

LAPORAN KUIS PEMROGRAMAN FRAMEWORK

SISTEM INFORMASI RESERVASI SPORT CENTER

BERBASIS WEBSITE

Disusun Oleh :

MUHAMMAD HAFIDZ SETYAWAN	NIM. 244107027013
M. SAIFULLOH BAHROIN GANA PERKASA	NIM. 244107027015
MUHAMAD HAFIIDH AFANDI	NIM. 244107027018
AHMAD NANA MAINGGA	NIM. 244107027007

DOSEN PENGAMPU :

M. HASYIM RATSANJANI, S.KOM., M.KOM.



D-IV TEKNIK INFORMATIKA

TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2024

BAB 1

PENDAHULUAN

1.1. Deskripsi Project

Sistem Informasi Reservasi Sport Center Berbasis Website adalah sebuah platform digital yang dirancang untuk memudahkan pengguna dalam melakukan pemesanan lapangan olahraga secara online. Sistem ini memungkinkan pengguna untuk melihat ketersediaan lapangan, melakukan reservasi, mengelola jadwal, serta melakukan pembayaran dengan metode yang tersedia. Dengan adanya sistem ini, diharapkan dapat meningkatkan kemudahan dalam pengelolaan sport center dan memberikan pengalaman yang lebih mudah bagi pengguna.

1.2. Maksud dan Tujuan Aplikasi

Tujuan utama dari pengembangan sistem ini adalah untuk :

- Mempermudah pengguna dalam melakukan reservasi lapangan olahraga secara online.
- Meningkatkan efisiensi dan transparansi dalam pengelolaan sport center.
- Mengurangi risiko double-booking atau kesalahan administrasi dalam pemesanan lapangan.
- Menyediakan fitur pembayaran yang fleksibel dan aman bagi pengguna.
- Menyediakan riwayat pemesanan sehingga pengguna dapat melacak transaksi yang telah dilakukan.

1.3. Kemampuan Aplikasi

Sistem ini memiliki berbagai fitur utama yang mendukung pengelolaan reservasi lapangan olahraga, antara lain :

- Manajemen User
 - a. Registrasi & Login User → Pengguna dapat mendaftar dan login ke dalam sistem.
 - b. Update Profil User → Pengguna dapat memperbarui informasi pribadi mereka.
 - c. Pencarian User → Admin dapat mencari pengguna berdasarkan nama atau email.

- Manajemen Lapangan
 - a. Melihat daftar lapangan → Pengguna dapat melihat daftar lapangan yang tersedia.
 - b. Menambah & Mengupdate lapangan → Admin dapat menambahkan atau mengubah informasi lapangan.
 - c. Menghapus lapangan → Admin dapat menghapus data lapangan jika tidak digunakan lagi.
- Manajemen Reservasi
 - a. Membuat Reservasi → Pengguna dapat melakukan booking lapangan berdasarkan jadwal yang tersedia.
 - b. Mengupdate atau Membatalkan Reservasi → Pengguna bisa mengubah atau membatalkan reservasi yang sudah dibuat.
 - c. Melihat Riwayat Reservasi → Pengguna bisa melihat riwayat booking mereka.
- Manajemen Pembayaran
 - a. Metode Pembayaran Online → Pengguna bisa membayar DP atau penuh melalui gateway pembayaran.
 - b. Konfirmasi Pembayaran → Sistem akan mengecek apakah pembayaran berhasil atau gagal.

1.4. Deskripsi User

Sistem ini dirancang untuk digunakan oleh beberapa jenis pengguna, yaitu:

- **User (Pelanggan)** : Pengguna yang ingin melakukan reservasi lapangan, mengelola pemesanan, dan melakukan pembayaran.
- **Admin** : Pengelola sport center yang bertanggung jawab untuk mengelola data lapangan, memverifikasi reservasi, serta memastikan pembayaran berhasil.

BAB II

IDENTIFIKASI PROSES BISNIS

2.1. Manajemen User

- Registrasi & Login
 - a. POST /users/register → User melakukan pendaftaran.
 - b. POST /users/login → User melakukan login.
- Pengelolaan Profil
 - a. GET /users/{id} → Mendapatkan data user berdasarkan ID.
 - b. PUT /users/{id} → Mengupdate informasi user.

2.2. Manajemen Lapangan

- Melihat Daftar Lapangan
 - a. GET /fields → User dapat melihat daftar lapangan yang tersedia.
- Pengelolaan Lapangan oleh Admin
 - a. POST /fields → Admin menambahkan lapangan baru.
 - b. PUT /fields/{id} → Admin mengupdate informasi lapangan.
 - c. DELETE /fields/{id} → Admin menghapus lapangan yang tidak digunakan.

2.3. Manajemen Reservasi

- Membuat dan Mengelola Reservasi
 - a. GET /reservations → Mendapatkan daftar reservasi.
 - b. POST /reservations → User melakukan reservasi lapangan.
 - c. PUT /reservations/{id} → User mengupdate reservasi.
 - d. DELETE /reservations/{id} → User membatalkan reservasi.

2.4. Manajemen Pembayaran

- Proses Pembayaran
 - a. POST /payments → User melakukan pembayaran DP atau penuh melalui sistem.
 - b. GET /payments/{id} → User atau admin melihat detail pembayaran.
- Konfirmasi Pembayaran
 - a. Sistem akan mengecek status pembayaran (berhasil/gagal).

BAB III

PHYSICAL DATABASE DESIGN

3.1. Tabel Users (Menyimpan Data User)

NAMA KOLOM	TIPE DATA	KETERANGAN
Id	INT (PK,AI)	ID Unik Pengguna
Name	VARCHAR (255)	Nama lengkap pengguna
Email	VARCHAR (255)	Email unik pengguna
Password	VARCHAR (255)	Password terenskripsi
Role	ENUM ('admin', 'user')	Peran pengguna dalam sistem
Created_at	TIMESTAMP	Waktu pembuatan data
Updated_at	TIMESTAMP	Waktu pembaruan data

3.2. Tabel Fields (Menyimpan Data Lapangan)

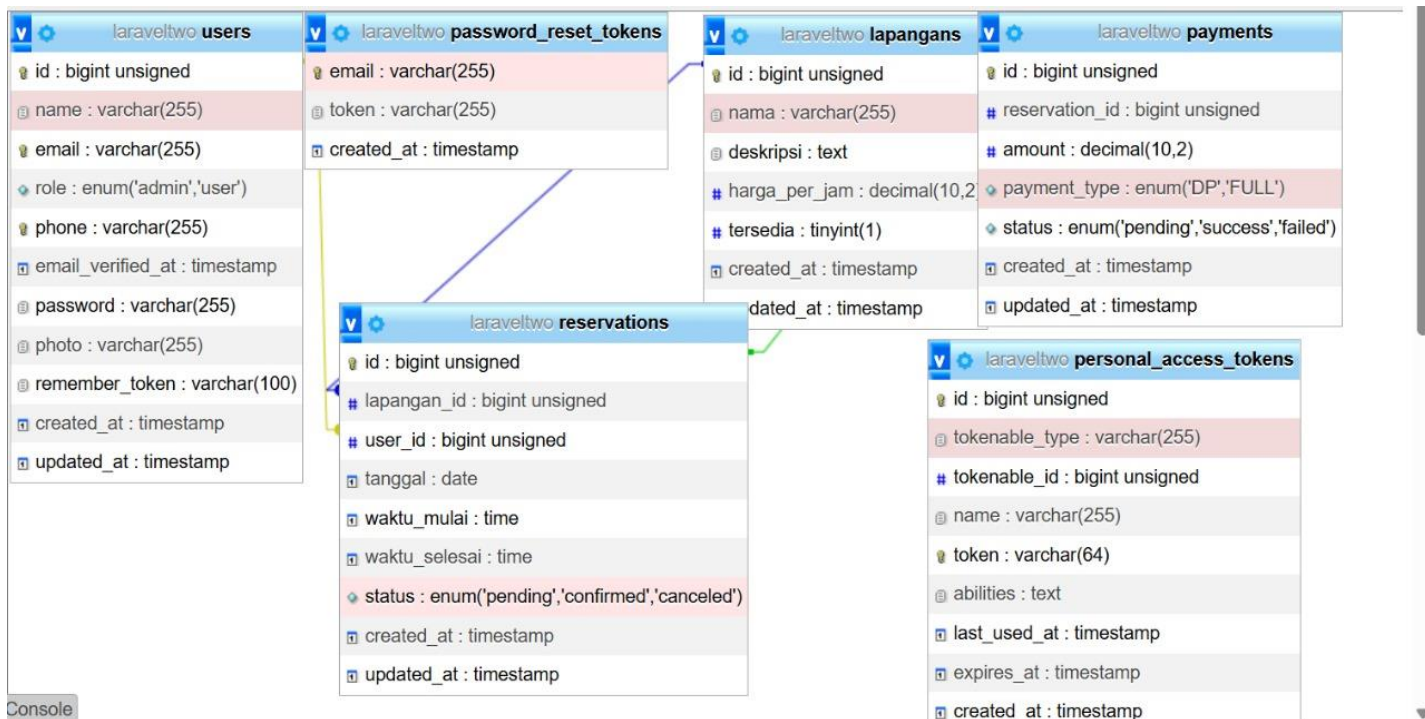
NAMA KOLOM	TIPE DATA	KETERANGAN
Id	INT (PK,AI)	ID unik lapangan
Name	VARCHAR (255)	Nama lapangan
Location	Text	Lokasi lapangan
Price	VARCHAR (255)	Harga sewa lapangan per jam
Status	ENUM ('availabe', 'unavailable')	Ketersediaan lapangan
Created_at	TIMESTAMP	Waktu pembuatan data
Updated_at	TIMESTAMP	Waktu pembaruan data

