

Cahier des Charges

Projet OrphanCare – Gestion des orphelins et des dons

Table des Matières

1. Présentation du projet
2. Problèmes à résoudre / besoins exprimés
3. Utilisateurs cibles
4. Fonctionnalités attendues
5. Contraintes techniques
6. Rôles et permissions
7. Lien avec les technologies
8. Livrables
9. User Stories
10. Priorisation des fonctionnalités
11. Tâches à réaliser
12. Estimations de durée
13. Outils utilisés

1. Présentation du projet

Nom du projet : OrphanCare – Gestion des orphelins et des dons

Brève description : Application web destinée à centraliser les données des orphelins, les dons, le suivi et les notifications pour une association caritative.

Objectif principal : Offrir un outil structuré, sécurisé et accessible aux responsables de l'association.

2. Problèmes à résoudre / besoins exprimés

Problèmes :

- Difficulté à gérer les données des orphelins
- Manque de suivi personnalisé des dons
- Absence de notifications et rappels

Besoins :

- Centralisation dans une seule plateforme
- Historique des dons et lien avec les utilisateurs
- Remarques médicales/scolaires
- Notifications automatiques par email

3. Utilisateurs cibles

- **Administrateur** : Gère l'application, les utilisateurs, les données et les notifications
- **Employé** : Gère les orphelins, enregistre les dons, ajoute des remarques

4. Fonctionnalités attendues

Administrateur :

- Gérer les utilisateurs et leurs rôles
- Gérer les orphelins et les dons
- Filtrer les dons, recevoir des notifications

Employé :

- Ajouter/modifier des orphelins
- Enregistrer des dons, ajouter des remarques
- Rechercher et filtrer les données

5. Contraintes techniques

- **Backend** : Laravel (PHP)
- **Frontend** : React (remplaçant Blade)
- **Base de données** : MySQL

- **Hébergement** : VPS ou mutualisé (OVH, Forge...)
- **Responsive** : Compatible desktop, mobile, tablette

6. Rôles et permissions

Administrateur :

- Gestion des utilisateurs, orphelins, dons
- Accès complet au tableau de bord et notifications

Employé :

- Ajouter/éditer orphelins et dons
- Ajouter des remarques
- Rechercher et consulter les données

7. Lien avec les technologies

Laravel :

- Authentification sécurisée
- Middleware (isAdmin, isEmployee)
- Gates/Policies

React :

- Composants conditionnels selon les rôles
- Routes privées avec vérification de l'authentification
- Contexte global pour la gestion de session

8. Livrables

- Application web fonctionnelle avec interface React
- Gestion complète des utilisateurs, orphelins et dons
- Notifications email
- Tableau de bord pour les statistiques

9. User Stories

- En tant qu'employé, je veux ajouter un orphelin pour mettre à jour la base.
- En tant qu'administrateur, je veux gérer les comptes utilisateurs.
- En tant qu'employé, je veux ajouter une note de suivi à un orphelin.
- En tant qu'administrateur, je veux recevoir des notifications lors des dons.

10. Priorisation des fonctionnalités

Essentielles :

- Authentification et rôles
- Gestion des orphelins, dons, remarques

Secondaires :

- Notifications email
- Statistiques et tableau de bord

Bonus :

- Interface publique pour les donateurs

11. Tâches à réaliser

Phase 1 – Analyse & Conception (8h) : Cahier des charges, conception MCD, wireframes

Phase 2 – Backend Laravel (20h) : Auth, CRUD, sécurité

Phase 3 – Frontend React (12h) : Composants, routes, responsive

Phase 4 – Tests & Déploiement (8h)

12. Estimations de durée

Total estimé : 52h + 8h de marge

13. Outils utilisés

- **Langages** : Laravel, React
- **BDD** : MySQL
- **IDE** : VS Code
- **Design** : Figma
- **Tests** : Postman
- **Contrôle de version** : GitHub