# Project 25:  Bundoora Family Clinic Digital Prescription and Medication Tracking Management System

# About the Client

Bundoora Family Clinic is nestled among gum trees in the west Melbourne, just 5 minutes from Belgrave and 45 minutes from the CBD. Bundoora is a semi-rural area with young families, retirees and alternative lifestyle enthusiasts. We offer our patients a variety of services and facilities to assist with their medical treatment.

The practice comprises 6 General Practitioners and host 1+ Registrar/s who practice quality medical care for patients incorporating best practice principles and up to date knowledge. They are supported by 2 practice nurses, a practice manager, office manager, 4 receptionists, on-site pathology, psychology and dietitian services.

**OUR VISION:**
Our vision is to deliver quality comprehensive medical care in a financially sustainable environment.

**OUR MISSION:**
The goal of the doctors and staff of Bundoora Family Clinic is to provide:

**Quality medical care** for patients incorporating best practice principles, up to date knowledge and technological advances.
**Quality service** for patients through awareness of individual health and social requirements, respect for the patients as people and confidentiality for any information gained.
**Quality working conditions** for doctors and staff through mutual respect and awareness individually of personal and social requirements.

# Project Brief & Business Problem Specifications:

Bundoora Family Clinic Management System needs to be developed to override the problems prevailing in the practicing manual system. This system should be supported to eliminate and in some cases reduce the hardships faced by the existing system. Moreover this system need to be designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application should reduce as much as possible to avoid errors while entering the data. It should also provide error message while entering invalid data. No formal knowledge should be needed for the user to use the system. Thus by this all it should prove it is a user-friendly system. Bundoora Family Clinic Management System, as described above, should lead to error free, secure, reliable and fast management system. It should assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it should help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Appointment, Clinic, Patient, Test, and Medicine. Every Clinic Automation system has different Clinic needs, therefore the system should be designed exclusive to employee management systems that are adapted to managerial requirements. The system should be designed to assist in strategic planning, and will help ensure that the organization is equipped with the right level of information and details for future goals. Also, for those busy executive who are always on the go, the system comes with online access features, which will allow the users to manage the workforce anytime, at all times. The system should ultimately allow the system users to better manage resources.

# Objective of Project

The main objective of the Project on Bundoora Family Clinic Management System is to manage the details of Clinic, Appointment, Doctor, Patient, and Medicine. It should manage all the information about Clinic, Test, Medicine, and Clinic. The project should totally built at administrative and user ends and only the administrator is guaranteed the complete access to the system. The purpose of the project is to build an application program to reduce the manual work for managing the Clinic, Appointment, Test, and Doctor. It should track all the details about the Doctor, Patient, and Medicine.

# Functional Requirements

**User Account and Authentication**

**FR1:** The system should allow new users (doctors, patients, pharmacists and admins) to register via a web form.
**FR2:** The system should validate all registration fields before submission.
**FR3:** The system should verify user accounts via email or OTP confirmation.
**FR4.** The system should allow users to log in using a valid username and password.
**FR5:** The system should allow password reset via a secure token link sent to the registered email.
**FR6:** The admin should be able to deactivate, activate or delete user accounts.

**Role-Based Access Control**

**FR7:** The system should assign roles (Doctor, Patient, Pharmacist, Admin) during registration.
**FR8:** Each role should have restricted access to authorised modules only.
**FR9:** The system should redirect users to their respective dashboards after login.

**Doctor Module**

**FR10:** The doctor should be able to create a new digital prescription for a registered patient.
**FR11:** The system should allow the doctor to select the patient from an existing database.
**FR12:** The system should allow doctors to add medicines with dosage,

frequency and duration.
**FR13:** The doctor should be able to attach diagnosis and clinical notes to each prescription.
**FR14:** The system should automatically generate a unique prescription ID.
**FR15:** The doctor should be able to view, edit, or cancel previously issued prescriptions.
**FR16:** The doctor should digitally sign prescriptions using a secure key or password confirmation.
**FR17:** The system should notify the patient and pharmacist when a new prescription is issued.

