
<Équipe 9>

<WALL-E2>
Plan de tests logiciels

Version 3.1

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

Historique des révisions

Date	Version	Description	Auteur
2017-04-06	1.0	Rédaction version initiale	Essaid Djoudi
2017-04-07	2.0	Correction et complétion de la partie : 2 (Exigences à tester)	Essaid Djoudi
2017-04-07	2.1	Correction et complétion des parties : 3,4,5	Essaid Djoudi
2017-04-08	3.0	Correction finale et mise en page	Essaid Djoudi
2017-04-08	3.1	Version pour le livrable final	Essaid Djoudi

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

Table des matières

1.	Introduction	4
2.	Exigences à tester	4
3.	Stratégie de test	11
3.1	Types de test	11
3.1.1	Tests de fonction	12
3.1.2	Tests d'interface usager	12
3.1.3	Tests d'intégrité des données	13
3.1.4	Tests de performance	13
3.1.5	Tests de charge	14
3.1.6	Tests de stress	14
3.1.7	Tests de volume	15
3.1.8	Tests de sécurité et de contrôle d'accès	15
3.1.9	Tests d'échec/récupération	15
3.1.10	Tests de configuration	16
3.1.11	Tests d'installation	17
3.2	Outils	17
4.	Ressources	17
4.1	Équipe de test	17
4.2	Système	17
5.	Jalons du projet	18

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

Plan de tests logiciels

1. Introduction

La qualité du développement du projet WALL-E2 nous impose à établir un plan de test.

Le plan de test nous permet d'identifier et de spécifier ce qu'on souhaite tester et comment effectuer ces tests. Ceci a pour but d'y trouver des anomalies ou des défauts liés à la qualité du projet, de s'assurer de la cohérence des données saisies (format des données, des contraintes sur les données, etc.) et de vérifier la cohérence des actions du système sur les données.

Alors dans ce plan de tests nous décrivons un inventaire d'éléments à tester dans l'application ainsi que les caractéristiques. Il s'agit de ce qui doit être testé. Ensuite nous procédons à une explication de l'approche à utiliser afin de créer les cas de tests pour chaque type de tests à effectuer tout en mentionnant les outils nécessaires à l'exécution des tests .

2. Exigences à tester

Dans cette section, nous identifions ce qui sera testé (cas d'utilisation, exigences fonctionnelles et exigences non fonctionnelles) séparé sous forme de deux grands tableaux :

- Un tableau qui identifiera les fonctionnalités à tester du côté du client lourd.
- Un tableau qui identifiera les fonctionnalités à tester du côté du client léger.

Alors le travail consiste à établir la traçabilité entre quelques exigences du SRS et leurs cas de test respectifs.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

2.1 Client lourd :

	Exigences à tester :	Tests à exécuter :
Clavardage :	SRS020 3.1.1.15.2 lorsqu'un contact d'un utilisateur connecté au serveur est en ligne, l'indication « en ligne » doit apparaître devant son nom	<ul style="list-style-type: none"> - ouvrir l'application client lourd. - Se connecter au serveur. - Créer un premier utilisateur - Ouvrir l'environnement du clavardage pour le premier utilisateur. - Tout en laissant le premier utilisateur connecté, créer un deuxième utilisateur. - Ouvrir son environnement du clavardage. - Vérifier à partir de l'environnement du clavardage du deuxième utilisateur la présence du mot 'En ligne' devant le premier utilisateur qui est toujours connecté. - Vérifier aussi à partir de l'environnement du clavardage du premier utilisateur la présence du mot 'En ligne' devant le deuxième utilisateur qui est toujours connecté.
	(SRS009 3.1.1.6)Il doit être possible pour un utilisateur d'envoyer des émoticônes lors d'une discussion.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un premier utilisateur - Ouvrir l'environnement du clavardage pour le premier utilisateur. - Tout en laissant le premier utilisateur connecté, créer un deuxième utilisateur. - Ouvrir son environnement du clavardage. - Envoyer une émoticône en message de l'environnement du clavardage du premier utilisateur vers celui du deuxième utilisateur. - Vérifier auprès du serveur la réception de l'émoticône en forme de string provenant de l'environnement du clavardage du premier utilisateur. - Vérifier auprès du serveur l'envoi de l'émoticône en forme de string vers de l'environnement du clavardage du deuxième utilisateur. - Vérifier la présence de l'émoticône dans la section des messages de l'environnement du clavardage du deuxième utilisateur.
	3.1.1.8 L'environnement de clavardage doit empêcher l'échange de mots inappropriés entre utilisateurs lors d'une discussion.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un premier utilisateur - Ouvrir l'environnement du clavardage pour le premier utilisateur. - Tout en laissant le premier utilisateur connecté, créer un deuxième utilisateur. - Ouvrir son environnement du clavardage. - Via la base de données, il faut accéder au

