Équipe 102

Fais-moi un dessin
Plan de tests logiciels

Version 1.1

Historique des révisions

Date	Version	Description	Auteur
2021-04-10	1.0	Rédaction initiale du document	Vlad Drelciuc
2021-04-19	1.1	Révision du document	Andi Podgorica

Table des matières

Plan de tests logiciels

1. Introduction

Ce document sert à planifier les tests que nous souhaitons effectuer sur notre logiciel. En particulier, la section 2 présentera les cas d'utilisations, les exigences fonctionnelles et les exigences non-fonctionnelles que nous testerons. La section 3 présentera les types de tests que nous souhaitons effectuer ainsi que les outils dont nous disposons. La section 4 détaillera les ressources dont nous disposerons pour effectuer nos tests. Enfin, la section 5 décrira l'échéancier de tests que nous comptons suivre.

2. Exigences à tester

2.1. Exigences fonctionnelles du client lourd

Exigences du SRS	**Types de tests associés**
3.1.1 Clavardage - intégration	Tests de fonction, tests d'interface usager
3.1.2 Canaux de discussion	Tests de fonction, tests d'interface usager, tests de stress
3.1.3 Profil utilisateur et historique	Tests de fonction, tests de sécurité et de contrôle d'accès
3.1.4 Options de jeu	Tests de fonction, tests d'interface usager
3.1.5 Lobby	Tests de fonction
3.1.6 Mode de jeu classique	Tests de fonction, tests d'interface usager, tests de performance, tests de volume, tests d'échec/récupération
3.1.7 Mode de jeu sprint solo	Tests de fonction, tests d'interface usager, tests de performance, tests de volume, tests d'échec/récupération
3.1.8 Mode de jeu sprint coopératif	(Non implémenté)
3.1.9 Choix de mots	(Non implémenté)
3.1.10 Création d'une paire mot-image	Tests de fonction, tests d'interface usager, tests de volume, tests d'intégrité des données
3.1.11 Personnalité des joueurs virtuels	Tests de fonction
3.1.12 Effets visuels et sonores	Tests de fonction, tests d'interface usager
3.1.13 Tutoriel	Tests de fonction, tests d'interface usager
3.1.14 Classements des joueurs	Tests de fonction, tests de performance

3.1.15 Progression / niveau	Tests de fonction, tests d'intégrité des données
3.1.16 Magasin	Tests de fonction, tests d'interface usager, tests de sécurité et de contrôle d'accès
3.1.17 Monnaie virtuelle	Tests de fonction, tests de sécurité et de contrôle d'accès
3.1.18 Réactions	(Non implémenté)
3.1.19 Badges	Tests de fonction, tests d'interface usager
3.1.20 Musique d'arrière-plan	Tests de fonction

2.2. Exigences fonctionnelles du client léger

Exigences du SRS	**Types de tests associés**
3.2.1 Clavardage - intégration	Tests de fonction, tests d'interface usager
3.2.2 Clavardage - canaux de discussions	Tests de fonction, tests d'interface usager, tests de stress
3.2.3 Profil utilisateur et historique	Tests de fonction, tests de sécurité et de contrôle d'accès
3.2.4 Options de jeu	Tests de fonction, tests d'interface usager
3.2.5 Lobby	Tests de fonction
3.2.6 Mode de jeu classique	Tests de fonction, tests d'interface usager, tests de performance, tests de volume, tests d'échec/récupération
3.2.7 Mode de jeu sprint solo	Tests de fonction, tests d'interface usager, tests de performance, tests de volume, tests d'échec/récupération
3.2.8 Mode de jeu sprint coopératif	(Non implémenté)
3.2.9 Choix des mots	(Non implémenté)
3.2.10 Personnalité des joueurs virtuels	Tests de fonction
3.2.11 Effets visuels et sonores	Tests de fonction, tests d'interface usager
3.2.12 Tutoriel	Tests de fonction, tests d'interface usager
3.2.13 Classements des joueurs	Tests de fonction, tests de performance
3.2.14 Progression / niveau	Tests de fonction, tests d'intégrité des données
3.2.15 Badges	Tests de fonction, tests d'interface usager

3.2.16 Musique d'arrière-plan	Tests de fonction

2.3. Exigences non-fonctionnelles

Exigences du SRS	**Types de tests associés**
4.1 Utilisabilité	Tests d'interface usager
4.2 Fiabilité	Tests de stress
4.3 Performance	Tests de performance
4.4 Maintenabilité	(Pas testable directement)
4.5 Contraintes de conception	(Pas testable directement)
4.6 Sécurité	Tests de sécurité et de contrôle d'accès
4.7 Exigences de la documentation	Tests d'interface usager

3. Stratégie de test

3.1. Types de test

3.1.1. Tests de fonction

Objectif de test:	Valider l'implémentation correcte des cas d'utilisation de l'application.
Technique:	Exécuter une liste d'actions permettant de valider une fonctionnalité ciblée.
Critère de complétion:	Vérifier que les résultats réels sont les mêmes que les résultats attendus.
Considérations spéciales:	Les cas d'utilisation à tester devraient refléter les exigences du SRS.

3.1.2. Tests d'interface usager

Objectif de test:	Valider que l'interface usager soit intuitive et adaptée aux usagers de l'application.
Technique:	Faire exécuter une liste d'actions à des utilisateurs externes faisant partie du public cible de l'application et prendre note de leurs commentaires et durée d'exécution.
Critère de complétion:	Les utilisateurs externes parviennent à accomplir la liste d'actions demandées dans un délai d'apprentissage correspondant à nos exigences.
Considérations spéciales:	Aucune.

