UJIAN TENGAH SEMESTER

PEMROGRAMAN PLATFORM KHUSUS

“LAYANAN WEB SERVICE MANAJEMEN PARKIR DI POLSTAT STIS”

Dosen Pengampu:

Ibnu Santoso

Disusun Oleh:

Dwinanda Muhammad Keyzha (222212576)

PROGRAM D-IV KOMPUTASI STATISTIK

POLITEKNIK STATISTIKA STIS

JL. OTTO ISKANDARDINATA, NO 64.C, JAKARTA TIMUR

2023/2024

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa, atas limpahan rahmat dan karunia-Nya, penulis dapat menyelesaikan tugas Ujian Tengah Semester (UTS) dalam bentuk laporan sistem berjudul **"Layanan Web Service Manajemen Parkir di Polstat STIS"** tepat waktu.

Laporan ini disusun sebagai salah satu syarat penilaian mata kuliah *Pemrograman Platform Khusus* yang bertujuan untuk mengembangkan pemahaman dan keterampilan mahasiswa dalam membangun layanan berbasis web. Sistem yang dirancang ini merupakan implementasi dari konsep-konsep yang telah diajarkan selama perkuliahan, baik melalui teori maupun praktikum.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Bapak **Ibnu Santoso**, selaku dosen pengampu mata kuliah *Pemrograman Platform Khusus*, atas ilmu, bimbingan, serta motivasi yang diberikan selama proses pembelajaran. Tak lupa, penulis juga berterima kasih kepada pihak-pihak lain yang telah memberikan dukungan, baik secara langsung maupun tidak langsung, sehingga laporan ini dapat diselesaikan dengan baik.

Penulis menyadari bahwa laporan dan sistem ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan untuk perbaikan di masa mendatang. Akhir kata, semoga laporan ini dapat memberikan manfaat bagi pihak yang berkepentingan, khususnya dalam pengelolaan parkir di lingkungan Politeknik Statistika STIS.

Jakarta, 15 November 2024

Dwinanda Muhammad Keyzha

DAFTAR ISI

UJIAN TENGAH SEMESTER PEMROGRAMAN PLATFORM KHUSUS 1
KATA PENGANTAR 2
BAB 1 PENDAHULUAN 4
1.1. Latar Belakang 4
1.2. Rumusan Masalah 5
1.3. Tujuan 5
BAB 2 PEMBAHASAN 6
2.1. Class Diagram 6
2.1.1 Rancangan Class Diagram 6
2.1.2 Penjelasan Class Diagram 6
2.2. Proses Bisnis 9
2.2.1 Rancangan Proses Bisnis 9
2.2.2 Penjelasan Proses Bisnis 9
2.3. Penjelasan Kode Program 11
2.3.1 Persiapan 11
2.3.2 Auth 12
2.3.3 Config 12
2.3.4 Controller 12
2.3.5 Data Transfer Object (DTO) 13
2.3.6 Entity 13
2.3.7 Mapper 14
2.3.8 Repository 14
2.3.9 Service 14
2.4. Dokumentasi 15
BAB 3 ISI DAN DOKUMENTASI 16
3.1. Hasil Pengujian 16
3.1.1 Register 16
3.1.2 Login 17
3.1.3 User 18
3.1.3 Admin 23
BAB 4 ISI DAN DOKUMENTASI 34
4.1 Kesimpulan 34
4.2 Saran 34

BAB 1
PENDAHULUAN

1.1. Latar Belakang

Politeknik Statistika STIS yang terletak di Jl. Otto Iskandardinata No. 64.C, Jakarta Timur, merupakan institusi pendidikan yang terus berkembang dengan jumlah mahasiswa, dosen, dan staf administratif yang meningkat setiap tahunnya. Dalam mendukung aktivitas sehari-hari komunitas akademik, kampus menyediakan fasilitas parkir yang terletak di basement dan halaman depan kampus.

Namun, pengelolaan parkir di kampus ini hingga saat ini masih dilakukan secara manual, sehingga sering kali menimbulkan berbagai tantangan, seperti:

1. Ketidakmampuan untuk memantau secara real-time ketersediaan parkir.
2. Kesulitan dalam mengelola data kendaraan yang masuk dan keluar.
3. Risiko kehilangan kendaraan akibat minimnya integrasi keamanan teknologi.

Kondisi ini menimbulkan kebutuhan akan sistem manajemen parkir yang lebih modern dan terintegrasi. Sistem ini diharapkan mampu mengoptimalkan pengelolaan parkir, memberikan data yang akurat untuk mendukung pengambilan keputusan, serta meningkatkan kenyamanan dan keamanan pengguna parkir. Dengan adanya solusi berbasis teknologi yang terotomatisasi, Politeknik Statistika STIS dapat memajukan infrastruktur kampusnya dan memberikan pelayanan terbaik kepada seluruh civitas akademika.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, terdapat beberapa permasalahan yang perlu diatasi, yaitu:

1. Bagaimana cara mengatasi keterbatasan pengelolaan parkir manual yang saat ini diterapkan di Politeknik Statistika STIS?
2. Bagaimana meningkatkan pengalaman pengguna parkir di kampus dengan memanfaatkan teknologi?
3. Bagaimana memastikan keamanan kendaraan di area parkir melalui integrasi teknologi modern?

1.3. Tujuan

Adapun tujuan dari pengembangan layanan web service manajemen parkir ini adalah sebagai berikut:

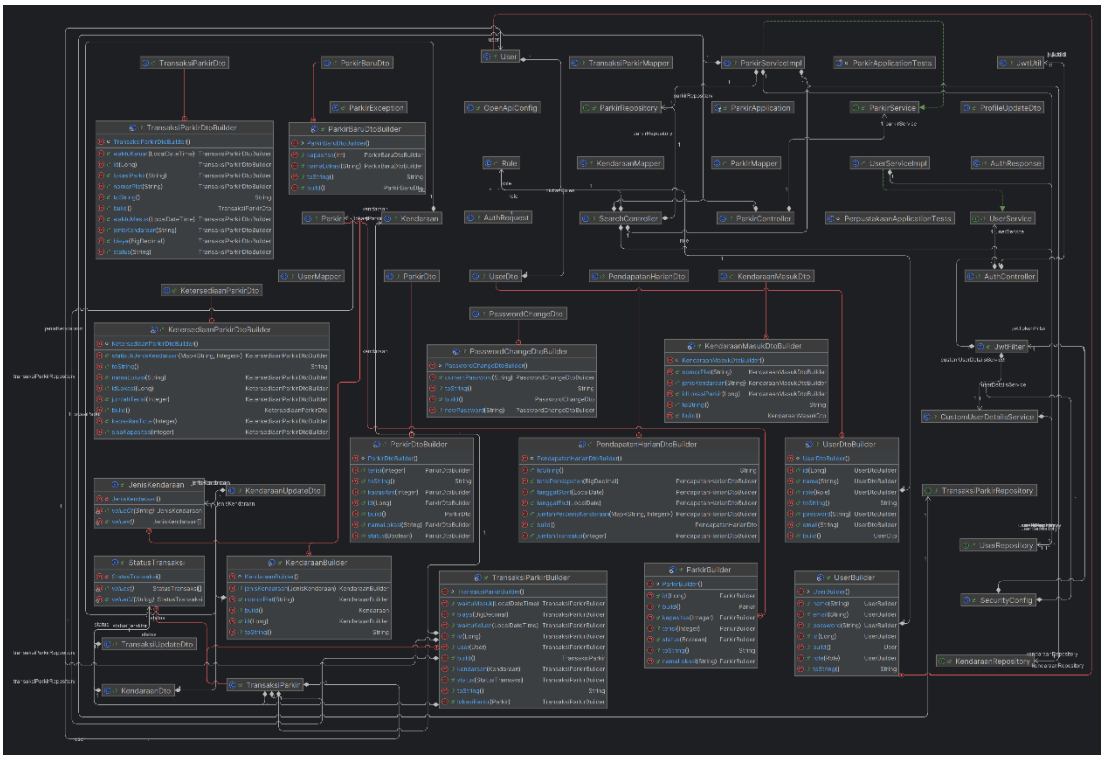
1. Membangun sistem manajemen parkir berbasis web yang dapat mengoptimalkan penggunaan fasilitas parkir di Politeknik Statistika STIS.
2. Meningkatkan kenyamanan pengguna parkir dengan menyediakan sistem reservasi, akses informasi ketersediaan parkir, dan pembayaran yang efisien.
3. Memperkuat keamanan area parkir melalui integrasi fitur teknologi seperti kamera pengawas, sistem pelacakan, dan identifikasi kendaraan.

BAB 2

PEMBAHASAN

2.1. Class Diagram

2.1.1 Rancangan Class Diagram



2.1.2 Penjelasan Class Diagram

1. Core Entities
    a. User/Profile Classes
        i. UserData
            1. Merupakan entitas utama yang menyimpan informasi pengguna
            2. Atribut: id, name, email, password, role
            3. Berhubungan dengan Profile untuk menyimpan detail tambahan pengguna
            4. Role menentukan hak akses (ADMIN, UMUM, DOSEN, MAHASISWA, KARYAWAN)
        ii. Profile
            1. Menyimpan informasi profil detail dari pengguna
            2. Atribut: id, userId, name, email
            3. Terhubung one-to-one dengan UserData
    b. Parking Management Classes
        i. LokasiParkir
            1. Mengelola informasi lokasi parkir yang tersedia
            2. Atribut: id, namaLokasi, kapasitas, terisi, status
            3. Berhubungan dengan TransaksiParkir untuk pencatatan penggunaan

ii. Kendaraan
    1. Menyimpan informasi kendaraan yang terdaftar
    2. Atribut: id, nomorPlat, jenisKendaraan
    3. Terhubung dengan TransaksiParkir untuk pencatatan aktivitas parker
iii. TransaksiParkir
    1. Mencatat seluruh transaksi parkir
    2. Atribut: id, nomorPlat, jenisKendaraan, waktuMasuk, waktuKeluar, biaya, status, lokasiParkir
    3. Memiliki relasi dengan LokasiParkir dan Kendaraan

2. Controller Classes
    a. Authentication Controllers
        i. AuthController
            1. Menangani proses autentikasi
            2. Metode: login(), register()
            3. Menggunakan JWT untuk manajemen token
            4. Mengelola operasi terkait profil pengguna
            5. Metode: getProfile(), updateProfile(), deleteProfile()
            6. Mengimplementasikan manajemen password
    b. Parking Management Controllers
        i. ParkirController
            1. Mengatur operasi utama parkir
            2. Metode: masukParkir(), keluarParkir()
            3. Mengelola transaksi parkir
            4. Mengelola lokasi parkir
            5. Metode: addLokasi(), updateLokasi(), deleteLokasi()
            6. Monitoring ketersediaan parkir
            7. Mengelola data kendaraan
            8. Metode: addKendaraan(), updateKendaraan(), deleteKendaraan()
            9. Pencarian berdasarkan plat nomor dan jenis
    c. Search and Report Controllers
        i. SearchController
            1. Menangani berbagai operasi pencarian
            2. Metode: searchTransaksi(), searchKendaraan(), searchUser()
            3. Implementasi filter dan kriteria pencarian
            4. Menghasilkan laporan dan statistik
            5. Metode: getPendapatan(), getUserStats(), getParkirSummary()
            6. Agregasi data untuk reporting
    d. Business Logic Services
        i. ParkirService
            1. Implementasi logika bisnis parkir
            2. Penghitungan biaya

