
Équipe 4

Poly Paint Pro
Spécifications des requis du système (SRS)

Version 3.0

Historique des révisions

Date	Version	Description	Auteur
2018-09-07	1.0	Séparation des tâches par section du document.	Audrey, Eric, J-P, Sébastien, Pascal
2018-09-08	1.01	Première rédaction des exigences d'utilisabilité.	Audrey
2018-09-08	1.02	Première rédaction des exigences de fiabilité, de maintenabilité et de sécurité	Sébastien
2018-09-08	1.03	Rédaction des exigences 4.7, de l'introduction ainsi que de la description globale.	Audrey
2018-09-09	1.04	Rédaction de la majorité des exigences fonctionnelles essentielles	Ayman
2018-09-11	1.05	Rédaction de quelques exigences fonctionnelles souhaitable	Ayman
2018-09-12	1.06	Rédaction de la section description globale	Audrey, Pascal
2018-09-12	1.07	Révision et rédaction d'exigences d'utilisabilité.	Audrey
2018-09-12	1.08	Révision et rédaction d'exigences de performance	Eric
2018-09-12	1.09	Révision globale du document	Mohamed Laziz Taouali
2018-09-13	1.10	Complétion des exigences fonctionnelles	Ayman
2018-09-13	1.11	Correction de la section 2 et implémentation des exigences d'utilisabilité	Pascal et Audrey
2018-09-13	1.12	Révision des exigences de fiabilité, de maintenabilité et de sécurité	Sébastien
2018-09-13	1.13	Correction d'exigences de performance	Eric
2018-09-13	1.14	Révision de tout le document.	Audrey, Eric, Sébastien, Pascal
2018-09-13	1.15	Mise en forme du document	Audrey
2018-09-13	1.16	Correction des fautes d'orthographe	Sébastien
2018-09-13	1.17	Ajout de contraintes de conception	Jean Paul
2018-09-14	2.0	Révision de la mise en page en vue de la remise de LOG3000	Audrey
2018-09-17	2.1	Révision et ajout d'exigences fonctionnelles personnalisées	Audrey
2018-09-18	2.2	Révision du format des exigences fonctionnelles et ajout d'exigences	Audrey
2018-09-20	2.3	Modification des exigences pour refléter les changements faits dans la liste des exigences.	Audrey

2018-09-24	2.4	Correction des exigences fonctionnelles : 3.1.1 à 3.1.9	Ayman
2018-09-25	2.5	Modification des exigences essentielles pour refléter la correction et modification du glossaire.	Audrey
2018-09-25	2.6	Révision des exigences fonctionnelles: 3.1.10 à 3.1.15 et toute la section 3.2.	Ayman
2018-09-25	2.7	Correction de la partie caractéristique des usagers.	Audrey
2018-09-26	2.8	Définition de ces termes dans le glossaire : forme basique, forme de connexion, utilisateur authentifié et droits administrateurs.	Ayman
2018-09-28	3.0	Correction des erreurs et de la mise en page.	Audrey et Sébastien

Table des matières

1. Introduction	5
1.1. But	5
1.2. Définitions, acronymes et abréviations	5
1.3. Vue d'ensemble du document	5
2. Description globale	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	6
3. Exigences fonctionnelles	7
3.1 Exigences essentielles du client lourd	7
3.2 Exigences essentielles du client léger	13
3.3. Exigences fonctionnelles souhaitables du client lourd	19
3.4. Exigences fonctionnelles souhaitables du client léger	22
4. Exigences non-fonctionnelles	24
4.1. Utilisabilité	24
4.2. Fiabilité	25
4.3. Performance	27
4.4. Maintenabilité	27
4.5. Contraintes de conception	28
4.6. Sécurité	29
4.7. Exigences de la documentation usager en ligne et du système d'assistance	29
4.8. Normes applicables	30
ANNEXE A: Glossaire	31

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

Pour la définition de mots, acronymes et abréviations, se référer à l'annexe A qui correspond au glossaire de ce document.

1.3. Vue d'ensemble du document

Le document suivant est présenté sous trois sections soit la description globale (section 2), les exigences fonctionnelles (section 3) et les exigences non-fonctionnelles (section 4). Également, une annexe est présentée à la fin du document.

2. Description globale

Le logiciel à développer, nommé Poly Paint Pro, est une version améliorée du logiciel Poly Paint qui permet de faire des dessins à l'aide de traits. Poly Paint Pro comprendra de nouvelles fonctionnalités comme l'édition en réseau, le clavardage, la possibilité de dessiner des formes et plus encore. Initialement, Poly Paint n'était seulement qu'une application Windows (client lourd). Cependant, une édition tablette (client léger), supportant la plateforme iOS, ainsi qu'un site web seront développés. De plus, certaines fonctionnalités de Poly Paint seront enlevées tandis que d'autres évolueront.

2.1. Caractéristiques des usagers

Les usagers de Poly Paint Pro n'ont pas besoin de connaissances particulières pour utiliser l'application et n'ont pas nécessairement utilisé Poly Paint auparavant. L'âge minimal des usagers n'a pas d'importance, il faut seulement qu'ils sachent lire, écrire et utiliser un ordinateur ou une tablette. L'application peut être utilisée dans n'importe quelle circonstance et par des personnes de tous les domaines. L'application peut être utilisée en ligne ou hors ligne, donc l'utilisateur n'a pas besoin d'internet pour l'utiliser. Par contre, le mode hors ligne ne comprend pas toutes les fonctionnalités disponibles. Puisque les utilisateurs peuvent créer des images simples à complexe (avec beaucoup de formes et connexion), l'interface sera épurée et l'application sera facile d'utilisation.

2.2. Interfaces

2.2.1. Interfaces usagers

Nous aurons trois types d'interfaces disponibles pour notre application: un client lourd pour ordinateur, un client léger pour iPad Mini 4, un site web et un serveur.

Les critères des interfaces devront respecter les critères ergonomiques de l'interaction entre humain et machine de Bastien et de Scapin. Nous tenterons d'implémenter ces critères le plus possible dans notre application, mais nous nous concentrerons surtout sur les principes de guidage, de charge cognitive, de gestion des erreurs et de consistance. De plus, le design des interfaces sera basé sur les jeux dessins et les applications de dessin et de clavardage les plus populaires.

2.2.2. Interfaces matérielles

Pour utiliser l'application, les utilisateurs utilisant le client lourd doivent avoir un ordinateur avec Windows 10 ayant une souris, un clavier et un écran. Pour ce qui est du client léger, l'utilisateur doit avoir accès à une tablette iPad Mini 4 disposant d'un écran tactile et d'un haut-parleur.

2.2.3. Interfaces logicielles

Le client lourd sera disponible sur Windows 10 et le client léger sera sur iOS. Les langages de programmation utilisés pour nos applications clients seront le C# et WPF (avec framework .NET) pour le client lourd et le Swift pour le client léger. Nous utiliserons Node.js pour l'application côté serveur et cette dernière sera déployée sur une machine Linux. Nous développerons aussi un site web codé en Typescript avec le cadriciel Angular qui sera disponible à partir d'un fureteur web comme Chrome ou Firefox. On utilisera des bases de données relationnelles et NoSQL pour gérer les données.

2.2.4. Interfaces de communication

Afin de pouvoir se connecter au serveur et d'accéder aux images, il est indispensable d'avoir un accès à internet à partir d'une adresse IP. Toutes les communications se feront par le protocole HTTP. Nous utiliserons les sockets afin d'établir la connexion client-serveur. Les communications entre le client lourd et le serveur utiliseront l'internet à partir du réseau filaire de Polytechnique ou à partir du réseau sans fil sécurisé eduroam. Les communications entre le client léger et le serveur se feront à travers l'internet. Le client léger se connectera à l'internet via le réseau sans fil sécurisé eduroam.

2.3. Contraintes générales

Un minimum de dix images doit pouvoir être édité en même temps. Chacune de ces images doit pouvoir être éditée par au moins cinq personnes à la fois. De plus, chaque image doit pouvoir supporter cinq canaux de communications. La cohérence visuelle est importante entre les applications. Les utilisateurs du client lourd et du client léger doivent pouvoir éditer la même image en même temps. La fluidité des clients est un aspect important pour la qualité d'utilisation de Poly Paint Pro.

Le client lourd est développé avec les langages C# et WPF et le client léger doit être développé en Swift. Le client léger fonctionnel avec l'iPad Mini 4 et le client lourd avec Windows 10. Le client lourd et le site web doivent pouvoir fonctionner sur les ordinateurs du local L-4810.

2.4. Hypothèses et dépendances

Nous supposons que l'utilisateur a accès au réseau sans fil eduroam. Évidemment, le débit du réseau peut affecter les performances de notre application pour les fonctionnalités qui utilisent internet. De plus, Poly Paint Pro doit respecter toutes les exigences spécifiées par le client.

