

**Turb0\_P41nt\_F1v3\_Th0u54nd.exe**  
**Spécifications des requis du système (SRS)**

**Version 2.1**

# Historique des révisions

Date	Version	Description	Auteur
2019-09-03	1.0	Ébauche - exigences fonctionnelles	Justine Lambert
2019-09-05	1.1	Exigences essentielles client lourd Description globale du système	Justine Lambert Christophe Carreau-Lacasse
2019-09-07	1.2	Exigences essentielles client léger, exigences souhaitables et exigences non fonctionnelles	Justine Lambert
2019-09-12	1.3	Introduction	Marie-Elaine Bérubé
2019-09-12	1.4	Révision du document	Justine Lambert
2019-09-19	2.0	Améliorations aux exigences fonctionnelles	Gabriel Bottari Justine Lambert Marie-Elaine Bérubé
2019-09-26	2.1	Améliorations aux exigences fonctionnelles	Christophe Carreau-Lacasse

# Table des matières

<b>1. Introduction</b>	4
1.1. But	4
1.2. Définitions, acronymes et abréviations	4
1.3. Vue d'ensemble du document	5
<b>2. Description globale</b>	5
2.1. Caractéristiques des usagers	5
2.2. Interfaces	5
2.2.1. Interfaces usagers	5
2.2.2. Interfaces matérielles	6
2.2.3. Interfaces logicielles	6
2.2.4. Interfaces de communication	6
2.3. Contraintes générales	6
2.4. Hypothèses et dépendances	7
<b>3. Exigences fonctionnelles</b>	7
3.1. Clavardage	7
3.2. Profil utilisateur	8
3.4. Modes de jeu	10
3.5. Construction du jeu	11
3.6. Joueurs virtuels	12
3.7. Tutoriel	13
3.8 Expérience utilisateur	14
3.9. Effets visuels et sonores	15
<b>4. Exigences non-fonctionnelles</b>	16
4.1. Utilisabilité	16
4.2. Fiabilité	16
4.3. Performance	16
4.4. Maintenabilité	16
4.5. Contraintes de conception	16
4.6. Sécurité	17
4.7. Exigences de la documentation usager en ligne et du système d'assistance	17

# Spécifications des requis du système (SRS)

## 1. Introduction

### 1.1. But

Le document de SRS décrit le comportement externe du logiciel *Fais-moi un dessin*. Il décrit tout d'abord globalement les différentes contraintes de l'application, puis les exigences fonctionnelles et les exigences non fonctionnelles du projet, ainsi que les autres facteurs nécessaires à la description complète des exigences du logiciel à développer.

### 1.2. Définitions, acronymes et abréviations

**Architecture client-serveur:** Type d'architecture utilisé pour la communication entre les clients et le serveur.

**Bonne réponse:** Le joueur réussit à deviner le mot ou l'expression représentant l'objet dessiné.

**Canal de discussion:** Un canal de discussion est une boîte de clavardage où il est possible de clavarder avec d'autres usagers.

**Client léger:** Application exécutable compatible avec un assistant personnel (PDA).

**Client lourd:** Application exécutable compatible avec un ordinateur personnel (PC).

**Construction du jeu:** Création d'une partie. Nécessite des informations telles qu'un terme ou une expression, une image et des indices.

**Dessin:** Ensemble de lignes tracées par un joueur lors d'une partie.

**Fonction de hachage:** Algorithme cryptographique utilisé pour stocker sécuritairement les mots de passe des utilisateurs.

**Groupe ou groupe de jeu:** Un groupe de jeu est un groupe composé de 2 à 4 joueurs réunis pour jouer une partie.

**Joueur virtuel:** Joueur contrôlé par l'application. Il ne peut que dessiner et peut avoir une personnalité distincte.

**Lobby:** Page d'accueil de l'application qui présente les parties en attentes et les parties en cours.

**Manche:** Débute au moment où un joueur commence à faire un dessin pour son coéquipier, et se termine lorsqu'un point est attribué à une des deux équipes ou si personne n'arrive à deviner correctement.

**Mauvaise réponse:** Le joueur tente de deviner le mot ou l'expression représentant l'objet dessiné mais ne réussit pas.

**Module de l'application:** Partie de l'application qui peut être développée indépendamment des autres.

**Partie:** Une suite de manches jusqu'à ce qu'une équipe obtienne un total de trois points, déclenchant la fin de la partie.

**Personnalité du joueur virtuel:** Les joueurs virtuels ont différentes formules typiques, ce qui donne l'impression de jouer contre plusieurs différents joueurs.

**Salle d'attente:** Page affichée pour les joueurs en attente d'une partie. Ils peuvent discuter avec les autres joueurs en attente.

**Serveur:** Dispositif informatique offrant des services et répondant aux requêtes des clients.

**Système:** Plateforme d'exécution et logiciel *Fais-moi un dessin*.

**Usager:** Le terme usager sera synonyme aux termes utilisateur du logiciel et joueur dans le présent document. Il s'agit de toute personne qui interagit avec l'application.

