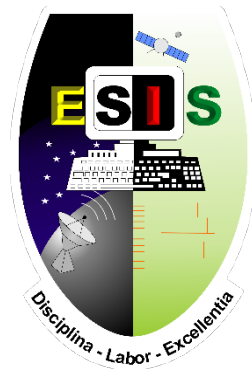


ÉCOLE SUPÉRIEURE D'INFORMATIQUE SALAMA

République Démocratique Du Congo

Province du Haut – Katanga

www.esisalama.org



CONCEPTION D'UN SYSTÈME D'APPEL AU SECOURS : CAS
D'ENLÈVEMENT DANS LA VILLE DE LUBUMBASHI

*Travail présenté et défendu en vue de
l'obtention du grade d'ingénieur technicien
en Génie Logiciel*

Par : **KALUME NSENGA YVES**

Option : Génie Logiciel

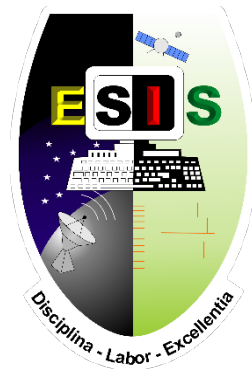
SEPTEMBRE 2023

ÉCOLE SUPÉRIEURE D'INFORMATIQUE SALAMA

République Démocratique Du Congo

Province du Haut – Katanga

www.esisalama.org



CONCEPTION D'UN SYSTÈME D'APPEL AU SECOURS : CAS
D'ENLÈVEMENT DANS LA VILLE DE LUBUMBASHI

*Travail présenté et défendu en vue de
l'obtention du grade d'ingénieur technicien
en Génie Logiciel*

Par : **KALUME NSENGA YVES**

Option : Génie Logiciel

Directeur : Mr. Yves NDETURUYE

SEPTEMBRE 2023

ÉPIGRAPHE

“L’homme et sa sécurité doivent constituer la première préoccupation de toute aventure technologique”

Albert Einstein

DÉDICACE

À chers parents MWANZA LUBAVU Matthieu et KAYUMBA MULUTU Anastasie,
mes frères KAZEMBE KIMBAYA Luc et LEWILE KIMBAYA Prodige ainsi qu'à ma
sœur LUBAVU MWAMBA Audrey.

REMERCIEMENTS

Au terme de notre périple académique à l'École Supérieure d'Informatique Salama (E.S.I.S), il est de notre devoir de rendre hommage aux personnes qui ont contribué de manière significative à notre parcours et à la réalisation de ce travail.

En tout premier lieu, nous exprimons notre profonde reconnaissance au Dieu Tout Puissant pour sa grâce inestimable, sa miséricorde constante et la santé qui nous a été octroyée tout au long de notre cursus à l'E.S.I.S jusqu'à l'accomplissement de ce présent travail.

Nous adressons nos remerciements à l'ensemble du corps administratif et professoral de l'École Supérieure d'Informatique Salama pour leur engagement inébranlable dans notre formation.

De manière particulière, nous souhaitons témoigner notre gratitude à Monsieur NDETURUYE Yves, en sa qualité de Directeur de notre travail. Son précieux temps et son expertise ont grandement contribué à la réalisation de ce travail.

À nos très chers parents, MWANZA Mathieu et KAYUMBA Anastasie, nous exprimons notre reconnaissance pour le soutien indéfectible sur les plans moral, matériel et spirituel dans les moments les plus exigeants.

À nos compagnons de lutte académique et d'amitié, avec lesquels nous avons partagé des moments de joie et de sacrifice considérables, mention spéciale à CHIBUMBU IRUNG Méschac, KAZADI KAYUMBA Alain, KALOMBO CIYA Peter, PUNGU WA MUPENDA Adalbert, MENDJE UPUKU Floribert et à l'ensemble de la promotion L4 Génie Logiciel.

À nos mentors qui ont éclairé notre chemin, nous disons merci pour leur précieux conseil, leur guidance et leur soutien constant. Mention spéciale à YOMBO Jonathan, AMPIRE Eric, TSHIOKUFILA Shekinah, BENI Roland. Vous êtes devenus bien plus que des mentors, vous êtes devenus des grands frères, des modèles et des amis.

Enfin, à tous nos frères et sœurs, tantes, oncles, neveux et nièces, amis et connaissances, dont les noms ne sont pas mentionnés ici, mais qui occupent une place spéciale dans nos cœurs, nous leur exprimons notre gratitude éternelle pour leur contribution à notre épanouissement personnel.

Que chacun de vous ressente la sincérité de notre reconnaissance et de notre gratitude, gravées dans notre cœur à jamais. Votre soutien et votre amour ont été les piliers de notre parcours, et pour cela, nous vous en sommes éternellement reconnaissants.

LISTE DES FIGURES

| | | |
|------|--|----|
| 1.1 | Carte d'une station de base LORAN | 7 |
| 1.2 | Principe du GPS dans un espace 1-D | 12 |
| 1.3 | Principe du GPS dans un espace 2-D | 13 |
| 1.4 | Caputre de l'application ICE GeoAlert | 14 |
| 1.5 | Caputre de l'application The Sorority | 16 |
| 2.1 | Aperçu général du processus DSDM[1] | 20 |
| 2.2 | Diagramme de cas d'utilisation | 23 |
| 2.3 | Diagramme de séquence S'authentifier | 29 |
| 2.4 | Diagramme de séquence Consulter contacts | 30 |
| 2.5 | Diagramme de séquence Ajouter contact | 31 |
| 2.6 | Diagramme de séquence Supprimer contact | 32 |
| 2.7 | Diagramme de séquence Modifier contact | 33 |
| 2.8 | Diagramme de séquence Activer tracking | 34 |
| 2.9 | Diagramme de séquence Désactiver tracking | 35 |
| 2.10 | Diagramme de séquence Envoyer alerte | 36 |
| 2.11 | Diagramme de séquence Consulter alertes | 37 |
| 2.12 | Diagramme de séquence Voir détails alertes | 38 |
| 2.13 | Diagramme de séquence Envoyer alerte | 39 |
| 2.14 | Diagramme d'activité administrateur | 40 |
| 2.15 | Modèle de domaine | 41 |
| 2.16 | Document utilisateur contenant des sous-documents Contact | 42 |
| 2.17 | Document Alert contenant des sous-documents AlertLocation | 43 |
| 3.1 | Présentation de l'architecture physique | 45 |
| 3.2 | Diagramme de déploiement | 47 |
| 3.3 | Communication des différentes composantes lors de l'envoi d'une alerte | 49 |
| 3.4 | Demande de permission et lancement du service TrackingService | 49 |
| 3.5 | Récupération de la localisation | 50 |
| 3.6 | Classification des entrées sonores | 51 |
| 3.7 | Détection d'une commande vocale sur les entrées audios | 52 |
| 3.8 | Envois d'alertes | 53 |
| 3.9 | Écran d'enregistrement du numéro de l'utilisateur | 54 |
| 3.10 | Écran de soumission du code de confirmation | 55 |

| | |
|--|----|
| 3.11 Écran d'enregistrement de l'identité de l'utilisateur | 56 |
| 3.12 Écran principal | 57 |
| 3.13 Écran permettant d'ajouter un contact | 58 |
| 3.14 Écran d'accueil, tracking activé | 59 |
| 3.15 Notification signalant que le tracking est en cours | 60 |
| 3.16 Écran principal, alertes en cours d'envoi | 61 |
| 3.17 SMS d'alerte | 62 |
| 3.18 Page de connexion administrateur | 63 |
| 3.19 Listes des alertes | 63 |
| 3.20 Détail d'une alerte | 64 |

LISTE DES TABLEAUX

| | | |
|-----|--|----|
| 2.1 | Classification des cas d'utilisation | 23 |
|-----|--|----|

LISTE DES ACRONYMES

DSDM : Dynamic Systems Development Method
GSM : Global System for Mobile communication
GPS: Global Positioning System
SMS: Short Message Service
LORAN: Long Range Navigation
UMTS : Universal Mobile Telecommunications Service
EOTD : Enhanced Observed Time Difference
Cell ID : Cellule IDentify
Wifi : Wireless Fidelity
GLONASS: Globalnaya Navigatsionnaya Sputnikovaya Sistema
FDMA : Frequency-division multiple access
GNSS : Global navigation satellite system
UML : Unified Modeling Language
UP : Unified Process
XP : Extreme programming
NoSQL : Not Only SQL
JSON : JavaScript Object Notation

TABLE DES MATIÈRES

| | |
|---|----------|
| INTRODUCTION GÉNÉRALE | 1 |
| 0.1 Problématique | 1 |
| 0.2 Hypothèse | 2 |
| 0.3 Choix et Intérêt du sujet | 2 |
| 0.3.1 Choix du sujet | 2 |
| 0.3.2 Intérêt du sujet | 2 |
| 0.3.2.1 Intérêt du personnel | 2 |
| 0.3.2.2 Intérêt scientifique | 2 |
| 0.4 Méthodes et techniques | 3 |
| 0.4.1 Méthodes | 3 |
| 0.4.2 Techniques | 3 |
| 0.4.3 État de l’art | 4 |
| 0.4.4 Délimitation du travail | 5 |
| 0.4.5 Subdivision du travail | 5 |
| 0.4.6 Outils logiciels et équipements utilisés | 5 |
| 1 GÉNÉRALITÉS ET ÉTUDE PRÉALABLE | 6 |
| 1.1 Introduction partielle | 6 |
| 1.2 Les moyens de localisation actuels à grande échelle | 6 |
| 1.2.1 Les systèmes de localisation par réseaux terrestres | 6 |
| 1.2.1.1 Le système LORAN C | 6 |
| 1.2.1.2 La localisation par réseau GSM | 8 |
| 1.2.1.3 La géolocalisation par Wi-Fi | 8 |
| 1.2.1.4 Limites des systèmes de localisation par réseaux terrestres | 8 |
| 1.2.2 Les systèmes de localisation satellitaires | 9 |
| 1.2.2.1 La localisation par GPS | 9 |
| 1.2.2.2 GLONASS | 10 |
| 1.2.2.3 Galileo | 10 |
| 1.3 Localisation d’un terminal par GPS | 11 |
| 1.3.1 La trilatération | 12 |
| 1.4 ÉTUDE PRÉALABLE | 13 |
| 1.4.1 Étude de l’existant | 14 |

| | | |
|----------|--|-----------|
| 1.4.1.1 | ICE GeoAlert | 14 |
| 1.4.1.2 | The Sorority | 16 |
| 1.4.2 | Critique de l'existant | 17 |
| 1.4.3 | Piste de solution | 17 |
| 1.5 | Conclusion partielle | 18 |
| 2 | CONCEPTION ET MODÉLISATION | 19 |
| 2.1 | Introduction Partielle | 19 |
| 2.1.1 | Présentation du langage UML | 19 |
| 2.1.2 | Présentation de la méthode DSDM | 19 |
| 2.1.3 | Utilisation de l'orienté objet avec le DSDM | 20 |
| 2.1.4 | Évaluation des risques | 21 |
| 2.2 | Besoins | 21 |
| 2.2.1 | Besoins fonctionnels | 21 |
| 2.2.2 | Besoins non fonctionnels | 22 |
| 2.3 | Identification des acteurs | 22 |
| 2.4 | Diagrammes | 22 |
| 2.4.1 | Diagramme de cas d'utilisation | 22 |
| 2.4.1.1 | Classification des cas d'utilisation | 23 |
| 2.4.1.2 | Descriptions textuelles des cas d'utilisation | 23 |
| 2.4.2 | Diagrammes de séquence | 28 |
| 2.4.2.1 | S'authentifier | 29 |
| 2.4.2.2 | Consulter contacts | 30 |
| 2.4.2.3 | Ajouter contact | 31 |
| 2.4.2.4 | Supprimer contact | 32 |
| 2.4.2.5 | Modifier contact | 33 |
| 2.4.2.6 | Activer tracking | 34 |
| 2.4.2.7 | Désactiver tracking | 35 |
| 2.4.2.8 | Envoyer alerte | 36 |
| 2.4.2.9 | Consulter alertes | 37 |
| 2.4.2.10 | Voir détails alertes | 38 |
| 2.4.3 | Diagramme d'activité | 38 |
| 2.4.3.1 | Activité utilisateur | 39 |
| 2.4.3.2 | Activité administrateur | 40 |
| 2.4.3.3 | Modèle de domaine | 41 |
| 2.5 | Modélisation de la base de données NoSQL orientée document | 41 |
| 2.5.1 | Règles de mappage pour une base de données orientée document | 42 |

| | | |
|----------|--|-----------|
| 2.5.1.1 | Relation par référence | 42 |
| 2.5.1.2 | Documents imbriqués | 42 |
| 2.5.2 | Modèle non relationnel | 42 |
| 2.6 | Conclusion partielle | 43 |
| 3 | PROCESSUS ET IMPLÉMENTATION | 44 |
| 3.1 | Introduction | 44 |
| 3.1.1 | Présentation de l'architecture physique | 44 |
| 3.2 | Technologies utilisées | 45 |
| 3.2.1 | Kotlin | 45 |
| 3.2.2 | TensorFlow lite | 45 |
| 3.2.3 | Mediapipe | 45 |
| 3.2.4 | Yamnet | 46 |
| 3.2.5 | Firestore | 46 |
| 3.2.6 | Angular | 46 |
| 3.2.7 | Android Studio | 46 |
| 3.3 | Diagramme de déploiement | 47 |
| 3.4 | Expérience utilisateur et fonctionnement | 47 |
| 3.4.1 | Expérience utilisateur | 47 |
| 3.4.2 | Fonctionnement | 48 |
| 3.5 | Captures d'écrans | 54 |
| 3.5.1 | Client mobile pour utilisateur | 54 |
| 3.6 | Difficultés rencontrées | 64 |
| 3.7 | Conclusion partielle | 65 |
| | CONCLUSION GÉNÉRALE | 66 |

AVANT-PROPOS

Dans un monde en constante évolution, la sécurité personnelle reste une préoccupation cruciale pour les individus. L'essor de la technologie a ouvert de nouvelles perspectives pour répondre à cette préoccupation, offrant la possibilité de concevoir des systèmes d'appel au secours innovants et efficaces.

L'enlèvement est un crime grave qui suscite une inquiétude croissante dans de nombreuses régions du monde, y compris Lubumbashi, une ville en constante expansion. Face à cette réalité préoccupante, il est impératif de développer des solutions technologiques innovantes pour renforcer la sécurité personnelle et la réactivité des forces de l'ordre.

Ce travail de recherche abordera les étapes clés de la conception d'un tel système, en commençant par une analyse critique des solutions existantes, en passant par la proposition de notre propre solution, et en concluant par sa mise en œuvre.

Notre objectif ultime est de contribuer à la sécurité des citoyens en offrant un outil puissant et accessible qui pourrait potentiellement sauver des vies. Nous sommes conscients des défis techniques et des enjeux sociaux associés à un tel projet, mais c'est avec une conviction profonde que nous entreprenons cette mission. L'innovation technologique peut être un puissant catalyseur de changement, et nous sommes déterminés à explorer comment elle peut être mise au service de la sécurité des individus.

Au fil des prochaines pages, nous vous invitons à plonger dans notre exploration du développement d'une solution répondant à un besoin criant dans notre société contemporaine : la sécurité personnelle. Bien que cette étude soit circonscrite dans le temps à l'année académique actuelle, notre ambition dépasse ces limites. Nous aspirons à contribuer à la création d'un environnement plus sûr pour nos concitoyens, en proposant une solution qui, nous l'espérons, pourra servir de modèle et inspirer d'autres initiatives de sécurité dans d'autres contextes.

INTRODUCTION GÉNÉRALE

L'importance de la sécurité est un enjeu majeur dans notre société contemporaine. Les défis sécuritaires sont nombreux et complexes, des problèmes tels que les enlèvements, la violence urbaine et la criminalité organisée continuent de menacer la tranquillité et la stabilité des communautés à travers le monde.

Face à ces défis, il devient essentiel de rechercher et de mettre en œuvre des solutions innovantes pour garantir la sécurité et protéger les individus contre ces menaces. De nouvelles technologies, des approches préventives et des méthodes d'intervention efficaces peuvent jouer un rôle crucial dans la création d'un environnement plus sûr. Dans le cadre de notre travail de fin d'études en Génie Logiciel, nous nous penchons sur la conception d'une solution qui contribuera à la lutte contre ce fléau en permettant à une victime d'enlèvement d'alerter ses proches et les services de sécurité, tout en laissant des traces qui faciliteront sa localisation.

0.1 Problématique

La ville de Lubumbashi fait face à une situation sécuritaire de plus en plus préoccupante, marquée par une série d'enlèvements. En février 2023, un article publié par 7sur7.cd a révélé la découverte de huit cadavres dans divers quartiers de la ville en l'espace d'une semaine¹. Un autre article de Radio Okapi publié 19 mai de cette même année met également en évidence la résurgence de la criminalité dans la ville de Lubumbashi depuis le début de l'année². La Commission Justice et Paix de l'archidiocèse de Lubumbashi a dénoncé publiquement cette situation préoccupante, citant des cas de corps sans vie retrouvés et l'enlèvement récent d'une religieuse. Ces cas ne représentent malheureusement qu'un exemple parmi tant d'autres.

Les enlèvements continuent de représenter une menace constante pour la sécurité des habitants de Lubumbashi, laissant place à de sérieuses inquiétudes concernant la stabilité et la quiétude de la ville. Cette situation soulève des questions cruciales qui nécessitent une attention approfondie dans le cadre de notre travail

- Comment peut-on développer des mécanismes discrets et efficaces pour permettre à une victime d'enlèvement d'alerter ses proches et les services de sécurité sans éveiller les soupçons du ravisseur ?

¹<https://7sur7.cd/2023/02/13/lubumbashi-8-corps-sans-vie-ramasses-en-une-semaine>

²<https://www.radiookapi.net/2023/05/19/actualite/securite/lubumbashi-la-commission-justice-et-paix-de-leglise-catholique-denonce>

- Comment peut-on aider les victimes d'enlèvement à laisser des traces ou des indices qui pourraient aider les forces de sécurité à les localiser ?

0.2 Hypothèse

Face au problème soulevé, nous pouvons répondre aux questions posées avec les solutions ci-après :

- Nous proposons de développer une application mobile permettant à une victime d'enlèvement de déclencher discrètement un appel à l'aide
- Notre système contiendra également un système de géolocalisation pour permettre de localiser la victime.

0.3 Choix et Intérêt du sujet

0.3.1 Choix du sujet

Pour être formé à la science, il fallait produire un travail scientifique vérifié, accepté et justifié dans notre domaine de formation, un travail mis à la disposition de tous pour faire évoluer le domaine de notre science. Notre choix s'est porté sur la conception d'un système d'appel au secours grâce à la reconnaissance vocale appliquée aux cas d'enlèvement afin de permettre aux personnes qui sont victimes de kidnapping de lancer un appel à l'aide afin d'être secourues.

0.3.2 Intérêt du sujet

0.3.2.1 Intérêt du personnel

Nous avons choisi un cas concret afin de matérialiser les connaissances acquises tout au long de notre parcours académique. Mais, également mettre à la disposition des personnes en danger un outil qui leur permettra d'alerter les proches et la sécurité pour ainsi être secourues

0.3.2.2 Intérêt scientifique

Nous avons choisi ce sujet dans le but de satisfaire aux exigences académiques imposées à tous les étudiants qui doivent réaliser un travail de fin d'études. Cependant, l'obtention du diplôme n'est pas le seul objectif de ce travail, car nous souhaitons également qu'il serve de modèle fiable et utile pour les chercheurs qui viendront après nous.

0.4 Méthodes et techniques

0.4.1 Méthodes

Pour concrétiser ce travail, nous avons décidé d'utiliser le Dynamic Systems Development Method (DSDM). Le DSDM est une méthode agile qui met l'accent sur la livraison rapide de fonctionnalités et la collaboration étroite avec les parties prenantes.

Le DSDM se compose de plusieurs phases, dont voici les principales :

- Étude de faisabilité : Cette phase consiste à établir les bases du projet, à définir son périmètre et à évaluer sa faisabilité. Elle permet de s'assurer que le projet est viable avant de s'engager pleinement.
- Étude business (ou analyse fonctionnelle) : Cette étape sert à la définition des spécifications, les fonctionnalités principales sont identifiées et priorisées en fonction de leur valeur ajoutée pour les utilisateurs.
- Modèle fonctionnel itératif : Dans cette phase, les fonctionnalités du système sont développées de manière itérative. Des courtes itérations permettent de tester rapidement et de recevoir les retours des parties prenantes, ce qui permet d'ajuster et d'améliorer le système au fil du temps.
- Mise en œuvre : Cette phase consiste à finaliser et à déployer le système dans son environnement de production.

0.4.2 Techniques

Pour pouvoir comprendre le fonctionnement du système que nous souhaitons implémenter, nous allons recourir à ces deux techniques qui sont :

- La technique documentaire

Nous allons utiliser la technique documentaire pour approfondir nos connaissances sur les concepts fondamentaux de la sécurité, la prévention des enlèvements, ainsi que les protocoles existant pour prévenir ce problème. Nous avons consulté des études académiques, des rapports gouvernementaux, des manuels de sécurité et d'autres ressources pertinentes pour nous aider à concevoir un système solide et efficace.

