



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**

Membre de **HONORIS UNITED UNIVERSITIES**

RAPPORT MINI PROJET DE DEVELOPPEMENT MOBILE

Plateforme de Signalement et de Gestion des Incidents

Encadré par : **Madame ZINEB NAJI**

Réalisé par **ADNAN Yasmine**
HAKIM Ilyass
RIZQY Nabila

Année Scolaire : 2025-2026
Filière : 4IIR G11

REMERCIEMENTS

Je tiens à exprimer mes sincères remerciements à toutes les personnes qui ont contribué, de près ou de loin, à la réalisation de ce mini projet.

Je remercie tout particulièrement **mon professeur académique Madame ZINEB NAJI** pour son accompagnement, ses conseils pertinents et sa disponibilité tout au long du projet. Ses orientations ont été déterminantes pour la bonne conduite de ce travail.

Je remercie également **l'équipe pédagogique** pour la qualité de la formation dispensée, qui m'a permis d'acquérir les compétences nécessaires à la réalisation de ce projet.

RÉSUMÉ

Ce mini projet porte sur la conception et le développement d'une **plateforme de signalement et de gestion des incidents**, destinée à centraliser, structurer et sécuriser le processus de déclaration et de traitement des incidents.

La solution proposée repose sur une application mobile permettant aux utilisateurs de signaler des incidents en fournissant une description, une catégorie, une localisation et des pièces jointes. Un backend cloud assure la gestion des données, l'authentification des utilisateurs, le suivi du cycle de vie des incidents ainsi que la traçabilité des actions réalisées. Une interface d'administration permet la supervision globale, l'assignation des incidents et le suivi des indicateurs de performance.

Les résultats obtenus montrent que la plateforme améliore significativement la réactivité des équipes, la coordination entre les acteurs et la visibilité sur l'état des incidents, tout en garantissant la sécurité et la confidentialité des données.

ABSTRACT

This project presents the design and implementation of a mobile-based incident reporting and management platform aimed at improving the efficiency, traceability, and security of incident handling processes. The proposed system centralizes incident reporting and ensures structured data management across all stages of the incident lifecycle.

The solution relies on a mobile application that allows users to submit incident reports including descriptive information, categorization, geolocation, and multimedia attachments. A cloud-based backend ensures secure user authentication, real-time data storage, incident tracking, and action traceability. Additionally, an administrative interface provides centralized supervision, incident assignment, and performance monitoring capabilities.

Experimental results indicate that the proposed platform enhances operational responsiveness, strengthens coordination among stakeholders, and improves visibility over incident status, while maintaining high standards of data security and confidentiality.

TABLE DE MATIÈRES

REMERCIEMENTS.....	3
RÉSUMÉ.....	4
ABSTRACT.....	5
I. INTRODUCTION	7
II. CONTEXTE ET PROBLÉMATIQUE	8
III. OBJECTIFS DU PROJET	9
3.1 OBJECTIF PRINCIPAL	9
3.2 OBJECTIFS SPÉCIFIQUES	9
IV. TECHNOLOGIES UTILISÉES	10
V. ARCHITECTURE TECHNOLOGIQUE DE LA PLATEFORME.....	12
VI. CONCEPTION :	13
1. DIAGRAMME DE CLASSES (CLASS DIAGRAM).....	13
2. DIAGRAMME DES CAS D'UTILISATION (USE CASE DIAGRAM)	14
3. DIAGRAMME DE SÉQUENCE.....	15
VII. RÉSULTATS ET RÉALISATIONS DE L'APPLICATION	19
VIII. CONCLUSION.....	30

I. INTRODUCTION

La gestion efficace des incidents constitue aujourd'hui un enjeu majeur pour les organisations, les institutions publiques et les entreprises, en raison de l'impact direct qu'elle peut avoir sur la continuité des services, la satisfaction des utilisateurs et la sécurité globale. Une prise en charge rapide et structurée des incidents permet non seulement de limiter les risques et les dysfonctionnements, mais également d'améliorer la qualité des services offerts.

Cependant, dans de nombreux contextes, les méthodes traditionnelles de signalement reposent encore sur des moyens non centralisés tels que les appels téléphoniques, les échanges par courrier électronique ou les formulaires papier. Ces approches présentent plusieurs limites, notamment la perte d'informations, l'absence de suivi en temps réel, les délais de traitement élevés et le manque de coordination entre les différents intervenants. De plus, elles ne permettent pas une traçabilité fiable des actions ni une analyse efficace des incidents sur le long terme.

Face à ces constats, ce mini projet vise la conception et le développement d'une plateforme numérique de signalement et de gestion des incidents. Cette solution, accessible via une application mobile et complétée par une interface d'administration, permet de centraliser l'ensemble des incidents et d'assurer une gestion complète de leur cycle de vie, depuis la déclaration jusqu'à la résolution finale. L'objectif est de proposer un système sécurisé, fiable et évolutif, favorisant la réactivité, la transparence et la collaboration entre les différents acteurs impliqués.

II. CONTEXTE ET PROBLÉMATIQUE

Dans de nombreuses structures, la gestion des incidents souffre d'un manque d'organisation dû à l'utilisation de canaux multiples et non structurés pour le signalement et le suivi. Cette dispersion de l'information rend difficile la priorisation des incidents, complique leur traitement et empêche une vision globale de la situation. Par conséquent, les équipes responsables interviennent souvent avec retard, ce qui peut engendrer des impacts négatifs sur les services et sur la satisfaction des utilisateurs.

Par ailleurs, l'absence d'un système centralisé limite fortement les possibilités d'analyse des données, telles que l'identification des incidents récurrents, l'évaluation des performances des équipes ou l'amélioration continue des processus. De plus, la communication entre les différents intervenants (utilisateurs, équipes techniques, responsables) reste souvent fragmentée, ce qui nuit à la coordination et à la transparence.

Dans ce contexte, la problématique principale de ce projet peut être formulée comme suit :
Comment concevoir et mettre en œuvre une solution centralisée, sécurisée et évolutive permettant de structurer le signalement, le suivi et le traitement des incidents, tout en facilitant la communication et la collaboration entre les différents acteurs ?

III. OBJECTIFS DU PROJET

3.1 Objectif principal

L'objectif principal de ce mini projet est de concevoir et développer une application mobile de signalement des incidents permettant aux utilisateurs de déclarer rapidement et facilement un incident, et aux équipes concernées de le traiter, le suivre et le résoudre de manière structurée. Cette solution vise à améliorer la réactivité, la traçabilité et l'efficacité globale du processus de gestion des incidents, tout en garantissant la sécurité des données et une adaptation aux différents profils d'utilisateurs.

3.2 Objectifs spécifiques

Afin d'atteindre l'objectif principal, plusieurs objectifs spécifiques ont été définis :

- Centraliser l'ensemble des incidents dans une base de données unique afin de faciliter leur consultation et leur suivi
- Réduire les délais de prise en charge et de résolution des incidents grâce à une meilleure organisation et à des notifications automatiques
- Assurer une traçabilité complète de toutes les actions effectuées sur chaque incident (création, mise à jour, validation, résolution)
- Adapter les fonctionnalités et les interfaces en fonction des rôles des utilisateurs (reporter, intervenant, superviseur, administrateur)
- Garantir la sécurité, la confidentialité et l'intégrité des données à travers des mécanismes d'authentification et de contrôle d'accès

IV. TECHNOLOGIES UTILISÉES

La conception et le développement de la plateforme reposent sur un ensemble de technologies modernes et éprouvées, sélectionnées pour leur performance, leur flexibilité et leur compatibilité avec les applications mobiles multiplateformes. L'architecture technologique adoptée vise à garantir une expérience utilisateur optimale, une maintenance simplifiée et une évolutivité du système.

React Native avec Expo

React Native est un framework open-source développé par Meta, permettant de créer des applications mobiles natives pour iOS et Android à partir d'un seul code source en JavaScript.

L'utilisation conjointe d'Expo offre un environnement de développement simplifié et performant, incluant des outils de déploiement rapide, la gestion des assets et l'accès à des APIs natives préconfigurées. Cette combinaison permet d'accélérer le développement tout en assurant la portabilité et la cohérence de l'application sur différentes plateformes.

JavaScript

Le langage JavaScript constitue le socle de la logique applicative. Sa nature interprétée et sa large adoption dans le développement web et mobile permettent d'assurer la cohérence du code, la facilité d'intégration des bibliothèques tierces et la gestion efficace des interactions utilisateurs et des flux de données.

Firebase Authentication

Firebase Authentication a été intégré pour assurer la gestion sécurisée des comptes utilisateurs. Ce service fournit des mécanismes robustes pour l'inscription, la connexion et la réinitialisation des mots de passe, tout en supportant plusieurs méthodes d'authentification, telles que le courrier électronique, Google, Facebook et d'autres fournisseurs OAuth.

Firebase

Firestore

La base de données NoSQL Firestore assure le stockage et la récupération des données en temps réel. Elle permet la synchronisation automatique entre le backend et les clients, offrant ainsi une gestion efficace des informations dynamiques telles que les profils utilisateurs, les publications et les messages. Cette approche garantit la cohérence des données tout en simplifiant leur manipulation dans un environnement multi-utilisateur.

Firebase

Storage

Pour la gestion et le stockage des fichiers volumineux, notamment les images, Firebase Storage a été utilisé. Ce service cloud sécurisé permet de stocker et de récupérer les fichiers de manière fiable et scalable, assurant ainsi leur disponibilité et leur intégrité.