**Patient Module**

**FR18:** The patient should be able to view all active and past prescriptions.
**FR19:** The patient should receive email or SMS notifications for new prescriptions.
**FR20:** The patient should receive automated reminders for medication timings.
**FR21:** The patient should be able to mark each medication dose as Taken or Missed.
**FR22:** The patient should be able to request a prescription renewal from their doctor.
**FR23:** The system should allow patients to update personal and contact information.
**FR24:** The patient should view dispensing status updates from the pharmacy.

**Pharmacist Module**

**FR25:** The pharmacist should view prescriptions assigned to their pharmacy.
**FR26:** The pharmacist should verify the prescription's authenticity using the digital signature.
**FR27:** The pharmacist should update the dispensing status (Pending, Ready, Collected).
**FR28:** The system should automatically reduce stock levels when prescriptions are dispensed.
**FR29:** The pharmacist should record batch number and expiry date for each dispensed drug.
**FR30:** The pharmacist should generate and print a daily dispensing report.

**Medication Inventory and Tracking**

**FR31:** The system should store medicine details, including name, manufacturer, price and expiry.
**FR32:** The system should display the available stock for each medicine in real time.

**FR33:** The system should trigger an alert when stock falls below a minimum threshold.
**FR34:** The system should generate automatic reports of expired or low-stock medicines.

### Notification and Communication

**FR35:** The system should send notifications to users via email or SMS using a third party API.
**FR36:** The system should generate refill alerts when a patient's prescription is about to expire.
**FR37:** The system should send alerts to doctors when patients miss multiple doses.

### Reporting and Analytics

**FR38:** The admin should generate reports of total prescriptions, users and medication usage.
**FR39:** The system should display charts for the most prescribed drugs and adherence rates.
**FR40:** The system should allow exporting reports in PDF and CSV formats.

### Security and Data Protection

**FR41:** The system should encrypt sensitive information, including passwords and medical data.
**FR42:** The system should log all user activities with timestamps and IP addresses.
**FR43:** The system should automatically log out inactive users after a defined timeout period.

### AI and Decision Support

**FR44:** The system should suggest possible medications based on diagnosis keywords using a local knowledge base.
**FR45:** The system should identify potential drug–drug interactions or allergy conflicts before prescription finalisation.

# Non Functional Requirements

There are a lot of software requirements specifications included in the non-functional requirements of the system, which contains various processes, namely Security, Performance, Maintainability and Reliability.

**Performance Requirements**

NFR1: The system should load any dashboard page within 3 seconds under normal load conditions.
NFR2: The system should handle at least 50 concurrent users without performance degradation.
NFR3: Database queries should execute within 2 seconds for standard retrieval operations.
NFR4: The system should be optimised to support 500 active prescriptions and 9000 records without loss of responsiveness.

**Security Requirements**

NFR5: All user passwords should be hashed and salted using algorithms such as bcrypt.
NFR6: The system should enforce HTTPS for all communications between client and server.
NFR7: Sensitive medical data should be encrypted in both storage and transmission.
NFR8: The system should implement role-based access control (RBAC) to prevent unauthorised access.
NFR9: User sessions should automatically expire after 15 minutes of inactivity.

**Usability Requirements**

NFR10: The user interface should be responsive, supporting desktop, tablet, and mobile devices.
NFR11: The system should comply with WCAG 2.1 AA accessibility standards.
NFR12: The navigation design should follow consistent colour themes, icons and font hierarchy for readability.
NFR13: The system should provide clear error messages and validation hints for all form inputs.

**Reliability and Availability**

NFR14: The system should maintain an uptime of 99 % during operational hours excluding maintenance.
NFR15: The system should automatically back up the database daily to prevent data loss.
NFR16: The system should recover from server or database failure within 30 minutes using the latest backup.

**Maintainability and Scalability**

NFR17: The application should follow the MVC architecture (Model View Controller) to simplify code maintenance.
NFR18: The codebase should include inline documentation and follow PHP-FIG PSR-12 coding standards.
NFR19: The database should be normalised to 3rd Normal Form (3NF) to ensure scalability and data integrity.

**Auditability and Compliance**

NFR20: The system should maintain audit logs for all critical activities like logins, prescriptions and data updates including storing user ID, timestamp and IP address for at least one year to comply with data retention policies.

# User Modules (User Frontend):

Developers need to research and discuss with the client to finalize the modules and requirements.

