

### INTRODUCTION

Le projet vise à concevoir et développer une application web pour les Associations caritatives fournissant une aide humanitaire, tels que des matelas, des vêtements, de la nourriture, etc., dans des zones reculées. L'objectif principal est de résoudre le problème des associations qui ont du mal à identifier les domaines non couverts par les subventions, en facilitant la coordination entre les associations, en apportant une aide ciblée, et en digitalisé le processus.

### FONCTIONALITE

#### Interface Admin :

- Gestion des associations
- Gestion des villes
- Gestion des dowars
- Gestion des Utilisateurs

#### Interface Association :

- Gestion des membres
- Gestion des biens essentiels (matelas, vêtements, nourriture...)
- voirStatistique des dowars
- Gestion des kafilas

### DESIGNE DE L'APPLICATION

- Conception d'interfaces utilisateur conviviales et intuitives pour les administrateurs, les associations , et aussi landing page pour demande de participation si membre ou bien association .
- Utilisation de principes de design UX pour garantir une expérience utilisateur optimale.

### Tests et déploiement

- Plan de tests exhaustif pour vérifier le bon fonctionnement de la application avant son déploiement.
- Déploiement sur un serveur sécurisé et fiable.

## Maintenance et évolutivité

Prévoir des mises à jour régulières pour améliorer les fonctionnalités existantes et ajouter de nouvelles fonctionnalités en fonction des retours des utilisateurs et des besoins émergents.

Garantir la maintenance continue de la plateforme pour assurer sa stabilité et sa performance.

## Technologies

### 1. Backend :

Framework Spring Boot pour le développement de l'application backend.

Spring Security pour la gestion de l'authentification, de l'autorisation et la sécurisation des Endpoint de l'application.

Base de données PostgreSQL pour le stockage des données. PostgreSQL est une base de données relationnelle robuste et largement utilisée, offrant des fonctionnalités avancées de gestion des données.

Utilisation de JPA (Java Persistence API) pour la gestion de la couche d'accès aux données et le mapping objet-relationnel.

### 2. Frontend :

Framework Angular pour le développement de l'interface utilisateur côté client.

Utilisation de Bootstrap pour le design et la mise en page responsives.

### 3. Déploiement :

Utilisation d'un serveur d'application tel qu'Apache Tomcat pour le déploiement de l'application backend.

Utilisation de services d'hébergement cloud comme AWS (Amazon Web Services) ou Google Cloud Platform pour le déploiement de l'application frontend et de la base de données PostgreSQL.

Utilisation de Docker pour la conteneurisation des applications et la gestion des environnements de développement, de test et de production.

### 4. Sécurité :

Intégration de JWT (JSON Web Token) avec Spring Security pour la gestion des sessions et l'authentification basée sur token pour front et back.

Configuration de mécanismes de sécurités avancés au niveau de la couche

frontale et de la couche backend pour prévenir les attaques et protéger les données sensibles.

## Contrainte de durée

- Le développement de la plateforme devra être complété avant la date limite du 24 mars 2024.
- Le délai total de développement est donc de 50 jours à compter de la date de début des travaux de développement.
- Un planning détaillé sera établi dès le début du projet, décomposant les différentes phases de développement, y compris la conception, la programmation, les tests, la mise en production, et la phase de correction des bugs.
- Une gestion de projet efficace sera mise en place pour s'assurer que les délais sont respectés et que les objectifs de développement sont atteints dans les temps impartis.
- Des réunions régulières seront organisées pour suivre l'avancement du projet, identifier les éventuels retards ou obstacles, et prendre les mesures correctives nécessaires pour respecter la date limite du 24 mars 2024