



1. Tim Developer Aplikasi Go kantin (Canelly)

- Ketua

Nama : Ichsan Fathur Rahman

NIM : 2022071023

Tugas

- Mengatur jalannya pengembangan proyek secara keseluruhan, termasuk menyusun rencana kerja dan timeline.
- Memimpin rapat tim, mendistribusikan tugas kepada anggota, dan memastikan setiap anggota bekerja sesuai dengan tanggung jawabnya.
- Mengembangkan modul backend utama, seperti sistem autentikasi pengguna dan pengolahan data pesanan menggunakan PHP dan MySQL.
- Bertanggung jawab terhadap integrasi seluruh komponen aplikasi agar berjalan dengan baik dan sesuai dengan kebutuhan proyek.

- Anggota 1

Nama : Ellyza Hardianty

NIM : 2022071015

Tugas

- Mengelola database MySQL, termasuk perancangan tabel untuk menyimpan data pengguna, dashboard dan wish list.
- Membantu pengembangan fungsi backend untuk mendukung proses pemesanan dan pengelolaan data secara efisien.
- Melakukan pengujian sistem database untuk memastikan data tersimpan dengan benar dan tidak terjadi kesalahan.

- Anggota 2

Nama : Anggi Saputri

NIM : 2022071042

Tugas

- Merancang antarmuka pengguna (UI) menggunakan Bootstrap 5 untuk memastikan aplikasi responsif dan mudah digunakan di berbagai perangkat.
- Menyusun halaman frontend utama, seperti halaman dashboard dan profil pengguna.
- Mengoptimalkan tampilan agar terlihat menarik dan intuitif, sesuai dengan preferensi pengguna target.
- Melakukan uji tampilan untuk memastikan kompatibilitas pada berbagai ukuran layar dan perangkat.

2. Deskripsi Aplikasi

Go Kantin adalah aplikasi berbasis web yang dirancang untuk memberikan kemudahan dalam membeli makanan dan minuman di kantin tanpa harus mengantri. Dikembangkan menggunakan Laravel, PHP, MySQL, dan Bootstrap 5, aplikasi ini menawarkan pengalaman yang lebih cepat dan efisien bagi pengguna. Berfokus pada antarmuka web responsif yang dapat diakses melalui berbagai perangkat, seperti desktop, laptop, dan tablet. Dengan lima menu utama—“Home,” “Dashboard,” “Food and Drink,” dan “Cart”. Pengguna dapat dengan mudah menavigasi seluruh fitur aplikasi, mulai dari melihat daftar menu hingga melakukan pemesanan.

Semua data yang ditampilkan, seperti menu makanan dan informasi pengguna, terhubung langsung dengan database MySQL melalui fitur Query Builder Laravel, memungkinkan pembaruan konten secara otomatis tanpa memerlukan perubahan manual pada kode aplikasi. Desain responsif yang didukung oleh Bootstrap 5 memastikan pengalaman pengguna tetap optimal di berbagai ukuran layar.

Dengan fitur-fitur praktis dan inovatif yang dioptimalkan untuk platform web, **Go Kantin** tidak hanya meningkatkan efisiensi operasional kantin tetapi juga memberikan pengalaman belanja yang lebih modern dan memuaskan bagi para pelanggan.

3. Spesifikasi Kebutuhan User

Spesifikasi Kebutuhan Input dan Output Data dari Sistem

1. Tampilan Login

- **Input:**
 - NIP/NIM/ID.
 - Password.
 - Tombol login.
- **Output:**
 - Pilihan Membuat Akun.
 - Menampilkan Menu Dashboard.

2. Tampilan SignUp

- **Input:**
 - Fullname.
 - NIP/NIM/ID.
 - Phone Number.
 - Password.
 - Tombol sign-up.

- **Output:**
 - Menu untuk Melakukan Login.

3. Dashboard

- **Input:**
 - Menu navigasi untuk akses ke halaman Dashboard, Food and Drink, Cart, Edit Profile, dan Logout.
- **Output:**
 - Menu navigasi untuk akses ke halaman Dashboard, Food and Drink, Cart, Edit Profile, dan Logout.
 - Welcome message dengan nama pengguna.

4. Food and Drink

- **Input**
 - Tidak ada.
- **Output:**
 - Daftar menu makanan dan minuman.
 - Harga per item.
 - Nama menu Makanan dan minuman.

5. Cart

- **Input:**
 - Pilihan menu item untuk ditambahkan ke keranjang.
 - Tombol Add.
 - Tombol "Proceed to Checkout".
 - Pilihan Payment Method
 - Tombol "Pay Now"
 - Tombol "Order More"
- **Output:**
 - Nama menu yang dipilih.
 - Total semua item yang dipilih
 - Notifikasi You have successfully made the payment!

6. Edit Pengguna

- **Input**
 - Dapat Mengubah Fullname
 - Dapat Mengubah NIP/NIM ID
 - Dapat Mengubah Phone Number
 - Dapat Mengubah Password
 - Tombol Edite Profile
- **Output:**
 - Nama lengkap pengguna.
 - Email pengguna.
 - Nomor telepon pengguna.
 - Foto profil pengguna.
 - Opsi untuk mengubah data profil (email, password).
 - Opsi untuk Edit Profile.

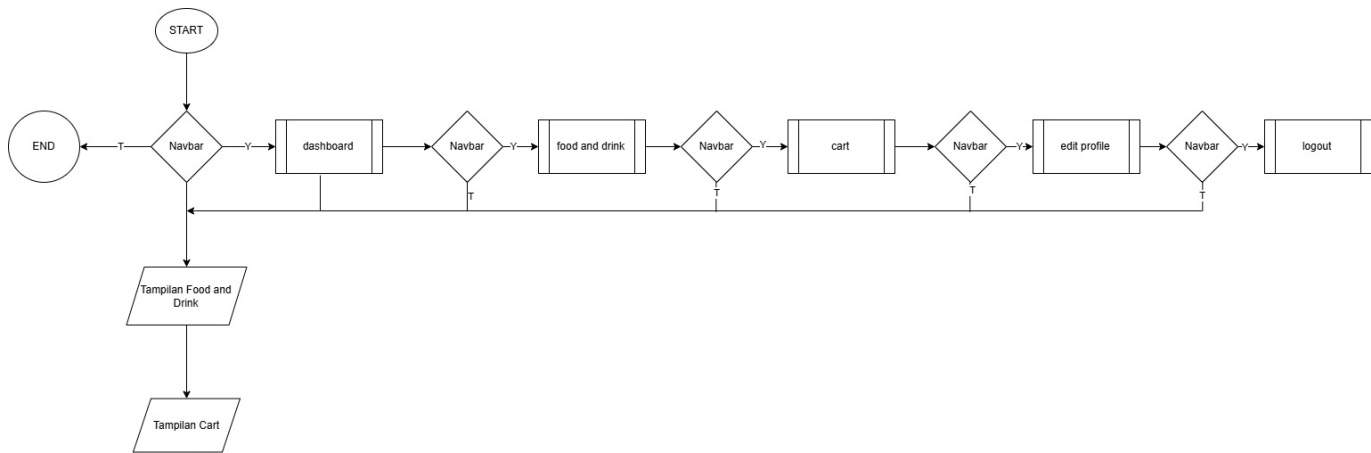
- Notifikasi Profile Berhasil di Update

7. Logout

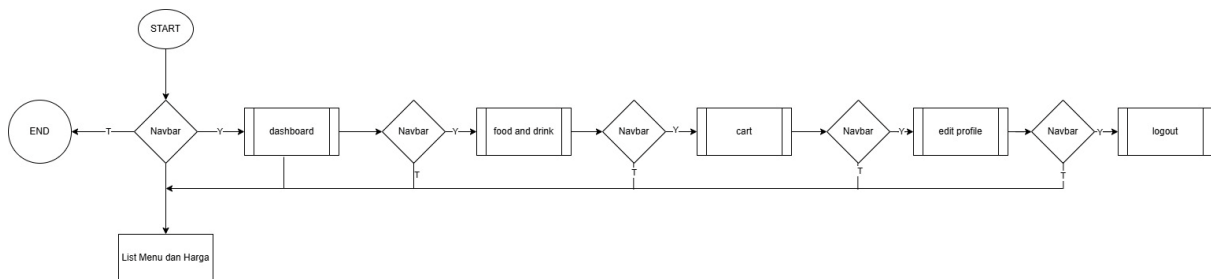
- **Input**
 - Pilihan tombol menu “logout/batal”
- **Output:**
 - Jika klik tombol “batal” akan ke menu dashboard
 - Jika klik “Ya, logout” akan keluar dan balik ke tampilan menu landing.

