

---

**Équipe 102**

---

**PolyScrabble**  
**Spécifications des requis du système (SRS)**

**Version 2.0**

## Historique des révisions

Date	Version	Description	Auteur
2022-09-01	1.0	Rédaction initiale du SRS	Équipe 102
2022-09-08	1.1	Modification des exigences	Équipe 102
2022-09-12	1.2	Modification des exigences non fonctionnelles	Équipe 102
2022-09-14	1.3	Modification des exigences fonctionnelles et rédaction finale	Équipe 102
2022-09-26	1.4	Correction du document entier après la première remise	Équipe 102
2022-11-19	1.5	Délétion des exigences souhaitable non réalisé, correction des sections 1 et 2 selon les critères de correction	Maximiliano
2022-11-20	1.6	Correction des exigences fonctionnelles communes	Maximiliano
2022-11-21	1.7	Correction des exigences fonctionnelles, non fonctionnelles.	Maximiliano
2022-11-22	1.8	Ajout des définitions	Maximiliano
2022-12-01	2.0	Mise en page	Maximiliano

# Table des matières

<b>1. Introduction</b>	<b>5</b>
1.1. But	5
1.2 Définitions, acronymes et abréviations	5
1.3. Vue d'ensemble du document	6
<b>2. Description globale</b>	<b>6</b>
2.1. Caractéristiques des usagers	6
2.2. Interfaces	6
2.2.1. Interfaces usagers	6
2.2.2. Interfaces matérielles	7
2.2.3. Interfaces logicielles	8
2.2.4. Interfaces de communication	8
2.3. Contraintes générales	8
2.4. Hypothèses et dépendances	8
<b>3. Exigences fonctionnelles</b>	<b>9</b>
3.1 Modes de jeu	9
3.1.1 [Essentiel] Mode de jeu classique	9
3.1.2 [Souhaitable] Mode de jeu "Cartes de pouvoir"	10
3.1.3 [Souhaitable] Mode de jeu "Cartes de pouvoir" configurable	11
3.2. Visibilité des parties [Essentiel]	11
3.3. Interaction avec le plateau [Essentiel]	12
3.4 Matches favoris [Souhaitable]	14
3.5. Compte Utilisateur [Essentiel]	14
3.6 Avatar [Souhaitable]	15
3.7 Clavardage Canaux de Discussions[Essentiel]	15
3.8 Configuration de l'application [Essentiel]	16
3.9 Création de Cookies [Souhaitable]	16
3.10 Système de classement [Souhaitable]	17
3.11 Actions provoque du son [Essentiel]	17
3.12 Résultat de fin de partie [Souhaitable]	18
<b>Section spécifique au Client Léger:</b>	<b>18</b>
3.13 Clavardage Intégration[Essentiel]	18

3.14 Avatar [Essentiel]	19
3.15 Barre de recherche [Essentiel]	19
<b>Section spécifique au Client Lourd:</b>	<b>19</b>
3.16 Clavardage Intégration[Essentiel]	19
<b>4. Exigences non-fonctionnelles</b>	<b>20</b>
4.1. Utilisabilité	20
4.2. Fiabilité	20
4.3. Performance	20
4.4. Maintenabilité	20
4.5. Contraintes de conception	21
4.6. Sécurité	21
<b>Annexe</b>	<b>22</b>

# Spécifications des requis du système (SRS)

## 1. Introduction

### 1.1. But

Le SRS décrit le comportement externe de notre application Poly Scrabble. Il décrit aussi les exigences fonctionnelles et non fonctionnelles, les contraintes de conception, ainsi que les autres facteurs nécessaires à la description complète des exigences du logiciel à développer.

### 1.2 Définitions, acronymes et abréviations

OS. : Operating System (Système d'opération)

ELO : système de rang permettant d'évaluer le niveau de chaque joueur

SDK : kit de développement logiciel

AWS : services web d'amazon

LINT : système d'automatisation pour éviter les erreurs de styles lorsque l'on écrit dans un langage de programmation

ELECTRON : environnement permettant de créer des applications multi-plateformes

TUILE : objet utilisé au scrabble contenant une lettre et son pointage

USB: L'USB est une norme de bus informatique en série qui sert à connecter des périphériques informatiques à un ordinateur

RJ45: est un connecteur permettant généralement de se connecter à des interfaces réseaux.

