Haute Ecole Arc - Ingénierie  
Projet SampleCenter  
Concept et plan de tests

Loris Ceschin – Nicolas Gonin – Joaquim Perez  
27.04.2017

Qualité du logiciel 2017  
Professeur : Ouerhani Nabil

# Table des matières

[1. Table des matières 3](#_Toc481130998)

[2. Contexte 4](#_Toc481130999)

[3. Stratégie de test 4](#_Toc481131000)

[3.1. Qualité 4](#_Toc481131001)

[3.2. Objectifs 4](#_Toc481131002)

[3.3. Périmètre 4](#_Toc481131003)

[3.4. Stratégie d’approche 5](#_Toc481131004)

[4. Plan de test 5](#_Toc481131005)

[4.1. Fonctionnalités 5](#_Toc481131006)

[4.2. Ergonomie 5](#_Toc481131007)

[4.3. Critères d’accep tance 5](#_Toc481131008)

[5. Procédure de test 6](#_Toc481131009)

[5.1. L’Equipe 6](#_Toc481131010)

[5.2. Environnement et outils de test 6](#_Toc481131011)

[5.3. Processus de test 6](#_Toc481131012)

[5.4. Environnements et Outils de test 6](#_Toc481131013)

# 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).

# Stratégie de test

## Qualité

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

* La navigation doit être intuitive
* L’ajout d’un sample doit être simple
* Les utilisateurs peuvent uploader des samples qui ont le même nom.
* Plusieurs utilisateurs peuvent lire le même sample au même moment
* 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 les navigateurs principaux.

## Objectifs

Les objectifs principaux de test sont :

* Garantir un produit fiable et intuitif pour l’utilisateur
* 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.
* Valider les fonctionnalités les plus importantes

## Périmètre

Voici ci-dessous le périmètre des tests, avec un grand niveau d’abstraction. Pour plus de détails, voir le chapitre “Périmètre de test”.

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

* Fonctionnalité : Lecture/upload de samples
* Portabilité : Produit fonctionnel sur différents navigateurs
* Ergonomie : Navigabilité entre les pages

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 d’approche

Pour garantir un produit fonctionnel et accessible, nous allons utiliser la stratégie suivante :

Les tests seront focalisés sur les fonctionnalités les plus importantes (upload / lecture de samples) et sur la navigabilité du site.

Pour garantir une utilisation du site agréable pour tous les utilisateurs, des tests de charge seront inclus. En effet, plusieurs utilisateurs doivent pouvoir accéder aux mêmes ressources dans un temps de réponse convenable.

# Plan de test

## Fonctionnalités

* Upload de samples : On doit pouvoir upload un sample sans problème tant que les champs sont corrects (pas de conflits de nom, url, ect)
* Lecture de samples : Un son uploadé doit pouvoir être lu
* Recherche d’un sample : La recherche doit retourner tous les samples dont le titre contient l’entrée.
* Si une recherche ne retourne aucun résultat, afficher une page adaptée

## Ergonomie

* Navigabilité : On doit pouvoir upload un sample depuis n'importe quelle page
* UI : Le site doit être facile de compréhension

## Critères de validation

|  |  |  |
| --- | --- | --- |
| **Quoi** | **Besoins** | **Validation** |
| Upload | Upload un sample sur le serveur | Le sample doit être uploadé si tous les champs requis sont remplis |
| Lecture | Lire un sample | Lire le fichier audio uploadé sur le serveur |
| Recherche | Pouvoir rechercher un sample | La recherche sort une liste de tous les samples ayant tout ou une partie de la valeur recherchée dans le titre |
| Navigabilité | Pouvoir upload un sample depuis partout | Toutes les pages ont un lien sur l’upload |
| UI | Facile d’utilisation | Tous les champs ont un texte qui spécifie quoi entrer. |

# Procédure de test

## L’Equipe

L’équipe de test se compose de :

* Joaquim Perez
* Nicolas Gonin
* Loris Ceschin

## Environnement et outils de test

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

## Processus de test

1. Prise en main des environnements et outils de tests
2. Implémentation des tests selon les directives.
3. Exécution des tests.
4. 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.
5. Lorsque tous les tests ont passés avec succès, clôture des tests et émission d’un rapport général.

## Environnements et 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, ainsi que du module Firefox Selenium IDE pour enregistrer les scripts.