4. Rekayasa Proses

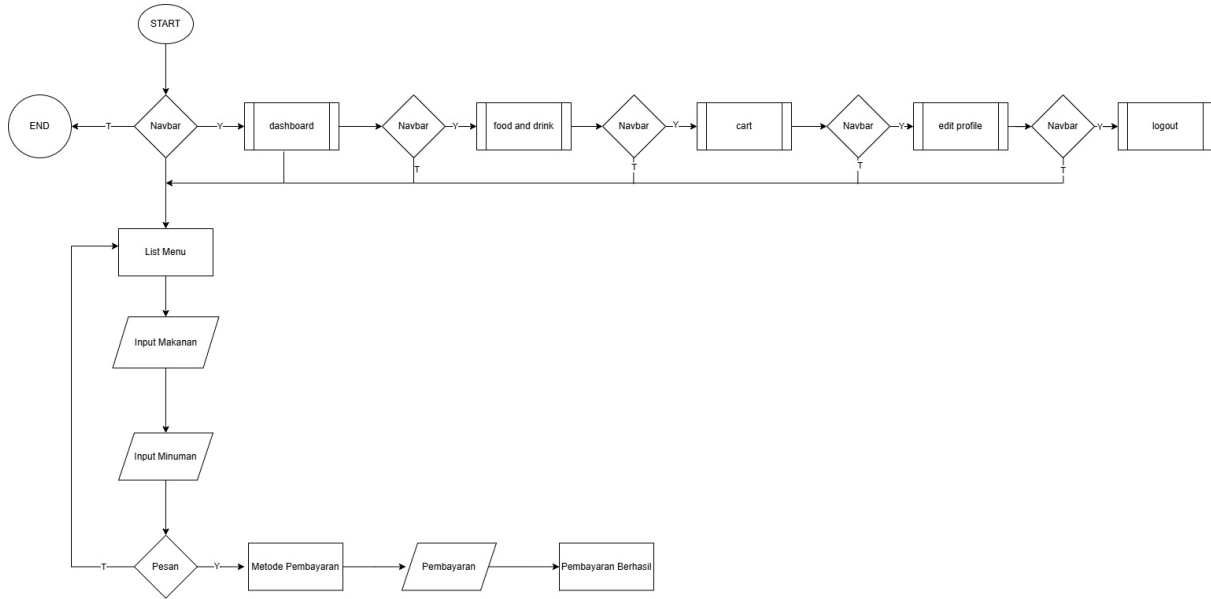
- Flowchart Dashboard



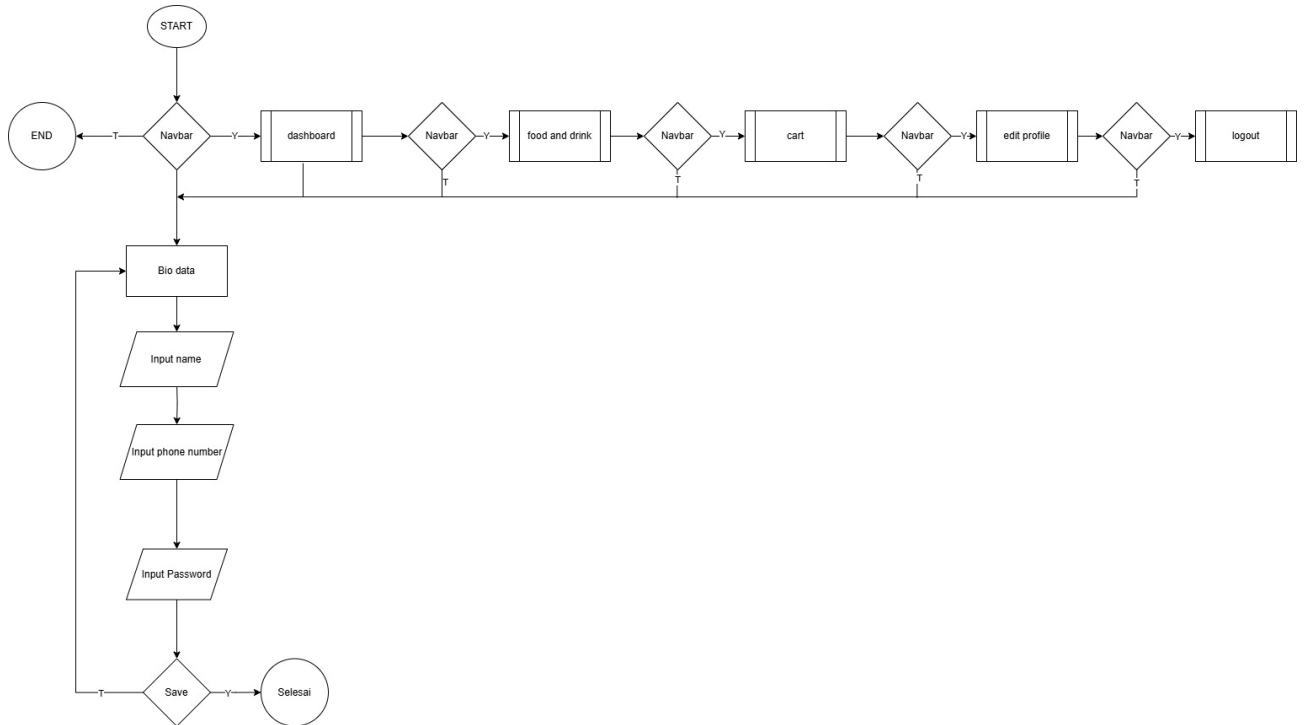
- Flowchart Menu



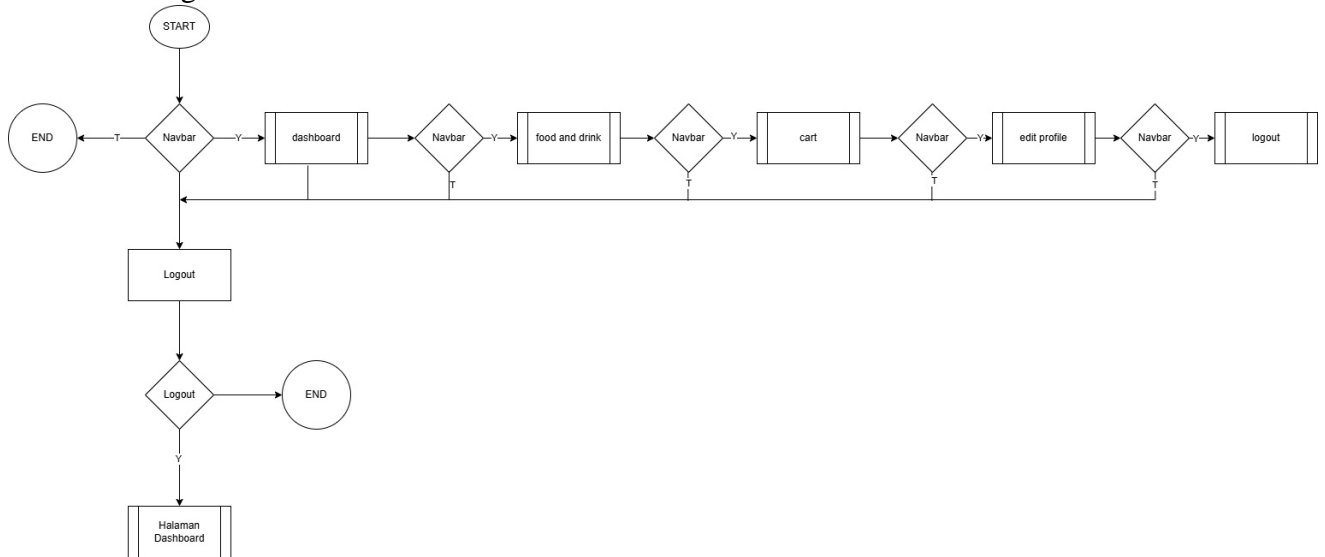
- Flowchart Cart



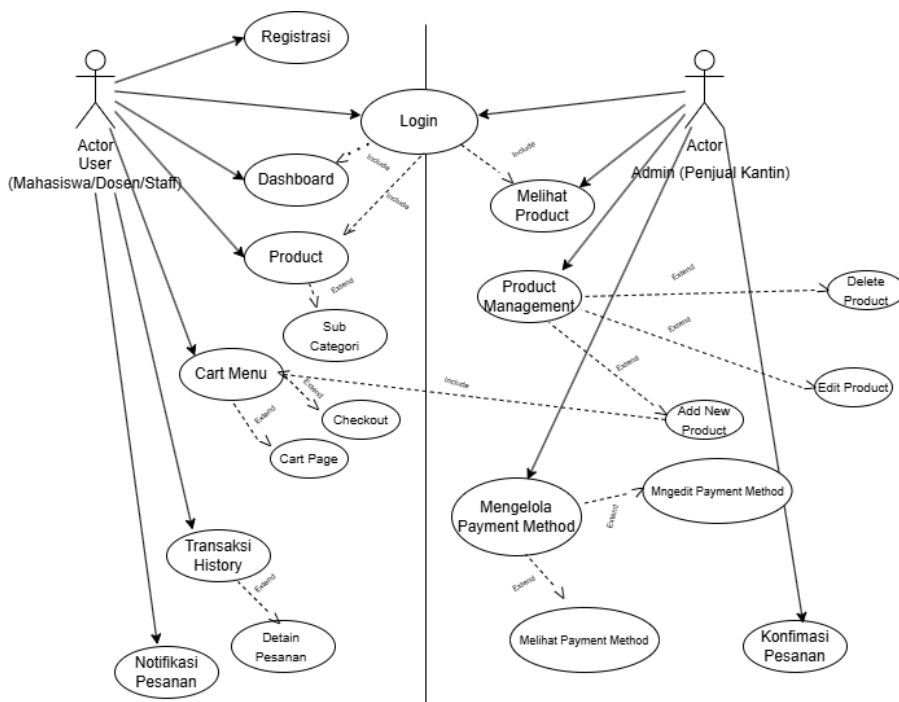
- Flowchart Edit Profile



- Flowchart Log out



- Diagram Use Case



- Scenario Usecase

Use Case login		
Deskripsi	:	Use case ini menangani verifikasi dan hak akses memasukkan data sebagai pengguna
Aktor	:	User
Kondisi Awal	:	Sistem menampilkan halaman login
Kondisi Akhir	:	Sistem berhasil masuk ke dashboard aplikasi atau mendapat pesan error jika login gagal.

Skenario Normal	
Aksi Aktor	Reaksi Sistem
1. User memasukkan NIP/NIM/ID dan password pada form login.	1. Sistem memvalidasi data yang dimasukkan pengguna.
	2. Sistem menampilkan halaman dashboard.
Skenario Gagal	
Aksi Aktor	Reaksi Sistem
1. User memasukkan NIP/NIM/ID dan password pada form login.	1. Sistem memvalidasi data yang dimasukkan pengguna.
	2. Sistem menampilkan pesan error "NIM/NIP/ID atau password salah".

Use Case Sign Up		
Deskripsi	:	Use case ini menangani pembuatan akun baru user
Aktor	:	User
Kondisi Awal	:	Sistem menampilkan halaman Sign Up
Kondisi Akhir	:	Sistem berhasil masuk ke login atau mendapat pesan error jika Sign Up gagal.
Skenario Normal		
Aksi Aktor		Reaksi Sistem
1. User memasukkan nama, NIP/NIM/ID, nomor telfon dan password pada form login.	2. Sistem memvalidasi data yang dimasukkan pengguna.	
	3. Sistem menampilkan halaman dashboard.	
Skenario Gagal		
Aksi Aktor		Reaksi Sistem
1. User memasukkan nama, NIP/NIM/ID, nomor telfon dan password pada form login.	2. Sistem memvalidasi data yang dimasukkan pengguna.	
	3. Sistem menampilkan pesan error "Data tidak valid".	

