

RAPPORT DE STAGE D'ÉTÉ

Diplôme National d'Ingénieur en Sciences Appliquées et Technologiques

Spécialité : Génie Logiciel et Systèmes d'Information

Par :

Roua Ben Amna

Conception et développement d'une application web pour la gestion des stations de lavage

Encadrant professionnel :

Mr. Anis Jaouadi

Réalisé au sein de Lab-It



Année Universitaire : 2023-2024

J'autorise l'étudiante à faire le dépôt de son rapport de stage en vue d'une soutenance.

Encadrant professionnel, **Anis Jaouadi**

Signature et cachet

DÉDICACE

“

Ma profonde gratitude à mes chers parents pour l'éducation qu'ils me prodiguent, avec tous les moyens et au prix de tous les sacrifices qu'ils ont consentis à mon égard, pour le sens du devoir qu'ils m'enseignent depuis mon enfance. Votre générosité et votre bonté ont toujours été un bon exemple pour nous tous. Veuillez trouver en ce travail le fruit de votre dévouement et l'expression de mon respect et mon profond amour.

mes proches et mes chers amis vous m'avez toujours soutenue durant toutes mes études, je vous souhaite beaucoup de bonheur et de réussite.

A tous mes camarades et collègues, je vous aime tous ...

”

- Roua Ben Amna

REMERCIEMENTS

Tout d'abord, je remercie Dieu le tout-puissant de m'avoir donné la santé et la volonté d'entamer et de terminer ce travail.

Mes sincères remerciements à mon encadreur de la société **Mr.Anis Jaouadi** pour son assistance et ses conseils continus durant toute la période du stage.

J'adresse également mes sincères remerciements à toutes les personnes qui ont contribué de près ou de loin au bon déroulement de ce travail.

Enfin, j'adresse aussi mes vifs remerciements à la personne qui va examiner et juger ce travail.

TABLE DES MATIÈRES

Dédicace	ii
Remerciements	iii
Introduction générale	1
1 Etude préalable	2
1 Organisme d'accueil	3
1.1 Hiérarchie de la société	4
1.2 Activité de la société	4
2 Problématique	6
3 Analyse et critique de l'existant	6
3.1 Etude de l'existant	6
3.2 Solution envisagée	7
4 Choix méthodologique	8
4.1 Méthodologie de développement adoptée	8
4.2 Justification du choix	10
5 Langage de modélisation	10
6 Environnement de développement	10
6.1 Environnement matériel	10
6.2 Environnement logiciel	11
7 Choix technique	13
8 Architecture	15

8.1	Architecture logique	15
8.2	Architecture de l'application	17
2	Planification et capture des besoins	20
1	Capture des Besoins	21
1.1	Identification des Acteurs	21
1.2	Besoins Fonctionnels	21
1.3	Besoins Non Fonctionnels	23
1.4	Diagramme de cas d'utilisation global	24
2	Pilotage avec SCRUM	25
2.1	Équipe et rôles	25
2.2	Plan de sprints	25
3	Backlog Produit	25
4	Définition de « Fini »	29
3	Étude et réalisation du sprint 1	30
1	Réunion de planification du sprint (Sprint planning meeting)	31
1.1	Objectif du sprint (Sprint goal)	31
1.2	Backlog du sprint	31
2	Description fonctionnelle des histoires du Sprint1	42
2.1	Diagramme de cas d'utilisation général du sprint 1	42
2.2	Diagramme de cas d'utilisation « S'inscrire »	43
2.3	Diagramme raffiné du cas d'utilisateur « S'authentifier »	43
2.4	Diagramme raffiné de cas d'utilisation « Gestion des voitures »	43
2.5	Description Textuelle de cas d'utilisation « Inscription »	44
2.6	Description Textuelle de cas d'utilisation « Chercher une voiture »	45
2.7	Description Textuelle de cas d'utilisation « Ajouter une voiture »	46
2.8	Description Textuelle de cas d'utilisation « Modifier une voiture »	47
2.9	Description Textuelle de cas d'utilisation « Supprimer une voiture »	48
3	Description comportementale des histoires du sprint 1	49
3.1	Diagramme de séquence « S'inscrire »	49
3.2	Diagramme de séquence « S'authentifier »	50
3.3	Diagramme de séquence « Changer mot de passe en cas d'oubli »	51
4	Description Structurale	52

4.1	Diagramme de classe global de sprint 1	52
4.2	Réalisation	52
5	Tests Postman	53
5.1	Test du cas d'utilisation « Inscription »	53
5.2	Test du cas d'utilisation « Liste des voitures »	54
6	La revue	56
7	Rétrospective	59
7.1	Scrum Board	60
7.2	Tableau de Rétrospective	60
4	Étude et réalisation du sprint 2	62
1	Réunion de planification du sprint (Sprint planning meeting)	63
1.1	Objectif du sprint (Sprint goal)	63
1.2	Backlog du sprint	63
2	Description fonctionnelle des histoires du Sprint 2	71
2.1	Diagramme de cas d'utilisation général du sprint 2	71
2.2	Diagramme raffiné du cas d'utilisation « Gestion des stations de lavage »	71
2.3	Diagramme raffiné du cas d'utilisation « Gestion des Sessions»	72
2.4	Diagramme raffiné du cas d'utilisation « Disponibilité en Temps Réel » .	72
2.5	Description Textuelle de cas d'utilisation « Ajouter une station »	73
2.6	Description Textuelle de cas d'utilisation « Modifier une station »	73
3	Description comportementale des histoires du sprint 2	75
3.1	Diagramme de séquence « Recherche les stations à proximité »	75
3.2	Diagramme de séquence « Ajouter une session »	75
4	Description Structurelle	76
4.1	Diagramme de classe global de sprint 2	76
4.2	Réalisation	77
5	Tests Postman	77
5.1	Test du cas d'utilisation « Modifier une session »	77
5.2	Test du cas d'utilisation « Disponibilité de station de lavage »	78
6	La revue	78
7	Rétrospective	81
7.1	Scrum Board	81
7.2	Tableau de Rétrospective	82

5 Étude et réalisation du sprint 3	83
1 Réunion de planification du sprint (Sprint planning meeting)	84
1.1 Objectif du sprint (Sprint goal)	84
1.2 Sprint Backlog	84
2 Description fonctionnelle des histoires du Sprint 3	92
2.1 Diagramme de cas d'utilisation général du sprint 3	92
2.2 Diagramme raffiné du cas d'utilisation « Gestion d'affluence »	92
2.3 Diagramme raffiné du cas d'utilisation « Supervision des utilisateurs et des gérants »	93
2.4 Description Textuelle de cas d'utilisation « Mettre à jour en temps réel le statut des sessions »	94
2.5 Description Textuelle de cas d'utilisation « Être notifié lorsque un tour de lavage approche »	95
2.6 Description Textuelle de cas d'utilisation « Filtrer la liste des sessions par date »	97
3 Description comportementale des histoires du sprint 3	98
3.1 Diagramme de séquence « Désactiver/Activer les comptes »	98
4 Description Structurale	99
4.1 Diagramme de classe global de sprint 3	99
5 Tests Postman	100
5.1 Test du cas d'utilisation « Afficher liste des utilisateurs et des managers »	100
5.2 Test du cas d'utilisation « Activer un compte »	100
6 La revue	101
7 Rétrospective	106
7.1 Scrum Board	106
7.2 Tableau de Rétrospective	107
Conclusion générale	108
webographie	112

TABLE DES FIGURES

1.1	Logo de société Lab-IT	3
1.2	La hiérarchie de la société Lab-IT	4
1.3	Interafce de ween.tn	7
1.4	Principe du travail avec le Framework Scrum	9
1.5	Logo UML	10
1.6	Logo IntelliJ IDEA	11
1.7	Logo Visual Studio Code	11
1.8	Logo WampServer	12
1.9	Logo Overleaf	12
1.10	Logo Draw.io	12
1.11	Logo Git et GitHub	12
1.12	Logo Trello	13
1.13	Logo Postman	13
1.14	Logo HTML5, CSS3, JavaScript	13
1.15	Logo Angular	14
1.16	Logo Spring boot	14
1.17	Logo My SQL	14
1.18	Architecture de notre API REST	15
1.19	Architecture MVVM dans notre application	16
1.20	Architecture MVC dans notre application Spring Boot	17
1.21	Architecture de l'application SpotWash	18
2.1	Diagramme de cas d'utilisation global	24

2.2	Plan de sprints	25
3.1	Diagramme de cas d'utilisation général du sprint 1	42
3.2	Diagramme de cas d'utilisation « S'inscrire »	43
3.3	Diagramme de cas d'utilisation « S'authentifier »	43
3.4	Diagramme de cas d'utilisation « Gestion des voitures »	43
3.5	Diagramme de séquence « S'inscrire »	49
3.6	Diagramme de séquence « S'authentifier »	50
3.7	Diagramme de séquence « Changer mot de passe en cas d'oubli »	51
3.8	Diagramme de classe global de sprint 1	52
3.9	Table : "user"	52
3.10	Table : "verification token"	53
3.11	Table : "reset password token"	53
3.12	Table : "car"	53
3.13	Code source du test du cas d'utilisation « Créer un compte »	54
3.14	Code source du test du cas d'utilisation « Liste des voitures »	55
3.15	SpotWash - Interface d'inscription	56
3.16	SpotWash - Verification email	56
3.17	SpotWash - Interface d'authentification	57
3.18	SpotWash - Interface de modification profil	57
3.19	SpotWash - interface affectation des permissions	58
3.20	SpotWash - Interface de page d'accueil	58
3.21	SpotWash - Interface d'une liste des voiture	59
3.22	SpotWash - Interface de modification voiture	59
3.23	Scrum board « jour 1 »	60
3.24	Scrum board « dernier jour »	60
4.1	Diagramme de cas d'utilisation général du sprint 2	71
4.2	Diagramme de cas d'utilisation « Gestion des stations de lavage »	71
4.3	Diagramme de cas d'utilisation « Gestion des Sessions »	72
4.4	Diagramme de cas d'utilisation « Disponibilité en Temps Réel »	72
4.5	Diagramme de séquence « Recherche les stations à proximité »	75
4.6	Diagramme de séquence « Ajouter session »	75
4.7	Diagramme de classe global de sprint 2	76
4.8	Table : "car wash station"	77

4.9	Table : "car wash session"	77
4.10	Code source du test du cas d'utilisation « Modifier session »	77
4.11	Code source du test du cas d'utilisation « Disponibilité de station de lavage »	78
4.12	SpotWash - Interface de la liste des stations de lavage	78
4.13	SpotWash - Interface d'ajout d'une station	79
4.14	SpotWash - Interface de suppression	79
4.15	SpotWash - Interface de Carte avec Stations de lavage	80
4.16	SpotWash - Interface d'ajout une réservation	81
4.17	Scrum board « jour 1 »	81
4.18	Scrum board «dernier jour»	82
5.1	Diagramme de cas d'utilisation général du sprint 3	92
5.2	Diagramme de cas d'utilisation « Gestion d'affluence »	92
5.3	Diagramme de cas d'utilisation « Supervision des utilisateurs et des gérants »	93
5.4	Diagramme de séquence « Désactiver/Activer les comptes »	98
5.5	Diagramme de classe global de sprint 3	99
5.6	Code source du test du cas d'utilisation « Afficher liste des utilisateurs et des manager »	100
5.7	Code source du test du cas d'utilisation « Activer un compte »	101
5.8	SpotWash - Interface de list des réservation d'un utilisateur	102
5.9	SpotWash - Interface de list des sessions	102
5.10	SpotWash - List des sessions filtrée	103
5.11	SpotWash - Rappel Email pour la Complétion du Lavage	103
5.12	SpotWash - Interface des statistiques	104
5.13	SpotWash - Interface d'administrateur	105
5.14	Scrum board «jour 1»	106
5.15	Scrum board «dernier jour»	106

LISTE DES TABLEAUX

1.1	Les points faibles et forts du ween.tn	7
1.2	Environnement matériel PC	11
2.1	Backlog produit	26
3.1	Backlog du sprint - Sprint 1	32
3.2	Description Textuelle de cas d'utilisation « Inscription »	44
3.3	Description Textuelle de cas d'utilisation « Chercher une voiture »	45
3.4	Description Textuelle de cas d'utilisation « Ajouter une voiture »	46
3.5	Description Textuelle de cas d'utilisation « Modifier une voiture »	47
3.6	Description Textuelle de cas d'utilisation « Supprimer une voiture »	48
3.7	Tableau de rétrospective « Sprint 1 »	61
4.1	Backlog du sprint - Sprint 2	64
4.2	Description Textuelle de cas d'utilisation « Ajouter une station »	73
4.3	Description Textuelle de cas d'utilisation « Modifier une station »	74
4.4	Tableau de rétrospective « Sprint 2 »	82
5.1	Backlog du sprint - Sprint 3	85
5.2	Description Textuelle de cas d'utilisation « Mettre à jour en temps réel le statut des sessions »	94
5.3	Description Textuelle de cas d'utilisation « Être notifié lorsque un tour de la- vage approche »	95
5.4	Description Textuelle de cas d'utilisation « Filtrer la liste des sessions par date »	97

5.5	Tableau de rétrospective « Sprint 3 »	107
-----	---	-----

INTRODUCTION GÉNÉRALE

CE rapport vise à développer une application Web innovante de « gestion de station de lavage ». L'application est destinée aux utilisateurs qui veulent rechercher rapidement les stations de lavage et estimer le temps qui leur faut pour laver leurs voitures. D'autre part, il s'agit également d'une précieuse ressource pour les gérants des stations de lavage de mieux contrôler le trafic et estimer avec une précision le temps de chaque lavage.

En fait, l'augmentation du nombre de voitures sur la route induit une augmentation de la demande de ce type de service automobile, tandis que les utilisateurs trouvent encore difficile de découvrir les stations de lavages. Ainsi, ce projet proposera une solution numérique à ces deux problèmes à la fois. Il permet aux utilisateurs de rechercher les station de lavage près de chez eux, afficher les données en direct sur la disponibilité de ces stations, recevoir des notifications qu'une station est ouverte ou quand leur tour se rapproche, et gérer leurs profils avec des données de leur voiture. D'autre part, le gérants peuvent utiliser l'application pour gérer les stations et mettre à jour les temps disponibles pour chaque station. Les différentes étapes suivies pour la réalisation de ce projet sont détaillées dans le présent rapport qui se compose de cinq chapitres. Le premier chapitre présente l'organisme d'accueil et propose une problématique, une étude de l'existant et la solution proposée. Les besoins fonctionnels, non fonctionnels et le backlog produit seront détaillés dans le deuxième chapitre. Les trois chapitres 3,4 et 5 exposeront les trois sprints et l'effort effectué pour concevoir l'application web du projet.

Enfin, nous conclurons ce rapport avec la conclusion qui résume l'ensemble du travail accompli, en précisant les compétences acquises durant cette expérience.

CHAPITRE 1

ETUDE PRÉALABLE

1	Organisme d'accueil	3
2	Problématique	6
3	Analyse et critique de l'existant	6
4	Choix méthodologique	8
5	Langage de modélisation	10
6	Environnement de développement	10
7	Choix technique	13
8	Architecture	15

Introduction

Ce chapitre est une présentation du cadre du projet. Tout d'abord, nous allons définir l'entreprise dans laquelle nous avons effectué notre stage d'été.

Ensuite, nous allons présenter une analyse de l'existant pour décortiquer les fonctionnalités déjà développées et surtout de mettre en relief leurs limites afin de dégager une solution.

Enfin, nous déterminerons par définir la méthodologie adaptée.

1 Présentation de l'organisme d'accueil

LAB-IT a été créé en 2020 par son fondateur M. Anis JAOUADI connaissant le secteur IT depuis 1986, LAB-IT est une société tunisienne. Spécialisés dans les services d'ingénierie informatique, LAB-IT se distingue particulièrement dans la conception, l'intégration et la maintenance de l'infrastructure informatique. Ainsi, que ce soit pour la consolidation, le stockage, la virtualisation, la sauvegarde et l'archivage de ces derniers, LAB-IT vend également des composants informatiques. Cette offre de service, à travers une gamme complète de solutions et produits en IT fait de LAB-IT un partenaire de choix dans un marché en pleine croissance.



FIGURE 1.1 : Logo de société Lab-IT

La fiche d'identité est présentée ci-dessous :

- Raison sociale : Lab-IT
- Personne : Anis Jaouadi
- Forme Juridique : Société Privée
- Création : 2020
- Siège Sociale : Rue catacombes, Cité ezzahra, souk lahad, Sousse, Tunisia
- Téléphone : +216 50 234 911
- Adresse Electronique : contact@lab-it.tn / anisjaouadi@lab-it.tn

- Web : <https://lab-it.tn/>

1.1 Hiérarchie de la société

La figure 1.2 présente la hiérarchie de la société Lab IT :

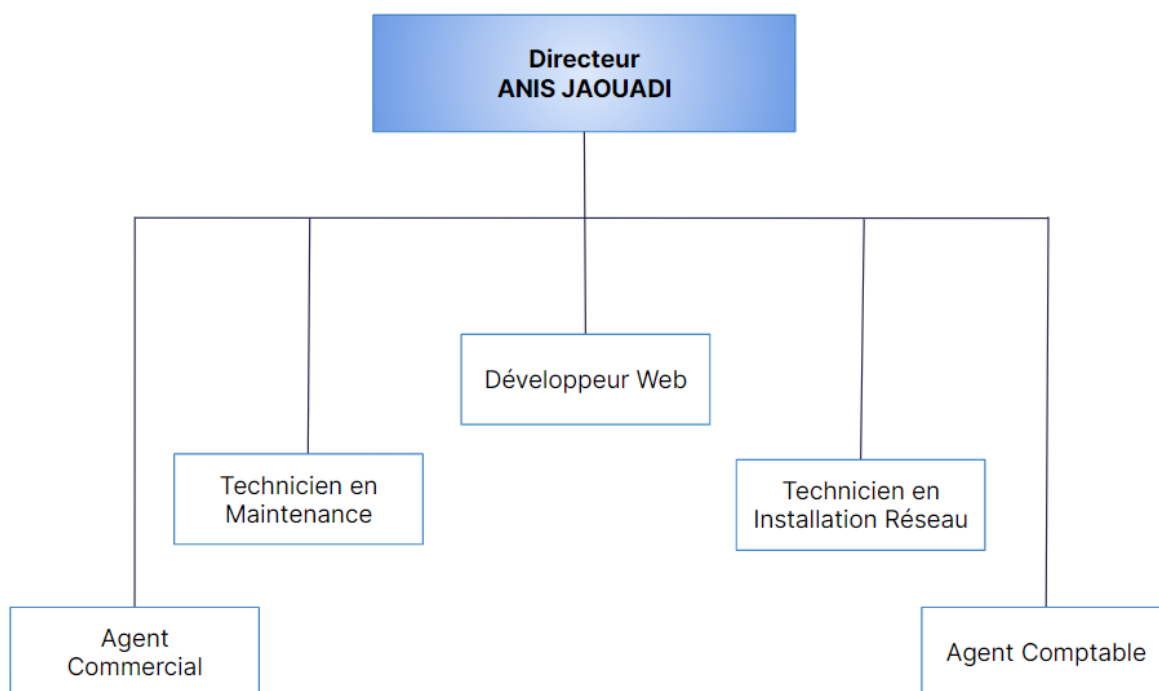


FIGURE 1.2 : La hiérarchie de la société Lab-IT

1.2 Activité de la société

1.2.1 Services developpement web

La société LAB-IT réalise la conception et le développement de site Web vitrine ou E-Commerce. Notre équipe d'infographiste réalise des sites internet pour tous les types et toutes les envergures de projets en fonction de toutes les possibilités techniques et financières de ses clients. Pour véhiculer une image sérieuse sur la toile il faut un site internet qui vous ressemble, professionnel, efficace et totalement personnalisé. Notre équipe réalise également, en nos propres moyens, les créations graphiques et leurs impressions pour l'ensemble des outils de communication : logo, carte de visite, flyer, supports commerciaux etc. L'offre va jusqu'à la création du Site Internet : met en place du site, du catalogue ou du site e-commerce, consulte également les annuaires et les moteurs de recherche, ainsi que des références naturelles ou payantes.

1.2.2 Services maintenance

L'équipe de techniciens intervient dans les meilleurs délais en cas de panne afin de rétablir le bon fonctionnement des systèmes informatiques. Ils déterminent l'origine de la panne et mettent en œuvre les procédures adaptées à des prix très attractifs. La société propose aussi des forfaits de maintenance, flexibles et adaptés à tous les besoins. Elle réalise un audit gratuit permettant de cibler au mieux la maintenance dont les besoin. De plus, pour les urgences, l'équipe intervienne durant toute la journée.

1.2.3 Vente

Lab-it est une boutique de vendte des matériels informatiques en Tunisie. La société propose un site marchand pour un usage informatique à ses clients : pc bureau, pc portables, tablettes etc Ainsi que toutes sortes d'accessoires comme : claviers, casques, souris etc

1.2.4 Installation réseau informatique et logiciels

Avant de procéder à toute installation d'un réseau informatique, l'entreprise doit d'abord définir clairement ses besoins. Il est conseillé de faire appel à un prestataire qui rédigera un cahier des charges complet et adapté à la structure. De nombreux éléments seront retenus lors de l'élaboration du cahier des charges comme la superficie totale à équiper et ses spécificités techniques et le nombre de serveurs et de postes nécessaires au bon fonctionnement de l'entreprise.

Installation des câbles

Le câblage consiste à mettre en place un réseau de câbles qui va permettre de relier les serveurs aux ordinateurs. Deux options s'offrent aux entreprises : - Il est possible de cacher les câbles dans les murs si le bâtiment est suffisamment récent.

- Dans le cas où le bâtiment est ancien, l'entreprise peut opter pour des planchers techniques, des faux-plafonds ou des bornes wifi.

Installation des serveurs

L'entreprise doit désormais installer des serveurs si elle n'en a pas et s'assurer qu'elle dispose de suffisamment de serveurs pour absorber ses besoins. Il est également nécessaire de vérifier la connectique : il faut compter un minimum de 2 ports par serveur. Les serveurs peuvent

fonctionner sous Windows ou Linux, l'entreprise devra donc choisir le système d'exploitation qui lui convient le mieux.

Installation des commutateurs réseaux

La liaison entre les serveurs et les équipements informatiques d'une entreprise (ordinateurs, imprimantes, etc.) se fait par le biais de commutateurs réseaux. Le nombre de commutateurs nécessaires dépendra de votre installation, du nombre de serveurs et d'équipements informatiques dans les locaux.

2 Problématique

Avec l'augmentation du nombre de véhicules, les stations de lavage deviennent de plus en plus importantes. Toutefois, il est compliqué pour les utilisateurs de localiser les lavages disponibles et de connaître la durée d'attente. En même temps, les stations de lavage font face à des difficultés pour assurer une gestion précise des véhicules et fournir des estimations du temps de lavage en raison de l'augmentation de la demande.

Comment peut-on faciliter la localisation des stations de lavage disponibles pour les utilisateurs, estimer précisément les temps d'attente et gérer l'affluence pour les responsables de stations de lavage pour résoudre cette problématique?

3 Analyse et critique de l'existant

La réalisation de tout projet doit être précédée par une étude précise de l'existant. De ce fait cette étude permettra de citer les solutions existantes et de dégager leurs limites sur lesquelles on se base pour proposer les solutions adoptées.

3.1 Etude de l'existant

Après une recherche approfondie que nous avons fait, nous avons trouver des applications qui peuvent nous aider pour réaliser notre projet.

Parmi ces solutions nous identifions :

ween.tn

Ween.tn est un site tunisien spécialisé dans l'annuaire. Il permet aux utilisateurs de chercher et de trouver des emplacements et des bons plans dans toute la Tunisie. Les stations de lavage font partie de ces secteurs.

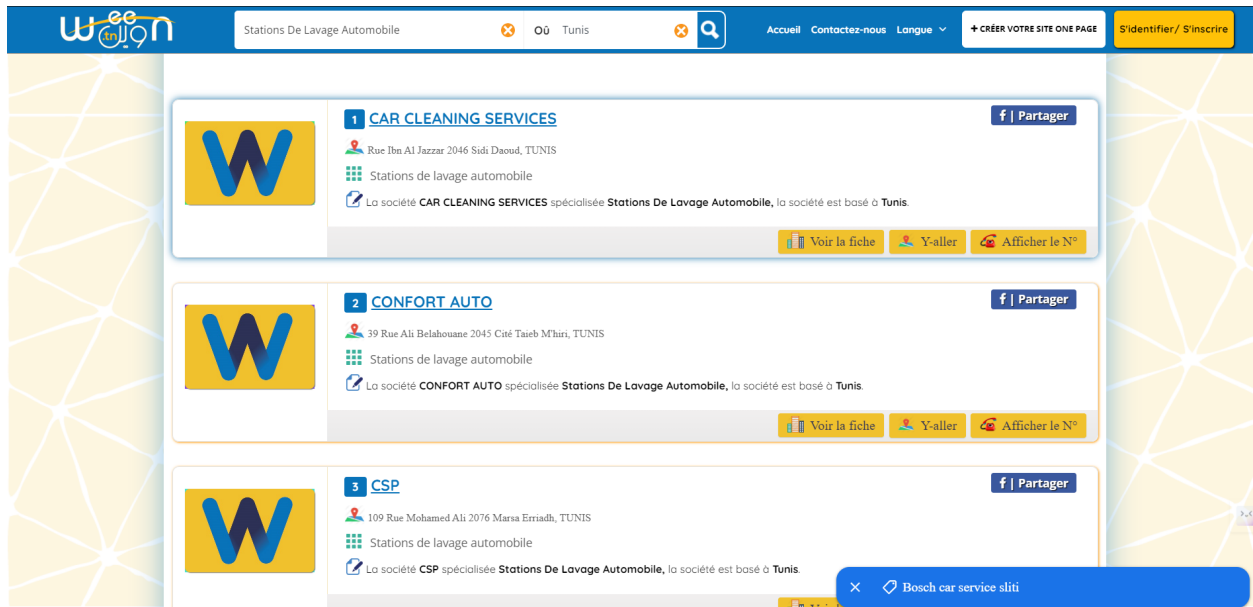


FIGURE 1.3 : Interface de ween.tn

TABLEAU 1.1 : Les points faibles et forts du ween.tn

Les points forts	Les points faibles
-Large couverture géographique. -Interface conviviale	-Nécessité de mise à jour régulière des informations. -Module de gestion des scéances est très limité. -Absence de fonctionnalité de réservation .

