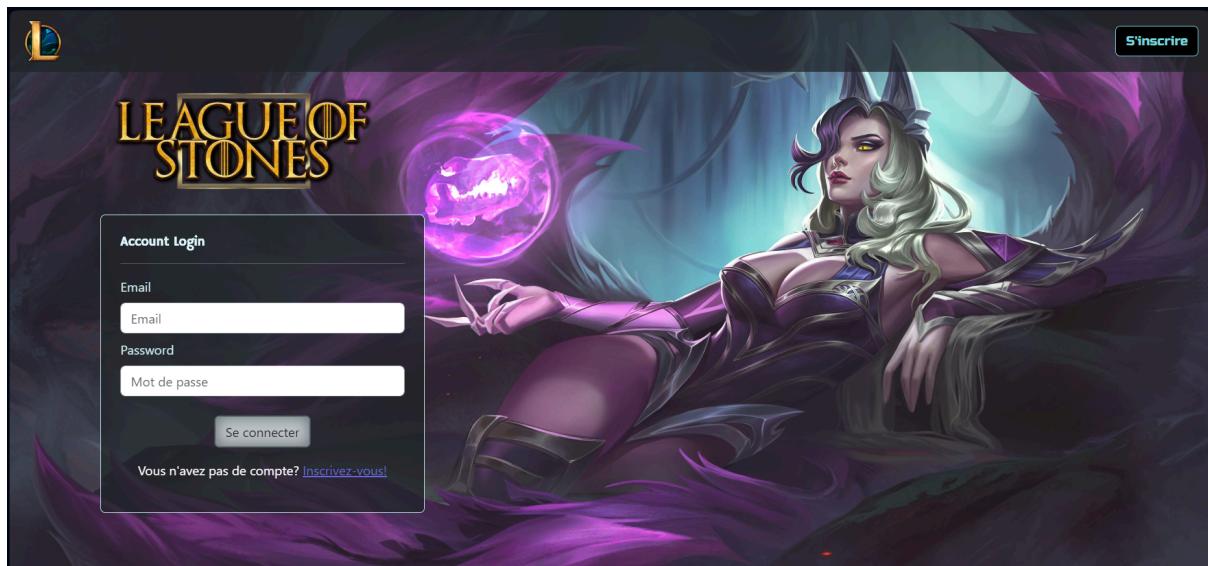
3. Apports personnels

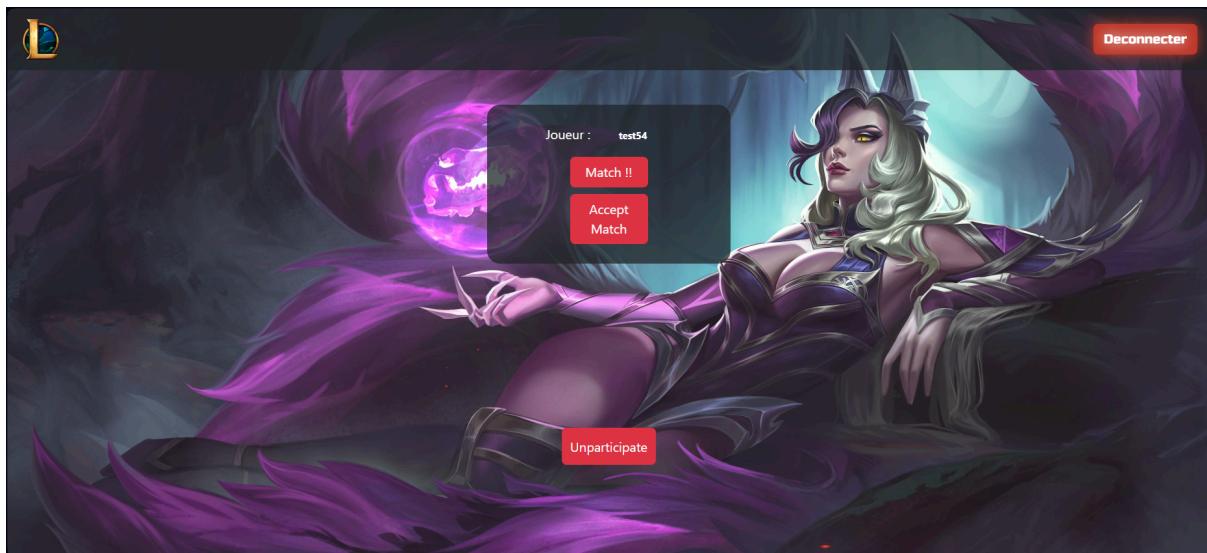
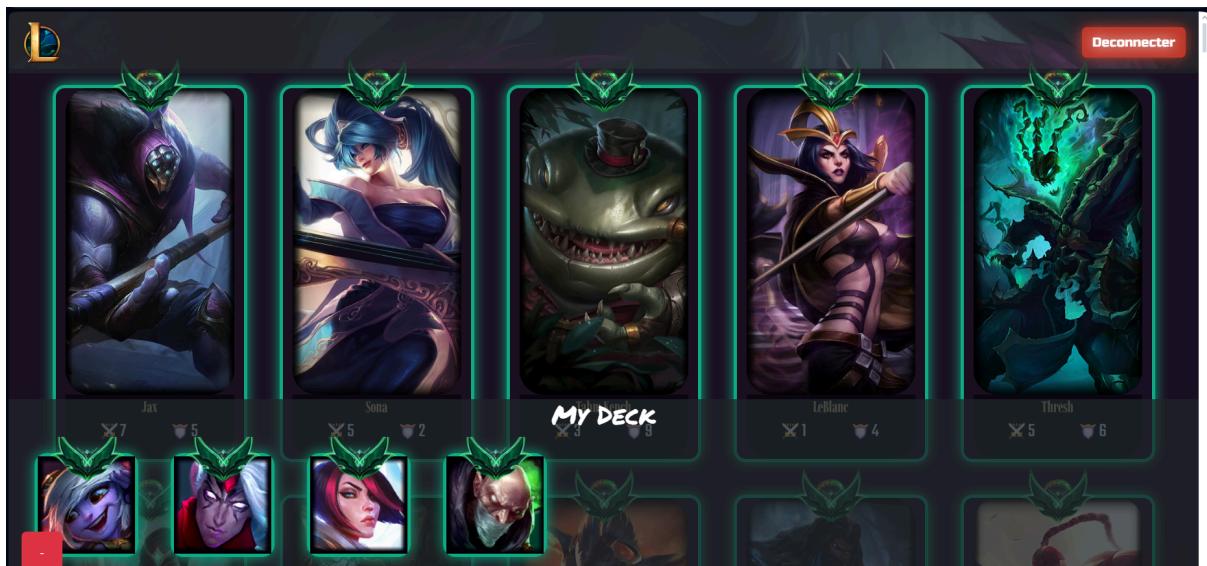
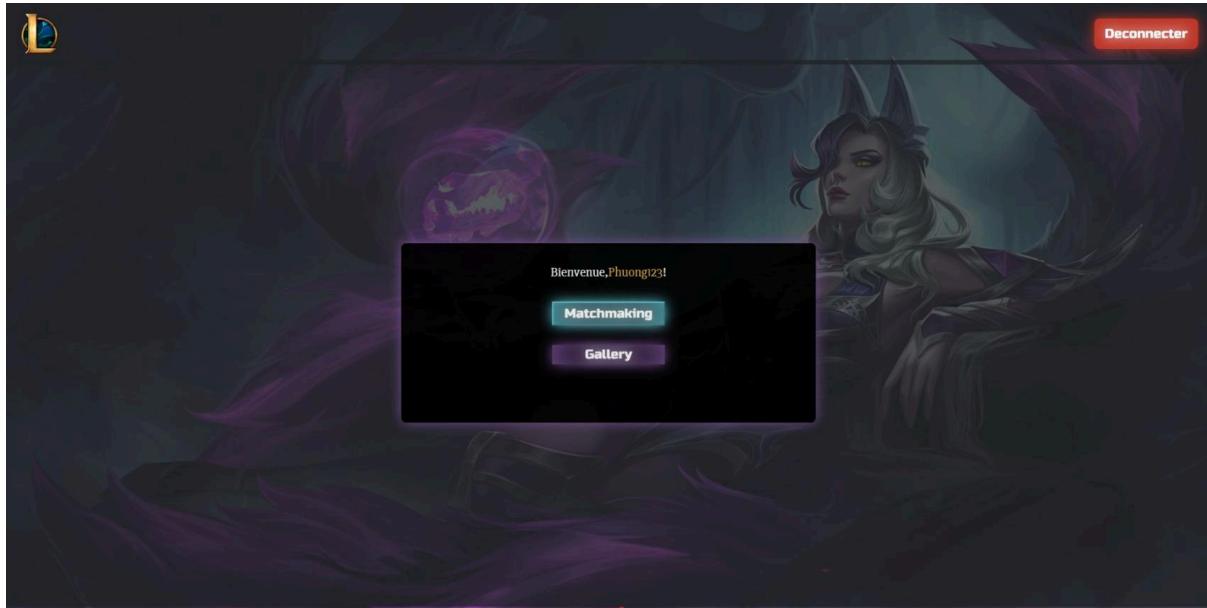
Lors de ce projet complet qui comprenait des phases de réflexion, veille, conception, modélisation et création, nous avons pu apprendre à :

- **Travailler en équipe** : tant au point de vue de l'ambiance que dans la répartition des tâches, l'esprit d'équipe s'est rapidement installé pour mener à bien le projet car sans cohésion ni dynamique de groupe, il n'aurait pu aboutir.