3.1.3. Tests d'intégrité des données

Objectif de test:	Valider que les interactions avec la base de données ne corrompent pas les données.
Technique:	Envoyer une série de données depuis un client et valider qu'elles ont bien été reçues et stockées dans la base de données afin de valider qu'un autre client les a lus correctement.
Critère de complétion:	Les données sont inchangées entre l'envoi et la réception.
Considérations spéciales:	Des données telles que les dates pourraient avoir un format différent entre les clients et le serveur, mais la valeur elle-même devrait être la même partout.

3.1.4. Tests de performance

Objectif de test:	Valider que les performances de l'application correspondent à nos exigences du SRS.
Technique:	Vérifier la taille des dossiers de build de l'application. Vérifier la consommation de RAM. Chronométrer le temps entre l'envoi d'une requête et la réception de la réponse du serveur.
Critère de complétion:	Les mesures de performance (telles que la latence) respectent les exigences du SRS pour les deux clients.
Considérations spéciales:	Aucune.

3.1.5. Tests de stress

Objectif de test:	Valider que l'envoi de plusieurs requêtes simultanées n'affecte pas l'intégrité des données.
Technique:	Envoyer plusieurs requêtes en même temps depuis plusieurs clients et valider l'ordre de réception.
Critère de complétion:	Tous les clients reçoivent toutes les requêtes qui les concernent et dans le bon ordre.
Considérations spéciales:	Nous testons principalement les messages car c'est l'élément qui a le plus grand potentiel de recevoir une grande charge provenant de plusieurs clients différents en même temps. Concernant le mode de jeu Classique, nous n'avons pas à nous en soucier car chaque utilisateur va dessiner quand son tour viendra, alors nous craignons moins qu'il y ait des conflits dans la séquence d'instructions.

3.1.6. Tests de volume

Objectif de test:	Valider qu'une grande quantité de données ne cause pas de panne de notre application ou dans notre base de données.
Technique:	Enregistrer des éléments très volumineux dans la base de données (ex: paire mot-image avec beaucoup de traits) et essayer de les lire depuis un autre client.
Critère de complétion:	L'écriture et la lecture d'éléments volumineux ne devraient pas causer de panne dans l'application.
Considérations spéciales:	Aucune.

3.1.7. Tests de sécurité et de contrôle d'accès

Objectif de test:	Valider la sûreté de l'application.
Technique:	Tenter d'accéder à une page privée (ex: aire de jeu) sans se connecter au préalable. Modifier manuellement les données dans la mémoire locale (localstorage) de l'application et vérifier le comportement de l'application.
Critère de complétion:	Les pages privées de l'application ne devraient pas être accessibles sans connexion préalable. La modification de la mémoire locale ne devrait pas donner accès aux informations du compte d'un autre utilisateur.
Considérations spéciales:	Aucune.

3.1.8. Tests d'échec/récupération

Objectif de test:	Valider qu'une fermeture soudaine de l'application d'un des clients ne compromet pas l'intégrité des données d'une partie en cours.
Technique:	Forcer la fermeture de l'application durant une partie et constater l'impact sur les autres joueurs dans la partie.
Critère de complétion:	Les autres utilisateurs devraient être notifiés de la fin de la partie. Les données de la partie devraient rester intactes dans la base de données.
Considérations spéciales:	Aucune.

3.2. Outils

Les outils suivants seront utilisés au sein de la discipline de test:

Type de test	Outil
Tests de performance	Task Manager de Windows 10
Tests de performance	Developer Tools de Google Chrome
Tests d'intégrité des données	UI web de Firestore

4. Ressources

4.1. Équipe de test

Rôle	Membre de l'équipe	Responsabilités
Testeur	Vlad Drelciuc	Tests de fonction pour le serveur, test d'intégrité des données, test de sécurité et de contrôle d'accès
Testeur	Benjamin Boucher	Tests de fonction pour le client lourd, tests d'interface usager du client lourd, test de performance, test de stress
Testeur	Justin Caisse	Tests de fonction pour le client léger, tests d'interface usager du client léger, test de volume, test d'échec/récupération

4.2. Système

- Tous les tests du client lourd seront effectués sur des ordinateurs roulant sous Windows 10 et ayant un minimum de 8 GB de RAM.
- Tous les tests du client léger Android se feront dans une émulation d'une tablette Samsung Galaxy de 10.1po ayant 2 GB de RAM. L'émulation se fera depuis Android Studio sur des ordinateurs roulant sous Windows 10 et ayant un minimum de 8 GB de RAM.
- Les tests nécessitant de consulter le contenu de la base de données Firestore seront faits dans un navigateur web Google Chrome depuis un ordinateur roulant sous Windows 10 et ayant un minimum de 8 GB de RAM.

5. Jalons du projet

Jalon	Effort	Date de début	Date de fin
Tests de fonction	10	2021-04-14	2021-04-19
Tests d'interface usager	2	2021-04-16	2021-04-17
Tests d'intégrité des données	2	2021-04-17	2021-04-17
Tests de performance	2	2021-04-17	2021-04-17
Tests de stress	2	2021-04-15	2021-04-15
Tests de volume	2	2021-04-15	2021-04-16
Tests de sécurité et de contrôle d'accès	2	2021-04-15	2021-04-16
Tests d'échec/récupération	2	2021-04-16	2021-04-17