Rapport de tests

Bancomat Web

**Java EE / QDL**

# Introduction

Dans le cadre du cours de la HE-ARC « Java Entreprise Édition » il nous est demandé, à titre éducatif, de développer une application web à l’aide du Framework JSF (Java Server Faces). Ce projet doit être suivi à l’aide de procédures de test pour assurer la qualité du logiciel. Ce document relate la stratégie, planification, spécification ainsi que le bilan des tests.

Le site web est un système de bancomat disposant d’une interface, destinée à des clients, de retrait / dépôt / transfert d’argent à partir de compte bancaire qu’ils peuvent demander à créer. Le programme disposera également d’une interface spéciale d’administration des comptes et des utilisateurs.

# Stratégie de test

La partie suivante aborde les différents aspects de la stratégie des tests.

## Qualité du produit

 Le produit doit fournir 2 rôles bien distincts qui n’ont pas les même droit et fonctionnalités :

* Le client doit pouvoir :
  + Demander l’ouverture d’un compte
  + Retirer
  + Déposer
  + Transférer de l’argent
* L’administrateur doit pouvoir :
  + Accepter les demandes d’ouverture de compte
  + Gérer les comptes clients
  + Gérer les personnes inscrites

## Objectifs de test

Les choses importantes à tester sont :

* Les accès utilisateurs. Il ne faut pas qu’on puisse accéder au compte d’un autre cela entrainerait un gros problème de sécurité. Entre autres, il serait idéal de tester toute la sécurité et d’effectuer des transferts cryptés en HTTPS au vu de l’importance des données.
* La procédure d’ouverture de compte car cela entrainera le blocage de l’utilisation de l’application.
* Le bon fonctionnement des retraits, dépôts, transferts clients. Autrement cela rendrait également l’utilisation obsolète du site.

Éventuellement la gérance des clients et comptes clients en cas de beug lors de la création. Les autres fonctionnalités sont prioritaires car cela n’empêche pas le déploiement de l’application.

## Périmètre de test

La partie test du projet sera défini uniquement par des tests systèmes (performances et sécurité) ainsi que par des tests de validation défini par les objectifs de test. Les programmes d’automatisation de tests tel que Selenium et JMeter permettant un travail réduit

# Plan de test

## Niveau de test

### Test système

## Organisation

## Infrastructure

# Spécifications détaillées des tests

## Clusters

## Conditions de test

# Rapport d’exécution des tests

# Liste des bugs

|  |  |  |
| --- | --- | --- |
|  |  |  |