​

|  |
| --- |
| ​​  Cégep Marie-Victorin |
| ​​Devoir 4 |
| ​​ CodeMasters – Mathis, Alan, Tri, Mathieu  2024-02-18​ |

​

​

Table des matières

[Partie 1 : Analyse 2](#_Toc245193209)

[Technique de tests utilisée dans des applications natives 3](#_Toc2098459511)

[Techniques et outils de tests envisagés 3](#_Toc952201000)

# Partie 1 : Analyse

## Technique de tests utilisée dans des applications natives

Lors de la réalisation d’une application de bureau, il y a plusieurs sortes de tests qui s’offre à nous. En voici une liste ainsi que leurs descriptions :

* Tests fonctionnels : Les tests fonctionnels sont une approche qui vise à tester l’application afin de vérifier si toutes les fonctionnalités correspondent aux demandes du client.
* Tests de compatibilité : Les tests de compatibilité vise à tester si l’application fonctionne bien sur d’autres systèmes d’exploitation et configurations matérielles.
* Tests de performances : Les tests de performance permettent, grâce à des simulations, de tester quels sont les performances de l’application selon différentes conditions.
* Tests d’utilisabilité : Les tests d’usabilité permettent de vérifier si l’application est facile à prendre en main et accessible à tous tout en identifiant les différents éléments à améliorer.
* Tests de sécurité : Les tests de sécurité permettent de vérifier si l’application est sécuritaire en identifiant les différentes vulnérabilités ou risques potentiels.
* Tests d’installation : Les tests d’installation permettent de vérifier si l’application s’installe facilement sur un nouvel appareil sans causer de problèmes ainsi que vérifier si celle-ci fonctionne comme elle était attendu.
* Tests de régression : Les tests de régression permettent de vérifier si l’application continue de fonctionner normalement après des mises à jour ou des modifications sur celle-ci.

## Techniques et outils de tests envisagés

Dans ce projet, nous envisageons d’utiliser les tests de fonctionnalité, les tests d’utilisabilités, de compatibilité, de performance ainsi que les tests de sécurité. Les outils de test que nous envisageons d’utiliser sont des outils d’automatisation de tests tel que Leapwork, Selenium, Test Studio.

Pour les tests de fonctionnalité, vérifier les diverses fonctionnalités de l’application étape par étape et vérifier que chaque fonctionne comme voulu.

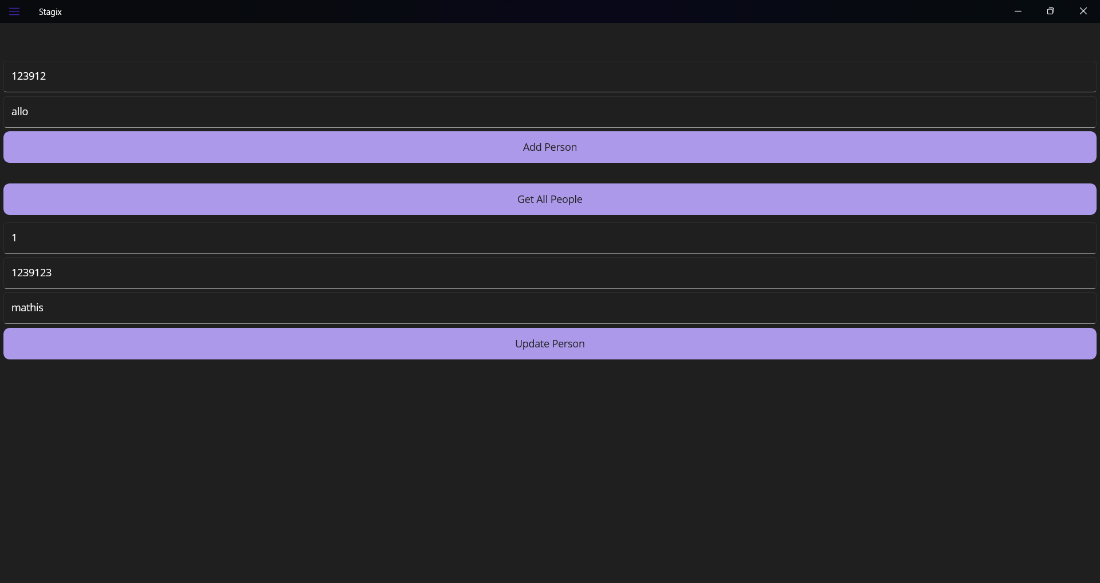
Pour les tests de compatibilité, tester sur différentes versions d’OS et différentes résolutions d’écrans.

Pour les tests d’usabilité, recruter des participants pour tester l'application et observer le taux de succès des défis d'usabilité prédéfinis par notre équipe, tels que la facilité de navigation et l'expérience utilisateur (UX).

Pour les tests de performance, évaluer la réactivité de l’application, les ressources qu’il occupe (CPU, RAM), et le soumettre à des stress tests avec JMeter.

Pour les tests de sécurité, examiner le code et tester sa vulnérabilité contres des attaques comme l’injection SQL et le XSS.





Demo crud:

<https://collegemv-my.sharepoint.com/:v:/g/personal/2289642_collegemv_qc_ca/Earx5DARAvhIqvbbHmLOlfEBH2Xoo0W9MNtyWmYAODkg7Q?e=rtTQx1>