WIFI: est un protocole de communication sans fil

WINDOWS: est un système d'exploitation pour ordinateur

LINUX: est un système d'exploitation pour ordinateur

ANDROID: est un système d'exploitation pour mobile

SCRABBLE: est un jeu de société où l'objectif est de cumuler des points, sur la base de tirages aléatoires de lettres, en créant des mots sur une grille carrée.

MONGODB: est un système de base de données

HTTP: "protocole de transfert hypertexte", est un protocole de communication client-serveur

WEBSOCKET: est un protocole réseau visant à créer des canaux de communication

REST: est un style d'architecture logicielle définissant un ensemble de contraintes à utiliser pour créer des services web.

### **1.3. Vue d'ensemble du document**

La section 2 de notre rapport contient une description générale du produit. La section 3 de notre rapport est consacrée à la description de nos exigences fonctionnelles, cela comprend le type d'exigence, souhaitable ou essentiel. La section 4 est utilisée pour décrire les exigences non fonctionnelles du logiciel.

## **2. Description globale**

PolyScrabble est un logiciel permettant de jouer en ligne au scrabble que cela soit individuellement contre un ordinateur ou de façon multijoueur en ligne, via une application de bureau (sous Windows 10) ou mobile (Android 9.0 Pie). Les utilisateurs auront des comptes personnalisés avec lesquels ils pourront jouer et clavarder. Il s'agit d'un projet d'évolution de logiciel.

### **2.1. Caractéristiques des usagers**

Nos usagers visés sont âgés entre 12 et 30 ans et ont une bonne familiarité avec le jeu Scrabble physique mais n'ont pas de connaissances poussées en informatique.

### **2.2. Interfaces**

#### **2.2.1. Interfaces usagers**

Le logiciel comprendra:

- Une page d'accueil donnant l'option de se connecter à son compte utilisateur pour ensuite accéder aux options de jeux (mode classique, mode elo).
- Une page de profil servant à afficher les informations de l'utilisateur connecté (nom d'utilisateur, avatar). Cette page affichera aussi l'historique de la partie de l'utilisateur et ses horaires de connexion/déconnexion ainsi que ses matchs favoris.
- Une page de jeu contenant le plateau de jeu ainsi que le chevalet des joueurs. Cette page contiendra aussi des panneaux d'informations sur l'avancement de la partie (nom du joueur jouant, score de chacun des joueurs et leurs nom respectifs), cette page inclut aussi le clavardage.

- Une interface de clavardage servant à communiquer à travers les canaux de discussion, cette interface sera disponible à travers toute l'application.
- Une interface de recherche d'utilisateurs permettant à un utilisateur de rechercher les informations d'autres joueurs pour voir leurs statistiques personnelles (historique des matchs).
- L'interface sera adaptée en fonction du client, que ce soit avec une souris et un clavier pour le client lourd ou une interface tactile pour le client léger.

Le serveur comprendra des interfaces pour faire des requêtes Rest ou Websockets:

- Nous avons différentes routes pour les requêtes de type REST. Nous avons les routes pour l'authentification, la connexion, aussi le serveur possède des routes pour certaines méthodes pour les usagers, comme la création d'un compte et la modification du compte. Les usagers pourront avoir des avatars associés à leurs comptes, donc nous avons une route qui permet de faire des opérations sur les avatars comme demander d'avoir tous les avatars sur le serveur pour la sélection, obtenir un avatar random, ou dans le cas du client léger, sauvegarder un avatar personnalisé. Finalement nous avons une fonctionnalité qui permet de sauvegarder certaines parties comme des parties favorites, donc nous avons une route pour sauvegarder une partie et obtenir une partie.
- Toutes les autres interfaces du serveur sont sous la forme de websockets. Cela inclut tout le système de jeu, de clavardage etc.

### 2.2.2. Interfaces matérielles

L'application Poly Scrabble sera développée pour un ordinateur ayant une souris, un clavier et un écran. Cet ordinateur devra être sous l'OS Windows 10. La résolution de l'écran devra être 1920 par 1080 pixels et avec mise à l'échelle de la résolution à 100%.

Selon les spécifications d'électron **[1] (voir annexe)** le client n'a pas de prérequis en termes d'installations nécessaires (tel que chromium). Par exemple, si on installe discord, on n'a pas besoin d'installer chromium.