3.2 Solution envisagée

Après avoir analysé les limitations des solutions existantes et identifié leurs lacunes, notre solution consiste à développer une application moderne et complète dédiée à la gestion de stations de lavage, nommée "SpotWash". Cette application sera conçue pour offrir une expérience utilisateur intuitive avec les fonctionnalités suivantes :

- **Interfaces :** L'application 'SpotWash' propose des interfaces utilisateur modernes et conviviales, ce qui rend la navigation et l'utilisation pour les utilisateurs finaux plus faciles.

- **Sécurité** : L'application 'SpotWash' garantie la confidentialité et la protection des données personnelles des utilisateurs.
- **Localisation des Stations de Lavage** : Les utilisateurs ont la possibilité de trouver facilement les stations de lavage à proximité.
- **Choisir une date** : Réserver à l'avance des dates pour laver.
- **Gestion des réservations** : Il est possible pour les utilisateurs de consulter l'état de leurs réservations directement depuis l'application.
- **Estimation du temps d'attente** : En utilisant l'application, on peut calculer en temps réel le temps d'attente pour chaque station de lavage, en prenant en considération la charge actuelle et les détails du lavage choisi.
- **Notifications instantanées** : Des notifications sont envoyées aux utilisateurs pour les informer du temps estimé restant avant la fin du lavage.

4 Choix méthodologique

Le succès d'un projet dépend de son adéquation au processus de développement. Le choix d'une méthodologie constitue une étape décisive pour l'élaboration d'une application indépendante de toute plateforme d'exécution et de tout langage de programmation.

4.1 Méthodologie de développement adoptée

Le choix de la méthodologie doit être précis pour bien encadrer notre projet. En effet, pour notre cas, nous avons adopté le Framework Scrum.

«Scrum est un Framework, utilisé depuis le début des années 90, au sein duquel les acteurs peuvent aborder des problèmes complexes et adaptatifs, en livrant de manière efficace et créative des produits de la plus grande valeur possible.

Scrum n'est pas un processus, une technique ou une méthode définitive. C'est plutôt un Framework dans lequel nous pouvons utiliser divers processus et techniques. Scrum montre clairement l'efficacité relative de la gestion de produit et des techniques de travail afin de pouvoir améliorer en permanence le produit, l'équipe et l'environnement de travail».

«La structure Scrum comprend les équipes Scrum et leurs rôles, les événements, les ar-

tefacts et les règles associés. Chaque composant du cadre sert un objectif spécifique et est essentiel au succès et à l'utilisation de Scrum».[1]



FIGURE 1.4 : Principe du travail avec le Framework Scrum

Comme le montre la figure, l'équipe Scrum se compose de :

- Product Owner (PO) : négocie avec les stakeholders et traduit les besoins dans le backlog produit
- Un Scrum master : assure le bon déroulement de la méthodologie de SCRUM et gère la coordination entre le product owner et l'équipe de développement.
- Une équipe de développement : constituée de 3 à 5 personnes qui réalisent le produit

Le sprint backlog est l'ensemble des éléments du Backlog de produit sélectionné pour le Sprint, plus un plan pour fournir l'incrément de produit et réaliser l'objectif de Sprint.

La Revue de sprint est un évènement organisé à la fin du sprint pour inspecter l'incrément et adapter le backlog de produit si nécessaire. Pendant la revue de Sprint, l'équipe Scrum et les parties prenantes échangent sur ce qui a été fait durant le Sprint et le comparent par la définition de « fini ».

La rétrospective du sprint est le dernier évènement du sprint qui donne l'occasion pour l'équipe Scrum de s'auto-inspecter et de créer un plan d'améliorations à mettre en œuvre lors du prochain sprint.

4.2 Justification du choix

Il a été choisi simplement parce que Scrum est léger et facile à comprendre, ce qui nous permet de démarrer Agile rapidement. Cette règle permet de faire face à la complexité.

Sur la base d'une approche incrémentale itérative, nous pouvons rapidement créer des incréments de produit fonctionnel et de haute qualité, mettre en évidence des objectifs clairs à chaque itération et obtenir une meilleure image de l'avancement des travaux.

De plus, il assure une :

- Meilleure qualité de communication : l'utilisateur a la possibilité de clarifier ses exigences au fur et à mesure.
- Meilleure visibilité : le client a une meilleure visibilité sur l'avancement des travaux. Il est juste présent à travers les réunions périodiques.
- Meilleur contrôle de la qualité : les tests sont effectués continuellement.
- Réduction des risques : les risques sont détectés à l'avance.
- Meilleur Contrôle des coûts : le projet peut être arrêté à défaut de budget.

5 Langage de modélisation



FIGURE 1.5 : Logo UML

Le langage UML, nous garantit une architecture d'information structurée et nous permet de présenter nos diagrammes de façon simple et compréhensible.

6 Environnement de développement

6.1 Environnement matériel

Pour la réalisation de ce travail nous avons utilisé un ordinateur portable ayant les caractéristiques suivantes :

TABEAU 1.2 : Environnement matériel PC

Caractéristiques	Description
Marque	PC PORTABLE Lenevo
Processeur	Intel Core i7 12 éme génération
Disque dur	500 GO ssd
RAM	24 GO
Carte Graphique	Intel(R) Graphics 620
Système d'exploitation	Windows 11 (64 bits)

6.2 Environnement logiciel

Au cours de cette section, nous présentons les outils et logiciels utilisés tout au long de notre projet.

IntelliJ IDEA

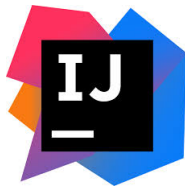


FIGURE 1.6 : Logo IntelliJ IDEA

IntelliJ IDEA, un environnement de développement intégré (IDE) pour Java et Kotlin, a été développé dans le but d'optimiser la productivité des développeurs. C'est une option idéale pour un projet Spring Boot.[2]

Visual Studio Code

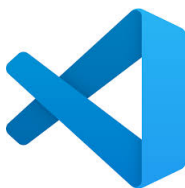


FIGURE 1.7 : Logo Visual Studio Code

Visual Studio Code est un outil de développement populaire pour le frontend, offrant des fonctionnalités redéfinies et optimisées dans la création et le débogage d'applications web.[3]

WampServer



FIGURE 1.8 : Logo WampServer

WampServer est une plateforme de développement Web, permettant de faire fonctionner localement des scripts PHP. [4]

Overleaf



FIGURE 1.9 : Logo Overleaf

Overleaf est un éditeur LaTeX en ligne, collaboratif en temps réel.

Draw.io



FIGURE 1.10 : Logo Draw.io

Diagrams.net (draw.io) est une solution en ligne gratuite pour créer des diagrammes et des organigrammes.[5]

Git et GitHub



FIGURE 1.11 : Logo Git et GitHub

Git est un logiciel de gestion de version qui nous permet d'enregistrer les différentes modifications apportées à un projet et de pouvoir revenir à une version antérieure du projet.

GitHub est une plateforme qui facilite l'hébergement de projets et favorise une collaboration efficace au sein des équipes de développement.[6]

Trello



FIGURE 1.12 : Logo Trello

Trello est un outil de collaboration qui organise les projets en tableaux. [7]

Postman



FIGURE 1.13 : Logo Postman

Postman est un outil gratuit permettant de tester des requêtes HTTP.

7 Choix technique

HTML5, CSS3, JavaScript :



FIGURE 1.14 : Logo HTML5, CSS3, JavaScript

Utilisés pour la création et le design des pages web, améliorant l'expérience utilisateur.

Angular



FIGURE 1.15 : Logo Angular

Angular, un framework JavaScript open-source écrit en TypeScript, est géré par Google. Son objectif principal est de développer des applications à page unique.[8]

Spring boot



FIGURE 1.16 : Logo Spring boot

Spring Boot est une collection d'outils qui étend le cadre Spring et facilite la création rapide d'applications. ce qui en fait un choix idéal pour les applications backend robustes et évolutives.[9]

My sql



FIGURE 1.17 : Logo My SQL

MySQL, Le système de gestion de base de données relationnelle open-source est souvent employé en raison de ses performances et de sa souplesse.[10]

API REST

Dans le cadre de notre projet, nous avons élaboré une API RESTful solide en utilisant Spring Boot pour offrir des services backend essentiels à notre application. Cette section expose en

détail l'architecture, les fonctionnalités clés et les choix techniques liés à notre service API RESTful.[11]

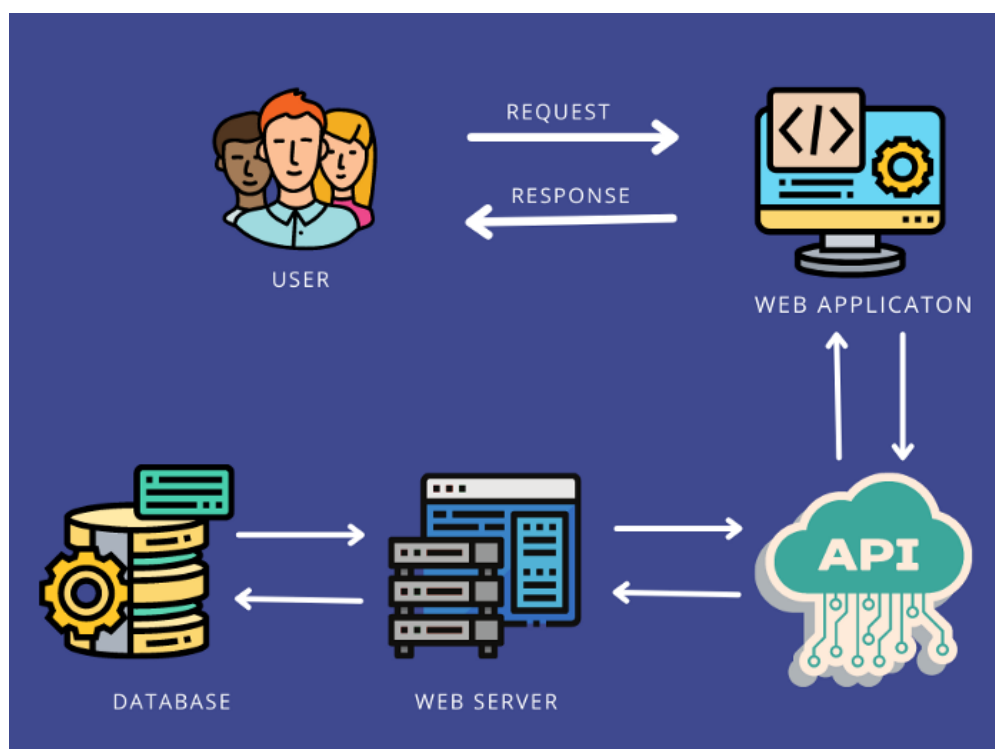


FIGURE 1.18 : Architecture de notre API REST

Principes de l'Architecture REST

Les API REST respectent des principes fondamentaux qui les rendent adaptées à notre projet :

- Ressources : Les URI sont utilisées pour identifier les données en tant que ressources.
- Opérations : les méthodes HTTP sont (GET, POST, PUT, DELETE).
- Sans état : Toutes les informations nécessaires sont incluses dans chaque requête HTTP afin de saisir la demande sans avoir besoin de contexte stocké du côté du serveur.
- Représentation : Les données sont généralement échangées sous forme de représentations JSON ou XML. [11]

8 Architecture

8.1 Architecture logique

L'architecture logique (ou logicielle) décrit les composants abstraits (souvent les services et leurs clients) et les interactions entre eux.

Architecture MVVM de angular

Pour la partie frontend de l'application, nous avons mis en place l'architecture MVVM d'angular.[12]

Principes de l'Architecture MVVM

L'architecture MVVM se compose des éléments suivants :

- Modèle : Contient les données de l'application et la logique métier.
- Vue : Représente l'interface utilisateur (UI) avec laquelle l'utilisateur interagit.
- VueModèle : Expose des méthodes, des commandes et des propriétés pour maintenir l'état de la vue. [12]

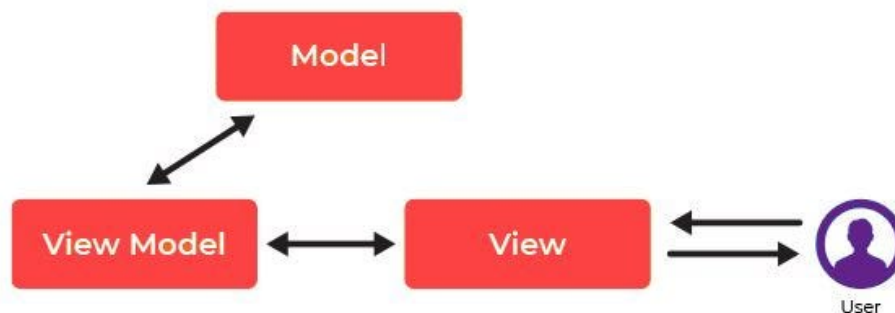


FIGURE 1.19 : Architecture MVVM dans notre application

Avantages de l'Architecture MVVM dans notre Projet

L'architecture MVVM a apporté plusieurs avantages :

- Séparation claire des responsabilités entre la Vue, le VueModèle et le Modèle.
- Réutilisabilité accrue des composants.
- Gestion efficace de l'état de l'interface utilisateur.

Architecture MVC de Spring Boot

En parallèle, pour la partie backend de notre application, nous avons utilisé l'architecture MVC de Spring Boot.[13]

Principes de l'Architecture MVC

L'architecture MVC se compose des éléments suivants :

- Modèle : Contient les données de l'application et la logique métier.
- Vue : Représente l'interface utilisateur (UI) avec laquelle l'utilisateur interagit.
- Contrôleur : contrôle la logique pour gérer les entrées utilisateur. [13]

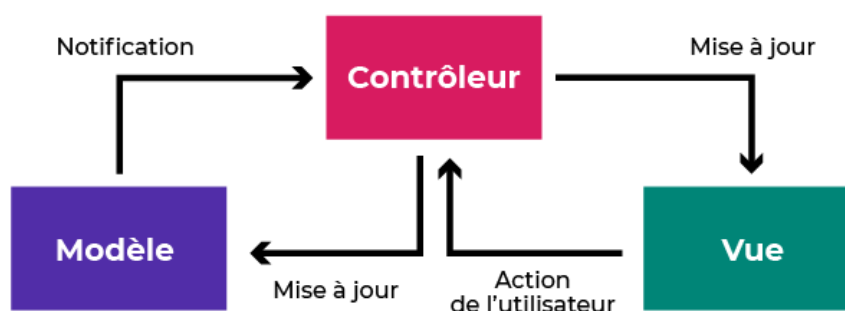


FIGURE 1.20 : Architecture MVC dans notre application Spring Boot

Avantages de l'Architecture MVC de Spring Boot

L'utilisation de l'architecture MVC dans notre projet Spring Boot offre plusieurs avantages :

- Organisation claire du code en séparant les préoccupations de la gestion des données, des opérations métier et de l'affichage.
- Facilité de maintenance et de test.
- Flexibilité dans la gestion des vues permettant d'adapter facilement l'application aux besoins des clients et aux changements dans l'interface utilisateur.

8.2 Architecture de l'application

L'architecture de l'application est représentée par le diagramme ci-dessous. Elle montre l'interaction entre les différentes technologies utilisées pour fournir des services aux utilisateurs , aux gérants et aux administrateurs.

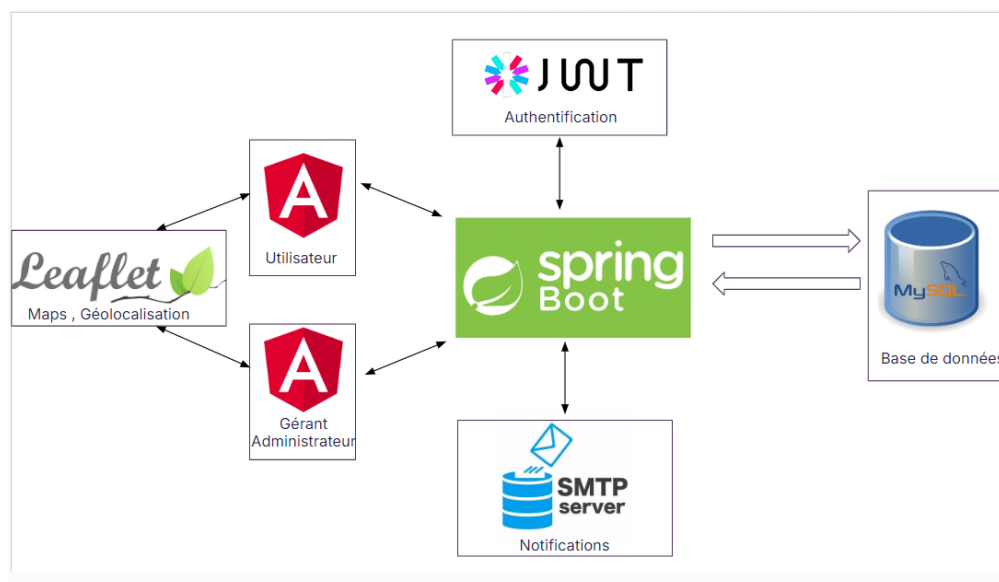


FIGURE 1.21 : Architecture de l'application SpotWash

8.2.1 Description des composants

- **Leaflet (Maps, Géolocalisation)** : Leaflet est une bibliothèque JavaScript qui permet de développer des cartes interactives. Elle est adoptée pour présenter à l'utilisateur des cartes, leur géolocalisation, leurs déplacement. [14]
- **angular (Utilisateur , Gérant et administrateur)** : Angular constitue un intégrateur web qui permet de créer des applications web dynamiques. Ces dernières sont au nombre de deux, l'une utilisée par les clients finaux et l'autre par les gérants et les administrateurs.[8]
- **Spring Boot** : Spring est un framework Java que nous avons utilisé pour développer notre application back-end. Il contrôle la base de données, gère la communication avec les utilisateurs et les gestionnaires, en plus de la logique métier API REST et l'authentification.[9]
- **JWT (JSON Web Token)** : JWT est un standard ouvert qui permet la création de jetons délivrés par le serveur de ressources pour l'utilisateur qui pourra vérifier son adresse de manière sécurisée.[15]
- **SMTP Server (Notifications)** : SMTP est utilisé pour l'envoi des e-mails. Dans notre cas, on l'a programmé pour qu'il s'occupe des notifications que nous envoyons aux utilisateurs.[16]
- **MySQL (Base de données)** : C'est un système de gestion de base de données qui contient toutes les données de l'application.[10]

Conclusion

Dans ce chapitre, nous avons brièvement présenté l'organisme d'accueil, la problématique, l'étude de l'existant, choix méthodologique, l'environnement de développement ainsi que l'architecture. Le prochain chapitre sera consacré à la planification du projet.

CHAPITRE 2

PLANIFICATION ET CAPTURE DES BESOINS

1	Capture des Besoins	21
2	Pilotage avec SCRUM	25
3	Backlog Produit	25
4	Définition de « Fini »	29

Introduction

Dans ce chapitre, nous clarifierons et présenterons les besoins de l'application en identifiant les acteurs, les besoins fonctionnels et les besoins non fonctionnels. Par la suite nous énoncerons les éléments du Backlog du produit et la planification de nos releases.

1 Capture des Besoins

1.1 Identification des Acteurs

Les acteurs dans l'application de gestion des stations de lavage sont :

- **Visiteur** : Utilisateur consultant la page d'accueil pour trouver une station de lavage et ayant la possibilité de s'inscrire.
- **Utilisateur** : Utilisateur final cherchant à trouver et réserver une station de lavage.
- **Gérant** : Propriétaire ou responsable de station de lavage, en charge de la gestion de la disponibilité et des informations.
- **administrateur** : Responsable de la consultation des comptes, des stations de lavage et des sessions associées.
- **Planificateur des tâches** : Responsable de la gestion des alertes et des rappels.

1.2 Besoins Fonctionnels

1. **S'inscrire** : Permet aux visiteurs la possibilité de créer un compte.
2. **Authentification** : Permet aux utilisateurs, gérants et administrateurs de se connecter à l'application en utilisant leurs identifiants sécurisés.
3. **Gestion des Profils** : Permet aux utilisateurs, gérants et administrateurs de modifier des informations personnelles.
4. **Gestion des Voitures** :
 - Permet aux utilisateurs d'ajouter, modifier et gérer les informations de leurs voitures.
 - Facilite les réservations en permettant de sélectionner une voiture déjà enregistré.

5. Recherche de Stations de Lavage :

- Permet aux visiteurs de rechercher des stations de lavage à proximité en utilisant la géolocalisation.
- Affiche les détails des stations comme le nom, l'adresse et les services offerts.

6. Réservation de Sessions :

- Permet aux utilisateurs de réserver une session de lavage pour leur voiture.
- Propose une estimation du temps de lavage en fonction de la charge actuelle des stations.

7. Disponibilité en Temps Réel :

- Fournit aux utilisateurs des informations en temps réel sur la disponibilité des stations de lavage.
- affiche le nombre de places disponibles et les temps d'attente estimés.

8. Gestion de l'affluence :

- Permet aux gérants de stations de mettre à jour en temps réel l'affluence des stations de lavage.
- Affiche le nombre de véhicules en cours de lavage pour aider les utilisateurs à choisir une station moins fréquentée.

9. Notifications :

- Envoie des notifications pour informer les utilisateurs des mises à jour importantes, telles que l'approche de leur tour de lavage ou la disponibilité de nouvelles places.
- Comprend des rappels de réservation et des alertes en cas de changements significatifs.

10. Statistiques :

- Fournit aux gérants des statistiques détaillées sur l'utilisation des stations de lavage.
- Nombre de sessions pour chaque station et chaque semaine.
- Identification de la station la plus fréquentée.

11. Supervision des Utilisateurs et des Gérants :

- Fournit aux administrateurs des informations détaillées sur les utilisateurs et les gérants.
- Permet aux administrateurs de désactiver ou activer les comptes des utilisateurs et des gérants.
- Affiche les stations gérées par chaque gérant.
- Affiche les sessions de lavage de chaque station.

1.3 Besoins Non Fonctionnels

- **Performance** : Les utilisateurs s'attendent à obtenir des informations en temps réel sur la disponibilité des stations et les temps d'attente sans délais perceptibles. Une bonne performance est donc nécessaire pour maintenir leur satisfaction et leur fidélité.
- **Sécurité** : La sécurité des données est primordiale pour gagner la confiance des utilisateurs et des gérants. En garantissant la protection des informations personnelles et sensibles, l'application répond aux exigences de conformité et prévient les risques de sécurité.
- **Convivialité** : Une interface utilisateur bien conçue améliore l'expérience utilisateur, rendant l'application plus accessible et agréable à utiliser. Cela peut également réduire la courbe d'apprentissage pour les nouveaux utilisateurs.
- **Scalabilité** : À mesure que l'application gagne en popularité et que de nouvelles stations de lavage sont ajoutées, le système doit être capable de gérer cette croissance sans dégradation des performances.
- **Fiabilité** : Une application fiable assure que les utilisateurs et les gérants peuvent accéder aux fonctionnalités à tout moment, augmentant ainsi la crédibilité et la confiance dans le service offert.

1.4 Diagramme de cas d'utilisation global



FIGURE 2.1 : Diagramme de cas d'utilisation global

2 Pilotage avec SCRUM

2.1 Équipe et rôles

- **Product Owner** : Anis Jaouadi, Responsable de la vision du produit et de la priorisation des fonctionnalités à développer.
- **Scrum Master** : Anis Jaouadi, Facilite les cérémonies SCRUM et assure que l'équipe suit les principes SCRUM.
- **Équipe de Développement** : Roua Ben Amna. Responsable de la conception, du développement et des tests des fonctionnalités.

2.2 Plan de sprints

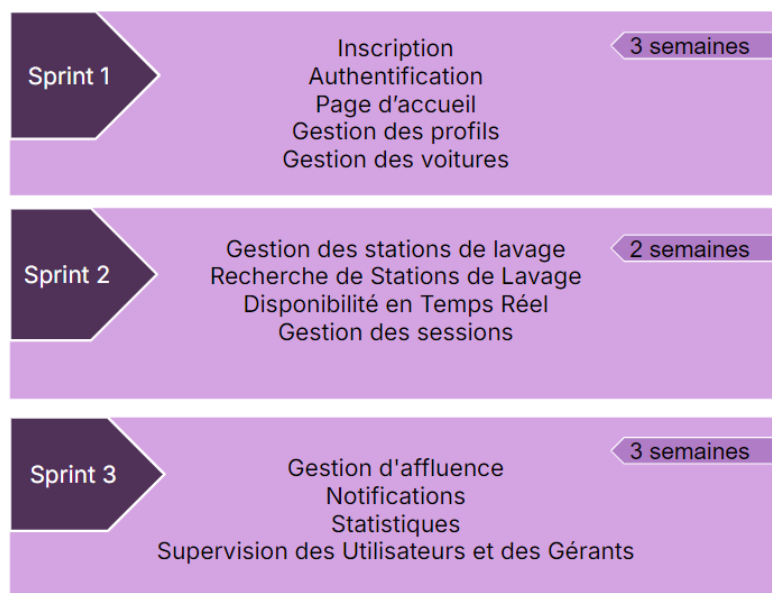


FIGURE 2.2 : Plan de sprints

3 Backlog Produit

Le Backlog Produit est une liste ordonnée de tous les éléments identifiés comme nécessaires au produit. Il constitue l'unique source d'exigences pour tout changement à apporter au produit.