Zustand

Zustand est une bibliothèque légère de gestion d'état global pour les applications React et React Native. Elle a été adoptée pour sa simplicité et son efficacité, permettant aux composants de partager et de mettre à jour les données de manière réactive. Comparée à d'autres solutions plus complexes, Zustand favorise une meilleure performance et une syntaxe concise, particulièrement adaptée aux contraintes des applications mobiles.

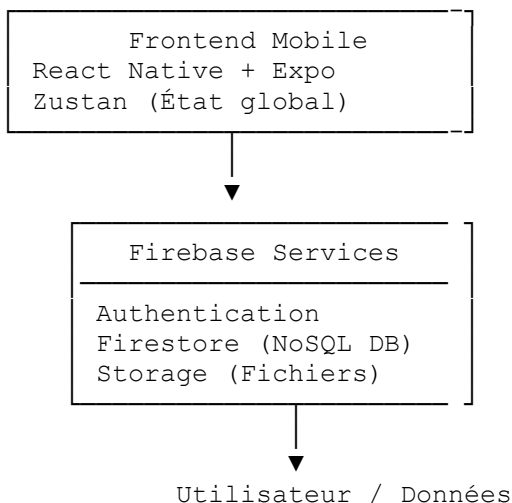
V. ARCHITECTURE TECHNOLOGIQUE DE LA PLATEFORME

L'architecture de la plateforme repose sur une structure modulaire permettant une communication fluide entre le frontend, le backend et les services cloud. Chaque composant a été sélectionné pour répondre à des besoins spécifiques en termes de performance, sécurité et évolutivité. Le schéma ci-dessous illustre l'organisation des différentes technologies au sein du système :

Description du schéma :

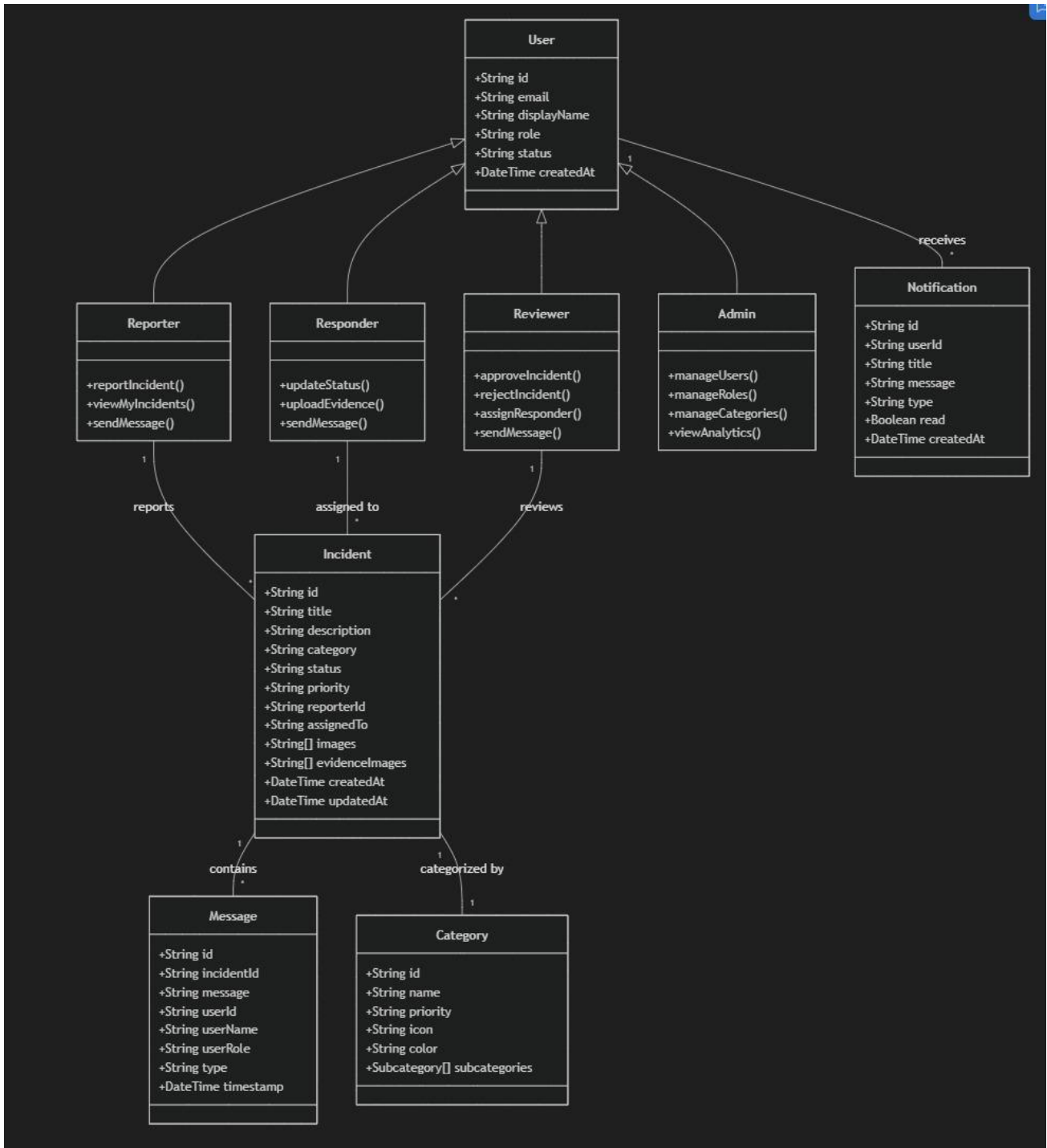
1. **Frontend Mobile** : Développé avec React Native et Expo, il constitue l'interface utilisateur et communique avec le backend via des appels API et Firebase.
2. **Gestion de l'état global** : Zustand est intégré au frontend pour gérer les données partagées entre les composants et assurer la réactivité de l'application.
3. **Backend / Services Cloud** :
 - **Firebase Authentication** : Gère l'authentification sécurisée des utilisateurs.
 - **Firebase Firestore** : Stocke les données structurées et synchronise en temps réel avec le frontend.
 - **Firebase Storage** : Permet le stockage et la récupération des fichiers volumineux, notamment les images.
4. **Communication et Flux de données** : Les données circulent entre le frontend et les services cloud via des requêtes API sécurisées et des synchronisations temps réel, garantissant cohérence et fiabilité.

Schéma (représentation conceptuelle) :



VI. CONCEPTION :

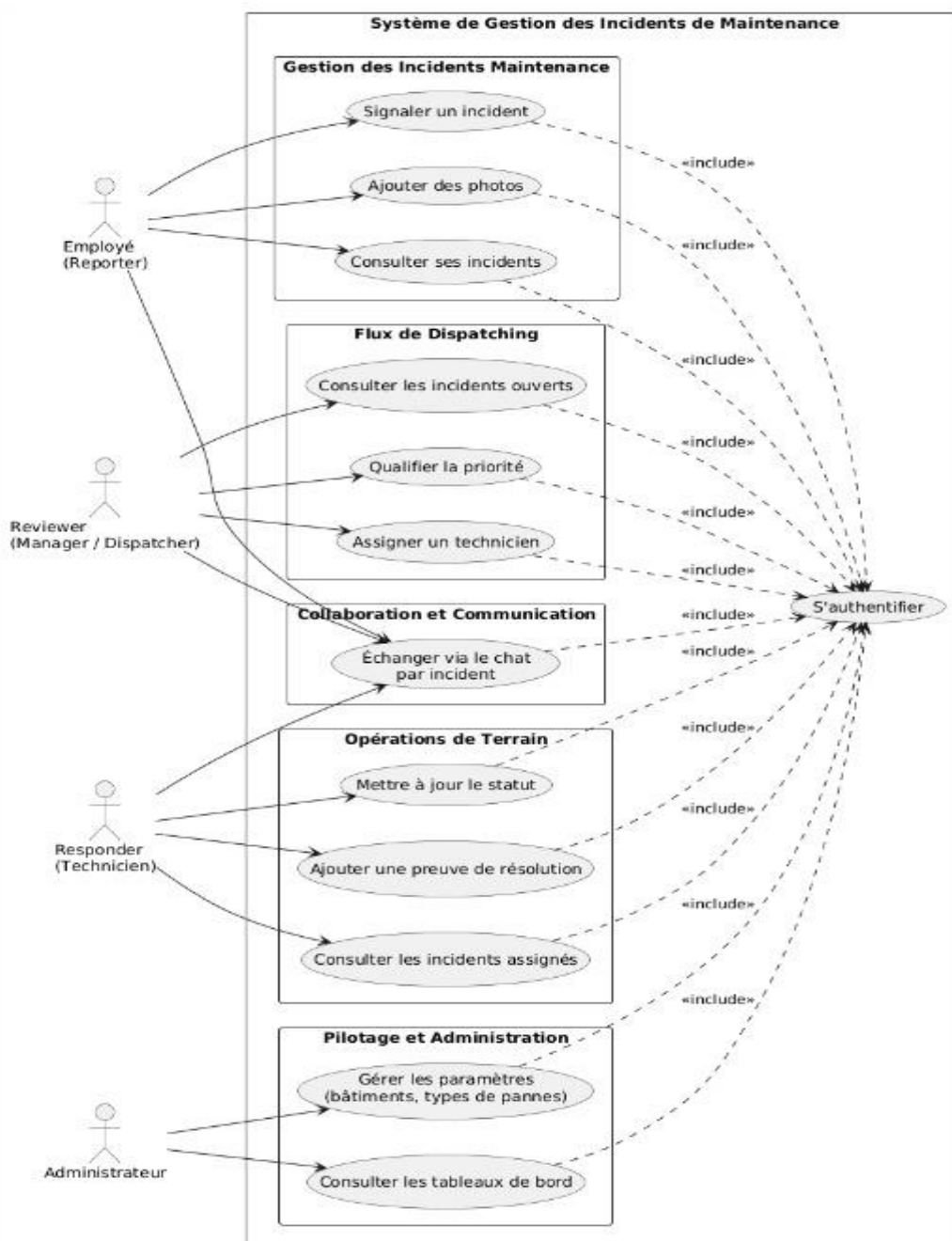
1. Diagramme de classes (Class Diagram)