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

		<p>fichier contenant la liste des mots inappropriés.</p> <ul style="list-style-type: none"> - Choisir n'importe quel mot appartenant à la liste des mots inappropriés. - Envoyer le mot choisi en message de l'environnement du clavardage du premier utilisateur vers celui du deuxième utilisateur. - Vérifier auprès du serveur la réception de du mot provenant de l'environnement du clavardage du premier utilisateur. - Vérifier auprès du serveur l'annulation de l'envoi du mot vers l'environnement du clavardage du deuxième utilisateur. - Vérifier l'absence du mot dans la section des messages de l'environnement du clavardage du deuxième utilisateur.
Profil utilisateur :	SRS043 3.1.4.1 Il doit être possible pour un utilisateur de créer son profil à partir du client lourd.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un nouveau compte utilisateur - Vérifier l'affichage des alertes appropriées à la mauvaise saisie de données. - Vérifier l'acceptation de la connexion auprès du serveur. - Vérifier l'ajout adéquat des informations du profil créé au niveau de la base de données
	SRS046 3.1.4.4 sur le site Web, le profil de chaque utilisateur doit contenir les informations USER ainsi que les statistiques USER.	<ul style="list-style-type: none"> - Ouvrir le site Web - Vérifier la présence des comptes utilisateur déjà créés. - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un nouveau compte utilisateur - Rouvrir le site Web . - Vérifier l'ajout du nouveau compte utilisateur Créé.
Édition et simulation en ligne :	SRS026 3.1.2.2 il doit être possible pour un utilisateur d'assister à une édition sans apporter aucune modification.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un premier compte utilisateur ou se connecter avec un compte qui existe déjà et qui n'est pas connecté. - Entamer une édition en ligne sur une carte publique. - Créer un deuxième compte utilisateur ou se connecter avec un compte qui existe déjà et qui n'est pas connecté. - Rejoindre l'édition en ligne sur une carte publique à laquelle le premier utilisateur est en train de participer. - Essayer de faire des actions sur la scène (ajout objet, suppression objet, etc.). - Vérifier l'incapacité de faire des actions sur la scène (car c'est un spectateur).

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

	SRS026 3.1.2.2 il doit être possible pour un utilisateur d'assister à une édition sans apporter aucune modification.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un premier compte utilisateur ou se connecter avec un compte qui existe déjà et qui n'est pas connecté. - Entamer une simulation en ligne sur une carte publique. - Créer un deuxième compte utilisateur ou se connecter avec un compte qui existe déjà et qui n'est pas connecté. - Rejoindre la simulation en ligne sur une carte publique à laquelle le premier utilisateur est en train de participer. - Vérifier l'absence du robot affilié au deuxième utilisateur et l'incapacité de ce dernier de faire des actions sur la scène (car c'est un spectateur).
	SRS028 3.1.2.4 en mode édition ou simulation en ligne, si un utilisateur reste inactif pendant 3 minutes, il doit automatiquement être déconnecté du serveur.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un compte utilisateur ou se connecter avec un compte qui existe déjà et qui n'est pas connecté. - Rejoindre une simulation en ligne sur une carte publique. - Mettre le robot sur contrôle manuel - Ne plus faire d'action pendant 3min. - Vérifier après les 3minute de la déconnexion du serveur.
Confidentialité des cartes en ligne	SRS055 3.1.5.2 lors de la création d'une carte, par défaut elle doit être publique	<ul style="list-style-type: none"> - ouvrir l'application client lourd. - Se connecter au serveur. - Créer un compte utilisateur ou se connecter avec un compte qui existe déjà. - Rejoindre l'édition en ligne. - Créer une carte - Sauvegarder cette carte - Vérifier auprès de la base de données que le type de carte est bien public.
	SRS058 3.1.5.5 il doit être possible pour le créateur d'une carte d'alterner entre ces deux types : carte publique et carte privée.	<ul style="list-style-type: none"> - Ouvrir l'application client lourd. - Se connecter au serveur. - Créer un compte utilisateur ou se connecter avec un compte qui existe déjà. - Rejoindre l'édition en ligne. - Créer une carte - Sauvegarder cette carte - Vérifier auprès de la base de données que le type de carte est bien public. - Changer son type de public en privé. - Vérifier auprès de la base de données que le

