

PROYEK AKHIR

PERANCANGAN DAN PENGEMBANGAN SISTEM INFORMASI POINT
OF SALE – myPOS.



NIM : 200120099

NAMA LENGKAP : BAYU GURIUM

PROGRAM STUDI SISTEM INFORMASI

INSTITUT TEKNOLOGI DAN BISNIS STIKOM AMBON

2024.

SURAT PERNYATAAN



INSTITUT TEKNOLOGI DAN BISNIS STIKOM AMBON

Prod.: Teknik Informatika, Sistem Informasi, Pariwisata, Bisnis Digital, Desain Komunikasi Visual, Perencanaan Wilayah dan Kota
Manajemen Informatika, Komputerisasi Akuntansi

Jl. A. Y. Patty No. 108, Ambon, Maluku. Telp. (0911) 315735, 342332 Fax. (0911) 316117,
Web : www.stikomambon.ac.id, Email : info@stikomambon.ac.id, akademik.stikomambon@gmail.com

SURAT PERNYATAAN

NAMA (SESUAI IJAZAH SMA)

BAYU GURIUM

TEMPAT DAN TANGGAL LAHIR (SESUAI IJAZAH SMA)

KULUGOWA, 14 MEI 2001

JENIS KELAMIN

LAKI-LAKI

PROGRAM STUDI

SISTEM INFORMASI

JUDUL PROJECT AKHIR atau TUGAS AKHIR (tidak boleh memakai singkatan):

PERANCANGAN DAN PENGEMBANGAN SISTEM INFORMASI POINT OF SALE

NO HP : 085142959238

Biodata diisi dengan sebenar-benarnya, jika terjadi kesalahan saya akan bertanggung jawab

Ambon,

TTD Mahasiwa



BAYU GURIUM

DAFTAR ISI

HALAMAN JUDUL.....	i
SURAT PERNYATAAN.....	ii
DAFTAR ISI.....	iii
DAFTAR TABEL.....	iv
DAFTAR GAMBAR.....	v
LATAR BELAKANG.....	vi
TUJUAN DAN MANFAAT PENGEMBANGAN SISTEM.....	1
a. Tujuan.....	1
b. Manfaat.....	2
PERANCANGAN.....	3
a. Kebutuhan Fungsionalitas Pengguna.....	3
DIAGRAM DAN DESAIN DATABASE.....	6
A. Diagram.....	6
b. Use Case Diagram.....	6
c. Activity Diagram.....	8
B. Desain Database.....	16
C. Entity-Relationship Diagram.....	21
DESAIN.....	23
a. Wireframe.....	23
IMPLEMENTASI.....	25
A. Metodologi Pengembangan.....	25

1. Metode Agile.....	25
B. Pengembangan Sistem.....	26
a. Teknologi yang digunakan.....	26
1. Perangkat Keras.....	26
2. Perangkat Lunak.....	27
b. Proses Coding dan version Control.....	28
TANGKAPAN LAYAR PROYEK.....	29
ALUR PROSES.....	31
PENGUJIAN.....	34
a. Jenis Pengujian.....	34
KESIMPULAN.....	36

DAFTAR TABEL

Tabel 1 : Detail Tabel Produk.....	17
Tabel 2 : Detail Tabel Penjualan.....	18
Tabel 3 : Tabel Detail Penjualan.....	20

DAFTAR GAMBAR

Gambar 1 : Use Case Diagram Point of Sale.....	7
Gambar 2 : Activity Diagram Login.....	9
Gambar 3 : Activity Diagram Proses Taransaksi Penjualan.....	10
Gambar 4 : Activity Diagram CRUD data Stok Masuk.....	11
Gambar 5 : Activity Diagram CRUD data Stok Keluar.....	12
Gambar 6 : Activity Diagram CRUD data Produk.....	13
Gambar 7 : Activity Diagram CRUD data Pengguna/User.....	14
Gambar 8 : Activity Diagram mengubah Paswword Pengguna/User.....	15
Gambar 9 : Activity Diagram mengubah Foto Profile Pengguna/User.....	16
Gambar 10 : Entity-Relationship-Diagram (ERD).....	22
Gambar 11 : Wireframe Halaman Dashboard Aplikasi POS.....	23
Gambar 12 : Wireframe Halaman Transaksi Penjualan.....	24
Gambar 13 : Wireframe Halaman Laporan Penjualan.....	24
Gambar 14 : Tampilan Halaman Dashboard.....	29
Gambar 15 : Tampilan Halaman Transaksi Penjualan.....	30
Gambar 16 : Tampilan Halaman Laporan Penjualan.....	31
Gambar 17 : Tampilan Halaman Data Produk.....	33

LATAR BELAKANG

Seiring dengan perkembangan teknologi informasi dan digitalisasi bisnis, penggunaan Point of Sale (POS) telah menjadi tidak terpisahkan dari operasional harian berbagai jenis bisnis. POS adalah sistem yang digunakan untuk melakukan transaksi penjualan secara elektronik, termasuk pengelolaan stok, pemrosesan pembayaran, dan pelaporan keuangan. Di era digital saat ini, POS telah berevolusi dari sistem kasir sederhana menjadi solusi yang canggih dan terintegrasi. Kemajuan Teknologi dalam era digital dan teknologi informasi telah berkembang pesat dan menjadi bagian integral dari berbagai aspek kehidupan, Kebutuhan akan Otomatisasi di Banyak bisnis, terutama di sektor ritel dan jasa, membutuhkan sistem yang dapat mengotomatisasi proses transaksi dan manajemen inventaris. Peningkatan Volume Transaksi Terutama di toko-toko dengan banyak cabang, diperlukan sebuah sistem yang dapat mencatat dan mengelola transaksi dengan cepat dan akurat. Aplikasi POS dapat membantu mengurangi waktu yang diperlukan untuk proses transaksi, dan setiap transaksi tercatat secara otomatis dan rinci, sehingga memudahkan pelacakan dan audit. Hal ini meningkatkan akuntabilitas dan transparansi dalam bisnis.

- **Pengelolaan Stok yang Lebih Baik**

Aplikasi POS dapat membantu dalam pengelolaan inventaris secara real-time, sehingga pemilik bisnis dapat mengetahui status stok dengan akurat dan mengurangi risiko kehabisan barang atau overstocking

- **Data dan Analitik**

Aplikasi POS memungkinkan pengumpulan data transaksi yang dapat dianalisis untuk mendapatkan wawasan bisnis yang berharga. Data ini dapat digunakan untuk membuat keputusan bisnis yang lebih baik.

- **Skalabilitas**

Aplikasi POS yang baik dapat dengan mudah diadaptasi dan dikembangkan sesuai dengan pertumbuhan bisnis, baik dari segi fitur maupun jumlah pengguna.

TUJUAN DAN MANFAAT PENGEMBANGAN SISTEM

a. Tujuan

Tujuan utama dari pengembangan POS adalah untuk meningkatkan efisiensi operasional dan pengalaman pelanggan di berbagai sektor bisnis. Dengan menerapkan POS, bisnis dapat mengotomatiskan proses transaksi, mengurangi kesalahan manusia dalam pencatatan dan inventaris, serta meningkatkan kecepatan layanan. Selain itu, POS juga membantu dalam analisis data penjualan dan pengambilan keputusan yang lebih tepat waktu dan akurat.

- Otomatisasi Proses Transaksi: Mengurangi kesalahan manusia dan mempercepat penjualan.
- Pengelolaan Inventaris Real-Time: Memastikan ketersediaan stok yang akurat.
- Meningkatkan Transparansi dan Akuntabilitas: Mencatat setiap transaksi dengan rinci.
- Meningkatkan Pengalaman Pelanggan: Mengurangi waktu tunggu pelanggan.
- Menghasilkan Data dan Analitik yang Terpusat: Mendapatkan wawasan bisnis yang berharga.

b. Manfaat

Penggunaan POS memiliki berbagai manfaat yang signifikan bagi bisnis, antara lain:

1. Manajemen Inventaris: POS memungkinkan pemantauan real-time terhadap stok barang, membantu bisnis untuk mengelola persediaan dengan lebih efisien dan menghindari kekurangan atau kelebihan stok.
2. Peningkatan Efisiensi: Dengan otomatisasi proses transaksi, POS mengurangi waktu yang dibutuhkan untuk setiap transaksi, memungkinkan karyawan untuk fokus pada pelayanan pelanggan atau kegiatan yang lebih strategis.
3. Analisis Penjualan: Data yang terkumpul melalui POS dapat digunakan untuk menganalisis tren penjualan, preferensi pelanggan, dan performa produk. Hal ini memungkinkan bisnis untuk mengidentifikasi peluang pasar baru dan merancang strategi pemasaran yang lebih efektif.
4. Peningkatan Pengalaman Pelanggan: Dengan proses pembayaran yang cepat dan akurat, POS membantu meningkatkan pengalaman pelanggan dengan memberikan layanan yang lebih efisien dan profesional.
5. Integrasi dengan Sistem Lain: POS dapat diintegrasikan dengan sistem manajemen inventaris, sistem akuntansi, dan platform e-commerce, memastikan alur kerja yang terhubung dan terkoordinasi dengan baik di seluruh perusahaan.

PERANCANGAN

a. Kebutuhan Fungsionalitas Pengguna

Aplikasi Manajemen Point of Sale (POS) ini dirancang untuk membantu pemilik usaha dalam mengelola bisnis mereka dengan lebih efektif dan efisien. Aplikasi ini dilengkapi dengan berbagai fitur yang memungkinkan pengguna untuk

mengelola data produk, transaksi penjualan, serta persediaan barang dengan mudah.

Berikut beberapa fitur utama yang terdapat didalam aplikasi Point of Sale :

a. Pengelolaan Data Produk (CRUD)

1. **Create** : Menambahkan data produk baru ke dalam sistem dengan informasi seperti nama produk, harga, kategori, dan jumlah stok.
2. **Read**: Melihat dan menelusuri daftar produk yang tersedia di dalam inventaris.
3. **Update**: Memperbarui informasi produk yang sudah ada, termasuk perubahan harga, deskripsi, atau jumlah stok.
4. **Delete**: Menghapus produk yang sudah tidak tersedia atau tidak lagi dijual.

b. Pengelolaan Data Satuan (CRUD)

1. **Create**: Menambahkan satuan unit baru untuk produk, seperti kilogram, liter, atau unit.
2. **Read**: Melihat daftar satuan yang sudah ada untuk memudahkan pengelompokan produk.
3. **Update**: Mengubah informasi satuan jika ada perubahan.
4. **Delete**: Menghapus satuan yang tidak digunakan lagi.

c. Pengelolaan Data Kategori (CRUD)

1. **Create**: Menambahkan kategori baru untuk mengelompokkan produk, seperti makanan, minuman, atau elektronik.

