Haute Ecole Arc - Ingénierie

Rapport  
SampleCenter

Rapport Qualité du logiciel 2017 – Test Plan de SampleCenter

Professeur : Ouerhani Nabil

# Table des matières

[1. Table des matières 4](#_Toc480977295)

[2. Contexte 5](#_Toc480977296)

[3. Plan de test 5](#_Toc480977297)

[3.1. Qualité 5](#_Toc480977298)

[3.2. Objectifs 5](#_Toc480977299)

[3.3. Périmètre 5](#_Toc480977300)

[3.4. Stratégie & Approche 6](#_Toc480977301)

[3.1. Procédure de test 6](#_Toc480977302)

[3.1.1. L’Equipe 6](#_Toc480977303)

[3.1.2. Les étapes principales 6](#_Toc480977304)

[3.1.1. Les environnements et les outils de test 7](#_Toc480977305)

[3.1.1. Plan de communication avec le projet 7](#_Toc480977306)

[3.2. Estimation de l’effort (budget) 7](#_Toc480977307)

[4. Annexes 7](#_Toc480977308)

[4.1. Dépôt du code 7](#_Toc480977309)

[5. Références 8](#_Toc480977310)

# Contexte

Ce document regroupe la stratégie de test et le plan de test pour le projet de gestion de sample « SampleCenter ».

Le but de la plateforme SampleCenter est de fournir une interface de gestions de samples. Les musiciens peuvent utiliser la plateforme pour trouver facilement le sample dont ils ont besoin et le télécharger. Ils pourraient aussi partager leurs propres samples et regrouper les samples qu'ils ont aimés. Une autre utilisation de la plateforme serait en tant qu'API, c'est-à-dire depuis une autre application (web ou autre).

# Plan de test

## Qualité

Les attentes du projet en termes de qualité du produit sont les suivantes :

* Le produit doit pouvoir supporter une grande charge sur la même fonctionnalité, en garantissant un délai de réponse correct.
* Le produit doit pouvoir tourner sur différents navigateurs à définir.
* Le produit doit garantir une expérience utilisateur efficace et intuitive.
* Les fonctionnalités qui sont disponibles sur le produit final remplissent pleinement leur rôle.

## Objectifs

Les objectifs principaux de test sont :

* Garantir un produit fiable et intuitif pour l’utilisateur, quelle que soit son degré de compétence en informatique, et quelle que soit son niveau d’interaction avec le produit.
* Repérer rapidement ou sont les erreurs, les problèmes, afin de pouvoir corriger ou améliorer le produit dans un délai réduit et ainsi optimiser le processus d’élaboration du produit.

## Périmètre

Voici ci-dessous le périmètre des tests, avec un grand niveau d’abstraction.

Il s’agira principalement de tests de validation sur les aspects suivants :

* Fonctionnalité : En particulier la conformité vis-à-vis des spécifications.
* Portabilité : Produit fonctionnel sur différents navigateurs
* Ergonomie : Produit intuitif/facile d’utilisation

Dans une moindre mesure, le périmètre portera également sur des tests système au niveau de la performance.

Il y aura aussi éventuellement quelques tests unitaires sur des classes/fonctions complexes pour garantir la fiabilité de celles-ci.

|  |  |
| --- | --- |
| Niveau de test | Type de test |
| Validation | Fonctionnalité, Portabilité, Ergonomie |
| Système | Performance |
| Unitaire | Fiabilité/Maturité |

## Stratégie & Approche

Afin de parvenir au principal objectif du projet, qui consiste à offrir un service de « livre de cuisine » en ligne pratique et accessible pour tout un chacun, nous allons mettre en place la stratégie de tests suivante :

Les tests seront avant tout focalisés sur la facilité d’utilisation, sur l’offre multiplateforme (le produit fonctionne sur différents navigateurs) et sur la garantie de fonctionnalités fiables et répondant aux attentes.

Dans ce même but, des tests de performance sur la charge seront inclus. En effet un grand nombre d’utilisateur doit pourvoir accéder à notre produit simultanément tout en bénéficiant d’un temps de réponse convenable.

Ainsi donc, pour chaque fonctionnalité, on contrôlera que les critères de charge sont respectés, que la portabilité est garantie et qu’elle remplit son rôle correctement.

## Procédure de test

### L’Equipe

L’équipe de test se compose de :

* Ceschin Loris
* Perez Joakim
* Gonin Nicolas Rapport

### Les étapes principales

* Prise en main des environnements et outils de tests,
* Implémentation des tests selon les directives,
* Exécution des tests,
* Emission d’un rapport de test, contenant le code ou la procédure du test, la version, ainsi que les résultats,
* Si les résultats escomptés ne sont pas obtenus, corrections/améliorations dans le projet et/ou dans les tests, puis réitération des point 3) et 4) jusqu’à obtenir des résultats escomptés,
* Lorsque tous les tests ont passés avec succès, clôture des tests et émission d’un rapport général.

### Les environnements et les outils de test

* Pour les tests unitaires, utilisation de la librairie JUnit.
* Pour les tests de performance, utilisation de la librairie JMeter,
* Pour les tests d’utilisation automatisés, utilisation de la librairie Selenium et de WebDriver.

### Plan de communication avec le projet

* A l’issue d’un test et selon les résultats du rapport, le Test Manager communique les erreurs/problèmes aux autres membres du projet et s’assure que l’équipe soit sur la même longueur d’onde pour trouver des solutions.
* Lorsqu’une solution a été mise en place, le Test Manager est averti et une nouvelle phase de test peut commencer.

## Estimation de l’effort (budget)

Nous utilisons ici la méthode d’estimation SDP pour l’effort à fournir.

L’unité utilisée est le Jours par Homme (Person per Days). Cela équivaut à une personne travaillant une journée entière (8H).

# Annexes

## Dépôt du code

* [GitHub](https://github.com/ceschinloris/JEE-SampleCenter)

# Références