3.3. Tabel Reservations (Menyimpan Data Reservasi)

NAMA KOLOM	TIPE DATA	KETERANGAN
Id	INT (PK,AI)	ID unik reservasi
User_id	INT (FK)	ID pengguna yang melakukan reservasi
Field_id	INT (FK)	ID lapangan yang di pesan
Date	DATE	Tanggal reservasi
Time_slot	TIME	Waktu mulai reservasi
Status	ENUM ('pending', 'confirmed','canceled')	Status reservasi
Created_at	TIMESTAMP	Waktu pembuatan data
Updated_at	TIMESTAMP	Waktu pembaruan data

3.4. Tabel Payments (Menyimpan Data Pembayaran)

NAMA KOLOM	TIPE DATA	KETERANGAN
Id	INT (PK,AI)	ID unik pembayaran
Reservation_id	INT (FK)	ID reservasi yang dibayar
Amount	DECIMAL (10,2)	Jumlah pembayaran
Method	ENUM('transfer', 'credit_card', 'e-wallet')	Metode pembayaran
Status	ENUM ('pending', 'confirmed','canceled')	Status pembayaran
Created_at	TIMESTAMP	Waktu pembuatan data
Updated_at	TIMESTAMP	Waktu pembaruan data



BAB IV

ENDPOINT API

4.1. User API

HTTP METHOD	ENDPOINT	DESKRIPSI
POST	/users/register	User melakukan pendaftaran
POST	/users/login	User melakukan login
GET	/users/{id}	Mendapatkan data user berdasarkan ID
PUT	/users/{id}	Mengupdate informasi user

4.2. Field API (Lapangan)

HTTP METHOD	ENDPOINT	DESKRIPSI
GET	/fields	Mendapatkan daftar lapangan.
POST	/fields	Admin menambahkan lapangan baru.
PUT	/fields/{id}	Admin mengupdate informasi lapangan.
DELETE	/fields/{id}	Admin menghapus lapangan.

4.3. Reservation API (Booking Lapangan)

HTTP METHOD	ENDPOINT	DESKRIPSI
GET	/reservations	Mendapatkan daftar reservasi.
POST	/reservations	Membuat reservasi baru.
PUT	/reservations/{id}	Mengupdate reservasi.
DELETE	/reservations/{id}	Membatalkan reservasi.

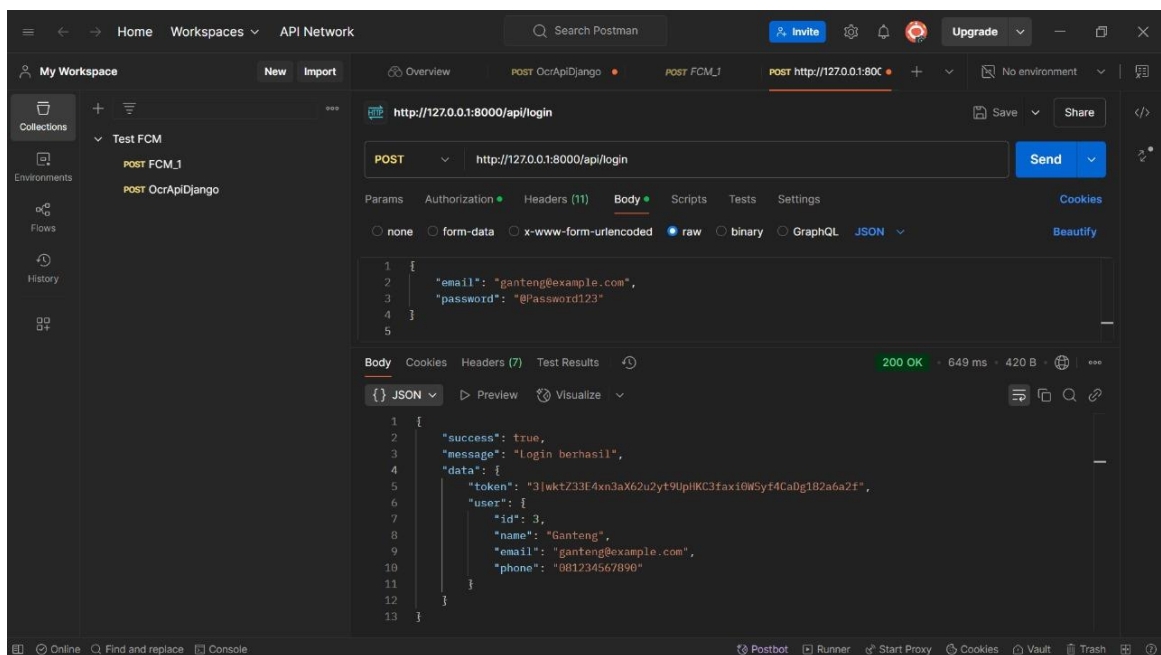
4.4. Payment API (Pembayaran)

HTTP METHOD	ENDPOINT	DESKRIPSI
GET	/payments/{id}	Melihat detail pembayaran.
POST	/payments	Melakukan pembayaran