Ce diagramme de classes représente l'architecture d'un système de gestion des incidents basé sur des rôles utilisateurs. La classe **User** est la classe centrale, contenant les informations communes (identifiant, email, rôle, statut, date de création) et servant de classe parente aux rôles **Reporter**, **Responder**, **Reviewer** et **Admin**, chacun disposant de responsabilités

spécifiques. Le **Reporter** peut signaler des incidents, les consulter et échanger des messages, tandis que le **Responder** est chargé de traiter les incidents qui lui sont assignés, de mettre à jour leur statut et d'ajouter des preuves. Le **Reviewer** intervient pour valider ou rejeter les incidents et assigner des responders, alors que l'**Admin** gère les utilisateurs, les rôles, les catégories et les statistiques du système. La classe **Incident** constitue le cœur fonctionnel du système : elle regroupe les informations relatives à un incident (description, statut, priorité, images, dates) et est liée au reporter, au responder et au reviewer. Chaque incident appartient à une **Category**, peut contenir plusieurs **Message** pour assurer la communication entre les acteurs, et génère des **Notification** afin d'informer les utilisateurs des événements importants. Ce diagramme met ainsi en évidence une structure claire, modulaire et orientée rôles, facilitant la gestion, le suivi et la communication autour des incidents dans l'application.

2. Diagramme des cas d'utilisation (Use Case Diagram)



Ce diagramme de cas d'utilisation illustre le fonctionnement global du **système de gestion des incidents de maintenance** ainsi que les interactions entre les différents acteurs et le système. Il met en évidence les rôles principaux impliqués dans le processus, à savoir l'**Employé (Reporter)**, le **Reviewer (Manager / Dispatcher)**, le **Responder (Technicien)** et l'**Administrateur**, chacun disposant de fonctionnalités spécifiques selon ses responsabilités.

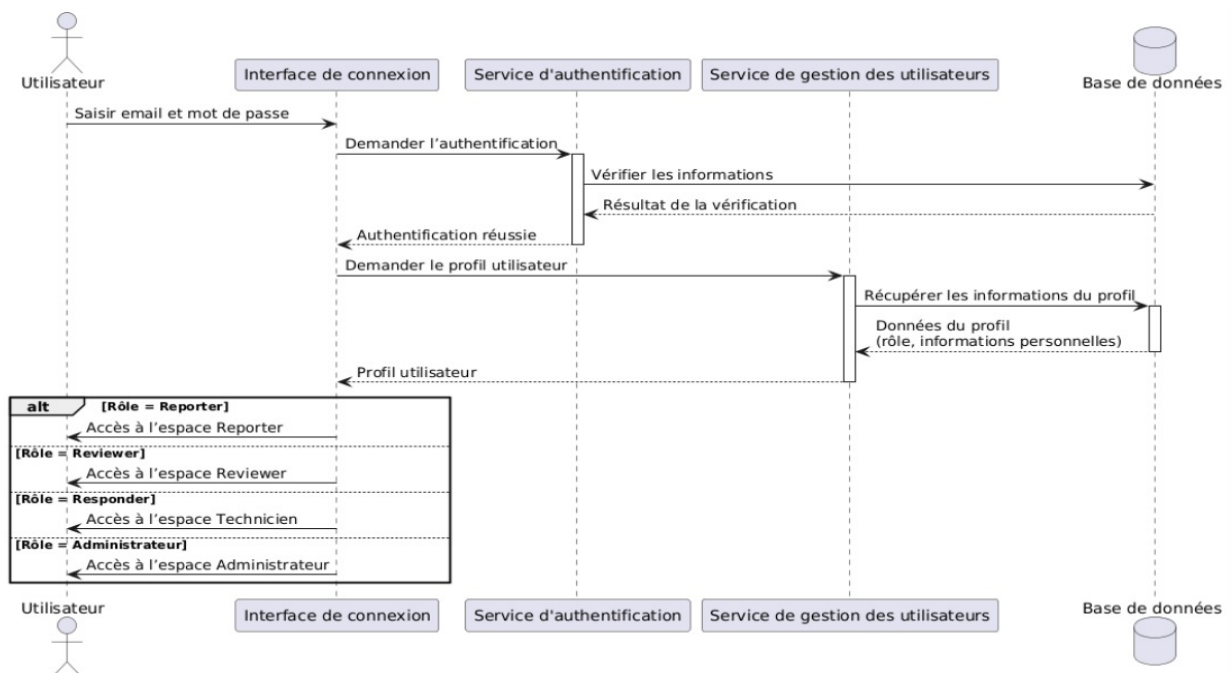
L'**Employé (Reporter)** est chargé de signaler les incidents de maintenance. Il peut déclarer un incident, y ajouter des photos afin de mieux décrire le problème rencontré, et consulter l'historique de ses incidents. Ces actions lui permettent de suivre l'évolution des demandes qu'il a émises. Le **Reviewer (Manager / Dispatcher)** intervient ensuite pour assurer le flux de dispatching : il consulte les incidents ouverts, qualifie leur priorité en fonction de leur criticité, puis assigne un technicien approprié pour leur prise en charge.

Le **Responder (Technicien)** est responsable des opérations de terrain. Il peut consulter les incidents qui lui sont assignés, mettre à jour leur statut au fur et à mesure de l'intervention, et ajouter des preuves de résolution (photos, commentaires) afin de confirmer la clôture de l'incident. La **collaboration et la communication** entre les différents acteurs sont assurées par un système de chat par incident, facilitant les échanges et la coordination tout au long du traitement.

Enfin, l'**Administrateur** dispose de fonctionnalités de pilotage et d'administration du système. Il peut gérer les paramètres globaux tels que les bâtiments et les types de pannes, ainsi que consulter des tableaux de bord permettant d'analyser les performances et l'état global des incidents. L'ensemble des cas d'utilisation inclut la fonctionnalité d'**authentification**, garantissant un accès sécurisé et adapté au rôle de chaque utilisateur. Ce diagramme met ainsi en évidence une organisation claire, structurée et orientée rôles du système, favorisant une gestion efficace et collaborative des incidents de maintenance.