TABEAU 2.1 : Backlog produit

N	User Story	Priorité
1	En tant que visiteur, je veux m'inscrire	M
2	En tant que visiteur, je veux vérifier mon email.	M
3	En tant qu'utilisateur, je veux réinitialiser mon mot de passe oublié en cliquant sur le lien "Mot de passe oublié?" via l'interface d'authentification	M
4	En tant que visiteur, je veux visiter la page d'accueil.	M
5	En tant que visiteur, je veux rechercher des stations de lavage à proximité.	M
6	En tant qu'utilisateur, je veux m'authentifier.	M
7	En tant qu'utilisateur, je veux consulter la liste de mes voitures.	S
8	En tant qu'utilisateur, je veux rechercher une voiture.	S
9	En tant qu'utilisateur, je veux ajouter une voiture.	S
10	En tant qu'utilisateur, je veux modifier une voiture.	S
11	En tant qu'utilisateur, je veux supprimer une voiture.	S
12	En tant qu'utilisateur, je veux consulter mon compte.	S
13	En tant qu'utilisateur, je veux modifier mon compte.	S
14	En tant qu'utilisateur, je veux consulter la liste de mes sessions.	S
15	En tant qu'utilisateur, je veux ajouter une session de lavage pour mon voiture.	M
16	En tant qu'utilisateur, je veux voir le nombre de places disponibles.	M
17	En tant qu'utilisateur, je veux voir les temps d'attente estimés.	M
18	En tant qu'utilisateur, je veux voir les informations asynchrones.	M
19	En tant qu'utilisateur, je veux modifier une session.	M
20	En tant qu'utilisateur, je veux annuler une session.	S

21	En tant qu'utilisateur, je veux être notifié lorsque mon tour de lavage approche, afin de pouvoir préparer mon véhicule à temps.	C
22	En tant qu'utilisateur, je veux être notifié de la disponibilité de nouvelles places.	C
23	En tant que gérant, je veux m'inscrire.	M
24	En tant que gérant, je veux vérifier mon email.	M
25	En tant que gérant, je veux m'authentifier.	M
26	En tant que gérant, je veux réinitialiser mon mot de passe.	M
27	En tant que gérant, je veux consulter la liste des stations de lavage.	S
28	En tant que gérant, je veux ajouter une station de lavage.	S
29	En tant que gérant, je veux modifier une station de lavage.	S
30	En tant que gérant, je veux supprimer une station de lavage.	S
31	En tant que gérant, je veux consulter la liste des sessions d'une station de lavage.	S
32	En tant que gérant, je veux filtrer la liste des sessions par date.	S
33	En tant que gérant, je veux filtrer la liste des sessions par statut.	S
34	En tant que gérant, je veux filtrer la liste des sessions par type de lavage.	S
35	En tant que gérant, je veux mettre à jour en temps réel le statut des sessions.	S
36	En tant que gérant, je veux consulter mon compte.	S
37	En tant que gérant, je veux modifier mon compte.	S
38	En tant que gérant, je veux voir des statistiques détaillées sur l'utilisation des stations de lavage.	C

39	En tant que gérant, je veux voir le nombre de sessions pour chaque station et chaque semaine.	C
40	En tant que gérant, je veux identifier la station la plus fréquentée.	C
41	En tant qu'administrateur, je veux m'authentifier.	M
42	En tant qu'administrateur, je veux réinitialiser mon mot de passe.	M
43	En tant qu'administrateur, je veux consulter mon compte.	S
44	En tant qu'administrateur, je veux modifier mon compte.	S
45	En tant qu'administrateur, je veux consulter la liste des utilisateurs et des gérants.	M
46	En tant qu'administrateur, je veux désactiver ou activer les comptes des utilisateurs et des gérants.	M
47	En tant qu'administrateur, je veux consulter la liste des stations de lavage gérées par chaque gérant.	M
48	En tant qu'administrateur, je veux consulter la liste des sessions de chaque gérant.	M

- Must have (doit) : Hautement prioritaires ; s'ils ne sont pas atteints, le projet sera considéré comme un échec.
- Should have (devrait) : Il s'agit de points importants mais le client a marqué son accord sur une éventuelle livraison différée.
- Could have (pourrait) : Regroupe des éléments planifiés dans le projet mais qui peuvent ne pas être réalisés en cas de retard ou de manque de ressources.
- Won't have (n'aura pas) : Des points souhaités mais qui sont écartés du projet en cours.

[17]

4 Définition de « Fini »

La définition de « Fini » (ou la Definition of Done «DOD» en anglais) est l'ensemble de critères définis par l'équipe de développement déterminant si une User Story peut se considérer comme traitée.

Le but de la DOD est de définir les critères qui font qu'une User Story a été totalement traitée. Après plusieurs recherches et point avec l'ensemble des équipes dans l'organisme on a choisi de commencer à travailler avec la Définition de « Fini » ci-dessous et l'améliorer à chaque rétrospective de sprint si on détecte des anomalies :

- Code révisé (Code reviewed)
- Critères d'acceptation remplis (Acceptance criteria met)
- Tests fonctionnels réussis (Non-Functional requirements met)[18]

Conclusion

Dans ce chapitre, nous avons présenté la Capture des besoins du projet, Pilotage avec SCRUM, le Backlog Produit ainsi que la définition de « Fini ».

Le prochain chapitre sera consacré à l'étude et réalisation du sprint 1.

CHAPITRE 3

ÉTUDE ET RÉALISATION DU SPRINT 1

1	Réunion de planification du sprint (Sprint planning meeting)	31
2	Description fonctionnelle des histoires du Sprint1	42
3	Description comportementale des histoires du sprint 1	49
4	Description Structurelle	52
5	Tests Postman	53
6	La revue	56
7	Rétrospective	59

Introduction

Dans ce chapitre, nous allons clarifier et présenter les besoins de l'application en identifiant les éléments du sprint 1. Nous aborderons les objectifs du sprint, le backlog du sprint, ainsi que les étapes de conception et de réalisation. Cela nous permettra de comprendre les différentes phases de développement et d'assurer une exécution réussie du projet.

1 Réunion de planification du sprint (Sprint planning meeting)

1.1 Objectif du sprint (Sprint goal)

Au cours du Sprint 1, nous avons développé les fonctionnalités suivantes : inscription, authentification, gestion des profils et gestion des voitures.

1.2 Backlog du sprint

Le Backlog du sprint est l'ensemble des éléments sélectionnés, pour le Sprint plus un plan pour livrer l'incrément du produit et réaliser l'objectif du Sprint.

TABLEAU 3.1 : Backlog du sprint - Sprint 1

Fonctionnalité	N°	User Story	Tâches Associées
Inscription	1	En tant que visiteur, je veux m'inscrire	• Création de la page d'inscription (angular) • Implémentation de l'aPI pour l'inscription (Spring) • Intégration de la logique d'inscription (angular) • Test de fonctionnalité avec Postman
	2	En tant que visiteur, je veux vérifier mon email.	• Création de la page de vérification d'email (angular) • Implémentation de l'API pour la vérification d'email (Spring) • Intégration de la logique de vérification d'email (angular) • Test de fonctionnalité avec Postman

	3	En tant que gérant, je veux m'inscrire.	• Création de la page d'inscription pour les gérants (angular) • Intégration de la logique d'inscription pour les gérants (angular) • Test de fonctionnalité avec Postman
	4	En tant que gérant, je veux vérifier mon email.	• Création de la page de vérification d'email pour les gérants (angular) • Intégration de la logique de vérification d'email pour les gérants (angular) • Test de fonctionnalité avec Postman

authentification	5	En tant qu'utilisateur, je veux m'authentifier.	• Création de la page d'authentification (angular) • Implémentation de l'API d'authentification (Spring) • Intégration de la logique d'authentification (angular) • Test de fonctionnalité avec Postman
	6	En tant qu'utilisateur, je veux réinitialiser mon mot de passe oublié en cliquant sur le lien «Mot de passe oublié?» via l'interface d'authentification.	• Création de la page de réinitialisation de mot de passe (angular) • Implémentation de l'API de réinitialisation de mot de passe (Spring) • Intégration de la logique de réinitialisation de mot de passe (angular) • Test de fonctionnalité avec Postman

	7	En tant que gérant, je veux m'authentifier.	• Création de la page d'authentification pour les gérants (angular) • Intégration de la logique d'authentification pour les gérants (angular) • Test de fonctionnalité avec Postman
	8	En tant que gérant, je veux réinitialiser mon mot de passe.	• Création de la page de réinitialisation de mot de passe pour les gérants (angular) • Intégration de la logique de réinitialisation de mot de passe pour les gérants (angular) • Test de fonctionnalité avec Postman

	9	En tant qu'administrateur, je veux m'authentifier.	• Intégration de la logique d'authentification pour les administrateurs (angular) • Test de fonctionnalité avec Postman
	10	En tant qu'administrateur, je veux réinitialiser mon mot de passe.	• Intégration de la logique de réinitialisation de mot de passe pour les administrateurs (angular) • Test de fonctionnalité avec Postman

Gestion des profils	11	En tant qu'utilisateur, je veux consulter mon compte.	• Création de la page de consultation de compte (angular) • Implémentation de l'API de consultation de compte (Spring) • Intégration de la logique de consultation de compte (angular) • Test de fonctionnalité avec Postman
	12	En tant qu'utilisateur, je veux modifier mon compte.	• Création de la page de modification de compte (angular) • Implémentation de l'API de modification de compte (Spring) • Intégration de la logique de modification de compte (angular) • Test de fonctionnalité avec Postman

	13	En tant que gérant, je veux consulter mon compte.	• Création de la page de consultation de compte pour les gérants (angular) • Intégration de la logique de consultation de compte pour les gérants (angular) • Test de fonctionnalité avec Postman
	14	En tant que gérant, je veux modifier mon compte.	• Création de la page de modification de compte pour les gérants (angular) • Intégration de la logique de modification de compte pour les gérants (angular) • Test de fonctionnalité avec Postman

	15	En tant qu'administrateur, je veux consulter mon compte.	• Intégration de la logique de consultation de compte pour les administrateurs (angular) • Test de fonctionnalité avec Postman
	16	En tant qu'administrateur, je veux modifier mon compte.	• Intégration de la logique de modification de compte pour les administrateurs (angular) • Test de fonctionnalité avec Postman
Gestion des voitures	17	En tant qu'utilisateur, je veux consulter la liste de mes voitures	• Création de la page de liste des voitures (angular) • Implémentation de l'API pour récupérer la liste des voitures (Spring) • Intégration de la logique pour récupérer la liste des voitures (angular) • Test de fonctionnalité avec Postman

	18	En tant qu'utilisateur, je veux rechercher une voiture.	• Création de la fonctionnalité de recherche pour les voitures (angular) • Implémentation de l'API pour rechercher des voitures (Spring) • Intégration de la logique pour rechercher des voitures (angular) • Test de fonctionnalité avec Postman
	19	En tant qu'utilisateur, je veux ajouter une voiture.	• Création de la page d'ajout de voiture (angular) • Implémentation de l'API pour ajouter une voiture (Spring) • Intégration de la logique pour ajouter une voiture (angular) • Test de fonctionnalité avec Postman

	20	En tant qu'utilisateur, je veux modifier une voiture.	• Création de la page de modification de voiture (angular) • Implémentation de l'API pour modifier une voiture (Spring) • Intégration de la logique pour modifier une voiture (angular) • Test de fonctionnalité avec Postman
	21	En tant qu'utilisateur, je veux supprimer une voiture.	• Création d'un bouton de suppression de voiture et d'une modal de confirmation (angular) • Implémentation de l'API pour supprimer une voiture (Spring) • Intégration de la logique pour supprimer une voiture (angular) • Test de fonctionnalité avec Postman

Page d'accueil	22	En tant que visiteur, je veux visiter la page d'accueil.	• Création de la page d'accueil (angular) • Intégration de la logique pour la page d'accueil (angular) • Test de fonctionnalité avec Postman
----------------	----	--	--

2 Description fonctionnelle des histoires du Sprint1

2.1 Diagramme de cas d'utilisation général du sprint 1

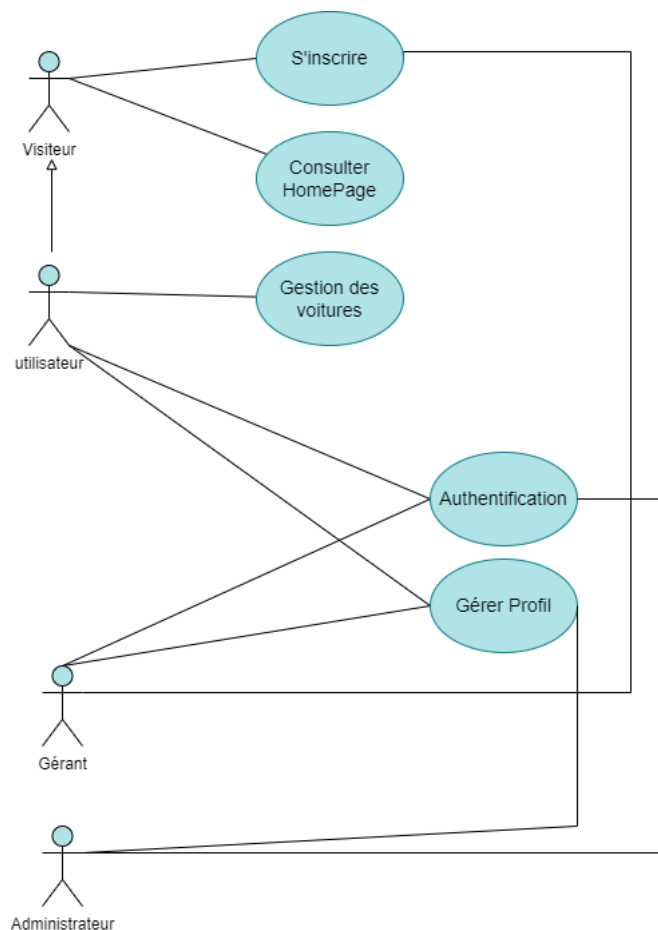


FIGURE 3.1 : Diagramme de cas d'utilisation général du sprint 1

2.2 Diagramme de cas d'utilisation « S'inscrire »

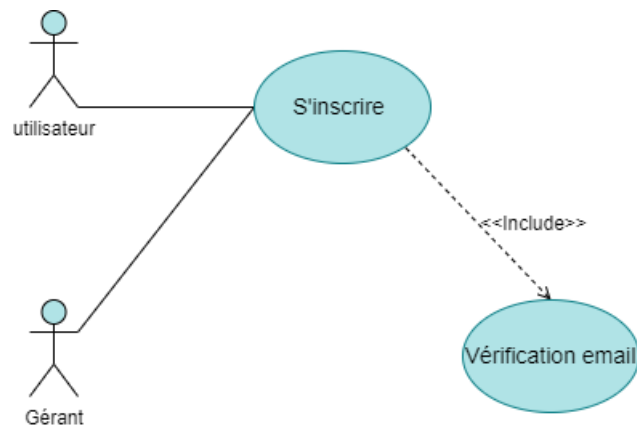


FIGURE 3.2 : Diagramme de cas d'utilisation « S'inscrire »

2.3 Diagramme raffiné du cas d'utilisateur « S'authentifier »

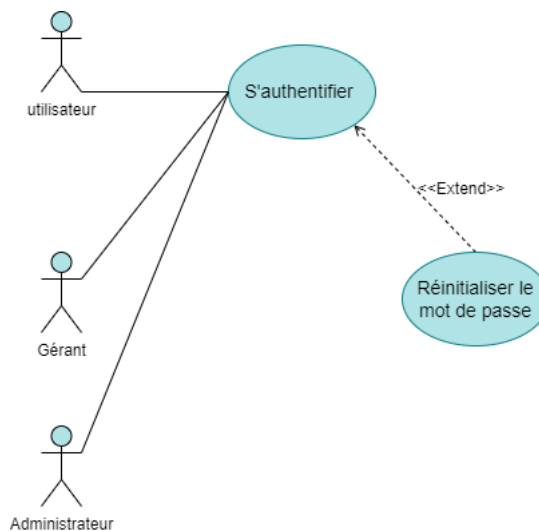


FIGURE 3.3 : Diagramme de cas d'utilisation « S'authentifier »

2.4 Diagramme raffiné de cas d'utilisation « Gestion des voitures »

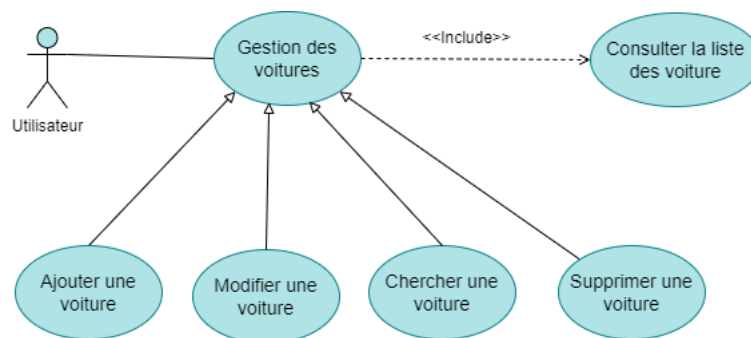


FIGURE 3.4 : Diagramme de cas d'utilisation « Gestion des voitures »

2.5 Description Textuelle de cas d'utilisation « Inscription »

TABLEAU 3.2 : Description Textuelle de cas d'utilisation « Inscription »

Titre	Inscription
Acteurs	Gérant, Utilisateur
Pré-condition	• Les acteurs doivent fournir des informations valides pour l'inscription. • L'utilisateur doit être sur la page d'inscription.
Scénario nominal	1. L'utilisateur accède à la page d'inscription. 2. Il remplit le formulaire avec ses informations. 3. Il soumet le formulaire. 4. Le système valide les informations. 5. Si tout est valide, l'utilisateur et le Gérant sont enregistrés et un e-mail de confirmation est envoyé.
Scénario alternatif	5. A1 - Si l'utilisateur ou le gérant entre des informations invalides, un message d'erreur s'affiche.
Post-condition	L'utilisateur ou le gérant est inscrit avec succès.

2.6 Description Textuelle de cas d'utilisation « Chercher une voiture »**TABLEAU 3.3 :** Description Textuelle de cas d'utilisation « Chercher une voiture »

Titre	Chercher voiture
Acteurs	Utilisateur
Pré-condition	• Utilisateur authentifié. • Liste de ses voitures affichée.
Scénario nominal	1. L'utilisateur lance une recherche rapide à partir des mots-clés. 2. Le système cherche le résultat avec les données saisies. 3. Le système affiche le résultat de recherche.
Scénario alternatif	2. A1 - Pas de résultat.
Post-condition	La voiture trouvée.

2.7 Description Textuelle de cas d'utilisation « Ajouter une voiture »

TABLEAU 3.4 : Description Textuelle de cas d'utilisation « Ajouter une voiture »

Titre	Ajouter voiture
Acteurs	Utilisateur
Pré condition	• Utilisateur authentifié. • Liste de ses voitures affichée.
Scénario nominal	1. L'utilisateur demande l'ajout d'une nouvelle voiture. 2. Le système redirige l'utilisateur vers la page contenant le formulaire d'ajout d'une voiture. 3. L'utilisateur saisit les informations nécessaires pour la nouvelle voiture. 4. L'utilisateur clique sur valider. 5. Le système vérifie les données saisies. 6. Le système redirige l'utilisateur vers la liste des voitures.
Scénario alternatif	5. A1 -le système affiche un message d'erreur si l'un des champs obligatoires est vide et/ou invalide. Le cas d'utilisation redémarre de l'étape 3 du scénario nominal.
Post condition	La voiture ajoutée.

2.8 Description Textuelle de cas d'utilisation « Modifier une voiture »

TABEAU 3.5 : Description Textuelle de cas d'utilisation « Modifier une voiture »

Titre	Modifier voiture
Acteurs	Utilisateur
Pré-condition	• Utilisateur authentifié. • Liste de ses voitures affichée. • La voiture existe.
Scénario nominal	1. L'utilisateur sélectionne la voiture. 2. L'utilisateur clique sur l'icône « Edit ». 3. Le système affiche les informations de la voiture dans un formulaire. 4. L'utilisateur modifie le formulaire. 5. L'utilisateur valide le formulaire. 6. Le système vérifie les informations saisies par l'utilisateur. 7. Le système redirige l'utilisateur vers la liste des voitures.
Scénario alternatif	5. A1 - Le système affiche un message d'erreur si l'un des champs obligatoires est vide et/ou invalide. Le cas d'utilisation redémarre de l'étape 3 du scénario nominal.
Post-condition	La voiture modifiée.

2.9 Description Textuelle de cas d'utilisation « Supprimer une voiture »

TABLEAU 3.6 : Description Textuelle de cas d'utilisation « Supprimer une voiture »

Titre	Supprimer voiture
Acteurs	Utilisateur
Pré-condition	• Utilisateur authentifié. • Liste de ses voitures affichée. • La voiture existe.
Scénario nominal	1. L'utilisateur clique sur l'icône de suppression. 2. Le système affiche une fenêtre de confirmation. 3. L'utilisateur clique sur le bouton « Confirm ». 4. Le système supprime la voiture .
Scénario alternatif	2. A1 - L'opération sera annulée si l'utilisateur clique sur le bouton « Cancel ».
Post-condition	La voiture supprimée.

3 Description comportementale des histoires du sprint 1

3.1 Diagramme de séquence « S'inscrire »

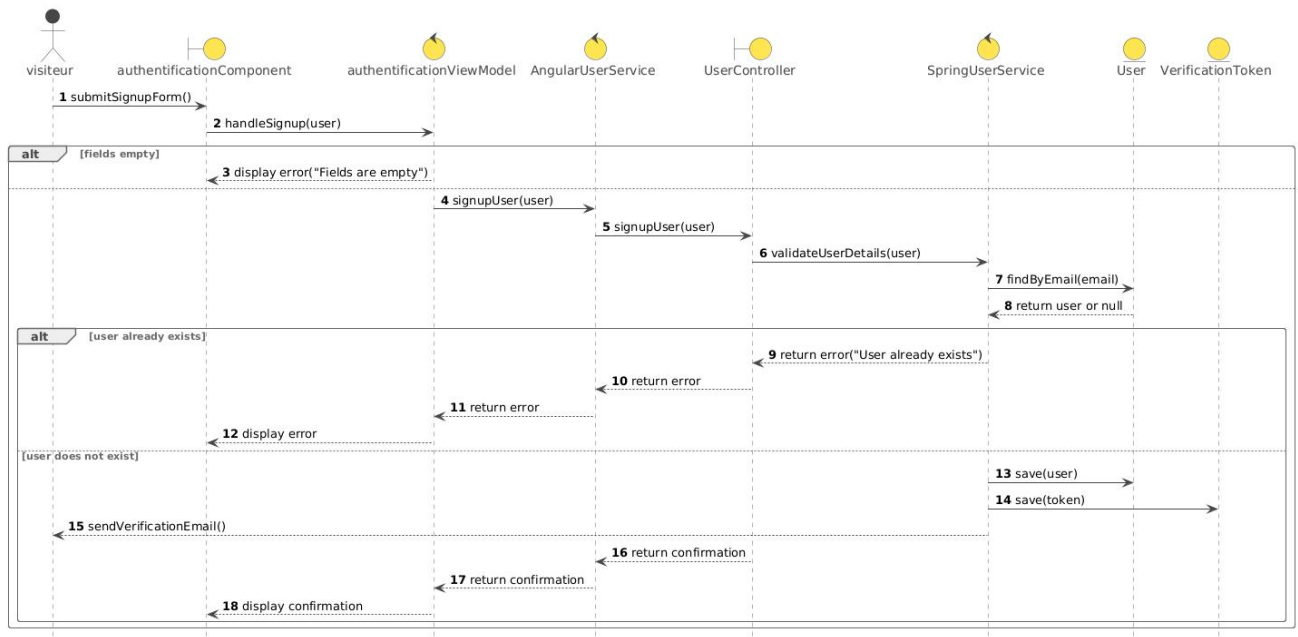


FIGURE 3.5 : Diagramme de séquence «S'inscrire»

Ce diagramme décrit le processus d'inscription d'un utilisateur via l'application. L'utilisateur interagit avec la vue modèle du formulaire d'inscription, puis lorsqu'il le remplit, toutes les données sont vérifiées sur le champ qu'elle n'est pas vide et qu'il n'est pas déjà enregistré dans la BD. Une fois que toutes les conditions sont satisfaites, l'utilisateur est créé en enregistré, l'utilisateur reçoit le courriel.

Le même processus s'applique pour l'inscription d'un gérant.

3.2 Diagramme de séquence « S'authentifier »

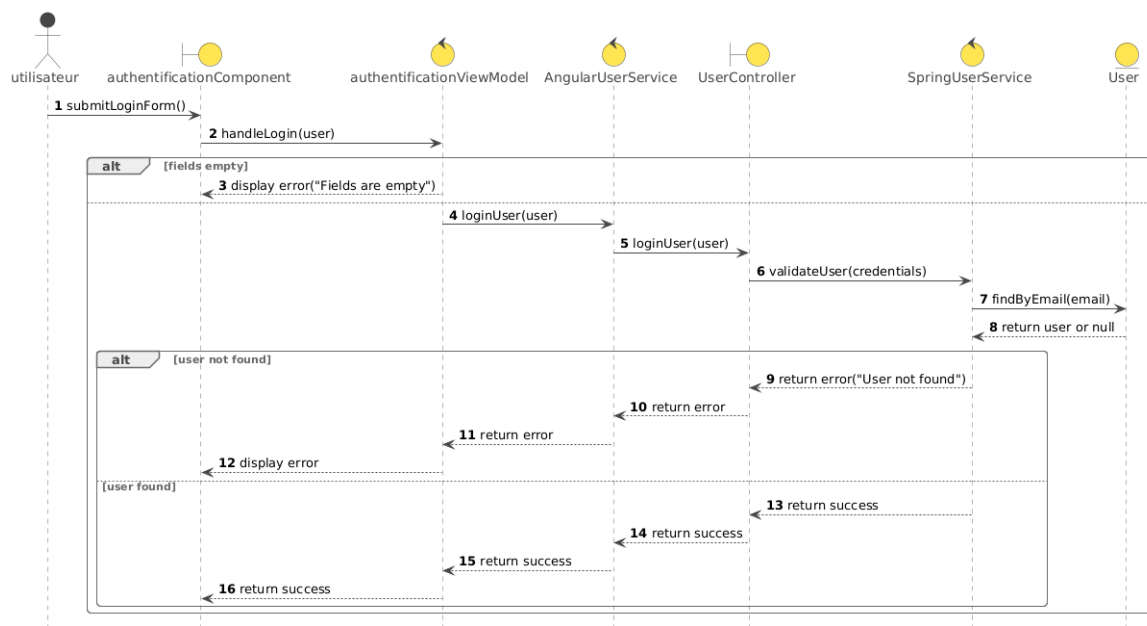


FIGURE 3.6 : Diagramme de séquence «S'authentifier»

Ce diagramme représente le processus d'authentification d'un utilisateur. Pour se connecter, l'utilisateur fournit les informations d'identification, et le système les vérifie s'il les a déjà. En cas de problème, un message est envoyé.