### 1.3. Vue d'ensemble du document

La section 2 contient une description globale de l'application, allant des contraintes générales à la description des usagers. Les sections 3 et 4 présentent les exigences fonctionnelles et non fonctionnelles.

## 2. Description globale

Le logiciel à développer consiste en un jeu du style *Fais-moi un dessin* en réseau, sur ordinateur et sur appareil Android. Le principe du jeu est simple : deux équipes de deux joueurs s'affrontent, et à tour de rôle, un joueur doit faire deviner à l'autre joueur de son équipe un mot en le dessinant à l'écran. Le dessin est retransmis en direct sur tous les appareils connectés, permettant ainsi aux joueurs à distance de deviner. Le jeu aura aussi un système intégré de messagerie instantanée, un système de gestion d'amis ainsi que plusieurs modes de jeu différents.

### 2.1. Caractéristiques des usagers

Le logiciel développé sera principalement utilisé par des ingénieurs logiciels. Il s'agit donc d'usagers ayant de bonnes connaissances techniques, ainsi que beaucoup d'expérience avec différents logiciels. Étant donné leur expertise, ils s'attendent à un logiciel ayant une interface complète, qui leur permet d'accéder rapidement et facilement à toutes les fonctionnalités disponibles. Nous pouvons assumer que les utilisateurs sont familiers avec les interfaces standardisées du marché; le logiciel peut donc les intégrer sans avoir à guider l'utilisateur. Il s'agit aussi d'usagers qui sont moins tolérants vis-à-vis des problèmes techniques qui pourraient survenir au cours de l'utilisation d'un logiciel.

### 2.2. Interfaces

#### 2.2.1. Interfaces usagers

Le jeu sera constitué de plusieurs interfaces distinctes, qui seront pour la plupart implémentées à la fois sur le client lourd et le client léger. Sur le client lourd, les interfaces seront définies avec des fichiers XAML utilisant la technologie WPF. Sur le client léger, les interfaces seront définies avec des fichiers XML.

La première interface est le *lobby*, ou encore la page d'accueil. Dans cette interface, l'utilisateur pourra voir sa liste d'amis, clavarder avec des amis, rejoindre une partie ou bien créer une partie. Il est à noter que seul un usager sur le client lourd peut créer une partie, le bouton de création ne sera donc pas affiché sur le client léger.

Lorsqu'un joueur crée une partie, il est amené à une seconde interface où il peut configurer la partie, en décidant du nombre de joueurs et du mode de jeu souhaité, ainsi que configurer les joueurs virtuels qui seront dans la partie.

Lorsqu'un joueur rejoint une partie existante, il est amené dans la salle d'attente, où il peut voir les joueurs de chaque équipe et clavarder avec tous les autres joueurs présents dans la salle d'attente. Lorsque le nombre minimal de joueurs requis pour commencer la partie est atteint dans la salle d'attente, le joueur qui a créé la partie peut la commencer, ce qui redirige les joueurs vers l'interface de jeu.

L'interface de jeu sera constituée d'un espace de clavardage avec tous les membres de la partie et d'un espace de dessin, où les joueurs vont soit y voir le dessin qu'un autre joueur est en train de dessiner, ou bien ce sera leur surface de dessin s'il s'agit de leur tour de dessin.

Une autre interface sera l'interface de clavardage externe, seulement disponible sur le client lourd. Cette interface sera dans une fenêtre séparée et permettra de clavarder avec ses amis.

Le serveur étant hébergé sur la plateforme infonuagique Microsoft Azure, il n'y a pas d'interface usager, car pour la configuration ainsi que le démarrage et l'arrêt du serveur il existe un portail de configuration en ligne sur le site d'Azure.

### *2.2.2. Interfaces matérielles*

Le logiciel sera disponible sur deux interfaces matérielles, soit un client lourd qui sera disponible sur les ordinateurs Windows et un client léger disponible sur les appareils Android. Le client léger sera optimisé pour fonctionner sur une tablette, et garantit un fonctionnement optimal sur une tablette Samsung Galaxy Tab A, puisqu'il s'agit du type de tablette pour laquelle il sera développé. Les méthodes d'entrées supportées par le client lourd sont le clavier et la souris, filaire ou sans-fil. Le client lourd ayant une taille dynamique, il supporte tous les types d'écrans qu'un ordinateur Windows supporte, peu importe leur résolution. Il est toutefois recommandé d'utiliser un écran ayant une résolution de 1920x1080 pixels afin d'avoir une expérience optimale. Les méthodes d'entrées supportées par le client léger sont l'écran tactile et le clavier virtuel Android. L'utilisation d'un clavier ou d'une souris externe est possible, mais n'est pas officiellement supportée et ne sera pas testée. L'interface du client léger sera faite de manière dynamique et devrait donc fonctionner sur tous les écrans peu importe leur résolution, mais sera optimisée pour l'utilisation sur un écran de 10.1 pouces avec une résolution de 1920x1200 pixels. Pour les deux clients, tous les sons générés seront transmis par la sortie standard du périphérique. Il faudra donc un système de son ou bien des écouteurs afin d'entendre les effets sonores du jeu.