Notre application sera aussi développée pour une interface mobile sous l'OS Android 9.0 Pie. La tablette devrait avoir une résolution de 1920 pixels par 1200 pixels. Dans notre cas, la tablette sur laquelle le logiciel sera testé est la Samsung Galaxy Tab A de 2019. Pour le client léger, nous avons aussi l'utilisation de l'appareil photo de la tablette afin de permettre à l'utilisateur d'avoir un avatar personnalisé.

### 2.2.3. Interfaces logicielles

Notre client léger sera déployé sur une tablette sous l'OS Android 9.0 Pie. Pour notre client lourd l'interface sur laquelle l'application sera déployée doit être Windows 10, vu que notre application sera déployée avec electron, l'utilisateur n'aura pas de prérequis à installer. Notre serveur sera déployé sur une instance EC2 utilisant comme OS Amazon Linux 2.

En plus de ces trois déploiements, nous utiliserons une base de données (MongoDb) pour nous permettre de stocker des données et de les rendre persistantes pour chaque utilisateur.

### 2.2.4. Interfaces de communication

Nos clients léger et lourds devront avoir une connexion à internet sans fil (par wifi) ou filaire (rg45, usb). Ils devront aussi posséder une carte réseau pour pouvoir gérer ces connexions. Le client léger et lourd ne se communiquent pas entre eux directement mais communiquent au serveur qui leur communique en retour. Le serveur est le seul qui communique directement à la base de données à travers le protocole http.

## 2.3. Contraintes générales

Nous possédons plusieurs contraintes :

- Le client lourd doit posséder un ordinateur sous Windows 10 avec une connexion internet.
- Le client léger doit posséder une tablette Galaxy Tab A datant de 2019 sous Android 9.0 Pie minimum avec une connexion internet.
- Le client lourd doit être programmé avec le langage Typescript avec le cadriciel Angular,
- Le client léger doit être programmé avec le langage Dart avec le cadriciel Flutter
- Notre application doit pouvoir exécuter plusieurs parties en même temps
- Notre application doit pouvoir gérer plusieurs utilisateurs connectés simultanément.
- Notre application doit offrir une expérience utilisateur non saccadée et fluide.
- Notre application doit offrir un clavardage en temps réel avec une persistance sur la base de données.

## 2.4. Hypothèses et dépendances

On suppose que les utilisateurs ont accès à une connexion internet avec une latence



acceptable (moins de 200ms). On suppose que les utilisateurs ont un ordinateur ou une tablette relativement récente (minimum android 9.0 Pie pour tablette et minimum windows 10 pour ordinateur).

## **3. Exigences fonctionnelles**

### **Section Commune au deux clients:**

### **3.1 Modes de jeu**

#### **3.1.1 [Essentiel] Mode de jeu classique**

3.1.1.1 Le système doit pouvoir créer un plateau afin de permettre de deux à quatre utilisateurs de jouer.

3.1.1.2 Le système doit faire commencer l'utilisateur avec sept lettres choisies aléatoirement dans la réserve.

3.1.1.3 Le système doit permettre à l'utilisateur d'exécuter une action parmi les suivantes:

3.1.1.3.1 Le système doit permettre à l'utilisateur de placer des lettres.

3.1.1.3.2 Le système doit permettre à l'utilisateur de pouvoir échanger des lettres.

3.1.1.3.3 Le système doit permettre à l'utilisateur de pouvoir passer son tour.

3.1.1.4 Le système doit limiter le temps de chaque tour pour les utilisateurs au temps choisi au début de la partie.

3.1.1.5 Le système doit passer le tour de l'utilisateur s' il dépasse le temps alloué.

3.1.1.6 Le système doit permettre à l'utilisateur de placer uniquement des lettres présentes sur son chevalet

3.1.1.7 Le système doit permettre à l'utilisateur de placer des mots verticalement.

3.1.1.8 Le système doit permettre à l'utilisateur de placer des mots horizontalement.

3.1.1.9 Le système doit obliger l'utilisateur de placer un mot sur la case centrale (position H8 de la grille) si c'est la première fois qu'un utilisateur joue dans la partie

3.1.1.10 Le système doit permettre à l'utilisateur de placer uniquement des mots qui touchent d'autres lettres déjà placé sur le plateau

3.1.1.11 Le système doit supporter l'utilisation d'accents pour le placement des lettres.

3.1.1.12 Le système doit pouvoir empêcher l'utilisateur de faire des échanges de lettres s' il ne reste pas assez de lettres dans la réserve (moins de 7 lettres).

