**📄 Patient Registration Module – Technical Requirements Document**

**Project: Eye Hospital Management System**

**Module: Patient Registration**

**Backend: Node.js**

**Database: MongoDB**

**Prepared For: Backend Developer**

**🧭 Objective**

The **Patient Registration Module** will be the **first step** in managing patient data at the Eye Hospital. Its purpose is to collect personal details of new patients and generate a unique **MR Number** for each. This module will serve as the base for all future patient interactions, including consultations, IP/OP classification, billing, and clinical records.

**🏗️ Scope of the Module**

* **Register new patients** with comprehensive personal information.
* **Auto-generate a unique MR Number** for each patient.
* **Store patient data** securely in a MongoDB database.
* **Redirect to IP/OP classification** (to be developed in the next phase).

**🧾 Data to be Captured**

The system must collect and store the following information during patient registration:

**1. Basic Details**

* **Name**
* **Father's / Spouse's Name**
* **Age**
* **Sex** (Male / Female / Other)
* **Date of Birth**
* **Age**
* **Marital Status**
* **Blood Group**

**2. Contact & Address**

* **Address Line 1**
* **Address Line 2**
* **Location**
* **City**
* **PIN Code**
* **State**
* **Country**
* **Mobile Number**
* **Email ID**
* **Next of Kin Mobile No.**
* **Relation with Next to kin**

**3. Identification & Other Info**

* **ABHA ID**
* **Aadhar Number**
* **Occupation**
* **Religion**
* **Source**
* **PAN No.**

If you enter the Date of Birth (DOB), the system will automatically calculate the Age. Similarly, if you enter the Age, the system will automatically calculate the approximate Date of Birth.

**🆔 MR Number Generation**

A unique **MR Number** must be generated automatically for every new patient. The MR number will follow a structured format:

MR + YYYYMMDD + XXXX

* YYYYMMDD → Date of registration
* XXXX → Daily serial number starting from 0001
* Example: MR202507150001

This MR number will be the **primary identifier** for the patient in the system and must never be duplicated.

**📤 Data Storage**

All registered patient data will be saved in the MongoDB database under a collection named patients.

* Each field from the form corresponds to a field in the database.
* MR Number should be marked as **unique** and used as the primary identifier for retrieval and updates.

**🔁 Workflow of the Module**

1. Staff/admin opens the **registration form**.
2. Fills in all the required patient details.
3. Upon submission:
   * A unique MR Number is generated.
   * The data is saved to the database.
4. The user is redirected to the **IP/OP selection page** (to be developed next).

**✅ Key Functional Requirements**

* **User-friendly form interface** to input data.
* **Validation rules** to ensure essential fields are filled correctly:
  + Name, Mobile Number, Date of Birth → Mandatory
  + Aadhar, Email → Optional but validated if entered
* **No duplicate MR Numbers** should be allowed.
* Date/time of registration must be captured automatically.
* Next of Kin Mobile No. and Relation with Next to Kin should appear next to Mobile Number in the UI.

**📌 Notes for Developer**

* Use **Node.js with Express.js** for API and server-side logic.
* Use **MongoDB** to store patient data in structured format.
* Ensure the MR Number is generated reliably and sequentially on a per-day basis.
* Add appropriate validations at both frontend and backend layers.
* Keep the code modular and scalable for future integration (like appointments, billing, etc.).