3. Exigences fonctionnelles

La section suivante présente les exigences fonctionnelles de Poly Paint Pro qui décrivent les fonctionnalités du produit incluant le client léger et le client lourd et comprend également le site web.

3.1 Exigences essentielles du client lourd

3.1.1. *Clavardage - intégration*

- 3.1.1.1. Le système doit permettre à l'utilisateur de clavarder s'il est connecté à internet.
- 3.1.1.2. Le système doit permettre à chaque utilisateur authentifié ou non authentifié de clavarder.
- 3.1.1.3. Le système doit permettre à l'utilisateur de clavarder avec n'importe quelle personne en ligne.
- 3.1.1.4. Le système doit proposer un mode de clavardage intégré et une mode de clavardage fenêtre.
- 3.1.1.5. Le système doit empêcher l'utilisateur d'avoir une boîte de texte en mode fenêtré et une boîte de texte en mode intégré en même temps.
- 3.1.1.6. Le système doit permettre à l'utilisateur d'alterner entre les deux modes de clavardage à n'importe quel moment.
- 3.1.1.7. Le système doit proposer le mode fenêtré en tant que mode de clavardage par défaut.
- 3.1.1.8. Le système doit permettre à l'utilisateur d'alterner au mode fenêtré grâce à un bouton sur la boîte de texte du mode intégré.
- 3.1.1.9. Le système doit permettre à l'utilisateur d'alterner au mode intégré grâce à un bouton sur la boîte de texte du mode fenêtré.
- 3.1.1.10. Le système doit inclure la boîte de texte en mode intégré dans la même interface que celle du dessin.
- 3.1.1.11. Le système doit avoir une interface qui contient la liste des utilisateurs connectés.
- 3.1.1.12. Le système doit permettre à l'utilisateur de retrouver un utilisateur connecté grâce à une barre de recherche dans l'interface de la liste des utilisateurs connectés.

3.1.2. *Clavardage - Canaux de discussion*

- 3.1.2.1. Le système doit permettre à l'utilisateur de participer à plusieurs canaux de discussion en même temps.
- 3.1.2.2. Le système doit regrouper les canaux de discussion dans la même interface.
- 3.1.2.3. Le système doit afficher un onglet pour chaque canal de discussion actif dans l'interface de chat.
- 3.1.2.4. Le système doit permettre à l'utilisateur de créer un canal de discussion.
- 3.1.2.5. Le système doit permettre à l'utilisateur d'inviter des personnes en ligne au canal créé.

3.1.2.6. Le système doit avoir une liste de canaux de discussion actifs.

3.1.2.7. Le système doit permettre à l'utilisateur de trouver un canal de discussion actif grâce à une barre de recherche dans l'interface de la liste des canaux de discussion actifs.

3.1.2.8. Le système doit permettre à l'utilisateur de joindre n'importe quel canal de discussion actif depuis la liste des canaux de discussion actifs.

3.1.2.9. Le système doit permettre à l'utilisateur de quitter un canal de discussion.

3.1.3. Édition des formes

3.1.3.1. Le système doit permettre à l'utilisateur de placer un rectangle dans le canevas..

3.1.3.2. Le système doit permettre à l'utilisateur de placer un triangle dans le canevas.

3.1.3.3. Le système doit permettre à l'utilisateur de placer une ellipse dans le canevas.

3.1.3.4. Le système doit permettre à l'utilisateur de placer une ligne dans le canevas.

3.1.3.5. Le système doit permettre à l'utilisateur de placer une flèche directionnelle dans le canevas.

3.1.3.6. Le système doit permettre à l'utilisateur de placer une flèche bidirectionnelle dans le canevas.

3.1.3.7. Le système doit permettre à l'utilisateur de placer un diagramme de classe dans le canevas.

3.1.3.8. Le système doit permettre à l'utilisateur de placer un diagramme de cas d'utilisation dans le canevas.

3.1.3.9. Le système doit permettre à l'utilisateur de choisir la couleur des traits d'une forme basique.

3.1.3.10. Le système doit permettre à l'utilisateur de choisir la couleur des traits d'une forme connexion.

3.1.3.11. Le système doit permettre à l'utilisateur de choisir la couleur de remplissage d'une forme basique.

3.1.3.12. Le système doit permettre à l'utilisateur de choisir la taille des traits d'une forme basique.

3.1.3.13. Le système doit permettre à l'utilisateur de choisir la taille des traits d'une forme de connexion.

3.1.3.14. Le système doit permettre à l'utilisateur de visualiser n'importe quelle forme lors de sa création pour indiquer son emplacement et sa taille.

3.1.3.15. Le système doit créer un point d'ancrage sur chaque côté d'un triangle.

3.1.3.16. Le système doit créer un point d'ancrage sur chaque côté d'un rectangle.

3.1.3.17. Le système doit créer quatre points d'ancrage uniformément distribué sur la bordure d'une ellipse.

3.1.3.18. Le système doit permettre à l'utilisateur de déplacer n'importe quelle forme.

3.1.3.19. Le système doit permettre à l'utilisateur de retirer n'importe quelle forme.

3.1.3.20. Le système doit permettre à l'utilisateur de relier des points d'ancrage en utilisant une ligne ou une flèche.

3.1.3.21. Le système doit ajuster les liens lors du déplacement de la forme.

3.1.3.22. Le système doit permettre l'utilisateur de créer une liaison composée d'un ou plusieurs segments de droite.

- 3.1.3.23. Le système doit créer un point d'ancrage sur chaque côté d'un diagramme de classe.
- 3.1.3.24. Le système doit créer trois sous-rectangles à l'intérieur d'une classe.
- 3.1.3.25. Le système doit permettre à l'utilisateur de créer un lien d'héritage.
- 3.1.3.26. Le système doit permettre à l'utilisateur de créer un lien de composition.
- 3.1.3.27. Le système doit permettre à l'utilisateur de créer une association unidirectionnelle.
- 3.1.3.28. Le système doit permettre à l'utilisateur de créer une association bidirectionnelle.
- 3.1.3.29. Le système doit permettre à l'utilisateur de créer un lien d'agrégation.
- 3.1.3.30. Le système doit permettre à l'utilisateur de libeller une relation de classes.
- 3.1.3.31. Le système doit permettre à l'utilisateur de définir la multiplicité d'une relation de classes.
- 3.1.3.32. Le système doit permettre à l'utilisateur de placer un acteur dans le diagramme de cas d'utilisations.
- 3.1.3.33. Le système doit permettre à l'utilisateur de nommer un acteur.
- 3.1.3.34. Le système doit permettre à l'utilisateur de placer un cas d'utilisation.
- 3.1.3.35. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'association.
- 3.1.3.36. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'inclusion.
- 3.1.3.37. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'extension.
- 3.1.3.38. Le système doit permettre à l'utilisateur de libeller une relation de cas d'utilisation.
- 3.1.3.39. Le système doit permettre à l'utilisateur de définir la multiplicité d'une relation de cas d'utilisation.

3.1.4. Édition collaborative

- 3.1.4.1. Le système doit permettre à l'utilisateur de participer à une édition collaborative tant qu'il est connecté à internet.
- 3.1.4.2. Le système doit demander un mot de passe à l'utilisateur s'il veut participer à une édition collaborative d'une image protégée.
- 3.1.4.3. Le système doit permettre à l'utilisateur d'ajouter du texte flottant au canevas.
- 3.1.4.4. Le système doit permettre à l'utilisateur de modifier le texte en le sélectionnant.
- 3.1.4.5. Le système doit empêcher l'utilisateur de modifier un texte sélectionné par un autre utilisateur.
- 3.1.4.6. Le système doit permettre à l'utilisateur de réinitialiser le canevas à n'importe quel moment.
- 3.1.4.7. Le système doit rendre le canevas blanc et supprimer tous ses éléments de dessin quand il est réinitialisé.
- 3.1.4.8. Le système doit interrompre toutes les opérations en cours quand le canevas est réinitialisé.
- 3.1.4.9. Le système doit permettre à l'utilisateur de retirer de la surface du dessin le dernier trait dessiné.

- 3.1.4.10 Le système doit mettre la partie retirée par l'utilisateur dans sa pile.
- 3.1.4.11. Le système doit permettre à l'utilisateur de placer le trait du dessus de sa pile sur la surface du dessin.
- 3.1.4.12. Le système doit permettre à l'utilisateur de dupliquer un élément sélectionné.
- 3.1.4.13 Le système doit désélectionner l'utilisateur de l'élément après sa duplication
- 3.1.4.14. Le système doit rendre l'élément créé par la duplication sélectionné par l'utilisateur qui a exécuté cette action.
- 3.1.4.15. Le système doit permettre à l'utilisateur de coller un élément copié en utilisant le bouton "dupliquer" si aucun élément n'est sélectionné par cet utilisateur.
- 3.1.4.16. Le système doit permettre à l'utilisateur de sélectionner et couper une partie du canevas.
- 3.1.4.17 Le système doit supprimer la partie coupée du canevas.
- 3.1.4.18. Le système doit permettre à l'utilisateur de coller la partie coupée avec le bouton "Dupliquer".
- 3.1.4.19. Le système doit empêcher l'utilisateur de sélectionner une partie déjà sélectionnée par un autre utilisateur.
- 3.1.4.20. Le système doit permettre à l'utilisateur d'effacer une forme sélectionnée par lui.
- 3.1.4.21. Le système doit afficher chaque modification pour chaque utilisateur qui participe à l'édition collaborative de l'image.
- 3.1.4.22. Le système doit permettre à l'utilisateur de redimensionner une surface sélectionnée par lui.

