# Medis Care

# (Medicine Management System)



# Session 2023 - 2027

# Submitted by:

Munees Tariq 2023-CS-32

# Supervised by:

Prof. Khaldoon Khurshid

# Course:

Advanced Database

# Department of Computer Science

# University of Engineering and Technology, Lahore, Pakistan

# 1. Introduction

**Medis Care** is a **Medicine Management Mobile App** is a cross-platform application developed using **React Native** and **Expo**, designed to facilitate the management of medicines in a medical setting. The app allows users (admin or pharmacy staff) to perform **CRUD operations** on medicine data, such as creating, updating, deleting, and viewing medicines. The app also allows filtering of medicines based on different criteria like price, expiry date, etc. Offline functionality is supported via **SQLite**, ensuring data persistence even without internet connectivity.

# 2. Features

2.1 Admin/Pharmacy Staff Features

* Create, Edit, and Delete Medicines: Add new medicines, update existing ones, or remove them from the database.
* **Add Medicine Details:** Include information such as name, brand, type, price, quantity, and expiry date.
* **Publish/Unpublish Medicines:** Mark medicines as available or unavailable for sale or distribution.
* **View Medicine Details:** View the complete details of each medicine including price, stock quantity, and expiry date.
* **Filter Medicines:** Filter the list of medicines based on:

**-** Price (greater than or less than a specific amount).

**-** Expiry Date (e.g., show medicines expiring after a certain year).

2.2 General User Features

* **View Medicines:** View a list of all available medicines.
* **Filter Medicine List:** Users can filter medicines based on price or expiry date.
* **Search for Medicines:** Search for medicines by name or type.
* **View Detailed Information:** View detailed information on individual medicines including price, quantity, and expiry date.

# 3. Technical Specifications

3.1 Tech Stack

* **Frontend:** React Native + Expo.
* **Database:** SQLite (local storage).
* **Navigation:** React Navigation to handle app navigation.

3.2 Database Schema

The SQLite database will include the following tables:

* **Medicines:** Contains details of the medicine (e.g., name, brand, type, price, quantity, expiry date).
* **Medicine Filters:** Stores filtering preferences for the user (e.g., price range, expiry date).

The **schema** for the Medicines table will look like this:

* name: primary key Name of the medicine (text).
* brand: Brand of the medicine (text).
* type: Type of the medicine (e.g., tablet, syrup) (text).
* price: Price of the medicine (float).
* quantity: Quantity available in stock (integer).
* expiryDate: Expiry date of the medicine (date).

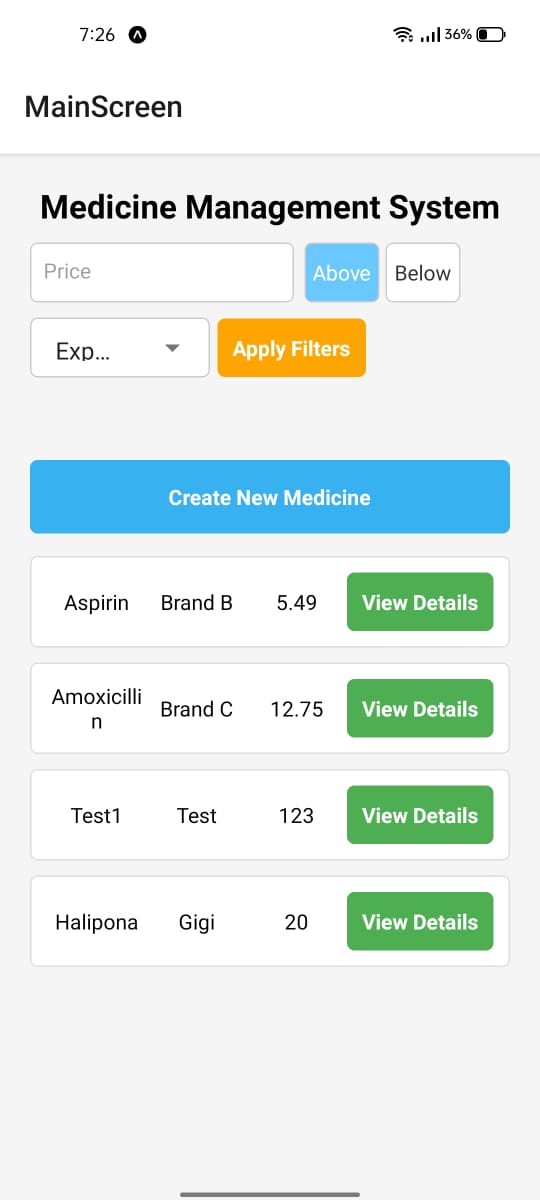
3.3 CRUD Operations

* **Create:** Admin or pharmacy staff can add new medicines to the database.
* **Read:** Users can view all medicines, and can filter them based on criteria such as price or expiry date.
* **Update:** Admin or pharmacy staff can update medicine details like price, quantity, and expiry date.
* **Delete:** Admin or pharmacy staff can remove medicines from the database.

# 4. Offline Functionality

* All data is **stored locally** via SQLite, ensuring that the app continues functioning even without internet access.
* Admin/Pharmacy staff can draft **new** medicines, **edit** existing ones, and apply **filters** offline. Changes sync automatically when the app reconnects to the internet.
* Users can **view** the available medicines and filter them even without internet access.
* **Performance:** The app is optimized to handle a large number of medicines efficiently even while offline.

# 5. Wireframes

********

# Table1: Loading Table2: Main Screen

# IMG-20250405-WA0011IMG-20250405-WA0012

# Table3: View Details Table4: Delete Medicine

# IMG-20250405-WA0009

# IMG-20250405-WA0010

# Table5: Create new Table6: Update Medicine

# IMG-20250405-WA0013

# Table7: Apply Filters

# 6. Conclusion

The Medicine Management System App showcases the **power** of React Native for **cross-platform** development, combined with efficient local data management using SQLite for **offline** functionality. This makes the app suitable for managing medicine inventories, especially in areas with limited **connectivity**.

GitHub Repository:

https://github.com/mooojn/meds-management