Le même processus s'applique pour l'inscription d'un gérant et d'un administrateur.

3.3 Diagramme de séquence « Changer mot de passe en cas d'oubli »

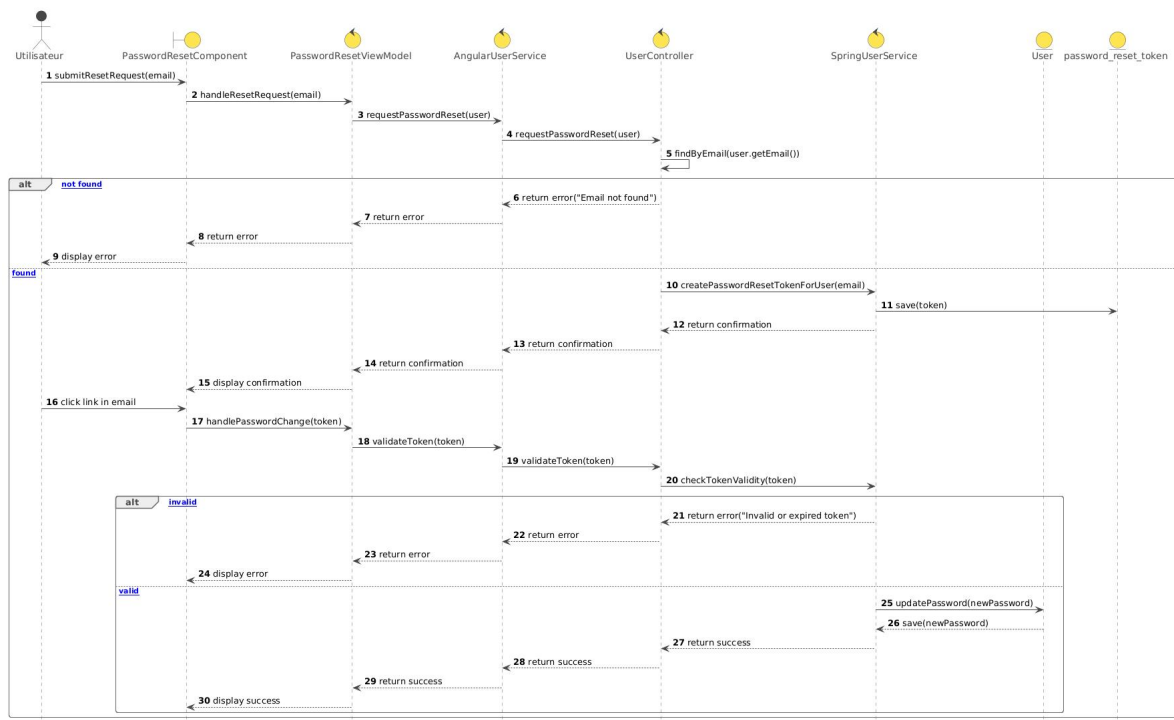


FIGURE 3.7 : Diagramme de séquence «Changer mot de passe en cas d'oubli»

Ce diagramme explique comment réinitialiser le mot de passe. À cet effet, l'utilisateur demande une réinitialisation en fournissant son adresse e-mail. Ensuite, le système fouille l'e-mail, génère un jeton qui est ensuite envoyé à l'utilisateur par messagerie.

Le même processus s'applique pour l'inscription d'un gérant et d'un administrateur.

4 Description Structurelle

4.1 Diagramme de classe global de sprint 1

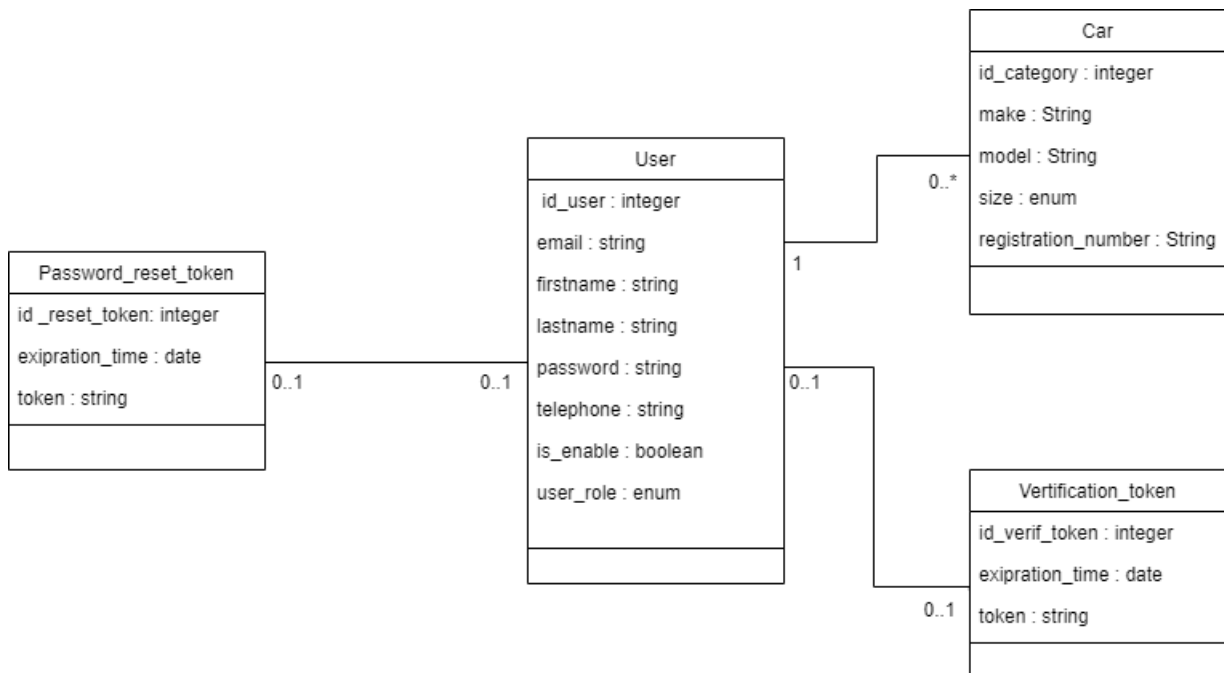


FIGURE 3.8 : Diagramme de classe global de sprint 1

4.2 Réalisation

Le modèle relationnel est basé sur une organisation des données sous forme de tables. La manipulation des données se fait selon le concept mathématique de relation de la théorie des ensembles, c'est-à-dire l'algèbre relationnelle.[19]

Dans cette section nous allons définir la structure de la base de données du premier sprint ainsi que les interfaces homme-machine les plus importantes.

Table structure									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	bigint			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	email	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐ 3	firstname	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐ 4	is_enabled	bit(1)			No	None			Change Drop More
☐ 5	lastname	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐ 6	password	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐ 7	telephone	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐ 8	user_role	tinyint			Yes	NULL			Change Drop More

FIGURE 3.9 : Table : "user"

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1 id	bigint			No	None		AUTO_INCREMENT	Change Drop More
☐	2 expiration_time	datetime(6)			Yes	NULL			Change Drop More
☐	3 token	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐	4 user_id	bigint			No	None			Change Drop More

FIGURE 3.10 : Table : "verification token"

Table structure		Relation view							
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1 id_token	bigint			No	None		AUTO_INCREMENT	Change Drop More
☐	2 expiration_time	datetime(6)			Yes	NULL			Change Drop More
☐	3 token	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐	4 user_id	bigint			No	None			Change Drop More

FIGURE 3.11 : Table : "reset password token"

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1 id	bigint			No	None		AUTO_INCREMENT	Change Drop More
☐	2 make	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐	3 model	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐	4 size	enum('LARGE', 'MEDIUM', 'SMALL')	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More
☐	5 user_id	bigint			Yes	NULL			Change Drop More
☐	6 registration_number	varchar(255)	utf8mb4_0900_ai_ci		Yes	NULL			Change Drop More

FIGURE 3.12 : Table : "car"

5 Tests Postman

Les tests Postman sont des scripts JavaScript exécutés après l'envoi d'une requête pour vérifier la validité et la performance de la réponse reçue. Ils permettent d'automatiser les vérifications et d'assurer que l'API fonctionne comme attendu.

5.1 Test du cas d'utilisation « Inscription »

Nous présentons ci-dessous, le code source et le résultat de cas de test :

1. Vérification du statut de la réponse
2. Vérification que la réponse est au format JSON
3. Vérification que l'utilisateur existe déjà

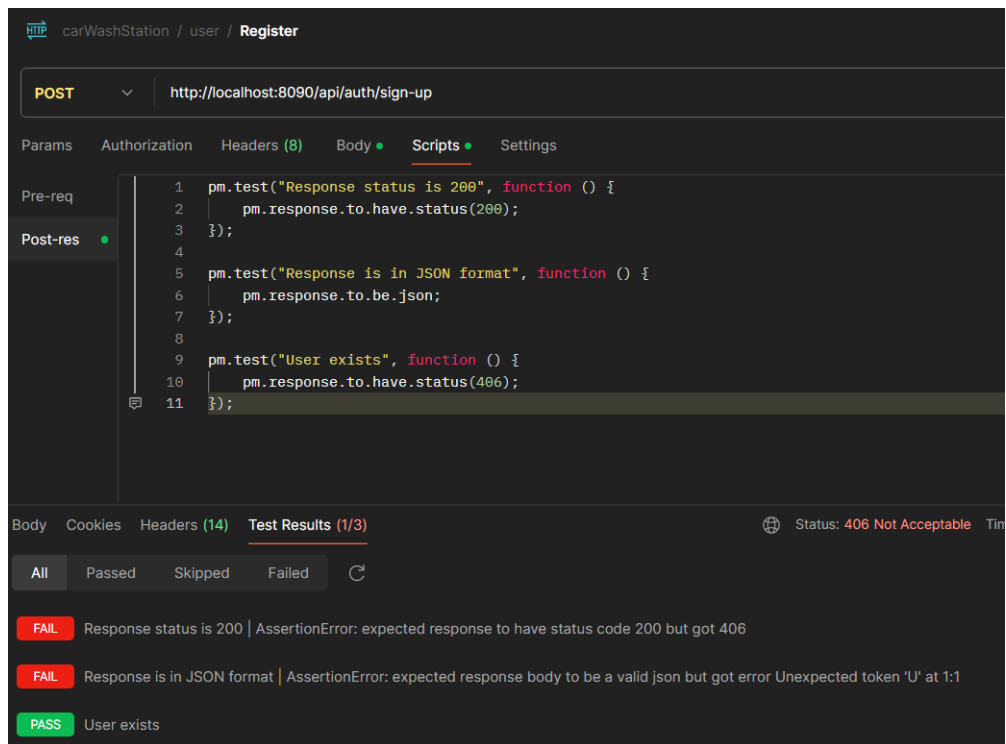


FIGURE 3.13 : Code source du test du cas d'utilisation « Créer un compte »

5.2 Test du cas d'utilisation « Liste des voitures »

Nous présentons ci-dessous, le code source et le résultat de deux cas de test :

1. Test des Propriétés de la Voiture
2. Test du Temps de Réponse

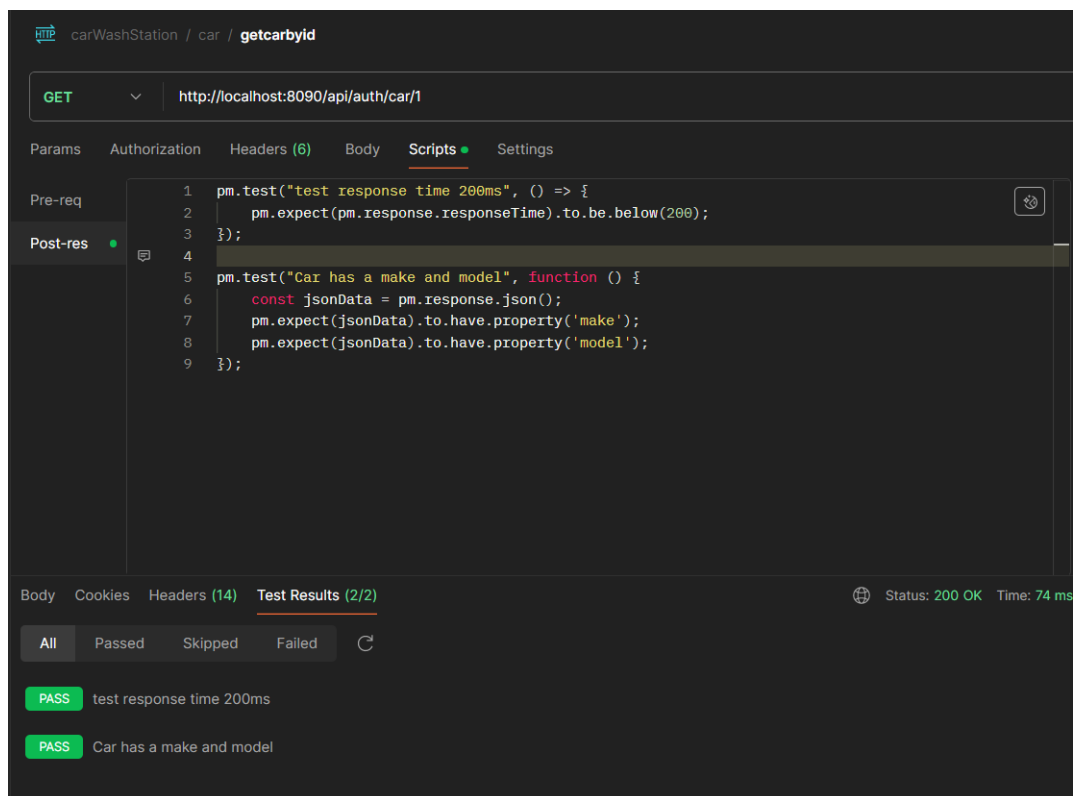


FIGURE 3.14 : Code source du test du cas d'utilisation « Liste des voitures »

6 La revue

- La figure 3.15 est illustrée l'interface d'inscription, accompagnée des validateurs qui gèrent les erreurs de saisie pour chaque champ :

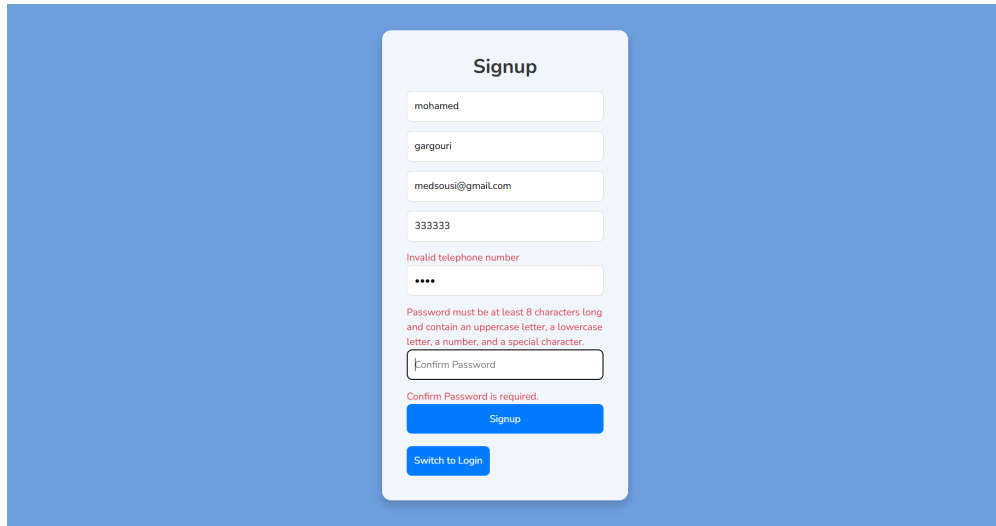


FIGURE 3.15 : SpotWash - Interface d'inscription

-La figure 3.16 montre l'email de vérification qui est envoyé aux utilisateurs après qu'ils ont complété le processus d'inscription.

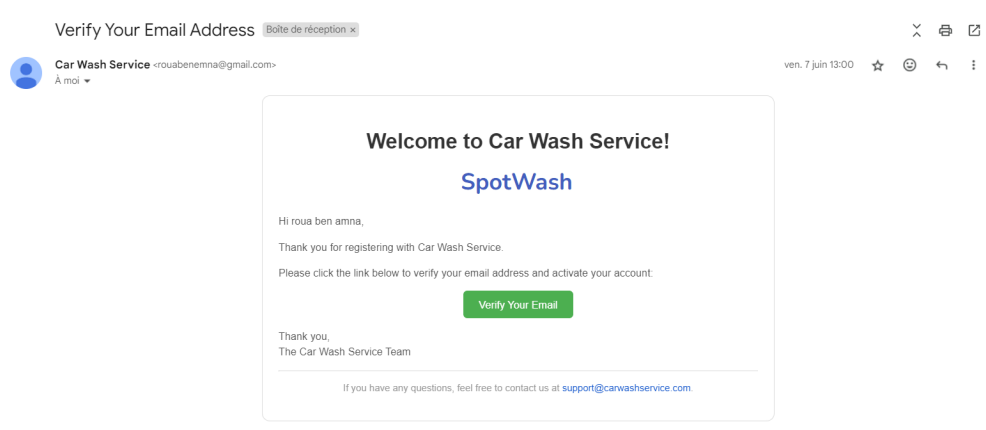


FIGURE 3.16 : SpotWash - Verification email

- La figure 3.17 montre l'interface d'authentification :

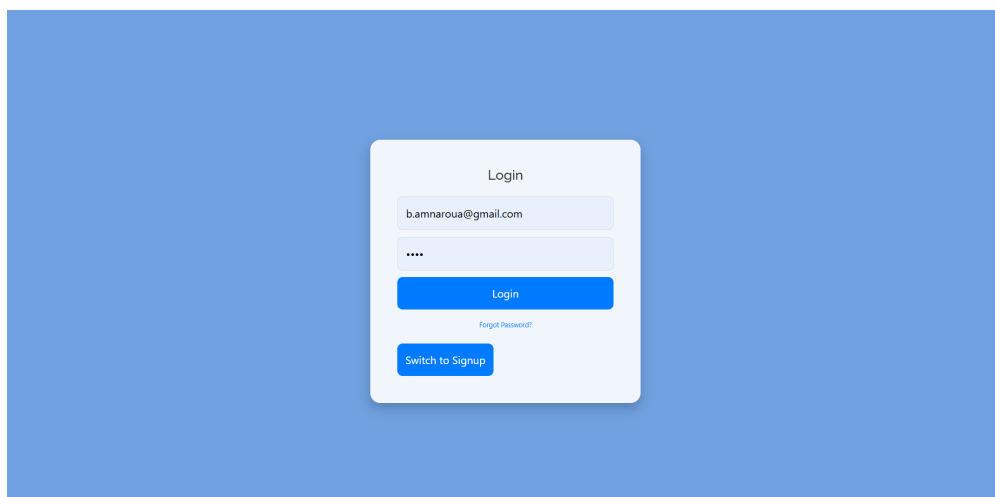


FIGURE 3.17 : SpotWash - Interface d'authentification

- La figure 3.18 montre l'interface de modification des données personnelles de l'administrateur :

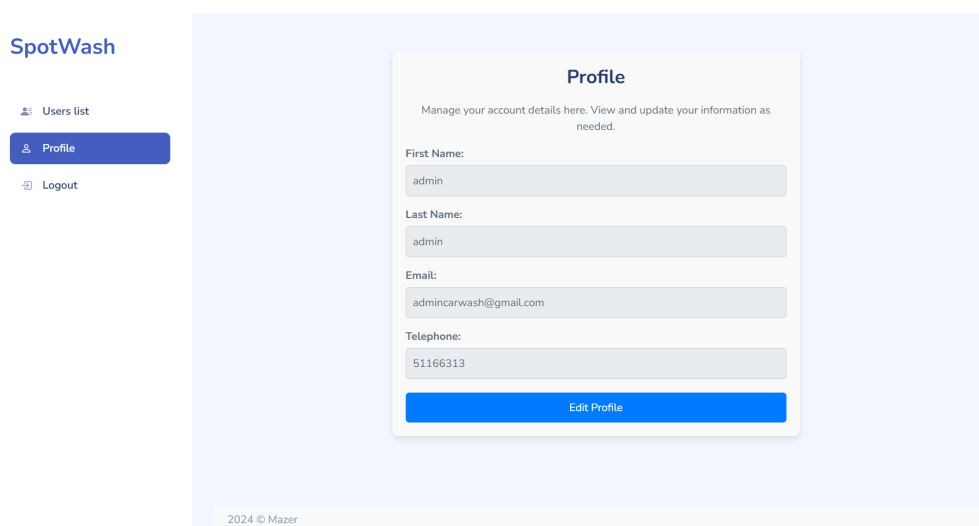


FIGURE 3.18 : SpotWash - Interface de modification profil

- La figure 3.19 illustre l'interface permettant aux utilisateurs de réinitialiser leur mot de passe en cas d'oubli :

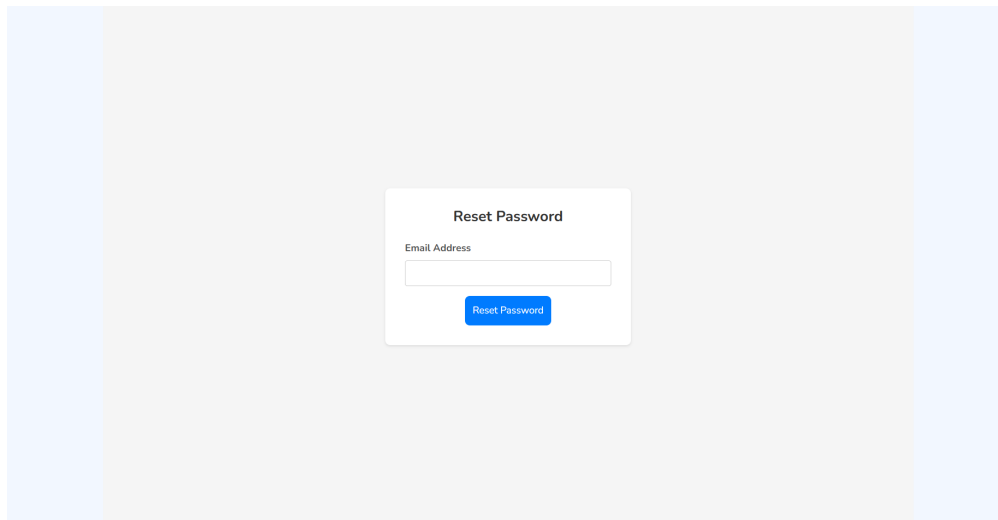


FIGURE 3.19 : SpotWash - interface affectation des permissions

- La figure 3.20 illustre l'interface de la page d'accueil du site :

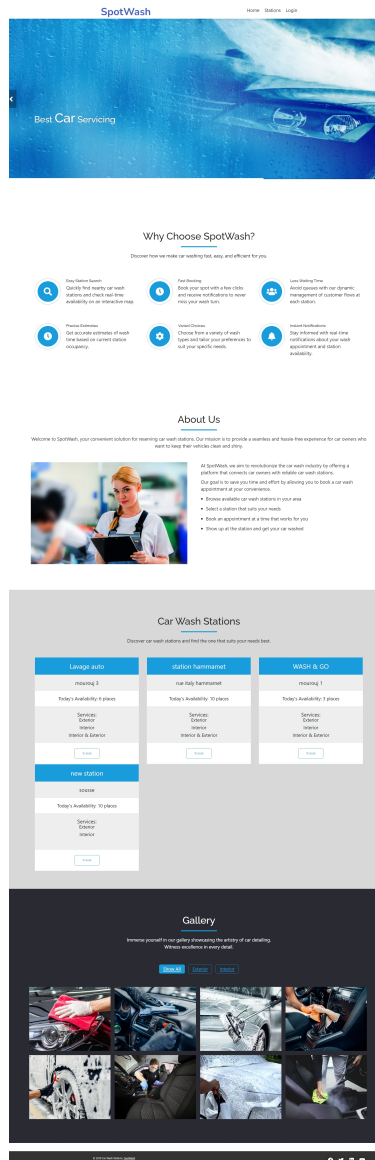


FIGURE 3.20 : SpotWash - Interface de page d'accueil

- La figure 3.21 illustre l'interface affichant la liste des voitures, ainsi que l'option de recherche disponible :

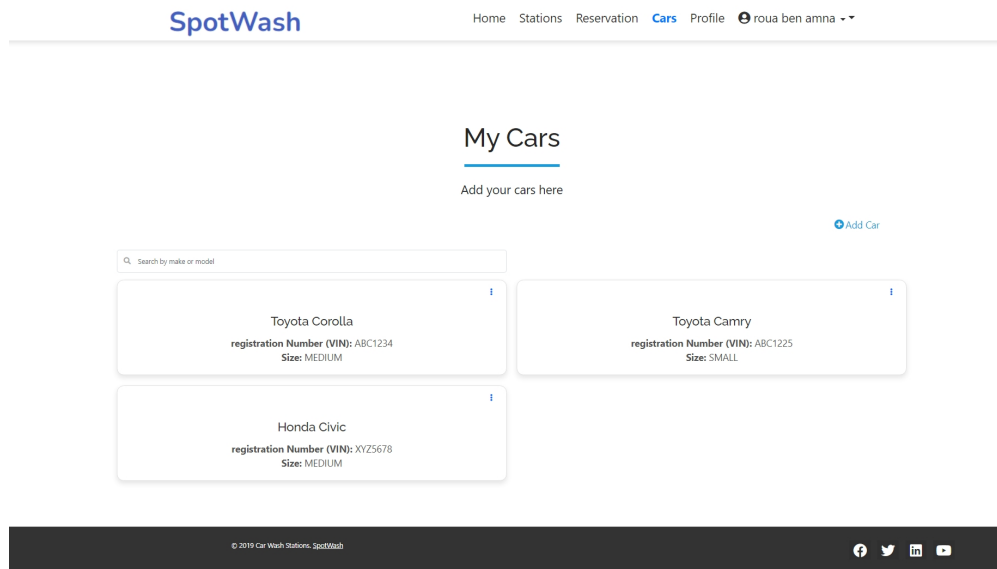


FIGURE 3.21 : SpotWash - Interface d'une liste des voiture

- La figure 3.22 illustre l'interface de modification de voiture, avec les champs automatiquement remplis par les informations précédemment saisies :

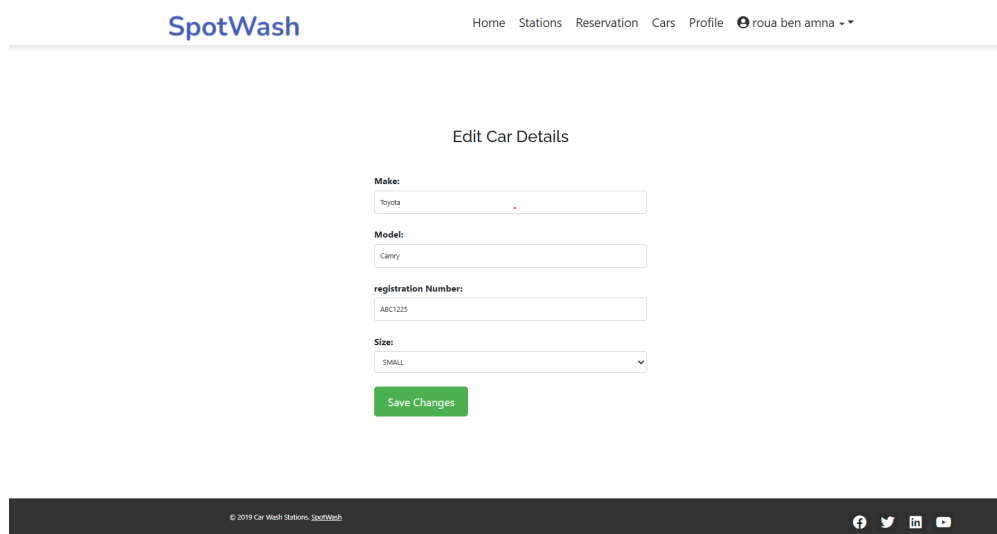


FIGURE 3.22 : SpotWash - Interface de modification voiture

7 Rétrospective

La rétrospective du sprint« a lieu après la revue du sprint et avant la prochaine planification du sprint. Le but de la rétrospective de Sprint est de :

- Inspecter la manière dont le dernier Sprint s'est déroulé en ce qui concerne les personnes,

les relations , les processus et les outils.