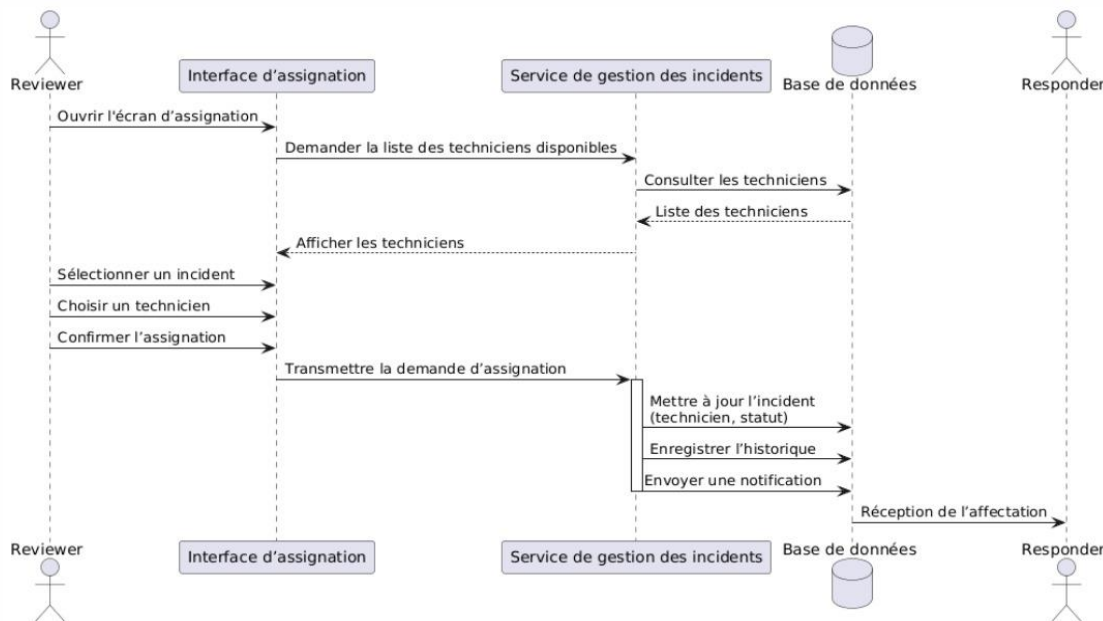
3. DIAGRAMME DE SÉQUENCE

- Authentification et Redirection



Ce diagramme de séquence décrit le **processus d'authentification et de redirection des utilisateurs** au sein de la plateforme de gestion des incidents. Il met en évidence les interactions entre l'utilisateur, l'interface de connexion (**LoginScreen**), les services d'authentification (**AuthService**), le service d'accès aux données (**FirestoreService**) et la base de données **Firestore**. Le processus débute lorsque l'utilisateur saisit son adresse e-mail et son mot de passe dans l'écran de connexion. Ces informations sont transmises au service d'authentification via la méthode *signInWithEmailAndPassword(email, pwd)*. Le service d'authentification procède alors à la vérification des identifiants auprès de Firestore. En cas de succès, un jeton d'authentification ainsi qu'un identifiant unique (UID) sont retournés, confirmant l'identité de l'utilisateur. Une fois l'utilisateur authentifié, l'application récupère les informations de son profil en appelant la méthode *getUserProfile(uid)*. Le service Firebase interroge la base de données Firestore afin d'obtenir le document utilisateur correspondant, contenant notamment le rôle, le nom et les autres informations nécessaires. Ces données sont ensuite renvoyées à l'application. Enfin, une structure conditionnelle (*alt*) permet de rediriger automatiquement l'utilisateur vers l'interface qui correspond à son rôle. Ainsi, un **Reporter** est orienté vers l'espace de signalement, un **Reviewer** vers l'interface de supervision et d'attribution, un **Responder** vers l'espace de traitement des incidents, et un **Admin** vers les interfaces d'administration. Ce mécanisme garantit un accès sécurisé et personnalisé aux fonctionnalités de la plateforme, en fonction des responsabilités de chaque utilisateur.

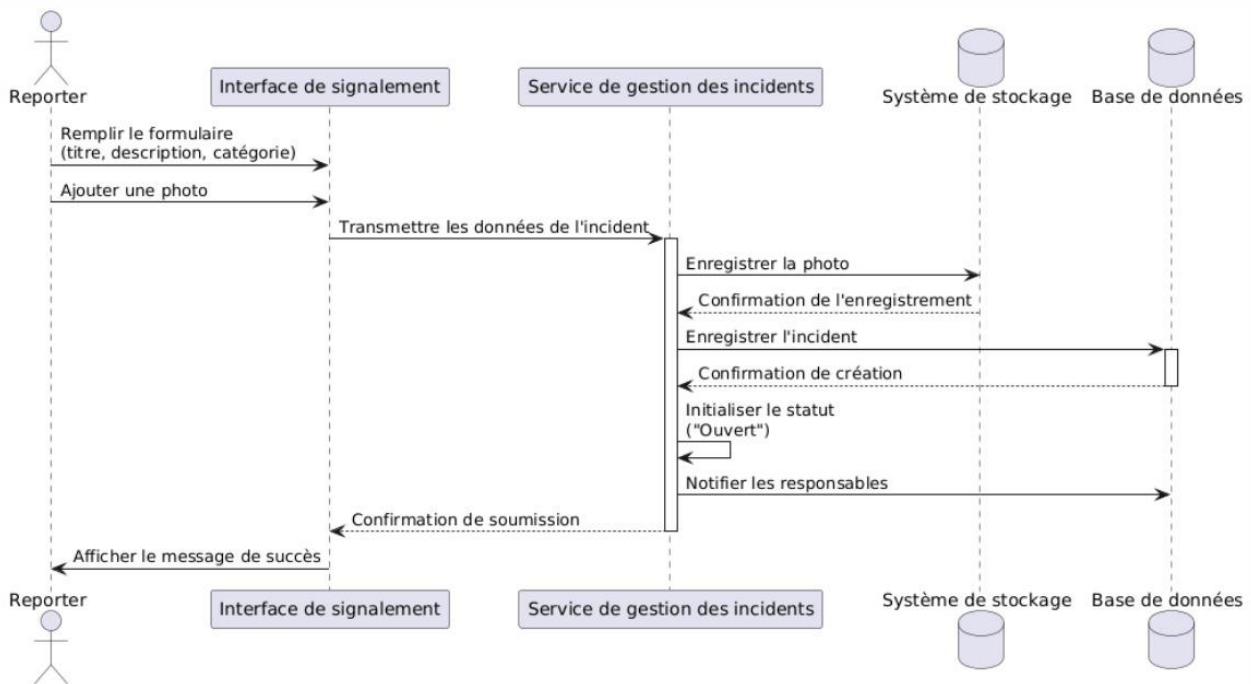
- Attribution d'un incident à un technicien



Ce diagramme de séquence détaille les interactions nécessaires à la réalisation de la fonctionnalité d'attribution d'un incident. Le processus s'initialise lorsque le **Reviewer** sollicite l'ouverture de l'interface d'attribution. Pour permettre une prise de décision éclairée, le système interroge automatiquement le **Service de gestion des incidents**, lequel communique avec la **Base de données** pour récupérer et afficher en temps réel la liste des techniciens disponibles. Cette étape garantit que l'administrateur dispose des ressources actualisées avant d'entamer toute action.

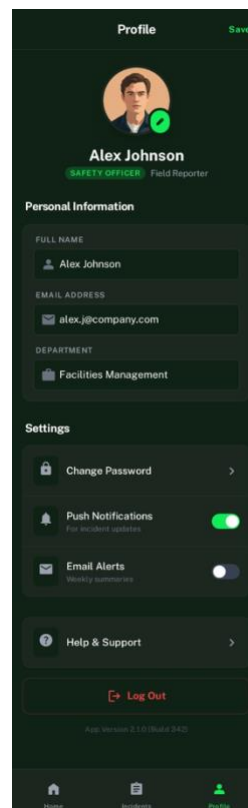
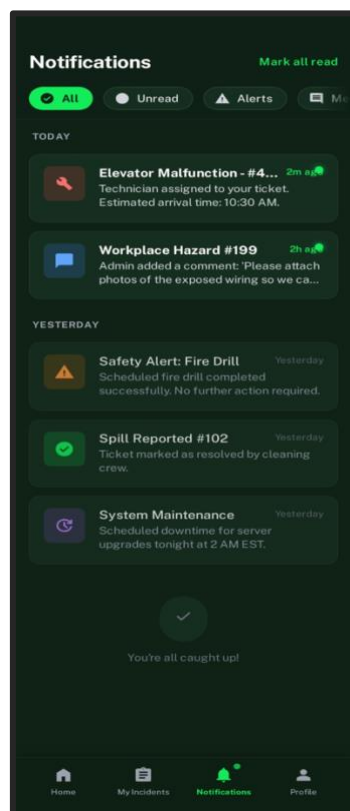
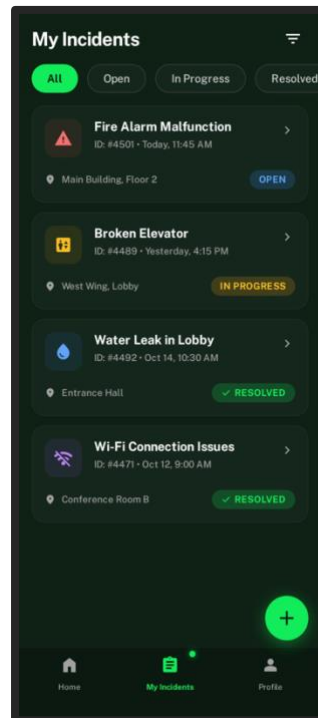
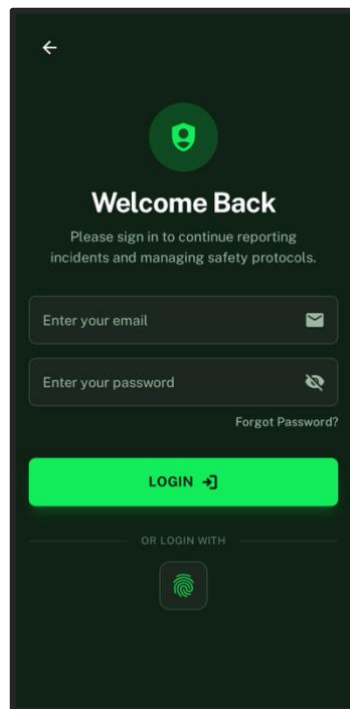
Dans un second temps, l'utilisateur procède à la sélection de l'incident et du technicien approprié, puis valide l'opération. Cette confirmation déclenche une série de traitements automatiques au niveau du serveur. Le service métier assure la mise à jour des informations de l'incident (statut et responsable), procède à l'archivage de l'action dans l'historique pour garantir la traçabilité, et génère enfin une notification. Le cycle se termine par la réception de l'affectation par le **Responder** (le technicien), bouclant ainsi le flux de communication entre la couche de gestion et l'exécution terrain.

