Table des matières

[1 Cadre et objectif 2](#_Toc69714642)

[1.1 Aperçu des solutions à implémenter 2](#_Toc69714643)

[1.2 Porté de test 2](#_Toc69714644)

[1.3 Objectif de test 2](#_Toc69714645)

[1.3.1 Test de l’interface 2](#_Toc69714646)

[1.3.2 Test de fonctionnalité 2](#_Toc69714647)

[1.3.3 Test de performance 2](#_Toc69714648)

[2 Approche de test 3](#_Toc69714649)

[2.1 Tableau de stratégie 3](#_Toc69714650)

[3 Activité de test et délivrable 3](#_Toc69714651)

[3.1 Test de l’application 3](#_Toc69714652)

[3.1.1 Plan de test de validation 3](#_Toc69714653)

[3.1.2 Plan de test de fonctionnalité 4](#_Toc69714654)

[3.1.3 Plan de test de performance 4](#_Toc69714655)

[4 Approche de communication 4](#_Toc69714656)

[5](#_Toc69714657)

[5 Infrastructure, environnement et outils de test 5](#_Toc69714658)

[5.1 Infrastructure 5](#_Toc69714659)

[5.2 Environnement et outils de test 5](#_Toc69714660)

[6 Supposition, contrainte et dépendance 5](#_Toc69714661)

[7 Issue et risque 5](#_Toc69714662)

# Cadre et objectif

## Aperçu des solutions à implémenter

Le but final de ce plan de test est de mettre en place des scripts Katalon/Selenium (intégrés dans Jenkins) ainsi que des scripts OctoPerf (non intégré dans Jenkins) afin de tester notre application web.

## Porté de test

Les différent test vont portés sur toute notre applications web développée en Java Entreprise Edition. Cela inclue :

* la partie frontend qui contient toute les pages de notre application
* la partie backend qui contient l’infrastructure du site web
* la base de données et les interaction entre le backend et celle-ci
* la communication entre le frontend et le backend

## Objectif de test

L’objectif de ces test de de contrôler le bon fonctionnement de l’application. Les objectifs principaux de ces test sont :

### Test de l’interface

L’objectif du test de l’interface est de vérifier si toutes les interface s’affichent correctement. Ce test concerne donc toutes les pages de notre application web. Des test de validation vont être réalisés visuellement sur chaque fonctionnalité des pages du site web. Cette partie de test va principalement être intégrée au développement des pages elles-mêmes ainsi qu’à la fin des sprints principaux.

### Test de fonctionnalité

L’objectif du test de fonctionnalité consistent à tester le bon fonctionnement général de notre applications. Différents test unitaires comme l’inscription d’un utilisateur, la récupération des données d’une écurie, l’ajout d’un mécanicien dans une équipe, … seront effectués régulièrement grâce à Katalon.

### Test de performance

Les test de performance ont comme objectif de tester le temps de réponse du site web lors d’une forte sollicitation. Ces test seront réalisés grâce aux outils comme OctoPerf. L’objectif est d’avoir un temps de réponse raisonnable pour une vingtaine d’utilisateurs simultanés.

# Approche de test

## Tableau de stratégie

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caractéristique | Classe de risque | Review | Test développement | Test système | Acceptation utilisateur | Acceptation production |
| Fonctionnalité | | | | | | |
| Register | B | \*\* | \*\*\* | \*\*\* | \* | \*\*\* |
| Login | B | \*\* | \*\*\* | \*\*\* | \* | \*\*\* |
| Visualiser l’écurie | C | \* | \* | \* | \*\*\* | \*\* |
| Visualiser les utilisateurs | C | \* | \* | \* | \*\*\* | \*\* |
| Inviter un utilisateur | B | \* | \*\* | \*\* | \*\* | \* |
| Accepter une invitation | B | \* | \*\* | \*\* | \*\* | \* |
| Security | | | | | | |
| Accès aux pages | A | \*\* |  | \*\* |  | \*\*\* |

|  |  |
| --- | --- |
| Risque élevé | A |
| Risque moyen | B |
| Risque faible | C |

|  |
| --- |
| Priorité des test |
| Priorité élevé | \*\*\* |
| Priorité moyenne | \*\* |
| Priorité basse | \* |

# Activité de test et délivrable

## Test de l’application

|  |  |  |
| --- | --- | --- |
| Activités de test | Délivrable | Responsable |
| Test de validation | Plan de test de validation | Sébastien Roso |
| Test de fonctionnalité | Plan de test de fonctionnalité | Alan Chambaz |
| Test de performance | Plan de test de performance | Colin Pelletier |

