

Projet de fin d'études

N° d'ordre:

MEMOIRE

Présenté à

Institut Supérieur d'Informatique de Mahdia

En vue de l'obtention

**De la Licence en Ingénierie des Systèmes Informatiques
Spécialité : Systèmes Embarqués et Internet des Objets**

par

**Mayoufi Amine
Jmai Wiem**

**Système à base de drones connectés et autonomes
pour la surveillance et l'alerte des incendies**

Soutenu le 15/06/2023, devant la commission d'examen :

Mr. Anis Haj Said

Président

Mr. Aloulou Hamdi

Encadrant

Mme Htit Zahra

Examinateur

Remerciements

C'est avec un grand plaisir que nous réservons ces lignes en signe de gratitude et de reconnaissance à tous ceux qui ont contribué de près ou de loin à l'élaboration de ce travail. Nous commençons par remercier nos chers parents Saida et Najib, Laila et Moktar qui ont toujours été là pour nous, qui ont tout sacrifié pour leurs enfants, n'épargnant ni santé ni efforts et qui nous avaient donné un magnifique modèle de labeur et de persévérance. Nous sommes redevables d'une éducation dont nous sommes fiers.

Ainsi nos sincères remerciements à notre encadrante à L'institut supérieure d'informatique de Mahdia (ISIMA), Monsieur Hamdi Aloulou pour avoir accepté de nous diriger patiemment, pour son soutien constant et pour sa présence et sa disponibilité tout au long de ce stage.

Nos vifs remerciements accompagnés de toute notre gratitude s'adressent également à Monsieur Ilyes Tlili qui a gardé un œil attentif sur le déroulement et l'avancement de notre projet, en ayant toujours des remarques très constructives ainsi que pour la marque de confiance qu'il nous a attribuée. Nous aimerons témoigner du plaisir qu'était pour nous de travailler sous leurs directives.

Nous exprimons également notre gratitude à notre directrice, Mme Nabila Torki, ainsi qu'à Mme Rim Bedoui, pour leur précieuse aide lors des démarches administratives liées à l'autorisation d'utilisation des drones. Leur soutien et leur expertise ont été d'une grande importance, et nous les remercions chaleureusement pour leur contribution, nous ne manquerons pas d'exprimer notre grande reconnaissance à tous les enseignants et administrateurs de l'ISIMA et tous les membres du jury pour avoir accepté d'évaluer notre travail.

Dédicaces

Jmai Wiem

Je dédie cette œuvre

À mes précieux parents

Je dédie ce travail à mes chers parents, Najib et Saida, qui sont la prunelle de mes yeux. Leur amour inconditionnel, leur tendresse infinie et leurs sacrifices incessants méritent d'être honorés.

Ma chère maman, Saida

Je tiens à vous exprimer toute l'ampleur de mon amour, de mon attachement et de ma plus profonde considération envers vous. Je suis extrêmement fière d'être votre fille et de pouvoir enfin réaliser ce que vous avez tant espéré et attendu de moi

Vous n'avez jamais cessé de déployer des efforts considérables pour subvenir à nos besoins, nous encourager et nous aider à choisir le chemin de la réussite. Votre patience infinie, votre bonne volonté, vos précieux conseils et votre confiance en moi ont joué un rôle primordial dans ma réussite.

Maman, je t'aime profondément et je suis immensément reconnaissante pour tout ce que tu as fait et continues de faire pour moi. Ta présence bienveillante et ton soutien indéfectible sont des piliers essentiels dans ma vie. Tu es une source constante d'inspiration et de force.

Je t'aime du plus profond de mon cœur.

Mon cher père, Najib

À l'homme qui illumine ma vie de son amour inconditionnel, mon guide éternel, mon roc indéfectible, et ma source constante de joie et de bonheur, à toi, mon père bien-aimé, qui n'a jamais ménagé ses efforts pour me voir triompher

Ma chérie tante Thouraya,

À ma chère tante, qui occupe une place si spéciale dans mon cœur, tu as été bien plus qu'une tante pour moi, tu as été comme une seconde mère. Ta bienveillance, ton amour inconditionnel et tes conseils précieux ont guidé mes pas et ont façonné la personne que je suis aujourd'hui. Je te suis éternellement reconnaissante pour ta présence aimante dans ma vie.

Le meilleur frère du monde, Wael,

La vie m'a offert de nombreux amis que j'aime profondément, mais toi, tu es le seul à partager mon sang, et nous avons un lien unique qui ne s'estompera jamais. Je suis reconnaissant(e) pour tout ce que tu fais, mon cher frère Wael. Je ne peux imaginer ma vie sans toi à mes côtés. Ta présence est un soutien inestimable et ton amour inconditionnel est une source de force et de bonheur pour moi. Merci d'être toujours là pour moi.

À mes chers Amine, Nada, Younes, Oumaima, Ranya, Nermine, Yuma, et Ahmed

pour leur aide et support dans les moments difficiles.

À tous mes amies qui m'ont toujours encouragé, et à qui je souhaite plus de succès.

A toute ma famille,

À tous ceux que j'aime et ceux qui m'aiment.

J'espère que ce projet soit le début de réalisation de mes rêves.

Dédicaces

Amine Mayoufi

Je dédie ce travail

À mes chers parents

Je dédie ce travail à la prunelle de mes yeux, mes très chers parents Moktar et Laila, pour tout leur amour, tendresse et sacrifices.

Mon cher père, Moktar

Je te suis profondément reconnaissant pour ton soutien et ton amour inconditionnel. Tu es mon modèle et ma source d'inspiration. Merci d'avoir toujours été là pour moi. .

Ma chère maman, Laila

Ta tendresse et ton dévouement ont façonné la personne que je suis aujourd'hui. Merci pour ton amour infini et ta présence réconfortante dans ma vie.

À Mes Adorables Sœurs,

Houda, Imen et Houyem, vous êtes mes complices et mes amies les plus précieuses.

Merci pour nos moments de rire et de soutien mutuel.

Votre amour et vos conseils ont laissé une empreinte indélébile dans mon cœur. Je vous suis reconnaissant pour tous les précieux souvenirs que nous avons partagés.

À tous mes amis

Ceux qui m'ont toujours souhaité le meilleur. Je leur désire la réussite dans leur vie.

Taif, Amine, Ali, Weam, Nada, Inaam, Ahmed, Youssef, Rihem et Aya, je vous remercie du fond du cœur. Votre amitié et votre soutien m'ont aidé à surmonter les difficultés et à partager des moments de joie et de complicité.

Enfin,

Je souhaite également dédier ce travail à mes grands-parents

Mohamed et Mahbouba,

À ma cousine Fatouma et son mari Riadh, pour leur soutien et leur accueil chaleureux à Mahdia. Leur hospitalité m'a fait me sentir chez moi. Je pense également à leurs adorables enfants, Adem, Youssef et Younes.

Table des matières

Liste des figures	xii
Liste des tableaux	xviii
Introduction générale	1
1 Étude de projet	4
1.1 Introduction	4
1.2 Champ d'application	4
1.3 Présentation de l'organisme d'accueil	4
1.3.1 Pépinière d'entreprises Mahdia : CITY Lab	4
1.3.2 Quetratech	5
1.4 Présentation du projet	5
1.4.1 Problématique	5
1.4.2 Objectifs du projet	6
1.5 Étude et critique de L'existant	6
1.5.1 Étude de l'existant	7
1.5.1.1 OroraTech	7
1.5.1.2 Pano wildfire detection stations	7
1.5.1.3 Crop-Care	8
1.5.2 Critique de l'existant	8
1.6 Solution Proposée	9
1.7 Méthodologie adoptée	11
1.7.1 Méthodologie retenue : Méthodologie agile	11

1.7.2	Kanban	12
1.7.3	Langage de conception	13
1.8	Conclusion	13
2	Sprint 0 : Phase préparatoire	14
2.1	Introduction	14
2.2	Analyse et spécifications des besoins	14
2.2.1	Identification des acteurs	14
2.2.2	Besoins Fonctionnels	15
2.2.2.1	Besoins Fonctionnels de Super Administrateur	16
2.2.2.2	Besoins Fonctionnelles de l'Administrateur	17
2.2.2.3	Besoins Fonctionnelles du client	18
2.2.3	Besoins non fonctionnels	19
2.3	Gestion de projet avec Kanban	19
2.4	Planification des sprints	21
2.5	Environnement logiciel	21
2.6	Architecture générale	28
2.7	Analyse	29
2.8	Motif d'architecture logicielle	31
2.9	Conclusion	31
3	Release 1 : Application Mobile SaviaBot	32
3.1	Introduction	32
3.2	SPRINT 1 : “Gestion des comptes”	32
3.2.1	Backlog du SPRINT 1	33
3.2.2	Analyse	33
3.2.2.1	Diagramme de cas d'utilisation	34
3.2.2.2	Description textuelle des cas d'utilisation	34
3.2.3	Conception	41
3.2.3.1	Diagrammes de séquence	41
3.2.3.2	Diagramme de Classes	50

3.2.4	Réalisation	51
3.2.5	Validation	54
3.3	SPRINT 2 : “Planification des missions et avis clients”	60
3.3.1	Backlog du SPRINT 2	60
3.3.2	Analyse	60
3.3.2.1	Diagramme de cas d’utilisation	61
3.3.2.2	Description textuelle de cas d’utilisation	61
3.3.3	Conception	65
3.3.3.1	Diagramme de séquence :	65
3.3.3.2	Diagramme de Classes	72
3.3.4	Réalisation	73
3.3.5	Validation	74
3.4	Conclusion	78
4	Release 2 : Réalisation de drone SaviaHawk	79
4.1	Introduction	79
4.2	SPRINT 3 : “Assemblage et navigation du drone”	79
4.2.1	Backlog du Sprint 3	80
4.2.2	Choix des composants du drone SaviaHawk	80
4.2.2.1	Microcontrôleur	80
4.2.2.2	Capteurs	81
4.2.2.3	Réseaux	83
4.2.2.4	Actionneur	83
4.2.2.5	Alimentation et circuit	84
4.2.2.6	Accessoire	85
4.2.3	Création du circuit final et assemblage	87
4.2.3.1	Circuit Générale	87
4.2.3.2	Assemblage	88
4.2.4	Drone : Simulation et Navigation	91
4.2.4.1	Approche Mathématique et Analyse des Composants . .	91

4.2.4.2	Calibrage de drone et simulation de Station	93
4.2.4.3	Communication Raspberry/PixHawk et Navigation	96
4.3	SPRINT 4 : “Détection et alertes précoce des incendies”	98
4.3.1	Backlog du sprint 4	98
4.3.2	Intégration des capteurs de détection d’incendie	98
4.3.2.1	Capteurs de substances gazeuses et de qualité de l’air (MQ135 et MQ2)	98
4.3.2.2	Caméra thermique AMG8833	99
4.3.2.3	Caméra Raspberry 5MP	99
4.3.2.4	Fusion des capteurs	100
4.3.3	Développement des algorithmes de détection	101
4.3.3.1	Dataset	101
4.3.3.2	YOLOv5	101
4.3.4	Réalisation de la fonctionnalité “Surveiller en temps réel l’état des cultures et les zones vertes ” et " Recevoir les alertes ”	104
4.3.4.1	Analyse	104
4.3.4.2	Conception	106
4.3.4.3	Réalisation	109
4.3.4.4	Validation	111
4.3.5	Enregistrer les données de détection d’incendie pour analyse ultérieure	112
4.4	Conclusion	113
5	Release 3 : Amélioration continue de la détection	114
5.1	Introduction	114
5.2	SPRINT 5 : “Surveillance et amélioration du système de détection du drone”	114
5.2.1	Backlog du Sprint 5	115
5.2.2	Analyse	115
5.2.3	Conception	118
5.2.4	Réalisation	122
5.2.5	Validation	124

5.3 Amélioration des algorithmes de détection d'incendies	126
5.4 Conclusion	127
Conclusion générale	128
Webographie	131

Table des figures

1.1	Logo de Pépinière d'entreprises et CITYLAB Mahdia	5
1.2	Logo Quetratech	5
1.3	Plateforme WILDFIRE d'OroraTech	7
1.4	Station de détection Pano AI	8
1.5	Solution Crop-Care de Robocare	8
1.6	L'écosystème internet des objets	11
1.7	Tableau kanban	12
1.8	Unified Modeling Language	13
2.1	Logo VS Code	22
2.2	Logo Flutter et Dart	22
2.3	Logo Android Studio	22
2.4	Logo Firebase	22
2.5	Logos Google Cloud Platform	23
2.6	Logo Figma	23
2.7	Logo RasbianOS	23
2.8	Logo Python	24
2.9	Logo Geany	24
2.10	Logo Notepad++	24
2.11	Logo QGroundControl	25
2.12	Logo YOLOv5	25
2.13	Logo librairie OpenCV	25
2.14	Logo bibliothèque Pytorch	25

2.15 Logo Github	26
2.16 Logo Draw.io	26
2.17 Logo Tinkercard	26
2.18 Logo Creality	26
2.19 Tableau Trello : Quetratech Intern	27
2.20 Architecture IoT	28
2.21 Diagramme de cas d'utilisation global	30
2.22 Model-View-Controller (MVC) [9]	31
 3.1 Sprint 1- Diagramme de cas d'utilisation	34
3.2 Sprint 1 -Diagramme de séquence “s’inscrire”	42
3.3 Sprint 1 -Diagramme de séquence “s’authentifier”	43
3.4 Sprint 1 -Diagramme de séquence “Mettre à jour son profil”	44
3.5 Sprint 1 -Diagramme de séquence “Changer mot de passe”	45
3.6 Sprint 1 -Diagramme de séquence “modifier espace”	46
3.7 Sprint 1 -Diagramme de séquence “Gérer les comptes d’administrateurs” .	47
3.8 Sprint 1 -Diagramme de séquence “Ajouter admin”	48
3.9 Sprint 1 -Diagramme de séquence “Supprimer admin”	49
3.10 Sprint 1 -Diagramme de séquence “Ajouter collaborateur”	50
3.11 Sprint 1 -Diagramme de Classes	51
3.12 Sprint 1 -Base de données No SQL Firestore ”	52
3.13 Sprint 1 -Intégration de la base de données à l’application SaviaBot” . .	52
3.14 Sprint 1 -Base de donnée No SQL Firestore	52
3.15 Sprint 1 -Initialisation de l’API Google Maps	53
3.16 Sprint 1 -Intégration de l’API Google Maps dans l’application SaviaBot .	53
3.17 Sprint 1 -Développement de Widget Login (Authentification)	53
3.18 Sprint 1-Interface d’inscription	54
3.19 Sprint 1-Interface d’inscription	54
3.20 Sprint 1 - interface HomePage Client	55
3.21 Sprint 1 – interface Profil Settings	56

3.22 Sprint 1 - interface Edit Profile	56
3.23 Sprint 1 – interface Edit Password	57
3.24 Sprint 1 – interface Edit Space	57
3.25 Sprint 1 - interface Home Super Administrateur	58
3.26 Sprint 1 - interface Management	58
3.27 Sprint 1 - interface Ajouter Admin	58
3.28 Sprint 1 - Interface Supprimer Admin	59
3.29 Sprint 1 - Interface Ajouter Collaborateur	59
3.30 Sprint 2 -Diagramme de Cas d'utilisation	61
3.31 Sprint 2 - Diagramme de séquence de cas d'utilisation “Accéder aux commentaires des clients”	66
3.32 Sprint 2 -Diagramme de séquence “Afficher les données des admins”	67
3.33 Sprint 2 -Diagramme de séquence “Consulter les demandes des clients”	68
3.34 Sprint 2 -Diagramme de séquence “Planifier les missions des drones et confirmer les demandes des clients”	69
3.35 Sprint 2 -Diagramme de séquence “Suivre l'état des drones”	70
3.36 Sprint 2 -Diagramme de séquence “Sélectionner son espace de surveillance”	71
3.37 Sprint 2 -Diagramme de séquence “Donner des commentaires sur les services qu'il a reçus”	72
3.38 Sprint 2 -Diagramme de Classes	73
3.39 Sprint 2 -Initialisation de l'API MapBox	74
3.40 Sprint 2 -Intégration de l'API MapBox dans l'application SaviaBot	74
3.41 Sprint 2 - Interface Accéder aux commentaires des clients	74
3.42 Sprint 2 - Interface Afficher les données des admins des admins.	75
3.43 Sprint 2 - Interface Dashboard AdminHome	75
3.44 Sprint 2 - Interface Pending Clients AdminHome	75
3.45 Sprint 2 - Interface Supprimer Admin	76
3.46 Sprint 2 - Section Suivre l'état des drones	76
3.47 Sprint 2 - Interface Sélectionner son espace de surveillance	77
3.48 Sprint 2 - Interface Donner des commentaires sur les services qu'il a reçus	77

4.1	Carte de développement Raspberry Pi 4	80
4.2	Carte de contrôle de vol Pixhawk 2.4.8	81
4.3	Caméra thermique IR AMG 8833	81
4.4	Caméra Raspberry Pi OmniVision OV5647	82
4.5	Capteur de gaz inflammables et de fumée MQ2	82
4.6	Capteur de gaz nocifs MQ135	82
4.7	Module GPS NEO-M8N	83
4.8	Télémesure radio 433 MHz	83
4.9	Contrôleur de vitesse ESC Brushless OPTO 40A	83
4.10	Les 4 Moteurs DJI 2212 920KV	84
4.11	La batterie Dupu Lipo 4s 20c	84
4.12	Module d'alimentation APM	84
4.13	Convertisseur 5V 3a	85
4.14	Plateforme de drone F450	85
4.15	Emetteur des signaux sonore PixHawk	85
4.16	Hélices du drone 1045 Pitch CW/CCW	86
4.17	Circuit Générale de SaviaHawk	87
4.18	Soudage des composentes	88
4.19	Portage des moteurs et ESC	88
4.20	Portage des moteurs et ESC	89
4.21	Améliorations 3D	89
4.22	Couvercle SaviaHawk	90
4.23	Prototype Final SaviaHawk v0	90
4.24	Calibrage du Boussole et Accéléromètre	94
4.25	Calibrage du Gyroscope et Niveau d'horizon	94
4.26	Calibration de la batterie et des ESC	94
4.27	Calibration du contrôleur	95
4.28	Modes de vol	95
4.29	API OpenWeatherMap	96
4.30	Station météorologique Simulée	96

4.31 L'armement du drone SaviaHawk	97
4.32 Installation des capteurs MQ135	99
4.33 Câblage I2C caméra thermique AMG8833	99
4.34 tests du capteur AMG 8833	99
4.35 Installation de Caméra Raspberry	100
4.36 tests du capteur AMG 8833	100
4.37 Exemple de Dataset	102
4.38 Réseau de neurones convolutifs	102
4.39 Graphes d'évolution, de perte (loss) et de précision (accuracy)	103
4.40 Tests et exécution Detect.py	103
4.41 Sprint 4-Diagramme de cas d'utilisation	104
4.42 Sprint 4 -Diagramme de séquence “Surveiller en temps réel l'état des cultures et les zones vertes”	107
4.43 Sprint 4 -Diagramme de séquence “recevoir les alertes”	108
4.44 Sprint 4 -Diagramme de Calssses	109
4.45 Sprint 4- Interface Client incluant la surveillance temps réel	110
4.46 Sprint 4 - Real-time Database update	110
4.47 Sprint 4 -Initialisation de Firebase Cloud Messaging	110
4.48 Sprint 4 - Interface Client incluant la surveillance temps réel	111
4.49 Sprint 4 - Test de l'envoi d'alerte	111
4.50 collection "Détection" - Firestore	112
4.51 Vidéo de Détection - Google Cloud Storage	113
5.1 Diagramme de cas d'utilisation du sprint 5	116
5.2 Sprint 5 -Diagramme de séquence “Explorer les données collectées par les drones”	119
5.3 Sprint 5- Diagramme de séquence “Recevoir les alertes de détection d'incendies à un stade précoce”	120
5.4 Sprint 5 -Diagramme de séquence “Confirmer l'incendie”	121
5.5 Sprint 5 -Diagramme de Classes	122
5.6 Sprint 5 -Initialisation de l'API SMS Twilio	123

5.7	Sprint 5 -FCM Auth Token	123
5.8	Sprint 5 -Push Notification + SMS	123
5.9	Sprint 5 - Explorer les données collectées par les drones	124
5.10	Sprint 5 - Recevoir les alertes de détection d'incendies à un stade précoce	125
5.11	Sprint 5 – Interface Historique des détections	125
5.12	saviahawk-capture google cloud bucket	126

Liste des tableaux

1.1	<i>Critique de l'existant</i>	9
2.1	Identification des acteurs	15
2.2	<i>Besoins Fonctionnels de Super Administrateur</i>	16
2.3	<i>Besoins Fonctionnelles de l'administrateur</i>	17
2.4	<i>Besoins Fonctionnelles du client</i>	18
2.5	<i>Backlog du produit</i>	20
2.6	<i>Planification des sprints</i>	21
3.1	Backlog du sprint 1 : "Gestion des comptes"	33
3.2	Description textuelle du scénario de cas d'utilisation « S'inscrire »	35
3.3	Description textuelle du scénario de cas d'utilisation « S'authentifier »	36
3.4	Description textuelle du scénario de cas d'utilisation « Mettre à jour son profil »	37
3.5	Description textuelle du scénario de cas d'utilisation « Changer mot de passe »	38
3.6	Description textuelle du scénario de cas d'utilisation « Modifier l'espace »	39
3.7	Description textuelle du scénario de cas d'utilisation « Gérer les comptes d'administrateurs »	39
3.8	Description textuelle du scénario de cas d'utilisation « Ajouter Admin »	40
3.9	Description textuelle du scénario de cas d'utilisation « Supprimer Admin »	40
3.10	Description textuelle du scénario de cas d'utilisation « Ajouter collaborateur »	41
3.11	Backlog du sprint 2 : "Planification des missions et avis clients"	60

3.12	Description textuelle du scénario de cas d'utilisation « Accéder aux commentaires des clients »	62
3.13	Description textuelle du scénario de cas d'utilisation « Afficher les données des admins »	62
3.14	Description textuelle du scénario de cas d'utilisation « Consulter les demandes des clients »	63
3.15	Description textuelle du scénario de cas d'utilisation « Planifier les missions des drones et confirmer les demandes des clients »	63
3.16	Description textuelle du scénario de cas d'utilisation « Suivre l'état des drones »	64
3.17	Description textuelle du scénario de cas d'utilisation « Sélectionner son espace de surveillance »	64
3.18	Description textuelle du scénario de cas d'utilisation « Donner des commentaires sur les services qu'il a reçus »	65
4.1	Backlog du sprint 3 : "Assemblage et navigation du drone"	80
4.2	Backlog du sprint 4 : "Détection et alertes précoce des incendies"	98
4.3	Description textuelle du scénario de cas d'utilisation "Surveiller en temps réel l'état des cultures et les zones vertes"	105
4.4	Description textuelle du scénario de cas d'utilisation "Recevoir les alertes"	106
5.1	Backlog du sprint 5 : "Surveillance et amélioration du système de détection du drone"	115
5.2	Description textuelle du scénario de cas d'utilisation "Explorer les données collectées par les drones"	116
5.3	Description textuelle du scénario de cas d'utilisation "Recevoir les alertes de détection d'incendies à un stade précoce"	117
5.4	Description textuelle du scénario de cas d'utilisation "Confirmer l'incendie"	118

Glossaire

AI : Intelligence Artificiel

API : Application Programming Interface

GPU : graphical processing unit

SMS : Short Message Service

IoT : Internet of Things

LoRa : Long Range

ML : Machine Learning

CNN : Convolutional Neural Networks

SDK : Software Development Kit

IDE : Integrated development environment

UML : Unified Modeling Language

LiPo : lithium polymer battery

Introduction générale

La détection précoce des feux de forêt est essentielle pour limiter leurs effets dévastateurs et durables sur l'environnement, la flore et la faune. Des statistiques récentes montrent que les incendies de forêt ont augmenté ces dernières années en Tunisie et dans le monde, soulignant l'importance d'interventions rapides et efficaces pour protéger les écosystèmes forestiers.

En Tunisie, les incendies de forêt causent de grands dégâts économiques et environnementaux. Plus de 1 000 incendies de forêt ont été enregistrés chaque année entre 2010 et 2021, avec une augmentation significative en 2022. Environ 2 500 hectares de forêt ont été détruits par des incendies en Tunisie en 2022, soit une augmentation de 20 % d'une année sur l'autre [8] [15].

Avec les incendies de forêt qui font rage en Australie, aux États-Unis, en Sibérie et dans d'autres parties du monde, la situation mondiale n'est guère meilleure. Par exemple, les incendies de forêt en Californie en 2022 ont brûlé plus de 2 millions d'acres de terres, soit une augmentation de 10 % par rapport à l'année précédente [10] [2].

De plus, Les incendies de forêt en Amazonie ont également augmenté de 15 % en 2022 par rapport à 2021, détruisant de vastes étendues de forêt tropicale [6] [7].

Ces incendies ont non seulement détruit de vastes étendues de forêts, mais ont également entraîné une perte de biodiversité, ainsi que des émissions massives de gaz à effet de serre et des effets néfastes sur la santé humaine. Selon Global Forest Watch, les émissions de CO₂ des feux de forêt atteindront 2,3 milliards de tonnes en 2022, soit une augmentation de 12 % par rapport à 2021 [3].

La détection précoce des incendies de forêt joue un rôle important dans la limitation de ces impacts. La détection rapide d'un départ de feu permet aux équipes d'intervention d'intervenir plus rapidement pour éteindre le feu et éviter sa propagation. Cela réduit la superficie brûlée, minimise la perte de biodiversité et atténue les impacts sur les écosystèmes forestiers.

Dans le climat actuel de changement climatique et de risque accru d'incendies de forêt, il est très important de développer et de mettre en œuvre des systèmes de détection précoce, efficaces et fiables. Les drones équipés de capteurs spécifiques comme celui présenté dans ce projet peuvent y jouer un rôle important. Les drones apportent une aide en surveillant les forêts à la recherche de signes d'incendie.

Dans ce cadre, se déroule notre projet de fin d'études qui consiste à concevoir et à développer un drone autonome lié à une application mobile qui détecte et identifie rapidement les incendies.

Par le biais de ce rapport, nous allons détailler les différents chapitres de réalisation de ce projet :

- Dans le premier chapitre « Étude de projet », nous introduirons tout d'abord, le contexte général de notre projet ainsi que l'organisme d'accueil. Nous présenterons ensuite l'étude préalable du projet concernant l'étude de l'existant et les principaux objectifs de notre solution logicielle et la demande du projet. Finalement, nous allons terminer ce chapitre par la description de la méthodologie pour laquelle nous avons opté.
- Dans le chapitre suivant « Phase préparatoire », nous commencerons par une étude préliminaire du projet tout en décrivant l'analyse et la spécification des besoins classés en deux branches telles que : les besoins fonctionnels et les besoins non fonctionnels ainsi l'identification des acteurs en élaborant un diagramme de d'utilisation global et un backlog du produit. Par la suite, il convient de répartir les besoins identifiés en sprints, tout en établissant un calendrier prévisionnel et en définissant l'architecture détaillée de système.
- Dans le chapitre trois qui est nommé, « Release 1 : application mobile SaviaBot », nous présenterons, dans un premier lieu, le premier sprint « Gestion des Comptes ». Dans un deuxième lieu, nous présenterons le deuxième sprint « Planification de missions et avis clients » dans chaque sprint, nous présentons l'organisation et son backlog de sprint, l'analyse de notre sprint, les diagrammes UML et l'exécution montrant quelques conceptions d'interfaces.
- Le quatrième chapitre intitulé « Release 2 : Réalisation de SaviaHawk drone », qui présente le design matériel utilisé. Premièrement, nous commençons par la présentation de l'organisation de notre projet et l'explication du choix du matériel adéquat. Enfin, nous décrivons le processus d'assemblage du matériel pour réaliser notre solution.
- Finalement, nous clôturons notre travail par le chapitre « Release 3 : Amélioration continue du système de la détection ». Ce chapitre présente la dernière version

de notre projet, il met en évidence le sprint 5 « surveillance et amélioration du système de détection du drone ». Ce sprint définit les aspects cruciaux des alertes de détection, de l'exploration des données collectées par les drones, ainsi que de la surveillance et de l'amélioration des systèmes.

- Pour résumer, une conclusion générale vient synthétiser le projet réalisé et proposer des perspectives à court et à moyen terme.

1.1 Introduction

Ce chapitre introductif sera consacré à la présentation générale du projet ainsi de l'organisme d'accueil dans lequel nous avons effectué notre stage. Dans une deuxième partie, nous allons livrer les critiques et les problèmes qui s'en dégagent pour formuler les solutions et les méthodologies à adopter. pour atteindre notre objectif.

1.2 Champ d'application

Le projet SaviaBot est une initiative d'entreprise dans le cadre de la préparation d'un projet de fin d'études en vue de l'obtention du diplôme de licence en Ingénierie de Système informatique à l'institut Supérieur d'Informatique de Mahdia (ISIMA). Ce projet académique sera mené sur une période de quatre mois, débutant le 06 février et s'achevant le 06 juin 2023.

1.3 Présentation de l'organisme d'accueil

Nous commencerons d'abord par présenter la Pépinière d'entreprises Mahdia : CITY Lab, puis décrire l'entreprise Quetratech.

1.3.1 Pépinière d'entreprises Mahdia : CITY Lab

Le FabLab Solidaire "CITY Lab", un atelier de fabrication numérique accessible à tous, est situé dans la pépinière d'entreprises de Mahdia. Il est géré en partenariat avec l'association Cit'ESS et offre à ses bénéficiaires un accès à divers outils numériques tels que l'imprimante 3D, la fraiseuse numérique, la découpeuse laser, pour la réalisation de

prototypes, ainsi que des équipements high-tech et un espace dédié pour l'immersion dans le monde de la réalité virtuelle/réalité augmentée.



FIGURE 1.1 — Logo de Pépinière d'entreprises et CITYLAB Mahdia

1.3.2 Quetratech

Pour notre stage de fin d'études, nous avons choisi Quetratech comme entreprise d'accueil, qui est une entreprise spécialisée dans la conception, le développement et l'invention de produits numériques pour les marques émergentes. Au sein de cette entreprise, le travail se concentre sur les technologies Web, les expériences mobiles et l'Internet des objets. QuetraTech offre des solutions de haute qualité pour les applications mobiles, la création de sites Web attrayants, la conception graphique pour une meilleure interaction entre les utilisateurs et les informations, ainsi que des solutions complètes pour l'Internet des objets, y compris la conception, le développement, l'installation et l'exploitation.



FIGURE 1.2 — Logo Quetratech

1.4 Présentation du projet

1.4.1 Problématique

La problématique centrale de ce projet est la détection précoce des incendies de forêt. Les incendies de forêt sont des catastrophes naturelles qui peuvent avoir des conséquences dévastatrices sur l'environnement, la biodiversité, les ressources naturelles et la sécurité des populations. Ils peuvent se propager rapidement, devenir incontrôlables et causer des dommages irréparables.

Les défis liés à la détection précoce des incendies de forêt résident dans les caractéristiques spécifiques de ces événements. Ils se produisent souvent dans des zones géogra-

phiques étendues et difficiles d'accès, ce qui rend la surveillance continue et l'intervention rapide difficiles. De plus, les incendies de forêt peuvent démarrer et se propager rapidement, rendant essentielle une détection précoce pour une intervention rapide et efficace.

Les systèmes de détection traditionnels, tels que les tours de guet et les patrouilles terrestres, peuvent présenter des limitations en termes de couverture, de réactivité et de précision. Il est donc nécessaire de développer des approches innovantes qui utilisent les technologies pour améliorer la détection des incendies de forêt.

1.4.2 Objectifs du projet

L'objectif principal de ce projet est de mettre en place un système avancé de détection précoce des incendies de forêt, afin de minimiser les risques et d'accélérer les interventions pour contenir et éteindre les incendies. Les objectifs spécifiques sont les suivants :

- Développer une solution technologique efficace et fiable pour la détection précoce des incendies de forêt.
- Mettre en place un système de capteurs pour collecter en temps réel des données environnementales et des indicateurs d'incendie.
- Développer des algorithmes d'analyse de données et d'intelligence artificielle pour détecter de manière fiable les signes d'incendie.
- Établir une communication robuste pour transmettre les données de détection en temps réel à une station centrale de contrôle.
- Intégrer un système de gestion et de coordination des interventions pour une réponse rapide et efficace aux incendies.
- Assurer l'accessibilité et l'interopérabilité du système avec d'autres systèmes existants.
- Garantir la sécurité des données et la confidentialité des informations collectées.

1.5 Étude et critique de L'existant

Dans cette partie, nous allons étudier les solutions présentes sur le marché. Cette étude permettra d'identifier les points forts et les points faibles chez chaque solution ainsi les technologies existantes afin de définir l'allure générale de notre solution proposée.

1.5.1 Étude de l'existant

L'étude de l'existant consiste à analyser les solutions actuelles pour la détection des incendies de forêt. Cela inclut l'examen des technologies, équipements et méthodes utilisés. Certaines solutions utilisent des caméras infrarouges ou des capteurs de fumée pour détecter les incendies. Les méthodes de communication, telles que les réseaux sans fil ou satellites, sont également prises en compte. L'analyse permet d'identifier les forces, les limites et les lacunes des solutions existantes, afin de développer une solution améliorée et adaptée aux besoins spécifiques du projet.

1.5.1.1 OroraTech

OroraTech [11] est une entreprise de technologie qui fournit des solutions d'intelligence thermique depuis l'espace. Leur solution Wildfire aide à protéger les actifs forestiers en informant des détections précoce de hotspots et en surveillant les zones à risque d'incendies de forêt. La solution fournit des données fiables provenant de plus de 20 sources satellites et d'algorithmes avancés pour l'évaluation des risques, la détection précoce, la surveillance en temps réel et l'analyse des dommages. Ils proposent également des solutions pour la planification urbaine résiliente au climat, la surveillance de l'infrastructure et de l'industrie et la gestion des ressources agricoles.

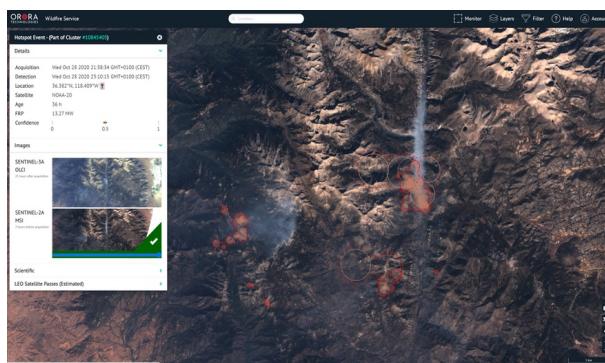


FIGURE 1.3 — Plateforme WILDFIRE d'OroraTech

1.5.1.2 Pano wildfire detection stations

Les stations de détection d'incendies de forêt **Pano AI** [12] sont des systèmes conçus pour surveiller les incendies de forêt en temps réel. Elles utilisent des technologies avancées de détection de feu, de suivi de la fumée et de diffusion d'alerte pour aider les secours à réagir rapidement et efficacement aux incendies. Les stations Pano peuvent être installées dans des zones à risque d'incendies et transmettent en temps réel des données sur la

localisation, la taille et la gravité des incendies.

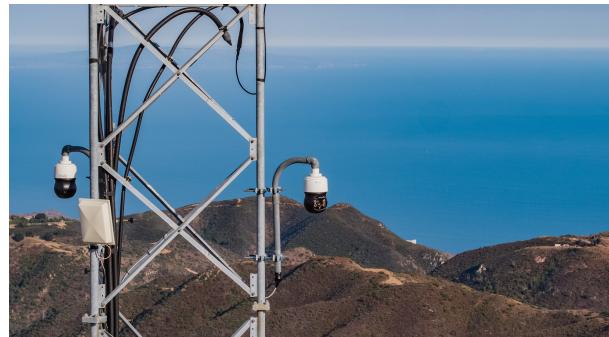


FIGURE 1.4 — Station de détection Pano AI

1.5.1.3 Crop-Care

Crop-Care est une solution d'agriculture de précision de la startup **Robocare** [13]. Elle utilise l'imagerie hyperspectrale et l'intelligence artificielle pour surveiller les cultures, détecter les maladies et fournir des recommandations de gestion optimale des terres et des cultures.

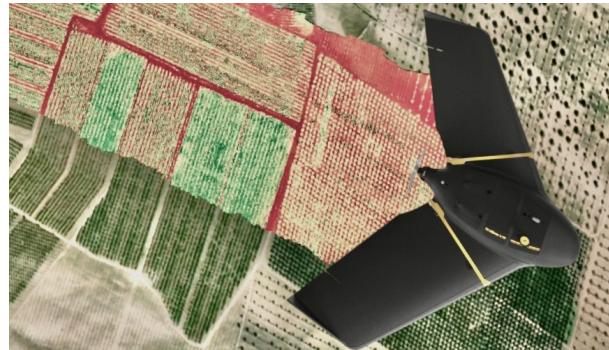


FIGURE 1.5 — Solution Crop-Care de Robocare

1.5.2 Critique de l'existant

Dans le tableau 1.1, nous présentons une critique des solutions existantes dans le domaine de la surveillance des incendies de forêt et de l'agriculture de précision. Cette étude met en évidence certaines limitations des solutions examinées, notamment :

Nom/Pays	Problème Résolu	Méthode	Les inconvénients
OroraTech (Allemagne)	la surveillance de l'infrastructure et de l'industrie et la gestion des ressources agricoles	Surveillance par Satellite	- Le coût est élevé. - Inutilisable dans toutes les conditions météorologiques.
Pano (États-Unis)	surveiller les incendies de forêt	stations de détection d'incendies (tour)	- pas pratique dans les zones avec un terrain surélevé et fourchu. - Inadapté dans les cas particuliers tels que les feux de camping. La diffusion d'alerte est instantanée, mais la position exacte de l'impact peut être déduite de manière inexacte.
Crop-Care (Tunisie)	capteur d'éclairage adaptatif	Drone / Software	- Le drone non autonome, nécessite un pilote - Le système effectue un scan du petit terrain.

TABLE 1.1 — Critique de l'existant

1.6 Solution Proposée

Après une analyse approfondie, nous avons constaté que les solutions existantes pour la surveillance des incendies dans les espaces verts et la gestion des ressources agricoles sont complexes et coûteuses pour les utilisateurs. Cependant, pour remédier à ces défis, nous proposons un système IoT autonome contrôlé par l'intelligence artificielle. Ce système utilisera les données collectées dans le but de fournir une interface conviviale pour les professionnels tels que les pompiers et les agriculteurs.

Ce système sera en mesure de surveiller les incendies dans les grands espaces verts tels que les parcs, les forêts, etc. Il permettra également une gestion efficace des ressources agricoles telles que les cultures de blé et d'autres cultures similaires. En exploitant les données collectées, ce système pourra prendre des décisions en temps réel pour éteindre les incendies et gérer les ressources agricoles de manière optimale.

En somme, ce système IoT autonome sera non seulement facile à utiliser pour les pro-

fessionnels, mais il sera également une solution rentable pour la surveillance des incendies et la gestion des ressources agricoles.

La solution repose sur un système IoT (Internet des objets) composé de drones autonomes, de stations de recharge et d'une application de contrôle. Les drones autonomes sont équipés de capteurs avancés qui leur permettent de collecter des données environnementales lors de leurs missions de surveillance des zones forestières. Ces données sont ensuite transmises aux stations de recharge, qui agissent comme des hubs de collecte et de régulation des données.

Les stations de recharge jouent un rôle clé dans le système IoT en collectant les données environnementales des drones et en les transmettant à une station centrale de contrôle via des connexions sans fil sécurisées. Ces données sont utilisées pour analyser les conditions environnementales en temps réel et réguler les missions des drones en fonction de ces informations.

L'application offre une interface conviviale aux utilisateurs tels que les professionnels du secteur forestier. Elle permet de visualiser les données collectées par les drones, de suivre leurs positions en temps réel et de prendre des décisions éclairées en cas de détection d'incendie. L'application permet également de planifier et de coordonner les missions des drones, d'effectuer des réglages et de surveiller l'état des batteries.

En combinant les capacités des drones autonomes, des stations de recharge et de l'application de contrôle, ce système IoT offre une surveillance continue, précise et réactive des zones forestières. Il permet une collecte efficace des données environnementales, une régulation optimisée des missions des drones et une prise de décision basée sur des informations en temps réel. Ainsi, il contribue à la prévention et à la gestion efficace des incendies de forêt.

Qu'est-ce que l'IoT :

L'Internet des objets (IoT) est une infrastructure réseau dynamique dotée de capacités d'auto-configuration basée sur des protocoles de communication standard et interopérable, où les "choses" physiques et virtuelles ont des identités, des attributs physiques et des personnalités virtuelles. Les interfaces intelligentes et leur intégration transparente dans le réseau d'information communiquent fréquemment des données liées aux utilisateurs et à leur environnement. Les objets connectés à Internet sont appelés Internet of Things (IoT) [5].