3.1.1.13 Le système doit changer les tours quand le tour est terminé

3.1.1.14 Le système doit valider le mot de l'utilisateur s' il fait partie du dictionnaire.

3.1.1.15 Le système doit remettre sur le chevalet les lettres utilisées pour former un mot à l'utilisateur si ce mot n'a pas été accepté par le système.

3.1.1.16 Le système doit donner des points aux utilisateurs pour leurs mots placés en fonction du pointage de chaque lettre utilisée

3.1.1.17 Le système doit prendre en compte les cases bonus lors du comptage de point pour chaque mot

3.1.1.18 Le système doit pouvoir donner 50 points supplémentaires à l'utilisateur qui place ses sept lettres de son chevalet en une fois.

3.1.1.19 Le système doit désactiver une case bonus (elle ne compte plus comme étant un bonus) si un mot à déjà été placé sur cette même case.

### 3.1.2 **[Souhaitable]** Mode de jeu "Cartes de pouvoir"

3.1.2.1 Le système doit donner l'option a l'utilisateur de jouer en mode "Carte de pouvoir"

3.1.2.2 Le système doit pouvoir attribuer des cartes de pouvoir à ce dernier à chaque trois mots placés par l'utilisateur.

3.1.2.3 Le système doit limiter le nombre maximum de cartes de pouvoir à trois par utilisateur.

3.1.2.4 Le système doit permettre a l'utilisateur de faire sauter le tour d'un utilisateur avec une carte pouvoir

3.1.2.5 Le système doit permettre à l'utilisateur de transformer une case vide en case bonus de couleur aléatoire avec une carte pouvoir

3.1.2.6 Le système doit permettre à l'utilisateur de réduire de moitié le temps de jeu des

autres utilisateurs dans la même partie avec une carte pouvoir

3.1.2.7 Le système doit permettre à l'utilisateur d'échanger une lettre de la réserve avec une lettre de son chevalet avec une carte pouvoir

3.1.2.8 Le système doit permettre à l'utilisateur d'échanger son chevalet avec celui d'un autre utilisateur avec une carte pouvoir

3.1.2.9 Le système doit permettre à l'utilisateur de retirer des points à l'utilisateur avec le plus de points et les répartir à tous les utilisateurs (incluant l'utilisateur visé) avec une carte pouvoir

### 3.1.3 [Souhaitable] Mode de jeu "Cartes de pouvoir" configurable

3.1.3.1 Le système doit limiter le nombre maximum de cartes de pouvoir à trois par utilisateur.

3.1.3.2 Le système doit permettre au créateur de la partie de pouvoir activer ou désactiver des cartes pouvoirs en début de partie.

3.1.3.3 Le système doit limiter le nombre minimal de cartes pouvoirs activées à un.

3.1.3.4 Le système doit offrir une nouvelle carte de pouvoir à chaque trois mot validé par l'utilisateur.

## 3.2. Visibilité des parties [Essentiel]

3.2.1 Le système doit permettre à l'utilisateur d'avoir accès à une interface lui montrant les parties accessibles.

3.2.1.1 Le système doit avoir une interface des parties accessibles qui affiche le nom de l'utilisateur qui a créé la partie.

3.2.1.2 Le système doit avoir une interface des parties accessibles qui affiche le nombre de joueurs réel et virtuel.

3.2.1.3 Le système doit avoir une interface des parties accessibles qui affiche le nombre d'observateurs.

3.2.2 Le système doit permettre le commencement d'un match lorsque il y a 2 ou plus joueurs réels dans un match.

3.2.3 Le système doit permettre à l'utilisateur de créer une partie privée.

3.2.3.1 Le système doit permettre au créateur d'une partie privée de pouvoir accepter ou

refuser les joueurs qui essayent de rejoindre la partie.

3.2.4 Le système doit permettre à l'utilisateur de pouvoir créer une partie publique.

3.2.4.1 Le système doit permettre à l'utilisateur de pouvoir observer une partie publique en tout temps.

3.2.4.2 Le système doit permettre au créateur de la partie publique d'avoir l'option de protéger l'accès à une partie à l'aide d'un mot de passe.