- La technique d'interview

Nous opterons pour la technique d'interview afin de comprendre pleinement les problèmes et les défis auxquels les individus sont confrontés en matière de sécurité contre l'enlèvement. Nous mènerons des entretiens avec des experts en sécurité,

des responsables des forces de l'ordre, des professionnels spécialisés dans la protection des personnes et des individus ayant été victimes d'enlèvement. Grâce à ces entretiens, nous pourrions recueillir des informations essentielles pour concevoir un système d'alerte efficace qui répondra aux préoccupations et aux besoins réels des personnes en matière de sécurité.

0.4.3 État de l'art

Au cours des dernières années, plusieurs travaux ont été entrepris afin de lutter contre les problèmes d'insécurité liés au cas d'enlèvement :

Premièrement, nous avons consulté le travail intitulé "Implémentation D'un Système De Prévention Sur L'enlèvement En Taxi" défendu à l'École Supérieure d'Informatique Salama, durant l'année académique 2020-2021 par Kalonga Mwamba Jonathan, en vue de l'obtention d'un grade d'ingénieur technicien informatique dans la filière Génie Logiciel, Système Informatique. Dans son travail, l'auteur a proposé une solution constituée d'une application mobile qui permettra à l'utilisateur de scanner la plaque du véhicule dans lequel il veut monter, de lancer une alerte qui enverra des informations sur la localisation et le taxi a des personnes prédéfinies, ainsi qu'à un tableau de bord géré par un administrateur qui pourra voir sur une carte la géolocalisation du téléphone.

Le deuxième travail est "Application de la GSM dans la géolocalisation en cas de détresse" également défendu à l'École Supérieure d'Informatique Salama durant l'année académique 2016-2017 par Tela Matondo Nathan. Dans ce travail, l'auteur propose une solution qui permettra à une victime de signaler sa position à ses proches sans passer par internet et sans attirer l'attention de ses ravisseurs. Pour utiliser cette solution, l'utilisateur doit pré-enregistrer une liste des personnes, puis lorsqu'il monte dans un taxi, il suffira qu'il lance l'application mobile et cette dernière enverra périodiquement les coordonnées géographiques du smartphone aux contacts pré-enregistrés.

Enfin, une recherche intitulée "Voice-Controlled Tool for Anytime Safety of Women" [2] publiée par Agrawal Agrima, Maurya Ankita et Patil Amruta. Ce travail vise à aider les femmes qui se sentent vulnérables sur le lieu du travail en raison de l'environnement et de l'absence de moyens de se défendre en cas de situation difficile.

Pour surmonter ces problèmes, un bracelet intelligent à commande vocale a été conçu pour les femmes. Ce dispositif est activé par la voix de la victime et il commence à suivre la localisation de la victime en utilisant le module GPS de l'appareil, chaque fois

que la commande vocale correspond, un message d'alerte est envoyé au numéro de contact pré-enregistré. Le GPS du téléphone mobile est utilisé pour localiser l'endroit le plus proche et envoyer un appel ou un SMS au poste de police le plus proche en mentionnant l'emplacement de la victime.

0.4.4 Délimitation du travail

Dans le souci et la quête d'un bon résultat, la restriction spatio-temporelle s'impose. Ainsi, nous limiterons notre champ de travail dans la province du Haut-Katanga et plus précisément dans la ville de Lubumbashi durant l'année académique 2022-2023.

0.4.5 Subdivision du travail

En plus de l'introduction générale et de la conclusion générale, notre travail sera divisé en 3 chapitres :

- Chapitre 1 : Généralités et étude préalable : Ce chapitre traitera des concepts de base qui nous permettront de nous mettre d'accord sur les grandes lignes de notre travail.
- Chapitre 2 : Conception Et Modélisation: Cela évoquera des fondements importants qui nous permettront de voir de près les différentes facettes de la solution que nous essayons de proposer.
- Chapitre 3 : Processus Et Implémentation : Chapitre présentera les étapes pratiques à suivre pour arriver à notre solution.

0.4.6 Outils logiciels et équipements utilisés

- Le langage de programmation Kotlin
- Le framework Angular, base sur Typescript
- Les IDEs IntelliJ IDEA et Android Studio
- Git et GitHub pour le versionnage de fichiers
- PlantUML et Draw.io
- Latex

CHAP I : GÉNÉRALITÉS ET ÉTUDE PRÉALABLE

1.1 Introduction partielle

La localisation, le repérage des pairs ou la navigation ont toujours posé des défis. À différentes époques, les solutions de localisation ont tiré parti des technologies de pointe disponibles: détermination de la position des étoiles [3], utilisation de phares maritimes (depuis l'Antiquité jusqu'à nos jours) et aéronautiques depuis les années 1920 [4]. Bien que la navigation ait longtemps été basée sur la référence à des points connus (villes, étoiles, phares), elle a progressivement cherché à améliorer sa précision. Deux approches ont émergé: la triangulation, où l'utilisateur mesure les angles par rapport aux points de référence pour déduire sa position (notamment grâce au sextant au XVIIIe siècle), et la navigation à l'estime, où l'utilisateur part de sa position initiale puis suit ses mouvements en utilisant des instruments de mesure tels que des compas, une méthode employée depuis le XVe siècle et popularisée par Christophe Colomb lors de sa découverte des Amériques [5]. Ce n'est qu'à la fin du XXe siècle que les systèmes de trilatération (d'abord terrestres, puis par satellites avec le GPS) ont pris le dessus en raison de leur précision et de leur couverture [6].

1.2 Les moyens de localisation actuels à grande échelle

1.2.1 Les systèmes de localisation par réseaux terrestres

Aujourd'hui, de nombreux réseaux cellulaires ou sans fil existent. Ces réseaux communiquent avec les équipements mobiles par radio. Le premier système de ce type est le système LORAN C dans les années 1960. Ensuite, les réseaux comme le GSM et l'UMTS ou le réseau de TNT (Télédiffusion Numérique Terrestre) sont apparus pour relayer le système LORAN C [7].

1.2.1.1 Le système LORAN C

LORAN est un acronyme de Long Range Navigation, l'évolution du concept original de LORAN pour répondre aux exigences opérationnelles pour une plus grande précision et une plus grande portée de service a donné lieu au développement de trois systèmes connexes : LORAN A, LORAN B et LORAN C.

Loran C est basé sur la mesure de la différence de temps d'arrivée des ondes provenant de deux stations terrestres différentes. Le lieu géométrique représenté par cette différence de temps est une hyperbole. Pour fonctionner, il est nécessaire de

connaître la position de chacune des stations terrestres et les stations terrestres doivent être synchronisées entre elles afin de comparer les instants d'arrivée des différents signaux [7].

Dans le cadre de LORAN, il y a une des stations qui sert de référence à toutes (appelée station maître), les autres stations sont dites esclaves.

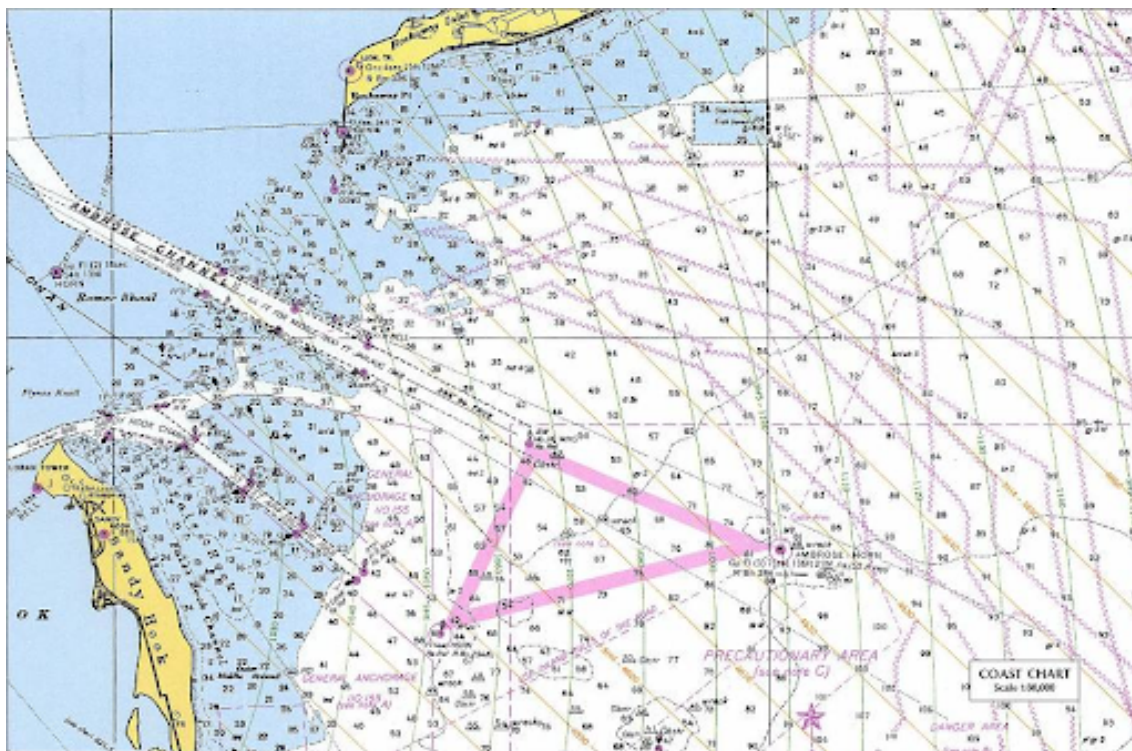


Figure 1.1: Carte d'une station de base LORAN

Le processus de localisation se déroule de la manière suivante : le dispositif mobile calcule la variation de temps entre l'arrivée des signaux de la station maître et d'une station esclave, ce qui génère une ligne de position. Une deuxième ligne de position est obtenue en mesurant une autre variation de temps entre la station principale et une deuxième station secondaire. La convergence de ces deux approches aboutit à la détermination de la position du dispositif mobile [7].

La navigation s'appuie sur des cartes préétablies. Le récepteur LORAN n'effectue pas de calculs de position par lui-même. Les cartes marines, telles que celle illustrée à la Figure 1.1, représentent les différents décalages observables. L'intersection des diverses ligne de position donne lieu à une approximation de la position du navire. Dans les récepteurs modernes, les valeurs affichées ne sont plus les variations de temps, mais plutôt les coordonnées de latitude et de longitude correspondantes. Les précisions données pour ce système de navigation sont de l'ordre de 500 m. Pour des applications de navigation maritime, cette précision était suffisante avant l'arrivée du GPS [7].

1.2.1.2 La localisation par réseau GSM

La localisation par GSM consiste à positionner un terminal GSM en se basant sur des informations liées aux antennes GSM auxquelles le terminal est connecté. La précision de la géolocalisation par GSM varie généralement de 200 mètres à plusieurs kilomètres, en fonction de la localisation du terminal, qu'il soit en milieu urbain ou rural.

Les trois technologies de géolocalisation distinctes utilisées par le GSM sont :

- Le différentiel temps dit EOTD (Enhanced Observed Time Difference) : Le téléphone mobile émet un signal vers les stations mobiles environnantes, celle qui est la plus proche lui renvoie ce signal. Le temps écoulé entre l'émission et la réception de cette onde est analysée par un serveur externe qui calculera la localisation du téléphone portable dans le réseau.
- Le système de l'identification de cellule ou Cell ID (Cellule IDentify) : Le système de l'identification par cellule est la technique de géolocalisation la plus simple et la moins coûteuse. Lorsque l'utilisateur se trouve dans une zone couverte par le réseau, il est localisé grâce à l'identification de la cellule à laquelle appartient l'antenne et par laquelle la communication est transmise.
- La triangulation : Le système de la triangulation repose sur le traitement croisé des informations provenant en permanence de trois relais émetteur et récepteur qui changent au fur et à mesure que l'antenne hertzienne utilisée par le portable de l'utilisateur se déplace.

1.2.1.3 La géolocalisation par Wi-Fi

La géolocalisation par Wi-Fi constitue une solution adaptée tant pour le positionnement à l'intérieur que l'extérieur des bâtiments. De manière similaire à la méthode Cell ID utilisée pour la géolocalisation sur un réseau GSM, un terminal Wi-Fi peut se localiser en se basant sur les identifiants de bornes Wi-Fi (adresses MAC) qu'il détecte. Des bases de données répertoriant de nombreuses bornes Wi-Fi ainsi que leurs positions géographiques existent. Ces bases de données peuvent être la propriété d'entreprises privées ou être mises à disposition gratuitement par des sociétés.

1.2.1.4 Limites des systèmes de localisation par réseaux terrestres

Ces systèmes terrestres possèdent une portée limitée, dans la mesure où ils supposent une surface de localisation plane, ce qui n'est pas le cas de la Terre. De plus, le signal envoyé est affaibli dans l'air, et peut même être dégradé par des obstacles ou par les conditions météorologiques. Malgré tout, ces systèmes sont toujours opérationnels,

pour des applications précises comme l'aéronautique, ou en complément des systèmes satellitaires [7].

1.2.2 Les systèmes de localisation satellitaires

L'émergence d'un système mondial de localisation s'est manifestée avec les avancées dans la conquête spatiale dans les années 1960. Les États-Unis ont introduit le tout premier système de positionnement par satellite en 1964, baptisé TRANSIT [8]. Ce système utilisait l'effet Doppler subi par le signal émis, lequel renfermait des informations précises sur la position du satellite émetteur, appelée éphéméride. Cette méthode permettait de se localiser dans le cadre du référentiel géodésique terrestre. Répondant aux exigences militaires, les États-Unis ont entrepris dans les années 1970 le développement d'une solution plus précise.

En effet, le système TRANSIT était entravé par les limitations imposées par la propagation du signal satellite dans l'atmosphère, incluant les affaiblissements dus à la troposphère et la dispersion des fréquences dans l'ionosphère. Ces facteurs ne permettaient qu'une précision de l'ordre de quelques centaines de mètres.

1.2.2.1 La localisation par GPS

Le GPS représente le système de localisation le plus répandu parmi les systèmes mondiaux de positionnement par satellites. Il est entièrement opérationnel et accessible partout. En effet, grâce au GPS, il est possible d'obtenir instantanément la localisation précise de n'importe quel endroit sur la planète.

Le système GPS englobe une constellation composée d'au moins 24 satellites tournant autour de la Terre suivant six orbites différentes. Chacune de ces orbites est inclinée à un angle de 55 degrés par rapport à l'équateur terrestre. Les orbites sont réparties de manière à être séparées de 60 degrés les unes des autres, garantissant ainsi une couverture de 260 degrés sur l'ensemble de la planète. Chaque orbite possède un rayon de 26 560 km. Ces orbites sont configurées de manière que les satellites effectuent une révolution complète autour de la Terre en l'espace d'une journée sidérale. Les quatre satellites se trouvant sur une orbite ne se trouvent pas à équidistance sur cette dernière. Deux des satellites sont espacés d'un angle variant entre 30,0 et 32,1 degrés. Les deux autres satellites forment un angle compris entre 92,28 et 130,98 degrés par rapport aux trois autres. Cette configuration a été mise en place afin de minimiser les effets en cas de dysfonctionnement d'un des satellites. Théoriquement, un récepteur GPS devrait être en mesure de capter entre 4 et 11 satellites depuis n'importe quel point de la surface terrestre [7].

1.2.2.2 GLONASS

GLONASS est un système russe de navigation par satellite fonctionnant dans le cadre d'un service de radionavigation par satellite. Il constitue une alternative au système de positionnement global et est le deuxième système de navigation en service avec une couverture mondiale et une précision comparable [9].

Le système GLONASS repose sur une constellation de 24 satellites en mouvement, répartis sur 3 plans orbitaux et situés à une altitude de 19 100 km. Le programme GLONASS a été initié en 1982 et a été déclaré opérationnel à part entière en 1993 par les autorités russes. Toutefois, en raison de contraintes financières rencontrées par l'Union Soviétique et de la relativement courte durée de vie des satellites (de 2 à 3 ans), la constellation a progressivement subi une détérioration. À l'heure actuelle, le système fonctionne en mode réduit avec seulement sept satellites opérationnels. Ce système trouve son application en géodésie, grâce à l'utilisation de mesures de phase.

L'intérêt de ce système de navigation réside en sa robustesse aux interférences. Chaque satellite émet sur sa propre fréquence (FDMA). Les satellites balaient une plus grande région du globe notamment les régions nord du fait des caractéristiques de la constellation de satellites et du plan d'inclinaison. Le principal défaut du système est qu'il n'est guère entretenu. L'entretien des satellites est très onéreux et du fait que les autorités russes manquent de moyens financiers, aujourd'hui seulement sept satellites sur les vingt quatre sont opérationnels [7].

1.2.2.3 Galileo

Galileo est destiné à être un GNSS civil de l'UE qui permet à tous les utilisateurs d'y accéder. À l'origine, le GPS réservait le signal de la plus haute qualité à un usage militaire, et le signal disponible pour un usage civil était intentionnellement dégradé (disponibilité sélective). Cette situation a changé lorsque le président Bill Clinton a signé une directive politique en 1996 pour désactiver la disponibilité sélective. Depuis mai 2000, le même signal de précision est fourni aux civils et aux militaires [9].

Sur le plan conceptuel, GALILEO ne différera pas de manière significative du GPS. Il reposera sur une constellation de 30 satellites, chacun ayant une masse d'environ 700 kg. Ces satellites seront agencés en trois plans orbitaux inclinés à un angle de 56 degrés par rapport à l'équateur, évoluant à une altitude de 23 500 km. Chaque satellite effectuera une rotation complète autour de la Terre en environ 14 heures. La gestion de la constellation sera assurée par un réseau global de stations terrestres [9].

1.3 Localisation d'un terminal par GPS

Le GPS est un système satellitaire qui est le plus populaire parmi les systèmes de positionnement mondial par satellites qui produit des informations en continu aux récepteurs GPS. Dans la vie quotidienne, les récepteurs GPS sont intégrés à de nombreux objets tels que les téléphones portables, les appareils photo et les automobiles. Ils sont également utilisés dans des dispositifs plus spécialisés tels que les robots d'exploration et les GPS de randonnée.

Il est constitué de trois parties distinctes nommées segments qui sont le segment spatial, le segment de contrôle et le segment utilisateur.

- *Le segment spatial*

Le segment spatial, composé des satellites, constitue le cœur du système GPS. Ces satellites forment une constellation en orbite autour de la Terre, évoluant à une altitude supérieure à 19 000 km. Cette altitude élevée permet une couverture étendue des signaux sur une large zone. Les orbites des satellites sont organisées de manière à ce qu'à tout moment, un récepteur GPS puisse capter les signaux d'au moins trois satellites simultanément.

Les satellites envoient des signaux radio à faible puissance et sur des fréquences différentes. Ces signaux sont capables de traverser des matériaux tels que le plastique, le verre et les nuages, mais ils sont bloqués par des objets denses ou solides tels que les bâtiments et les montagnes. Chaque satellite émet un code unique, ce qui permet aux récepteurs GPS d'identifier et de différencier les signaux émanant de chaque satellite.

- *Le segment de contrôle :*

Il assure la supervision des satellites GPS, en les suivant et en leur fournissant des ajustements au niveau du temps et des orbites. Ce segment comprend cinq stations de contrôle réparties dans le monde, chacune étant située à différents endroits autour de la Terre. Parmi ces cinq stations, quatre sont automatisées et sont chargées de la surveillance des satellites, tandis qu'une station principale est dédiée au contrôle global. Les stations automatiques reçoivent des données des satellites et transmettent ces informations à la station de contrôle principale. Cette dernière procède à la mise à jour et à la correction des données reçues, puis renvoie ces informations aux satellites à l'aide de deux antennes situées sur deux autres.

- *Le segment utilisateur*

Le segment utilisateur comprend les utilisateurs militaires et civils qui reçoivent et utilisent les informations émanant des satellites via un récepteur GPS. Ce segment englobe diverses catégories d'utilisateurs, notamment les forces de police, la gendarmerie, les militaires (armée de terre, marine, etc.) dans le domaine militaire. Dans le domaine civil, ce segment rassemble les pilotes, les navigateurs maritimes, les pêcheurs, les sportifs, les chasseurs, les conducteurs d'engins, les randonneurs, et bien d'autres.

1.3.1 La trilatération

Afin de fonctionner correctement, le récepteur GPS doit acquérir deux informations fondamentales : la distance entre lui et les satellites ainsi que les positions exactes de ces satellites. Après avoir capté les signaux émis par les satellites, le récepteur mesure le temps nécessaire à la propagation de ces ondes, ce qui lui permet de calculer la distance qui le sépare de chaque satellite. Cette opération est ensuite répétée pour les autres satellites en vue. Une fois que les distances entre le récepteur et les satellites sont connues, il devient possible de calculer et de déterminer la position grâce à une technique de triangulation. Pour obtenir une position en deux dimensions, trois satellites sont nécessaires, tandis que pour une position en trois dimensions, quatre satellites sont requis.

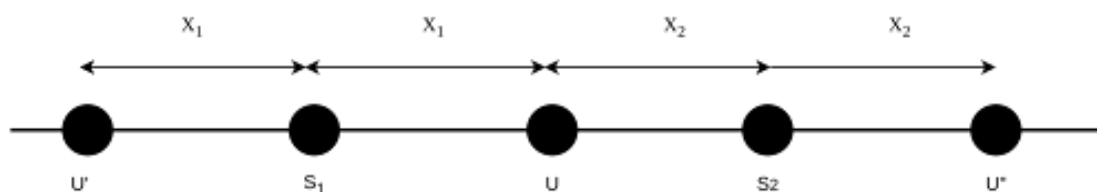


Figure 1.2: Principe du GPS dans un espace 1-D

Lorsqu'un équipement U (qu'il s'agisse d'un satellite ou d'un appareil mobile) effectue une mesure et évalue la distance qui le sépare du satellite S_1 , cela engendre deux positions potentielles pour U : soit à gauche de S_1 , soit à sa droite. Pour lever cette incertitude de position, l'utilisation d'un second satellite devient nécessaire. Dans ce scénario, c'est le satellite S_2 qui intervient pour résoudre cette ambiguïté. Lorsque S_2 évalue la distance entre lui et l'équipement U à x_2 , cela permet de déterminer la seule position plausible, comme illustrée par U sur la figure 1.2.

