API DOCTOR BOOKING

# <https://localhost:7031/swagger/v1/swagger.json>

# 1. Entities / Model

- Users

- Specializations

- Doctors

- Providers

- DoctorSchedules

- Appointments

- Reviews

# 2. DTO yang Digunakan

Doctors  
- DoctorListDto – GET /api/doctors (list)  
- DoctorDetailDto – GET /api/doctors/{id}  
- DoctorCreateDto – POST /api/doctors  
- DoctorResponseDto – response setelah create/update  
  
DoctorSchedules  
- DoctorScheduleDto – untuk menampilkan jadwal dokter  
- TimeSlotDto – response GET /available  
  
Appointments  
- AppointmentCreateDto – POST appointment  
- AppointmentUpdateDto – PUT appointment  
- AppointmentSummaryDto – list ringkas appointment  
- AppointmentDetailDto – detail appointment  
  
Reviews  
- ReviewDto – GET review  
- ReviewCreateDto – POST review

# 3. Endpoints

Users  
- POST /api/users/register  
- POST /api/users/login  
- PUT /api/users/{id}  
  
Doctors  
- GET /api/doctors  
- GET /api/doctors/{id}  
- POST /api/doctors  
- PUT /api/doctors/{id}  
- DELETE /api/doctors/{id}  
  
Specializations  
- GET /api/specializations  
- GET /api/specializations/{id}  
- POST /api/specializations  
- PUT /api/specializations/{id}  
- DELETE /api/specializations/{id}  
  
Providers  
- GET /api/providers  
- POST /api/providers  
- PUT /api/providers/{id}  
- DELETE /api/providers/{id}  
  
DoctorSchedules  
- GET /api/doctorschedules/doctor/{doctorId}  
- GET /api/doctorschedules/available?doctorId=1&date=2025-07-20  
- POST /api/doctorschedules  
- PUT /api/doctorschedules/{id}  
- DELETE /api/doctorschedules/{id}  
  
Appointments  
- GET /api/appointments (dengan filter status/date)  
- GET /api/appointments/history/{userId}  
- GET /api/appointments/{id}  
- POST /api/appointments  
- POST /api/appointments/qr-appointment  
- GET /api/appointments/queue/{appointmentId}  
- PUT /api/appointments/{id}  
- DELETE /api/appointments/{id}  
  
Reviews  
- GET /api/reviews/doctor/{doctorId}  
- POST /api/reviews  
- DELETE /api/reviews/{id}

# 4. Filtering

- Appointments: Filter by status/date → GET /api/appointments?status=Scheduled&date=2025-07-20  
- DoctorSchedules: GET /api/doctorschedules/available?doctorId=1&date=YYYY-MM-DD

# 5. Contoh Testing (POST & PUT)

## POST Doctor

POST /api/doctors  
Content-Type: application/json  
  
{  
 "fullName": "Dr. Siti Rahma",  
 "email": "siti@clinic.com",  
 "phoneNumber": "08123456789",  
 "photo": "https://example.com/photo.jpg",  
 "specializationId": 1,  
 "providerId": 1,  
 "licenseNumber": "ABC123",  
 "consultationFee": 150000,  
 "biography": "Spesialis jantung dengan pengalaman 10 tahun."  
}

## PUT Doctor

PUT /api/doctors/1  
Content-Type: application/json  
  
{  
 "doctorId": 1,  
 "fullName": "Dr. Siti Rahma Update",  
 "email": "sitiupdate@clinic.com",  
 "phoneNumber": "08123456789",  
 "photo": "https://example.com/photo\_new.jpg",  
 "specializationId": 1,  
 "providerId": 1,  
 "licenseNumber": "ABC123",  
 "consultationFee": 200000,  
 "biography": "Update - Spesialis jantung."  
}

## POST Appointment

POST /api/appointments  
Content-Type: application/json  
  
{  
 "userId": 1,  
 "doctorId": 2,  
 "appointmentDate": "2025-07-22",  
 "appointmentTime": "09:00:00",  
 "patientNotes": "Keluhan sakit kepala selama 3 hari."  
}

## PUT Appointment

PUT /api/appointments/3  
Content-Type: application/json  
  
{  
 "appointmentDate": "2025-07-23",  
 "appointmentTime": "10:30:00",  
 "status": "Rescheduled",  
 "patientNotes": "Update - sakit kepala makin parah."  
}

## POST Review

POST /api/reviews  
Content-Type: application/json  
  
{  
 "doctorId": 2,  
 "userId": 1,  
 "appointmentId": 3,  
 "rating": 5,  
 "comment": "Dokternya sangat ramah dan profesional."  
}

## Users

- POST /api/users/register – Daftar user baru  
- POST /api/users/login – Login user  
- GET /api/users/{id} – Ambil profil user  
- PUT /api/users/{id} – Update profil user  
- DELETE /api/users/{id} – Nonaktifkan user (opsional)

# DTO untuk Users

UserRegisterDto  
- FullName  
- Email  
- PhoneNumber  
- Password  
  
UserLoginDto  
- Email  
- Password  
  
UserProfileDto  
- UserId  
- FullName  
- Email  
- PhoneNumber  
- DateOfBirth  
- Gender  
- Address

# Contoh Testing Users

## POST Register

POST /api/users/register  
Content-Type: application/json  
  
{  
 "fullName": "Budi Santoso",  
 "email": "budi@example.com",  
 "phoneNumber": "08123456789",  
 "password": "password123"  
}

## POST Login

POST /api/users/login  
Content-Type: application/json  
  
{  
 "email": "budi@example.com",  
 "password": "password123"  
}

## PUT User

PUT /api/users/1  
Content-Type: application/json  
  
{  
 "fullName": "Budi Santoso Update",  
 "email": "budiupdate@example.com",  
 "phoneNumber": "08123456789",  
 "dateOfBirth": "2000-01-15",  
 "gender": "M",  
 "address": "Jl. Merdeka No. 20, Jakarta"  
}