### *2.2.3. Interfaces logicielles*

Le logiciel est composé de trois interfaces logicielles. En premier lieu, il y aura un client lourd sur l'ordinateur, développé en C# avec le cadriciel de présentation WPF (Windows Presentation Foundation) qui sera compilé en exécutable compatible uniquement avec le système d'exploitation Windows. Le client lourd sera compilé de façon à être compatible avec Windows 10 seulement.

Sa deuxième interface logicielle est un serveur .NET Core écrit en C# hébergé sur la plateforme infonuagique Azure de Microsoft, qui communiquera avec tous les clients du logiciel et une base de données. La base donnée sera de type MySQL et sera aussi hébergé sur la plateforme Azure de Microsoft.

Sa troisième interface est une application native Android optimisée pour une utilisation sur tablette avec Android version 9.0 (*Android Pie*), qui sera développée en Kotlin.

Finalement, nous aurons un engin de conversion de SVG, qui prendra en paramètre une image et qui retournera un fichier SVG reproduisant l'image. Cet engin de conversion sera intégré à l'exécutable du serveur.

### *2.2.4. Interfaces de communication*

Afin que les différents clients communiquent entre eux, nous utiliserons l'interface de communication Internet sans-fil (WiFi) pour le client léger, et l'interface Ethernet câblé pour les clients lourds. Il s'agit du meilleur moyen de connecter deux appareils, particulièrement mobiles, puisque la fonctionnalité sans-fil est disponible sur la grande majorité des appareils, est rapide et assure la transmission de toutes les données. De plus, nous ferons l'usage de la technologie WebSocket afin de permettre l'envoi et la réception en continu d'information entre les clients et le serveur.

## **2.3. Contraintes générales**

Afin de répondre aux demandes et critères du document de vision, le logiciel se doit de respecter les contraintes suivantes:

- Le client lourd doit intégrer *PolyPaint*.
- Le serveur doit être en mesure de supporter plusieurs parties simultanément.

- Les clients doivent être fluides (pas d'effets saccadés) sur le réseau local de Polytechnique Montréal.
- L'interface du client léger doit être adaptée pour une tablette Android et plus spécifiquement, il doit bien fonctionner sur la tablette Samsung Galaxy Tab A.

## 2.4. Hypothèses et dépendances

Afin d'assurer un fonctionnement fluide et sans saccade, le logiciel dépend d'une connexion stable à un réseau Internet sans-fil.

## 3. Exigences fonctionnelles

Cette section présente les exigences fonctionnelles pour le système. Les exigences essentielles et souhaitables sont présentées conjointement, mais leur type est clairement identifié.

### 3.1. Clavardage

#### Exigences communes au client lourd et client léger

3.1.1 (Essentielle) Le système doit permettre à l'utilisateur de pouvoir clavarder avec d'autres utilisateurs connectés.

#### Exigences propres au client lourd

3.1.1.1 (Essentielle) Le système doit permettre à l'utilisateur de communiquer avec les autres joueurs de sa partie lors d'une partie.

3.1.1.2 (Essentielle) Le système doit permettre à l'utilisateur de communiquer avec les autres utilisateurs connectés au serveur en tout temps, même s'ils ne sont pas dans la même partie.

3.1.2 (Essentielle) Le système doit permettre à l'utilisateur de clavarder à même une fenêtre intégrée à l'application.

3.1.3 (Essentielle) Le système doit permettre à l'utilisateur de clavarder dans une fenêtre externe à l'application.

3.1.4 (Essentielle) Le système doit permettre de basculer de la fenêtre intégrée à la fenêtre externe.

3.1.5 (Souhaitable) Le système doit permettre à l'utilisateur de créer un canal de discussion.

3.1.6 (Souhaitable) Le système doit permettre à l'utilisateur de joindre un canal de discussion existant.

3.1.6.1 (Souhaitable) Le système doit permettre à l'utilisateur de faire partie de plusieurs canaux de discussion simultanément.

3.1.6.2 (Souhaitable) Le système doit permettre à l'utilisateur de quitter un canal de communication.

3.1.7 (Souhaitable) Le système doit conserver et afficher un historique de clavardage pour chaque canal de discussion.

#### Exigences propres au client léger

3.1.8 (Essentielle) Le système doit permettre à l'utilisateur de pouvoir accéder à une interface de clavardage à partir de toute vue de l'application.

3.1.9 (Essentielle) Le système doit notifier l'utilisateur de la réception de chaque nouveau message via un indicateur visuel et un effet sonore.

