# **Cahier des charges : FoundIt!**

## **Une solution communautaire pour retrouver ce qui est perdu!**

## **Contexte du Projet**

### **Introduction au problématique :**

#### **L’histoire d’Ahmed :**

Un mercredi soir, après une longue journée à YouCode, Ahmed et son équipe se rendent au local du centre-ville pour finaliser leur projet. À 23h, épuisés mais fiers de leur travail, ils quittent précipitamment le lieu. Dans la précipitation, Ahmed oublie son chargeur sous une table. Quelques minutes plus tard, Anass, un autre étudiant, quitte le local et trouve le chargeur. Il poste un message sur Discord : "Chargeur trouvé ce soir au centre-ville – modèle USB-C noir. Si c’est le vôtre, contactez-moi !". Mais entre les memes et autres messages, l'annonce disparaît rapidement.

Le lendemain matin, Ahmed réalise son oubli. Il consulte Discord, mais ne voit pas l'annonce d'Anass. Il poste à son tour : "Urgent ! Chargeur perdu hier soir au centre-ville – modèle USB-C noir." Malheureusement, son message subit le même sort. Ce n’est que le soir, en retournant au centre-ville, qu’il croise Anass par hasard et apprend que son chargeur était entre de bonnes mains.

### **Le problème :**

* Les annonces sur Discord se perdent rapidement.
* Aucun système de priorité ou de géolocalisation pour les objets trouvés.
* Les étudiants ne savent pas où chercher les objets trouvés, même si une annonce existe.

### **Pourquoi Discord ne suffit pas ?**

* **Messages éphémères** : Les annonces disparaissent sous les discussions.
* **Manque de structure** : Pas de filtres par lieu, date ou catégorie.
* **Aucune visibilité centralisée** : Les objets trouvés ne sont pas facilement accessibles.

## **Solution proposée : FoundIt!**

### **Fonctionnalités clés :**

1. **Annonces structurées et géolocalisées**
   * Possibilité de déclarer une perte ou une trouvaille avec lieu et horaire.
   * Annonces automatiquement taguées par emplacement (campus, centre-ville).
2. **Notifications ciblées**
   * Notifications push ou SMS en cas de correspondance avec une annonce.
3. **Messagerie sécurisée**
   * Communication interne sans partage de coordonnées personnelles.
4. **Carte interactive** (Bonus)  
   * Visualisation des zones de pertes récurrentes.

## **Objectifs de la Plateforme**

* **Réduction des pertes** : Centraliser les déclarations d’objets.
* **Connectivité** : Mettre en relation les personnes concernées.
* **Soutien communautaire** : Renforcer l’entraide.
* **Accessibilité** : Interface intuitive et responsive.

## **Architecture Fonctionnelle**

### **Partie Utilisateur**

* **Accueil** : Présentation du projet, recherche rapide.
* **Inscription/Connexion** : Formulaire d’inscription, authentification.
* **Tableau de bord** : Gestion des annonces et de la messagerie.
* **Messagerie interne** : Communication entre utilisateurs.
* **Déclaration d’objets** : Ajout d’annonces avec filtres.
* **Recherche avancée** : Tri par localisation, catégorie, date.
* **Notifications** : Alertes sur correspondances.

### **Partie Administrateur**

* **Panneau de gestion** : Gestion des annonces et utilisateurs.
* **Modération** : Surveillance des annonces et commentaires.
* **Statistiques et rapports** : Suivi de l'activité.

## **Gestion des risques**

1. **Fausse déclarations**
   * Vérification des comptes utilisateurs via email, téléphone ou carte d'étudiant.
   * Système de signalement et modération.
   * Possibilité de bannir les utilisateurs faisant de fausses déclarations.
2. **Abus sur la messagerie** 
   * Signaler les messages offensants.
   * Possibilité de bloquer et signaler un utilisateur.
3. **Adoption de la plateforme**
   * Campagne de sensibilisation au sein de YouCode.
   * Notifications et rappels aux utilisateurs.

## **Conformité et Sécurité**

* **Protection des données personnelles** : Stockage sécurisé et chiffrement des informations sensibles.
* **Conformité RGPD** : Les utilisateurs ne peuvent pas supprimer leurs données, mais celles-ci seront protégées.
* **Prévention des abus** : Signaler les utilisateurs faisant de fausses déclarations.

## **Charte Graphique**

* **Palette de couleurs** : Blanc, Blue Vert.
* **Typographie** : Inter.
* **Composants graphiques** : Boutons, icônes, cartes interactives.

## **Rétroplanning & Timeline**

### **Phase 1 : Analyse et Planification (1 semaine)**

* Définition des besoins et architecture technique.
* Conception des maquettes (Figma/Adobe XD).

### **Phase 2 : Conception et Développement (7 semaines)**

* Backend : Laravel (authentification, API, base de données).
* Frontend : React (composants réutilisables, intégration des API).
* Intégration base de données (CRUD annonces et utilisateurs).
* Implémentation messagerie en temps réel (Pusher).

### **Phase 3 : Tests et Validation (1,5 semaine)**

* Tests utilisateurs, correction de bugs, vérifications de sécurité.

### **Phase 4 : Livraison Finale et Déploiement (1,5 semaine)**

* Ajustements finaux et déploiement sur AWS.

## **Stack Technologique**

* **Backend** : Laravel
* **Base De Données** : Postgre
* **Messagerie en temps réel** : Pusher
* **Frontend** : React
* **Hébergement** : AWS

## **Sécurité**

* Protection contre les attaques XSS et CSRF.
* Authentification par mot de passe haché.
* Surveillance et modération des contenus.