 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing		PAGE 1	PAGES 11


PROPOSITION
Projet de fin d'études
Département de génie logiciel et des TI

IoT Systems Testing

Auteurs
Armand Dim Dim - DIMA01048904
Maximilien Blanchard-Bizien - BLAM13099804

Professeur superviseur
NAOUEL MOHA

Date
23 mai 2024

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing		PAGE 2	PAGES 11

Suivi des changements
 *A – Ajouté M – Modifié S – Supprimé

NUMÉRO DE VERSION	DATE aaaa/mm/jj	NUMÉRO DE FIGURE, TABLE OU SECTION	A* M S	BRÈVE DESCRIPTION DU CHANGEMENT	NUMÉRO DE DEMANDE CHANGEMENT



 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 3	PAGES 11	

TABLE DES MATIÈRES

1. PROBLÉMATIQUE ET CONTEXTE	4
2. OBJECTIFS DU PROJET	4
3. MÉTHODOLOGIE	5
4. COMPOSITION DE L'ÉQUIPE ET RÔLES	6
5. LIVRABLES ET PLANIFICATION	6
5.1 Description des artéfacts	6
5.2 Planification	7
6. RISQUES	8
7. TECHNIQUES ET OUTILS	9
8. RÉFÉRENCES	10
ANNEXE A : PLAN DE TRAVAIL	11

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 4	PAGES 11	

1. PROBLÉMATIQUE ET CONTEXTE

La complexité des tests manuels dans les systèmes IoT, caractérisés par leur diversité de composants et de couches, les rend sujets à des erreurs et difficiles à réaliser. Ceci est particulièrement problématique dans des projets comme "Where is my Professor" (WIMP), où différents outils et applications doivent fonctionner de concert. Pour garantir le bon fonctionnement de ce projet et de ses outils associés, tels que le robot "Buddy", un outil de tests spécifiques doit être mis en place, notamment la génération de "Payloads" contenant les instructions nécessaires aux différents éléments du projet. Une automatisation adaptée des tests devient ainsi cruciale, non seulement pour optimiser les processus de test, mais également pour assurer la performance, la sécurité et la fiabilité des logiciels IoT. Cependant, le développement d'une telle solution doit relever des défis tels que la compréhension de la complexité des systèmes IoT, la prise en compte de leur hétérogénéité, et l'intégration harmonieuse avec les outils existants. En résumé, l'enjeu principal réside dans la mise en place d'un processus de test automatisé et complet, capable de répondre aux spécificités des systèmes IoT et d'assurer leur qualité globale.

Ensuite, il est nécessaire d'adapter les techniques de test aux spécificités des composants et des couches logicielles des systèmes IoT. Les tests doivent être capables de vérifier la communication entre les appareils, la gestion des données, ainsi que les interactions avec les services cloud. Par exemple, dans le cas du projet WIMP, cela pourrait impliquer des tests de connectivité entre le robot et les smartphones des utilisateurs, des tests de gestion des données de localisation et des tests de résistance aux pannes de connexion.


La conception et l'implémentation d'un outil de test automatisé IoT constitue le travail à faire dans le cadre de ce projet de fin d'étude. Cependant, un tel projet pose de nombreux défis à savoir :

- Comment créer des payloads représentatifs pour tester efficacement les fonctionnalités des systèmes IoT et obtenir une couverture exhaustive des tests?
- Comment élaborer des tests couvrant toutes les couches et fonctionnalités des systèmes IoT, tout en restant représentatifs des cas d'utilisation réels ?
- Comment garantir que les tests automatisés évaluent efficacement la performance, la sécurité et la fiabilité des systèmes IoT.

La mise en œuvre d'un outil de test automatisé adapté aux systèmes IoT est indispensable pour garantir leur performance et fiabilité. Cependant, cela nécessite une approche soigneusement élaborée pour surmonter les défis spécifiques liés à la diversité des composants et couches logicielles.

2. OBJECTIFS DU PROJET

Pour ce projet, nous devons travailler sur le logiciel "Where is my Professor?", ou WIMP, principalement en créant un outil de tests fonctionnels, et qui permettent de confirmer que les fonctionnalités du programme marchent correctement et de voir si un changement apporté dans le code brise ces fonctionnalités. Le logiciel est composé et utilise différents composants, tel qu'un petit robot appelé "Buddy", et une montre "Fitbit", pour permettre aux étudiants de mieux savoir où se trouve leur professeur, et voir s'il est disponible ou non. L'objectif principal étant de développer un outil d'automatisation complet dédié aux tests des systèmes IoT. Cet outil

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 5	PAGES 11	

permettra de surmonter les défis inhérents aux tests manuels et aux limitations des outils d'automatisation existants, contribuant ainsi à une amélioration significative de la qualité logicielle, de la rapidité de développement et de la fiabilité des systèmes IoT. Nous devons implémenter une logique de type microservice à notre outil, permettant de mieux apporter des modifications au logiciel. À la fin du projet, et avec cet outil, nous serons capable de tester avec succès la majorité des fonctionnalités du robot “Buddy” de façons rapide et automatisé.