3. Validasi ketersediaan

ii. UserService

1. Manajemen pengguna
2. Validasi role dan perizinan
3. Pengelolaan profil

e. Utility Services

i. . JwtService

1. Manajemen token JWT
2. Enkripsi dan dekripsi
3. Validasi token

ii. ValidationService

1. Validasi input
2. Verifikasi format data
3. Pengecekan constraint

iii. Repository Classes

1. Mengimplementasikan akses ke database
2. Menyediakan metode CRUD dasar
3. Menggunakan JPA Repository
4. Termasuk: UserRepository, ProfileRepository, ParkirRepository, KendaraanRepository, TransaksiRepository

f. Relationships

i. One-to-One

1. User dengan Profile
2. Kendaraan dengan TransaksiParkir aktif

ii. One-to-Many

1. LokasiParkir dengan TransaksiParkir
2. User dengan Kendaraan
3. User dengan TransaksiParkir

iii. Many-to-One

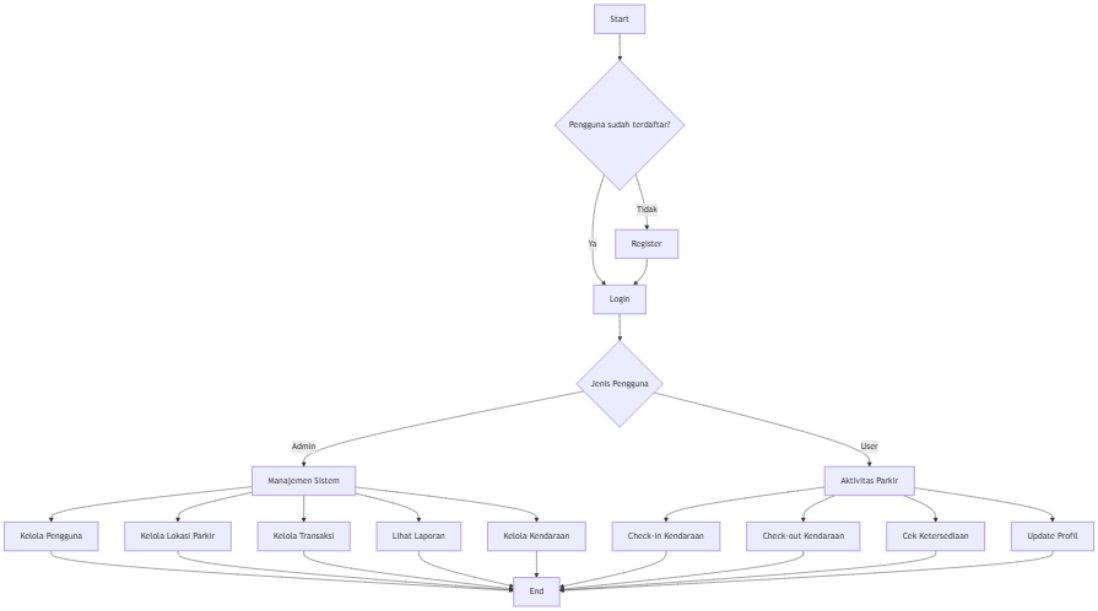
1. TransaksiParkir dengan LokasiParkir
2. TransaksiParkir dengan User

g. Security and Validation

i. Implementasi Spring Security
ii. Role-based access control
iii. Input validation pada level service
iv. Exception handling terpusat

2.2. Proses Bisnis

2.2.1 Rancangan Proses Bisnis



2.2.2 Penjelasan Proses Bisnis

1. Pendahuluan
   Sistem Manajemen Parkir Polstat STIS merupakan layanan web service yang dirancang untuk mengoptimalkan pengelolaan parkir di lingkungan kampus. Sistem ini mengakomodasi berbagai kebutuhan pengguna mulai dari administrasi hingga penggunaan fasilitas parkir sehari-hari. Pada Sistem ini perhitungan biaya parkir akan dikenakan ke user umum. Akan tetapi, khusus untuk civitas akademika tidak akan dikenai biaya.
   parkir alias gratis.
2. Proses Bisnis Utama
   a. Proses Registrasi dan Autentikasi
      i. Registrasi Pengguna Baru
         1. Pengguna mengisi form registrasi dengan data:
            a. Nama
            b. Email
            c. Password
            d. Role (UMUM/DOSEN/MAHASISWA/KARYAWAN)
            e. Sistem memvalidasi dan menyimpan data pengguna
         2. Proses Login
            a. Pengguna memasukkan email dan password
            b. Sistem memverifikasi kredensial

c. Sistem memberikan token autentikasi untuk akses selanjutnya

b. Proses Administrasi (Admin)
   i. Manajemen Pengguna
      1. Melihat daftar pengguna
      2. Mengelola profil pengguna
      3. Mengubah password pengguna
      4. Menghapus akun pengguna
   ii. Manajemen Lokasi Parkir
      1. Menambah lokasi parkir baru
      2. Mengatur kapasitas parkir
      3. Memperbarui status lokasi
      4. Menghapus lokasi parkir
   iii. Manajemen Kendaraan
      1. Melihat daftar kendaraan terdaftar
      2. Menambah data kendaraan baru
      3. Memperbarui informasi kendaraan
      4. Menghapus data kendaraan
      5. Mencari kendaraan berdasarkan nomor plat
      6. Mencari kendaraan berdasarkan jenis
   iv. Manajemen Transaksi
      1. Melihat seluruh transaksi
      2. Memperbarui detail transaksi
      3. Menghapus transaksi
      4. Menghasilkan laporan pendapatan
      5. Menghapus transaksi berdasarkan ID kendaraan
      6. Menghapus transaksi berdasarkan ID profil

c. Proses Operasional Parkir (User)
   i. Check-in Kendaraan
      1. User memilih lokasi parkir
      2. Memasukkan data kendaraan (plat nomor, jenis)
      3. Sistem mencatat waktu masuk
      4. Sistem memperbarui ketersediaan parkir
   ii. Check-out Kendaraan
      1. Sistem mencatat waktu keluar
      2. Menghitung biaya parkir
      3. Memperbarui status parkir
      4. Mengupdate ketersediaan lokasi
   iii. Monitoring Ketersediaan
      1. Melihat kapasitas real-time
      2. Mengecek status setiap lokasi
      3. Melihat statistik penggunaan

d. Fitur Pendukung
   i. Pencarian dan Pelaporan
      1. Pencarian transaksi berdasarkan periode
      2. Pencarian kendaraan berdasarkan plat nomor

3. Pencarian kendaraan berdasarkan jenis
4. Laporan pendapatan harian/periodik
5. Statistik penggunaan parkir

ii. Manajemen Profil
1. Update data pribadi
2. Perubahan password
3. Riwayat penggunaan parkir

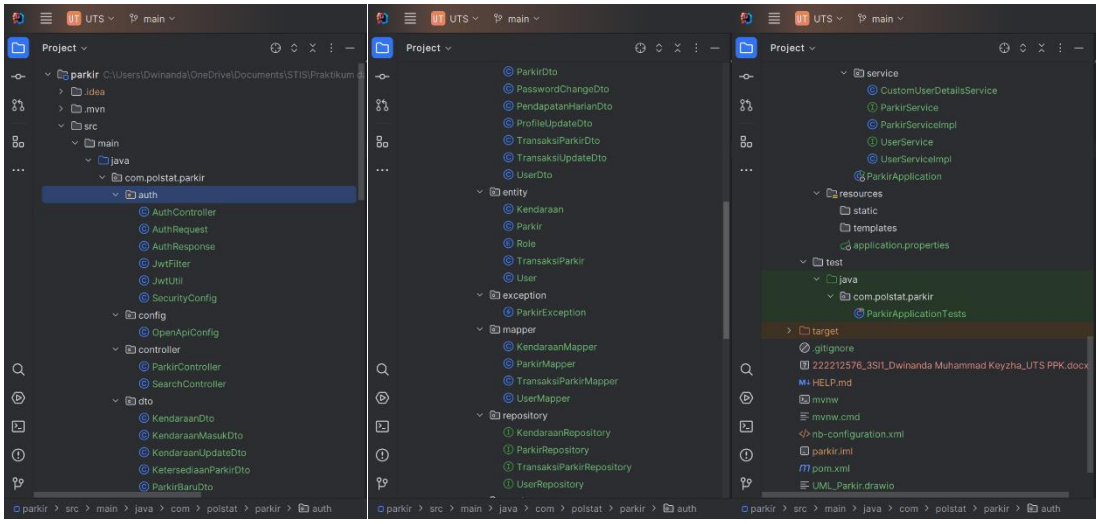
iii. Keamanan Sistem
1. Implementasi token-based authentication
2. Pembatasan akses berdasarkan role
3. Validasi input untuk setiap endpoint
4. Proteksi endpoint sensitive

iv. Integrasi Sistem
1. REST API endpoints terstandarisasi
2. Format data JSON untuk request/response
3. Dukungan untuk berbagai jenis client
4. Dokumentasi API yang lengkap

3. Kesimpulan

Sistem Manajemen Parkir ini dirancang dengan mempertimbangkan kebutuhan berbagai pemangku kepentingan dan mengimplementasikan praktik terbaik dalam pengembangan web service. Sistem menyediakan solusi komprehensif untuk manajemen parkir mulai dari operasional harian hingga pelaporan administratif, termasuk pengelolaan data kendaraan yang memungkinkan pelacakan dan manajemen kendaraan yang efisien.

2.3. Penjelasan Kode Program



2.3.1 Persiapan

1. Database Parkir
   a. Buat database dengan nama parkir.
   b. Struktur tabel akan dibuat otomatis oleh Hibernate berdasarkan entity classes.
   c. Komponen utama database meliputi tabel:

i. users (pengguna sistem)
    1. Menyimpan informasi pengguna aplikasi
    2. Termasuk kredensial dan data pribadi
    3. Manajemen pengguna
ii. kendaraan (data kendaraan)
    1. Informasi detail kendaraan yang parkir
    2. Manajemen kendaraan
iii. parkir (informasi parkir)
    1. Status dan lokasi parkir real-time
    2. Manajemen lokasi parkir
iv. transaksi_parkir (transaksi parkir)
    1. Record semua aktivitas parkir
    2. Data pembayaran dan durasi

2.3.2 Auth

Dalam implementasinya, package Auth memegang peran krusial dalam mengamankan sistem dengan menerapkan JWT (JSON Web Token) untuk autentikasi dan mengatur kontrol akses berbasis peran. Package ini bekerja sama dengan Config yang mengatur konfigurasi global aplikasi termasuk dokumentasi API melalui Swagger/OpenAPI, memastikan sistem berjalan dengan optimal dan terstandarisasi.

1. AuthController
    a. Menangani endpoint untuk login dan register
    b. Implementasi JWT authentication
    c. Pengelolaan role-based access control
2. SecurityConfig
    a. Konfigurasi keamanan aplikasi
    b. Setting JWT filter
    c. Konfigurasi CORS dan CSRF
    d. Password encryption
3. JwtUtil
    a. Pembuatan dan validasi token
    b. Manajemen user authentication
4. AuthRequest/AuthResponse
    a. DTO untuk request login/register
    b. Response berisi token dan user info

2.3.3 Config

Package Config bertanggung jawab untuk mengatur dan mengelola berbagai konfigurasi aplikasi secara terpusat. Package ini memastikan standarisasi dokumentasi API dan pengaturan sistem yang konsisten untuk mendukung operasional aplikasi secara keseluruhan.

1. OpenApiConfig
    a. Konfigurasi Swagger/OpenAPI documentation
    b. API endpoint documentation
    c. Security schemes

2.3.4 Controller

Layer Controller bertindak sebagai gerbang utama yang menangani semua request HTTP, mengorganisir endpoint-endpoint API untuk operasi parkir, pencarian kendaraan, dan manajemen transaksi. Controller berkomunikasi dengan layer service melalui Data

Transfer Objects (DTOs) yang berperan sebagai pembawa data antar layer, sekaligus memastikan validasi dan format data yang konsisten.

1. ParkirController
    a. Check-in kendaraan
    b. Check-out kendaraan
    c. Manajemen parkir
2. SearchController
    a. Implementasi pencarian kendaraan
    b. Filter dan sorting
3. TransaksiParkirController
    a. Manajemen transaksi parkir
    b. Endpoint pembayaran
    c. Reporting endpoints

2.3.5 Data Transfer Object (DTO)

Package DTO berfungsi sebagai pembawa data antar layer aplikasi, menyediakan struktur yang tervalidasi untuk pertukaran informasi. DTO membantu dalam memisahkan representasi data eksternal dari model internal, meningkatkan keamanan dan fleksibilitas sistem.

1. KendaraanDto
    a. Data kendaraan
    b. Validasi data
2. ParkirDto
    a. Data lokasi parkir
    b. Status parkir
    c. Informasi durasi
3. TransaksiParkirDto
    a. Detail transaksi
    b. Informasi pembayaran
    c. Status transaksi
4. UserDto
    a. Informasi pengguna
    b. Role information

2.3.6 Entity

Entity package mendefinisikan struktur data utama yang mencerminkan tabel-tabel dalam database, dilengkapi dengan anotasi JPA untuk mapping yang tepat. Package Mapper berperan penting dalam melakukan transformasi data antara Entity dan DTO, menggunakan MapStruct untuk mengoptimalkan proses konversi dan menjaga konsistensi data.

1. Kendaraan
    a. Properties kendaraan
2. Parkir
    a. Informasi lokasi
    b. Status parkir
    c. Relasi dengan kendaraan
3. TransaksiParkir
    a. Detail transaksi
    b. Relasi dengan parkir
    c. Informasi pembayaran

4. User
    a. Data pengguna
    b. Role dan permissions
    c. Authentication info

2.3.7 Mapper

Package Mapper berperan sebagai penghubung antara Entity dan DTO, menyediakan metode-metode untuk transformasi data yang efisien. Menggunakan MapStruct, package ini memastikan konsistensi data saat konversi antar objek dilakukan, mengurangi potensi error dan meningkatkan maintainability kode.

