Équipe 210

Draw It Up!
Plan de tests logiciels

Version 1.2

Historique des révisions

Date	Version	Description	Auteur
2021-04-15	1.0	Introduction et création des sections	Marc-André Filion
2021-04-16	1.1	Avancement des sections 3.1, 4.1 et 5	Marc-André Filion
2021-04-19	1.2	Révision finale de l'équipe avant la remise	Équipe 210

Table des matières

Plan de tests logiciels

1. Introduction

Ce document a pour but de présenter la planification prévue en ce qui a trait aux tests de l'application Fais-moi un dessin afin d'en assurer la qualité. Il sera d'abord question des exigences fonctionnelles et non fonctionnelles, conformément à la spécification des requis, qui seront testés. Ensuite sera présentée la stratégie de test qui fera part des différents types de tests prévus, en indiquant les objectifs visés par chacun, la technique et les outils à employer, ainsi que le critère de complétion. Enfin, il sera question des ressources nécessaires avec les jalons associés aux étapes importantes de la discipline de tests.

2. Exigences à tester

Exigences	Tests associés
3.1 Clavardage	Tests de fonction, d'intégrité des données et de performance
3.2. Profil de l'utilisateur	Tests de fonction, test de sécurité et de contrôle d'accès
3.3. Modes de jeu	Tests de fonction, d'intégrité des données, de performance et de stress
3.4. Création d'une paire mot-image	Tests de fonction, d'intégrité des données et de performance
3.5. Tutoriel	Tests de fonction, d'interface usager
3.6. Trophée	Tests de fonction
3.7. Classement	Tests de fonction, d'interface usager
3.8. Joueur virtuel	Tests de fonction, de performance
3.10. Support pour mot de passe oublié	Tests de fonction, d'interface usager, de sécurité et de contrôle d'accès
3.11. Effets visuels et sonores	Test de fonction et d'interface usager
3.12. Filtre de blasphème	Tests de fonction

4.1. Convivialité	Tests d'interface usager
4.2. Fiabilité	Test d'échec/récupération
4.3. Performance	Test de charge, de stress et de volume
4.6. Sécurité	Test de sécurité et de contrôle d'accès

3. Stratégie de test

3.1. Types de test

3.1.1. Test de fonction

Objectif de test:	Vérifier que chaque fonctionnalité assure le rôle qu'elle doit accomplir et assure conformité envers les cas d'utilisation rattachés.
Technique:	Suivre les étapes indiquées dans les cas d'utilisation et observer si celles-ci sont adéquates par rapport à l'application.
Critère de complétion:	Les cas d'utilisation représentent bien les étapes nécessaires à accomplir l'exigence.
Considérations spéciales:	

3.1.2. Tests d'interface usager

Objectif de test:	Vérifier que les utilisateurs aient accès à l'information dont ils ont besoin, et rapidement.
Technique:	Procéder à des tests utilisateurs avec 6 individus. Noter les commentaires de rétroaction.
Critère de complétion:	Recevoir un avis de satisfaction auprès de la majorité (4/6) des individus.
Considérations spéciales:	Les individus doivent être familiers avec les ordinateurs/tablettes.

3.1.3. Tests d'intégrité des données

Objectif de test:	S'assurer que les attributs devant être définis pour chaque entrée dans les collections le sont. (Validité)
Technique:	Envoi de requêtes à la base de données pour obtenir les objets ayant un attribut portant une valeur non définie (vide).
Critère de complétion:	Les requêtes ne renvoient aucun élément.
Considérations spéciales:	

3.1.4. Tests de performance

Objectif de test:	Vérifier que les échanges de données se font dans un temps raisonnable et respectent les durées définies dans le SRS.
Technique:	Profiler l'application lors d'opérations client-serveur et analyser les temps de réponse
Critère de complétion:	Les chiffres obtenus sont inférieurs aux valeurs définies dans le SRS
Considérations spéciales:	

3.1.5. Tests de charge

Objectif de test:	Vérifier qu'une partie avec le maximum de joueurs possible demeure en temps réel.
Technique:	Effectuer des parties en mode classique avec le nombre maximal de joueurs (4) et noter l'expérience de jeu
Critère de complétion:	L'expérience est fluide et ne crée pas de distractions par rapport à la partie en cours.
Considérations spéciales:	