3.1.9.1 (Essentielle) Le système doit faire disparaître l'indicateur visuel à l'ouverture du clavardage.

- 3.1.10 (Essentielle) Le système doit permettre à l'utilisateur de créer un canal de discussion.
- 3.1.11 (Essentielle) Le système doit permettre à l'utilisateur de joindre un canal de discussion existant.
  - 3.1.11.1 (Essentielle) Le système doit permettre à l'utilisateur de faire partie de plusieurs canaux de discussion simultanément.
  - 3.1.11.2 (Souhaitable) Le système doit permettre à l'utilisateur de quitter un canal de communication.
- 3.1.12 (Essentielle) Le système doit conserver et afficher un historique de clavardage pour chaque canal de discussion.

## **3.2. Profil utilisateur**

### Exigences communes au client lourd et client léger

- 3.2.1 (Essentielle) Le système doit permettre de créer un profil utilisateur pour le joueur.
  - 3.2.1.1 (Essentielle) Le système doit permettre de créer un profil utilisateur en utilisant une adresse email.
    - 3.2.1.1.1 (Essentielle) Le système doit permettre d'entrer un nom d'utilisateur.
    - 3.2.1.1.2 (Essentielle) Le système doit permettre d'entrer un pseudonyme.
    - 3.2.1.1.3 (Essentielle) Le système doit permettre d'entrer un mot de passe.
    - 3.2.1.1.4 (Essentielle) Le système doit permettre de choisir un avatar parmi une liste prédéterminée de huit avatars.
  - 3.2.1.2 (Souhaitable) Le système doit permettre d'utiliser Google pour créer son profil.
    - 3.2.1.1.1 (Essentielle) Le système doit permettre d'obtenir le nom du joueur via son profil Google.
    - 3.2.1.1.2 (Essentielle) Le système doit permettre d'entrer un pseudonyme.
    - 3.2.1.1.3 (Essentielle) Le système doit permettre d'obtenir l'avatar du joueur via son profil Google.
  - 3.2.1.3 (Souhaitable) Le système doit permettre d'utiliser Facebook pour créer son profil.
    - 3.2.1.3.1 (Essentielle) Le système doit permettre d'obtenir le nom du joueur via son profil Facebook..
    - 3.2.1.3.2 (Essentielle) Le système doit permettre d'entrer un pseudonyme.
    - 3.2.1.3.3 (Essentielle) Le système doit permettre d'obtenir l'avatar du joueur via son profil Facebook.
- 3.2.2 (Essentielle) Le système doit permettre à l'utilisateur de s'authentifier au système à l'aide de son profil utilisateur.
- 3.2.3 (Essentielle) Le système doit afficher la partie publique du profil utilisateur de chaque joueur à tous les utilisateurs.
  - 3.2.3.1 (Essentielle) Le système doit afficher le pseudonyme de chaque joueur à tous les utilisateurs.
  - 3.2.3.2 (Essentielle) Le système doit afficher les avatars de chaque joueur à tous les utilisateurs.
- 3.2.4 (Essentielle) Le système doit afficher le profil utilisateur complet au joueur authentifié.
  - 3.2.4.1 Le système doit afficher le pseudonyme du joueur authentifié.
  - 3.2.4.1 Le système doit afficher le nom d'utilisateur du joueur authentifié.
  - 3.2.4.1 Le système doit afficher l'avatar du joueur authentifié.
- 3.2.5 (Essentielle) Le système doit permettre au joueur authentifié de consulter ses statistiques de jeu.



- 3.2.5.1 (Essentielle) Le système doit afficher au joueur authentifié les date et heure de chacune de ses déconnexions au système.
- 3.2.5.2 (Essentielle) Le système doit afficher au joueur authentifié le nombre de parties qu'il a jouées.
- 3.2.5.3 (Essentielle) Le système doit afficher au joueur authentifié le pourcentage de ses victoires.
- 3.2.5.4 (Essentielle) Le système doit afficher au joueur authentifié la durée moyenne d'une de ses parties.
- 3.2.5.5 (Essentielle) Le système doit afficher au joueur authentifié son temps total passé à jouer.
- 3.2.6 (Essentielle) Le système doit permettre à l'utilisateur de consulter son historique de jeu détaillé.
  - 3.2.6.1 (Essentielle) Le système doit afficher au joueur authentifié les date et heure de chacune de ses connexions au système.
  - 3.2.6.2 (Essentielle) Le système doit afficher au joueur authentifié les date et heure de chacune de ses déconnexions au système.
  - 3.2.6.3 (Essentielle) Le système doit afficher au joueur authentifié l'historique de ses parties jouées.
  - 3.2.6.4 (Essentielle) Le système doit afficher au joueur authentifié le résultat de chacune de ses parties.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.3 Lobby de jeu**