2. **Read:** Melihat daftar kategori yang tersedia untuk mengelola produk dengan lebih terstruktur.
3. **Update:** Memperbarui informasi kategori jika ada perubahan.
4. **Delete:** Menghapus kategori yang tidak lagi relevan atau digunakan.

d. Pengelolaan Data Supplier (CRUD)

5. **Create:** Menambahkan supplier baru dengan informasi lengkap seperti nama, kontak, dan alamat.
6. **Read:** Melihat daftar supplier yang sudah ada dan memudahkan pencarian informasi supplier.
7. **Update:** Memperbarui informasi supplier, seperti perubahan alamat atau nomor telepon.
8. **Delete:** Menghapus supplier yang sudah tidak bekerja sama lagi.

e. Pengelolaan Data Customer (CRUD)

1. **Create:** Menambahkan customer baru dengan informasi lengkap seperti nama, kontak, dan alamat.
2. **Read:** Melihat daftar customer yang sudah ada untuk memudahkan pencarian dan manajemen data customer.
3. **Update:** Memperbarui informasi customer jika ada perubahan data.
4. **Delete:** Menghapus customer yang sudah tidak bertransaksi lagi.

f. Transaksi Penjualan

1. Memungkinkan pengguna untuk melakukan transaksi penjualan produk kepada pelanggan dengan mudah dan cepat.

2. Pengguna dapat memilih produk, menentukan jumlah yang dijual, dan menghasilkan struk penjualan.
3. Mencatat setiap transaksi penjualan untuk keperluan laporan dan analisis.

g. Pengelolaan Data Stok Masuk (CRUD)

1. **Create:** Menambahkan data stok masuk untuk mencatat penerimaan barang baru dari supplier.
2. **Read:** Melihat daftar stok masuk dengan rincian seperti tanggal, produk, dan jumlah.
3. **Update:** Mengubah informasi stok masuk jika ada kesalahan atau perubahan.
4. **Delete:** Menghapus data stok masuk yang tidak relevan atau sudah tidak diperlukan.

h. Pengelolaan Data Stok Keluar (CRUD)

1. **Create:** Menambahkan data stok keluar untuk mencatat barang yang keluar dari stok, baik untuk penjualan maupun alasan lainnya.
2. **Read:** Melihat daftar stok keluar dengan rincian seperti tanggal, produk, dan jumlah.
3. **Update:** Mengubah informasi stok keluar jika ada kesalahan atau perubahan.
4. **Delete:** Menghapus data stok keluar yang tidak relevan atau sudah tidak diperlukan

i. Laporan Penjualan

1. Menghasilkan laporan dari hasil penjualan untuk membantu analisis bisnis.

2. Laporan dapat mencakup informasi seperti total penjualan, produk terlaris, dan analisis penjualan berdasarkan periode waktu tertentu.
3. Memudahkan pemilik usaha untuk membuat keputusan berdasarkan data yang akurat dan up-to-date.

DIAGRAM DAN DESAIN DATABASE

A. Diagram

b. Use Case Diagram

Diagram Use Case adalah salah satu jenis diagram yang digunakan dalam pemodelan perangkat lunak untuk menggambarkan bagaimana pengguna (aktor) berinteraksi dengan sistem dan bagaimana sistem memenuhi kebutuhan pengguna tersebut.

Elemen-elemen di dalam Use Case Diagram :

1. Aktor

Entitas yang berinteraksi dengan sistem. Aktor bisa berupa pengguna manusia, sistem lain, atau perangkat keras.

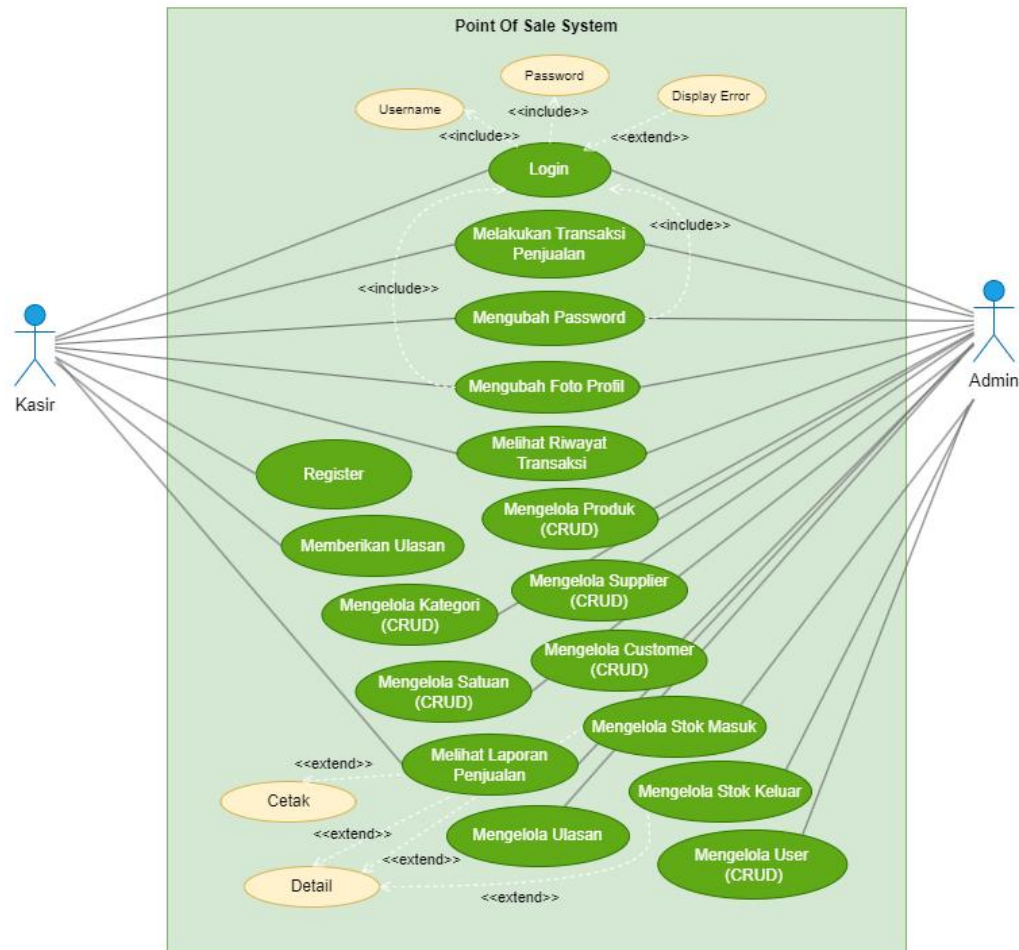
2. Use Case

Fungsi atau layanan yang disediakan oleh sistem kepada aktor. Use Case menggambarkan tujuan atau hasil yang dicapai oleh aktor melalui interaksi dengan sistem.

3. Relasi

1. Associations: Garis yang menghubungkan aktor dengan use case, menunjukkan bahwa aktor berinteraksi dengan use case tertentu.
2. Include: Hubungan antara use case di mana satu use case selalu melibatkan use case lainnya.

3. Extend: Hubungan di mana satu use case opsional dapat memperluas fungsionalitas use case lainnya.



Gambar 1 : Use Case Diagram Point of Sale

c. Activity Diagram

Activity Diagram adalah salah satu jenis diagram dalam UML (Unified Modeling Language) yang digunakan untuk menggambarkan alur kerja atau proses dalam sistem. Diagram ini fokus pada aktivitas dan aliran kerja di dalam sistem dan menunjukkan langkah-langkah yang dilakukan untuk mencapai tujuan tertentu. Tujuan utama activity diagram adalah untuk memodelkan proses, alur kerja, atau aktivitas sistem secara rinci. Ini dapat mencakup alur proses dalam aplikasi, interaksi antar sistem, atau langkah-langkah dalam suatu prosedur bisnis.

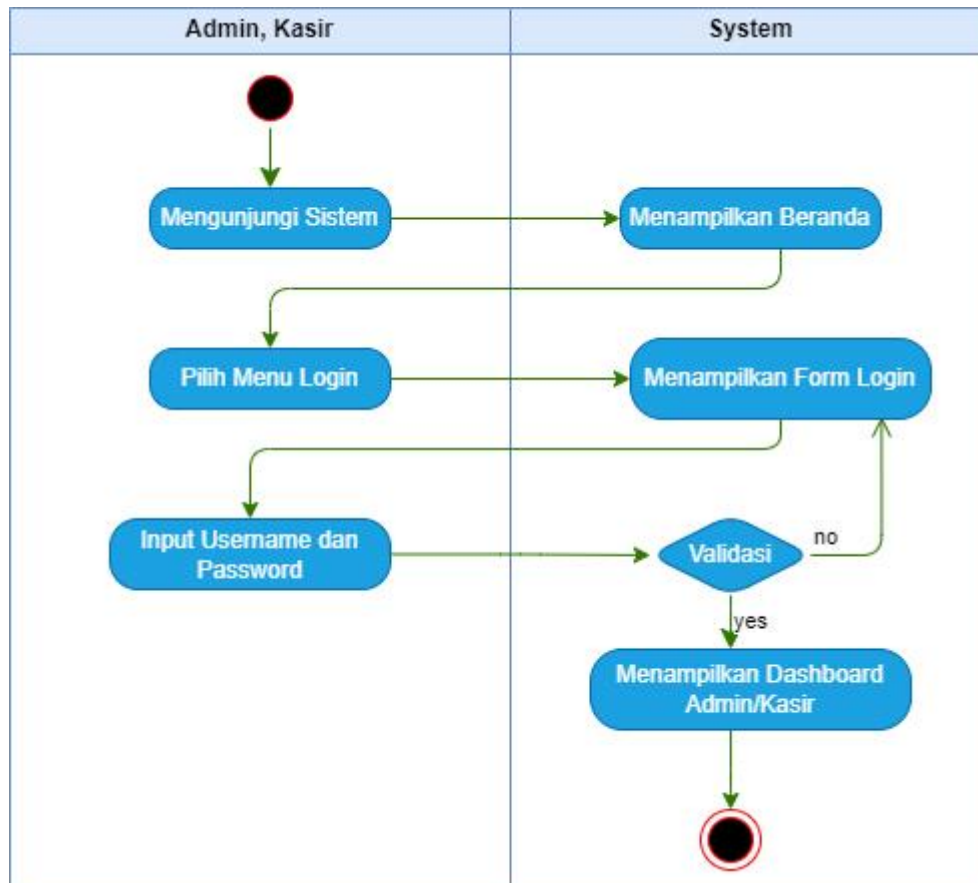
Berikut beberapa elemen penting yang ada di dalam activity diagram :

1. Aktivitas: Representasi dari tindakan atau pekerjaan yang dilakukan. Biasanya digambarkan dengan kotak atau oval.
2. Transisi: Panah yang menunjukkan alur atau urutan aktivitas. Menunjukkan bagaimana proses berpindah dari satu aktivitas ke aktivitas berikutnya.
3. Keputusan: Titik di mana alur proses bercabang berdasarkan kondisi atau keputusan. Digambarkan dengan bentuk berlian.
4. Awal dan Akhir: Titik awal dan akhir dari proses. Titik awal biasanya digambarkan dengan lingkaran hitam penuh, sedangkan titik akhir dengan lingkaran hitam dengan lingkaran putih di dalamnya.