1. KendaraanMapper
    a. Konversi entity ke DTO
    b. Custom mapping methods
2. ParkirMapper
    a. Konversi entity ke DTO
    b. Custom mapping methods
3. TransaksiParkirMapper
    a. Mapping transaksi data
    b. Complex object mapping
4. UserMapper
    a. User data mapping
    b. Custom conversions

2.3.8 Repository

Repository package menyediakan interface untuk mengakses database, mengimplementasikan JPA Repository dengan custom query untuk kebutuhan spesifik. Layer ini bekerja sama dengan Service package yang mengandung seluruh business logic aplikasi, menangani operasi-operasi utama seperti manajemen parkir, pemrosesan transaksi, dan pengelolaan pengguna.

1. KendaraanRepository
    a. CRUD operations
    b. Custom queries
2. ParkirRepository
    a. CRUD operations
    b. Custom queries
    c. Search functionality
3. TransaksiParkirRepository
    a. Transaction management
    b. Reporting queries
    c. Statistical data retrieval

2.3.9 Service

Package Service merupakan lapisan yang mengimplementasikan seluruh business logic aplikasi. Package ini bertanggung jawab untuk mengkoordinasikan operasi-operasi kompleks, memproses data dari Repository, dan menyediakan fungsionalitas inti sistem parkir seperti manajemen pengguna, pemrosesan transaksi, dan pengelolaan parkir.

1. ParkirService
    a. Check-in/out management
    b. Parkir management
2. UserService

    a. User management
    b. Authentication logic
    c. Role management
3. TransaksiService
    a. Transaction processing
    b. Payment handling
    c. Reporting functions
4. CustomUserDetailsService
    a. User authentication
    b. User authorization
    c. Security implementation

2.4. Dokumentasi

Dokumentasi untuk API ini dibuat menggunakan postman dengan link di bawah ini:

https://documenter.getpostman.com/view/35042842/2sAY55bdg2

Dokumentasi untuk API ini juga dibuat menggunakan Swagger UI dengan link di bawah ini:

http://localhost:8080/docs/swagger-ui

Untuk kode lengkap ada di link git repository di bawah ini:

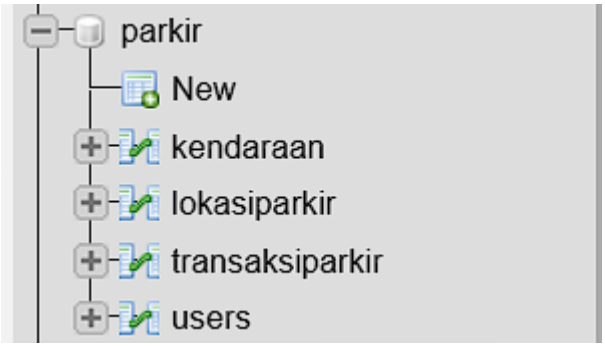
https://github.com/dwinandakey/Projek-UTS-PPK

https://git.stis.ac.id/222212576/Projek-UTS-PPK

BAB 3

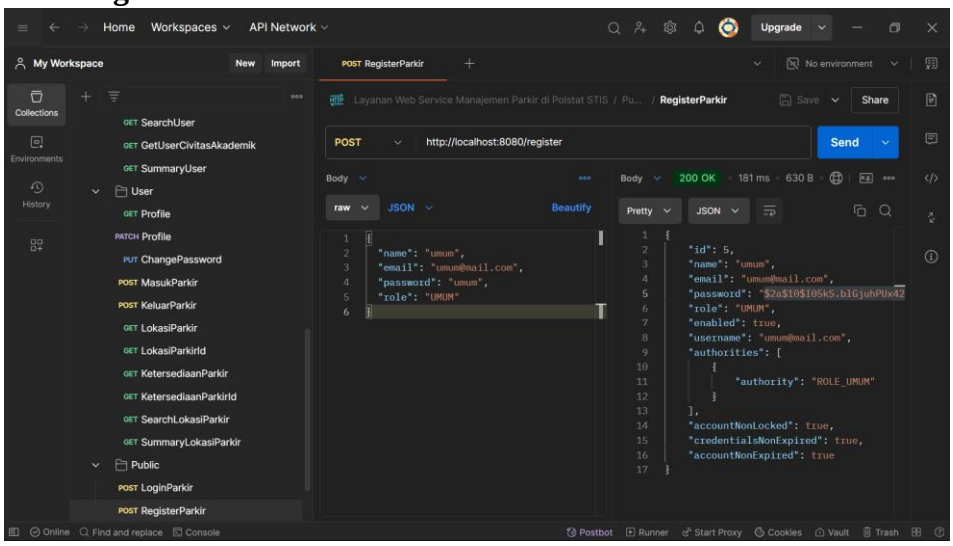
ISI DAN DOKUMENTASI

3.1. Hasil Pengujian



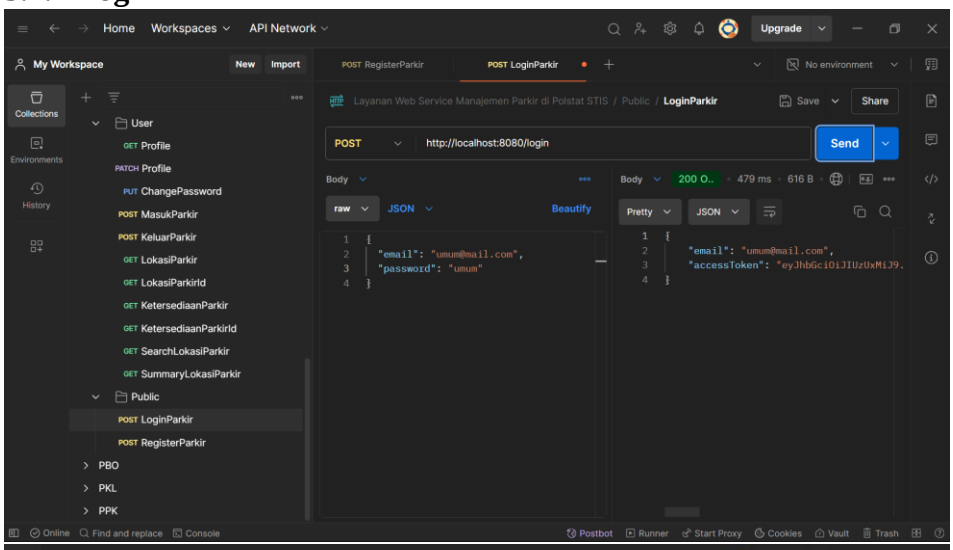
Sebelum melakukan pengujian web service, buat terlebih dahulu database parkir. Tabel-tabel yang diperlukan akan otomatis terbentuk ketika menjalankan web service.

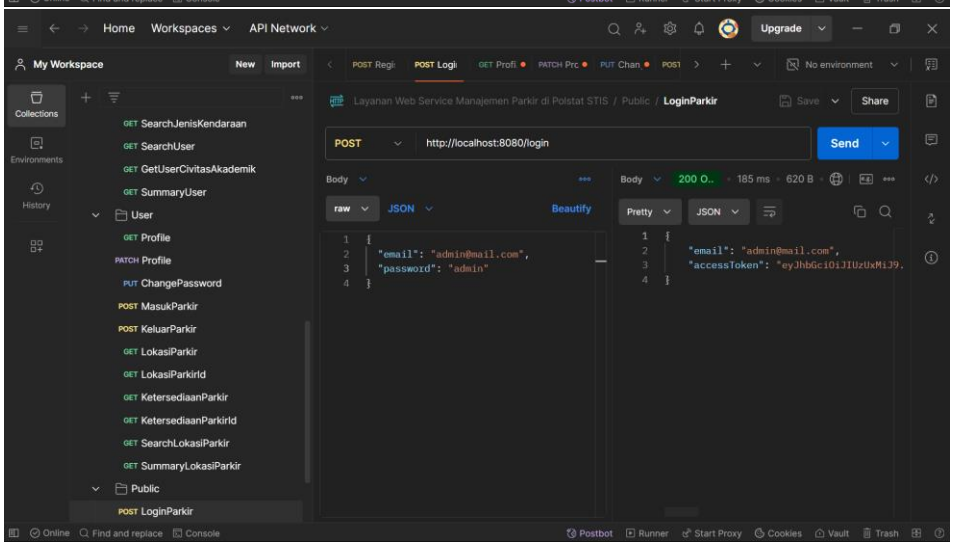
3.1.1 Register



Proses pendaftaran pengguna baru ke dalam sistem parkir. Pengguna dapat mendaftar dengan memilih peran seperti "UMUM", "DOSEN", "MAHASISWA", atau "KARYAWAN" sesuai dengan ketentuan sistem.

3.1.2 Login

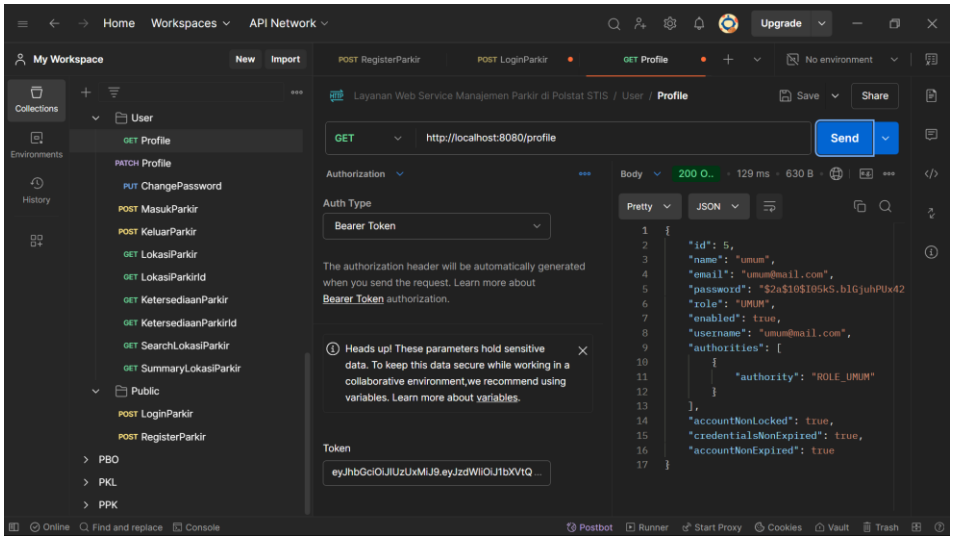




Proses autentikasi pengguna untuk mendapatkan akses ke sistem parkir. Setelah berhasil login, pengguna akan mendapatkan token autentikasi yang diperlukan untuk mengakses fitur-fitur lainnya.

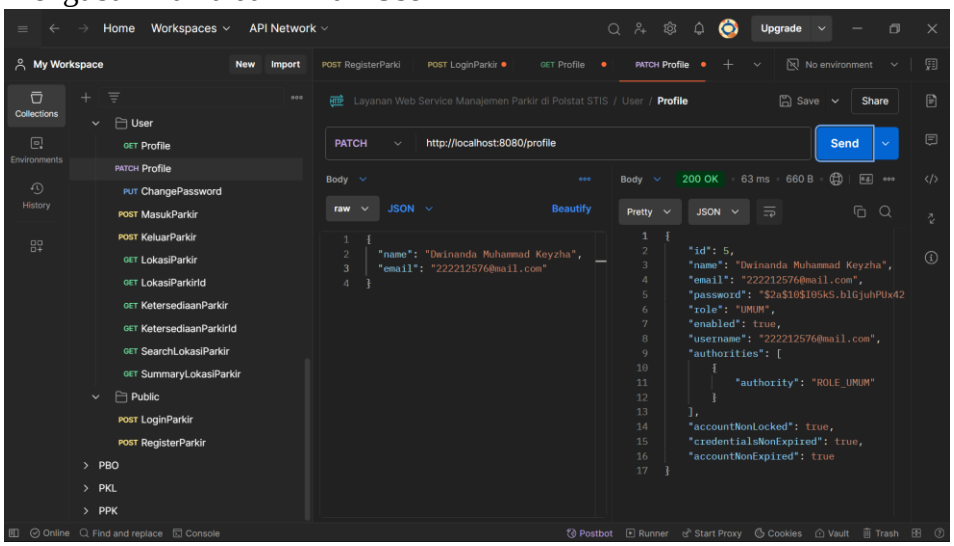
3.1.3 User

1. Melihat Profil User



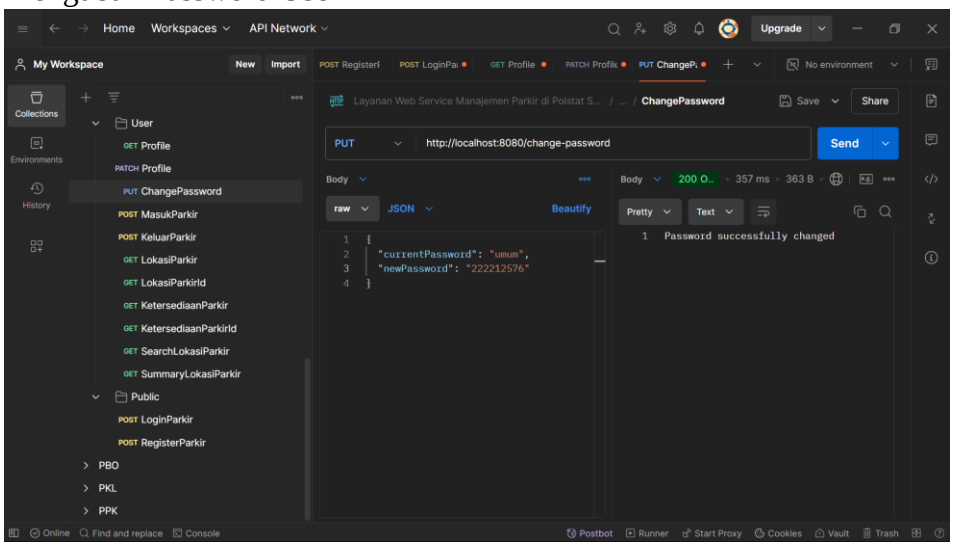
Mengambil informasi profil pengguna yang sedang login, termasuk nama, email, dan detail lainnya.

