

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi informasi yang semakin pesat menyebabkan berkembangnya kebutuhan akan informasi dan digitalisasi data dalam semua bidang kehidupan, termasuk dalam bidang bisnis. Pemanfaatan teknologi informasi dalam bidang bisnis ini merupakan salah satu pemenuhan terhadap persaingan usaha dan bisnis oleh pemilik bisnis untuk senantiasa meningkatkan kualitas usahanya. Pencatatan terhadap produk yang dibeli oleh konsumen di setiap cabang bisnis awalnya dilakukan secara manual. Data transaksi pembeli tersebut dianalisis untuk mendapatkan evaluasi atau informasi yang dibutuhkan untuk pengambilan keputusan. Pendataan transaksi pada produk yang dilakukan secara manual memiliki kekurangan, seperti kesalahan tulis pada jumlah transaksi produk, kesalahan harga, dan kesalahan perhitungan yang dapat menyebabkan kerugian bagi pemilik bisnis tersebut. Selain itu, pencatatan secara manual juga mempersulit proses analisis data apabila ingin mengintegrasikan data yang berada di cabang yang lain. Salah satu penerapan teknologi informasi dalam bidang usaha ini adalah aplikasi *Point of Sales* (POS).

Aplikasi POS merupakan suatu sistem untuk membuat proses transaksi menjadi lebih efisien dan sekaligus untuk manajemen bisnis secara keseluruhan. Aplikasi POS dapat meminimalisir kesalahan dalam pendataan transaksi dan produk, serta kesalahan analisis yang dapat muncul ketika dilakukan secara manual. Di sisi lain, aplikasi POS juga dapat melakukan manajemen bisnis, seperti manajemen karyawan, pelanggan, cabang, produk, dan kategori produk.

CV. Gama Putra Santosa merupakan perusahaan yang didirikan pada tahun 2011 yang memiliki fokus di bidang IT dan videografi. CV. Gama Putra Santosa juga menawarkan pengembangan aplikasi untuk kebutuhan UMKM dan juga *game* edukasi. Salah satu aplikasi yang akan dibuat adalah aplikasi *mobile* POS yang ditujukan untuk UMKM, khususnya warung atau restoran. Aplikasi *mobile* POS yang dibuat harus dapat terkoneksi dengan internet, sehingga pemilik bisnis dapat mengecek transaksi yang terjadi dan mengelola bisnis secara *real-time*. Pengecekan

dan pengelolaan secara *real-time* dalam aplikasi *mobile* dapat diraih dengan menyiapkan tempat penyimpanan data atau *database di cloud*, kemudian aplikasi *mobile* berkomunikasi dengan *database* melalui *endpoint API (Application Programming Interface)*.

Pengembangan aplikasi *mobile* POS di CV. Gama Putra Santosa dilakukan secara tim oleh empat orang mahasiswa magang yang terdiri dari *system analyst*, *UI/UX designer*, *front-end mobile developer*, dan *back-end developer*. Penulis bertanggung jawab sebagai *back-end developer* dalam pengembangan aplikasi ini. Seorang *back-end developer* bertugas untuk membuat struktur *database* sebagai tempat penyimpanan data dan API sebagai penghubung antara *database di cloud* dan aplikasi *mobile*.

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, maka dapat diidentifikasi bahwa masalah yang muncul adalah sebagai berikut:

1. Sistem pencatatan transaksi produk yang dilakukan secara manual rawan akan kesalahan.
2. Integrasi data yang terdapat di cabang yang berbeda sulit dilakukan apabila data masih dicatat secara manual.
3. Data transaksi dan analisis tidak dapat dilihat secara *real-time* oleh pemilik bisnis.

C. Batasan Masalah

Adapun batasan masalah pada kegiatan praktik magang pengembangan sistem *back-end* aplikasi POS ini, antara lain:

1. Pengembangan aplikasi POS berfokus pada transaksi atau penjualan produk pada pelanggan dan manajemen bisnis (karyawan, cabang, pelanggan, produk, dan kategori produk)
2. Pengembangan sistem *back-end* aplikasi POS dilakukan menggunakan bahasa pemrograman PHP dengan *framework* Laravel.
3. Pengembangan sistem *back-end* dibuat dalam bentuk REST API.

D. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka terdapat rumusan masalah yaitu bagaimana cara membangun dan mengembangkan sistem *back-end* aplikasi POS.

E. Tujuan Praktik Magang

Tujuan dilaksanakannya kegiatan praktik magang pengembangan sistem *back-end* aplikasi POS ini adalah sebagai berikut:

1. Membangun sistem *back-end* aplikasi POS.
2. Membuat dokumentasi API untuk sistem *back-end* aplikasi POS.

F. Manfaat Praktik Magang

Manfaat dilaksanakannya kegiatan praktik magang pengembangan sistem *back-end* aplikasi POS ini adalah sebagai berikut:

1. Mahasiswa terlatih untuk menerapkan keterampilan yang telah diperoleh selama studi ke dalam ranah industri.
2. Mahasiswa mendapat pengalaman sebagai persiapan untuk menghadapi tantangan dunia kerja.
3. Mahasiswa dapat mengembangkan keterampilan untuk berkomunikasi dan bekerja sama, serta memahami tugasnya dalam tim.

BAB II

GAMBARAN INSTANSI

A. Umum

1. Sejarah Instansi

Gama Putra Santosa (GPS) Studio didirikan pada tahun 2011 oleh Bambang Robi'in dan Ari Hadiyanto dengan tujuan membentuk perusahaan yang berfokus pada industri IT dan videografi. Pada tanggal 28 Mei 2011, pendirian perusahaan ini secara resmi tercatat melalui akta notaris nomor 17 yang disahkan oleh Siti Asmaul Khusnah, S.H.

Fokus utama GPS Studio adalah menyediakan layanan di bidang IT dan videografi, seperti memberikan konsultasi kepada pengguna untuk pembuatan aplikasi, serta menyediakan peralatan yang diperlukan untuk kegiatan videografi, termasuk kamera, *lighting*, dan tempat. CV. Gama Putra Santosa juga telah aktif sebagai mitra pemerintah dalam penyediaan barang dan jasa melalui proses tender (LPSE).

Pada tahun 2015, CV. Gama Putra Santosa mengalihkan fokusnya ke bidang perangkat lunak, dengan kegiatan utama sebagai konsultan IT, pengembangan *software* sesuai kebutuhan, dan pembuatan aplikasi. Pada tahun 2016, GPS Studio memulai pengembangan aplikasi *mobile* dengan nama pengembang Sahabat Studio. Hingga saat ini, GPS Studio tetap aktif dalam industri IT dan videografi, mengikuti perkembangan teknologi.

Pada tahun 2021, selama masa pandemi, GPS Studio lebih banyak terlibat dalam pembuatan *podcast* untuk menyampaikan informasi tentang usaha UMKM di Yogyakarta, bekerja sama dengan Dinas Perindustrian dan Perdagangan Daerah. Selain itu, GPS Studio juga memulai pembangunan *game* edukasi berbasis *mobile* untuk meningkatkan proses pembelajaran.



Gambar 2.1 Logo CV. Gama Putra Santosa

2. Visi dan Misi

Adapun visi dan misi CV. Gama Putra Santosa adalah:

a. Visi:

Menjadi Perusahaan IT yang Dinamis dan Edukatif.

b. Misi:

- 1) Menyediakan lapangan pekerjaan,
- 2) Menyediakan wadah belajar dan mengembangkan ilmu untuk kebutuhan industri,
- 3) Mengembangkan produk-produk inovatif edukatif.

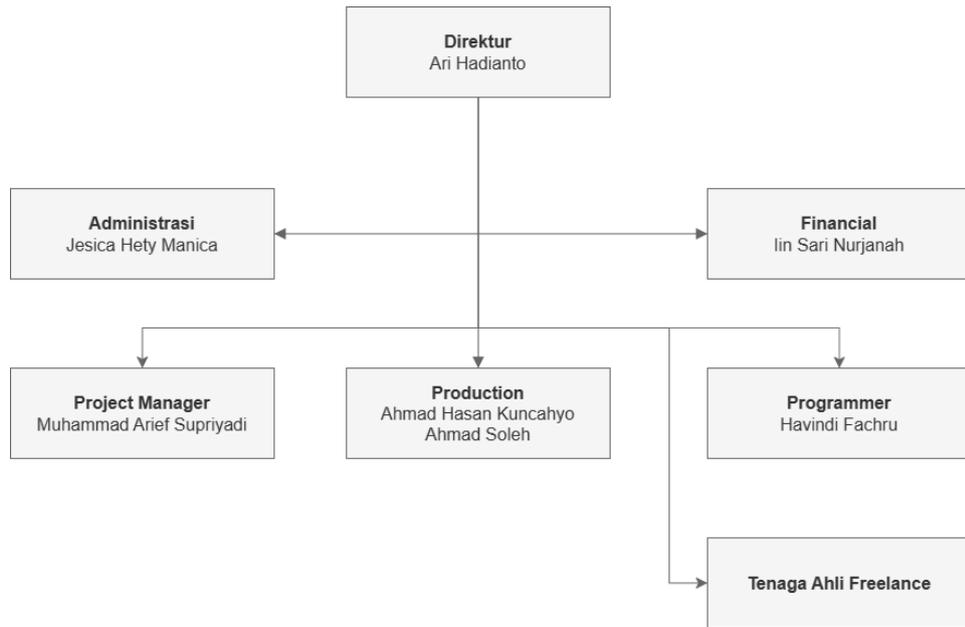
3. Alamat dan Kontak Instansi

CV. Gama Putra Santosa berlokasi di Jalan Wiyoro Lor No. 143, Baturetno, Kecamatan Banguntapan, Kabupaten Bantul, Daerah Istimewa Yogyakarta 55197. Kontak 081575625555.

B. Struktur Organisasi di Tempat Magang

CV. Gama Putra Santosa merupakan perusahaan yang aktif di bidang videografi dan IT dimana perusahaan GPS sebagai penyedia jasa videografi dan menjadi konsultan IT. Tentu perusahaan sulit untuk berjalan jika hanya 1 orang di dalamnya, oleh karena itu perusahaan ini memiliki struktur organisasi sebagai penentu keberhasilan perusahaan dalam mencapai tujuan. Struktur organisasi menggambarkan suatu hubungan, wewenang dan tanggung jawab setiap tingkatan yang dimiliki seseorang dalam organisasi. Struktur organisasi disusun dengan tujuan keberhasilan dalam bekerja sama dalam mengorganisir suatu progres perusahaan.

Struktur organisasi pada CV. Gama Putra Santosa ditunjukkan pada Gambar 2.2 berikut.



Gambar 2.2 Struktur Organisasi CV. Gama Putra Santosa

C. Sumber Daya Manusia dan Sumber Daya Fisik di Lokasi Magang

Sumber Daya Manusia pada CV. Gama Putra Santosa dapat dilihat pada Tabel 2.1.