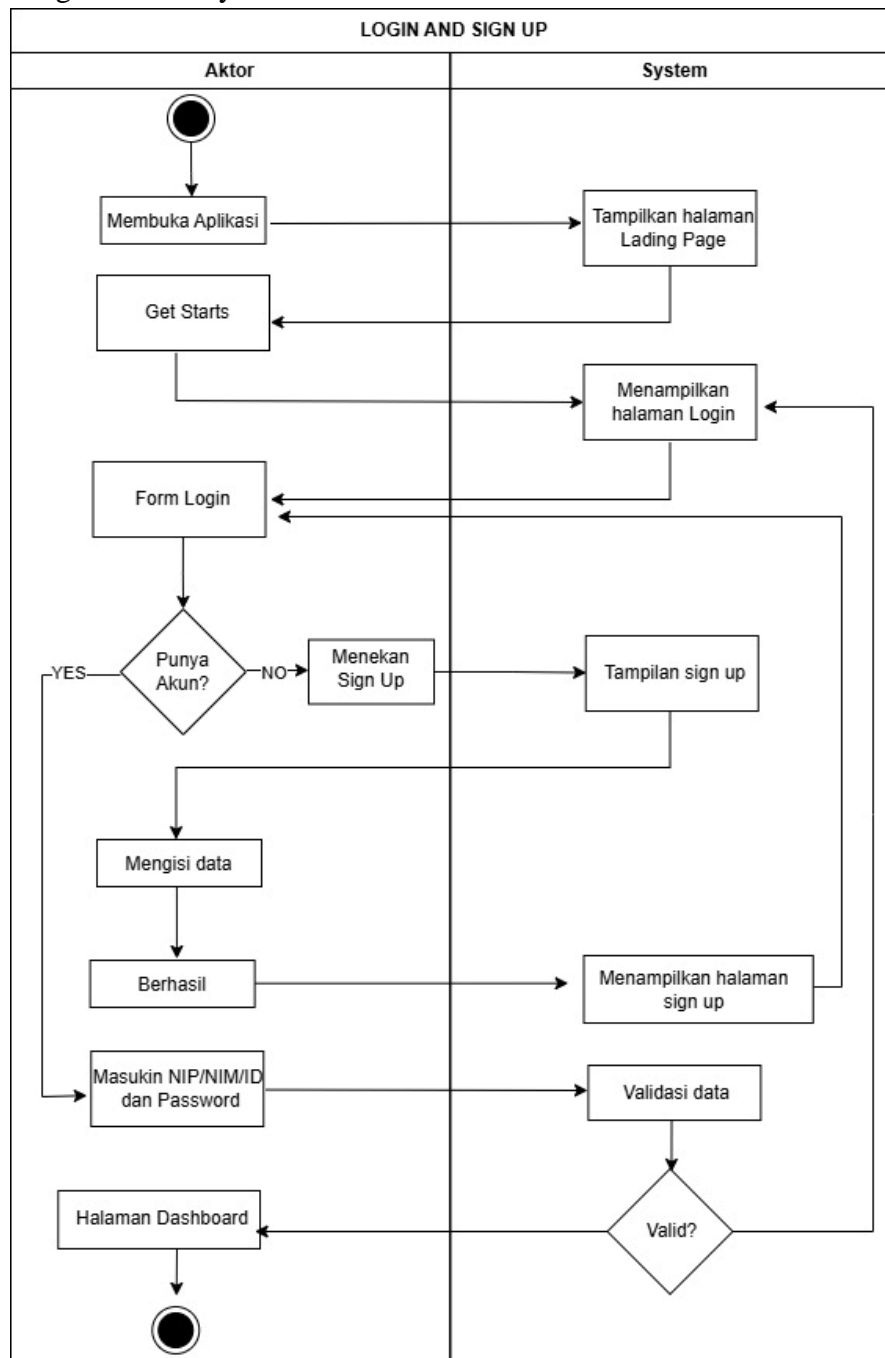
Use Case Edit Profile		
Deskripsi	:	Use case ini menangani mengedit data akun seperti nama atau email user.
Aktor	:	User
Kondisi Awal	:	User sudah login dan berada di halaman "Edit Profile".
Kondisi Akhir	:	Profile user berhasil diperbarui
Skenario Normal		
Aksi Aktor		Reaksi Sistem

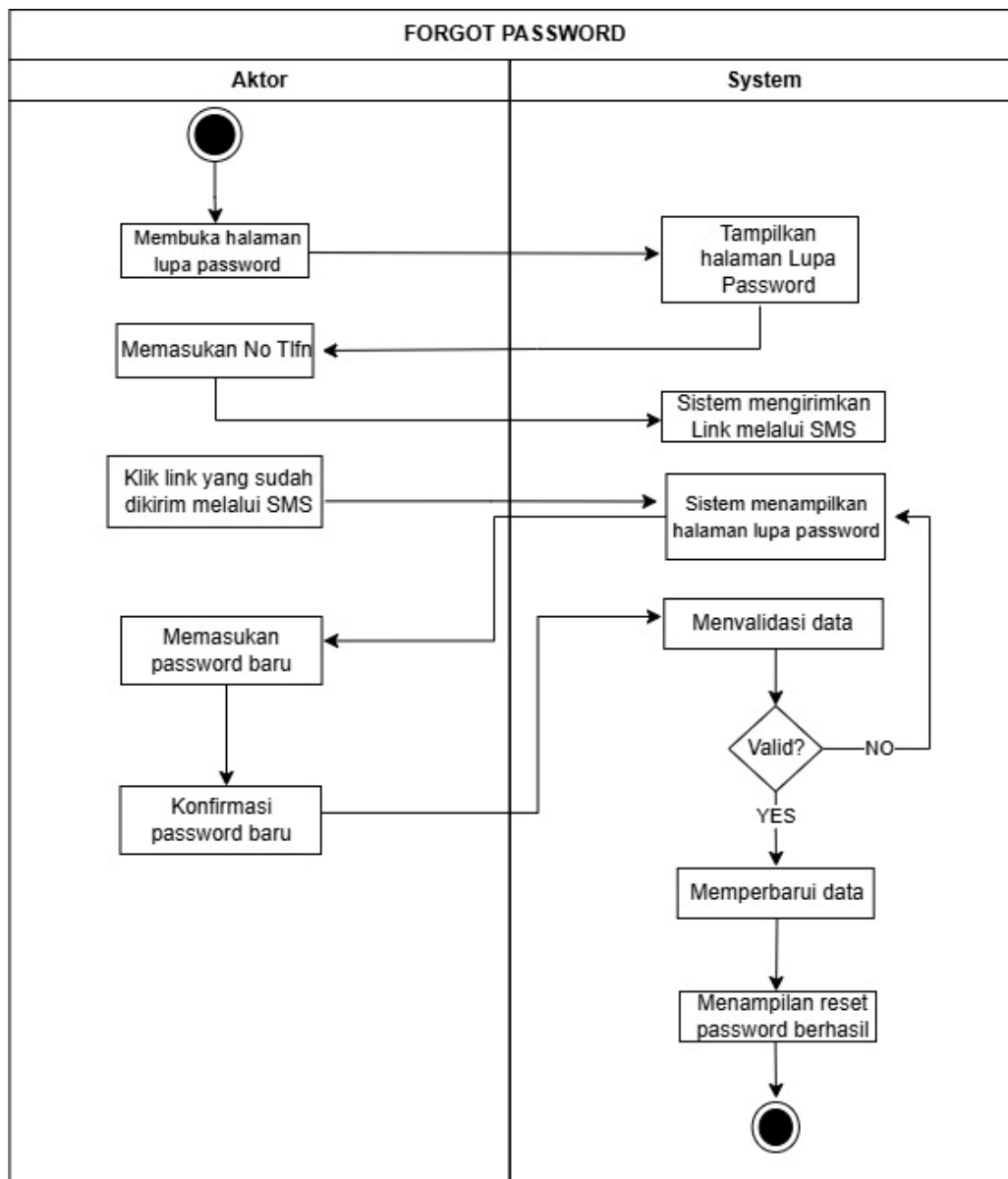
1. Pengguna membuka halaman "Edit Profile".	
2. User Mengubah data	
3. Menekan tombol "save".	
	4. Sistem memvalidasi data yang diubah.
	5. Sistem memperbarui data di database.
	Sistem menampilkan pesan "Profil berhasil diperbarui".
Skenario Gagal	
1. Pengguna membuka halaman "Edit Profile".	
2. User Mengubah data	
3. Menekan tombol "save".	
	4. Sistem memvalidasi data yang diubah. (contoh: email salah format).
	5. Sistem menampilkan pesan error "Email tidak valid".