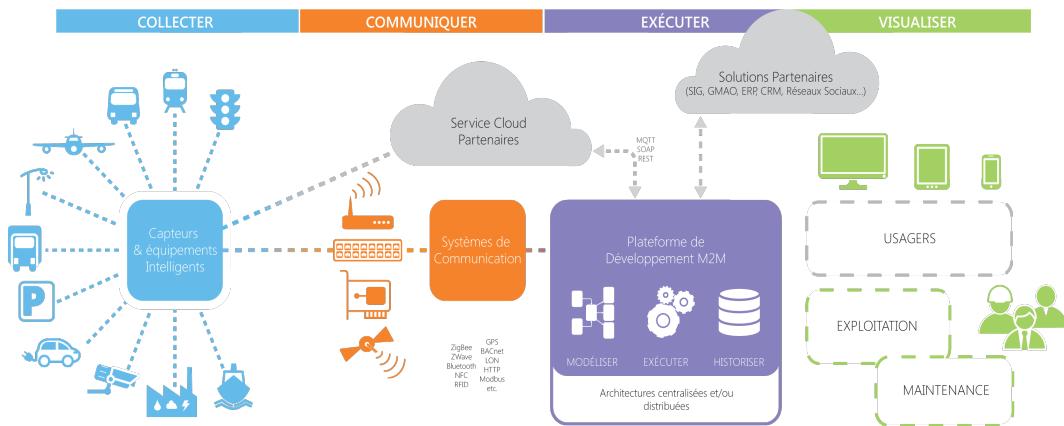


FIGURE 1.6 — L'écosystème internet des objets

1.7 Méthodologie adoptée

Construire un projet est pareil que de construire une maison.

Pour cela, une méthode de travail est essentielle pour réaliser des projets de haute gamme qui garantit un niveau de qualité acceptable et éviter tout débordement au niveau des délais.

Une méthodologie de travail nous aide à accomplir chaque étape de notre projet, de la planification à la mise en œuvre, dans le souci d'efficacité et rentabilité.

Il existe donc plusieurs méthodologies, chacune ayant ses propres principes et outils pour gérer les différentes phases du projet. Cependant, pour faire un choix judicieux, il est primordial de prendre en compte divers facteurs tels que la nature et la complexité du projet, les ressources disponibles, les délais impartis, etc. Une compréhension approfondie des principes de chaque méthodologie est donc essentielle pour choisir celle qui conviendrait le mieux au projet.

1.7.1 Méthodologie retenue : Méthodologie agile

La méthodologie Agile est une approche souple et adaptable pour la gestion de projets, notamment dans les domaines du développement de logiciels et de la production de produits. Elle met l'accent sur la collaboration, l'adaptabilité face aux changements et la livraison continue de valeur. Les équipes de développement travaillent en cycles courts pour produire des parties fonctionnelles du produit, appelées "itérations", tout en communiquant régulièrement avec les clients pour s'assurer que le produit final réponde à

leurs besoins et attentes. L'approche Agile favorise également une culture de transparence et de collaboration entre les membres de l'équipe, ce qui conduit souvent à des résultats de qualité supérieure.

1.7.2 Kanban

Kanban est une méthodologie de gestion de projet et de flux de travail conçue pour optimiser la productivité, la qualité et la fiabilité en réduisant les déchets, les temps d'attente et en limitant les travaux en cours. Elle permet donc de représenter les flux de travail, les tâches à effectuer, les tâches en cours et les tâches terminées à l'aide de schémas visuels ou électroniques. Les utilisateurs peuvent facilement voir la progression du travail, les goulots d'étranglement et les défis potentiels, permettant à la collaboration de surmonter ces obstacles. Kanban est couramment utilisé dans la fabrication, les services informatiques et les projets de développement de logiciels, mais peut être appliquée à presque tous les types de flux de travail. La méthode Kanban est basée sur des principes tels que la transparence, l'équilibre de la charge de travail et l'amélioration continue.

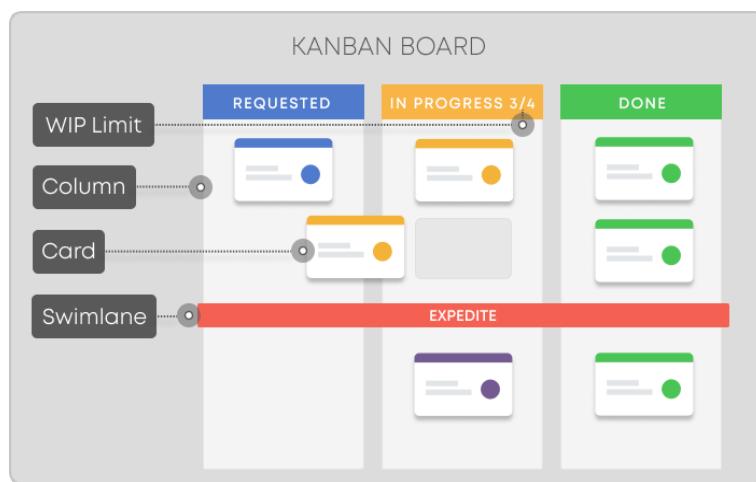


FIGURE 1.7 — Tableau kanban

Le tableau Kanban est un outil visuel utilisé pour représenter le flux de travail et suivre l'état d'avancement d'un projet. Il est généralement constitué de plusieurs colonnes qui représentent les différentes étapes du processus, telles que "À faire", "En cours" et "Terminé". Chaque tâche est représentée par une carte qui contient des informations telles que la description de la tâche, la personne responsable, la date d'échéance, etc.

Scrumban - Kanban avec la notion de sprints : Scrumban est une méthodologie de gestion de projet qui combine deux stratégies Agile courantes : Scrum et Kanban

La méthode Scrumban combine les principes de Scrum et de Kanban pour la gestion de projet. Elle a été développée pour aider les équipes à effectuer une transition entre ces deux méthodologies Agile. Scrumban offre une flexibilité aux équipes qui souhaitent intégrer des éléments de Scrum, tels que les sprints et la planification itérative, tout en utilisant les principes visuels et d'amélioration continue de Kanban. Cette approche est particulièrement adaptée aux équipes qui ont une bonne connaissance préalable de Scrum et de Kanban, leur permettant de tirer parti des avantages des deux méthodologies. Pour notre équipe, Scrumban offre une solution hybride qui peut être adaptée en fonction des besoins spécifiques du projet, combinant la structure des sprints avec la flexibilité et la visualisation du flux de travail de Kanban.

1.7.3 Langage de conception



FIGURE 1.8 — Unified Modeling Language

Pour concevoir correctement notre application, nous allons appliquer les principes et les règles du langage de modélisation UML (Unified Modeling Language) dans ce travail. UML est un langage unifié adapté à la modélisation visuelle, riche sémantiquement et syntaxiquement. Ce langage est destiné à l'architecture, la conception et la mise en œuvre des systèmes logiciels complexes, tant au niveau de leur structure que de leur comportement.

1.8 Conclusion

Dans ce chapitre, nous avons examiné les concepts fondamentaux nécessaires à la compréhension des attentes de notre projet. Nous avons également décrit la méthodologie et les outils utilisés pour mener à bien le projet.

Dans le chapitre suivant, nous aborderons l'analyse des besoins fonctionnels et non fonctionnels ainsi que le planning prévisionnel du projet.

2.1 Introduction

Après avoir établi le contexte général de notre projet, nous allons désormais entamer la première phase de la méthodologie adoptée, à savoir la phase de planification, également connue sous le nom de “Sprint 0”.

Pour amorcer le sprint 0, il est essentiel de planifier et de définir l’architecture de la solution envisagée. Tout d’abord, une analyse et une spécification approfondies des besoins doivent être réalisées, ainsi l’identification des acteurs, en élaborant un diagramme de d’utilisation global et un backlog du produit. Par la suite, il convient de répartir les besoins identifiés en sprints, tout en établissant un calendrier prévisionnel et en définissant l’architecture détaillée de système.

2.2 Analyse et spécifications des besoins

Cette étape vise à apprêhender le contexte du système afin de définir les besoins fonctionnels et non fonctionnels, d’identifier les acteurs impliqués de spécifier les exceptions éventuelles et finalement de déterminer les cas d’utilisation initiaux.

2.2.1 Identification des acteurs

Les acteurs dans un projet peuvent être définis comme des individus, des groupes ou des entités externes qui ont une interaction directe avec le système modélisé, chacun jouant un rôle clé dans la réussite du projet. Notre application implique la participation de trois acteurs clés, à savoir :

Acteur	Rôle
Super Administrateur	Le super administrateur gère les comptes des administrateurs par région, assiste les avis des clients et affiche également les données des administrateurs.
Administrateur	L'administrateur gère et supervise toutes les opérations de drone, de la configuration à la maintenance, afin de prévenir et de gérer les incendies.
Client	Le client surveille les cultures, détecte les incendies et planifie la maintenance, tout en accédant aux données historiques et météorologiques, personnalisant son expérience et bénéficiant d'analyses approfondies.

TABLE 2.1 — Identification des acteurs

2.2.2 Besoins Fonctionnels

L'objectif de notre projet est de satisfaire un besoin. Il faut l'exprimer clairement avant d'imposer une solution. Les besoins fonctionnels définissent les fonctionnalités et les actions que le système informatique doit être capable d'exécuter pour répondre aux besoins de l'utilisateur ou du client. Ils sont déterminés en identifiant les tâches et les fonctions que le système doit effectuer pour remplir sa fonction principale. Ils sont essentiels pour s'assurer que le système répond aux besoins de l'utilisateur de manière efficace et fiable. Après l'étude de l'existant, nous allons dégager les besoins fonctionnels requis :

2.2.2.1 Besoins Fonctionnels de Super Administrateur

Fonctionnalité	Description
S'authentifier	Le super admin peut s'identifier en utilisant son adresse e-mail et/ou son numéro de téléphone.
Gérer des comptes administrateurs régionaux	Le super admin peut créer et gérer les comptes d'administrateurs pour chaque région afin d'assurer une gestion efficace des tâches.
Afficher les données des Admins	Le super admin peut suivre l'activité des administrateurs, y compris leurs tâches et leur emplacement, pour une gestion efficace des ressources.
Accéder aux commentaires des clients	Le super admin peut accéder aux commentaires des clients pour comprendre leurs préoccupations et y répondre rapidement, améliorant ainsi la satisfaction client.

TABLE 2.2 — Besoins Fonctionnels de Super Administrateur

2.2.2.2 Besoins Fonctionnelles de l'Administrateur

Fonctionnalité	Description
S'authentifier	L'administrateur peut s'identifier en utilisant ses informations d'authentification obtenue depuis le super admin.
Mettre à jour son profil	L'administrateur peut modifier ses informations personnelles et de connexion.
Recevoir les alertes	L'administrateur peut gérer les alertes pour répondre rapidement aux problèmes.
Suivre l'état des drones	L'administrateur peut s'assurer que les drones sont maintenus en bon état de fonctionnement.
Explorer les données collectées par les drones	L'administrateur peut explorer les données collectées par les drones pour améliorer les opérations.
Consulter les demandes des clients	L'administrateur peut consulter les demandes émises par les clients afin de les traiter et de les confirmer.

TABLE 2.3 — Besoins Fonctionnelles de l'administrateur

2.2.2.3 Besoins Fonctionnelles du client

Fonctionnalité	Description
S'authentifier	Le client peut s'identifier en utilisant son adresse e-mail et/ou son numéro de téléphone.
S'inscrire et activer son compte	Le client peut créer un compte et de l'activer pour accéder à l'application.
Mettre à jour son profil	Le client peut modifier ses informations personnelles, telles que son nom, son adresse, son adresse e-mail, son numéro de téléphone, etc.
Sélectionner son espace de surveillance	Le client peut sélectionner l'espace vert qu'il souhaite surveiller et demander le service SaviaBot pour la maintenance et la surveillance.
Donner des commentaires sur les services qu'il a reçus	Le client peut donner des commentaires et des retours d'expérience sur les services qu'il a reçus pour améliorer la qualité des services.
Surveiller en temps réel l'état des cultures et les zones vertes	Le client peut surveiller en temps réel l'état des cultures et des zones vertes, y compris les niveaux d'humidité, de température, de luminosité, etc.
Recevoir des alertes de détection d'incendies à un stade précoce	Permettre à l'utilisateur de détecter les incendies à un stade précoce grâce à des capteurs et des alertes automatiques (SMS + Push Notif).
Ajouter Collaborateur	Le client peut ajouter un ami/collègue qui va recevoir les notifications de détection et peut accéder au dashboard de l'utilisateur.

TABLE 2.4 — *Besoins Fonctionnelles du client*

2.2.3 Besoins non fonctionnels

Une fois les besoins fonctionnels satisfaits, des besoins non fonctionnels ou opérationnels doivent être pris en compte tout au long du développement de l'application.

Les besoins non fonctionnels sont des exigences qui décrivent les propriétés globales d'un système informatique plutôt que ses fonctions spécifiques. Ils comprennent des critères de qualité et des contraintes qui sont importants pour l'utilisateur ou le client.

- **Performance** : Le système doit être performant pour détecter rapidement les signes précurseurs d'un incendie et envoyer des alertes en temps réel.
- **Sécurité** : Le système doit être sécurisé pour éviter toute alternative de piratage ou de sabotage. Les données collectées doivent être protégées pour préserver la vie privée des utilisateurs.
- **Fiabilité** : Le système doit être très fiable, car la détection précoce d'un incendie est cruciale pour sauver des vies et limiter les dégâts matériels.
- **Disponibilité** : Le système doit être disponible 24h/24h et 7j/7j pour surveiller en permanence les zones à risque d'incendie et réagir rapidement en cas de détection d'un incendie.
- **L'énergie** : Le système doit être conçu pour minimiser la consommation d'énergie et prolonger la durée de vie de ses batteries, afin de maintenir une surveillance continue sans nécessiter de recharges fréquentes.

2.3 Gestion de projet avec Kanban

En kanban, il est important de définir les exigences des utilisateurs en termes de fonctionnalités et de les hiérarchiser en tant que backlog du produit.

Backlog du produit : Le backlog du produit est une liste priorisée et constamment mise à jour des fonctionnalités, améliorations et tâches requises pour un produit. Cet outil essentiel pour les équipes agiles permet d'organiser le travail, de se concentrer sur les éléments cruciaux et de communiquer efficacement avec les parties prenantes.

TABLE 2.5 — Backlog du produit

ID	Fonctionnalité	Récit utilisateur	Priorité
1	S'inscrire	En tant qu'utilisateur (client), je peux m'inscrire.	Haute
2	Authentification	En tant qu'utilisateur (admin, ou client), je peux me connecter avec mon e-mail et mon mot de passe.	Haute
2	Gérer les comptes	En tant que super admin, je peux créer et gérer les comptes admin régionaux.	Haute
3	Gérer les demandes des clients	En tant qu'administrateur, je peux accorder les systèmes de détection à chaque client et valider leurs demandes.	Haute
4	Recevoir des alertes de détection d'incendie	En tant que client, je peux recevoir des alertes de détection d'incendie précoce par SMS et notifications push.	Haute
5	Gérer les stations/drones	En tant qu'admin, je peux accorder ou supprimer des drones et stations aux clients.	Moyenne
6	Suivre l'état des drones	En tant qu'admin, je peux vérifier les positions et l'état des drones en temps réel.	Moyenne
7	Gérer les paramètres du compte	En tant qu'utilisateur (admin ou client), je peux modifier mon profil.	Moyenne
8	Accéder aux commentaires des clients	En tant que super admin, je peux accéder aux commentaires des clients pour améliorer la satisfaction client.	Moyenne
9	Gérer les zones à surveiller	En tant que client, je peux spécifier, ajouter ou modifier des zones à surveiller.	Moyenne
10	Consulter l'historique	En tant qu'admin, je peux consulter l'historique des positions visitées par le drone.	Faible
11	Afficher les données des admins	En tant que super admin, je peux suivre l'activité des admins, y compris leurs tâches.	Faible

2.4 Planification des sprints

Le processus de planification de chaque sprint dans le cadre de kanban permet de déterminer les tâches à accomplir sur une période spécifique. Après une réflexion approfondie, nous avons identifié trois versions pour notre projet. Le tableau affiche la planification des sprints pour chaque version (Release).

TABLE 2.6 — Planification des sprints

Release	Sprint ID	Nom du Sprint	Durée/Jours
Release 1	1	Gestion des comptes	8
	2	Planification des missions et avis clients	10
Release 2	3	Assemblage et navigation de drone	15
	4	Détection et alertes précoce des incendies	12
Release 3	5	Surveillance et amélioration du système de détection du drone	10

2.5 Environnement logiciel

Au sein de cette partie, nous exposons les langages de programmation ainsi que les logiciels que nous avons mis en œuvre pour la réalisation de notre projet. Nous avons utilisé comme logiciel :

- **Visual Studio Code**

Nous utilisons Visual Studio Code, un éditeur de code multiplateforme, open source et gratuit développé par Microsoft, pour développer notre application Flutter et déboguer facilement notre code. Avec son interface conviviale et ses fonctionnalités puissantes, Visual Studio Code nous permet d'écrire, de tester et de déployer notre application de manière efficace.



FIGURE 2.1 — Logo VS Code

- **Flutter (Dart)**

Nous utilisons Flutter, un framework open source de Google, pour développer une application Android en utilisant un code unique pour différentes plateformes.



FIGURE 2.2 — Logo Flutter et Dart

- **Android Studio**

Android Studio est un IDE (Integrated Development Environment) développé par Google, qui offre une solution complète pour créer des applications Flutter multiplate-formes. Nous avons également utilisé l'émulateur fourni par Android Studio pour tester notre application Android pendant le processus de développement.



FIGURE 2.3 — Logo Android Studio

- **Firebase**

Firebase est une plateforme de développement d'applications de Google qui simplifie l'intégration de services tels que l'authentification des utilisateurs et le stockage de données. Nous utilisons Firebase, en particulier Firestore pour la gestion des données et l'authentification, afin de créer une expérience utilisateur sécurisée et réactive dans notre application.



FIGURE 2.4 — Logo Firebase

- **Google Cloud Platform**

Google Cloud Platform (GCP) est une plateforme de services cloud complète offrant un large éventail de fonctionnalités. Nous avons utilisé GCP, notamment Google Cloud Storage, Firestore, Google Maps et Google Cloud Messaging, pour héberger, stocker, gérer et intégrer des services dans nos applications, en bénéficiant de la puissance, de la sécurité et de la fiabilité de l'infrastructure cloud de Google.



Google Cloud Platform

FIGURE 2.5 — Logos Google Cloud Platform

• Figma

Figma est un outil de conception d'interface utilisateur en ligne que nous avons utilisé pour concevoir notre application avant le développement.



FIGURE 2.6 — Logo Figma

• Raspbian 64bit

Nous avons utilisé Raspbian, une distribution Linux dédiée aux cartes Raspberry Pi, pour fusionner les capteurs, exploiter les données, contrôler le drone de manière autonome et communiquer avec la station.



FIGURE 2.7 — Logo RaspbianOS

• Python 3

Nous avons utilisé Python 3 pour exécuter tous les programmes de détection et de vols de drone, ainsi que pour connecter ces programmes entre eux afin de contrôler l'en-

semble en fonction de conditions spécifiques. Python 3 nous a permis de développer des programmes efficaces, lisibles et modulaires pour la réalisation de ces fonctionnalités.



FIGURE 2.8 — Logo Python

- **Geany**

Geany est un éditeur de texte léger et convivial pour Raspberry Pi, offrant des fonctionnalités d'édition de base et une interface intuitive pour les développeurs.



FIGURE 2.9 — Logo Geany

- **Notepad++**

Notepad++ est un éditeur de texte puissant utilisé pour éditer le code sur notre station de travail. Il offre des fonctionnalités avancées telles que la coloration syntaxique pour plusieurs langages, y compris Python 3, ainsi que des outils de recherche et de remplacement avancés. Avec Notepad++, nous avons pu personnaliser notre environnement de développement en ajoutant des plugins pour répondre à nos besoins spécifiques.



FIGURE 2.10 — Logo Notepad++

- **QGroundControl**

Nous avons utilisé QGroundControl, un logiciel de contrôle de vol open-source, pour flasher et calibrer le contrôleur de vol. QGroundControl nous a permis de configurer et de gérer le drone de manière précise, en calibrant les capteurs tels que l'accéléromètre, le gyroscope et la boussole. Cela a assuré des mesures précises et une stabilité accrue lors des vols.



FIGURE 2.11 — Logo QGroundControl

- **YOLOv5**

Nous avons utilisé YOLOv5 pour entraîner et exécuter notre modèle de détection d'objets. C'est un algorithme performant et polyvalent, offrant une précision élevée et des vitesses d'inférence rapides.



FIGURE 2.12 — Logo YOLOv5

- **OpenCV**

OpenCV est une librairie de vision informatique open source très utilisée. Elle offre des fonctionnalités performantes pour le traitement d'images et de vidéos ainsi que des algorithmes avancés de détection d'objets et de reconnaissance faciale.



FIGURE 2.13 — Logo librairie OpenCV

- **PyTorch**

PyTorch : bibliothèque open source de machine learning de Facebook, idéale pour créer et entraîner des réseaux de neurones profonds avec support GPU performant. Populaire auprès des chercheurs et des développeurs en IA.



FIGURE 2.14 — Logo bibliothèque Pytorch

- **GitHub**

Nous avons utilisé GitHub pour partager et gérer notre code source. C'est une plateforme de développement collaboratif qui utilise Git, offrant des fonctionnalités de col-

laboration et de suivi des modifications pour faciliter l'amélioration continue entre les développeurs.



FIGURE 2.15 — Logo Github

• Draw.io

Draw.io est un outil en ligne puissant pour créer des diagrammes de toutes sortes, tels que des organigrammes, des cartes mentales, des diagrammes de flux et bien plus encore. Son interface conviviale permet une utilisation facile et intuitive, tandis que sa fonctionnalité de collaboration en temps réel en fait un choix idéal pour les équipes travaillant à distance.



FIGURE 2.16 — Logo Draw.io

• Tinkercard

Tinkercad est un outil de modélisation 3D simple et accessible, idéal pour les débutants et les projets d'impression 3D. Il offre une large gamme de fonctionnalités pour créer, manipuler et assembler des objets en 3D, le tout dans une interface conviviale et intuitive.



FIGURE 2.17 — Logo Tinkercard

• Creality

Creality Slicer est un logiciel de tranchage de modèles 3D conçu pour les imprimantes 3D Creality. Il permet de préparer et d'optimiser les modèles 3D pour l'impression en 3D.



FIGURE 2.18 — Logo Creality

• Trello

Trello est une application de gestion de projet en ligne basée sur la méthode Kanban. Cela permet aux équipes de créer des tableaux pour représenter les projets et des listes pour organiser les différentes phases et catégories. Les cartes sont utilisées pour représenter des tâches individuelles et peuvent être déplacées entre les listes pour montrer les progrès. Cette approche nous permet de suivre efficacement la progression des tâches, de regrouper les cartes par statut et de collaborer de manière transparente. Trello fournit ainsi une vue d'ensemble claire du projet, facilite la communication entre les membres de l'équipe et augmente la productivité.

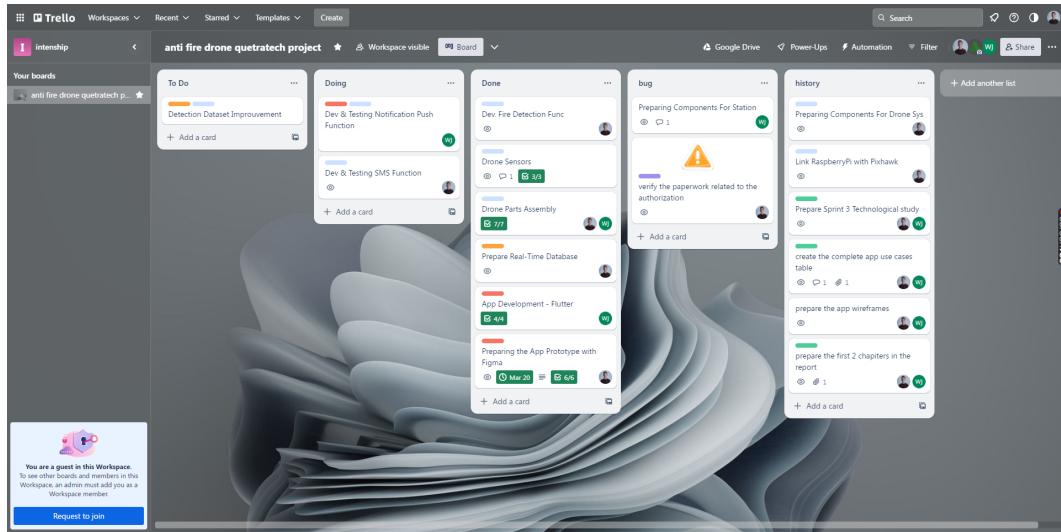


FIGURE 2.19 — Tableau Trello : Quetratech Intern

2.6 Architecture générale

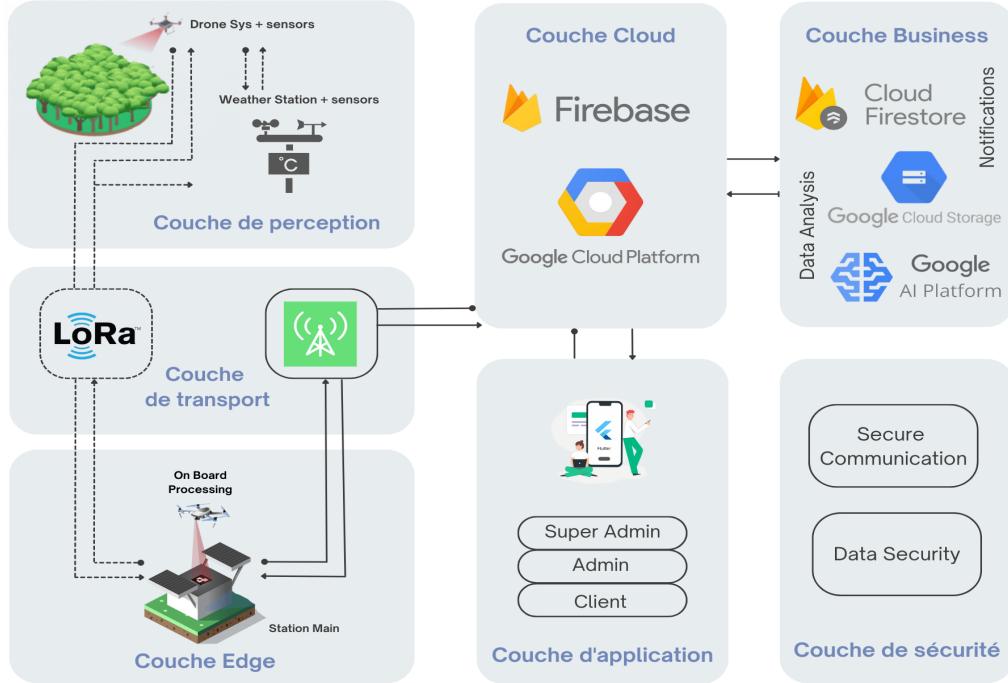


FIGURE 2.20 — Architecture IoT

Couche de perception (appareils et capteurs) :

- Drone : Microcontrôleurs, Capteurs et module de communication LoRa.
- Stations au sol : Capteurs météo et modules de communications LoRa.

Couche de transport (communication) :

- LoRa : Communication entre le drone et les stations au sol
- 4G : Connexion internet et communication avec le cloud pour la station principale

Couche Edge (calcul et traitement des données en local) :

- Raspberry Pi : Exécute le script Python pour la détection d'incendies, le contrôle du drone et la communication avec les stations au sol

Couche Cloud (services et infrastructures cloud) :

- Google Cloud Firestore : Stocke les informations des utilisateurs, les affectations de drones et de stations, et les événements de détection d'incendie
- Google Cloud Storage : Stocke les datasets prises par le drone
- Google Cloud AI Platform : Entraîne et déploie le modèle de détection pour améliorer

la détection d'incendies

Couche d'application (interface utilisateur et contrôle) :

-Application mobile (Flutter) : Développée avec différents rôles d'utilisateur (utilisateur, administrateur, super-administrateur) et connectée aux services cloud pour accéder aux données en temps réel et contrôler le système

Couche Business (analyse de données, prise de décision et notifications) :

-Analyse des données : Traite les données collectées par les drones et les stations au sol, analyse les conditions météo et prend des décisions en fonction des analyses

-Notifications : Utilise Firebase Cloud Messaging pour les notifications push et un service tiers pour les notifications SMS

Couche de sécurité (sécurité des données et des communications) :

-Communication sécurisée : Emploie des protocoles cryptés et des méthodes d'authentification sécurisées pour garantir une communication sécurisée entre les appareils, le cloud et les interfaces utilisateur

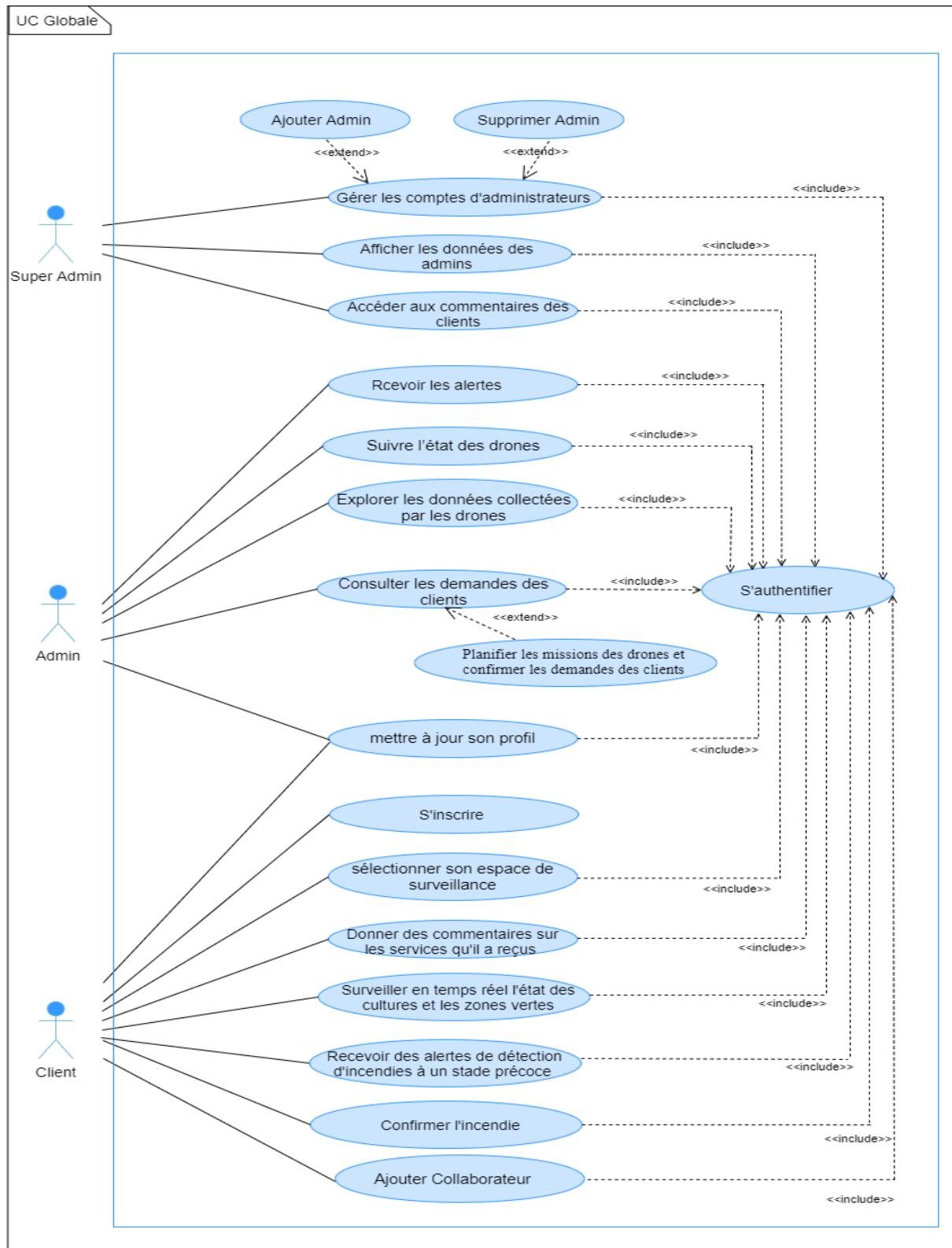
-Sécurité des données : Assure la confidentialité et la protection des données grâce à des mécanismes de contrôle d'accès appropriés, au cryptage et à des solutions de stockage sécurisées sur le cloud.

2.7 Analyse

Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation est un outil graphique essentiel pour représenter l'interaction entre un utilisateur et un système ainsi que ses fonctionnalités. Il est crucial pour comprendre les besoins des utilisateurs et communiquer les exigences fonctionnelles. Dans notre projet, nous avons utilisé ce diagramme de manière efficace pour visualiser ces interactions et fonctionnalité, comme illustré dans le diagramme 2.21 ci-dessus :

Le diagramme de la figure suivante présente les principales fonctionnalités offertes par l'application :

**FIGURE 2.21** — Diagramme de cas d'utilisation global

2.8 Motif d'architecture logicielle

Nous utilisons l'architecture Modèle-Vue-Contrôleur(MVC) pour décrire l'interaction entre les éléments du système et les acteurs à travers des diagrammes de séquence.

MVC (Model-View-Controller) : Model-View-Controller (MVC) est un modèle d'architecture logicielle d'interface utilisateur graphique introduit en 1978 qui sépare les responsabilités de chaque composant.

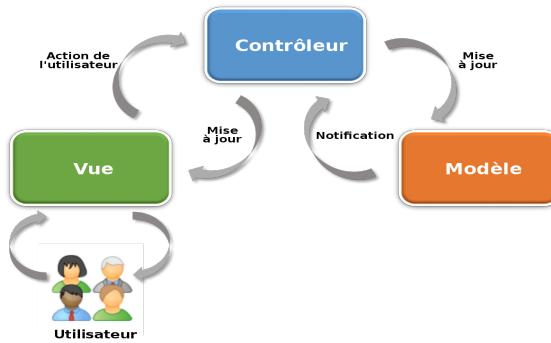


FIGURE 2.22 — Model-View-Controller (MVC) [9]

Modèle : Un modèle représente les données utilisées par l'application. Il peut être une base de données, d'un fichier ou simplement un objet contenant les informations nécessaires. Le modèle est responsable de la gestion et de la manipulation des données.

Vue : La vue est une représentation de l'objet dans l'application. Elle est chargée d'afficher les données au format approprié pour les utilisateurs. La vue peut également permettre aux utilisateurs d'interagir avec les données et de les modifier si nécessaire.

Contrôleur : Le contrôleur agit comme une interface entre le modèle et la vue. Il gère toute la logique métier de l'application, et traite les requêtes entrantes des utilisateurs et utilise les composants du modèle pour manipuler les données. Le contrôleur interagit aussi avec la vue pour rendre le résultat final visible aux utilisateurs.

2.9 Conclusion

Dans ce chapitre, nous avons utilisé la méthodologie Kanban pour planifier notre projet en détail. Nous avons créé le backlog du produit et décomposé le projet en sprints. Nous avons également identifié les besoins fonctionnels et non fonctionnels. Enfin, nous avons défini les technologies et l'environnement de travail à utiliser tout au long du projet.

3.1 Introduction

Dans ce chapitre, nous présentons la première version qui contient principalement deux sprints : le premier sprint «Gestion des Comptes », le deuxième sprint « Planification de missions et avis clients». En effet, dans chaque sprint, nous présentons l’organisation et son backlog de sprint, l’analyse de notre sprint, les diagrammes UML et l’exécution montrant quelques conceptions d’interfaces.

3.2 SPRINT 1 : “Gestion des comptes”

Dans cette section, nous classifions nos parties de sprint, l’organisation et le backlog de sprint, l’analyse, les diagrammes UML et l’exécution.

3.2.1 Backlog du SPRINT 1

Le tableau 3.1 donne un aperçu détaillé sur le Backlog du sprint qui prend en charge la fonctionnalité "Gestion des comptes".

ID	Sprint	User Story	Jour
1	Gestion des Comptes	Réalisation de la fonctionnalité "s'inscrire"	2
		Réalisation de la fonctionnalité "s'authentifier"	3
		Réalisation de la fonctionnalité "Mettre à jour son profil"	2
		Réalisation de la fonctionnalité "Changer mot de passe"	2
		Réalisation de la fonctionnalité "Modifier l'espace"	2
		Réalisation de la fonctionnalité "Gérer les comptes d'administrateurs"	2
		Réalisation de la fonctionnalité "Ajouter Admin"	2
		Réalisation de la fonctionnalité "Supprimer Admin"	2
		Réalisation de la fonctionnalité "Ajouter collaborateur"	2
Test			3

TABLE 3.1 — Backlog du sprint 1 : "Gestion des comptes"

3.2.2 Analyse

Dans cette partie, nous présentons le diagramme de cas d'utilisation, les descriptions textuelles pour chaque cas, ainsi que les diagrammes de séquence.

3.2.2.1 Diagramme de cas d'utilisation

La figure 3.1 représente le diagramme de cas d'utilisation pour le sprint 1.

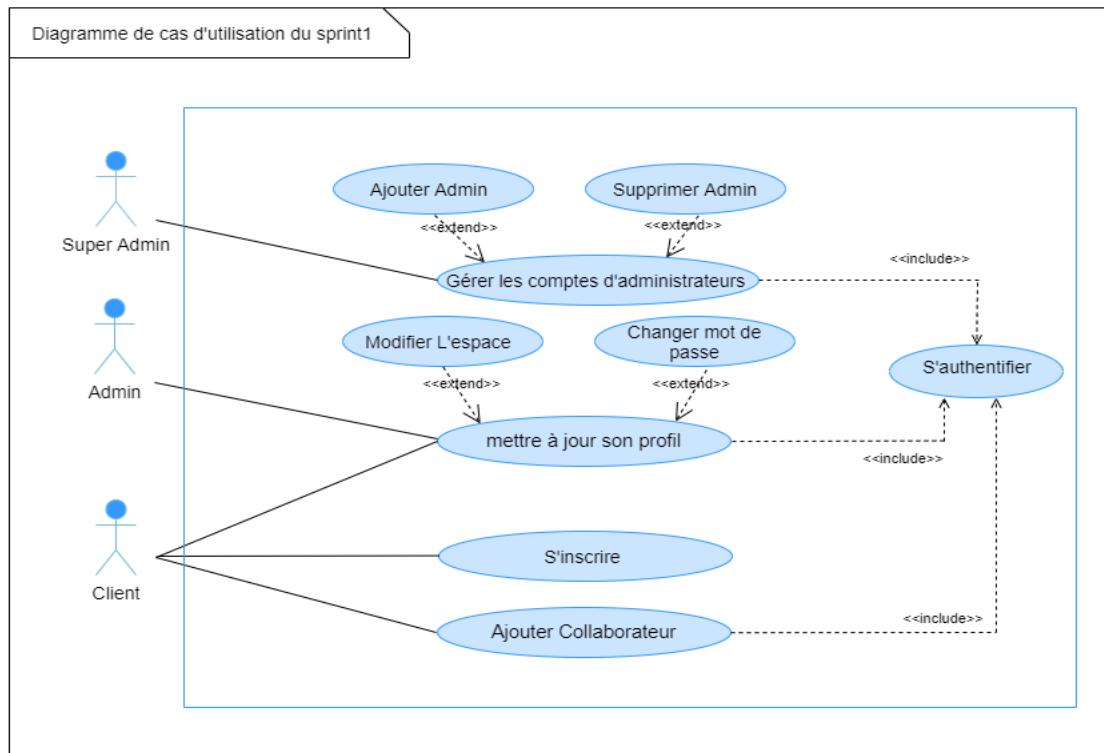


FIGURE 3.1 — Sprint 1- Diagramme de cas d'utilisation

3.2.2.2 Description textuelle des cas d'utilisation

- Description textuelle de cas d'utilisation « S'inscrire »

Nom du CU	S'inscrire
Acteurs	Client
Résumé	Le client s'inscrit pour bénéficier des fonctionnalités de "Saviabot".
Pré-conditions	L'utilisateur accède à l'interface d'inscription.
cénario Nominal	<p>1-Le système affiche l'écran d'intégration.</p> <p>2-Le client demande de rejoindre.</p> <p>3-Le client remplit le formulaire et valide.</p> <p>4-Le système effectue une vérification des champs obligatoires pour s'assurer qu'ils ne sont pas vides.</p> <p>5-Le système vérifie la validité des informations fournies.</p> <p>6-Le système vérifie l'existence du client.</p> <p>7-Le système envoie un courrier électronique de confirmation.</p> <p>8-L'utilisateur suit le lien de vérification dans l'e-mail.</p> <p>9-Le système enregistre l'utilisateur et affiche un message de réussite de l'inscription.</p>
Enchaînement Alternatif	<p>4.1 Les champs obligatoires sont vides.</p> <p>4.2 Le système affiche le message d'erreur : "Given String is empty".</p> <p>4.3 Retour au point 3 pour que le client puisse remplir à nouveau le formulaire.</p> <p>5.1 Les informations ne sont pas valides.</p> <p>5.2 Le système affiche le message d'erreur : "The email is badly formatted" si l'adresse e-mail est mal formatée.</p> <p>5.2.1 Le système affiche le message d'erreur : "Password should be at least 6 characters" si le mot de passe est inférieur à six caractères.</p> <p>5.3 Retour au point 3 pour que le client puisse modifier les informations fournies.</p> <p>6.1 Le compte existe déjà.</p> <p>6.2 Le système affiche le message d'erreur : "The email is already in use by another account".</p> <p>6.3 Retour au point 3 pour que le client puisse fournir de nouvelles informations.</p>
Post-conditions	Le client est dirigé vers la page "Pick location".