Tabel 2.1 Sumber Daya Manusia

Nama	Jabatan
Ari Hadiano	Direktur
lin Sari Nurjanah	Financial
Jesica Hety Manica	Administrasi
Muhammad Arief Supriyadi	Project Manager
Ahmad Hasan Kuncahyo	Production
Ahmad Soleh Seftian	Production
Havindi Fahru	Programmer

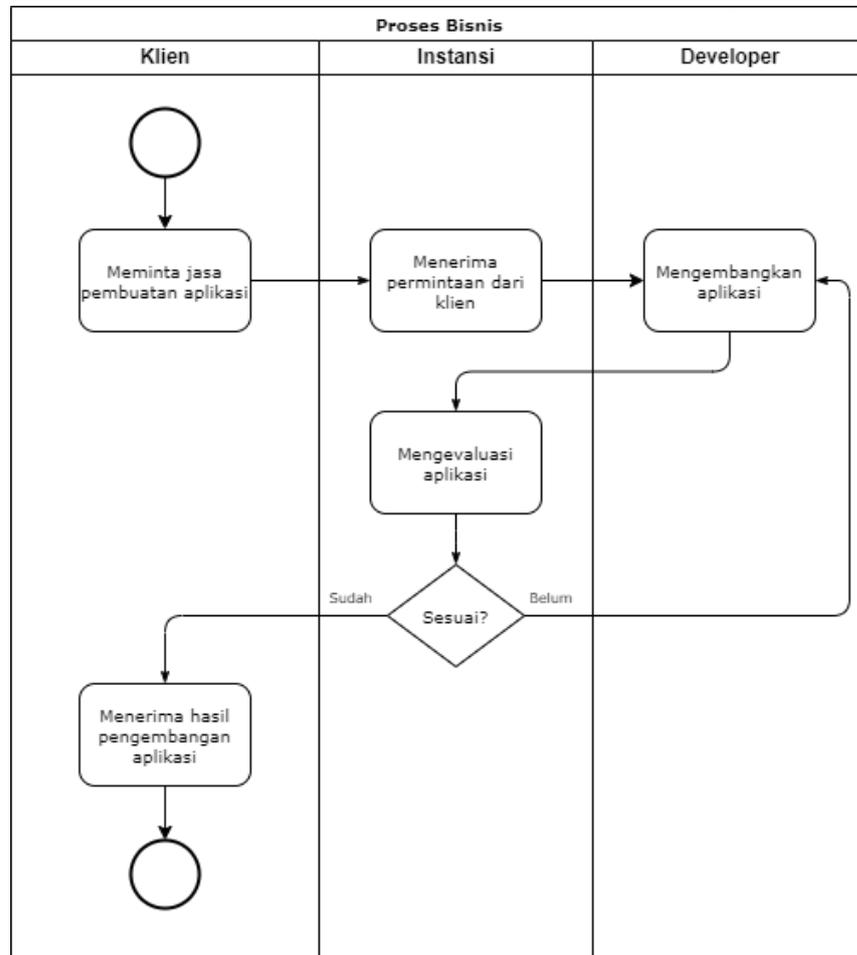
Sumber Daya Fisik pada CV. Gama Putra Santosa dapat dilihat pada Tabel 2.2.

Tabel 2.2 Sumber Daya Fisik

Nama	Keterangan
Ruang Meeting	1 Ruang
Ruang Tamu	1 Ruang
Studio Podcast	1 Ruang

D. Proses Bisnis saat ini yang Berhubungan dengan Tema Praktik Magang

Proses bisnis saat ini yang berhubungan dengan tema praktik magang yang dilakukan saat di CV. Gama Putra Santosa dapat dilihat pada Gambar 2.3.



Gambar 2.3 Proses Bisnis saat ini yang Berhubungan dengan Tema Praktik Magang

BAB III

TAHAPAN KEGIATAN PRAKTIK MAGANG

A. Lokasi Praktik Magang, Alamat, Kontak

Lokasi Praktik Magang berada di Kantor CV. Gama Putra Santosa dengan alamat di Jalan Wiyoro Lor No. 143, Baturetno, Kecamatan Banguntapan, Kabupaten Bantul, Daerah Istimewa Yogyakarta 55197. Kontak pembimbing lapangan Praktik Magang yang dapat dihubungi yaitu Muhammad Arief Supriyadi dengan nomor kontak 081575625555.

B. Rencana Observasi

Pelaksanaan observasi tempat magang dilakukan pada tanggal 3 Oktober 2023. Observasi dilakukan untuk mengetahui tentang proyek yang akan dilakukan dan tugas penulis sebagai pemegang dengan mewawancarai narasumber. Berdasarkan hasil observasi, didapatkan bahwa proyek yang akan dikerjakan yaitu aplikasi *Point of Sales (POS)*, sementara penulis diberikan tugas sebagai *Back End Developer* untuk merancang *database* dan API untuk menghubungkan *database* dengan aplikasi *mobile*.

C. Rancangan Jadwal Kegiatan Magang

Rancangan jadwal kegiatan magang yang dilaksanakan selama 3 bulan atau sekitar 14 minggu dapat dilihat pada Tabel 3.1.

Tabel 3.1 Rencana dan Realisasi Kegiatan Praktik Magang

No	Nama Kegiatan	Minggu Pelaksanaan														Realisasi		
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	Ya/Tidak	%	
1.	Observasi dan pembagian tugas.																Ya	100
2.	Mempelajari dasar framework Laravel untuk REST API.																Ya	100
3.	Brainstorming dan perencanaan proyek.																Ya	100
4.	Membuat rencana REST API dan rancangan database.																Ya	100

5.	Implementasi REST API.																	Ya	100
6.	Pembuatan dokumentasi REST API.																	Ya	100

BAB IV

HASIL PRAKTIK MAGANG

A. Hasil Observasi Magang

1. Proses Bisnis

Berdasarkan proses bisnis yang didapat setelah melakukan observasi magang sebagai *back-end developer* di CV. Gama Putra Santosa, tahapan pertama adalah mencari referensi aplikasi sejenis kemudian menganalisis kebutuhan aplikasi bersama dengan anggota tim. Berdasarkan analisis kebutuhan aplikasi yang telah ada, *back-end developer* mulai merancang struktur *database* yang akan digunakan. Setelah itu, *UI/UX designer*, *mobile front-end developer*, dan *back-end developer* berkomunikasi untuk membuat rancangan REST API, terutama mengenai data apa yang perlu dikirimkan dan ditampilkan. Berdasarkan rancangan REST API, *back-end developer* akan membuat implementasi dalam bentuk kode dan juga disertai dengan dokumentasi API untuk mempermudah *mobile front-end developer* memakai API tersebut.

2. Rincian Pekerjaan

Rincian pekerjaan penulis selama praktik magang di CV. Gama Putra Santosa sebagai *back-end developer* adalah sebagai berikut.

- a. Menganalisis struktur dan relasi pada database yang dibutuhkan untuk aplikasi *Point of Sales*.
- b. Mempelajari arsitektur dalam *framework* Laravel dan fitur-fitur yang dibutuhkan untuk pengembangan sistem *back-end* aplikasi *Point of Sales* menggunakan REST API, seperti:
 - 1) Database
 - 2) Eloquent ORM
 - 3) Routing
 - 4) Model
 - 5) Controller
 - 6) Middleware
 - 7) API resource
 - 8) HTTP Request

- 9) Mail
- 10) File storage
- 11) Validation
- 12) Security dan Authorization
- c. Praktik pembuatan REST API menggunakan *framework* Laravel untuk sistem *back-end* aplikasi *Point of Sales*.
 - 1) Mengimplementasikan rancangan struktur dan relasi pada *database* menggunakan *database migrations*.
 - 2) Mengimplementasikan CRUD untuk tabel *users, employees, outlets, products, categories,* dan *customers* menggunakan model dan *controller*.
 - 3) Mengimplementasikan fitur keranjang dengan banyak item dan *invoice*.
 - 4) Mengimplementasikan token untuk *user* dan *employee* sebagai pengenalan dan *authorization*.
- d. Membuat dokumentasi API menggunakan OpenAPI.

B. Pembahasan Magang

1. Problem yang Ditemukan di Tempat Magang sebagai Topik Praktik Magang

Masalah yang ditemukan di CV. Gama Putra Santosa sebagai topik praktik magang sebagai *back-end developer* adalah:

- a. Proses pengembangan aplikasi *mobile* POS memerlukan *database* untuk sebagai tempat penyimpanan data.
- b. Proses pengembangan aplikasi *mobile* POS memerlukan REST API sebagai jembatan komunikasi antara aplikasi *mobile* dengan *database* yang akan ada di *cloud*.
- c. Diperlukan dokumentasi API sebagai acuan bagi *mobile front-end developer* dalam menggunakan API.

2. Analisis terhadap Hasil Observasi

Setelah melakukan observasi terhadap bagaimana permasalahan dalam proses pengembangan aplikasi *mobile* POS, maka berikut adalah hasil analisisnya:

- a. Analisis struktur dan relasi pada *database*.

Database untuk sistem aplikasi POS memiliki beberapa tabel, yaitu tabel *users*, *employees*, *employees_outlets*, *outlets*, *products*, *categories*, *customers*, *invoices*, *invoice_items*, dan *carts*. Masing-masing tabel memiliki relasi satu sama lain, yaitu:

- 1) Tabel *users* memiliki relasi dengan *employees*, *outlets*, *products*, *categories*, dan *customers* dengan relasi *one to many*.
 - 2) Tabel *employees* memiliki relasi *many to many* dengan *outlets*, sehingga diperlukan tabel baru bernama *employees_outlets*. Tabel *employees* juga memiliki relasi *one to many* dengan *invoices* dan *carts*.
 - 3) Tabel *outlets* memiliki relasi *one to many* dengan *invoices*.
 - 4) Tabel *products* memiliki relasi *many (optional) to one (optional)* dengan *categories*. Tabel *products* juga memiliki relasi *one to many* dengan *invoice_items* dan *carts*.
 - 5) Tabel *customers* memiliki relasi *one (optional) to many* dengan *invoices*.
 - 6) Tabel *invoices* memiliki relasi *one to many* dengan *invoice_items*.
- b. Analisis REST API yang dibutuhkan untuk aplikasi POS menggunakan *framework* Laravel.

Berikut adalah kebutuhan API yang dibutuhkan sebagai perantara komunikasi aplikasi *mobile* POS dengan *database*.

- 1) Membuat *user* baru.
- 2) Mengirim kode OTP ke *email user*.
- 3) Memverifikasi *user* menggunakan kode OTP yang dikirimkan ke *email user*.
- 4) *Login user*.
- 5) Mendapatkan data *user* yang sedang masuk.
- 6) Mengubah data *user* yang sedang masuk.
- 7) *Logout user*.
- 8) Membuat *outlet* atau cabang baru bagi *user*.
- 9) Mendapatkan semua *outlet* yang terhubung dengan *user*.
- 10) Mendapatkan data *outlet* dari suatu *user* menggunakan id *outlet*.
- 11) Mengubah data *outlet* dari suatu *user* menggunakan id *outlet*.
- 12) Menghapus *outlet* milik *user* menggunakan id *outlet*.

