

# ***MediManager - Application de Gestion des Patients***

**Iyed Kadri**  
**Rani Charradi**  
**Youssef Zaghouuni**  
**TP3**

## **Description du Projet**

**MediManager** est une application Android native développée pour faciliter la gestion quotidienne des patients d'un cabinet médical. Elle permet au médecin de gérer efficacement les informations des patients, leurs consultations et leurs rendez-vous via une interface intuitive et moderne.

## **Objectifs du Projet**

- Centraliser les informations médicales des patients
- Suivre l'historique des consultations
- Gérer les rendez-vous avec système de notifications
- Offrir une interface simple et rapide d'utilisation
- Garantir la persistance des données via SQLite

## **Cahier des Charges Fonctionnel**

### **1 Gestion des Patients (CRUD Complet)**

**\*\*Fonctionnalités :\*\***

- Ajouter un nouveau patient avec informations complètes
- Modifier les informations d'un patient existant
- Supprimer un patient (avec dialogue de confirmation)
- Rechercher un patient par nom/prénom
- Afficher la liste complète des patients
- Consulter les détails complets d'un patient

**\*\*Informations patient :\*\***

- Nom et Prénom
- Date de naissance (calcul automatique de l'âge)
- Genre (Homme/Femme/Autre)
- Téléphone
- Email
- Adresse complète
- Groupe sanguin (A+, A-, B+, B-, AB+, AB-, O+, O-)
- Allergies connues
- Date de création du dossier

## **2** Gestion des Consultations

**\*\*Fonctionnalités :\*\***

- Ajouter une consultation pour un patient
- Voir l'historique complet des consultations
- Modifier une consultation existante
- Supprimer une consultation
- Filtrer les consultations par date

**\*\*Informations consultation :\*\***

- Patient concerné (liaison avec table patients)
- Date de la consultation
- Diagnostic médical
- Traitement prescrit
- Ordonnance détaillée
- Notes complémentaires du médecin
- Date d'enregistrement

## **3** Gestion des Rendez-vous

**\*\*Fonctionnalités :\*\***

- Planifier un nouveau rendez-vous
- Modifier un rendez-vous existant
- Annuler un rendez-vous
- Marquer un rendez-vous comme "complété"
- Voir les rendez-vous du jour
- Voir les rendez-vous de la semaine
- Voir tous les rendez-vous
- Notifications pour les rendez-vous à venir

**\*\*Informations rendez-vous :\*\***

- Patient concerné
- Date du rendez-vous
- Heure du rendez-vous
- Motif de consultation
- Statut (Programmé/Complété/Annulé)
- Notes
- Date de création

## **4** Dashboard & Statistiques

**\*\*Affichage sur l'écran principal :\*\***

-  Nombre total de patients enregistrés
-  Nombre de consultations du mois en cours
-  Nombre de rendez-vous à venir
-  Liste des patients récemment ajoutés

## Base de Données SQLite

\*\*Structure des Tables\*\*

Table `patients`

```
CREATE TABLE patients (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    first_name TEXT NOT NULL,
    last_name TEXT NOT NULL,
    date_of_birth TEXT,
    gender TEXT,
    phone TEXT,
    email TEXT,
    address TEXT,
    blood_group TEXT,
    allergies TEXT,
    created_at TEXT DEFAULT CURRENT_TIMESTAMP
);
```

Table `consultations`

```
CREATE TABLE consultations (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    patient_id INTEGER NOT NULL,
    consultation_date TEXT NOT NULL,
    diagnosis TEXT,
    treatment TEXT,
    prescription TEXT,
    notes TEXT,
    created_at TEXT DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (patient_id) REFERENCES patients(id) ON DELETE CASCADE
);
```

Table `appointments`

```
CREATE TABLE appointments (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    patient_id INTEGER NOT NULL,
    appointment_date TEXT NOT NULL,
    appointment_time TEXT NOT NULL,
    reason TEXT,
    status TEXT DEFAULT 'scheduled',
    notes TEXT,
    created_at TEXT DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (patient_id) REFERENCES patients(id) ON DELETE CASCADE
);
```

## Interfaces Utilisateur (XML Layouts)

### **\*\*Activités Principales\*\***

<b>Fichier XML   Description  </b>
`activity_main.xml`   Écran d'accueil avec dashboard, liste des patients et barre de recherche
`activity_add_patient.xml`   Formulaire d'ajout/modification d'un patient
`activity_patient_details.xml`   Détails complets d'un patient avec onglets consultations/rendez-vous
`activity_add_consultation.xml`   Formulaire d'ajout/modification de consultation
`activity_add_appointment.xml`   Formulaire de planification de rendez-vous
`activity_appointments_list.xml`   Liste des rendez-vous avec filtres (jour/semaine/tous)