- Création d'un Incident

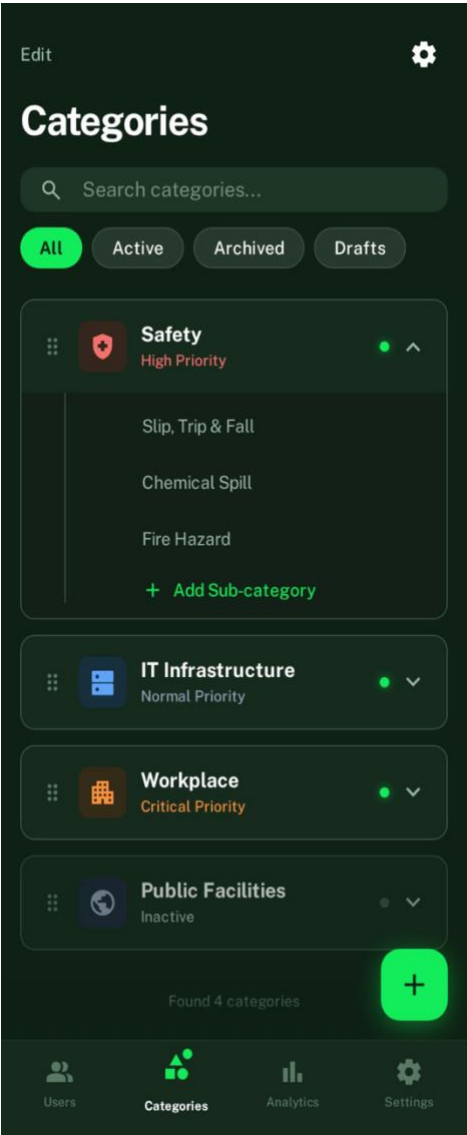
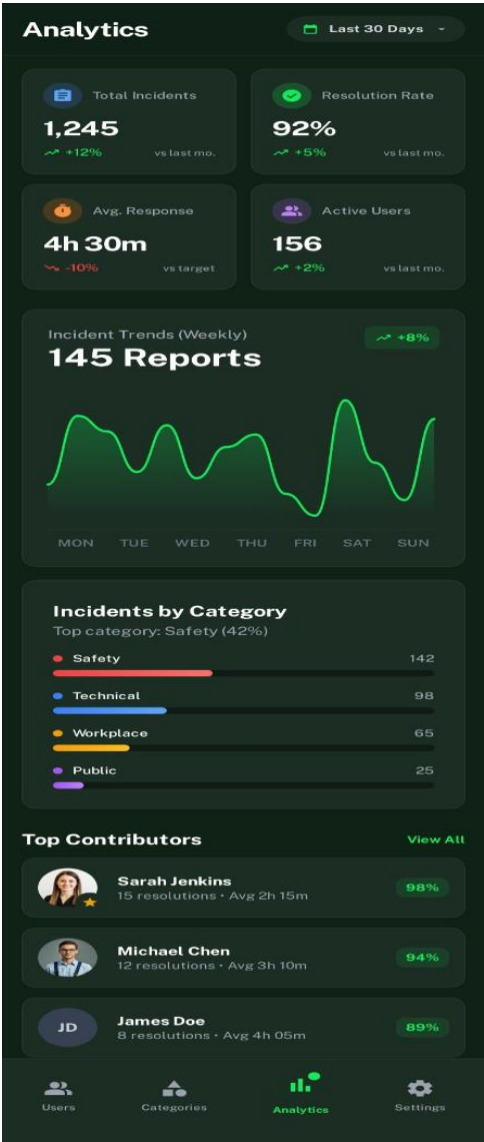


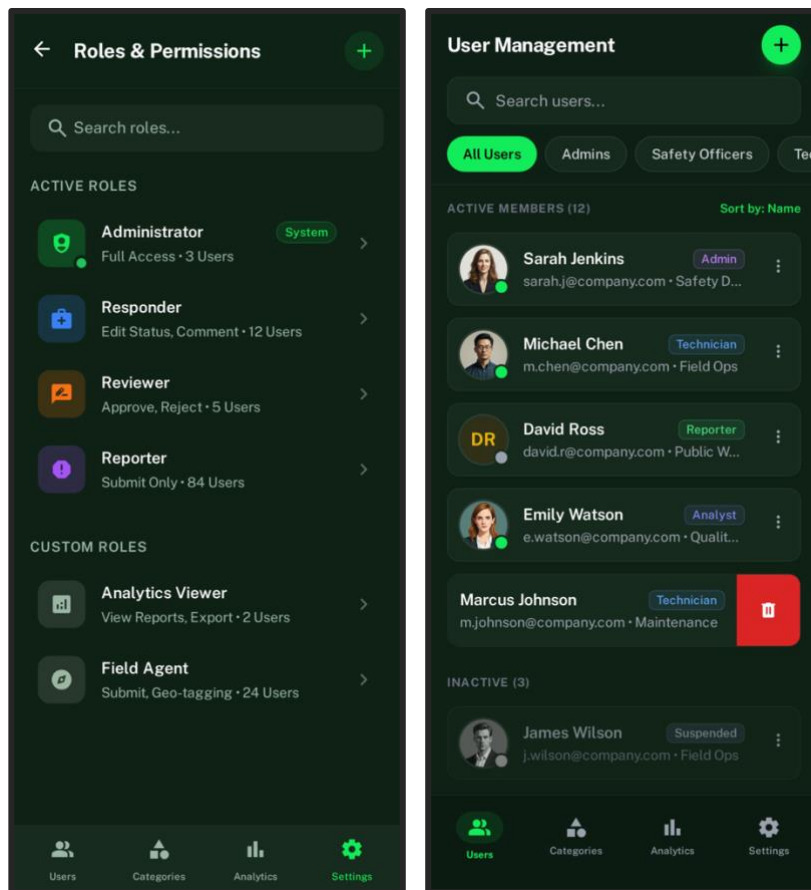
VII. RÉSULTATS ET RÉALISATIONS DE L'APPLICATION

- Responder



- Reporter :





21:21

Appareil p...

←

Create Account

Join to report and manage incidents effectively.

Full Name

👤

Email Address

✉️

Password

👁️

Confirm Password

👁️

Register as

👤

Reporter

🛡️

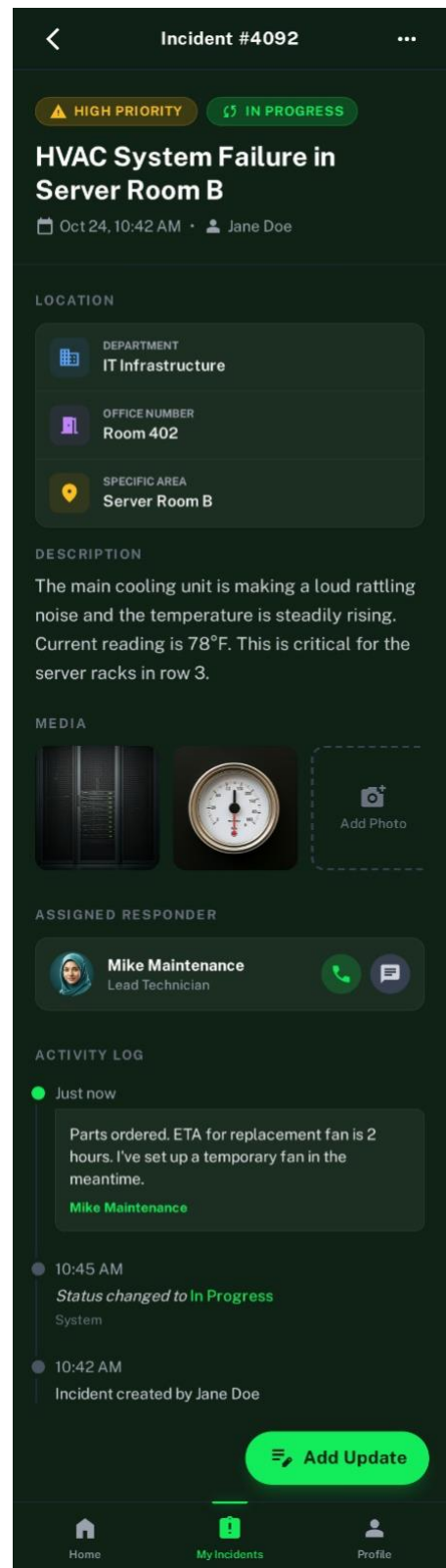
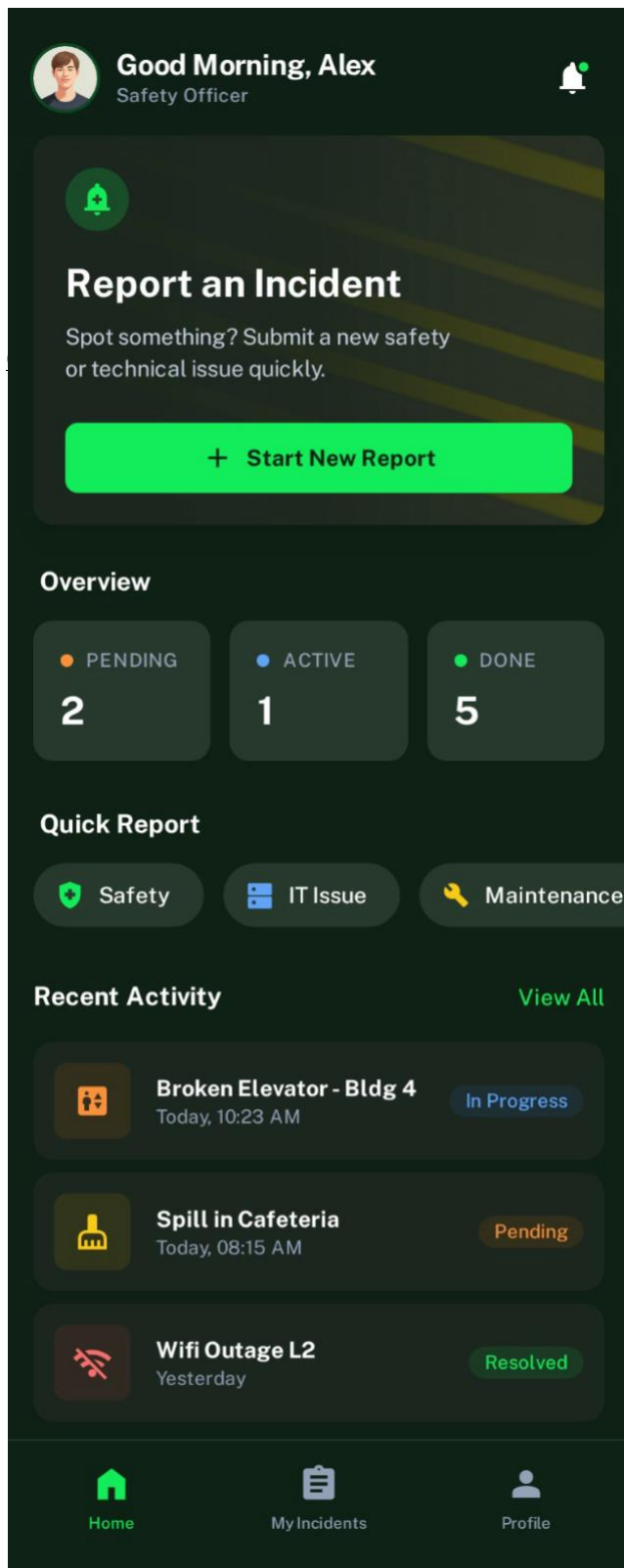
Reviewer