Le processus de détermination de la position d'un équipement dans un espace à deux ou trois dimensions suit une logique similaire à celle utilisée pour la localisation en une dimension. Dans cette configuration, l'obligation est d'avoir trois satellites à partir desquels trois distances sont mesurées. Le lieu géométrique associé à une distance spécifique par rapport à un point fixe est un cercle dans le cas bidimensionnel. La fig-

ure 1.3 démontre que deux satellites conduisent à l'estimation de deux positions dans l'espace. Cependant, l'introduction d'un troisième cercle est nécessaire pour déterminer la position unique de l'utilisateur.

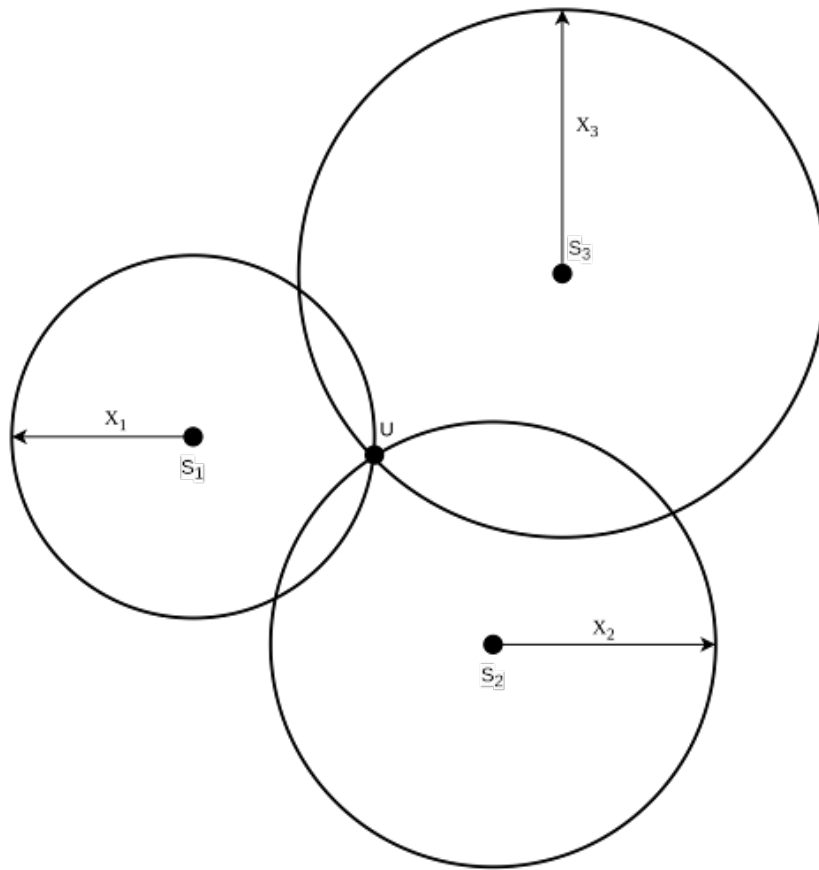


Figure 1.3: Principe du GPS dans un espace 2-D

De la même manière, on aboutit à la nécessité de disposer de quatre satellites pour parvenir à une localisation dans un espace à trois dimensions. Dans cet espace, le lieu géométrique correspondant est une sphère dont le rayon est la distance entre le satellite et l'utilisateur. On sait que l'intersection de deux sphères est un cercle. L'intersection de ce cercle avec une sphère génère deux points dans l'espace, et la troisième sphère détermine quelle position parmi les deux est occupée par l'équipement mobile.

1.4 ÉTUDE PRÉALABLE

Il existe aujourd'hui toute une panoplie des applications utilisant la géolocalisation et pouvant permettre à une personne en danger d'être localisé et de bénéficier d'une assistance. Dans cette partie, nous allons présenter ces applications, ensuite, nous allons établir une étude critique et enfin, nous allons proposer des pistes de solution qui nous permettront d'avoir plus de lumière avant de passer à la conception de notre solution.

1.4.1 Étude de l'existant

1.4.1.1 ICE GeoAlert



Figure 1.4: Caputre de l'application ICE GeoAlert

ICE GeoAlert¹ est une application qui permet aux utilisateurs d'alerter leurs familles. Pour envoyer un message d'alerte contenant sa géolocalisation à ses proches, la personne en danger doit avoir ICE GeoAlert installé au préalable, et envoie une alerte en cliquant sur le bouton rouge. L'alerte est reçue par message et par email.

- Avantages :
 - La possibilité d'envoyer rapidement sa géolocalisation aux proches en cas de danger.
 - Le déclenchement d'une alerte via un bouton dédié, ce qui peut être rapide et intuitif dans une situation d'urgence.

¹<https://icegeoalert.com>

- L’alerte est envoyée par message et par e-mail, ce qui garantit que les proches soient informés via plusieurs canaux.
- Inconvénients :
 - Dépendance de la connexion Internet pour envoyer les alertes, ce qui pourrait poser un problème dans des endroits avec une couverture réseau limitée.
 - Nécessité d’une action consciente de la part de la personne en danger pour déclencher l’alerte, ce qui pourrait ne pas être possible dans toutes les situations d’urgence.

1.4.1.2 The Sorority



Figure 1.5: Caputre de l'application The Sorority

L'application The Sorority² Mise sur l'entraide pour lutter contre le harcèlement de rue... en créant une communauté de femmes et personnes issues des minorités de genre, avec des utilisateurs volontaires et aux profils vérifiés.

En cas de danger, on peut afficher un message en grand sur l'écran de son smart-

²<https://www.jointhesorority.com/>

phone pour interpellier directement les personnes autour, déclencher une alarme sonore pour attirer l'attention et dissuader l'agresseur et également appeler les autorités via un raccourci.

Grâce à la géolocalisation, les cinquante utilisateurs les plus proches sont aussi alertés... L'application permet aussi de trouver un lieu sûr où se réfugier en cas de besoin/danger immédiat. L'application aide aussi à lutter contre les violences conjugales, intra-familiales et contre toutes les formes de harcèlement.

- Avantages :

- Encouragement à la solidarité et à l'entraide entre utilisatrices pour lutter contre le harcèlement et les agressions.
- Fonctionnalités telles que l'affichage de messages d'urgence, le déclenchement d'alarmes sonores et l'appel aux autorités, qui peuvent dissuader les agresseurs.
- Géolocalisation pour alerter les utilisateurs proches en cas de danger, ce qui peut permettre une réponse plus rapide.

- Inconvénients :

- Limitation à un groupe spécifique d'utilisateurs (femmes et minorités de genre), excluant potentiellement d'autres groupes susceptibles d'être en danger.
- Dépendance de la participation volontaire des utilisateurs, ce qui pourrait réduire la portée de l'application si le nombre d'utilisateurs n'est pas suffisamment élevé.

1.4.2 Critique de l'existant

Les applications étudiées visent à répondre à des situations d'urgence cruciales, mais elles présentent des défis significatifs. La dépendance à la connectivité Internet peut s'avérer problématique dans les régions où la couverture réseau est limitée. Une autre limitation importante réside dans la nécessité d'une action manuelle de la part de l'utilisateur en danger pour déclencher une alerte. Dans des scénarios de danger extrême, cette action peut être difficile, voire impossible à réaliser, réduisant ainsi l'utilité de ces applications dans des situations critiques.

De plus, les solutions examinées ont des portées limitées. Certaines applications sont restreintes à des groupes spécifiques d'utilisateurs, excluant potentiellement d'autres personnes en danger. Dans l'environnement spécifique de Lubumbashi, il est impératif de développer des solutions qui surmontent ces limites pour garantir une protection adéquate et une assistance rapide aux individus en danger.

1.4.3 Piste de solution

La conception d'un système d'appel au secours efficace en cas d'enlèvement à Lubumbashi requiert une approche réfléchie et innovante pour surmonter les limites identifiées dans les applications existantes, telles qu'ICE GeoAlert 2014 et The Sorority. Ce chapitre explore diverses pistes de solutions qui pourraient être envisagées pour répondre aux défis spécifiques de la région.

- **Alertes automatisées par Reconnaissance Sonore :** L'intégration d'un système de reconnaissance sonore sur les smartphones des utilisateurs pourrait permettre la détection automatique de signaux de danger, tels que cris ou bruits suspects. Cette fonctionnalité automatisée renforcerait la réactivité du système, déclenchant des alertes sans nécessiter d'action manuelle de la part de l'utilisateur.
- **Envoi d'Alertes par Différents Canaux :** Pour assurer que les alertes soient reçues rapidement, il serait bénéfique d'envisager l'envoi d'alertes par différents canaux. Cela inclut non seulement les réseaux mobiles GSM, mais aussi les applications de messagerie instantanée, les courriels et les notifications push.

1.5 Conclusion partielle

Dans ce chapitre, les principales méthodes de localisation des utilisateurs mobiles ont été présentées. On distingue les systèmes de localisation par réseaux terrestres tels que le LORAN C, le GSM ou encore le WIFI qui communiquent avec les équipements mobiles par radio. Ces systèmes qui, malgré leurs limites causées par les obstacles et les conditions météorologiques, sont toujours opérationnels, dans des domaines tels que l'aéronautique, ou en complément des systèmes satellitaires. Nous avons également présenté quelques différents systèmes de localisation satellitaire qui sont le GPS, GLONASS et Galileo. Enfin, nous avons présenté le fonctionnement de localisation d'un terminal mobile grâce aux systèmes satellitaire GPS qui est un système compatible avec la majorité des objets de notre vie quotidienne tels que les smartphones et les véhicules automobiles.

CHAP II : CONCEPTION ET MODÉLISATION

2.1 Introduction Partielle

Dans ce présent chapitre, nous allons procéder à la réalisation des diagrammes UML qui nous permettront de bien comprendre notre système, en plus des diagrammes, nous allons définir le processus de développement utilisé et du langage utilisé pour dégager lesdits diagrammes.

2.1.1 *Présentation du langage UML*

UML est un langage de modélisation très complet, qui couvre de nombreux aspects du développement des logiciels, comme les exigences, l'architecture, les structures et les comportements [10]. UML est un langage de modélisation visuelle à usage général pour les systèmes. Bien qu'UML soit le plus souvent associé à la modélisation de systèmes logiciels orientés objet, son application est beaucoup plus large que cela [11]. UML n'est pas lié à une méthodologie ou à un cycle de vie spécifique. Il peut d'ailleurs être utilisé avec toutes les méthodologies existantes. La méthode UP par exemple utilise UML comme syntaxe de modélisation visuelle sous-jacente. Elle est la méthode préférée pour UML, car elle est la mieux adaptée, mais UML fournit le support de modélisation visuelle pour d'autres méthodes [11].

2.1.2 *Présentation de la méthode DSDM*

La méthode DSDM, encore appelée Méthode de développement de systèmes dynamiques, est une méthode de gestion de projet de la catégorie des méthodes agiles développée en Grande-Bretagne à partir de 1994. La méthode DSDM fait référence à un processus organisé et fondé sur le bon sens qui donne la priorité à la mise en œuvre rapide et efficace de projets. À bien des égards, elle est comparable à XP et à Scrum, mais son principal avantage par rapport à ces deux méthodes est qu'il s'agit de la meilleure méthodologie pour laquelle les délais sont fixes. Au lieu de se concentrer sur les activités des équipes de développement, DSDM met l'accent sur la livraison de la solution. Cette méthodologie fait le nécessaire pour s'assurer que le sens commercial et la faisabilité de chaque projet ont été établis avant la conception et la mise en œuvre [12].

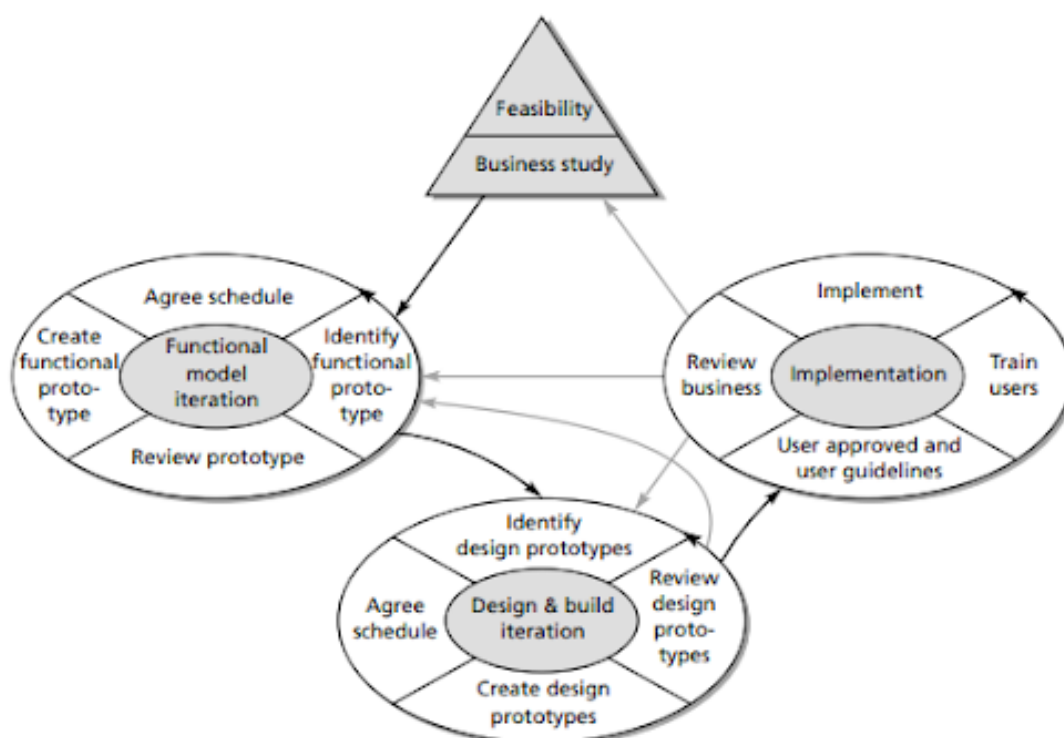


Figure 2.1: Aperçu général du processus DSDM[1]

2.1.3 Utilisation de l'orienté objet avec le DSDM

L'analyse orientée objet a commencé par la décomposition du système proposé en cas d'utilisation, qui, pour nos besoins, peuvent être considérés comme des processus métier que le système aidera à résoudre. Cela correspond bien au domaine d'activité de DSDM [1].

- Identifier les besoins métier qui devraient être pris en charge par le système proposé.
- Définir les besoins en information des processus métier qui seront pris en charge.
- Identifier les catégories d'utilisateurs impactés par le développement et l'introduction du système proposé.
- Identifier les processus métier et les scénarios métier qui doivent être modifiés.
- Clarifier toutes les interfaces avec d'autres systèmes (humains ou automatisés).

La conception orientée objet consiste essentiellement à sélectionner une architecture (généralement multiniveaux), les protocoles de communication entre les niveaux, une stratégie de gestion au sein des niveaux et les langages appropriés pour construire les niveaux. Encore une fois, cela correspond bien aux objectifs de définition de l'architecture système :

- Fournir une compréhension commune des architectures techniques à utiliser pendant le développement et la mise en œuvre.
- Décrire la plateforme cible et la plateforme de développement.
- Fournir une description générale de l'architecture logicielle (c'est-à-dire les principaux objets ou composants logiciels, à la fois processus et données, et leurs interactions).

La technologie orientée objet facilite le développement incrémental du code requis; les concepts d'orientation objet encouragent le développement d'opérations/méthodes de classe petites et réutilisables, qui peuvent être ajoutées au besoin même après que la classe a été déployée [1].

2.1.4 Évaluation des risques

Chaque fonctionnalité d'un logiciel est associée à un risque, qu'il s'agisse d'un risque lié à la disponibilité, à l'évolutivité ou à l'intégrité des données. L'analyse des risques liés à l'architecture est l'une des activités clés de la gestion des projets. En analysant continuellement les risques, l'architecte peut remédier aux lacunes d'un projet et prendre des mesures correctives pour atténuer les risques [13].

2.2 Besoins

La conception d'un système exige de connaître les besoins des utilisateurs concernés, voici ci-dessous les besoins que nous avons pu constater; et que nous utiliserons par la suite à des fins de modélisation, enfin de mieux comprendre le système de manière générale.

2.2.1 Besoins fonctionnels

1. L'utilisateur doit s'authentifier avant d'accéder à l'application
2. L'administrateur doit s'authentifier avant d'accéder à l'espace d'administration
3. L'utilisateur peut ajouter des contacts qui recevront les alertes
4. L'utilisateur peut consulter la liste de ses contacts.
5. L'utilisateur peut modifier les contacts qu'il a ajoutés
6. L'utilisateur peut supprimer des contacts parmi les contacts qu'il a ajoutés
7. L'utilisateur peut envoyer une alerte
8. L'administrateur peut consulter les alertes actives de tous les utilisateurs

9. L'administrateur peut retransmettre les alertes aux contacts de l'utilisateur ou a un centre de sécurité

2.2.2 Besoins non fonctionnels

1. l'application doit être intuitive et facile à manipuler
2. l'application doit être en mesure de fonctionner sur des périphériques non performant
3. les informations liées à la géolocalisation des utilisateurs doivent être le plus précises possible

2.3 Identification des acteurs

1. Utilisateur : Une personne qui peut être victime de kidnapping et qui aura besoin d'être assistée
2. Administrateur : chargé de la gestion du tableau de bord du système
3. Service de messagerie : un système logiciel externe qui permet d'envoyer des SMS

2.4 Diagrammes

2.4.1 Diagramme de cas d'utilisation

Dans le cadre de notre travail de fin de cycle (TFC), le diagramme de cas d'utilisation revêt une importance cruciale. Il nous sert avant tout à comprendre les besoins et les exigences fonctionnelles du système que nous analysons ou que nous sommes en train de concevoir. En identifiant les acteurs impliqués et en décrivant comment ils interagissent avec le système, ce diagramme nous permet d'établir une base solide pour la planification et la réalisation de notre solution.