3.1.5. Sauvegarde d'image et chargement

- 3.1.5.1. Le système doit sauvegarder l'image en cours d'édition à chaque intervalle de temps tant que l'application est connectée à internet.
- 3.1.5.2. Le système doit permettre à l'utilisateur de sauvegarder l'image localement.

3.1.6. Accessibilité des images

- 3.1.6.1. Le système doit permettre à l'utilisateur de créer un mot de passe pour son image afin de la rendre protégée s'il est authentifié à son compte.
- 3.1.6.2. Le système doit permettre à l'utilisateur d'accéder à une image protégée s'il saisit le bon mot de passe.
- 3.1.6.3. Le système doit permettre à l'utilisateur d'accéder à une image protégée sans saisir de mot de passe si il en est l'auteur.
- 3.1.6.4. Le système doit permettre à l'utilisateur de rendre son image non protégée à n'importe quel moment.
- 3.1.6.5. Le système doit éjecter tous les utilisateurs, sauf l'auteur, de l'édition de l'image si elle devient protégée.

3.1.7. Profil utilisateur et galerie

- 3.1.7.1. Le système doit permettre à l'utilisateur de créer un compte.
- 3.1.7.2. Le système doit permettre à l'utilisateur de s'authentifier à l'application avec son compte.

- 3.1.7.3. Le système doit permettre à l'utilisateur de consulter les images publiques.
- 3.1.7.4. Le système doit présenter une galerie d'images privées pour chaque utilisateur authentifié.
- 3.1.7.5. Le système doit permettre à l'utilisateur authentifié de consulter sa galerie privée.
- 3.1.7.6. Le système doit empêcher l'utilisateur de consulter la galerie privée d'un autre utilisateur.
- 3.1.7.7. Le système doit permettre à l'utilisateur authentifié de rendre son image privée.
- 3.1.7.8. Le système doit permettre à l'utilisateur authentifié de rendre son image publique.
- 3.1.7.9. Le système doit permettre à l'utilisateur authentifié de modifier son pseudonyme.
- 3.1.7.10. Le système doit permettre à l'utilisateur authentifié de modifier son mot de passe.
- 3.1.7.11. Le système doit afficher les images sous forme de liste
- 3.1.7.12. Le système doit permettre à l'utilisateur de visualiser une image en la sélectionnant de la liste.
- 3.1.7.13. Le système doit permettre l'utilisateur de passer à l'édition de l'image lors de sa visualisation grâce à un bouton.
- 3.1.7.14. Le système doit demander un mot de passe

3.1.8. Site web

- 3.1.8.1. Le système doit permettre à l'utilisateur de consulter la liste des images publiques.
- 3.1.8.2. Le système doit afficher l'auteur d'une image.
- 3.1.8.3. Le système doit afficher la date de création d'une image.
- 3.1.8.4. Le système doit afficher l'état de protection d'une image.
- 3.1.8.5. Le système doit afficher le nom d'une image.
- 3.1.8.6. Le système doit permettre à l'utilisateur de visualiser l'image publique en la sélectionnant de la liste.
- 3.1.8.7. Le système doit permettre à l'utilisateur de consulter la liste des profils des utilisateurs inscrits.
- 3.1.8.8. Le système doit afficher le nom d'un profil.
- 3.1.8.9. Le système doit afficher le nombre d'images publiques d'un profil.
- 3.1.8.10. Le système doit permettre à l'utilisateur de consulter le profil d'un utilisateur inscrit.
- 3.1.8.11. Le système doit permettre à l'utilisateur de consulter les images publiques d'un utilisateur inscrit à partir de son profil.
- 3.1.8.12. Le système doit afficher une barre de recherche dans l'interface de la liste des profils.
- 3.1.8.13. Le système doit afficher une barre de recherche dans l'interface de la liste des images.
- 3.1.8.14. Le système doit permettre à l'utilisateur de filtrer la liste des images par auteur.
- 3.1.8.15. Le système doit permettre à l'utilisateur de filtrer la liste des profils par nom.
- 3.1.8.16. Le système doit permettre à l'utilisateur de s'authentifier à son profil.
- 3.1.8.17. Le système doit permettre à l'utilisateur authentifié de consulter son image privée.
- 3.1.8.18. Le système doit permettre à l'utilisateur authentifié de modifier son pseudonyme.

- 3.1.8.19. Le système doit permettre à l'utilisateur authentifié de modifier son mot de passe.
- 3.1.8.20. Le système doit permettre à l'administrateur de s'authentifier sur le site web.
- 3.1.8.21. Le système doit permettre à l'administrateur de consulter n'importe quel profil.
- 3.1.8.22. Le système doit permettre l'administrateur de consulter n'importe quelle image publique.
- 3.1.8.23. Le système doit permettre l'administrateur de consulter n'importe quelle image privée.
- 3.1.8.24. Le système doit permettre à l'administrateur de créer un utilisateur.
- 3.1.8.25. Le système doit permettre à l'administrateur de supprimer un utilisateur.
- 3.1.8.26. Le système doit déconnecter un utilisateur authentifié qui vient d'être supprimé.
- 3.1.8.27. Le système doit supprimer les images publiques d'un utilisateur supprimé.
- 3.1.8.28. Le système doit permettre à l'administrateur de supprimer une image d'un profil.
- 3.1.8.29. Le système doit éjecter chaque utilisateur qui participe à l'édition d'une image qui vient d'être supprimée.
- 3.1.8.30. Le système doit permettre à l'administrateur d'attribuer le droit de créer un compte utilisateur à un utilisateur.
- 3.1.8.31. Le système doit permettre à l'administrateur de retirer le droit de créer un compte utilisateur à un utilisateur.
- 3.1.8.32. Le système doit permettre à l'administrateur d'attribuer le droit de supprimer un compte utilisateur à un utilisateur.
- 3.1.8.33. Le système doit permettre à l'administrateur de retirer le droit de supprimer un compte utilisateur à un utilisateur.
- 3.1.8.34. Le système doit permettre à l'administrateur d'attribuer le droit de supprimer une image à un utilisateur.
- 3.1.8.35. Le système doit permettre à l'administrateur de retirer le droit de supprimer une image à un utilisateur.
- 3.1.8.36. Le système doit permettre à l'administrateur d'attribuer le droit d'attribution de droits administrateur à un utilisateur.
- 3.1.8.37. Le système doit permettre à l'administrateur de retirer le droit d'attribution de droits administrateur à un utilisateur.

3.1.9. Tutoriel

- 3.1.9.1. Le système doit proposer une suite d'images qui expliquent le fonctionnement du mode d'édition lorsqu'un utilisateur l'utilise pour la première fois.
- 3.1.9.2. Le système doit permettre à l'utilisateur de passer d'une image explicative à une autre grâce à un bouton.
- 3.1.9.3. Le système doit permettre à l'utilisateur de fermer le tutoriel grâce à un bouton.
- 3.1.9.4. Le système doit permettre à l'utilisateur de consulter le tutoriel de nouveau à n'importe quel moment.

3.1.10. Galerie - J'aime et commentaires

3.1.10.1. Le système doit permettre à l'utilisateur, lorsqu'il est authentifié, de mettre un "J'aime" sur une image publique.

3.1.10.2. Le système doit permettre à l'utilisateur, après qu'il ait mis un "J'aime" sur une image publique de retirer ce "J'aime".

3.1.10.3. Le système doit empêcher l'utilisateur non authentifié de mettre un "J'aime" pour une image publique.

3.1.10.4 Le système doit empêcher l'utilisateur non authentifié de mettre un commentaire pour une image publique.

3.1.10.5. Le système doit permettre à l'utilisateur authentifié de mettre un commentaire pour une image publique.

3.1.10.6. Le système doit afficher les commentaires d'une image publique.

3.1.10.7. Le système doit afficher les "J'aime" d'une image publique.

3.1.10.8. Le système doit permettre à l'utilisateur d'actualiser la page afin de voir les nouveaux "J'aime".

3.1.10.9. Le système doit permettre à l'utilisateur d'actualiser la page afin de voir les nouveaux commentaires..

3.1.11. Photos profil

3.1.11.1. Le système doit permettre à l'utilisateur authentifié de choisir une photo de profil.

3.1.11.2. Le système doit permettre à l'utilisateur de modifier sa photo de profil avec une image depuis son disque dur.

3.1.11.3. Le système doit mettre par défaut une photo de profil vierge comme image de profil si l'utilisateur n'a pas de photo de profil.