Activity Diagram digunakan untuk membantu dalam memahami dan menganalisis alur proses atau prosedur dan mendokumentasikan cara kerja sistem atau prosedur bisnis serta membantu dalam merancang dan merencanakan proses atau alur kerja sistem yang efisien.

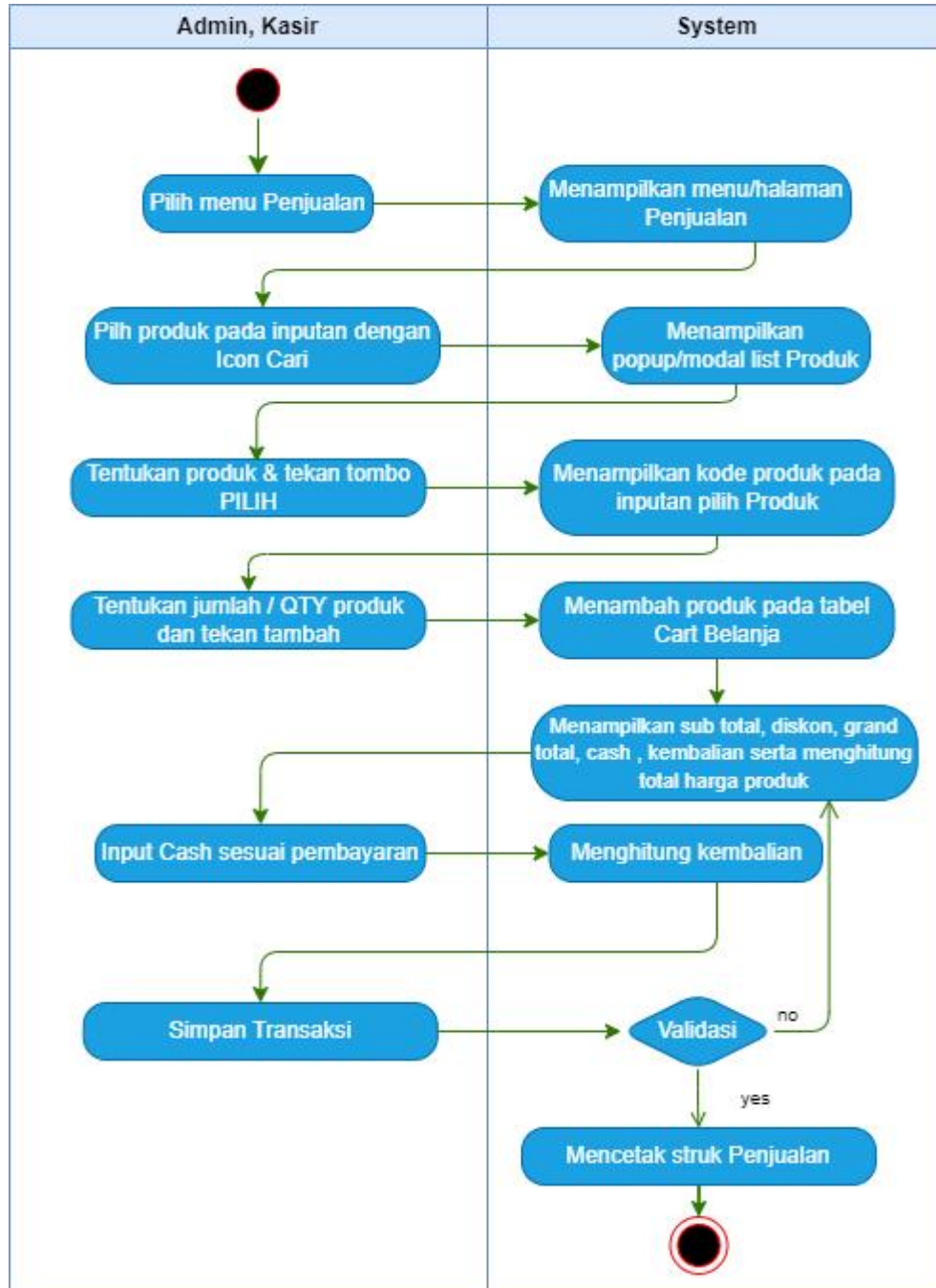
Berikut adalah Activity Diagram untuk Sistem Informasi Point of Sale (POS) :