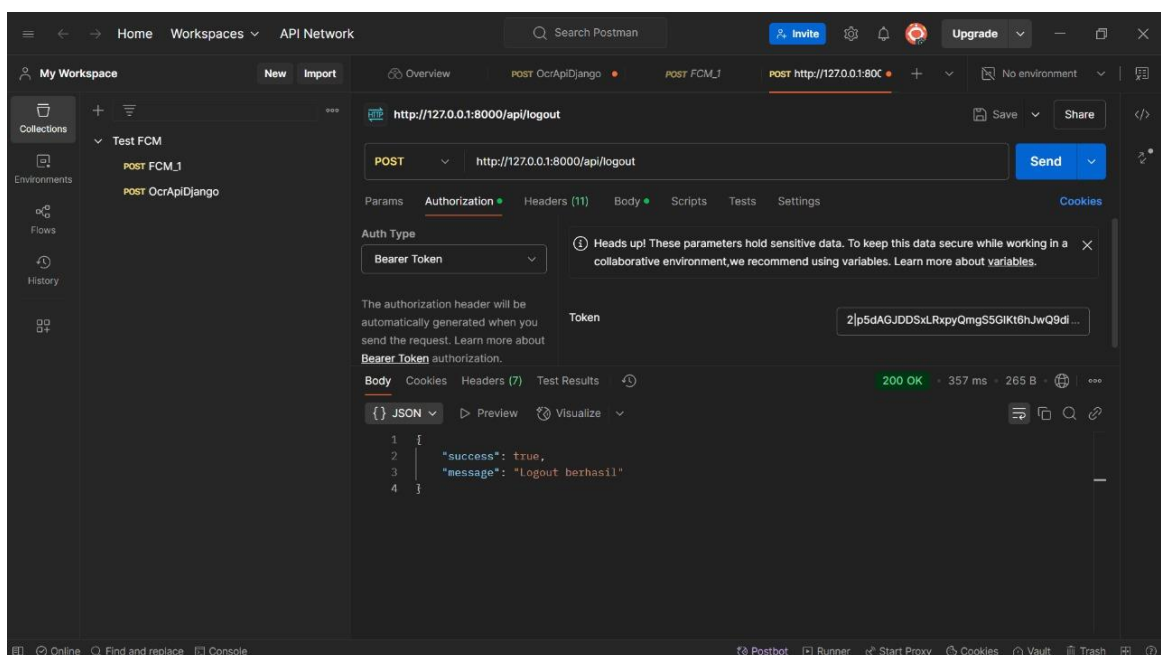
BAB V

DOKUMENTASI API

5.1. LOGIN & LOGOUT API

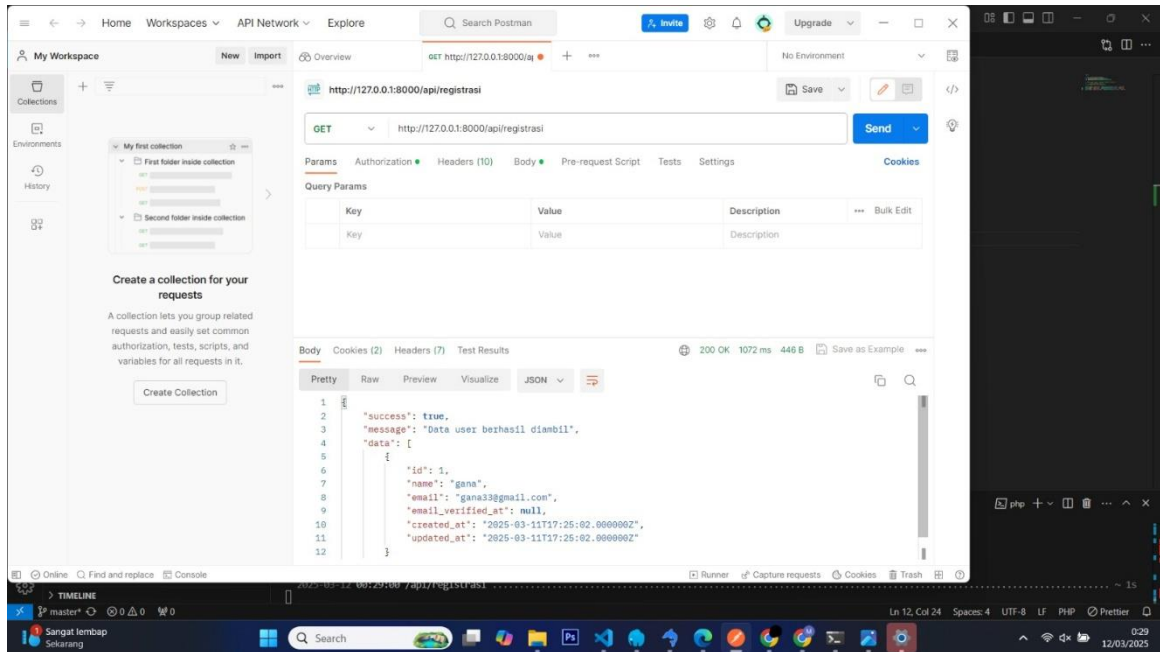


LOGIN API

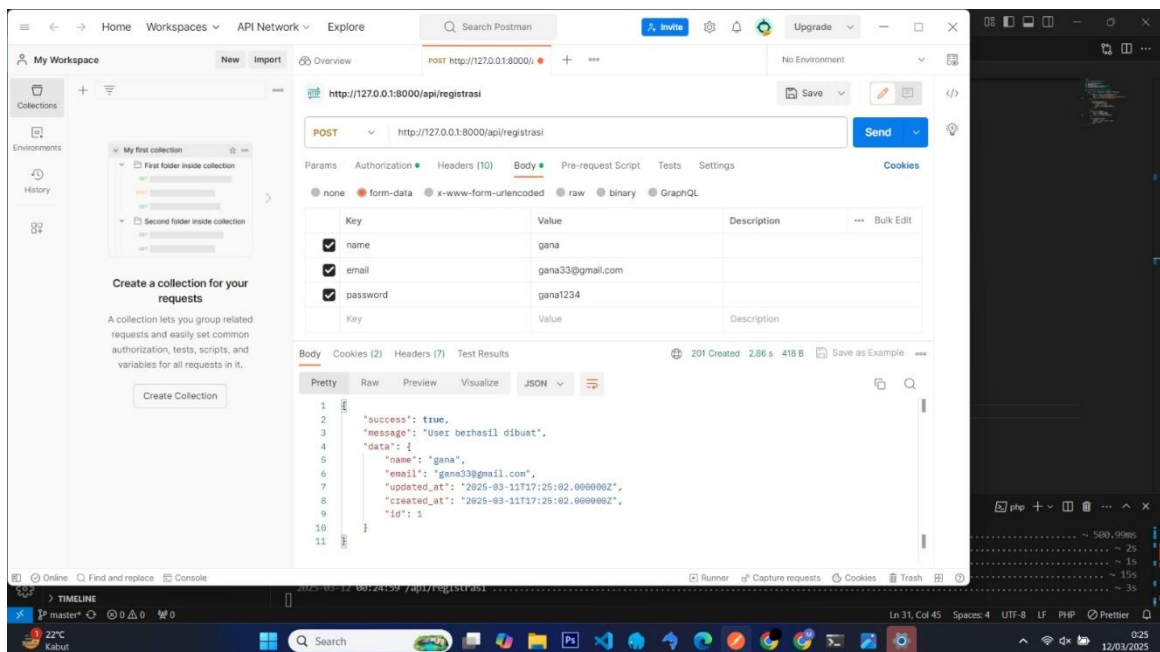


LOGOUT API

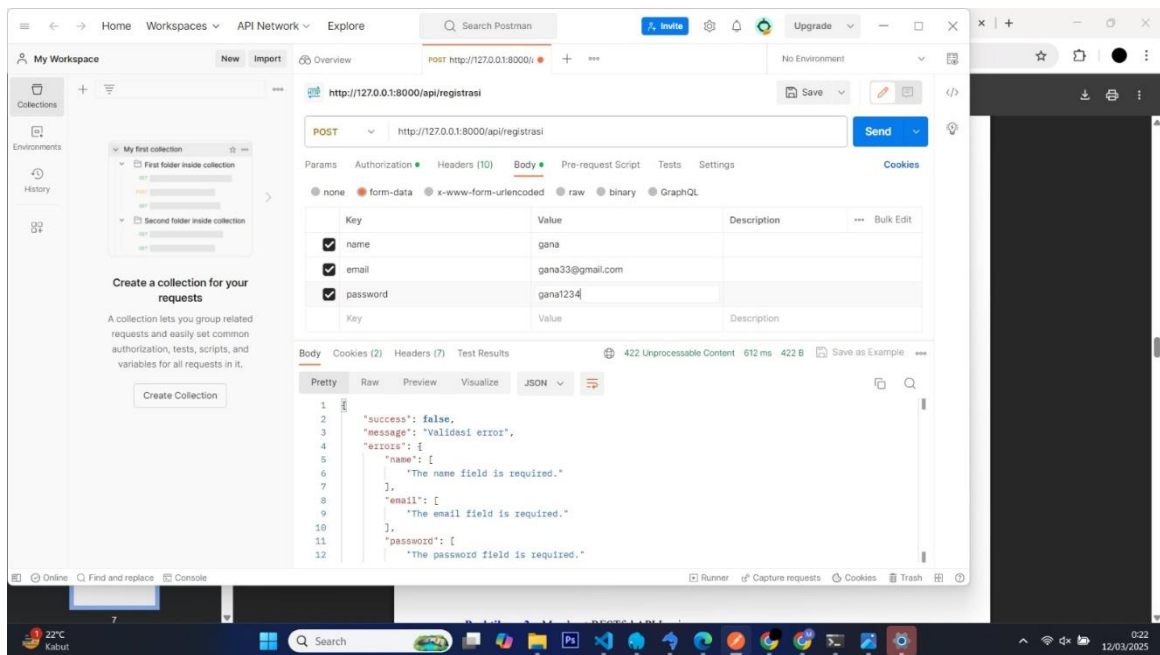
5.2. USER API



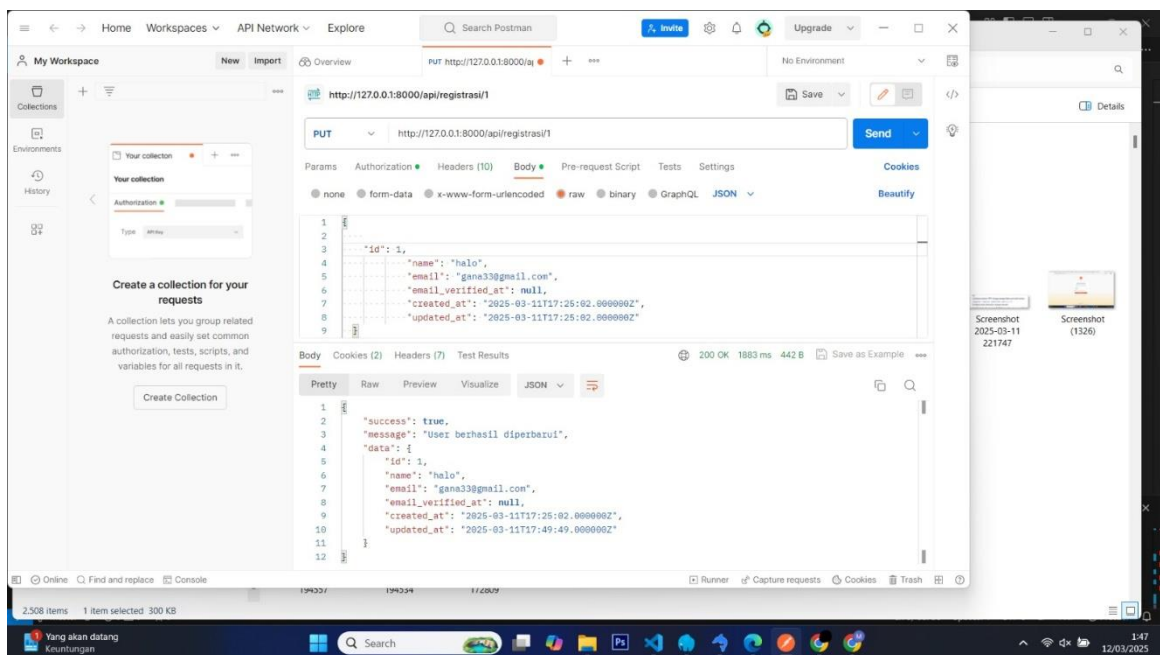
GET USER API



POST USER SESUDAH MENGIRIM DATA

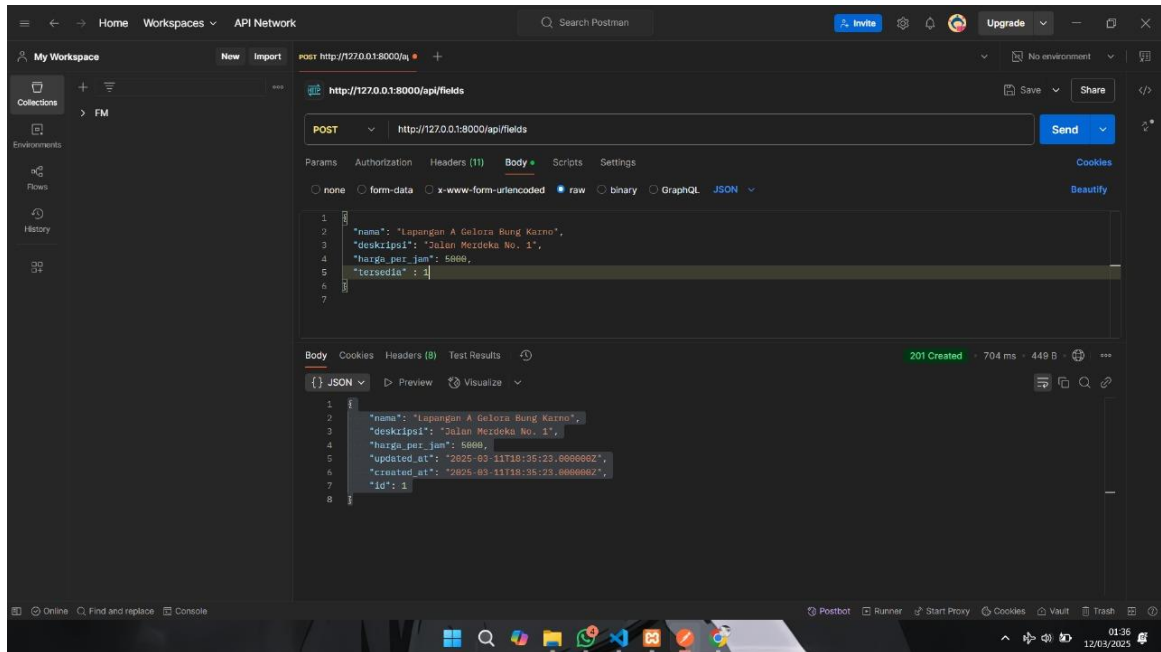


POST USER SEBELUM DATA ADA

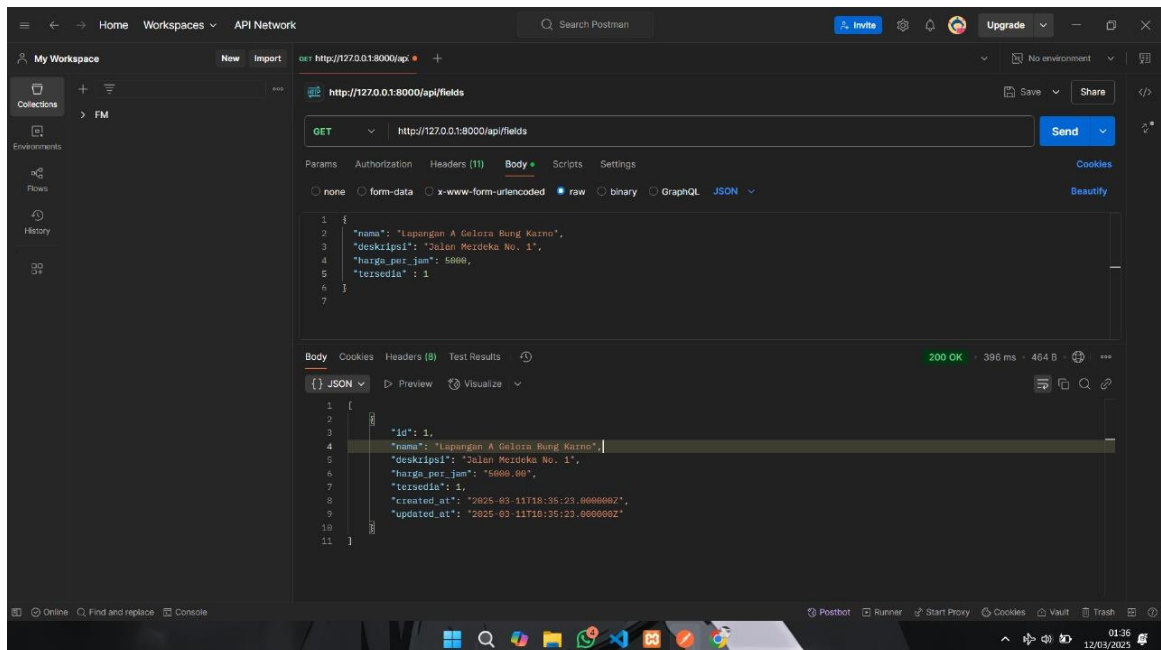


PUT USER API

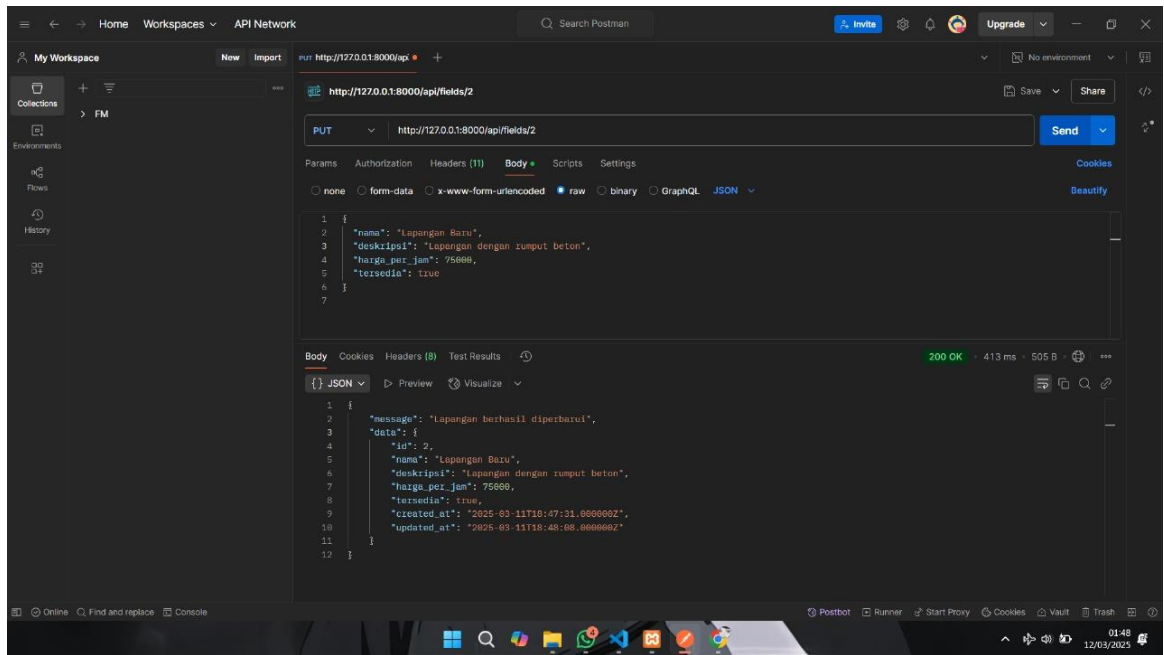
5.3. FIELDS API



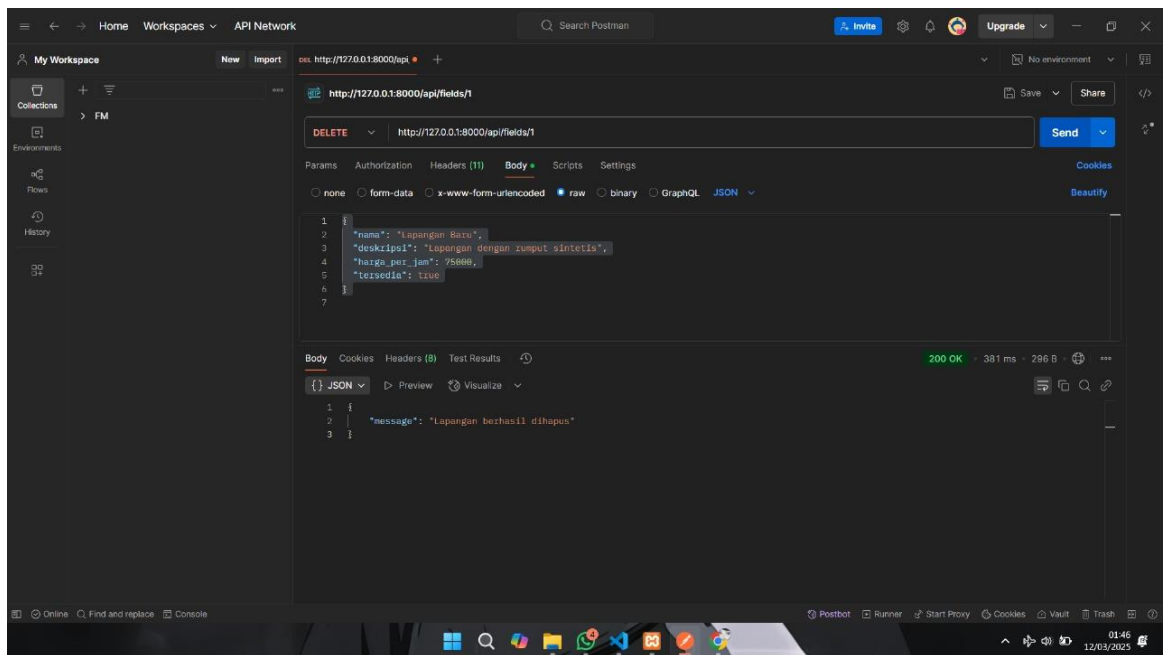
POST FIELDS API



GET FIELDS API



PUT FIELDS API



DELETE FIELDS API

5.4. RESERVATIONS API

