**Évaluation 1 – Stratégie de tests**

[Sous-titre du document]

Yolaine Courteau

**Résumé**

[Tapez le résumé du document ici. Il s'agit généralement d'une courte synthèse du document.]

Table des matières

Objectifs 3

Approche 3

Activités principales 3

Niveau de criticité des fonctionnalités 3

Niveau d’intégrité des fonctionnalités à tester : 3

Techniques et méthodes utilisées 4

Liste de tests en fonction des risques : 4

Liste des tests en fonction des niveaux 5

Critères d’acceptation 6

Analyse des risques 8

Liste de tests en fonction des risques : 8

Niveau d’intégrité 9

Critères d’entrée et sortie 9

Environnement 9

Environnement 9

Outils de test 11

Gestion de configuration 11

Analyse des risques 12

Revues et approbations 14

# Objectifs

S’assurer de la conformité des fonctions du logiciel, choisi par l’utilisateur et développé dans le processus d’itération du cycle de développement, pour s’assurer de la qualité du logiciel avant la mise en production d’une nouvelle itération ou de la fin du processus d’implémentation, jusqu’aux tests de recette.

S’assurer de la fonctionnalité et de l’utilisabilité du logiciel ainsi que l’exactitude des données

Le but de ce document est de décrire la stratégie de test de haut niveau pour le projet Denrée O Suivant, de définir la portée du test préliminaire, les activités de test de haut niveau et l'organisation, ainsi que la gestion des tests pour le projet. La stratégie de test fournit le cadre pour s’assurer de la fonctionnalité et de l’utilisabilité des 3 fonctionnalités suivantes:

-**Ajout de marchandises à donner**

**-Liste des marchandises à collecter**

**-Réservation de marchandises à collecter**

Le but de cette stratégie de test, est de clarifier les activités de test, les rôles et les responsabilités, les processus et la pratique à utiliser dans ce projet.

Dans les cas où des activités ne sont pas conformes à ce qui est couvert par la présente stratégie de test, les exceptions seront détaillées dans le plan de test.

# Approche

## Activités principales

### Niveau de criticité des fonctionnalités

Fonctionnalités à tester selon le niveau de criticité.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fonctionnalités à tester / Niveau de criticité | 1 | 2 | 3 | 4 |
| Ajout de marchandises à donner. |  |  |  | X |
| Liste des marchandises à collecter. |  |  |  | X |
| Réservation de marchandises à collecter. |  |  |  | X |

### Niveau d’intégrité des fonctionnalités à tester :



## Techniques et méthodes utilisées

### Liste de tests en fonction des risques :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tests/ Risques | 1 | 2 | 3 | 4 |
| Ad hoc | X |  |  |  |
| Analyse de traçabilité |  |  |  | X |
| Revue de code | X |  |  |  |
| Revues des exigences |  |  |  | X |
| Test black box |  |  | X |  |
| Test d’acceptation |  |  | X |  |
| Test de données |  |  |  | X |
| Test de valeurs limites |  |  |  | X |
| Test des composantes |  |  | X |  |
| Test exploratoires |  |  | X |  |
| Utilisabilité |  |  | X |  |

### Liste des tests en fonction des niveaux

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Unitaires | Intégration | Système | Acceptation |
| Ad hoc | X |  |  |  |
| Analyse de traçabilité | X | X | X | X |
| Revue de code | X |  |  |  |
| Revues des exigences | X | X | X |  |
| Test black box | X | X |  |  |
| Test d’acceptation |  |  | X |  |
| Test de données | X |  |  |  |
| Test de valeurs limites | X |  |  |  |
| Test des composantes | X |  |  |  |
| Test exploratoires |  |  | X |  |
| Utilisabilité |  |  | X |  |



## Critères d’acceptation

* Acceptation du client
* Le succès de 95% de l’ensemble des tests fonctionnels
* Un maximum de 2h de tests par sprint.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fonctionnalité** | **Critères d’entrée** | **Critères de sortie** | **Résultat attendu** | **Acceptation** |
| **Ajout de marchandises à donner** | Toutes les informations qui concernent la marchandise (ID, nom du produit etc …) et ceux qui concerne l’authentification du donneur | Ajout de la marchandise avec succès.  Ajout de la marchandise dans la base de données | Ajout de la marchandise avec succès.  Ajout de la marchandise dans la base de données | Succès ou échec |
| **Liste des marchandises à collecter** | Authentification de l’utilisateur et sélection de la liste des marchandises à collecter | Affichage de la liste des marchandises à collecter | Affichage de la liste des marchandises à collecter | Succès ou échec |
| **Réservation de marchandises à collecter** | Authentification de l’utilisateur et réservation de la marchandise désirée | Changement du statut de la marchandise de « disponible » à « réservé ». | Changement du statut de la marchandise de « disponible » à « réservé ». | Succès ou échec |

Voici un tableau qui permet de résumer les critères d’entrée et de sortie pour le cas de notre projet :

|  |  |
| --- | --- |
| Comportement fonctionnel |  |
| Caractéristiques de qualité | Fiabilité : Il faut que les données soient affichées en temps réel car deux organismes ne peuvent réserver la même marchandise. Aussitôt qu’un organisme fait une réservation, cette marchandise doit apparaître comme réservé pour tout le monde.  Exactitude : Donné un temps raisonnable à l’organisme pour qu’il collecte sa marchandise, sinon elle redeviendra accessible à tous.  Utilisabilité : Facile à utiliser. |
| Scénarios (cas d'utilisation) |  |
| Règles commerciales | Avant d’accepter des alimentes de la part d’un donneur on doit vérifier la date de péremption.  Reçus d’impôts pour dons |
| Interfaces externes | Efficacité : Lien avec le serveur doit être fonctionnel. |
| Contraintes | Sécurité : Certaines options du logiciel ne doivent être vus par le publique comme la listes des marchandises disponibles à collecter pour éviter les collecteurs malhonnêtes. |