De plus, nous devons aussi créer une interface pour ce logiciel pour pouvoir mieux exécuter les tests. Non seulement ça, mais cette interface doit aussi être capable de prendre des “Payloads”, des fichiers JSON qui contiennent des instructions à envoyer aux autres applications. Il faudra donc que cette application, et le lien entre les autres outils, fonctionnent parfaitement pour pouvoir exécuter les tests avec succès.

3. MÉTHODOLOGIE

Pour atteindre les objectifs de ce projet, nous adopterons une approche itérative basée sur les meilleures pratiques en matière de développement logiciel et de tests. Notre méthodologie comprendra les activités suivantes :


Analyse initiale : Nous commencerons par une analyse approfondie des spécifications du projet WIMP ainsi que des exigences fonctionnelles et non fonctionnelles. Cette analyse nous permettra de comprendre le code source disponible, les fonctionnalités, ainsi que les besoins en termes de tests et d'identifier les différents composants et couches logicielles du système IoT.

Conception de l'outil de tests : Sur la base de l'analyse initiale, nous concevons un processus de test exhaustif qui couvre toutes les couches et fonctionnalités du système IoT. Nous définirons les cas de test, les scénarios d'utilisation et les critères d'acceptation pour chaque composant du système.

Implémentation de l'outil de tests automatisés : Nous mettrons en œuvre des tests automatisés adaptés aux spécificités des systèmes IoT. Cela inclura le développement de scripts de test, l'intégration d'outils de test automatisés et la configuration des environnements de test.

Exécution des tests : Nous exécuterons les tests automatisés dans l'outil de tests développé en utilisant des payloads et captureront les résultats pour analyse des défauts et les problèmes rencontrés pour chaque cas.

Analyse des résultats : Nous analyserons les résultats des tests pour évaluer la qualité du logiciel IoT et identifier les zones d'amélioration. Nous utiliserons des métriques de qualité telles que le taux de réussite des tests, le temps moyen de détection des défauts et la couverture des tests pour évaluer la performance du processus de test.

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing		PAGE 6	PAGES 11

La méthodologie Agile Scrum se présente comme une approche idéale pour relever les défis des tests automatisés dans les systèmes IoT, tels que définis dans notre problématique. En établissant une vision claire du processus de test automatisé, en formant une équipe Scrum multidisciplinaires et en planifiant des sprints avec des objectifs spécifiques, nous pouvons progressivement concevoir et mettre en œuvre des solutions de test adaptées à la diversité des composants et à la complexité des couches logicielles des systèmes IoT. En intégrant les tests automatisés dans le flux de travail de développement, en gérant efficacement l'hétérogénéité des environnements IoT et en accordant une attention particulière à la performance, la sécurité et la fiabilité, cette approche Agile Scrum nous permettra de garantir une qualité globale des logiciels IoT tout en nous adaptant aux évolutions constantes de ce domaine technologique.

4. COMPOSITION DE L'ÉQUIPE ET RÔLES

Membre de l'ÉTS

Prénom	Rôle(s) (voir annexe A)	Responsabilités
1.Maximilien	Développeur	Développeur backend et aide la révision et documentation.
2.Armand	Développeur	Développeur frontend et aide la révision et documentation.


Membre de Concordia

Prénom	Rôle(s) (voir annexe A)	Responsabilités
1.Jean	Chef de projet	Supervise le projet.
2.Yahia	Responsable technique	Choix technologique et aide au développement.

5. LIVRABLES ET PLANIFICATION


5.1 Description des artefacts

Nom de l'artefact	Description
GTI_LOG_795_Plan de projet.docx	Document présent qui décrit la planification du projet de fin d'étude.
GTI_LOG_795 Rapport Technique.docx	Rapport contenant l'ensemble des réalisations du projet.
Presntation Finale.pptx	Document de présentation orale du projet de fin d'étude.
Code source	Code source outil de test et scripts de tests automatisés.