- Créer un plan pour mettre en œuvre des améliorations sur la façon dont l'équipe Scrum fait son travail ». [20]

7.1 Scrum Board

Le scrum board représente l'état de chaque sprint backlog item : s'il est à faire, en cours ou terminé.

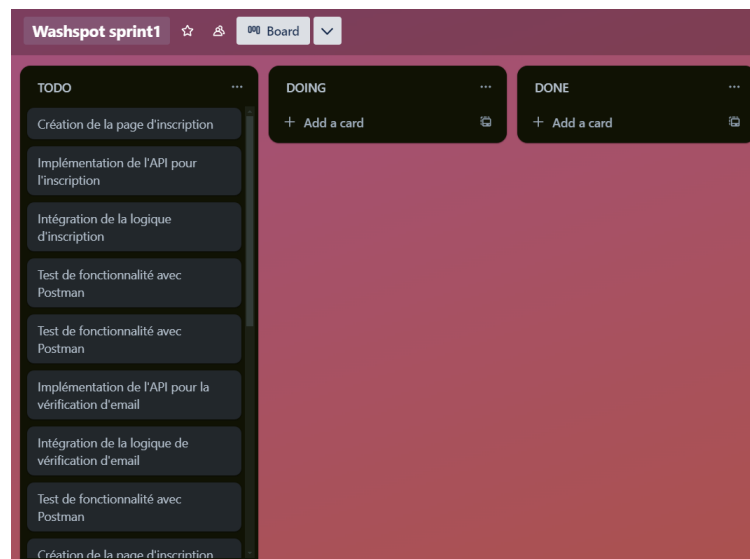


FIGURE 3.23 : Scrum board « jour 1 »

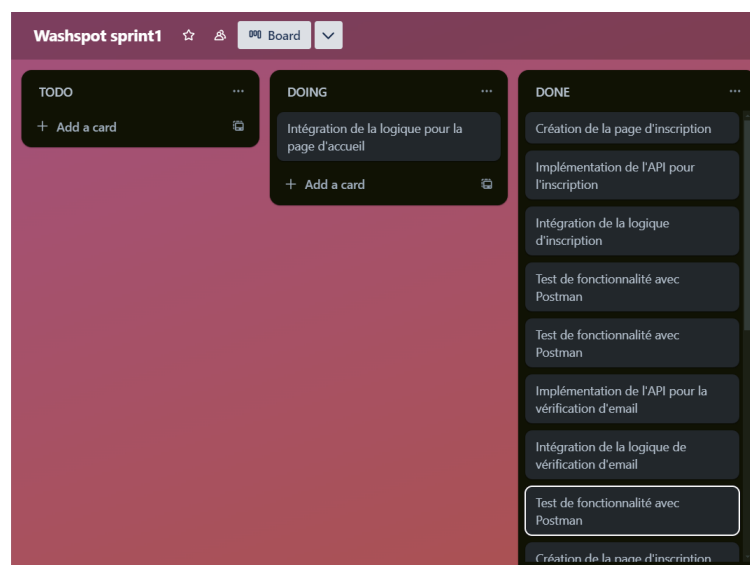


FIGURE 3.24 : Scrum board «dernier jour»

7.2 Tableau de Rétrospective

TABLERAU 3.7 : Tableau de rétrospective « Sprint 1 »

Questions	Réponses
Qu'est-ce qu'on a bien fait ?	• La conception • Respecter le sprint goal • Le travail effectué est bien conforme au besoin
Qu'est ce qui ne s'est pas bien passé ?	• Bien maîtriser le Framework et les outils de travail
Que devrions-nous arrêter de le faire ?	• Se soucier du deadline du sprint
Que devrions-nous commencer à faire ?	• Se concentrer sur la valeur à livrer au lieu du deadline

Conclusion

A travers ce chapitre, nous avons présenté le sprint backlog, la conception, la revue et la rétrospective qui reflètent l'incrément du travail potentiellement livrable effectué au cours du sprint 1.

Le chapitre qui suit sera dédié à un nouveau sprint dont l'objectif sera spécifié au cours du sprint planning meeting.

CHAPITRE 4

ÉTUDE ET RÉALISATION DU SPRINT 2

1	Réunion de planification du sprint (Sprint planning meeting)	63
2	Description fonctionnelle des histoires du Sprint 2	71
3	Description comportementale des histoires du sprint 2	75
4	Description Structurale	76
5	Tests Postman	77
6	La revue	78
7	Rétrospective	81

Introduction

Ce chapitre débutera par une clarification des objectifs du sprint 2, suivie d'une présentation détaillée du backlog du sprint. Nous terminerons par une description des étapes de conception et de réalisation, en expliquant les méthodes et processus utilisés pour accomplir cette itération.

1 Réunion de planification du sprint (Sprint planning meeting)

1.1 Objectif du sprint (Sprint goal)

Au cours de ce sprint, nous réalisons la gestion des stations de lavage, Recherche de Stations de Lavage, disponibilité en temps réel et réservation des sessions .

1.2 Backlog du sprint

Le Backlog du Sprint est l'ensemble des éléments sélectionnés, pour le Sprint plus un plan pour livrer l'incrément du produit et réaliser l'objectif du Sprint.

TABLERAU 4.1 : Backlog du sprint - Sprint 2

Fonctionnalité	N°	User Story	Tâches Associées
Gestion des Stations de Lavage	1	En tant que gérant, je veux consulter la liste des stations de lavage.	• Création de la page de liste des stations (angular) • Implémentation de l'API pour récupérer la liste des stations (Spring) • Intégration de la logique pour récupérer la liste des stations (angular) • Test de fonctionnalité avec Postman

	2	En tant que gérant, je veux ajouter une station de lavage.	• Création d'un formulaire pour ajouter une station, intégré dans une modal (angular) • Implémentation de l'API pour ajouter une station (Spring) • Intégration de la logique pour ajouter une station (angular) • Test de fonctionnalité avec Postman
	3	En tant que gérant, je veux modifier une station de lavage.	• Création d'un formulaire pour modifier une station, intégré dans une modal (angular) • Implémentation de l'API pour modifier une station (Spring) • Intégration de la logique pour modifier une station (angular) • Test de fonctionnalité avec Postman

	4	En tant que gérant, je veux supprimer une station de lavage.	• Création d'un bouton pour supprimer une station et d'une modal de confirmation (angular). • Implémentation de l'API pour supprimer une station (Spring) • Intégration de la logique pour supprimer une station (angular) • Test de fonctionnalité avec Postman
--	---	--	--

Recherche de Stations de Lavage	5	En tant que visiteur, je veux rechercher des stations de lavage à proximité.	• Création de la fonctionnalité de recherche pour les stations avec intégration de la géolocalisation sur une carte (angular) • Implémentation de l'API de recherche (Spring) • Intégration de la logique de recherche (angular) • Test de fonctionnalité avec Postman
Disponibilité en Temps Réel	6	En tant qu'utilisateur, je veux voir le nombre de places disponibles.	• Implémentation de l'API pour récupérer les places disponibles (Spring) • Intégration de la logique pour récupérer les places disponibles (angular) • Test de fonctionnalité avec Postman

	7	En tant qu'utilisateur, je veux voir les informations asynchrones.	• Mise en place de la fonctionnalité pour récupérer et afficher les informations en temps réel (angular) • Intégration de la logique pour afficher les informations en temps réel (angular) • Test de fonctionnalité avec Postman
Gestion des Sessions	8	En tant qu'utilisateur, je veux consulter la liste des sessions.	• Création de la page de liste des sessions (angular) • Implémentation de l'API pour afficher les sessions (Spring) • Intégration de la logique pour afficher les sessions (angular)

	9	En tant qu'utilisateur, je veux ajouter une session de lavage pour ma voiture.	• Création de la page de réservation de session (angular) • Implémentation de l'API pour réserver une session (Spring) • Intégration de la logique pour réserver une session (angular) • Test de fonctionnalité avec Postman
	10	En tant qu'utilisateur, je veux modifier une session.	• Création de la page pour modifier une session (angular) • Implémentation de l'API pour modifier une session (Spring) • Intégration de la logique pour modifier une session (angular) • Test de fonctionnalité avec Postman

	11	En tant qu'utilisateur, je veux annuler une session.	• Création d'un bouton pour annuler une session et d'une modal de confirmation (angular) • Implémentation de l'API pour annuler une session (Spring) • Intégration de la logique pour annuler une session (angular) • Test de fonctionnalité avec Postman
	12	En tant qu'utilisateur, je veux voir les temps d'attente estimés.	• Implémentation de l'API pour récupérer les temps d'attente (Spring) • Intégration de la logique pour afficher les temps d'attente (angular) • Test de fonctionnalité avec Postman

2 Description fonctionnelle des histoires du Sprint 2

2.1 Diagramme de cas d'utilisation général du sprint 2

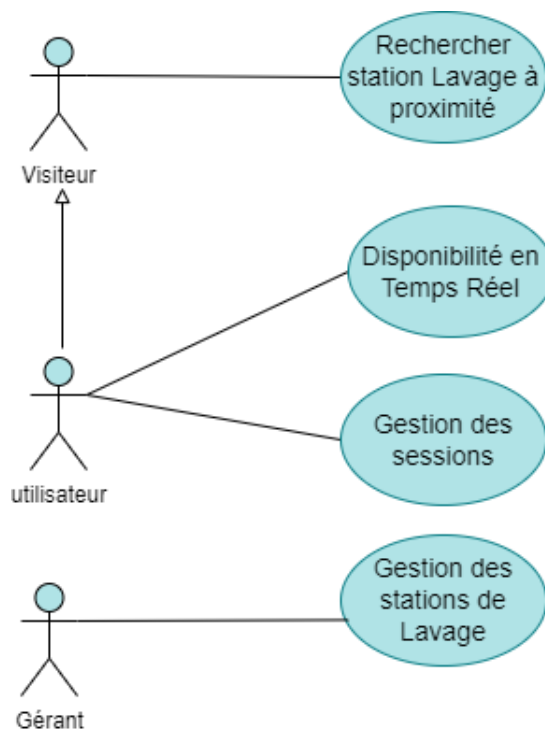


FIGURE 4.1 : Diagramme de cas d'utilisation général du sprint 2

2.2 Diagramme raffiné du cas d'utilisation « Gestion des stations de lavage »

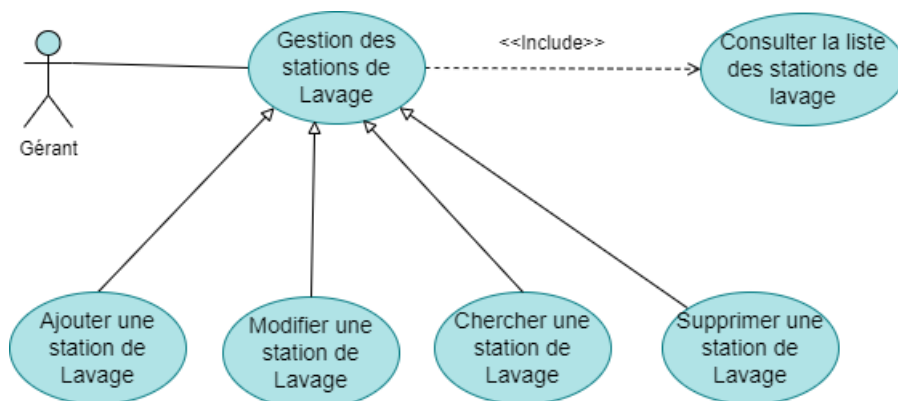


FIGURE 4.2 : Diagramme de cas d'utilisation «Gestion des stations de lavage»

2.3 Diagramme raffiné du cas d'utilisation « Gestion des Sessions »

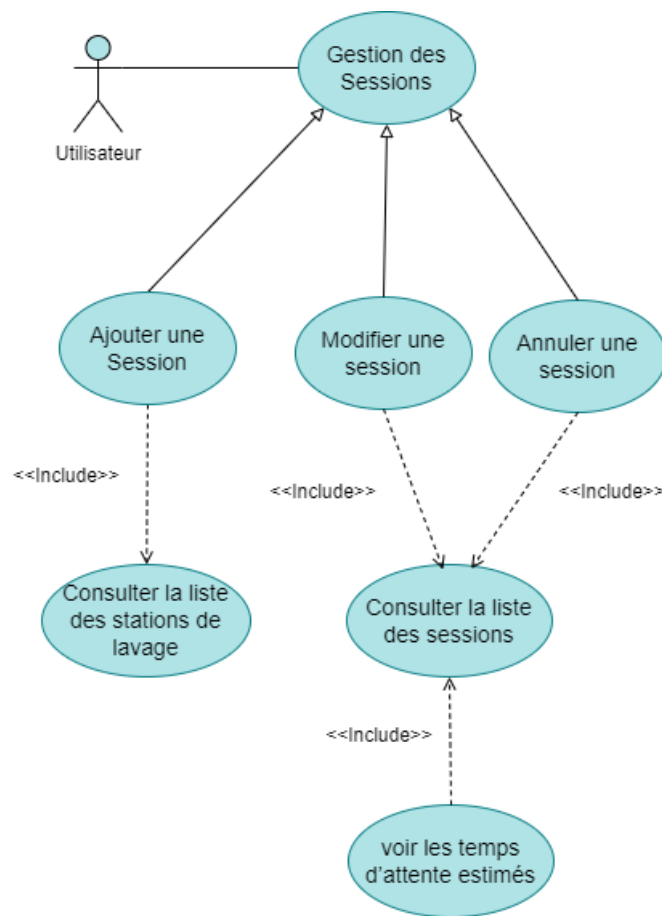


FIGURE 4.3 : Diagramme de cas d'utilisation « Gestion des Sessions »

2.4 Diagramme raffiné du cas d'utilisation « Disponibilité en Temps Réel »

»

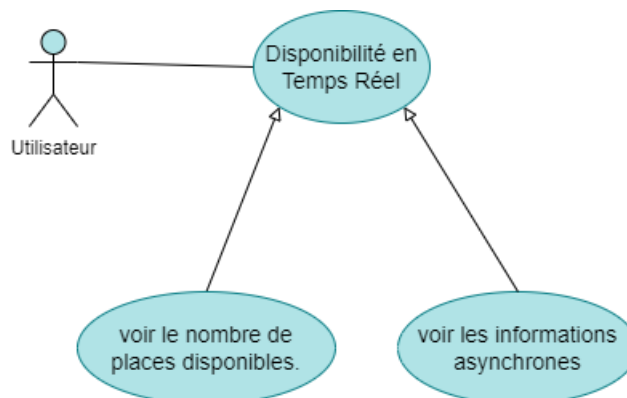


FIGURE 4.4 : Diagramme de cas d'utilisation « Disponibilité en Temps Réel »

Afin d'éviter de surcharger le backend, nous avons privilégié le polling dans Angular pour présenter les informations asynchrones à l'utilisateur. Tout d'abord, nous avons opté pour un intervalle de polling équilibré, qui était initialement de 10 secondes.

2.5 Description Textuelle de cas d'utilisation « Ajouter une station »

TABLEAU 4.2 : Description Textuelle de cas d'utilisation « Ajouter une station »

Titre	Ajouter station
Acteurs	Gérant
Pré condition	• Gérant authentifié. • Liste des stations affichée.
Scénario nominal	1. Le gérant demande l'ajout d'une nouvelle station. 2. Le système affiche le formulaire d'ajout d'une station. 3. Le gérant saisit les informations nécessaires de nouvelle station. 4. Le gérant clique sur "Add". 5. le système vérifie les données saisies.
Scénario alternatif	5. A1 -le système affiche un message d'erreur si l'un des champs obligatoires est vide et/ou invalide ou une erreur lors de l'ajout des séances. Le cas d'utilisation redémarre de l'étape 3 du scénario nominal.
Post condition	La station ajoutée.

Lorsqu'une station est ajoutée, le gérant peut directement indiquer l'emplacement exact sur la carte en utilisant la bibliothèque Leaflet. Cette fonctionnalité permet au gérant de sélectionner précisément la position de la station de lavage, garantissant ainsi une localisation précise et fiable pour les utilisateurs finaux.

2.6 Description Textuelle de cas d'utilisation « Modifier une station »

TABLERAU 4.3 : Description Textuelle de cas d'utilisation « Modifier une station »

Titre	Modifier station
Acteurs	Gérant
Pré condition	• Gérant authentifié • Liste des stations affichée. • station existe.
Scénario nominal	1. Le gérant sélectionne une station. 2. Le gérant clique sur l'icône « Edit ». 3. Le système affiche les informations de la station dans un formulaire . 4. Le gérant valide le formulaire. 5. Le système vérifie les informations saisies par le gérant.
Scénario alternatif	5. A1 -le système affiche un message d'erreur si l'un des champs obligatoires est vide et/ou invalide. Le cas d'utilisation redémarre de l'étape 3 du scénario nominal.
Post condition	Le station modifiée.

3 Description comportementale des histoires du sprint 2

3.1 Diagramme de séquence « Recherche les stations à proximité »

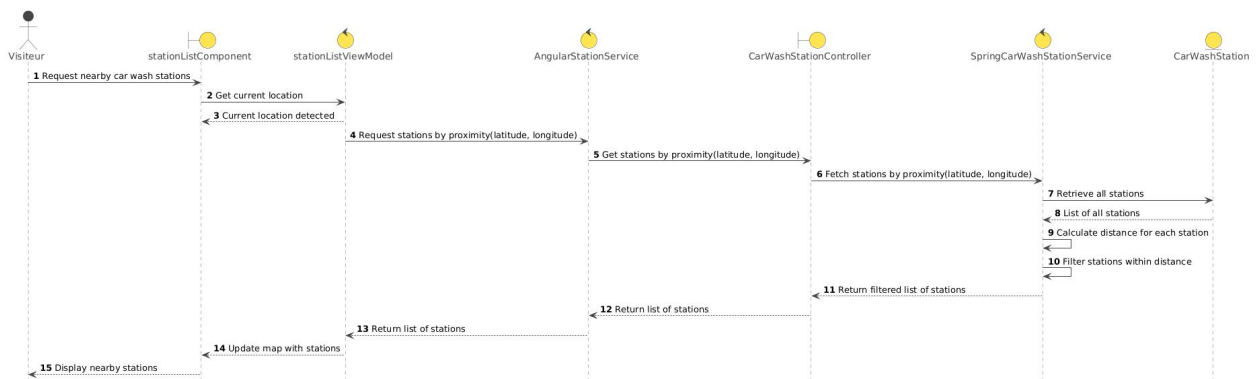


FIGURE 4.5 : Diagramme de séquence « Recherche les stations à proximité »

Grâce à la carte interactive, les visiteurs et les utilisateurs de l'application peuvent facilement repérer les stations de lavage à proximité. Les utilisateurs peuvent également consulter des informations supplémentaires sur les services offerts directement à partir de la carte géolocalisée.

3.2 Diagramme de séquence « Ajouter une session »

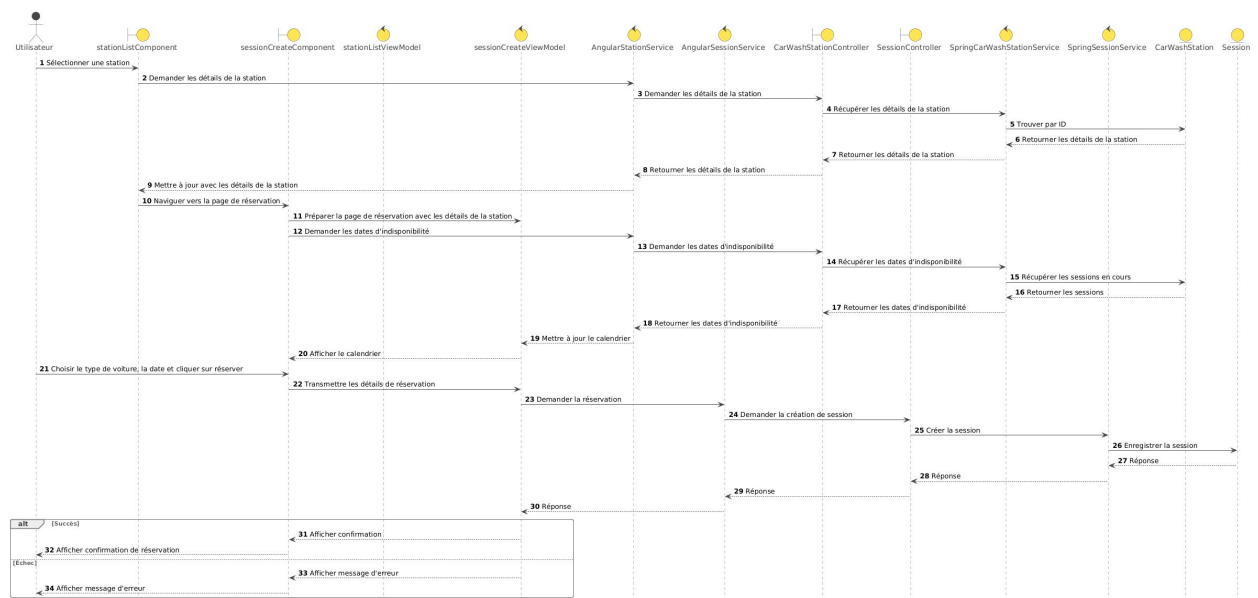


FIGURE 4.6 : Diagramme de séquence « Ajouter session »

