|  |
| --- |
|  |
| ConnectTo |
| Plan de tests |
|  |
| **Dylan De Pignata, Paul Amy Jambon, Cédric Pahud** |
| **Date (24.04.2018)** |

|  |
| --- |
|  |

Table des matières

[1. Introduction 3](#_Toc512364152)

[1.1 Description du projet 3](#_Toc512364153)

[1.2 Buts du plan de tests 3](#_Toc512364154)

[1.3 Portée du plan de tests 3](#_Toc512364155)

[1.4 Références 3](#_Toc512364156)

[2. Inventaire des tests 4](#_Toc512364157)

[3. Stratégie de tests 4](#_Toc512364158)

[3.1 <Type ou portion 1> 4](#_Toc512364159)

[3.1.1 Intégrité de données 4](#_Toc512364160)

[3.1.2 Fonctionnel 4](#_Toc512364161)

[3.1.3 Performance 5](#_Toc512364162)

[3.1.4 Sécurité 5](#_Toc512364163)

[3.1.5 Interface utilisateur 5](#_Toc512364164)

[3.1.6 Interface matérielle 6](#_Toc512364165)

[3.1.7 Outils 6](#_Toc512364166)

[4. Cas de tests 6](#_Toc512364167)

[4.1 <Cas 1> 6](#_Toc512364168)

[4.1.1 Objet du test 6](#_Toc512364169)

[4.1.3 Scénario de test 6](#_Toc512364170)

[4.1.3 Points de contrôle 7](#_Toc512364171)

# 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

La portée du plan ne concerne que notre application car elle ne doit pas être intégrée dans une quelconque architecture. Il est cependant important de noter que notre application est connectée à une base de données et que tout ce qui concerne la base de données et la connexion avec celle-ci ne sera pas testé. Tout ce qui concerne le fonctionnement des services en interne (c’est-à-dire tout sauf la connexion entre eux) ne sera pas testé non plus.

## 1.4 Références

Ref-1 : Cahier des charges ConnectTo Janvier 2018

# 2. Inventaire des tests

* **Test d’authentification** : tester qu’un utilisateur puisse s’authentifier sur le site web et qu’il ait accès seulement aux pages disponibles pour son rôle.
* **Test d’ajout, modification et suppression d’un service** : tester que les fonctionnalités de base du CRUD des services marche
* **Test d’ajout, modification et suppression d’un signal** : tester que les fonctionnalités de base du CRUD des signaux marche
* **Test d’ajout, modification et suppression d’une action**: tester que les fonctionnalités de base du CRUD des actions marche
* **Test d’ajout, modification et suppression d’un utilisateur**: tester que les fonctionnalités de base du CRUD des utilisateurs marche
* **Test de connexion des services** : tester que deux services connectés puisse communiquer entre eux via un signal et déclencher une action
* **Test de recherche**: tester que la fonction de recherche marche et retourne bien les résultats demandé.
* **Test de charge utilisateurs** : tester que plusieurs utilisateurs puissent se logger en même temps et interagir avec le site sans problème
* **Test de charge services :** tester que plusieurs service connectés en même temps puissent s’envoyer des signaux sans perturbation.

# 3. Stratégie de tests

[Expliquez l’approche à utiliser afin de créer les cas de tests pour chaque type de tests à effectuer ou portion de votre application, si votre application est divisée en différentes couches.]

## 3.1 <Type ou portion 1>

### 3.1.1 Intégrité de données

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment et avec quoi doit-on tester? (ex : Fichiers de départ, configuration initiale, etc.)] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test] |

### 3.1.2 Fonctionnel

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment doit-on tester?] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test] |

### 3.1.3 Performance

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment doit-on tester?] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test] |

### 3.1.4 Sécurité

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment doit-on tester?] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test] |

### 

### 3.1.5 Interface utilisateur

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment doit-on tester?] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test.  Considérer que les tests doivent aider à démontrer les facteurs suivants, tirés de la norme ISO 25000 :   * Pertinence * Facilité d’apprentissage * Opérabilité * Protection de l’utilisateur face à une erreur * Interface utilisateur esthétique, * Accessibilité.] |

### 3.1.6 Interface matérielle

|  |  |
| --- | --- |
| Objectif: | [Que doit-on tester?] |
| Technique: | [Comment doit-on tester?] |
| Critère de succès: | [Comment déterminer que le logiciel passe le test?] |
| Considérations particulières: | [Notes spécifiques à ce type de test.  Considérer que les tests doivent concerner autant l’intégration avec les autres logiciels embarqués sur l’appareil client que sur l’intégration avec ces composantes physiques.] |

### 3.1.7 Outils

[Liste des outils à utiliser pour les tests. Dans le cadre de certain projet, il pourrait être nécessaire de devoir faire une ou plusieurs des activités suivantes :

* Enregistrement des tests exécutés
* Établissement d’un calendrier de test
* Élaboration d’une matrice de traçabilité entre les différents tests et les requis du logiciel.

Identifiez et numérotez les outils en utilisant le préfixe OUT.]

# 4. Cas de tests

## 4.1 <Cas 1>

### 4.1.1 Objet du test

[Indiquez à quels tests de l’inventaire (section 2) ce cas de test fait référence. Il peut y en avoir plus d’un. De plus, expliquez ce qu’on veut contrôler par ce test.]

### 4.1.3 Scénario de test

[Étapes précises à appliquer par le testeur, avec les données de tests. Numérotez chacune des étapes à effectuer par le testeur.

Les points de contrôle indiquent les endroits où vous devez comparer les résultats obtenus par vos tests et les valeurs attendues, représentant une exécution sans problèmes.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Étape** | **Point de contrôle?** [Oui / Non] | **Description** | **Valeur à saisir** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### 4.1.3 Points de contrôle

[Inscrivez les valeurs obtenues aux différents points de contrôles.]

|  |  |  |
| --- | --- | --- |
| **Point de contrôle** | **Valeur observée** | **Valeur attendue** |
|  |  |  |
|  |  |  |
|  |  |  |