



**TECHNOLOGIE DE L'INFOROUTE**  
**INF37407 (MS)**

UNIVERSITÉ DU QUÉBEC À RIMOUSKI  
DÉPARTEMENT DE MATHÉMATIQUES, D'INFORMATIQUE ET DE GÉNIE

---

**RAPPORT**  
**Travail Pratique**

---

**ÉQUIPE :**  
Byckel Reauchel Kenfack Kemda  
Joel Kayemba

**PROFESSEUR :**  
Yacine Yaddaden, Ph. D.

Date : 27 octobre 2024

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Objectif de l'application . . . . .	3
<b>2</b>	<b>Technologies utilisées et Structure de l'application</b>	<b>3</b>
<b>3</b>	<b>Actions Redux, Reducers et Interface utilisateur</b>	<b>4</b>
<b>4</b>	<b>Conclusion</b>	<b>4</b>

# 1 Introduction

Ce rapport présente le projet de développement de l'application Web MECAZEN pour la gestion des rendez-vous et de la facturation dans le cadre des réparations automobiles, réalisé pour le cours INF37407. MECAZEN permet aux utilisateurs de gérer leurs véhicules, de planifier des rendez-vous avec des mécaniciens, et d'accéder à des informations de sécurité automobile via une API externe. Elle utilise React.js et Redux pour la gestion de l'état global de l'application.

## 1.1 Objectif de l'application

L'objectif principal de cette application est de fournir aux utilisateurs une plateforme intuitive et fonctionnelle pour faciliter la gestion de leurs véhicules, ainsi que la planification des rendez-vous de maintenance et de réparation. En tant que clients, ils peuvent non seulement ajouter, modifier ou supprimer des véhicules dans leur profil, mais également planifier, modifier ou annuler des rendez-vous pour des interventions techniques avec des mécaniciens. Afin de renforcer la sécurité des utilisateurs, l'application permet également de rechercher et afficher des informations critiques pour la sécurité des véhicules en s'appuyant sur l'API NHTSA. Ce système rend ainsi l'expérience utilisateur complète, en centralisant toutes les informations nécessaires à la gestion des véhicules et en offrant une visibilité accrue sur leur état de sécurité.

# 2 Technologies utilisées et Structure de l'application

Cette application repose sur plusieurs technologies modernes. Elle utilise principalement React.js pour concevoir une interface utilisateur fluide et dynamique. Grâce à Redux, la gestion de l'état global de l'application devient plus efficace, permettant une synchronisation en temps réel des données à travers toute l'interface. L'implémentation de Redux Thunk facilite la gestion des actions asynchrones, comme les requêtes vers les API externes, tandis qu'Axios simplifie les appels HTTP. L'interface est enrichie par React Bootstrap, offrant un design moderne et réactif, tandis que FontAwesome est intégré pour ajouter des icônes interactives qui dynamisent davantage l'application. L'architecture de l'application est organisée en modules pour assurer une gestion optimale des fonctionnalités. [3]

La gestion de l'authentification et des utilisateurs est assurée par Redux, ce qui permet aux utilisateurs de se connecter et de modifier leurs informations de profil de manière sécurisée. L'application distingue entre le profil de client, qui a la possibilité de gérer les véhicules et de planifier des rendez-vous, et celui de mécanicien, qui accède à la liste des rendez-vous et peut gérer les modifications ou annulations demandées par les clients. [4]

Le module de gestion des véhicules offre aux clients une interface simple et efficace pour enregistrer des informations sur leurs véhicules, soit en saisissant les détails manuellement, soit en les récupérant automatiquement à partir de l'API NHTSA en renseignant le VIN ou les informations de base comme la marque, le modèle et l'année. Lors de l'ajout ou de la modification d'un véhicule, l'application affiche automatiquement les rappels de sécurité associés. Cette fonction de rappel garantit aux utilisateurs une connaissance constante des informations de sécurité. [6]

Le module de gestion des rendez-vous est conçu pour faciliter la prise de rendez-vous entre les clients et les mécaniciens. Les clients peuvent réserver un créneau pour une intervention technique, modifier leur rendez-vous si nécessaire, ou l'annuler en fonction de leur disponibilité et de leurs besoins. Ces actions sont suivies d'une validation par le mécanicien, qui peut également fournir une estimation des coûts et de la durée des travaux. Le module permet également aux mécaniciens de suivre et gérer les demandes de modifications, en acceptant ou refusant les propositions faites par les clients.