4 Description Structurale

4.1 Diagramme de classe global de sprint 2

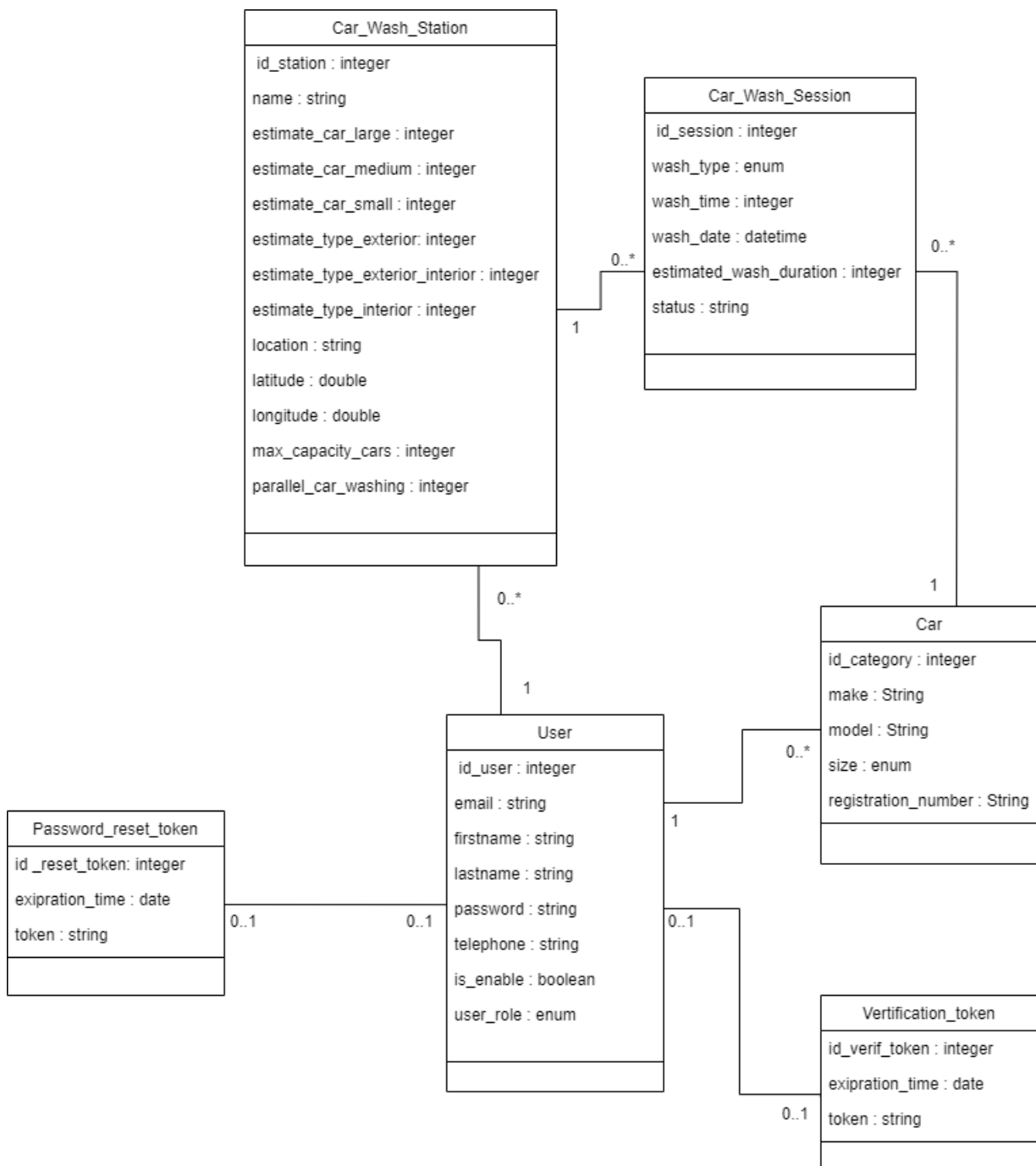


FIGURE 4.7 : Diagramme de classe global de sprint 2

4.2 Réalisation

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	bigint			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	name	varchar(255)	utf8mb4_0900_ai_ci		Oui	NULL			Modifier Supprimer Plus
3	estimate_car_large	bigint			Non	Aucun(e)			Modifier Supprimer Plus
4	estimate_car_medium	bigint			Non	Aucun(e)			Modifier Supprimer Plus
5	estimate_car_small	bigint			Non	Aucun(e)			Modifier Supprimer Plus
6	estimate_type_exterior	bigint			Non	Aucun(e)			Modifier Supprimer Plus
7	estimate_type_exterior_interior	bigint			Non	Aucun(e)			Modifier Supprimer Plus
8	estimate_type_interior	bigint			Non	Aucun(e)			Modifier Supprimer Plus
9	location	varchar(255)	utf8mb4_0900_ai_ci		Oui	NULL			Modifier Supprimer Plus
10	latitude	double			Non	Aucun(e)			Modifier Supprimer Plus
11	longitude	double			Non	Aucun(e)			Modifier Supprimer Plus
12	max_capacity_cars	int			Oui	NULL			Modifier Supprimer Plus
13	parallel_car_washing	int			Oui	NULL			Modifier Supprimer Plus
14	user_id	bigint			Oui	NULL			Modifier Supprimer Plus

FIGURE 4.8 : Table : "car wash station"

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	bigint			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	estimated_wash_duration	bigint			Oui	NULL			Modifier Supprimer Plus
3	status	varchar(255)	utf8mb4_0900_ai_ci		Oui	NULL			Modifier Supprimer Plus
4	wash_time	bigint			Oui	NULL			Modifier Supprimer Plus
5	wash_type	enum('EXTERIOR', 'EXTERIOR_INTERIOR', 'INTERIOR')	utf8mb4_0900_ai_ci		Oui	NULL			Modifier Supprimer Plus
6	car_id	bigint			Oui	NULL			Modifier Supprimer Plus
7	station_id	bigint			Oui	NULL			Modifier Supprimer Plus
8	wash_date	datetime(6)			Oui	NULL			Modifier Supprimer Plus

FIGURE 4.9 : Table : "car wash session"

5 Tests Postman

5.1 Test du cas d'utilisation « Modifier une session »

Nous présentons ci-dessous le code source ainsi que le résultat de deux cas de test :

1. Vérification du type de lavage initial, qui doit être « INTERIOR ».
2. Vérification du type de lavage après modification, qui doit être « EXTERIOR ».

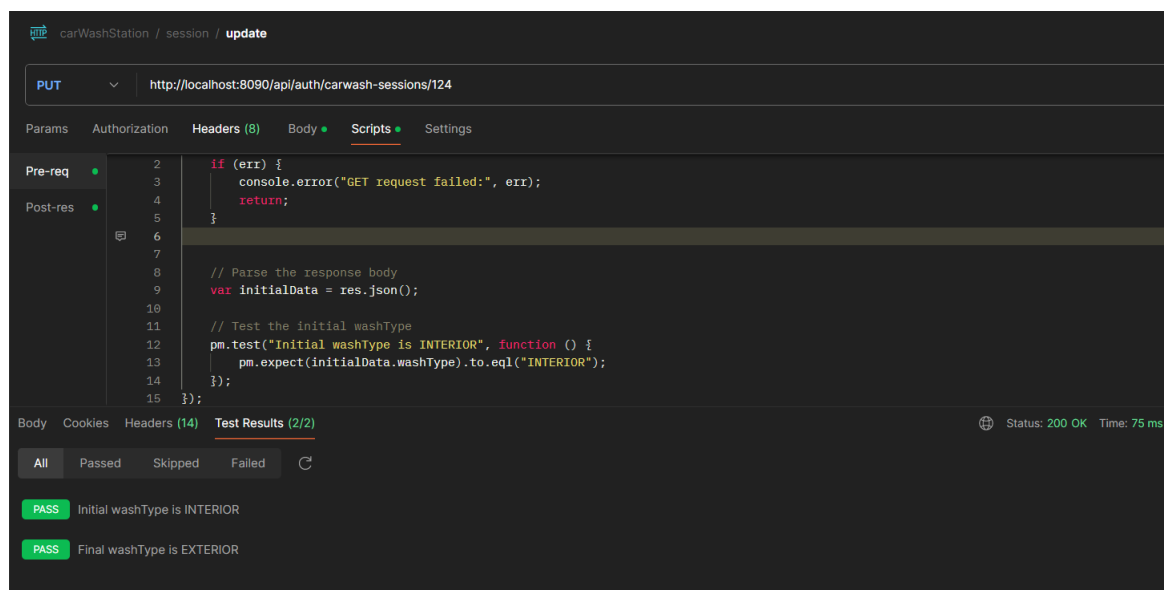


FIGURE 4.10 : Code source du test du cas d'utilisation « Modifier session »

5.2 Test du cas d'utilisation « Disponibilité de station de lavage »

Nous présentons ci-dessous le code source ainsi que le résultat du deux cas de test :

1. Vérification du statut de la réponse.
2. Vérification les dates non disponibles.

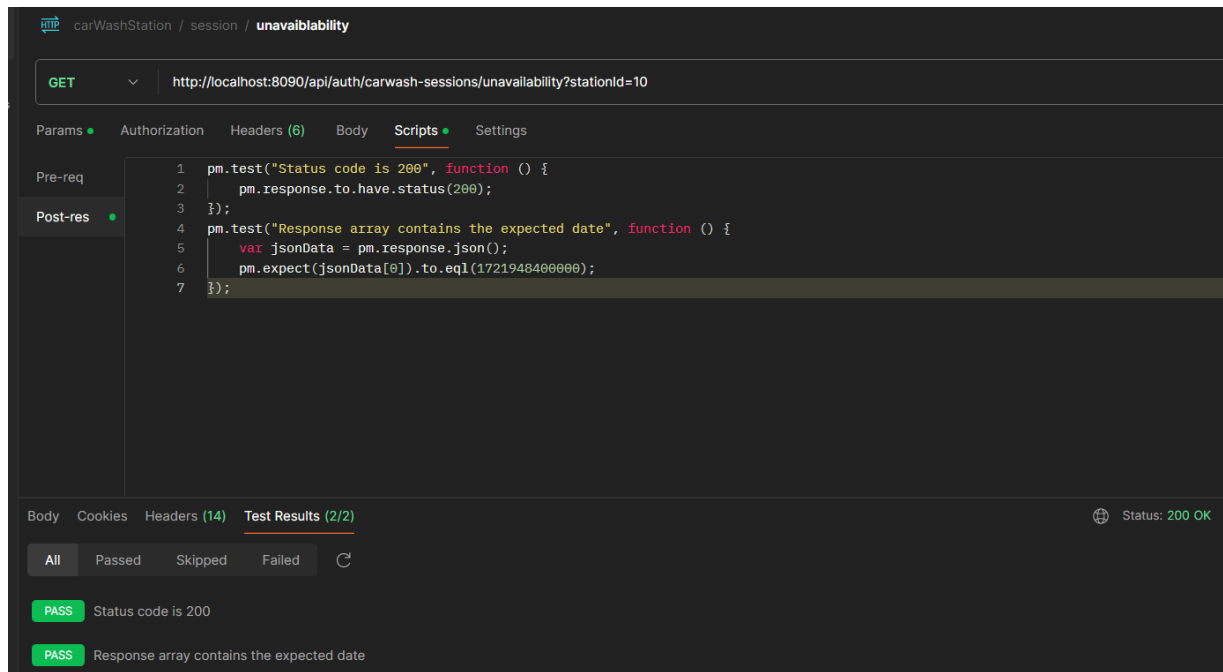


FIGURE 4.11 : Code source du test du cas d'utilisation « Disponibilité de station de lavage »

6 La revue

- La figure 4.12 illustre l'interface affichant la liste des stations de lavage, ainsi que les options de gestion des stations, telles que les boutons pour ajouter, modifier et supprimer :

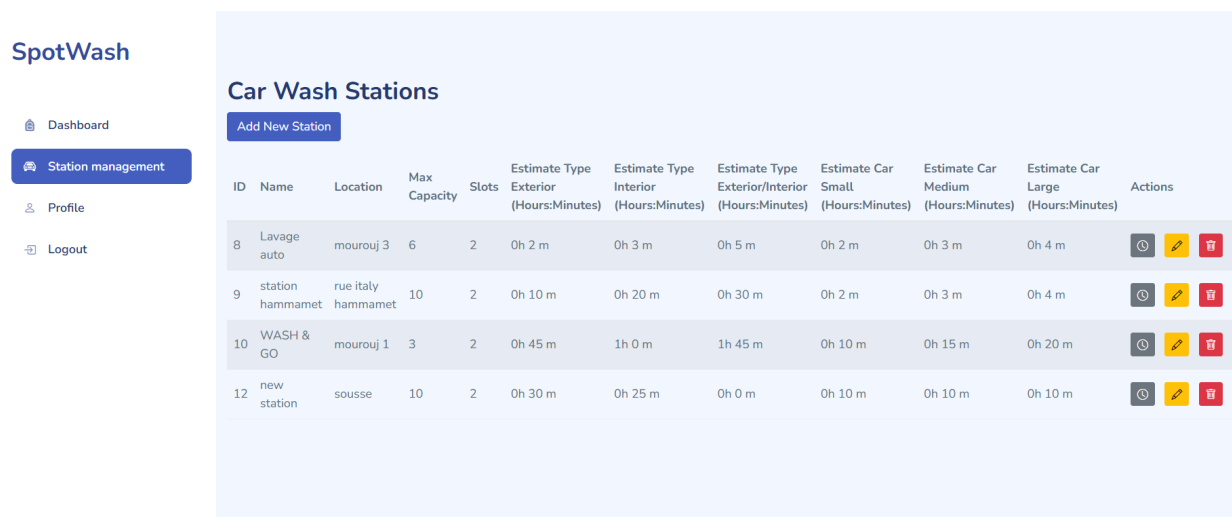


FIGURE 4.12 : SpotWash - Interface de la liste des stations de lavage

- En cliquant sur le bouton « Add New Station », une fenêtre modale pour ajouter une nouvelle station s'affiche, comme illustré dans la figure 4.13 :

SpotWash

Dashboard
Station management
Profile
Logout

Stations

Location	Max Capacity	Station Type
mourouj 3	6	2
rue italy hammamet	10	2
mourouj 1	3	2
sousse	10	2

Estimate Type Exterior/Interior (Hours:Minutes)
hh:mm

Estimate Car Small (Hours:Minutes) *
hh:mm

Estimate Car Medium (Hours:Minutes) *
hh:mm

Estimate Car Large (Hours:Minutes) *
hh:mm

Coordinates *
Latitude, Longitude:

Map showing location selection.

Add

Station Name	Estimate Type Exterior/Interior (Hours:Minutes)	Estimate Car Small (Hours:Minutes)	Estimate Car Medium (Hours:Minutes)	Estimate Car Large (Hours:Minutes)	Actions
mourouj 3	0h 3 m	0h 4 m			[Clock] [Edit] [Delete]
rue italy hammamet	0h 3 m	0h 4 m			[Clock] [Edit] [Delete]
mourouj 1	0h 15 m	0h 20 m			[Clock] [Edit] [Delete]
sousse	0h 10 m	0h 10 m			[Clock] [Edit] [Delete]

FIGURE 4.13 : SpotWash - Interface d'ajout d'une station

- La figure 4.14 illustre la fenêtre modale de confirmation pour la suppression d'une station de lavage :

SpotWash

Dashboard
Station management
Profile
Logout

Car Wash

Add New Station

Confirm Delete

Are you sure you want to delete this station?

Cancel Delete

ID	Name	Location	Max Capacity	Station Type	Estimate Type Exterior/Interior (Hours:Minutes)	Estimate Car Small (Hours:Minutes)	Estimate Car Medium (Hours:Minutes)	Estimate Car Large (Hours:Minutes)
8	Lavage auto	mourouj 3	6	2	0h 2 m	0h 3 m	0h 5 m	0h 4 m
9	station hammamet	rue italy hammamet	10	2	0h 10 m	0h 20 m	0h 30 m	0h 4 m
10	WASH & GO	mourouj 1	3	2	0h 45 m	1h 0 m	1h 45 m	0h 20 m
12	new station	sousse	10	2	0h 30 m	0h 25 m	0h 0 m	0h 10 m

FIGURE 4.14 : SpotWash - Interface de suppression

- La figure 4.15 montre l'interface de la carte indique où se trouve actuellement l'utilisateur. Elle montre aussi les stations à proximité dans un rayon de 20 km.

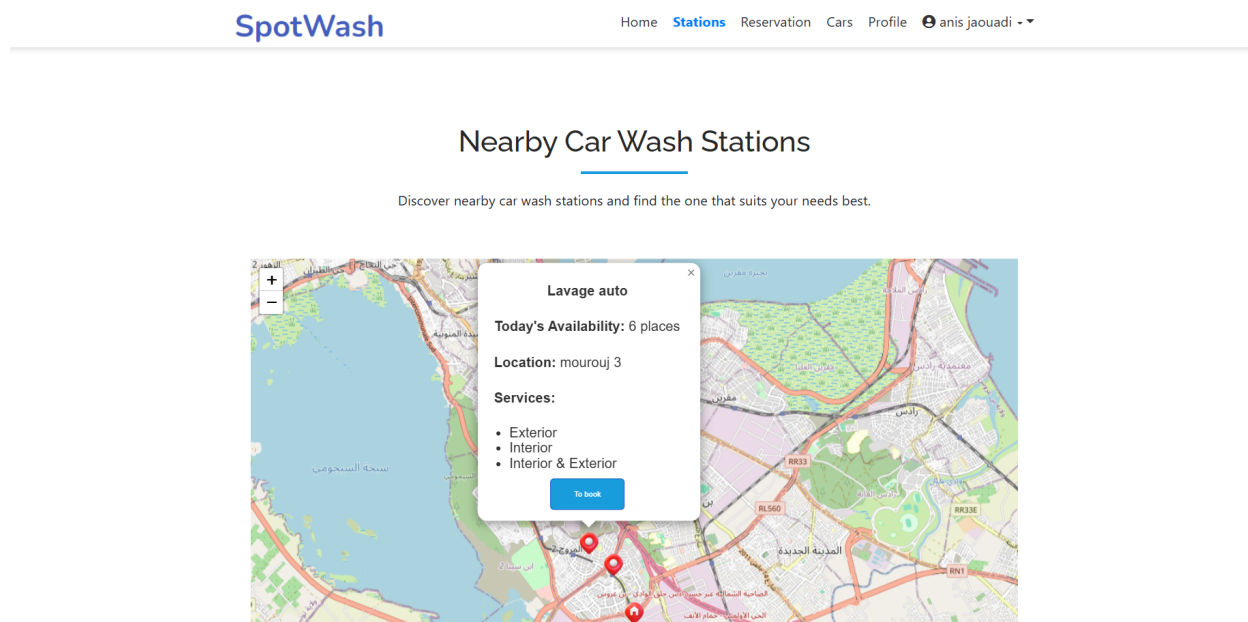


FIGURE 4.15 : SpotWash - Interface de Carte avec Stations de lavage

- La figure 4.16 montre l'interface permet d'ajouter une réservation en sélectionnant la voiture et le type de lavage. Ensuite, l'utilisateur peut choisir la date souhaitée en utilisant un calendrier.

Les dates non disponibles sont colorées en rouge.

En cliquant sur une date disponible, l'utilisateur peut également voir la charge de travail de la station pour cette date.

SpotWash

[Home](#) [Stations](#) [Reservation](#) [Cars](#) [Profile](#) [roua ben amna](#)

Create Car Wash Session

Car Wash Station:

WASH & GO

Car:

Camry

Wash Type:

Exterior

Wash Date:

< August 2024 >

Wk	Su	Mo	Tu	We	Th	Fr	Sa
30	28	29	30	31	1	2	3
31	4	5	6	7	8	9	10
32	11	12	13	14	15	16	17
33	18	19	20	21	22	23	24
34	25	26	27	28	29	30	31

Selected date is not available.

Create Session

FIGURE 4.16 : SpotWash - Interface d'ajout une réservation

7 Rétrospective

7.1 Scrum Board

Le scrum board représente l'état de chaque sprint backlog item : s'il est à faire, en cours ou terminé.

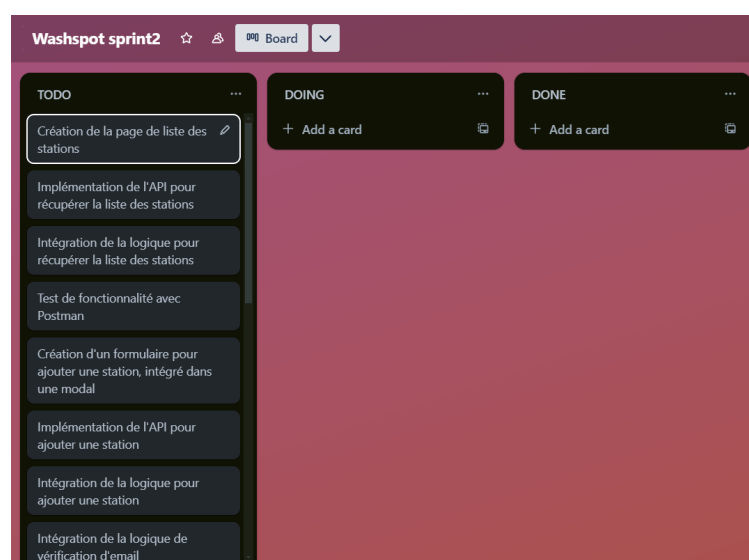


FIGURE 4.17 : Scrum board « jour 1 »

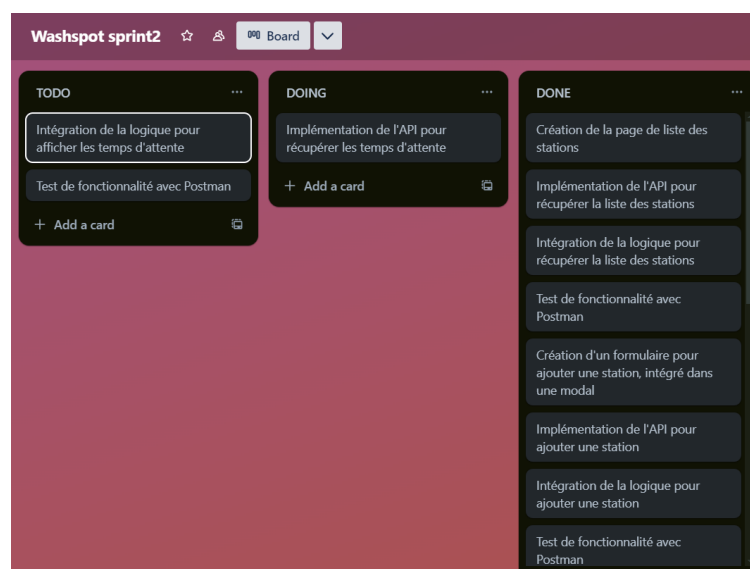


FIGURE 4.18 : Scrum board «dernier jour»

7.2 Tableau de Rétrospective

TABLEAU 4.4 : Tableau de rétrospective « Sprint 2 »

Questions	Réponses
Qu'est-ce qu'on a bien fait?	• La conception • Respecter le sprint goal • Bien maitriser le Framework et les outils de travail
Qu'est ce qui ne s'est pas bien passé?	• Délai dans la livraison d'une user story.
Que devrions-nous arrêter de le faire?	• Travailler en amant pour bien comprendre et assimiler le besoin
Que devrions-nous commencer à faire	• Essayer d'être auto organisé

Conclusion

À la fin de ce sprint, nous avons passé la phase d'étude et la réalisation du sprint 2 . Nous avons donc un deuxième incrément potentiellement livrable de notre logiciel.

Le prochain chapitre sera consacré à la production de notre dernier sprint.

CHAPITRE 5

ÉTUDE ET RÉALISATION DU SPRINT 3

1	Réunion de planification du sprint (Sprint planning meeting)	84
2	Description fonctionnelle des histoires du Sprint 3	92
3	Description comportementale des histoires du sprint 3	98
4	Description Structurale	99
5	Tests Postman	100
6	La revue	101
7	Rétrospective	106

Introduction

Durant ce chapitre, nous visons à finaliser des fonctionnalités essentielles pour améliorer l'application avec le Sprint 3. Nous clarifierons les objectifs de ce sprint, puis nous présenterons le backlog correspondant. Enfin, nous détaillerons les étapes de conception et de réalisation pour cette itération.

1 Réunion de planification du sprint (Sprint planning meeting)

1.1 Objectif du sprint (Sprint goal)

au cours de ce sprint, nous réalisons la gestion d'affluence, notifications, statistiques et supervision des utilisateurs et des gérants.

Il est à noter qu'une user story du Sprint 2 "voir les temps d'attente estimés" n'a pas été entièrement complétée. Par conséquent, cette user story sera reportée et intégrée dans le Backlog du Sprint 3, où elle sera finalisée. Cette intégration garantit que l'objectif initial de la user story sera atteint.

1.2 Sprint Backlog

Le Backlog Sprint est l'ensemble des éléments sélectionnés, pour le Sprint plus un plan pour livrer l'incrément du produit et réaliser l'objectif du Sprint.