2. Mengubah Nama dan Email User



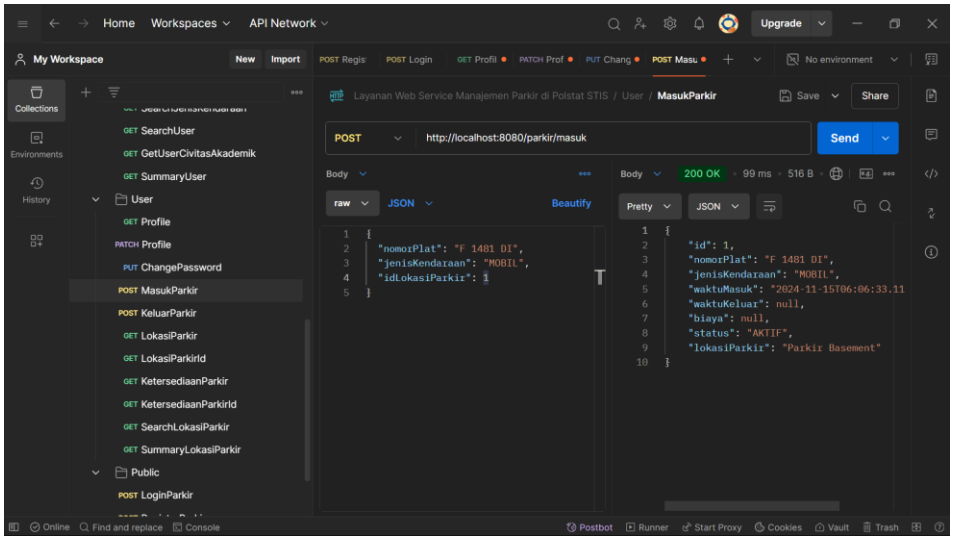
Memungkinkan pengguna untuk memperbarui informasi profil seperti nama dan alamat email.

3. Mengubah Password User



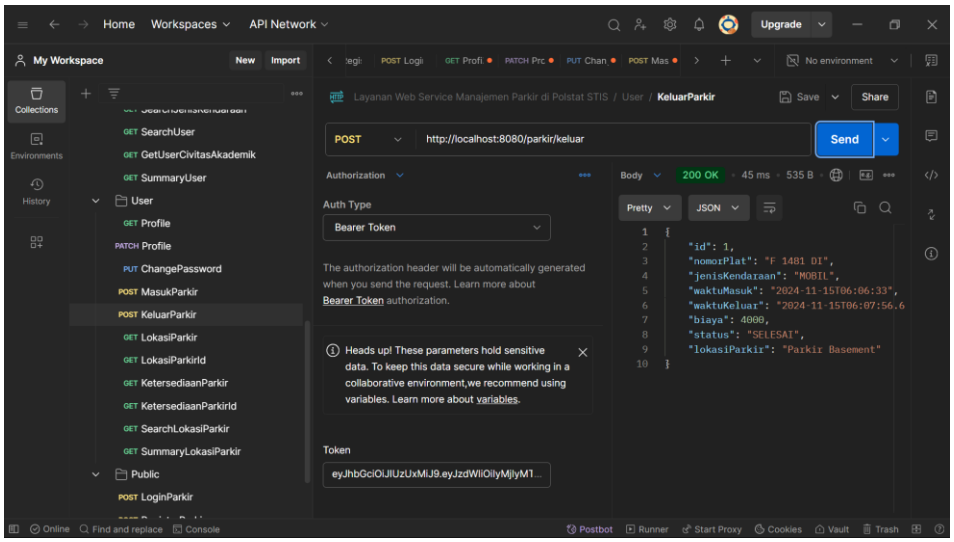
Fitur untuk mengganti kata sandi pengguna yang sedang login dengan verifikasi password lama.

4. Masuk Parkir



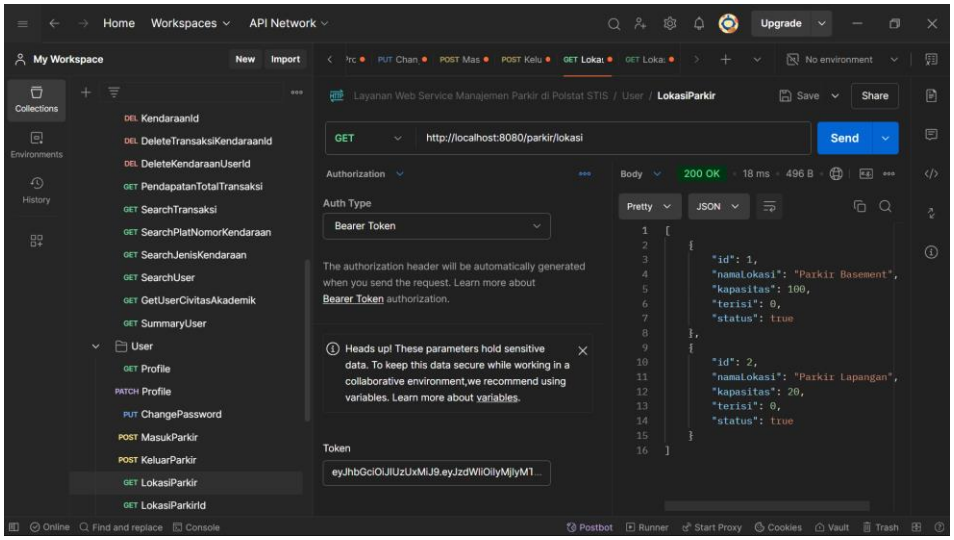
Mencatat kendaraan yang masuk ke area parkir dengan detail nomor plat, jenis kendaraan, dan lokasi parkir.

5. Keluar Parkir



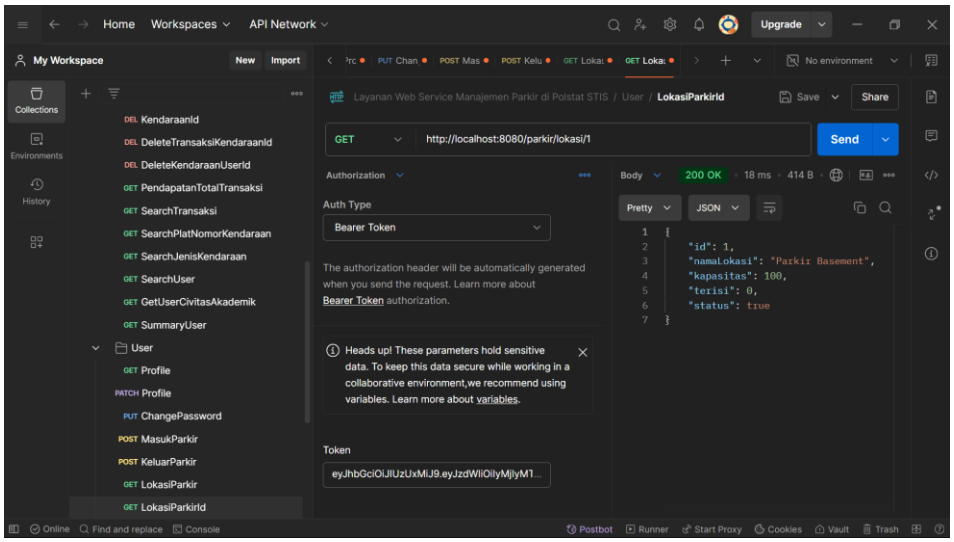
Mencatat waktu keluar kendaraan, menghitung biaya parkir, dan memperbarui status parkir.

6. Melihat Lokasi Parkir



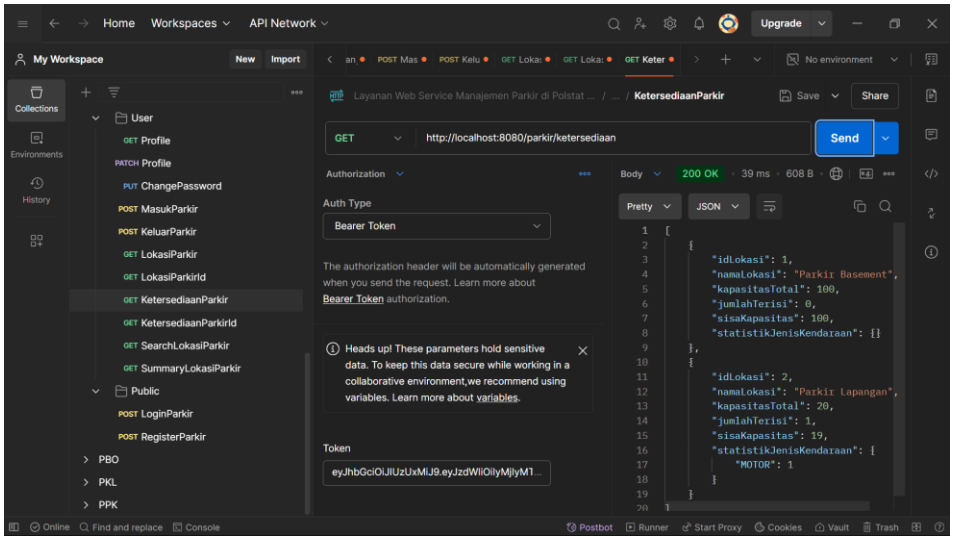
Menampilkan daftar seluruh lokasi parkir yang tersedia dalam sistem.

7. Melihat Lokasi Parkir Berdasarkan Id



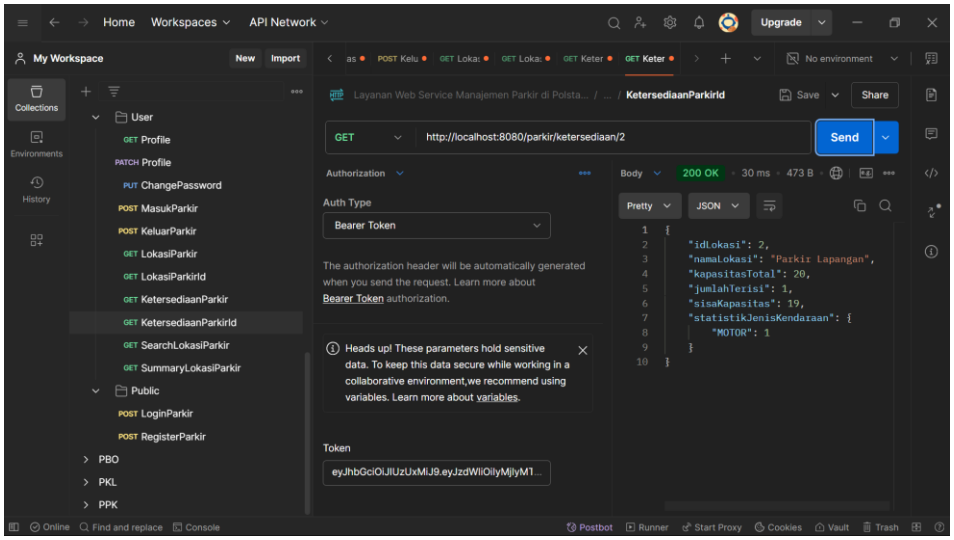
Menampilkan detail spesifik dari lokasi parkir tertentu berdasarkan ID.