TABLE 3.2 — Description textuelle du scénario de cas d'utilisation « S'inscrire »

- Description textuelle de cas d'utilisation « S'authentifier »

Nom du CU	S'authentifier
Acteurs	Super Administrateur, Administrateur et le Client
Résumé	Pour tirer parfaitement parti des fonctionnalités offertes par "Savia-Bot", les utilisateurs doivent s'authentifier en tant que client, administrateur ou super administrateur.
Pré-conditions	L'utilisateur a déjà un compte. L'utilisateur accède au menu « Sign In ».
Scénario Nominal	<p>1- Le système affiche l'écran d'intégration.</p> <p>2- L'utilisateur demande de s'authentifier.</p> <p>3- L'utilisateur fournit ses identifiants dans les champs appropriés et valide.</p> <p>4- Le système vérifie que tous les champs requis sont renseignés.</p> <p>5- Le système effectue une validation des identifiants saisis pour s'assurer de leur validité.</p>
Enchaînement Alternatif	<p>4.1 Certains champs sont vides.</p> <p>4.2 Le système affiche le message d'erreur : "The given string is empty or NULL".</p> <p>4.3 Retour au point 3 pour que l'utilisateur puisse remplir à nouveau les champs.</p> <p>5.1 L'e-mail ou le mot de passe est incorrect.</p> <p>5.2 Le système affiche le message d'erreur : "There is no user record corresponding to this identifier, the user may have been deleted".</p> <p>5.3 Retour au point 3 pour que l'utilisateur puisse saisir de nouveaux identifiants.</p>
Post-conditions	Une fois authentifié par le système, l'utilisateur est automatiquement redirigé vers le tableau de bord qui correspond à son rôle - que ce soit en tant que super administrateur, administrateur ou client.

TABLE 3.3 — Description textuelle du scénario de cas d'utilisation « S'authentifier

»

- Description textuelle de cas d'utilisation «Mettre à jour son profil».

Nom du CU	Mettre à jour son profil
Acteurs	Client/Administrateur
Résumé	L'utilisateur modifie les informations de son profil.
Pré-conditions	L'utilisateur s'est authentifié.
Scénario Nominal	<p>1- L'utilisateur sélectionne l'option "My Profile".</p> <p>2- Le système affiche l'interface "Profile Settings".</p> <p>3- L'utilisateur modifie ses informations.</p> <p>4- L'utilisateur enregistre les modifications effectuées.</p> <p>5- Le système vérifie que les champs ne sont pas vides.</p>
Enchaînement Alternatif	<p>5.1 Champ vide.</p> <p>5.2 Le système affiche le message d'erreur : "The given String is empty or NULL".</p> <p>5.3 Retour au point 3 pour permettre au client de modifier à nouveau ses informations.</p>
Post-conditions	Les informations de l'utilisateur ont été modifiées avec succès.

TABLE 3.4 — Description textuelle du scénario de cas d'utilisation « Mettre à jour son profil »

- Description textuelle de cas d'utilisation «Changer mot de passe».

Nom du CU	Changer mot de passe
Acteurs	Administrateur/Client
Résumé	L'utilisateur modifie les informations de son profil.
Pré-conditions	L'utilisateur s'est authentifié.
Scénario Nominal	<p>1- L'utilisateur sélectionne l'option "My Profile".</p> <p>2- Le système affiche l'interface "Profile Settings".</p> <p>3- L'utilisateur souhaite modifier son mot de passe.</p> <p>4- L'utilisateur saisit son ancien mot de passe.</p> <p>5- L'utilisateur crée son nouveau mot de passe et enregistre les modifications.</p>
Enchaînement Alternatif	<p>5.1 Mot de passe ancien incorrect.</p> <p>5.1.1 Le système affiche le message d'erreur : "Invalid Old password. If you forget your password, click on forgot password below to recover your password".</p> <p>5.1.2 Retour au point 4 pour que l'utilisateur puisse saisir à nouveau son ancien mot de passe correctement.</p> <p>5.2 Nouveau mot de passe trop court.</p> <p>5.2.1 Le système affiche le message d'erreur : "Password should be at least 6 characters".</p> <p>5.2.2 Retour au point 5 pour que l'utilisateur puisse créer un nouveau mot de passe valide.</p>
Post-conditions	Le mot de passe de l'utilisateur a été modifié avec succès.

TABLE 3.5 — Description textuelle du scénario de cas d'utilisation « Changer mot de passe »

- Description textuelle de cas d'utilisation «Modifier l'espace».

Nom du CU	Modifier l'espace
Acteurs	Client
Résumé	L'utilisateur modifie l'espace à surveiller.
Pré-conditions	L'utilisateur s'est authentifié.
Scénario Nominal	<p>1- Le client accède à son profil.</p> <p>2- Le système affiche l'interface correspondant au profil du client.</p> <p>3- Le client souhaite modifier l'espace à surveiller.</p> <p>4- Le client effectue les modifications souhaitées concernant l'espace à surveiller et valide.</p> <p>5- Le système enregistre les changements et envoie une notification push à l'administrateur du client mettant le client en attente de validation de l'administrateur.</p>
Post-conditions	L'espace de surveillance du client a été modifié avec succès.

TABLE 3.6 — Description textuelle du scénario de cas d'utilisation « Modifier l'espace »

- Description textuelle de cas d'utilisation «Gérer les comptes d'administrateurs».

Nom du CU	Gérer les comptes d'administrateurs
Acteurs	Super Administrateur
Résumé	Le super administrateur peut se charger de la gestion des administrateurs.
Pré-conditions	Le super administrateur s'est authentifié.
Scénario Nominal	<p>1- Le super administrateur accède à son profil.</p> <p>2- Le système charge et affiche l'interface correspondant au profil du super administrateur.</p> <p>3- Le super administrateur demande la liste des administrateurs existants.</p> <p>4- Le système traite la demande du super administrateur et affiche la liste des administrateurs existants.</p>
Post-conditions	La liste des administrateurs est affichée.

TABLE 3.7 — Description textuelle du scénario de cas d'utilisation « Gérer les comptes d'administrateurs »

- Description textuelle de cas d'utilisation «Ajouter Admin».

Nom du CU	Ajouter Admin
Acteurs	Super Administrateur
Résumé	Le super administrateur peut ajouter des administrateurs.
Pré-conditions	Le super administrateur s'est authentifié.
Scénario Nominal	<p>1- Le super administrateur saisit les données du nouvel administrateur, telles que l'adresse e-mail, le mot de passe et la région attribuée.</p> <p>2- Le système vérifie les informations saisies.</p> <p>3- Le système enregistre le nouvel administrateur.</p> <p>4- Le super administrateur reçoit un message de succès.</p>
Enchaînement Alternatif	2.1 Le système affiche un message d'erreur si l'administrateur existe déjà ou si les champs sont non valides.
Post-conditions	Administrateur ajouté.

TABLE 3.8 — Description textuelle du scénario de cas d'utilisation « Ajouter Admin »

»

- Description textuelle de cas d'utilisation «Supprimer Admin».

Nom du CU	Supprimer Admin
Acteurs	Super Administrateur
Résumé	Le super administrateur peut supprimer des administrateurs.
Pré-conditions	Le super administrateur s'est authentifié.
Scénario Nominal	<p>1- Le super administrateur charge la liste des administrateurs.</p> <p>2- Le super administrateur choisit l'administrateur qu'il souhaite supprimer.</p> <p>3- Le système affiche une alerte de confirmation pour confirmer la suppression.</p> <p>4- Le super administrateur valide la suppression.</p> <p>5- Le système affiche un message de succès de suppression.</p>
Enchaînements Alternatifs	3.1 Si le super administrateur ne confirme pas la suppression, le système affiche la liste des administrateurs. La liste reste inchangée.
Post-conditions	Administrateur supprimé.

TABLE 3.9 — Description textuelle du scénario de cas d'utilisation « Supprimer Admin »

- Description textuelle de cas d'utilisation «Ajouter collaborateur».

Nom du CU	Ajouter collaborateur
Acteurs	Client
Résumé	Le client peut ajouter des collaborateurs pour participer à la surveillance de son espace et ceux-ci recevront une alerte en cas d'incendie.
Pré-conditions	Le client s'est authentifié.
Scénario Nominal	1- Le client accède à son profil. 2- Le système affiche l'interface correspondant au profil du client. 3- Le client souhaite ajouter un collaborateur. 4- Le client saisit les informations correspondantes au collaborateur telles que le nom, l'adresse e-mail et le numéro de téléphone, puis valide.
Post-conditions	Collaborateur ajouté.

TABLE 3.10 — Description textuelle du scénario de cas d'utilisation « Ajouter collaborateur »

3.2.3 Conception

Dans cette section, nous présentons les diagrammes de séquences et de classes utilisés dans le cadre du premier sprint. Les diagrammes de séquence illustrent la séquence des messages échangés entre les objets nécessaires pour réaliser les fonctionnalités du scénario, tandis que les diagrammes de classes décrivent la structure du système.

3.2.3.1 Diagrammes de séquence

Diagramme de séquence d'inscription :

La figure 3.2 illustre le déroulement du cas d'utilisation “s’inscrire”. Le client remplit les champs requis et valide les informations. Si les données sont valides, le système enregistre l’inscription et affiche un message de confirmation. En cas de données invalides, un message d’erreur est affiché pour informer le client du problème.

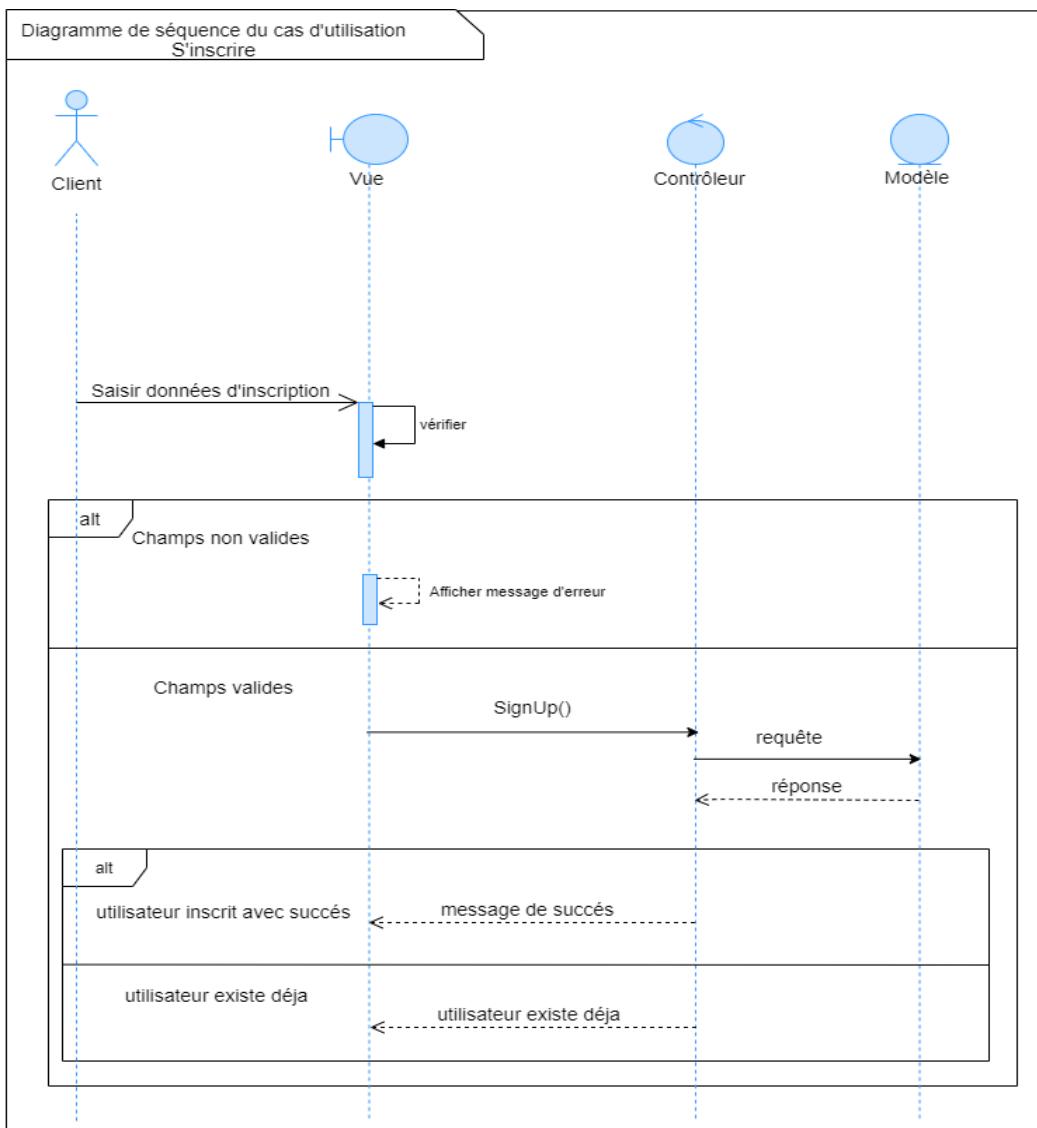


FIGURE 3.2 — Sprint 1 -Diagramme de séquence “s’inscrire”

Diagramme de séquence d’authentification :

La figure 3.3 représente le déroulement du processus d’authentification. L’utilisateur entre ses identifiants de connexion. Le système vérifie la validité de ces identifiants. Si les identifiants sont corrects, l’utilisateur accède à l’application, sinon, un message d’erreur d’authentification est affiché.

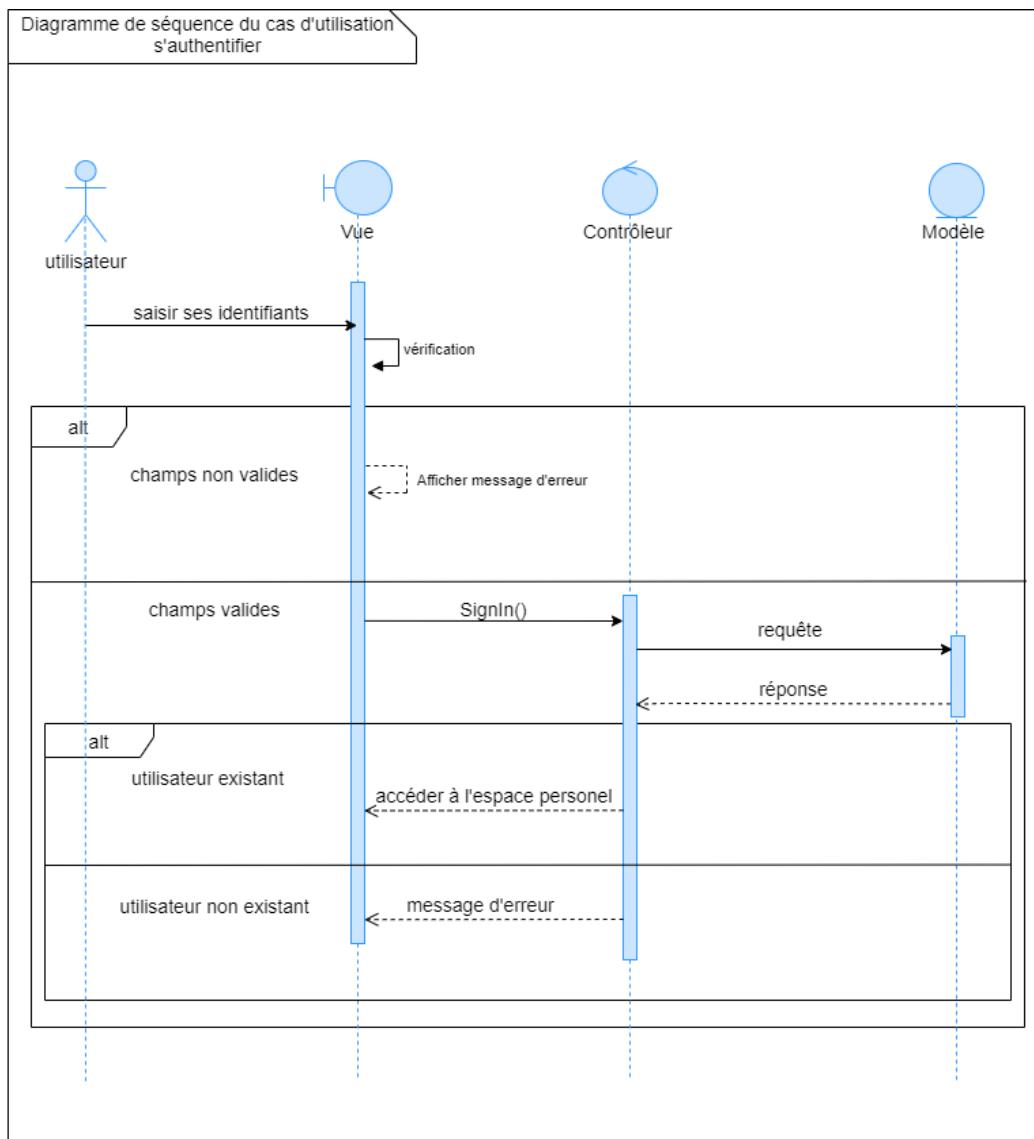


FIGURE 3.3 — Sprint 1 -Diagramme de séquence “s’authentifier”

Diagramme de séquence du cas d'utilisation “Mettre à jour son profil” :

La figure 3.4 représente le déroulement du processus “Mettre à jour son profil”. Une fois authentifié, l’utilisateur peut mettre à jour son profil en remplittant les champs du formulaire et en validant. Le système vérifie les données et met à jour le profil de l’utilisateur, affichant une confirmation en cas de succès. En cas d’erreur, un message d’erreur approprié guide l’utilisateur pour corriger les données fournies.

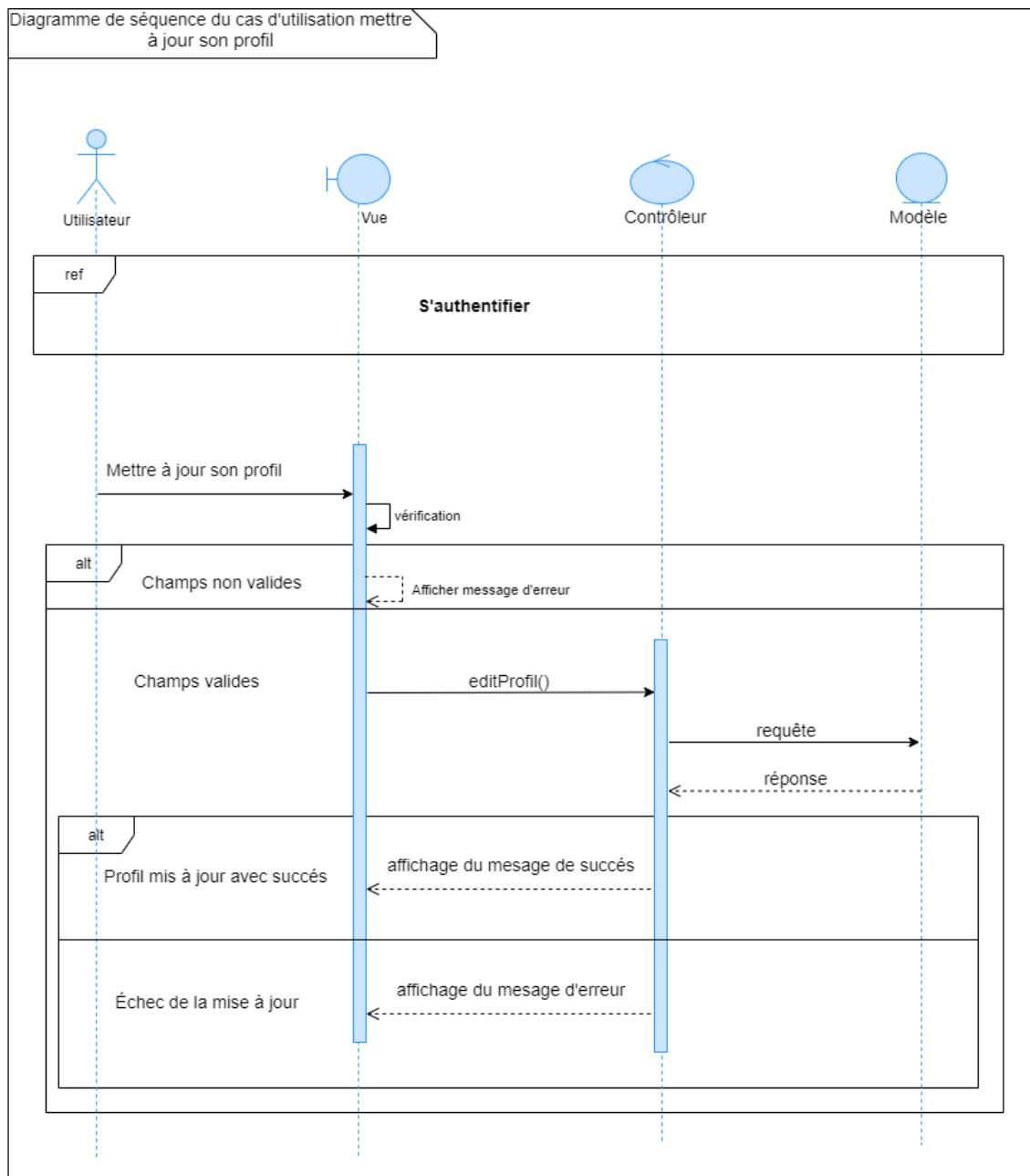


FIGURE 3.4 — Sprint 1 -Diagramme de séquence “Mettre à jour son profil”

Diagramme de séquence du cas d'utilisation “Changer mot de passe” :

La figure 3.4 représente le déroulement du processus “Changer mot de passe”. Une fois authentifié, l’utilisateur peut modifier son mot de passe en saisissant l’ancien et le nouveau mot de passe dans un formulaire. Après validation, le système vérifie la validité de l’ancien mot de passe et enregistre le nouveau. Une confirmation est affichée pour indiquer que le changement a été effectué avec succès. En cas d’erreur, un message d’erreur approprié guide l’utilisateur.

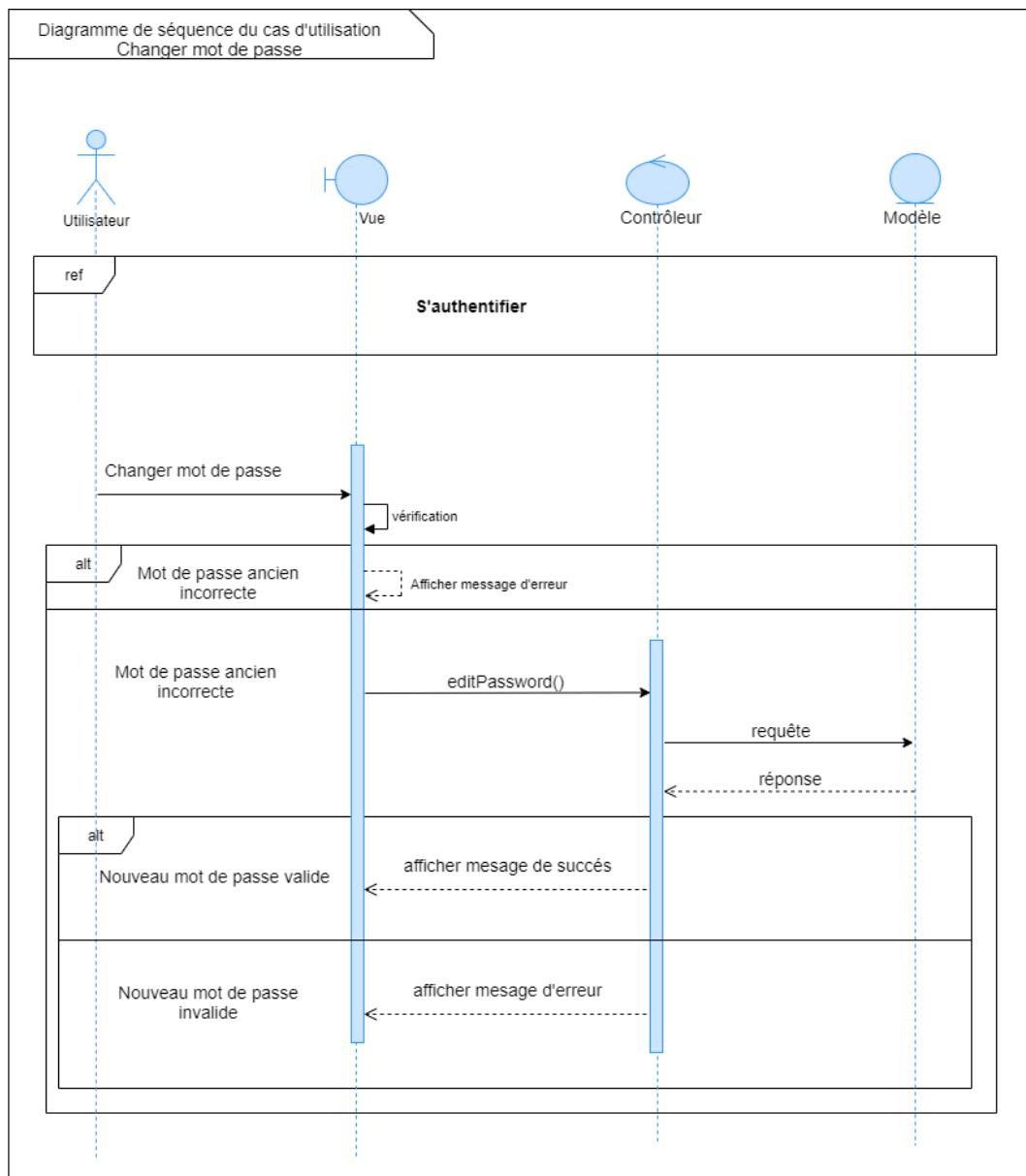


FIGURE 3.5 — Sprint 1 -Diagramme de séquence “Changer mot de passe”

Diagramme de séquence du cas d'utilisation “Modifier espace” :

La figure 3.6 représente le déroulement du processus “modifier espace”. Une fois authentifié, l’utilisateur peut modifier son espace de surveillance en choisissant une nouvelle zone sur une carte et en remplissant les informations correspondantes. Après validation, le système vérifie les données fournies et met à jour les informations de l’espace de surveillance. Une confirmation est affichée pour indiquer que les modifications ont été effectuées avec succès.

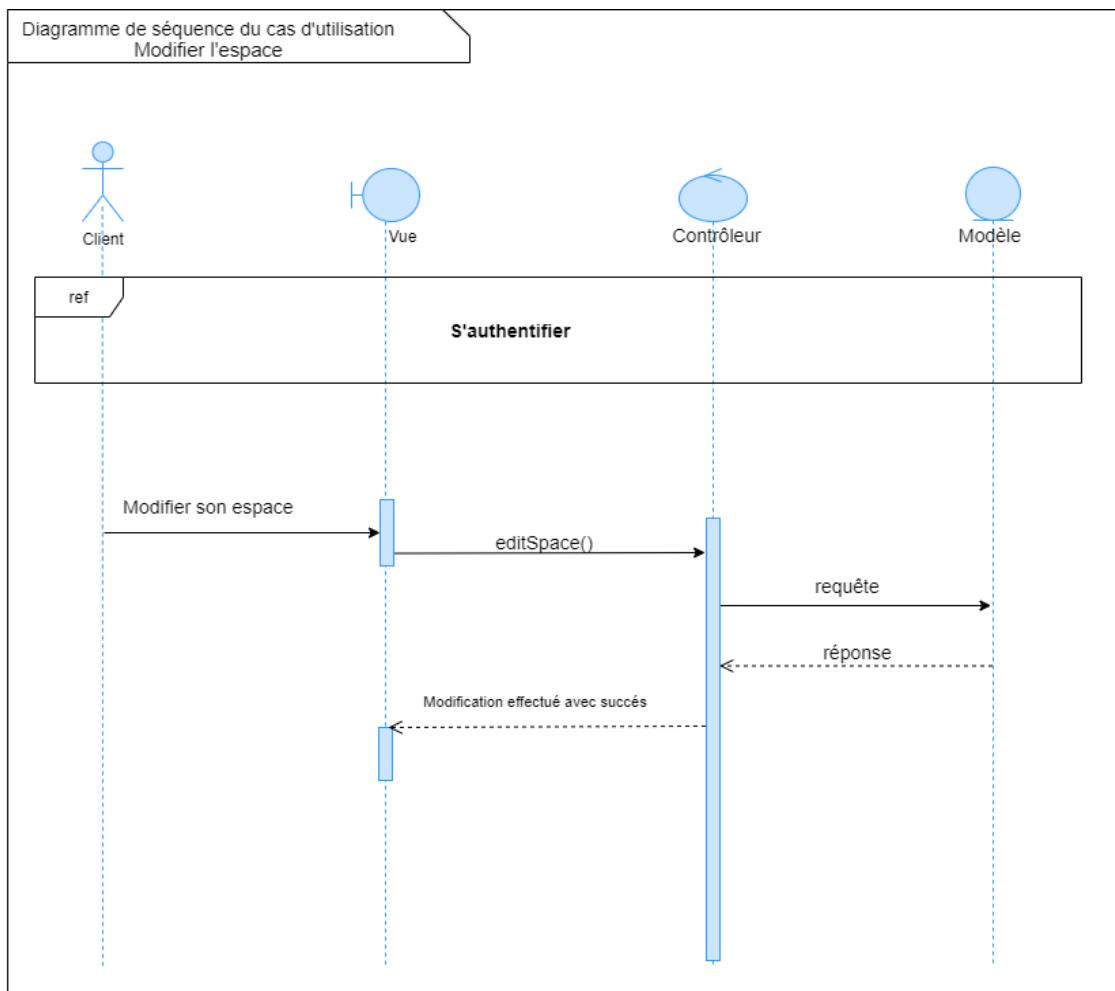


FIGURE 3.6 — Sprint 1 -Diagramme de séquence “modifier espace”

Diagramme de séquence du cas d'utilisation “Gérer les comptes d’administrateurs” :

La figure 3.7 représente le déroulement du processus “Gérer les comptes d’administrateurs”. Elle présente les actions du super administrateur pour la gestion des comptes d’administrateurs. Après s’être authentifié, le super administrateur peut supprimer un administrateur existant en le sélectionnant dans la liste des administrateurs. Il peut également ajouter un nouvel administrateur en remplittant un formulaire avec les informations requises. Ainsi, le super administrateur a le pouvoir de supprimer et d’ajouter des comptes d’administrateurs, facilitant ainsi la gestion de l’application.

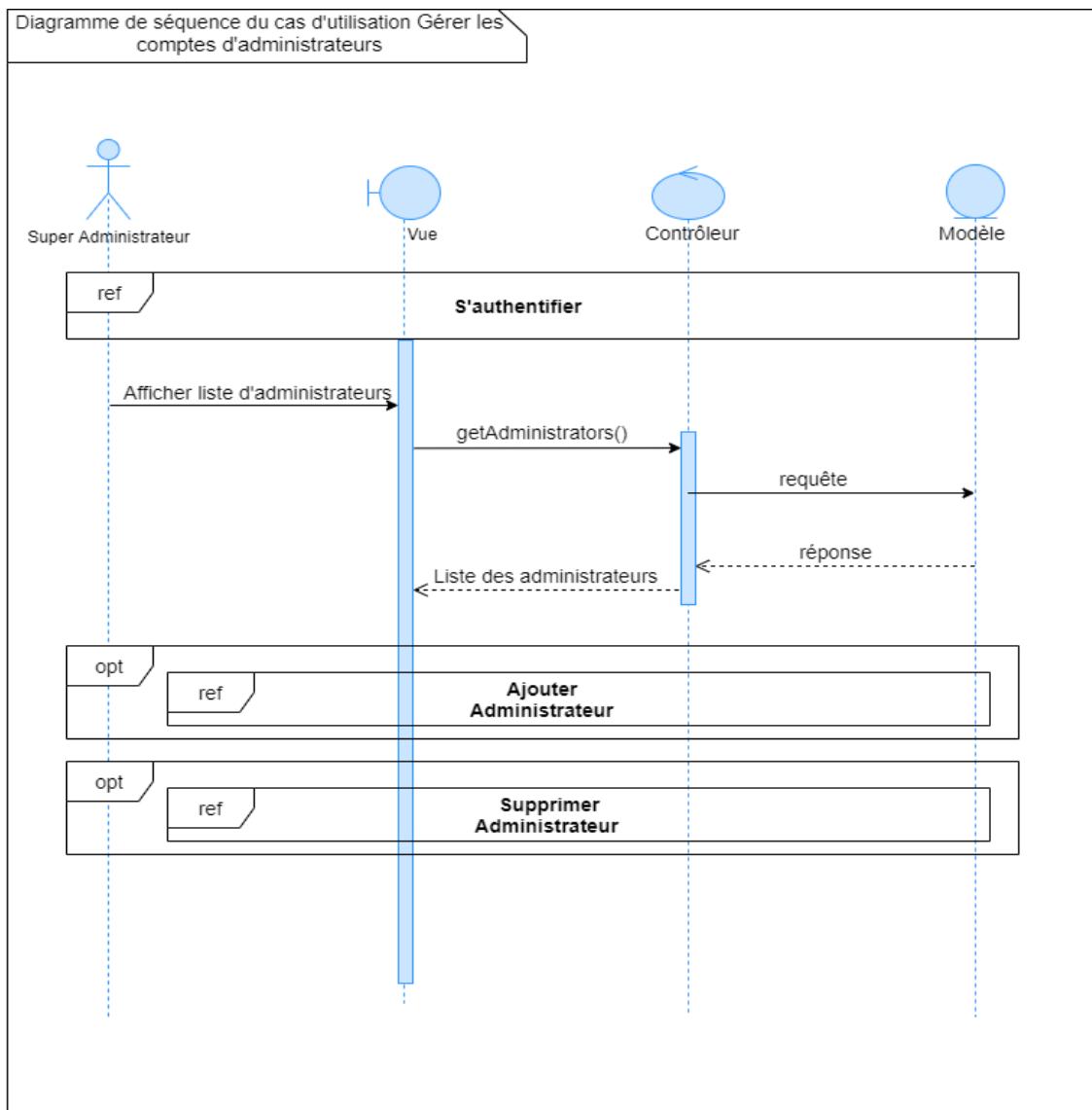


FIGURE 3.7 — Sprint 1 -Diagramme de séquence “Gérer les comptes d’administrateurs”

Diagramme de séquence du cas d’utilisation “Ajouter admin” :

La figure 3.8 représente le déroulement du processus “Ajouter admin”. Après s’être authentifié, le super administrateur remplit les informations nécessaires pour le nouvel administrateur. Si les informations sont valides, l’administrateur est ajouté avec succès. En cas d’informations invalides, une erreur est renvoyée, indiquant au super administrateur de corriger les champs incorrects.

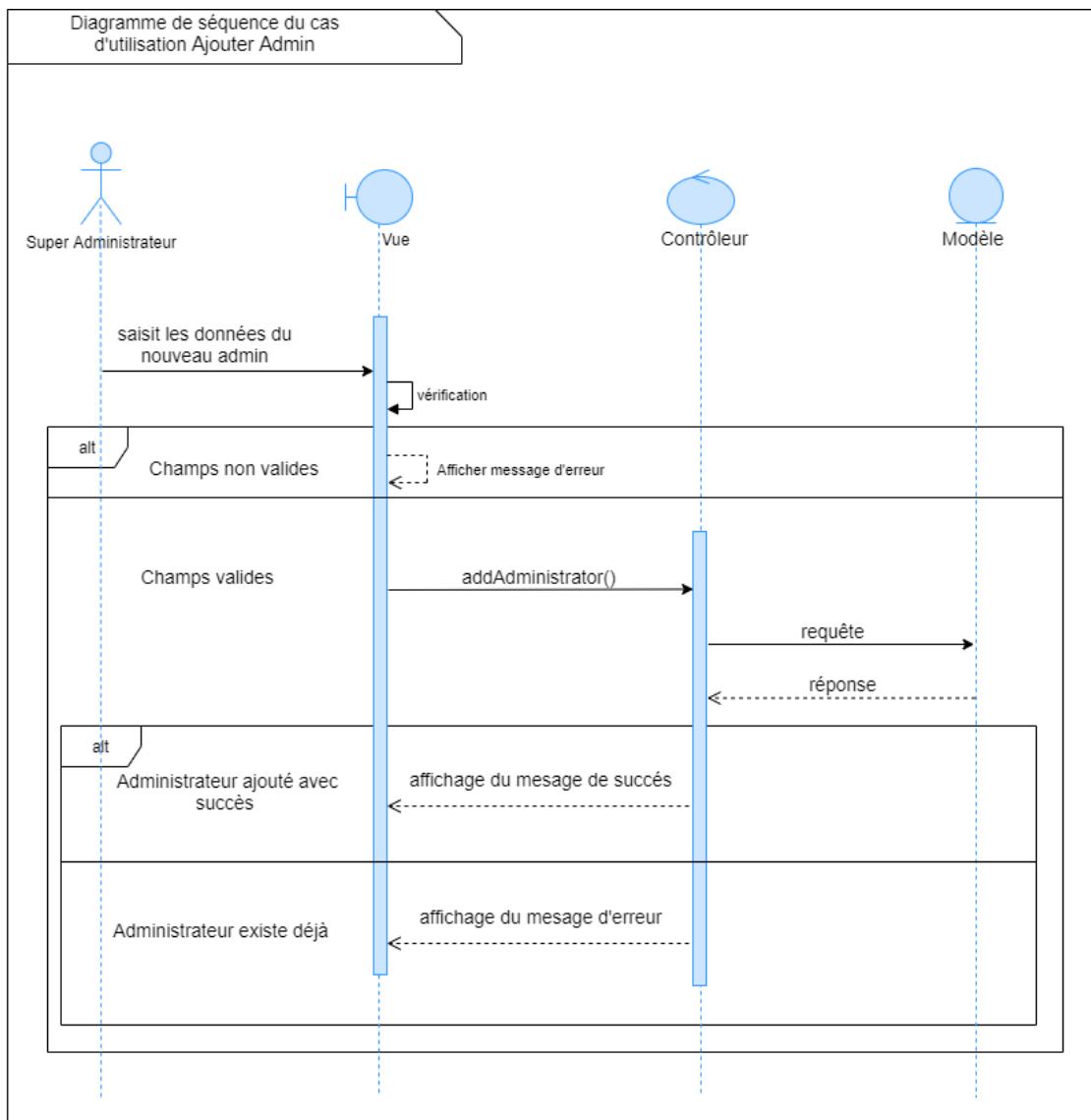


FIGURE 3.8 — Sprint 1 -Diagramme de séquence “Ajouter admin”

Diagramme de séquence du cas d'utilisation “Supprimer admin” :

La figure 3.9 représente le déroulement du processus “Supprimer admin” par le super administrateur. Après s’être authentifié, le super administrateur sélectionne l’administrateur à supprimer. Ensuite, il est invité à confirmer la suppression. Si la confirmation est donnée, l’administrateur est supprimé avec succès. Sinon, le super administrateur est redirigé vers la liste des administrateurs sans effectuer la suppression.

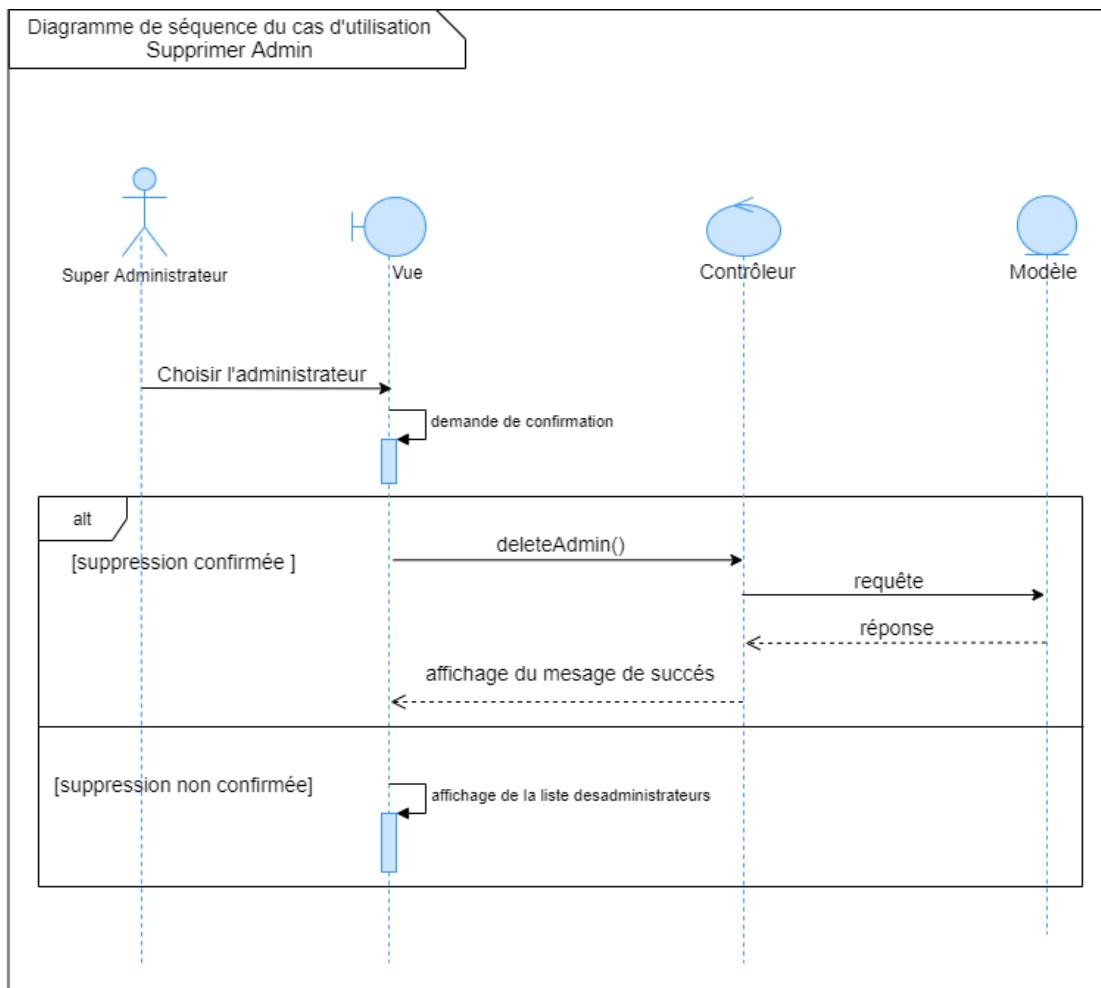


FIGURE 3.9 — Sprint 1 -Diagramme de séquence “Supprimer admin”

Diagramme de séquence du cas d'utilisation “Ajouter collaborateur” :

La figure 3.29 représente le déroulement du processus “Ajouter collaborateur” par le client. Après s’être authentifié, le client remplit les champs requis pour le collaborateur et valide l’ajout. Le système enregistre avec succès le collaborateur ajouté.

