

Fais-moi un dessin
Spécifications des requis du système (SRS)

Version 1.7

Historique des révisions

Date	Version	Description	Auteur
2021-01-25	1.0	Rédaction de l'introduction et de la description globale	Guillaume Thibault
2021-01-27	1.1	Rédaction des exigences fonctionnelles	Équipe 207
2021-01-29	1.2	Rédaction des exigences non-fonctionnelles	Équipe 207
2021-02-1	1.3	Révision du document	Guillaume Thibault et Marie-Ève Patron
2021-02-02	1.4	Révision du document et correction des erreurs de langue	Julien Witty
2021-02-03	1.5	Révision des différences entre les exigences du client lourd et celles du client léger	Samuel Ouvrard
2021-02-03	1.6	Correction d'exigences relatives à la devinette des dessins	Marc-Olivier Riopel
2021-02-19	1.7	Correction selon commentaires du correcteur	Guillaume Thibault, Samuel Ouvrard et Marie-Ève Patron

Table des matières

1. Introduction	5
1.1. But	5
1.2. Définitions, acronymes et abréviations	5
1.3. Vue d'ensemble du document	6
2. Description globale	7
2.1. Caractéristiques des usagers	7
2.2. Interfaces	7
2.2.1. Interfaces usagers	7
2.2.2. Interfaces matérielles	7
2.2.3. Interfaces logicielles	7
2.2.4. Interfaces de communication	8
2.3. Contraintes générales	8
2.4. Hypothèses et dépendances	8
3. Exigences fonctionnelles	9
3.1. Client lourd	9
3.1.1. Clavardage	9
3.1.1.1 Clavardage - Intégration	9
3.1.1.2. Clavardage - Canaux de discussion	9
3.1.2 Profil utilisateur et historique	9
3.1.3. Modes de jeu	10
3.1.4. Création d'une paire mot-image	13
3.1.5. Personnalité des joueurs virtuels	14
3.1.6. Effets visuels et sonores	14
3.1.7. Tutoriel	15
3.1.7.3. Tutoriel (essentiel)	15
3.1.7.4. Tutoriel interactif (souhaitable).	15
3.1.8. Tableau de classement	15
3.1.9. Suggestion de mots	15
3.1.10. Téléversement d'un dessin d'une partie vers le joueur virtuel	15
3.1.11. Partie privée avec identifiant de la partie	16
3.1.12. Vote sur dessins du joueur virtuel	16
3.1.13. Liens d'amitié	17
3.1.14. Fil d'image des parties jouées	17
3.1.15. Rejoindre ses amis	17
3.1.16. Invitation des amis à une partie privée	17
3.2. Client léger	18
3.2.1. Clavardage	18
3.2.1.1. Clavardage - Intégration	18
3.2.1.2. Clavardage - Canaux de discussion	18
3.2.2. Profil utilisateur et historique	18
3.2.3. Modes de jeu	18

3.2.4. Effets visuels et sonores	18
3.2.5. Tutoriel	18
3.2.6. Suggestion de mots	18
3.2.7. Téléversement d'un dessin d'une partie vers la banque de dessins des joueurs virtuels.	18
3.2.8. Partie privée avec identifiant de la partie	18
3.2.9. Vote sur dessins du joueur virtuel	18
3.2.10. Liens d'amitié	18
3.2.11. Rejoindre ses amis	19
3.2.12. Invitation des amis a une partie privée	19
4. Exigences non-fonctionnelles	20
4.1. Utilisabilité	20
4.2. Fiabilité	20
4.3. Performance	20
4.4. Maintenabilité	21
4.5. Contraintes de conception	21
4.6. Sécurité	21
4.7. Exigences de la documentation usager en ligne et du système d'assistance	21
ANNEXE A: Nomenclature des versions	23

Spécifications des requis du système (SRS)

1. Introduction

1.1. But

Le SRS décrit le comportement externe d'une application. Il décrit aussi les exigences 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

- Android: Système d'exploitation mobile fondé sur le noyau Linux.
- Angular: *Framework* permettant de faire des applications web basées sur une seule page.
- API : Application Programming Interface permet à deux applications de communiquer ensemble.
- Atlas: Service infonuagique de la base de données MongoDB.
- Mb/s: Mégabit par seconde, correspondant à 1 000 000 bits par seconde.
- Client lourd: Version sur ordinateur de notre application.
- Client léger: Version sur mobile de notre application.
- CSS: Cascading style sheets est un langage décrivant comment les éléments HTML sont représentés.
- Electron: *Framework* permettant de créer une application de bureau à partir d'une application web.
- Endpoints: Extrémité d'un canal de communication HTTP.
- Express: Cadriciel pour construire des applications web basées sur Node.js.
- Framework: Cadriciel facilitant le développement logiciel.
- HTML: Hypertext Markup Language est un langage pour structurer des documents web.
- HTTP: HyperText Transfer Protocol est un protocole de communication permettant la transmission de documents hypermédias comme HTML.
- HTTPS: HyperText Transfer Protocol Secure est une extension de HTTP avec de l'information cryptée.
- Joueur réel: Un humain jouant au jeu.
- Joueur virtuel: Faux joueur simulé par le serveur.
- JSON: Format de données textuelles dérivé de la notation des objets du langage JavaScript.
- JSON Web Token : Méthode pour certifier l'accès à un utilisateur.
- Kotlin: Langage de programmation permettant de développer des applications mobiles sous Android.
- Lobby: Groupe de joueurs rassemblés ensemble pour jouer à un jeu.
- Manche de partie classique: Moment de la partie lorsqu'une équipe dessine et l'autre équipe est en attente. Dans l'équipe qui dessine un joueur a le rôle de dessiner et l'autre a le rôle de deviner.
- MongoDB: Système de gestion de base de données orientée documents.
- Node.js: Plateforme logicielle libre en JavaScript pour la création d'application réseau événementielle.
- NoSQL: Type de base de données s'éloignant du paradigme classique des bases relationnelles.
- Outil annuler: Outil permettant d'enlever la dernière action fait dans un canevas