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

		type de carte est devenu privé. - Recharger son type de privé en public. - Vérifier auprès de la base de données que le type de carte est redevenu public.
Tutoriel	SRS073 3.1.7.1 au niveau de l'édition, le client lourd doit fournir un tutoriel interactif. L'utilisateur choisit une opération d'édition et le tutoriel le guide afin qu'il puisse la réaliser.	L'exécution de ce test repose sur l'expérience utilisateur ; donc il faut faire recours à 4 utilisateurs en dehors du domaine du génie logiciel (échantillon aléatoire) pour essayer d'interagir avec ce tutoriel et voir si ce tutoriel les a aidés à comprendre le fonctionnement de notre application. À la fin du tutoriel, ces 4 utilisateurs feront face à l'utilisation directe de l'application, et à partir de l'adéquat de leurs interactions avec l'application que les défauts du tutoriel seront découverts.

2.2 Client léger :

	Exigences à tester :	Tests à exécuter :
Clavardage :	SRS122 3.2.4.15.2 il doit être possible pour un utilisateur de participer à une ou plusieurs discussions en même temps.	<ul style="list-style-type: none"> - Ouvrir l'application client léger. - Cliquer sur le bouton Clavardage - S'authentifier et entrer l'IP et le port du serveur. - Se connecter - Choisir un utilisateur dans la liste des utilisateurs connectés - Envoyer et recevoir des messages de ce dernier - Choisir un autre utilisateur dans la liste des utilisateurs connectés - Envoyer et recevoir des messages de ce nouvel utilisateur
Création de profils	SRS147 3.1.8.1.1 l'utilisateur doit avoir la possibilité de créer un profil utilisateur dans le client léger	<ul style="list-style-type: none"> - Ouvrir l'application client léger - Cliquer sur le bouton création profil - Entrer un identifiant non utilisé - Remplir les informations demandées pour la création d'un profil - Cliquer sur le bouton s'inscrire - Attendre l'alerte indiquant la création effective du profil

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

	SRS148 3.1.8.1.2 l'utilisateur doit recevoir un courriel de confirmation après avoir créée son profil	<ul style="list-style-type: none"> - Ouvrir l'application client léger - Cliquer sur le bouton création profil - Entrer un identifiant non utilisé - Remplir les informations demandées pour la création d'un profil, dont une adresse courriel valide - Cliquer sur le bouton s'inscrire - Attendre la réception du courriel dans la boîte de réception du courriel indiqué.
Authentification	SRS150 3.2.8.2.1 l'utilisateur du client léger doit pouvoir se connecter à son profil à partir de son identifiant et de son mot de passe	<ul style="list-style-type: none"> - Ouvrir l'application client léger - Cliquer sur clavardage. - S'authentifier - Entrer l'IP et le port du serveur. - Attendre le message de confirmation de la connexion (la connexion n'est possible que si le profil existe)
	SRS151 3.2.8.2.2 l'utilisateur du client léger doit pouvoir se connecter à son profil en utilisant la fonctionnalité Touch Id présente sur l'iPad	<ul style="list-style-type: none"> - Ouvrir l'application client léger - Cliquer sur clavardage. - S'authentifier en laissant le pouce appuyé (Matched Touch Id) sur le bouton home de l'iPad - Entrer l'IP et le port du serveur. - Cliquer sur le bouton se connecter - Vérifier que les champs de la connexion et du mot de passe ont bien été préremplis - Attendre le message de confirmation de la connexion (la connexion n'est possible que si le profil existe)
	.	
Présentation des cartes	SRS130 3.2.5.1 il est possible de visionner les différentes cartes sélectionnées	<ul style="list-style-type: none"> - ouvrir l'application client léger. - Cliquer sur le bouton Édition. - Cliquer sur le bouton nouveau carte - Ajouter des objets - Cliquer sur l'outil enregistrer - Nommer la carte - Retourner à l'interface de présentation de cartes - Vérifier que la carte en question fait partie de la liste des cartes enregistrées - Sélectionner et vérifier que la vignette et le nom correspondent à cette carte
	SRS132 3.2.5.2 les vignettes doivent être utilisées pour représenter les cartes enregistrées	<ul style="list-style-type: none"> - ouvrir l'application client léger. - Cliquer sur le bouton Édition. - Cliquer sur le bouton nouvelle carte - Enregistrer plusieurs cartes différentes avec des noms différents - Retourner à l'interface de présentation de