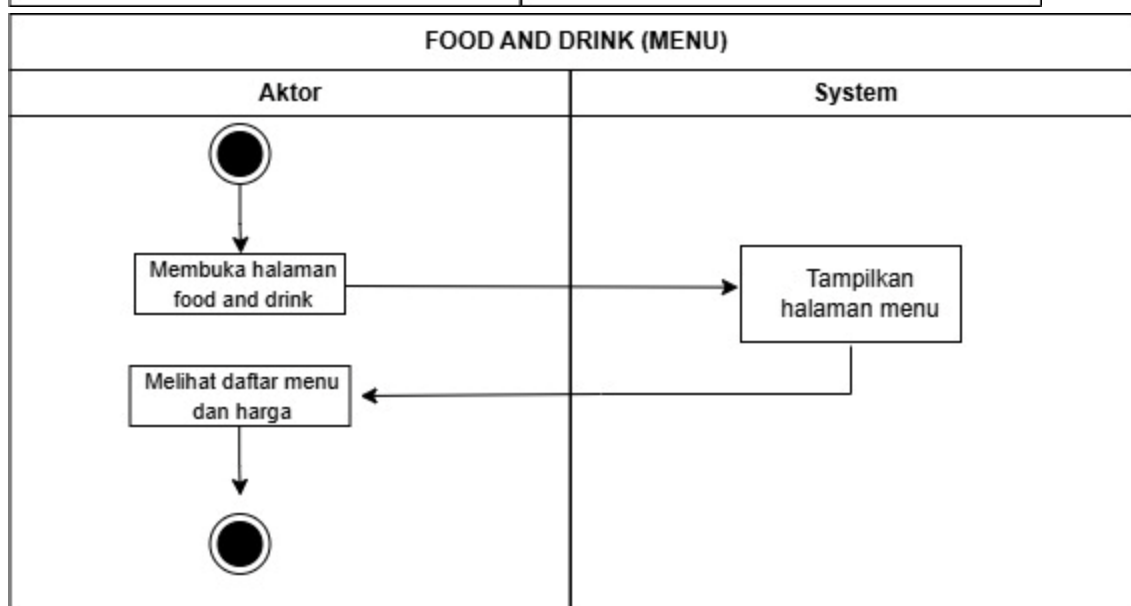
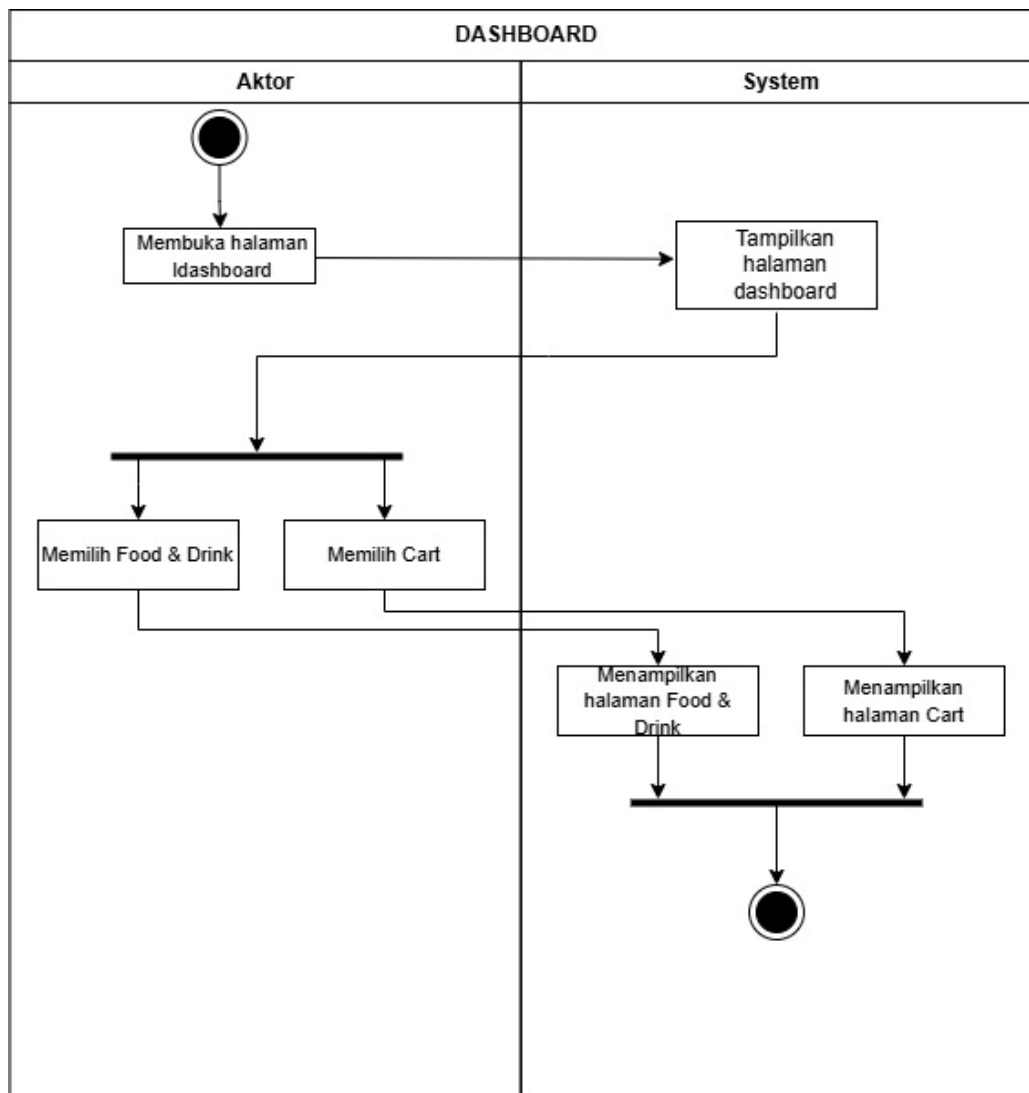
Use Case Food And Drink		
Deskripsi	:	Use case ini menampilkan daftar makanan atau minuman yang tersedia.
Aktor	:	User
Kondisi Awal	:	Sistem menampilkan tampilan menu makanan atau minuman
Kondisi Akhir	:	Item berhasil ditambahkan ke cart pengguna.
Skenario Normal		
Aksi Aktor		Reaksi Sistem
1. Pengguna membuka halaman menu makanan.		
		2. Sistem menampilkan daftar makanan yang tersedia.
Skenario Gagal		
Aksi Aktor		Reaksi Sistem
1. Pengguna membuka halaman menu makanan.		
		2. Sistem gagal memuat data makanan (contoh: masalah koneksi).
		3. Sistem menampilkan pesan error "Gagal memuat data".

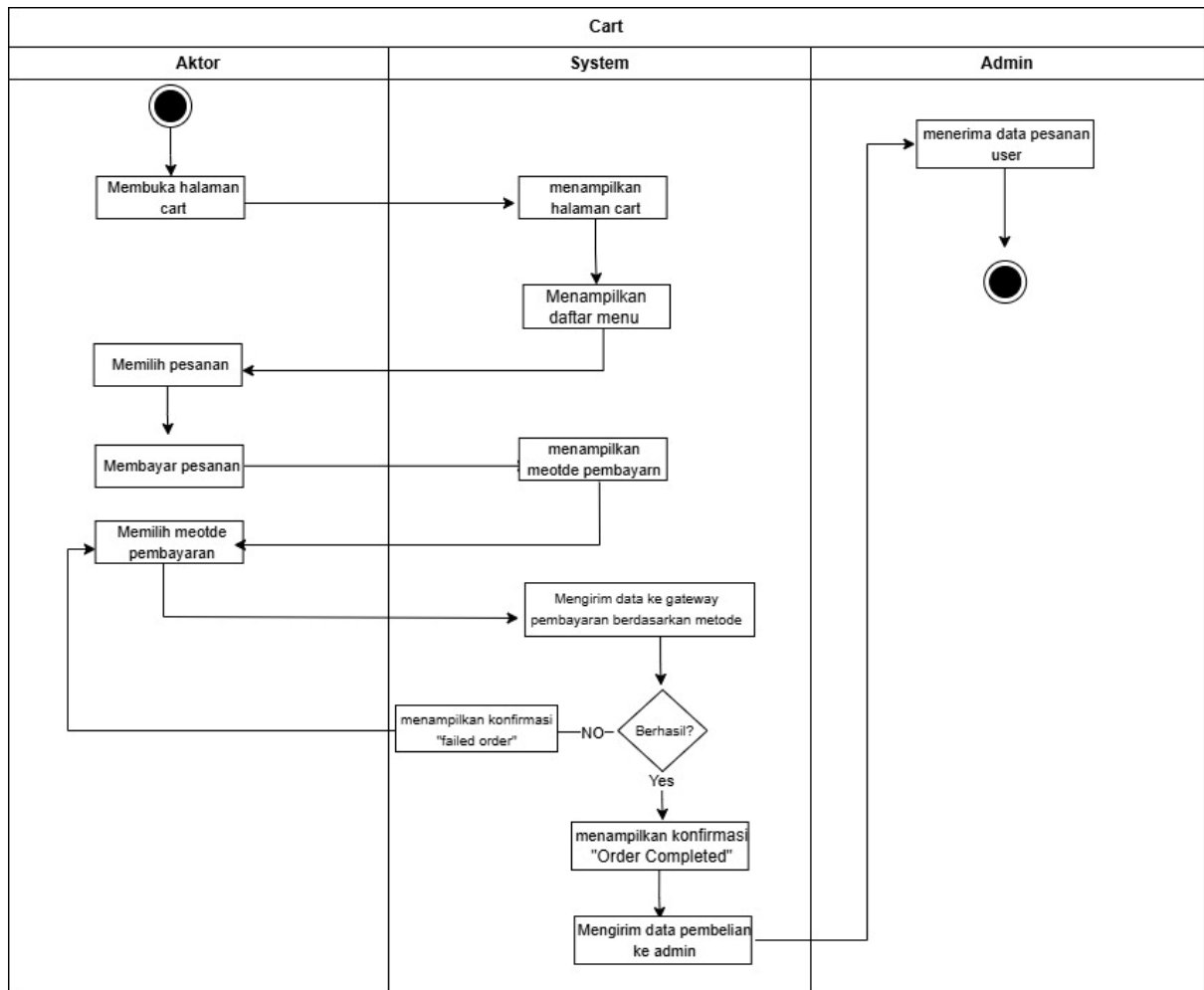
Use Case cart		
Deskripsi	:	Use case melakukan proses pembayaran untuk item di cart.
Aktor	:	User
Kondisi Awal	:	Sistem menampilkan tampilan cart
Kondisi Akhir	:	Pembayaran selesai, dan user mendapat konfirmasi "Order Completed".
Skenario Normal		
Aksi Aktor		Reaksi Sistem
1. User membuka halaman cart		
2. User memilih pesanan		
3. User melakukan checkout		
		4. Sistem memproses pembayaran dan menampilkan metode pembayaran
5. User memilih metode pembayaran		
		5. Sistem mengalihkan pembayaran ke aplikasih sesuai metode pembayaran
6. User melakukan transaksi		7. Sistem memproses transaksi dengan menampilkan “waiting for payment” 8. Sistem menampilkan halaman "Order Completed" sebagai konfirmasi. 9. Sistem mengirimkan data pesanan user kepada admin.
Skenario Gagal		
Aksi Aktor		Reaksi Sistem
1. User membuka halaman cart		
2. User memilih pesanan		
3. User melakukan checkout		
		4. Sistem memproses pembayaran dan menampilkan metode pembayaran
4. User memilih metode pembayaran		
		5. Sistem mengalihkan pembayaran ke aplikasih sesuai metode pembayaran
6. User melakukan transaksi		7. Sistem gagal memproses pembayaran (contoh: saldo tidak cukup). 8. Sistem menampilkan pesan error "Pembayaran gagal".