- Outil crayon: Outil permettant de faire des traits dans un canevas
- Outil grille: Outil permettant de faire apparaître et disparaître une grille qui est superposé sur un canevas
- Outil refaire: Outil permettant de remettre la dernière action enlevée dans un canevas
- Paire mot-image: Une image et un mot permettant d'être utilisé par un joueur virtuel
- RESTful: Style d'architecture logicielle définissant un ensemble de contraintes à utiliser pour créer des services web.
- Sockets: Type de terminal logiciel qui fonctionne pour établir une liaison réseau bidirectionnelle entre un serveur et un client.
- Typescript : Langage de programmation issu du JavaScript permettant d'avoir du code mieux structuré.
- TCP: Transmission Control protocol est un protocole de communication.
- Tour de partie classique: Deux manches d'une partie classique
- Upvote d'un dessin: Somme des votes positifs moins les votes négatifs sur les paires mot/image soumises pour les joueurs virtuels.
- XML : Type de langage déclaratif semblable au HTML.

1.3. Vue d'ensemble du document

Dans la prochaine section, une description globale du logiciel est fournie afin d'avoir la vue d'ensemble du projet. Par la suite, la section 3 est dédiée aux exigences fonctionnelles qui sont les fonctions du système à développer. Finalement, la dernière section décrit les exigences non-fonctionnelles décrivant les propriétés du système.

2. Description globale

Le projet consiste à créer un logiciel permettant à ses utilisateurs de jouer à un jeu en réseau consistant à deviner un dessin que l'autre joueur fait. Le logiciel est composé de trois sous-logiciels, soit un serveur, un client lourd et un client léger.

2.1. Caractéristiques des usagers

Les usagers visés sont des étudiants en génie logiciel. Ces personnes possèdent un ordinateur ou une tablette Android. Ces personnes, ayant accès à ces outils informatiques, sont habituées à utiliser des applications ou des logiciels et ils ont une bonne connaissance de la signification des icônes et des interfaces utilisées dans des jeux.

2.2. Interfaces

2.2.1. Interfaces usagers

Pour ce projet, trois interfaces usagers seront développées. Le premier est pour le client lourd. Pour celui-ci, nous utiliserons le projet Poly Dessin développé en Angular, Typescript, HTML et CSS et l'adapterons pour convenir aux exigences de ce nouveau projet. Le client lourd fonctionnera sous Windows 10 ainsi que MacOS Big Sur. La deuxième interface est pour le client léger, celle-ci sera développée en Kotlin. Le client léger devra minimalement fonctionner fluidement sur une tablette Samsung Galaxy Tab A 2019 utilisant Android Pie. Le serveur sera développé en Node.js en utilisant le *framework* Express. Ce serveur pourra être utilisé à l'aide d'un terminal.

2.2.2. Interfaces matérielles

L'utilisateur aura besoin de différents matériels selon la plateforme qu'il désire utiliser. Celui-ci a le choix entre l'utilisation d'une tablette Android supportant le touché tactile et comportant un clavier intégré ou d'un ordinateur comportant un moniteur, une souris et un clavier. De plus, un système sonore est recommandé sur les deux plateformes pour que l'utilisateur puisse entendre les effets sonores.

2.2.3. Interfaces logicielles

Pour le développement du client lourd, le *framework* Electron sera utilisé. Celui-ci sert à développer une application de bureau en utilisant des technologies web. La technologie web utilisée pour le développement sera le *framework* Angular ce qui permet la réutilisation d'une partie du code du projet Poly Dessin. L'application sur ordinateur utilisera le système d'exploitation Windows 10, car celui-ci est le système d'exploitation le plus utilisé par l'audience visée. Pour le développement du client léger, le langage de programmation Kotlin sera utilisé sur Android Studio. Le serveur développé en Node.js à l'aide d'Express sera hébergé sur Atlas. Les données seront sauvegardées dans une base de données de type NoSQL sur MongoDB et celles-ci pourront être accédées à l'aide de requêtes, mais aussi grâce à une interface web fournie par MongoDB.

2.2.4. Interfaces de communication

Le client lourd et le client léger se connecteront au serveur via internet et wifi. Un des types de communication générale qui sera utilisé par l'application est TCP. Ce type de communication est utilisé par les requêtes HTTP et HTTPS. Un autre type de communication qui sera utilisé est les sockets. Ce sont des points terminaux de communication à deux sens entre deux programmes. Grâce aux sockets utilisés au travers de la librairie socket.io, les utilisateurs de l'application pourront se connecter avec le serveur et ainsi échanger des informations en temps réel avec celui-ci.

2.3. Contraintes générales

Une contrainte majeure que le système doit comporter est que plusieurs joueurs puissent jouer ensemble de façon simultanée et que tout changement soit synchronisé pour les joueurs connectés dans une salle d'attente (lobby). Le nombre de joueurs à supporter au maximum dans une salle d'attente est de 4. De plus, on doit pouvoir supporter minimalement 4 salles d'attente fonctionnant en même temps.

2.4. Hypothèses et dépendances

Tout d'abord, on suppose que le serveur est hébergé et accessible au moment de l'utilisation des applications. De plus, on suppose que l'utilisateur a accès à une connexion internet fiable et haute vitesse, soit ayant un débit supérieur à 8Mb/s à partir de son ordinateur ou de sa tablette Android. Finalement on présume que la tablette et l'ordinateur sont assez à jour pour supporter la version de leur plateforme respective.