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

		<p>cartes</p> <ul style="list-style-type: none"> - Vérifier que les cartes en questions font partie de la liste des cartes enregistrées - Sélectionner à tour de rôle chacune des cartes pour vérifier que la vignette correspond à la carte sélectionnée
Tutoriel	SRS117 3.2.3.1 le tutoriel doit s'activer automatiquement lorsque l'utilisateur entre pour la première fois dans le mode éditeur	<ul style="list-style-type: none"> - Supprimer l'application de l'iPad - Réinstaller l'application sur l'iPad - Ouvrir l'application client léger - Cliquer sur l'Édition - Cliquer sur l'édition nouvelle Carte - Vérifier que l'interface active est celle du tutoriel
Interface utilisateur	SRS079 3.2.1.3 Il doit être possible pour un utilisateur d'utiliser trois gestuelles dans le client léger (la rotation, la pression longue, le pincement)	<ul style="list-style-type: none"> - Ouvrir l'application client léger. - Cliquer sur le bouton édition - Cliquer sur le bouton nouvelle carte - Ajouter un mur à la scène - Cliquer sur l'outil sélection - Sélectionner le mur en cliquant sur le mur - Cliquer sur l'outil rotation - Effectuer la gestuelle rotation - Vérifier que l'angle de l'objet est modifié - Cliquer sur l'outil mise à l'échelle - Effectuer la gestuelle pincement - Vérifier que la taille de l'objet a bien été modifiée - Ajouter plusieurs segments de ligne - Effectuer la gestuelle pression longue - Ajouter de nouveau un autre segment de ligne - Vérifier que ce segment ne s'ajoute pas automatiquement à la suite de l'avant-dernier segment ajouté
Personnalisation de l'application	SRS166 3.2.12.3 l'utilisateur doit avoir la possibilité de changer la présentation des cartes. À tout moment, il peut choisir de les présenter dans une liste ou dans un tableau. Cette modification est visible au moment où il doit choisir la carte qu'il souhaite éditer	<ul style="list-style-type: none"> - ouvrir l'application client léger. - Cliquer sur le bouton réglages - Cliquer sur le bouton grille de l'interface de réglages - Retourner au menu d'accueil en cliquant sur le bouton menu - Cliquer sur le bouton édition - Vérifier que les cartes enregistrées au préalable sont présentées sous la forme grille
Édition des objets de la scène	SRS086 3.2.2.6 une indication claire doit apparaître lorsque l'objet est sélectionné	<ul style="list-style-type: none"> - ouvrir l'application client léger. - Cliquer sur le bouton édition - Cliquer sur le bouton nouvelle carte - Ajouter un objet - Choisir l'outil sélection - Cliquer sur l'objet - Vérifier que l'objet a changé de couleur

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

	SRS106 3.2.2.26 Il est possible pour un utilisateur de modifier la position d'un objet	<ul style="list-style-type: none"> - ouvrir l'application client léger. - Cliquer sur le bouton édition - Cliquer sur le bouton nouvelle carte - Ajouter un objet - Choisir l'outil sélection - Cliquer sur l'objet - Vérifier que les propriétés de l'objet sont affichées dans la barre de propriétés - Cliquer sur le champ éditable de la position x de l'objet - Modifier la valeur du champ - Vérifier que l'objet s'est déplacé sur l'axe des x (horizontal) en fonction de la valeur entrée - Cliquer sur le champ éditable de la position y de l'objet - Modifier la valeur du champ - Vérifier que l'objet s'est déplacé sur l'axe des y (vertical) en fonction de la valeur entrée

3. Stratégie de test

La troisième section du document plan de projet représente la stratégie de test. C'est une approche que nous avons utilisée afin de créer les cas de tests pour chaque type de tests à effectuer.

