

# Labloom Healthcare API Documentation

**Version:** 1.0.0

**Base URL:** `https://labloom-malabar.vercel.app`

**Description:** This document provides a complete technical reference for the Labloom Backend.



## Authentication

Use these endpoints to manage user registration, login, and profile security. All private endpoints require a `Bearer <token>` in the Authorization header.

### 1. Register User

`POST /api/auth/register`

- Description:** Creates a new patient account.
- Body:**

```
{
  "name": "John Doe",
  "phone": "+1234567890",
  "email": "john@example.com",
  "password": "securepassword",
  "privacyPolicyAccepted": true
}
```

- Response:** `201 Created` with user details and JWT token.

### 2. Login (Password)

`POST /api/auth/login`

- Body:** `{"email": "john@example.com", "password": "securepassword"}`

### 3. Login (OTP Flow)

- Step 1: Send OTP:** `POST /api/auth/login-otp` | Body: `{"phone": "+1234567890"}`
- Step 2: Verify OTP:** `POST /api/auth/verify-otp` | Body: `{"phone": "+1234567890", "otp": "1234"}`



## Clinical Discovery

Endpoints for searching doctors and lab tests.

### 1. Search Doctors

`GET /api/doctors`

- Query Params:** `query`, `specialization`, `sort`
- Response:** Array of doctor objects (Name, Photo, Bio, Rating, Price).

### 2. Get All Lab Tests

GET /api/tests

- **Description:** Returns availability and pricing for blood tests, scans, and panels.
- 

## Booking System

Unified engine for scheduling appointments and tests.

### 1. Create Booking

POST /api/bookings

- **Body:**

```
{
  "bookingType": "doctor",
  "doctorId": "65ab...",
  "date": "2024-10-25",
  "time": "10:30 AM",
  "appointmentMode": "Video call"
}
```

### 2. My Bookings

GET /api/bookings/my

- **Access:** Private (Bearer Token)
  - **Response:** List of upcoming and past visits.
- 

## Medical Records

Detailed historical data including lab results and medication.

### 1. Lab Reports

GET /api/medical-records/lab-reports

- **Status Mapping:** Maps internal results to "Good", "Borderline", or "Bad" for UI display.

### 2. Prescriptions

GET /api/medical-records/prescriptions

- **Tabs:** `actual` (current) or `history` (expired).
  - **Actions:**
    - **Refill Request:** POST /api/medical-records/prescriptions/:id/refill
    - **Set Reminder:** PUT /api/medical-records/prescriptions/:id/reminder
- 

## Health Metrics

Historical tracking for vital signs.

### 1. Get Trend Data

GET /api/metrics/:type

Types: "Blood Pressure", "Weight", "Oxygen Saturation", etc.

### 2. Add New Record

POST /api/metrics

- **Body:** {"type": "Weight", "value": 72, "unit": "kg"}

---

### Settings & Utility

- **Favorite Doctors:** GET /api/users/favorites
- **Notifications:** GET /api/notifications
- **Payment Methods:** GET /api/payments/methods
- **City Search:** GET /api/utls/cities?query=Bos

---

### Error Codes

Code	Meaning
400	Bad Request (Missing fields or invalid data)
401	Unauthorized (Missing or invalid token)
404	Not Found (Resource does not exist)
500	Internal Server Error (Server or Database failure)