3. Exigences fonctionnelles

3.1. Client lourd

3.1.1. Clavardage

3.1.1.1 Clavardage - Intégration

- 3.1.1.1.1. Le système doit permettre à l'utilisateur d'avoir accès en tout temps à une boîte de texte dans une fenêtre séparée pour le clavardage (essentiel).
- 3.1.1.1.2. Le système doit permettre à l'utilisateur d'avoir accès en tout temps au clavardage à même l'environnement de l'application (essentiel).
- 3.1.1.1.3. Le système doit permettre à l'utilisateur d'alterner entre le mode de clavardage intégré et celui à même l'environnement (essentiel).
- 3.1.1.1.4. Le système doit permettre à l'utilisateur de recevoir une notification lors de la réception de nouveaux messages s'il se trouve dans l'application (essentiel).

3.1.1.2. Clavardage - Canaux de discussion

- 3.1.1.2.1. Le système doit permettre à l'utilisateur d'accéder à un clavardage qui regroupe tous les utilisateurs (essentiel).
- 3.1.1.2.2. Le système doit permettre à l'utilisateur de créer des canaux de discussion en choisissant le nom du canal (essentiel).
- 3.1.1.2.3. Le système doit permettre à l'utilisateur d'accéder à des canaux de discussion en le choisissant dans une liste (essentiel).
- 3.1.1.2.4. Le système doit permettre à l'utilisateur de supprimer un canal de discussion en le choisissant dans une liste si personne ne se trouve dans le canal à ce moment-là (essentiel).
- 3.1.1.2.5. Le système doit permettre à l'utilisateur de décider d'afficher l'historique de clavardage complet du canal qui est normalement caché par défaut (essentiel).
- 3.1.1.2.6. Le système doit permettre à l'utilisateur de quitter un canal de discussion qu'il a préalablement accédé (essentiel).

3.1.2 Profil utilisateur et historique

- 3.1.2.1. Le système doit permettre à l'utilisateur de s'authentifier avec un pseudonyme et le mot de passe correspondant pour se connecter au serveur de l'application (essentiel).
- 3.1.2.2. Le système doit permettre à l'utilisateur de se créer un compte s'il n'en possède pas (essentiel).
- 3.1.2.3. Le système doit obliger l'utilisateur à fournir un nom lors de la création d'un compte (essentiel).
- 3.1.2.4. Le système doit obliger l'utilisateur à fournir un prénom lors de la création d'un compte (essentiel).
- 3.1.2.5. Le système doit obliger l'utilisateur à choisir un avatar lors de la création d'un compte (essentiel).
- 3.1.2.6. Le système doit obliger l'utilisateur à fournir un pseudonyme lors de la création d'un compte (essentiel).
- 3.1.2.7. Le système doit obliger l'utilisateur à fournir un mot de passe lors de la création d'un compte (essentiel).

- 3.1.2.8. Le système doit obliger l'utilisateur à confirmer son mot de passe lors de la création d'un compte (essentiel).
- 3.1.2.9. Le système doit permettre à l'utilisateur de consulter son nom dans son propre profil (essentiel).
- 3.1.2.10. Le système doit permettre à l'utilisateur de consulter son prénom dans son propre profil (essentiel).
- 3.1.2.11. Le système doit permettre à l'utilisateur de consulter le pseudonyme d'un joueur dans le profil de celui-ci (essentiel).
- 3.1.2.11. Le système doit permettre à l'utilisateur de consulter l'avatar d'un joueur dans le profil de celui-ci (essentiel).
- 3.1.2.12. Le système doit permettre à l'utilisateur de consulter le meilleur score en sprint solo d'un joueur dans le profil de celui-ci (essentiel).
- 3.1.2.13. Le système doit permettre à l'utilisateur de consulter le meilleur score en sprint coop d'un joueur dans le profil de celui-ci (essentiel).
- 3.1.2.14. Le système doit permettre à l'utilisateur de consulter le ratio partie classique gagnée/perdue d'un joueur dans le profil de celui-ci (essentiel).
- 3.1.2.15. Le système doit permettre à l'utilisateur de consulter son temps total passé sur l'application dans son propre profil (essentiel).
- 3.1.2.16. Le système doit permettre à l'utilisateur de consulter son temps moyen d'une partie dans son propre profil (essentiel).
- 3.1.2.17. Le système doit permettre à l'utilisateur de consulter un relevé de ses connexions dans son propre profil (essentiel).
- 3.1.2.18. Le système doit permettre à l'utilisateur de consulter un relevé de ses déconnexions dans son propre profil (essentiel).
- 3.1.2.19. Le système doit permettre à l'utilisateur de consulter un relevé de l'heure de ses parties jouées dans son propre profil (essentiel).
- 3.1.2.20. Le système doit permettre à l'utilisateur de consulter un relevé des dates de ses parties jouées dans son propre profil (essentiel).
- 3.1.2.21. Le système doit permettre à l'utilisateur de consulter un relevé des joueurs présents lors de ses parties jouées dans son propre profil (essentiel).
- 3.1.2.22. Le système doit permettre à l'utilisateur de consulter un relevé du mode de ses parties jouées dans son propre profil (essentiel).

3.1.3. Modes de jeu

- 3.1.3.1. Le système doit permettre à l'utilisateur d'avoir accès à un menu principal (essentiel).
- 3.1.3.2. Le système doit permettre à l'utilisateur de créer une partie à partir du menu principal (essentiel).
- 3.1.3.3. Le système doit obliger l'utilisateur de choisir un mode de jeu lors de la création d'une partie (essentiel).
- 3.1.3.4. Le système doit obliger l'utilisateur de choisir une difficulté lors de la création d'une partie (essentiel).
- 3.1.3.5. Le système doit permettre à l'utilisateur de consulter la liste des parties disponibles à partir du menu principal (essentiel).
- 3.1.3.6. Le système doit permettre à l'utilisateur de se joindre à une des parties disponibles de la

liste des parties disponibles (essentiel).

3.1.3.7. Le système doit permettre à l'utilisateur de filtrer la liste des parties disponibles en fonction du mode de jeu (essentiel).