- 13) Mendaftarkan akun karyawan atau *employee* dengan *role* pemilik menggunakan data *user*.
- 14) Menambahkan karyawan dari suatu *user*.
- 15) *Login* akun karyawan dari *user*.
- 16) Mendapatkan semua data karyawan dari suatu *user* sekaligus melakukan pencarian berdasarkan nama atau *outlet* tempat karyawan berada.
- 17) Mendapatkan data karyawan dari suatu *user* menggunakan id karyawan.
- 18) Mengubah data karyawan dari suatu *user* menggunakan id karyawan.
- 19) Menghapus data karyawan dari suatu *user* menggunakan id karyawan.
- 20) *Logout* akun karyawan.
- 21) Menambahkan pelanggan atau *customer* dari *user* tertentu.
- 22) Mendapatkan semua data pelanggan dari *user* tertentu sekaligus melakukan pencarian berdasarkan nama.
- 23) Mendapatkan data pelanggan dari *user* menggunakan id pelanggan.
- 24) Mengubah data pelanggan dari *user* menggunakan id pelanggan.
- 25) Menghapus data pelanggan dari *user* menggunakan id pelanggan.
- 26) Menambahkan kategori produk dari *user*.
- 27) Mendapatkan semua data kategori produk dari *user* sekaligus melakukan pencarian berdasarkan nama kategori.
- 28) Mendapatkan data kategori produk dari *user* menggunakan id kategori produk.
- 29) Mengubah data kategori produk dari *user* menggunakan id kategori produk.
- 30) Menghapus data kategori produk dari *user* menggunakan id kategori produk.
- 31) Menambahkan produk dari *user*.
- 32) Mendapatkan semua data produk dari *user* sekaligus melakukan pencarian berdasarkan nama dan id kategori, serta dapat diurutkan berdasarkan abjad atau tanggal dibuat.
- 33) Mendapatkan data produk dari *user* menggunakan id produk.
- 34) Mengubah data produk dari *user* menggunakan id produk.

- 35) Menghapus data produk dari *user* menggunakan id produk.
 - 36) Menghapus gambar produk dari *user* menggunakan id produk.
 - 37) Menambahkan atau memperbarui item produk dengan kuantitas atau diskon tertentu ke dalam keranjang.
 - 38) Mendapatkan data keranjang.
 - 39) Menghapus seluruh item produk dalam keranjang.
 - 40) Menghapus item produk tertentu dalam keranjang menggunakan id produk.
 - 41) Melakukan *checkout*.
 - 42) Menampilkan seluruh *invoice* atau riwayat transaksi yang terhubung dengan *user* sekaligus melakukan penyaringan data berdasarkan id *outlet* maupun tanggal.
 - 43) Mendapatkan detail *invoice* yang terhubung dengan *user* menggunakan id *invoice*.
 - 44) Mengubah data *invoice* yang terhubung dengan *user* menggunakan id *invoice*.
 - 45) Menghapus *invoice* yang terhubung dengan *user* menggunakan id *invoice*.
 - 46) Mendapatkan laporan total penjualan, total transaksi, produk terlaris, dan jenis pembayaran paling sering digunakan dari semua *outlet* atau di *outlet* tertentu sekaligus terdapat penyaringan tanggal.
- c. Membuat dokumentasi API.

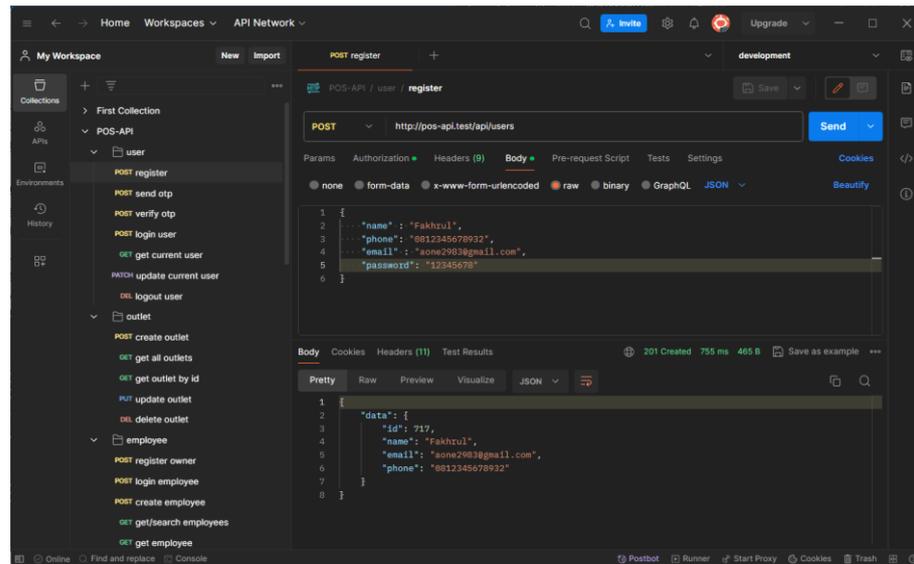
Dokumentasi API dibuat sebagai acuan bagi *mobile front-end developer* untuk menggunakan API yang telah dibuat. Dokumentasi API yang dibuat berisi deskripsi singkat tentang API, daftar semua *endpoint*, parameter, *response*, dan keamanan yang disediakan oleh API, ini juga termasuk format dan tipe data yang digunakan. Dokumentasi API dibuat menggunakan *OpenAPI Specification (OAS)*, yaitu sebuah standar format dokumentasi API yang terstruktur dan terstandar, serta dapat diakses secara publik.

3. Capaian Magang

Setelah melaksanakan praktik magang selama 3 bulan sebagai *back-end developer* di CV. Gama Putra Santosa, terdapat capaian magang sebagai berikut.

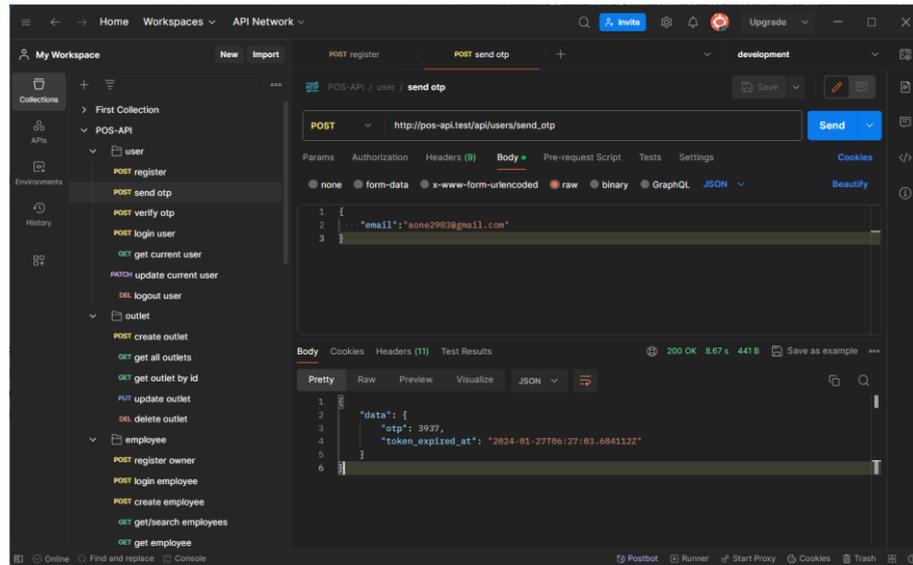
a. Implementasi REST API yang telah dibuat.

Berikut adalah implementasi dan pengujian REST API menggunakan *software* Postman.



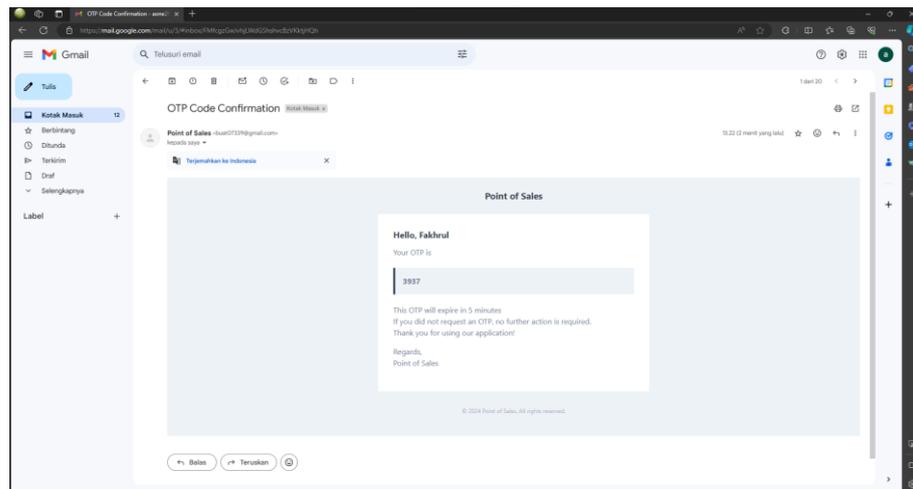
Gambar 4.1 Endpoint untuk Registrasi User

Gambar 4.1 di atas merupakan penerapan *endpoint* untuk registrasi *user*. Data yang dibutuhkan yaitu nama, nomor telepon, *email*, dan *password*. Nomor telepon dan *email* merupakan nilai *unique* yang tidak boleh sama dengan *user* lain. *User* yang telah melakukan registrasi tidak bisa langsung melakukan *login* karena masih perlu proses untuk verifikasi *email user*.



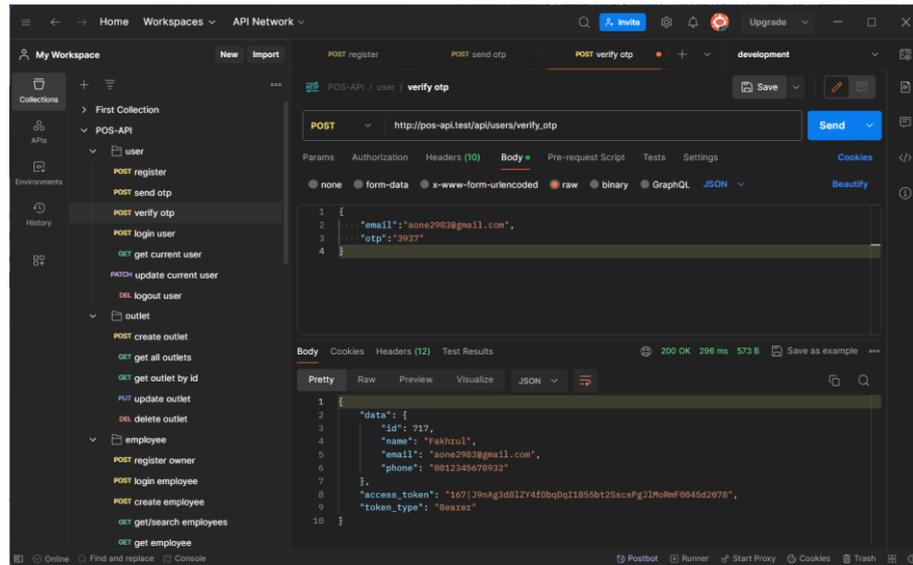
Gambar 4.2 Endpoint untuk Meminta Kode OTP

Kode OTP yang diminta akan dikirimkan melalui *email* yang dimasukkan ke *Request Body*. REST API juga mengirimkan kode OTP melalui *response* yang dimaksudkan untuk proses *testing* menggunakan *email* acak.