## Analyse des risques

### Liste de tests en fonction des risques :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tests/ Risques | 1 | 2 | 3 | 4 |
| Ad hoc | X |  |  |  |
| Analyse de traçabilité |  |  |  | X |
| Revue de code | X |  |  |  |
| Revues des exigences |  |  |  | X |
| Test black box |  |  | X |  |
| Test d’acceptation |  |  | X |  |
| Test de données |  |  |  | X |
| Test de valeurs limites |  |  |  | X |
| Test des composantes |  |  | X |  |
| Test exploratoires |  |  | X |  |
| Utilisabilité |  |  | X |  |

### Niveau d’intégrité

Niveau d’intégrité exigé pour les fonctionnalités.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Unitaires | Intégration | Système | Acceptation |
| Ajout de marchandises à donner. |  |  |  | X |
| Liste des marchandises à collecter. |  |  |  | X |
| Réservation de marchandises à collecter. |  |  |  | X |

# Critères d’entrée et sortie

* Acceptation du client
* Le succès de 95% de l’ensemble des tests fonctionnels
* Un maximum de 2h de tests par sprint.
* Acceptation du client
* 80% des tests fonctionnels exécutés avec succès
* Maximum 1h de tests par itération
* Revues et approbations
* Approuvé

# Environnement

## Environnement

|  |
| --- |
| Windows 7, 8, 10 |
| MacOSX |
| Java 1.8 |
| JUnit 2,3 |
| Office 2007 |
| GIT |

|  |  |
| --- | --- |
| Spécifications techniques | |
| PHP Version | 5.4.37 |
| MySQL Version | 5.5.44-MariaDB-cll-lve |
| Architecture | x86\_64 |
| Operating System | Linux |
| Perl Version | 5.10.1 |
| Kernel Version | 2.6.32-531.29.2.lve1.3.11.1.el6.x86\_64 |
| Apache Version | 2.4.16 |

|  |  |
| --- | --- |
| Choix du langage | |
| Langage | JAVA (langage open source) |
| Grâce à ce langage, nous pouvons : | Programmer une application mobile(Android),  Programmer un service WEB,  Rendre ces programmes multiplateformes  Utiliser un modèle MVC |
| Langue du projet | Français |

Java 1.8

PHP Version 5.4.37

MySQL Version 5.5.44-MariaBD-cll-lve

Architecture x86\_64

Operating System linux

Perl Version 5.10.1

Kernel Version 2.6.32-531.29.2.lve1.3.11.1.el6.x86

Apache Version 2.4.16

## Outils de test

|  |  |
| --- | --- |
| Eclipse | Pour faire la revue de code |
| Word | Pour rédiger les rapports, spécifications des tests |
| Excel | Pour compiler les données de différents tests |
| JUnit | Pour les tests automatisés |
| Test ad hoc | Word ou Exel pour compiler les données |
| Test exploratoire | Word ou Exel pour compiler les données |

- Selenium IDE et Selenium WebDriver pour l’automatisation des tests

- JMeter pour mesurer la performance du logiciel

- Eclipe java pour le code

- Word pour la spécification des tests

- Excel pour la compilation des défauts

## Gestion de configuration



# Analyse des risques

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Niveau de criticité** | | | |
| **Liste des risques** | **Indolores (1)** | **Limites (2)** | **Graves (3)** | **Dramatiques (4)** |
| R1 - Manque d’expérience des programmeurs |  |  |  | **x** |
| R2 - Détection de graves manquements aux règles et aux standards de développement |  |  | **x** |  |
| R3 - Manque d’expérience des testeurs |  |  |  | **x** |
| R4 - Absence prolongée d’un testeur |  |  | **x** |  |
| R5 – Méconnaissance de l’application par les testeurs. | **X** |  |  |  |
| R6 – Le logiciel ne fonctionne pas correctement sur une telle ou telle version d’Android |  |  | **x** |  |
| R7 – Manque de disponibilité du matériel requis. |  |  |  | **X** |
| R8 – Non-respect de l’échéance |  |  |  | **x** |
| R9 – Non-respect du budget | **x** |  |  |  |
| R10 – Tests automatiques mal programmés |  | **x** |  |  |

|  |  |
| --- | --- |
| **Risque** | **Plan d’atténuation** |
| R1 | Offrir aux programmeurs une formation d’appoint accélérée |
| R2 | Faire une revue de code par un programmeur senior connaissant les règles et standards de programmation |
| R3 | Offrir aux testeurs une formation d’appoint accélérée sur l’utilisation des outils de tests qui seront employés pour tester le logiciel « denrée ô suivant » |
| R4 | Assurer la polyvalence et le partage des tâches entre les différents testeurs pour pallier à toute éventualité. Dans le cas d’une absence très prolongée, on essayera de négocier de nouveaux délais avec le client.  (Sous-traitance si nécessaire) |
| R6 | Revoir les performances (rapidité …) du logiciel ainsi que la performance des émulateurs Android déployés |
| R7 | Prévoir avec le support technique de nouvelles ressources afin de faire un lissage des ressources disponibles |
| R8 | Renégocier les délais le plutôt possible avec le client |

# Revues et approbations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Nom | Historique | Date | Approbation |
|  |  |  |  |  |