3.2.4.3 Le système doit permettre à l'utilisateur d'entrer le bon mot de passe pour rentrer dans une partie publique protégée.

3.2.5 Le système doit permettre à l'observateur d'être ajouté au canal de discussion lorsqu'il rejoint une partie.

3.2.6 Le système doit permettre à l'observateur de voir le plateau.

3.2.6.1 Le système doit désactiver la fonctionnalité chevalier pour l'observateur.

3.2.6.2 Le système doit permettre à l'observateur d'observer les chevaliers de l'ensemble des joueurs.

3.2.6.3 Le système doit empêcher l'observateur d'avoir des tours.

3.2.7 Le système doit permettre à l'observateur de pouvoir prendre la place d'un joueur virtuel (ordinateur).

3.2.7.1 Le système doit permettre à l'observateur de continuer la partie avec le même nombre de points que le joueur virtuel remplacé

3.2.7.2 Le système doit permettre à l'observateur de continuer la partie avec les mêmes lettres que le joueur virtuel remplacé.

### **3.3. Interaction avec le plateau [Essentiel]**

3.3.1 Le système doit permettre aux joueurs de pouvoir placer des lettres de leur chevalet sur le plateau avant de commettre leur choix.

3.3.2 Le système doit permettre à la synchronisation des tuiles en direct lorsqu'elles sont glissées-déposées pour l'utilisateur qui déplace les tuiles

3.3.2.1 Le système doit afficher l'action de placement aux autres joueurs seulement

lorsque le placement est confirmé.

3.3.2.2 Le système doit afficher l'action de placement aux autres joueurs seulement lorsque le placement est confirmé.

3.3.2.3 Le système doit afficher aux autres joueurs uniquement un indicateur visuel de la case du départ du joueur actif lors du placement.

3.3.2.4 Le système doit permettre au joueur de placer des lettres à travers une action de Glisser-déposer sur le plateau.

3.3.2.5 Le système doit permettre au joueur de pouvoir sélectionner une tuile de son chevalet

3.3.2.6 Le système doit permettre au joueur de pouvoir glisser une tuile vers le plateau de jeu

3.3.2.7 Le système doit permettre au joueur de pouvoir déposer une tuile sur une case vide.

3.3.2.8 Le système doit retourner la tuile dans le chevalet si celle-ci est déposée sur tout autre élément que le plateau de jeu lors du placement du joueur actif.

3.3.2.9 Le système doit permettre au joueur de déplacer les tuiles placées pendant son tour sur d'autres cases vides avec une action de Glisser-déposer.

3.3.2.10 Le système doit permettre au joueur de remettre les tuiles placées pendant son tour sur son chevalet en les glissant-déposant sur celui-ci.

3.3.2.11 Le système doit confirmer le placement à travers le bouton d'entrée (Enter).

3.3.2.12 Le système doit vérifier qu'un seul mode de placement est en cours d'utilisation à la fois.

3.3.2.13 Le joueur doit enlever toutes les lettres placées pour changer de mode de placement.

3.3.3 Le système doit permettre la synchronisation en continu

3.3.3.1 Le système doit propager et afficher l'action de placement du joueur actif aux autres joueurs en continu.

3.3.3.2 Le système doit mettre à jour les plateaux des autres joueurs avec des tuiles

placées ou retirées par ce dernier à chaque trois mots placés par l'utilisateur

3.3.3.3 Une fois le placement confirmé par le joueur, le système doit mettre à jour le visuel habituel du plateau contenant les nouvelles tuiles.

3.3.3.4 Le système doit, dans le cas d'un placement Glisser-déposer, transmettre l'affichage de la tuile glissée aux autres joueurs pendant son déplacement.