The screenshot shows the Postman interface with a workspace named "My Workspace". The left sidebar shows a collection named "Salakinaja" with a sub-collection "Field Reservations Api". The main panel displays a POST request to `http://127.0.0.1:8000/api/reservations`. The request body is a JSON object:

```
{  "message": "Reservasi berhasil dibuat",  "data": {    "lapangan_id": 1,    "user_id": 2,    "tanggal": "2025-03-15",    "waktu_mulai": "14:00",    "waktu_selesai": "16:00",    "status": "pending",    "updated_at": "2025-03-11T19:54:52.000000Z",    "created_at": "2025-03-11T19:54:52.000000Z",    "id": 1  }  }
```

The response is a 201 Created status with a response time of 190 ms and a body size of 519 B. The response body is a JSON object:

```
{  "message": "Reservasi berhasil dibuat",  "data": {    "lapangan_id": 1,    "user_id": 2,    "tanggal": "2025-03-15",    "waktu_mulai": "14:00",    "waktu_selesai": "16:00",    "status": "pending",    "updated_at": "2025-03-11T19:54:52.000000Z",    "created_at": "2025-03-11T19:54:52.000000Z",    "id": 1  }  }
```

POST RESERVATIONS API

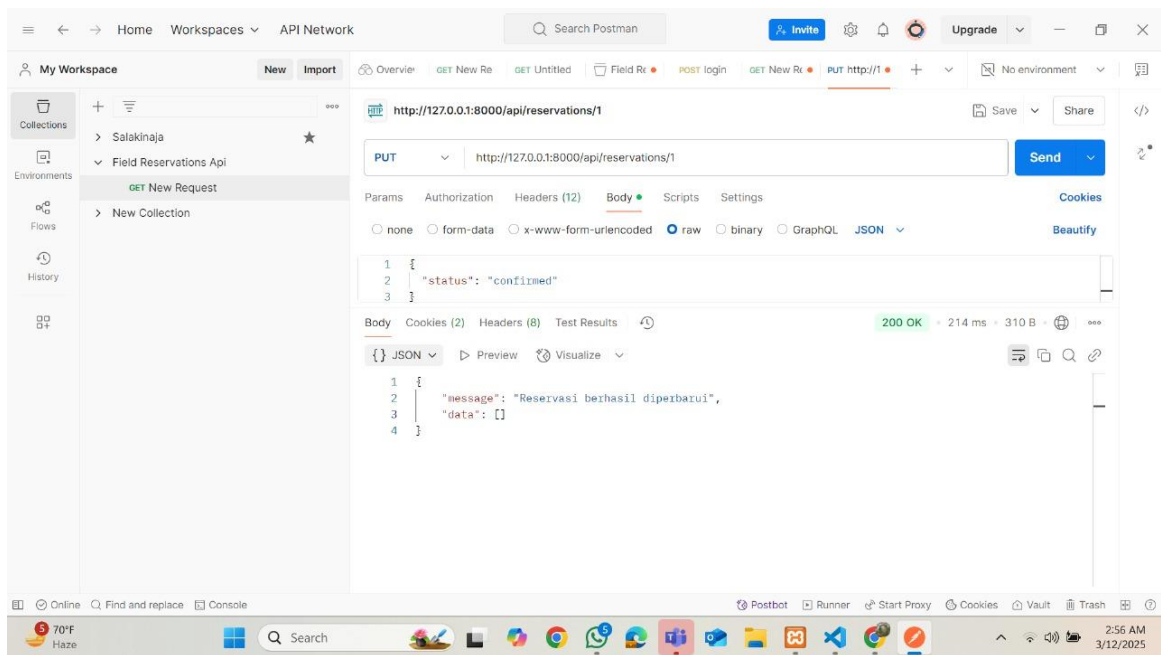
The screenshot shows the Postman interface with a workspace named "My Workspace". The left sidebar shows a collection named "Salakinaja" with a sub-collection "Field Reservations Api". The main panel displays a GET request to `http://127.0.0.1:8000/api/reservations`. The request headers are:

Key	Value	Description
Content-Type	application/json	

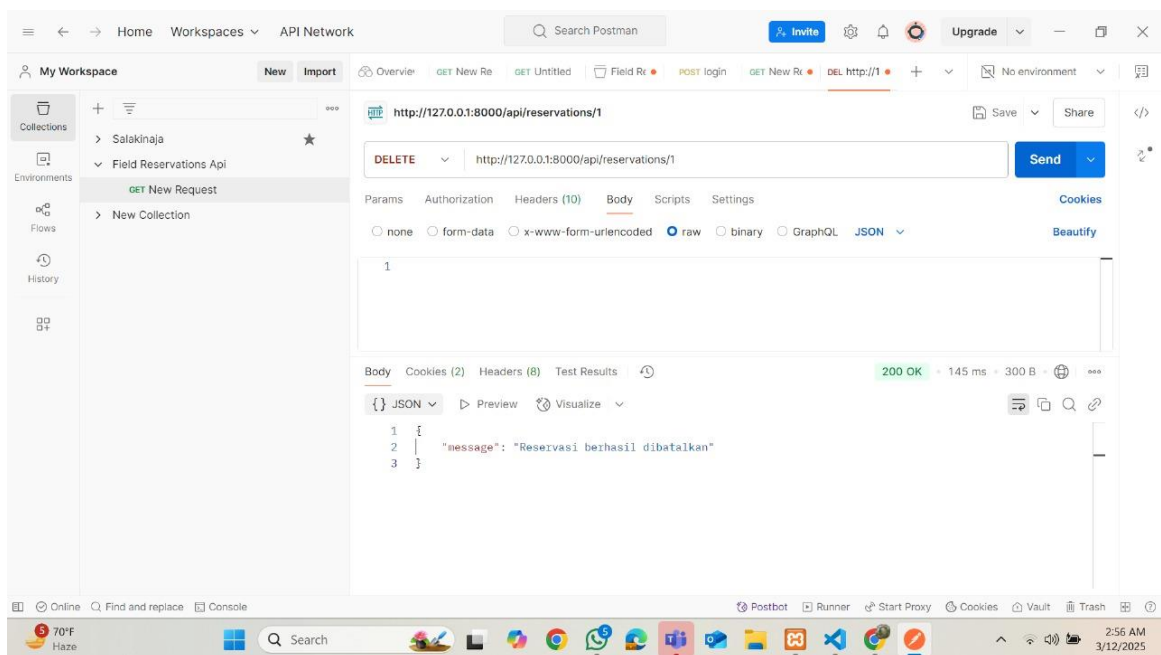
The response is a 200 OK status with a response time of 214 ms and a body size of 268 B. The response body is a JSON object:

```
{  "data": []  }
```