🚚

Responder

👑

Admin



Cancel

New Incident

Draft

i

CORE DETAILS

Incident Title

e.g., Water Leak in Hallway B

Category

Select Category

⌵

📄

WHAT HAPPENED?

Provide a detailed description of the incident, including who was involved and immediate actions taken...

📍

LOCATION

Department

Select Department

⌵

Office Number

Select Office

⌵

Specific Area

Select Area

⌵

📷

EVIDENCE

+

Add Media

🔑

×

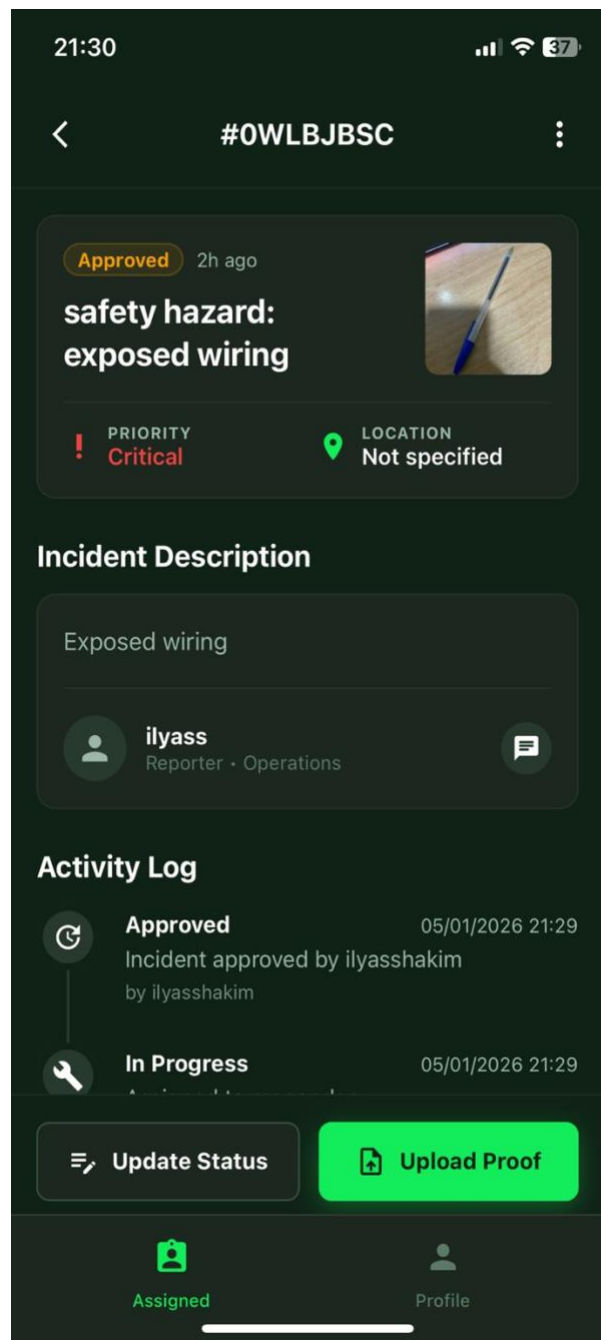
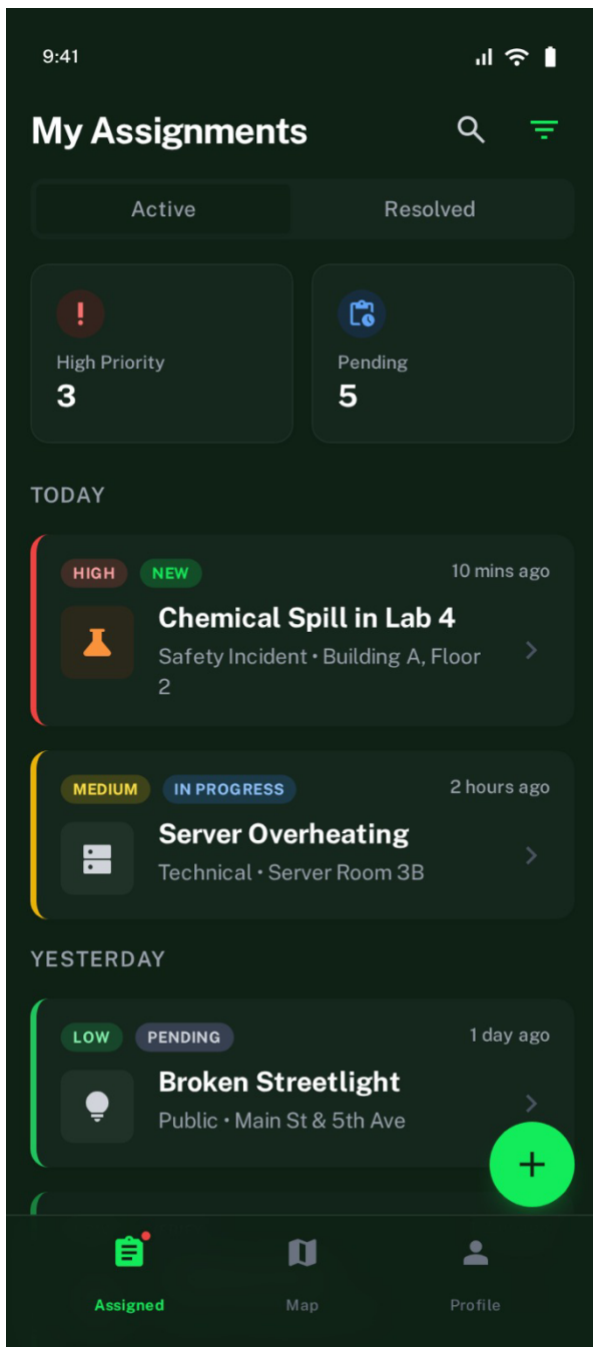
🖼️

×

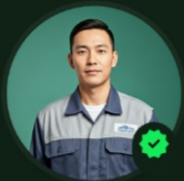
Submit Incident >

23

- Reviewer:



Profile



Alex Mercer


Unit 404 - Safety Technician

Availability Status

Go online to receive assignments


☐

ACCOUNT



Personal Information


>



Security & Password


>

PREFERENCES




Notifications

>



Location Settings

>



Help & Support

>

Log Out

App Version 2.1.0

Assigned

Map

Profile


×


Update Status

Save

Incident #4029

Broken HVAC Unit (Level 2)


