
Équipe 210

Draw It Up!
Spécifications des requis du système (SRS)

Version 1.8

Historique des révisions

Date	Version	Description	Auteurs
2021-01-26	1.0	Rédaction initiale du SRS.	Benjamin Saint-Cyr
2021-01-30	1.1	Ajout d'exigences non-fonctionnelles.	Bouchra-nour-el-yakin Dahamni
2021-01-31	1.2	Ajout d'exigences non-fonctionnelles.	Benjamin Saint-Cyr, Marc-André Fillion
2021-02-02	1.3	Ajout d'exigences fonctionnelles et début de la rédaction de l'introduction.	Benjamin Saint-Cyr, Gérard Akkerhuis
2021-02-03	1.4	Finalisation des exigences fonctionnelles. Finalisation des exigences non-fonctionnelles. Écriture et finalisation de la description globale. Vérification du contenu et correction primaire.	Benjamin Saint-Cyr, Bouchra-nour-el-yakin Dahamni, Charles-Édouard Turcot, Gérard Akkerhuis, Marc-André Fillion, Vincenzo Pucci-Barbeau
2021-02-04	1.5	Complétion de la section 1.2. Validation et correction finale.	Benjamin Saint-Cyr, Bouchra-nour-el-yakin Dahamni, Charles-Édouard Turcot, Gérard Akkerhuis, Marc-André Fillion
2021-02-19	1.6	Correction des erreurs en fonction des commentaires des chargés et révision finale.	Charles-Édouard Turcot, Gérard Akkerhuis
2021-04-18	1.7	Correction et révision de l'ensemble du document pour le rendre conforme au produit final	Marc-André Fillion
2021-04-19	1.8	Révision finale de l'équipe avant la remise	Équipe 210

Table des matières

1. Introduction	4
1.1. But	4
1.2. Définitions, acronymes et abréviations	4
1.2.23 Utilisateur normal	5
1.2.24 Utilisateur spécialisé	5
1.3. Vue d'ensemble du document	5
2. Description globale	6
2.1. Caractéristiques des usagers	6
2.2. Interfaces	6
2.2.1. Interfaces usagers	6
2.2.2. Interfaces matérielles	6
2.2.3. Interfaces logicielles	6
2.2.4. Interfaces de communication	7
2.3. Contraintes générales	7
2.4. Hypothèses et dépendances	7
3. Exigences fonctionnelles	7
3.1. Clavardage	7
3.2. Profil de l'utilisateur	8
3.3. Modes de jeu	8
3.4. Création d'une paire mot-image	10
3.5. Tutoriel	11
3.6. Trophée	11
3.7. Classement	11
3.8. Joueur virtuel	12
3.9. Système d'équipe dans le jeu	12
3.10. Support pour mot de passe oublié	13
3.11. Effets visuels et sonores	13
3.12. Filtre de blasphème	14
4. Exigences non-fonctionnelles	14
4.1. Convivialité	14
4.2. Fiabilité	14
4.3. Performance	15
4.4. Maintenabilité	15
4.5. Contraintes de conception	16
4.6. Sécurité	16
4.7. Portabilité	17

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

1.2.1. Angular :

Désigne un cadriciel pour la production d'applications web basé sur le TypeScript.

1.2.2. Architecture client/serveur :

Désigne un mode de communication au sein d'un réseau basé sur un principe de requête-réponse. Les requêtes étant effectuées par les systèmes clients au serveur. Lesquels reçoivent par la suite une réponse de la part du serveur dans le but de fournir des services.

1.2.3. AWS :

Amazon Web Service est un service web offert par Amazon qui permet de déployer son programme dans le nuage.

1.2.4. Base de données :

Désigne un système qui permet de stocker, organiser et analyser des données.

1.2.5. Cadriciel:

Ensemble de bibliothèques, d'outils et de conventions destiné à la création d'une application logicielle.

1.2.6. Client léger :

Désigne la partie de l'application exécutable sur la version tablette Android. Celui-ci contient les fonctionnalités de dessin propre au client léger et envoie des requêtes au serveur.

1.2.7. Client lourd :

Désigne la partie de l'application exécutable sur la version PC. Celui-ci contient les fonctionnalités de dessin propre au client lourd et envoie des requêtes au serveur.

1.2.8. Docker :

Désigne un logiciel permettant l'empaquetage d'applications et leur exécution.

1.2.9. Electron :

Désigne un environnement permettant le développement d'applications multi-plateformes avec des technologies web. (<https://www.electronjs.org/>)

1.2.10. GB, MB :

Désignant un “*gigabyte*” (gigaoctet en français) et un “*megabyte*” (mégaoctet en français) respectivement. Ils sont tous les deux utilisés en tant qu'unité de mesure de mémoire.

1.2.11. Go:

Désigne un langage de programmation généralement, et surtout dans notre cas, utilisé pour les serveurs.

1.2.12. HTML :

Désigne le langage de balisage utilisé pour représenter les pages web.