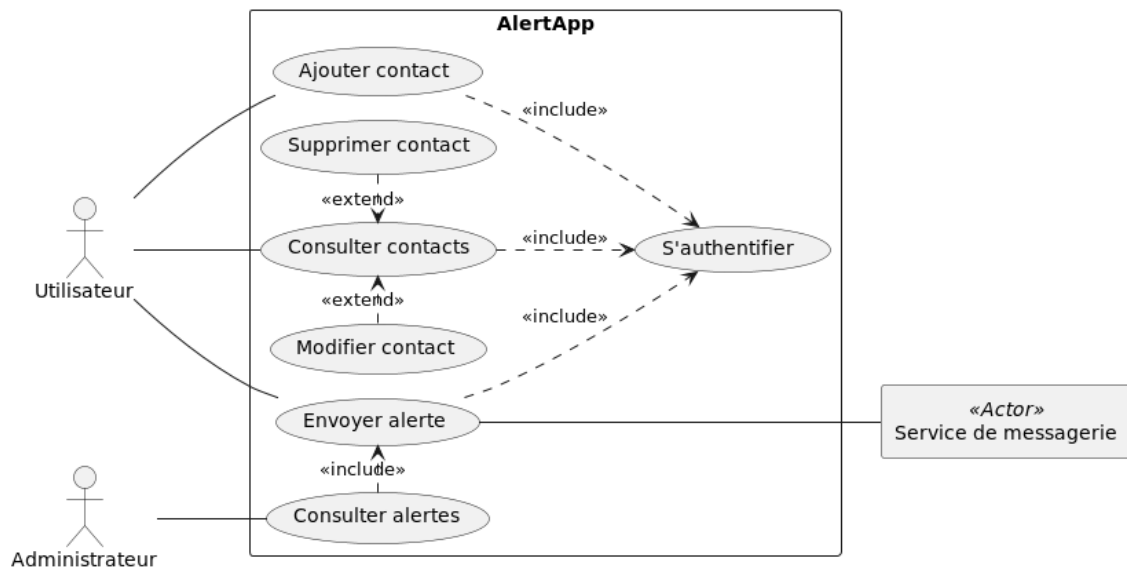


Figure 2.2: Diagramme de cas d'utilisation

2.4.1.1 Classification des cas d'utilisation

Table 2.1: Classification des cas d'utilisation

| Cas d'utilisation | Priorité | Risque |
|---------------------|----------|--------|
| S'authentifier | Élevée | Moyen |
| Consulter contacts | Élevée | Moyen |
| Ajouter contact | Élevée | Moyen |
| Modifier contact | Faible | Faible |
| Supprimer contact | Moyenne | Faible |
| Activer tracking | Élevée | Moyen |
| Désactiver tracking | Élevée | Faible |
| Envoyer alerte | Élevée | Élevée |
| Consulter alertes | Élevée | Faible |

2.4.1.2 Descriptions textuelles des cas d'utilisation

a. S'authentifier

- Objectif : permettre à l'utilisateur de fournir son identité au système
- Acteurs :
 - Acteur principal : Utilisateur,Administrateur
 - Acteur secondaire : -
- Précondition : -

- Scenario nominal :
 1. L'utilisateur soumet son numéro de téléphone
 2. Le système envoie un code de validation au numéro
 3. Le système affiche le formulaire de soumission du code de confirmation
 4. L'utilisateur soumet le code de confirmation
 5. Le système vérifie la validité du code de confirmation
 6. Le système affiche un formulaire d'insertion d'identité
 7. L'utilisateur soumet son identité
 8. Le système affiche un message de succès
- Scenario alternatif :
 - 6 .a. Le code de confirmation est invalide
 - 6 .a.1. Le système affiche un message d'erreur
 - 6 .a.2. On rentre au scenario 1
- Scenario d'exception :
 1. L'utilisateur quitte l'application avant la fin de l'authentification
- Postcondition : L'utilisateur est authentifié et peut accéder aux autres fonctionnalités

b. Consulter contacts

- Objectif : permettre à l'acteur de voir la liste de ses contacts
- Acteurs :
 - Acteur principal : Utilisateur
 - Acteur secondaire : -
- Précondition : S'authentifier
- Scenario nominal :
 1. L'acteur accède à la page des contacts
 2. Le programme affiche la liste de tous ses contacts
- Scenario alternatif : -
- Scenario d'exception : -
- Postcondition : -

c. Ajouter contact

- Objectif : permettre à l'acteur d'ajouter un contact
- Acteurs :

- Acteur principal : Utilisateur
- Acteur secondaire : -
- Précondition : -
- Scenario nominal :
 1. L'utilisateur accède à la page d'ajout de contact
 2. Le formulaire d'ajout est affiché
 3. L'utilisateur soumet le formulaire
 4. Le système vérifie que les informations sont valides
 5. Le système enregistre le contact
 6. Le système affiche un message de succès
- Scenario alternatif :
 - 5 .a. Les informations sont invalides
 - 5 .a.1. Le système affiche un message d'erreur
 - 5 .a.2. On rentre au scenario 2
- Scenario d'exception : -
- Postcondition : Un nouveau contact est ajouté

d. Modifier contact

- Objectif : permettre à l'utilisateur de modifier un contact
- Acteurs :
 - Acteur principal : Utilisateur
 - Acteur secondaire : -
- Précondition : Consulter contact
- Scenario nominal :
 1. L'utilisateur sélectionne un contact
 2. L'utilisateur sélectionne l'option de modification du contact
 3. Le système affiche le formulaire de modification du contact
 4. L'utilisateur modifie le contact
 5. L'utilisateur soumet le formulaire
 6. Le système vérifie les nouvelles informations
 7. Le système enregistre les modifications du contact
 8. Le système affiche un message de succès
- Scenario alternatif :
 - 7 .a. Les informations sont invalides

7 .a.1. Le système affiche un message d'erreur

7 .a.2. On rentre au scenario 3

- Scenario d'exception :

1. L'utilisateur n'a aucun contact

2. L'utilisateur quitte la page de modification du contact avant de l'avoir soumis

- Postcondition : Un contact a été modifié

e. Supprimer contact

- Objectif : permettre à l'utilisateur de supprimer un contact

- Acteurs :

- Acteur principal : Utilisateur

- Acteur secondaire : -

- Précondition : Consulter contact

- Scenario nominal :

1. L'utilisateur sélectionne un contact

2. L'utilisateur sélectionne l'option de suppression

3. Le système demande à l'utilisateur de confirmer la suppression

4. L'utilisateur confirme la suppression

5. Le contact est supprimé

6. Le système affiche un message de succès

- Scenario alternatif :

- 4 .a. L'utilisateur annule la suppression

- 4 .a.1. Le système affiche la liste des contacts

- Scenario d'exception :

1. L'utilisateur n'a aucun contact

- Postcondition : Un contact a été supprimé

f. Activer tracking

- Objectif : permettre à l'utilisateur d'activer l'écoute de la commande vocale

- Acteurs :

- Acteur principal : Utilisateur

- Acteur secondaire : -

- Précondition : S'authentifier

- Scenario nominal :
 1. L'utilisateur clique sur l'option d'activation de tracking
- Scenario alternatif : -
- Scenario d'exception : -
- Postcondition : Le système est en attente d'une commande vocale

g. Désactiver tracking

- Objectif : permettre à l'acteur d'annuler l'écoute de la commande vocale
- Acteurs :
 - Acteur principal : Utilisateur
 - Acteur secondaire : -
- Précondition : Activer tracking
- Scenario nominal :
 1. L'utilisateur clique sur l'option de désactivation du tracking
- Scenario alternatif : -
- Scenario d'exception : -
- Postcondition : L'écoute de la commande vocale est désactivée

h. Envoyer alerte

- Objectif : permettre à l'utilisateur d'envoyer une alerte à ses contacts
- Acteurs :
 - Acteur principal : Utilisateur
 - Acteur secondaire : Service de messagerie
- Précondition : Activer tracking
- Scenario nominal :
 1. L'utilisateur énonce une commande vocale
 2. Le système vérifie la validité de la commande
 3. Tant que le tracking est active, Le service de messagerie envoie une alerte aux contacts de l'utilisateur
- Scenario alternatif :
 - 1 .a. Le système détecte un son suspect
 - 1 .a.1. On passe au scenario 3
- Scenario d'exception :
 1. Le Service de messagerie ne fonctionne pas

2. Fausse alerte

- Postcondition : Alertes envoyées avec succès

i. Consulter alertes

- Objectif : permettre à l'administrateur de voir les alertes envoyées par les utilisateurs
- Acteurs :
 - Acteur principal : Administrateur
 - Acteur secondaire : -
- Précondition : S'authentifier
- Scenario nominal :
 1. L'administrateur demande la page des alertes
 2. Le système affiche la liste de toutes les alertes
- Scenario alternatif : -
- Scenario d'exception : -
- Postcondition : -

j. Voir détails alerte

- Objectif : permettre à l'administrateur de voir les détails d'une alerte envoyée par les utilisateurs
- Acteurs :
 - Acteur principal : Administrateur
 - Acteur secondaire : -
- Précondition : Consulter alertes
- Scenario nominal :
 1. L'administrateur demande la page des détails d'une alerte
 2. Le système affiche les détails de l'alerte
- Scenario alternatif : -
- Scenario d'exception : -
- Postcondition : -

2.4.2 Diagrammes de séquence

Dans le cadre de notre travail, le diagramme de séquence nous permet de modéliser les interactions entre les objets ou composants du système, en mettant en évidence la séquence des messages échangés au fil du temps. Ce diagramme est essentiel pour comprendre le comportement dynamique du système, ce qui facilite la conception ainsi que le développement.