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing		PAGE 7	PAGES 11


5.2 Planification

Nous envisageons d'adapter la méthodologie Agile Scrum pour répondre aux besoins de notre planification déjà établie. Chaque itération de trois semaines correspondra à un sprint, avec des rencontres régulières avec la professeure superviseure pour présenter l'avancement, fournir les livrables et planifier la prochaine étape. Entre ces périodes, des rencontres journalières avec le chef de projet responsable de Concordia permettront de discuter des problèmes rencontrés et de décider des stratégies à adopter. En dehors de ces périodes, l'équipe se réunira autant de fois que nécessaire pour coordonner les rencontres à venir et planifier les présentations. Cette approche itérative et collaborative permettra de maintenir une communication efficace, de s'adapter aux exigences changeantes du projet et de garantir une progression continue vers les objectifs fixés. Le tableau de l'annexe A présente donc la planification pour la réalisation des tâches que nous aurons à effectuer.

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 8	PAGES 11	

6. RISQUES

Risque	Impact	Probabilité	Mitigation / atténuation
Disponibilités basses pour les membres de l'équipe.	Limite la capacité de travailler sur le projet.	Haut	<p>Si l'un des membres de l'équipe ne peut pas venir au laboratoire à Concordia, il pourra travailler à la place sur une autre partie du projet, comme le rapport ou sur le prototypage de l'interface.</p> <p>Pour les cas où un membre de l'équipe ne peut absolument pas venir, un autre membre de l'équipe pourra faire un résumé de ce qui c'est passé cette journée-là, ou poser des questions que le membre manquant aurait aimé poser. Ensuite, ce coéquipier peut relier les informations qu'il a reçues à son collègue absent.</p>
Détermination des rôles et responsabilités des membres de l'équipe ne sont pas très précis.	Ralentir le projet. Peut causer une mauvaise approche sur la réalisation de certaines tâches.	Faible	<p>Si un membre de l'équipe n'est pas sûr des tâches qu'il est supposé faire, il peut demander à l'un des membres de l'équipe Ptidej, soit directement, par courriel ou avec "Rock.Chat".</p> <p>Aussi, utiliser un système de billets, tel que les outils Jira ou Kaban, peut aider à mettre en place quelles tâches le coéquipier doit compléter.</p>
Peu d'expérience en réseau et envers l'Internet des objets.	Peut causer une mauvaise approche sur la réalisation de certaines tâches. Ralentir le projet.	Moyen	Si un membre de l'équipe a besoin de clarification ou d'explications sur une partie du projet, il pourra tout simplement demander à l'un des membres de l'équipe Ptidej pour aider à clarifier les explications. De plus, si le problème est plus général, le coéquipier peut aussi juste faire ses propres recherches en ligne.
La taille et la complexité du projet est moyennement grande dû à la multitude d'objets impliqués.	Ralentir le projet. Plus grande compréhension requise.	Moyen	Si on n'a pas besoin de travailler sur certaines parties du projet, on peut juste les ignorer pour le moment. Aussi, pour les parties du code qui ne sont pas très claires, poser des questions aux développeurs peut permettre aux membres de l'équipe de clarifier ces parties.

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 9	PAGES 11	

Certaines des technologies utilisées, comme le robot “Buddy”, sont assez récentes et complexes.	Plus grande compréhension requise.	Moyen	Poser des questions aux membres de Concordia peut aider à clarifier comment fonctionne les technologies utilisées peut aider. Aussi, rechercher des ressources en ligne peut aider, si la technologie est publiée depuis un bon moment.
---	------------------------------------	-------	---


7. TECHNIQUES ET OUTILS

Outils pour générer les livrables:

- **Android Studio et Android SDK:** Outils essentiels pour développer des applications Android comme le robot Buddy. Android Studio fournit un environnement de développement intégré (IDE) complet pour la création d'applications Android, tandis que le SDK Android offre les bibliothèques et les outils nécessaires pour exécuter et tester ces applications sur des appareils ou des émulateurs Android;
- **Visual Studio Code:** Outil pour travailler sur différents types d'applications. Pour ce projet, l'outil est utilisé principalement pour programmer en Javascript;
- **Node.js:** Cet environnement d'exécution JavaScript permet d'exécuter du code JavaScript en dehors d'un navigateur web. Il est particulièrement utile pour développer des applications et des outils côté serveur;
- **Vue.js:** Ce framework JavaScript progressif simplifie le développement d'interfaces utilisateur (IU) pour les applications web. Il propose une approche déclarative et basée sur des composants pour construire des IU;
- **Vuetify:** Ce “framework material design” pour Vue.js fournit une collection de composants et de styles pré-construits pour créer des IU cohérentes et visuellement attrayantes qui respectent les directives material design;
- **Android Debug Bridge (adb):** Cet outil en ligne de commande sert de pont polyvalent entre Android Studio et les appareils ou émulateurs Android. Il permet le débogage, les transferts de fichiers et le contrôle des appareils;
- **Buddy SDK:** Ce kit de développement logiciel (SDK) spécialement conçu pour le robot Buddy fournit les outils et les API nécessaires pour interagir avec les capacités matérielles et logicielles du robot;
- **Google Drive:** Pour les fichiers collaboratif;
- **Google docs:** Pour nous permettre de rédiger nos rapports et présentations du projet ensemble, et sur une seule copie de ces documents;
- **Serveur Discord:** Pour la communication entre les membres de l'équipe et la planification des rencontres;
- **GitHub:** Pour le contrôle de version et la centralisation du code.