1.2.13. joueur virtuel :

Désigne la représentation informatique d'un utilisateur créé par le système.

1.2.14. Kotlin :

Désigne un langage de programmation utilisé pour le développement d'applications Android.

1.2.15. Librairie :

Ensemble de sous-logiciel utilisé dans un système informatique

1.2.16 Linting :

Désigne l'action d'exécuter un programme afin de trouver des erreurs et des fautes de style.

1.2.17. MongoDB :

Désigne une base de données basée sur des documents conçue pour les applications modernes.

(<https://www.mongodb.com/>)

1.2.18. NodeJS :

Désigne une plateforme et environnement permettant l'exécution de code JavaScript à l'extérieur d'un navigateur web.

1.2.19. PC :

Désigne un ordinateur personnel.

1.2.20. Serveur :

Désigne la partie de l'application responsable de répondre aux requêtes clients et de contrôler l'accès des utilisateurs.

1.2.21. TypeScript :

Désigne un langage de programmation utilisé pour le développement d'applications, il s'agit d'un langage sur-ensemble de JavaScript.

1.2.22. Utilisateur :

Désigne une personne qui utilise notre application, et ce, à partir du client léger ou du client lourd.

1.2.23 Utilisateur normal

Désigne un utilisateur qui a peu ou pas d'expérience avec les ordinateurs et qui ne comprend pas intuitivement l'interface.

1.2.24 Utilisateur spécialisé

Un utilisateur spécialisé est un utilisateur qui a beaucoup d'expérience avec les ordinateurs et qui comprend intuitivement l'interface.

1.3. Vue d'ensemble du document

La section 2 du SRS présente la description globale du produit incluant les caractéristiques des

utilisateurs, les interfaces utilisateurs qui seront implémentées et les contraintes générales de l'application. Les exigences fonctionnelles et non fonctionnelles sont ensuite décrites dans les sections 3 et 4 respectivement.

2. Description globale

Le logiciel "PolyDessin 2" est un jeu multiplateforme, semblable à Pictionary, permettant de faire des parties avec d'autres utilisateurs ou bien avec des utilisateurs virtuels. Les parties consistent à occuper l'un de deux rôles, celui d'un dessinateur qui doit faire un dessin représentant un mot donné par l'application et celui du devineur qui doit tenter de trouver le mot représenté par le dessin. Le jeu est accessible par un client lourd, sur PC, et un client léger, sur mobile. Le jeu est produit à partir de l'application de dessin qui avait été développée en projet 2.

2.1. Caractéristiques des usagers

Les utilisateurs du logiciel sont des étudiants en génie logiciel ou des ingénieurs logiciel. En conséquence, l'utilisateur typique a des connaissances techniques avancées et est familier avec un logiciel de dessin de ce type. Les usagers sont donc capables d'utiliser un ordinateur ou bien une tablette pour démarrer une application. Ils savent lire, écrire et sont capables de faire des associations d'images avec des mots. Ils ont au moins 18 ans, mais leur âge spécifique importe peu. L'application est donc développée en prenant en compte ces caractéristiques.

2.2. Interfaces

2.2.1. Interfaces usagers

L'interface utilisateur sur le client lourd est développée en HTML avec le cadriciel Angular et possède une structure simple basée sur un menu principal qui permet d'avoir accès aux différentes fonctionnalités de l'application. Un système de clavardage est intégré à l'application, mais peut également être déplacé sur une autre fenêtre. L'interface utilisateur du client léger est développée en Kotlin et comporte une structure similaire de menu principal permettant d'accéder au reste de l'application. Le serveur est accessible par une interface de type "entrée de commande" sur console.

2.2.2. Interfaces matérielles

Le client lourd supporte comme interface matérielle de résolution 1920x1080p une souris et un clavier. Le client léger a pour interface un écran tactile multipoint de résolution 1920x1200p. Afin de pouvoir entendre les effets sonores, l'appareil doit également posséder un système de son approprié.

2.2.3. Interfaces logicielles

Le langage de programmation Go accompagné de sa librairie standard sera utilisé pour créer le serveur. Les communications avec la base de données seront faites avec la librairie libre de MongoDB, de la base

de données du même nom. Le serveur sera déployé avec une image Docker sur le nuage AWS avec une licence gratuite.

Le client lourd utilise le langage de programmation TypeScript avec la plateforme NodeJS et le cadriciel Angular.

Pour ce qui est du client léger, le langage de programmation multiplateforme Kotlin sera utilisé. Afin d'aider à l'implémentation, il sera accompagné des librairies Android Jetpack.

2.2.4. Interfaces de communication

La communication entre les clients et le serveur disponible sur l'Internet est faite avec un protocole d'envoi de websockets et en utilisant REST. Les clients communiquent avec le serveur grâce à une connexion internet établie à l'aide de WIFI.