3.1.12. Gestion des amis

3.1.12.1. Le système doit permettre à l'utilisateur authentifié d'ajouter des utilisateurs authentifiés (en ligne) en tant qu'amis, à partir de la liste des utilisateurs.

3.1.12.2. Le système doit permettre à l'utilisateur authentifié d'ajouter des utilisateurs hors ligne en tant qu'amis à partir de la liste des utilisateurs.

3.1.12.2. Le système doit permettre à l'utilisateur de consulter sa liste d'amis.

3.1.12.3. Le système doit permettre à l'utilisateur de débiter une discussion avec un ami à partir de la liste de ses amis.

3.1.12.4. Le système doit permettre à l'utilisateur de retirer des amis de sa liste d'amis à n'importe quel moment.

3.1.12.5. Le système doit permettre à l'utilisateur d'accepter une demande d'amitié provenant d'un autre utilisateur.

3.1.12.6. Le système doit permettre à l'utilisateur de refuser une demande d'amitié provenant d'un autre utilisateur.

3.1.12.7. Le système doit ajouter un ami dans la liste d'amis d'un utilisateur lorsque l'ami a accepté la demande d'amitié.

3.2 Exigences essentielles du client léger

3.2.1. Clavardage - intégration

3.2.1.1. Le système doit permettre à l'utilisateur de clavarder s'il est connecté à internet.

3.2.1.2. Le système doit permettre à chaque utilisateur authentifié ou non authentifié de clavarder.

3.2.1.3. Le système doit permettre à l'utilisateur de clavarder avec n'importe quelle personne en ligne.

3.2.1.4. Le système doit avoir une interface de clavardage intégrée.

3.2.1.5 Le système doit permettre à l'utilisateur d'accéder à l'interface de clavardage à tout moment.

3.2.1.6. Le système doit générer un effet visuel lorsque l'utilisateur reçoit un message.

3.2.1.7. Le système doit générer un effet sonore lorsque l'utilisateur reçoit un message.

3.2.1.8. Le système doit enlever l'effet visuel lorsque l'utilisateur ouvre tous les canaux de discussion contenant un nouveau message.

3.2.1.9. Le système doit avoir une interface qui contient la liste des utilisateurs connectés.

3.2.1.10. Le système doit permettre à l'utilisateur de retrouver un utilisateur connecté grâce à une barre de recherche dans l'interface de la liste des utilisateurs connectés.

3.2.2. Clavardage - Canaux de discussion

3.2.2.1. Le système doit permettre à l'utilisateur de participer à plusieurs canaux de discussion en même temps.

3.2.2.2. Le système doit regrouper les canaux de discussion dans la même interface.

3.2.2.3 Le système doit afficher un onglet pour chaque canal de discussion actif dans l'interface de chat.

3.2.2.4. Le système doit permettre à l'utilisateur de créer un canal de discussion.

3.2.2.5. Le système doit permettre à l'utilisateur d'inviter des personnes en ligne au canal créé.

3.2.2.6. Le système doit avoir une liste de canaux de discussion actifs.

3.2.2.7. Le système doit permettre à l'utilisateur de trouver un canal de discussion actif grâce à une barre de recherche dans l'interface de la liste des canaux de discussion actifs.

3.2.2.8. Le système doit permettre à l'utilisateur de joindre n'importe quel canal de discussion actif depuis la liste des canaux de discussion actifs.

3.2.2.9. Le système doit permettre à l'utilisateur de quitter un canal de discussion.

3.2.3. Édition des formes

3.2.3.1. Le système doit permettre à l'utilisateur de placer un rectangle dans le canevas..

3.2.3.2. Le système doit permettre à l'utilisateur de placer un triangle dans le canevas.

3.2.3.3. Le système doit permettre à l'utilisateur de placer une ellipse dans le canevas.

3.2.3.4. Le système doit permettre à l'utilisateur de placer une ligne dans le canevas.

- 3.2.3.5. Le système doit permettre à l'utilisateur de placer une flèche directionnelle dans le canevas.
- 3.2.3.6. Le système doit permettre à l'utilisateur de placer une flèche bidirectionnelle dans le canevas.
- 3.2.3.7. Le système doit permettre à l'utilisateur de placer un diagramme de classe dans le canevas.
- 3.2.3.8. Le système doit permettre à l'utilisateur de placer un diagramme de cas d'utilisation dans le canevas.
- 3.2.3.9. Le système doit permettre à l'utilisateur de choisir la couleur des traits d'une forme basique.
- 3.2.3.10. Le système doit permettre à l'utilisateur de choisir la couleur des traits d'une forme connexion.
- 3.2.3.11. Le système doit permettre à l'utilisateur de choisir la couleur de remplissage d'une forme basique.
- 3.2.3.12. Le système doit permettre à l'utilisateur de choisir la taille des traits d'une forme basique.
- 3.2.3.13. Le système doit permettre à l'utilisateur de choisir la taille des traits d'une forme de connexion.
- 3.2.3.14. Le système doit permettre à l'utilisateur de visualiser n'importe quelle forme lors de sa création pour indiquer son emplacement et sa taille.
- 3.2.3.15. Le système doit créer un point d'ancrage sur chaque côté d'un triangle.
- 3.2.3.16. Le système doit créer un point d'ancrage sur chaque côté d'un rectangle.
- 3.2.3.17. Le système doit créer quatre points d'ancrage uniformément distribué sur la bordure d'une ellipse.
- 3.2.3.18. Le système doit permettre à l'utilisateur de déplacer n'importe quelle forme.
- 3.2.3.19. Le système doit permettre à l'utilisateur de retirer n'importe quelle forme.
- 3.2.3.20. Le système doit permettre à l'utilisateur de relier des points d'ancrage en utilisant une ligne ou une flèche.
- 3.2.3.21. Le système doit ajuster les liens lors du déplacement de la forme.
- 3.2.3.22. Le système doit permettre l'utilisateur de créer une liaison composée d'un ou plusieurs segments de droite.
- 3.2.3.23. Le système doit créer un point d'ancrage sur chaque côté d'un diagramme de classe.
- 3.2.3.24. Le système doit créer trois sous-rectangles à l'intérieur d'une classe.
- 3.2.3.25. Le système doit permettre à l'utilisateur de créer un lien d'héritage.
- 3.2.3.26. Le système doit permettre à l'utilisateur de créer un lien de composition.
- 3.2.3.27. Le système doit permettre à l'utilisateur de créer une association unidirectionnelle.
- 3.2.3.28. Le système doit permettre à l'utilisateur de créer une association bidirectionnelle.
- 3.2.3.29. Le système doit permettre à l'utilisateur de créer un lien d'agrégation.
- 3.2.3.30. Le système doit permettre à l'utilisateur de libeller une relation de classes.
- 3.2.3.31. Le système doit permettre à l'utilisateur de définir la multiplicité d'une relation de classes.
- 3.2.3.32. Le système doit permettre à l'utilisateur de placer un acteur dans le diagramme de cas d'utilisations.

3.2.3.33. Le système doit permettre à l'utilisateur de nommer un acteur.

3.2.3.34. Le système doit permettre à l'utilisateur de placer un cas d'utilisation.

3.2.3.35. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'association.

3.2.3.36. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'inclusion.

3.2.3.37. Le système doit permettre à l'utilisateur de relier un acteur et un cas d'utilisation avec une relation d'extension.

3.2.3.38. Le système doit permettre à l'utilisateur de libeller une relation de cas d'utilisation.

3.2.3.39. Le système doit permettre à l'utilisateur de définir la multiplicité d'une relation de cas d'utilisation.

3.2.4. Édition collaborative

3.2.4.1. Le système doit permettre à l'utilisateur de participer à une édition collaborative tant qu'il est connecté à internet.

3.2.4.2. Le système doit demander un mot de passe à l'utilisateur s'il veut participer à une édition collaborative d'une image protégée.