3.1.3.8. Le système doit permettre à l'utilisateur de filtrer la liste des parties disponibles en fonction de la difficulté (essentiel).

3.1.3.9. Le système doit permettre à l'utilisateur de consulter la liste des joueurs dans une partie disponible (essentiel).

3.1.3.10. Le système doit permettre à l'utilisateur d'ajouter des joueurs virtuels à une partie disponible tout en respectant les règles des différents modes de jeu (essentiel).

3.1.3.11. Le système doit permettre à l'utilisateur de retirer des joueurs virtuels à une partie disponible tout en respectant les règles des différents modes de jeu (essentiel).

3.1.3.12. Le système doit créer un nouveau canal de discussion lors de la création d'une nouvelle partie (essentiel).

3.1.3.13. Le système doit ajouter l'utilisateur au canal de discussion de la partie lorsqu'il rejoint une partie disponible (essentiel).

3.1.3.14. Le système doit permettre à l'utilisateur de rejoindre la salle d'attente d'une partie disponible (essentiel).

3.1.3.15. Le système doit amener l'utilisateur dans la salle d'attente lorsqu'il rejoint une partie disponible (essentiel).

3.1.3.16. Le système doit permettre à l'utilisateur de quitter la salle d'attente pour retourner au menu principal (essentiel).

3.1.3.17. Le système doit permettre à tous les utilisateurs dans la salle d'attente de démarrer la partie lorsque le nombre minimum de joueurs est atteint (essentiel).

3.1.3.17. Mode classique

3.1.3.17.1. Le système doit permettre à l'utilisateur d'ajouter un maximum de deux joueurs virtuels dans la salle d'attente (essentiel).

3.1.3.17.2. Le système doit permettre à l'utilisateur d'enlever les joueurs virtuels dans la salle d'attente (essentiel).

3.1.3.17.3. Le système doit former deux équipes de deux aléatoirement au début de la partie (essentiel).

3.1.3.17.4. Le système doit s'assurer que chaque équipe de deux contient au maximum un joueur virtuel au début de la partie (essentiel).

3.1.3.17.5. Le système doit aléatoirement choisir quelle équipe commence au début de la partie (essentiel).

3.1.3.17.6. Le système doit aléatoirement choisir quel joueur dessine en premier dans l'équipe qui commence (essentiel).

3.1.3.17.7. Le système doit allouer au joueur qui devine un essai pour deviner le mot ou l'expression (essentiel).

3.1.3.17.8. Le système doit allouer au joueur qui devine un maximum de temps pour deviner le mot ou l'expression en fonction de la difficulté de la partie (essentiel).

3.1.3.17.9. Le système doit faire gagner un point à l'équipe qui dessine si le joueur qui devine trouve le bon mot ou expression (essentiel).

3.1.3.17.10. Le système doit allouer à l'autre équipe un essai pour deviner le mot ou

l'expression si le joueur qui devine soumet le mauvais mot ou expression ou que le temps est écoulé (essentiel).

3.1.3.17.11. Le système doit allouer à l'autre équipe un maximum de 10 secondes pour deviner le mot ou l'expression (essentiel).

3.1.3.17.12. Le système doit faire gagner un point à l'équipe en attente si le joueur de cette équipe devine le mot ou l'expression (essentiel).

3.1.3.17.13. Le système doit n'attribuer aucun point si aucune des équipes ne devine le mot ou l'expression correctement(essentiel).

3.1.3.17.14. Le système doit inverser le rôle des équipes à la fin de chaque manche (essentiel).

3.1.3.17.15. Le système doit alterner le rôle des membres de l'équipe qui dessine à chaque tour (essentiel).

3.1.3.17.16. Le système doit toujours donner le rôle de celui qui devine à l'utilisateur s'il joue avec un joueur virtuel lorsque leur équipe est celle qui dessine(essentiel).

3.1.3.17.17. Le système doit permettre à l'utilisateur d'avoir accès à la surface de dessin lorsqu'il dessine (essentiel).

3.1.3.17.18. Le système doit permettre à l'utilisateur d'avoir accès à l'outil Crayon lorsqu'il dessine (essentiel).

3.1.3.17.19. Le système doit permettre à l'utilisateur d'avoir accès à l'outil Efface lorsqu'il dessine (essentiel).

3.1.3.17.20. Le système doit permettre à l'utilisateur d'avoir accès à l'outil Annuler lorsqu'il dessine (essentiel).

3.1.3.17.21. Le système doit permettre à l'utilisateur d'avoir accès à l'outil Refaire lorsqu'il dessine (essentiel).

3.1.3.17.22. Le système doit permettre à l'utilisateur d'avoir accès à l'outil Grille lorsqu'il dessine (essentiel).

3.1.3.17.23. Le système doit permettre à l'utilisateur de changer la couleur de l'outil Crayon lorsqu'il dessine (essentiel).

3.1.3.17.24. Le système doit permettre à l'utilisateur de changer la taille de l'outil Crayon lorsqu'il dessine (essentiel).

3.1.3.17.25. Le système doit permettre à l'utilisateur de changer la taille de l'outil Efface lorsqu'il dessine (essentiel).

3.1.3.17.26. Le système doit permettre aux autres joueurs de voir apparaître en temps réel le dessin du joueur qui dessine (essentiel).

3.1.3.17.27. Le système doit permettre au joueur qui a le rôle de deviner le mot ou l'expression à tout moment (essentiel).

3.1.3.18. Sprint solo

3.1.3.18.1. Le système doit permettre à l'utilisateur de deviner un maximum de mots ou d'expressions à partir de dessins réalisés par un joueur virtuel (essentiel).

3.1.3.18.2. Le système doit allouer à l'utilisateur 3 minutes pour deviner un maximum de mots et expressions (essentiel).

3.1.3.18.3. Le système doit allouer à l'utilisateur un laps de temps limité en fonction de la difficulté de la partie pour deviner chaque mot ou expression (essentiel).