2.3. Contraintes générales

Le client lourd doit être une application Electron avec le cadriciel Angular et doit être fonctionnel sur un ordinateur qui a comme système d'exploitation Windows 10. Pour ce qui est du client léger, il doit être utilisé sur une tablette Samsung Galaxy Tab A 2019 utilisant le système d'exploitation Android 9.0 (Pie) fonctionnant avec 2GB de mémoire vive et 32GB d'espace disponible. Le serveur possède un disque de 1GB, plus de 512MB seront attribués à la base de données. Il est constitué d'un processeur à 2.5GHz avec un débit réseau maximum de 300 Mb/s. En conséquence, une des contraintes principales de l'application est de respecter les limites de mémoire vive, plus spécifiquement sur le client léger, afin d'éviter d'avoir des manques de mémoire vive lors de l'exécution du programme.

2.4. Hypothèses et dépendances

La dépendance externe principale de l'application est le bon fonctionnement du nuage AWS. On pose d'abord l'hypothèse que pour le bon fonctionnement du jeu, les serveurs AWS sont fonctionnels en tout temps et qu'aucun problème de disponibilité n'aura lieu. On pose aussi l'hypothèse que l'accès internet de la tablette Android et l'ordinateur Windows est fiable et qu'aucune interruption de service n'aura lieu.

3. Exigences fonctionnelles

3.1. Clavardage

3.1.1. **[Essentiel]** Client lourd et léger : Le système doit permettre à l'utilisateur d'accéder à une interface de clavardage.

3.1.2. **[Essentiel]** Client lourd : Le système doit permettre ~~au client lourd~~ à l'utilisateur de changer entre le mode intégré et le mode fenêtre du clavardage.

~~3.1.3. [Essentiel] Client lourd et léger : Le système doit permettre à l'utilisateur d'utiliser des autocollants personnalisés.~~

~~3.1.3.1. [Essentiel] Client lourd: Le système doit permettre à l'utilisateur de créer ses propres autocollants.~~

~~3.1.3.2. [Essentiel] Client lourd: Le système doit permettre à l'utilisateur d'enregistrer ses autocollants.~~

~~3.1.3.3. [Essentiel] Client lourd et léger: Le système doit afficher les autocollants des utilisateurs dans le clavardage.~~

3.1.4. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de rejoindre un ou plusieurs canaux de discussion.

3.1.5. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir l'historique de conversation de tous les canaux dont il est membre.

~~3.1.6. [Essentiel] Client léger: Le système doit notifier l'utilisateur s'il a un nouveau message à l'aide d'un indicateur visuel et un effet sonore.~~

3.2. Profil de l'utilisateur

3.2.1. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir **son courriel**. ~~sur son profil toutes les informations suivantes:~~

~~3.2.1.1. [Essentiel] Client lourd et léger : (informations privées) son prénom, nom, adresse courriel et mot de passe.~~

~~3.2.1.2. [Essentiel] Client lourd et léger: (informations publiques) son pseudonyme, son avatar, dates et heure de connexion et déconnexion, date et heure des parties jouées, nom des utilisateurs des parties jouées ainsi que le résultat pour chaque partie, le nombre de parties jouées, le pourcentage de victoires, le temps moyen d'une partie, le temps total passé à jouer, les trophées gagnés ainsi que les trophées pas encore gagnés.~~

~~3.2.3. [Essentiel] Client lourd et léger: Le système doit permettre à chaque utilisateur de cliquer sur le nom d'un autre utilisateur dans le chat afin d'accéder à son profil et de voir ses informations publiques. (Voir 3.2.1.2)~~

3.2.2. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir son pseudonyme.

3.2.3. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir son avatar.

3.2.4. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir chaque heure où il s'est connecté.

3.2.5. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir chaque heure où il s'est déconnecté.

3.2.6. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de voir la date de chaque partie qu'il a jouée.

3.2.7. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir l'heure de fin de chaque partie qu'il a jouée.

3.2.8. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir, pour chaque partie qu'il a jouée, le courriel de chaque joueur ayant participé.

3.2.9. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le résultat de chaque partie qu'il a jouée.

3.2.10. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le nombre de parties qu'il a jouées, tout mode confondu.

3.2.11. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir son ratio de victoires/défaites.

3.2.12. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le temps moyen qu'il passe en partie.

3.2.13. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le temps de jeu total lui étant associé.

3.2.14. [Essentiel] Client lourd: Le système doit permettre à l'utilisateur de voir chaque trophée qu'il a débloqué.

3.3. Modes de jeu

3.3.1. [Essentiel] Client lourd et léger : Mode classique

3.3.1.1. [Essentiel] Client lourd et léger: Le système doit permettre ~~les parties de~~ à l'utilisateur de jouer à une partie de deux versus deux.

3.3.1.2. ~~[Optionnel] Client lourd et léger: Le système doit permettre aux autres utilisateurs de joindre une partie à l'aide d'un code alphanumérique de salle.~~

3.3.1.3. [Essentiel] Client lourd et léger: Le système doit permettre au maximum 1 joueur virtuel dans une équipe. ~~, seulement s'il n'y a pas de deuxième utilisateur humain dans l'équipe.~~

3.3.1.4. [Essentiel] Client lourd et léger: Le système doit permettre à un membre de l'équipe de dessiner pendant que l'autre membre devine.