 Building B, Floor 3



Current Status

☒ In Progress

Currently working on the fix



☐ Waiting for Resources


Parts or team unavailable

☐ Resolved

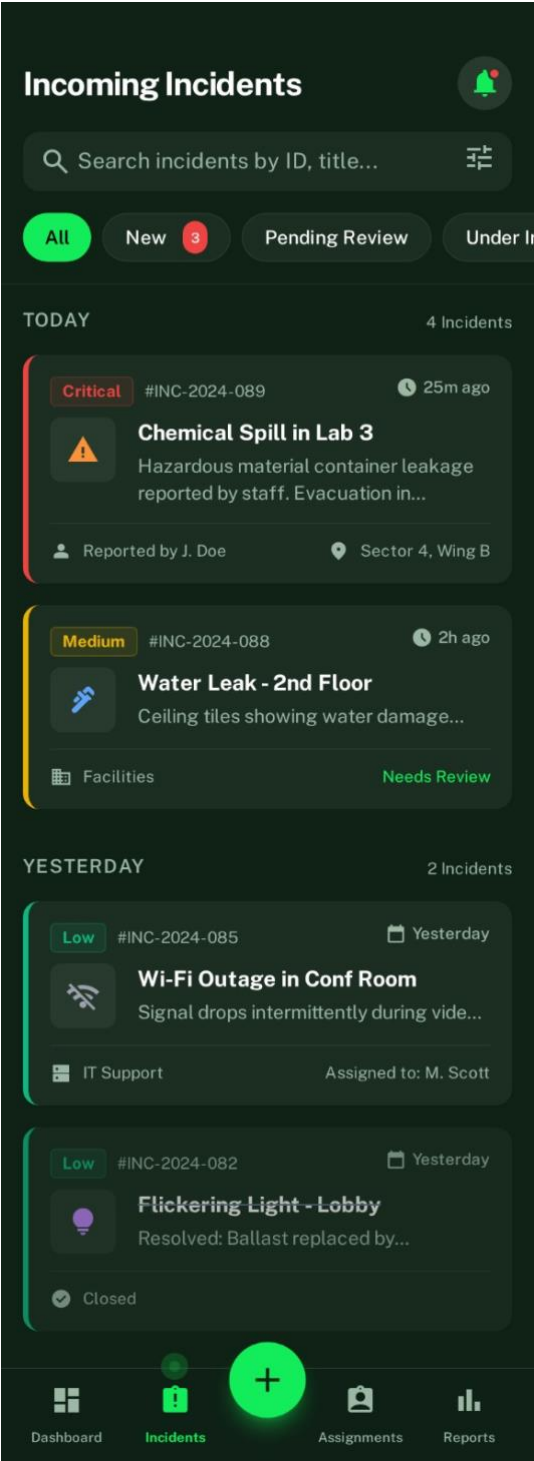
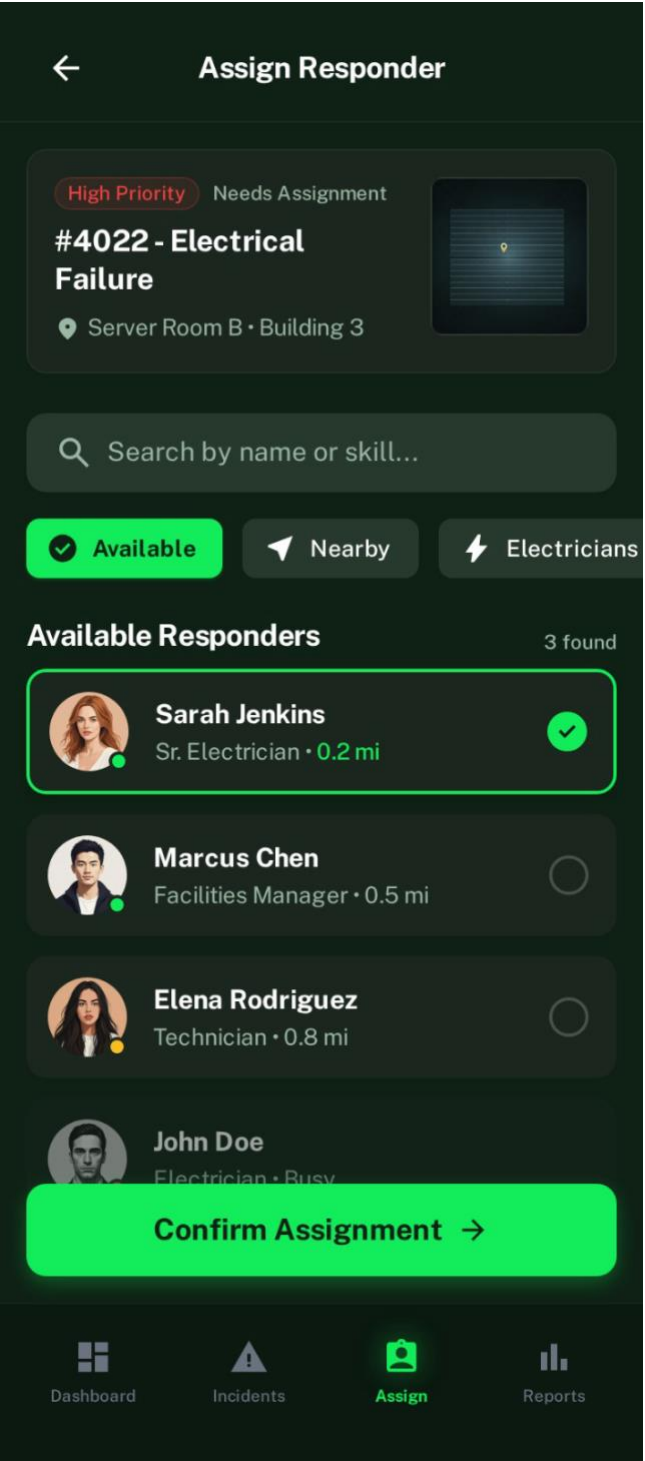
Issue fixed and verified

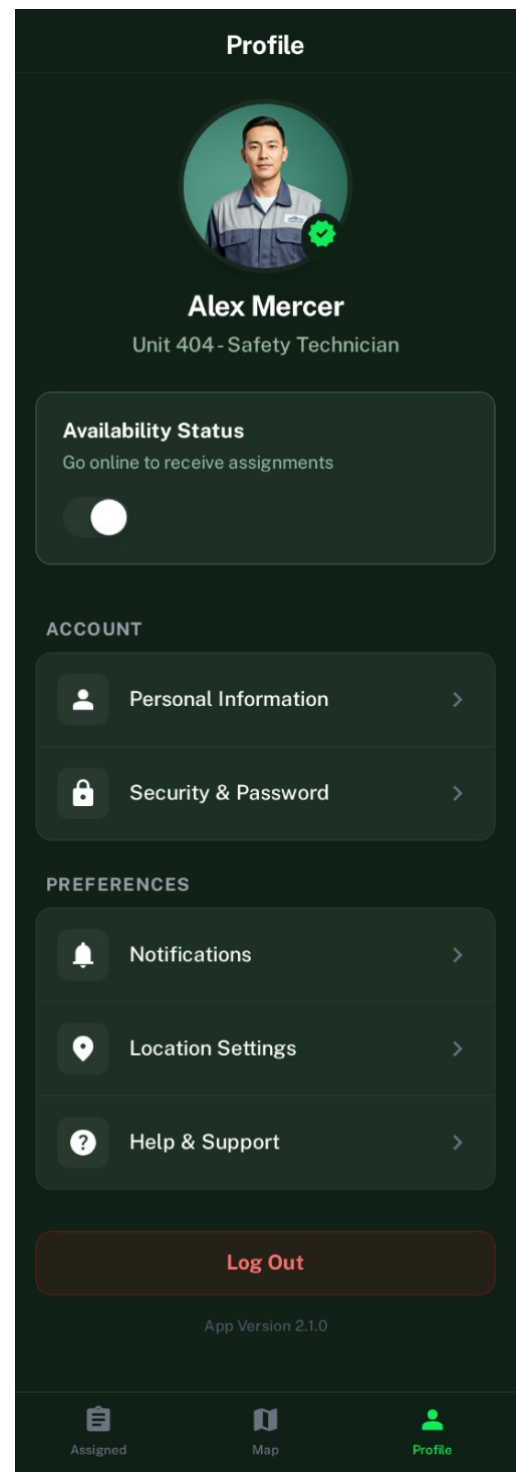
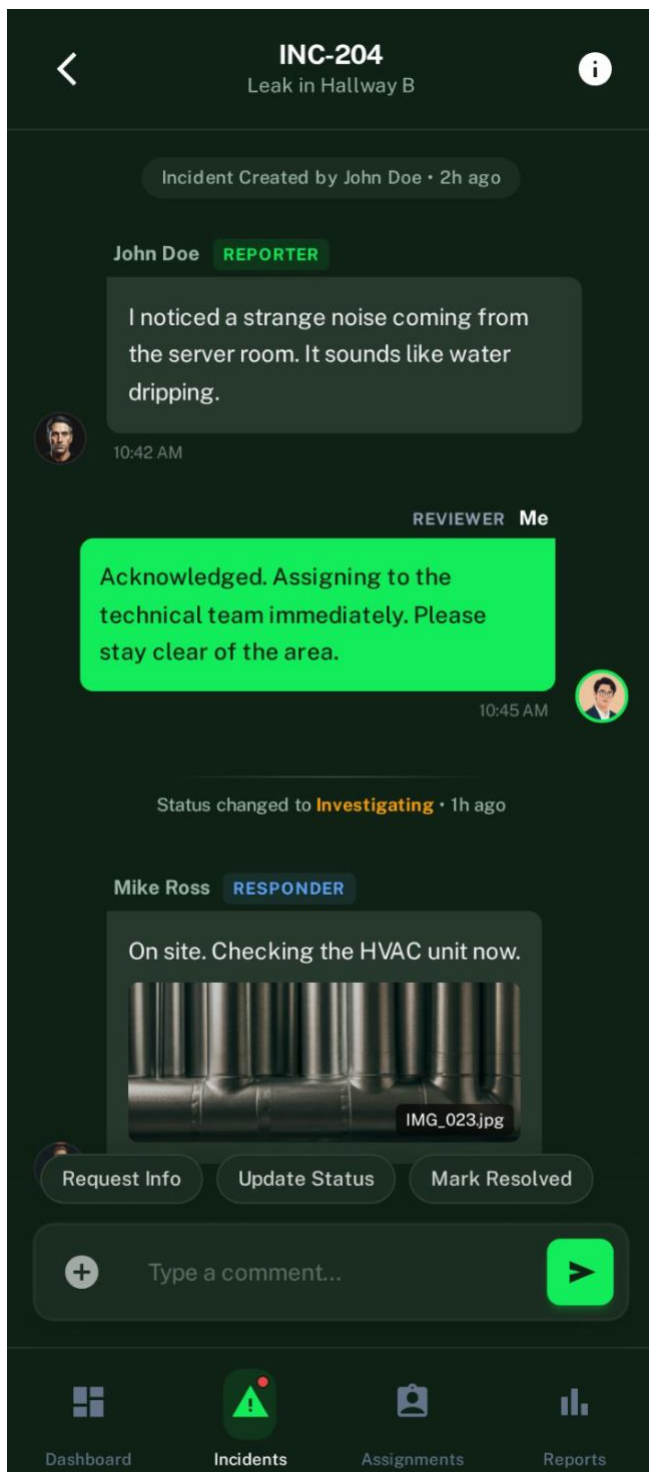
Add Note