- **Poser les bonnes questions** : pour cerner correctement le problème, ce qui a donc développé les qualités de communicant de chacun.
- **Respecter des délais** : sans doute l'une des grandes difficultés du projet a consisté à respecter au mieux le planning pour fournir un projet fini en temps voulu.
- **Aboutir à un résultat** : même si l'intérêt premier des projets est de mettre à profit nos connaissances et à en acquérir de nouvelles nous nous devions de fournir une application fonctionnelle répondant au cahier des charges.

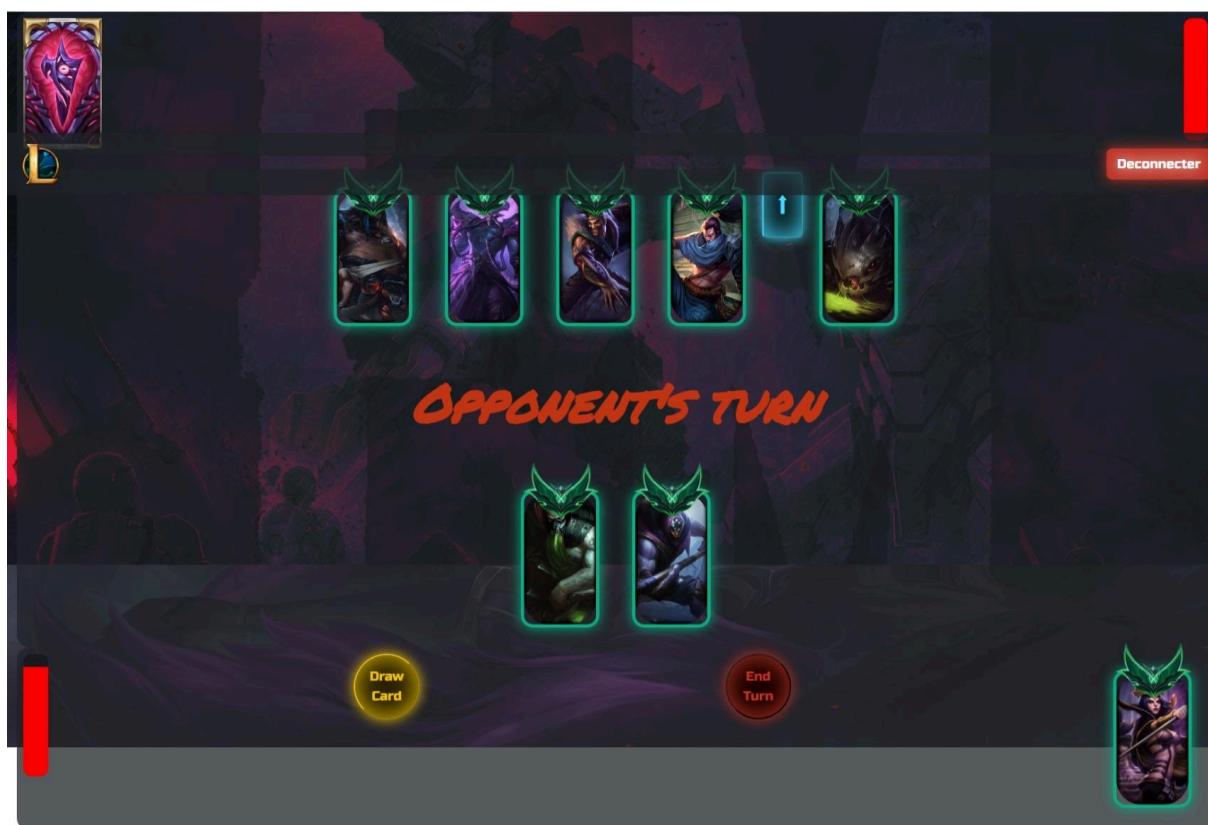
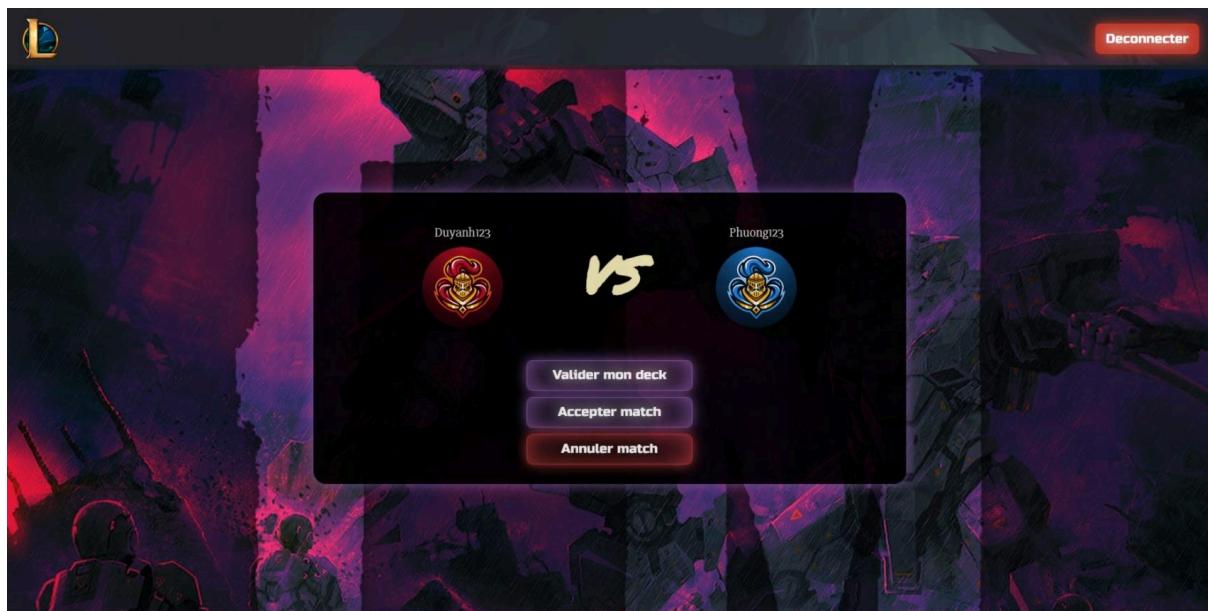
VII – Annexes