3.1.3.18.4. Le système doit allouer à l'utilisateur un nombre limité d'essais pour deviner

le mot ou l'expression en fonction de la difficulté de la partie (essentiel).

3.1.3.18.5. Le système doit permettre à l'utilisateur de voir le temps total de la partie restant à tout moment pendant la partie (essentiel).

3.1.3.18.6. Le système doit permettre à l'utilisateur de voir le temps restant pour deviner le mot ou l'expression à tout moment pendant la partie (essentiel).

3.1.3.18.7. Le système doit permettre à l'utilisateur de voir le nombre d'essais restants pour deviner le mot ou l'expression à tout moment pendant la partie (essentiel).

3.1.3.18.8. Le système doit permettre à l'utilisateur de voir son score à tout moment pendant la partie (essentiel).

3.1.3.18.9. Le système doit donner un point à l'utilisateur pour chaque bonne réponse (essentiel).

3.1.3.18.10. Le système doit ajouter du temps au compte à rebours total de la partie à chaque bonne réponse en fonction de la difficulté (essentiel).

3.1.3.18.11. Le système doit passer à l'image suivante si l'utilisateur n'a plus d'essais pour deviner le mot ou l'expression (essentiel).

3.1.3.18.12. Le système doit passer à l'image suivante si l'utilisateur n'a plus de temps pour deviner le mot ou l'expression (essentiel).

3.1.3.18.13. Le système doit arrêter la partie lorsque le compte à rebours total de la partie atteint zéro (essentiel).

3.1.3.18.14. Le système doit afficher le score final de la partie lorsque celle-ci est terminée (essentiel).

3.1.3.19. Sprint coopératif

3.1.3.19.1. Le système doit permettre à l'utilisateur de se joindre à un maximum de trois autres joueurs humains pour ce mode de jeu (essentiel).

3.1.3.19.2. Le système doit permettre aux joueurs de deviner un maximum de mots ou d'expressions à partir de dessins réalisés par un joueur virtuel (essentiel).

3.1.3.19.3. Le système doit allouer aux joueurs un maximum collectif d'essais défini selon la difficulté (essentiel).

3.1.3.19.4 à 3.1.3.19.15. Les exigences 3.1.3.18.2 à 3.1.3.18.14. à l'exception de l'exigence 3.1.3.18.4 du sprint solo s'appliquent aussi au sprint coop (essentiel).

3.1.4. Création d'une paire mot-image

3.1.4.1. Le système doit permettre à l'utilisateur d'avoir accès à une interface de création d'une paire mot-image à partir du menu principal (essentiel).

3.1.4.2. Le système doit obliger l'utilisateur d'entrer le mot ou l'expression qu'il va dessiner (essentiel).

3.1.4.3. Le système doit permettre à l'utilisateur de dessiner le mot ou l'expression dans une zone de dessin (essentiel).

3.1.4.4. Le système doit permettre à l'utilisateur de simuler la création de son dessin en accéléré par un joueur virtuel pour lui donner un aperçu du résultat final lors d'une partie (essentiel).

3.1.4.5. Le système doit obliger l'utilisateur d'ajouter au minimum un indice (essentiel).

3.1.4.6. Le système doit permettre à l'utilisateur d'ajouter autant d'indices qu'il le désire dans un champ d'entrées dynamique (essentiel).

- 3.1.4.7. Le système doit permettre à l'utilisateur de décider d'un niveau de difficulté pour son dessin (essentiel).
- 3.1.4.8. Le système doit prendre en compte le niveau de difficulté de la paire mot-image pour allouer le nombre d'essais pour deviner (essentiel).
- 3.1.4.9. Le système doit prendre en compte le niveau de difficulté de la paire mot-image pour déterminer la vitesse d'apparition du dessin (essentiel).
- 3.1.4.10. Le système doit prendre en compte le niveau de difficulté de la paire mot-image pour déterminer le temps alloué pour deviner le dessin (essentiel).

3.1.5. Personnalité des joueurs virtuels

- 3.1.5.1. Le système doit permettre à l'utilisateur de recevoir des messages de la part de joueurs virtuels avec des personnalités variées pendant la partie (essentiel).
- 3.1.5.2. Le système doit permettre à l'utilisateur de recevoir des messages de la part des joueurs virtuels faisant référence à des parties déjà jouées (essentiel).
- 3.1.5.3. Le système doit permettre à l'utilisateur de recevoir des messages personnalisés de la part de joueurs virtuels faisant référence à son nombre de parties gagnées (essentiel).
- 3.1.5.4. Le système doit permettre à l'utilisateur de recevoir des messages de la part des joueurs virtuels faisant référence à son nombre de parties jouées (essentiel).
- 3.1.5.5. Le système doit permettre à l'utilisateur de recevoir des messages de la part des joueurs virtuels faisant référence à son meilleur score de sprint solo (essentiel).
- 3.1.5.6. Le système doit permettre à l'utilisateur de recevoir des messages de la part des joueurs virtuels faisant référence à son meilleur score de sprint coopératif (essentiel).
- 3.1.5.7. Le système doit permettre à l'utilisateur de recevoir des indices de la part de joueurs virtuels lorsque demandé (essentiel).
- 3.1.5.8. Le système doit permettre à l'utilisateur de recevoir des félicitations de la part de joueurs virtuels (essentiel).
- 3.1.5.9. Le système doit permettre à l'utilisateur de recevoir des encouragements de la part de joueurs virtuels (essentiel).
- 3.1.5.10. Le système doit allouer une personnalité aléatoire à chaque joueur virtuel entre gentil, anxieux, passif agressif et compétitif (essentiel).