3.3.1.5. [Essentiel] Client lourd et léger: Le système doit attribuer ~~40 points~~ **1 point** à une équipe lors de la résolution d'un mot.

3.3.1.6. [Essentiel] Client lourd et léger: Le système doit donner la victoire à l'équipe avec le plus de points à la fin de la partie.

3.3.1.7. ~~[Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de choisir le temps accordé pour deviner un mot lors de la création d'une partie.~~

3.3.1.8. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le score en tout temps pendant la partie.

3.3.1.9. [Essentiel] Client lourd et léger: Le système doit permettre un droit de réplique lorsqu'une équipe échoue à deviner un mot. ~~L'équipe ayant le droit de réplique a une chance de~~

~~faire un dessin et deviner le mot.~~

3.3.2. [Essentiel] Client lourd et léger: Mode sprint solo.

~~3.3.2.1. [Essentiel] Client lourd et léger: Le système doit permettre à un utilisateur de commencer un sprint solo de 100 secondes.~~

3.3.2.2. [Essentiel] Client lourd et léger: Le système doit permettre à un joueur virtuel de dessiner une image, prenant un maximum de ~~20~~ **10** secondes pour le dessin complet.

3.3.2.3. [Essentiel] Client lourd et léger: Lorsque l'utilisateur trouve le mot, le système doit permettre au joueur virtuel de commencer un nouveau dessin.

3.3.2.4. [Essentiel] Client lourd et léger: Lorsque l'utilisateur trouve le mot, ~~il doit obtenir 10 points~~ **le système doit lui octroyer 1 point** et un ajout de temps équivalent au nombre d'essais restants multiplié par 3.

~~3.3.2.5. [Essentiel] Client lourd et léger: Lorsque l'utilisateur devine un mot, il doit disposer de 3 essais en difficulté facile, 2 essais en difficulté moyenne et 1 essai seulement en difficulté difficile.~~

~~3.3.2.6. [Optionnel] Client lourd et léger: Le système doit permettre à l'utilisateur de passer un mot, s'il est incapable de le trouver.~~

~~3.3.2.7. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de voir le score, le temps restant et le nombre d'essais restant en tout temps pendant la partie.~~

3.3.2.8. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de sélectionner un niveau de difficulté entre facile, moyen ou difficile.

~~3.3.2.9. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de sélectionner un mode de dessin.~~

3.3.2.10. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur d'obtenir un bonus de temps de 8 secondes en difficulté facile lorsqu'il devine le mot.

3.3.2.11. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur d'obtenir un bonus de temps de 6 secondes en difficulté moyenne lorsqu'il devine le mot.

3.3.2.12. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur d'obtenir un bonus de temps de 4 secondes en difficulté difficile lorsqu'il devine le mot.

3.3.2.13. [Essentiel] Client lourd et léger: Le système doit offrir 5 essais à l'utilisateur par mot à deviner en difficulté facile.

3.3.2.14. [Essentiel] Client lourd et léger: Le système doit offrir 3 essais à l'utilisateur par mot à deviner en difficulté moyenne.

3.3.2.15. [Essentiel] Client lourd et léger: Le système doit offrir 1 essai à l'utilisateur par mot à deviner en difficulté difficile.

3.3.2.16. [Essentiel] Client lourd et léger: Le système doit offrir à l'utilisateur un temps initial de 2 minutes en difficulté facile.

3.3.2.17. [Essentiel] Client lourd et léger: Le système doit offrir à l'utilisateur un temps

initial de 1 minute en difficulté moyenne.

3.3.2.18. [Essentiel] Client lourd et léger: Le système doit offrir à l'utilisateur un temps initial de 30 secondes en difficulté difficile.

3.3.2.19. [Essentiel] Client lourd et léger: Le système doit faire en sorte que le joueur virtuel change de dessin lorsque l'utilisateur n'a plus d'essai.

3.3.2.20. [Essentiel] Client lourd et léger: Le système doit terminer la partie lorsque le temps est écoulé.

3.3.2.21. [Essentiel] Client lourd et léger: Le système doit afficher le score en tout temps en cours de partie.

3.3.2.22. [Essentiel] Client lourd et léger: Le système doit afficher le temps restant en tout temps en cours de partie.

3.3.2.23. [Essentiel] Client lourd et léger: Le système doit afficher le nombre d'essais restant en tout temps en cours de partie.