#### Exigences communes au client lourd et client léger

- 3.3.1 (Essentielle) Le système doit permettre un lobby de jeu permettant de sélectionner un mode jeu.
- 3.3.2 (Essentielle) Le système doit permettre un lobby de jeu permettant de joindre un groupe de joueurs existant, s'il y a lieu.
- 3.3.3 (Essentielle) Le système doit permettre un lobby de jeu permettant de créer un nouveau groupe de joueurs.
- 3.3.4 (Essentielle) Le système doit permettre un lobby de jeu permettant d'afficher les informations des groupes de joueurs existants.
  - 3.3.4.1 Le système doit permettre un lobby de jeu permettant d'afficher la liste des groupes existants.
  - 3.3.4.2 Le système doit permettre un lobby de jeu permettant d'afficher la liste des joueurs humains faisant partie des groupes existants.
  - 3.3.4.3 Le système doit permettre un lobby de jeu permettant d'afficher la liste des joueurs virtuels faisant partie des groupes existants.
  - 3.3.4.3 Le système doit permettre un lobby de jeu permettant d'afficher les équipes de joueurs faisant partie des groupes existants.
- 3.3.5 (Essentielle) Le système doit permettre aux joueurs humains d'ajouter un ou des joueurs virtuels à un groupe.
  - 3.3.5.1 (Essentielle) Le système doit permettre la création d'au plus un joueur virtuel par équipe de deux joueurs en mode classique.

3.3.5.2 (Essentielle) Le système doit permettre la création d'au plus un joueur virtuel dans un groupe de jeu en mode sprint solo.

3.3.6 (Essentielle) Le système doit amener le joueur qui joint un groupe à une salle d'attente de jeu.

3.3.7 (Essentielle) Le système doit permettre à n'importe quel joueur de démarrer la partie dès que le nombre de joueurs requis dans le groupe est atteint dans la salle d'attente.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.4. Modes de jeu**

#### Exigences communes au client lourd et client léger

3.4.1 (Essentielle) Le système doit permettre au groupe de joueurs de s'affronter en mode classique, soit une équipe de deux contre une autre équipe de deux.

3.4.1.1 (Essentielle) Le système doit permettre au joueur connecté de pouvoir, à tour de rôle, dessiner et deviner.

3.4.1.1.1 (Essentielle) Le système doit permettre au joueur de pouvoir dessiner un objet à faire deviner à son coéquipier à l'aide des outils de dessin.

3.4.1.1.2 (Essentielle) Le système doit permettre au joueur d'avoir la possibilité de deviner le dessin fait par son coéquipier.

3.4.1.1.3 (Essentielle) Le système doit permettre au joueur de pouvoir dessiner en mode classique, sans défis additionnels.

3.4.1.1.4 (Souhaitable) Le système doit permettre au joueur de pouvoir dessiner en mode aveugle, c'est-à-dire que seul son coéquipier peut voir son dessin.

3.4.1.1.5 (Souhaitable) Le système doit permettre au joueur de pouvoir dessiner en mode "un seul trait", c'est-à-dire que le dessin s'arrête dès qu'il termine son premier trait.

3.4.1.2 (Essentielle) Le système doit permettre à l'équipe d'obtenir un point si le joueur qui devine a une bonne réponse.

3.4.1.3 (Essentielle) Le système doit donner le droit de réplique à l'équipe adverse si le joueur a trois mauvaises réponses consécutives.

3.4.1.4 (Essentielle) Le système doit permettre à l'équipe adverse d'obtenir un point si le joueur qui devine a une bonne réponse lors du droit de réplique.

3.4.1.5 (Essentielle) Le système doit terminer la ronde avec un score nul si le joueur adverse obtient une mauvaise réponse lors du droit de réplique.

3.4.1.6 (Essentielle) Le système doit terminer la partie et déclarer l'équipe vainqueur lorsqu'une équipe obtient un total de trois points.

3.4.2 (Souhaitable) Le système doit permettre à un joueur de jouer en mode sprint solo, soit de deviner un nombre maximum de dessins tracés par le joueur virtuel dans un certain laps de temps.