2.4.2.1 S'authentifier

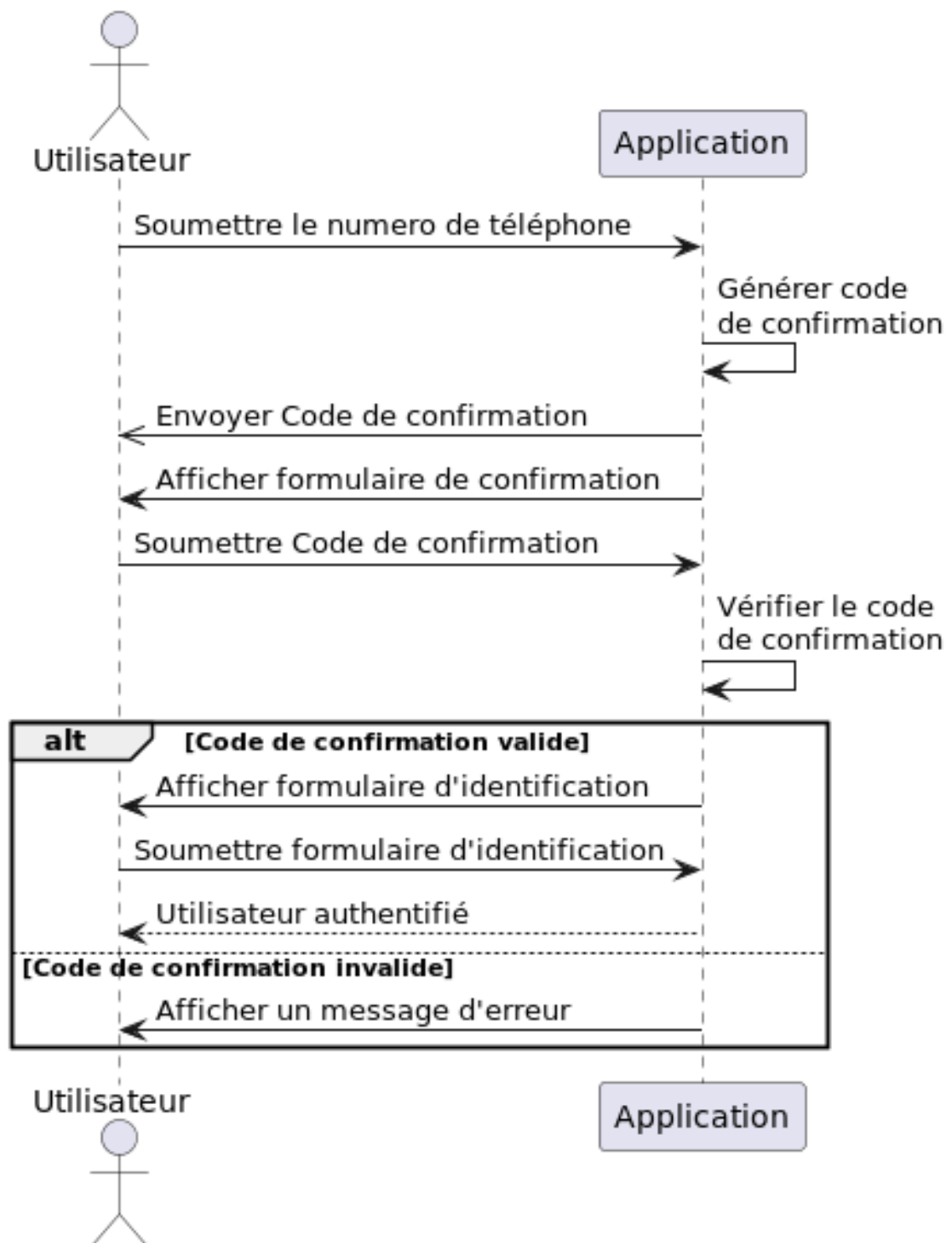


Figure 2.3: Diagramme de séquence S'authentifier

2.4.2.2 Consulter contacts

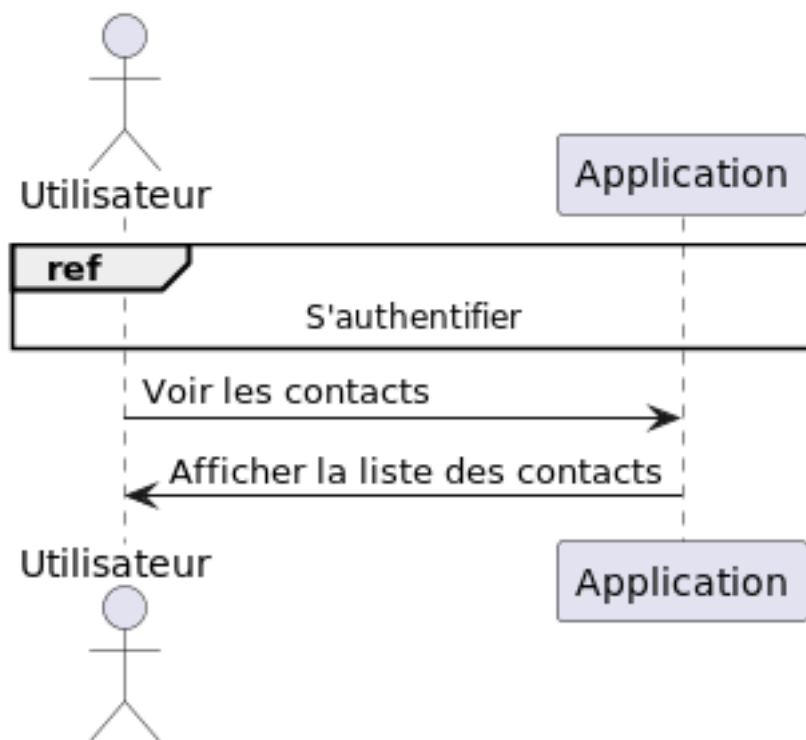


Figure 2.4: Diagramme de séquence Consulter contacts

2.4.2.3 Ajouter contact

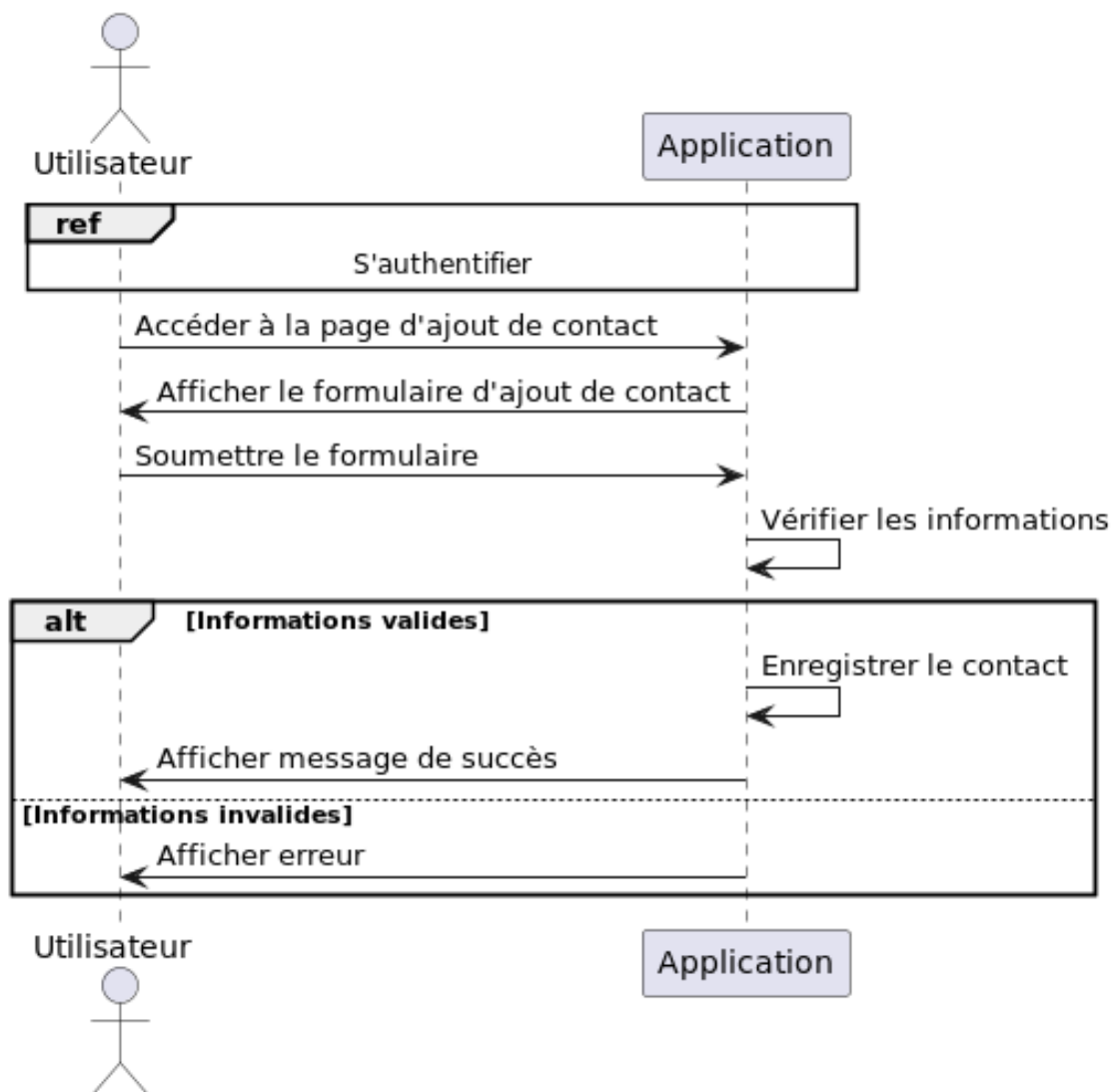


Figure 2.5: Diagramme de séquence Ajouter contact

2.4.2.4 Supprimer contact

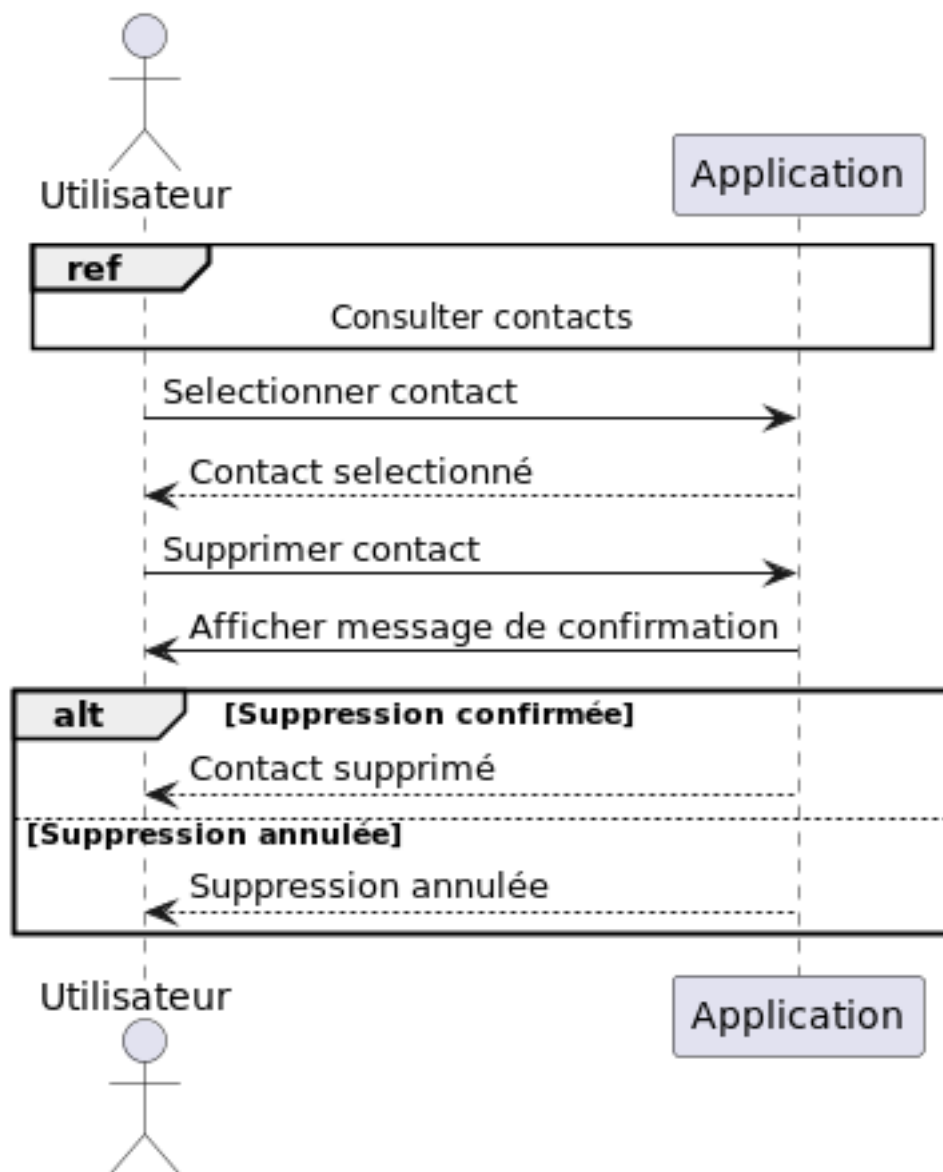


Figure 2.6: Diagramme de séquence Supprimer contact

2.4.2.5 Modifier contact

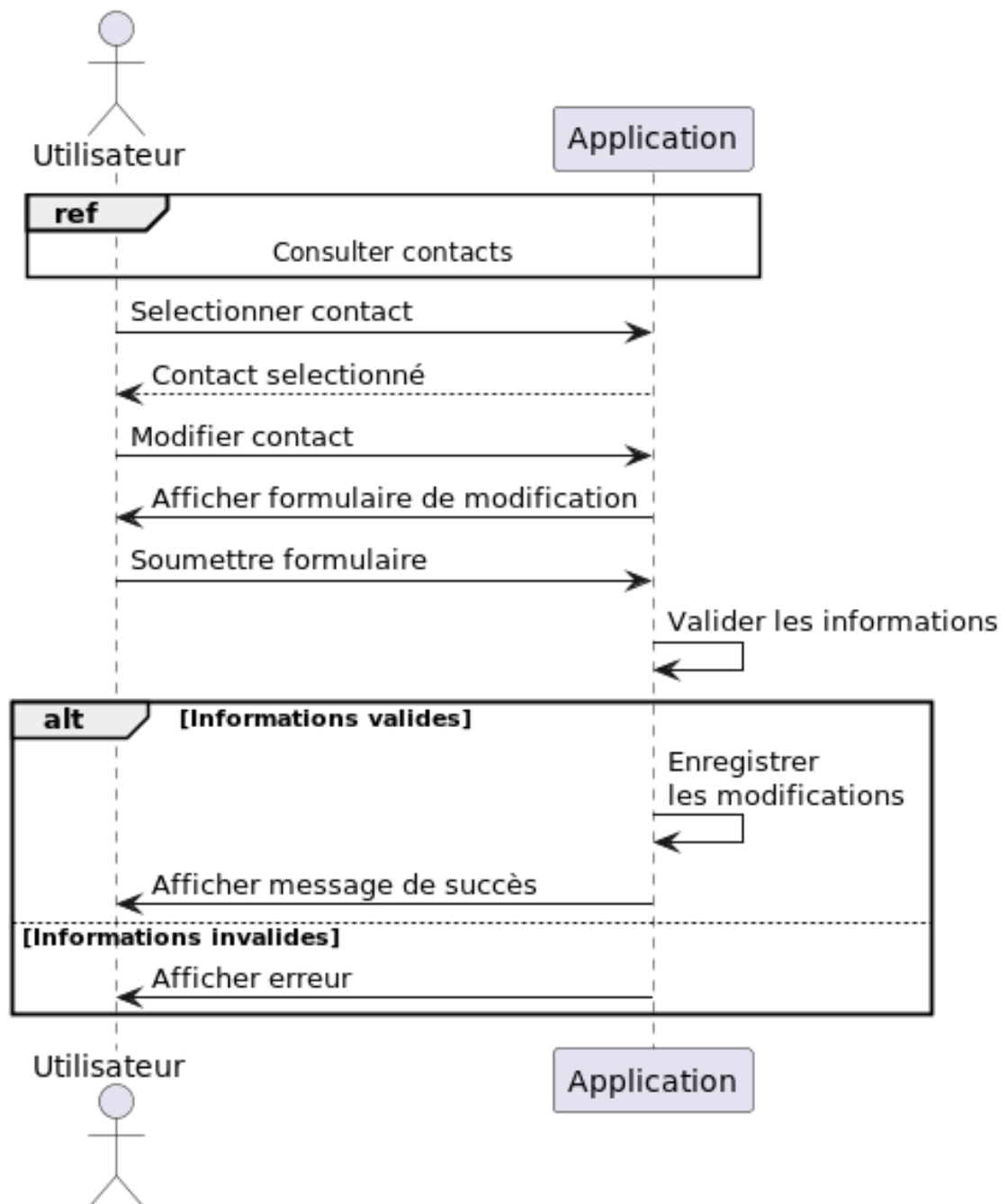


Figure 2.7: Diagramme de séquence Modifier contact

2.4.2.6 Activer tracking

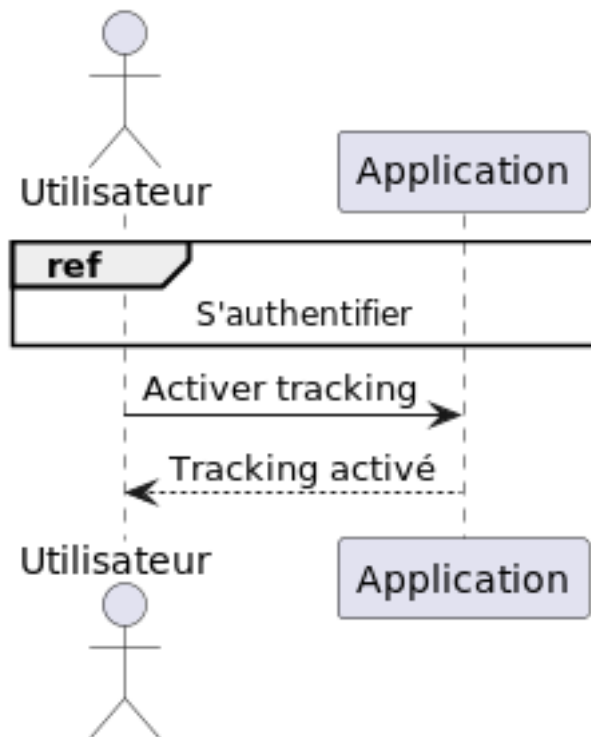


Figure 2.8: Diagramme de séquence Activer tracking

2.4.2.7 Désactiver tracking

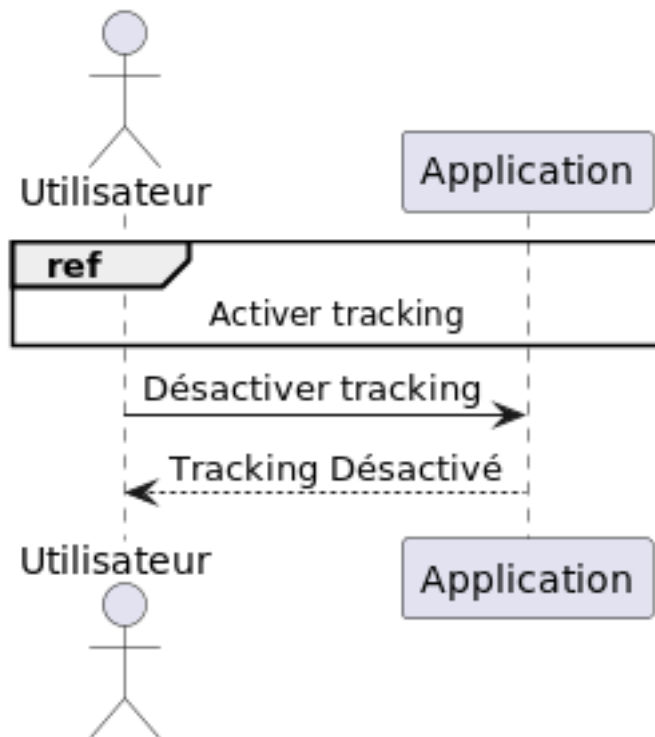


Figure 2.9: Diagramme de séquence Désactiver tracking

2.4.2.8 Envoyer alerte

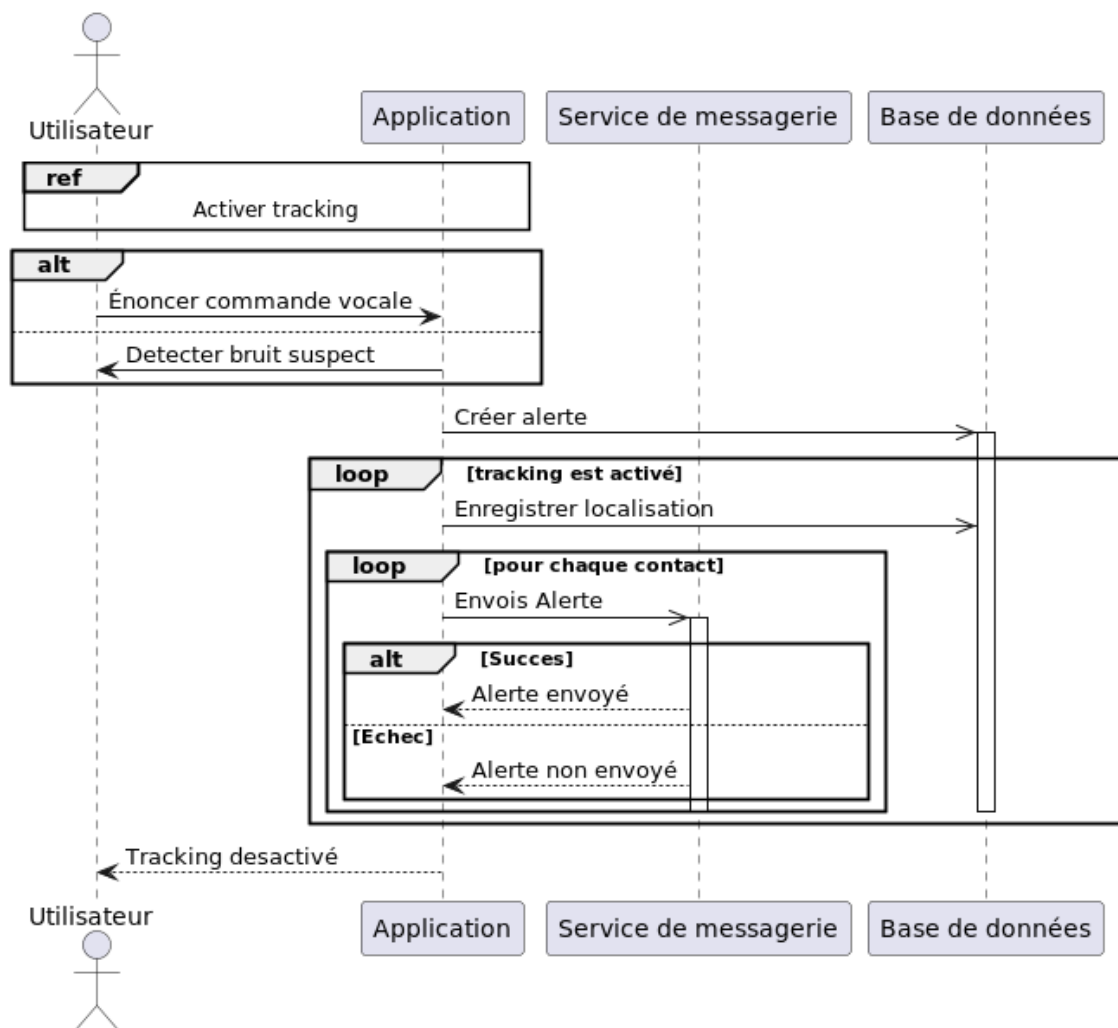


Figure 2.10: Diagramme de séquence Envoyer alerte

2.4.2.9 Consulter alertes

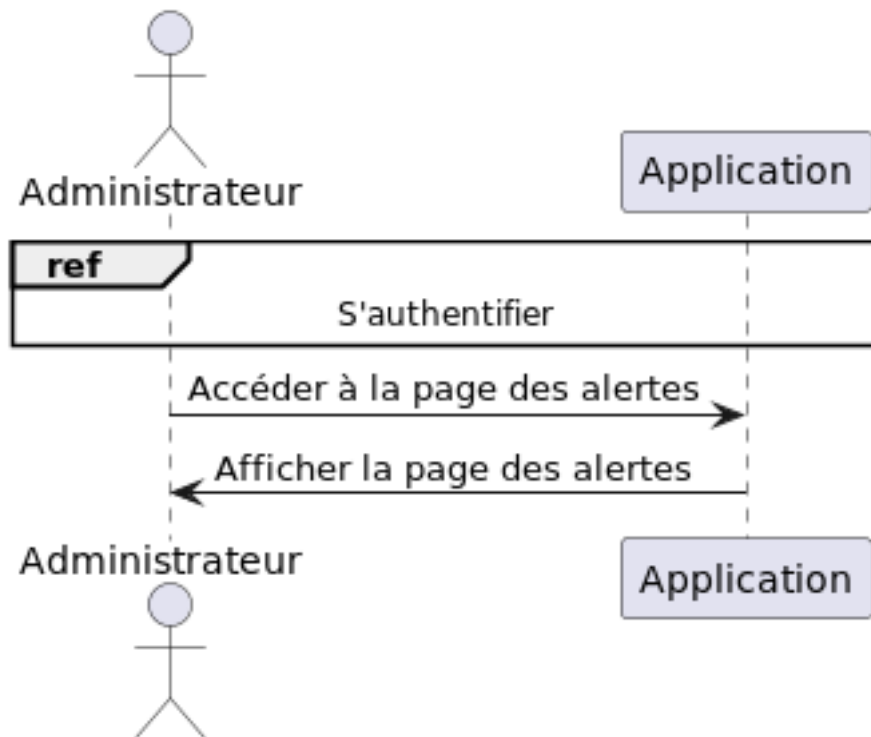


Figure 2.11: Diagramme de séquence Consulter alertes

2.4.2.10 Voir détails alertes

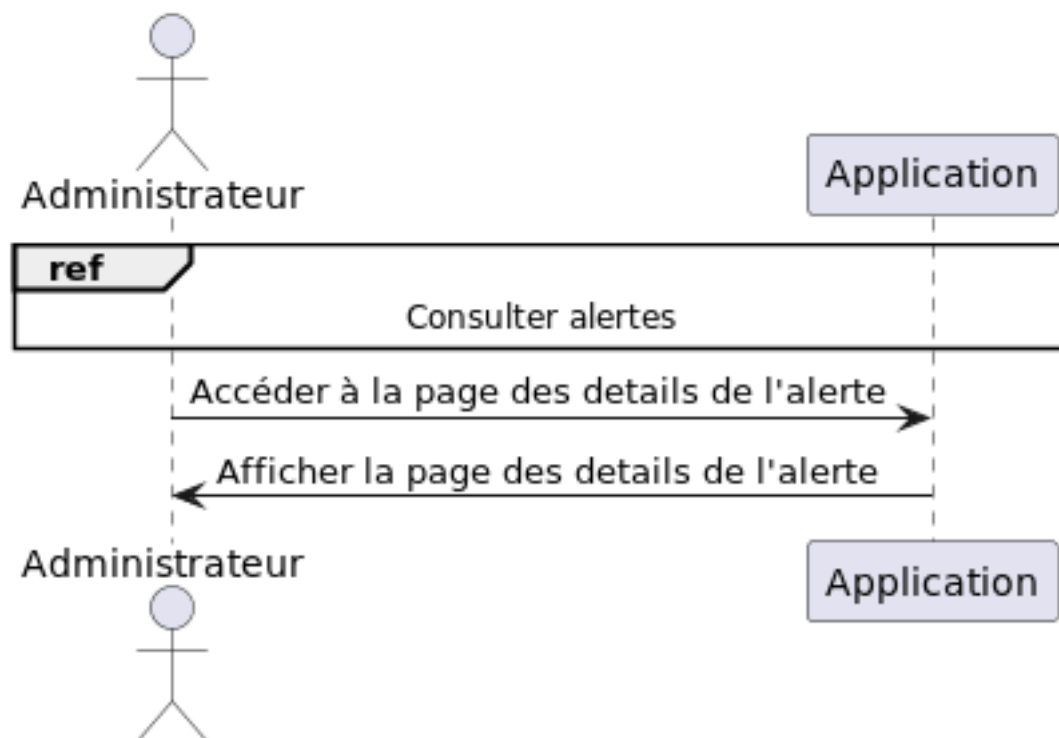


Figure 2.12: Diagramme de séquence Voir détails alertes

2.4.3 Diagramme d'activité

le diagramme d'activité quant à lui nous permet de modéliser visuellement les processus, les flux de travail, et les décisions au sein du système. En utilisant ce diagramme, nous pouvons mieux comprendre et représenter les séquences d'actions, les transitions, et les conditions qui gouvernent le comportement du système. Cela facilite la planification, la conception et l'optimisation des processus.

2.4.3.1 Activité utilisateur

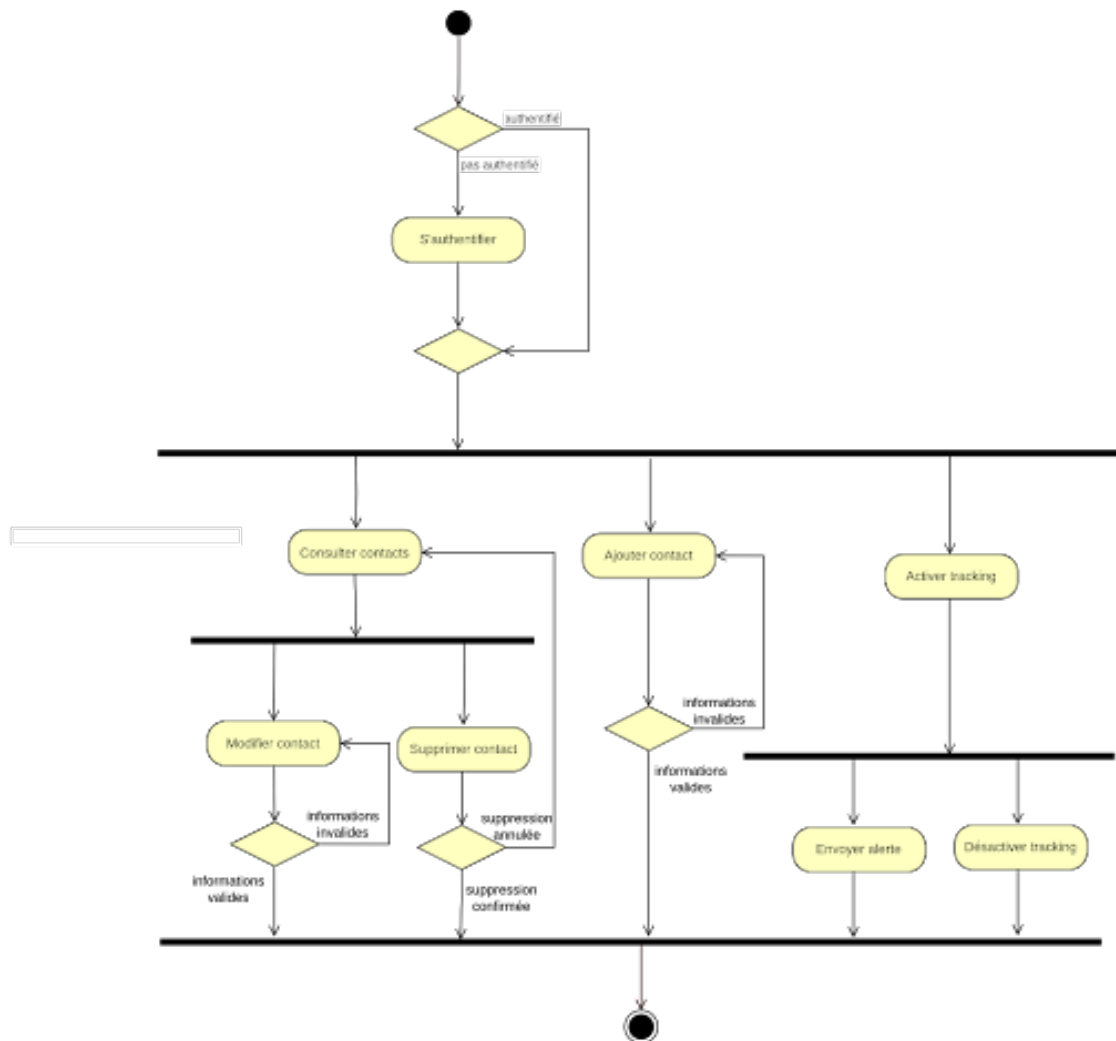


Figure 2.13: Diagramme de séquence Envoyer alerte

2.4.3.2 Activité administrateur

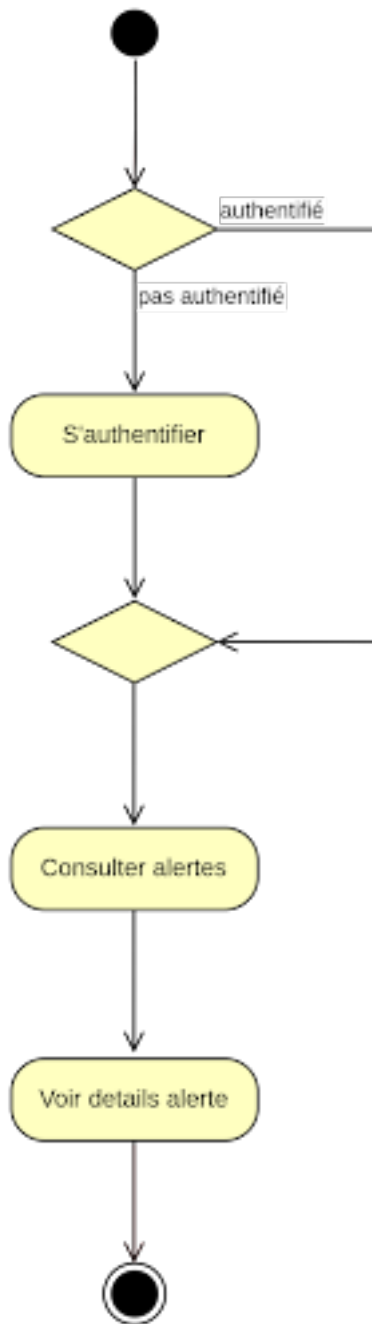


Figure 2.14: Diagramme d'activité administrateur

2.4.3.3 Modèle de domaine

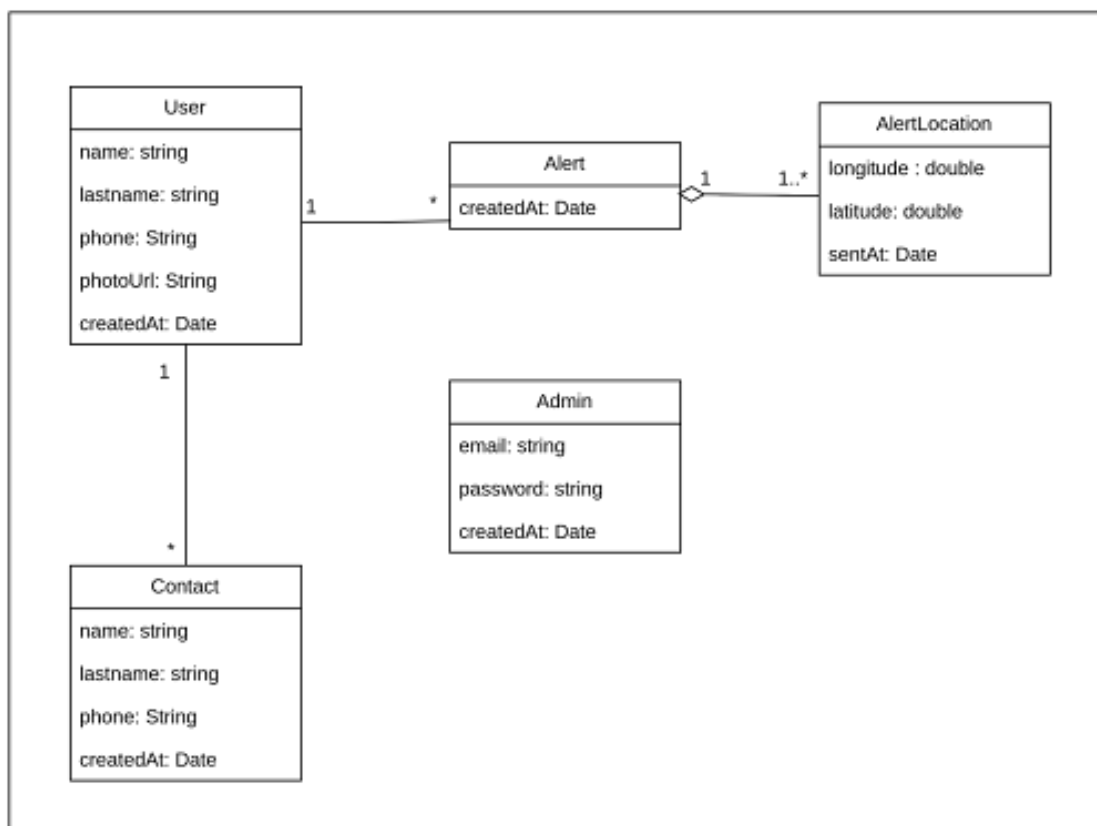


Figure 2.15: Modèle de domaine

2.5 Modélisation de la base de données NoSQL orientée document

La modélisation de la base de données NoSQL constitue une étape cruciale de l'implémentation de notre système. Nous avons opté d'utiliser le service firestore qui propose une base de données non relationnelle, sa capacité à gérer efficacement des données géospatiales, sa prise en charge de la synchronisation en temps réel, ainsi que sa scalabilité, en font un choix idéal pour notre application qui nécessite une réactivité immédiate en cas d'urgence.