8. Melihat Ketersediaan Parkir



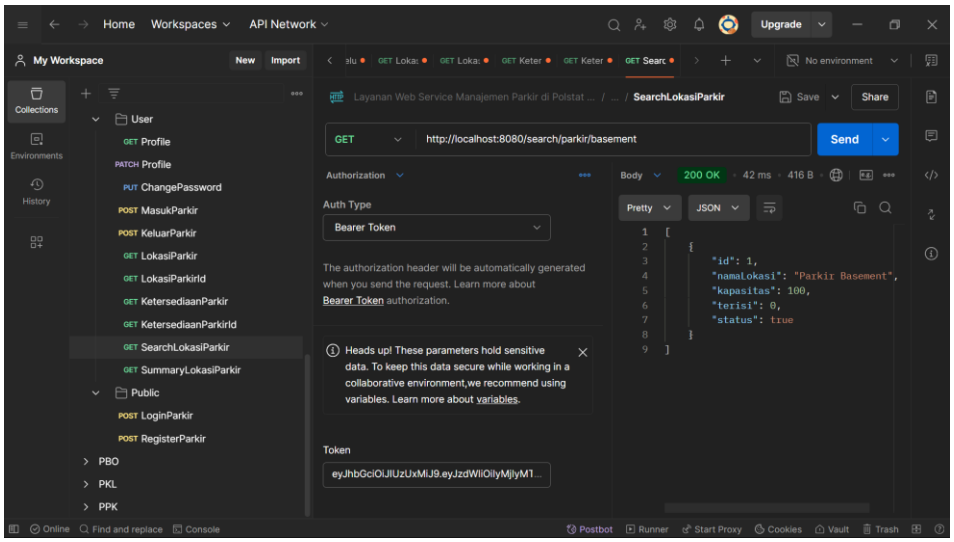
Menampilkan informasi ketersediaan parkir di seluruh lokasi.

9. Melihat Ketersediaan Parkir Berdasarkan Id



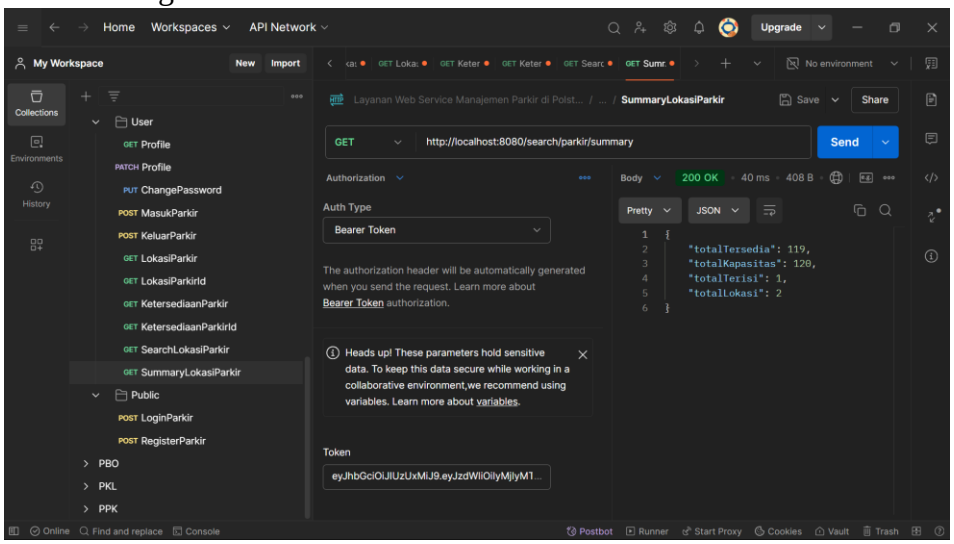
Menampilkan informasi ketersediaan parkir untuk lokasi tertentu.

10. Mencari Lokasi Parkir Berdasarkan Nama



Fitur pencarian lokasi parkir menggunakan kata kunci nama lokasi.

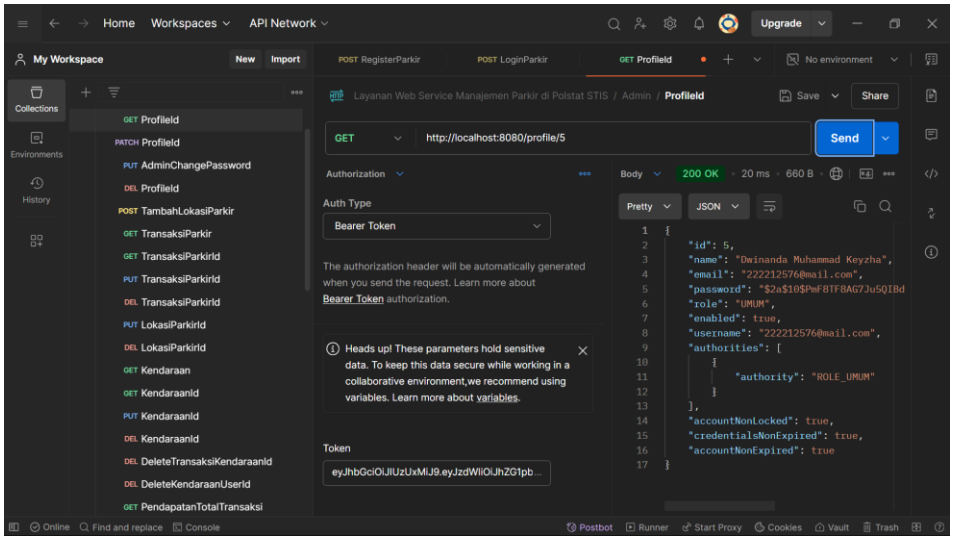
11. Melihat Ringkasan Data dari Lokasi Parkir



Menampilkan statistik dan ringkasan informasi tentang lokasi parkir, termasuk kapasitas dan ketersediaan.

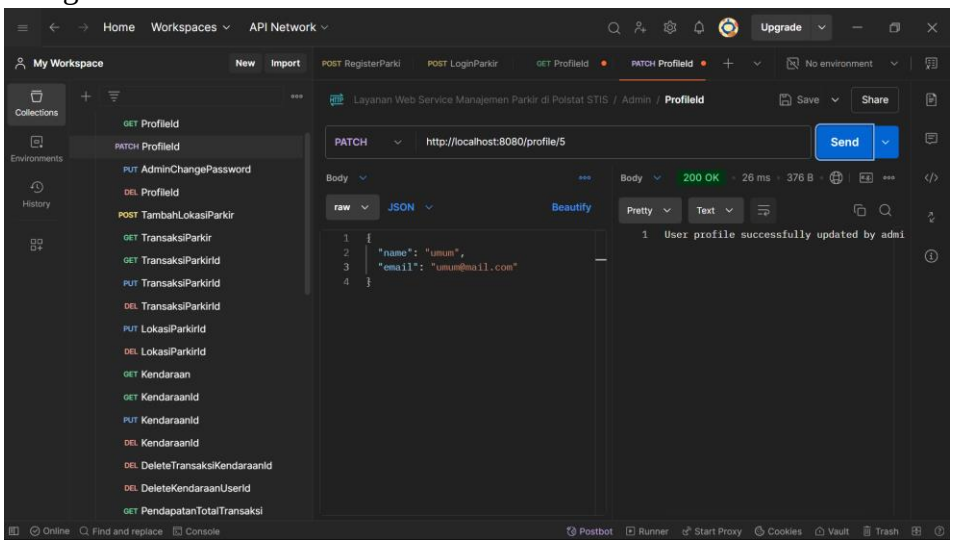
3.1.3 Admin

1. Melihat Profil User Berdasarkan Id



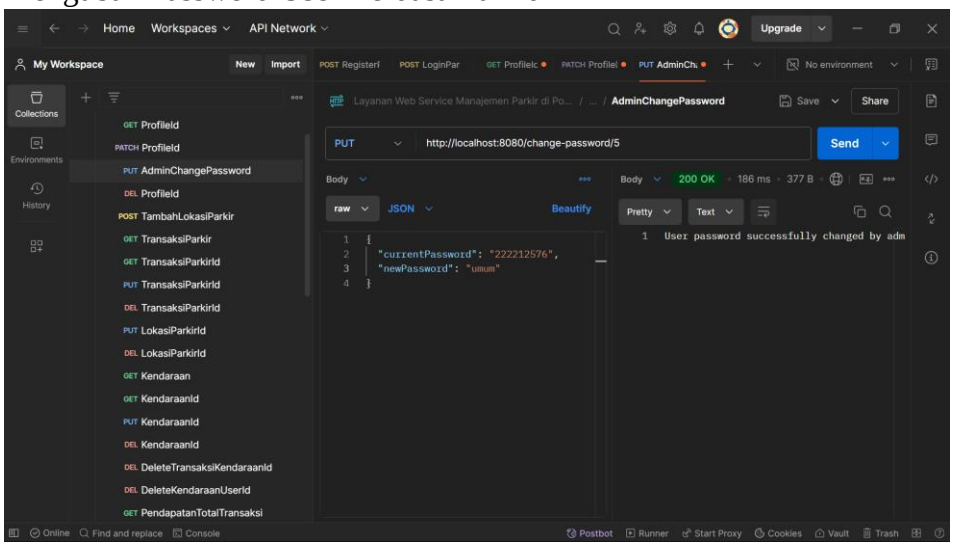
Admin dapat melihat detail profil pengguna tertentu berdasarkan ID.

2. Mengubah Nama dan Email User Berdasarkan Id



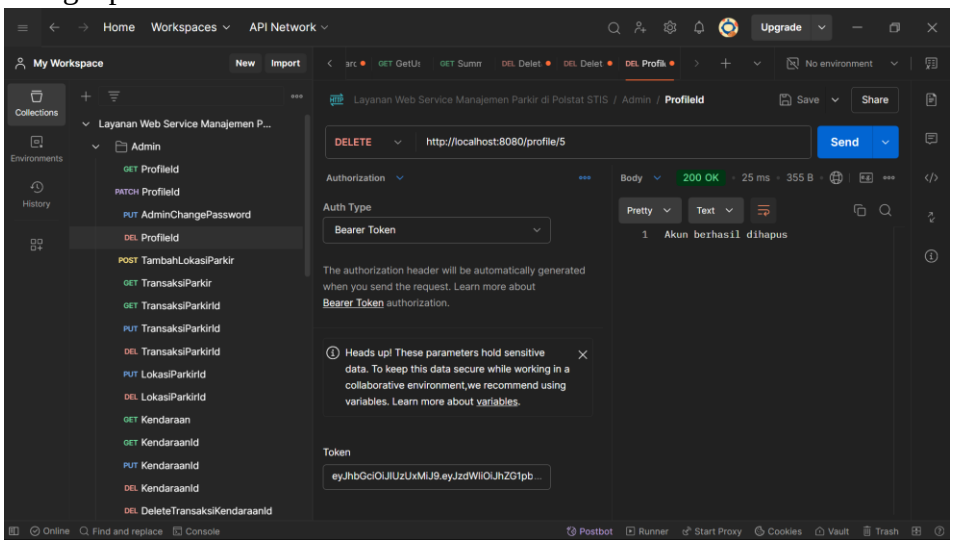
Admin dapat memperbarui informasi profil pengguna tertentu.

3. Mengubah Password User Berdasarkan Id



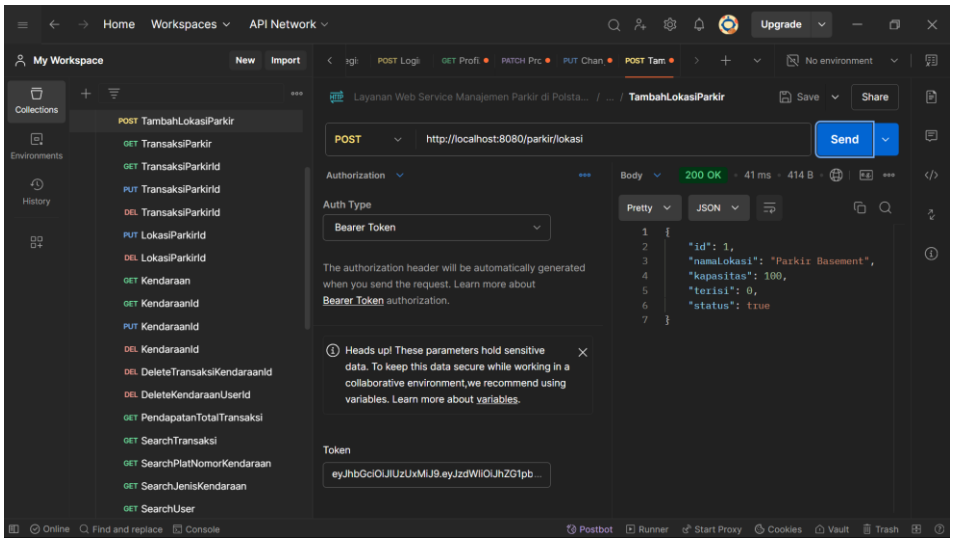
Admin dapat mengatur ulang atau mengubah password pengguna tertentu.