Gambar 4.3 Tampilan Email yang Dikirim Oleh Sistem

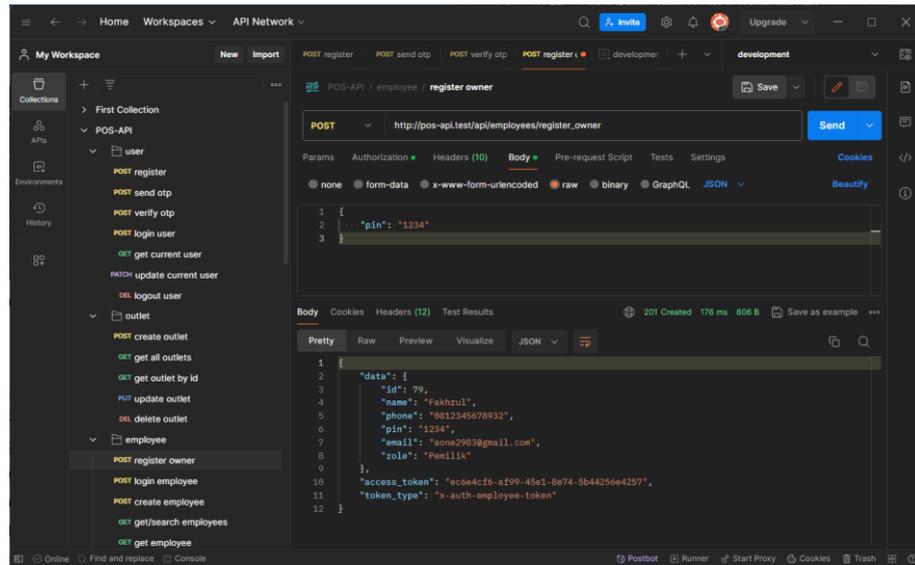
Gambar 4.3 menampilkan kode OTP yang dikirim melalui *email* oleh sistem *back-end*. Kode ini yang nantinya digunakan untuk memverifikasi *email user*, sehingga *user* dapat melakukan *login*.



Gambar 4.4 Verifikasi User Menggunakan Kode OTP

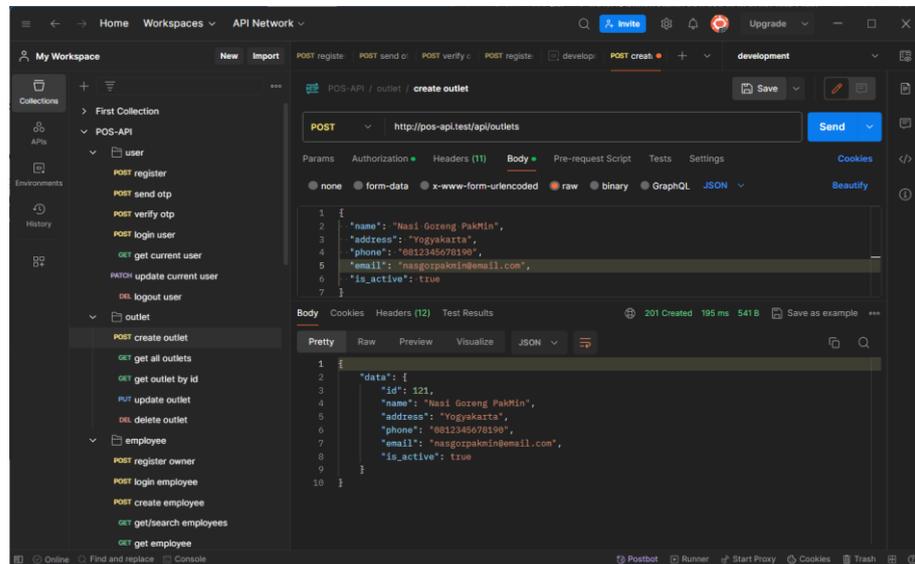
Gambar 4.4 menampilkan kode OTP yang digunakan sebagai proses verifikasi *user*. Endpoint ini juga mengirimkan token akses kepada *user*, sehingga *user* tidak perlu melakukan proses *login*. Token akses ini berguna untuk mengakses *endpoint* lain. Tanpa token akses, *endpoint* lain akan menampilkan pesan error “Unauthorized”.

Kemudian untuk mengakses *endpoint* dalam fitur utama, seperti manajemen *outlet*, *employee*, *customer*, *product*, *category*, *cart*, *invoice*, dan *report*, diperlukan token akun *employee*. Token ini juga sebagai penanda *role* bagi *employee* dan juga berlaku sebagai pembatas hak akses terhadap *endpoint* tertentu. Karena itu, *user* diwajibkan untuk mendaftarkan akun *employee* mereka menggunakan *endpoint* “*api/employees/register_owner*” untuk mendaftarkan diri secara otomatis menggunakan data yang terdapat dalam akun *user*. Di sini hanya memerlukan masukan pin untuk menambahkan data pin pada akun *employee* yang memiliki *role* sebagai *owner* atau pemilik.



Gambar 4.5 Endpoint Register Owner

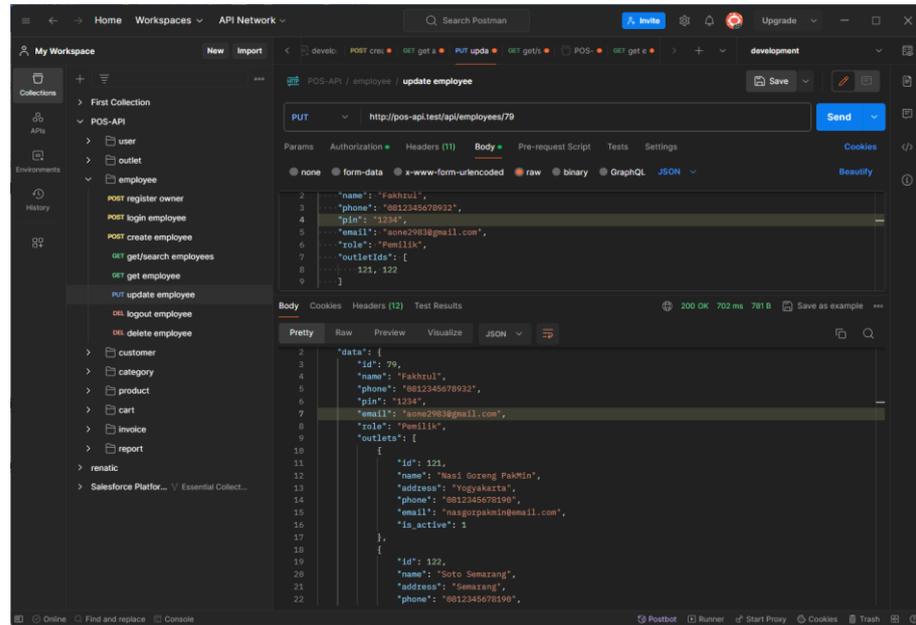
Gambar 4.5 menunjukkan *endpoint register owner* yang memberikan *response* yang berisi data akun karyawan yang memiliki *role* sebagai pemilik. *Response* juga berisi token akses beserta tipe token yang akan digunakan pada *endpoint-endpoint* yang lain bersamaan dengan token *user*. Token akses *employee* ini juga dapat diperoleh menggunakan fitur *login employee*.



Gambar 4.6 Endpoint untuk Menambahkan Outlet

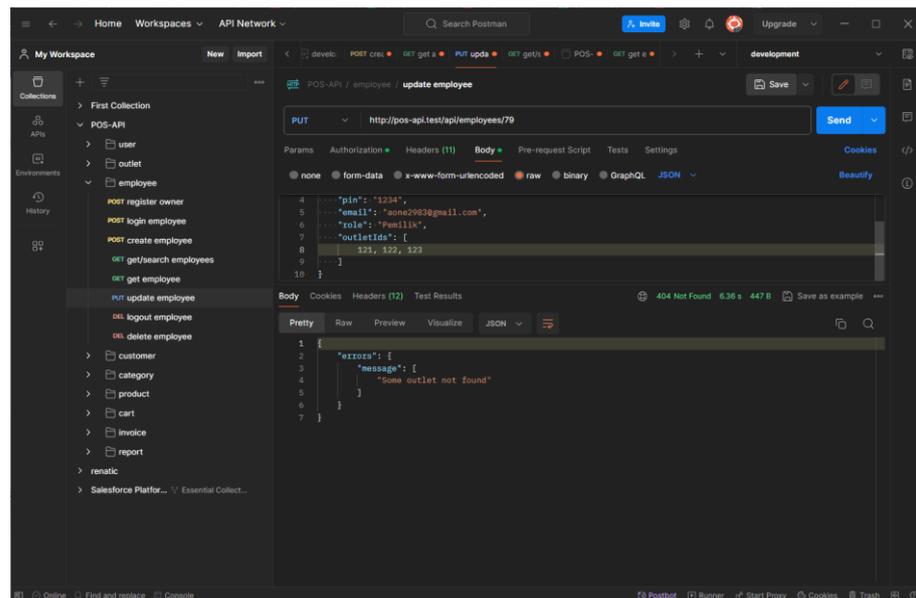
Gambar 4.6 menunjukkan *endpoint* untuk menambahkan *outlet* ke *user* yang terhubung. Proses menghubungkan *outlet* dengan akun *user* ini dilakukan secara otomatis di dalam sistem *back-end*. Untuk menghubungkan *outlet* dengan *employee*, diperlukan *test* perubahan melalui *endpoint update*

employee dengan menambahkan id *outlet* yang ingin dihubungkan, seperti yang terlihat pada Gambar 4.7 berikut.



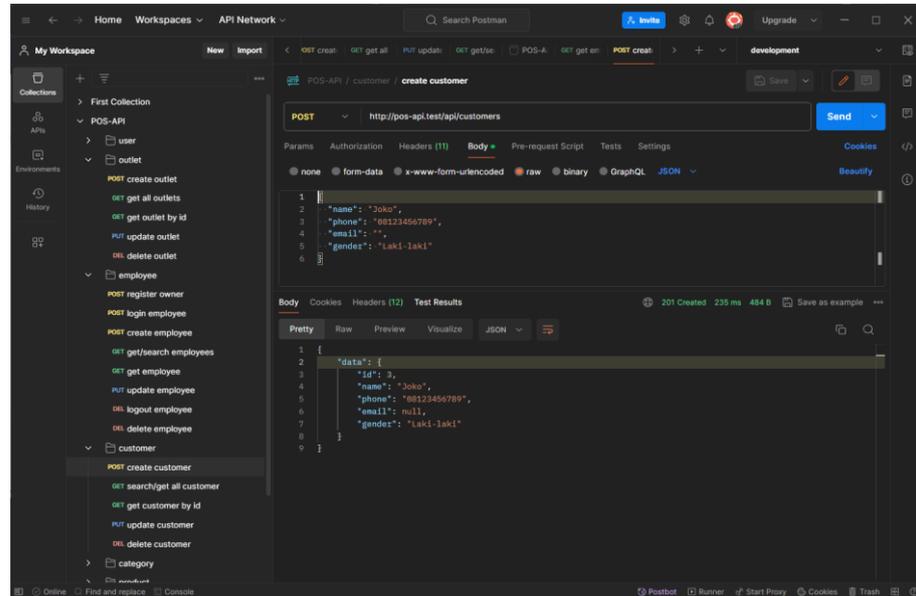
Gambar 4.7 Endpoint Update Employee

Gambar 4.7 menunjukkan *endpoint* untuk mengubah data *employee* dengan id *employee* untuk mengidentifikasi data *employee* mana yang akan diubah. Untuk menambahkan outlet yang terhubung dengan *employee*, tambahkan id *outlet* pada *field* "outletIds" dalam bentuk *array*. *Response* akan menghasilkan error apabila terdapat id *outlet* yang tidak sesuai atau tidak ada.