### Plan de test de validation

Ci-dessous le plan de test de validation. Il comprend un test de l’utilisation, de l’ergonomie et fonctionnalité dans sa globalité.

|  |  |  |
| --- | --- | --- |
| **Objectif** | **Commentaire** | **Résultat** |
| Page de création de compte | Visualisation global de la page de création de compte |  |
| Page de login | Visualisation de la page de login |  |
| Page de l’écurie | Visualisation globale de la page de l’écurie |  |
| Budget de l’écurie | Visualisation correcte du budget de l’écurie |  |
| Formule 1 | Visualisation des formules 1 de l’écurie |  |
| Pièces de la formule 1 | Affichage correcte de toutes les pièces de la formules 1 |  |
| Amélioration des pièces de la formule 1 | Affichage correcte du niveau d’amélioration des pièces de la formule 1 |  |
| Usure des pièces de la formule 1 | Affichage correcte de l’usure des pièces de la formule 1 |  |
| Membre de l’écurie | Visualisation correcte des membres de l’écurie |  |
| Ensemble des utilisateurs | Visualisation de l’ensemble des utilisateurs qui ne sont pas encore dans une équipe |  |
| Invitation | Affichage correcte d’une invitation à rejoindre une écurie |  |

### Plan de test de fonctionnalité

|  |  |  |
| --- | --- | --- |
| **Objectif** | **Commentaire** | **Résultat** |
| Création d’un compte | Création d’un compte avec des informations correcte | ✓ |
| Création d’un compte | Création d’un compte avec une adresse email déjà utilisée | ✓ |
| Création d’un compte | Création d’un compte avec une adresse email incorrecte |  |
| Création d’un compte | Création d’un compte avec un mot de passe non conforme | ✓ |
| Authentification | Authentification avec le bon utilisateur et mot de passe | ✓ |
| Authentification | Authentification avec le bon utilisateur et mauvais mot de passe | ✓ |
| Authentification | Authentification avec un utilisateur inexistant | ✓ |
|  |  |  |

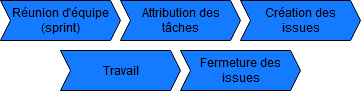
Le plan de test de fonctionnalité se fait via Katalon

### Plan de test de performance

Plan de test de performance se fait via OctoPerf

# Approche de communication

La communication lors des phases de test se fera principalement au travers des séances hebdomadaires comme toutes les autres tâches du projets.

Lors des séances, l’état actuel du projet est fait et les tâches du sprint suivant sont distribuées aux membres. Chacun des membres va ensuite créer des issues, travailler sur ses tâches puis fermer les issues si celles-ci sont terminées.

# 

Hors des séances, la communication se fait via un salon textuel Discord.

# Infrastructure, environnement et outils de test

## Infrastructure

Au vu de la situation sanitaire, aucun local n’est mis à disposition. Les séances se font à distance avec les logiciels Microsoft Teams et Discord.

## Environnement et outils de test

L’exécution des tests se fait principalement via deux outils

### Katalon

Katalon nous permet de réaliser des tests unitaires sur le site web en simulant des requêtes d’utilisateurs. La version Katalon utilisée est Katalon studio 7.9.1

### OctoPerf

OctoPerf nous permet de réaliser des tests de performances en simulant plusieurs utilisateurs en parallèle.

# Supposition, contrainte et dépendance

# Risques

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Description | Source | Catégorie | Probabilité | Impact | Criticité | Remédiation |
| Manque de compétence | Interne | Compétence | 0.8 | 0.5 | 0.4 | Relecture du cours, recherche de tutoriel sur internet, poser des questions à une personne externe en cas de blocage. |
| Mauvaise utilisation des outils de test | Interne | Compétence | 0.2 | 0.6 | 0.12 | Etude de la documentation des outils de test |
| Retard sur la livraison du projet | Interne | Temporel | 0.7 | 0.5 | 0.35 | Travailler en dehors des cours |
| Perte des protocoles de test | Externe | Donnée | 0.05 | 1 | 0.05 | Réaliser des copies des données (git, NAS, sauvegarde externe) |
| Arrêt de la maintenance des serveurs OctoPerf | Externe | Outils | 0.05 | 0.5 | 0.025 | Créer un docker OctoPerf, chercher un autre programme de test de performance |