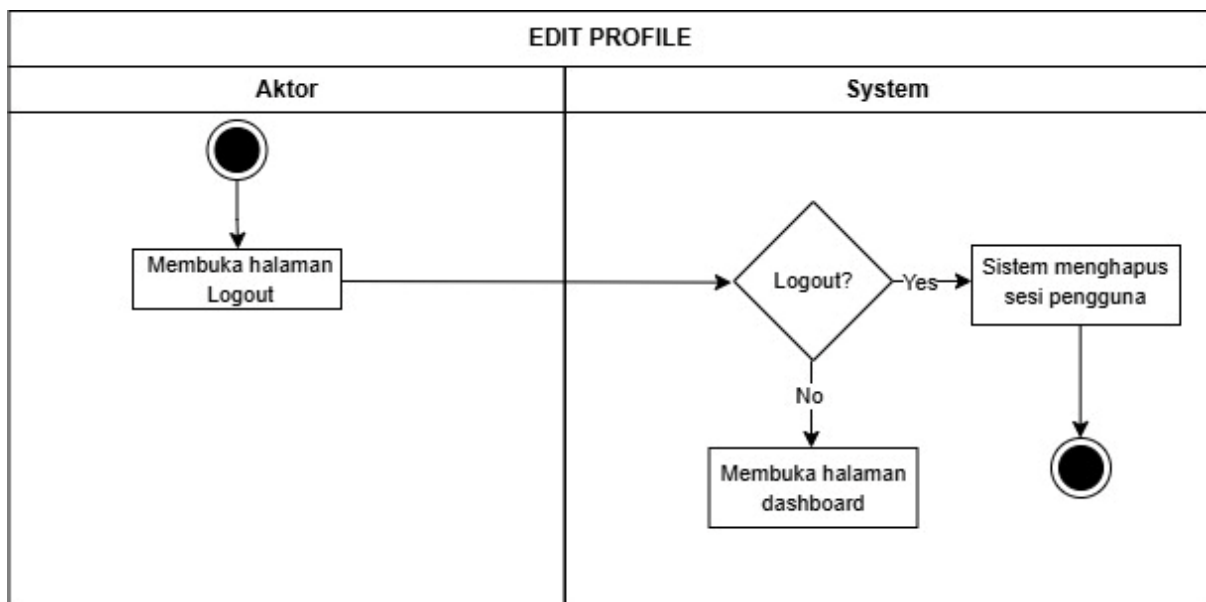
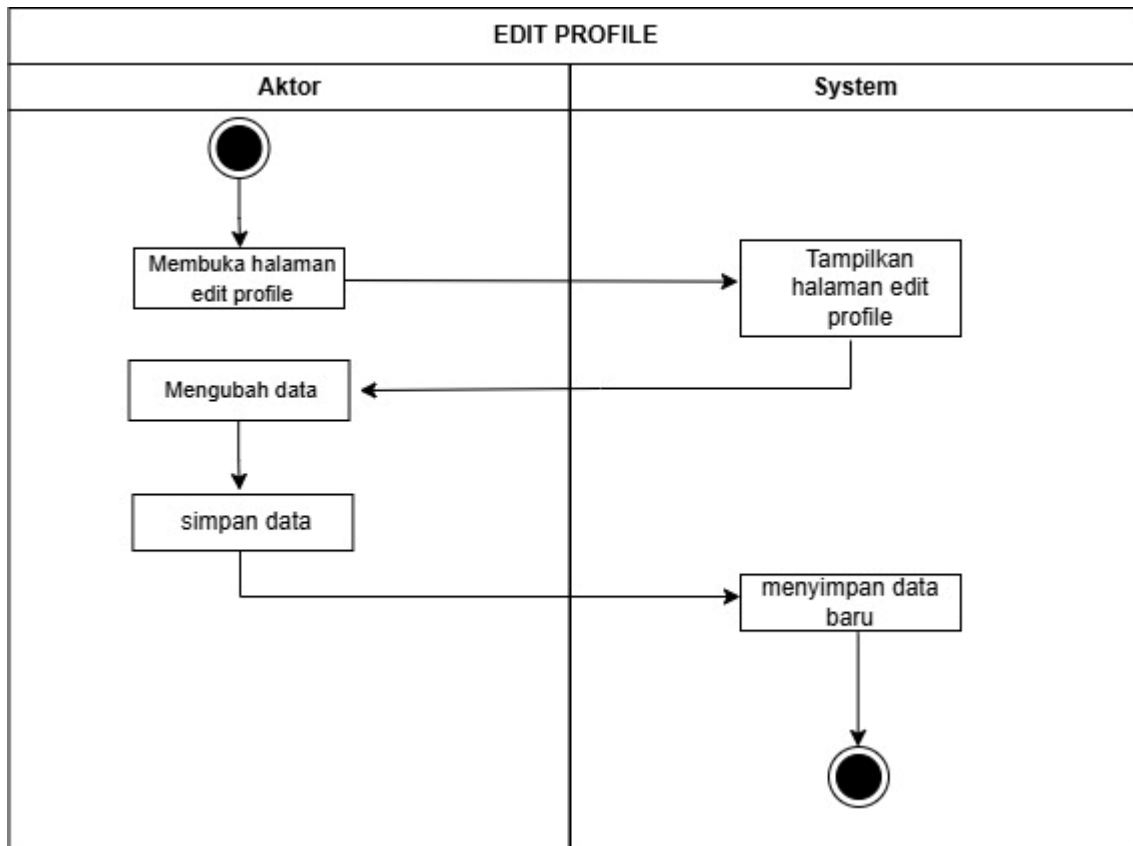
- Diagram Activity











A. Spesifikasi Database

- Tabel User dan Admin

	id	full_name	nip_nim_id	phone	role	password
e	1	Anggy cantik	2022071028	087652678937	user	\$2y\$12
e	3	Admin	01100	082178372219	admin	\$2y\$12
e	4	orang1	2022071025	0865672863	user	\$2y\$12

Selected: Edit Copy Delete Export

Rows: 25 Filter rows: Search this table Sort

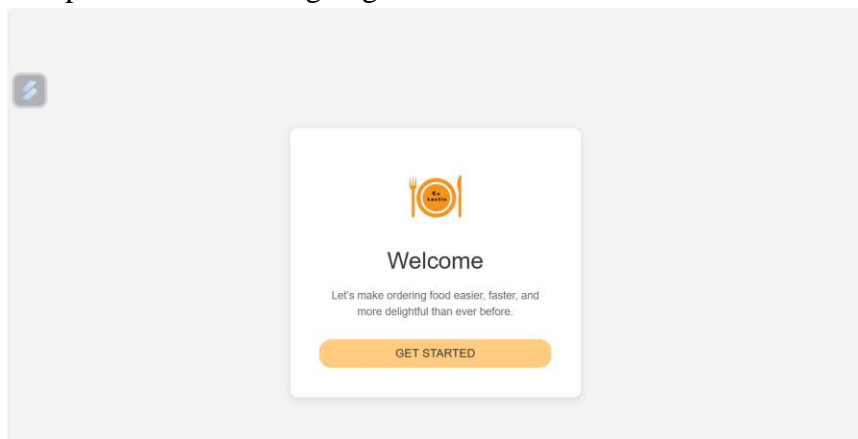
- Tabel Menu

id	name	price	image_path	created_at	updated_at
6	pizza mini	5.0	img/1736390067_pizza-mini.jpg	2025-01-08 19:34:27	2025-01-08 19:34:27
7	ayam geprek	17.0	img/1736390139_geprek.png	2025-01-08 19:35:39	2025-01-08 19:35:39
8	Dimsum	17.0	img/1736391031_dimsum.png	2025-01-08 19:50:31	2025-01-08 19:50:31
9	Sop buah	10.0	img/1736391050_sop.jpg	2025-01-08 19:50:50	2025-01-08 19:50:50
10	Soto ayam	20.0	img/1736391066_soto.jpeg	2025-01-08 19:51:06	2025-01-08 19:51:06
11	Teh manis dingin	5.0	img/1736391247_esteh.jpeg	2025-01-08 19:54:07	2025-01-08 19:54:07
12	Teh manis tawar	3.0	img/1736391271_esteh.jpeg	2025-01-08 19:54:31	2025-01-08 19:54:31
13	Roti bakar	15.0	img/1736391497_roti bakar.png	2025-01-08 19:58:17	2025-01-08 19:58:17
14	mie goreng aceh	25.0	img/1736405318_mie aceh.png	2025-01-08 23:48:38	2025-01-08 23:48:38

5. Implementasi

a) Tampilan Antarmuka Aplikasi

1. Tampilan Awal/Landing Page.



Penjelasan: Halaman landing page dari website "Go Kantin" dirancang sebagai halaman utama yang menyambut pengguna dengan informasi singkat dan jelas tentang layanan pembelian makanan di kantin. Elemen utama halaman ini meliputi:

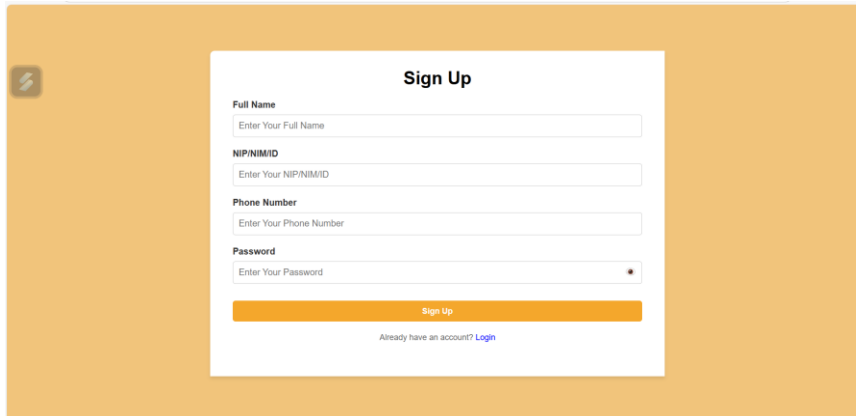
- **Hero Section**

Bagian utama yang menyambut pengguna dengan judul besar "Welcome" disertai deskripsi singkat tentang misi layanan, yaitu membuat pemesanan makanan menjadi lebih mudah, cepat, dan menyenangkan. Terdapat tombol aksi "GET STARTED" yang mengarahkan pengguna untuk memulai eksplorasi layanan.