Des étapes de fonctionner du jeu:

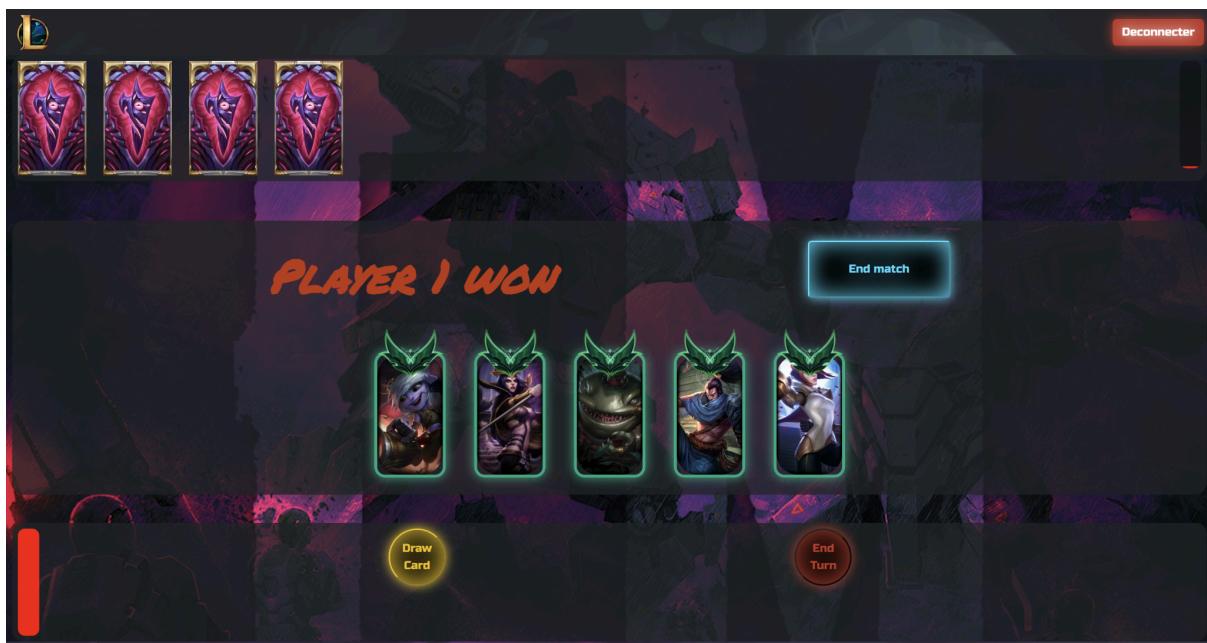
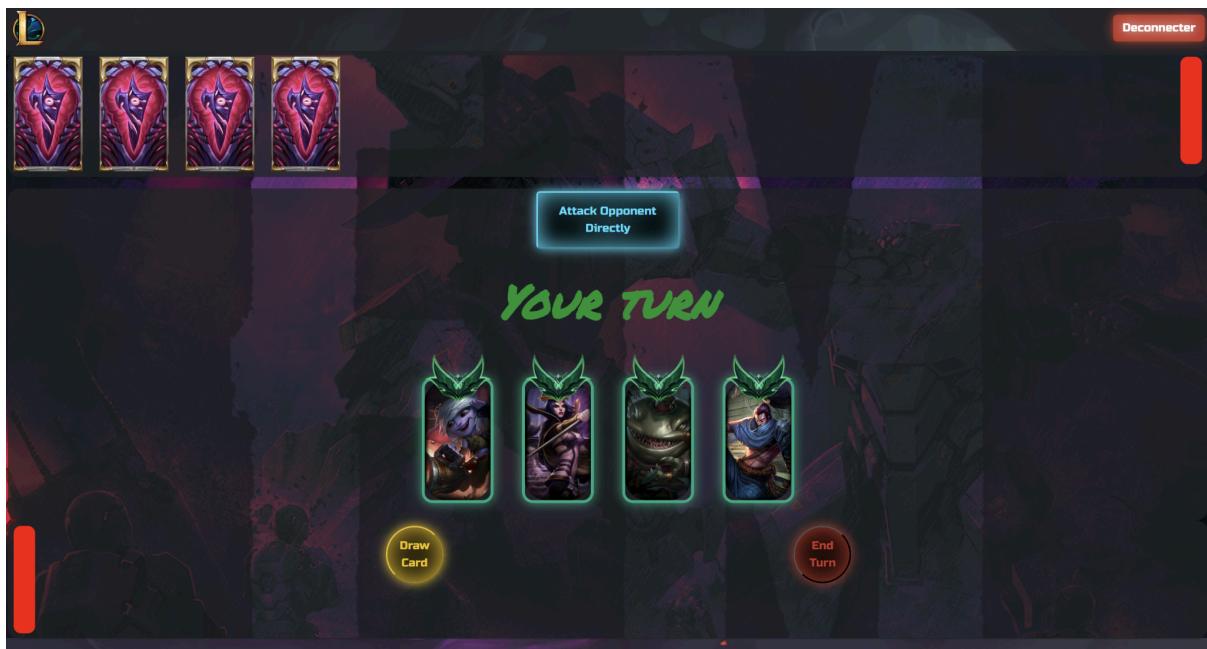




url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web



url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web



url GitLab: https://mi-git.univ-tlse2.fr/duy-anh.nguyen/projet_web