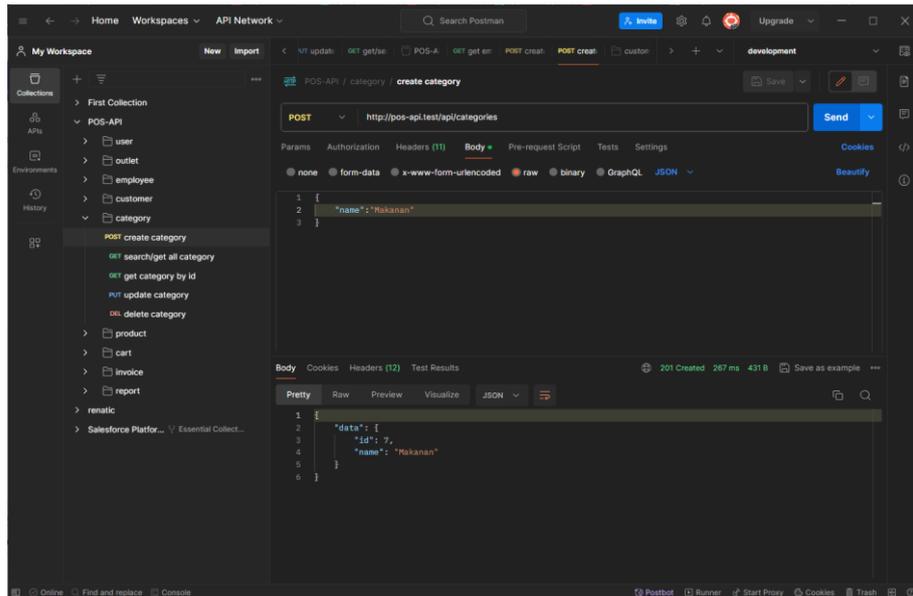
Gambar 4.8 Error Handling Ketika Id Outlet Tidak Ditemukan

Gambar 4.8 menunjukkan *error handling* yang dilakukan ketika id *outlet* yang dimasukkan ke *endpoint* tidak ada di dalam *database*. *Error handling* yang sama juga berlaku pada *endpoint* lain, seperti *endpoint* untuk mendapatkan data *outlet*, *employee*, *customer*, *category*, *product*, dan *invoice* menggunakan id terkait.



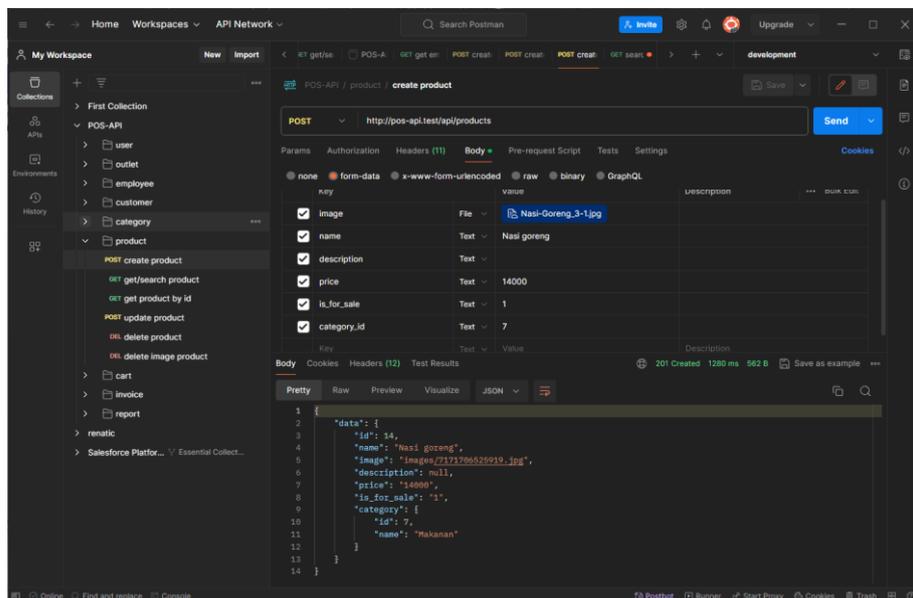
Gambar 4.9 Endpoint untuk Menambahkan Customer

Gambar 4.9 merupakan *endpoint* untuk menambahkan *customer* pada akun *user* terkait. Data *customer* ini dapat digunakan ketika sedang dilakukan transaksi. *Endpoint* lain untuk *customer* ini hanya berupa operasi dasar CRUD (*Create, Read, Update, Delete*). Operasi dasar ini juga diterapkan dalam *endpoint* yang berhubungan dengan *category*.



Gambar 4.10 Endpoint untuk Menambahkan Category

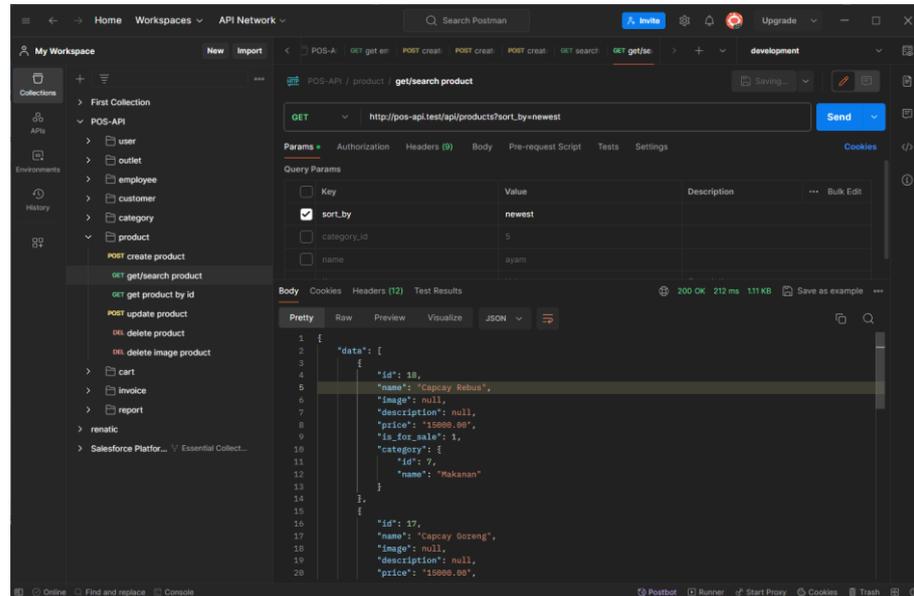
Gambar 4.10 menunjukkan *endpoint* untuk menambahkan *category* pada akun *user*. *Category* ini digunakan untuk mengkategorikan jenis produk dan untuk digunakan sebagai *filter* apabila dibutuhkan ketika mencari produk.



Gambar 4.11 Endpoint untuk Menambahkan Product

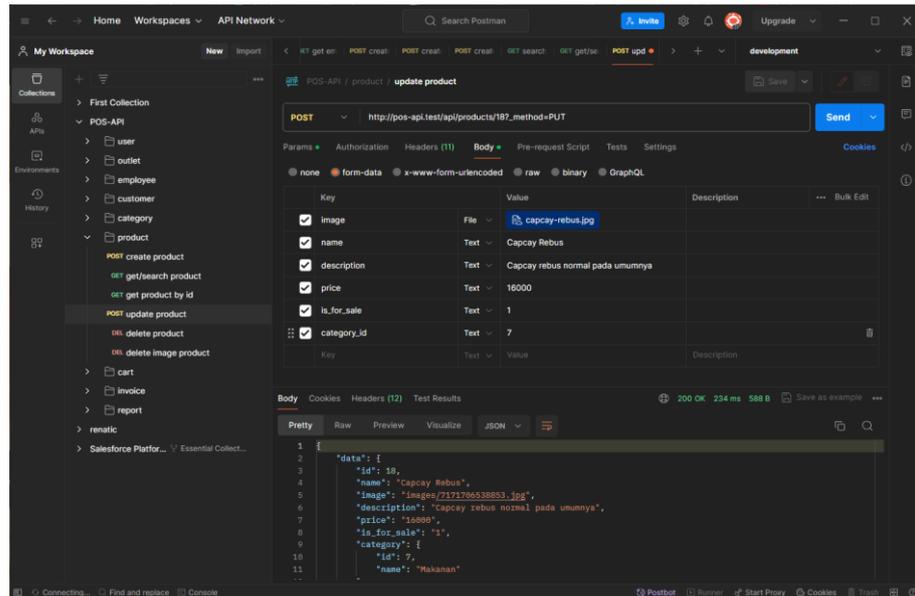
Gambar 4.11 menunjukkan *endpoint* untuk menambahkan *product* pada akun *user*. Produk ini memiliki tiga *field* yang harus diisi, yaitu *name*, *price*, dan *is_for_sale*, sedangkan *field* lain seperti *image*, *description*, dan

category_id bisa dibiarkan kosong. Pengiriman dilakukan menggunakan *form-data* apabila terdapat *file* atau dalam hal ini gambar yang disertakan.



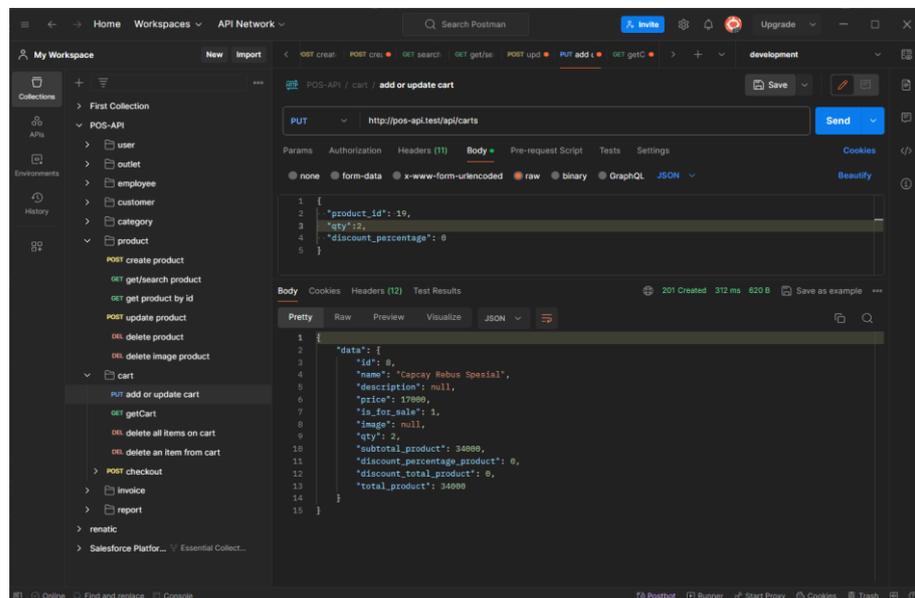
Gambar 4.12 Endpoint untuk Mencari atau Mendapatkan Data Product

Gambar 4.12 menunjukkan *endpoint* untuk mencari atau mendapatkan data *product* yang terhubung dengan *user*. Terdapat beberapa parameter yang dapat dimasukkan ke dalam *endpoint* ini, seperti *sort_by*, *category_id*, dan *name*. Parameter *sort_by* dapat mengurutkan data berdasarkan abjad ataupun waktu ditambahkan. Parameter *category_id* dapat menyaring data hanya pada data *product* yang memiliki id *category* yang sesuai. Parameter *name* dapat menyaring data hanya pada data *product* yang memiliki nama yang sesuai.