- a. Activity Diagram proses login Admin/Kasir



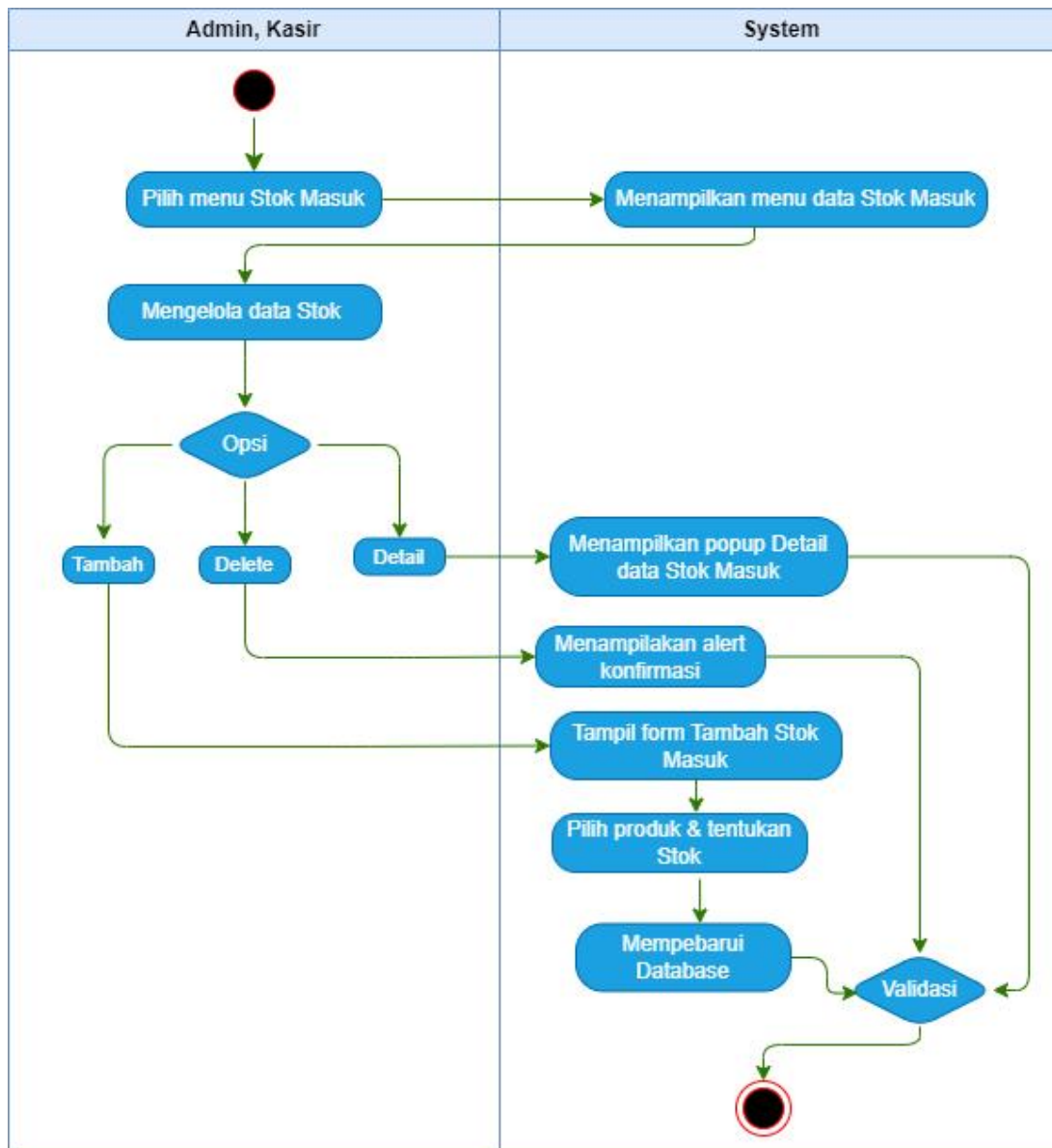
Gambar 2 : Activity Diagram Login

- b. Activity Diagram melakukan proses transaksi penjualan produk Admin / Kasir



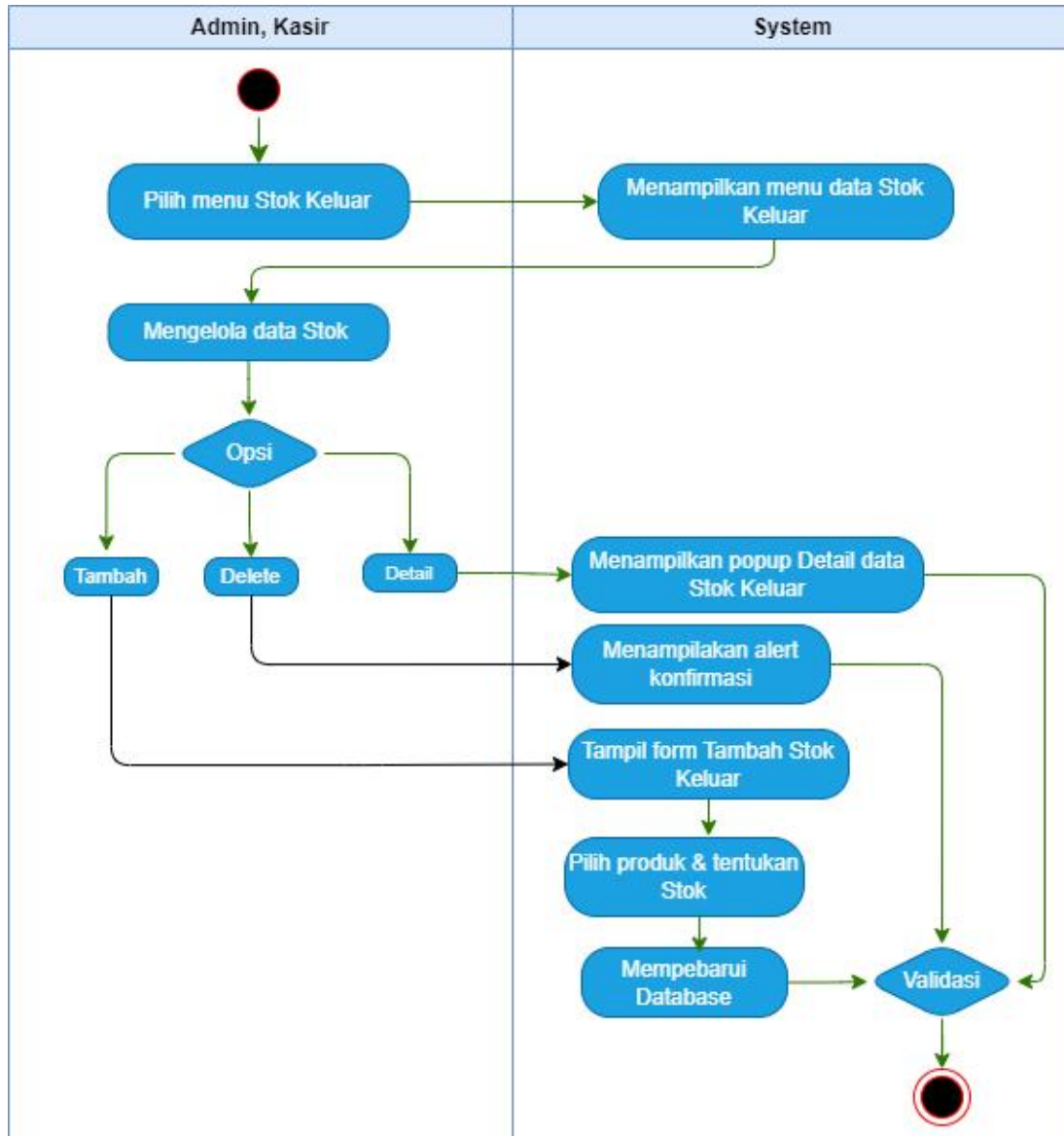
Gambar 3 : Activity Diagram poses Transaksi Penjualan

- c. Activity Diagram mengelola data stok masuk - CRUD (*Create, Read, Update dan Delete*) oleh Admin/Kasi



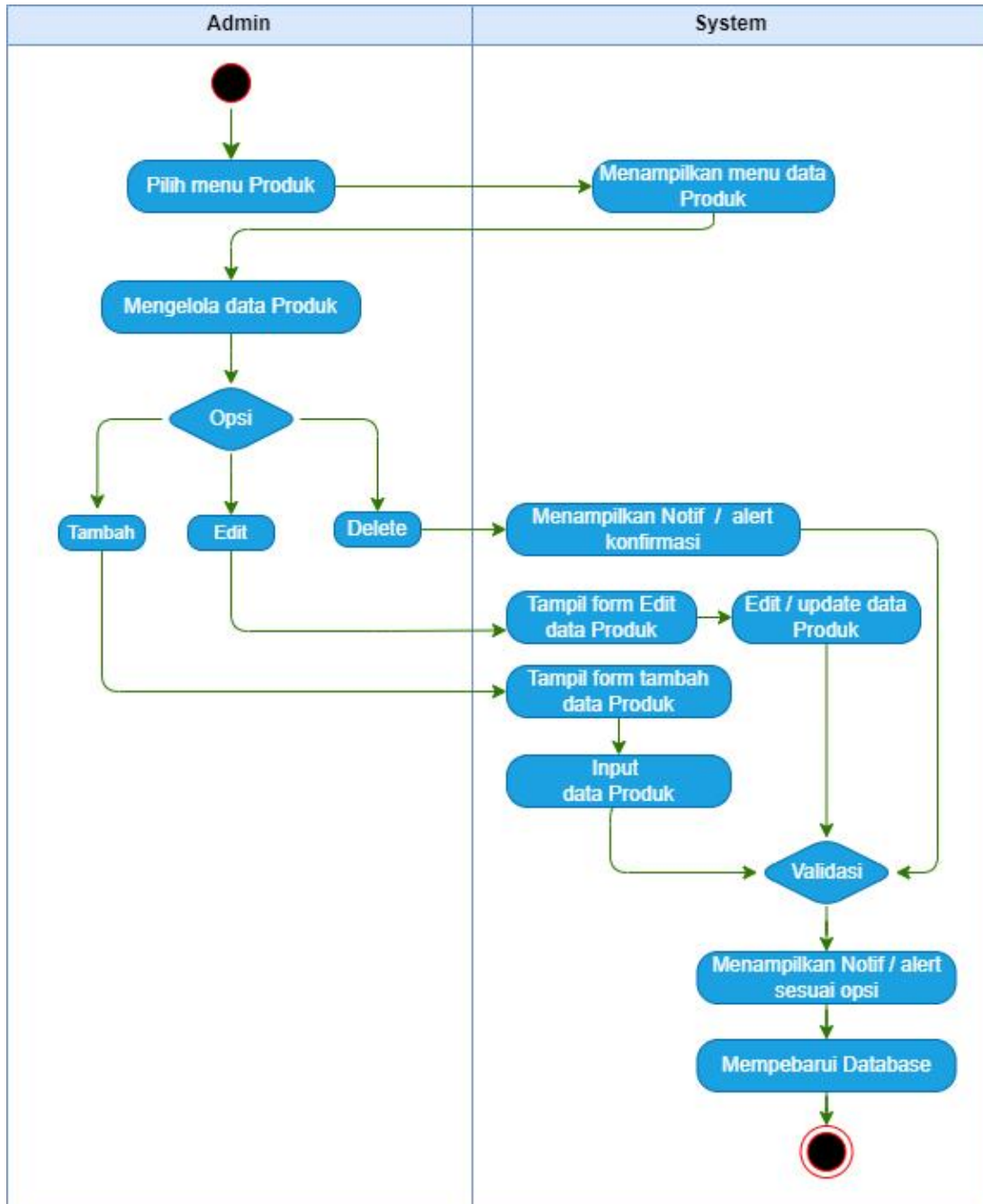
Gambar 4 : Activity Diagram CRUD data Stok Masuk

- d. Activity Diagram mengelola data stok keluar - CRUD (*Create, Read, Update dan Delete*) oleh Admin/Kasir



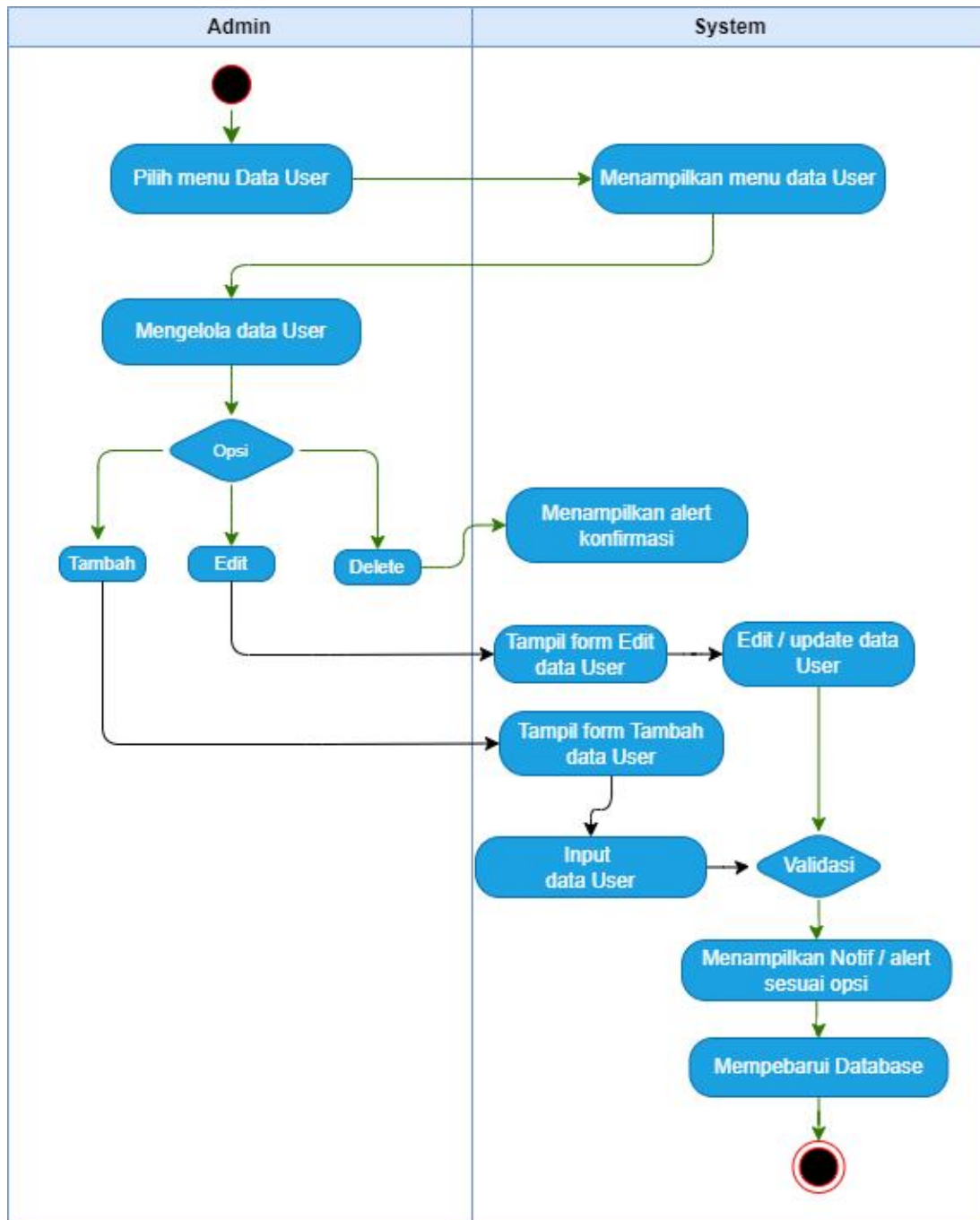
Gambar 5 : Activity Diagram CRUD data Stok Keluar