3.2.4.3. Le système doit permettre à l'utilisateur d'ajouter du texte flottant au canevas.

3.2.4.4. Le système doit permettre à l'utilisateur de modifier le texte en le sélectionnant.

3.2.4.5. Le système doit empêcher l'utilisateur de modifier un texte sélectionné par un autre utilisateur.

3.2.4.6. Le système doit permettre à l'utilisateur de réinitialiser le canevas à n'importe quel moment.

3.2.4.7 Le système doit rendre le canevas blanc et supprimer tous ses éléments de dessin quand il est réinitialisé.

3.2.4.8. Le système doit interrompre toutes les opérations en cours quand le canevas est réinitialisé.

3.2.4.9. Le système doit permettre à l'utilisateur de retirer de la surface du dessin le dernier trait dessiné.

3.2.4.10 Le système doit mettre la partie retirée par l'utilisateur dans sa pile.

3.2.4.11. Le système doit permettre à l'utilisateur de placer le trait du dessus de sa pile sur la surface du dessin.

3.2.4.12. Le système doit permettre à l'utilisateur de dupliquer un élément sélectionné.

3.2.4.13 Le système doit désélectionner l'utilisateur de l'élément après sa duplication

3.2.4.14. Le système doit rendre l'élément créé par la duplication sélectionné par l'utilisateur qui a exécuté cette action.

3.2.4.15. Le système doit permettre à l'utilisateur de coller un élément copié en utilisant le bouton "dupliquer" si aucun élément n'est sélectionné par cet utilisateur.

3.2.4.16. Le système doit permettre à l'utilisateur de sélectionner et couper une partie du canevas.

3.2.4.17 Le système doit supprimer la partie coupée du canevas.

3.2.4.18. Le système doit permettre à l'utilisateur de coller la partie coupée avec le bouton "Dupliquer".

3.2.4.19. Le système doit empêcher l'utilisateur de sélectionner une partie déjà sélectionnée par un autre utilisateur.

3.2.4.20. Le système doit permettre à l'utilisateur d'effacer une forme sélectionnée par lui.

3.2.4.21. Le système doit afficher chaque modification pour chaque utilisateur qui participe à l'édition collaborative de l'image.

3.2.4.22. Le système doit permettre à l'utilisateur de redimensionner une surface sélectionnée par lui.

3.2.5. Sauvegarde d'image et chargement

3.2.5.1. Le système doit sauvegarder l'image en cours d'édition à chaque intervalle de temps tant que l'application est connectée à internet.

3.2.5.2. Le système doit permettre à l'utilisateur de sauvegarder l'image localement.

3.2.6. Accessibilité des images

3.2.6.1. Le système doit permettre à l'utilisateur de créer un mot de passe pour son image afin de la rendre protégée s'il est authentifié à son compte.

3.2.6.2. Le système doit permettre à l'utilisateur d'accéder à une image protégée s'il saisit le bon mot de passe.

3.2.6.3. Le système doit permettre à l'utilisateur d'accéder à une image protégée sans saisir de mot de passe s'il en est l'auteur.

3.2.6.4. Le système doit permettre à l'utilisateur de rendre son image non protégée à n'importe quel moment.

3.2.6.5. Le système doit éjecter tous les utilisateurs, sauf l'auteur, de l'édition de l'image si elle devient protégée.

3.2.7. Profil utilisateur et galerie

3.2.7.1. Le système doit permettre à l'utilisateur de créer un compte.

3.2.7.2. Le système doit permettre à l'utilisateur de s'authentifier à l'application avec son compte.

3.2.7.3. Le système doit permettre à l'utilisateur de consulter les images publiques.

3.2.7.4. Le système doit présenter une galerie d'images privées pour chaque utilisateur authentifié.

3.2.7.5. Le système doit permettre à l'utilisateur authentifié de consulter sa galerie privée.

3.2.7.6. Le système doit empêcher l'utilisateur de consulter la galerie privée d'un autre utilisateur.

3.2.7.7. Le système doit permettre à l'utilisateur authentifié de rendre son image privée.

3.2.7.8. Le système doit permettre à l'utilisateur authentifié de rendre son image publique.

3.2.7.9. Le système doit permettre à l'utilisateur authentifié de modifier son pseudonyme.

3.2.7.10. Le système doit permettre à l'utilisateur authentifié de modifier son mot de passe.

3.2.7.11. Le système doit afficher les images sous forme de liste

3.2.7.12. Le système doit permettre à l'utilisateur de visualiser une image en la sélectionnant de la liste.

3.2.7.13. Le système doit permettre l'utilisateur de passer à l'édition de l'image lors de sa visualisation grâce à un bouton.

3.2.7.14. Le système doit demander un mot de passe

3.2.8. Tutoriel

3.2.8.1. Le système doit proposer une suite d'images qui expliquent le fonctionnement du mode d'édition lorsqu'un utilisateur l'utilise pour la première fois.

3.2.8.2. Le système doit permettre à l'utilisateur de passer d'une image explicative à une autre grâce à un bouton.

3.2.8.3. Le système doit permettre à l'utilisateur de fermer le tutoriel grâce à un bouton.

3.2.8.4. Le système doit permettre à l'utilisateur de consulter le tutoriel de nouveau à n'importe quel moment.

3.2.9. Interface utilisateur

3.2.9.1. L'utilisateur peut utiliser une gesture dans plusieurs scénarios d'utilisation. Cette gesture est la translation d'un élément dans le canevas.

3.2.9.2. L'utilisateur peut utiliser deux gestures de façon concurrent dans au moins un scénario d'utilisation. Ces deux gestures sont la translation et l'agrandissement d'un élément dans le canevas.

3.2.10. Effets visuels et sonores

3.2.10.1. Le système doit permettre à l'application d'avoir un effet sonore lors d'une notification de la réception d'une image.

3.2.10.2. Le système doit permettre à l'application d'avoir un effet sonore lors d'une notification de la réception d'une demande d'amitié.

3.2.10.3. Le système doit permettre à l'application d'avoir un effet visuel lorsque l'utilisateur accède à un canal de discussion. Les messages se dérouleront à l'écran.

3.2.10.4. Le système doit permettre à l'application d'avoir un effet visuel lorsque l'utilisateur affiche le menu en touchant le bouton réservé à cet effet. Le menu glissera à partir d'un côté de l'écran.

3.2.11. Galerie - J'aime et commentaires

3.2.11.1. Le système doit permettre à l'utilisateur, lorsqu'il est authentifié, de mettre un "J'aime" sur une image publique.

3.2.11.2. Le système doit permettre à l'utilisateur, après qu'il ait mis un "J'aime" sur une image publique de retirer ce "J'aime".

3.2.11.3. Le système doit empêcher l'utilisateur non authentifié de mettre un "J'aime" pour une image publique.

3.2.11.4. Le système doit empêcher l'utilisateur non authentifié de mettre un commentaire pour une image publique.

3.2.11.5. Le système doit permettre à l'utilisateur authentifié de mettre un commentaire pour une image publique.

3.2.11.6. Le système doit afficher les commentaires d'une image publique.

3.2.11.7. Le système doit afficher les "J'aime" d'une image publique.

3.2.11.8. Le système doit permettre à l'utilisateur d'actualiser la page afin de voir les nouveaux "J'aime".

3.2.11.9. Le système doit permettre à l'utilisateur d'actualiser la page afin de voir les nouveaux commentaires.

3.2.12. Photos profil

3.2.12.1. Le système doit permettre à l'utilisateur authentifié de choisir une photo de profil.

3.2.12.2. Le système doit permettre à l'utilisateur de modifier sa photo de profil avec une image depuis son disque dur.

3.2.12.3. Le système doit mettre par défaut une photo de profil vierge comme image de profil si l'utilisateur n'a pas de photo de profil.

3.2.13. Gestion des amis

3.2.13.1. Le système doit permettre à l'utilisateur authentifié d'ajouter des utilisateurs authentifiés (en ligne) en tant qu'amis, à partir de la liste des utilisateurs.

3.2.13.2. Le système doit permettre à l'utilisateur authentifié d'ajouter des utilisateurs hors ligne en tant qu'amis à partir de la liste des utilisateurs.

3.2.13.3. Le système doit permettre à l'utilisateur de consulter sa liste d'amis.

3.2.13.4. Le système doit permettre à l'utilisateur de débiter une discussion avec un ami à partir de la liste de ses amis.

3.2.13.5. Le système doit permettre à l'utilisateur de retirer des amis de sa liste d'amis à n'importe quel moment.

3.2.13.6. Le système doit permettre à l'utilisateur d'accepter une demande d'amitié provenant d'un autre utilisateur.

3.2.13.7. Le système doit permettre à l'utilisateur de refuser une demande d'amitié provenant d'un autre utilisateur.

3.2.13.8. Le système doit ajouter un ami dans la liste d'amis d'un utilisateur lorsque l'ami a accepté la demande d'amitié.

3.3. Exigences fonctionnelles souhaitables du client lourd

3.3.1. Édition collaborative

3.3.1.1. Le système doit permettre à l'utilisateur d'insérer une image dans le canevas à partir d'un fichier choisi.

3.3.1.2. Le système doit sélectionner et réplique automatiquement une image lorsque celle-ci est ajoutée par l'utilisateur.

3.3.1.3. Le système doit permettre à l'utilisateur d'effectuer une rotation d'un élément qu'il a sélectionné dans le canevas.