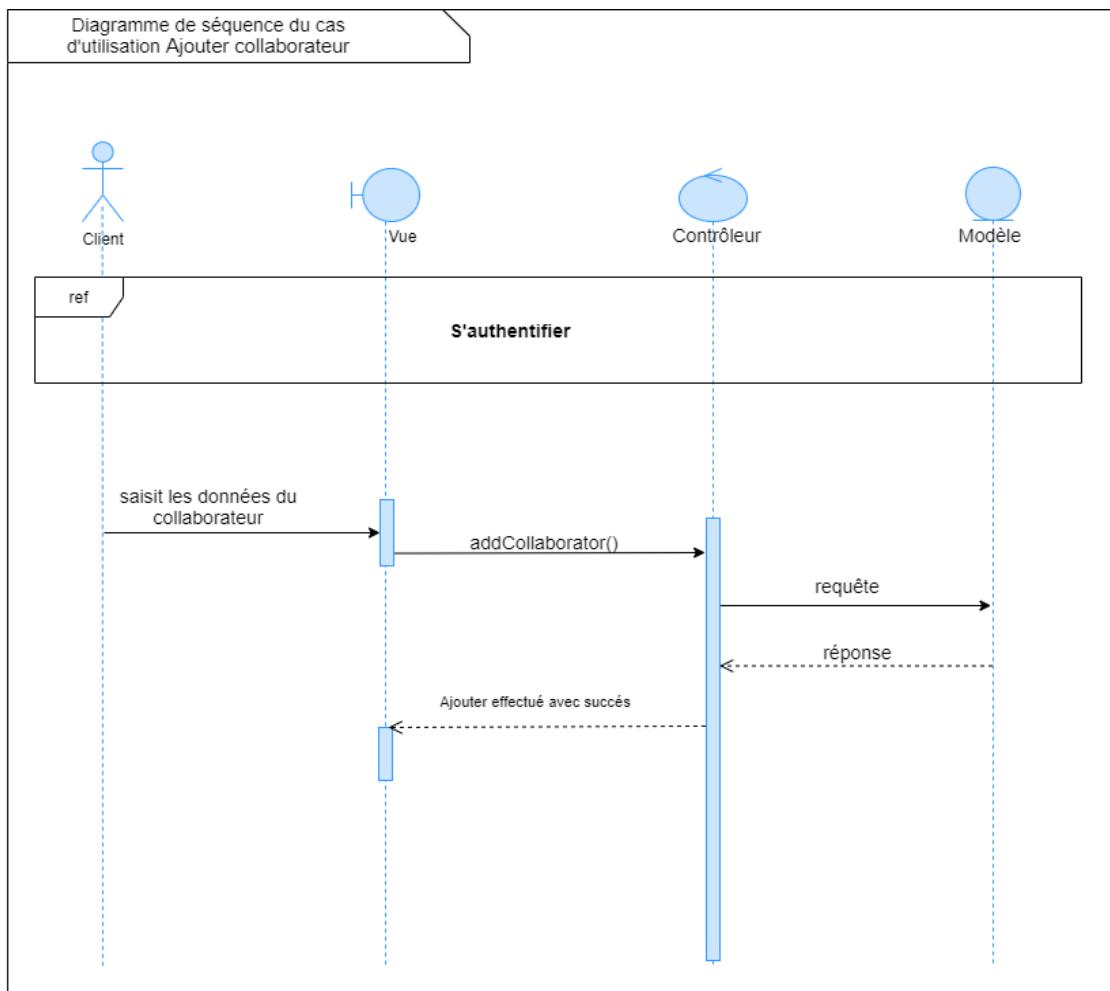


FIGURE 3.10 — Sprint 1 -Diagramme de séquence “Ajouter collaborateur”

3.2.3.2 Diagramme de Classes

Le diagramme de classes est un outil de modélisation utilisé pour représenter la structure d'un système. Il permet de décrire les entités et leurs relations au sein de notre application.

Les classes :

Les classes sont des éléments qui permettent de décrire de manière formelle un groupe d'objets partageant des caractéristiques et une sémantique commune. Elles sont représentées graphiquement par un rectangle divisé en trois sections distinct. La section supérieure correspond au nom de la classe, la section centrale présente les propriétés ou les attributs de la classe, tandis que la section inférieure liste les méthodes associées à la classe. Cette représentation visuelle permet de structurer et d'organiser les informations relatives à la classe, en mettant en évidence ses différents composants.

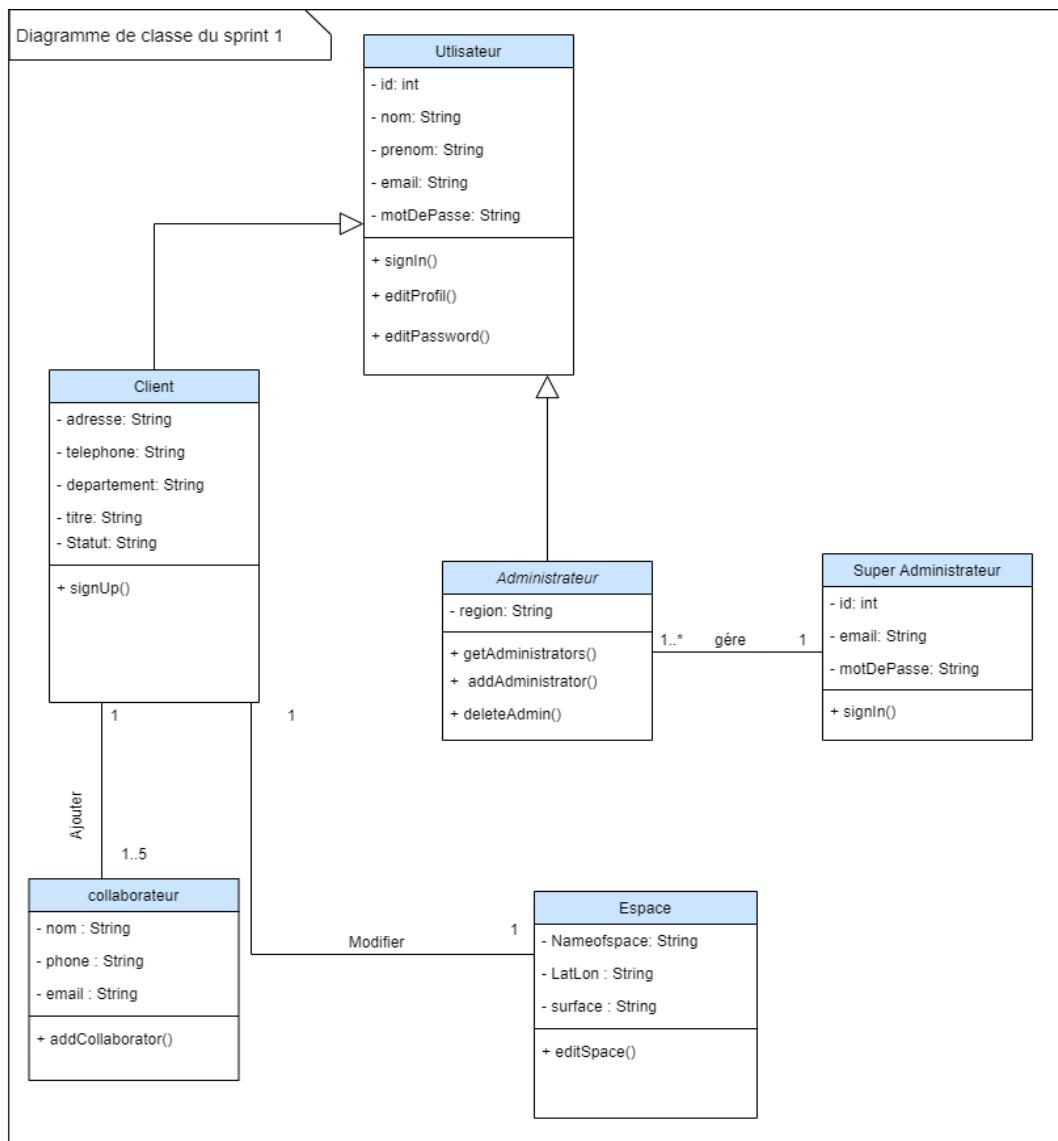


FIGURE 3.11 — Sprint 1 -Diagramme de Classes

3.2.4 Réalisation

Après avoir analysé le premier sprint et la phase de conception, cette section présente quelques étapes de développement des interfaces dans le cadre de ce sprint.

- Configuration de la base de données Firestore (No SQL Database)

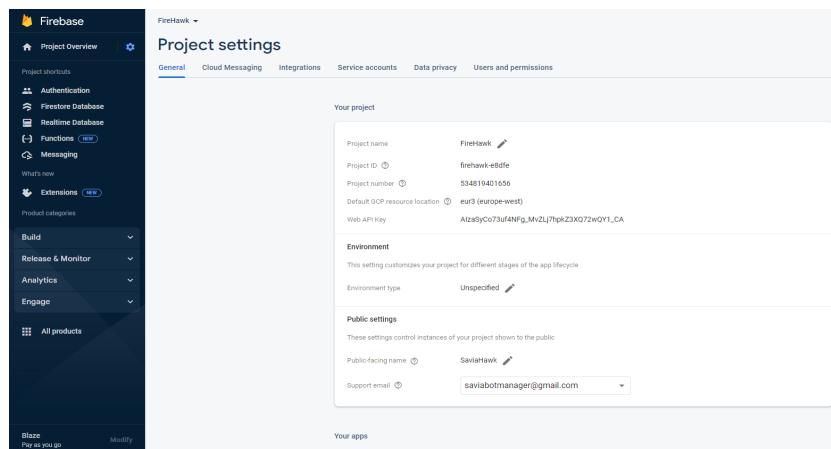


FIGURE 3.12 — Sprint 1 -Base de données No SQL Firestore ”

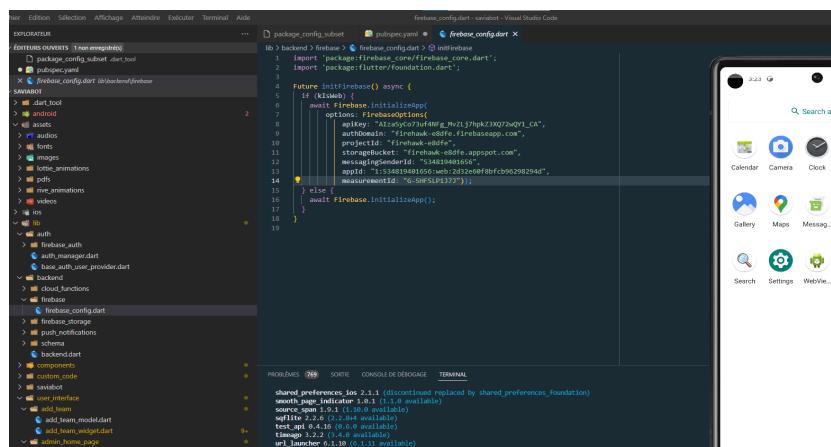


FIGURE 3.13 — Sprint 1 -Intégration de la base de données à l’application Savia-Bot”

- Activation de L’authentification par Email - Firebase

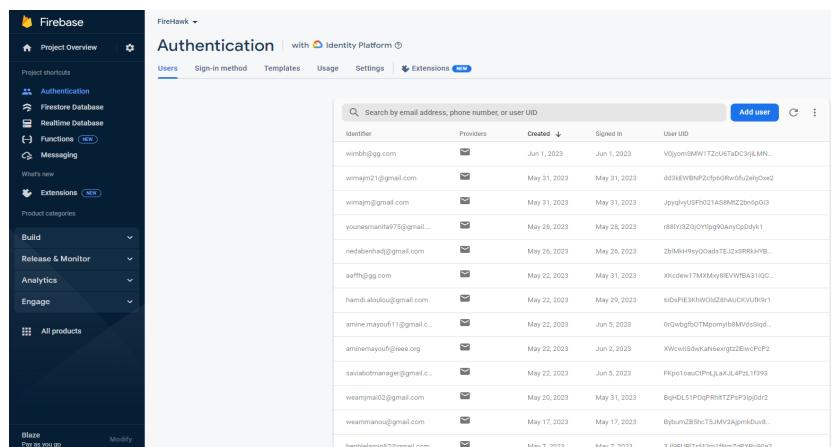


FIGURE 3.14 — Sprint 1 -Base de donnée No SQL Firestore

- Intégration de l'API Google Maps

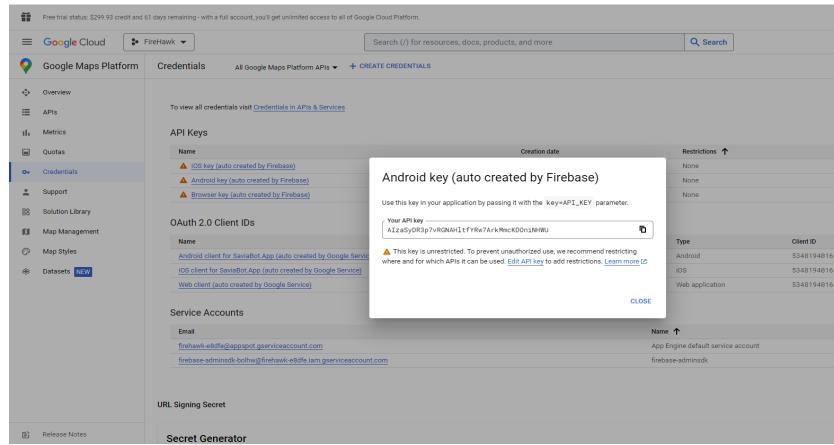


FIGURE 3.15 — Sprint 1 -Initialisation de l'API Google Maps

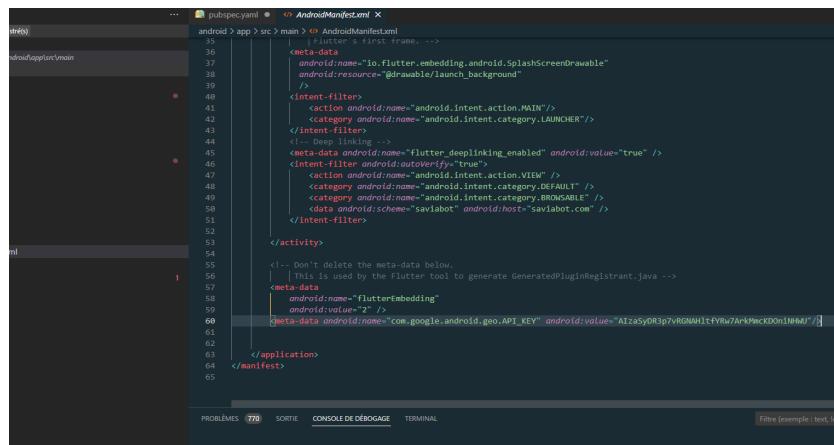


FIGURE 3.16 — Sprint 1 -Intégration de l'API Google Maps dans l'application SaviaBot

- Réalisation des interfaces :

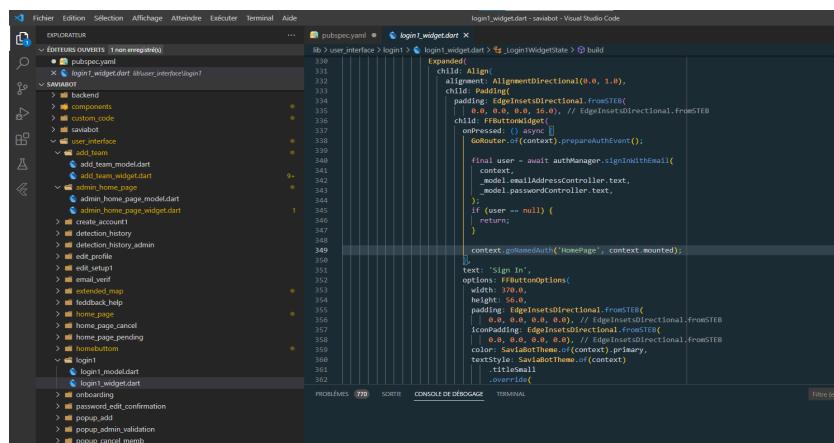


FIGURE 3.17 — Sprint 1 -Développement de Widget Login (Authentification)

3.2.5 Validation

Après avoir réalisé les interface, cette section présente les tests des interfaces qui ont été développées dans le cadre de ce sprint.

- Interface d'inscription :

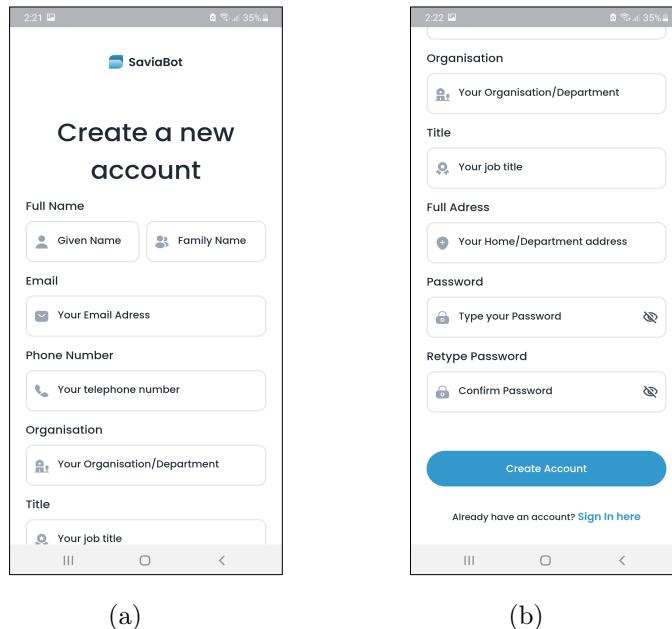


FIGURE 3.18 — Sprint 1-Interface d'inscription

- Interface d'authentification :

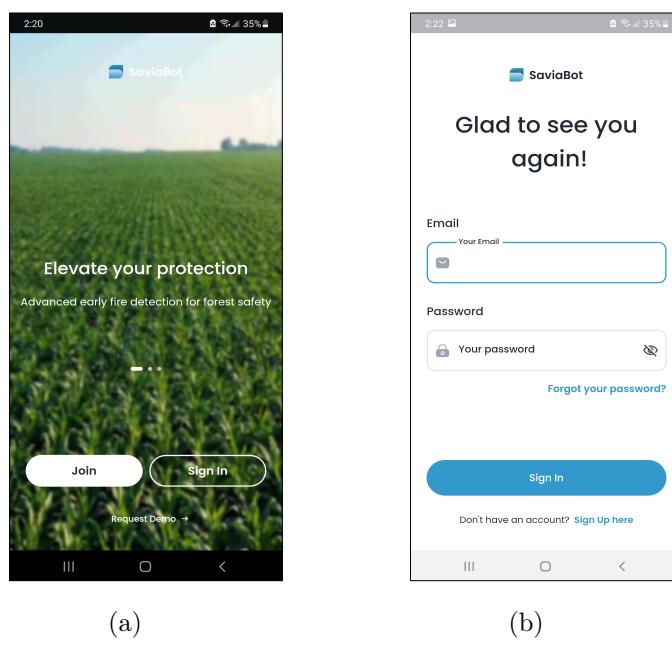
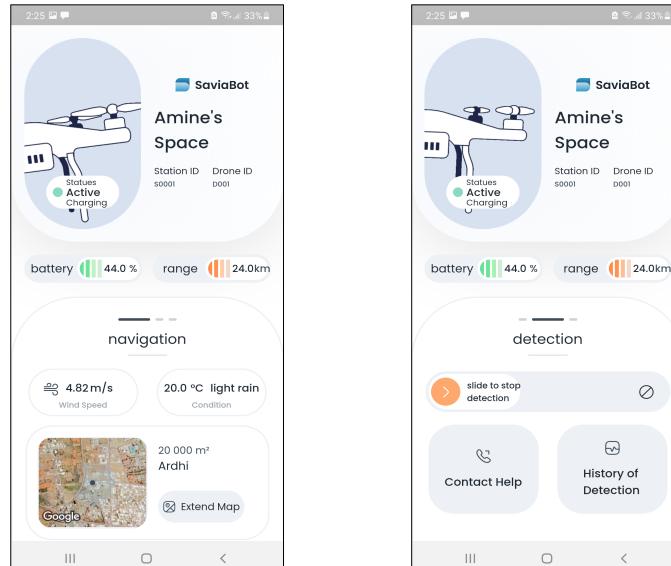


FIGURE 3.19 — Sprint 1-Interface d'inscription

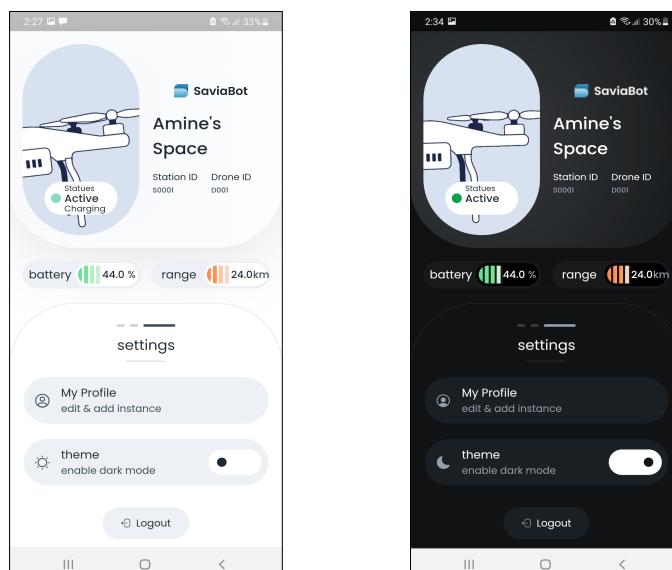
- Interface Home Client :

Une fois le Client est authentifié, il est redirigé vers la page home.



(a)

(b)



(c)

(d)

FIGURE 3.20 — Sprint 1 - interface HomePage Client

- Interface Profil Settings :

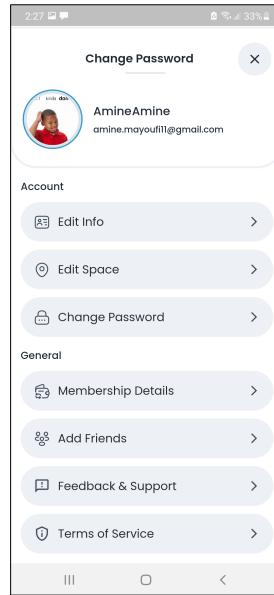


FIGURE 3.21 — Sprint 1 – interface Profil Settings

- Interface Edit Profile :

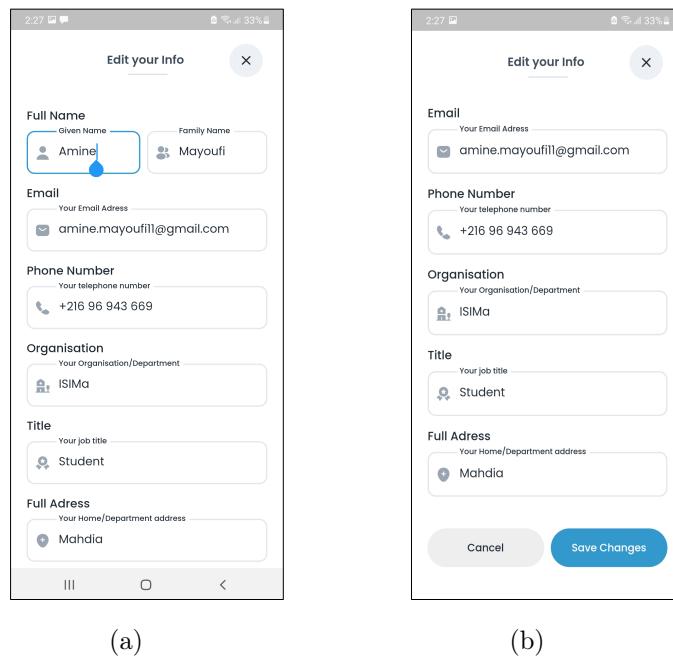


FIGURE 3.22 — Sprint 1 - interface Edit Profile

- Interface Edit Password :

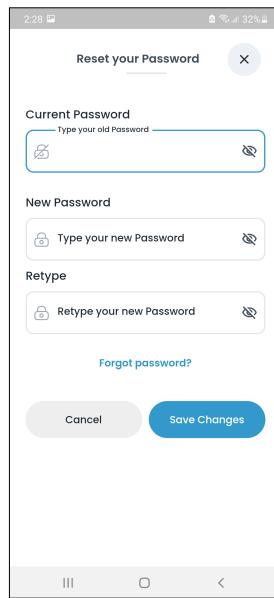


FIGURE 3.23 — Sprint 1 – interface Edit Password

- Interface Edit Space :



FIGURE 3.24 — Sprint 1 – interface Edit Space

- Interface Home Super Administrateur :

Une fois le super administrateur est authentifié, il est redirigé vers la page home.

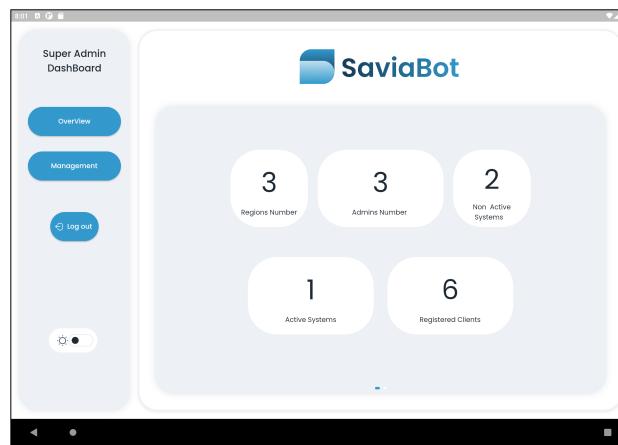


FIGURE 3.25 — Sprint 1 - interface Home Super Administrateur

- Interface Management :

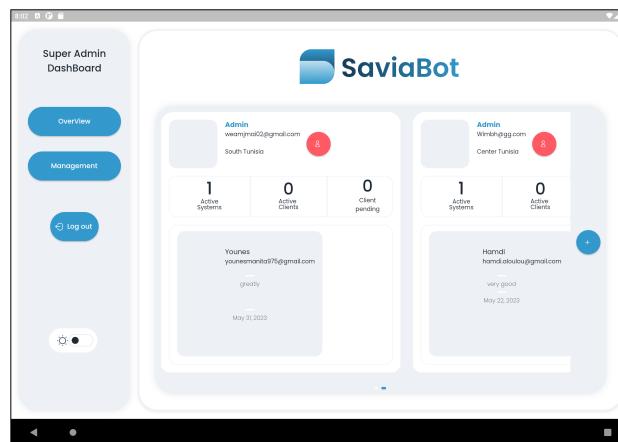


FIGURE 3.26 — Sprint 1 - interface Management

- Interface Ajouter Admin :

The screenshot shows a modal dialog titled 'Admin Account Creation'. It contains three input fields: 'Admin Email Address' (with placeholder 'admin's Email Address'), 'Admin Password' (with placeholder 'Admin's account password'), and 'Admin Region' (a dropdown menu with placeholder 'Please select a region...'). At the bottom of the dialog is a blue 'Create Admin' button.

FIGURE 3.27 — Sprint 1 - interface Ajouter Admin

- Interface Supprimer Admin :

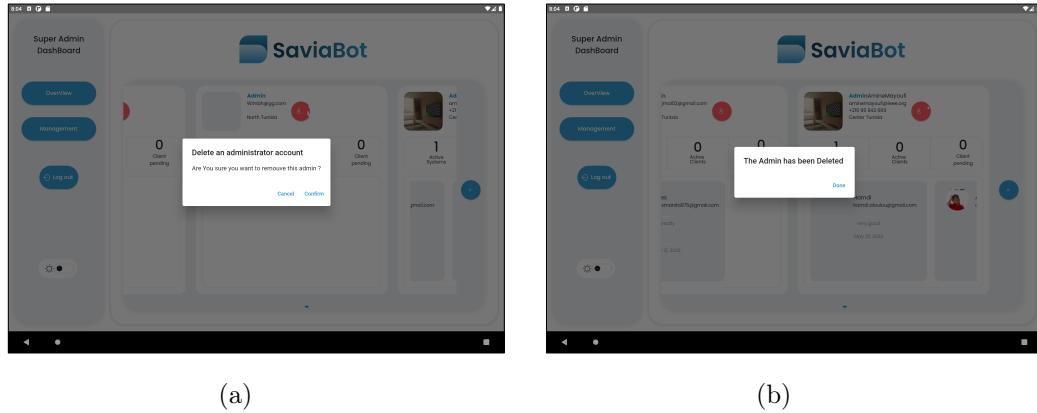


FIGURE 3.28 — Sprint 1 - Interface Supprimer Admin

- Interface Ajouter Collaborateur :

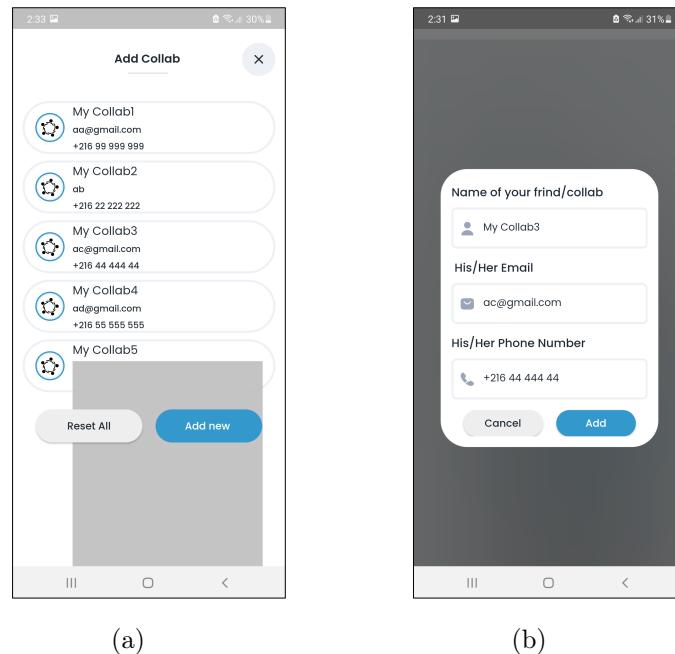


FIGURE 3.29 — Sprint 1 - Interface Ajouter Collaborateur

3.3 SPRINT 2 : "Planification des missions et avis clients"

Dans cette section, nous présentons, tout d'abord, l'organisation et le backlog du deuxième sprint. Ensuite, nous présentons la phrase d'analyse et la solution conceptuelle. Enfin, les différentes réalisations sont illustrées.

3.3.1 Backlog du SPRINT 2

Le tableau 3.11 donne un aperçu détaillé sur le Backlog du sprint qui prend en charge la fonctionnalité « Planification des missions et avis clients ».

ID	Sprint	User Story	Jour
2	Planification des missions avis clients	Réalisation de la fonctionnalité "Accéder aux commentaires des clients"	2
		Réalisation de la fonctionnalité "Afficher les données des admins"	2
		Réalisation de la fonctionnalité "Consulter les demandes des clients"	3
		Réalisation de la fonctionnalité "Confirmer les demandes des clients"	2
		Réalisation de la fonctionnalité "Contrôler les missions des drones"	3
		Réalisation de la fonctionnalité "Suivre l'état des drones"	1
		Réalisation de la fonctionnalité "Sélectionner son espace de surveillance"	3
		Réalisation de la fonctionnalité "Donner des commentaires sur les services qu'il a reçus"	3
Test			4

TABLE 3.11 — Backlog du sprint 2 : "Planification des missions et avis clients"

3.3.2 Analyse

Pendant cette étape, nous décrivons en détail les diverses fonctionnalités du système, un diagramme de cas d'utilisation, les descriptions textuelles pour chaque cas, ainsi que les diagrammes de séquence.

3.3.2.1 Diagramme de cas d'utilisation

La figure représente le diagramme de cas d'utilisation pour le sprint 2.

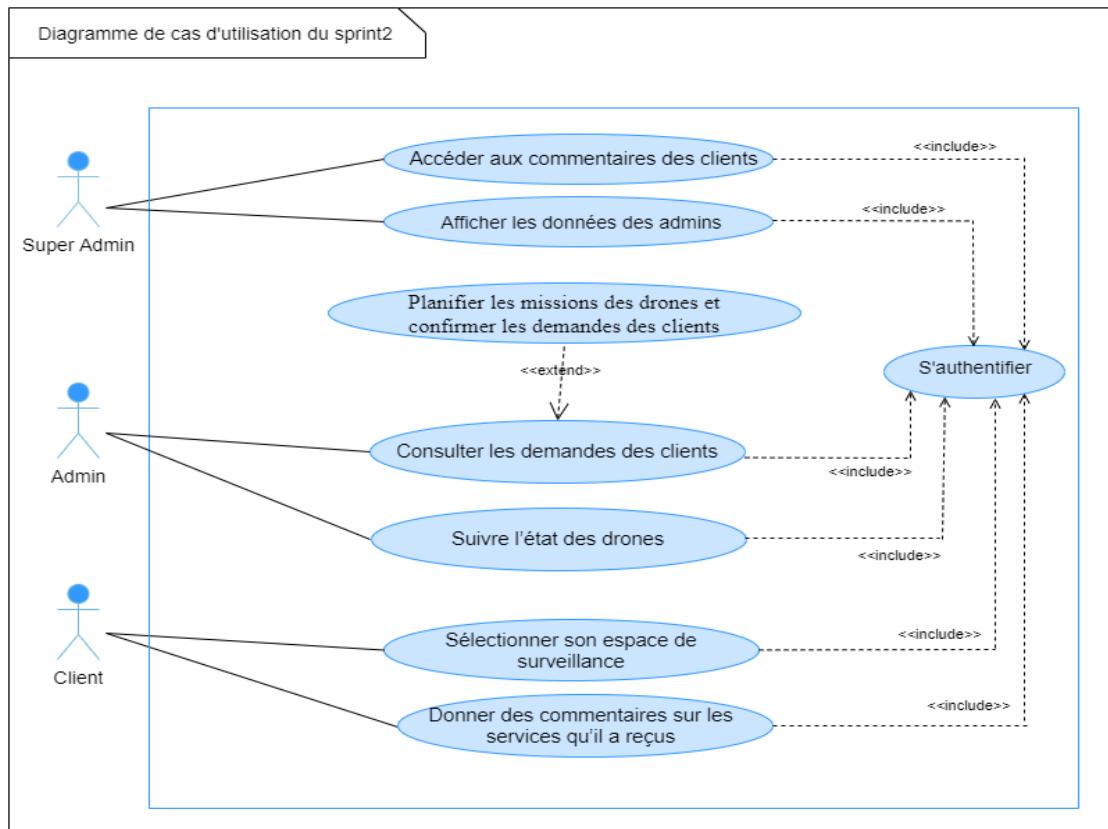


FIGURE 3.30 — Sprint 2 -Diagramme de Cas d'utilisation

3.3.2.2 Description textuelle de cas d'utilisation

- Description textuelle de cas d'utilisation «Accéder aux commentaires des clients»

Nom du CU	Accéder aux commentaires des clients
Acteurs	Super administrateur
Résumé	Le super administrateur accède aux commentaires laissés par les clients.
Pré-conditions	Le Super administrateur s'est authentifié.
Scénario Nominal	<p>1- Le super administrateur accède à son profil.</p> <p>2- Le système charge et affiche l'interface correspondant au profil du super administrateur.</p> <p>3- Le super administrateur accède à la section "management".</p> <p>4- Le système affiche la liste des administrateurs et présente les commentaires disponibles des clients pour chaque administrateur.</p>
Post-conditions	Le super administrateur a pu consulter les commentaires des clients.

TABLE 3.12 — Description textuelle du scénario de cas d'utilisation « Accéder aux commentaires des clients »

- Description textuelle de cas d'utilisation «Afficher les données des admins»

Nom du CU	Afficher les données des admins
Acteurs	Super administrateur
Résumé	Le super administrateur peut afficher les données des administrateurs.
Pré-conditions	Le Super administrateur s'est authentifié.
Scénario Nominal	<p>1- Le super administrateur charge la liste des administrateurs.</p> <p>2- Le super administrateur choisit l'administrateur dont il souhaite observer les données.</p> <p>3- Le système affiche les détails et les activités en cours de l'administrateur sélectionné, tels que les clients en attente, les systèmes actifs, les clients actifs et la liste des commentaires.</p>
Post-conditions	Le super administrateur a pu afficher les données de l'administrateur sélectionné.

TABLE 3.13 — Description textuelle du scénario de cas d'utilisation « Afficher les données des admins »

- Description textuelle de cas d'utilisation «Consulter les demandes des clients»

Nom du CU	Consulter les demandes des clients
Acteurs	Administrateur
Résumé	L'administrateur consulte les demandes des clients reçues.
Pré-conditions	L'administrateur s'est authentifié.
Scénario Nominal	<p>1- L'administrateur reçoit une notification push pour être informé d'une nouvelle demande.</p> <p>2- L'administrateur charge l'interface des clients en attente.</p> <p>3- La liste des clients en attente se charge, triée par ordre de priorité décroissante.</p> <p>4- L'administrateur consulte les détails de la première demande.</p>
Post-conditions	L'administrateur a consulté les demandes des clients reçues.

TABLE 3.14 — Description textuelle du scénario de cas d'utilisation « Consulter les demandes des clients »

- Description textuelle de cas d'utilisation «Planifier les missions des drones et Confirmer les demandes des clients»

Nom du CU	Planifier les missions des drones et confirmer les demandes des clients
Acteurs	Administrateur
Résumé	L'administrateur planifie les missions des drones et confirme les demandes des clients.
Pré-conditions	L'administrateur s'est authentifié.
Scénario Nominal	<p>1- L'administrateur sélectionne une demande pour l'examiner et la valider.</p> <p>2- L'administrateur choisit le chemin à parcourir par le drone et la position de station de contrôle pour cette demande en utilisant un algorithme de calcul approprié.</p> <p>3- L'administrateur enregistre les modifications et valide la demande.</p>
Post-conditions	L'administrateur a contrôlé avec succès les missions des drones et confirmé les demandes.

TABLE 3.15 — Description textuelle du scénario de cas d'utilisation « Planifier les missions des drones et confirmer les demandes des clients »

- Description textuelle de cas d'utilisation «Suivre l'état des drones»

Nom du CU	Suivre l'état des drones
Acteurs	Administrateur
Résumé	L'administrateur surveille l'état des drones.
Pré-conditions	L'administrateur s'est authentifié.
Scénario Nominal	<p>1- L'administrateur accède à son profil.</p> <p>2- Le système charge et affiche l'interface correspondant au profil de l'administrateur.</p> <p>3- L'administrateur charge la liste des clients actifs.</p> <p>4- L'administrateur sélectionne un client pour suivre l'état de son drone.</p> <p>5- Le système affiche l'état actuel du drone.</p>
Post-conditions	L'administrateur a suivi avec succès l'état du drone du client sélectionné.

TABLE 3.16 — Description textuelle du scénario de cas d'utilisation « Suivre l'état des drones »

- Description textuelle de cas d'utilisation «Sélectionner son espace de surveillance»

Nom du CU	Sélectionner son espace de surveillance
Acteurs	Client
Résumé	Le client sélectionne l'espace qu'il souhaite surveiller.
Pré-conditions	Le Client s'est authentifié.
Scénario Nominal	<p>1- Le système affiche une carte permettant au client de sélectionner son espace de surveillance.</p> <p>2- Le client choisit l'espace en cliquant sur la carte.</p> <p>3- Le client remplit le formulaire avec le nom de l'espace et la taille de surface.</p> <p>4- Le système affiche l'interface de paiement.</p> <p>5- Le client choisit le mode de paiement approprié et valide la sélection.</p>
Post-conditions	Le client a sélectionné l'espace qu'il souhaite surveiller.

TABLE 3.17 — Description textuelle du scénario de cas d'utilisation « Sélectionner son espace de surveillance »

- Description textuelle de cas d'utilisation «Donner des commentaires sur les services qu'il a reçus»

Nom du CU	Donner des commentaires sur les services qu'il a reçus
Acteurs	Client
Résumé	Le client peut donner son avis sur le système.
Pré-conditions	Le Client s'est authentifié.
Scénario Nominal	<p>1- Le client accède à son profil.</p> <p>2- Le système affiche l'interface correspondant au profil du client.</p> <p>3- Le client souhaite écrire un commentaire.</p> <p>4- Le client saisit son avis dans le champ dédié et valide.</p> <p>5- Le système affiche un message de confirmation de l'enregistrement du commentaire.</p>
Post-conditions	L'avis du client a été enregistré et envoyé au super administrateur.