- e. Activity Diagram mengelola data produk - CRUD (*Create, Read, Update dan Delete*) oleh Admin



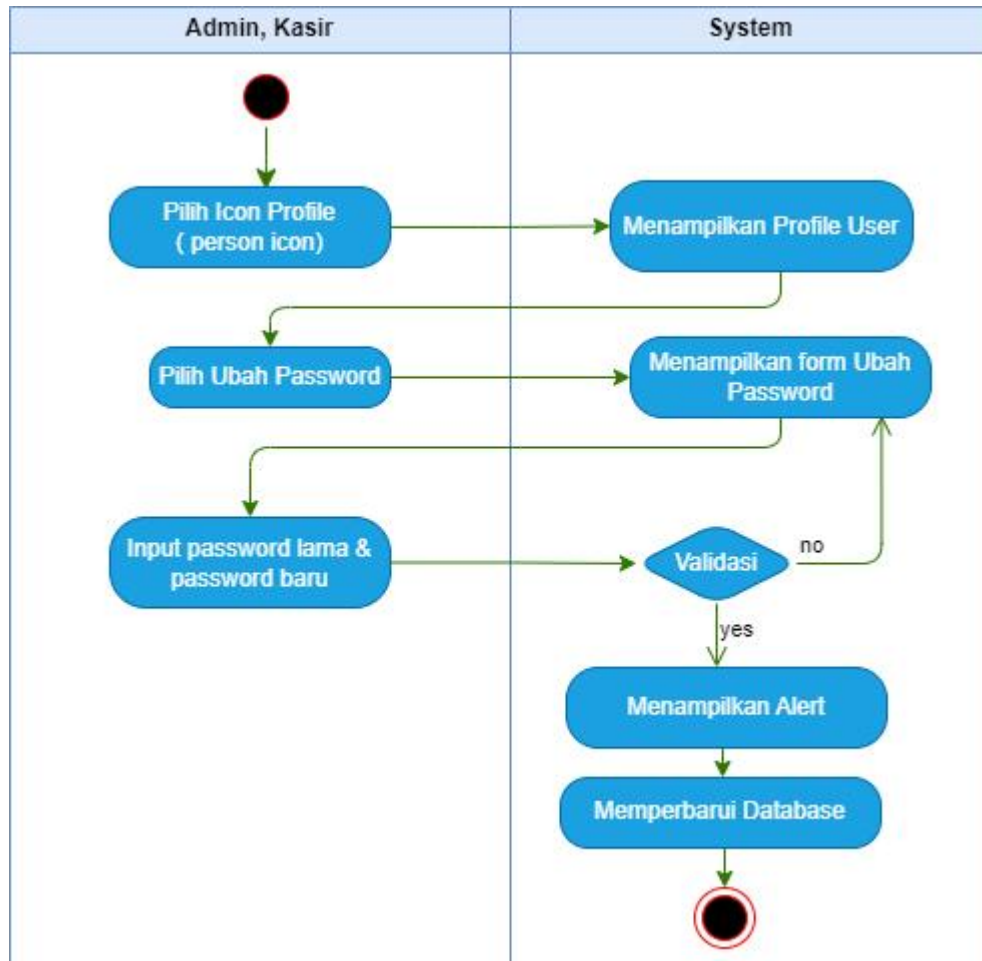
Gambar 6 : Activity Diagram CRUD data Produk

- f. Activity Diagram mengelola data Pengguna/User - CRUD (*Create, Read, Update dan Delete*) oleh admin



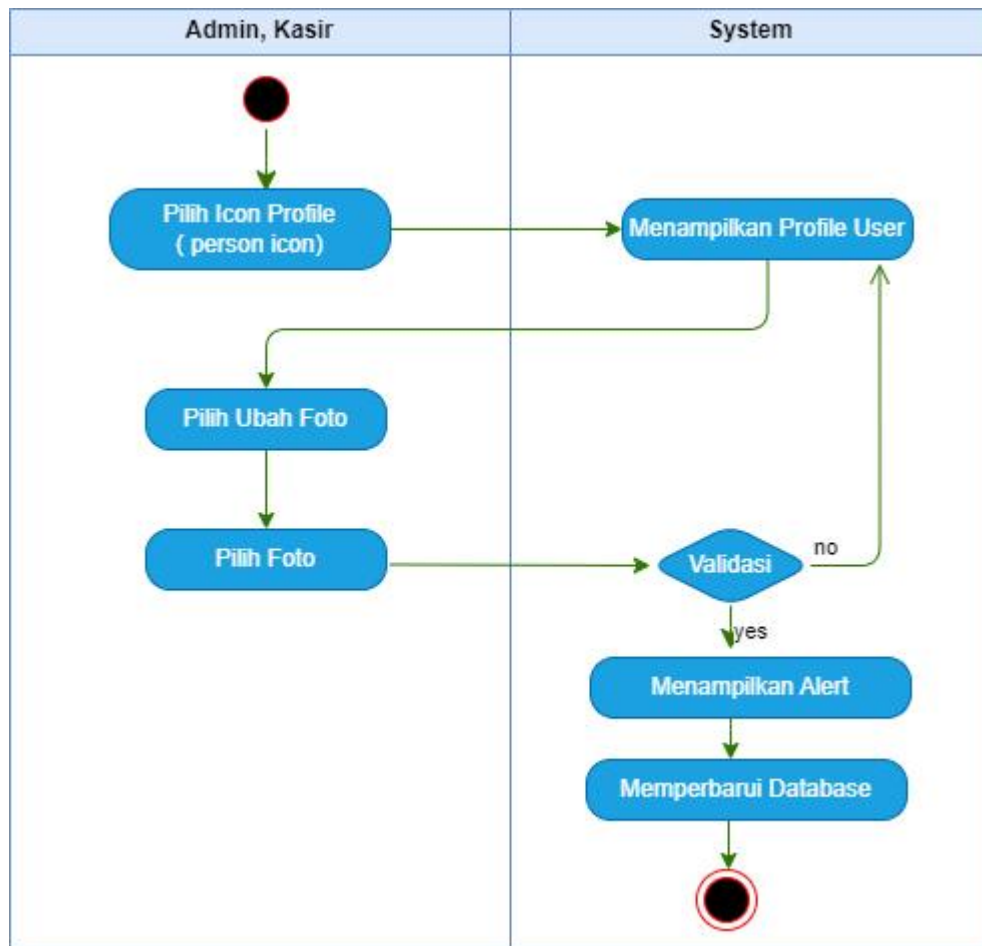
Gambar 7 : Activity Diagram CRUD data User

- g. Sctivity Diagram mengelola profile user (Ubah Password) oleh Admin/Kasir



Gambar 8 :Activity Diagram Ubah Password (Admin/Kasir)

- h. Activity Diagram mengelola profile user (Ubah Foto profil User)
oleh Admin/Kasir



Gambar 9 : Activity Diagram Ubah Foto Profile (Admin/Kasir)

B. Desain Database

1. Tabel Produk (`produk`)

Tabel ini menyimpan informasi tentang produk yang tersedia di sistem. Ini mencakup nama produk, harga, kategori, dan satuan. Tabel ini adalah komponen utama dalam manajemen inventaris dan penjualan.

Table 1 : Detail Tabel Produk

Kolom	Tipe Data	Keterangan
id_produk	INT	Primary key, auto-increment
kode_produk	VARCAHAR	Kode unik untuk mengidentifikasi produk
nama_produk	VARCAHAR	Nama dari produk
id_kategori	INT	Kunci asing ke tabel Kategori
id_satuan	INT	Kunci asing ke table Satuan
harga	VARCHAR	Harga satuan produk
stok	VARCHAR	Stok produk saat ini
created_at	TIMESTAMP	Tanggal dan waktu saat dibuat
updated_at	TIMESTAMP	Tanggal dan waktu terakhir diperbarui

Deskripsi Kolom:

- **id_produk:** Identifikasi unik untuk setiap produk dalam sistem. Digunakan sebagai kunci primer.
- **kode_produk:** Kode unik yang sering digunakan untuk referensi cepat atau identifikasi produk.
- **nama_produk:** Nama lengkap produk yang dijual.
- **id_kategori:** Menghubungkan produk dengan kategori di tabel Kategori, memberikan konteks tentang jenis produk.

- **id_satuan:** Menghubungkan produk dengan satuan pengukuran di tabel Satuan, menentukan unit yang digunakan.
- **harga:** Harga per unit produk yang ditetapkan untuk penjualan.
- **stok:** Menunjukkan jumlah stok produk yang tersedia di inventaris.
- **created_at:** Waktu dan tanggal saat produk pertama kali ditambahkan ke sistem.
- **updated_at:** Waktu dan tanggal saat data produk terakhir kali diperbarui.

2. Tabel Penjualan (`penjualan`)

Tabel penjualan mencatat semua transaksi penjualan yang terjadi dalam sistem. Tabel ini penting untuk melacak penjualan dan mengelola informasi terkait transaksi dengan menyimpan data-data pada table dibawah.