- 3.4.2.1 (Souhaitable) Le système doit permettre au joueur d'obtenir un point à chaque bonne réponse.
- 3.4.2.2 (Souhaitable) Le système doit permettre au joueur d'obtenir un bonus de temps à chaque bonne réponse.
- 3.4.2.3 (Souhaitable) Le système doit imposer au joueur un nombre maximal d'essais.
- 3.4.2.4 (Souhaitable) Le système doit permettre au joueur de passer à l'image suivante sans point s'il a atteint le nombre maximal d'essais sans avoir de bonne réponse.
- 3.4.2.5 (Souhaitable) Le système doit permettre au joueur de passer à l'image suivante sans bonus de temps s'il a atteint le nombre maximal d'essais sans avoir de bonne réponse.
- 3.4.3 (Souhaitable) Le système doit permettre à un joueur de collaborer avec d'autres joueurs pour deviner le dessin tracé par le joueur virtuel dans un certain laps de temps (mode sprint coopératif).
  - 3.4.3.1 (Souhaitable) Le système doit permettre à l'équipe d'obtenir un point à chaque bonne réponse.
  - 3.4.3.2 (Souhaitable) Le système doit permettre à l'équipe d'obtenir un bonus de temps à chaque bonne réponse.
  - 3.4.3.3 (Souhaitable) Le système doit imposer à l'équipe un nombre maximal d'essais.
  - 3.4.3.4 (Souhaitable) Le système doit permettre à l'équipe de passer à l'image suivante sans point si elle a atteint le nombre maximal d'essais sans avoir de bonne réponse.
  - 3.4.3.5 (Souhaitable) Le système doit permettre à l'équipe de passer à l'image suivante sans bonus de temps si elle a atteint le nombre maximal d'essais sans avoir de bonne réponse.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.5. Construction du jeu**

#### Exigences communes au client lourd et client léger

Cette fonctionnalité est offerte sur le client lourd seulement.

#### Exigences propres au client lourd

- 3.5.1 (Essentielle) Le système doit permettre à l'utilisateur de créer un jeu de manière manuelle.
  - 3.5.1.1 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec des mots ou expressions recherchés.
  - 3.5.1.2 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec des indices.
  - 3.5.1.3 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec une image en format vectoriel.
  - 3.5.1.4 (Essentielle) Le système doit permettre à l'utilisateur de tracer le dessin dans une interface pour fournir l'image qui sera utilisé par le joueur virtuel.

- 3.5.1.5 (Essentielle) Le système doit permettre à l'utilisateur de choisir un niveau de difficulté à assigner à son image.
- 3.5.1.6 (Essentielle) Le système doit générer un dessin de l'image en accéléré pour fournir un aperçu au joueur.
- 3.5.2 (Essentielle) Le système doit permettre à l'utilisateur de créer un jeu de manière assistée.
  - 3.5.2.1 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec des mots ou expressions recherchés.
  - 3.5.2.2 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec des indices.
  - 3.5.2.3 (Essentielle) Le système doit permettre à l'utilisateur de créer le jeu avec une image en format matricielle.
  - 3.5.2.1 (Essentielle) Le système doit convertir l'image matricielle en image vectorielle.
  - 3.5.2.2 (Essentielle) Le système doit permettre à l'utilisateur de configurer les paramètres de la conversion d'image.
  - 3.5.2.3 (Essentielle) Le système doit permettre à l'utilisateur de sélectionner le mode de dessin du joueur virtuel après avoir fourni l'image.
    - 3.5.2.3.1 (Essentielle) Le mode aléatoire doit afficher chaque élément du dessin dans un ordre décidé aléatoirement.
    - 3.5.2.3.2 (Essentielle) Le mode panoramique doit afficher chaque élément du dessin dans l'ordre de sa position par rapport au plan cartésien.
    - 3.5.2.3.3 (Essentielle) Le mode centré doit afficher chaque élément du dessin dans l'ordre de sa position par rapport au centre de l'image.
- 3.5.3 (Souhaitable) Le système doit permettre à l'utilisateur de créer un jeu en entrant un mot seulement.
  - 3.5.3.1 (Souhaitable) Le système doit pouvoir proposer des images correspondant au mot entré par l'utilisateur.
    - 3.5.3.1.1 (Souhaitable) Le système doit permettre à l'utilisateur de choisir entre les images proposées.
    - 3.5.3.1.2 (Souhaitable) Le système doit permettre à l'utilisateur de rafraîchir les images proposées.
- 3.5.4 (Souhaitable) Le système doit pouvoir proposer des indices liés à l'image fournie à tous les modes de jeux.
  - 3.5.4.1 (Souhaitable) Le système doit permettre à l'utilisateur de rafraîchir les indices proposés.
- 3.5.5 (Souhaitable) Le système doit permettre à l'utilisateur de créer un jeu de manière aléatoire sans entrer d'information.
  - 3.5.5.1 (Souhaitable) Le système doit générer aléatoirement un mot.
  - 3.5.5.2 (Souhaitable) Le système doit générer aléatoirement un dessin correspondant au mot généré.
  - 3.5.5.3 (Souhaitable) Le système doit générer aléatoirement des indices correspondant au mot généré.

#### Exigences propres au client léger

Cette fonctionnalité est offerte sur le client lourd seulement.