- **Visual dan Brand**

Dihiasi elemen grafis sederhana dengan ikon piring, garpu, dan sendok yang melambangkan layanan terkait makanan. Desain minimalis dan modern memberikan kesan profesional, dengan dominasi warna oranye untuk menciptakan daya tarik visual serta membangun identitas merek.

2. Sign Up User.

A screenshot of a 'Sign Up' form on a cream-colored background. The form is white with a thin orange border. It features four input fields: 'Full Name' (placeholder: Enter Your Full Name), 'NIP/NIM/ID' (placeholder: Enter Your NIP/NIM/ID), 'Phone Number' (placeholder: Enter Your Phone Number), and 'Password' (placeholder: Enter Your Password, with a toggle icon). Below the fields is an orange 'Sign Up' button. At the bottom, there is a link: 'Already have an account? [Login](#)'.

Penjelasan:

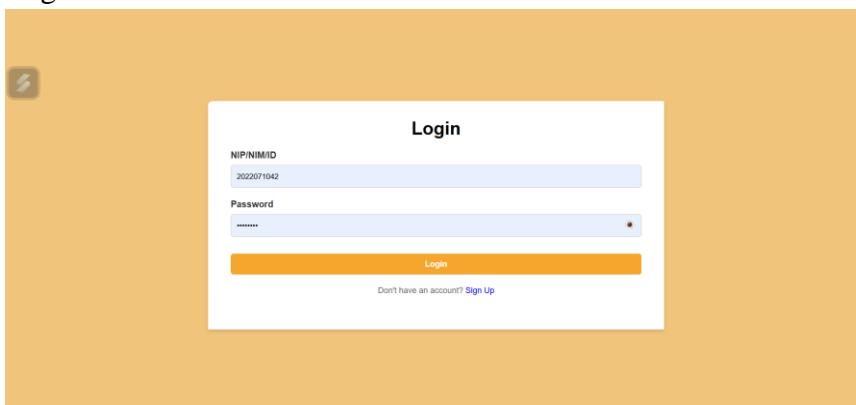
- **Halaman Signup**

Halaman signup dirancang untuk menginformasikan bahwa ini adalah tempat untuk mendaftarkan akun. Bagian utamanya menampilkan judul besar "Signup" yang memberi penekanan pada tujuan halaman ini. Formulir singkat disediakan untuk mengumpulkan informasi dasar seperti full name, NIP/NIM/ID, phone number, dan password. Tombol aksi "Signup" disertakan untuk memudahkan pengguna menyelesaikan proses pendaftaran akun dengan cepat dan efisien.

- **Visual dan Brand**

Halaman ini dihiasi dengan elemen grafis sederhana, seperti ikon pengguna atau desain minimalis, yang menciptakan tampilan ramah dan mudah dipahami. Warna yang digunakan selaras dengan identitas visual merek, memberikan kesan profesional, menarik, dan nyaman bagi pengguna saat mengisi formulir.

3. Login.

A screenshot of a 'Login' form on a cream-colored background. The form is white with a thin orange border. It features two input fields: 'NIP/NIM/ID' (placeholder: 2022071042) and 'Password' (placeholder: *****, with a toggle icon). Below the fields is an orange 'Login' button. At the bottom, there is a link: 'Don't have an account? [Sign Up](#)'.

Penjelasan: Halaman **Login** didesain dengan tampilan sederhana, minimalis, dan fokus pada kemudahan pengguna untuk mengakses aplikasi. Dengan latar belakang berwarna krem

lembut, formulir login yang berada di tengah halaman menjadi fokus utama, memudahkan pengguna dalam mengisi informasi yang diperlukan.

Struktur Halaman

1. Judul Halaman:

- Teks "**Login**" terletak di bagian atas formulir dengan ukuran font yang jelas dan tegas, memberikan penanda utama bahwa ini adalah halaman login.

2. Input Data Pengguna:

- **NPM/NIM/ID**: Kolom input pertama untuk memasukkan identitas pengguna. Kolom ini memiliki placeholder bertuliskan "NPM/NIM/ID", memberikan panduan kepada pengguna tentang data yang harus dimasukkan.
- **Password**: Kolom input kedua untuk memasukkan kata sandi pengguna. Kolom ini dirancang dengan keamanan, di mana karakter yang dimasukkan otomatis disembunyikan.

3. Tombol Login:

- Tombol berwarna **oranye cerah** dengan teks "**Login**" di tengah, memberikan aksentuasi visual yang menarik sekaligus memudahkan pengguna untuk langsung mengidentifikasi fungsi tombol ini.

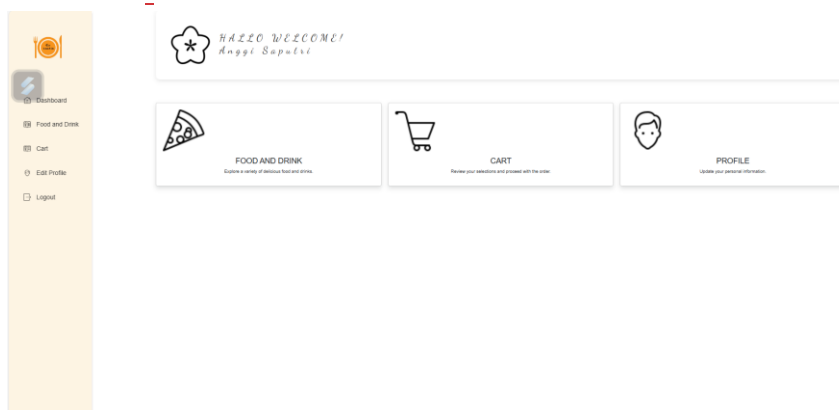
4. Tautan Sign Up:

- Di bagian bawah formulir terdapat teks "**Don't have an account? Sign U**". Tautan ini dirancang dengan warna biru, mengundang pengguna untuk mendaftar jika belum memiliki akun.

Desain Visual

- **Kontras Warna**: Latar belakang berwarna krem memberikan kontras yang lembut terhadap formulir login yang putih, menciptakan tampilan bersih dan profesional.
- **Tipografi**: Teks menggunakan font sederhana yang mudah dibaca, dengan hierarki yang jelas pada judul, placeholder, dan tombol.
- **Pemosisian Tengah**: Formulir ditempatkan secara sentral pada halaman untuk memastikan fokus langsung pada proses login.

4. Dashboard



Penjelasan:

Halaman Dashboard

Pada halaman dashboard, terdapat **sidebar** yang berisi menu navigasi utama dengan masing-masing dilengkapi **ikon yang relevan** untuk memudahkan identifikasi fungsi setiap menu. Menu pada sidebar meliputi:

- **Dashboard:** Ikon berupa rumah (home) sebagai tanda halaman utama.
- **Cart:** Ikon keranjang belanja untuk meninjau pesanan.
- **Food and Drink:** Ikon piring dan sendok garpu untuk memilih makanan dan minuman.
- **Edit Profile:** Ikon pengguna (user) untuk memperbarui informasi profil.
- **Logout:** Ikon pintu keluar (logout) untuk mengakhiri sesi pengguna.

Di bagian atas halaman, terdapat sapaan pembuka dengan pesan "**Hello, Welcome**", diikuti dengan nama pengguna, memberikan kesan personal dan ramah.

Menu Utama Halaman Halaman utama menampilkan tiga menu utama dengan desain yang lebih besar dan jelas:

- **Food and Drink:** Menampilkan daftar makanan dan minuman yang tersedia untuk dipilih.
- **Cart:** Memungkinkan pengguna untuk memeriksa dan mengelola pesanan sebelum checkout.
- **Profile:** Menyediakan akses cepat ke informasi pribadi pengguna.

Tampilan Visual dan Fungsionalitas

Tiap menu dan sidebar dirancang dengan elemen visual seperti **ikon, warna yang konsisten, dan tipografi yang bersih**. Sidebar tetap terlihat di sisi kiri layar sehingga pengguna dapat dengan mudah berpindah antar halaman tanpa kesulitan.

- **Ikon pada Sidebar**

Memberikan kemudahan identifikasi fungsi menu tanpa perlu membaca teks secara detail.

- **Responsivitas**

Desain antarmuka yang fleksibel untuk memastikan navigasi tetap nyaman di berbagai perangkat.

Desain ini dibuat dengan fokus pada **kemudahan navigasi** dan **tampilan profesional** sehingga memberikan pengalaman yang menyenangkan bagi pengguna saat menjelajahi aplikasi.

5. Product Page (Food and Drink).



Penjelasan: Pada halaman **Food and Drink**, tampilan antarmuka didesain untuk memberikan pengalaman yang menyenangkan dalam memilih berbagai jenis makanan dan minuman. Sidebar tetap ada, seperti pada halaman **Dashboard**, dengan menu navigasi yang memudahkan pengguna untuk berpindah antar halaman. Sidebar ini terdiri dari:

- **Dashboard:** Ikon rumah (home) yang mengarah kembali ke halaman utama.
- **Cart:** Ikon keranjang belanja untuk melihat dan mengelola pesanan.
- **Food and Drink:** Ikon piring dan sendok garpu sebagai menu utama untuk memilih makanan dan minuman.
- **Edit Profile:** Ikon pengguna (user) untuk memperbarui profil pengguna.
- **Logout:** Ikon pintu keluar untuk keluar dari aplikasi.