Table 2 : Detail Tabel Penjualan

Kolom	Tipe Data	Keterangan
id_penjualan	INT	Primary key, auto-increment
invoice	VARCAHAR	Nomor invoice untuk transaksi
id_customer	INT	Kunci asing ke table customer
total_harga	INT	Total harga dari transaksi
diskon	INT	Diskon dari transaksi
harga_bayar	INT	Harga yang di bayar oleh pelanggan
cash	INT	Jumlah uang tunai yang diterima
kembalian	INT	Jumlah uang kembalian

nota	TEXT	Catatan transaksi
tanggal	DATE	Tanggal transaksi
id_user	INT	Kunci asing ke table user (Kasir)
created_at	TIMESTAMP	Tanggal dan waktu saat dibuat
updated_at	TIMESTAMP	Tanggal dan waktu terakhir diperbarui

Deskripsi Kolom:

- **id_penjualan:** Identifikasi unik untuk setiap transaksi penjualan. Digunakan sebagai kunci primer.
- **invoice:** Nomor invoice yang digunakan untuk referensi transaksi.
- **id_customer:** Menghubungkan transaksi dengan pelanggan yang melakukan pembelian di tabel Customer.
- **total_harga:** Total harga dari semua produk yang dibeli dalam transaksi.
- **diskon:** Jumlah diskon yang diterapkan pada total harga.
- **harga_bayar:** Jumlah yang dibayar oleh pelanggan setelah diskon diterapkan.
- **cash:** Jumlah uang tunai yang diterima dari pelanggan saat pembayaran.
- **kembalian:** Jumlah uang kembalian yang harus diberikan kepada pelanggan jika pembayaran tunai melebihi harga bayar.
- **nota:** Catatan tambahan yang mungkin diperlukan untuk transaksi.
- **tanggal:** Tanggal transaksi penjualan dilakukan.
- **id_user:** Menghubungkan transaksi dengan kasir yang melakukan penjualan di tabel User.

- **created_at:** Waktu dan tanggal saat data transaksi dibuat.
- **updated_at:** Waktu dan tanggal saat data transaksi terakhir kali diperbarui.

3. Tabel Detail Penjualan (`detail_penjualan`)

Tabel `detail_penjualan` menyimpan rincian tentang setiap item dalam transaksi penjualan. Data yang dicatat meliputi produk yang terlibat, harga per unit, jumlah yang terjual, diskon yang diterapkan, dan total harga untuk item tersebut. Tabel ini memberikan detail penting yang melengkapi informasi transaksi penjualan di tabel penjualan.

Table 3: Tabel Detail Penjualan

Kolom	Tipe Data	Keterangan
<code>id_detail</code>	INT	Primary key, auto-increment
<code>id_penjualan_detail</code>	INT	Kunci asing ke table penjualan
<code>id_produk_detail</code>	INT	Kunci asing ke table produk
<code>harga_detail</code>	INT	Harga per unit produk dalam detail transaksi
<code>qty_detail</code>	INT	Jumlah unit produk yang terjual
<code>diskon_detail</code>	INT	Diskon yang diterapkan pada item ini
<code>total_detail</code>	INT	Total harga untuk item (harga x qty - diskon)
<code>created_at</code>	TIMESTAMP	Tanggal dan waktu saat dibuat
<code>updated_at</code>	TIMESTAMP	Tanggal dan waktu terakhir diperbarui

Deskripsi Kolom:

- **id_detail**: Identifikasi unik untuk setiap entri detail penjualan. Digunakan sebagai kunci primer.
- **id_penjualan_detail**: Menghubungkan detail dengan transaksi penjualan di tabel penjualan.
- **id_produk_detail**: Menghubungkan detail dengan produk yang terjual di tabel produk.
- **harga_detail**: Harga per unit produk yang terjual dalam transaksi ini.
- **qty_detail**: Jumlah unit produk yang terjual dalam transaksi ini.
- **diskon_detail**: Jumlah diskon yang diterapkan pada item ini.
- **total_detail**: Total harga untuk item ini, dihitung sebagai $\text{harga_detail} * \text{qty_detail} - \text{diskon_detail}$.
- **created_at**: Waktu dan tanggal saat data detail penjualan dibuat.
- **updated_at**: Waktu dan tanggal saat data detail penjualan terakhir kali diperbarui.

C. Entity - Relationship Diagram (ERD)

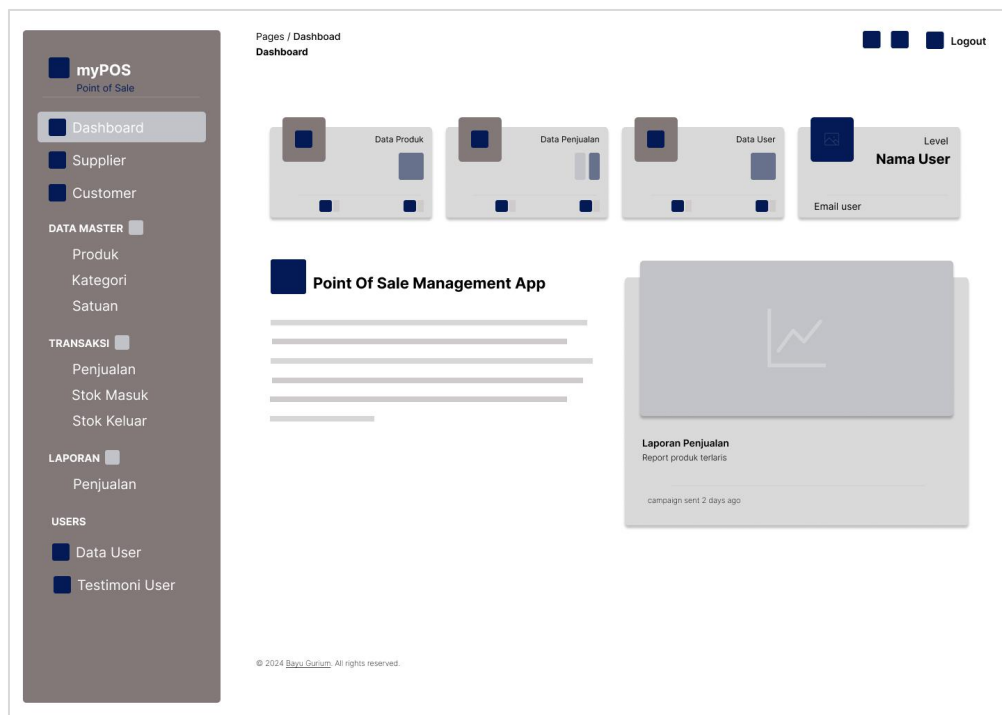
ERD (Entity Relationship Diagram) atau diagram hubungan entitas adalah diagram yang digunakan untuk perancangan suatu database dan menunjukkan

DESAIN

a. Wireframe

Pada tahap desain, saya membuat wireframe untuk aplikasi POS sebagai panduan awal untuk tata letak dan fungsionalitas antarmuka pengguna. Wireframe ini menggambarkan elemen-elemen utama seperti tombol, formulir, dan navigasi tanpa rincian visual akhir.

Dalam laporan ini, saya menyertakan gambaran umum wireframe yang menampilkan halaman utama dan fitur kunci aplikasi, seperti dashboard, halaman transaksi, dan halaman laporan. Untuk detail lebih lanjut, wireframe tambahan disertakan sebagai lampiran di akhir laporan atau dapat diakses melalui [tautan](#). Hal ini bertujuan untuk memberikan konteks dan pemahaman yang jelas tentang bagaimana aplikasi dirancang tanpa membebani laporan dengan terlalu banyak detail.



Gambar 11 :Wireframe halaman Dashboard Aplikasi POS

[illegible]

Gambar 12 : Wireframe halaman transaksi Penjualan Aplikasi POS

myPOS

Point of Sale

Dashboard

Supplier

Customer

DATA MASTER

Produk

Kategori

Satuan

TRANSAKSI

Penjualan

Stok Masuk

Stok Keluar

LAPORAN

Penjualan

USERS

Data User

Testimoni User

Pages / laporan penjualan

Laporan-Penjualan

Logout

Laporan Penjualan

Point Of Sale Management

Show 10 entries

Search :

INVOICE	TANGGAL	CUSTOMER	TOTAL	DISKON	GRAND TOTAL
Kode Invoice	Tanggal	Customer	Total	Diskon	Grand Total
Kode Invoice 2	Tanggal 2	Customer 2	Total 2	Diskon 2	Grand Total 2

Showing 2 to 2 of 2 entries

Previous

1

Next

Gambar 13 : Wireframe halaman Laporan Penjualan Aplikasi POS

IMPLEMENTASI

A. Metode Pengembangan

1. Metodologi Agile

Metode agile adalah kumpulan beberapa metode pengembangan perangkat lunak secara berulang dan bertahap sesuai kebutuhan. Kepopuleran metode ini karena saat pembuatan software sedang berlangsung tetap efisien dan fleksibel terhadap perubahan yang terjadi.

- **Iteraktif dan Inkremental:**

Pengembangan dilakukan dalam siklus berulang (iterasi) yang menghasilkan peningkatan atau fitur baru pada setiap siklus.

- **Fleksibilitas dan Adaptabilitas:**

Agile memungkinkan perubahan dan penyesuaian sepanjang siklus pengembangan, sehingga mudah untuk merespons kebutuhan yang berubah atau umpan balik dari pengguna.

- **Kolaborasi dan Komunikasi:**

Agile mendorong komunikasi yang erat dan kolaborasi antara pengembang, pengguna, dan pemangku kepentingan untuk memastikan semua pihak memiliki pemahaman yang sama tentang tujuan proyek.

- **Umpan Balik Berkelanjutan:**

Setelah setiap iterasi, umpan balik dikumpulkan dan digunakan untuk meningkatkan produk pada iterasi berikutnya, memastikan produk terus berkembang sesuai dengan kebutuhan pengguna.

- **Penekanan pada Kualitas:**

Agile mempromosikan praktik terbaik seperti pengujian otomatis dan integrasi berkelanjutan untuk memastikan produk berkualitas tinggi

- **Respons Cepat terhadap Perubahan:**

Dapat menyesuaikan dengan cepat terhadap perubahan kebutuhan bisnis.

- **Pengembangan Bertahap:**

Fitur dikembangkan dan diimplementasikan secara bertahap, memungkinkan pengujian dan perbaikan berkelanjutan.