TABLE 3.18 — Description textuelle du scénario de cas d'utilisation « Donner des commentaires sur les services qu'il a reçus »

3.3.3 Conception

3.3.3.1 Diagramme de séquence :

- Diagramme de séquence du cas d'utilisation “Accéder aux commentaires des clients” :

La figure3.41 représente le déroulement du processus “Accéder aux commentaires des clients”. Après avoir réussi l'authentification, le super administrateur demande la liste des administrateurs disponibles. Ensuite, il sélectionne un administrateur spécifique pour consulter les avis des clients.

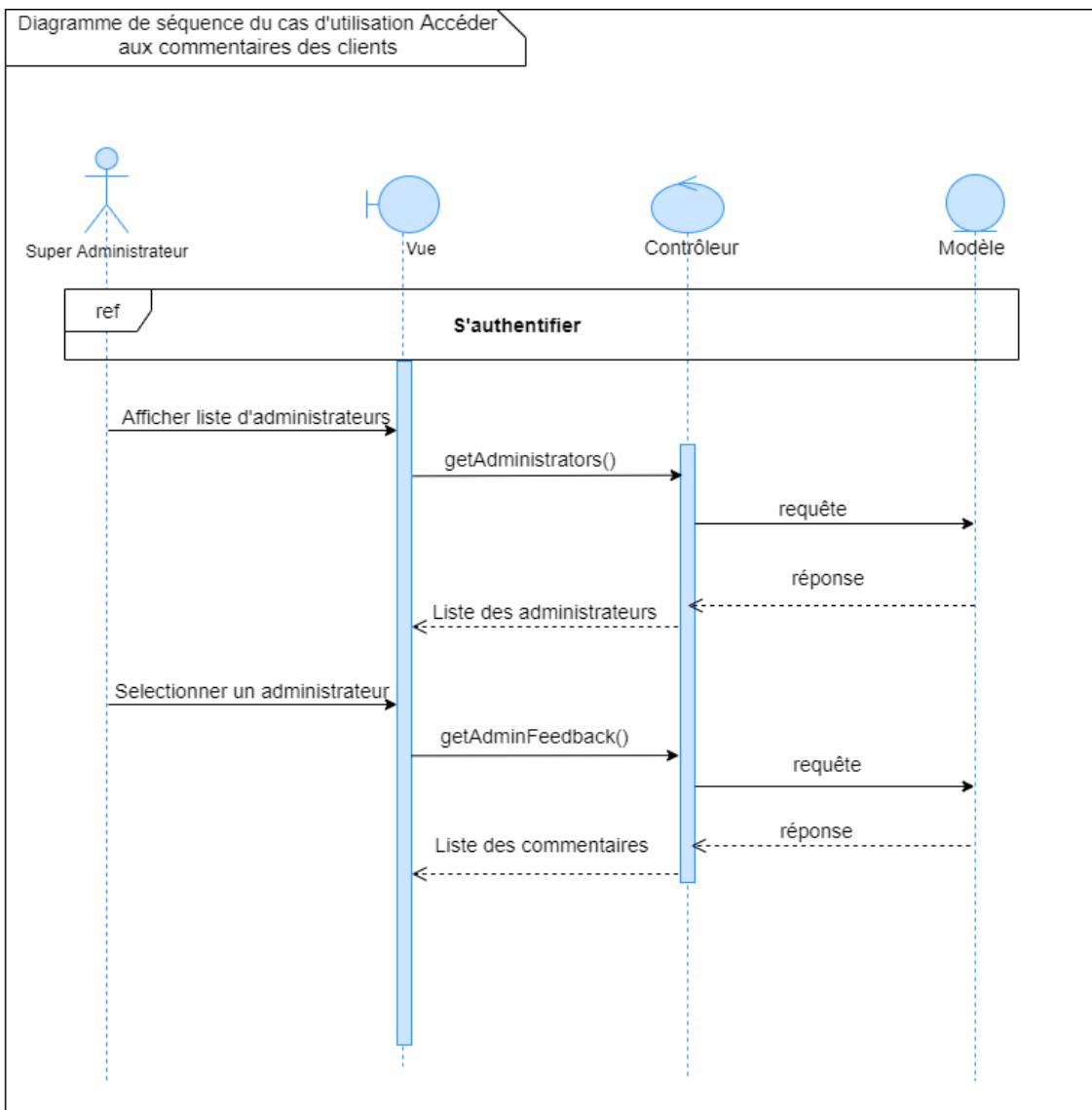


FIGURE 3.31 — Sprint 2 - Diagramme de séquence de cas d'utilisation “Accéder aux commentaires des clients”

- Diagramme de séquence du cas d'utilisation “Afficher les données des admins” :

La figure 3.32 représente le déroulement du processus “Afficher les données des admins”. Après avoir réussi l’authentification, le super administrateur demande la liste des administrateurs disponibles. Ensuite, il sélectionne un administrateur spécifique pour surveiller ses activités.

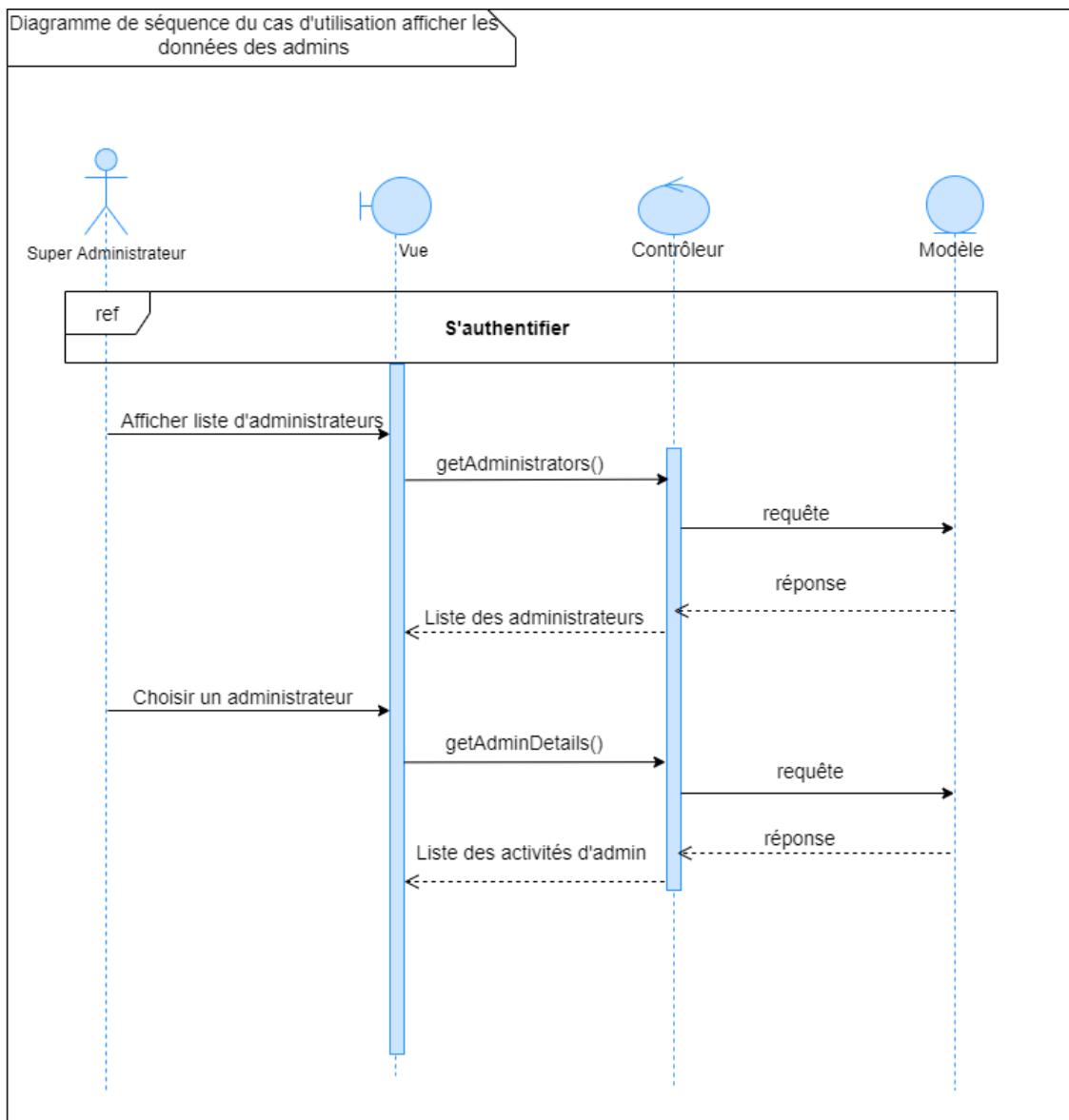


FIGURE 3.32 — Sprint 2 -Diagramme de séquence “Afficher les données des admins”

- Diagramme de séquence du cas d'utilisation “Consulter les demandes des clients” :

La figure 3.33 représente le déroulement du processus “Consulter les demandes des clients”. Après avoir réussi l’authentification, l’administrateur demande la liste des clients en attente de traitement. Ensuite, il sélectionne la première demande de la liste afin de l’examiner et de prendre les mesures appropriées en conséquence.

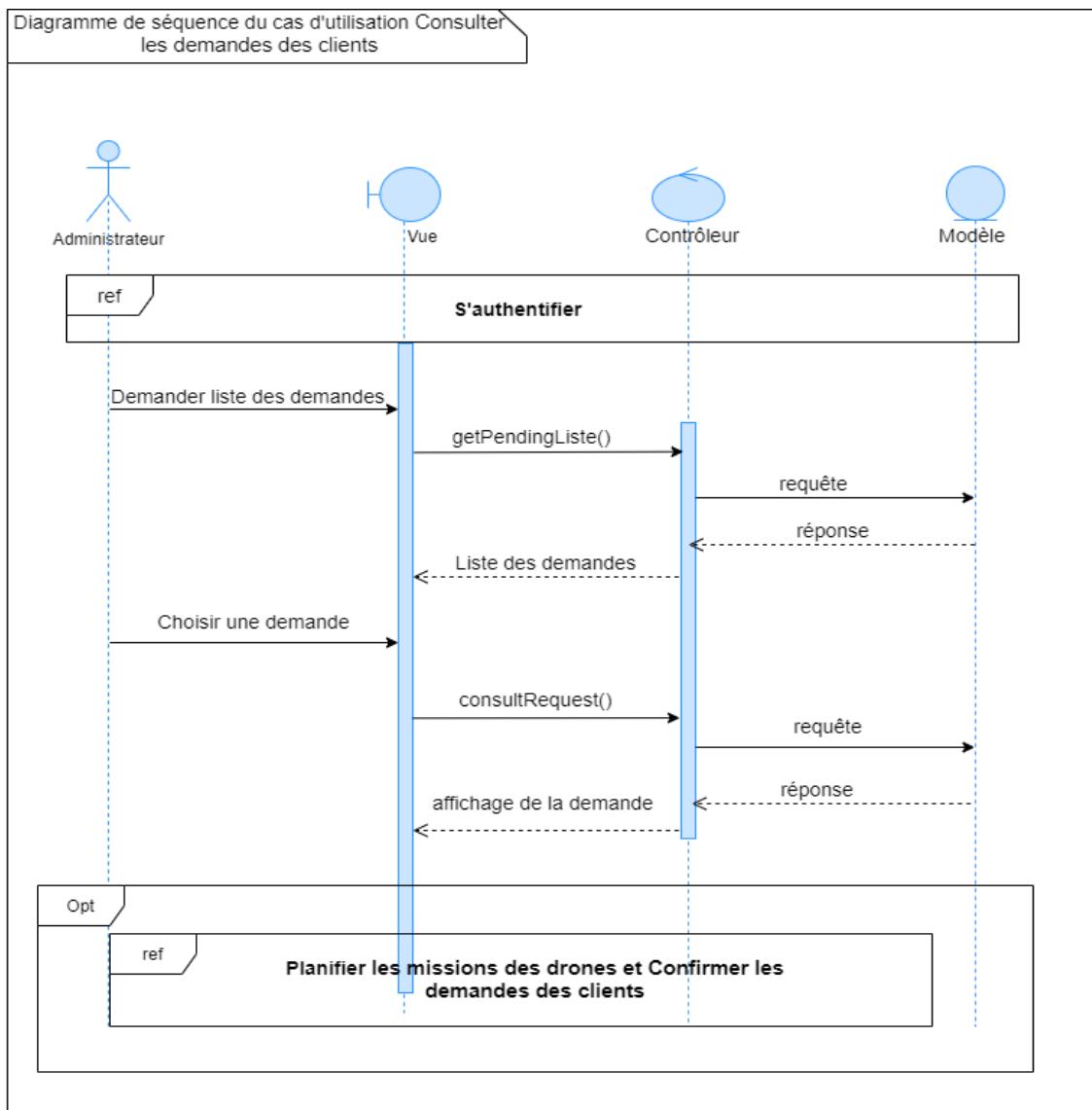


FIGURE 3.33 — Sprint 2 -Diagramme de séquence “Consulter les demandes des clients”

- Diagramme de séquence du cas d'utilisation “Planifier les missions des drones et confirmer les demandes des clients” :

La figure 3.34 représente le déroulement du processus de planifier des missions des drones et la confirmation de demande de client par l’administrateur. L’administrateur traite la demande en appliquant les mesures de contrôle appropriées. Puis, il valide la demande.

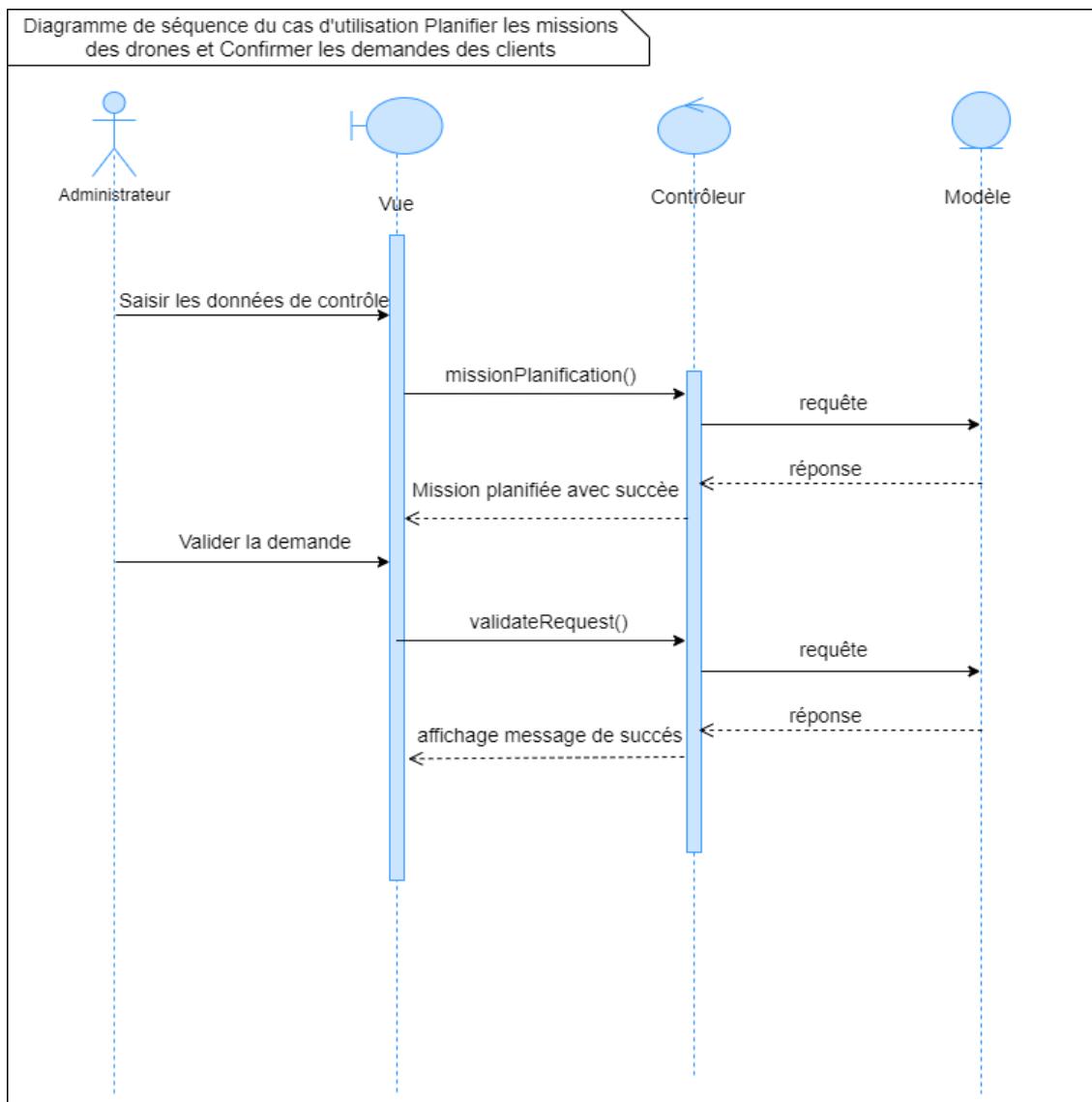


FIGURE 3.34 — Sprint 2 -Diagramme de séquence “Planifier les missions des drones et confirmer les demandes des clients”

- Diagramme de séquence du cas d'utilisation “Suivre l’état des drones” :

La figure 3.35 représente le déroulement du processus de Suivi de l’état des drones par l’administrateur. Après avoir réussi l’authentification, l’administrateur demande la liste des clients actifs. Le système affiche la liste des clients, puis l’administrateur sélectionne un client pour suivre l’état actuel de son drone.

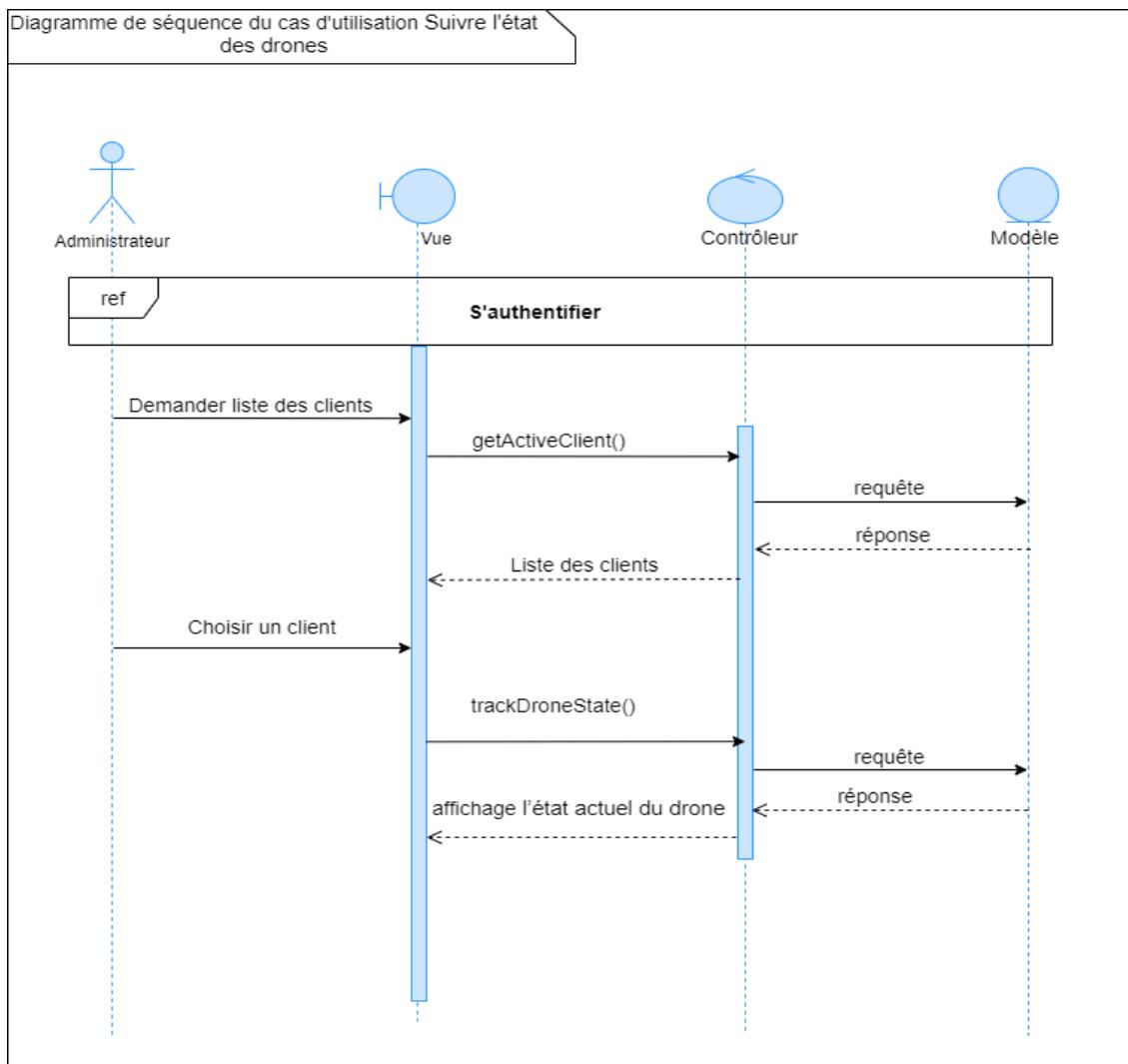


FIGURE 3.35 — Sprint 2 -Diagramme de séquence “Suivre l’état des drones”

- Diagramme de séquence du cas d’utilisation “Sélectionner son espace de surveillance” :

La figure 3.36 représente le déroulement du processus de sélection de l’espace de surveillance par le client. Après avoir réussi l’authentification, le système affiche une carte au client pour qu’il puisse choisir l’espace qu’il souhaite surveiller. Une fois l’espace sélectionné, le client remplit un formulaire avec les informations de l’espace et valide sa sélection. Le système affiche ensuite une demande de choix du mode de paiement au client. Après avoir choisi le mode de paiement approprié, le client le valide. Enfin, le système affiche un message de succès.

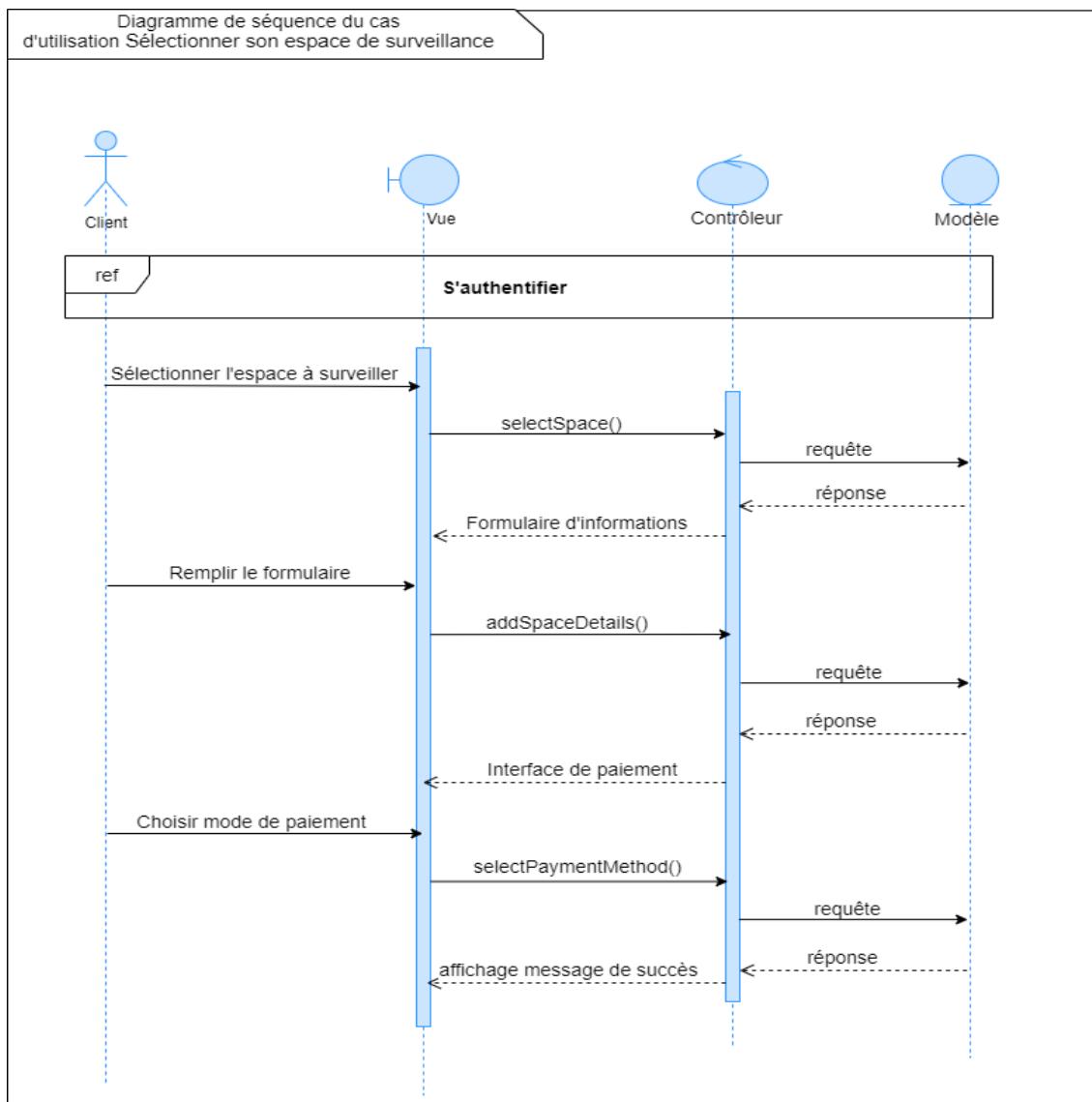


FIGURE 3.36 — Sprint 2 -Diagramme de séquence “Sélectionner son espace de surveillance”

- Diagramme de séquence du cas d'utilisation “Donner des commentaires sur les services qu'il a reçus” :

La figure 3.37 représente le déroulement du processus de soumission de commentaires sur les services reçus par le client. Après avoir réussi l'authentification, le client remplit un formulaire de feedback contenant son commentaire sur le service qu'il a reçu. Une fois le formulaire rempli, le client le valide. Le système enregistre alors le feedback du client et envoie un message de confirmation pour indiquer que l'opération s'est déroulée avec succès.

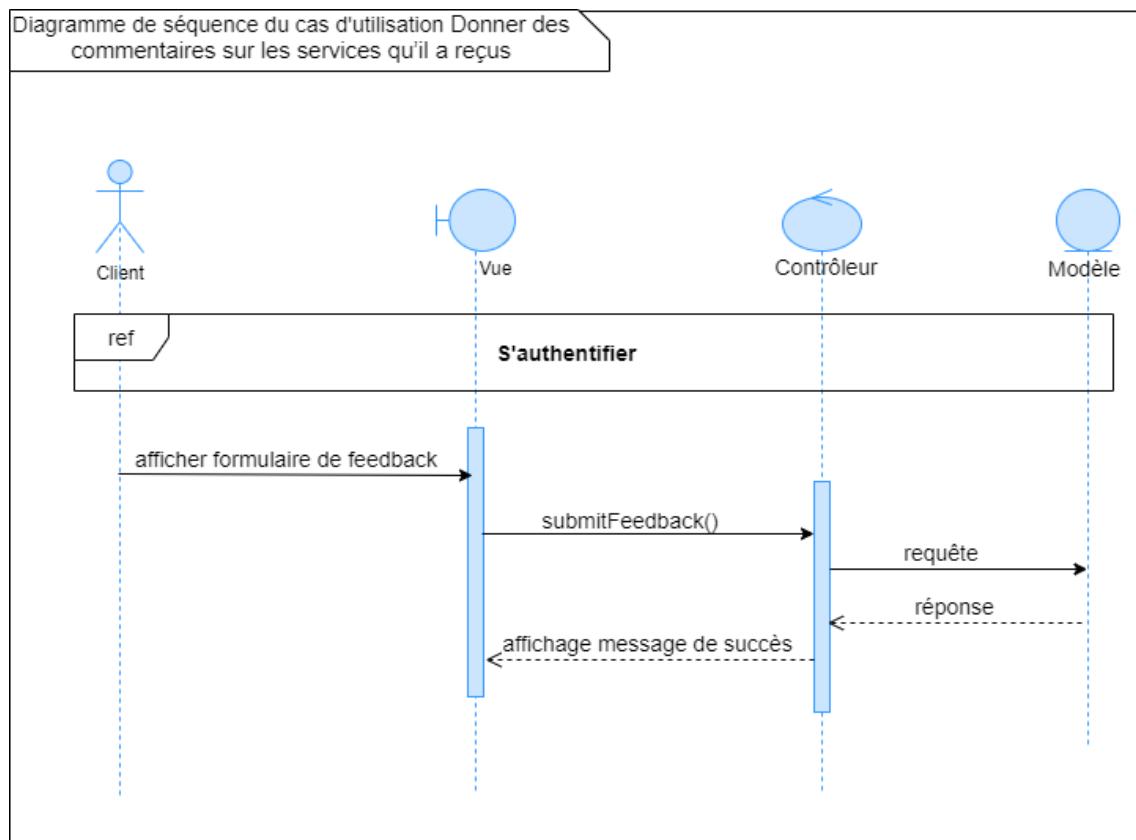


FIGURE 3.37 — Sprint 2 -Diagramme de séquence “Donner des commentaires sur les services qu'il a reçus”

3.3.3.2 Diagramme de Classes

Le diagramme de classes est un outil de modélisation utilisé pour représenter la structure d'un système. Il permet de décrire les entités et leurs relations au sein de notre application.

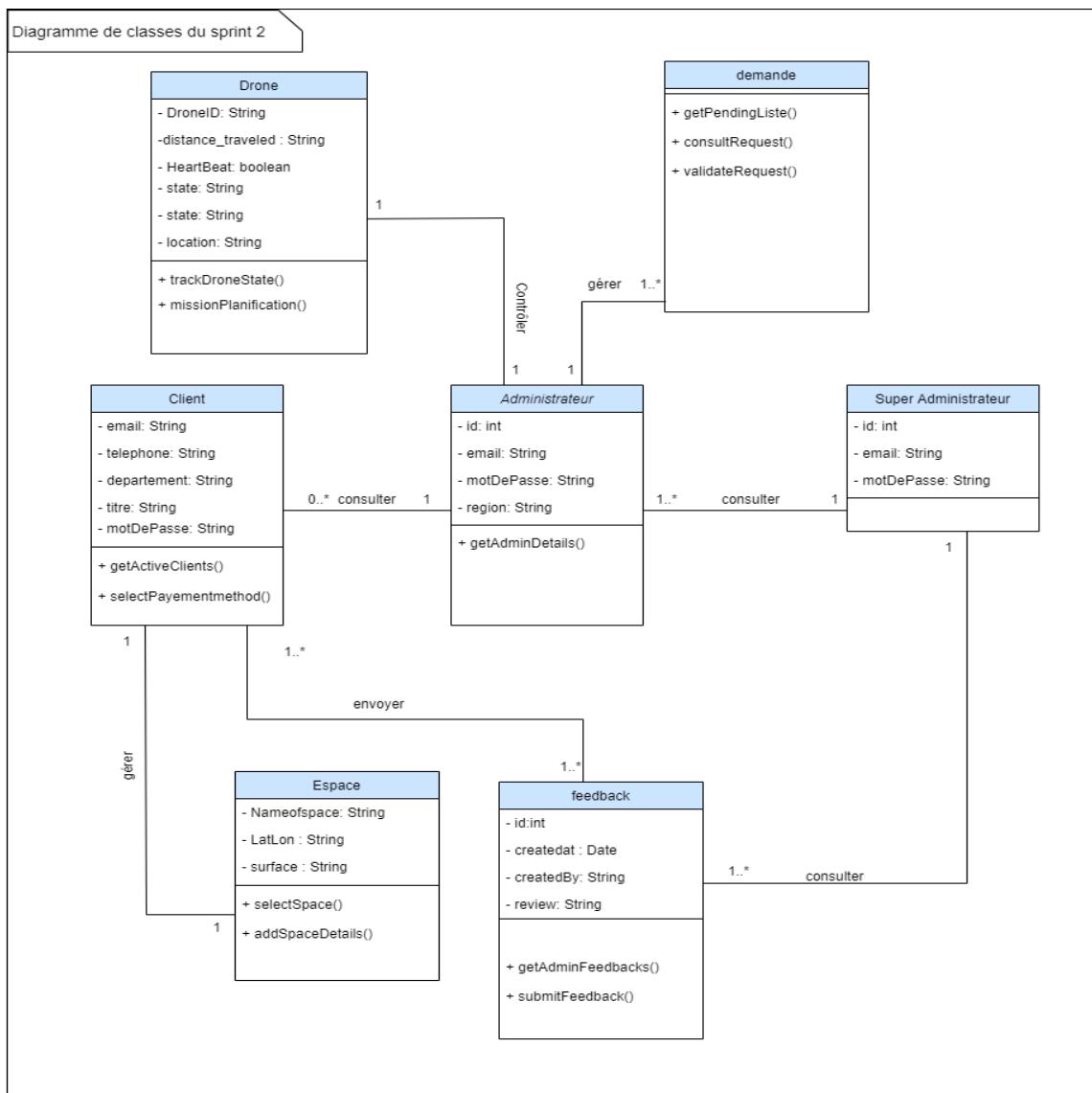


FIGURE 3.38 — Sprint 2 -Diagramme de Classes

3.3.4 Réalisation

Après avoir analysé le sprint 2 et la phase de conception, cette section présente les interfaces qui ont été développées dans le cadre de ce sprint.

- Intégration de l'API Google Maps

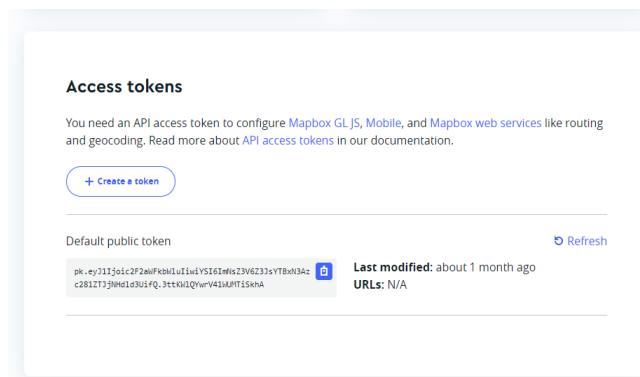


FIGURE 3.39 — Sprint 2 -Initialisation de l'API MapBox

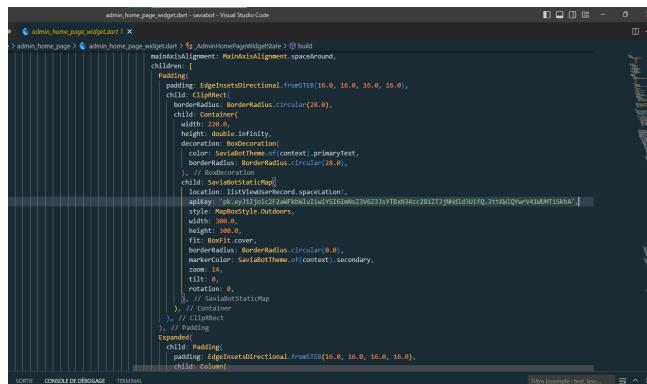


FIGURE 3.40 — Sprint 2 -Intégration de l'API MapBox dans l'application SaviaBot

3.3.5 Validation

Après avoir réalisé les interface, cette section présente les tests des interfaces qui ont été développées dans le cadre de ce sprint.

- Interface Accéder aux commentaires des clients :

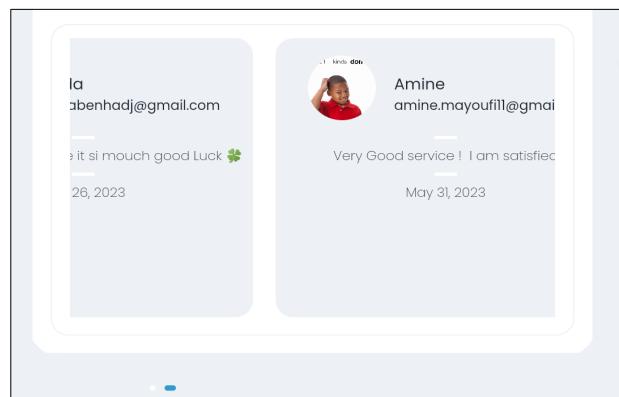


FIGURE 3.41 — Sprint 2 - Interface Accéder aux commentaires des clients

- Interface Afficher les données des admins :

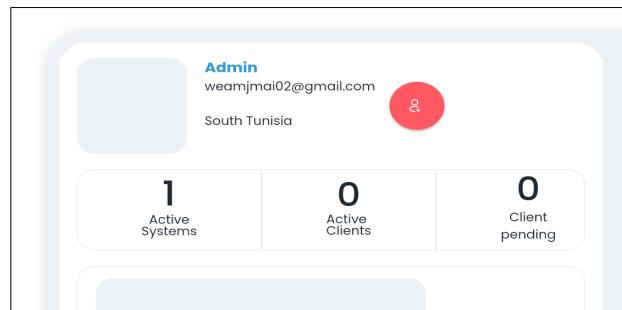


FIGURE 3.42 — Sprint 2 - Interface Afficher les données des admins des admins.

- Interface Consulter les demandes des clients :

Une fois l'Administrateur est authentifié, il est redirigé vers la page AdminHome.

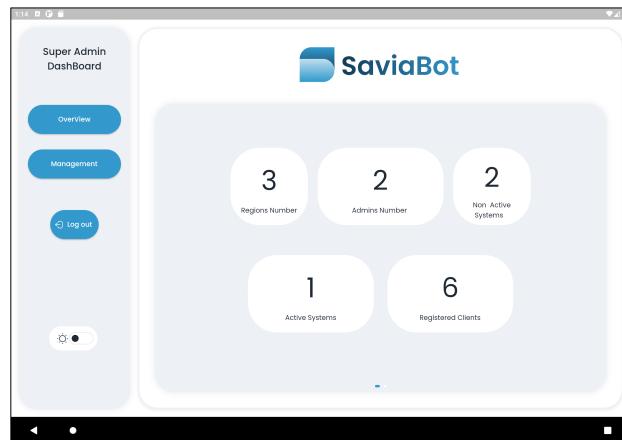


FIGURE 3.43 — Sprint 2 - Interface Dashboard AdminHome

- Interface Confirmer les demandes des clients :



FIGURE 3.44 — Sprint 2 - Interface Pending Clients AdminHome

- Interface Contrôler les missions des drones :

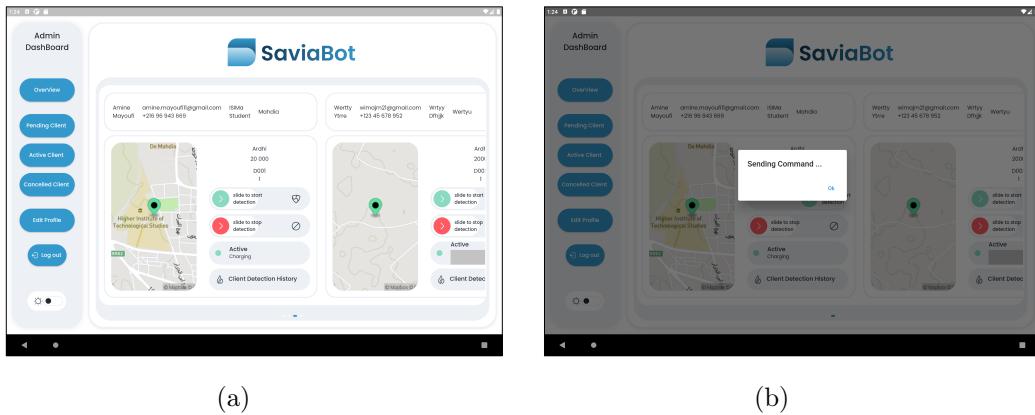


FIGURE 3.45 — Sprint 2 - Interface Supprimer Admin

- Interface Suivre l'état des drones :

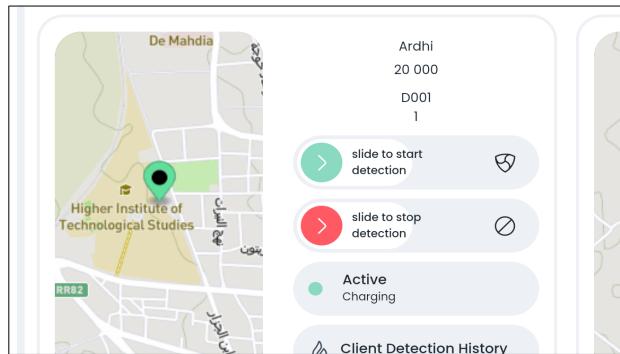


FIGURE 3.46 — Sprint 2 - Section Suivre l'état des drones

- Interface Sélectionner son espace de surveillance :

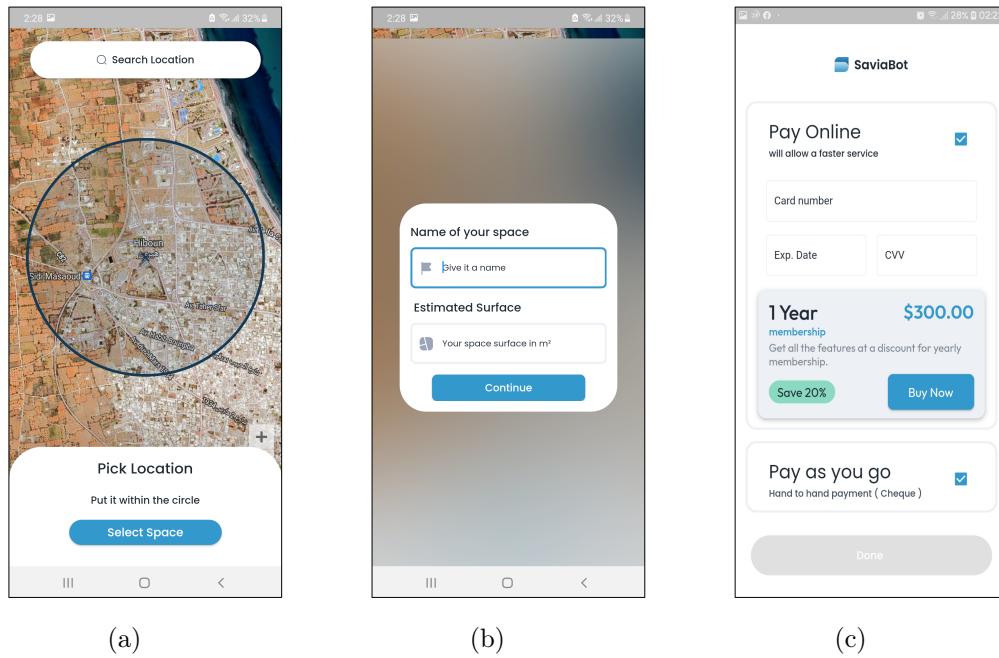


FIGURE 3.47 — Sprint 2 - Interface Sélectionner son espace de surveillance

- Interface Donner des commentaires sur les services qu'il a reçus :

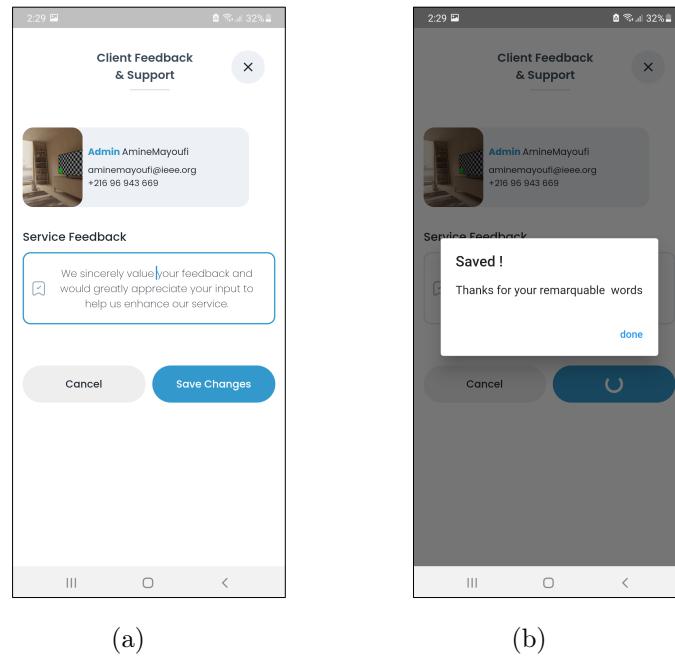


FIGURE 3.48 — Sprint 2 - Interface Donner des commentaires sur les services qu'il a reçus

3.4 Conclusion

Dans cette release, nous avons achevé l'analyse, la conception et la réalisation du premier sprint “Gestion des comptes” ainsi que du deuxième sprint “Planification de missions et avis clients”. Dans le chapitre suivant, nous entamerons la deuxième release de notre projet.