# Reports Requirements:

- Users should easily export PDF for the Clinic, Test, Patient
- Application should also provide excel export for Appointment, Doctor, Medicine
- Also export the report into csv format for Clinic, Appointment, Medicine

# System Modules

**1. User Management Module**
Handles user registration, authentication, and profile maintenance for doctors, patients, pharmacists and administrators.
**2. Role-Based Access Control (RBAC) Module**
Implements access permissions, ensuring users only access the features allowed for their role.
**3. Doctor Module**
Provides doctors with tools to create, edit, sign and manage patient prescriptions, including diagnosis and dosage details.

**4. Patient Module**
Allows patients to view prescriptions, track medication progress, request renewals and receive reminders.
**5. Pharmacist Module**
Enables pharmacists to verify digital prescriptions, dispense medications, manage inventory updates and record transactions.
**6. Prescription Management Module**
Stores and processes all prescription data, linking doctors, patients and pharmacists in a unified workflow.
**7. Medication Inventory Management Module**
Maintains drug stock details, expiry dates, suppliers and automated low-stock alerts for pharmacies.
**8. Notification & Reminder Module**
Sends automated alerts for new prescriptions, medication schedules, refills and missed doses via email or SMS.
**9. Reporting & Analytics Module**
Generates visual and tabular reports on prescriptions, adherence rates and usage statistics for admin and healthcare monitoring.
**10. Security & Audit Log Module**
Monitors all user activities, logs sensitive transactions and provides an audit trail for accountability and data protection.
**11. Data Backup & Recovery Module**
Performs regular database backups, restoration procedures and system recovery after failure.
**12. System Administration Module**
Manages system settings, user privileges, performance monitoring and general maintenance operations.

**13. AI Decision Support Module (Optional / Extension)**
Suggests medication options based on diagnosis, detects drug-drug interactions and assists doctors with prescription validation.

# Input Data and Validation Requirements

- All the fields such as Clinic, Doctor, Medicine should be validated and does not take invalid values
- Each form for Clinic, Appointment, Test should not accept blank value fields
- Avoiding errors in data
- Controlling amount of input
- Integration of all the modules/forms in the system.
- Preparation of the test cases.
- Preparation of the possible test data with all the validation checks.
- Black-box/White-box testing.
- Recording of all the reproduced errors.
- Modifications done for the errors found during feeling.
- Prepared the test result scripts after rectification of the errors.
- Functionality of the entire module/forms.
- Validations for user input
- Checking of the coding standards to be maintained during coding.
- Testing the module with all the possible test data.
- Testing of the functionality involving all type of calculations etc.

- Creating & Changing Issues at ease
    - Query Issue List to any depth
- Multi-level Priorities & Severities.
- Targets & Milestones for guiding the programmers
- Attachments & Additional Comments for more information
- Robust database back-end
- Various level of reports available with a lot of filters criterias
- It should contain better storage capacity.
- Accuracy in work.
- Easy & fast retrieval of information.
- Well-designed reports.
- Decrease the load of the person involve in existing manual system.
- Access of any information individually.
- Work becomes very speedy.
- Easy to update information

# The proposed system requirements:

- System needs store information about new entry of Clinic.
- System needs to help the internal staff to keep information of Appointment and find them as per various queries.
- System need to maintain quantity record.
- System need to keep the record of Doctor.
- System need to update and delete the record.
- System also needs a search area.
- It also needs a security system to prevent data.

# UI Design Requirements

User Interface Design is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired inputs and outputs. The overall flow of screens and messages is called a dialogue.

1. The system user should always be aware of what to do next.
2. The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
3. Message, instructions or information should be displayed long enough to allow the system user to read them.
4. Use display attributes sparingly.
5. Default values for fields and answers to be entered by the user should be specified.
6. A user should not be allowed to proceed without correcting an error.
7. The system user should never get an operating system message or fatal error.

# Existing System

- Lack of security of data.
- More man power.
- Time consuming.
- Consumes large volume of pare work.
- Needs manual calculations.
- No direct role for the higher officials

The aim of proposed system is to develop a system of improved facilities. The proposed system should overcome all the limitations of the existing system. The system should provide proper security and reduces the manual work.

- Security of data.
- Ensure data accuracy's.
- Proper control of the higher officials.

- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.
- Minimum time required.

**Hardware Requirement: Should be recommended by the developers.**

**Software Requirement: Should be recommended by the developers.**