### **3.6. Joueurs virtuels**

#### Exigences communes au client lourd et client léger

- 3.6.1 (Essentielle) Le système doit pouvoir dessiner un objet à faire deviner à son coéquipier.
- 3.6.2 (Essentielle) Le système doit publier un message dans la fenêtre de clavardage à chaque début de partie.
- 3.6.3 (Essentielle) Le système doit publier un message à chaque fin de tour de l'équipe.
- 3.6.4 (Essentielle) Le système doit répondre par un indice lorsque son coéquipier lui en demande un.
- 3.6.5 (Essentielle) Le système doit pouvoir créer des joueurs virtuels ayant chacun leur propre personnalité.
  - 3.6.5.1 (Essentielle) Le système doit pouvoir créer des joueurs virtuels qui publient des messages reflétant leur personnalité.
- 3.6.6 (Souhaitable) Le système doit pouvoir envoyer des messages personnalisés au joueur,
  - 3.6.6.1 (Souhaitable) Le système doit pouvoir envoyer des messages sur des statistiques de jeu
  - 3.6.6.2 (Souhaitable) Le système doit pouvoir envoyer des messages sur l'historique des parties d'un joueur.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.7. Tutoriel**

#### Exigences communes au client lourd et client léger

- 3.7.1 (Essentielle) Chaque mode de jeu doit être accompagné d'un tutoriel qui présente une suite d'images explicatives.
  - 3.7.1.1 (Essentielle) L'utilisateur doit pouvoir accéder au tutoriel à partir du mode de jeu.
- 3.7.2 (Essentielle) Le système doit présenter un tutoriel complet au joueur lorsqu'il se connecte pour la première fois. Le tutoriel complet est composé de tous les tutoriels des modes de jeu.
- 3.7.3 (Souhaitable) Le système doit accompagner le mode de jeu classique d'un tutoriel interactif.
  - 3.7.3.1 (Souhaitable) Le système doit expliquer de manière dynamique le mode de jeu classique en faisant accomplir des actions à l'utilisateur pour qu'il puisse poursuivre le tutoriel
- 3.7.4 (Souhaitable) Le système doit accompagner le mode de jeu sprint solo d'un tutoriel interactif.
  - 3.7.4.1 (Souhaitable) Le système doit expliquer de manière dynamique le mode de jeu sprint solo en faisant accomplir des actions à l'utilisateur pour qu'il puisse poursuivre le tutoriel
- 3.7.5 (Souhaitable) Le système doit accompagner le mode de jeu sprint coopératif d'un tutoriel interactif.
  - 3.7.5.1 (Souhaitable) Le système doit expliquer de manière dynamique le mode de jeu sprint solo en faisant accomplir des actions à l'utilisateur pour qu'il puisse poursuivre le tutoriel

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.8 Expérience utilisateur**

#### Exigences communes au client lourd et client léger

3.8.1 (Souhaitable) Le système doit permettre au joueur de consulter la liste de ses amis.

3.8.2 (Souhaitable) Le système doit permettre au joueur de consulter le statut de connexion de chacun de ses amis.

3.8.3 (Souhaitable) Le système doit permettre au joueur de rechercher un autre joueur en entrant le pseudonyme de celui-ci.

3.8.4 (Souhaitable) Le système doit permettre au joueur d'envoyer une demande d'ami à un autre joueur.

3.8.4.1 (Souhaitable) Le système doit permettre au joueur de supprimer un ami de sa liste d'amis.

3.8.5 (Souhaitable) Le système doit envoyer une notification lorsqu'un joueur reçoit une demande d'ami.

3.8.5.1 (Souhaitable) Le système doit permettre au joueur d'accepter une demande d'ami.

3.8.5.2 (Souhaitable) Le système doit permettre au joueur de refuser une demande d'ami.

3.8.6 (Souhaitable) Le système doit permettre au joueur d'inviter un ami à rejoindre une partie lorsqu'il crée une partie dans le lobby de jeu.

3.8.6.1 (Souhaitable) Le système doit notifier le joueur qui se fait inviter à joindre une partie.

3.8.6.2 (Souhaitable) Le système doit permettre au joueur qui crée la partie de la lancer à tout moment.

3.8.6.3 (Souhaitable) Le système doit émettre une alerte si un joueur essaie de joindre une partie déjà commencée.

3.8.7 (Souhaitable) Le système doit permettre au joueur de partager ses résultats sur Facebook à chaque fin de partie.

3.8.7.1 (Souhaitable) Le système doit permettre au joueur de partager sa victoire sur Facebook.

3.8.7.2 (Souhaitable) Le système doit permettre au joueur de partager sa défaite sur Facebook.

3.8.7.3 (Souhaitable) Le système doit permettre au joueur de partager son dessin sur Facebook.

3.8.7.4 (Souhaitable) Le système doit rediriger le joueur vers le site Web de Facebook.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

### **3.9. Effets visuels et sonores**

#### Exigences communes au client lourd et client léger

- 3.9.1 (Essentielle) Le système doit jouer un son de victoire à la fin de chaque partie remportée.
- 3.9.2 (Essentielle) Le système doit jouer un son de défaite à la fin de chaque partie perdue.
- 3.9.3 (Essentielle) Le système doit jouer une animation visuelle de victoire à la fin de chaque partie remportée.
- 3.9.4 (Essentielle) Le système doit jouer une animation visuelle de défaite à la fin de chaque partie perdue.
- 3.9.5 (Souhaitable) Le système doit afficher une notification de type popup lorsqu'un ami se connecte au jeu.