3.3.1.4. Le système doit permettre à l'utilisateur d'exporter une image sous format PNG.

3.3.1.5. Le système doit permettre à l'utilisateur d'exporter une image sous format JPG.

3.3.2 Édition des formes

3.3.2.1. Le système doit permettre à l'utilisateur d'ajouter un texte à l'intérieur d'une forme basique.

- 3.3.2.2. Le système doit permettre à l'utilisateur d'ajouter un texte à l'intérieur d'une forme de connexion.
- 3.3.2.3. Le système doit faire flotter le texte ajouté à une forme de connexion autour de cette forme.
- 3.3.2.4. Le système doit permettre à l'utilisateur de déplacer le texte d'une forme de connexion.
- 3.3.2.5. Le système doit permettre à l'utilisateur de redimensionner une forme de connexion.
- 3.3.2.6. Le système doit permettre à l'utilisateur de redimensionner un diagramme.
- 3.3.2.7. Le système doit permettre à l'utilisateur de modifier la couleur du trait des formes basiques et des formes de connexion.
- 3.3.2.8. Le système doit permettre à l'utilisateur de modifier la couleur de remplissage d'une forme basique.
- 3.3.2.9. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait plein.
- 3.3.2.10. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait pointillé.
- 3.3.2.11. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait d'union.
- 3.3.2.12. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait plein.
- 3.3.2.13. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait pointillé.
- 3.3.2.14. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait d'union.
- 3.3.2.15. Le système doit permettre à l'utilisateur de modifier l'épaisseur de trait d'une forme de connexion.

3.3.3. Tutoriel

- 3.3.3.1. Le système doit permettre à l'utilisateur de relancer le tutoriel à n'importe quel moment.
- 3.3.3.2. Le système doit permettre à l'utilisateur de quitter le tutoriel.
- 3.3.3.3. Le système doit permettre à l'utilisateur de réaliser des actions précises afin de progresser à travers le tutoriel.

3.3.4. Sauvegarde d'image et chargement

- 3.3.4.1. Le système doit synchroniser toutes les images du profil actuel lorsqu'il y a un changement.
- 3.3.4.2. Le système doit synchroniser toutes les images du profil actuel lorsqu'un client se connecte.

3.3.5. Galerie - J'aime et commentaires

- 3.3.5.1. Le système doit mettre à jour dynamiquement (en temps réel) les commentaires, lorsqu'un utilisateur consulte une image.
- 3.3.5.2. Le système doit mettre à jour dynamiquement (en temps réel) les "J'aimes", lorsqu'un utilisateur consulte une image.

3.3.6. Accessibilité des images - Téléchargement des images à partir de la galerie

3.3.6.1. Le système doit permettre à l'utilisateur de télécharger une image publique de la galerie à partir du site web et la sauvegarder localement.

3.3.6.2. Le système doit permettre à l'utilisateur de télécharger une image privée, dont il est l'auteur, à partir du site web et de la sauvegarder localement.

3.3.7. Accessibilité des images - Partage d'image par raccourci d'URL

3.3.7.1. Le système doit permettre à l'utilisateur de partager via un lien URL une image privée à un utilisateur anonyme.

3.3.7.2. Le système doit permettre à l'utilisateur anonyme de consulter sur le site web une image privée via le lien URL qu'un utilisateur lui a partagé.

3.3.8. Meilleur filtre pour chercher les images

3.3.8.1. Le système doit permettre à l'utilisateur de nommer une image lors de sa création.

3.3.8.2. Le système doit permettre à l'utilisateur de rechercher une image par son nom.

3.3.9. Historique des versions - Création et restauration

3.3.9.1. Le système doit permettre à l'utilisateur de restaurer une version précédente d'une image.

3.3.9.2. Le système doit permettre à l'utilisateur de consulter l'historique d'une image.

3.3.9.3. Le système doit enregistrer dans l'historique d'une image toutes ses versions à partir de sa création.

3.3.9.4. Le système doit créer une version d'une image automatiquement à la fin d'une séance d'édition (lorsque tous les utilisateurs ont quitté le mode édition).

3.3.9.5. Le système doit permettre à une image de contenir tous les éléments de l'image.

3.3.9.6. Le système doit permettre à une image de contenir l'historique des actions d'édition.

3.3.9.7. Le système doit copier l'image restituée par l'utilisateur au moment de la création de la version par l'utilisateur.

3.3.9.8. Le système doit automatiquement transférer l'utilisateur à l'édition de l'image lorsqu'il a restitué une version.

3.3.9.9. Le système doit permettre à l'utilisateur qui édite une image de continuer à l'éditer sans dérangement, lorsqu'un autre utilisateur restaure une version de cette image.

3.3.10. Historique des versions - Création avancée

3.3.10.1. Le système doit permettre à l'utilisateur de créer une version manuellement en cliquant sur le bouton réservé à cette fin.

3.3.11. Historique des versions - Historique

3.3.11.1. Le système doit garder tout l'historique des actions d'édition de la version restaurée d'une image lorsque l'utilisateur restaure une version de cette image.

3.3.12. Historique des versions - Comparaison

3.3.12.1. Le système doit permettre à l'utilisateur de visionner deux versions différentes d'une image.

3.3.12.2. Le système doit afficher les deux versions à comparer côte à côte.

3.3.13. Contrôle de la police

3.3.13.1. Le système doit permettre à l'utilisateur de modifier la police des libellés lors de l'édition d'une image.

3.3.13.2. Le système doit permettre à l'utilisateur de choisir parmi 10 polices.

3.3.13.3. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en gras.

3.3.13.4. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en italique.

3.3.13.5. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en souligné.

3.4. Exigences fonctionnelles souhaitables du client léger

3.4.1. Édition collaborative

3.4.1.1. Le système doit permettre à l'utilisateur d'insérer une image dans le canevas à partir d'un fichier choisi.

3.4.1.2. Le système doit sélectionner et réplique automatiquement une image lorsque celle-ci est ajoutée par l'utilisateur.

3.4.1.3. Le système doit permettre à l'utilisateur d'effectuer une rotation d'un élément qu'il a sélectionné dans le canevas.

3.4.1.5. Le système doit permettre à l'utilisateur d'exporter une image sous format PNG.

3.4.1.6. Le système doit permettre à l'utilisateur d'exporter une image sous format JPG.

3.4.2. Profil utilisateur et galerie

3.4.2.1. Le système doit afficher les images dans un présentoir sophistiqué et dynamique.

3.4.2.2. Le système doit permettre à l'utilisateur de retourner à l'image précédente en faisant glisser l'image présentée à droite.

3.4.2.3. Le système doit permettre à l'utilisateur de passer à l'image suivante en faisant glisser l'image présentée à gauche.

3.4.3 Édition des formes

3.4.3.1. Le système doit permettre à l'utilisateur d'ajouter un texte à l'intérieur d'une forme basique.

3.4.3.2. Le système doit permettre à l'utilisateur d'ajouter un texte à l'intérieur d'une forme de connexion.

3.4.3.3. Le système doit faire flotter le texte ajouté à une forme de connexion autour de cette forme.

3.4.3.4. Le système doit permettre à l'utilisateur de déplacer le texte d'une forme de connexion.

3.4.3.5. Le système doit permettre à l'utilisateur de redimensionner une forme de connexion.

3.4.3.5. Le système doit permettre à l'utilisateur de redimensionner un diagramme.

3.4.3.6. Le système doit permettre à l'utilisateur de modifier la couleur du trait des formes basiques et des formes de connexion.

3.4.3.7. Le système doit permettre à l'utilisateur de modifier la couleur de remplissage d'une forme basique.

3.4.3.8. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait plein.

3.4.3.9. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait pointillé.

3.4.3.10. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme basique en trait d'union.

3.4.3.11. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait plein.

3.4.3.12. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait pointillé.

3.4.3.13. Le système doit permettre à l'utilisateur de modifier le style de trait d'une forme de connexion en trait d'union.

3.4.3.14. Le système doit permettre à l'utilisateur de modifier l'épaisseur de trait d'une forme de connexion.

3.4.4. Tutoriel

3.4.4.1. Le système doit permettre à l'utilisateur de relancer le tutoriel à n'importe quel moment.

3.4.4.2. Le système doit permettre à l'utilisateur de quitter le tutoriel.

3.4.4.3. Le système doit permettre à l'utilisateur de réaliser des actions précises afin de progresser à travers le tutoriel.

3.4.5. Sauvegarde d'image et chargement

3.4.5.1. Le système doit synchroniser toutes les images du profil actuel lorsqu'il y a un changement.

3.4.5.2. Le système doit synchroniser toutes les images du profil actuel lorsqu'un client se connecte.

3.4.6. Interface utilisateur

3.4.6.1. Le système doit permettre à l'utilisateur d'utiliser trois gestes de façon concurrent dans au moins un scénario d'utilisation. Ces gestes sont la translation, l'agrandissement et la rotation d'un élément dans le canevas.