Pour chaque cas de test, notre approche définit les éléments suivants :

- **Objectif de test** : ceci représente la vérification de l'adéquat de la cible de test (la navigation, la saisie de données, le traitement et la récupération) pour chaque cas d'utilisation.
- **Technique** : ceci représente l'exécution de chaque chemin du cas d'utilisation en utilisant des données valides (résultats attendus) et non valides (messages d'erreur et avertissement).
- **Critère de complétion** : ceci représente l'ensemble des tests prévus complétés ainsi que les anomalies identifiées et enregistrées.
- **Considérations particulières** : ceci représente les impacts sur l'implémentation et l'exécution d'un test fonctionnel.

3.1 Types de test

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

3.1.1 Tests de fonction

Objectif de test:	Permet de valider une solution proposée à la partie prenante lors du développement d'un produit logiciel. En autre terme ceci permet de vérifier si toutes les spécifications logicielles fournies par la partie prenante et les exigences du point de vue des cas d'utilisation sont respectées.
Technique:	À partir du SRS de l'équipe-9, chaque grand cas d'utilisation ainsi que l'ensemble des exigences sont validés isolément puis par la suite tous les cas d'utilisation sont validés ensemble. La technique de la boîte noire nous permet de réaliser ces tests. Elle consiste dans un premier temps à une vérification du produit logiciel et de ses processus internes et par la suite à une analyse des résultats.
Critère de complétion:	<ul style="list-style-type: none"> - La totalité des tests de fonctions prévus a été exécutée. - L'identification des bogues issus des tests a été établie - Des corrections ont été apportées à tout bogue identifié.
Considérations spéciales:	La vérification et la validation régulières des solutions proposées auprès du client (chargé de laboratoire du cours LOG3900), dans les heures de rencontre (séances de laboratoire) permettent d'assurer la bonne voie du développement du produit à livrer.

3.1.2 Tests d'interface usager

Objectif de test:	Fournit un moyen opérationnel pour répondre concrètement aux questions qui se posent lors de la conception de l'interface. En autre terme c'est une méthode permettant d'évaluer un produit en le faisant tester par des utilisateurs étrangers au projet. Ceci à pour ultime but d'observer les facilités, les difficultés ainsi que les hésitations à manipuler les différentes fonctionnalités par ces utilisateurs et de corriger les déficits afin d'enrichir l'application de convivialité.
Technique:	Tout en leur fournissant le matériel et temps suffisant, il faut déléguer l'utilisation de l'application à un ensemble de 4 utilisateurs différents qui étudie dans un autre programme que le génie logiciel et n'ayant jamais collaboré à ce projet (élément neutre). Ensuite il s'agit de les observer interagir avec l'application et enregistrer les véritables difficultés qu'ils rencontrent, les erreurs et points d'amélioration.
Critère de complétion:	L'évaluation de toutes les heuristiques de Bastian et Scapin.
Considérations spéciales:	<p>les tests doivent aider à démontrer les facteurs suivants, tirés de la norme ISO 25000 :</p> <ul style="list-style-type: none"> • Pertinence • Facilité d'apprentissage • Opérabilité • Protection de l'utilisateur face à une erreur • Interface utilisateur esthétique, • Accessibilité.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

3.1.3 Tests d'intégrité des données

Objectif de test:	Vérifier si les données contenues dans la base de données sont précises et fonctionnent comme prévu dans l'application. Aussi s'assurer des méthodes d'accès aux données et d'ajout, de modification et de suppression des données de la base de données.
Technique:	<ul style="list-style-type: none"> - Vérifier la connexion et la déconnexion à la base de données - Vérifier pour chaque valeur si elle est enregistrée avec succès à la base de données. - Vérifier la mise à jour des données en cas d'ajout, de modification et de suppression des données - Vérification si oui ou non une valeur vide ou par défaut peut être récupérée à partir de la base de données.
Critère de complétion:	Compatibilité des données avec leurs mises à jour (ce qui prouve à la fois l'accès adéquat à la base de données et l'application des modifications sur les données)
Considérations spéciales:	Vu la dissemblance il faut se servir d'un système de gestion de base de données orientée documents, répartissable sur un nombre quelconque d'ordinateurs pour appliquer la technique des tests d'intégrité de données.