Gambar 4.13 Endpoint Update Product

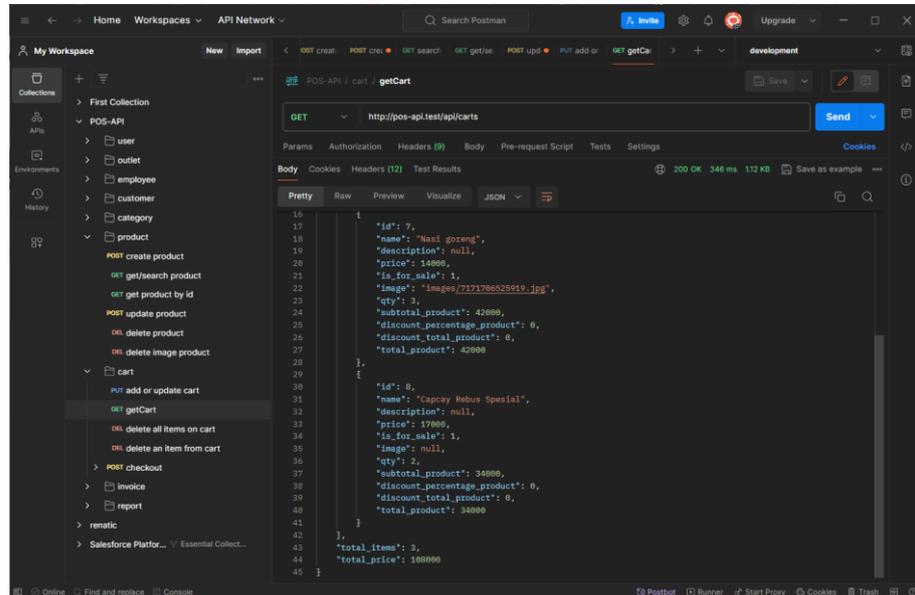
Gambar 4.13 menunjukkan *endpoint* yang digunakan untuk mengubah data *product*. *Endpoint* yang digunakan untuk mengubah data biasanya menggunakan *method* PUT, tetapi terdapat hal khusus untuk data yang dikirim dalam bentuk form-data karena keterbatasan PHP. Karena itu, digunakanlah *method* POST dengan tambahan parameter “`_method=PUT`”.



Gambar 4.14 Endpoint Tambah atau Ubah Item di Cart

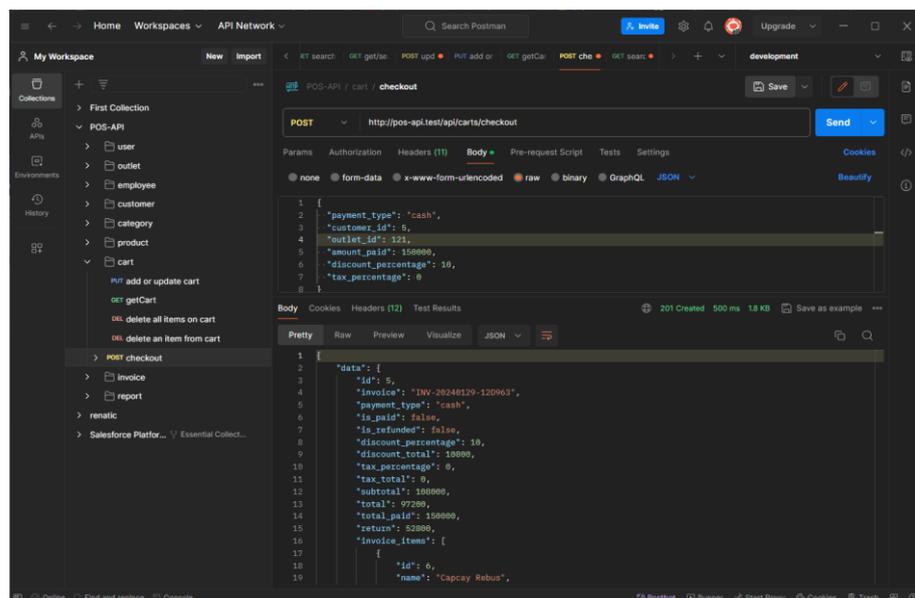
Gambar 4.14 merupakan *endpoint* untuk menambahkan atau mengubah kuantitas atau persentase diskon dalam keranjang yang terhubung ke masing-masing *employee*. Sistem akan secara otomatis

menambahkan item ke dalam keranjang apabila belum ada item tersebut dan memperbarui data kuantitas dan persen diskon apabila item tersebut sudah ada di dalam keranjang.



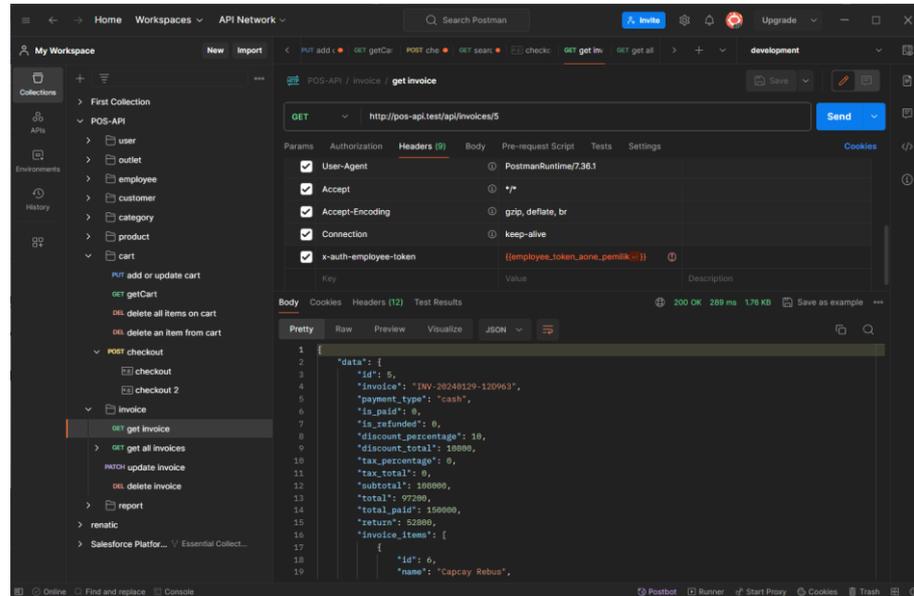
Gambar 4.15 Endpoint Get Cart

Gambar 4.15 merupakan *endpoint* untuk mendapatkan semua isi keranjang yang terhubung dengan akun *employee*. Sistem juga akan menghitung diskon dan total harga per item, serta menghitung total item dan total harga semua item.



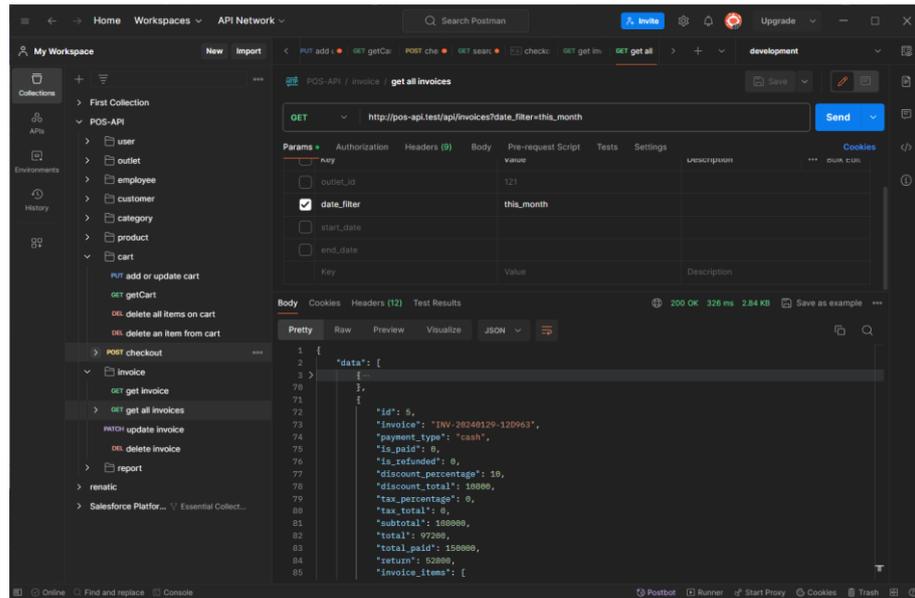
Gambar 4.16 Endpoint Cart Checkout

Gambar 4.16 merupakan *endpoint* untuk melakukan *checkout* item yang terdapat dalam keranjang *employee*. Endpoint ini juga sekaligus membuat *invoice* data invoice dan detail *invoice* berdasarkan item-item yang terdapat dalam keranjang beserta dengan perhitungan total dan uang kembalian jika ada.



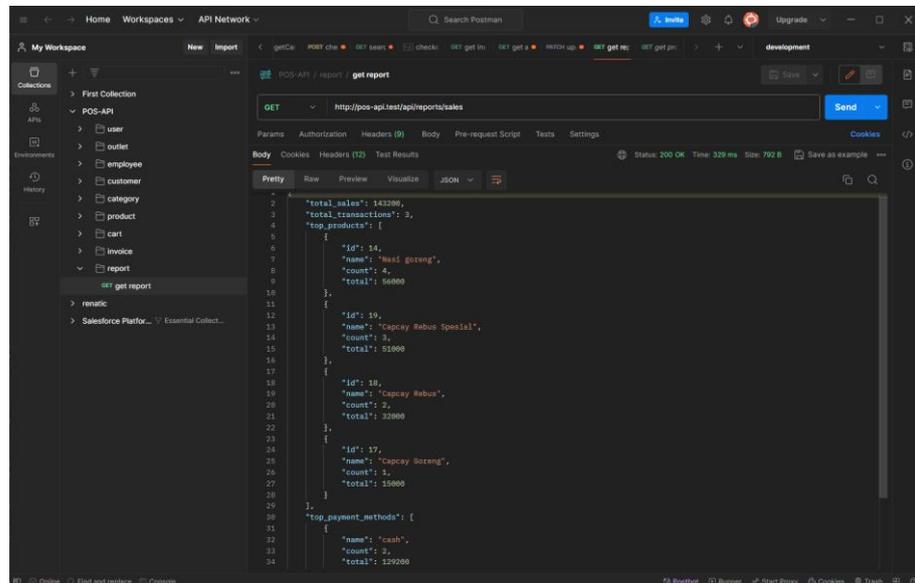
Gambar 4.17 Endpoint Get Invoice Menggunakan Id Invoice

Keranjang yang telah dilakukan *checkout* dapat juga dilihat di *endpoint* *get invoice* dengan memberikan id *invoice* yang telah diberikan di dalam *response checkout*, seperti yang terlihat pada Gambar 4.17 di atas. Untuk melihat semua *invoice* atau riwayat transaksi yang telah dilakukan, bisa dengan menggunakan *endpoint* *get all invoice*, seperti yang ditunjukkan pada Gambar 4.18 berikut.