~~3.3.3. [Souhaitable] Client lourd et léger: Mode sprint coopératif.~~

~~3.3.3.1. [Souhaitable] Client lourd et léger: Le système doit permettre à un utilisateur de commencer un sprint coopératif de 100 secondes.~~

~~3.3.3.2. [Souhaitable] Client lourd et léger: Le système doit permettre aux autres utilisateurs de joindre la partie à l'aide d'un code alphanumérique de salle.~~

~~3.3.3.3. [Souhaitable] Client lourd et léger: Le système doit permettre à un joueur virtuel de dessiner une image, jusqu'à ce qu'un utilisateur la devine.~~

~~3.3.3.4. [Souhaitable] Client lourd et léger: Lorsque les utilisateurs trouvent le mot, le système doit permettre au joueur virtuel de commencer un nouveau dessin.~~

~~3.3.3.5. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de sélectionner un niveau de difficulté entre facile, moyen ou difficile.~~

~~3.3.3.6. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de sélectionner un mode de dessin.~~

~~3.3.3.7. [Essentiel] Client lourd et léger: Lorsque les utilisateurs devinent un mot, ils doivent disposer de 3 essais par utilisateur en difficulté facile, 2 essais par utilisateur en difficulté moyenne et 1 essai par utilisateur seulement en difficulté difficile.~~

~~3.3.3.8. [Essentiel] Client lourd et léger: Lorsque les utilisateurs n'ont plus d'essais, le joueur virtuel commence à dessiner un nouveau dessin.~~

3.4. Création d'une paire mot-image

3.4.1. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de **dessiner l'image d'une nouvelle paire** ~~fournir un mot, un ou plusieurs indices et un dessin.~~

3.4.2. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de visionner un aperçu

accéléré de l'animation de son dessin.

3.4.3. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur ~~d'avoir la possibilité~~ d'associer un niveau de difficulté à son dessin: **facile, moyen ou difficile.**

3.4.4. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de choisir comment le joueur virtuel dessine son dessin.

3.4.4.1. **[Essentiel]** ~~Dans le mode conventionnel, le dessin d'un joueur virtuel doit se faire~~ **Le système doit faire en sorte que le joueur virtuel dessine** dans le même ordre que sa création par l'utilisateur **dans le mode conventionnel.**

3.4.4.2. **[Essentiel]** ~~Dans le mode aléatoire, le dessin d'un joueur virtuel doit se faire~~ **Le système doit faire en sorte que le joueur virtuel dessine** dans un ordre aléatoire et unique ~~à chaque partie~~ **dans le mode aléatoire.**

3.4.4.3. **[Essentiel]** ~~Dans le mode panoramique, le dessin d'un joueur virtuel doit se faire~~ **Le système doit faire en sorte que le joueur virtuel dessine** selon la position du dessin **dans le mode panoramique.**

3.4.4.3.1. **[Essentiel]** Le système doit dessiner de gauche à droite.

3.4.4.3.2. **[Essentiel]** Le système doit dessiner de droite à gauche.

3.4.4.3.3. **[Essentiel]** Le système doit dessiner de haut en bas.

3.4.4.3.4. **[Essentiel]** Le système doit dessiner de bas en haut.

3.4.4.4. **[Essentiel]** ~~Dans le mode centré, le dessin d'un joueur virtuel doit apparaître~~ **Le système doit faire en sorte que le joueur virtuel dessine** selon la position de départ du trait.

3.4.4.4.1. **[Essentiel]** Le système doit dessiner de l'intérieur vers l'extérieur.

3.4.4.4.2. **[Essentiel]** Le système doit dessiner de l'extérieur vers l'intérieur.

3.4.5. **[Essentiel]** Client lourd et léger: **Le système doit obliger l'utilisateur à fournir entre 1 et 15 indices pour la paire.**

3.4.6. **[Essentiel]** Client lourd et léger: **Le système doit permettre à l'utilisateur d'associer un mot à son image pour la paire.**

3.5. Tutoriel

3.5.1. **[Essentiel]** Client lourd et léger: Le système doit permettre l'accès à un tutoriel qui présente ~~des images et des explications sur l'application~~ **une animation pour chaque instruction.**

3.5.2. **[Essentiel]** Client lourd et léger: Le système doit permettre **à l'utilisateur de naviguer d'une explication à l'autre** ~~l'accès à un bouton "Suivant" pour passer à la prochaine explication.~~

3.5.3. **[Essentiel]** ~~Client lourd et léger: Le système doit faire apparaître le tutoriel automatiquement à la première connexion de l'utilisateur.~~

3.5.4. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de consulter le tutoriel à tout moment hors partie.

3.5.5. [Essentiel] Client lourd et léger: Le système doit permettre l'accès à un tutoriel qui présente une explication pour chaque animation.

3.6. Trophée

3.6.1. **[Essentiel]** Client lourd et léger: Le système doit notifier l'utilisateur lorsqu'il débloque un nouveau trophée.

3.6.2. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il **réalise un accomplissement**. ~~atteint un des accomplissements ci-dessous: Il garde le trophée même s'il ne remplit plus les conditions:~~

3.6.2.1. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a gagné sa première partie.

3.6.2.2. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a gagné sa ~~1^{ère}~~ **deuxième** partie.

3.6.2.3. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a passé le cap de ~~5h~~ **minutes** dans le jeu.