TABLEAU 5.1 : Backlog du sprint - Sprint 3

Fonctionnalité	N°	User Story	Tâches Associées
Réservation de Session	1	En tant qu'utilisateur, je veux voir les temps d'attente estimés.	• Implémentation de l'API pour récupérer les temps d'attente (Spring) • Intégration de la logique pour afficher les temps d'attente (angular) • Test de fonctionnalité avec Postman
Gestion d'affluence	2	En tant que gérant, je veux consulter la liste des sessions d'une station de lavage.	• Création de la page de liste des sessions (angular) • Implémentation de l'API pour récupérer la liste des sessions (Spring) • Intégration de la logique pour récupérer la liste des sessions (angular) • Test de fonctionnalité avec Postman

	3	En tant que gérant, je veux filtrer la liste des sessions par date.	• Intégration de la logique pour filtrer la liste des sessions (angular) • Test de fonctionnalité avec Postman
	4	En tant que gérant, je veux filtrer la liste des sessions par statut.	• Intégration de la logique pour filtrer la liste des sessions (angular)
	5	En tant que gérant, je veux filtrer la liste des sessions par type de lavage.	• Intégration de la logique pour filtrer la liste des sessions (angular)

	6	En tant que gérant, je veux mettre à jour en temps réel le statut des sessions.	• Mise en place de la fonctionnalité de mise à jour du statut des sessions (angular) • Implémentation de l'API pour mettre à jour le statut des sessions (Spring) • Intégration de la logique pour mettre à jour le statut des sessions (angular) • Test de fonctionnalité avec Postman
Notifications	7	En tant qu'utilisateur, je veux être notifié lorsque mon tour de lavage approche, afin de pouvoir préparer mon véhicule à temps.	• Implémentation des mécanismes pour envoyer des notifications (Spring)
	8	En tant qu'utilisateur, je veux être notifié de la disponibilité de nouvelles places.	• Implémentation des mécanismes pour envoyer des notifications (Spring)

Statistiques	9	En tant que gérant, je veux voir des statistiques détaillées sur l'utilisation des stations de lavage.	• Création de la page de statistiques (angular) • Implémentation de l'API pour récupérer les statistiques (Spring) • Intégration de la logique pour afficher les statistiques (angular) • Test de fonctionnalité avec Postman
	10	En tant que gérant, je veux voir le nombre de sessions pour chaque station et chaque semaine.	• Implémentation de l'API pour récupérer le nombre de sessions (Spring) • Intégration de la logique pour afficher le nombre de sessions (angular) • Test de fonctionnalité avec Postman

	11	En tant que gérant, je veux identifier la station la plus fréquentée.	• Implémentation de l'API pour récupérer les données de fréquentation (Spring) • Intégration de la logique pour afficher les données de fréquentation (angular) • Test de fonctionnalité avec Postman
Supervision des utilisateurs et des gérants	12	En tant qu'administrateur, je veux consulter la liste des utilisateurs et des gérants.	• Création de la page de liste des utilisateurs et des gérants (angular) • Implémentation de l'API pour récupérer la liste des utilisateurs et des gérants (Spring) • Intégration de la logique pour afficher la liste des utilisateurs et des gérants (angular) • Test de fonctionnalité avec Postman

	13	En tant qu'administrateur, je veux désactiver ou activer les comptes des utilisateurs et des gérants.	• Mise en place de la fonctionnalité pour désactiver ou activer les comptes (angular) • Implémentation de l'API pour désactiver ou activer les comptes (Spring) • Intégration de la logique pour désactiver ou activer les comptes (angular) • Test de fonctionnalité avec Postman
	14	En tant qu'administrateur, je veux consulter la liste des stations de lavage gérées par chaque gérant.	• Implémentation de l'API pour récupérer la liste des stations par gérant (Spring) • Intégration de la logique pour afficher la liste des stations par gérant (angular) • Test de fonctionnalité avec Postman

	15	En tant qu'administrateur, je veux consulter la liste des sessions de chaque gérant.	• Implémentation de l'API pour récupérer la liste des sessions par gérant (Spring) • Intégration de la logique pour afficher la liste des sessions par gérant (angular) • Test de fonctionnalité avec Postman
	16	En tant qu'administrateur, je veux filtrer la liste des sessions par date.	• Intégration de la logique pour filtrer la liste des sessions (angular) • Test de fonctionnalité avec Postman
	17	En tant qu'administrateur, je veux filtrer la liste des sessions par statut.	• Intégration de la logique pour filtrer la liste des sessions (angular)
	18	En tant qu'administrateur, je veux filtrer la liste des sessions par type de lavage.	• Intégration de la logique pour filtrer la liste des sessions (angular)

2 Description fonctionnelle des histoires du Sprint 3

2.1 Diagramme de cas d'utilisation général du sprint 3

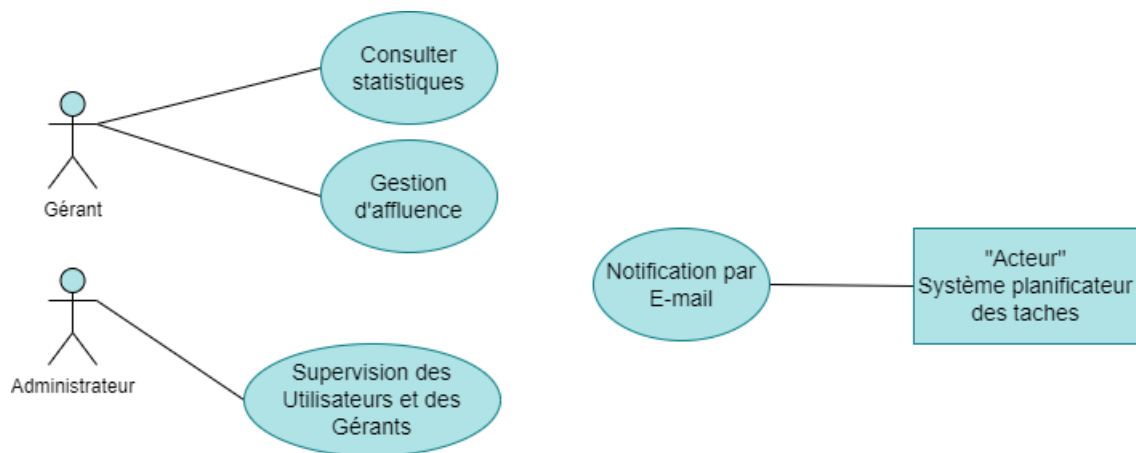


FIGURE 5.1 : Diagramme de cas d'utilisation général du sprint 3

2.2 Diagramme raffiné du cas d'utilisation « Gestion d'affluence »

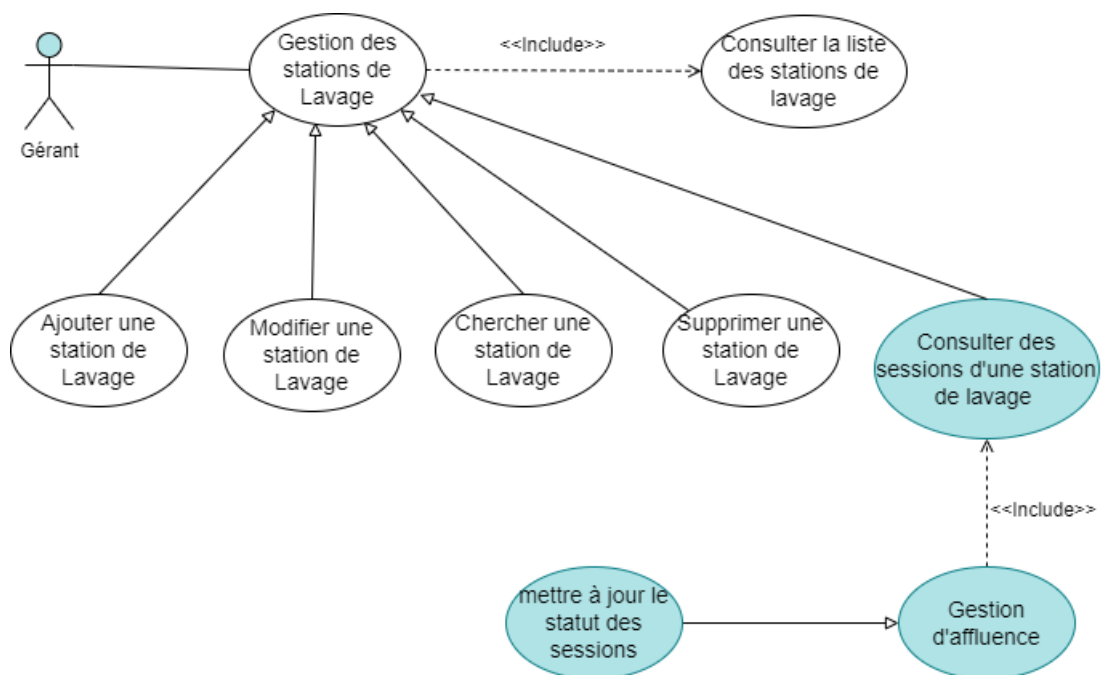


FIGURE 5.2 : Diagramme de cas d'utilisation « Gestion d'affluence »

2.3 Diagramme raffiné du cas d'utilisation « Supervision des utilisateurs et des gérants »

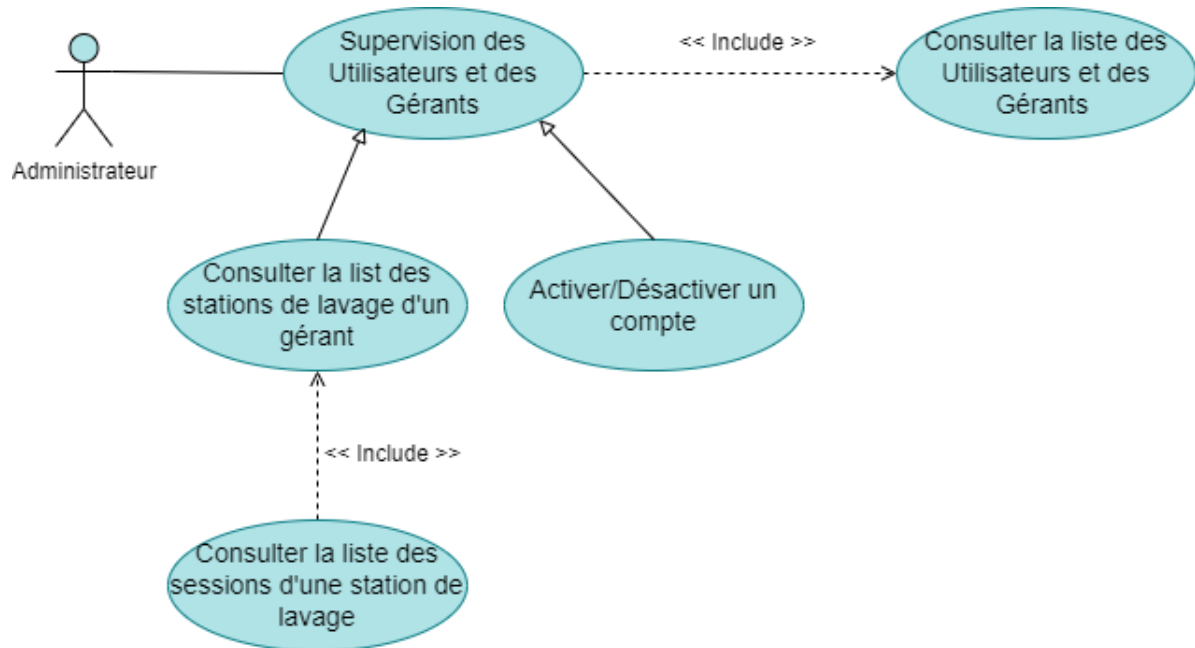


FIGURE 5.3 : Diagramme de cas d'utilisation « Supervision des utilisateurs et des gérants »

2.4 Description Textuelle de cas d'utilisation « Mettre à jour en temps réel le statut des sessions »

TABLEAU 5.2 : Description Textuelle de cas d'utilisation « Mettre à jour en temps réel le statut des sessions »

Titre	Mettre à jour en temps réel le statut des sessions
Acteurs	Gérant
Pré condition	• Gérant authentifié • Liste des sessions affichée
Scénario nominal	1. Le gérant sélectionne une session dans la liste. 2. Le gérant vérifie le statut actuel de la session. 3. Le gérant choisit de mettre à jour le statut de la session : - Si le statut actuel est « pending », le gérant peut le modifier en « in progress ». - Si le statut actuel est « in progress », le gérant peut le modifier en « completed ». 4. Le système enregistre la modification du statut. 5. Le statut mis à jour est reflété en temps réel dans la liste des sessions.

Scénario alternatif	Néant
Post condition	La liste des sessions affichée reflète le nouveau statut en temps réel.

2.5 Description Textuelle de cas d'utilisation « Être notifié lorsque un tour de lavage approche »

TABLEAU 5.3 : Description Textuelle de cas d'utilisation « Être notifié lorsque un tour de lavage approche »

Titre	Être notifié lorsque un tour de lavage approche
Acteurs	Utilisateur
Pré condition	• L'utilisateur authentifié. • L'utilisateur a réservé un créneau pour un lavage.

Scénario nominal	1. Le système surveille les créneaux de lavage réservés par les utilisateurs. 2. Lorsque le moment de lavage approche 15 minutes avant l'heure prévue), le système déclenche une notification. 3. La notification est envoyée à l'utilisateur par e-mail. 4. L'utilisateur reçoit la notification sur son appareil.
Scénario alternatif	Néant
Post condition	• L'utilisateur est informé de l'approche de son créneau de lavage. • Le système a enregistré que la notification a été envoyée.

2.6 Description Textuelle de cas d'utilisation « Filtrer la liste des sessions par date »

TABEAU 5.4 : Description Textuelle de cas d'utilisation « Filtrer la liste des sessions par date »

Titre	Filtrer la liste des sessions par date
Acteurs	Administrateur
Pré condition	• L'administrateur authentifié. • La liste des sessions est affichée.
Scénario nominal	1. L'administrateur accède à la liste des sessions. 2. L'administrateur sélectionne un champ de date. 3. L'administrateur applique le filtre. 4. Le système affiche les sessions qui correspondent à la date sélectionnée.
Scénario alternatif	4. A.1 -Si aucune session ne correspond aux critères de filtrage, le système affiche un message indiquant qu'aucune session n'a été trouvée pour cette date. 4. A.2 -Si l'administrateur souhaite annuler le filtre, il peut réinitialiser les critères de filtrage pour afficher de nouveau toutes les sessions.

Post condition

La liste des sessions est mise à jour pour afficher uniquement celles qui correspondent à la date sélectionnée.

3 Description comportementale des histoires du sprint 3

3.1 Diagramme de séquence « Désactiver/Activer les comptes »

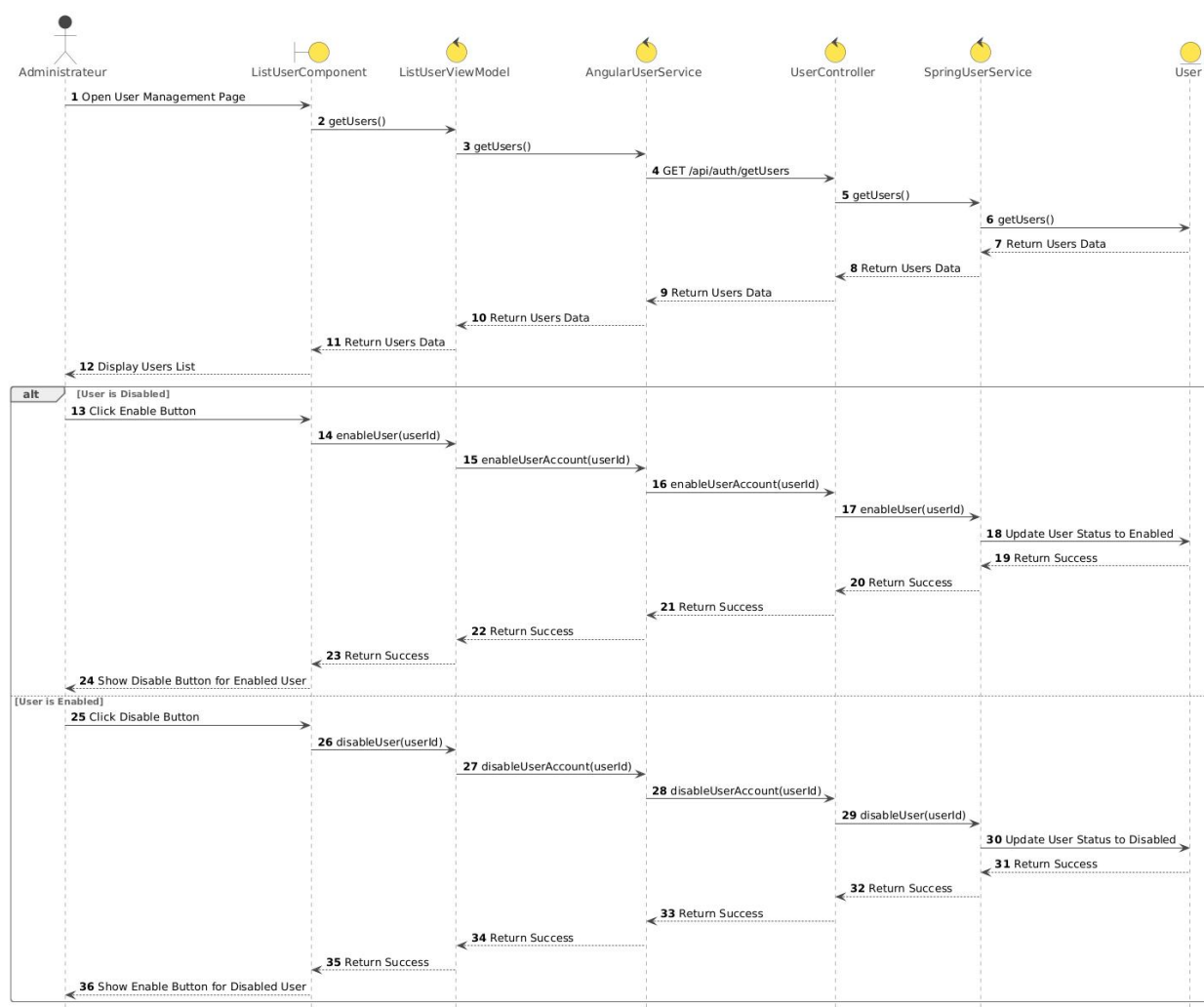


FIGURE 5.4 : Diagramme de séquence « Désactiver/Activer les comptes »

4 Description Structurale

4.1 Diagramme de classe global de sprint 3

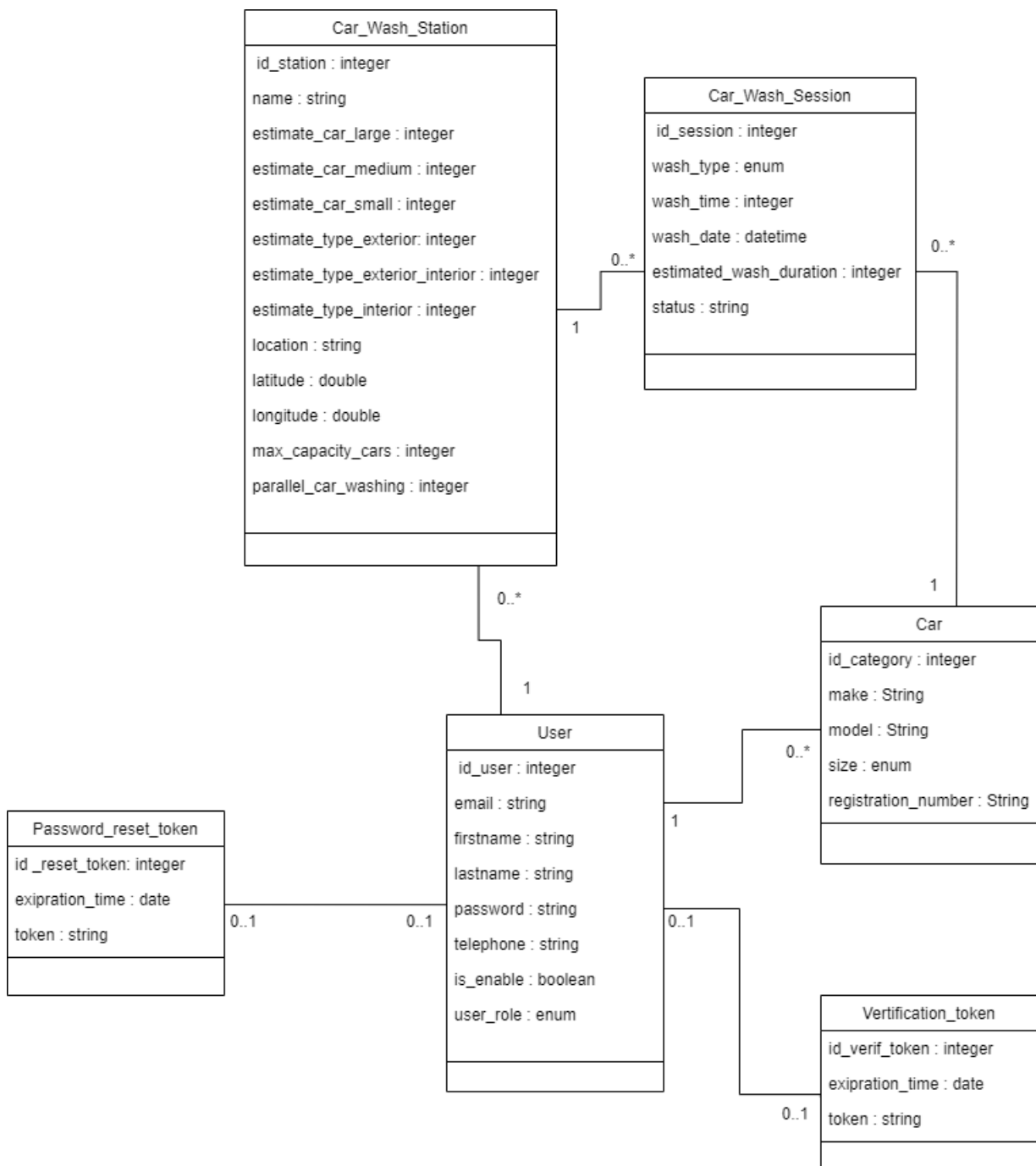


FIGURE 5.5 : Diagramme de classe global de sprint 3

5 Tests Postman

5.1 Test du cas d'utilisation « Afficher liste des utilisateurs et des managers »

Nous présentons ci-dessous le code source ainsi que le résultat des cas de test :

1. Vérification que la liste des utilisateurs et des managers contient exactement 5 éléments.
2. Vérification des champs présents pour chaque utilisateur et manager dans la liste.

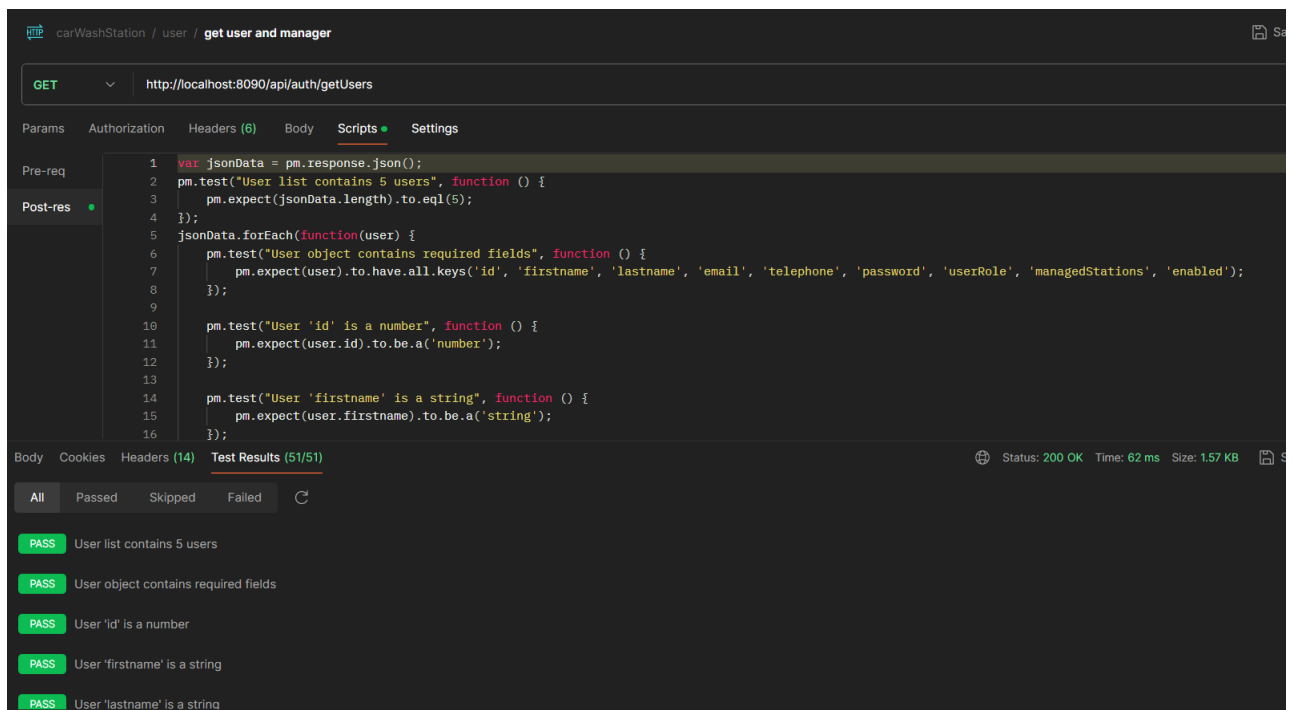


FIGURE 5.6 : Code source du test du cas d'utilisation « Afficher liste des utilisateurs et des manager »

5.2 Test du cas d'utilisation « Activer un compte »

Nous présentons ci-dessous le code source ainsi que le résultat du cas de test :

1. Vérification du message indiquant que le compte est déjà activé.

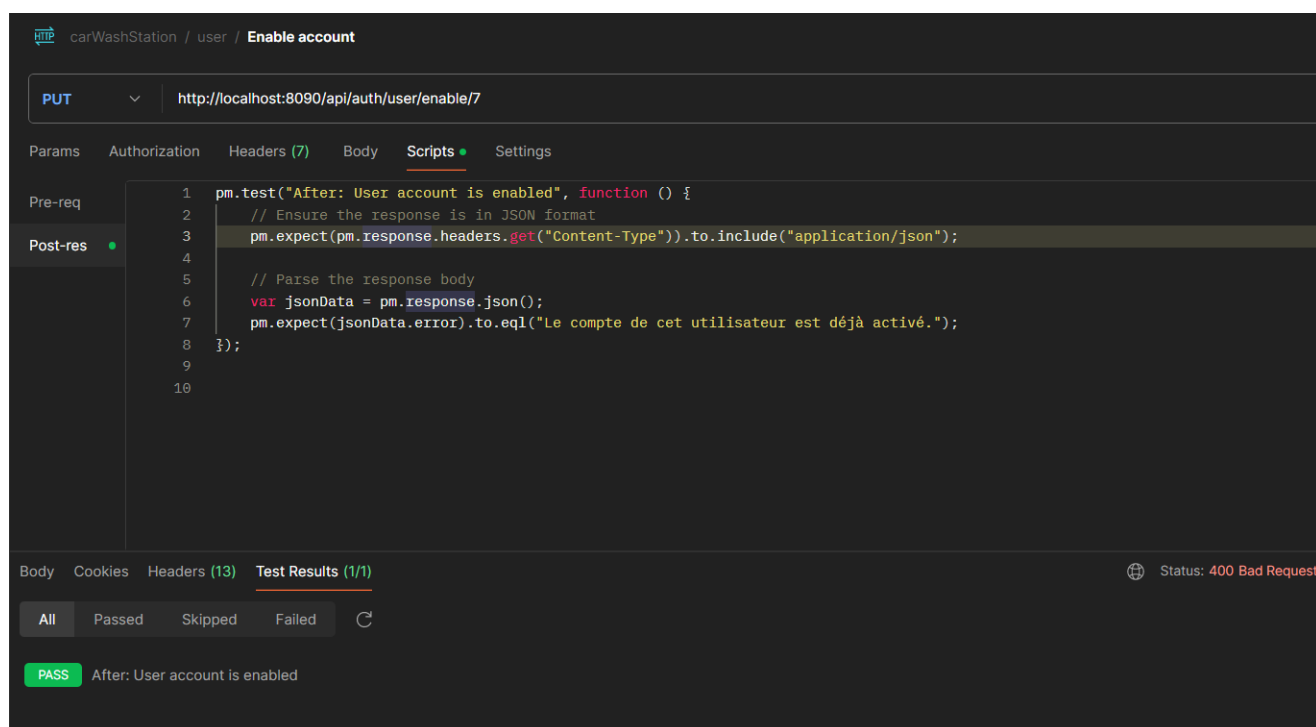


FIGURE 5.7 : Code source du test du cas d'utilisation « Activer un compte »

6 La revue

La figure 5.8 présente l'interface de gestion des réservations pour un utilisateur. Cette interface a été incluse dans le sprint 3, car nous avons finalisé la fonctionnalité d'estimation du temps. Le temps d'attente, ou temps estimé, indique la durée approximative nécessaire pour compléter le lavage.