3.1.6. Effets visuels et sonores

- 3.1.6.1. Le système doit permettre à l'utilisateur de percevoir une indication sonore de chronomètre lorsqu'il ne reste plus beaucoup de temps pour deviner le dessin (essentiel).
- 3.1.6.2. Le système doit permettre à l'utilisateur de percevoir le décompte de début de partie pour le mode de jeu sprint «3, 2, 1, GO » (essentiel).
- 3.1.6.3. Le système doit permettre à l'utilisateur de percevoir une indication sonore de « bip, bip bip heiiiiiiiiin! » pour le décompte de début de partie pour le mode de jeu sprint (essentiel).
- 3.1.6.4. Le système doit permettre à l'utilisateur de percevoir une indication sonore lorsqu'il gagne (essentiels).
- 3.1.6.5. Le système doit permettre à l'utilisateur de percevoir une indication sonore lorsqu'il Aaaawwwwww! (essentiel).

3.1.7. Tutoriel

3.1.7.1.1. Le système doit obliger l'utilisateur de suivre le tutoriel lors de la première connexion à l'application (essentiel).

3.1.7.1.2. Le système doit permettre à l'utilisateur de revoir le tutoriel en tout temps (essentiel).

3.1.7.3. Tutoriel (essentiel)

3.1.7.3.1. Le système doit permettre à l'utilisateur d'avoir accès à une suite d'images explicatives pour les différents modes de jeu de l'application.

3.1.7.3.2. Le système doit permettre à l'utilisateur de passer à l'image suivante pour continuer le tutoriel.

3.1.7.4. Tutoriel interactif (souhaitable).

3.1.7.4.1. Le système doit permettre à l'utilisateur de suivre un tutoriel interactif à la première utilisation de l'application

3.1.7.4.2. Le système doit permettre à l'utilisateur de sauter le diapositive expliquant les fonctionnalités de l'application.

3.1.8. Tableau de classement

3.1.8.1. Le système doit permettre à l'utilisateur de consulter la liste des joueurs ayant le plus de parties gagnées en mode classique (essentiel).

3.1.8.2. Le système doit permettre à l'utilisateur de consulter la liste des joueurs ayant le plus de parties jouées (essentiel).

3.1.8.3. Le système doit permettre à l'utilisateur de consulter la liste des joueurs ayant le meilleur score pour les sprints solos (essentiel).

3.1.8.4. Le système doit permettre à l'utilisateur de consulter la liste des joueurs ayant le meilleur score pour les sprints coopératifs (essentiel).

3.1.8.5. Le système doit permettre à l'utilisateur de consulter la liste des joueurs ayant le plus de upvotes cumulatifs sur les paires mot-image (essentiel). **voir les exigences 3.1.12 pour comprendre le système de vote sur les paires mot-image.

3.1.9. Suggestion de mots

3.1.9.1. Le système doit afficher une suggestion de trois mots ou expressions au joueur qui dessine au début de son tour lors d'une partie classique (essentiel).

3.1.9.1. Le système doit permettre à l'utilisateur de sélectionner un mot à dessiner à partir des 3 mots qui lui sont suggérés dans le mode de jeu classique (essentiel).

3.1.9.1. Le système doit permettre à l'utilisateur de rafraîchir au maximum 3 fois cette suggestion de mots (essentiel).

3.1.9.1. Le système doit empêcher le joueur de revoir les mots déjà suggérés.

3.1.10. Téléversement d'un dessin d'une partie vers le joueur virtuel

3.1.10.1. Le système doit permettre à l'utilisateur de choisir un de ses propres dessins à la fin d'une partie classique dans le but de l'ajouter à la banque de dessin des joueurs virtuels (essentiel).

- 3.1.10.2. Le système doit envoyer l'utilisateur à la page de création d'une paire mot-image (essentiel).
- 3.1.10.3. Le système doit automatiquement remplir le champ dessin avec le dessin préalablement choisi (essentiel).
- 3.1.10.4. Le système doit automatiquement remplir le champ mot ou dessin avec le mot ou l'expression utilisée pour dessiner l'image (essentiel).
- 3.1.10.5. Le système doit obliger l'utilisateur à remplir les autres champs nécessaires à la création d'une paire mot-image avant de la soumettre ****références aux exigences 3.1.4.5. à 3.1.4.10**** (essentiel).
- 3.1.10.6. Le système doit permettre à l'utilisateur de retourner au menu de fin de partie en annulant la création de la paire mot-image (essentiel).
- 3.1.10.7. Le système doit retourner l'utilisateur au menu de fin de partie après le téléversement de la paire mot-image (essentiel).

3.1.11. Partie privée avec identifiant de la partie

- 3.1.11.1. Le système doit permettre à l'utilisateur de créer une partie privée pour le mode classique (souhaitable).
- 3.1.11.2. Le système doit permettre à l'utilisateur de créer une partie privée pour le mode sprint coopératif (souhaitable).
- 3.1.11.3. Le système doit permettre à l'utilisateur de voir l'identifiant de la partie privée après sa création (souhaitable).
- 3.1.11.4. Le système doit permettre à l'utilisateur d'entrer l'identifiant d'une partie privée qu'il souhaite rejoindre à partir du menu principal (souhaitable).
- 3.1.11.5. Le système doit ajouter l'utilisateur à la partie privée dont il a entré l'identifiant si celui-ci est valide (souhaitable).
- 3.1.11.6. Le système doit avertir l'utilisateur que la partie est pleine s'il essaie de rejoindre une partie où le nombre maximum de personnes est atteint (souhaitable).
- 3.1.11.7. Le système doit avertir l'utilisateur que la partie est déjà commencée et qu'il ne peut pas rejoindre si le jeu est déjà en cours (souhaitable).