Gambar 4.18 Endpoint Get All Invoices

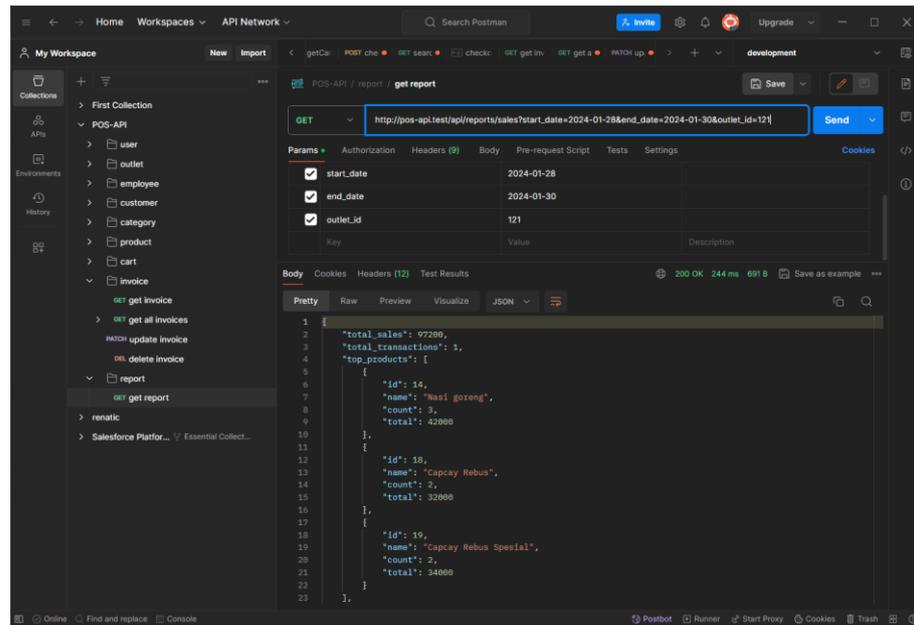
Gambar 4.18 menunjukkan *endpoint* untuk mendapatkan semua data *invoice* atau riwayat transaksi. Data ini dapat disaring berdasarkan hari ini, kemarin, minggu ini, bulan ini, ataupun rentang tanggal tertentu. Data ini juga dapat disaring berdasarkan *outlet* dengan memberikan id *outlet* pada parameter, jika tidak diberikan maka akan memberikan data pada semua *outlet* yang terhubung dengan *user*.



Gambar 4.19 Endpoint Get Report

Gambar 4.19 menunjukkan *endpoint* untuk mendapatkan laporan terkait total penjualan, banyaknya jumlah transaksi, produk terlaris, dan jenis pembayaran yang paling banyak dilakukan. Berbeda dengan *endpoint get all*

invoices yang menampilkan semua *invoice* tanpa ada ketentuan khusus selain penyaring yang digunakan sebagai parameter, laporan ini hanya menghitung *invoice* atau transaksi yang sudah dibayar dan tidak terdapat pengembalian dana atau *refund*. Namun, *endpoint* ini memiliki parameter yang sama seperti *endpoint get all invoices* yaitu parameter untuk menyaring *outlet* dan tanggal tertentu.

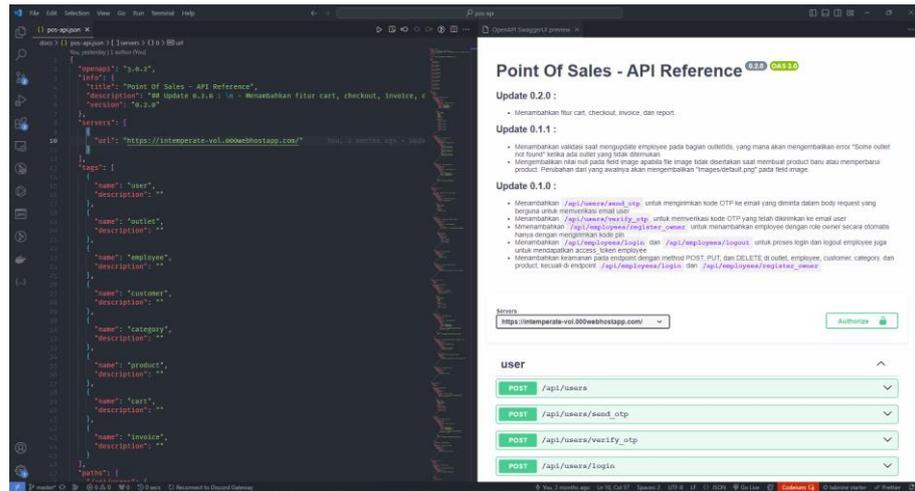


Gambar 4.20 Endpoint Get Report dengan Penyaring

Gambar 4.20 menunjukkan *endpoint get report* yang diberi penyaring untuk tanggal dan *outlet* tertentu. Terlihat bahwa *response* yang diberikan berbeda dengan yang ditunjukkan pada Gambar 4.19. Pada Gambar 4.20 hanya terdapat satu transaksi, sedangkan pada Gambar 4.19 terdapat total tiga transaksi.

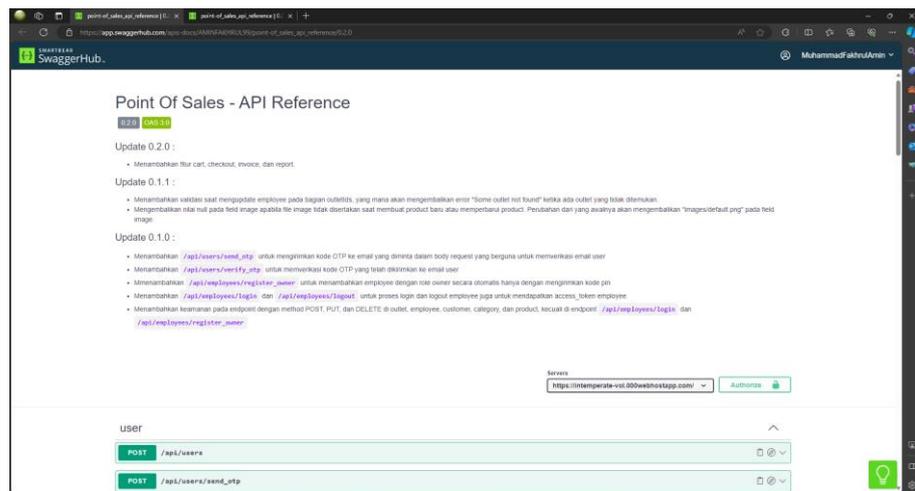
b. Dokumentasi API.

Dokumentasi API dibuat dalam bentuk JSON (*Javascript Object Notation*) menggunakan *OpenAPI Specification*. Berikut adalah tampilan dokumentasi API yang telah dibuat.



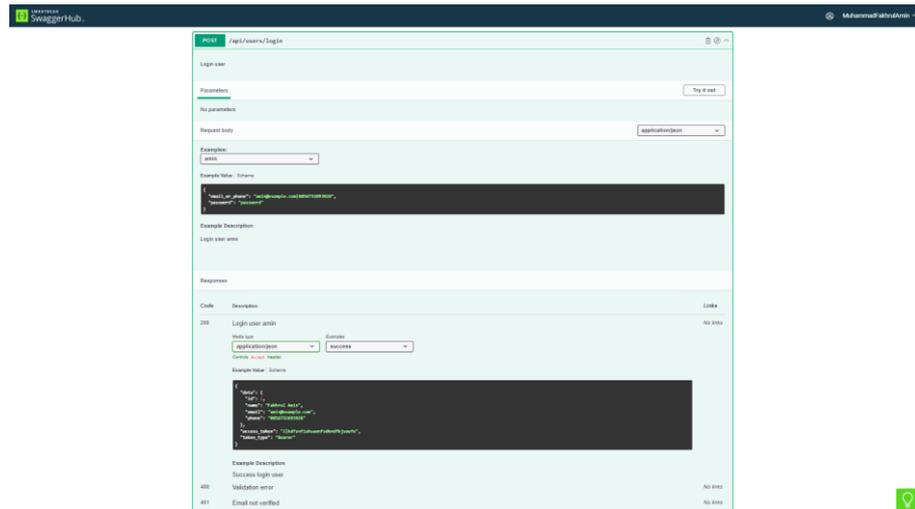
Gambar 4.21 Dokumentasi API

Gambar 4.21 menunjukkan dokumentasi API yang dibuat dalam bentuk JSON di dalam kode editor lokal. Dokumentasi ini juga bisa diakses secara publik melalui SwaggerHub, seperti yang dapat dilihat pada Gambar 4.22 berikut.



Gambar 4.22 Dokumentasi API Publik

Gambar 4.22 menunjukkan dokumentasi API yang tersedia secara publik. Penyediaan dokumentasi API secara publik ini untuk mempermudah akses terhadap dokumentasi API bagi *mobile front-end developer*, sehingga tidak perlu mengakses ataupun memahami kode *back-end* untuk mencoba menggunakan API.



Gambar 4.23 Dokumentasi API untuk Endpoint Login User

Gambar 4.23 menunjukkan dokumentasi API untuk *endpoint* login user yang menampilkan contoh *request body*, tipe data, dan beberapa *response* yang mungkin muncul.

4. Keberlanjutan

Aplikasi yang dibuat masih dalam tahap staging dan memerlukan pengembangan yang lebih lanjut, tetapi beban magang sudah selesai karena sudah melakukan kewajiban magang selama 3 bulan. Adapun pengembangan yang lebih lanjut, yaitu mengembangkan fitur riwayat transaksi agar dapat dikelompokkan menurut hari dan dapat menampilkan total transaksi setiap kelompok tersebut dan menambahkan fitur laporan yang dapat memberikan data penjualan yang terbagi perhari, perminggu, perbulan, atau pertahun yang dapat diaplikasikan dalam bentuk diagram pada sisi *mobile*.

BAB V

PENUTUP

A. Kesimpulan

Pengembangan sistem *back-end* aplikasi *mobile Point of Sales* dengan menggunakan *framework* Laravel berbasis REST API di CV. Gama Putra Santosa telah selesai dilakukan. API yang dibuat sudah berhasil digunakan melalui *software* Postman. Dokumentasi API juga sudah berhasil dibuat dan dapat diakses oleh publik melalui SwaggerHub, yaitu salah satu tempat untuk mempublikasikan OpenAPI *Specification*. Melalui praktik magang ini penulis mendapatkan pengalaman untuk bekerja sama dalam tim mengerjakan proyek di ranah industri.