3.4.7. Effets visuels et sonores

3.4.7.1. Le système doit permettre à l'application d'avoir un effet sonore lorsqu'un utilisateur ajoute un commentaire à une photo.

3.4.7.2. Le système doit permettre à l'application d'avoir un effet visuel lorsqu'un commentaire à une photo est ajouté.

3.4.8. Accessibilité des images - Partage d'image par raccourci d'URL

3.4.8.1. Le système doit permettre à l'utilisateur de partager via un lien URL une image privée à un utilisateur anonyme.

3.4.8.2. Le système doit permettre à l'utilisateur anonyme de consulter sur le site web une image privée via le lien URL qu'un utilisateur lui a partagé.

3.4.9. Meilleur filtre pour chercher les images

3.4.9.1. Le système doit permettre à l'utilisateur de nommer une image lors de sa création.

3.4.9.2. Le système doit permettre à l'utilisateur de rechercher une image par son nom.

3.4.10. Contrôle de la police

3.4.10.1. Le système doit permettre à l'utilisateur de modifier la police des libellés lors de l'édition d'une image.

3.4.10.2. Le système doit permettre à l'utilisateur de choisir parmi 10 polices.

3.4.10.3. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en gras.

3.4.10.4. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en italique.

3.4.10.5. Le système doit permettre à l'utilisateur de changer le style de police d'un libellé en souligné.

4. Exigences non-fonctionnelles

La section suivante comprend les exigences non-fonctionnelles de Poly Paint Pro. Les exigences non-fonctionnelles correspondent à des exigences par rapport à la qualité du produit.

4.1. Utilisabilité

4.1.1. Après avoir fait le tutoriel, l'utilisateur sera en mesure d'utiliser les fonctionnalités du mode édition.

Le tutoriel doit permettre à l'utilisateur d'apprendre rapidement, après une lecture, comment utiliser les fonctionnalités d'édition de Poly Paint Pro. Ces fonctionnalités incluent l'édition d'un dessin, de ses formes, et de l'édition collaborative.

4.1.2. Après avoir exploré la fonctionnalité de clavardage, l'utilisateur sera en mesure de clavarder avec d'autres usagers.

L'utilisateur sera en mesure d'utiliser toutes les fonctionnalités liées au clavardage après avoir accédé aux interfaces de clavardages. Ces fonctionnalités seront très similaires à ce qui est développé dans d'autres logiciels et applications web, donc clavarder se fera intuitivement.

4.1.3. Un nouvel utilisateur pourra se créer et accéder à son compte en moins de 2 minutes.

Un nouvel usager pourra se créer un nouveau compte d'utilisateur et y accéder en moins de 2 minutes.

4.1.4. Une icône de chargement apparaîtra lorsque l'application prendra plus de 1 seconde à répondre.

Pour indiquer à l'utilisateur que l'application prend un certain délai à effectuer une action, une icône de chargement va s'afficher dans le milieu de l'écran après une seconde de délai. Cela permettra d'indiquer à l'utilisateur que l'application est toujours active et que l'écran n'a pas figé.

4.1.5. Tous les boutons suivront une convention.

Afin de permettre à l'utilisateur d'identifier facilement un bouton, les exigences suivantes ont été définies.

- 4.1.5.1. Tous les boutons contenant du texte auront la même forme et réagiront de la même manière pour tous les états.

La seule caractéristique qui peut différencier les boutons contenant du texte est la couleur.

- 4.1.5.2. Les boutons avec une image ou une image et un texte n'auront pas nécessairement les mêmes formes, mais ils réagiront de la même façon pour tous les états.

Ces types de boutons incluent ceux avec une image ou ceux avec une image en plus d'un texte. Leur forme peut varier, ainsi que leur couleur. Par contre, ils réagiront de la même façon pour les différents états.

- 4.1.5.3. Un bouton possède quatre états soit normal, survolé, cliqué et désactivé.

Tous les boutons, peu importe leurs formes et caractéristiques, possèdent quatre états. L'état survolé est lorsque le curseur est au-dessus du bouton (sans le cliquer). L'état cliqué est lorsque le bouton est cliqué avec le curseur. L'état inactif est lorsque le bouton est désactivé, il n'effectuera aucune action lorsqu'il est cliqué. Finalement, l'état normal est lorsque le bouton n'est pas survolé ni cliqué ni désactivé.

- 4.1.5.4. Lorsque le curseur survole un bouton ou un item d'un menu, ceux-ci changent de couleur.

Lorsque le curseur survole un bouton ou un item du menu, ce qui correspond à l'état survolé, ce dernier changera de couleur.

- 4.1.6. *Il n'y aura pas plus de trois niveaux dans les menus.*

Il y a un maximum de trois niveaux pour les menus. Cela permettra à l'utilisateur d'effectuer toute action en moins de trois cliques.

- 4.1.7. *Les sous-menus regroupent les fonctions en lien avec le menu parent.*

Avec ces regroupements, cela va créer un entonnoir de recherche qui filtre les opérations désirables afin de faciliter leur accès à l'utilisateur.

- 4.1.8. *Les menus sont présentés de façon uniforme graphiquement.*

Les différents menus auront la même forme, même organisation selon les différents clients (lourd et léger) et le site web.

- 4.1.9. *Uniformité des éléments des menus à travers les clients et le site web pour les fonctionnalités qui sont identiques.*

Les fonctionnalités identiques à travers les clients et le site web seront représentées par les mêmes noms dans les différents menus.

- 4.1.10. *Si la connexion au serveur est perdue après plus de 10 secondes, l'utilisateur sera informé par une fenêtre popup.*

Suite à la perte de connexion avec le serveur, l'utilisateur utilise maintenant l'application hors ligne donc, il deviendra un utilisateur anonyme. Après 10 secondes de hors connexion, une fenêtre popup apparaîtra pour avertir l'utilisateur.

- 4.1.11. *Les interfaces doivent être agréables à regarder pour l'utilisateur, être intuitives et ergonomiques.*

Les interfaces seront simples, jolies et demanderont une faible charge cognitive à l'utilisateur. Elles respecteront les critères de Bastien et Scapin suivants: guidage, charge de travail, signifiante des codes, gestion des erreurs, adaptabilité et contrôle explicite.

4.1.12. Deux actions opposées auront des formes ou des couleurs différentes.

Deux actions comme “Défaire” et “Refaire” seront représentées par des symboles ou des couleurs opposées (ou différents).

4.1.13. Le site web est adapté pour toutes les plateformes.

Le site web sera disponible en version mobile, sur le client lourd et le client léger.

4.2. Fiabilité

4.2.1. Disponibilité

Le serveur et le site web doivent être disponible 99.9% du temps lors de la journée entre 6:00 et 22:00. Un bloc de 15 minutes est réservé de 3:45 à 4:00 pour les mises à jour du serveur et le site web. Le serveur et le site web doivent être disponible minimum 23h par jours.

4.2.2. Fréquence d'une panne

Une panne doit survenir au maximum une fois par jour. De plus, il doit y avoir un maximum de 10 pannes par mois.

4.2.3. Temps moyen jusqu'à la réparation

Une réparation minimale doit être faite dans un délai de 90 minutes suite à une panne pour remettre en service le système. Une révision de code pour s'assurer que le défaut qui a causé la panne ne soit plus présent dans le programme doit être faite dans les 24 heures suite à la panne.

4.2.4. Disponibilité de l'application mobile

L'application mobile (iOS) doit être disponible sur l'App Store 100% du temps.

4.3. Performance

- 4.3.1. *Le système doit gérer au moins 100 canaux de discussions concurrents.*
- 4.3.2. *Le système doit transmettre un message envoyé par un utilisateur dans un canal de discussion à tous les autres participants du canal en moins de 200 ms.*
- 4.3.3. *Le système doit charger une image sauvegardée en moins d'une seconde.*
- 4.3.4. *Le système doit supporter au moins 10 actions d'édition de formes par seconde.*
- 4.3.5. *Le système doit sauvegarder localement chaque changement fait à une image en moins de 300 ms.*
- 4.3.6. *Le système doit sauvegarder à distance chaque changement fait à une image en moins de 500 ms.*
- 4.3.7. *Le système doit traiter chaque recherche par nom d'image en moins de 500 ms.*
- 4.3.8. *Le système doit restaurer une version précédente de l'image en moins d'une seconde si les fonctionnalités d'historique des versions sont développées.*
- 4.3.9. *Le système doit permettre à l'utilisateur de voir au moins 10 images dans la galerie en moins d'une seconde s'il y a au moins 10 images déjà sauvegardées.*
- 4.3.10. *Le système doit supporter au moins 5 utilisateurs concurrents par image.*
- 4.3.11. *Le système doit supporter l'édition d'au moins 10 images en concurrence.*
- 4.3.12. *Le système doit créer un socket pour chaque canal de discussion.*
- 4.3.13. *Le système doit avoir au moins 5 Go de disque pour stocker les images.*