3.6.2.4. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a ouvert un chat privé **pour la première fois**.

3.6.2.5. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a fini le tutoriel.

3.6.2.6. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a fini un sprint solo avec ~~40~~ **2** dessins devinés.

~~3.6.2.7. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a créé 5 autocollants.~~

3.6.2.8. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a fait ~~5~~ **2** dessins non devinés par le partenaire en mode classique.

3.6.2.9. **[Essentiel]** Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il atteint un ratio de victoire de 2 et plus.

3.6.2.10. [Essentiel] Client lourd et léger: Le système doit permettre à l'utilisateur de recevoir un nouveau trophée lorsqu'il a créé sa première paire mot-image.

3.7. Classement

3.7.1. **[Essentiel]** Client lourd et léger: Le système doit permettre l'affichage d'un classement sur le menu principal.

3.7.2.**[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon les catégories ci-dessous :

3.7.2.1. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon le nombre de parties jouées le plus élevé.

3.7.2.2. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de

classement tous modes confondus, selon le nombre de victoires le plus élevé.

3.7.2.3. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon le nombre de trophées débloqués le plus élevé.

3.7.2.4. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon le nombre d'heures jouées le plus élevé.

3.7.2.5. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon le meilleur ratio de victoire.

3.7.2.6. **[Essentiel]** Client lourd et léger: Le système doit permettre d'afficher des tableaux de classement tous modes confondus, selon les meilleures équipes.

3.8. Joueur virtuel

3.8.1. **[Essentiel]** Client lourd et léger : Le système doit s'assurer qu'un joueur virtuel ~~dans une partie~~ possède une personnalité distincte conservée tout au long ~~de la d'une partie : arrogant, amical, gêné ou~~ euphorique.

3.8.1.1. **[Essentiel]** Client lourd et léger : Le système doit pouvoir donner une personnalité arrogante à un joueur virtuel.

3.8.1.2. **[Essentiel]** Client lourd et léger : Le système doit pouvoir donner une personnalité amicale à un joueur virtuel.

3.8.1.3. **[Essentiel]** Client lourd et léger : Le système doit pouvoir donner une personnalité gênée à un joueur virtuel.

3.8.1.4. **[Essentiel]** Client lourd et léger : Le système doit pouvoir donner une personnalité euphorique à un joueur virtuel.

3.8.2. **[Essentiel]** Client lourd et léger : Le système doit permettre qu'en présence d'un joueur virtuel, **ce dernier envoie** un message simple ~~est envoyé~~ par clavardage mettant sa personnalité de l'avant.

3.8.3. **[Essentiel]** Client lourd et léger : Le système doit permettre que lors d'une victoire, un joueur virtuel envoie un message simple par clavardage ~~exprimant sa satisfaction~~, quelle que soit sa personnalité.

~~3.8.4. **[Souhaitable]** Client lourd et léger : Le système doit permettre qu'un joueur virtuel envoie un message personnalisé selon les statistiques d'un utilisateur : "Prépare toi à te faire détrôner!", "C'est ici que ton règne s'arrête", "Tu gagnes souvent!", "J'aspire avoir autant de victoires que toi", "Tu dois manquer d'espace avec tous ces trophées...", "À ce que je vois, tu adores ce jeu!"~~

~~3.8.5. **[Souhaitable]** Client lourd et léger : Le système doit permettre qu'un joueur virtuel envoie un message en référence à une partie jouée dans le passé.~~

~~3.8.6. **[Souhaitable]** Client lourd et léger : Le système doit permettre qu'un joueur virtuel envoie un message en référence à un utilisateur rencontré dans le passé.~~

~~3.9. Système d'équipe dans le jeu~~

~~3.9.1. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur de créer une équipe.~~

3.9.2. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir une liste de toutes les équipes existantes.

3.9.3. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir le profil de chaque équipe pour voir les informations ci-dessous:

3.9.3.1. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir le profil de chaque équipe pour voir le nombre de victoires associées à l'équipe.

3.9.3.2. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir le profil de chaque équipe pour voir les membres de l'équipe.

3.9.3.3. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir le profil de chaque équipe pour voir leur position actuelle dans le classement des équipes selon le nombre de victoires total de tous ses membres.

3.9.3.4. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur d'aller voir le profil de chaque équipe pour voir leur meilleure position dans le classement des équipes selon le nombre de victoires total de tous ses membres.

3.9.4. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur de faire une demande pour rejoindre une équipe.

3.9.5. **[Essentiel]** Client lourd et léger : Le système doit permettre, lors de la création d'une équipe, un canal de conversation pour celle-ci.

3.9.6. **[Essentiel]** Client lourd et léger : Le système doit permettre, lors de l'adhésion d'un utilisateur à une équipe, son ajout au canal de conversation de l'équipe.

3.9.7. **[Essentiel]** Client lourd et léger : Le système doit permettre de créer une partie uniquement accessible par les membres de la même équipe.