Outils pour gérer le projet:

- **Github:** Outil qui permet de gérer et partager les projets en ligne. Utile pour travailler sur le projet à distance et séparément;
- **Kaban:** Sous-outil de Github qui permet de gérer et assigner des tâches sous forme de billets (“Tickets” en anglais);

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing	PAGE 10	PAGES 11	

Techniques supplémentaires:

- **Drawio:** Outil de dessin en ligne gratuit qui permet de créer différents types de diagrammes, tels que des organigrammes, des diagrammes UML, des diagrammes de flux et des schémas réseau;

8. RÉFÉRENCES

Landry, Guillaume. Mahara Moukene, Moïse. Rousseau, Guyllaume. Houle-Legault, Étienne. 2022. « Proposition : Migration de systèmes logiciels monolithiques vers des Microservices par le biais de l'intelligence artificielle et de l'analyse sémantique ». En ligne, documentation référenciel. 10 p.

<https://drive.google.com/file/d/1Ld28Bxuk2_zG5v9e9Plcwp_ASQVz9Ofa/view?usp=sharing>. Consulté le 13 mai 2024.

Hoang, Keving. 2022. « BUDDYOS : Manual installation guide v 1.0 ». En ligne. 7 p.


<<https://drive.google.com/file/d/1si8FIQ8idoKaCI5HLJJ-KAJqUtQE4fhR/view?usp=sharing>>. Consulté le 8 mai 2024.

Hoang, Kevin. Semisalov, Ilya. Valentin, Jacques. Cocquempot, Mickael. 2022. « BUDDY SDK : User Guide v 2.3.10 ». En ligne. 79 p.

<<https://drive.google.com/file/d/15T9eGU40uOGK0B2lrclgvvOAQHq1parL/view?usp=sharing>>. Consulté le 8 mai 2024.

Baptiste Minani, Jean. 2024. « IoT System Testing Project ». En ligne. 20 p.

<https://docs.google.com/presentation/d/11Xk_BMXKy2PGeDxHG4XppG4ysxQVnv7bL0Co59AEtCQ/edit?usp=sharing>. Consulté le 6 mai 2024.

 Département de génie logiciel et des TI	PROJET LOG/GTI 795	DOCUMENT NO.	DATE 2024-05-23	VERSION 1.0
	TITRE IoT Systems Testing		PAGE 11	PAGES 11

ANNEXE A : PLAN DE TRAVAIL

Le tableau suivant présente la planification pour la réalisation des tâches ou artefacts décrits précédemment.

#	Commence	Termine	Tâches/Jalon	Livrable(s)	Responsable(s)
0.0	2024-05-03	2024-05-03	Première rencontre avec la professeure, son adjoint et le chef de projet à Concordia	S.O.	Professeure superviseure
0.1	2024-05-06	2024-05-10	Analyse initiale et analyse du code WIMP	S.O.	Professeure superviseure
0.2	2024-05-11	2024-05-24	Rencontre quotidienne sur l'avancement de l'analyse du code et objectifs	S.O.	Chef de projet
0.3	2024-05-11	2024-05-24	Rencontre – professeure superviseure. Remise du plan de projet et planification prochaine itération	Plan de projet	Professeure superviseure
1.0	2024-05-25	2024-05-31	Conception et implémentation du frontend de l'outil de test	Outil de test IoT	Chef de projet
1.1	2024-06-01	2024-06-07	Implémentation du backend de l'outil de test	Outil de test IoT	Chef de projet
1.2	2024-06-08	2024-06-16	Connexion backend et frontend	Outil de test IoT	Chef de projet
1.3	2024-06-10	2024-06-20			
2.0	2024-07-23		Rencontre – Professeur superviseur	Architecture	Professeure
2.1	2024-08-02		Rencontre – Professeur superviseur	Rapport d'étape	Professeure
2.2	2024-08-23		Présentation itération 3	Présentation finale	Professeure