

Rapport de Projet RPA

Introduction

Ce rapport décrit la réalisation d'un projet d'automatisation des processus robotiques (RPA) utilisant Selenium pour la partie RPA, Next.js pour le frontend, et Node.js pour le backend. Le projet est centré sur trois fonctionnalités principales : la planification des visites, la génération automatisée de rapports de visite, et la mise à jour des données des contacts. Chaque partie du projet a été développée par différents membres de l'équipe, avec une collaboration étroite assurant une compréhension et une implication mutuelles.

Fonctionnalités

1. Planification des Visites

Responsable : KONE Ousmane

Description

La première fonctionnalité consiste en l'automatisation de la planification des visites chez les professionnels de la santé. Les principales caractéristiques incluent :

- Automatisation des Plannings : Utilisation de Selenium pour automatiser la planification des visites en fonction des disponibilités des délégués médicaux et des priorités définies.
- Intégration des Calendriers : Intégration avec les calendriers des délégués pour éviter les conflits de rendez-vous. Cette intégration assure que les visites sont planifiées de manière optimale sans chevauchement.

Procédures et Outils Utilisés

- Selenium : Pour l'automatisation des tâches répétitives de planification.

- YouTube et Documentation : Des tutoriels sur YouTube et la documentation officielle de Selenium ont été utilisés pour surmonter les difficultés techniques rencontrées.
- Méthodologie MERISE : Une analyse préalable des besoins a été effectuée à l'aide de la méthode MERISE pour assurer une bonne modélisation des données et des processus.

2. Rapport de Visite Automatisé

Responsable : Angui David Hermas

Description

La deuxième fonctionnalité concerne l'automatisation de la génération de rapports de visite. Les principales caractéristiques incluent :

- Génération Automatisée des Rapports : Après chaque rencontre avec un professionnel de la santé, des rapports sont générés automatiquement.
- Uniformité des Informations : Utilisation de modèles de rapport pour standardiser les informations recueillies, assurant une présentation cohérente et uniforme.

Procédures et Outils Utilisés

- Node.js : Utilisé pour le backend, assurant la communication entre le frontend et les systèmes de gestion des données.
- Selenium : Intégré pour extraire automatiquement les données nécessaires à partir des systèmes de gestion des visites.
- Tutoriels en Ligne et Documentation : Des ressources en ligne, y compris des tutoriels et la documentation de Node.js, ont été utilisées pour résoudre les problèmes techniques.

3. Mise à Jour des Données des Contacts

Responsable : KOUASSI Christ

Description

La troisième fonctionnalité concerne l'automatisation de la mise à jour des informations de contact des professionnels de la santé. Les principales caractéristiques incluent :

- Mise à Jour Automatique : Utilisation de robots RPA pour récupérer les données des e-mails ou des formulaires en ligne et les mettre à jour dans la base de données.
- Précision des Données : Assurer que les informations de contact sont toujours à jour et précises.

Procédures et Outils Utilisés

- Selenium et Email Parsers : Utilisation de Selenium pour l'automatisation et de parsers pour extraire les données des e-mails.
- Ressources en Ligne : Utilisation de tutoriels et de documentation en ligne pour résoudre les problèmes rencontrés.
- Analyse MERISE : Une analyse préalable à l'aide de la méthode MERISE pour une bonne conception de la base de données.

Collaboration et Répartition du Travail

Bien que chaque partie du projet ait été dirigée par un membre spécifique de l'équipe, nous avons tous collaboré et contribué à chaque partie du projet. Cela a permis une compréhension globale des différentes composantes et une implication de chacun dans le projet. Le développement du frontend avec Next.js a été réalisé en collaboration par tous les membres de l'équipe.

Difficultés Rencontrées et Solutions

Nous avons rencontré plusieurs difficultés techniques tout au long du projet. Voici quelques-unes des principales difficultés et les solutions apportées :

- Problèmes d'intégration avec les calendriers : Résolus en utilisant des tutoriels sur YouTube et la documentation des APIs de calendriers.
- Génération de rapports standardisés : Résolus en utilisant des moteurs de templates et en consultant des ressources en ligne pour la manipulation des templates.
- Extraction et mise à jour automatique des données : Résolus en utilisant des parsers pour les e-mails et en intégrant Selenium pour l'automatisation des mises à jour.

Conclusion

Ce projet RPA a permis de réaliser des automatisations cruciales pour la gestion des visites et des rapports dans le domaine de la santé. Grâce à une collaboration efficace et à l'utilisation d'outils et de ressources adaptés, nous avons pu surmonter les défis et aboutir à une solution fonctionnelle et efficace. L'analyse préalable à l'aide de la méthode MERISE a également joué un rôle clé dans la réussite de ce projet.