B. Saran

Sistem *back-end* aplikasi POS yang telah dibangun masih memiliki kekurangan, baik dari segi keamanan maupun fitur. Untuk itu, sistem ini perlu dikembangkan lebih lanjut untuk memenuhi kebutuhan pengguna. Beberapa saran yang dapat diberikan sebagai acuan pengembangan sistem *back-end* aplikasi POS antara lain:

1. Menambahkan fitur laporan yang dapat memberikan data penjualan yang terbagi perhari, perminggu, perbulan, atau pertahun yang dapat diaplikasikan dalam bentuk diagram pada sisi *mobile*.
2. Mengembangkan fitur riwayat transaksi agar dapat dikelompokkan menurut hari dan dapat menampilkan total transaksi setiap kelompok tersebut.

LAMPIRAN

A. Form Penilaian Pembimbing Lapangan

FORM PENILAIAN PEMBIMBING LAPANGAN

Yang bertanda tangan di bawah ini :
 Nama : MUHAMMAD ARIEF SUPRIYADI
 NIK / NIP :
 Jabatan : PROJEK MANAGER

Sebagai Pembimbing Lapangan Praktik Magang :
 Nama : Muhammad Fakhru Amin
 NIM : 2000018277

Setelah mengikuti pelaksanaan Praktik Magang mahasiswa tersebut, memberikan nilai sebagai berikut :

ASPEK PENILAIAN	Paramater Penilaian					Skor
	Sangat Baik 5	Baik 4	Cukup 3	Kurang 2	Buruk 1	
1. Komunikasi dan Pemahaman Instruksi Kerja		✓				4
2. Keaktifan dan Respon di Tempat Kerja		✓				4
3. Kesesuaian Materi KP dengan Keilmuan Mahasiswa		✓				4
4. Disiplin Kerja		✓				4
5. Pengerjaan Tugas dan Ketrampilan Teknis		✓				4
Total						20

Bojonegara 25 Januari 2024

 Pembimbing Lapangan*
 MUHAMMAD ARIEF SUPRIYADI
 NIK / NIP :

* Pengisian lapangan harus ditandatangani dan dibubuhi cap

B. Log Book Praktik Magang

Sebelum UTS

**LOG BOOK PRAKTIK MAGANG MAHASISWA
 PROGRAM STUDI S1 INFORMATIKA T.A 2023 / 2024
 (WAJIB DIISI DAN MASUK DALAM PENILAIAN)**

Nim : 2000018277
 Nama Mahasiswa : Muhammad Fakhru Amin
 Judul Praktik Magang : Pengembangan Back-End Aplikasi POS
 Dosen Pembimbing : Bambang Robiin, S.T., M.T.
 Pembimbing Lapangan : Muhammad Arief Supriyadi

Petunjuk Pengisian Log Book

1. Log book di isi per minggu
2. Log book ditulis tangan
3. Setiap kegiatan di paraf oleh pembimbing lapangan/ dosen pembimbing Praktik Magang
4. Log book per minggu di paraf oleh dosen pengampu kelas Praktik Magang
5. Jumlah bimbingan minimal 4x

Logbook Minggu 4 sd 7 (sebelum UTS)

No	Kegiatan dan Lokasi Praktik Magang	Waktu Pelaksanaan		Hasil	Kendala, Rencana Perubahan (Jika ada)	Paraf Pembimbing Lapangan	Paraf Dosen Pembimbing Praktik Magang
		Hari/TGL	Jam Durasi				
1.	Pembahasan mengenai proyek dan pembagian jobdesk. Lokasi kantor GPS studio	Minggu ke-1 Rabu, 4 Oktober 2023	2,5 jam	Proyek yang akan dikerjakan merupakan suatu aplikasi POS mobile untuk kasir. Mendapatkan pembagian jobdesk.			
2.	Pembahasan rencana proyek dan mencoba aplikasi lain yang sejenis.	Rabu, 9 Oktober 2023	3 jam	Mendapatkan gambaran yang lebih jelas mengenai proyek yang akan dikerjakan.			
3.	Melakukan konsultasi terhadap rancangan aplikasi yang sudah dibuat.	Jumat, 10 Oktober 2023	2,5 jam	Memperjelas pemahaman mengenai proses bisnis aplikasi dan rancangan database yang akan dibuat.			
4.	Membuat rencana REST API yang akan digunakan,	Selasa, Rabu, dan	3 x 3 jam	Menyelesaikan rencana REST API yang berhubungan dengan tabel			

	model, dan migration tabel.	Minggu, 23, 24, dan 29 Oktober 2023		user, outlet, dan employee menggunakan OpenAPI. Kemudian membuat model dan migration tabel untuk ketiganya menggunakan Laravel.			
5.	Mengerjakan REST API yang berhubungan dengan tabel user.	Senin-Jumat, 30 Oktober - 3 November 2023	5 x 3 jam	Mengimplementasikan 5 fitur REST API, yaitu register, login, update, get current user, dan logout. Termasuk membuat unit test case.			
6.	Mengerjakan REST API yang berhubungan dengan Outlet dan Employee	Selasa - Sabtu, 7 - 11 November 2023	4 x 3 jam	Mengimplementasikan semua fitur yang ada di outlet, seperti create, get, update, delete, dan get all outlet. Membuat fitur create employee. Keduanya sudah termasuk membuat unit test case.			
7.	Mengerjakan REST API yang berhubungan dengan Employee serta mencari solusi terhadap kendala yang sedang dihadapi	Selasa, Rabu, Jumat, 14, 15, dan 17 November 2023	3 x 2,5 jam	Mengimplementasikan fitur-fitur yang berhubungan dengan employee, seperti get, update, delete, dan get all.	Terdapat kendala untuk mengimplementasikan fitur dua kali login (login user dan karyawan), terutama pada proses otentikasi dan otorisasi.		

Catatan Pembimbing Lapangan/Dosen Pembimbing Praktik Magang / Dosen Pengampu Kelas Praktik Magang:

.....

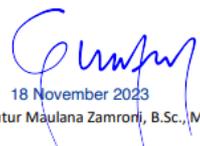
.....

.....

.....

.....

Dosen Pengampu Kelas Praktik Magang


 18 November 2023
 (Guntur Maulana Zamroni, B.Sc., M.Kom.)

Yogyakarta, 17 November 2023

Mahasiswa


 (Muhammad Fakhru Amin)

Setelah UTS

LOG BOOK PRAKTIK MAGANG MAHASISWA
PROGRAM STUDI S1 INFORMATIKA, UAD T.A 2023 / 2024
 (WAJIB DIISI DAN MASUK DALAM PENILAIAN)

Nim : 2000018277
 Nama Mahasiswa : Muhammad Fakhru Amin
 Judul Praktik Magang : Pengembangan Aplikasi POS
 Dosen Pembimbing : Bambang Robiin, S.T., M.T.
 Pembimbing Lapangan : Muhammad Arief Supriyadi

Petunjuk Pengisian Log Book

1. Log book di isi per minggu
2. Log book ditulis tangan
3. Setiap kegiatan di paraf oleh pembimbing lapangan/ dosen pembimbing Praktik Magang
4. Log book per minggu di paraf oleh dosen pengampu kelas Praktik Magang
5. Jumlah bimbingan minimal 3x

Logbook Minggu 8 sd 10 (setelah UTS)

No	Kegiatan dan Lokasi Praktik Magang	Waktu Pelaksanaan		Hasil	Kendala, Rencana Perubahan (Jika ada)	Paraf Pembimbing Lapangan	Paraf Dosen Pembimbing Praktik Magang
		Hari/TGL	Jam Durasi				
8.	Mengerjakan REST API yang berhubungan dengan Customer dan Category (Product), serta melengkapi fitur search employee.	20-26 November 2023	3 x 3 jam	Mengimplementasikan fitur CRUD Customer dan Category (Product), serta melengkapi fitur search employee by name.			
9.	Mengerjakan REST API yang berhubungan dengan Product dan membuat dokumentasi API untuk Category (Product) dan Product.	27 November - 3 Desember 2023	3 x 3 jam	Mengimplementasikan fitur CRUD Product dan melengkapinya dengan fitur search dan filter pengurutan produk. Membuat dokumentasi API untuk Category (Product) dan Product untuk digunakan di sisi mobile.			
10.	Mendeploy REST API beserta dokumentasinya	4 Desember	6 x 3 jam	Mencoba untuk mendeploy aplikasi (di 000webhost) dan dokumentasi	Proses deploy aplikasi di tempat hosting		

	agar bisa digunakan bersama dengan mobile developer secara online. Menyiapkan auth dan verifikasi OTP untuk user.	– 17 Desember 2023		API (di swagger). Mengimplementasikan personal access token untuk user karena akun user dapat diakses di beberapa device. Mengimplementasikan verifikasi akun user menggunakan kode OTP yang dikirim melalui email user.	gratis memerlukan waktu yang lama dan juga file yang dapat disimpan terbatas, sehingga perlu cara lain untuk menjalankan aplikasi agar lebih efisien ketika ada perubahan atau penambahan endpoint.		
11.	Mengerjakan fitur login bagi employee dan memperbarui dokumentasi API.	18 Desember – 31 Desember 2023	6 x 3 jam	Berhasil mengatasi kendala fitur double login untuk user dan karyawan, serta proses otentikasi untuk keduanya dengan menggunakan dua header berbeda untuk mengidentifikasi akun user dan karyawan. Memperbarui dokumentasi API dengan menambahkan deskripsi mengenai pembaruan dan terkait security dan auth.			
12.	Mengerjakan fitur keranjang dan invoice, serta dokumentasi API-nya.	1 Januari – 14 Januari 2024	6 x 3 jam	Mengimplementasikan fitur keranjang dengan banyak barang berbeda, beserta dengan fitur checkout yang secara otomatis memasukkan semua barang ke dalam invoice atau riwayat. Berhasil mengatasi kendala pemakaian REST API di mobile, yaitu dengan menjalankannya di lokal bersama dengan aplikasi mobile menggunakan emulator atau device yang tersambung ke Wi-Fi yang sama.			

13.	Mengerjakan fitur report, dokumentasi API-nya dan laporan praktik magang.	15 Januari - 27 Januari 2024	3 x 3 jam	Mengimplementasikan fitur report atau laporan dalam API serta dokumentasi API-nya. Mengerjakan laporan praktik magang Bab 1 sampai Bab 5.			
-----	---	------------------------------	-----------	---	--	---	---

Catatan Pembimbing Lapangan/Dosen Pembimbing Praktik Magang / Dosen Pengampu Kelas Praktik Magang:

.....

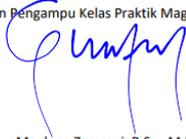
.....

.....

.....

.....

Dosen Pengampu Kelas Praktik Magang



(Guntur Maulana Zamroni, B.Sc., M.Kom.)

Yogyakarta, 4 Februari 2024

Mahasiswa



(Muhammad Fakhru Amin)

C. Lembar Persetujuan Instansi

HALAMAN PERSETUJUAN INSTANSI

Laporan Praktik Magang ini telah disetujui sebagai proses pengajuan seminar bagi

Nama : Muhammad Fakhru Amin
 NIM : 2000018277
 Tempat Praktik Magang : CV. Gama Putra Santosa
 Realisasi Waktu Pelaksanaan : 3 bulan

Yogyakarta, 4 Februari 2024

Menyetujui,
Ketua/Kepala/Direktur



Muhammad Arief Supriyadi

D. Dokumentasi

Berikut dokumentasi saat observasi magang.



Berikut dokumentasi saat pengerjaan praktik magang.