Contrairement aux bases de données relationnelles traditionnelles qui ont une forme simple de disposition en lignes et en colonnes. Une base de données orientée documents stocke les informations au format texte, qui consiste en des collections d'enregistrements organisés selon le concept clé-valeur, c'est-à-dire que pour chaque valeur représentée, un nom (ou étiquette) est attribué, qui décrit sa signification. Ce modèle de stockage est connu sous le nom d'objet JSON.

2.5.1 Règles de mappage pour une base de donnée orientée document

2.5.1.1 Relation par référence

Ce type de relation stocke les données en incluant des liens ou des références, d'un document à l'autre. Les applications peuvent résoudre ces références pour accéder aux données connexes dans la structure du document lui-même [14].

2.5.1.2 Documents imbriqués

Ce type de relation est stocké dans une structure de document unique, où les documents imbriqués sont disposés dans un champ ou un tableau. Ces modèles de données dénormalisés permettent de manipuler les données en une seule transaction [14].

2.5.2 Modèle non relationnel

Voici la manière dont nous avons organisé nos données, en priorisant la réduction du nombre de requêtes lors de la lecture.

```
USER : {  
  id : string  
  name : string  
  lastName : string  
  phone : string  
  photoUrl : string  
  createdAt : Date  
  ▼ contacts : [ 1 item  
    ▼ 0 : {  
      phone : string  
      name : string  
      lastName : string  
      createdAt : Date  
    }  
  ]  
}
```

Figure 2.16: Document utilisateur contenant des sous-documents Contact


```
Alert : {  
  id : string  
  userId : string  
  userName : string  
  createdAt : Date  
  ▼ locations : [ 1 item  
    ▼ 0 : {  
      id : string  
      userId : string  
      longitude : double  
      latitude : double  
      sentAt : Date  
    }  
  ]  
}
```

Figure 2.17: Document Alert contenant des sous-documents AlertLocation

2.6 Conclusion partielle

Pour chaque système d'information, une étape essentielle consiste à le modéliser pour orienter sa concrétisation. Le but de ce chapitre était d'expliquer en détail le fonctionnement du système en utilisant des diagrammes UML. Nous aspirons à atteindre notre objectif, car l'utilisation de multiples diagrammes UML nous a offert une variété de perspectives qui nous seront utiles dans la partie suivante de notre travail, qui sera dédiée à la mise en place de notre solution.

CHAP III : PROCESSUS ET IMPLÉMENTATION

3.1 Introduction

Nous en sommes au dernier chapitre de notre travail, ce chapitre se concentrera sur la mise en œuvre de notre solution. Il aura pour mission de mettre en évidence toute la partie technique de notre système. Nous dévoilerons également les défis rencontrés en cours de route, ainsi que les solutions que nous avons adoptées afin de les surmonter. Ce chapitre marque le point culminant de nos efforts pour créer un système d'appel au secours efficace et centré sur l'expérience utilisateur.

3.1.1 *Présentation de l'architecture physique*

Une architecture est appelée architecture physique, parfois aussi architecture technique quand elle décrit la façon dont les composants matériels du système communiquent entre eux, ces éléments peuvent être :

- Le poste de travail
- Matériel de stockage
- Équipement de réseau
- Etc. . .

Dans le cadre de notre application, nous avons opté pour une architecture serverless, un élément clé à noter est que la logique métier de l'application réside au sein des clients eux-mêmes. Contrairement à une architecture traditionnelle où la logique métier est souvent centralisée sur un serveur distant, notre approche décentralisée permet aux clients d'exécuter localement la logique métier.

Cette conception distribuée permet une plus grande réactivité de l'application, réduisant ainsi les dépendances vis-à-vis du serveur distant. Firebase, quant à lui, joue un rôle essentiel en fournissant les services nécessaires pour stocker et synchroniser les données, gérer l'authentification des utilisateurs et faciliter la communication entre les clients et les services cloud. Ainsi, Firebase agit comme une infrastructure backend sans héberger directement la logique métier, offrant ainsi une solution serverless robuste pour notre application.

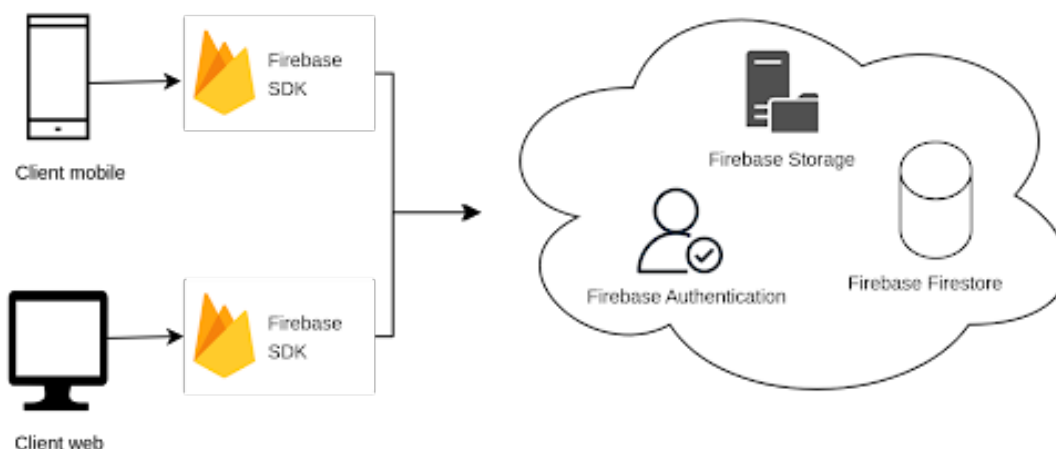


Figure 3.1: Présentation de l'architecture physique

3.2 Technologies utilisées

3.2.1 Kotlin

Kotlin est un langage de programmation orienté objet, à typage statique, interopérable avec la machine virtuelle Java, les bibliothèques de classes Java et Android. Il peut être utilisé presque partout où Java est utilisé, pour le développement côté serveur, les applications Android et bien plus encore. Kotlin fonctionne parfaitement avec toutes les et s'exécute avec le même niveau de performance que Java [15].

3.2.2 TensorFlow lite

TensorFlow Lite est une version légère du framework TensorFlow, spécialement conçue pour exécuter des modèles d'apprentissage automatique sur des appareils mobiles et des systèmes embarqués. Son principal avantage réside dans sa capacité à maintenir des performances élevées tout en étant optimisé pour des environnements aux ressources limitées. TensorFlow Lite permet de déployer des modèles d'intelligence artificielle directement sur des smartphones, des objets connectés, et même des microcontrôleurs, ouvrant ainsi la porte à une gamme étendue d'applications, de la vision par ordinateur à la reconnaissance vocale, le tout en conservant une faible empreinte mémoire et une exécution rapide [16].

3.2.3 Mediapipe

MediaPipe est un outil développé par Google qui offre des outils et des modèles pour le traitement des médias en temps réel, y compris la vision par ordinateur et le suivi de mouvement. Il simplifie le déploiement des modèles de machine learning sur

des périphériques tels que des smartphones avec des bibliothèques low code. Ces bibliothèques comprennent actuellement des outils pour trois catégories : la vidéo, l'audio et le texte.

3.2.4 Yamnet

YAMNet[17] est un réseau neuronal préformé qui utilise l'architecture de convolution séparable en profondeur MobileNetV1 [18]. Il peut utiliser une forme d'onde audio comme entrée et faire des prédictions indépendantes pour chacun des 521 événements audio du corpus AudioSet [19].

3.2.5 Firebase

Firebase est une plateforme de développement d'applications mobiles et web appartenant à Google, qui offre un ensemble de services cloud, tels que des bases de données en temps réel, l'authentification des utilisateurs, l'hébergement web, le stockage de fichiers, et des outils d'analyse avancés. Elle simplifie considérablement le développement d'applications en fournissant des fonctionnalités prêtes à l'emploi, ce qui permet aux développeurs de se concentrer sur la création de fonctionnalités uniques tout en bénéficiant de la puissance du cloud pour la gestion des données et des utilisateurs.

3.2.6 Angular

Angular est un framework de développement d'applications web open-source développé par Google. Il est largement utilisé pour la création d'applications web modernes et dynamiques. Angular est basé sur le langage TypeScript, une version améliorée de JavaScript, et offre un ensemble complet de fonctionnalités pour simplifier le développement front-end.

3.2.7 Android Studio

Android Studio est un environnement de développement intégré officiellement pris en charge par Google pour la création d'applications Android. Il offre aux développeurs un ensemble complet d'outils pour concevoir, développer, déboguer et déployer des applications Android.

3.3 Diagramme de déploiement

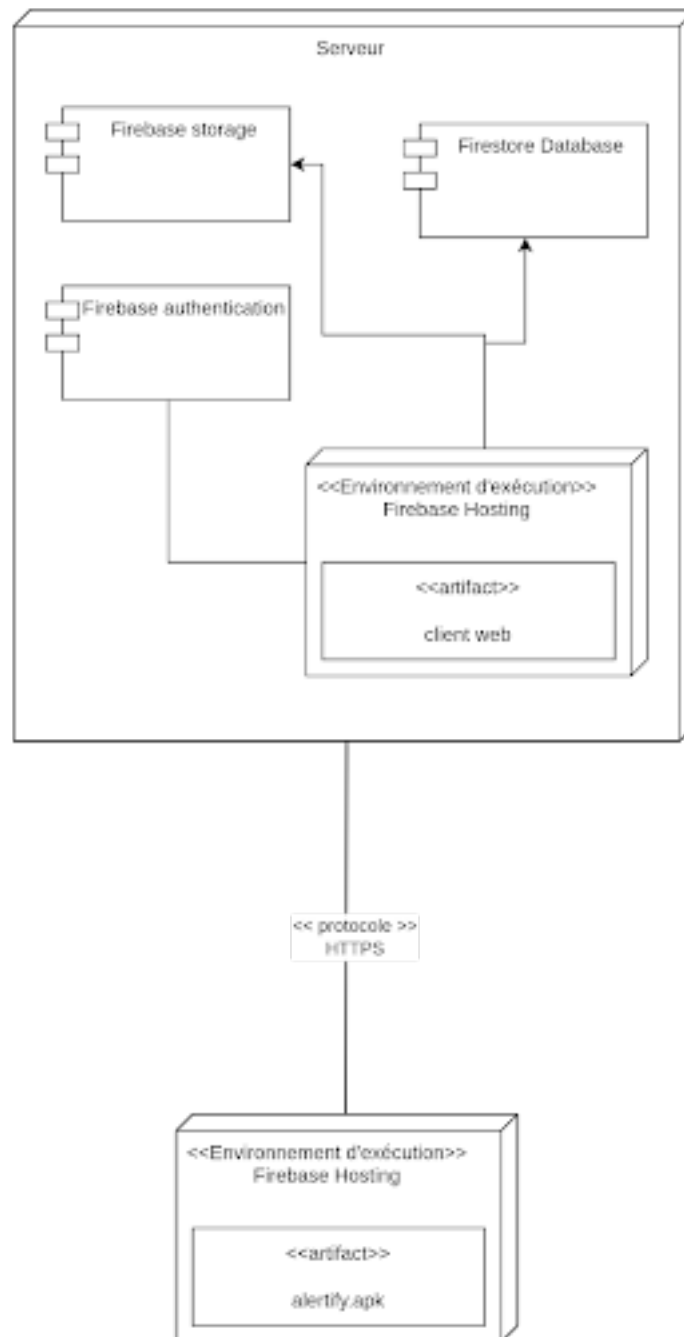


Figure 3.2: Diagramme de déploiement

3.4 Expérience utilisateur et fonctionnement

3.4.1 *Expérience utilisateur*

Le processus de conception de l'expérience utilisateur consiste à s'assurer qu'aucun aspect de l'expérience de l'utilisateur avec une solution ne se produit sans l'intention consciente et explicite de ce dernier. Cela signifie qu'il faut prendre en compte chaque possibilité de chaque action que l'utilisateur est susceptible de faire et de comprendre ses attentes à chaque étape du processus [20].

Dans le cadre de notre travail, l'attention portée à l'UX est cruciale pour garantir l'adoption et l'efficacité du système par les utilisateurs finaux.

Tout d'abord, il est essentiel de comprendre les besoins, les compétences techniques et les préférences des utilisateurs potentiels à Lubumbashi. Cette compréhension approfondie permettra de concevoir une interface utilisateur conviviale et intuitive, adaptée au contexte local. Il est crucial que l'application soit facile à comprendre et à utiliser, même pour les utilisateurs peu familiers avec les technologies modernes.

3.4.2 *Fonctionnement*

Pour permettre à notre application mobile de recevoir des commandes, nous avons recouru au composant d'application appelé service. Un service est un composant d'application qui peut effectuer des opérations de longue durée en arrière-plan. Il ne fournit pas d'interface utilisateur. Une fois lancé, un service peut continuer à fonctionner pendant un certain temps, même après que l'utilisateur a basculé vers une autre application [21]. Mais avant tout, nous devrions commencer par demander la permission d'accéder à la géolocalisation ainsi qu'au périphérique audio du smartphone, notamment le microphone.

Une fois que l'application a obtenu l'accès au microphone, les données captées sont évaluées à l'aide de MediaPipe et TensorFlow Lite. MediaPipe et TensorFlow Lite ont respectivement pour objectif de détecter des bruits tels que les pleurs et les gémissements, qui peuvent indiquer qu'une personne est en danger, ainsi que des commandes vocales spécifiques. Ces scénarios déclenchent alors le lancement d'alertes. Les alertes entraînent l'envoi de messages contenant la géolocalisation du smartphone aux contacts préenregistrés, ainsi qu'au serveur backend, dans le cas où le smartphone dispose d'une connexion Internet.

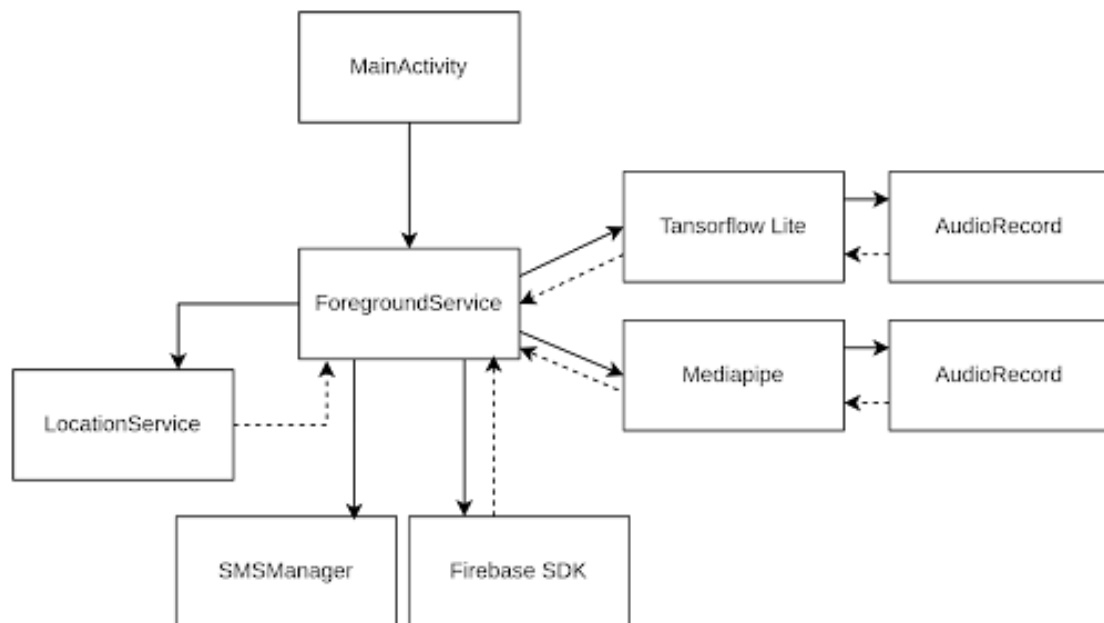


Figure 3.3: Communication des différentes composantes lors de l'envoi d'une alerte

```

val serviceIntent = remember {
    Intent(context, TrackingService::class.java)
}

val locationPermissionRequest = rememberLauncherForActivityResult(
    ActivityResultContracts.RequestMultiplePermissions()
) { permissions →
    if (permissions.containsKey(false)) {
        Toast.makeText(
            context,
            text: "La localisation est nécessaire pour le tracage",
            Toast.LENGTH_SHORT
        ).show()
    } else {
        context.startForegroundService(serviceIntent)
    }
}

```

Figure 3.4: Demande de permission et lancement du service TrackingService

```
locationCallback = object : LocationCallback() {

    override fun onLocationResult(locationResult: LocationResult) {
        super.onLocationResult(locationResult)
        contacts.forEach { contact →
            CoroutineScope(Dispatchers.IO).launch { this: CoroutineScope
                for (location in locationResult.locations) {
                    Timber.e(message: "Location: ${location.latitude} ${location.longitude}")
                    sendAlert(alertId, location, contact, user)
                }
            }
        }
    }
}

fusedLocationClient = LocationServices.getFusedLocationProviderClient(context: this)
val locationRequest = LocationRequest.Builder(intervalMillis: 15000)
    .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY)
    .build()

fusedLocationClient.requestLocationUpdates(
    locationRequest,
    locationCallback,
    Looper.getMainLooper()
)
```

Figure 3.5: Récupération de la localisation


```

yveskalume
suspend fun startAudioClassification() {
    if (recorder?.recordingState == AudioRecord.RECORDSTATE_RECORDING) {
        return
    }

    recorder?.startRecording()

    val lengthInMilliseconds =
        ((REQUIRE_INPUT_BUFFER_SIZE * 1.0f) / SAMPLING_RATE_IN_HZ) * 1000

    val interval = (lengthInMilliseconds * (1 - (DEFAULT_OVERLAP * 0.25))).toLong()

    classifyJob = CoroutineScope(Dispatchers.Default).launch { this: CoroutineScope
        while (true) {
            recorder?.let { classifyAudioAsync(it) }
            delay(interval)
        }
    }
}

yveskalume
private fun classifyAudioAsync(audioRecord: AudioRecord) {
    val audioData = AudioData.create(
        AudioDataFormat.create(recorder!!.format), SAMPLING_RATE_IN_HZ
    )
    audioData.load(audioRecord)

    val inferenceTime = SystemClock.uptimeMillis()
    audioClassifier?.classifyAsync(audioData, inferenceTime)
}

```

Figure 3.6: Classification des entrées sonores

```

for (i in 0 ≤ until < RECORDING_LENGTH) {
    floatInputBuffer[i][0] = inputBuffer[i] / 32767.0f
}
val inputArray = arrayOf<Any>(floatInputBuffer, sampleRateList)
val outputMap: MutableMap<Int, Any> = HashMap()
outputMap[0] = outputScores

// Run the model.
tfLiteLock.lock()
try {
    tfLite!!.runForMultipleInputsOutputs(inputArray, outputMap)
} finally {
    tfLiteLock.unlock()
}

// Use the smoother to figure out if we've had a real recognition event.
val currentTime = System.currentTimeMillis()
val result: RecognizeCommands.RecognitionResult =
    recognizeCommands!!.processLatestResults(outputScores[0], currentTime)
lastProcessingTimeMs = Date().time - startTime
if (!result.foundCommand.startsWith(prefix: "_") && result.isNewCommand) {
    val score: String = (result.score * 100).roundToInt().toString() + "%"
    Timber.e(result.foundCommand)
    Timber.e(message: "Score: $score")
} else {
    Timber.e(result.foundCommand)
}
delay(MINIMUM_TIME_BETWEEN_SAMPLES_MS)