4.1 Introduction

Ce chapitre est consacré à notre première version intégrant le design matériel. Nous commençons par présenter l'organisation de notre projet, puis nous expliquons comment nous avons choisi le matériel adéquat. Enfin, nous décrivons le processus d'assemblage du matériel pour réaliser notre solution.

4.2 SPRINT 3 : “Assemblage et navigation du drone”

Nous introduisons dans cette section l'environnement technique adopté à travers la présentation des outils de développement des technologies utilisées pour la réalisation de notre projet.

4.2.1 Backlog du Sprint 3

Le tableau 4.1 donne un aperçu détaillé sur le Backlog du sprint 3 qui prend en charge la fonctionnalité "Assemblage et navigation du drone".

ID	Sprint	User Story	Jour
3	Assemblage et navigation de drone	3.1 Choix du matériel	6
		3.2 Création du circuit final et assemblage.	12
		3.3 Simulation et navigation	4

TABLE 4.1 — Backlog du sprint 3 : “Assemblage et navigation du drone”

4.2.2 Choix des composants du drone SaviaHawk

Nous abordons dans cette section la sélection du matériel qui a été effectuée le prototype SaviaHawk v0.

4.2.2.1 Microcontrôleur

- Raspberry Pi 4 Modèle B (4Gb)



FIGURE 4.1 — Carte de développement Raspberry Pi 4

La Raspberry Pi 4 Modèle B (4Gb) est un ordinateur monocarte à faible coût, petit et puissant, conçu pour les projets informatiques. Nous avons utilisé la Raspberry Pi 4 Modèle B (4Gb) pour contrôler le drone, récupérer les données du drone vers la station et exécuter l'algorithme de détection.

- **Pixhawk PX4 Flight Controller Autopilot PIX 2.4.8**



FIGURE 4.2 — Carte de contrôle de vol Pixhawk 2.4.8

Le Pixhawk PX4 Flight Controller Autopilot PIX 2.4.8 est une carte de contrôle de vol haute performance pour les drones. Équipé d'un puissant microprocesseur de 168 MHz avec 256 Ko de RAM et d'un coprocesseur de sauvegarde STM32F103 pour une redondance et une fiabilité accrues, le Pixhawk PX4 est le choix idéal pour la construction d'un drone personnalisé avec une Raspberry Pi comme unité de traitement embarquée. Ses capteurs de haute précision, tels que le gyroscope L3GD20 à trois axes et l'accéléromètre LSM303D à trois axes, garantissent des performances de vol optimales.

4.2.2.2 Capteurs

- **AMG 8833 IR Thermal Camera Sensor**

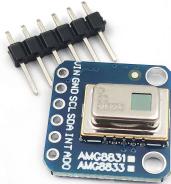


FIGURE 4.3 — Caméra thermique IR AMG 8833

Le capteur de caméra thermique IR AMG 8833 offre une détection précise de la température corporelle à l'aide de 64 capteurs infrarouges intégrés. Ce capteur est idéal pour une utilisation dans les systèmes de contrôle de température et de surveillance de la santé, offrant une résolution d'image élevée et une grande précision dans une plage de température allant jusqu'à 80 degrés Celsius.

- Camera Raspberry Pi



FIGURE 4.4 — Caméra Raspberry Pi OmniVision OV5647

La caméra OV5647 est une caméra de 5 MP conçue pour la Raspberry Pi. Elle capture des images de haute qualité et est utilisée dans diverses applications avec la Raspberry Pi.

- Capteur MQ-2



FIGURE 4.5 — Capteur de gaz inflammables et de fumée MQ2

Le capteur MQ-2 est un capteur de gaz inflammables et de fumée. Il détecte les gaz tels que le méthane, le butane, le propane, l'alcool, l'hydrogène et le monoxyde de carbone. Il est utilisé dans des applications de sécurité et de surveillance pour prévenir les accidents et les incendies.

- capteur MQ-135



FIGURE 4.6 — Capteur de gaz nocifs MQ135

Le capteur MQ135 est utilisé pour détecter la présence de gaz nocifs dans l'air, notamment le dioxyde de carbone, l'azote et le monoxyde de carbone, qui sont généralement libérés lors des incendies de forêt.

- NEO-M8N GPS Module



FIGURE 4.7 — Module GPS NEO-M8N

Le module GPS NEO-M8N est un dispositif de localisation précis et fiable, facile à intégrer dans des systèmes de navigation de véhicules, des drones et d'autres applications nécessitant un positionnement précis. Il offre une précision de positionnement jusqu'à 2,5 mètres et dispose d'une antenne intégrée, d'une faible consommation d'énergie et d'une interface série compatible avec divers microcontrôleurs.

4.2.2.3 Réseaux

- Radio 433 MHz Telemetry



FIGURE 4.8 — Télémétrie radio 433 MHz

La télémétrie radio 433 MHz est un système de communication sans fil fiable et simple à utiliser, utilisé pour transmettre des données sur une longue distance dans les applications de télémétrie, de surveillance et de contrôle à distance. Avec une fréquence de transmission de 433 MHz, il offre une excellente portée de communication avec une faible consommation d'énergie.

4.2.2.4 Actionneur

- ReadytoSky 40A OPTO Brushless ESC



FIGURE 4.9 — Contrôleur de vitesse ESC Brushless OPTO 40A

L'ESC Brushless OPTO 40A de ReadytoSky est un contrôleur de vitesse électrique pour moteurs brushless dans les drones, offrant une capacité de courant de 40A. Il est programmable, compatible avec la plupart des contrôleurs de vol, et dispose d'un système de protection intégré.

- **Moteurs DJI 2212 920KV**



FIGURE 4.10 — Les 4 Moteurs DJI 2212 920KV

Les 4 moteurs DJI 2212 920KV sont des moteurs puissants et fiables pour les drones, offrant une propulsion efficace avec une faible consommation d'énergie.

4.2.2.5 Alimentation et circuit

- **Dupu lipo 4s 20c battery, 14.8 v, 5200 mAh**



FIGURE 4.11 — La batterie Dupu Lipo 4s 20c

La batterie Dupu Lipo 4s 20c est une batterie haute performance pour le drone. Elle a une tension de 14,8 V, une capacité de 5200 mAh, un taux de décharge de 20C et est équipée d'un connecteur XT60 et d'un régulateur de tension. Elle est durable et fiable pour des temps de vol prolongés.

- **APM Power Module 5.3 V DC BEC XT60**



FIGURE 4.12 — Module d'alimentation APM

L'APM Power Module 5.3 V DC BEC XT60 est un module d'alimentation utilisé pour fournir une tension de 5.3 V de manière stable à la carte de contrôle de vol

Pixhawk 2.4.8. Il est équipé d'un connecteur XT60 pour faciliter la connexion à la batterie.

- **Convertisseur De Sortie USB 5V 3a**



FIGURE 4.13 — Convertisseur 5V 3a

Le convertisseur de sortie USB 5V DC 7V-24V à 5V 3A convertit une tension d'entrée de 7V à 24V en une tension de sortie de 5V à un courant maximal de 3A pour alimenter des appareils USB.

4.2.2.6 Accessoire

- **Carcasse Drone F450**



FIGURE 4.14 — Plateforme de drone F450

Le F450 est une plateforme de drone à structure en carcasse, conçue pour offrir une configuration modulaire et polyvalente. Avec sa conception légère et robuste, le DJI F450 permet d'intégrer facilement différents composants tels que les moteurs, les contrôleurs de vol et les systèmes de caméra, offrant ainsi une expérience de vol personnalisée et adaptée aux besoins spécifiques des utilisateurs.

- **Buzzer**



FIGURE 4.15 — Emetteur des signaux sonore PixHawk

Le buzzer émet des signaux sonores pour indiquer l'état de vol et fournir des notifications importantes.

- **1045 Pitch CW/CCW Propeller**



FIGURE 4.16 — Hélices du drone 1045 Pitch CW/CCW

Les 4 hélices 1045 sont des pales efficaces pour la propulsion des drones, avec une poussée stable et une faible consommation d'énergie.

4.2.3 Création du circuit final et assemblage

4.2.3.1 Circuit Générale

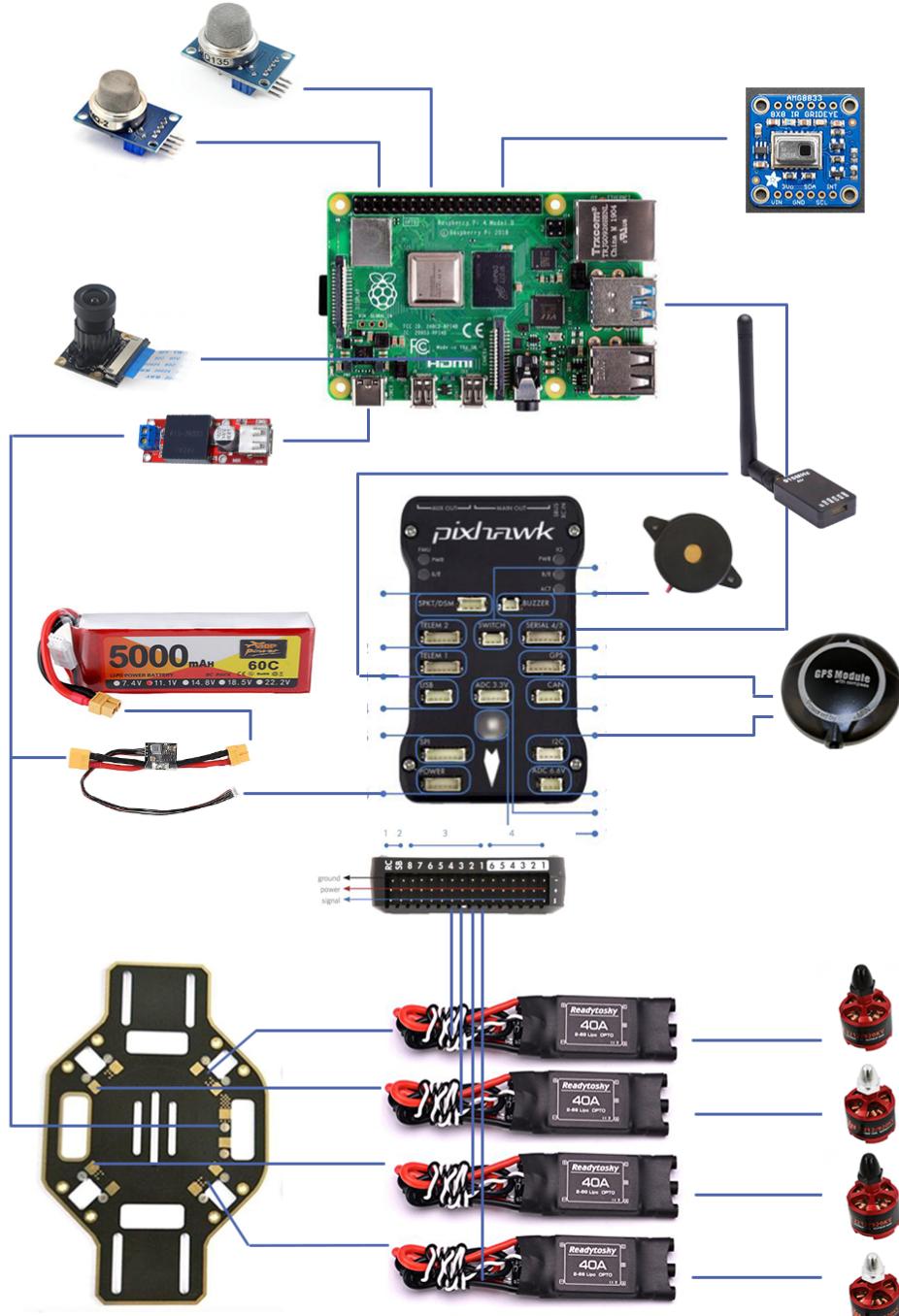


FIGURE 4.17 — Circuit Générale de SaviaHawk

4.2.3.2 Assemblage

- **Phase 1 : Soudage**

Pendant la phase 1, nous réalisons le soudage de quatre ESC (ReadytoSky 40A, OPTO Brushless ESC), du module d'alimentation APM Power Module 5.3V DC BEC XT60 et de deux câbles du convertisseur de sortie USB 5V DC 7V-24V à 5V 3A sur la plaque inférieure (bottom plate) du drone F450. Cette étape permet de préparer l'intégration des composants nécessaires à l'alimentation et au contrôle des moteurs du drone, ainsi qu'à l'alimentation de la Raspberry Pi ou d'autres appareils nécessitant une tension stable de 5V.

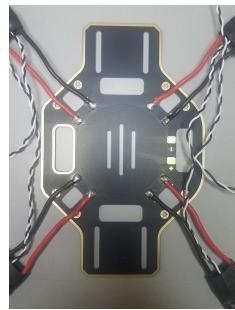


FIGURE 4.18 — Soudage des composentes

- **Phase 2 : Portage des moteurs et ESC sur le F450**

Dans la phase 2, nous procédons au montage et au câblage des quatre moteurs DJI 2212 920KV ainsi que des ESC sur le châssis du drone F450. Nous fixons solidement les moteurs sur les bras du châssis à l'aide de vis appropriées. Ensuite, nous connectons les câbles des moteurs aux ESC correspondants en respectant la correspondance des fils (trois fils pour chaque moteur : alimentation, terre et signal)



FIGURE 4.19 — Portage des moteurs et ESC



FIGURE 4.20 — Portage des moteurs et ESC

- **Phase 3 Modification Qualificatif de F450**

Le support anti-vibration pour Raspberry Pi 4 et Pixhawk réduit les vibrations indésirables pour une meilleure stabilité en vol. Les jambes du drone F450 sont imprimées en 3D dans une conception en forme de pattes d'araignée pour améliorer la stabilité et le centre de gravité. Le support de batterie 4S assure une protection sécurisée pour la battery LiPo. Enfin, la protection des hélices prévient les dommages potentiels aux hélices du drone SaviaHawk.

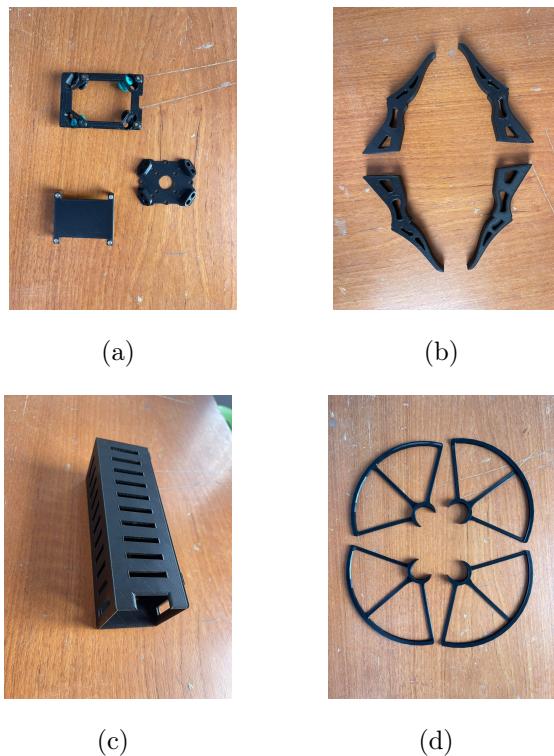


FIGURE 4.21 — Améliorations 3D

Le couvercle SaviaHawk v0 offre une protection sécurisée à l'ensemble du système contre les conditions de vol et autres facteurs environnementaux. Il assure la sécurité des composants internes en les protégeant des chocs, de la poussière et des interférences externes, conçu avec soin et minutie avec Tinkercad.

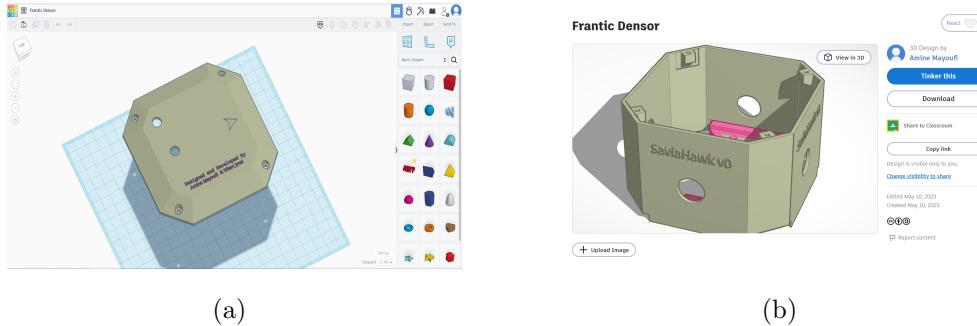


FIGURE 4.22 — Couvercle SaviaHawk

- **Phase 4 : Prototype final :**

Cette étape implique la connexion de tous les composants électroniques, l'organisation et la fixation des câbles, ainsi que la configuration finale du système.



FIGURE 4.23 — Prototype Final SaviaHawk v0

4.2.4 Drone : Simulation et Navigation

4.2.4.1 Approche Mathématique et Analyse des Composants

Estimation de la Capacité de Charge Utile d'un Drone

Un aspect critique de la conception de drones est la capacité de charge utile, qui détermine le poids total qu'un drone peut transporter en plus de son propre poids. Dans cet article, nous examinerons comment estimer la potentialité de charge utile d'un drone en utilisant des expressions mathématiques.

Formule de Calcul de la Capacité de Charge Utile

La capacité de charge utile d'un drone peut être calculée en utilisant la formule suivante :

$$\text{Capacité de charge utile} = \text{Poids de décollage maximal} - \text{Poids à vide}$$

où le poids de décollage maximal est le poids total qu'un drone peut soulever lorsqu'il décolle et le poids à vide est le poids du drone sans aucune charge utile.

Estimation du Poids de Décollage Maximal

Le poids de décollage maximal d'un drone peut être estimé en utilisant la puissance de ses moteurs. Comme règle générale, la poussée combinée de tous les moteurs devrait être deux fois le poids total du drone pour un vol stable. Cette règle est également connue comme le rapport de poussée-poids .

Dans notre cas, nous utilisons quatre moteurs DJI 2212 920KV. Chacun de ces moteurs peut produire environ 860g de poussée avec une batterie LiPo 4S. Par conséquent, la poussée totale est de $4 * 860g = 3440g$. Selon le rapport de poussée-poids de 2 :1, le poids de décollage maximal de notre drone ne doit pas dépasser la moitié de cette valeur, soit $3440g / 2 = 1720g$.

Estimation du Poids à Vide

Le poids à vide du drone est le poids total du drone sans la charge utile. Dans notre cas, cela inclut le poids des moteurs, de la batterie, des hélices, des ESC, de la radio télécommande, du capteur de caméra thermique IR, du contrôleur de vol Pixhawk, de la Raspberry Pi, du cadre du drone F450 et des autres parties imprimées en 3D.

Le poids total de ces composants est d'approximativement 280g (moteurs) + 450g (batterie) + 40g (hélices) + 100g (ESC) + 40g (télécommande) + 4g (capteur) + 32g (Pixhawk) + 46g (Raspberry Pi) + 282g (cadre F450) + 310g (parties imprimées en 3D) = 1584g.

Estimation de la Capacité de Charge Utile

En utilisant les valeurs calculées ci-dessus, nous pouvons estimer la capacité de charge utile de notre drone comme suit : Capacité de charge utile = 1720g (Poids de décollage maximal) - 1584g (Poids à vide) = 136g.

Cela signifie que notre drone devrait être capable de transporter une charge utile supplémentaire de 136g, en plus de son propre poids. Cependant, il convient de noter que ces valeurs sont des estimations basées sur les spécifications des composants et les performances réelles peuvent varier en fonction des conditions réelles de vol et de l'efficacité du moteur.

Estimation de la Durée de Vol

La durée de vol d'un drone est généralement déterminée par la capacité de sa batterie. Dans notre cas, nous utilisons une batterie LiPo 4S de 5200 mAh. Pour calculer le temps de vol estimé, nous devons connaître la consommation d'énergie totale de tous les composants du drone. Supposons qu'elle soit d'environ 25C, c'est-à-dire que le drone consomme 25 fois la potentialité de la batterie en une heure.

Donc, le temps de vol estimé serait capacité de la batterie / consommation d'énergie = $5200\text{mAh} / 25\text{C} = 0.208 \text{ heures} = 12.48 \text{ minutes}$. Ainsi, le prototype SaviaHawk V0 serait capable de voler approximativement 12 minutes en une seule charge de batterie, dans des conditions météorologiques claires.

Estimation de la Zone Couverte

La zone couverte par un drone pendant un seul vol peut être estimée en connaissant la vitesse de vol et la durée de vol. Supposons que la vitesse de vol de notre drone soit de 10 m/s. Alors, la distance totale parcourue durant un vol serait vitesse de vol x durée de vol = $10 \text{ m/s} \times 12.48 \text{ minutes} \times 60 \text{ (pour convertir les minutes en secondes)} = 7488 \text{ m}$.

Maintenant, avec une caméra thermique qui a un champ de vision (FoV) de 60°, la largeur de la zone couverte à une altitude donnée peut être calculée en utilisant la formule de la tangente : Largeur = $2 \times (\text{Tangente}(\text{FoV}/2)) \times \text{Altitude}$. Supposons que le drone vole à une altitude de 100m, alors la largeur de la zone couverte serait $2 \times (\text{Tangente}(60/2)) \times 100\text{m} = 173.2 \text{ m}$.

Par conséquent, la zone totale couverte serait distance parcourue x largeur de la zone couverte = $7488\text{m} \times 173.2\text{m} = 1.3 \text{ km}^2$.

Cela signifie que le prototype SaviaHawk V0 serait capable de couvrir environ 1.3 km² en une seule charge de batterie, dans des conditions météorologiques claires.

Couverture de la Zone Rectangulaire

Lorsqu'un drone effectue une mission de surveillance ou de numérisation, il suit généralement un modèle de vol en grille ou en échiquier pour couvrir l'ensemble de la zone cible. Dans ce modèle, le drone vole en ligne droite sur une longueur, tourne de 180°, se déplace d'une largeur, tourne de 180° et continue ce motif jusqu'à ce que l'ensemble de la zone soit couvert.

Pour notre prototype SaviaHawk V0, à une altitude de vol de 12m, la largeur de la zone couverte par la caméra thermique avec un champ de vision (FoV) de 60° serait de $2 \times (\text{Tangente}(\text{FoV}/2)) \times \text{Altitude} = 2 \times (\text{Tangente}(30)) \times 12\text{m} = 13.86 \text{ m}$.

Donc, si nous devons couvrir une zone rectangulaire de, disons, 1 km de longueur et 500 m de largeur, le drone devrait voler environ $500\text{m} / 13.86\text{m} = 36$ fois le long de la longueur pour couvrir l'ensemble de la zone. C'est-à-dire, le drone aurait à voyager à 36 points différents le long de la largeur pour couvrir toute la zone rectangulaire.

Amélioration de la Couverture avec un Meilleur Capteur ou une Meilleure Caméra

L'efficacité de la couverture d'une zone par un drone peut être améliorée en utilisant un capteur ou une caméra de meilleure qualité. Par exemple, si nous utilisons une caméra avec un champ de vision plus large, disons 90°, la largeur de la zone couverte à une altitude de 12m serait de $2 \times (\text{Tangente}(90/2)) \times 12\text{m} = 27.71 \text{ m}$. Cela signifie que le drone aurait besoin de moins de points de passage pour couvrir la même zone, soit approximativement $500\text{m} / 27.71\text{m} = 18$ points le long de la largeur.

De même, si nous utilisons un capteur capable de détecter des objets à une plus grande distance, le drone pourrait voler à une altitude plus élevée, ce qui augmenterait également la largeur de la zone couverte.

Il est à noter que l'utilisation d'un capteur ou d'une caméra de meilleure qualité peut augmenter la charge utile du drone, ce qui pourrait réduire la durée de vol. Par conséquent, un équilibre doit être trouvé entre la qualité du capteur ou de la caméra et la capacité de charge utile du drone.

4.2.4.2 Calibrage de drone et simulation de Station

Calibrage de PixHawk2.4.8 avec Ggroundcontrol : Flashage de firmware PX4 sur la carte Pixhawk 2.4.8 pour assurer une installation correcte et à jour du micrologiciel.

- Configuration des capteurs :

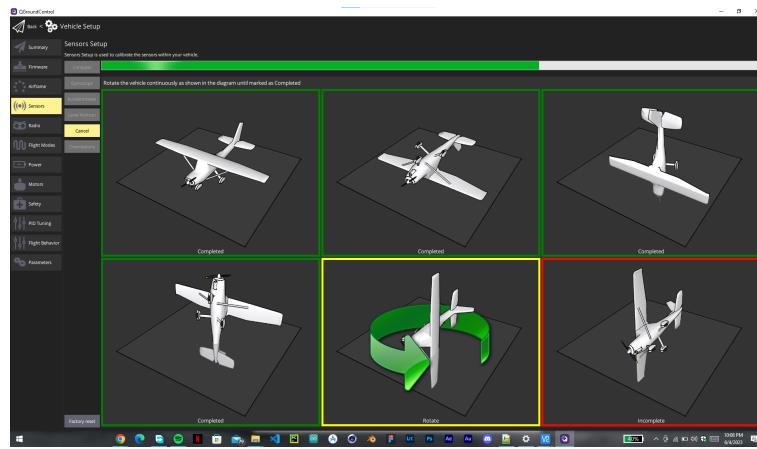


FIGURE 4.24 — Calibrage du Boussole et Accéléromètre

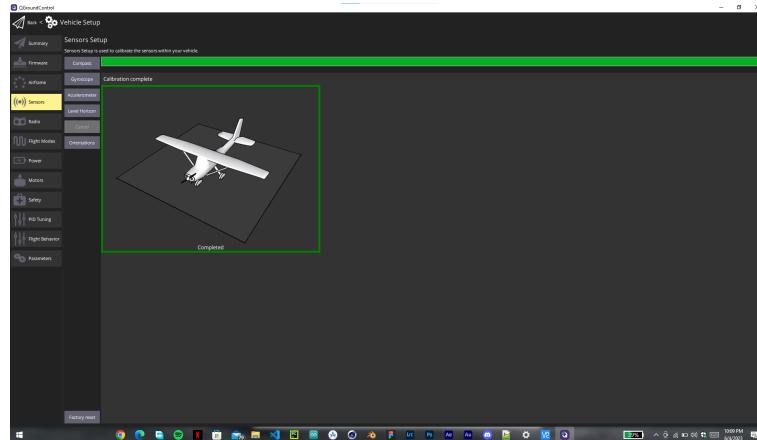


FIGURE 4.25 — Calibrage du Gyroscope et Niveau d'horizon

- Calibration de la batterie et des ESC (Electronic Speed Controllers) selon les instructions du fabricant pour garantir des mesures précises et un fonctionnement correct.
- image

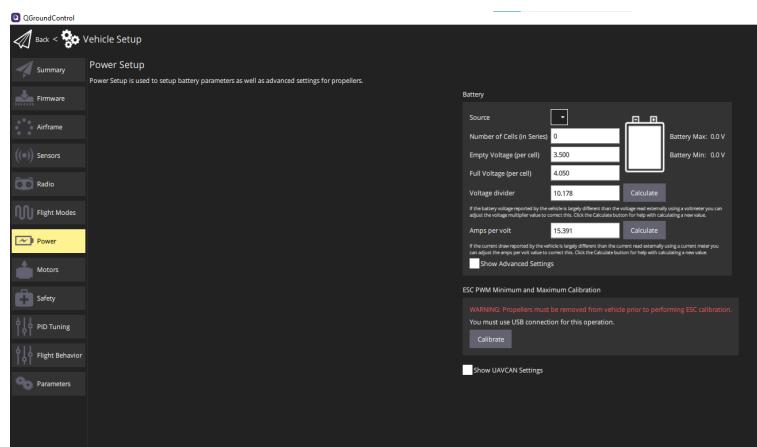


FIGURE 4.26 — Calibration de la batterie et des ESC

- Calibration du contrôleur Turnigy 9x conformément aux instructions spécifiques fournies par le fabricant.



FIGURE 4.27 — Calibration du contrôleur

- Configurer les Modes de vol en correspondant à la configuration du Turnigy 9x

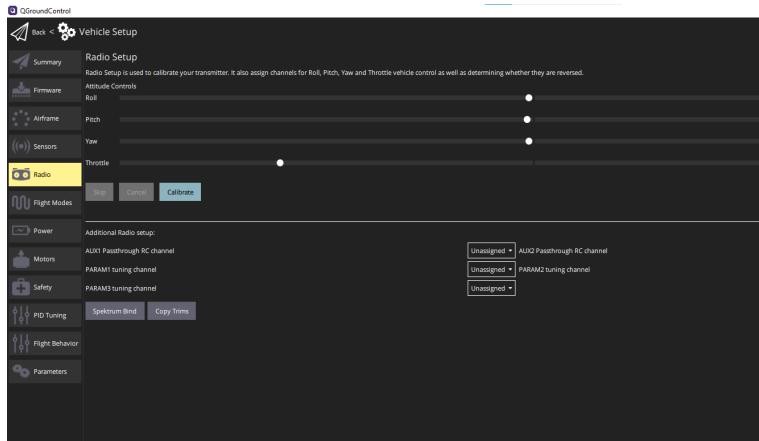


FIGURE 4.28 — Modes de vol

Simulation de station météorologique :

La station météo simulée reçoit des informations météorologiques spécifiques à l'aide de l'API OpenWeatherMap. Comme montre la figure ci-contre :

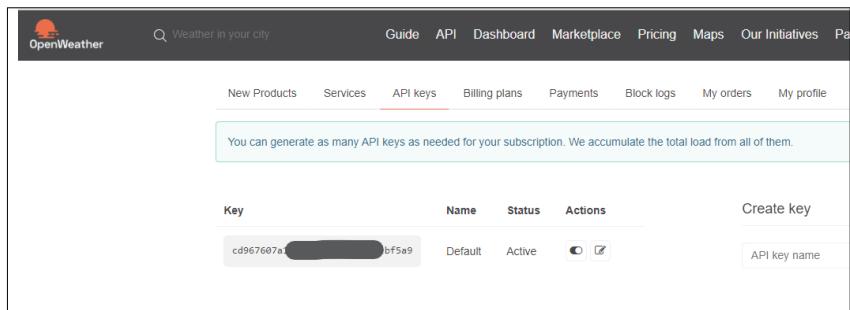


FIGURE 4.29 — API OpenWeatherMap

Les coordonnées géographiques de la station (latitude et longitude) sont utilisées pour interroger l’API et récupérer les données météorologiques appropriées. La température actuelle, la vitesse du vent et une description de l’état du temps dans la région sont des données envoyées à Firestore, comme indiqué dans la figure suivante :

```
api_key = "cd967607e-----j4ebf5a9"
lat = str(station_latitude)
lon = str(station_longitude)
url = f"https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid={api_key}"
response = requests.get(url)
weather_data = response.json()

# Convert temperature to Celsius
temp_in_celsius = int(weather_data['main']['temp']) - 273.15

# Extract wind speed
wind_speed = weather_data['wind']['speed']

# Extract weather description
weather_description = weather_data['weather'][0]['description']
```

FIGURE 4.30 — Station météorologique Simulée

Ces données sont enregistrées, en temps réel, dans Firestore sous un ID de station spécifique afin qu’elles puissent être utilisées dans l’application SaviaBot.

4.2.4.3 Communication Raspberry/PixHawk et Navigation

Bibliothèque MAVSDK-Python

Pour la communication entre la Raspberry Pi et le PixHawk, le script mavproxy.py est utilisé. Ce script utilise une bibliothèque qui permet de contrôler le drone et de lui donner des missions à partir de la Raspberry Pi. Il agit comme une interface de communication entre les deux appareils, permettant à la Raspberry Pi de transmettre des commandes au PixHawk.

Pour faciliter la communication et le contrôle du drone depuis la Raspberry Pi, la bibliothèque mavsdk est utilisée. Cette bibliothèque fournit une API qui permet de communiquer avec le drone et d’exécuter différentes tâches, telles que l’envoi de commandes de vol, la planification de missions, la récupération de données de capteurs, etc.

Déplacer vers des positions relatives

Le script (Python) commence par initialiser la connexion avec le drone et configurer certains paramètres. Ensuite, il effectue une série de mouvements en utilisant les coordonnées de position relatives (NED) et la fonction set-position-ned. Ces mouvements comprennent un décollage, un déplacement vers différentes positions relatives et un atterrissage. Pendant l'exécution, le script affiche également l'altitude actuelle du drone et le mode de vol.

Contrôle hors-bord en utilisant les coordonnées de vitesse

Ce script utilise la fonction set-velocity-body pour définir la vitesse du drone en utilisant les coordonnées de vitesse du corps (body coordinates). Le script effectue une série de mouvements en faisant tourner le drone dans le sens des aiguilles d'une montre et en montant, puis en effectuant d'autres mouvements. Il affiche de plus l'altitude actuelle du drone pendant l'exécution.

Enfin, après plusieurs tentatives, nous avons réussi à effectuer un vol réussi avec un vol préconfiguré utilisant Raspberry Pi 4, comme le montre la figure ci-dessous.



FIGURE 4.31 — L'armement du drone SaviaHawk

4.3 SPRINT 4 : “Détection et alertes précoce des incendies”

4.3.1 Backlog du sprint 4

Le tableau 4.2 donne un aperçu détaillé sur le Backlog sprint qui prend en charge la fonctionnalité "Détection et alertes précoce des incendies".

ID	Sprint	User Story	Jour
4	Détection et alertes précoce des incendies	4.1 Intégration des capteurs de détection d’incendie	3
		4.2 Développement des algorithmes de détection	4
		4.3 Réalisation de la fonctionnalité “Surveiller en temps réel l’état des cultures et les zones vertes ”	2
		4.4 Réalisation de la fonctionnalité “Recevoir les alertes ”	3
		4.5 Enregistrer les données de détection d’incendie pour analyse ultérieure	4

TABLE 4.2 — Backlog du sprint 4 : “Détection et alertes précoce des incendies”

4.3.2 Intégration des capteurs de détection d’incendie

4.3.2.1 Capteurs de substances gazeuses et de qualité de l’air (MQ135 et MQ2)

- Installation des capteurs MQ135 et MQ2 sur le circuit et le câblage avec Raspberry Pi 4.

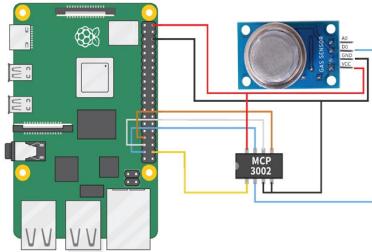


FIGURE 4.32 — Installation des capteurs MQ135

4.3.2.2 Caméra thermique AMG8833

- Connexion de la caméra thermique AMG8833 à la Raspberry Pi 4 via le circuit et le câblage nécessaires

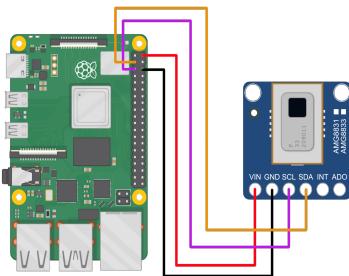


FIGURE 4.33 — Câblage I₂C caméra thermique AMG8833

- Réalisation de tests du capteur de la caméra thermique en utilisant la bibliothèque Adafruit.

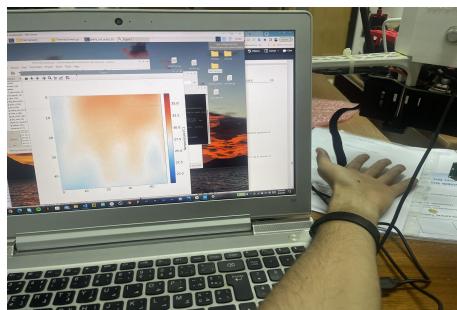


FIGURE 4.34 — tests du capteur AMG 8833

4.3.2.3 Caméra Raspberry 5MP

- Installation de la caméra Raspberry 5MP sur le système.

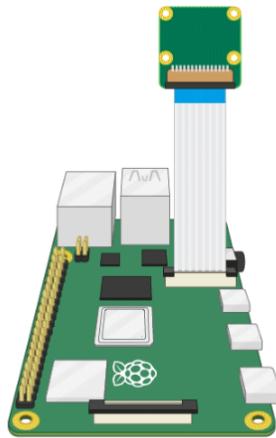


FIGURE 4.35 — Installation de Caméra Raspberry

- Réalisation de tests du capteur de la caméra en utilisant la bibliothèque CV2 d'OpenCV.

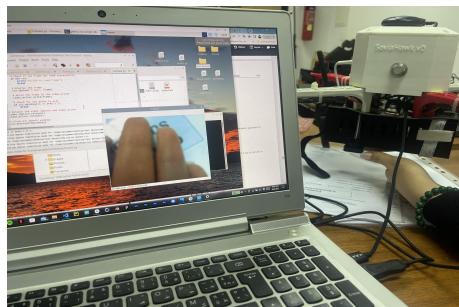


FIGURE 4.36 — tests du capteur AMG 8833

4.3.2.4 Fusion des capteurs

La fusion des capteurs est un processus qui combine les informations provenant de différents capteurs pour obtenir une vision plus complète et précise d'une situation donnée. Dans le contexte de la détection d'incendie, cela impliquerait de combiner les données des capteurs de substances gazeuses, de qualité de l'air, de caméra thermique et de caméra Raspberry 5 MP pour obtenir une évaluation globale de la présence et de l'intensité d'un incendie. Ce processus de fusion permet d'améliorer la fiabilité et l'efficacité du système de détection.

4.3.3 Développement des algorithmes de détection

Dans cette section, nous décrirons le développement des algorithmes de détection pour la détection des incendies à l'aide de l'intelligence artificielle et de caméras de drones.