- **Peningkatan Berkelanjutan:**

Evaluasi rutin dan umpan balik berkelanjutan memastikan produk terus berkembang dan memenuhi standar kualitas yang tinggi.

- **Pengelolaan Risiko yang Lebih Baik:**

Risiko diidentifikasi dan dimitigasi lebih awal dalam siklus pengembangan.

- **Fleksibilitas Individu:**

Memberikan fleksibilitas bagi pengembang individu untuk mengatur waktu dan prioritas proyek.

B. Pengembangan Sistem

Proses pengembangan sistem terdiri dari beberapa tools yang dibutuhkan untuk mencapai tujuan utama pengembangan sistem :

a. Teknologi yang Digunakan

1. Perangkat Keras / Hardware

- Leptop HP
- Processor AMD A4-9125 RADEON R3
- RAM 8GB DDR 4
- Sistem Operasi Windows 11 22H2
- Eksternal Keryboard Mecanical Keyboard
- Printer Thermal EPPOS RPP02

2. Perangkat Lunak / Software

- Codeigniter 4

Codeigniter 4 adalah sebuah framework/ kerangka kerja dari bahasa pemrograman PHP yang *small and powerfull* yang dirancang untuk mempermudah dan mempercepat proses pengembangan aplikasi web dengan arsitektur MVC (Model-View-Controller) yang membantu dalam pengembangan backend dengan fitur-fitur yang memudahkan pengelolaan database, routing, dan keamanan.

- Bootstrap v5

Bootstrap adalah salah satu framework dari CSS yang digunakan untuk desain front end, termasuk komponen UI, tata letak responsif, dan gaya visual. Bootstrap memudahkan pembuatan tampilan yang konsisten dan responsif di berbagai perangkat.

- MySQL dan MySQL Workbench

MySQL digunakan sebagai sistem manajemen basis data (DBMS) untuk menyimpan dan mengelola data aplikasi POS. MySQL Workbench adalah alat untuk merancang, mengelola, dan mengelola database MySQL secara visual.

- XAMPP

XAMPP adalah paket perangkat lunak yang mencakup Apache, MySQL, PHP, dan Perl, dan digunakan sebagai server lokal untuk mengembangkan dan menguji aplikasi secara lokal sebelum di-deploy ke lingkungan produksi.

- jQuery AJAX

Digunakan untuk melakukan permintaan HTTP asinkron ke server tanpa memuat ulang halaman. Ini memungkinkan interaksi dinamis dengan server, seperti memuat data baru, mengirim data formulir, atau memperbarui bagian halaman secara real-time.

- ResponsivelyApp

Adalah software yang digunakan untuk memantau tampilan website di semua device mulai dari tampilan desktop, tablet sampai tampilan mobile

- Visual Studio Code

Visual Studio Code (VS Code) adalah editor kode sumber yang ringan dan dapat dikustomisasi, yang mendukung berbagai bahasa pemrograman dan alat pengembangan.

- Composer

Composer adalah alat manajemen dependensi untuk PHP yang memungkinkan kita dapat mengelola pustaka pihak ketiga yang dibutuhkan oleh proyek.

-Draw.io

Draw.io (sekarang dikenal sebagai diagrams.net) adalah alat untuk membuat diagram dan grafik, yang berguna untuk membuat diagram alur, diagram kelas, dan lainnya.

- Figma

Figma adalah salah satu tools berbasis website yang digunakan untuk mendesain kapan saja dan dimanapun melalui internet. Pada

umumnya Figma digunakan untuk mendesain interface dari sebuah aplikasi yang ingin dibuat.

b. Proses Coding dan Version Control

Pada tahap coding, saya menggunakan CodeIgniter 4 sebagai framework PHP dan Bootstrap 5 untuk antarmuka pengguna. Struktur aplikasi mengikuti pola Model-View-Controller (MVC), yang memisahkan logika aplikasi, tampilan, dan data. saya menerapkan standar penamaan variabel yang konsisten dan menulis komentar yang jelas untuk setiap fungsi. Fitur utama meliputi manajemen produk, transaksi penjualan, dan laporan, dengan setiap fitur diuji secara menyeluruh untuk memastikan fungsionalitas yang tepat.

- Version Control

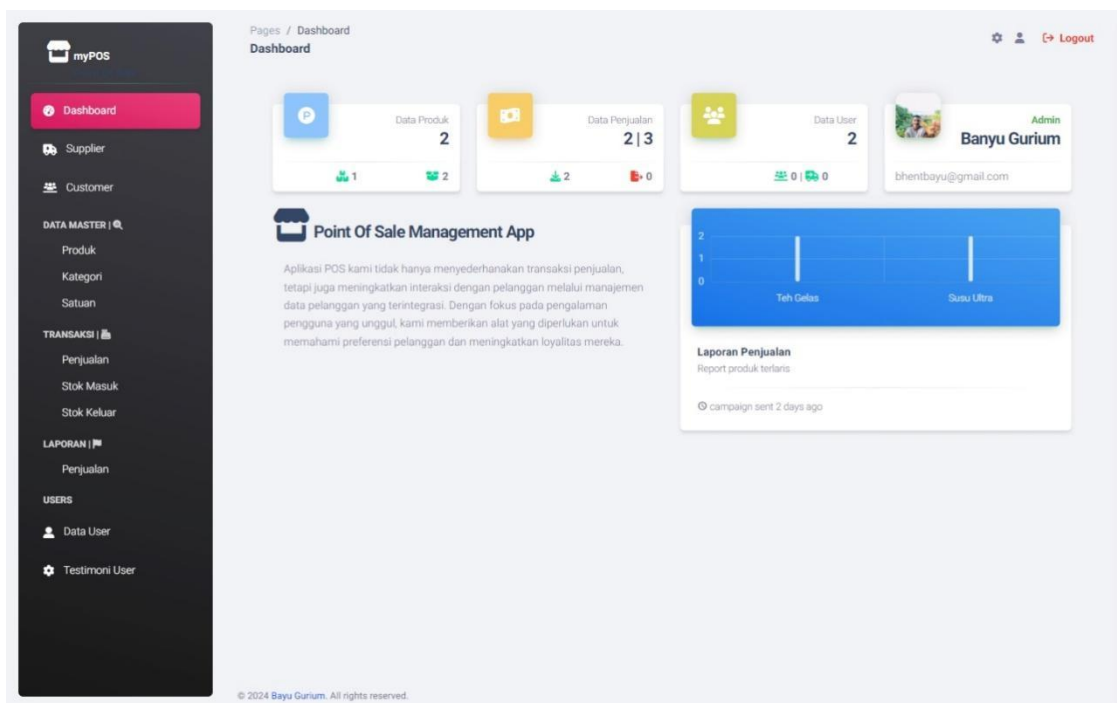
Untuk version control, saya menggunakan Git sebagai alat pengelola versi. Repositori proyek disimpan di GitHub, dan saya menerapkan strategi branching dengan menggunakan branch utama untuk produksi dan branch terpisah untuk pengembangan fitur baru. Setiap commit disertai dengan pesan deskriptif yang menjelaskan perubahan yang dilakukan. saya juga melakukan merge dari branch fitur ke branch utama setelah melalui proses review dan pengujian untuk memastikan kualitas kode.

TANGKAPAN LAYAR PROYEK

Dalam laporan ini, saya menyertakan beberapa tampilan umum yang menampilkan halaman utama dan fitur kunci aplikasi, seperti dashboard, halaman transaksi, dan halaman laporan. Untuk detail lebih lanjut tentang tampilan sistem dapat diakses melalui [tautan](#).

1. Halaman Dashboard

Pada halaman dashboard sistem menampilkan bebrapa data seperti data produk, penjualan dan data user. Dan sudah responsive pada semua perangkat.



Gambar 14 : Tampilan halaman Dashboard

2. Halaman Transaksi Penjualan

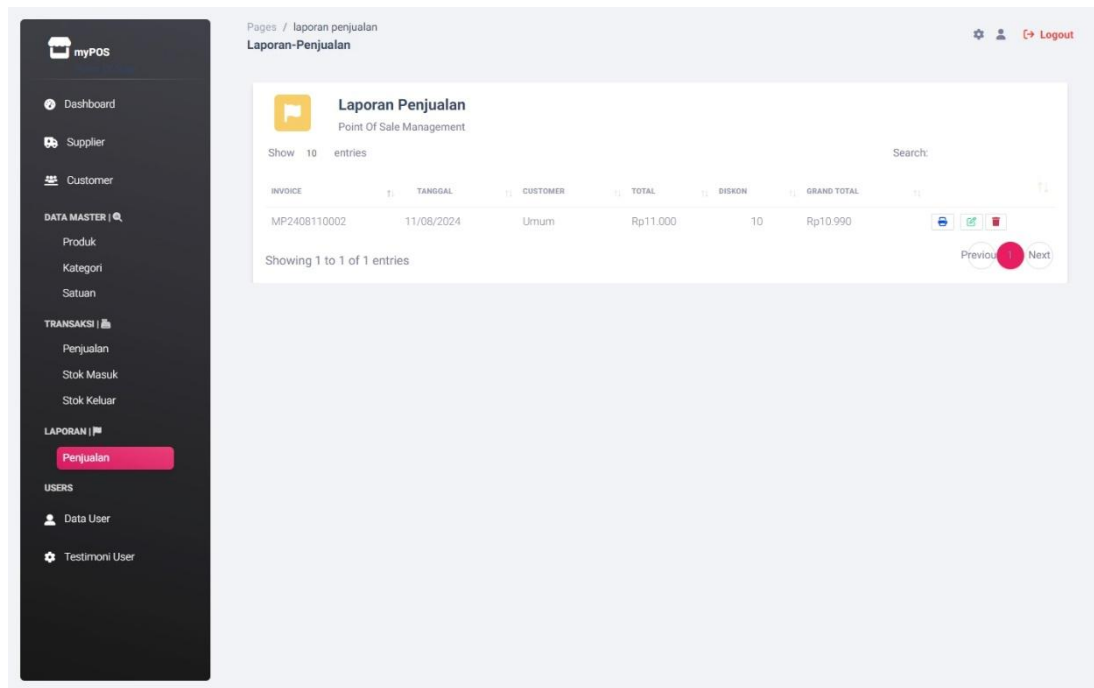
Menampilkan layout dan penempatan elemen proses transaksi penjualan produk halaman transaksi penjualan juga sudah responsive dan tampil baik di semua perangkat.