4. Menghapus User Berdasarkan Id



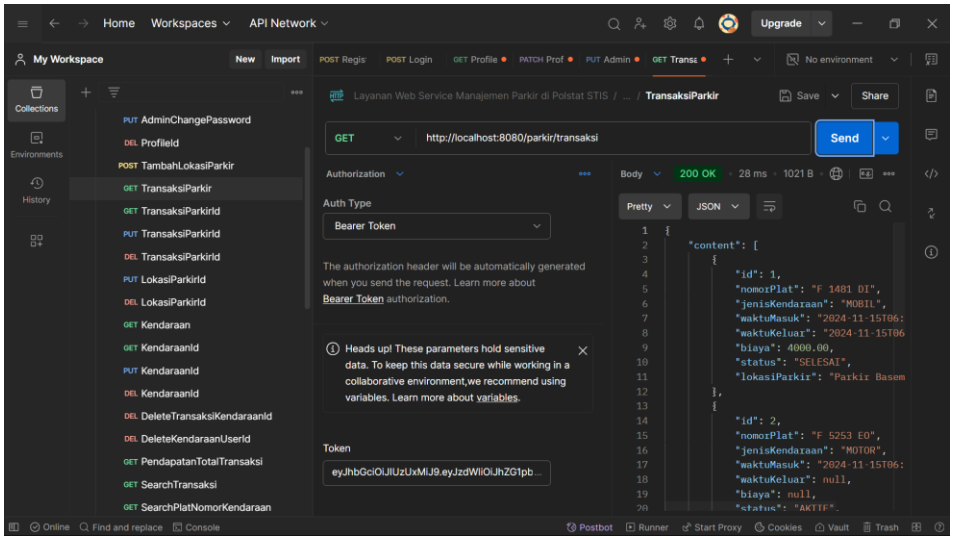
Admin dapat menghapus akun pengguna dari sistem.

5. Menambah Lokasi Parkir



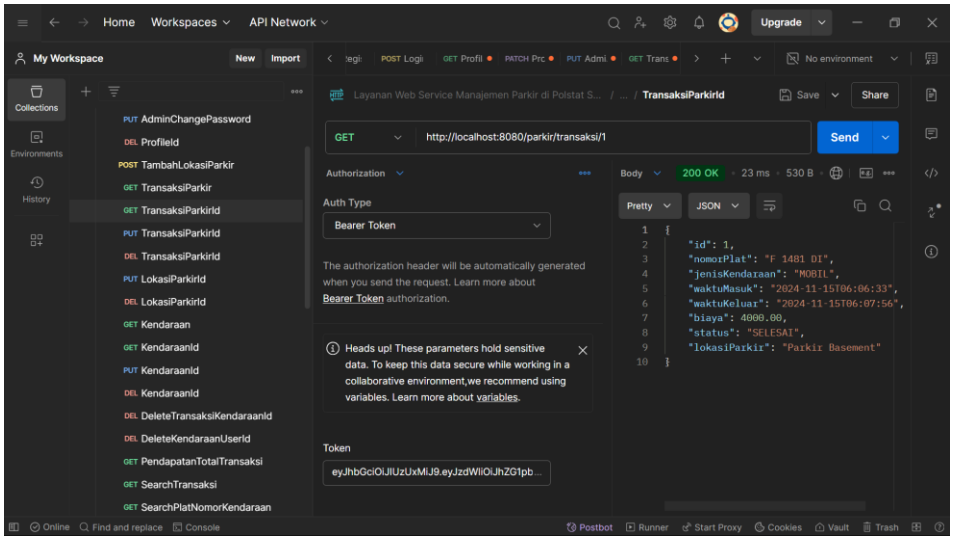
Admin dapat mendaftarkan lokasi parkir baru beserta kapasitasnya.

6. Melihat Transaksi



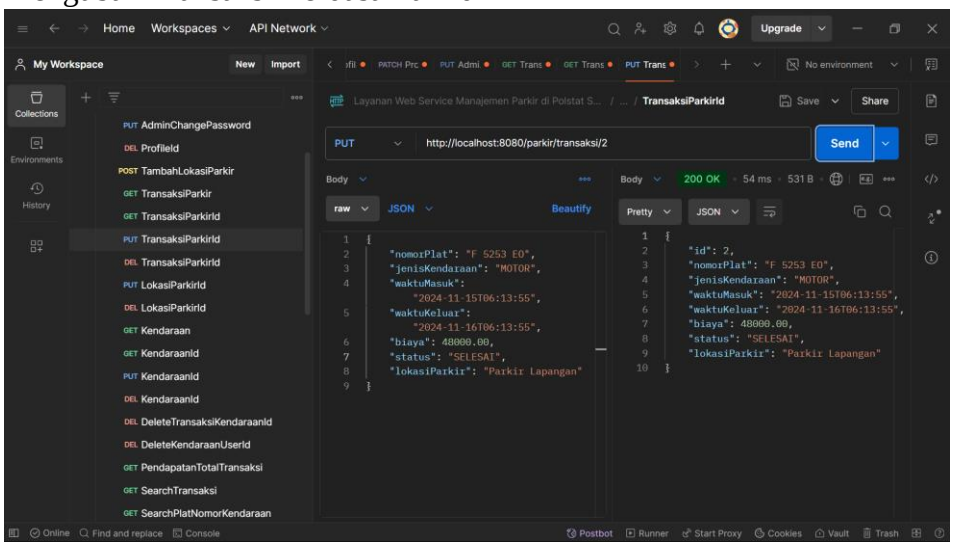
Menampilkan seluruh daftar transaksi parkir dalam sistem.

7. Melihat Transaksi Berdasarkan Id



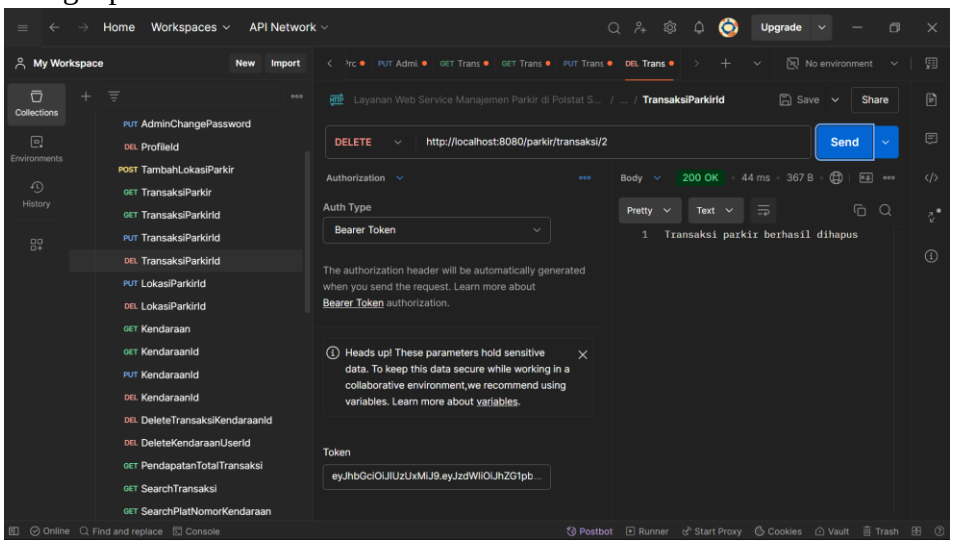
Menampilkan detail transaksi parkir spesifik.

8. Mengubah Transaksi Berdasarkan Id



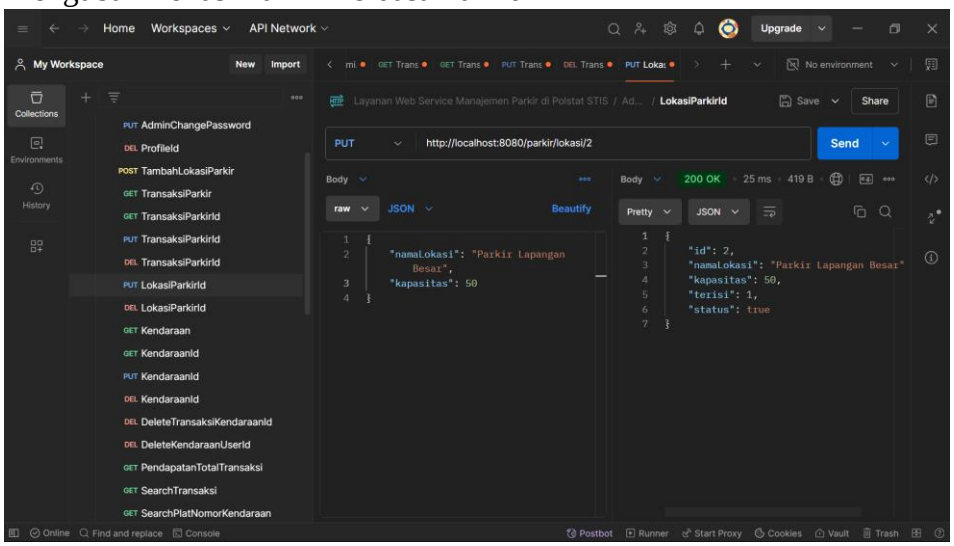
Admin dapat memperbarui informasi transaksi seperti biaya, waktu, dan status.

9. Menghapus Transaksi Berdasarkan Id



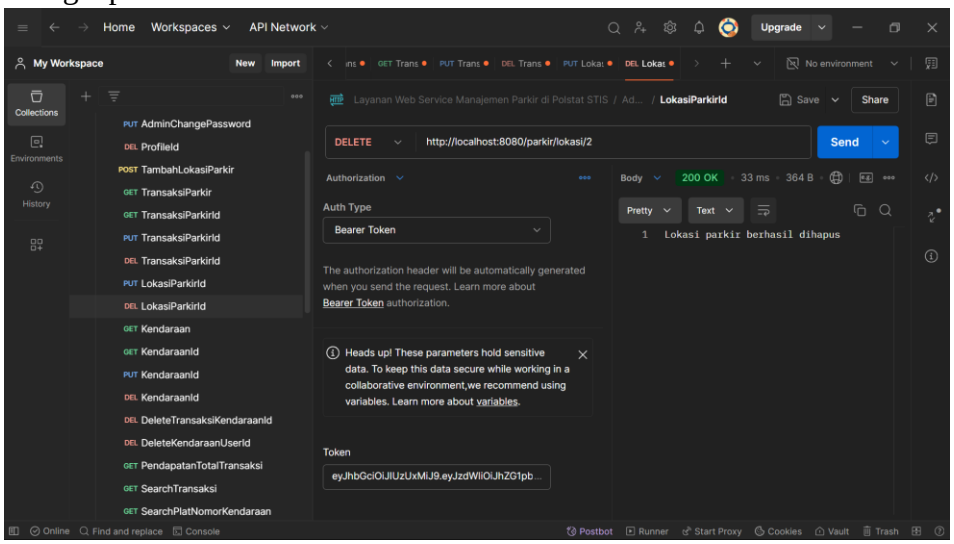
Admin dapat menghapus record transaksi tertentu.

10. Mengubah Lokasi Parkir Berdasarkan Id



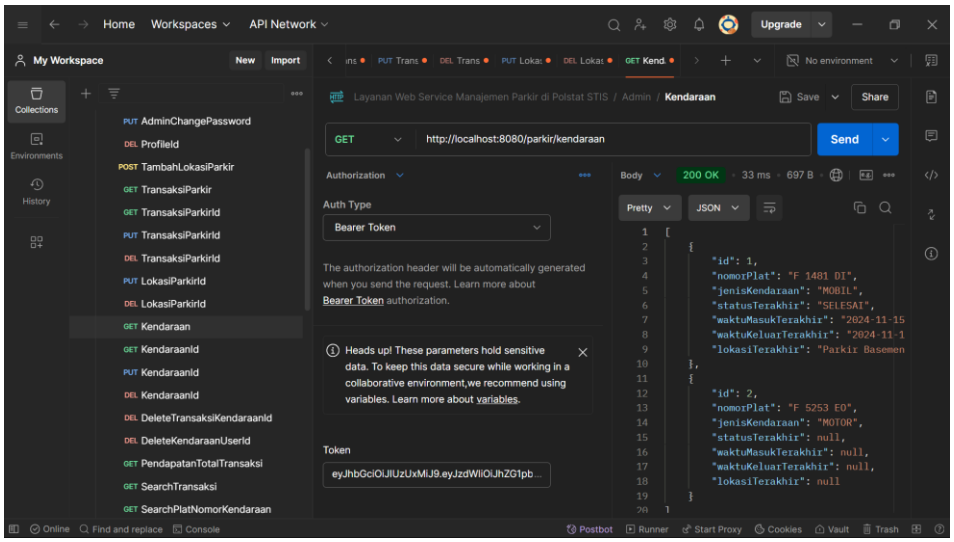
Admin dapat memperbarui informasi lokasi parkir seperti nama dan kapasitas.