4.3.3.1 Dataset

- Collecte :

Pour garantir que nos modèles peuvent gérer différents types d'incendies en forêt (incendie au sol, incendie des troncs et incendie de la canopée), nous avons collecté des images à partir de plusieurs ensembles de données publics sur les incendies. Parmi ces ensembles de données, nous avons utilisé BowFire [1], ForestryImages [4], ieee-dataport [14] , et d'autres sources similaires. Ces ensembles de données contiennent des images capturées dans divers scénarios d'incendies.

- Nettoyage des ensembles de données :

Après la collecte, nous avons procédé à une étape de nettoyage rigoureuse pour garantir la qualité des données utilisées pour l'entraînement de notre algorithme. Nous avons effectué une filtration manuelle afin de supprimer les doublons et d'éliminer les données corrompues ou inappropriées. De plus, nous avons normalisé les annotations et les métadonnées associées aux images pour assurer une cohérence et une compréhension optimale.

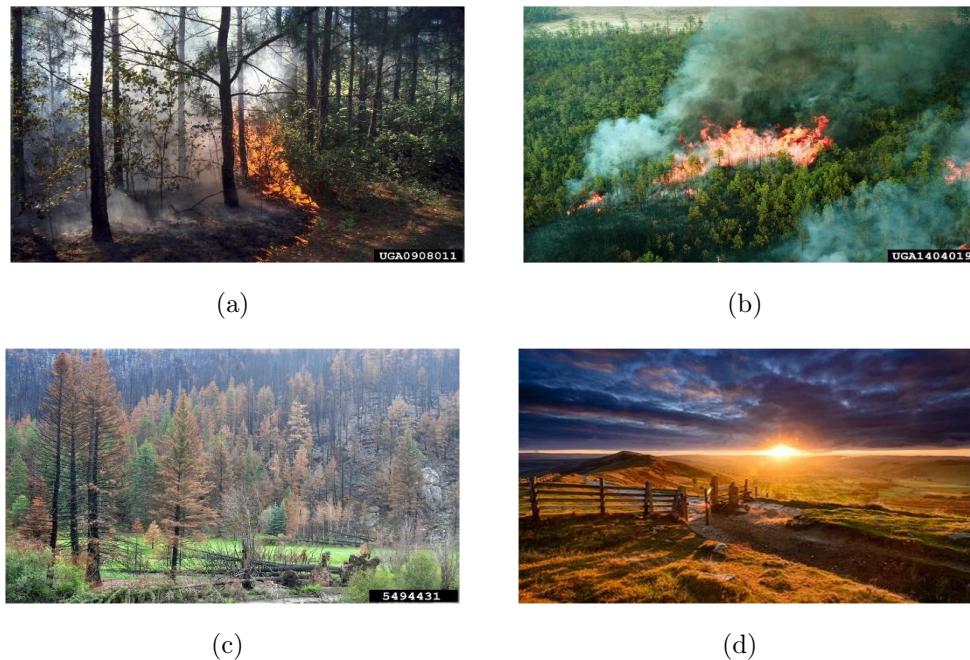
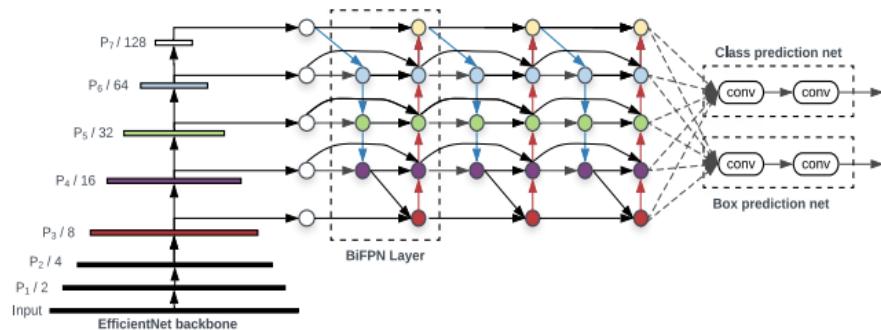
- Création d'un ensemble de données intégré (dataset) :

Suite au processus de nettoyage, nous avons créé un ensemble de données intégré sur les incendies en forêt. Cet ensemble de données comprend un total de 952 images, comprenant 423 images d'incendies en forêt [a] [b] et 529 images sans incendie [c] [d]. Les échantillons représentatifs de notre ensemble de données sont présentés dans les Figures.

4.3.3.2 YOLOv5

- Architecture du réseau (Réseau de neurones convolutifs) :

Pour notre algorithme de détection, nous avons choisi d'utiliser YOLOv5, qui est basé sur une architecture de réseau de neurones convolutifs (CNN). Le CNN est un type de modèle d'apprentissage profond spécifiquement conçu pour traiter des données d'images. Il est capable d'extraire automatiquement des caractéristiques significatives des images, ce qui est essentiel pour la détection des incendies.

**FIGURE 4.37** — Exemple de Dataset**FIGURE 4.38** — Réseau de neurones convolutifs

- Entraînement avec l'ensemble de données collecté :

Nous avons ensuite procédé à l'entraînement de notre modèle YOLOv5 en utilisant l'ensemble de données collecté et nettoyé. L'entraînement a été effectué sur un ordinateur portable équipé d'un GPU GTX950m, d'un processeur i7 6700HQ et de 16 Go de RAM, exécutant Ubuntu 22.04 64 bits avec l'accélération CUDA. Le temps total d'entraînement a été de 12 heures, 35 minutes et 10 secondes.

- Évaluation des performances :

Nous avons évalué les performances de notre modèle en suivant l'évolution de la perte (loss) et de la précision (accuracy) pendant l'entraînement. Des graphiques ont été générés pour illustrer ces métriques, tant pour l'ensemble de données d'entraînement que pour

l'ensemble de données de validation.

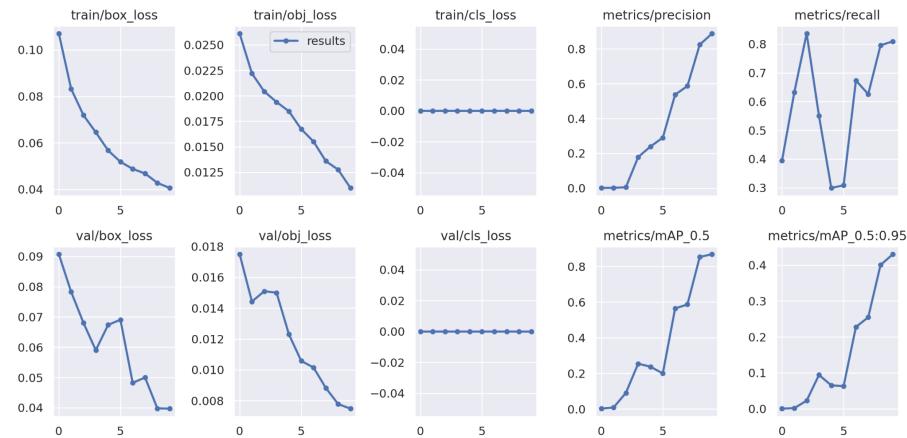


FIGURE 4.39 — Graphes d'évolution, de perte (loss) et de précision (accuracy)

- Tests et exécution avec les poids optimisés :

Une fois l'entraînement terminé, nous avons utilisé les poids optimisés obtenus pour tester et exécuter notre algorithme de détection. Les sorties générées par le modèle ont été évaluées et comparées aux annotations de référence pour évaluer la précision et les performances de détection des incendies.



FIGURE 4.40 — Tests et exécution Detect.py

4.3.4 Réalisation de la fonctionnalité “Surveiller en temps réel l’état des cultures et les zones vertes ” et ” Recevoir les alertes ”

4.3.4.1 Analyse

Dans cette partie, nous présentons le diagramme de cas d'utilisation, les descriptions textuelles pour chaque cas, ainsi que les diagrammes de séquence.

- Diagramme de cas d'utilisation :

La figure représente le diagramme de cas d'utilisation pour le sprint 4.

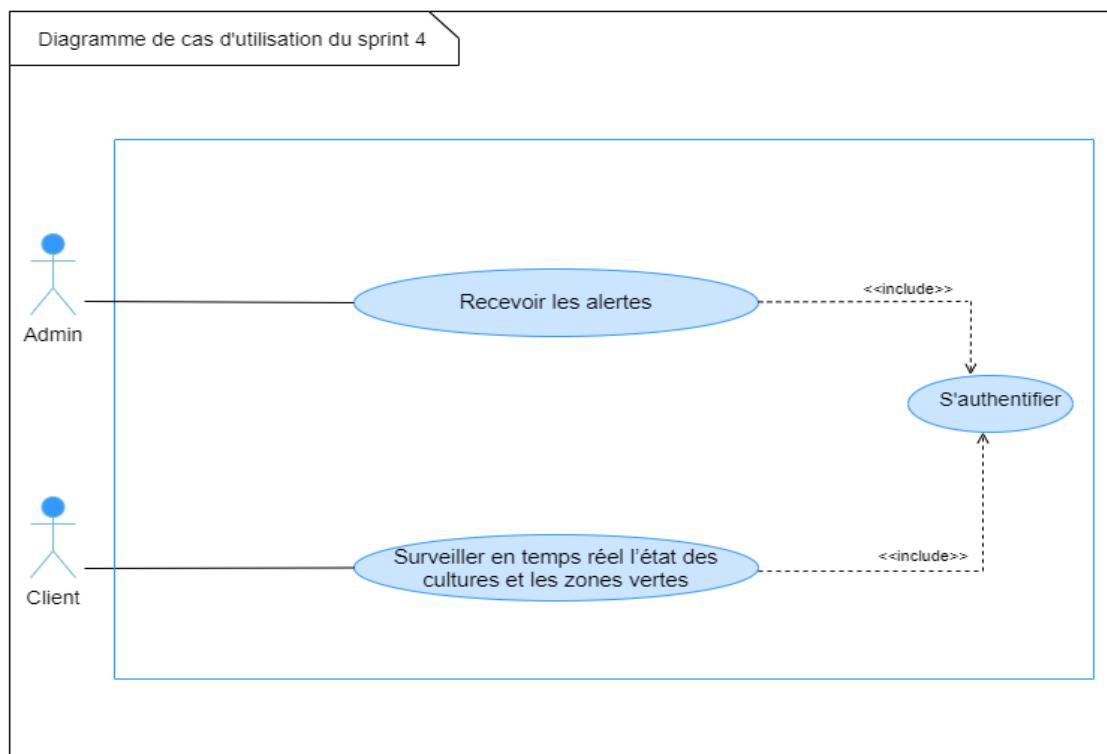


FIGURE 4.41 — Sprint 4-Diagramme de cas d'utilisation

- Description textuelle des cas d'utilisation :

- Description textuelle de cas d'utilisation « Surveiller en temps réel l'état des cultures et les zones vertes»

Nom du CU	Surveiller en temps réel l'état des cultures et les zones vertes
Acteurs	Client
Résumé	Le client surveille en temps réel l'état des cultures et les zones vertes.
Pré-conditions	Le client est authentifié.
Scénario Nominal	<p>1-Le client accède à l'écran principal "home screen" du système.</p> <p>2-Le système affiche les informations de l'état du drone, telles que la batterie, les vents et la météo.</p> <p>3-Le système présente une carte qui permet au client de visualiser sa zone verte.</p> <p>4-Le client observe en temps réel l'état des cultures dans sa zone à l'aide de la carte.</p> <p>5-Le système met à jour les informations en temps réel, fournissant au client les données les plus récentes sur l'état des cultures.</p>
Post-conditions	Le client a pu surveiller en temps réel l'état des cultures et les zones vertes.

TABLE 4.3 — Description textuelle du scénario de cas d'utilisation "Surveiller en temps réel l'état des cultures et les zones vertes"

- Description textuelle de cas d'utilisation «Recevoir les alertes»

Nom du CU	Recevoir les alertes
Acteurs	Administrateur
Résumé	L'administrateur reçoit les alertes du système, y compris les alertes de système et les alertes de détection d'incendie.
Pré-conditions	L'administrateur est authentifié.
Scénario Nominal	<p>1-Le système surveille en continu les différents paramètres et détecte les événements importants, tels que les problèmes techniques ou les détections d'incendie.</p> <p>2-Lorsqu'une alerte est générée, le système envoie une notification à l'administrateur.</p> <p>3-L'administrateur reçoit les alertes.</p> <p>4-L'administrateur examine les alertes pour évaluer leur gravité et prendre les mesures nécessaires en conséquence.</p>
Post-conditions	L'administrateur a reçu les alertes.

TABLE 4.4 — Description textuelle du scénario de cas d'utilisation "Recevoir les alertes"

4.3.4.2 Conception

Dans cette section, nous présentons les diagrammes de séquences et de classes utilisés dans le cadre du sprint. Les diagrammes de séquence illustrent la séquence des messages échangés entre les objets nécessaires pour réaliser les fonctionnalités du scénario, tandis que les diagrammes de classes décrivent la structure du système.

- Diagramme de séquence :

- Diagramme de séquence de cas d'utilisation “Surveiller en temps réel l'état des cultures et les zones vertes” :

La figure représente le déroulement du cas d'utilisation “Surveiller en temps réel l'état des cultures et les zones vertes”. Une fois authentifié et connecté à son compte, le client peut surveiller en temps réel l'état de l'environnement et des zones vertes. Le client demande spécifiquement d'afficher l'état de sa zone verte. Le système répond en affichant en temps réel la zone verte du client sur la carte. Cette fonctionnalité permet au client de suivre de près l'état de sa zone verte.

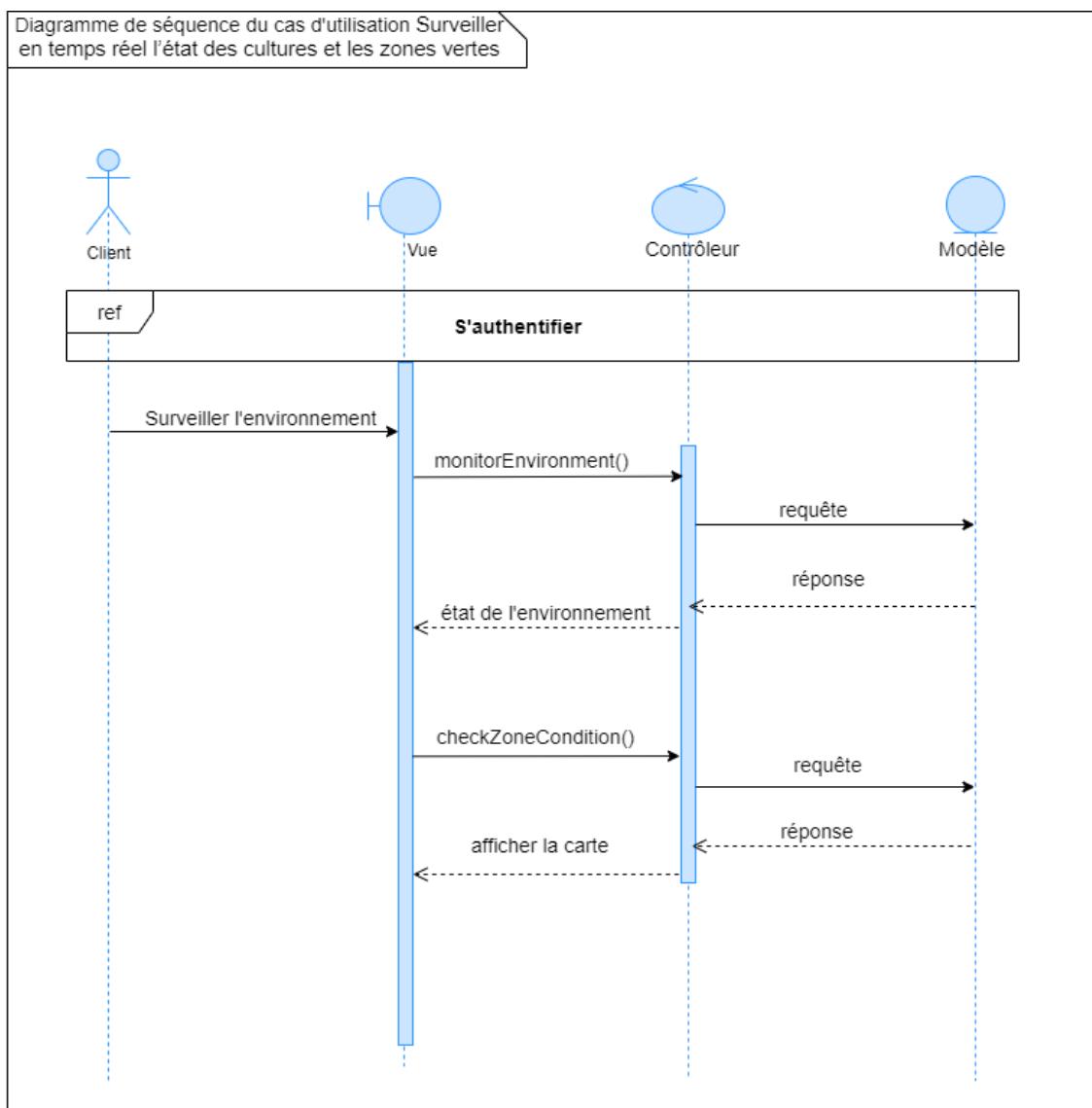


FIGURE 4.42 — Sprint 4 -Diagramme de séquence “Surveiller en temps réel l'état des cultures et les zones vertes”

- Diagramme de séquence de cas d'utilisation “Recevoir les alertes” :

La figure représente le déroulement du processus "Recevoir les alertes". Après avoir réussi l'authentification, l'administrateur peut recevoir en temps réel des alertes concernant les problèmes techniques ou système du drone.

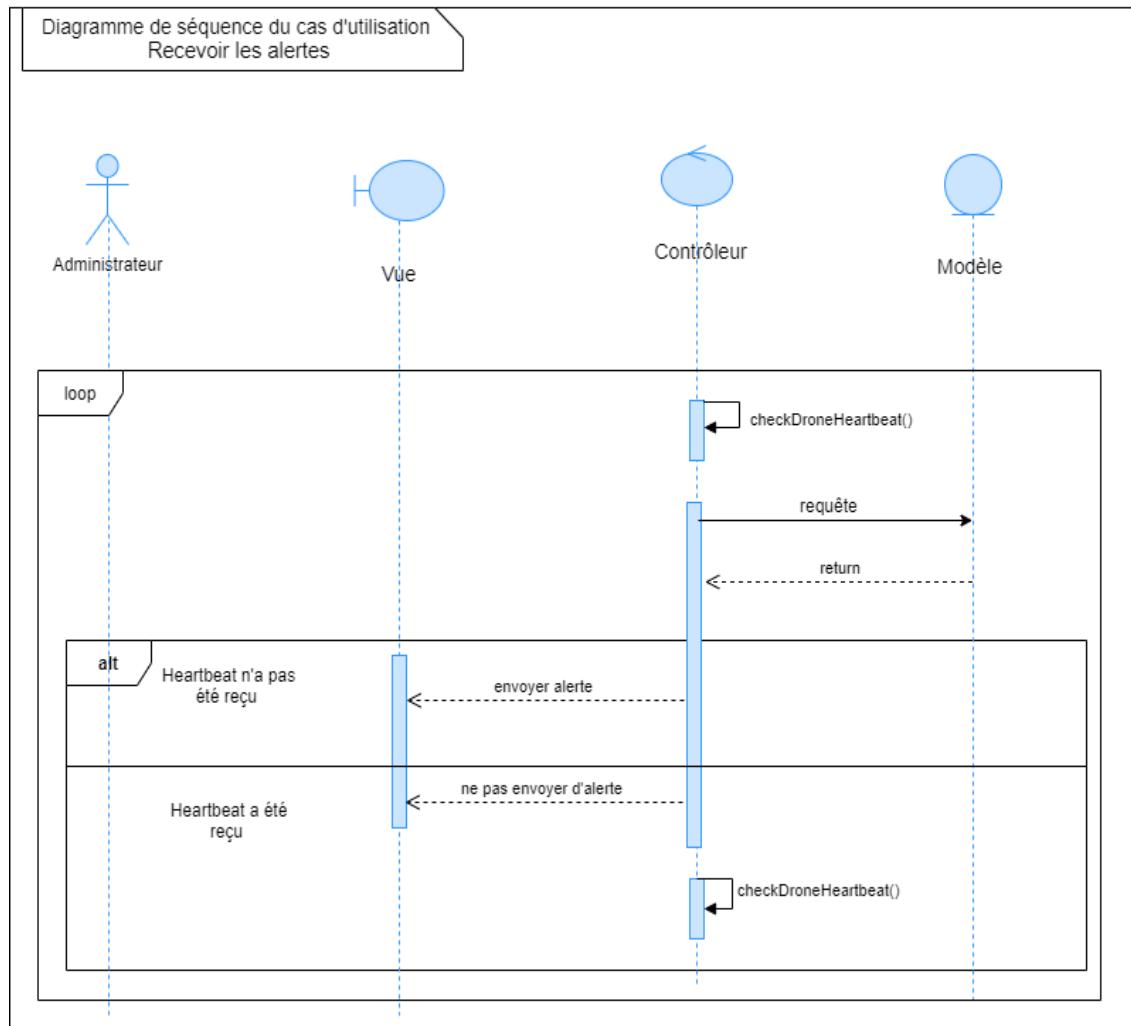


FIGURE 4.43 — Sprint 4 -Diagramme de séquence “recevoir les alertes”

- **Diagramme de classe :**

Dans cette sous-section, nous présenterons le diagramme de classe du sprint 4.

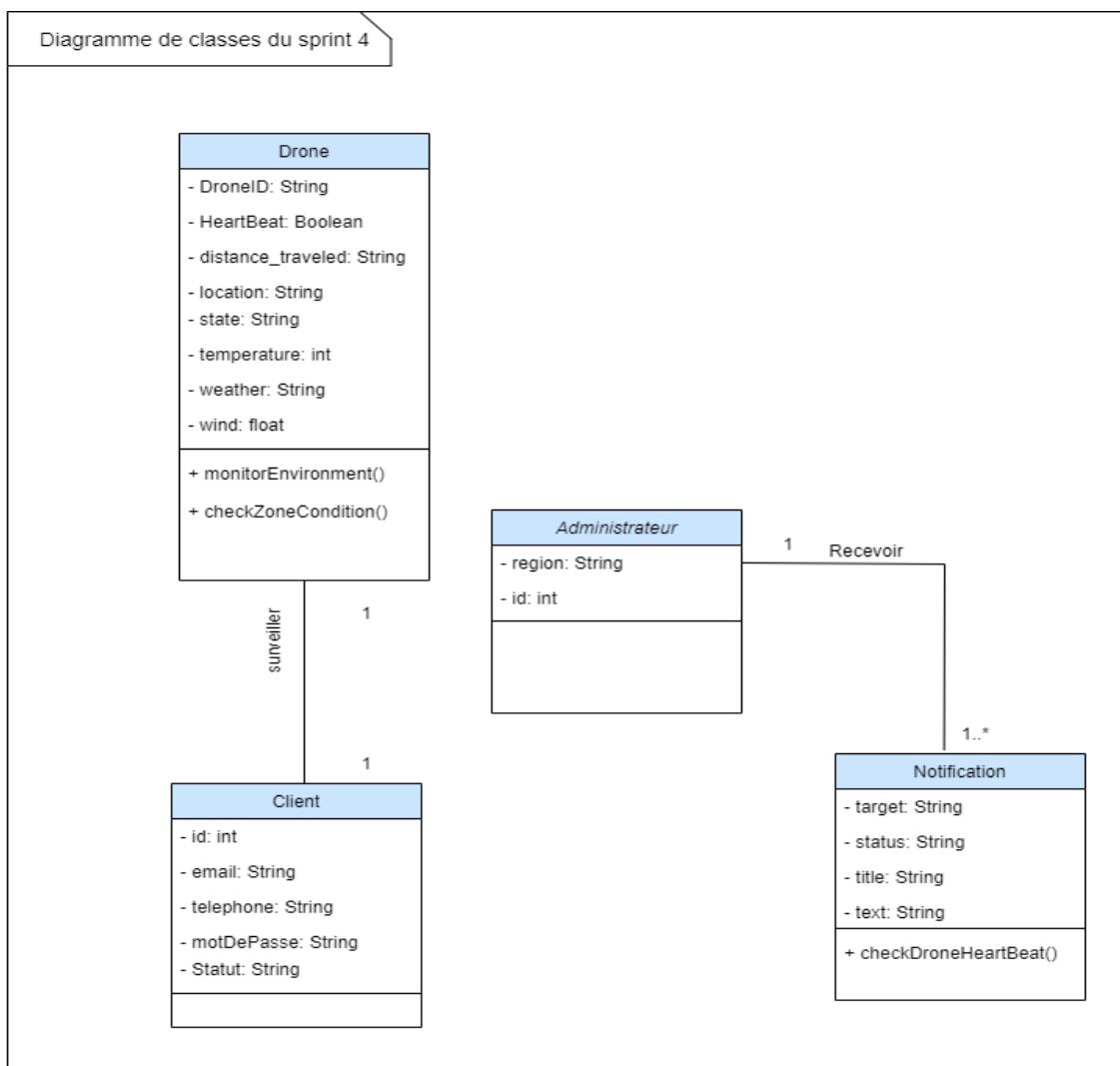
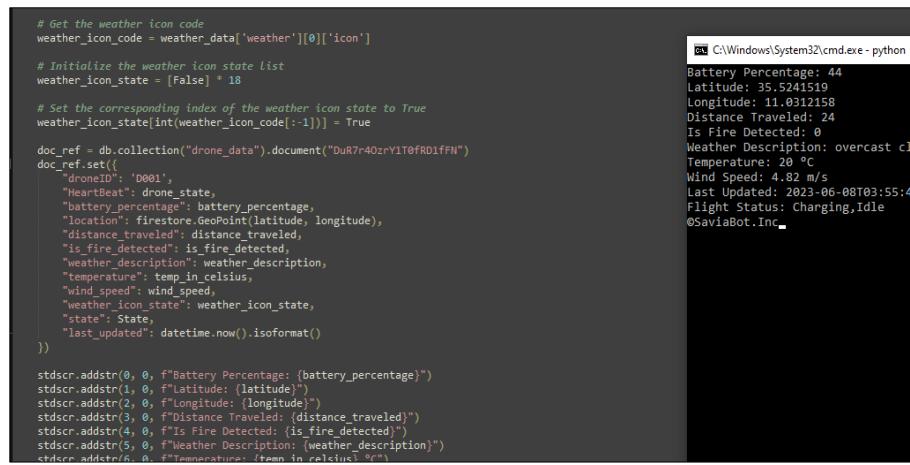


FIGURE 4.44 — Sprint 4 -Diagramme de Classes

4.3.4.3 Réalisation

- Envoie des données en temps réel du SaviaStation vers la Base de données Firestore



```
# Get the weather icon code
weather_icon_code = weather_data['weather'][0]['icon']

# Initialize the weather icon state list
weather_icon_state = [False] * 18

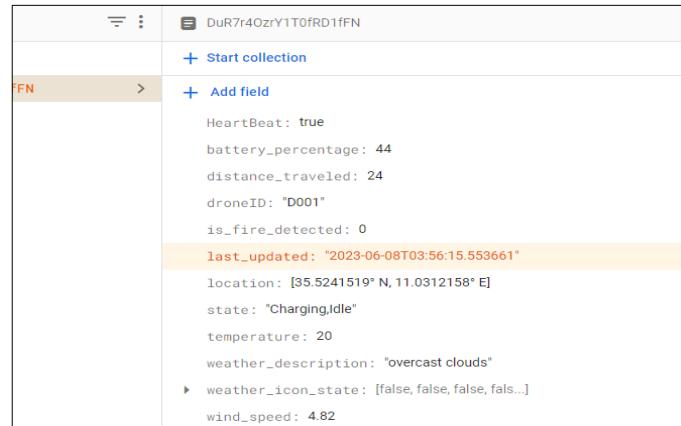
# Set the corresponding index of the weather icon state to True
weather_icon_state[int(weather_icon_code[-1])] = True

doc_ref = db.collection("drone_data").document("DuR7r4OzrY1T0fRD1fFN")
doc_ref.set({
    "droneID": 'D001',
    "HeartBeat": drom_state,
    "battery_percentage": battery_percentage,
    "location": firestore.Geopoint(latitude, longitude),
    "distance_traveled": distance_traveled,
    "is_fire_detected": is_fire_detected,
    "weather_description": weather_description,
    "temperature": temp_in_celsius,
    "wind_speed": wind_speed,
    "weather_icon_state": weather_icon_state,
    "state": State,
    "last_updated": datetime.now().isoformat()
})

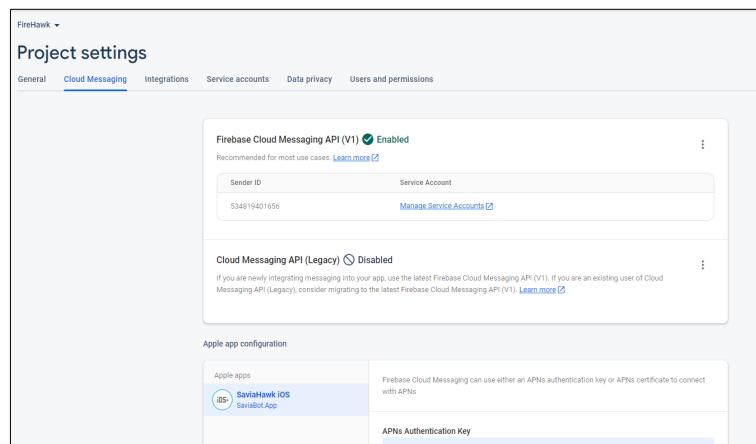
stdscr.addstr(0, 0, f"Battery Percentage: {battery_percentage}")
stdscr.addstr(1, 0, f"Latitude: {latitude}")
stdscr.addstr(2, 0, f"Longitude: {longitude}")
stdscr.addstr(3, 0, f"Distance Traveled: {distance_traveled}")
stdscr.addstr(4, 0, f"Is Fire Detected: {is_fire_detected}")
stdscr.addstr(5, 0, f"Weather Description: {weather_description}")
stdscr.addstr(6, 0, f"Temperature: {temp_in_celsius} °C")

```

Battery Percentage: 44
Latitude: 35.5241519
Longitude: 11.0312158
Distance Traveled: 24
Is Fire Detected: 0
Weather Description: overcast clouds
Temperature: 20 °C
Wind Speed: 4.82 m/s
Last Updated: 2023-06-08T03:55:44
Flight Status: Charging,Idle
@SaviaBot.Inc

FIGURE 4.45 — Sprint 4- Interface Client incluant la surveillance temps réel**FIGURE 4.46** — Sprint 4 - Real-time Database update

- Initialisation de Firebase Cloud Messaging

**FIGURE 4.47** — Sprint 4 -Initialisation de Firebase Cloud Messaging

4.3.4.4 Validation

Après avoir réalisé les interfaces, cette section présente les tests effectués sur les interfaces développées lors de ce sprint.

- Test de Surveillance en temps réel l'état des cultures et les zones vertes

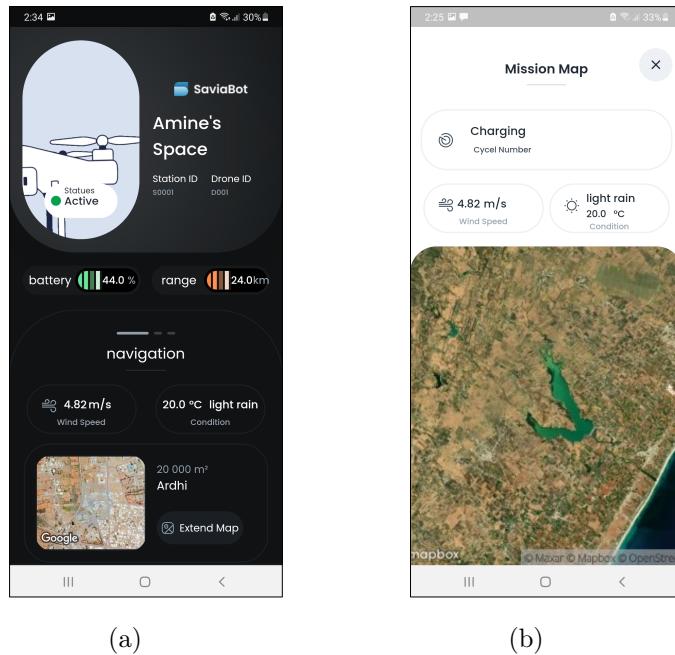


FIGURE 4.48 — Sprint 4 - Interface Client incluant la surveillance temps réel

- Test de fonction “recevoir les alertes”

_data	DuR7r4OzrY1T0fRD1ffN
document	+ Start collection + Add field <pre> HeartBeat: true battery_percentage: 44 distance_traveled: 24 droneID: "D001" is_fire_detected: 0 last_updated: "2023-06-04T23:41:38.863751" location: [35.5241519° N, 11.0312158° E] state: "Charging" temperature: 20 weather_description: "light rain" ▶ weather_icon_state: [false, false, false, false...] wind_speed: 4.82 </pre>
r40zrY1T0fRD1ffN	>

FIGURE 4.49 — Sprint 4 - Test de l'envoi d'alerte

4.3.5 Enregistrer les données de détection d'incendie pour analyse ultérieure

Une fois un incendie détecté, le script effectue les actions suivantes :

- Il crée un nouvel enregistrement dans la collection "Détection" de la base de données Firestore.
- Les informations liées à la détection sont ajoutées à cet enregistrement, telles que l'identifiant du drone (Prédéfinit), l'heure et la date de l'événement, ainsi que la géolocalisation de l'incendie (latitude et longitude).
- Les données de détection sont sauvegardées dans la base de données Firestore.



```

while True:
    # Read temperature data from AMG8833 sensor
    temperatures = [row for row in amg.pixels]

    # Check for fire detection
    is_fire_detected = any(temperature > fire_threshold for row in temperatures for temperature in row)

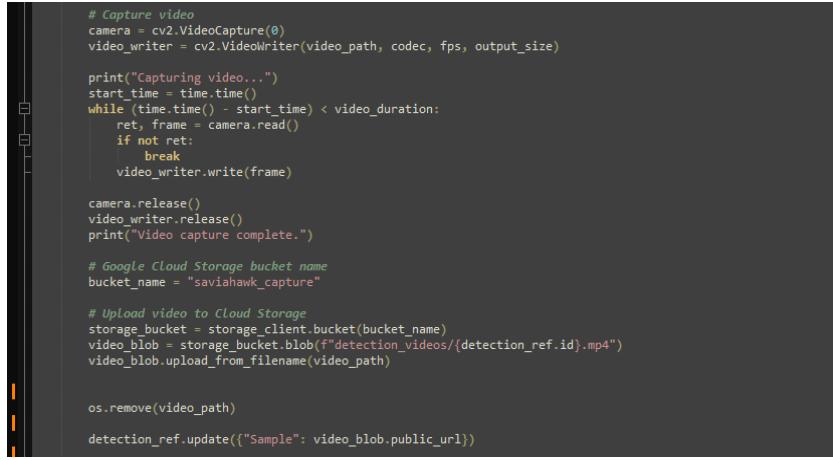
    if is_fire_detected:
        print("Fire detected!")

        # Retrieve registration tokens from Firestore based on DroneID
        db = firestore.client()
        users_ref = db.collection("User")
        query = users_ref.where("DroneID", "==", "D001")
        docs = query.get()

        # Add new Detection Document
        detection_ref = db.collection("Detection").document()
        detection_ref.set({
            "droneID": "D001",
            "EventTimeDate": datetime.now(),
            "Location": firestore.GeoPoint(latitude, longitude)
        })
    
```

FIGURE 4.50 — collection "Détection" - Firestore

- Une vidéo est capturée à l'aide de la caméra Raspberry. La durée de la vidéo est définie par la variable "video-duration".
- La vidéo capturée est enregistrée dans un fichier sur le système.
- La vidéo est téléchargée vers un compartiment (bucket) Google Cloud Storage spécifié.
- Le fichier local de la vidéo est supprimé pour libérer de l'espace de stockage.
- L'URL de la vidéo téléchargée est ajoutée à l'enregistrement de détection dans la base de données Firestore.



```
# Capture video
camera = cv2.VideoCapture(0)
video_writer = cv2.VideoWriter(video_path, codec, fps, output_size)

print("Capturing video...")
start_time = time.time()
while (time.time() - start_time) < video_duration:
    ret, frame = camera.read()
    if not ret:
        break
    video_writer.write(frame)

camera.release()
video_writer.release()
print("Video capture complete.")

# Google Cloud Storage bucket name
bucket_name = "saviahawk_capture"

# Upload video to Cloud Storage
storage_bucket = storage_client.bucket(bucket_name)
video_blob = storage_bucket.blob(f"detection_videos/{detection_ref.id}.mp4")
video_blob.upload_from_filename(video_path)

os.remove(video_path)
detection_ref.update({"Sample": video_blob.public_url})
```

FIGURE 4.51 — Vidéo de Détection - Google Cloud Storage

Les données de détection d’incendie sont ainsi sauvegardées dans la base de données Firestore pour une analyse ultérieure. Cela permet de conserver un historique des détections d’incendie, y compris les informations telles que la date, l’heure, la géolocalisation et les vidéos associées.

4.4 Conclusion

Dans ce chapitre, nous avons présenté le design matériel intégré dans notre projet, ainsi que la présentation de l’organisation de notre travail et l’explication du choix du matériel adéquat. Finalement, nous avons décrit le processus d’assemblage du matériel pour réaliser notre solution.

5.1 Introduction

Ce chapitre présente la dernière version de notre projet, mettant en évidence le sprint «surveillance et amélioration du système de détection du drone». Ce sprint se concentre sur les aspects cruciaux des alertes de détection, de l'exploration des données collectées par les drones, ainsi que de la surveillance et de l'amélioration des systèmes. Dans un monde en constante évolution, ces éléments sont indispensables pour assurer la sécurité et l'efficacité de notre projet.

5.2 SPRINT 5 : “Surveillance et amélioration du système de détection du drone”

Dans cette section, nous abordons le dernier sprint intitulé «surveillance et amélioration du système de détection du drone». Nous commencerons par présenter l'organisation mise en place ainsi que le backlog de ce sprint, qui détaille les tâches et les fonctionnalités à développer. Ensuite, nous passerons à la phase d'analyse et la solution conceptuelle. Enfin, nous présenterons les différentes réalisations et les progrès accomplis dans le cadre de ce sprint.

5.2.1 Backlog du Sprint 5

Le tableau 5.1 donne un aperçu détaillé sur le Backlog du sprint qui prend en charge la fonctionnalité "Surveillance et amélioration du système de détection du drone".

ID	Sprint	User Story	jour
5	Surveillance et amélioration du système de détection du drone	5.1 Réalisation de la fonctionnalité "Explorer les données collectées par les drones"	2
		5.2 Réalisation de la fonctionnalité "Recevoir les alertes de détection d'incendies à un stade précoce"	3
		5.2.1 Réalisation de la fonctionnalité "Confirmer l'incendie"	3
		5.3 Amélioration des algorithmes de détection d'incendies	3

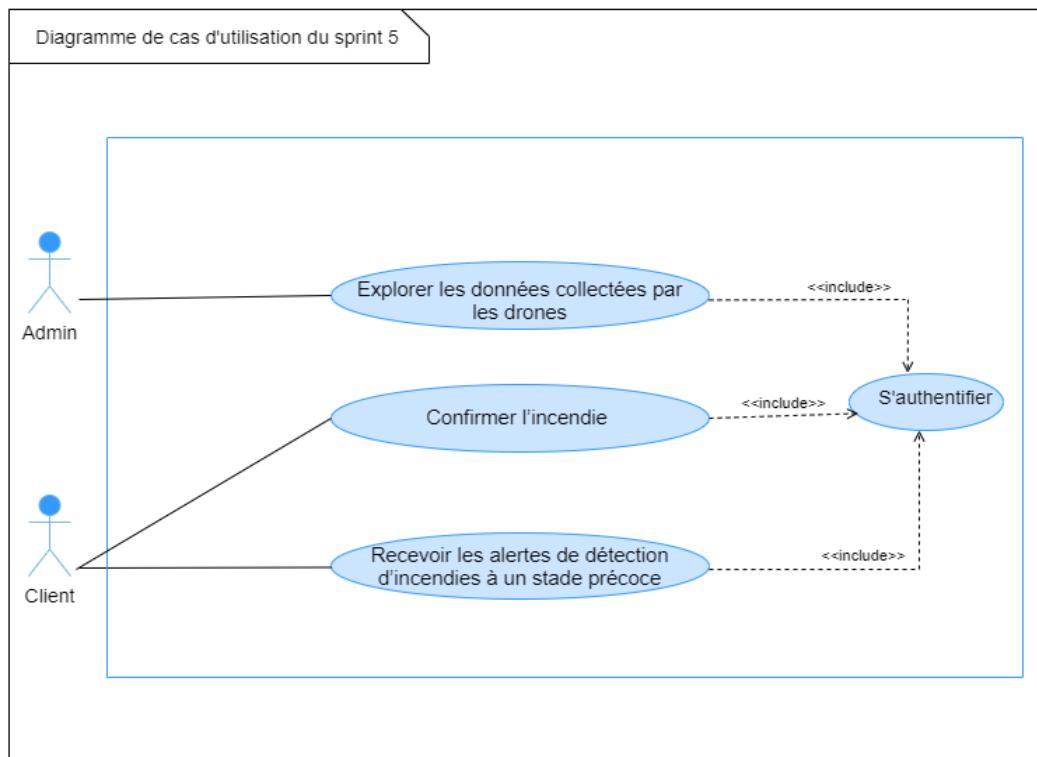
TABLE 5.1 — Backlog du sprint 5 : "Surveillance et amélioration du système de détection du drone"

5.2.2 Analyse

Dans cette partie, nous présentons le diagramme de cas d'utilisation, les descriptions textuelles pour chaque cas, ainsi que les diagrammes de séquence.