Facteurs pris en compte : L'estimation du temps tient compte de plusieurs facteurs, tels que l'affluence actuelle et les caractéristiques de la voiture à laver (taille et type de lavage requis).

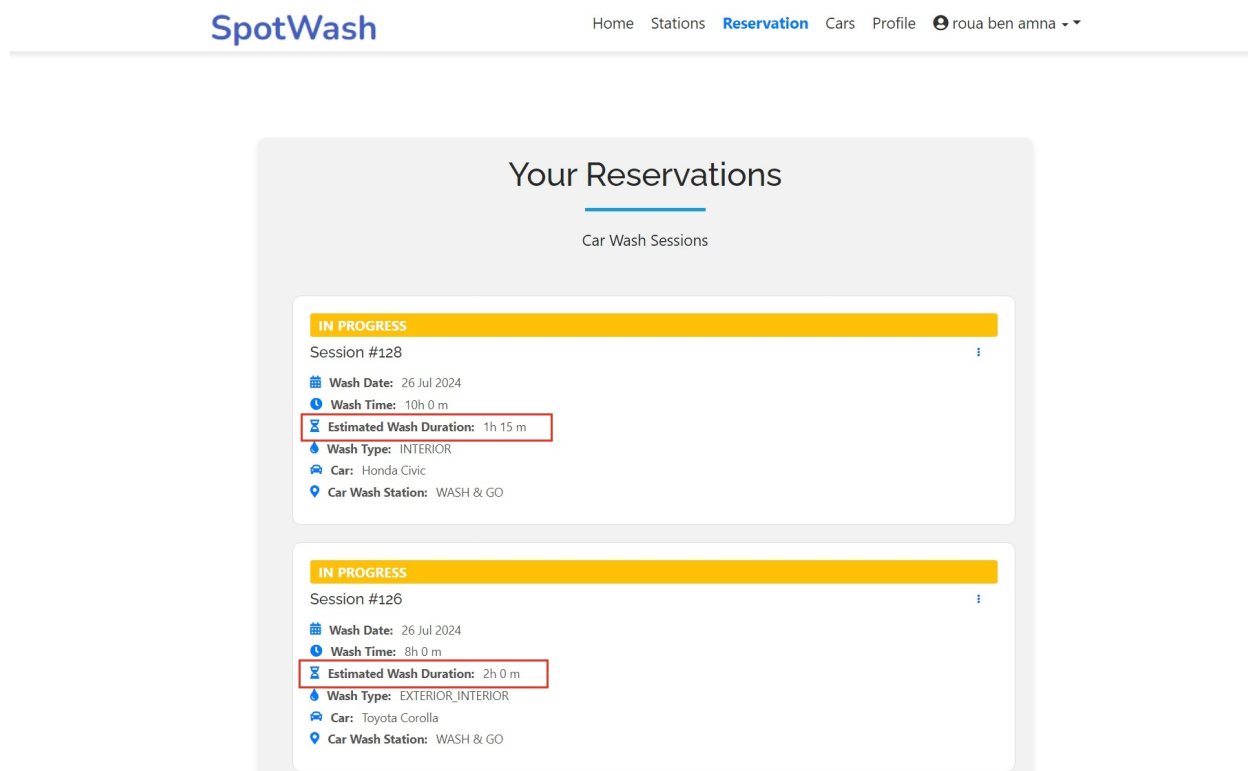


FIGURE 5.8 : SpotWash - Interface de list des réservation d'un utilisateur

- La figure 5.9 montre l'interface de la liste des sessions (ou des réservations), avec la possibilité de modifier les statuts.

The screenshot displays the 'Sessions List' page. On the left is a sidebar with navigation links: Dashboard, Station management, Profile, and Logout. The main area contains a table with the following data:

ID	Car Model	Car Registration Number	Size of Car	Wash Type	Wash Date	Estimated Wash Duration	Status	Actions
127	Honda Civic	XYZ5678	MEDIUM	INTERIOR	26 Jul 2024 At 8h 0m	1h 15m	Pending	[Edit] [Delete]
129	Honda Civic	XYZ5678	MEDIUM	INTERIOR	24 Jul 2024 At 16h 48m	1h 15m	In Progress	[Pause] [Delete]
126	Toyota Corolla	ABC1234	MEDIUM	EXTERIOR_INTERIOR	26 Jul 2024 At 8h 0m	2h 0m	In Progress	[Pause] [Delete]
128	Honda Civic	XYZ5678	MEDIUM	INTERIOR	26 Jul 2024 At 10h 0m	1h 15m	In Progress	[Pause] [Delete]
113	Toyota Corolla	ABC1234	MEDIUM	INTERIOR	2 Jul 2024 At 12h 46m	1h 15m	Completed	[Delete]
114	Toyota Corolla	ABC1234	MEDIUM	INTERIOR	2 Jul 2024 At 12h 46m	1h 15m	Completed	[Delete]
115	Toyota Corolla	ABC1234	MEDIUM	INTERIOR	2 Jul 2024 At 12h 47m	1h 15m	Completed	[Delete]
116	Toyota Camry	ABC1225	SMALL	INTERIOR	4 Jul 2024 At 8h 0m	1h 10m	Completed	[Delete]
117	Toyota Camry	ABC1225	SMALL	EXTERIOR_INTERIOR	4 Jul 2024 At 8h 0m	1h 55m	Completed	[Delete]
118	Toyota Corolla	ABC1234	MEDIUM	EXTERIOR_INTERIOR	4 Jul 2024 At 9h 55m	2h 0m	Completed	[Delete]

FIGURE 5.9 : SpotWash - Interface de list des sessions

- La figure 5.10 montre une interface filtrée selon le type de lavage.
D'autres filtres sont également disponibles pour le tableau, tels que par date, par statut et par type.

SpotWash

Dashboard

Station management

Profile

Logout

Sessions List

ID	Car Model	Car Registration Number	Size of Car	Wash Type	Wash Date	Estimated Wash Duration	Status	Actions
127	Honda Civic	XYZ5678	MEDIUM	All	8h 0m	1h 15m	Pending	
128	Honda Civic	XYZ5678	MEDIUM	INTERIOR	10h 0m	1h 15m	In Progress	
113	Toyota Corolla	ABC1234	MEDIUM	EXTERIOR_INTERIOR	2h 46m	1h 15m	Completed	
114	Toyota Corolla	ABC1234	MEDIUM	EXTERIOR	2 Jul 2024 At 12h 46m	1h 15m	Completed	
115	Toyota Corolla	ABC1234	MEDIUM	INTERIOR	2 Jul 2024 At 12h 47m	1h 15m	Completed	
116	Toyota Camry	ABC1225	SMALL	INTERIOR	4 Jul 2024 At 8h 0m	1h 10m	Completed	
123	Toyota Camry	ABC1225	SMALL	INTERIOR	21 Jul 2024 At 10h 0m	1h 10m	Completed	
129	Honda Civic	XYZ5678	MEDIUM	INTERIOR	24 Jul 2024 At 16h 48m	1h 15m	Completed	

FIGURE 5.10 : SpotWash - List des sessions filtrée

- La figure 5.11 montre l'email envoyé au utilisateur pour les rappeler 15 minutes avant la fin du lavage .

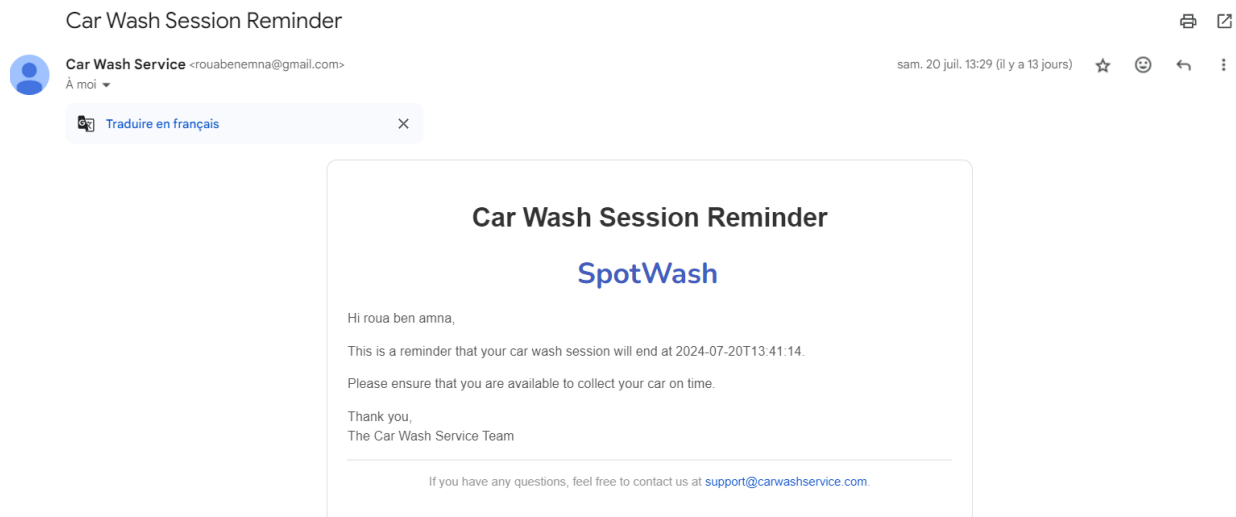


FIGURE 5.11 : SpotWash - Rappel Email pour la Complétion du Lavage

- La figure 5.12 montre l'interface des statistiques destinée au gérant.

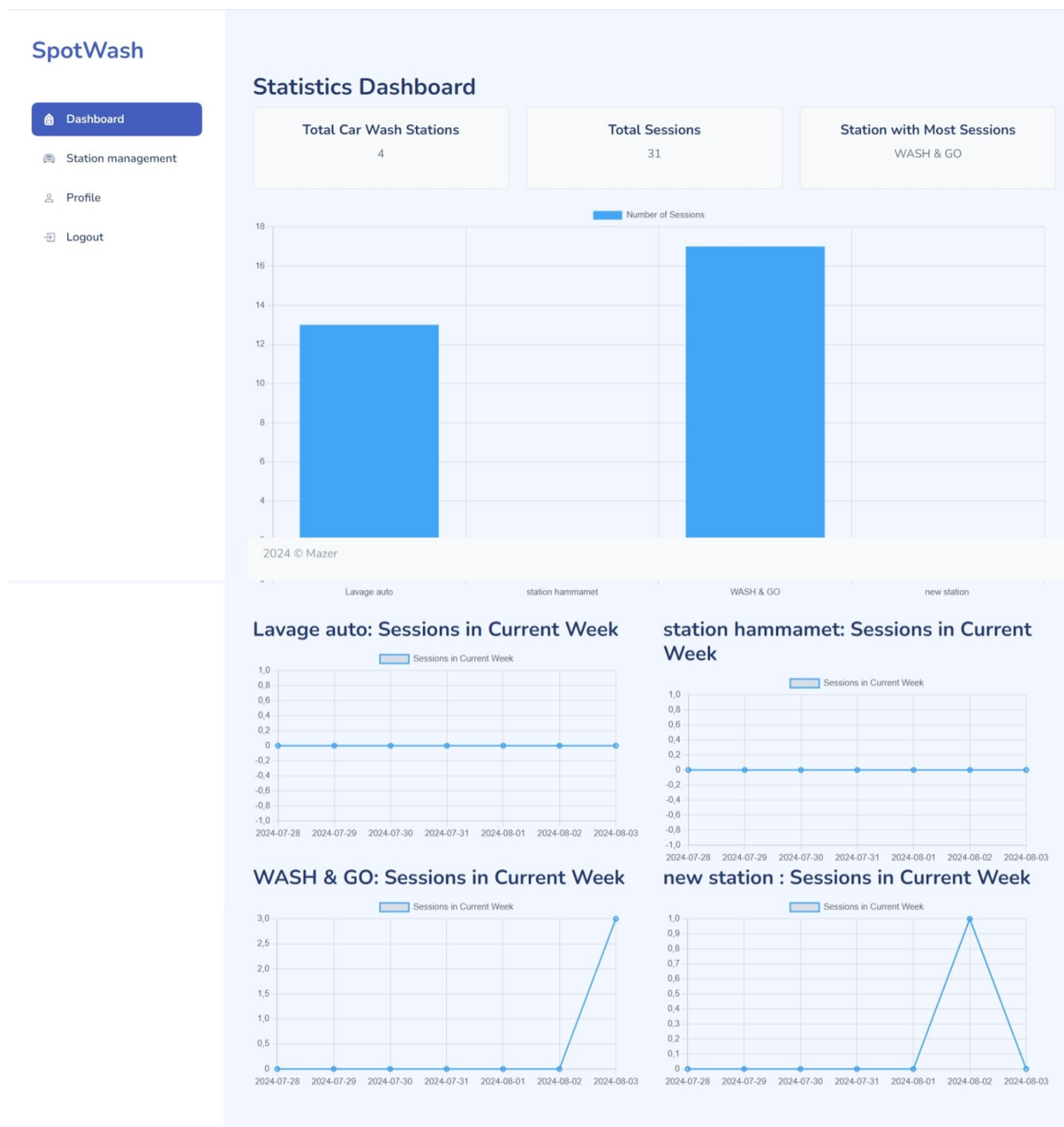


FIGURE 5.12 : SpotWash - Interface des statistiques

- La figure 5.13 montre la liste des utilisateurs et des gérants accessible à l'administrateur. Cette interface permet d'activer ou de désactiver les utilisateurs et de consulter les stations de lavage ainsi que les sessions associées à chaque station de lavage.

SpotWash

Users list

Profile

Logout

User Management

ID	First Name	Last Name	Email	Telephone	Role	Enabled	Actions
1	roua	ben amna	b.amnaroua@gmail.com	94369874	USER	Yes	
7	roua	ben emna	roua.benamna@etudiant-isi.utm.tn	22149314	USER	No	
9	roua	ben emna	rouabenemna@gmail.com	54149314	MANAGER	Yes	
10	anis	jaouadi	anissjaouadi@gmail.com	54149313	USER	Yes	

roua ben emna's Stations

Back

ID	Name	Location	Max Capacity	Sessions	Actions
8	Lavage auto	mourouj 3	6	13	
9	station hammamet	rue italy hammamet	10	0	
10	WASH & GO	mourouj 1	3	14	
12	new station	sousse	10	0	

Sessions for Lavage auto

Back

ID	Wash Date	Wash Type	Status
112	26 Jun 2024	EXTERIOR_INTERIOR	Completed
109	26 Jun 2024	EXTERIOR_INTERIOR	Completed
111	26 Jun 2024	EXTERIOR_INTERIOR	Completed
94	27 Jun 2024	EXTERIOR	Completed
95	27 Jun 2024	EXTERIOR_INTERIOR	Completed
102	27 Jun 2024	EXTERIOR_INTERIOR	Completed
105	27 Jun 2024	INTERIOR	Completed
106	27 Jun 2024	EXTERIOR	Completed
107	27 Jun 2024	INTERIOR	Completed
108	28 Jun 2024	EXTERIOR_INTERIOR	Completed
110	29 Jun 2024	INTERIOR	Completed
119	9 Jul 2024	EXTERIOR_INTERIOR	Completed
120	9 Jul 2024	INTERIOR	Completed

2024 © Mazer

FIGURE 5.13 : SpotWash - Interface d'administrateur

7 Rétrospective

7.1 Scrum Board

Le scrum board représente l'état de chaque sprint backlog item : s'il est à faire, en cours ou terminé.

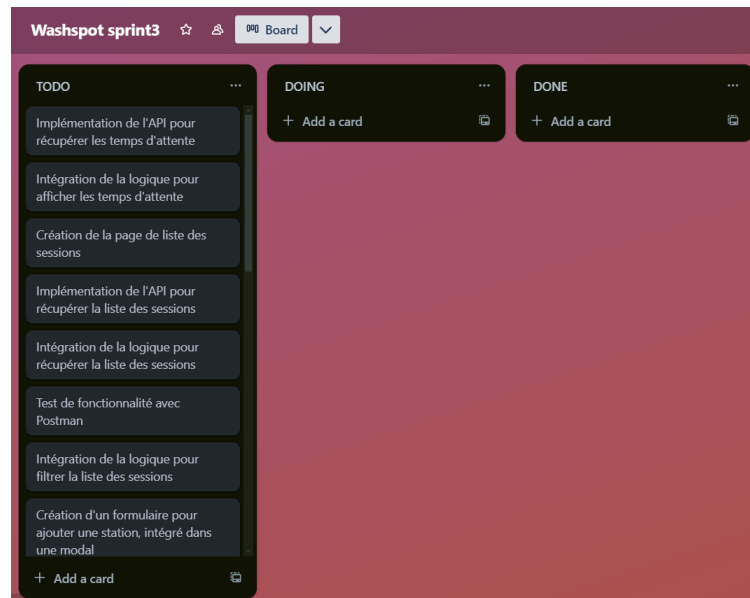


FIGURE 5.14 : Scrum board «jour 1»

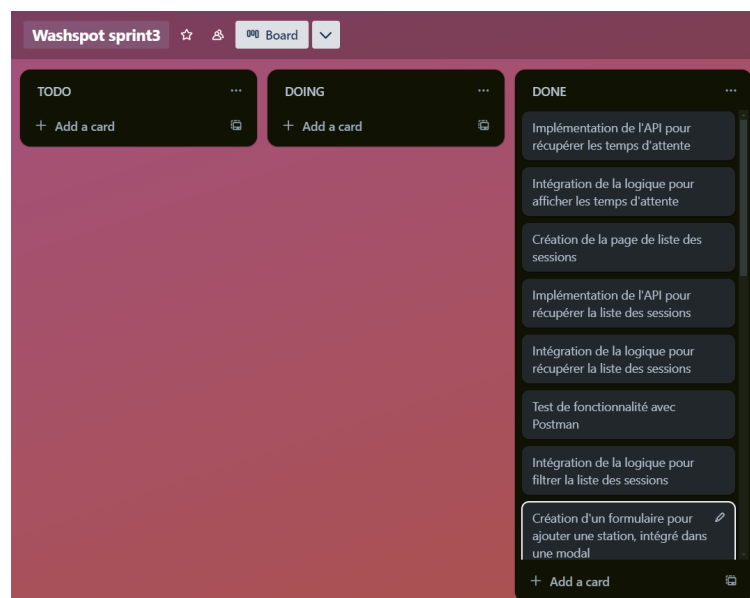


FIGURE 5.15 : Scrum board «dernier jour»

7.2 Tableau de Rétrospective

TABLEAU 5.5 : Tableau de rétrospective « Sprint 3 »

Questions	Réponses
Qu'est-ce qu'on a bien fait?	• La conception • Respecter le sprint goal • Le travail effectué est bien conforme au besoin • Bien maitriser les nouveaux outils de travail

Conclusion

À ce stade, notre projet d'études a atteint sa fin. Tout au long de ce chapitre, nous avons traité le troisième et dernier sprint, nous sommes passés par l'objectif, la conception et la réalisation.

CONCLUSION GÉNÉRALE

Gâce à l'application web "SpotWash" un problème bien spécifique a été résolu de manière innovante et efficace.

Dans ce projet, on a tenu compte de l'analyse approfondie de l'existant, de la définition des besoins et de la mise en place d'une architecture robuste. De plus, les choix technologiques, tels que Spring Boot pour le backend et Angular pour le frontend, ont joué un rôle crucial. En outre, la méthodologie Scrum était également importante pour assurer une gestion de projet rigoureuse, conduisant à un produit de haute qualité. Alors, on peut dire que ce stage a été très bénéfique. Tout d'abord, mes compétences techniques ont été améliorées non seulement car j'ai appris à utiliser des technologies telles que Spring Boot et Angular, mais j'ai également acquis des compétences en gestion de projet et en travail d'équipe en écrivant du code agile et en utilisant le cadre Scrum.

Pour l'avenir, il y a plusieurs domaines d'amélioration et de développement possibles dans le cadre de ce projet. En ce qui concerne le développement de nouvelles fonctionnalités, cela pourrait être le service de recommandations personnalisées pour les utilisateurs, qui sera basé sur leurs préférences et leur historique d'utilisation. En outre, les gérants auront la possibilité d'utiliser des algorithmes d'apprentissage automatique pour analyser les données collectées et obtenir des insights avancés sur les tendances et le comportement des utilisateurs. Enfin, il serait possible de mettre en œuvre des mécanismes d'obtention de feedback de la part des utilisateurs, qui peuvent être utilisés pour guider les futures améliorations et rendre l'application encore mieux adaptée à leurs besoins spécifiques.

En conclusion, je veux mentionner que ce projet m'a donné une opportunité sans précé-

dent de mettre mes connaissances théoriques en pratique et de contribuer à quelque chose d'utile et d'innovant.

WEBOGRAPHIE

- [1] Scrum.org. The scrum team. Available at <https://www.scrum.org/learning-series/what-is-scrum/the-scrum-team>, 08 mars 2022. Consulté le 21 juillet 2024.
- [2] jetbrains. IntelliJ idea overview. Available at <https://www.jetbrains.com/help/idea/discover-intellij-idea.html>, 04 juillet 2024. Consulté le 21 juillet 2024.
- [3] BDM. Découvrir visual studio code. Available at <https://www.blogdumoderateur.com/tools/visual-studio-code>, 16 juillet 2023. Consulté le 21 juillet 2024.
- [4] Hostinger. What is wamp. Available at <https://www.hostinger.com/tutorials/what-is-wamp>, 21 septembre 2022. Consulté le 21 juillet 2024.
- [5] draw.io. About draw.io. Available at <https://www.hostinger.com/tutorials/what-is-wamp>, 02 janvier 2021. Consulté le 21 juillet 2024.
- [6] Pierre Giraud. Présentation de git et de github. Available at <https://www.pierre-giraud.com/git-github-apprendre-cours/presentation-git-github/>, 11 février 2024. Consulté le 21 juillet 2024.
- [7] BDM. Découvrir trello. Available at <https://www.blogdumoderateur.com/tools/trello>, 05 décembre 2023. Consulté le 21 juillet 2024.
- [8] Agence 50A. Qu'est-ce que angular et à quoi sert-il? Available at <https://www.50a.fr/0/angular>, 04 juin 2024. Consulté le 21 juillet 2024.
- [9] IBM. Qu'est-ce que java spring boot? Available at <https://www.ibm.com/fr-fr/topics/java-spring-boot>, 10 mars 2024. Consulté le 21 juillet 2024.

- [10] Ionos. Qu'est-ce que mysql? Available at <https://www.ionos.fr/digitalguide/serveur/know-how/quest-ce-que-mysql/>, 18 janvier 2023. Consulté le 21 juillet 2024.
- [11] Red Hat. Une api rest, qu'est-ce que c'est? Available at <https://www.redhat.com/fr/topics/api/what-is-a-rest-api>, 02 février 2024. Consulté le 22 juillet 2024.
- [12] Deepakjose. Mvvm architecture in angular. Available at <https://deepakjosecodes.com/mvvm-architecture-in-angular-guide-with-code-examples/>, 02 juin 2023. Consulté le 22 juillet 2024.
- [13] Medium. Mastering mvc in java spring boot. Available at <https://medium.com/@TechiesSpot/mastering-mvc-in-java-spring-boot-a-comprehensive-guide>, 01 novembre 2023. Consulté le 23 juillet 2024.
- [14] Leafletjs. what's leaflet? Available at <https://leafletjs.com/>, 18 mai 2023. Consulté le 23 juillet 2024.
- [15] Tutorialspoint. Spring security - jwt. Available at https://www.tutorialspoint.com/spring_security/spring_security_with_jwt.htm#:~:text=JSON%20Web%20Token%20or%20JWT,public%20key%2Fprivate%20key%20pair., 06 avril 2023. Consulté le 23 juillet 2024.
- [16] Alexander S. Gillis. Smtip (simple mail transfer protocol). Available at <https://www.techtarget.com/whatis/definition/SMTP-Simple-Mail-Transfer-Protocol>, 03 novembre 2023. Consulté le 23 juillet 2024.
- [17] Monday blog. What is moscow prioritization? Available at <https://monday.com/blog/project-management/moscow-prioritization-method/>, 22 septembre 2022. Consulté le 01 aout 2024.
- [18] Derek Huether. Definition of done. Available at <https://www.leadingagile.com/2017/02/definition-of-done/>, 08 février 2017. Consulté le 29 juillet 2024.
- [19] IMT Atlantique. Le modèle relationnel. Available at https://formations.imt-atlantique.fr/bd_ihm/fr/intro_db/relational_model/#:~:text=Exemple%20%3A%20Consid%C3%A9rons%20les%20donn%C3%A9es%20des,sous%20forme%20d'une%20table., 03 mars 2023. Consulté le 04 aout 2024.
- [20] Atlassian. Comment réussir une rétrospective de sprint. Available at <https://www.atlassian.com/fr/team-playbook/plays/retrospective#:~:text=Qu'>

est%2Dce%20qu', lesquels%20vous%20pouvez%20vous%20am%C3%A9liorer., 21
février 2020. Consulté le 29 juillet 2024.

- [21] Tokay Torkut. Excelling in software development with scrum methodology. Available at <https://blog.stackademic.com/excelling-in-software-development-with-scrum-methodology-part-1-762a73c6a6ec>, 25 août 2023. Consulté le 04 septembre 2024.