3.1.4 Tests de performance

Objectif de test:	Mesurer la performance du système informatique par rapport aux contraintes et exigences identifiées au début du projet (temps de réponse maximum admissible décrivant une latence que le serveur prend afin de transmettre l'information a un nombre précis d'utilisateurs connectés à ce serveur).
Technique:	Lancer le serveur, connecter un utilisateur du client lourd/léger au serveur, calculer les délais moyens des envois des requêtes et des réponses suite à ces requêtes et comparer avec le délai maximal de réponse prévu.
Critère de complétion:	Tous les délais de traitement et de réponse des requêtes par le serveur sont inférieurs à 3 secondes.
Considérations spéciales:	S'assurer que les requêtes passées au serveur soient plus ou moins de même taille, sinon les diviser en sous-requêtes afin d'assurer la stabilité de la performance.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

3.1.5 Tests de charge

Objectif de test:	Permet de mettre en évidence les points sensibles et critiques de l'architecture technique du serveur et de calculer le dimensionnement du serveur ainsi que de la bande passante réseau nécessaire pour la mise en production de notre application.
Technique:	On simule l'utilisation progressive de l'application par la connexion d'un nombre d'utilisateurs au serveur jusqu'à arriver à <i>4 utilisateurs</i> parallèles et ainsi valider l'application pour une charge attendue d'utilisateurs.
Critère de complétion:	Tous les délais de traitement et de réponse des requêtes par le serveur lié à la présence d'un seul utilisateur doivent être approximativement les mêmes à la présence de 4 utilisateurs.
Considérations spéciales:	Vu que l'évaluation de ce projet se basera sur la présence de 4 utilisateurs , alors le système doit supporter minimum 4 utilisateurs connectés au serveur en même temps et qui interagissent simultanément avec le serveur.

3.1.6 Tests de stress

Objectif de test:	Permet d'identifier la manière avec laquelle le système réagit au maximum de l'activité attendue des utilisateurs.
Technique:	On simule l'activité maximale attendue de tous les scénarios fonctionnels confondus en heures de pointe de l'application. L'idée est de connecter au serveur, un ensemble de 10 utilisateurs différents où chacun d'eux enverra des requêtes en permanence pendant un laps de temps et au testeur d'analyser, après que ce temps s'écoule, si le serveur continue et résiste à la charge du traitement de toutes les requêtes.
Critère de complétion:	Le serveur reste toujours fonctionnel et ne plante pas en vue de la charge.
Considérations spéciales:	Le traitement de toutes requêtes est important dans le test de stress, mais l'augmentation des délais de traitement d'informations est facultative.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

3.1.7 Tests de volume

Objectif de test:	C'est un type spécifique de test de résistance au stress, qui permet à savoir si le système testé est capable de supporter une activité intense aussi il nous permet d'identifier la manière avec laquelle le système réagit aux limites du modèle d'usage de l'application.
Technique:	On simuler une activité bien supérieure à l'activité normale L'idée est de connecter au serveur, un ensemble de 10 utilisateurs différents et que chacun d'eux enverra des images simultanément et en permanence via l'environnement de clavardage. (Note : l'envoi d'une image par message est une requête considérée de taille importante à la vision du serveur)
Critère de complétion:	Les limites du serveur dépassent les différentes fonctionnalités sur lesquelles notre projet sera évalué.
Considérations spéciales:	Le traitement de toutes requêtes est important dans le test de stress, mais l'augmentation des délais de traitement d'informations est facultative si et seulement s'il y a plus de quatre utilisateurs.

3.1.8 Tests de sécurité et de contrôle d'accès

Objectif de test:	Permet d'identifier les failles sécuritaires de notre système. Aussi d'assurer la confidentialité et la protection des informations personnelles des utilisateurs.
Technique:	Ouvrir le client lourd/léger, lancer le serveur, essayer de connecter au serveur avec des données d'un profil (connexion et mot de passe) qui n'existe pas , vérifier l'affichage des messages d'alertes et d'avertissement ,ressaisir des informations, mais qui doivent existées (acceptées par le système), vérifier l'autorisation d'accès par le système.(cette technique peut s'élargir en faisant des vérifications sur les mots de passe liés aux cartes privées enregistrées dans le serveur)
Critère de complétion:	Suite à la saisie de mauvaises données, les alertes doivent être affichées. Aussi, la bonne saisie de données doit engendrer l'autorisation d'accès par le système.
Considérations spéciales:	La bonne saisie des données doit correspondre à des données qui existent déjà dans la base de données.