#### Exigences propres au client lourd

- 3.9.6 (Souhaitable) Le système doit afficher un effet de disparition et apparition en fondu à chaque changement de page de l'application.

#### Exigences propres au client léger

- 3.9.7 (Souhaitable) Le système doit émettre une vibration lorsque le temps alloué pour une partie est écoulé.

### **3.10 Expérience de jeu**

#### Exigences communes au client lourd et client léger

- 3.10.1 (Souhaitable) Le système doit permettre au joueur de consulter le classement global des joueurs sur le tableau des leaders.
- 3.10.1 (Souhaitable) Le système doit permettre au joueur de consulter son propre classement dans sa page de profil.
- 3.10.2 (Souhaitable) Le système doit empêcher le joueur qui dessine de tricher en écrivant le mot qu'il doit faire deviner plutôt que de le dessiner.
  - 3.10.2.1 (Souhaitable) Le système doit détecter si une lettre est écrite par le joueur.
  - 3.10.2.2 (Souhaitable) Le système doit avertir le joueur qu'une lettre a été détectée.
    - 3.10.2.2.1 (Souhaitable) Le système doit détecter si une lettre additionnelle est écrite par le joueur.
    - 3.10.2.2.2 (Souhaitable) Le système doit retirer un point au joueur à chaque lettre additionnelle détectée qui est présente dans le mot à faire deviner.

#### Exigences propres au client lourd

Aucune exigence spécifique au client lourd.

#### Exigences propres au client léger

Aucune exigence spécifique au client léger.

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

### **4.1. Utilisabilité**

- 4.1.1 Le temps de formation nécessaire à un utilisateur régulier doit être inférieur à 15 minutes.
- 4.1.2 Le temps de formation nécessaire à un superutilisateur doit être inférieur à 10 minutes.
- 4.1.3 Le nombre d'erreurs d'utilisation doit être inférieur à 5 par heure d'utilisation pour un utilisateur régulier.
- 4.1.4 Le nombre d'erreurs d'utilisation doit être inférieur à 3 par heure d'utilisation pour un superutilisateur.
- 4.1.5 Le temps nécessaire à un utilisateur pour créer un jeu en mode manuel doit être inférieur à 2 minutes.
- 4.1.6 Le temps nécessaire à un utilisateur pour créer un jeu en mode assisté doit être inférieur à 1 minute.

### **4.2. Fiabilité**

- 4.2.1 Le système doit être disponible au moins 99% du temps.
- 4.2.2 Le temps moyen entre les pannes doit être au moins de 1 mois.
- 4.2.3 Le temps moyen jusqu'à la réparation après une panne doit être au plus de 5 heures.

### **4.3. Performance**

- 4.3.1 Le serveur doit permettre la connexion simultanée d'au moins 8 utilisateurs.
- 4.3.2 Le temps de réponse pour l'authentification au serveur doit être inférieur à 5 secondes.
- 4.3.3 Le temps de réponse pour le chargement d'une partie doit être inférieur à 3 secondes.
- 4.3.4 Le temps de réponse pour le téléversement d'une image doit être inférieur à 5 secondes.
- 4.3.5 Le temps de réponse pour l'envoi d'un message dans le clavardage doit être inférieur à 2 secondes.

### **4.4. Maintenabilité**

- 4.4.1 Le code source doit être écrit en anglais.
- 4.4.2 Les conventions de nommage pour le code source doivent respecter les meilleures pratiques en vigueur pour chaque langage de programmation utilisé.
- 4.4.3 Le code source doit inclure des commentaires explicatifs.
- 4.4.4 Tous les développeurs affectés au projet doivent avoir accès au code source de manière à pouvoir en effectuer la maintenance.

### **4.5. Contraintes de conception**

- 4.5.1 Le client lourd doit être écrit avec le langage C# en utilisant WPF.
- 4.5.2 Le client léger doit être écrit avec le langage Kotlin.
- 4.5.3 Le serveur doit être implémenté avec .NET Core.



#### **4.6. Sécurité**

4.6.1 La connexion à chaque profil utilisateur doit exiger le bon mot de passe.

4.6.2 Les mots de passe des utilisateurs doivent être cryptés à l'aide d'une fonction de hachage.

#### **4.7. Exigences de la documentation usager en ligne et du système d'assistance**

4.7.1 Le système doit être accompagné d'une documentation pour les utilisateurs.

4.7.1.1 La documentation doit expliquer de manière succincte les principaux modules de l'application.

4.7.1.2 La documentation doit pouvoir être comprise facilement par un utilisateur régulier.

4.7.2 Les développeurs peuvent être contactés par courriel pour répondre à des demandes de soutien applicatif.