3.1.12. Vote sur dessins du joueur virtuel

- 3.1.12.1. Le système doit permettre à l'utilisateur de voir une liste avec les paires mot-image des joueurs virtuels de la partie lorsque celle-ci se termine (souhaitable).
- 3.1.12.1. Le système doit permettre à l'utilisateur de voir la valeur du upvote de chaque paire mot-image dans la liste de l'exigence 3.1.12.1. (souhaitable)
- 3.1.12.2. Le système doit permettre à l'utilisateur d'incrémenter le upvote d'une paire mot-image s'il a aimé celle-ci (souhaitable).
- 3.1.12.3. Le système doit permettre à l'utilisateur décrémenter le upvote d'une paire mot-image s'il n'a pas aimé celle-ci (souhaitable).
- 3.1.12.4. Le système doit automatiquement retirer une paire mot-image la banque de dessin des joueurs virtuels si les upvote deviennent inférieur à -5 (souhaitable).

3.1.12.5. Le système doit assigner un upvote de valeur 0 lors de la création d'une paire mot-image (souhaitable).

3.1.13. Liens d'amitié

3.1.13.1. Le système doit permettre à l'utilisateur d'envoyer des demandes d'amitiés à d'autres joueurs (souhaitable).

3.1.13.2. Le système doit permettre à l'utilisateur d'accepter des demandes d'amitiés reçues par d'autres joueurs (souhaitable).

3.1.13.3. Le système doit permettre à l'utilisateur d'initialiser une conversation avec un ou plusieurs amis à partir de sa liste d'amitié (souhaitable).

3.1.13.4. Le système doit permettre à l'utilisateur de retirer des amis de sa liste d'amitié (souhaitable).

3.1.14. Fil d'image des parties jouées

3.1.14.1. Le système doit permettre à l'utilisateur de visualiser un fil d'image sur le menu principal qui contient un maximum de 50 images (souhaitable).

3.1.14.2. Le système doit sélectionner au hasard un dessin fait par un joueur réel durant chaque partie du mode classique pour l'afficher au début du fil d'image des parties jouées (souhaitable).

3.1.14.3. Le système doit enlever la plus ancienne image du fil lorsque celui-ci en contient 50 (souhaitable).

3.1.15. Rejoindre ses amis

3.1.15.1. Le système doit permettre à l'utilisateur de voir les groupes contenant des personnes de sa liste d'amitié en premier lorsqu'il recherche un groupe pour une partie publique dans la liste des parties (souhaitable).

3.1.16. Invitation des amis à une partie privée

3.1.16.1. Le système doit permettre à l'utilisateur d'inviter des amis lorsqu'il crée une partie privée pour le mode classique (souhaitable).

3.1.16.2. Le système doit permettre à l'utilisateur d'inviter des amis lorsqu'il crée une partie privée pour le mode sprint coopératif (souhaitable).

3.1.16.3. Le système doit permettre à l'utilisateur de recevoir un lien pour rejoindre la partie dans un groupe de clavardage comportant toutes les personnes qui se sont fait inviter par le créateur de la partie (souhaitable).

3.1.16.4. Le système doit permettre à l'utilisateur se fait ajouter au lobby de la partie privée automatiquement lorsqu'il appuie sur le lien.

3.1.16.5 Le système doit avertir l'utilisateur si le lobby est complet lorsqu'il appuie sur lien pour rejoindre une partie privée. (Souhaitable).

3.2. Client léger

3.2.1. Clavardage

3.2.1.1. Clavardage - Intégration

Les mêmes exigences que la section 3.1.1.1. sont appliquées (essentiel) à l'exception des exigences 3.1.1.1.1 et 3.1.1.1.3.

3.2.1.2. Clavardage - Canaux de discussion

Les mêmes exigences que la section 3.1.1.2. sont appliquées (essentiel).

3.2.2. Profil utilisateur et historique

Les mêmes exigences que la section 3.1.2. sont appliquées (essentiel).

3.2.3. Modes de jeu

Les mêmes exigences que la section 3.1.3. sont appliquées (essentiel).

3.2.4. Effets visuels et sonores

Les mêmes exigences que la section 3.1.6. sont appliquées (essentiel).

3.2.5. Tutoriel

Les mêmes exigences que la section 3.1.7.1. sont appliquées (essentiel).

3.2.6. Suggestion de mots

Les mêmes exigences que la section 3.1.9. sont appliquées (souhaitable).

3.2.7. Téléversement d'un dessin d'une partie vers la banque de dessins des joueurs virtuels.

Les mêmes exigences que la section 3.1.10. sont appliquées (essentiel).

3.2.8. Partie privée avec identifiant de la partie

Les mêmes exigences que la section 3.1.11. sont appliquées (souhaitable).

3.2.9. Vote sur dessins du joueur virtuel

Les mêmes exigences que la section 3.1.12. sont appliquées (souhaitable).

3.2.10. Liens d'amitié

Les mêmes exigences que la section 3.1.13. sont appliquées (souhaitable).

3.2.11. Rejoindre ses amis

Les mêmes exigences que la section 3.1.15. sont appliquées (souhaitable).

3.2.12. Invitation des amis à une partie privée

Les mêmes exigences que la section 3.1.16. sont appliquées (souhaitable).

4. Exigences non-fonctionnelles

4.1. Utilisabilité

- 4.1.1. Le système doit permettre à un nouvel utilisateur d'être en mesure de créer son profil en moins de 2 minutes.
- 4.1.2. Le système doit permettre à un nouvel utilisateur d'être en mesure de joindre une partie en moins de 30 secondes après avoir visualisé le tutoriel.
- 4.1.3. Le système doit permettre à un nouvel utilisateur d'être en mesure de créer une partie en moins de 1 minute après avoir visualisé le tutoriel.
- 4.1.4. Le système doit permettre à un nouvel utilisateur d'être en mesure de créer une paire mot-image pour le joueur virtuel après avoir visualisé le tutoriel sur le client lourd.
- 4.1.5. Le système doit permettre à l'utilisateur d'être en mesure d'utiliser le clavier à partir de toutes les interfaces du client léger et du client lourd.
- 4.1.6. Le système doit permettre de faire une rétroaction visuelle lorsque l'utilisateur clique sur un bouton.
- 4.1.7. La disposition des éléments des interfaces entre les différentes plateformes doit être spatialement équivalente.
- 4.1.8. L'utilisateur doit être en mesure de lire le texte complet disponible dans une interface en moins d'une minute.
- 4.1.9. Le système doit permettre de faire une demande de confirmation à l'utilisateur avant une action destructive.
- 4.1.10. Le style de police utilisé doit être l'un de ceux-ci : Arial, Courier New, Calibri, Verdana et Times New Roman pour faciliter la lisibilité.
- 4.1.11. Le système doit avoir des icônes identiques entre les différentes plateformes.
- 4.1.12. Le système doit permettre à l'utilisateur de seulement jouer en mode paysage sur la tablette.
- 4.1.13. Le système doit présenter aux utilisateurs le dessin de façon simultanée.