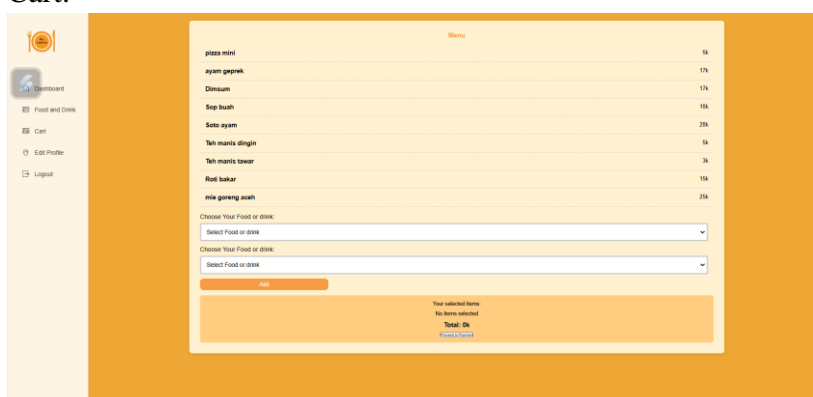
Tampilan Halaman

- **Food and Drink:** menampilkan berbagai pilihan makanan dan minuman dalam bentuk daftar yang terstruktur dengan baik. Setiap item disertai dengan **gambar** yang jelas, **nama menu**, dan **harga per item** yang tertera dengan jelas di bawah deskripsi makanan atau minuman. Ini memudahkan pengguna dalam membuat pilihan.
- **Gambar:** Setiap makanan dan minuman dilengkapi dengan gambar visual yang menggugah selera, memungkinkan pengguna untuk melihat dengan jelas tampilan menu.
- **Nama Menu:** Nama dari setiap item makanan atau minuman tertera dengan jelas di samping gambar.
- **Harga:** Setiap menu juga dilengkapi dengan harga per item yang terlihat jelas untuk memudahkan pengguna dalam memilih sesuai anggaran.

Desain Visual dan Responsivitas

- **Tampilan Bersih dan Minimalis:** Setiap elemen desain mengutamakan kejelasan dan kemudahan navigasi. Warna yang konsisten dan tipografi yang bersih digunakan untuk menjaga tampilan profesional dan mudah dibaca.
- **Responsivitas:** Desain halaman ini sepenuhnya responsif, sehingga tampilan akan tetap nyaman untuk digunakan di berbagai perangkat, baik desktop, tablet, maupun ponsel.
- **Ikon Menu:** Ikon yang digunakan di sidebar dan menu utama memberikan visual yang jelas, membantu pengguna untuk mengenali fungsi dengan cepat tanpa harus membaca teks panjang.

6. Cart.



Penjelasan: Halaman **Cart** adalah tempat pengguna dapat melihat, menambahkan, dan mengelola pesanan mereka. Berikut adalah penjelasan detail dari elemen-elemen pada halaman ini:

- **Tampilan Sidebar**

Sidebar pada sisi kiri halaman tetap konsisten dengan menu navigasi utama, yaitu:

1. Dashboard
2. Food and Drink
3. Cart
4. Edit Profile
5. Logout

Ikon dan label pada sidebar membantu pengguna berpindah antar halaman dengan mudah.

- **Daftar Menu Makanan dan Minuman**

Bagian utama halaman menampilkan daftar menu makanan dan minuman lengkap dengan nama dan harga setiap item. Contoh menu:

- Pizza Mini – 9k
- Ayam Geprek – 17k
- Dimsum – 17k
- Soto Ayam – 20k
- Dan lainnya.

Harga ditampilkan secara jelas dalam satuan ribu rupiah (k) untuk memudahkan pengguna memahami informasi harga.

- **Form Pilihan Pesanan**

Di bawah daftar menu, terdapat formulir untuk menambahkan item ke dalam pesanan:

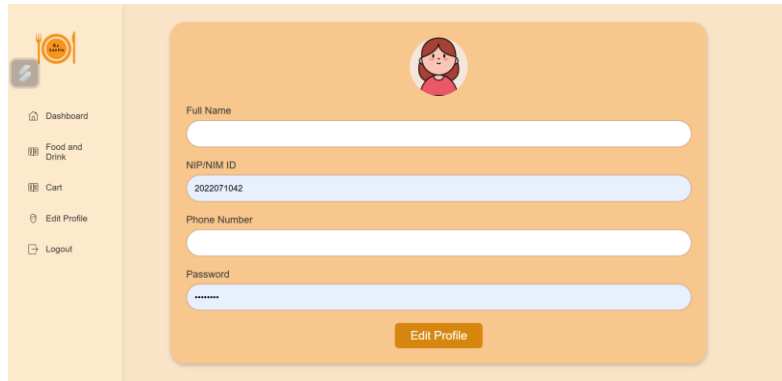
1. **Dropdown Menu**
 - Pengguna dapat memilih makanan atau minuman dari daftar yang sudah tersedia melalui dropdown menu.
2. **Tombol Add**
 - Tombol ini digunakan untuk menambahkan item yang dipilih kedalam daftar pesanan.
3. **Ringkasan Pesanan**

Bagian bawah halaman menampilkan ringkasan pesanan yang dipilih pengguna, termasuk:

- **Your Selected Items:** Menampilkan daftar makanan/minuman yang telah dipilih.
- **Total:** Menghitung total harga semua item yang dipilih.
- **Proses Checkout:** Tombol untuk melanjutkan proses pembayaran setelah semua pesanan selesai dipilih.
- **Tampilan Visual**
- **Warna Dominan:** Menggunakan warna oranye dan kuning yang memberikan nuansa hangat dan menggugah selera.
- **Desain Minimalis:** Elemen desain dibuat sederhana agar pengguna dapat fokus pada fungsi utama halaman.

Halaman ini dirancang untuk memastikan pengguna dapat mengelola pesanan mereka dengan nyaman dan mudah, mulai dari memilih makanan/minuman hingga melanjutkan ke proses pembayaran.

7. Edit Profile.



Penjelasan: Halaman **Edit Profile** dirancang untuk memungkinkan pengguna memperbarui informasi pribadi mereka dengan mudah. Berikut adalah penjelasan elemen-elemen pada halaman ini:

1. Tampilan Sidebar

Sidebar di sisi kiri halaman tetap konsisten dengan navigasi di seluruh aplikasi, memuat menu:

- Dashboard
- Food and Drink
- Cart
- Edit Profile
- Logout
- Formulir Edit Profile

Di bagian utama halaman, terdapat formulir sederhana yang memungkinkan pengguna memperbarui data mereka. Formulir ini terdiri dari:

1. **Full Name:** Kolom untuk mengubah nama lengkap pengguna.
2. **NIP/NIM/ID:** Kolom untuk memperbarui nomor identitas pengguna (NIP, NIM, atau ID).
3. **Phone Number:** Kolom untuk memperbarui nomor telepon pengguna.
4. **Password:** Kolom untuk memperbarui kata sandi akun.
5. **Tombol "Edit Profile":** Tombol aksi untuk menyimpan perubahan yang dilakukan pengguna pada data mereka.

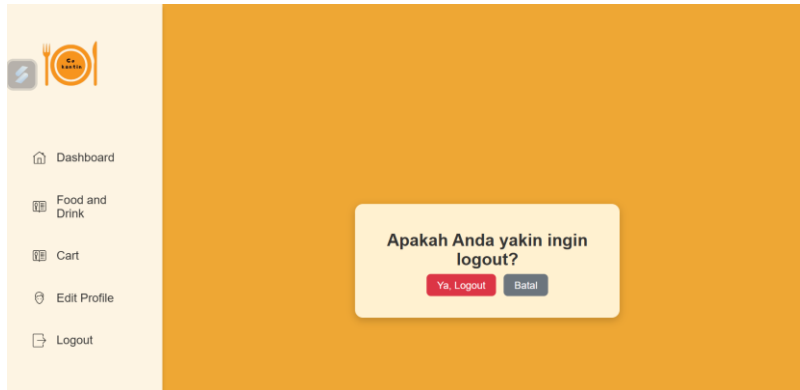
1. Visual dan Desain

- **Ilustrasi Profil:** Di bagian atas formulir terdapat ikon ilustrasi pengguna yang memberikan kesan personal.
- **Desain Minimalis:** Menggunakan elemen visual sederhana dengan dominasi warna oranye lembut, menciptakan tampilan yang bersih dan ramah.
- **Tipografi Jelas:** Font yang digunakan memastikan setiap teks mudah dibaca.
- **Fungsi dan Kegunaan**
 - Halaman ini memudahkan pengguna untuk memperbarui informasi pribadi mereka kapan saja.

- Dengan desain yang intuitif dan kolom-kolom yang jelas, pengguna dapat dengan cepat memahami dan menggunakan fitur ini tanpa hambatan.

Halaman **Edit Profile** tidak hanya berfungsi untuk memperbarui data tetapi juga memberikan pengalaman pengguna yang nyaman dan profesional.

8. Logout.



Penjelasan: Halaman **Logout** dirancang untuk memberikan proses keluar dari aplikasi dengan cepat dan mudah, sekaligus memastikan bahwa pengguna menyadari tindakan yang sedang dilakukan. Antarmuka pada halaman ini dibuat sederhana namun efektif, dengan elemen visual yang jelas.

Struktur Halaman

1. Pesan Konfirmasi:

- Teks utama berupa pesan konfirmasi, misalnya "**Are you sure you want to logout?**", tampil di bagian tengah halaman. Pesan ini memberi pengguna kesempatan untuk memastikan sebelum keluar dari akun mereka.

2. Tombol Aksi:

- **Logout**: Tombol dengan warna mencolok (seperti merah atau oranye), memberikan penanda yang jelas untuk keluar dari aplikasi. Tombol ini biasanya disertai ikon pintu keluar untuk memudahkan identifikasi.
- **Cancel**: Tombol alternatif dengan warna netral (seperti abu-abu atau putih), memungkinkan pengguna untuk membatalkan proses logout dan kembali ke halaman sebelumnya.