3.9.8. **[Essentiel]** Client lourd et léger : Le système doit permettre à un utilisateur de joindre une et une seule équipe à la fois.

3.10. Support pour mot de passe oublié

3.10.1. **[Essentiel]** Client lourd et léger : Le système doit permettre à l'utilisateur de changer son mot de passe d'entrer son adresse courriel, avant d'être connecté, à laquelle sera envoyé un code numérique aléatoire.

3.10.2. **[Essentiel]** Client lourd et léger : Le système doit permettre à l'utilisateur d'entrer le code numérique aléatoire qu'il a reçu par courriel afin de s'authentifier.

3.10.3. **[Essentiel]** Client lourd et léger : Le système doit permettre à l'utilisateur, si le code alphanumérique aléatoire est validé, de changer son mot de passe d'entrer un nouveau mot de passe pour son compte après que ce dernier ait entré le bon code.

3.11. Effets visuels et sonores

3.11.1. **[Essentiel]** Client lourd et léger : Le système doit ~~activer des effets sonores et visuels lorsqu'un utilisateur débloquent un nouveau trophée~~ **faire apparaître une notification de type toast avec un effet sonore spécifique lorsqu'un utilisateur débloquent un nouveau trophée.**

3.11.2. **[Essentiel]** Client lourd et léger : Le système doit ~~activer des effets sonores et visuels lors de victoires ou de défaites.~~

3.11.2.1. **[Essentiel]** Client lourd et léger : ~~Un ruban de victoire vert, pour une victoire, avec des petites particules. Un ruban de défaite d'une couleur rouge pour une défaite.~~

3.11.2.2. **[Essentiel]** Client lourd et léger : ~~Un effet sonore de type "victoire", pour une victoire et un effet sonore de type "défaite", pour une défaite.~~

3.11.3. **[Essentiel]** Client lourd et léger : Le système doit ~~afficher un compteur pour montrer le temps restant.~~

3.11.4. **[Essentiel]** Client lourd et léger : Le système doit ~~activer un son de compte à rebours lorsqu'il reste 10 secondes au tour.~~

3.11.5. **[Essentiel]** Client lourd et léger : Le système doit activer une rétroaction sonore spécifique après chaque tentative de résolution.

3.11.6 **[Essentiel]** Client lourd et léger : Le système doit faire une animation de transition lors d'un changement de page.

3.11.7 **[Essentiel]** Client lourd et léger : Le système doit faire jouer un effet sonore dans le canal de clavardage ouvert lorsqu'il reçoit un message.

3.11.8 **[Essentiel]** Client lourd et léger : Le système doit faire jouer un effet de confettis à la fin de la partie pour chaque utilisateur.

3.11.9 **[Essentiel]** Client lourd et léger : Le système doit, à la fin de la partie, faire jouer un effet sonore pour chaque utilisateur dans l'équipe gagnante.

3.11.10 **[Essentiel]** Client lourd et léger : Le système doit, à la fin de la partie, faire jouer un effet sonore pour chaque utilisateur dans l'équipe perdante.

3.12. Filtre de blasphème

3.12.1. **[Essentiel]** Client lourd et léger : Le système doit refuser le pseudonyme ou le nom d'utilisateur s'il contient du blasphème lorsqu'il s'inscrit.

3.12.2. **[Essentiel]** Client lourd et léger : Le système doit censurer tous les mots considérés comme du blasphème dans le clavardage **en envoyant le nom de l'utilisateur suivi de "a osé profaner le chat".**

4. Exigences non-fonctionnelles

4.1. Convivialité

4.1.1. L'interface du jeu doit être ergonomique et ne nécessiter aucune connaissance préalable pour l'utilisateur.

~~4.1.2. Le temps de formation moyen requis à un utilisateur normal pour comprendre les modes de jeu doit être de 10 minutes et pour un utilisateur spécialisé est 5 minutes.~~

4.1.3. Le temps moyen de configuration d'une partie par l'utilisateur doit être de 3 minutes.

4.1.4. Le temps moyen pour créer un profil utilisateur doit être de 3 minutes.

4.1.5. Le temps moyen pour afficher l'historique du clavardage de l'utilisateur doit être de 2 minutes.

4.2. Fiabilité

4.2.1. Les différentes parties du système doivent être exemptes de bogues critiques qui empêchent l'utilisation du système.

4.2.2. Le système doit avoir un temps moyen maximum de 4 mois entre les pannes.

4.2.3. Le nombre maximal moyen de défaillances par année doit être de 3.

4.2.4. Le système doit être hors d'usage pendant un maximum de 3 heures lors d'une panne.

4.2.5. Le système doit être disponible à 99 % du temps.

4.3. Performance

4.3.1. Le système doit permettre à l'utilisateur de recevoir chaque message dont il est le destinataire depuis le clavardage instantané au maximum 200 millisecondes après l'envoi du message, et ce, qu'il y ait des blasphèmes ou non.