### **\*\*Layouts RecyclerView (Items)\*\***

<b>Fichier XML   Description  </b>
`item_patient.xml`   Card affichant un patient (nom, âge, téléphone)
`item_consultation.xml`   Card affichant une consultation (date, diagnostic)
`item_appointment.xml`   Card affichant un rendez-vous (date, heure, statut)

### **\*\*Dialogs\*\***

<b>Fichier XML   Description  </b>
`dialog_confirm_delete.xml`   Confirmation avant suppression
`dialog_date_picker.xml`   Sélecteur de date
`dialog_filter.xml`   Filtres avancés pour les listes

## Architecture du Projet

com.votrenom.medimanager/

```
    └── activities/
        ├── MainActivity.java      # Écran principal avec dashboard
        ├── AddPatientActivity.java # Ajout/Modification patient
        ├── PatientDetailsActivity.java # Détails d'un patient
        ├── AddConsultationActivity.java # Ajout/Modification consultation
        ├── AddAppointmentActivity.java # Planification rendez-vous
        └── AppointmentsListActivity.java # Liste des rendez-vous

    └── adapters/
        ├── PatientAdapter.java     # Adapter RecyclerView patients
        ├── ConsultationAdapter.java # Adapter RecyclerView consultations
        └── AppointmentAdapter.java # Adapter RecyclerView rendez-vous

    └── database/
        ├── DatabaseHelper.java     # Gestion SQLite et création tables
        ├── PatientDAO.java         # Opérations CRUD patients
        ├── ConsultationDAO.java    # Opérations CRUD consultations
        └── AppointmentDAO.java     # Opérations CRUD rendez-vous

    └── models/
        ├── Patient.java            # Modèle de données Patient
        ├── Consultation.java       # Modèle de données Consultation
        └── Appointment.java        # Modèle de données Rendez-vous

    └── utils/
        ├── Constants.java          # Constantes de l'application
        ├── DateUtils.java           # Utilitaires pour gestion dates
        └── ValidationUtils.java     # Validation des formulaires
```

## Charte Graphique

**\*\*Couleurs Principales (Theme Médical)\*\***

```
<!-- res/values/colors.xml -->
<resources>
    <!-- Couleurs principales -->
    <color name="primary">#1976D2</color>      <!-- Bleu médical -->
    <color name="primary_dark">#1565C0</color>   <!-- Bleu foncé -->
    <color name="accent">#4CAF50</color>       <!-- Vert santé -->

    <!-- Backgrounds -->
    <color name="background">#F5F5F5</color>     <!-- Gris clair -->
    <color name="card_background">#FFFFFF</color> <!-- Blanc -->

    <!-- Textes -->
    <color name="text_primary">#212121</color>   <!-- Noir -->
    <color name="text_secondary">#757575</color> <!-- Gris -->

    <!-- Statuts rendez-vous -->
    <color name="status_scheduled">#FF9800</color> <!-- Orange -->
    <color name="status_completed">#4CAF50</color> <!-- Vert -->
    <color name="status_cancelled">#F44336</color> <!-- Rouge -->
</resources>
```

## Spécifications Techniques

### **\*\*Configuration Minimale\*\***

- **SDK Minimum :** Android 5.0 (API 21)
- **SDK Cible :** Android 14 (API 34)
- **Langage :** Java
- **Base de données :** SQLite
- **Architecture :** MVC (Model-View-Controller)

### **\*\*Permissions Nécessaires\*\***

```
```xml
<!-- AndroidManifest.xml -->
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.SCHEDULE_EXACT_ALARM" />
<uses-permission android:name="android.permission.POST_NOTIFICATIONS" />
```

### **\*\*Bibliothèques Recommandées\*\***

- **RecyclerView :** Liste performante des patients
- **CardView :** Affichage en cartes
- **Material Components :** Interface moderne
- **FloatingActionButton :** Bouton d'ajout rapide

## Critères de Validation

### **\*\*Tests Fonctionnels\*\***

- Ajout d'au moins 5 patients avec données complètes
- Modification et suppression sans erreurs
- Recherche fonctionnelle et rapide
- Consultations correctement liées aux patients
- Rendez-vous avec notifications fonctionnelles
- Statistiques dashboard correctes
- Pas de crash lors de la navigation
- Données persistantes après fermeture de l'app

### **\*\*Tests Techniques\*\***

- Base de données créée correctement
- Relations Foreign Key fonctionnelles
- Suppression en cascade opérationnelle
- Validation des champs obligatoires
- Gestion des erreurs (try-catch)
- Pas de fuite mémoire
- Interface responsive sur différentes tailles d'écran