### **3.4 Matches favoris [Souhaitable]**

3.4.1 Le système doit permettre à l'utilisateur de maintenir un historique de tous ses matchs favoris.

3.4.2 Le système doit permettre à l'utilisateur de sauvegarder un match dans ses favoris.

### **3.5. Compte Utilisateur [Essentiel]**

3.5.1 Le système doit permettre à un utilisateur de se créer un compte

3.5.1.1 Le système doit limiter le nom d'utilisateur à un minimum de trois lettres.

3.5.1.2 Le système doit limiter le nom d'utilisateur pour qu'il soit alphanumérique

3.5.1.3 Le système doit vérifier la validité du courriel utilisé.

3.5.2 Le système doit permettre à l'utilisateur de se connecter à un compte existant

3.5.3 Le système doit permettre de modifier le nom d'utilisateur

3.5.4 Le système doit contenir une page utilisateur qui contient les éléments suivants

3.5.4.1 Le système doit contenir une page utilisateur qui contient le nom de l'utilisateur.

3.5.4.2 Le système doit contenir une page utilisateur qui contient le elo de l'utilisateur.

3.5.4.3 Le système doit contenir une page utilisateur qui contient l'avatar de l'utilisateur.

3.5.4.4 Le système doit contenir une page utilisateur qui contient l'historique des connexions de l'utilisateur.

3.5.4.5 Le système doit contenir une page utilisateur qui contient les statistiques de l'utilisateur.

3.5.5 Le système doit contenir au sein de la page utilisateur, une interface statistiques avec les éléments suivants:

3.5.5.1 Le système doit contenir une interface statistique qui contient le nombre de parties jouées.

3.5.5.2 Le système doit contenir une interface statistique qui contient le nombre de parties gagnées.

3.5.5.3 Le système doit contenir une interface statistique qui contient le temps moyen.

3.5.5.4 Le système doit contenir une interface statistique qui contient les points moyen par partie.

3.5.6 Le système doit contenir au sein de la page utilisateur, une interface historique de parties avec les éléments suivants:

3.5.6.1 Le système doit contenir une interface historique de parties qui contient les parties avec leur date

3.5.6.2 Le système doit contenir une interface historique de parties qui contient les parties avec les résultat finaux respectifs

3.5.6.3 Le système doit contenir une interface statistique qui contient le(s) nom(s) du(des) adversaire(s).

3.5.6.4 Le système doit contenir une interface statistique qui contient les scores.

## **3.6 Avatar [Souhaitable]**

3.6.1 Le système doit permettre à l'utilisateur au moment de la création du compte de pouvoir choisir son avatar à partir d'une liste prédéfinie d'images.

3.6.2 Le système doit pouvoir laisser l'utilisateur modifier son avatar avec une liste prédéfinie d'avatars.

3.6.3 Le système doit présenter l'avatar de l'utilisateur à des endroits pertinents:

3.6.3.1 Le système doit présenter l'avatar sur la page d'accueil

3.6.3.2 Le système doit présenter l'avatar sur la page profil

3.6.3.3 Le système doit présenter l'avatar sur la page de jeu

3.6.3.4 Le système doit présenter l'avatar sur la page de clavardage

## **3.7 Clavardage Canaux de Discussions[Essentiel]**

3.7.1 Le système doit permettre à l'utilisateur de clavarder avec d'autres utilisateurs.

- 3.7.2 Le système doit offrir un canal spécifique à un match en cours
- 3.7.3 Le système doit permettre à tous les utilisateurs de clavarder dans un canal commun
- 3.7.4 Le système doit empêcher un utilisateur de quitter le canal commun
- 3.7.5 Le système doit empêcher la déléation du canal commun
- 3.7.6 Le système doit permettre de créer des canaux de discussions
- 3.7.7 Le système doit permettre à l'utilisateur de joindre un canal de discussion
- 3.7.8 Le système doit offrir une liste de tous les canaux disponibles
- 3.7.9 Le système doit permettre à un utilisateur de quitter un canal de discussion
- 3.7.10 Le système doit permettre à au créateur du canal de supprimer le canal de discussion
- 3.7.11 Le système doit permettre à l'utilisateur de passer d'un canal à l'autre
- 3.7.12 Le système doit permettre à l'utilisateur de voir tout son historique de clavardage à l'exception du canal d'un match.

## **3.8 Configuration de l'application [Essentiel]**

- 3.8.1 Le système doit permettre de changer le thème de l'application:
  - 3.8.1.1 Le système doit offrir un mode clair.
  - 3.8.1.2 Le système doit offrir un mode sombre.
- 3.8.2 Le système doit appliquer le thème choisi par l'utilisateur lors de la connexion.
- 3.8.3 Le système doit permettre de changer la langue de l'application en anglais
- 3.8.4 Le système doit permettre de changer la langue de l'application en francais
- 3.8.5 Le système doit préserver ces configurations entre chaque connexion.