GET RESERVATIONS API

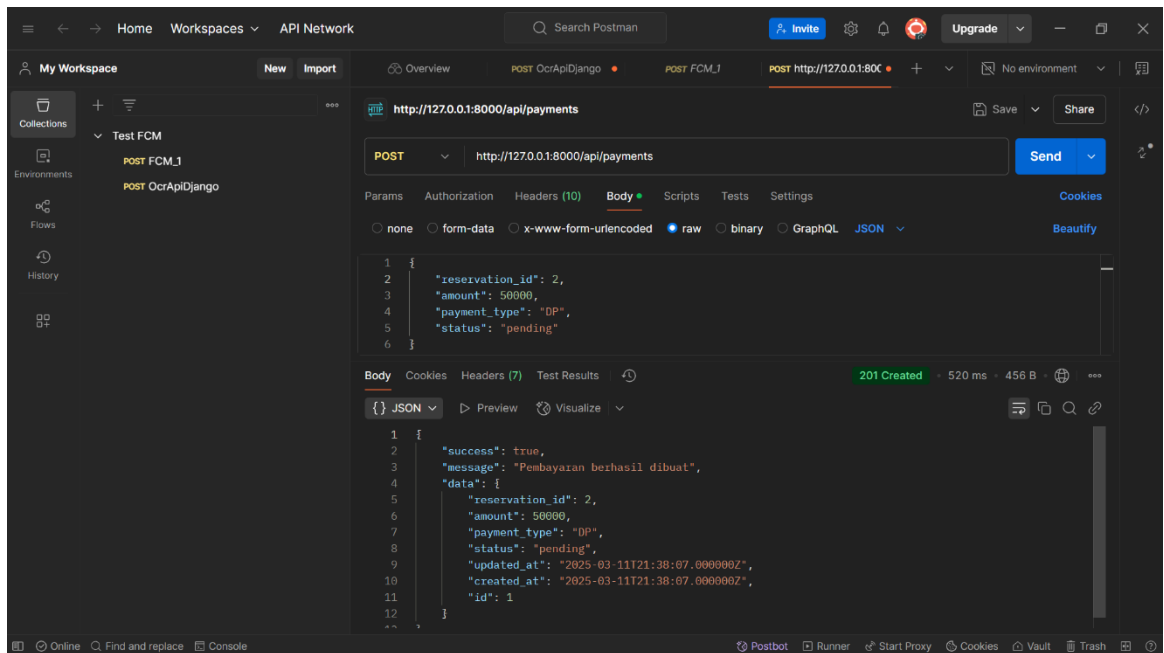


PUT RESERVATIONS API

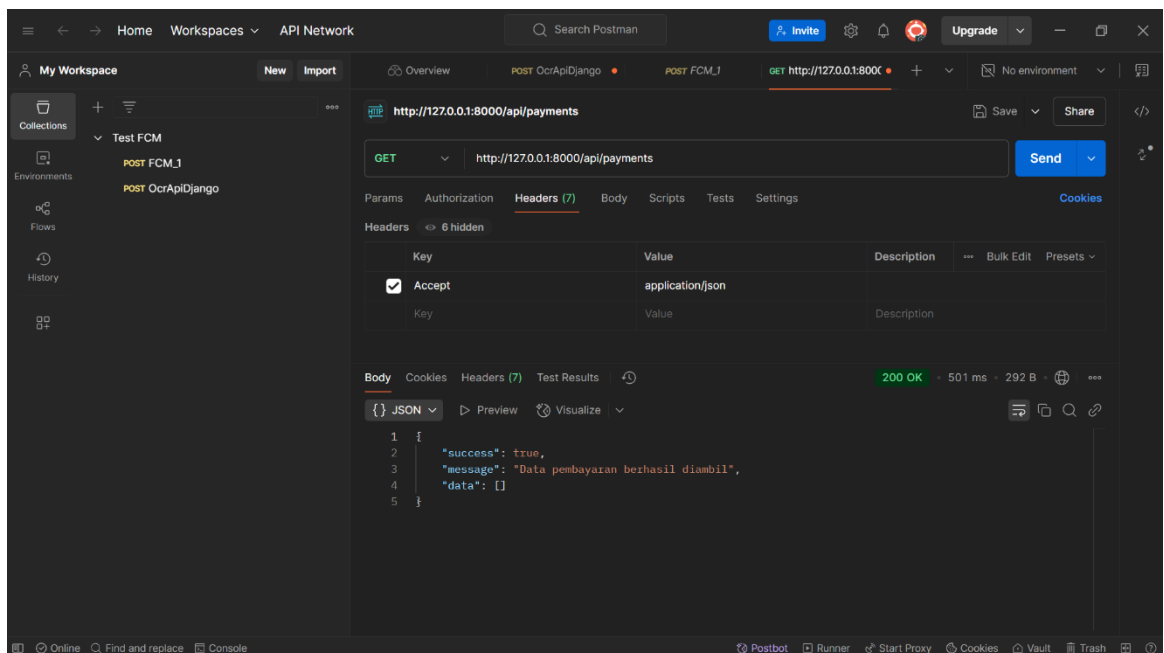


DELETE RESERVATIONS API

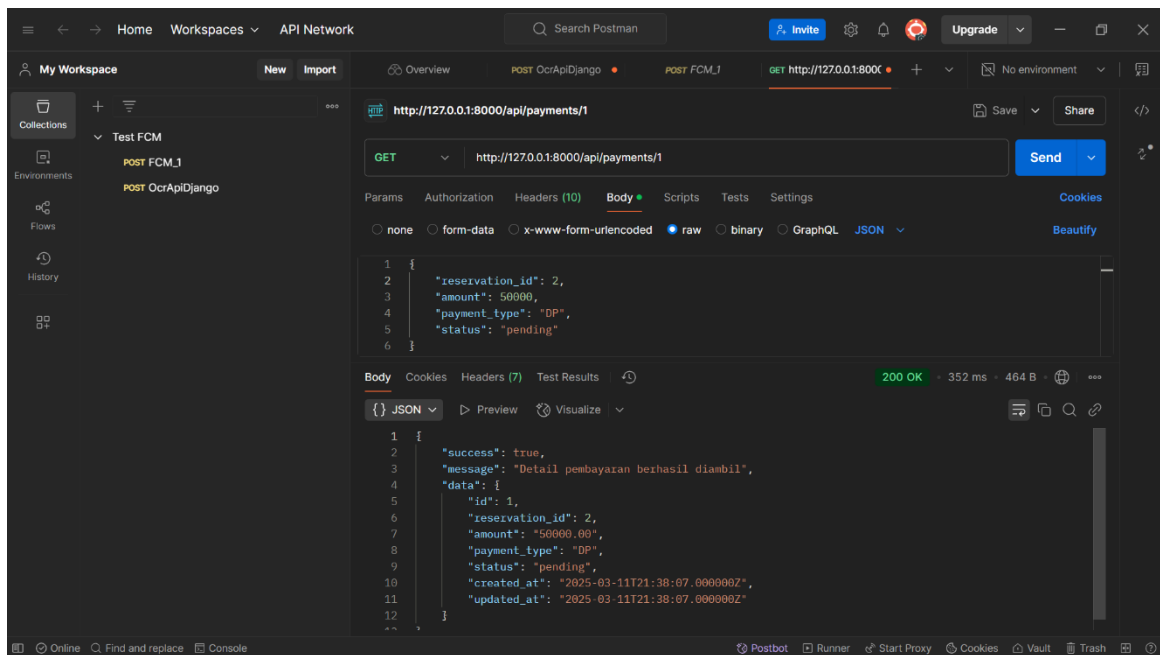
5.5. PAYMENTS API



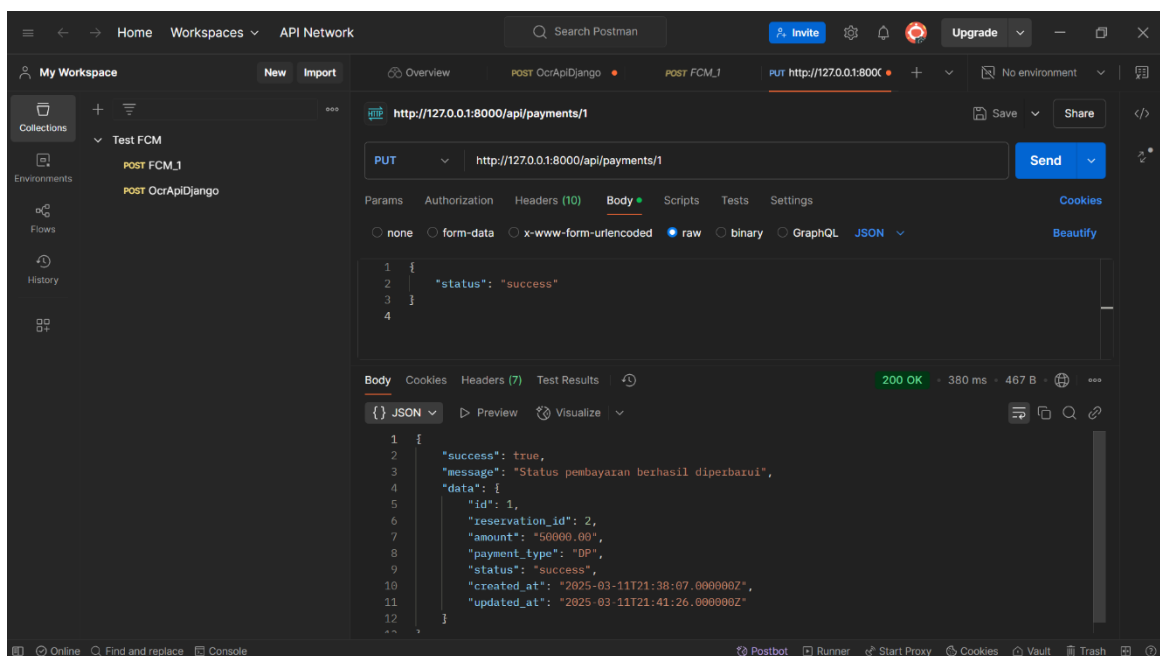