Diagramme de cas d'utilisation :

La figure 5.1 représente le diagramme de cas d'utilisation pour le sprint 5.

**FIGURE 5.1** — Diagramme de cas d'utilisation du sprint 5**Diagramme de cas d'utilisation :**

- Description textuelle de cas d'utilisation «Explorer les données collectées par les drones»

Nom du CU	Explorer les données collectées par les drones
Acteurs	Administrateur
Résumé	L'administrateur explore les données collectées par les drones.
Pré-conditions	L'administrateur s'est authentifié.
Scénario Nominal	1-L'administrateur accède à son profil. 2-Le système charge et affiche l'interface correspondant au profil de l'administrateur. 3-L'administrateur charge la liste des clients actifs. 4-L'administrateur sélectionne un client pour explorer les données collectées par son drone.
Post-conditions	L'administrateur a exploré avec succès les données collectées par le drone du client.

TABLE 5.2 — Description textuelle du scénario de cas d'utilisation "Explorer les données collectées par les drones"

- Description textuelle de cas d'utilisation «Recevoir les alertes de détection d'incendies à un stade précoce»

Nom du CU	Recevoir les alertes de détection d'incendies à un stade précoce
Acteurs	Client
Résumé	Le client reçoit une alerte de détection d'incendie.
Pré-conditions	Le client s'est authentifié.
Scénario Nominal	<p>1-Le client reçoit une alerte de détection d'incendie (notification push + SMS).</p> <p>2-Le client accède à son interface d'historique de détection d'incendie.</p> <p>3-Le système charge et affiche l'interface correspondant à l'historique de détection d'incendie du client.</p> <p>4-Le client visualise la date, l'heure et l'emplacement de l'incendie détecté sur la carte.</p>
Post-conditions	L'administrateur a exploré avec succès les données collectées par le drone du client.

TABLE 5.3 — Description textuelle du scénario de cas d'utilisation "Recevoir les alertes de détection d'incendies à un stade précoce"

- Description textuelle de cas d'utilisation «Confirmer l'incendie»

Nom du CU	Confirmer l'incendie
Acteurs	Client
Résumé	Le client confirme la détection d'un incendie.
Pré-conditions	Le client s'est authentifié.
Scénario Nominal	1-Le client accède à son interface d'historique de détection d'incendie. 2-Le système charge et affiche l'interface correspondant à l'historique de détection d'incendie du client. 3-Le client visualise la date, l'heure et l'emplacement de l'incendie détecté sur la carte. 4-Si la détection est confirmée, le client valide en sélectionnant "oui". 5-Si la détection est infirmée, le client valide en sélectionnant "non".
Post-conditions	La confirmation de l'incendie a été enregistrée avec succès.

TABLE 5.4 — Description textuelle du scénario de cas d'utilisation "Confirmer l'incendie"

5.2.3 Conception

Dans cette section, nous présentons les diagrammes de séquences et de classes utilisés dans le cadre du sprint. Les diagrammes de séquence illustrent la séquence des messages échangés entre les objets nécessaires pour réaliser les fonctionnalités du scénario, tandis que les diagrammes de classes décrivent la structure du système.

Diagramme de séquence :

- Diagramme de séquence du cas d'utilisation “Explorer les données collectées par les drones” :

La figure 5.2 illustre le déroulement du cas d'utilisation Explorer les données collectées par les drones. L'administrateur, une fois authentifié et accédant à son compte, a la possibilité de consulter l'historique de détection de chaque client. Après avoir choisi un client spécifique, l'administrateur explore en détail l'historique de détection pour examiner les données collectées par les drones. Cette fonctionnalité permet à l'administrateur d'analyser en détail les informations enregistrées.

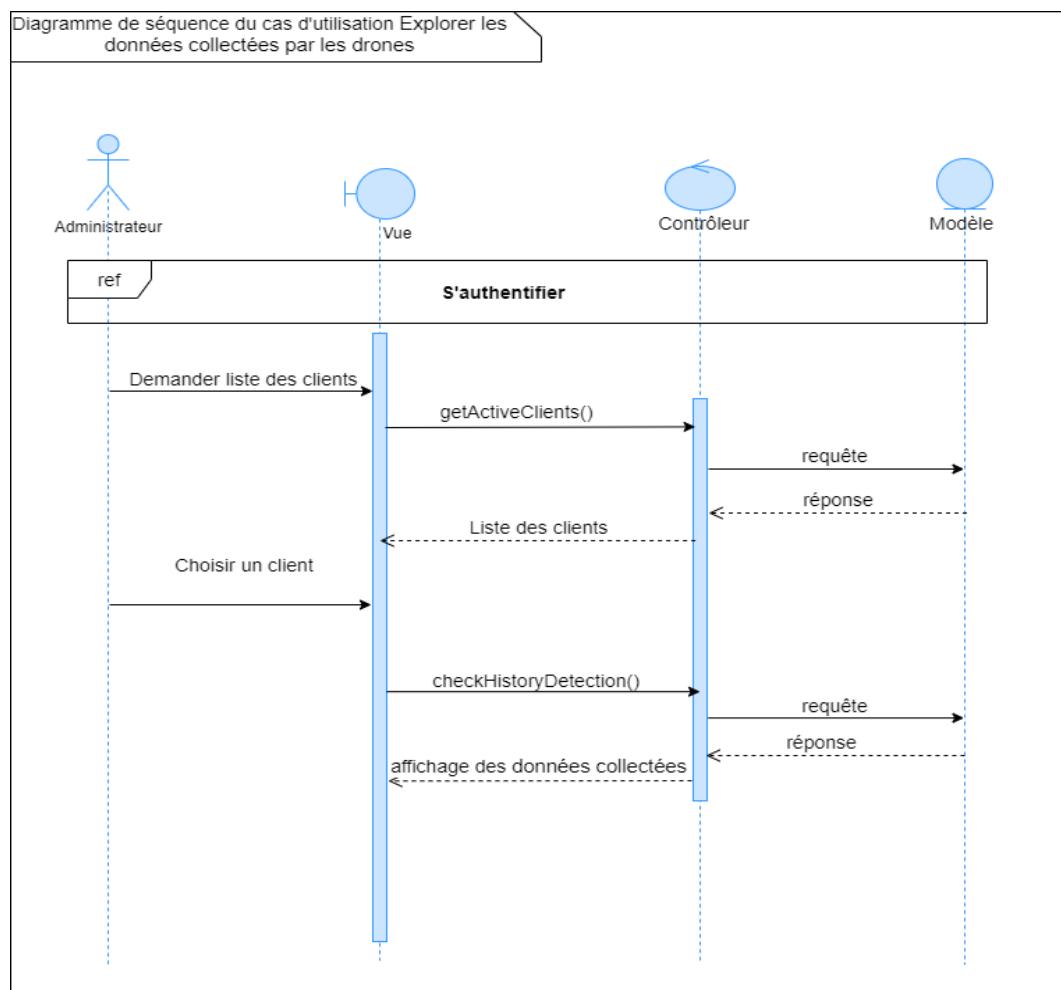


FIGURE 5.2 — Sprint 5 -Diagramme de séquence “Explorer les données collectées par les drones”

- Diagramme de séquence du cas d'utilisation “Recevoir les alertes de détection d’incendies à un stade précoce” :

La figure 5.3 illustre le déroulement du cas d’utilisation “Recevoir les alertes de détection d’incendies à un stade précoce”. Une fois que le client reçoit une alerte de détection d’incendie, il accède à son compte pour obtenir les informations de détection. De plus, le client a la possibilité d’accéder à une carte pour visualiser l’emplacement exact de l’incendie.

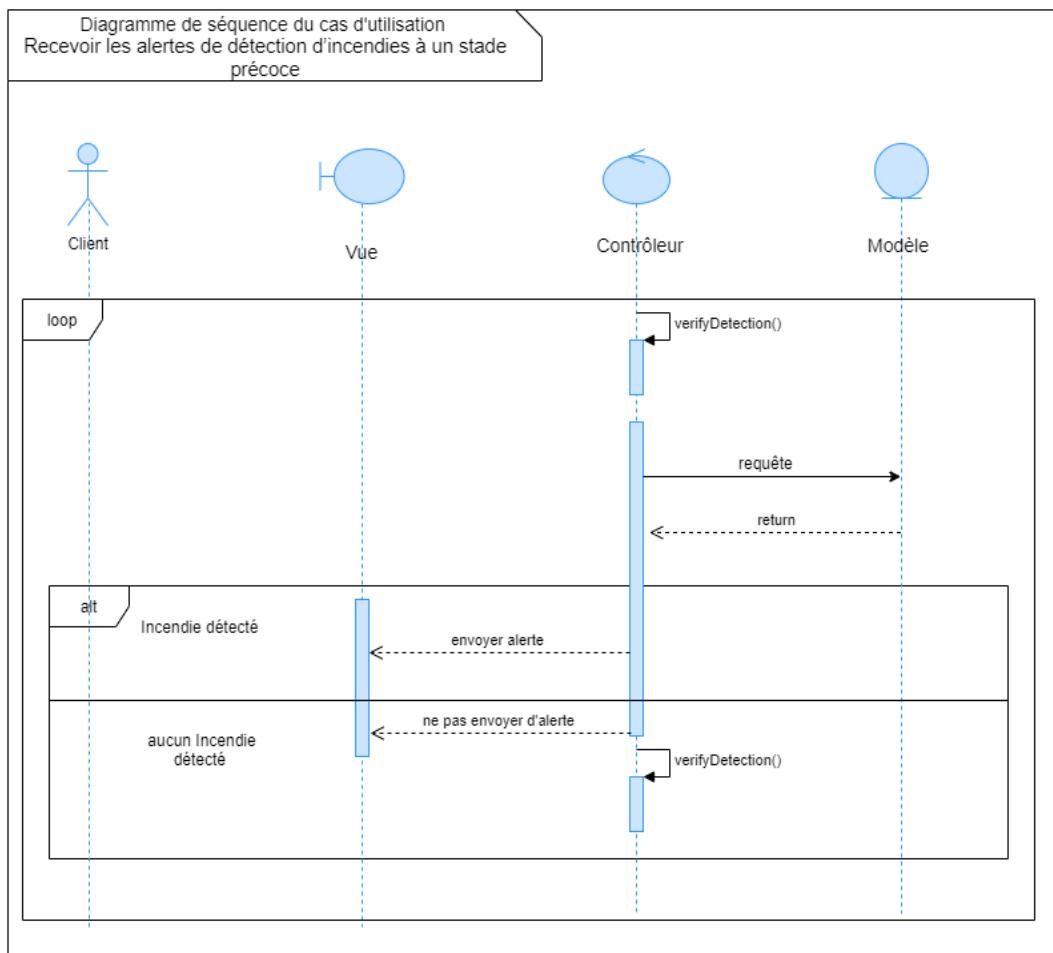


FIGURE 5.3 — Sprint 5- Diagramme de séquence “Recevoir les alertes de détection d’incendies à un stade précoce”

- Diagramme de séquence du cas d’utilisation “Confirmer l’incendie” :

La figure 5.4 illustre le déroulement du cas d’utilisation “Confirmer l’incendie”. Après avoir reçu et vérifié l’alerte d’incendie, le client accède à son compte pour confirmer si l’incendie est précis ou non. Cette fonctionnalité est essentielle pour améliorer la détection, car elle permet de valider et de préciser l’exactitude des alertes d’incendie.

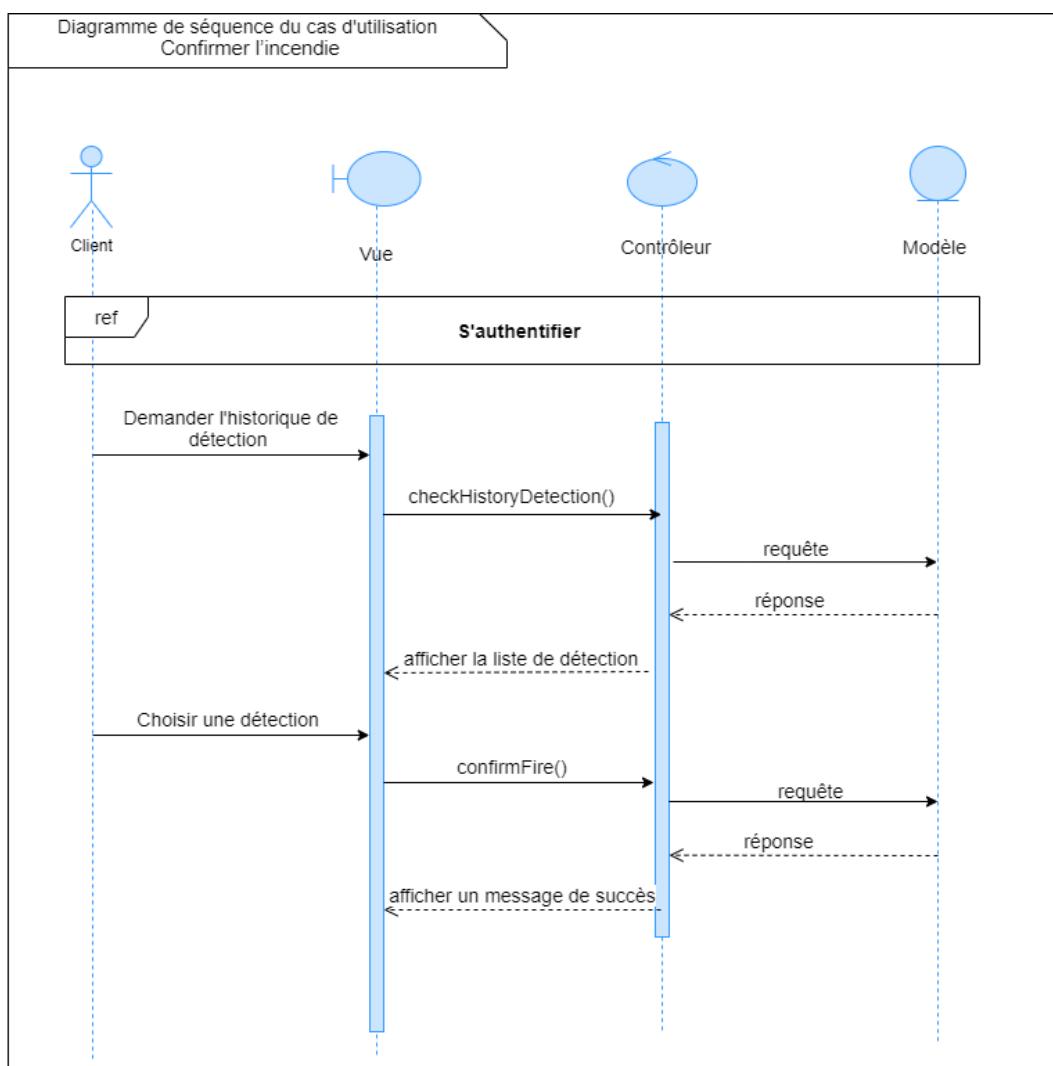


FIGURE 5.4 — Sprint 5 -Diagramme de séquence “Confirmer l'incendie”

Diagramme de Classes :

Dans cette sous-section, nous présenterons le diagramme de classes du sprint 4.

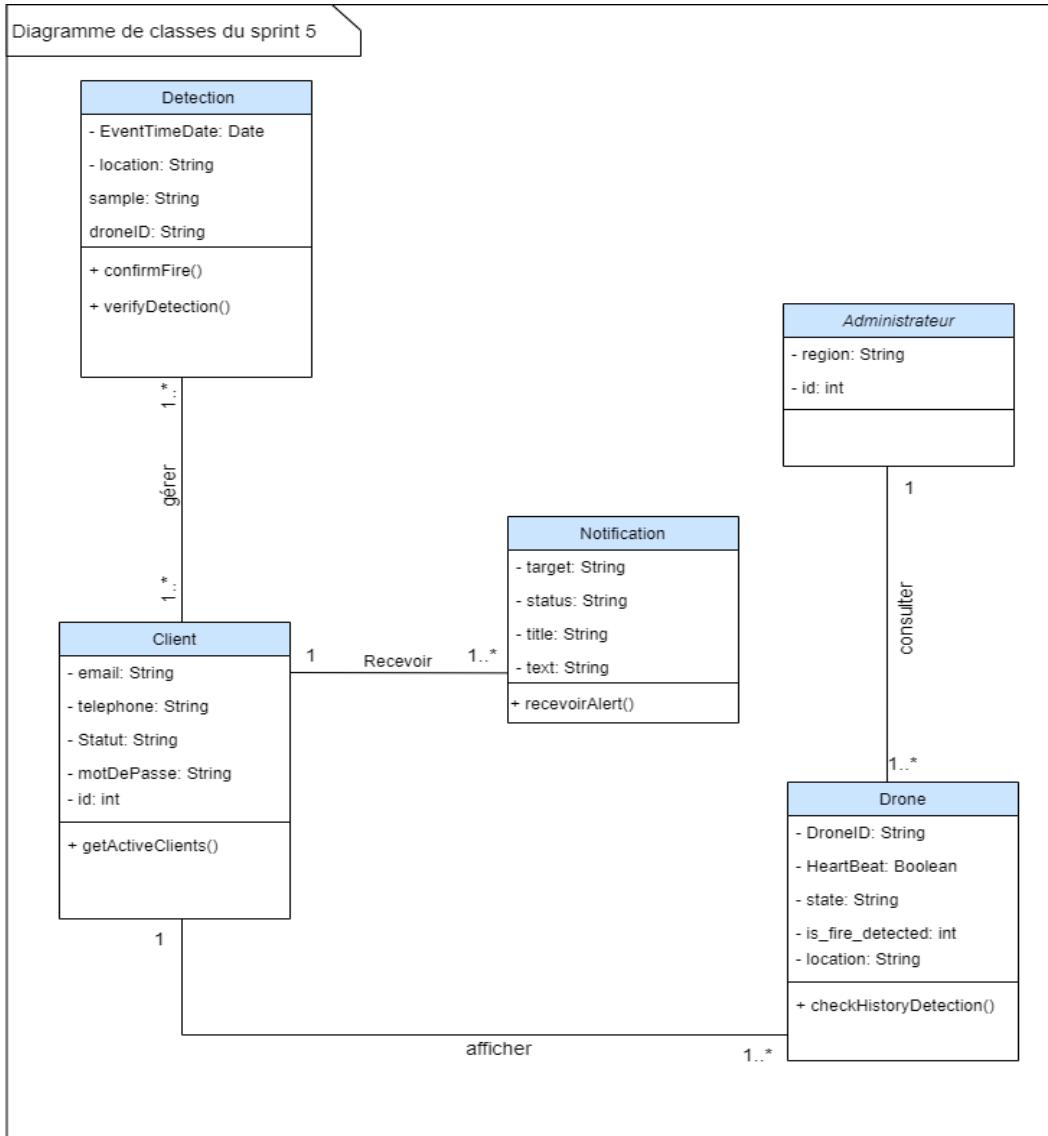


FIGURE 5.5 — Sprint 5 -Diagramme de Classes

5.2.4 Réalisation

- Initialisation de l'API SMS Twilio :

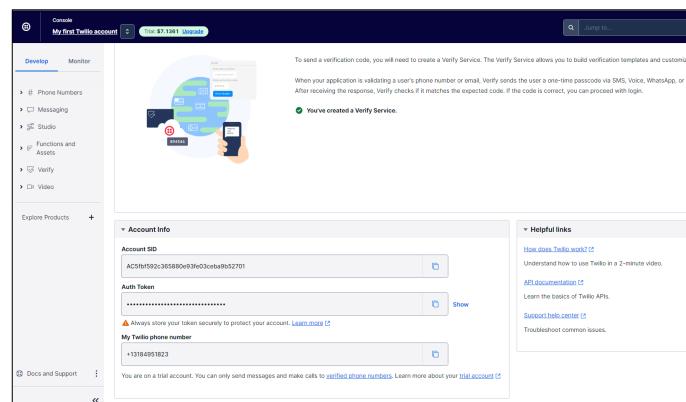


FIGURE 5.6 — Sprint 5 -Initialisation de l'API SMS Twilio

- Initialisation FCM Auth Token dans l'application SaviaBot :

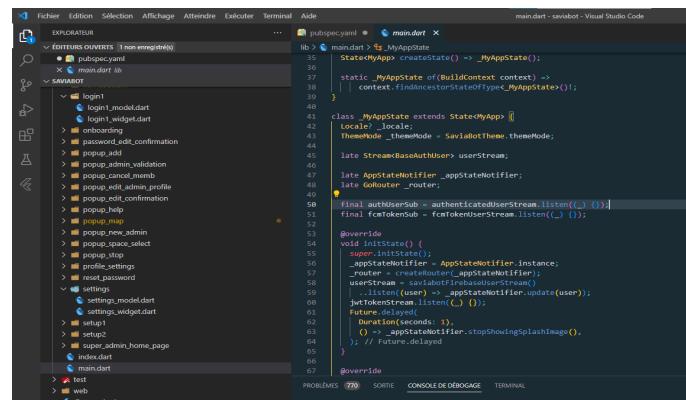


FIGURE 5.7 — Sprint 5 -FCM Auth Token

- Déclenchement d'action du drone SaviaHawk :

```

51
52   # Send push notifications to all FCM tokens
53   for registration_token in registration_tokens:
54     message = messaging.Message(
55       data={
56         "title": "New fire detected!",
57         "body": "Tap to view the detection history",
58         "click_action": "OPEN_DETECTION_HISTORY"
59       },
60       token=registration_token,
61     )
62     response = messaging.send(message)
63     print("Push notification sent successfully to", registration_token)
64
65
66   # Send SMS notifications
67   for doc in docs:
68     data = doc.to_dict()
69     phone_number = data.get("phone_number")
70     if phone_number:
71       try:
72         message = client.messages.create(
73           body="New fire detected in your space. Click here to view the location: {map\_url}",
74           from=_twilio_phone_number,
75           to=phone_number
76         )
77       print("SMS notification sent to", phone_number)
78     except twilio.base.exceptions.TwilioRestException as e:
79       print("Twilio error:", e)
80
81 ...
82 ...
83 ...
84 ...
85 ...
86 ...
87 ...
88 ...
89 ...
89 ...
90 ...
91 ...
92 ...
93 ...
94 ...
95 ...
96 ...
97 ...
98 ...
99 ...
100 ...
101 ...
102 ...
103 ...
104 ...
105 ...
106 ...
107 ...
108 ...
109 ...
110 ...
111 ...
112 ...
113 ...
114 ...
115 ...
116 ...
117 ...
118 ...
119 ...
120 ...
121 ...
122 ...
123 ...

```

FIGURE 5.8 — Sprint 5 -Push Notification + SMS

5.2.5 Validation

- Interface Explorer les données collectées par les drones :

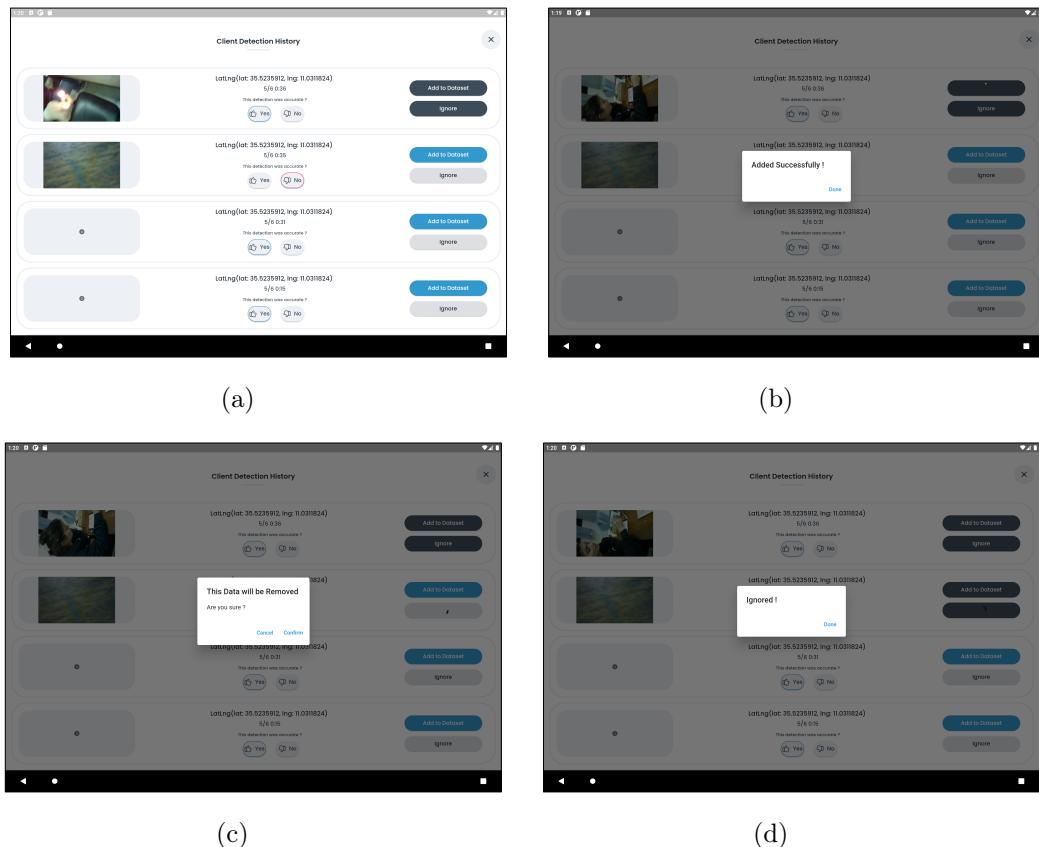


FIGURE 5.9 — Sprint 5 - Explorer les données collectées par les drones

- Recevoir les alertes de détection d'incendies à un stade précoce :

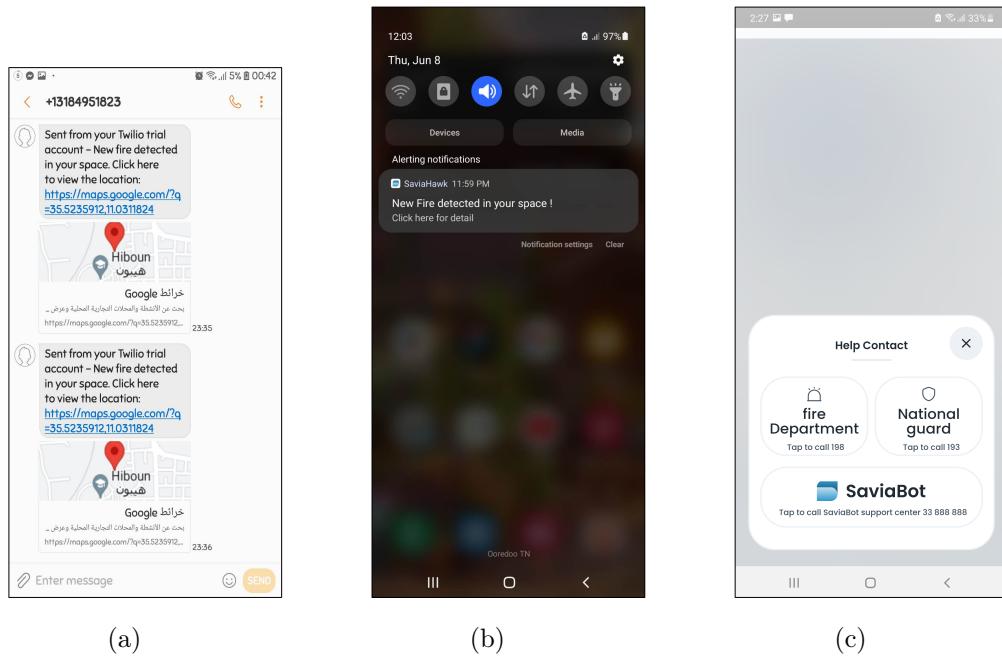


FIGURE 5.10 — Sprint 5 - Recevoir les alertes de détection d’incendies à un stade précoce

- Confirmation l’incendie de la part du Client - Interface Historique des détections :

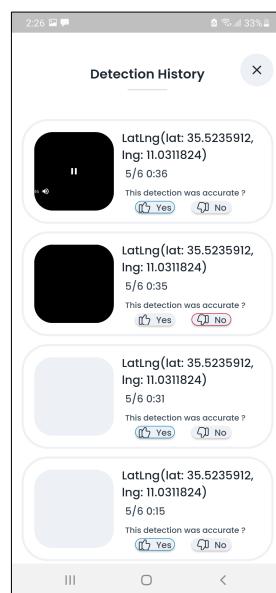


FIGURE 5.11 — Sprint 5 – Interface Historique des détections

5.3 Amélioration des algorithmes de détection d'incendies

Dans cette étude, nous avons mis en place un système d'amélioration continue de nos algorithmes de détection d'incendies. Nous avons développé un mécanisme de collecte et de stockage des données provenant du drone lorsqu'il détecte un incendie. Lorsqu'un incendie est détecté, le drone capture une courte vidéo de 5 secondes et enregistre les données provenant de sa caméra thermique. Ces données sont ensuite associées et synchronisées, comme montre la figure ci-contre :

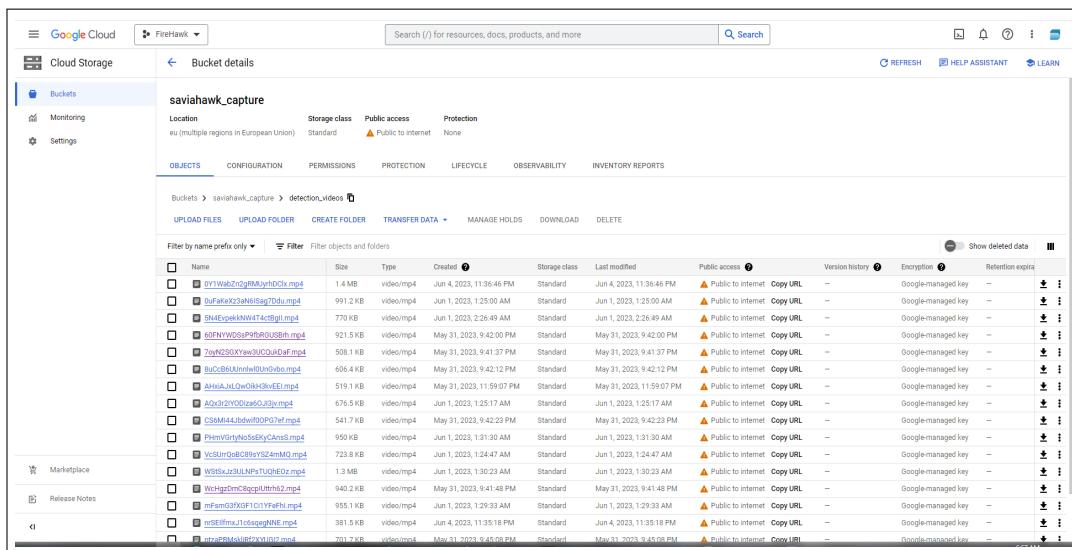


FIGURE 5.12 — saviahawk-capture google cloud bucket

Nous avons également intégré une fonctionnalité interactive pour l'utilisateur. L'utilisateur dispose d'un bouton permettant d'indiquer si la détection de l'incendie était précise ou non. Cette rétroaction utilisateur est importante pour améliorer la précision de nos algorithmes de détection.

Les données collectées, comprenant les vidéos, les données thermiques et les rétroactions utilisateur, sont ensuite stockées dans le cloud. Nous utilisons ces données pour re-entraîner notre ensemble de données. En utilisant ces nouvelles données, nous pouvons affiner et ajuster notre modèle de détection d'incendies. Nous effectuons un nouvel entraînement en utilisant des algorithmes d'apprentissage automatique avancés.

Une fois le nouvel entraînement terminé, nous exportons de nouveaux fichiers de poids (weights) contenant les améliorations apportées à notre modèle. Ces fichiers sont ensuite envoyés de manière sécurisée vers le drone, qui les utilise pour mettre à jour son système de détection d'incendies.

Cette approche d'amélioration continue garantit que nos algorithmes de détection d'incendies évoluent et s'adaptent en fonction des nouvelles données et des retours utilisateurs. Cela permet d'améliorer progressivement la précision de notre système de détection d'incendies et d'assurer une réponse plus fiable et efficace aux situations d'urgence.

5.4 Conclusion

Dans ce chapitre, nous avons mis en avant les fonctionnalités essentielles telles que l'exploration des données collectées et la réception d'alertes d'incendie. De plus, nous avons souligné l'importance de l'amélioration continue des algorithmes grâce à la collecte de données. Ces efforts sont orientés vers l'objectif de garantir une sécurité optimale et une réponse efficace aux situations d'urgence, contribuant ainsi à renforcer la fiabilité et l'efficacité de notre projet.

Conclusion générale

Ce travail s'intègre dans le cadre du projet de fin d'études au sein de l'entreprise Quetrattech et en vue d'obtention du diplôme de licence en Ingénierie de Système informatique à l'institut Supérieur d'Informatique de Mahdia (ISIMA). Il a pour objectif de proposer une solution sous forme d'un drone autonome qui identifie rapidement les incendies afin d'intervenir rapidement et minimiser ces dommages.

La mise en place d'une application mobile connecté a été la réponse adéquate à cette problématique.

Le présent manuscrit détaille toutes les étapes par lesquelles nous sommes passées pour arriver au résultat attendu. Nous avons essayé tout au long de notre stage de construire notre projet sprint par sprint en utilisant la méthodologie Kanban.

Ce stage a été, sous plusieurs façades, riches d'enseignements. Nous avons commencé tout d'abord par comprendre le contexte général de notre projet ainsi que l'organisme d'accueil. Nous avons présenté ensuite l'étude de l'existant et les principaux objectifs de notre solution logicielle et la demande du projet. Nous avons préparé par la suite le planning de travail en respectant les priorités de nos besoins et la description de la méthodologie pour laquelle nous avons opté.

Dans le deuxième chapitre, nous avons commencé par une étude préliminaire du projet tout en décrivant l'analyse et la spécification des besoins au bout de laquelle nous avons abouti à une définition précise des différentes fonctionnalités attendues de la solution envisagée. Ces fonctionnalités ont été définies à travers la description des besoins et des différents cas d'utilisation à envisager, ainsi l'identification des acteurs en élaborant un diagramme de d'utilisation global et un backlog du produit. Par la suite, il convient de répartir les besoins identifiés en sprints, tout en établissant un calendrier prévisionnel et en définissant l'architecture détaillée de système.

Dans le chapitre trois, nous avons présenté, dans un premier lieu, le premier sprint « Gestion des Comptes ». Dans un deuxième lieu, nous présenterons le deuxième sprint

« Planification de missions et avis clients » dans chaque sprint, nous présentons l'organisation et son backlog de sprint, l'analyse de notre sprint, la conception en décrivant son architecture. Nous nous sommes ensuite intéressés à la description détaillée des différents modules à l'aide du diagramme de classes, cas d'utilisation et de séquence selon le formalisme UML et l'exécution montrant quelques conceptions d'interfaces.

Le quatrième chapitre, qui présente le design matériel utilisé. Premièrement, nous avons commencé par la présentation de l'organisation de notre projet et l'explication du choix du matériel adéquat. Enfin, nous avons décrit le processus d'assemblage du matériel pour réaliser notre solution.

Finalement, nous clôturons notre travail par le chapitre cinq, Ce chapitre présente la dernière version de notre projet, il met en évidence le sprint 5 « surveillance et amélioration du système de détection du drone ». Ce sprint définit les aspects cruciaux des alertes de détection, de l'exploration des données collectées par les drones, ainsi que de la surveillance et de l'amélioration des systèmes.

Nous finalisons ce travail par diverses perspectives que nous allons présenter. Premièrement, nous n'avons pas pu réaliser l'automatisation de la planification des trajets des drones. Cela s'explique par le fait que cette action se réalise manuellement. De plus, nous aurions souhaité ajouter un capteur LiDAR qui aurait permis améliorer l'autonomie du drone. C'est pour une raison financière que nous n'avons pas pu ajouter ce capteur trop coûteux. Nous aurions également aspiré à la création d'un affichage en 3D de l'espace en utilisant des scans LiDAR en 3D de l'environnement. À cela s'ajoute l'ambition de pouvoir développer un algorithme ayant la capacité de détecter les maladies ou les problèmes des espaces verts, ainsi que d'évaluer leur état de végétation.

Le développement d'un deuxième drone serait favorable. Dans l'idéal, celui-ci serait plus grand, afin que le système puisse agir et éteindre les flammes dès leur apparition, sans nécessiter quelque intervention humaine.

Finalement, malgré les limites financières et matérielles, nous avons pu parvenir à un très bon résultat concernant notre travail.

Ce projet nous a été très instructif du point de vue des connaissances acquises. Il nous a permis la mise en œuvre de notre savoir-faire en le concrétisant par la réalisation pratique. Il nous a donné l'occasion d'approfondir notre maîtrise des différentes phases de création d'une application mobile connecté. Nous estimons que, bien que notre projet soit loin d'être fini, il représente néanmoins, un potentiel énorme. En effet, ce projet n'est qu'une première contribution vers une solution plus moderne pour être l'un des applications le plus fiable et afin de faciliter la détection des incendies. Nous tenons finalement à exprimer

notre grande fierté du travail réalisé dont nous garderions un excellent souvenir et notre satisfaction d'avoir pu travailler en bonnes conditions.

En conclusion, nous souhaitons que ce modeste travail apporte la satisfaction aux responsables de Quetratech ainsi qu'aux membres de Jury.

Webographie

- [1] *BoWFireDataset.zip*. URL : <https://bitbucket.org/gbdi/bowfire-dataset/downloads/> (visité le 18/04/2023).
- [2] *En Californie, le plus vaste incendie de l'année progresse*. URL : <https://www.lacroix.com/En-Californie-vaste-incendie-annee-progresse-2022-08-01-1301227190> (visité le 06/03/2023).
- [3] *Feux de forêt en 2022 : record de surfaces brûlées et d'émissions de carbone en Europe*. URL : https://www.francetvinfo.fr/faits-divers/incendie/feux-de-foret-en-2022-record-de-surfaces-brulees-et-d-emissions-de-carbone-en-europe_5542065.html (visité le 05/03/2023).
- [4] *forestryimages.org*. URL : <https://www.forestryimages.org/browse/subthumb.cfm?sub=740> (visité le 30/04/2023).
- [5] *L'écosystème internet des objets*. URL : <http://www.univ-bejaia.dz/xmlui/bitstream/handle/123456789/8036/Introduction%C3%A0%20l%20internet%20de%20l%E2%80%99objet%20et%20r%C3%A9alisation%20d%E2%80%99un%20syst%C3%A8me%20domotique.pdf?sequence=1&isAllowed=y> (visité le 01/06/2023).
- [6] *Les incendies de forêt en Tunisie*. URL : <https://news.mongabay.com/2022/11/2022-amazon-fires-tightly-tied-to-recent-deforestation-new-data-show/> (visité le 06/03/2023).
- [7] *Les incendies de forêt en Tunisie*. URL : <https://www.sudouest.fr/faits-divers/incendies/bresil-deja-plus-d-incendies-en-amazonie-sur-les-neuf-premiers-mois-de-2022-que-sur-toute-l-annee-2021-12337353.php> (visité le 06/03/2023).
- [8] *Lien GitHub du Project SaviaBot*. URL : <https://github.com/nomino0/SaviaBot.git>.

- [9] *Modèle-vue-contrôleur*. URL : <https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur> (visité le 30/04/2023).
- [10] *Nouvel incendie d'ampleur en Californie, deux morts et des milliers d'évacués*. URL : <https://www.lesoleil.com/2022/09/06/nouvel-incendie-d'ampleur-en-californie-deux-morts-et-des-milliers-devacués-2cd945b5f3c8483eeb26149d631451ed/> (visité le 06/03/2023).
- [11] *Ororatech™ - Startup de surveillance des incendies de forêt par Satellites*. URL : <https://spacenews.com/german-startup-raises-funding-for-wildfire-monitoring-satellites/> (visité le 15/03/2023).
- [12] *Pano AI active wildfire detection*. URL : <https://www.pano.ai/> (visité le 15/03/2023).
- [13] *Robocare Solutions*. URL : <https://robocare.tn/#Our-Solution> (visité le 15/03/2023).
- [14] *THE FLAME DATASET : AERIAL IMAGERY PILE BURN DETECTION USING DRONES (UAVS)*. URL : <https://ieee-dataport.org/open-access/flame-dataset-aerial-imagery-pile-burn-detection-using-drones-uavs> (visité le 30/04/2023).
- [15] *TUNISIE : LE NOMBRE D'INCENDIES DE FORÊTS EN 2020 DÉPASSE CE-LUI ENREGISTRÉ ENTRE 2011 ET 2019*. URL : <https://mapecology.ma/actualites/tunisie-nombre-dincendies-de-forets-2020-depasse-celui-enregistre-entre-2011-2019/> (visité le 05/03/2023).

Ce rapport décrit les différentes étapes nécessaires pour la réalisation du travail dans le cadre du projet de fin d'études en vue de l'obtention de la licence en informatique de l'Institut Supérieur d'informatique Mahdia. Notre projet consiste à développer un système complet de détection précoce des incendies de forêt en utilisant un drone autonome et une application mobile. L'objectif principal de ce projet est d'améliorer la sécurité en fournissant une solution efficace et rapide pour la détection et la notification des incendies de forêt, afin de minimiser les dégâts environnementaux et économiques.