## **3.9 Création de Cookies [Souhaitable]**

- 3.9.1 Le système doit fournir l'utilisateur d'un cookie lors d'une connexion.
- 3.9.2 Le système doit vérifier la validité du cookie.
- 3.9.3 Le système doit préserver le cookie tant que l'utilisateur reste connecté



3.9.4 Le système doit connecter automatiquement le client si le cookie est valide.

3.9.5 Le système doit protéger certaines fonctionnalités derrière la vérification du cookie

### **3.10 Système de classement [Souhaitable]**

3.10.1 Le système doit accorder un classement de base lors de la création d'un compte.

3.10.2 Le système doit offrir un mode de jeu classé à l'utilisateur.

3.10.3 Le système doit modifier le classement selon le résultat de la partie.

3.10.4 Le système doit chercher un adversaire qui est proche de l'elo de l'utilisateur.

3.10.4.1 Le système doit permettre à l'utilisateur de choisir la différence maximale d'elo entre le sien et son adversaire.

3.10.5 Le système doit permettre de présenter à l'utilisateur une interface montrant son adversaire, lorsque l'utilisateur trouve une partie.

3.10.5.1 Le système doit permettre à l'utilisateur d'accepter ou de refuser le match.

3.10.7 Le système doit permettre à l'utilisateur de voir le changement de son elo à la fin de la partie.

### **3.11 Actions provoque du son [Essentiel]**

3.11.1 Le système doit produire du son lorsque l'utilisateur place un bon mot.

3.11.2 Le système doit produire du son lorsque l'utilisateur place une lettre sur le plateau.

3.11.3 Le système doit produire du son lorsque l'utilisateur place un mot non valide.

3.11.4 Le système doit produire du son lorsque l'utilisateur place supprime des lettres du plateau.

3.11.5 Le système doit produire un son lorsque l'utilisateur gagne une partie.

3.11.6 Le système doit produire un son lorsque l'utilisateur perd une partie.

3.11.7 Le système doit permettre à l'utilisateur de désactiver les sons produits par le système.

### **3.12 Résultat de fin de partie [Souhaitable]**

- 3.12.1 Le système doit automatiquement ouvrir une page de résultats à la fin de la partie
- 3.12.2 Le système doit afficher sur la page des résultats le vainqueur du match.
- 3.12.3 Le système doit afficher sur la page des résultats les vaincus du match.
- 3.12.4 Le système doit afficher sur la page des résultats les observateurs du match.
- 3.12.5 Le système doit afficher sur la page des résultats le score de chacun des joueurs.
- 3.12.6 Le système doit afficher sur la page des résultats le temps du match du joueur.
- 3.12.7 Le système doit afficher sur la page des résultats le nombre de tour dans le match
- 3.12.8 Le système doit afficher sur la page des résultats la date du match
- 3.12.9 Le système doit afficher sur la page des résultats le nombre de lettres restante dans la réserve

## **Section spécifique au Client Léger:**

### **3.13 Clavardage Intégration[Essentiel]**

- 3.13.1 Le système doit permettre aux utilisateurs de clavarder pendant un match
- 3.13.2 Le système doit permettre aux utilisateurs de clavarder en dehors d'un match.
- 3.13.3 Le système doit avertir l'utilisateur lors de la réception d'un nouveau message
  - 3.13.3.1 Le système doit présenter un effet visuel lors de la réception d'un message
  - 3.13.3.2 Le système doit présenter un effet sonore lors de la réception d'un nouveau message
  - 3.13.3.3 Le système doit retirer cet effet visuel lorsque le message est lu.

### **3.14 Avatar [Essentiel]**

3.14.1 Le système doit permettre à l'utilisateur de choisir une image à partir des avatars prédéfinis.

3.14.2 Le système doit permettre à l'utilisateur de modifier son avatar

3.14.2.1 Le système doit permettre de capturer une image à l'aide de sa caméra pour son avatar.

3.14.2.2 Le système doit permettre à l'utilisateur de choisir une image à partir des avatars prédéfinis.

### **3.15 Barre de recherche [Essentiel]**

3.15.1 Le système doit permettre de trouver des utilisateurs à travers une barre de recherche.

3.15.2 Le système doit permettre d'afficher un maximum de 5 utilisateurs qui correspondent partiellement ou complètement au contenu de la recherche.