3.1.6. Tests de stress

Objectif de test:	Vérifier que les salons de clavardage puissent transmettre les messages des utilisateurs en grande quantité et les traiter rapidement en respectant l'ordre d'arrivée. Vérifier qu'un nombre élevé de parties puissent être jouées simultanément sans que cela ait un impact sur l'ensemble des joueurs.
Technique:	Lancer un grand nombre de parties en mode classique à nombre de joueurs variables, ainsi qu'en mode solo, tout en envoyant beaucoup de messages dans le clavardage, puis observer les impacts sur le délai de traitement des échanges humain-ordinateur.
Critère de complétion:	Aucun impact n'est observé ou celui-ci est négligeable.
Considérations spéciales:	

3.1.7. Tests de volume

Objectif de test:	Vérifier que la limite de requêtes que le serveur peut traiter en une seconde est grande et est adéquate dans le contexte de notre jeu de dessin simple.
Technique:	Envoyer en rafale des requêtes au serveur.
Critère de complétion:	Le serveur tombe en panne après avoir traité au moins 2000 requêtes.
Considérations spéciales:	Utiliser jmeter

3.1.8. Tests de sécurité et de contrôle d'accès

Objectif de test:	Vérifier que les paquets réseau contiennent les mots de passe sous forme cryptée et qu'ils sont protégés contre le *sniffing* (reniflage), pour ainsi qu'il n'y a pas d'accès illégal à l'application.
Technique:	Analyser les paquets du réseau et tenter de repérer les mots de passe.
Critère de complétion:	Aucun mot de passe trouvé n'est apparu sous sa réelle valeur.
Considérations spéciales:	

3.1.9. Tests d'échec/récupération

Objectif de test:	Vérifier que le serveur est capable de se rétablir lors de défaillances afin de demeurer en ligne.
Technique:	Exécuter des actions selon les cas limites (*edge cases*) afin d'essayer d'introduire une erreur au sein du serveur.
Critère de complétion:	Le serveur n'est pas tombé en panne
Considérations spéciales:	

3.2. Outils

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

À noter que la version du client roulant sur Android sera émulée, alors nous n'aurons pas besoin de tablette sous Android.

Type de test	Outil
Fonction	PC sous Windows 10
Interface usager	PC sous Windows 10
Intégrité des données	PC sous Windows 10, serveur AWS et base de données MongoDB
Performance	PC sous Windows 10
Charge	PC sous Windows 10, serveur AWS
Stress	PC sous Windows 10, serveur AWS
Volume	PC sous Windows 10, serveur AWS, jmeter
Sécurité et contrôle d'accès	PC sous Windows 10, serveur AWS, base de données MongoDB, Wireshark
Échec/récupération	PC sous Windows 10, serveur AWS

4. Ressources

4.1. Équipe de test

Rôle	Membre de l'équipe	Responsabilités
Testeur du serveur	Benjamin	Écrire et exécuter les tests relatifs au serveur et rapporter les résultats
Testeur du client lourd	Vincenzo	Écrire et exécuter les tests relatifs au client lourd et rapporter les résultats
Testeur du client léger	Charles-Édouard	Écrire et exécuter les tests relatifs au client léger et rapporter les résultats
Conception des tests pour exigences fonctionnelles et non-fonctionnelles	Marc-André	Rédiger les cas de tests conformément à la stratégie de tests et les exigences fonctionnelles et non fonctionnelles à tester
Correction des bogues	Gérard	Corriger les anomalies logicielles en fonction des résultats des tests
Correction des bogues	Bouchra	Corriger les anomalies logicielles en fonction des résultats des tests

4.2. Système

Aucune ressource système particulière n'est requise. Nous n'aurons besoin que d'un PC pour chaque testeur et que celui-ci soit configuré avec le système d'opération Windows 10, avec un environnement de développement intégré (IDE) et un logiciel d'analyse de paquets du réseau installés (pour les tests de sécurité).

5. Jalons du projet

Effort : 1 (très peu d'effort) à 5 (beaucoup d'effort)

Jalon	Effort	Date de début	Date de fin
Définition des cas de tests	2	15-04-21	15-04-21
Implémentation des tests de fonction	3	16-04-21	17-04-21
Implémentation des tests d'interface usager	2	16-04-21	17-04-21
Implémentation des tests d'intégrité des données	3	16-04-21	17-04-21
Implémentation des tests de performance	3	16-04-21	17-04-21
Implémentation des tests de charge	2	16-04-21	17-04-21
Implémentation des tests de stress	3	16-04-21	17-04-21
Implémentation des tests de volume	2	16-04-21	17-04-21
Implémentation des tests de sécurité	3	16-04-21	17-04-21
Implémentation des tests d'échec/récupération	3	16-04-21	17-04-21
Exécution et récolte des résultats	2	16-04-21	17-04-21
Correction des bogues	5	17-04-21	19-04-21