11. Menghapus Lokasi Parkir Berdasarkan Id



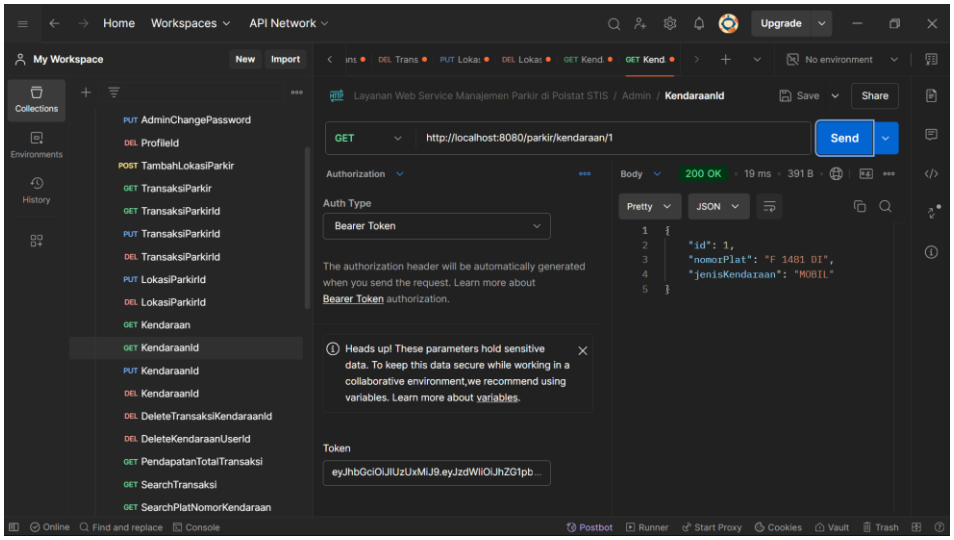
Admin dapat menghapus lokasi parkir dari sistem.

12. Melihat Kendaraan



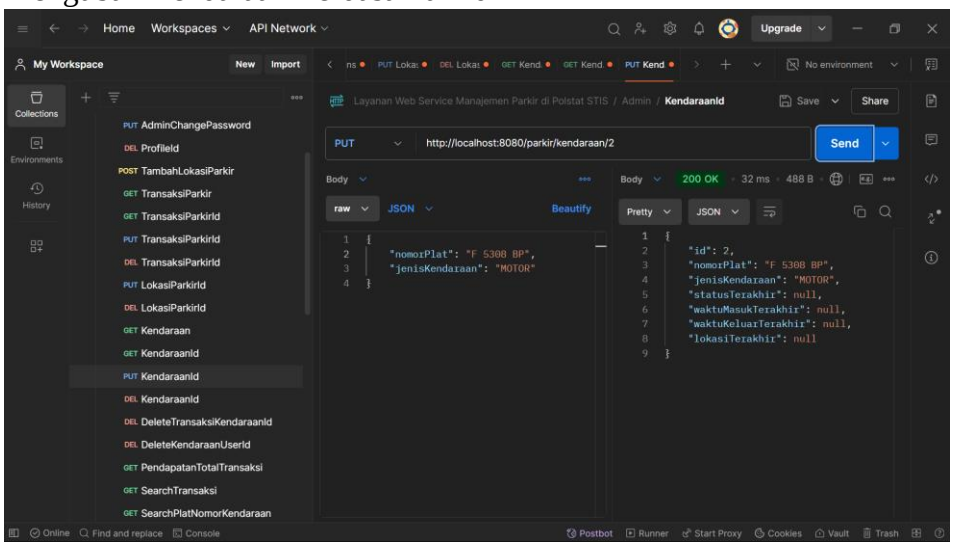
Menampilkan daftar seluruh kendaraan yang terdaftar.

13. Melihat Kendaraan Berdasarkan Id



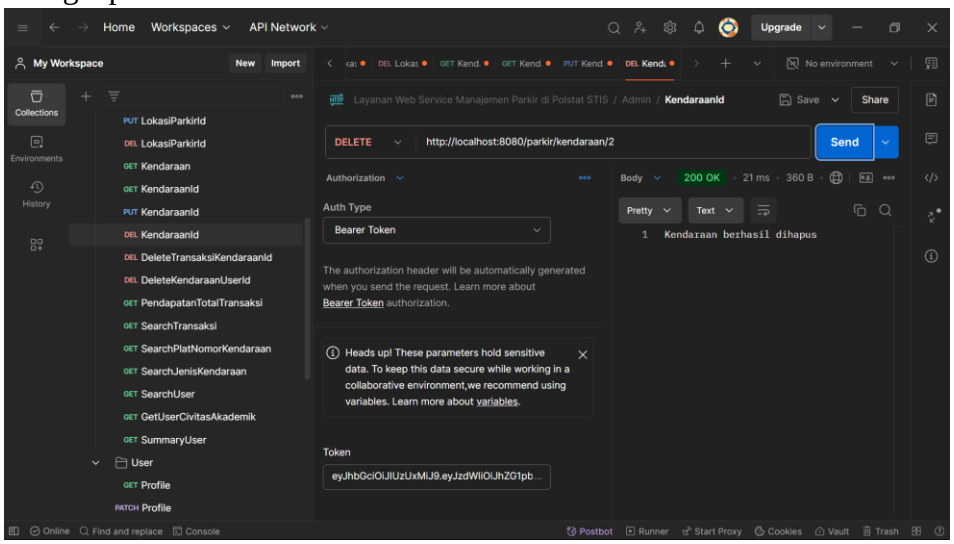
Menampilkan detail kendaraan spesifik.

14. Mengubah Kendaraan Berdasarkan Id



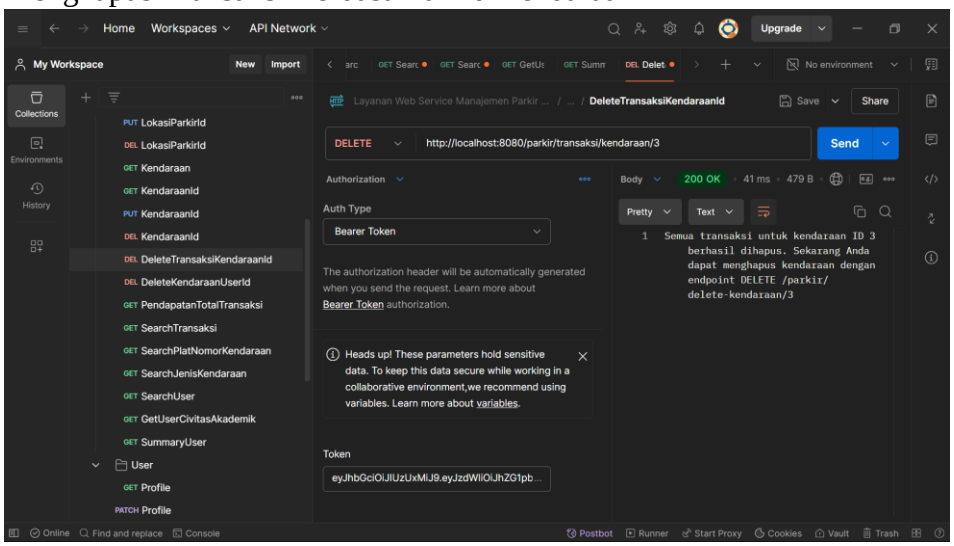
Admin dapat memperbarui informasi kendaraan seperti nomor plat dan jenis.

15. Menghapus Kendaraan Berdasarkan Id



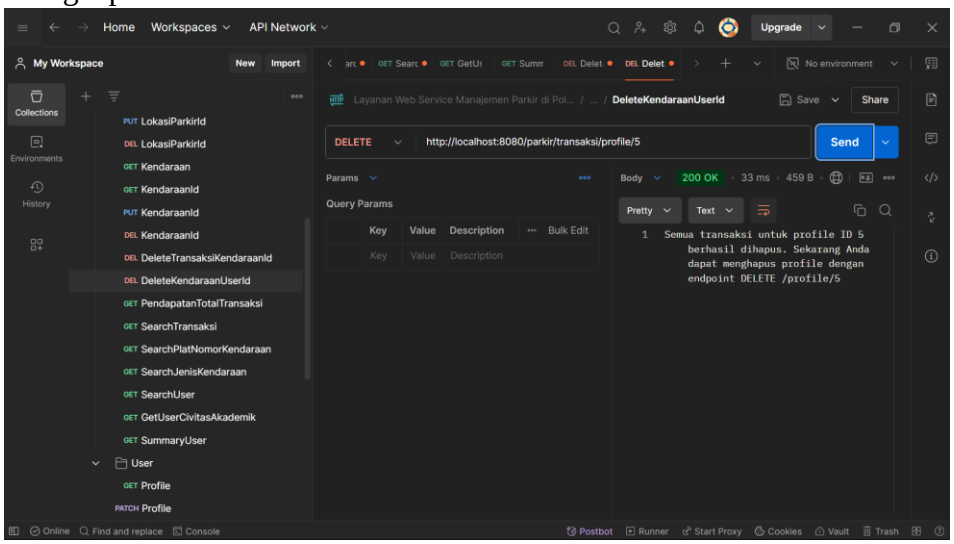
Admin dapat menghapus data kendaraan dari sistem.

16. Menghapus Transaksi Berdasarkan Id Kendaraan



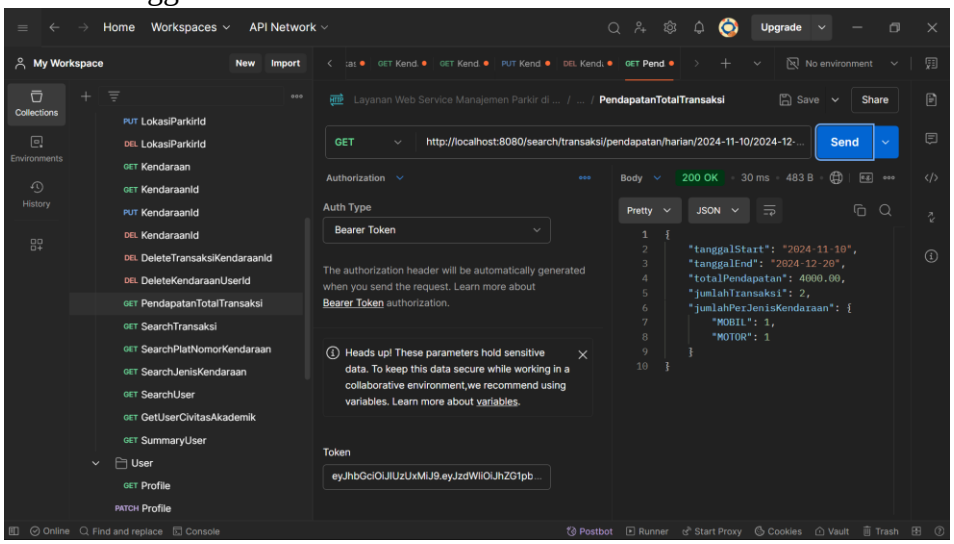
Admin dapat menghapus semua transaksi terkait dengan kendaraan tertentu.

17. Menghapus Transaksi Berdasarkan Id User



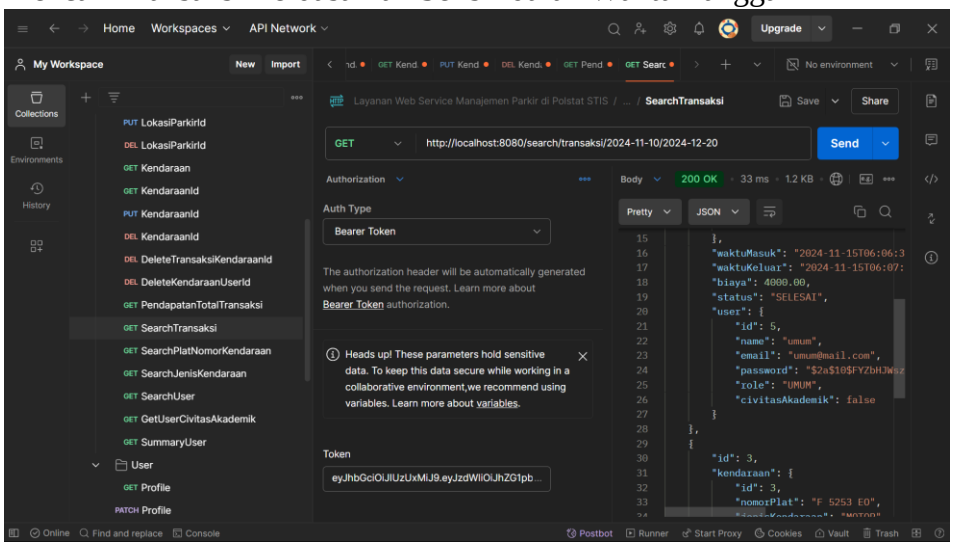
Admin dapat menghapus semua transaksi terkait dengan pengguna tertentu.