POST PAYMENTS API



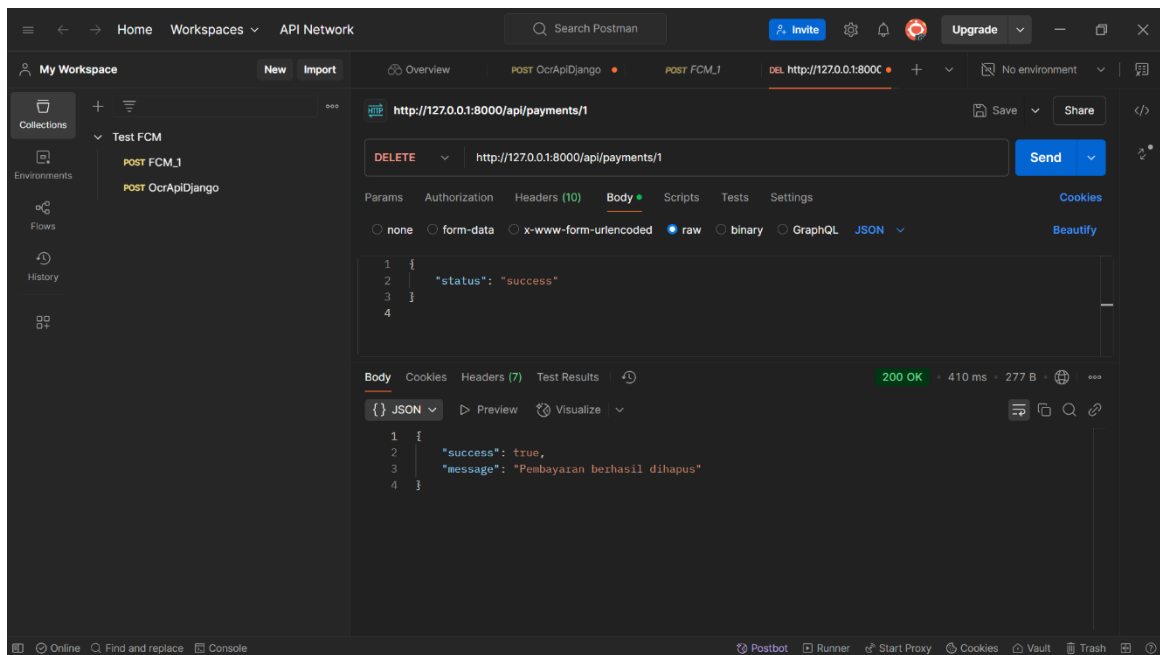
GET SEMUA DATA PAYMENTS API



GET SALAH SATU ID PAYMENTS API

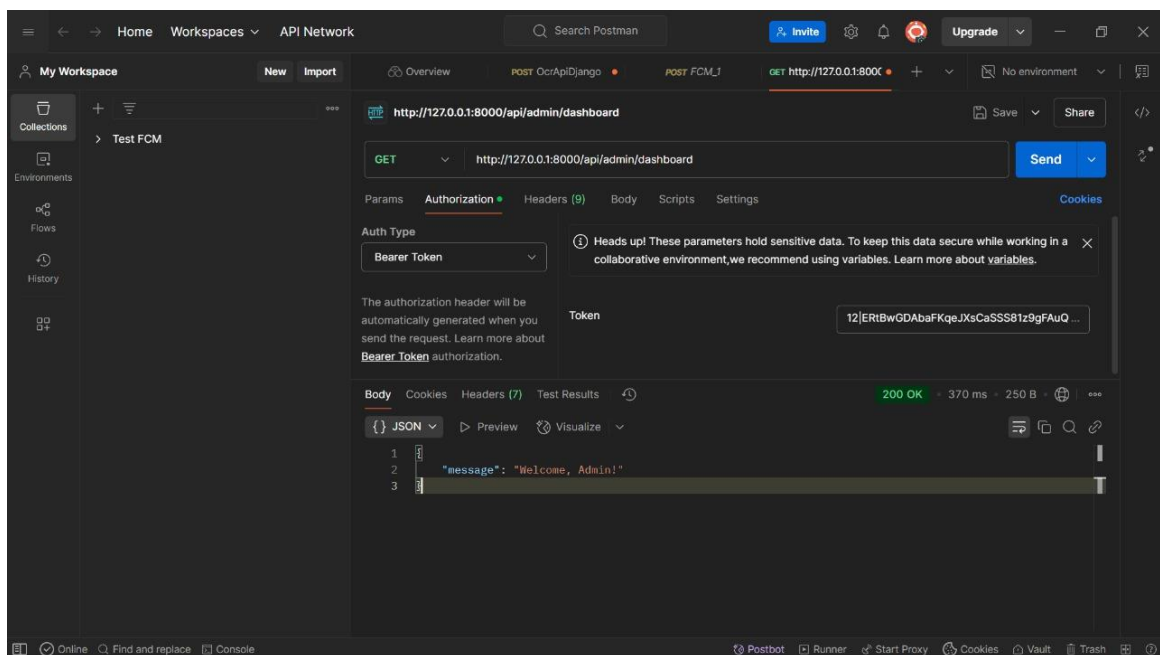


PUT PAYMENTS API

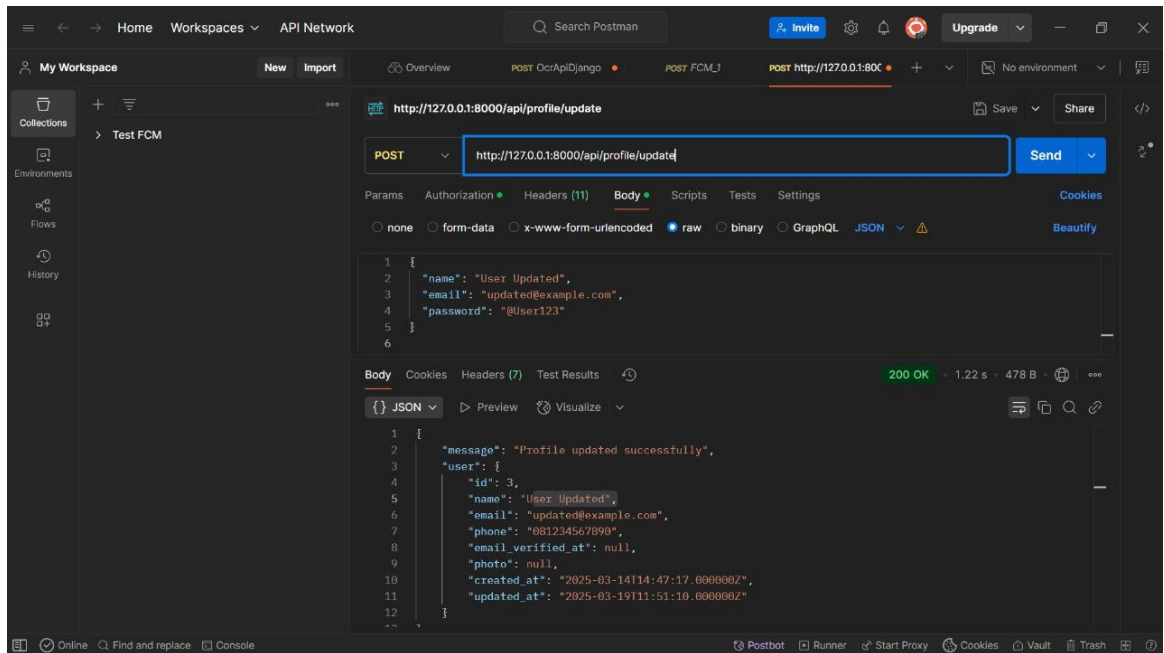


DELETE PAYMENT API