4.2. Fiabilité

- 4.2.1. Un bogue empêchant l'utilisateur de jouer une partie doit être corrigé dans un délai maximal de 24 heures à partir de leur découverte.
- 4.2.2. Le système doit être opérationnel 96% du temps

4.3. Performance

- 4.3.1. Le système doit permettre l'utilisation simultanée de l'application par 8 utilisateurs.
- 4.3.1 Le système doit être capable de supporter 4 parties simultanément.
- 4.3.2. Le délai de traitement d'une requête au serveur doit être au maximum 500ms.
- 4.3.3. Le système doit supporter au moins 25 canaux de discussion concurrents
- 4.3.4. Le client léger doit être supporté par la tablette Samsung Tab A 2019 avec un rafraîchissement d'image d'au minimum 24 images par seconde.
- 4.3.5. Le client lourd doit démarrer en moins de 15 secondes sur un ordinateur ayant 8Go de Ram, un processeur Intel i5 et un SSD.
- 4.3.6. L'exécutable du client lourd doit être au maximum de 8Go.

4.4. Maintenabilité

- 4.4.1. Le système doit avoir une copie qui doit permettre la récupération des données des utilisateurs.

- 4.4.2. Le système doit permettre à un administrateur d'avoir accès à toutes les données de la base données. Ils doivent pouvoir effectuer des opérations manuellement: modifier, ajouter, supprimer.
- 4.4.3. Le système doit être maintenu à l'aide du numéro de version incrémentant à chaque déploiement de nouvelles fonctionnalités. La nomenclature des versions est disponible dans l'annexe A du document.
- 4.4.4. Le système doit permettre à l'utilisateur de conserver ses données lors d'une mise à jour du logiciel.
- 4.4.5. Le système doit être écrit avec des codes respectant les normes de codage du projet.
 - 4.4.5.1. L'indentation doit être d'une grandeur de 4 espaces est utilisée.
 - 4.4.5.2. Le langage utilisé lors de la programmation doit être l'anglais.
 - 4.4.5.3. Le nom d'une constante doit être en majuscules (Ex: NOM_UTILISATEUR)
 - 4.4.5.4. Le nom d'une variable doit être écrit en camelCase.
 - 4.4.5.5. Le nom d'une fonction doit être en camelCase
 - 4.4.5.6. Le nom d'un objet doit être en PascalCase
 - 4.4.5.7. Une ligne doit être au maximum 80 caractères.
- 4.4.6 Le code doit être documenté à l'aide de commentaires expliquant le contenu du fichier et le fonctionnement d'une fonction.

4.5. Contraintes de conception

- 4.5.1. Le client léger doit être programmé avec les langages Kotlin et XML.
- 4.5.2. Le serveur doit être programmé avec le langage JavaScript.
- 4.5.3. Le serveur doit fonctionner dans l'environnement d'exécution Node.JS.
- 4.5.6. Le client lourd doit être programmé avec les langages TypeScript, HTML et CSS.
- 4.5.7. La base de données doit être construite avec MongoDB sur la plateforme Atlas.
- 4.5.8. Le serveur doit communiquer avec les clients pour la transmission d'images et de messages des utilisateurs avec socket.io.
- 4.5.9. Le serveur doit utiliser le *framework* Express.js.
- 4.5.10. Le client lourd doit être transformé en une interface utilisateur graphique fonctionnant sur le bureau de Windows et macOS en utilisant Electron.

4.6. Sécurité

- 4.6.1. Le système doit crypter un mot de passe durant la transmission.
- 4.6.2. Le système doit crypter un mot de passe lorsqu'il est stocké dans la base de données.
- 4.6.3. Le système doit crypter un message d'un canal de discussion durant leur transmission.
- 4.6.4 Le système doit utiliser les JSON Web Token pour s'assurer que l'utilisateur est bien celui qu'il dit être.
- 4.6.5 Le système doit permettre à l'utilisateur d'être connecté qu'à une seule place en même temps.

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

- 4.7.1. Le système doit mettre à la disposition de l'utilisateur un système d'aide sur chaque interface utilisateur de l'application.
- 4.7.2. Le système doit afficher un message d'erreur à l'utilisateur dans un langage vulgarisé soit un langage sans terme technique.

ANNEXE A: Nomenclature des versions

La nomenclature de la version des programmes est importante pour garder de la cohérence dans les différentes versions du logiciel, la gestion des bogues et la gestion des nouvelles fonctionnalités.

Les versions du programme sont de la forme suivante. X.Y.Z.A/a

X: est un chiffre qui correspond au numéro de version du logiciel. L'incrément de ce chiffre se fait à chaque fois que de nouvelles fonctionnalités importantes changeant l'expérience de l'utilisateur sont ajoutées.

Y: est un chiffre qui correspond à l'ajout de petites fonctionnalités qui ne changent pas beaucoup l'expérience de l'utilisateur.

Z: est un chiffre qui correspond à une nouvelle version du logiciel lorsque des bogues mineurs sont réglés.

A/a: est une lettre qui correspond à quel type d'application on utilise. La lettre majuscule correspond au client lourd et la lettre minuscule correspond au client léger.