Enfin, l'application est prête pour intégrer une solution de paiement. Bien que les paiements ne soient pas

gérés directement, les clients pourront, dans les futures versions, sauvegarder leurs moyens de paiement et recevoir des factures une fois que les réparations sont acceptées et finalisées.

### 3 Actions Redux, Reducers et Interface utilisateur

La gestion de l'état global est assurée par Redux, qui centralise les actions et les reducers pour optimiser la synchronisation des données. Les actions Redux sont structurées en modules pour chaque entité. Par exemple, pour les utilisateurs, certaines actions assurent la gestion de la connexion, de la déconnexion et de la mise à jour du profil utilisateur. La gestion des véhicules comprend des actions permettant l'ajout, la modification, et la suppression de véhicules, ainsi que la récupération des rappels de sécurité associés. Le module de rendez-vous inclut également des actions pour planifier, mettre à jour et annuler les rendez-vous. [5]

Les reducers sont conçus pour traiter chaque action et maintenir l'état à jour. Le reducer dédié aux utilisateurs gère l'état de connexion et les informations de profil, tandis que le reducer des véhicules conserve les informations liées aux véhicules des clients, les rappels de sécurité, et gère les erreurs potentielles lors des appels API. Le reducer des rendez-vous centralise toutes les informations liées aux rendez-vous, permettant une gestion fluide des demandes et des modifications entre clients et mécaniciens. [2]

L'interface de l'application, développée avec React Bootstrap, garantit une expérience utilisateur fluide et réactive. Les fonctionnalités sont intuitivement organisées pour permettre une navigation simple et rapide. Le module de gestion des véhicules propose des formulaires pour ajouter, modifier ou supprimer des véhicules, avec une intégration à l'API NHTSA qui permet de compléter automatiquement les informations via le VIN ou les détails du véhicule. La gestion des rendez-vous est également intuitive, permettant aux utilisateurs de consulter l'historique de leurs rendez-vous et de vérifier l'état de leurs demandes. Enfin, chaque utilisateur a accès à un profil personnel, où il peut mettre à jour ses informations et consulter les rappels de sécurité liés à ses véhicules. [1]

### 4 Conclusion

Le développement de cette application web a permis de mettre en œuvre des concepts clés de React.js et Redux, tout en intégrant des APIs externes pour enrichir les données et améliorer l'expérience utilisateur. L'application offre une solution complète pour la gestion de véhicules et de rendez-vous de réparation, tout en assurant une interface réactive et intuitive pour les clients et les mécaniciens. L'intégration avec l'API NHTSA pour la gestion des rappels de sécurité est un aspect crucial qui ajoute de la valeur en automatisant l'acquisition d'informations importantes pour l'utilisateur.

## Références

- [1] Dan Abramov and Andrew Clark. Redux documentation, 2023. <https://redux.js.org/>. pages 4
- [2] National Highway Traffic Safety Administration. Nhtsa vehicle safety api, 2023. <https://vpic.nhtsa.dot.gov/api/>. pages 4
- [3] Axios Project Contributors. Axios documentation, 2023. <https://axios-http.com/docs/intro>. pages 3
- [4] FontAwesome. Fontawesome documentation, 2023. <https://fontawesome.com/>. pages 3
- [5] Inc. Meta Platforms. React documentation, 2023. <https://reactjs.org/docs/getting-started.html>. pages 4
- [6] React Bootstrap Team. React bootstrap documentation, 2023. <https://react-bootstrap.github.io/>. pages 3