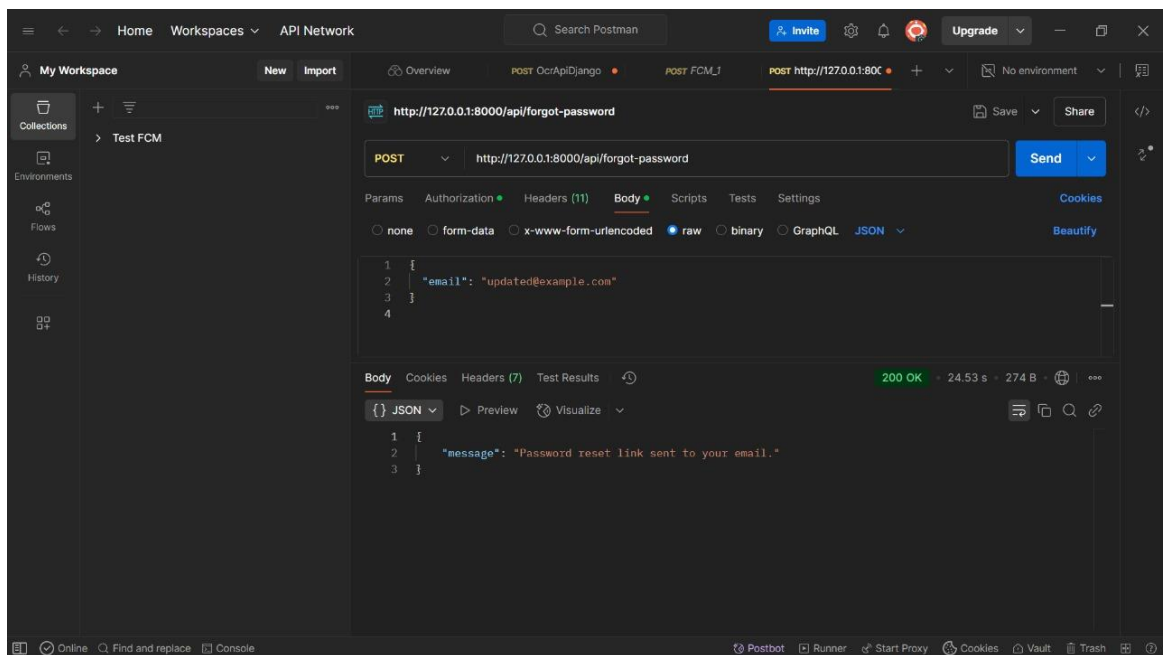
5.6. MANAJEMEN PROFIL API



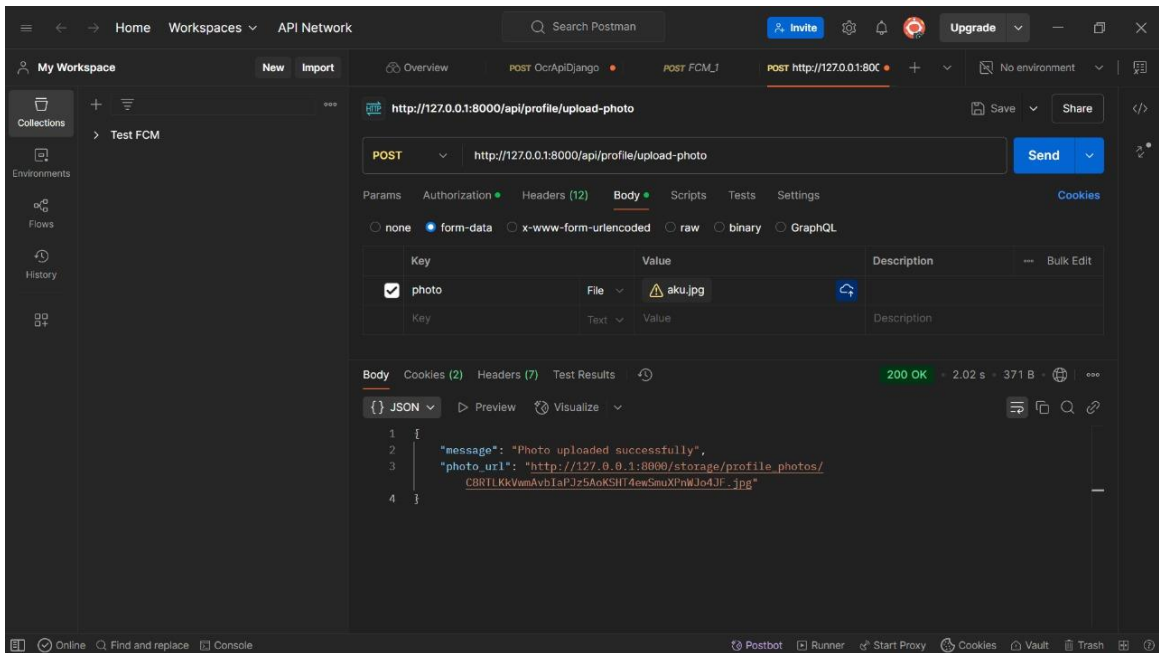
DASHBOARD API



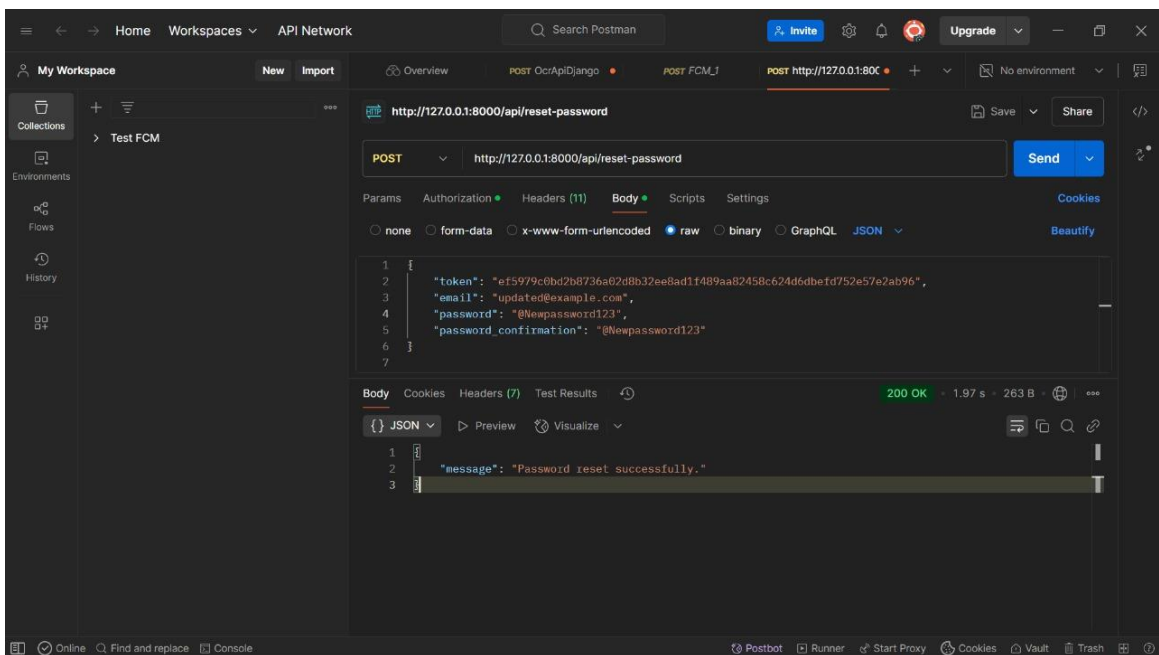
UPDATE PROFIL API



FORGET PASSWORD API



UPLOAD PHOTO PROFIL API



RESET PASSWORD API