The screenshot displays the 'myPOS' sales transaction interface. On the left is a dark sidebar menu with categories: 'Dashboard', 'Supplier', 'Customer', 'DATA MASTER' (including Produk, Kategori, Satuan), 'TRANSAKSI' (with 'Penjualan' highlighted), 'LAPORAN' (including Penjualan), and 'USERS' (including Data User, Testimoni User). The main content area is titled 'Penjualan'. It features input fields for 'Tanggal' (08/11/2024), 'Kasir' (Banyu Gurum), and 'Customer' (Umum). A 'Produk' section includes a search bar and a 'Jumlah' (1) field with a 'TAMBAH' button. Below this is a table with headers: 'Produk', 'Nama Produk', 'Harga', 'Jumlah', 'Diskon', and 'Total'. To the right of the table is a summary section with fields for 'Sub Total', 'Diskon', 'Grand Total', 'Cash', 'Kembalian', and 'Nota', all showing '0'. A large 'Rp 0' is prominently displayed. At the bottom right are 'CLEAR' and 'SIMPAN TRANSAKSI' buttons. The top right corner has a 'Logout' link.

Gambar 15: Tampilan halaman Transaksi Penjualan

3. Halaman Laporan Penjualan

Menampilkan data laporan penjualan mulai dari data kasir sampai dengan data produk yang terjual.



Gambar 16 : Tampilan Halaman Laporan Penjualan

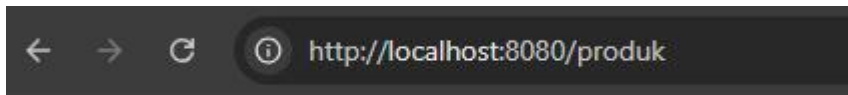
ALUR PROSES

Dengan menggunakan pendekatan MVC (*Model, View, Controller*) saya akan menjelaskan proses request dan respon data menggunakan pendekatan codeigniter 4.

Sebagai contoh saya akan menjelaskan langkah-langkah bagaimana proses menampilkan halaman data produk dengan cpdeigniter 4

1. Penerimaan Permintaan atau Request

User mengakses URL missal `/produk` di browser



Selanjutnya request akan diarahkan ke `Router/ rute` CodeIgniter 4 menggunakan sistem routing untuk memetakan URL yang diminta ke controller dan metode yang sesuai.

```
$routes->get('/produk', 'ProductController::index');
```

Ini berarti ketika URL /produk diakses, CodeIgniter akan memanggil metode index dari ProductController.

2. Diarahkan ke Controller

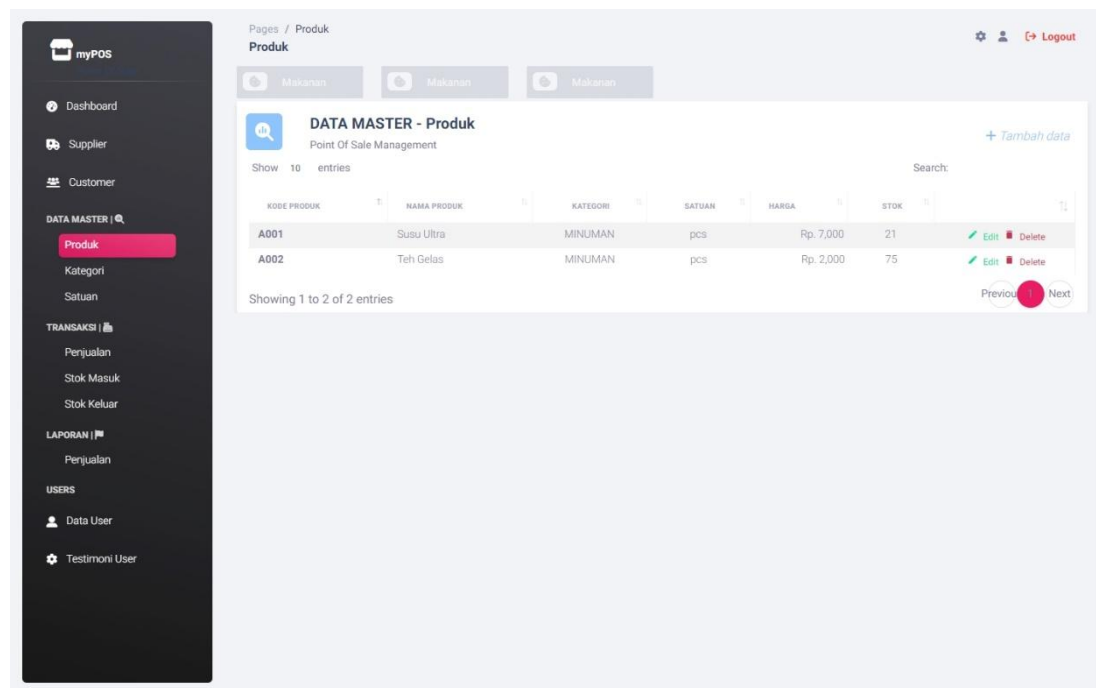
Controller adalah komponen yang mengatur logika bisnis dan memproses data sebelum dikirim ke view.

```
2
3 namespace App\Controllers;
4
5 use App\Controllers\BaseController;
6 use CodeIgniter\HTTP\ResponseInterface;
7
8 class Produk extends BaseController
9 {
10     public function index()
11     {
12
13         // load model ProdukModel
14         $produkModel = $this->loadModel('ProdukModel');
15
16         $data = [
17             'title'      => 'Produk',
18             'breadcrumb' => 'Produk',
19             'data_produk' => $produkModel->selectAllProduk(),
20
21         ];
22         return view('produk/index', $data);
23     }
```

Dalam metode index, controller memproses data yang diperlukan dan memanggil view dengan data tersebut. Contohnya data Produk

3. Pengembalian Data dan Menampilkan View

Dilihat pada method `index()` ada sebuah variable `$produkModel` maksud dari code tersebut adalah untuk memanggil model `ProdukModel` di dalam method `index` tersebut agar semua data yang ada pada model dapat di gunakan untuk ditampilkan pada view produk. `ProdukModel` bertugas untuk mengelola data produk. Kode selanjutnya yaitu `return view('produk/index', $data)` maksud dari kode ini yaitu method `index()` akan memanggil sebuah tampilan di dalam folder `views/produk/index.php` jadi tampilannya ada di dalam folder tersebut dan jika ada tampilannya maka akan tampil :



Gambar 17 : Tampilan halaman Data Produk

Dengan menggunakan framework codeigniter 4 sangat memudahkan dan mempercepat proses pengembangan karena memiliki banyak fitur-fitur yang dapat kita kembangkan sesuai kebutuhan. Dengan pattern MVC kita dapat dengan mudah manajemen tampilan maupun data pada proyek kita karena sudah di pisahkan sesuai kategorinya masing-masing.

PENGUJIAN

a. Jenis pengujian yang digunakan yaitu :

- Unit Testing

Metode pengujian ini yaitu menguji bagian kecil dari kode (unit) secara individual, mulai dari penamaan variable, query dll.

- Integration Testing

Menguji dan memastikan setiap fitur yang berintegrasi dengan banyak data sekalipun tidak mengalami kendala dalam proses GET dan POS.

Dengan melakukan beberapa pengujian di atas saya menemukan beberapa (bug/error) terutama pada saat proses transaksi penjualan dan sudah saya lakukan perbaikan pada masalah tersebut dan sistem kembali stabil.

Melalui beberapa tahapan di atas saya dapat mengembangkan sistem ini sesuai desain yang telah saya tentukan yaitu

- Fungsionalitas Utama

Manajemen Produk: Telah berhasil mengimplementasikan fitur CRUD untuk produk, termasuk kemampuan pencarian dan filter.

Transaksi Penjualan: Sistem dapat melakukan transaksi penjualan dan mencetak struk dengan informasi yang akurat.

Laporan dan Analitik: Menyediakan laporan penjualan dan grafik untuk analisis performa.

- Peningkatan Performa:

Optimisasi Kode: Kode telah diperiksa dan dioptimalkan untuk memastikan performa yang lebih baik dan responsif.

Penghapusan Kode Tidak Perlu: Kode yang tidak diperlukan telah dihapus untuk meningkatkan efisiensi sistem.

- Keamanan dan Akses:

Autentikasi dan Otorisasi: Sistem telah dilengkapi dengan kontrol akses dan autentikasi untuk melindungi data sensitif.

Keamanan Data: Enkripsi data dan perlindungan terhadap potensi ancaman telah diterapkan.

- User Experience (UX)

Antarmuka Pengguna: Antarmuka telah dirancang untuk kemudahan penggunaan dengan feedback positif dari pengguna.

Dukungan dan Dokumentasi: Dokumentasi pengguna dan pelatihan telah disediakan untuk memudahkan pengguna dalam menggunakan sistem.

- Feedback Pengguna:

Umpan Balik: Umpan balik dari pengguna telah diterima dan diintegrasikan untuk meningkatkan fungsionalitas dan pengalaman pengguna.

KESIMPULAN

Dengan adopsi yang luas di berbagai sektor bisnis, Point of Sale tidak hanya menjadi alat penting untuk meningkatkan efisiensi operasional, tetapi juga untuk menghadirkan pengalaman pelanggan yang lebih baik dan mendukung pertumbuhan bisnis secara keseluruhan. Di era digital dan kompetitif seperti sekarang, implementasi yang tepat dari teknologi POS menjadi kunci keberhasilan dan keberlanjutan bisnis.

Aplikasi Manajemen Point of Sale (POS) ini dirancang untuk membantu pemilik usaha dalam mengelola berbagai aspek bisnis mereka dengan lebih efektif dan efisien. Dengan fitur-fitur lengkap seperti pengelolaan data produk, transaksi penjualan, manajemen persediaan, dan laporan penjualan, aplikasi ini dapat menjadi alat yang sangat berguna untuk mengoptimalkan operasional bisnis.

LAMPIRAN

Link Github data Projek My Pos : <https://github.com/banyu142001/POS--Assets>

Link Github : <https://github.com/banyu142001>

Link Website Asset Pojek : <https://banyu142001.github.io/bayu-mypos-assets/assets.html>