# Concept et plan de tests

Projet Jamais404

Vallat Quentin, Friche Colin, Jurasz Loïc

INF3DLM-A – HE-ARC



## 1 Introduction

Le site web jamais404 est un projet qui permet de trouver des pages sur le site n’ayant pas encore été trouvée. On peut ensuite mettre un commentaire sur une page déjà trouvée et également voir les pages que nous avons trouvées.

Ce document décrit le plan pour les tests pour l’application Jamais404. Ce plan s’organise autour des objectifs suivants :

* Identifier les composants de l’application devant être testés
* Expliquer la stratégie de test employée
* Identifier les ressources nécessaires
* Lister les spécifications de tests recommandés

## 2. Périmètre de test

1. Interface utilisateur
   1. Page de login
   2. Page de logout
   3. Page d’accueil
   4. Page de profil
   5. Page recherchée
   6. Espace commentaire
2. Communication avec la base de données
   1. Lecture des données
      1. Affichage des pages trouvées par les utilisateurs
      2. Visualisation des commentaires
      3. Visualisation des pages trouvées par l’utilisateur
   2. Ecriture des données
      1. Page trouvée par l’utilisateur
      2. Commentaire ajouté
3. Test de performance
4. Test d’intégration

### 2.1 Spécification de tests

Tests d’intégrité de la base de données

Tests de système

Tests de l’interface utilisateur

Tests de performance

Tests de charge

Tests de sécurité d’accès

Tests de basculement et de récupération

### 2.2 Stratégie et approche détaillées de test

### Tests d’intégrité de la base de données

#### Objectif

L’objectif est de s’assurer que les méthodes d’accès à la base de données fonctionnent correctement et ne puisse pas corrompre les données.

Technique de test

Peupler la base de données et tester les méthodes d’accès afin de s’assurer qu’elle fonctionne correctement.

#### Critère de complétion

Le test sera complété quand toutes les méthodes fonctionnent comme spécifié et sans corruption des données.

### Tests de système

Les méthodes de ce test seront automatisées avec Katalon

#### Objectif

Vérifier que le déploiement soit fait correctement

#### Technique de test

Déploiement manuel

#### Critère de complétion

L’application est accessible depuis les différents navigateurs internet.

### Tests de l’interface utilisateur

#### Objectif

Vérifier la navigation des pages

#### Technique de test

Tests automatisés sur Katalon

#### Critère de complétion

Obtention des résultats attendus

### Tests de performance

Ces tests seront réalisés grâce à Octoperf

#### Objectif

Vérifier les temps d’accès et les temps de réponses de l’application

#### Technique de test

Réalisation de scripts de test

#### Critère de complétion

Complétion des tests sans erreur des fonctionnalités.

### Tests de charge

Octoperf sera utilisé pour simuler plusieurs connexions et actions simultanées

#### Objectif

S’assurer de la résistance du système à plusieurs connexions simultanées ainsi qu’à plusieurs actions simultanées faite par plusieurs utilisateurs.

#### Technique de test

Simuler plusieurs connexions et actions simultanées.

#### Critère de complétion

Complétion des tests sans erreur des fonctionnalités

### Tests de sécurité d’accès

#### Objectif

Vérifier l’inscription et le login

#### Technique de test

Tests manuels

#### Critère de complétion

Vérifier qu’on ne puisse pas accéder à l’interface utilisateur sans être login et vérifier

### Tests de basculement et de récupération

#### Objectif

Vérifier la disponibilité de l’application

#### Technique de test

Tenter d’accéder à l’application, redémarrer le serveur et tester l’accès.

#### Critère de complétion

Vérifier que l’accès à l’application fonctionne.

## 3. Infrastructure de test

|  |  |
| --- | --- |
| **Tests** | **Outils** |
| Tests de performance | Octoperf |
| Tests d’interface utilisateur | Katalon |
| Qualité du code | Sonar Cloud |
| Gestion de projet | Git |
| Unit Tests | JUnit |

## 4. Gestion des risques

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **N°** | **Description** | **Source** | **Impact** | **Proba-**  **bilité** | **Gravité** | **Mitigation** |
| 1 | Difficulté de développement de l’application | Interne | 0.8 | 0.2 | 0.16 | Documentation sur les outils de développement existants |
| 2 | Mauvaise communication dans l’équipe | Interne | 0.7 | 0.2 | 0.14 | Séance de discussion (rencontre ou vocal) et respect de l’organisation mise en place |
| 3 | Difficulté de l’utilisation des outils | Interne | 0.6 | 0.4 | 0.24 | Documentation et lecture du cours |
| 4 | Non-respect des délais | Interne | 0.8 | 0.2 | 0.16 | Travail ponctuel et planning |