```

Figure 3.7: Détection d'une commande vocale sur les entrées audios

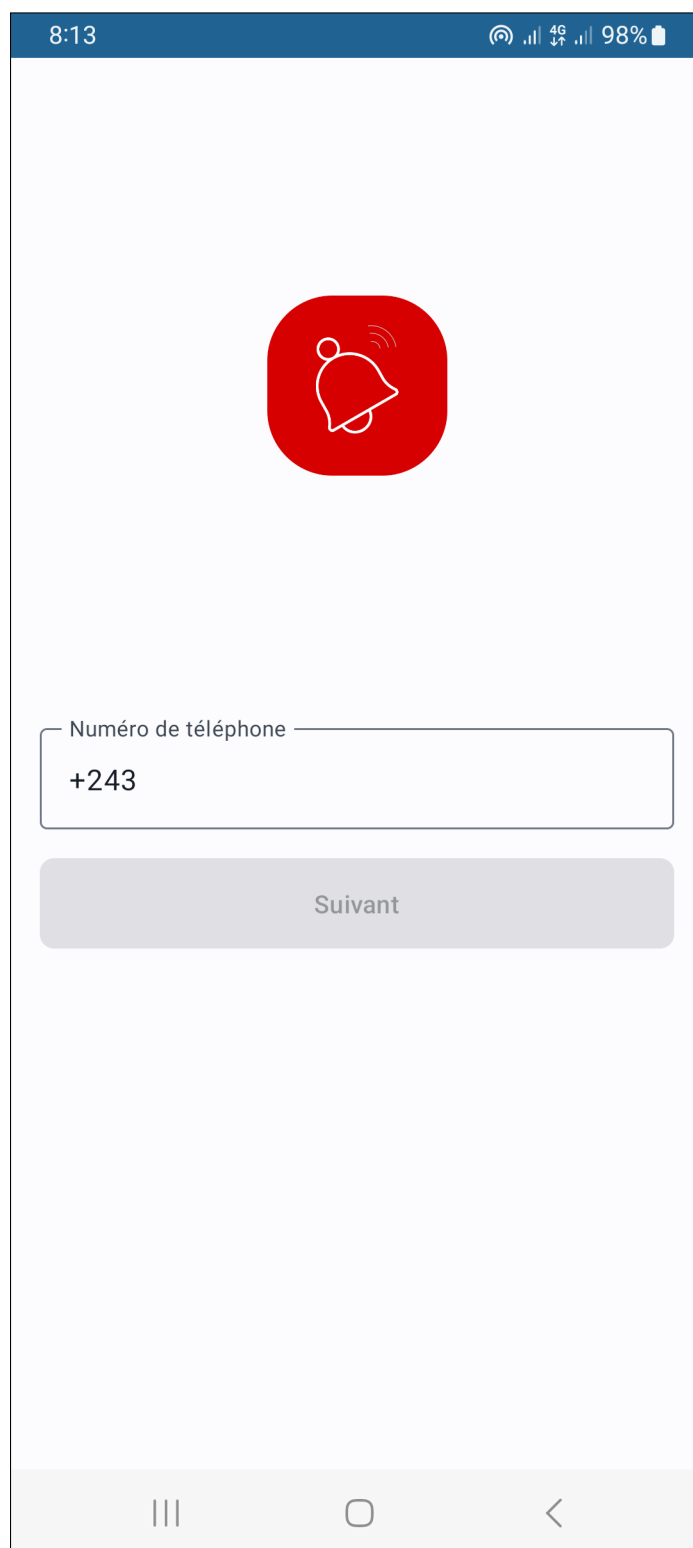
```
fun sendAlert(alertId: String, location: Location, contact: Contact, user: FirebaseUser) {
    try {
        CoroutineScope(SupervisorJob()).launch { this: CoroutineScope
            val smsManager: SmsManager =
                this@TrackingService.getSystemService(SmsManager::class.java)
            smsManager.sendTextMessage(
                contact.phone,
                scAddress: null,
                text: "Alerte ici ${user.displayName} ! je suis en danger. " +
                    "\nvoici ma position: " +
                    "https://maps.google.com/?ll=${location.latitude},${location.longitude}",
                sentIntent: null,
                deliveryIntent: null
            )
        }

        val alertLocation = AlertLocation(
            id = UUID.randomUUID().toString(),
            longitude = location.longitude,
            latitude = location.latitude,
            createdAt = Date()
        )
        firestore.document( documentPath: "alerts/$alertId/locations/${alertLocation.id}")
            .set(alertLocation)
    } catch (e: Exception) {
        Timber.e(e)
    }
}
```

Figure 3.8: Envois d'alertes

3.5 Captures d'écrans

3.5.1 Client mobile pour utilisateur



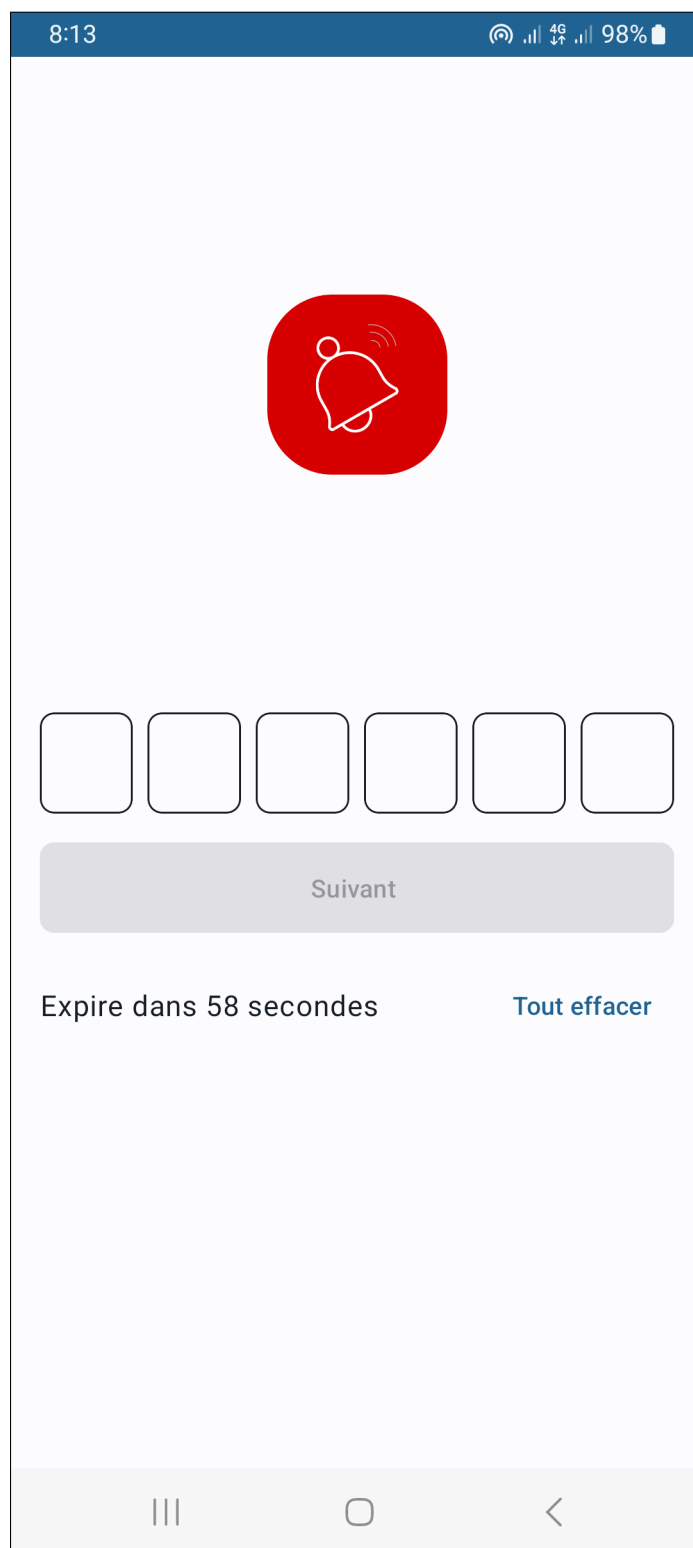
8:13 4G 98%

Numéro de téléphone

+243

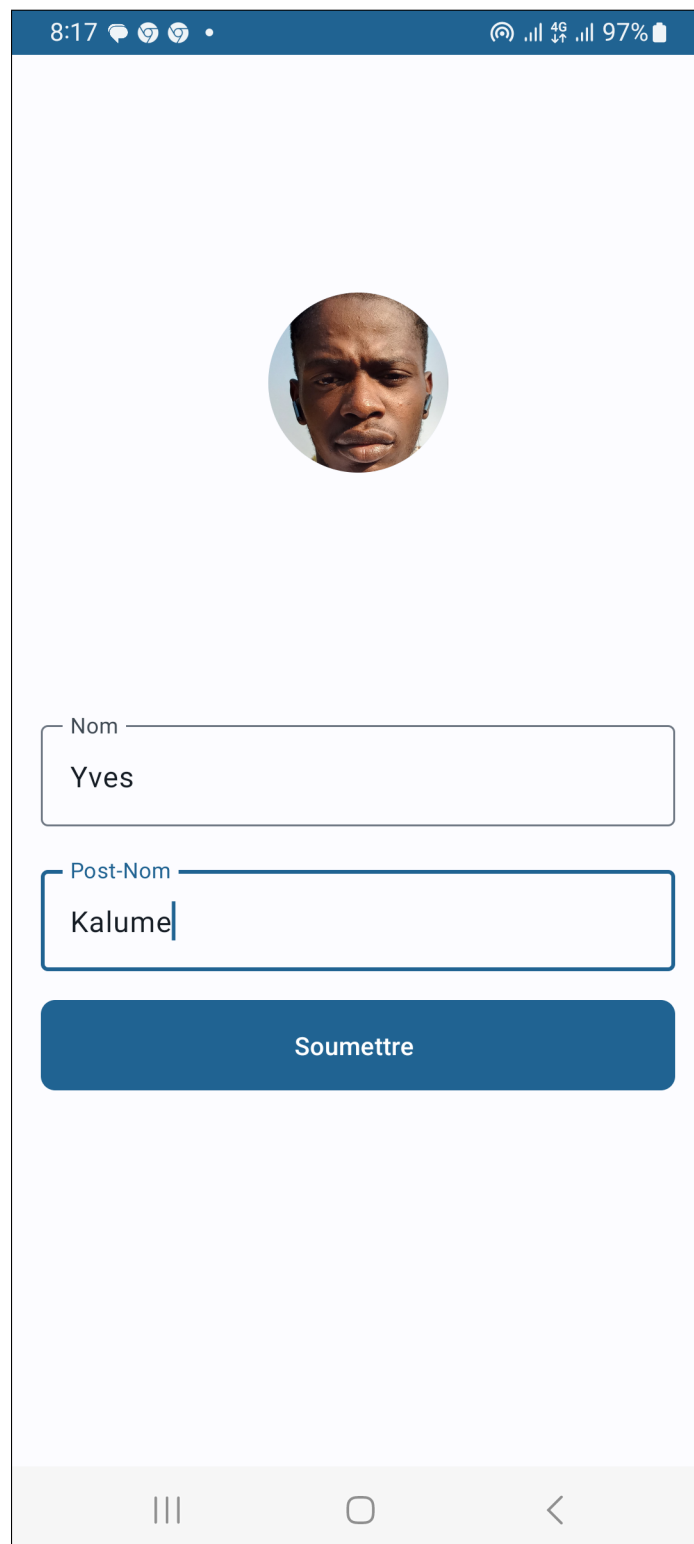
Suivant

Figure 3.9: Écran d'enregistrement du numéro de l'utilisateur



The image shows a mobile application screen for submitting a confirmation code. At the top, a dark blue status bar displays the time 8:13, signal strength, 4G network, and 98% battery. The main content area has a light blue background. In the center is a large red rounded square icon with a white bell and signal waves. Below this is a row of six empty square input boxes. Under the boxes is a grey button labeled 'Suivant'. At the bottom of the input area, it says 'Expire dans 58 secondes' and a blue link 'Tout effacer'. The bottom of the screen features a grey navigation bar with three icons: a hamburger menu, a circle, and a back arrow.

Figure 3.10: Écran de soumission du code de confirmation



The image shows a mobile application interface for user registration. At the top, a status bar displays the time 8:17, signal strength, 4G network, and 97% battery. The main content area has a light blue background. In the center, there is a circular profile picture of a man. Below the photo, there are two text input fields. The first field is labeled 'Nom' and contains the text 'Yves'. The second field is labeled 'Post-Nom' and contains the text 'Kalume'. Below these fields is a large blue button with the text 'Soumettre'. At the bottom of the screen, there is a grey navigation bar with three icons: a hamburger menu, a home button, and a back arrow.