3.1.9 Tests d'échec/récupération

Objectif de test:	Permet d'illustrer le privilège qu'un utilisateur actif en mode simulation ayant perdu sa connexion au serveur (échec)peut rejoindre la simulation une fois la connexion rétablie(récupération).
Technique:	Ouvrir le client lourd, se connecter au serveur avec un compte utilisateur valide , passer en simulation , interrompre la connexion de cet utilisateur au serveur , rétablir la connexion et visualiser si l'utilisateur pourra continuer sa simulation selon les informations de la scène que le serveur à sauvegarder pour lui.
Critère de complétion:	L'utilisateur pourra rejoindre sa simulation une fois la connexion rétablie.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

Considérations spéciales:	Le serveur doit être capable de sauvegarder les informations de la scène en permanence.
---------------------------	---

3.1.10 Tests de configuration

Objectif de test:	<ul style="list-style-type: none"> - Configuration logicielle : Permet de vérifier le fonctionnement adéquat du site Web en correspondance avec un serveur Web lancé avec le logiciel NodeJS (la seule qui se trouve dans les ordinateurs du local-L4810) - Configurations matérielles: Permet de vérifier le fonctionnement adéquat du client lourd sur les ordinateurs et de vérifier le fonctionnement adéquat du client léger sur un iPad.
Technique:	<ul style="list-style-type: none"> - Configuration logicielle : Lancement du serveur Web via l'application NodeJS (la seule qui se trouve dans les ordinateurs du local-L4810) , vérifier sur un autre ordinateur et à partir d'un fureteur si le site Web est présent. - Configurations matérielles: Sauvegarder l'application du client lourd sur un ordinateur du local-L4810, vérifier si toutes les librairies nécessaires fonctionnent soit présentes et fonctionnel. Sauvegarder l'application du client léger sur un iPad emprunté de l'école Polytechnique de Montréal, vérifier si toutes les librairies nécessaires au fonctionnement sont présentes et fonctionnelles.
Critère de complétion:	<p>L'application du client lourd s'exécute parfaitement sur les ordinateurs du local-L4810.</p> <p>L'application du client léger s'exécute parfaitement sur l'iPad emprunté.</p> <p>Le site Web se lance parfaitement via les fureteurs d'un ordinateur en dehors de celui qui a lancé le serveur Web.</p>
Considérations spéciales:	<p>S'assurer que l'espace mémoire sur les ordinateurs du local-L4810 soit suffisant pour supporter l'application ainsi que les fichiers générés par le serveur.</p> <p>S'assurer que le site Web est supporté par plusieurs fureteurs différents.</p>

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

3.1.11 Tests d'installation

Ces tests ne sont pas pertinents, car pour notre projet aucune nouvelle installation, mise à jour ou installation complète/personnalisée n'est présente.

3.2 Outils

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

Type de test	Outil
Tests de charge	Moniteur de ressources Windows.

4. Ressources

4.1 Équipe de test

Rôle	Membre de l'équipe	Responsabilités
Test designer	1	C'est le concepteur du document plan de tests, alors il se charge de définir les cas de tests ainsi que la stratégie selon laquelle ces tests seront exécutés
Testeur	2	C'est le concepteur du document résultats des tests, alors en se basant sur le plan de tests et le code à tester le tester se charge l'exécution des tests et d'insérer leurs résultats dans le document résultats de tests,
Implémenter	3	C'est le correcteur des bogues visualisé lors des tests, alors à partir du document résultant des tests l'implémenter se charge de revoir le code non fonctionnel et trouver des solutions pour le rendre fonctionnel.

4.2 Système

L'exécution des tests se portera sur le réseau de l'école Polytechnique de Montréal.

À l'école Polytechnique de Montréal, le local L4810 est dédié pour les équipes de projet 3 (LOG3990). Dans ce local on retrouve des ordinateurs sur lesquelles le client lourd sera testé.

Pour tester le client lourd, ces ordinateurs sont accessibles 24h par jour et 7 jours par semaine et sont équipés d'un système d'exploitation Windows10 sous lequel on trouve l'environnement de développement Visual studio 22015.

En ce qui concerne le client léger, les tests seront exécutés sur un iPad (Ios 9.3.4) emprunté de l'école Polytechnique de Montréal.

<WALL-E2>	Version: 3.1
Plan de tests logiciels	Date: 2017-04-11

5. Jalons du projet

Jalon	Effort	Date de début	Date de fin
Plan de tests	15 heures	2017-04-06	2017-04-08
Résultats des tests	15 heures	2017-04-08	2017-04-10