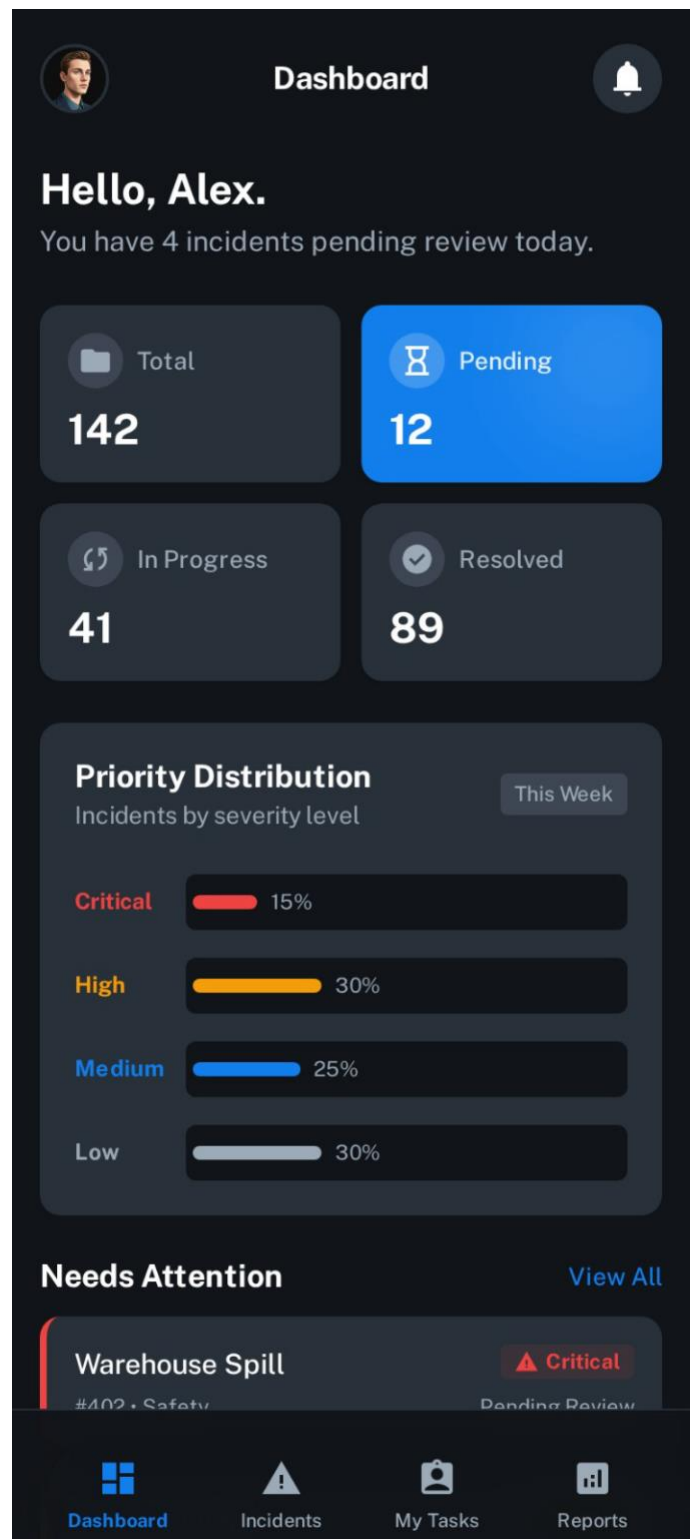
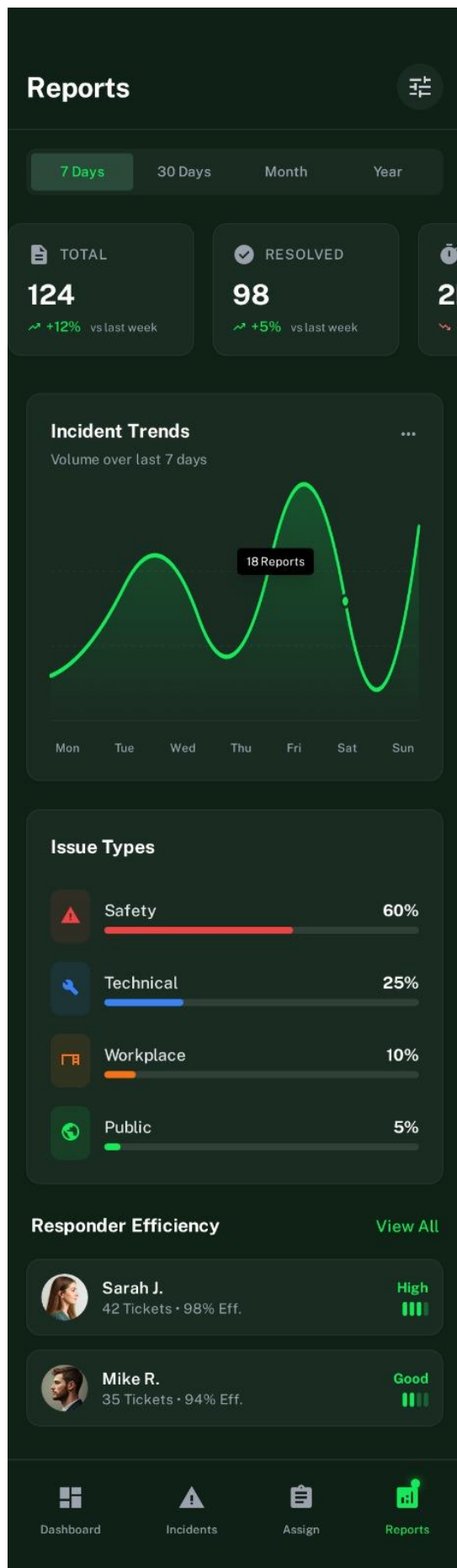
Describe the work performed or resources needed...

 Add Photo

0/500







#INC-2048


!

Pending Review

🚩

High Priority

Broken Hydraulic Press in Sector 4



Operator reported smoke and loud grinding noises coming from the main cylinder during the morning shift. Machine has been halted immediately to prevent further damage. Maintenance team has been notified but requires reviewer approval for replacement parts.

REPORTER

John Doe

TIME

Today, 09:42 AM

LOCATION

CATEGORY

Reject

Approve & Assign

Dashboard

Incidents

Assignments

Reports

Unassigned

! Priority Level

High

Activity History

● Incident Reported

Created by John Doe via Mobile App

2h ago

● System Flag

Auto-categorized as "Safety Incident" based on keywords.

2h ago

⚠ Warning: Multiple reports from Sector 4 this week.

● Pending Review

Waiting for reviewer action.

1h ago

VIII. CONCLUSION

Au terme de ce mini projet, il ressort que la conception et le développement d'une plateforme de signalement et de gestion des incidents constituent une réponse efficace aux problématiques liées à la dispersion de l'information, aux délais de traitement et au manque de coordination entre les différents acteurs. Le travail réalisé a permis de mettre en place une solution numérique centralisée, sécurisée et structurée, couvrant l'ensemble du cycle de vie d'un incident, depuis sa déclaration jusqu'à sa résolution finale.

L'application développée offre une interface mobile intuitive permettant aux utilisateurs de signaler rapidement des incidents, d'y joindre des informations pertinentes et de suivre leur évolution en temps réel. Grâce à une gestion des rôles bien définie (reporter, reviewer, responder et administrateur), la plateforme assure une répartition claire des responsabilités et un accès adapté aux fonctionnalités. L'architecture technologique adoptée, basée sur des services cloud et une séparation des couches, garantit la fiabilité, la traçabilité et la sécurité des données.

Les résultats obtenus démontrent une amélioration notable de la réactivité des équipes d'intervention, une meilleure communication entre les parties prenantes et une visibilité accrue sur l'état des incidents. Ce projet a également permis de consolider des compétences techniques et méthodologiques, notamment en conception UML, en développement mobile, en intégration de services cloud et en gestion de la sécurité.

En perspective, plusieurs axes d'amélioration peuvent être envisagés, tels que l'intégration de notifications en temps réel plus avancées, l'exploitation de tableaux de bord analytiques plus détaillés, l'ajout de la géolocalisation en temps réel ou encore l'utilisation de techniques d'intelligence artificielle pour la priorisation automatique des incidents. Ces évolutions permettraient d'enrichir davantage la plateforme et de renforcer son efficacité opérationnelle.

En conclusion, ce mini projet constitue une base solide pour le développement d'un système de gestion des incidents performant et évolutif, répondant aux besoins actuels des organisations tout en ouvrant la voie à de futures améliorations.