|  |
| --- |
|  |
| ConnectTo |
| Plan de tests |
|  |
| **Dylan Santos de Pinho, Paul Jeanbourquin, Cédric Pahud** |
| **Date (24.04.2018)** |

|  |
| --- |
|  |

Table des matières

[1. Introduction 3](#_Toc512458469)

[1.1 Description du projet 3](#_Toc512458470)

[1.2 Buts du plan de tests 3](#_Toc512458471)

[1.3 Portée du plan de tests 3](#_Toc512458472)

[1.4 Références 3](#_Toc512458473)

[2. Stratégie de tests 4](#_Toc512458474)

[3. Objectifs de qualité 4](#_Toc512458475)

[4. Approche de tests 5](#_Toc512458476)

[4.1 Authentification 5](#_Toc512458477)

[4.1.1 Fonctionnel 5](#_Toc512458478)

[4.1.1 Sécurité 5](#_Toc512458479)

[4.2 Recherche 6](#_Toc512458480)

[4.2.1 Fonctionnel 6](#_Toc512458481)

[4.3 CRUD 6](#_Toc512458482)

[4.3.1 Fonctionnel 6](#_Toc512458483)

[4.4 API 6](#_Toc512458484)

[4.4.1 Fonctionnel 6](#_Toc512458485)

[4.4.2 Performance 7](#_Toc512458486)

[4.5.1 Outils 7](#_Toc512458487)

[5. Livrables 7](#_Toc512458488)

# 1. Introduction

## 1.1 Description du projet

Le but de ce projet est d’avoir un site web permettant de connecter des services web entre eux en mettant en place un système de signal et d’actions. Un signal représenté par une url déclenche l’action d’un service enregistré sur notre site (par exemple le déclanchement d’une alarme). Tout ceci en utilisant Java Entreprise Edition.

## 1.2 Buts du plan de tests

Valider la fonctionnalité basique du site c’est-à-dire la connexion entre deux services. Mais aussi les fonctionnalités des CRUD des services/actions/signaux et pour finir l’authentification d’utilisateurs avec gestion des droits.

## 1.3 Portée du plan de tests

Nous allons focaliser les tests sur les sections critiques de notre application qui sont les suivantes :

* **Authentification**
  + Admin : tous les droits
  + Standard : gérer les connections
  + Service : gérer actions, signaux, services et connections
* **Recherche**
  + Simple
  + Avancée
* **Les différents CRUD**
  + Services
  + Actions
  + Connections
  + Signaux
  + Utilisateurs
* **L’API**
  + Connection entre les signaux

## 1.4 Références

Ref-1 : Cahier des charges ConnectTo Janvier 2018

# 2. Stratégie de tests

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caractéristique | Risque | Vérification | Test Dev | Test Sys | User  Acceptance | Acceptation  App |
| Fonctionnalité |  |  |  |  |  |  |
| Authentification | A | ●● | ●● |  | ●● |  |
| CRUD | B | ● | ●● | ● |  |  |
| Recherche | C |  | ● | ● | ● |  |
| API | A |  | ●●● | ●●● | ●● | ● |
| Affichage |  |  |  |  |  |  |
| Ergonomie | C | ● |  |  | ●● |  |
| Performance |  |  |  |  |  |  |
| User | B |  |  |  | ● |  |
| Service | A |  | ●● | ● |  |  |
| Sécurité | A | ● | ●● | ● |  | ● |

A : très risqué ● : rigueur limité S : test statique  
B : risqué ●● : rigueur moyenne I : test implicite  
C : peu risqué ●●● : rigueur haute

# 3. Objectifs de qualité

Notre application doit atteindre les objectifs de qualité suivant :

* **Sécurité** : Les utilisateurs ne peuvent atteindre que les pages sur lesquels ils ont des droits
* **Performance** : L’application doit pouvoir tenir une grande charge d’utilisateurs et une grande charge de services
* **Compatibilité :** L’application doit être fonctionnelle sur tous les navigateurs actuels
* **Utilisabilité :** L’application doit être facile de prise en main pour un utilisateur lambda

# 4. Approche de tests

## 4.1 Authentification

### 4.1.1 Fonctionnel

|  |  |
| --- | --- |
| Objectif: | Tester qu’un utilisateur puisse se connecter |
| Technique: | Entrer login et mot de passe sur la page d’accueil du site |
| Critère de succès: | La page de bienvenue s’affiche |
| Considérations particulières: | Les utilisateurs ne peuvent pas s’inscrire seul les administrateurs peuvent créer des utilisateurs il faut donc taper un login et mot de passe fourni par l’admin |

### 4.1.1 Sécurité

|  |  |
| --- | --- |
| Objectif: | Tester que les utilisateurs n’ait que les accès défini par leur rôle |
| Technique: | Se connecter en tant qu’utilisateur de chaque rôle et tester les accès aux différentes pages |
| Critère de succès: | Les utilisateurs n’ont accès qu’aux pages définies par leurs rôles |
| Considérations particulières: | Le rôle admin a accès à tout  Le rôle service à accès à actions, signaux, services et connections  Le rôle standard à seulement accès aux connections |

## 

## 4.2 Recherche

### 4.2.1 Fonctionnel

|  |  |
| --- | --- |
| Objectif: | Tester que la recherche retournes des résultats adéquats |
| Technique: | Rechercher un service via le lien de recherche |
| Critère de succès: | Une liste de services correspondant à la recherche s’affiche |

## 4.3 CRUD

### 4.3.1 Fonctionnel

|  |  |
| --- | --- |
| Objectif: | Tester les fonctionnalités du CRUD pour chaque entité |
| Technique: | Ajouter, Voir, Modifier et Supprimer chaque entité |
| Critère de succès: | Les éléments sont bien mis à jour dans la base de données |
| Considérations particulières: | Les entités à tester sont : Services, Signaux, Utilisateurs, Actions et Connections |

## 4.4 API

### 4.4.1 Fonctionnel

|  |  |
| --- | --- |
| Objectif: | Tester que deux services connectés puisse communiquer avec le système de signal/action |
| Technique: | Envoyer un signal en appelant une url |
| Critère de succès: | L’action connectée doit se déclencher |
| Considérations particulières: | Dans notre cas le service est représenté par un serveur python qui déclenche une alarme en recevant le signal ALARM |

### 4.4.2 Performance

|  |  |
| --- | --- |
| Objectif: | Tester que plusieurs services puisse communiquer en même temps dans ralentissement |
| Technique: | Simuler des services avec JMeter |
| Critère de succès: | Le temps de réponse ne doit pas être ralenti |

### 4.5.1 Outils

Pour effectuer ces tests nous avons utilisé les deux outils suivants :

* JMETER : pour les tests de charge
* Selenium pour les tests fonctionnels

# 5. Livrables

* Script de tests Selenium
  + Login
  + Sécurité
  + Recherche
* Plan de test JMeter
* Graphiques tests de charge
  + Users
  + Services
* Rapport de tests Selenium