



Cahier des Charges Fonctionnel et Technique

Application mobile & web de gestion des alertes et des crises

□ Technologies : Ionic + Angular

□ 1. Présentation Générale du Projet

L'application permet à des individus de :

- **Signaler des alertes** (type d'alertes : Agression, Enlèvements, Exécution Sommaire, Harcèlement, Cambriolage, Accident de la route, Risque de catastrophe naturelles...).
 - **Suivre en temps réel les alertes** (incidents, crises, événements majeurs).
 - **Assigner des intervenants** à chaque alerte.
 - **Afficher une carte interactive** avec la géolocalisation des incidents.
 - **Générer des statistiques et rapports** d'analyse.
 - **Gérer les membres de l'équipe** et consulter les procédures d'urgence.
-

□ 2. Objectifs du Projet

- Permettre une **visualisation rapide et centralisée** des alertes.
 - Réduire les temps de réaction face aux des citoyens faces aux menaces et crises.
 - Fournir des **outils d'analyse** pour anticiper et gérer les situations à risques.
 - Permettre un **accès mobile et web multiplateforme** (Android, iOS, navigateur).
-

□ 3. Fonctionnalités

□ Page d'accueil / Tableau de bord

- Liste des alertes en cours, triées par gravité (critique, élevée, modérée, faible).
- Indicateurs visuels (cartes, compteurs, jauges).
- Nombre total d'alertes actives / résolues/ En cours/ Confirmées/ Fausse/ Archivées.
- Alertes classées par date et par type.
- Bouton de déclaration d'alerte visible en permanence

☐ Carte interactive

- Affichage de toutes les alertes sur une carte (Google Maps ou Leaflet/OpenStreetMap).
- Clic sur une alerte → infos : type, statut, date, localisation, personnes affectées.

☐ Gestion des alertes

- Diffusion des alertes à tous les intervenants
- Filtres par : type d'incident, gravité, date, statut.
- Détails de chaque alerte : description, date de début, statut (en cours, résolue, etc.), localisation, participants, documents liés.
- Option pour changer le statut d'une alerte. (S'il s'agit d'un super-utilisateur)

☐ Gestion des utilisateurs / intervenants

- Liste des utilisateurs : rôle, statut (actif, inactif), missions assignées.
- Attribution manuelle d'une alerte à un ou plusieurs intervenants.
- Suivi du statut de chaque intervenant (disponible, en mission).
- Validation des Super-Utilisateurs (PWA) *accessible uniquement à l'administrateur*

☐ Inscription / Authentification

- Inscription des utilisateurs : Informations personnelles (email, nom, prénom,..).
- Création de leur profil d'alerteur.
- Authentification par email et password/ Numéro de téléphone en MFA.

☐ Plan d'intervention / documentation

- Liste de protocoles prédéfinis par type de crise (incendie, attaques armée, etc.).
- Consultation rapide pendant une crise.
- Possibilité de filtrer par catégorie.

☐ Rapports & statistiques

- Graphiques : nombre d'alertes par période, par gravité, par localisation.
- Délai moyen de traitement des alertes.
- Export possible (PDF, CSV).

☐ 4. Structure Technique & Architecture

☐ Stack technique :

- **Frontend** : Ionic + Angular
- **Backend** : Node.js avec Firebase
- **Base de données** : Firebase Firestore, ou autre selon besoin (MongoDB, MySQL...)
- **Cartographie** : Google Maps ou Leaflet/OpenStreetMap
- **Notifications** : Firebase Cloud Messaging (FCM)

☐ Cible plateformes :

- Android
 - iOS
 - Web (PWA)
-

☐ 5. UX/UI & Design

- Design moderne, responsive et sobre.
 - Couleurs utilisées :
 - ☐ Rouge → alertes critiques
 - ☐ Orange → alertes élevées
 - ☐ Vert → alertes modérées ou résolues
 - ☐ Blanc/Gris → interface de fond sobre
 - Icônes intuitives pour chaque type d'alerte.
 - Navigation via **menu latéral (side menu)**.
-

☐ 6. Sécurité

- Authentification (login + mot de passe)
 - Rôles : administrateur, intervenant, superviseur
 - Sécurisation des données et communications (HTTPS, Firestore rules ou JWT)
-

☐ 7. Livrables attendus

- Documentation technique
- APK Android et fichier iOS (ou soumission à l'App Store/TestFlight)
- Manuel utilisateur simple (PDF ou Web)

□ 8. Planning prévisionnel

Étape	Durée estimée
Spécifications détaillées	3-5 jours
Maquettes / Design UI	2 semaines
Développement front (Ionic)	3-7 semaines
Intégration backend	1-3 semaines
Tests & corrections	1-3 semaines
Livraison finale	Après validation finale

□ 9. Équipe & Rôles

- **Chef de projet** (MajOrWare) : suit l'avancement, valide les livrables.
 - **Développeur Ionic Angular** : développe l'application.
 - **UX/UI designer** (*optionnel*) : propose des maquettes.
 - **Testeurs** : vérifient le bon fonctionnement.
-