4.4. Maintenabilité

- 4.4.1. *Un serveur de backup pouvant supporter au moins 50% de la charge habituelle doit en tout temps être en marche. Si un des clients ne peut pas se connecter au serveur habituel, il tente de se connecter au serveur backup à la place.*
- 4.4.2. *Les administrateurs de Poly Paint peuvent gérer les comptes utilisateurs en cas de problème par le biais de leurs sessions web.*
 - 4.4.2.1. *Les administrateurs peuvent consulter les profils et images de tous les autres utilisateurs.*
 - 4.4.2.2. *Les administrateurs peuvent créer et supprimer des utilisateurs.*
 - 4.4.2.3. *Les administrateurs peuvent changer la protection de n'importe quelle image.*
 - 4.4.2.4. *Les administrateurs peuvent supprimer les images privées et publiques.*
 - 4.4.2.5. *Les administrateurs peuvent attribuer et retirer des droits d'administration aux autres utilisateurs.*
- 4.4.3. *Le code doit respecter les normes de codage de l'équipe.*
 - 4.4.3.1. *Une fonction n'a qu'une seule responsabilité. Les fonctions devraient être le plus possible des "fonctions pures"*
 - 4.4.3.2. *Une indentation de quatre espaces est utilisée.*

- 4.4.3.3. *L'ouverture de l'accolade doit se faire sur la même ligne que la commande de bloc.*
- 4.4.3.4. *L'entête de fichier doit contenir le nom du créateur du fichier, de ses différents auteurs, ainsi qu'une brève description du contenu du fichier.*
- 4.4.3.5. *Les lignes de doivent pas dépasser 80 colonnes (caractères).*
- 4.4.3.6. *Le langage de programmation est l'anglais.*
- 4.4.3.7. *Les variables doivent être écrites en camelCase. Le nom des constantes doit être en majuscules (e.g. SPEED_OF_LIGHT = 300000). Le nom de fonctions et classes doit être en PascalCase pour le C# et en camelCase pour les autres langages.*

4.5. Contraintes de conception

- 4.5.1. *Langage de programmation pour le client lourd*
Le langage de programmation utilisé pour notre application sera le C# pour le client lourd.
- 4.5.2. *Langage de programmation pour le client léger*
Le langage de programmation utilisé pour notre application sera le langage Swift pour le client léger.
- 4.5.3. *Langage de programmation pour le serveur*
Le langage de programmation utilisé dans le cas du serveur sera le Typescript
- 4.5.4. *Cadriciel pour le serveur*
Le cadriciel sera Node.js pour notre serveur.
- 4.5.5. *Langage de programmation pour le site web*
Le langage de programmation utilisé pour notre application sera le langage Typescript pour le site web.
- 4.5.6. *Cadriciel pour le site web*
Le cadriciel sera Angular pour notre site web.
- 4.5.7. *Environnement de travail pour le client lourd*
L'environnement de développement sera Visual Studio pour le C#.
- 4.5.8. *Environnement de travail pour le client léger*
L'environnement de développement sera XCode pour le Swift.
- 4.5.9. *Méthodologie de travail*
Le produit sera développé en mode itératif soit en 4 sprints de 2 semaines respectivement.
- 4.5.10. *Début de sprint*
Un sprint débutera par la planification des fonctionnalités qui seront implémentées lors des prochaines semaines
- 4.5.11. *Fin de sprint*
La fin d'un sprint se conclura par un retour en arrière sur la qualité du dernier livrable.

4.6. Sécurité

- 4.6.1. Les mots de passe doivent être "hash" avant d'être sauvegardés dans la base de données.*
- 4.6.2. La base de données doit bloquer toute tentative de connexion autre que celles envoyées par le serveur.*
- 4.6.3. Les images privées doivent être cachées pour tous les utilisateurs sauf les administrateurs et le propriétaire de l'image.*
- 4.6.4. Les images privées volontairement partagées par un utilisateur doivent être disponibles en mode vue seulement.*

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

- 4.7.1. Un bouton d'aide sera présent dans toutes les interfaces.*

Lorsque cliqué, le bouton d'aide affichera une rubrique d'aides pour l'utilisateur. La rubrique présentera le tutoriel, ainsi que des conseils si nécessaire.

- 4.7.2. Le tutoriel sera disponible en tout temps pour l'utilisateur.*

L'utilisateur pourra consulter le tutoriel lors de sa première entrée dans l'application et pourra le consulter à nouveau à partir du menu ou en cliquant sur le bouton d'aide dans n'importe quelle interface.

- 4.7.3. Les messages d'erreurs indiquent à l'utilisateur les détails vulgarisés (pas de termes techniques) et comment régler l'erreur.*

Lorsqu'une erreur survient, l'utilisateur sera immédiatement averti par l'apparition d'une fenêtre popup ou par un champ texte en rouge. Les détails de l'erreur seront indiqués (sans termes techniques) et comment l'utilisateur peut régler cette erreur.

- 4.7.4. Si l'utilisateur détecte une anomalie, il pourra la rapporter aux développeurs de l'application en leur écrivant un courriel.*

Si l'utilisateur rencontre une anomalie, il pourra envoyer un courriel aux développeurs de Poly Paint Pro. Cette information sera disponible dans la rubrique d'aide de l'application.

4.8. Normes applicables

4.8.1. Le logiciel de gestion de versions est Git pour chacune des parties de PolyPaint.

4.8.1.1. Le développement est fait sur la branche “develop” puis le tout est mis sur la branche “master” lorsque le produit est prêt à être livré.

4.8.1.2. Il ne peut y avoir qu’une fonctionnalité par branche.

4.8.1.3. Systématiquement faire un merge de la branche “develop” avant de commencer tout travail sur une branche.

4.8.1.4. Avant chaque merge d’une branche dans “develop”, merger le “develop” dans la branche, résoudre les conflits et s’assurer que tous les tests passent.

4.8.1.5. Les messages de commits doivent clairement énoncer les changements apportés.

4.8.1.6. Un commit doit toucher au maximum 50 lignes.

4.8.1.7. Des pull requests doivent être faits et révisé par au minimum une autre personne dans l’équipe avant tout merge dans la master.

4.8.2. Les tests doivent respecter la liste d’exigence choisie par l’équipe

4.8.2.1. Chaque fonction a au moins 2 tests dont minimalement un common use case et minimalement un edge case.

4.8.2.2. Ne jamais faire de push lorsqu’une ou plusieurs fonctions n’ont pas de tests.

4.8.2.3. Ne pas faire de push sur une branche si tous les tests ne passent pas avec succès.

ANNEXE A: Glossaire

Terme	Description
Commande de bloc	if, while, for, etc.
Commit	Il s'agit de la commande effectuée lorsqu'un changement a été fait dans le code.
Fonction pure	Une fonction pure est une fonction qui va toujours retourner les mêmes résultats pour les mêmes arguments. Elle ne va pas non plus avoir d'effets secondaires.
Forme basique	Une forme basique est soit un rectangle, une ellipse ou un triangle.
Forme de connexion	Une forme de connexion est une ligne, un flèche directionnelle ou une flèche bidirectionnelle.
Hash	Fonction mathématique unidirectionnelle qui calcule une empreinte presque unique pour chaque entrée. Les résultats ont des fortes probabilités d'être uniques (environ 1 sur 2^{256} pour SHA-256).
Image privée	Image dont seul l'auteur peut y avoir accès.
Image protégée	Image qui est seulement accédée à partir d'un mot de passe valide.
Image publique	Image disponible à tous les usagers (administrateur, utilisateur ayant un compte et utilisateur anonyme).
Forme basique	Rectangle, triangle ou ellipse.
Forme de connexion	Flèche ou ligne.
Utilisateur authentifié	Un utilisateur qui s'est authentifié à son profil.
Droits administrateur	Les droits de création et de suppression de profils, suppression de dessins et de gestion de droits. La gestion des droits est le fait d'attribuer ou de retirer ces droits administrateur à un utilisateur.
Merge	Il s'agit de l'opération de combiner deux branches sur GitHub.
Mode fenêtré	Une fenêtre à part de l'application.
Mode intégré	Fait partie de l'application.
Point d'ancrage	Point où l'élément peut s'accrocher à un autre élément.
Popup	Message que l'application envoie à l'utilisateur.
Pull Request	Lorsqu'une requête est effectuée pour merge le dernier commit effectué sur une branche locale.
Sockets	Port de communication TCP permettant d'envoyer et de recevoir des messages à d'autres ports sur le même ordinateur et sur d'autres ordinateurs.
Texte flottant	Texte pouvant être déplacé sans restriction.
Utilisateur anonyme	Utilisateur ne possédant pas de compte dans l'application ou n'étant pas connecté.