3.15.3 Le système doit permettre à l'utilisateur de se rediriger vers la page de l'utilisateur lorsqu'il sélectionne un profil dans la barre de recherche.

3.15.3.1 Le système doit permettre d'afficher les données du profil sélectionné suite à sa recherche.

## **Section spécifique au Client Lourd:**

### **3.16 Clavardage Intégration[Essentiel]**

3.16.1 Le système doit permettre aux utilisateurs de clavarder pendant un match

3.16.2 Le système doit permettre aux utilisateurs de clavarder en dehors d'un match.

3.16.3 Le système doit permettre à l'utilisateur de détacher la boîte de clavardage

3.16.4 Le système doit permettre à l'utilisateur de fermer la page détaché pour qu'elle revienne à sa place

## **4. Exigences non-fonctionnelles**

### **4.1. Utilisabilité**

- 4.1.1. L'utilisateur doit être capable de se créer un compte en maximum 30 secondes
- 4.1.2. L'utilisateur doit être capable de créer une partie multijoueur en moins de 25 secondes.
- 4.1.3. L'utilisateur doit être capable de modifier l'interface visuelle en moins de 2 minutes.
- 4.1.4 L'utilisateur doit être capable de rentrer un mot en moins de 10 secondes.
- 4.1.5. L'utilisateur doit être capable de se connecter à son compte en moins de 1 minute

### **4.2. Fiabilité**

- 4.2.1. Le système doit être en panne moins de 1 heure par semaine.
- 4.2.2 Le temps moyen entre panne moyenne doit être de 7 jours.
- 4.2.3 Le temps moyen de panne à sa réparation est de 1 heure.

### **4.3. Performance**

- 4.3.1. Le temps maximum pour qu'un utilisateur se connecte est de 2 secondes.
- 4.3.2. L'application client doit utiliser moins de 100 mégabytes de ram pour un ordinateur ou une tablette.
- 4.3.3. L'application client lourd doit prendre au maximum 150 mégabytes de place sur le disque.
- 4.3.4 Le serveur doit supporter un minimum de 100 clients connectés simultanés
- 4.3.5 Le serveur doit supporter le clavardage simultané de ces 100 joueurs.
- 4.3.6 Le serveur doit offrir une latence de 50 ms pour l'envoi et la réception d'un message.

### **4.4. Maintenabilité**

- 4.4.1 Le client lourd doit utiliser un système de lint (voir annexe)
- 4.4.2 Les membres de l'équipe auront accès au serveur AWS et pourront le redémarrer et regarder ce qui l'a fait crash.

4.4.3 Les membres de l'équipe doivent utiliser des merge requests pour intégrer leur code.

## **4.5. Contraintes de conception**

4.5.1 Les langages de programmation pour réaliser le logiciel sont le Typescript et Dart.

4.5.2 La plateforme de développement pour le client léger doit être Android Studio.

4.5.3 La plateforme de développement pour le client lourd doit être Visual Code.

4.5.4 Le SDK Android utilise vise la version Android 10 (SDK niveau 29)

4.5.5 Le client léger utilisera le cadre flutter, le client lourd utilisera Angular avec Electron, notre serveur roulera sous nodeJS avec le cadre express.

4.5.6 La base de données utilisée sera mongoDb.

4.5.7 Les OS sur lesquels l'application pourra fonctionner sont Windows, MacOS et Linux pour le client lourd et uniquement Android pour le client léger.

4.5.8 Le serveur doit être hébergé sur un serveur EC2 sur AWS.

4.5.9 Le développeurs doivent utiliser un environnement local pour développer une fonctionnalité

4.5.9.1 La base de données utilisera une base de données pour développement et non de production

4.5.9.2 Le client lourd et léger utiliseront une instance du serveur local.

## **4.6. Sécurité**

4.6.1 Les mots de passe des utilisateurs doivent être hachés sur le serveur et sauvegardés sur la base de données lors de la création du compte.

4.6.2 Le système doit refuser l'accès aux mots de passe à une source autre que les développeurs du logiciel.

4.6.3 Le système doit protéger l'accès à certaines API's réservés à des utilisateurs connectés et authentifiés.

4.6.4 Le système doit utiliser http.

## **Annexe**

1. <https://www.electronjs.org/docs/latest/tutorial/tutorial-prerequisites>