18. Melihat Pendapatan Total Harian Transaksi Berdasarkan Selisih Jarak Waktu Tanggal



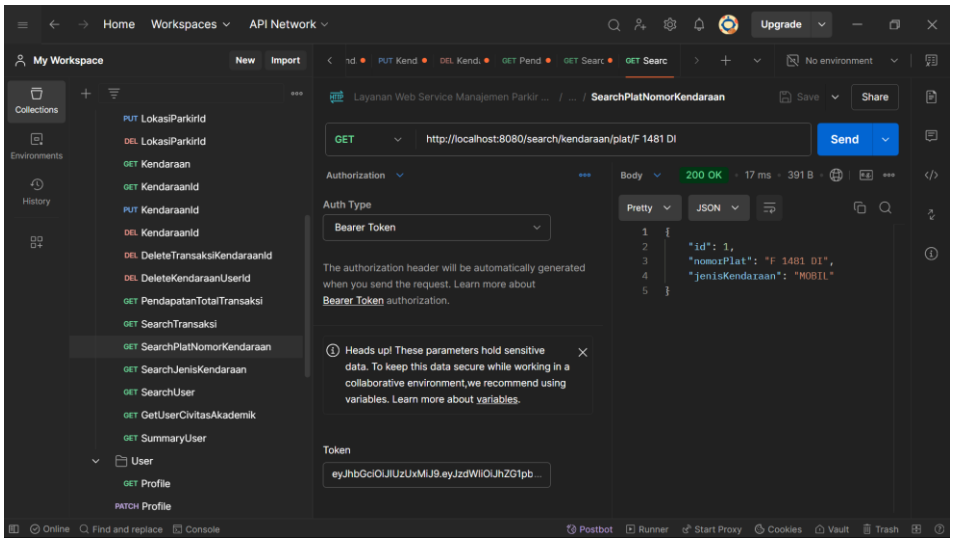
Menampilkan laporan pendapatan dalam rentang waktu tertentu.

19. Mencari Transaksi Berdasarkan Selisih Jarak Waktu Tanggal



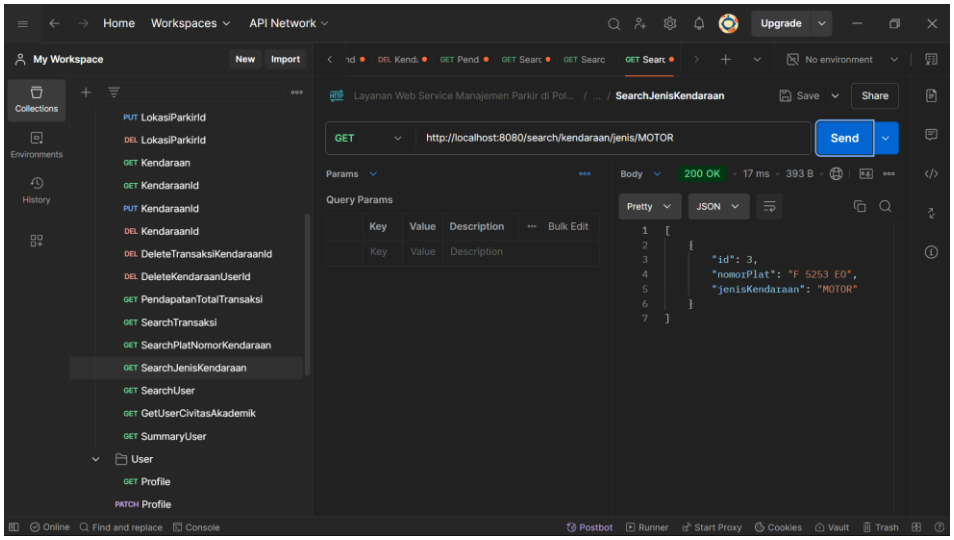
Mencari dan menampilkan transaksi dalam periode waktu tertentu.

20. Mencari Kendaraan Berdasarkan Pelat Nomor



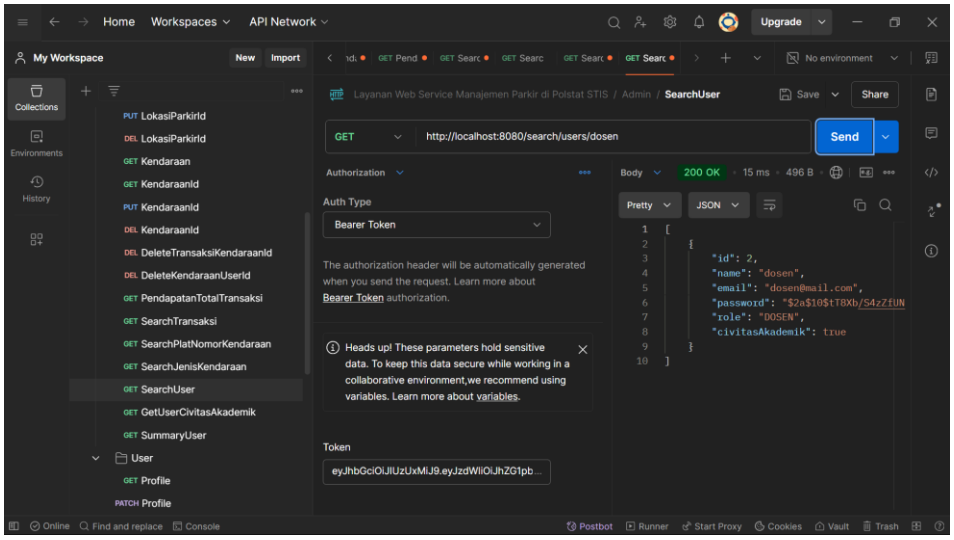
Fitur pencarian kendaraan menggunakan nomor plat.

21. Mencari Kendaraan Berdasarkan Jenis Kendaraan



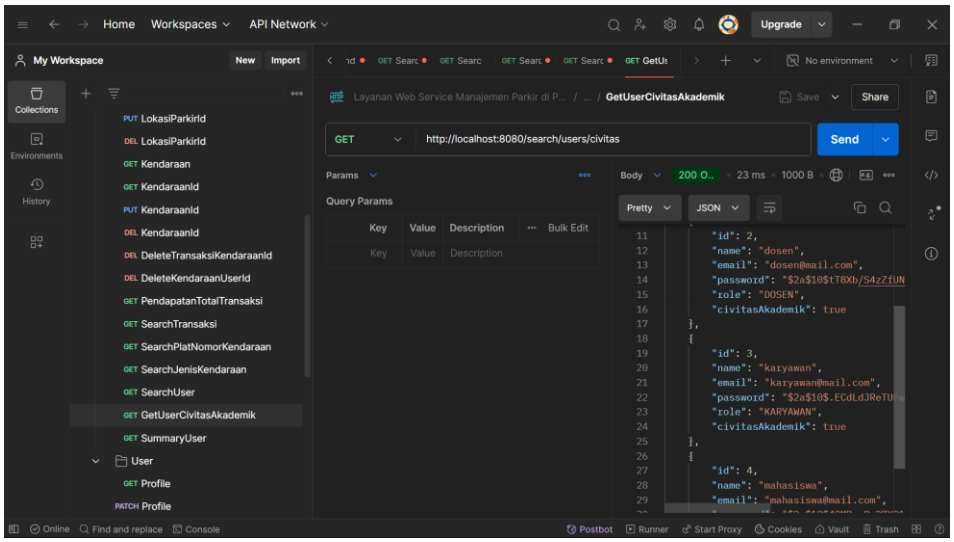
Fitur pencarian kendaraan berdasarkan jenisnya (MOTOR/MOBIL).

22. Mencari User Berdasarkan Nama atau Email



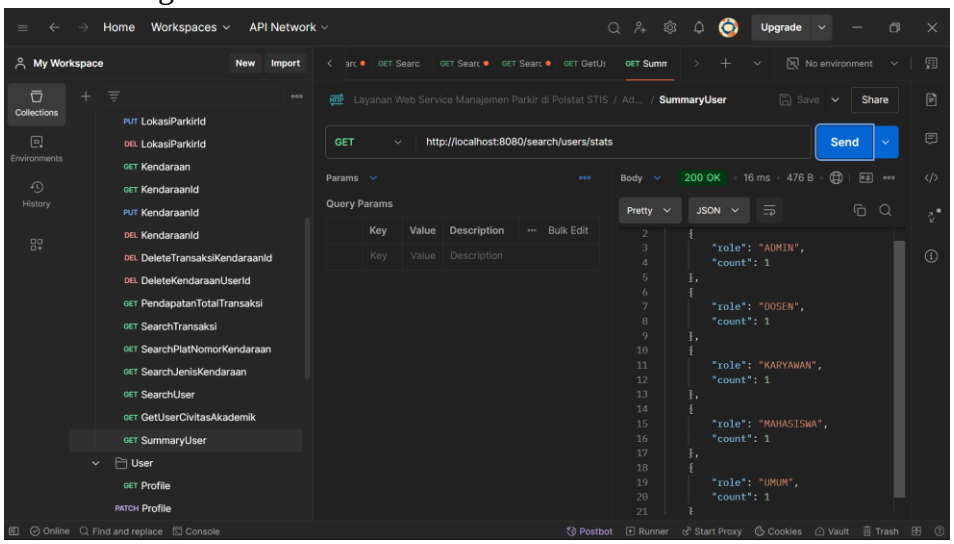
Fitur pencarian pengguna menggunakan kata kunci nama atau email.

23. Melihat User Berdasarkan Role Civitas Akademik



Menampilkan daftar pengguna yang termasuk dalam kategori civitas akademik.

24. Melihat Ringkasan Data dari User



Menampilkan statistik dan ringkasan informasi tentang pengguna sistem.

BAB 4
ISI DAN DOKUMENTASI

4.1 Kesimpulan

Berdasarkan hasil pengembangan dan pengujian Layanan Web Service Manajemen Parkir di Polstat STIS, dapat ditarik beberapa kesimpulan:

1. Sistem berhasil mengotomatisasi pengelolaan parkir yang sebelumnya dilakukan secara manual, dengan mengimplementasikan fitur-fitur utama seperti:
   a. Registrasi dan autentikasi pengguna dengan berbagai role (UMUM, DOSEN, MAHASISWA, KARYAWAN)
   b. Manajemen check-in dan check-out kendaraan
   c. Pemantauan ketersediaan parkir secara real-time
   d. Pencatatan dan pelacakan transaksi parkir
   e. Diferensiasi biaya parkir antara civitas akademika dan umum
2. Sistem menyediakan antarmuka REST API yang komprehensif dengan dokumentasi yang jelas melalui Swagger UI dan Postman, memudahkan integrasi dengan berbagai platform client.
3. Implementasi keamanan sistem telah diterapkan dengan baik melalui:
   a. JWT (JSON Web Token) untuk autentikasi
   b. Role-based access control untuk otorisasi
   c. Validasi input pada setiap endpoint
   d. Enkripsi password pengguna
4. Sistem berhasil mengakomodasi kebutuhan berbagai pemangku kepentingan:
   a. Admin dapat mengelola seluruh aspek sistem
   b. Pengguna dapat dengan mudah mengakses informasi parkir
   c. Civitas akademika mendapatkan privilege khusus terkait biaya parkir

4.2 Saran

Untuk pengembangan sistem di masa mendatang, beberapa saran yang dapat dipertimbangkan:

1. Peningkatan Fungsionalitas:
   a. Implementasi sistem notifikasi real-time untuk status parkir
   b. Penambahan fitur reservasi parkir
   c. Integrasi dengan sistem pembayaran digital untuk pengguna umum
   d. Pengembangan dashboard analitik yang lebih komprehensif
2. Optimasi Performa:
   a. Implementasi caching untuk meningkatkan kecepatan respons
   b. Optimasi query database untuk operasi yang sering digunakan
   c. Penerapan pagination untuk data dalam jumlah besar
3. Peningkatan Keamanan:
   a. Implementasi rate limiting untuk mencegah abuse
   b. Penambahan sistem logging yang lebih detail
   c. Penerapan monitoring sistem secara real-time
4. Pengembangan Antarmuka:

   a. Pembuatan aplikasi mobile untuk kemudahan akses
   b. Pengembangan dashboard admin yang lebih interaktif
   c. Integrasi dengan sistem kamera pengawas
5. Dokumentasi dan Testing:
   a. Penambahan unit testing yang lebih komprehensif
   b. Pembuatan dokumentasi teknis yang lebih detail
   c. Penyusunan user guide untuk pengguna akhir

Dengan implementasi saran-saran di atas, diharapkan sistem dapat berkembang menjadi solusi manajemen parkir yang lebih handal dan memberikan pengalaman pengguna yang lebih baik di masa mendatang.