8:17 4G 97%

Nom

Yves

Post-Nom

Kalume

Soumettre

Figure 3.11: Écran d'enregistrement de l'identité de l'utilisateur

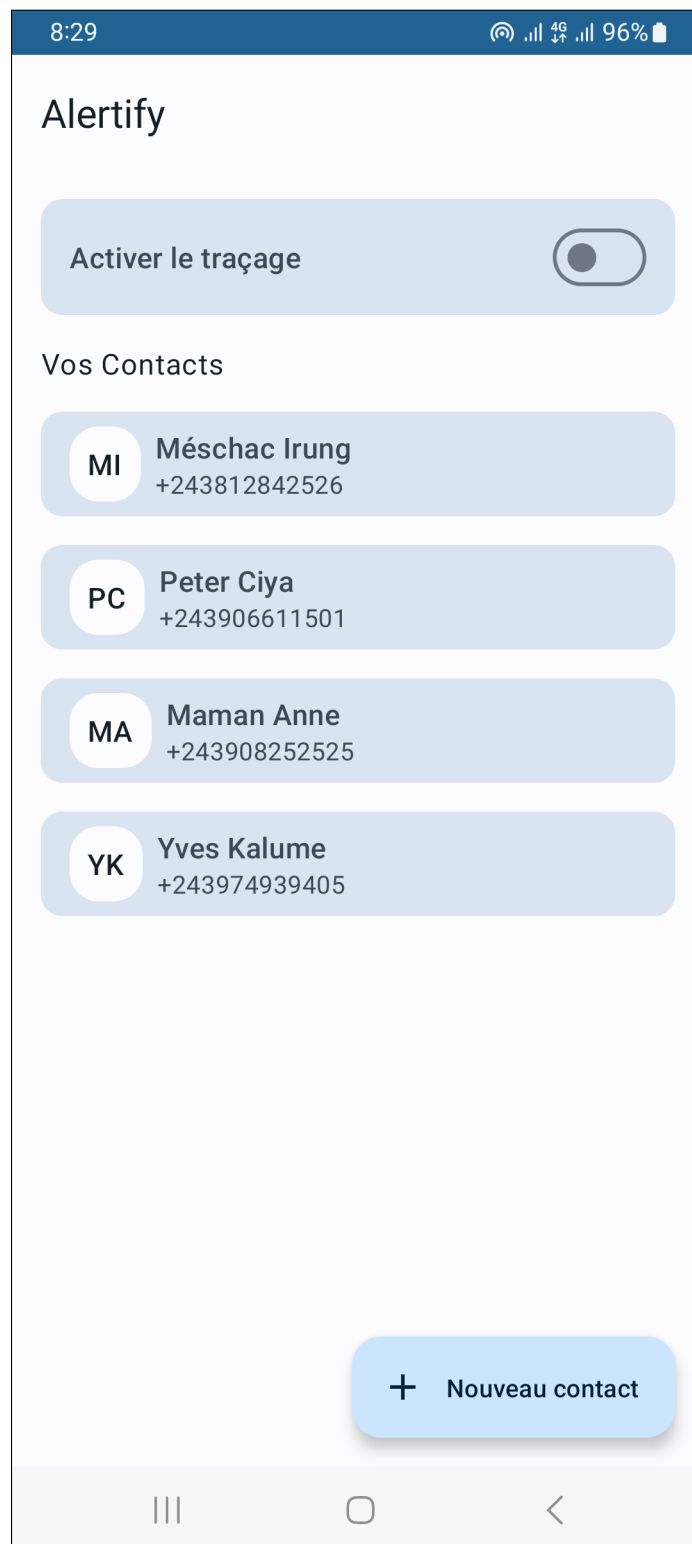


Figure 3.12: Écran principal

9:44 4G 67%

Nouveau contact

Nom
Yves

Post-nom
Kalume

Numero de telephone
+243

Enregistrer

1 2 ABC 3 DEF
4 GHI 5 JKL 6 MNO
7 PQRS 8 TUV 9 WXYZ
⌫ 0 ↩

Figure 3.13: Écran permettant d'ajouter un contact

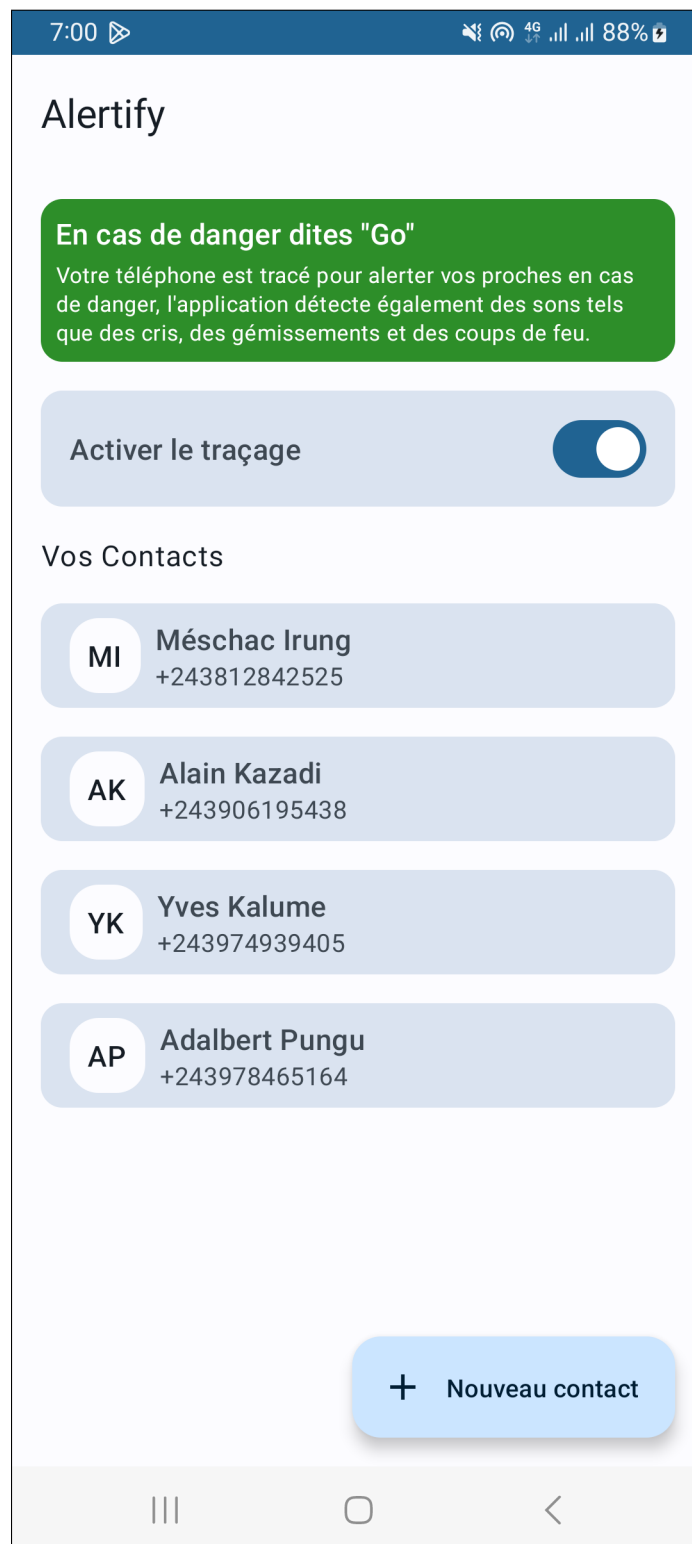


Figure 3.14: Écran d'accueil, tracking activé

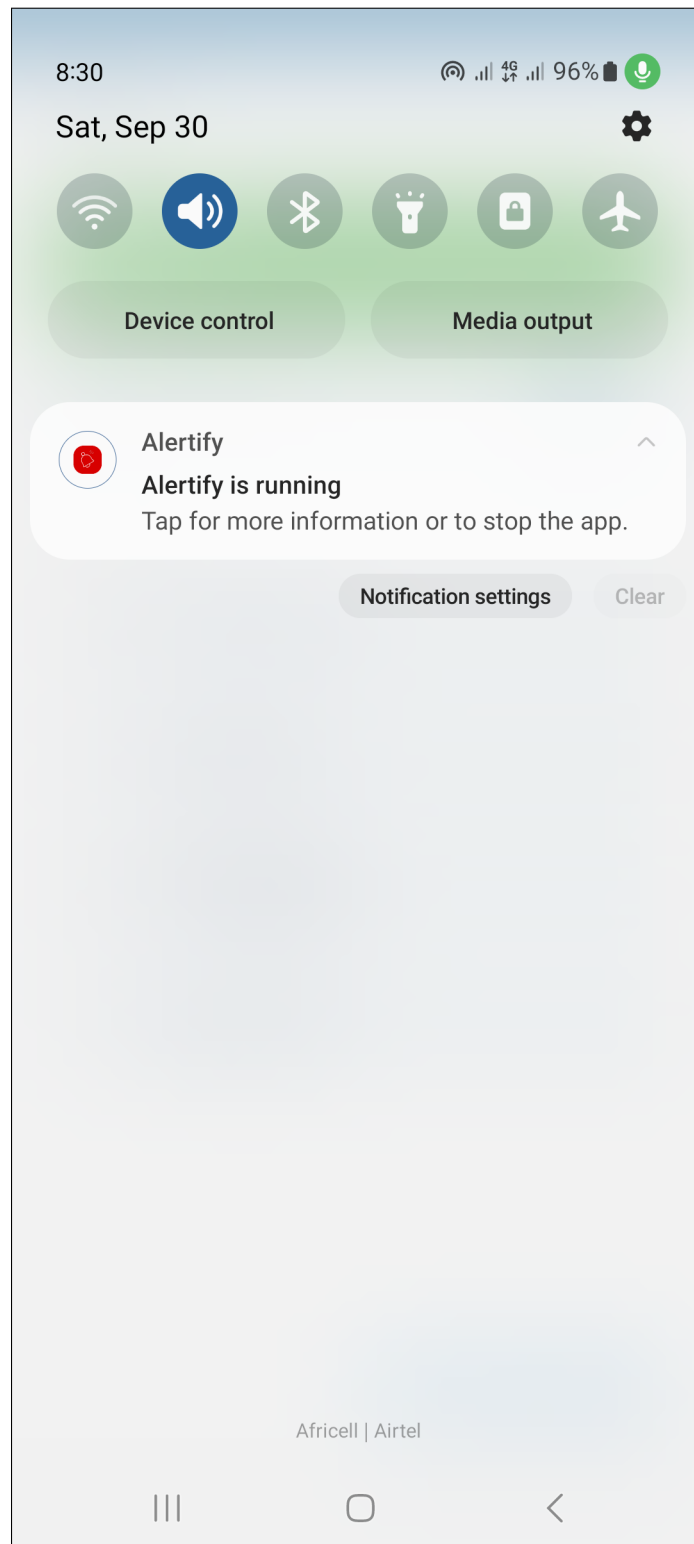


Figure 3.15: Notification signalant que le tracking est en cours

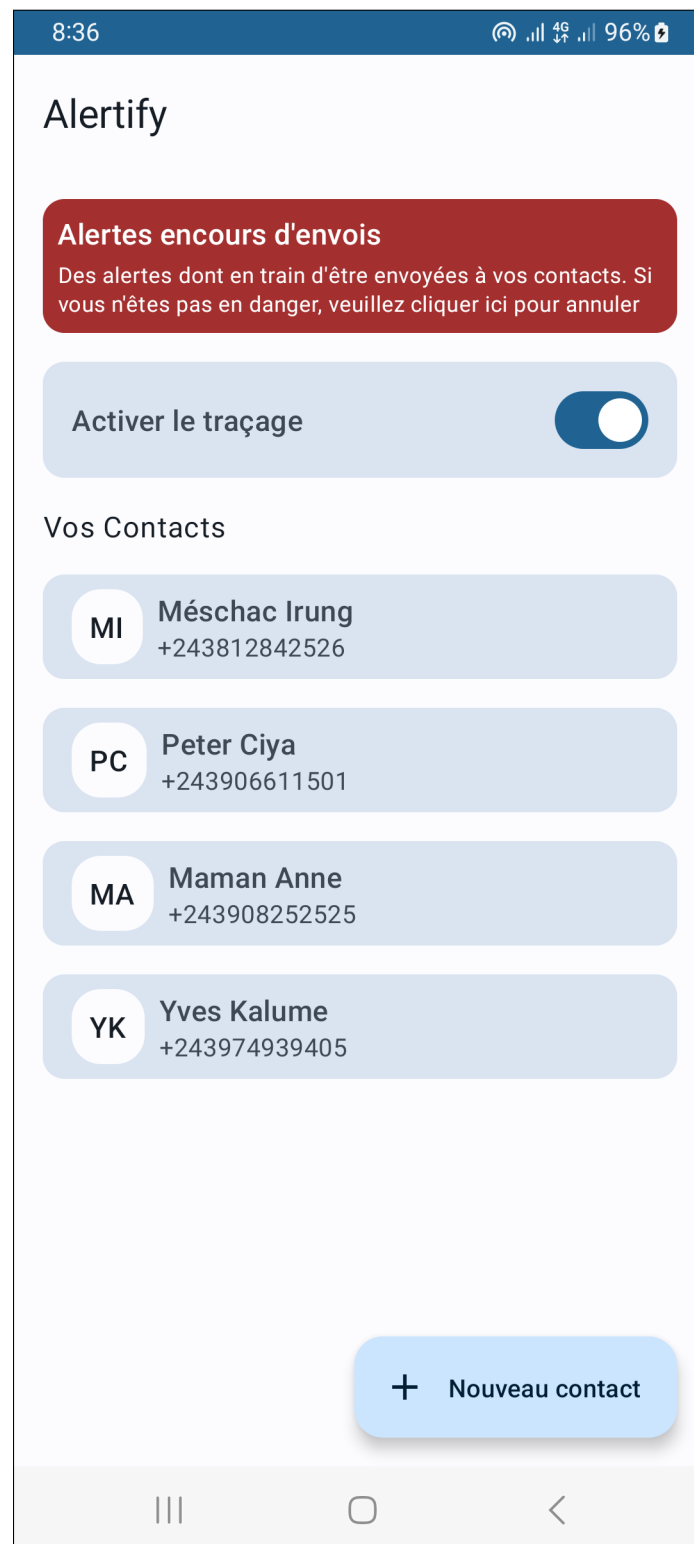


Figure 3.16: Écran principal, alertes en cours d'envoi

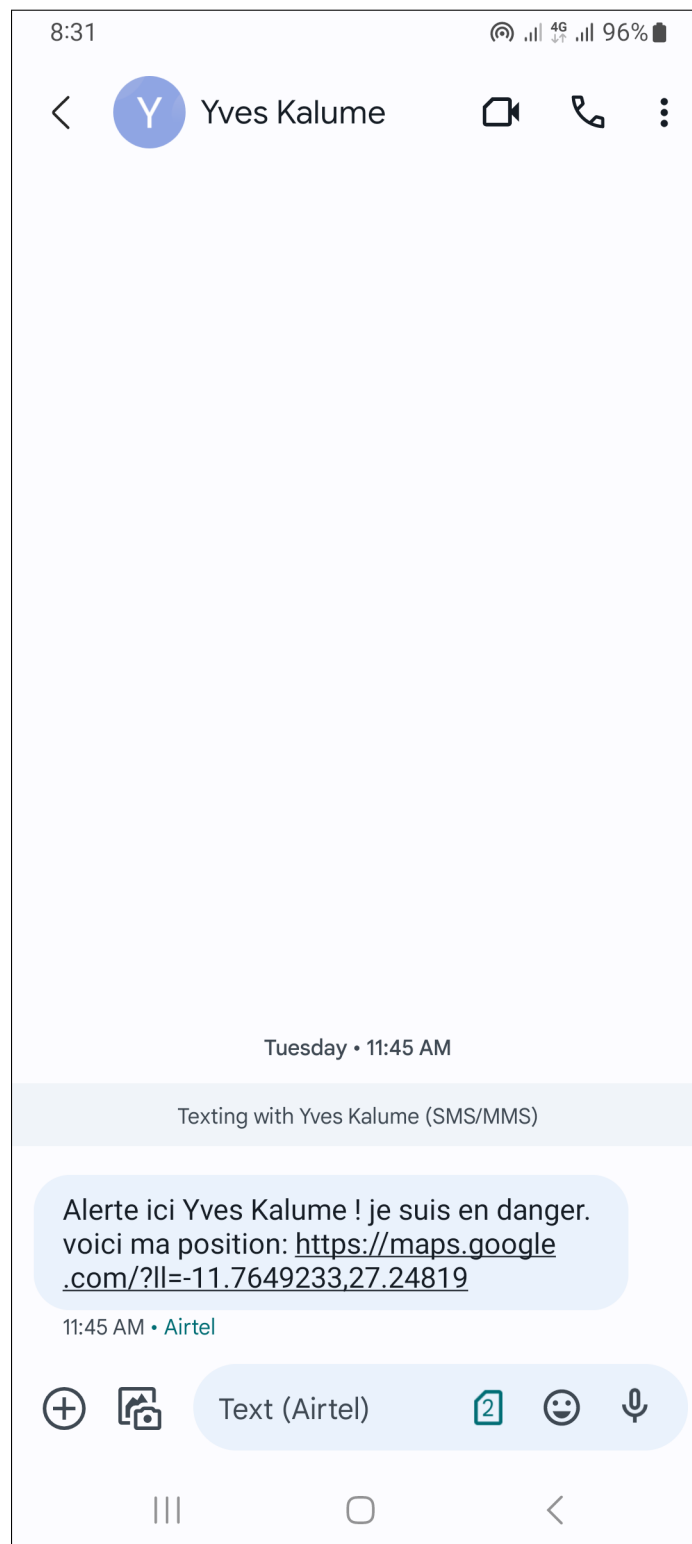


Figure 3.17: SMS d'alerte

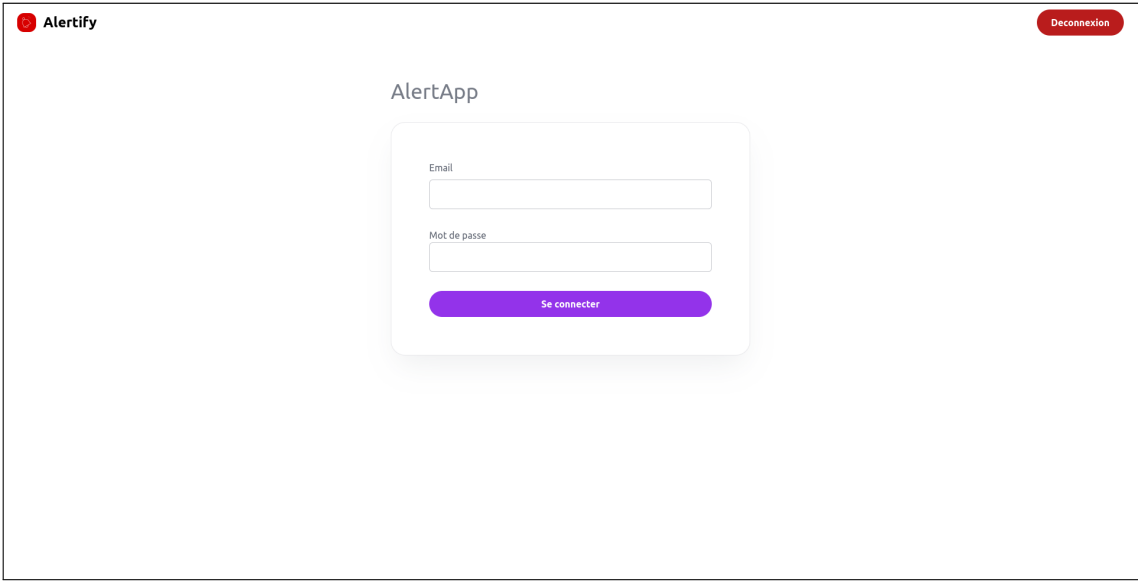


Figure 3.18: Page de connexion administrateur

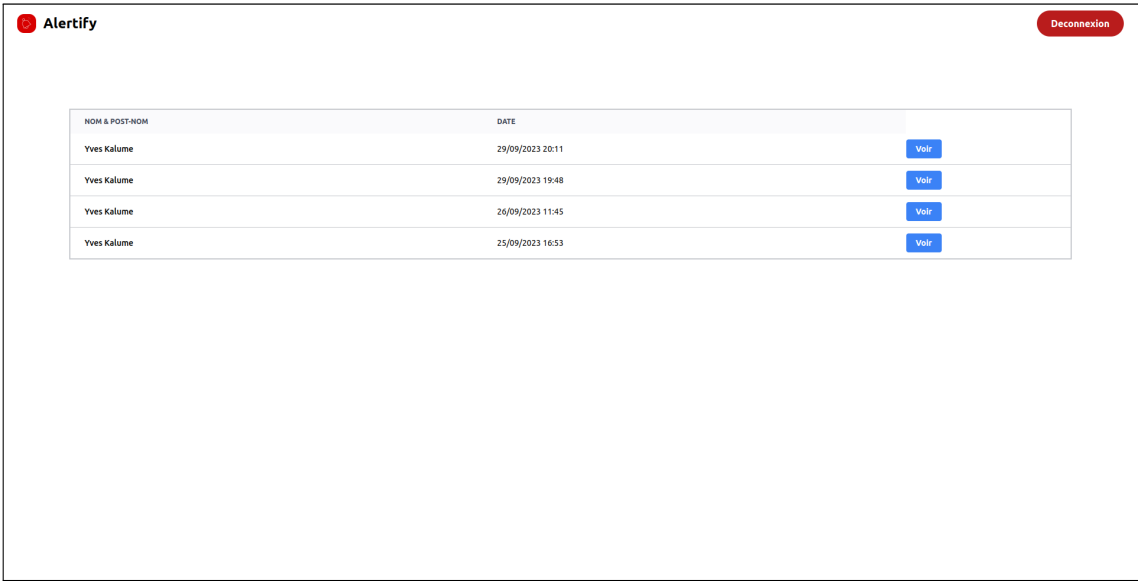


Figure 3.19: Listes des alertes

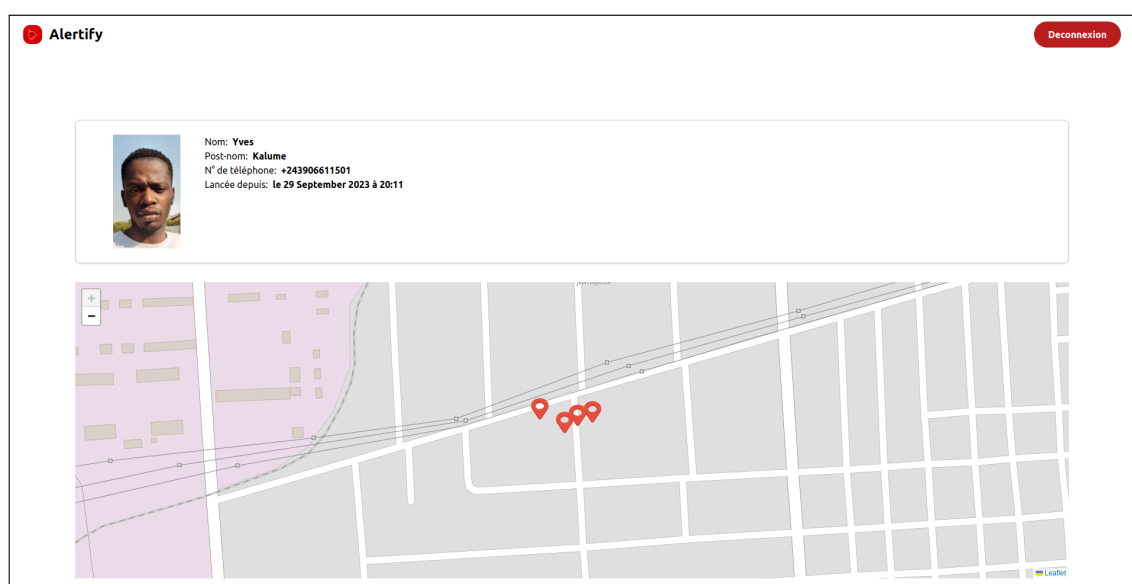


Figure 3.20: Détail d'une alerte

3.6 Difficultés rencontrées

Un travail scientifique vise toujours à résoudre un problème spécifique rencontré en apportant des solutions acceptables. C'est dans la recherche de cette solution que l'on oriente ses recherches dans un domaine bien déterminé et qu'on fixe les limites.

- *Manque d'informations des sources officielles*

Le manque d'informations fiables provenant de sources officielles a constitué un obstacle majeur. Les données sur les enlèvements, leur fréquence, leurs lieux et leurs circonstances étaient lacunaires et difficiles d'accès. Cela a compliqué notre travail d'analyse des tendances et d'adaptation de notre système aux besoins spécifiques de la ville de Lubumbashi. Nous avons dû recourir à des données disponibles, souvent limitées, et à des témoignages de la communauté locale pour informer notre approche.

- *Simulation de cas réels d'enlèvement*

Recréer fidèlement des cas réels d'enlèvement pour tester notre solution dans un contexte authentique s'est avéré une entreprise complexe, imprévisible et potentiellement dangereuse. La sécurité des participants est une priorité absolue, et la reproduction de situations de kidnapping authentiques implique des risques considérables. En conséquence, nous avons dû concevoir des méthodes alternatives de test qui garantissent la sécurité des participants tout en fournissant des résultats valables pour évaluer l'efficacité de notre système.

Ces difficultés ont mis à l'épreuve notre capacité à concevoir et à mettre en œuvre une solution efficace, tout en soulignant l'importance cruciale de l'adaptabilité et de la flexibilité pour répondre aux défis réels du terrain.

3.7 Conclusion partielle

Ce chapitre a mis en lumière la mise en œuvre technique de notre système d'appel au secours dédié à la prévention des enlèvements dans la ville de Lubumbashi. Nous avons plongé dans les détails des algorithmes essentiels, des outils et des technologies utilisés. Au-delà des aspects techniques, nous avons également accordé une attention particulière à l'expérience utilisateur qui est un élément central pour garantir que notre solution est accessible, conviviale et efficace, enfin, nous avons parlé des défis que nous avons rencontrés et les solutions que nous avons apportées.

CONCLUSION GÉNÉRALE

Nous arrivons au terme de notre travail consacré à la conception d'un système d'appel au secours en se basant sur le cas d'enlèvement dans la ville de Lubumbashi. Nous avons parcouru un chemin jalonné de défis, d'analyses approfondies et de développements techniques pour répondre à la problématique de renforcer la sécurité et le bien-être de la communauté.

Nous avons avancé l'hypothèse qu'un système d'appel au secours, permettant à une victime d'enlèvement de déclencher discrètement un appel à l'aide, pourrait jouer un rôle clé dans l'amélioration de la sécurité de la ville. Pour atteindre cet objectif, nous avons exploré et développé des solutions basées sur la commande vocale et la reconnaissance sonore permettant d'alerter efficacement les contacts d'urgence et envoyer sa localisation en cas de danger imminent.

En fin de compte, nous espérons que notre travail contribuera à améliorer la sécurité dans la ville de Lubumbashi et à offrir aux résidents une nouvelle couche de protection en cas de danger. Cependant, nous sommes conscients que toute œuvre est teintée d'imperfections, et nous encourageons d'autres chercheurs à explorer ce même chemin pour éclaircir davantage les zones d'ombre qui subsistent. Notre travail ne s'achève pas ici, il marque le début d'un engagement continu envers la sécurité et le bien-être de la communauté.

BIBLIOGRAPHIE

- [1] The DSDM Consortium, *DSDM Business Focused Development*, second edition ed.
- [2] A. Agrawal, A. Maurya, A. Patil, Student, and R. Guide, “Voice-controlled tool for anytime safety of women,” 05 2021.
- [3] “Polaris: How to find the North Star.” [Online]. Available: <http://www.space.com/15567-north-star-polaris.html>
- [4] Michael Brian Schiffer, *The Electric Lighthouse in the Nineteenth Century: Aid to Navigation and Political Technology*, 2005.
- [5] Frederic Chapin Lane, *Venice, A Maritime Republic*, 1973.
- [6] P. Daly, “Navstar GPS and GLONASS: Global satellite navigation systems,” *IOSR JECE : Journal of Electronics and Communication Engineering*, vol. 5.
- [7] Frédéric Evennou, “Techniques et technologies de localisation avancées pour terminaux mobiles dans les environnements indoor,” Ph.D. dissertation, Université Joseph-Fourier - Grenoble I, 2007.
- [8] W. Guier and G. Weiffenbach, “Genesis of Satellite Navigation,” *Johns Hopkins APL Digest*, 1997.
- [9] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and Elmar Wasle, *GNSS – Global Navigation Satellite Systems GPS, GLONASS, Galileo, and more*.
- [10] Xavier Blanc and Isabelle Mounier, *UML2 pour les développeurs : Cours avec exercices corrigés*, éd. eyrolles ed.
- [11] JIM ARLOW and ILA NEUSTADT, *UML 2.0 and the unified process: practical object-oriented analysis and design*, westford, massachusetts: addison-wesley. ed.
- [12] Jason Bennett and Jennifer Bowen, *Agile Project Management Step-by-Step Guide to Agile Project Management*.
- [13] Mark Richards and Neal Ford, *Fundamentals of Software Architecture*. O’Reilly Media, Inc.
- [14] Harley Vera, Wagner Boaventura, Maristela Holanda, Valeria Guimaraes, and Fernanda Hondo, “Data Modeling for NoSQL Document-Oriented Databases,” *Department of Computer Science University of Brasília*.
- [15] DMITRY JEMEROV and SVETLANA ISAKOVA, *Kotlin in Action*. ed. MANNING.

-
- [16] “ML for Mobile and Edge Devices - TensorFlow Lite,” Sep. 2023.
- [17] “Classification du son avec YAMNet.” [Online]. Available: <https://www.tensorflow.org/hub/tutorials/yamnet?hl=fr>
- [18] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.”
- [19] “A large-scale dataset of manually annotated audio events.” [Online]. Available: <https://research.google.com/audioset/>
- [20] Jesse James Garrett, *The elements of user experience*.
- [21] “Services overview,” En ligne, consulté le: 21 septembre 2023. [Online]. Available: <https://developer.android.com/guide/components/services>