4.3.2. Le système doit permettre un affichage fluide pour la formation des traits lorsque l'utilisateur observe un autre utilisateur (humain ou virtuel) se servir de l'outil Crayon.

4.3.3. Le système doit permettre un affichage fluide pour la formation des traits lorsque l'utilisateur observe un autre utilisateur (humain ou virtuel) se servir de l'outil Efface.

4.3.4. Le serveur doit permettre à l'utilisateur de sauvegarder un autocollant en moins de 3 secondes sur la base de données.

4.3.5. Le serveur doit prendre au maximum 5 GB de mémoire.

4.3.6. La base de données doit prendre au maximum 512 MB.

4.3.7. Le serveur doit permettre l'enregistrement d'au plus 1000 messages échangés par le clavardage instantané sur la base de données.

4.3.8. Le serveur doit permettre à l'utilisateur d'enregistrer sur la base de données 10 autocollants.

4.3.9. Le serveur doit permettre la connexion simultanée de 50 utilisateurs.

4.3.10. Le système doit permettre de fournir une expérience de jeu avec une latence maximale de 150 ms.

- 4.3.11. Le serveur doit chercher un dessin dans la base de données en moins de 200 millisecondes.
- 4.3.12. Le serveur doit vérifier dans la base de données s'il existe un nom d'utilisateur entré par l'utilisateur qui tente de se connecter en moins de 200 millisecondes.
- 4.3.13. Le serveur doit commencer la partie en moins de 200 millisecondes après que le nombre minimal d'utilisateurs est connecté.
- 4.3.14. Le serveur doit stocker les différents trophées pour chaque utilisateur dans la base de données en moins de 200 millisecondes.
- 4.3.15. Le serveur doit stocker les différentes équipes dans la base de données en moins de 200 millisecondes.
- 4.3.16. Le serveur doit ajouter un utilisateur dans une équipe en moins de 200 millisecondes .
- 4.3.17. Le serveur doit être en mesure de mettre à jour la base de données à chaque connexion et partie jouée d'un utilisateur.
- 4.3.18. Le serveur doit stocker les formules génériques pour les personnalités des utilisateurs virtuels sur la base de données.
- 4.3.19. Le serveur doit supporter une partie de deux utilisateurs contre deux.
- 4.3.20. Le serveur doit être inaccessible aux utilisateurs pour un maximum de 2 heures lors d'une maintenance.**

4.4. Maintenabilité

- ~~4.4.1. Le serveur ne doit pas être accessible aux utilisateurs lors d'une maintenance durant un maximum de 2 heures.~~
- 4.4.2. Le code du client léger doit respecter les normes et standards de programmation du langage Kotlin disponible sur ce lien : <https://kotlinlang.org/docs/reference/coding-conventions.html>
- 4.4.3. Le code du client lourd doit respecter les normes et standards de programmation du langage Typescript disponible sur ce lien : <https://basarat.gitbook.io/typescript/styleguide>
- 4.4.4. Le code du serveur doit respecter les normes et standards de programmation du langage Go en utilisant l'outil de formatage standard (go fmt).
- 4.4.5. Le code du serveur doit respecter les normes et standards de programmation du langage Go en utilisant l'outil de *linting* (golint).
- 4.4.6. Le client lourd doit être codé en anglais.
- 4.4.7. Le client léger doit être codé en anglais.
- 4.4.8. Le serveur doit être codé en anglais.

4.5. Contraintes de conception

- 4.5.1. Le serveur doit être codé avec le langage Go.
- 4.5.2. Le client lourd doit être codé avec le cadriciel Angular en utilisant ~~les langages HTML, SCSS et Typescript~~ **le langage HTML.**

- 4.5.3. Le client lourd doit être une application Electron.
- 4.5.4. Le client léger doit être codé sur l'environnement de développement Android Studio ~~avec le langage Kotlin.~~
- 4.5.5. Le protocole de communication doit être websocket et REST.
- 4.5.6. La base de données doit être MongoDB.
- 4.5.7. Le serveur doit être hébergé sur Amazon Web Service.
- 4.5.8. Le client lourd doit être codé avec le cadriciel Angular en utilisant le langage SCSS.**
- 4.5.9. Le client lourd doit être codé avec le cadriciel Angular en utilisant le langage Typescript.**
- 4.5.10. Le client léger doit être codé avec le langage Kotlin.**

4.6. Sécurité

- 4.6.1. Le système doit crypter le mot de passe d'un utilisateur.
- 4.6.2. Le système doit permettre aux utilisateurs d'être les seuls à pouvoir voir le prénom, nom et mot de passe associés à leur profil.
- 4.6.3. Le système doit permettre l'affichage de l'historique de clavardage d'un canal uniquement aux personnes faisant partie du canal.

4.7. Portabilité

- 4.7.1. Le client lourd doit fonctionner sur Electron.
- 4.7.2. Le client lourd doit fonctionner sur le système d'exploitation Windows 10.
- 4.7.3. Le client léger doit fonctionner sur une tablette Samsung Galaxy Tab A 2019.