3. Tautan Opsional:

- Di bagian bawah, bisa ditambahkan tautan "**Back to Dashboard**" untuk memberi opsi tambahan jika pengguna berubah pikiran dan ingin langsung kembali ke halaman utama.

b) Kode Aplikasi

- Penjelasan program yang dibuat spt Controller, Model, dan View

```

app > Models > Menu.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Menu extends Model
9  {
10     use HasFactory;
11
12     protected $fillable = ['name', 'price', 'image_path'];
13 }
14

```

Penjelasan: Model Menu digunakan untuk merepresentasikan data menu dalam database dan mengatur hubungan antara aplikasi Laravel dengan tabel database yang sesuai. Model ini menggunakan namespace App\Models dan dilengkapi dengan trait HasFactory, yang memungkinkan pembuatan data dummy menggunakan factory. Properti \$fillable pada model ini mendefinisikan kolom-kolom yang dapat diisi secara massal, yaitu name, price, dan image_path. Dengan model ini, berbagai operasi CRUD (Create, Read, Update, Delete) dapat dilakukan secara efisien dan terstruktur sesuai standar Laravel Eloquent ORM.

```

class User extends Authenticatable
{
    use HasApiTokens, HasFactory, Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */
    protected $fillable = [
        'full_name',
        'nip_nim_id',
        'phone',
        'password',
        'role'
    ];
}

```

Penjelasan: Model User digunakan untuk merepresentasikan data pengguna dalam database dan mengatur hubungan antara aplikasi Laravel dengan tabel yang sesuai. Model ini mewarisi class Authenticatable dan menggunakan beberapa trait, seperti HasApiTokens, HasFactory, dan Notifiable, yang mendukung pengelolaan token API, pembuatan data dummy, serta pengiriman notifikasi. Properti \$fillable mendefinisikan kolom-kolom yang dapat diisi secara massal, yaitu full_name, nip_nim_id, phone, password, dan role. Dengan model ini, operasi CRUD terhadap data pengguna dapat dilakukan secara aman dan terstruktur, sesuai standar Laravel Eloquent ORM.

```

class AdminController extends Controller
{
    public function dashboard()
    {
        return view('admin.dashboard');
    }

    public function cardadmin()
    {
        $items = Menu::all();
        return view('admin.cartadmin', compact('items'))
    }
}

```

Penjelasan: Class AdminController pada gambar adalah controller dalam Laravel yang digunakan untuk mengelola logika terkait halaman admin. Controller ini memiliki dua method utama, yaitu dashboard() yang mengembalikan view admin.dashboard sebagai halaman utama dashboard admin, dan cardadmin() yang mengambil semua data dari model Menu menggunakan Menu::all() lalu mengirimkan data tersebut ke view admin.cartadmin melalui fungsi compact. Dengan controller ini, pengelolaan logika dan tampilan halaman admin dapat dilakukan secara terstruktur sesuai dengan standar MVC Laravel.

```

class AuthController extends Controller
{
    public function index() {
        return view('signup');
    }
    public function viewLogin() {
        return view('login');
    }
    public function edit() {
        return view('user.editprofile');
    }
    public function logout() {
        return view('logout');
    }
    public function signup(Request $request)
    {
        // Validasi input
        $request->validate([
            'full_name' => 'required|string|max:255',
            'nip_nim_id' => 'required|string|unique:users,nip_nim_id',
            'phone' => 'required|string|unique:users,phone',
            'password' => 'required|string|min:8',
        ]);

        // Simpan data ke database
        User::create([
            'full_name' => $request->full_name,
            'nip_nim_id' => $request->nip_nim_id,
            'phone' => $request->phone,
        ]);
    }
}

```

Penjelasan: Class AuthController pada kode digunakan untuk menangani logika autentikasi dalam aplikasi Laravel. Controller ini memiliki beberapa method, di antaranya index() yang mengembalikan view signup sebagai halaman pendaftaran, viewLogin() untuk menampilkan halaman login, edit() untuk mengarahkan ke halaman edit profil pengguna, dan logout() untuk mengakses halaman logout. Selain itu, terdapat method signup() yang bertugas memproses

pendaftaran pengguna baru. Method ini memvalidasi input seperti full_name, nip_nim_id, phone, dan password, serta menyimpan data pengguna ke database melalui model User. Dengan controller ini, proses autentikasi dan pengelolaan pengguna menjadi lebih terstruktur

```
class FoodController extends Controller
{
    public function food(){
        $foodItems = Menu::all();
        return view('user.food',compact('foodItems'));
    }
}
```

Penjelasan: Class FoodController pada kode digunakan untuk menangani logika terkait data makanan dalam aplikasi Laravel. Controller ini memiliki method food() yang mengambil semua data dari model Menu menggunakan Menu::all() dan menyimpannya dalam variabel \$foodItems. Data tersebut kemudian dikirim ke view user.food menggunakan fungsi compact, sehingga dapat diakses dalam view.

```
class MenuController extends Controller
{
    // Menampilkan form untuk membuat menu baru
    public function create()
    {
        return view('menu.create'); // Pastikan view ini ada
    }

    // Menyimpan menu baru

    public function store(Request $request)
    {
        $request->validate([
            'name' => 'required|string|max:255',
            'price' => 'required|numeric',
            'image' => 'required|image|mimes:jpeg,png,jpg,gif|max:2048',
        ]);

        // Upload gambar ke folder public/img
        $imagePath = null;
        if ($request->hasFile('image')) {
            $file = $request->file('image');
            $imageName = time() . '_' . $file->getClientOriginalName();
            $file->move(public_path('img'), $imageName);
            $imagePath = 'img/' . $imageName;
        }

        // Simpan data ke database
        $menu = Menu::create([
            'name' => $request->name,
            'price' => $request->price,
            'image_path' => $imagePath,
        ]);
    }
}
```

Penjelasan: Class MenuController pada kode digunakan untuk mengelola logika terkait pembuatan menu baru dalam aplikasi Laravel. Controller ini memiliki dua method utama. Method create() digunakan untuk menampilkan form pembuatan menu baru melalui view menu.create. Method store() bertanggung jawab untuk menyimpan menu baru ke database.

Pada method ini, input dari pengguna divalidasi, termasuk validasi nama, harga, dan gambar. Jika gambar diunggah, file tersebut diproses dan disimpan ke folder public/img, lalu path-nya disimpan dalam database bersama data lainnya seperti nama dan harga menu.

```
class UserController extends Controller
{
    public function dashboard()
    {
        return view('user.dashboard');
    }
    public function card()
    {
        $foodItems = Menu::all();
        return view('user.cartuser', compact('foodItems'));
    }

    public function update(Request $request, $id)
    {
        $request->validate([
            'full_name' => 'required|string|max:255',
            'nip_nim_id' => 'required|string|unique:users,nip_nim_id,' . $id,
            'phone' => 'required|string|max:15|unique:users,phone,' . $id,
            'password' => 'nullable|string|min:6',
        ]);

        $user = User::findOrFail($id);

        $user->full_name = $request->input('full_name');
        $user->nip_nim_id = $request->input('nip_nim_id');
        $user->phone = $request->input('phone');

        if ($request->filled('password')) {
            $user->password = Hash::make($request->input('password'));
        }

        $user->save();

        return response()->json([
            'success' => true,
            'message' => 'User profile updated successfully.'
        ]);
    }
}
```

Penjelasan: Class UserController pada kode digunakan untuk mengelola logika terkait dengan pengelolaan profil pengguna dalam aplikasi Laravel. Controller ini memiliki tiga method utama. Method dashboard() digunakan untuk menampilkan halaman dasbor pengguna melalui view user.dashboard. Method card() bertanggung jawab untuk mengambil semua data item menu menggunakan Menu::all() dan mengirimkannya ke view user.cartuser. Sedangkan method update() digunakan untuk memperbarui data profil pengguna. Di dalamnya, data yang dikirimkan divalidasi terlebih dahulu, seperti memastikan nip_nim_id dan phone unik, serta password yang diperbarui harus memiliki panjang minimal 6 karakter. Setelah validasi, data pengguna diperbarui, dan jika ada perubahan password, password baru akan dienkripsi sebelum disimpan ke database.

```

resources > views > layout > sidebar.blade.php
1  <div class="col-md-3 col-lg-2 d-md-block bg-light sidebar position-fixed h-100">
2    <div class="logo p-3">
3      
4    </div>
5    <ul class="nav flex-column">
6      <li class="nav-item"><a class="nav-link" href="{{ route('dashboard') }}">
8      <li class="nav-item">
9        @if(auth()->check() && auth()->user()->role === 'user')
10          <a class="nav-link" href="{{ route('food') }}">
11            
12            Food and Drink
13          </a>
14        @endif
15      </li>
16      <li class="nav-item">
17        <a class="nav-link" href="{{ auth()->user()->role === 'admin' ? route('cardadmin') : route('card') }}">
18           Card
19        </a>
20      </li>
21
22      <li class="nav-item">
23        @if(auth()->check() && auth()->user()->role === 'user')
24          <a class="nav-link" href="{{ route('edit') }}">
25            
26            Edit Profile
27          </a>
28        @endif
29      </li>
30
31      <li class="nav-item"><a class="nav-link" href="{{ route('logout') }}"> Log
32      </li>
33    </ul>
34    <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display: inline;">
35      @csrf
36      <button type="submit" style="background: none; border: none; color: #333; font-size: 1.4rem; cursor: pointer; padding: 15px; border-ra
37    </form>
38  </div>
39
40  <li class="nav-item"><a class="nav-link" href="{{ route('logout') }}"> Log
41  </li>
42  <!-- <li>
43    <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display: inline;">
44      @csrf
45      <button type="submit" style="background: none; border: none; color: #333; font-size: 1.4rem; cursor: pointer; padding: 15px; border-ra
46      
47      Logout
48    </button>
49    </form>
50  </li> -->
51 </ul>
52 </div>
53 </div>

```

Penjelasan: Ini adalah kode layout/app dari web Go Kantin

```

<body>
  @include('layout.sidebar')
  <div class="main-content">
    <!-- Menu Container -->
    <div class="menu-container" id="menuContainer">
      <h2>Menu</h2>
      @foreach ($foodItems as $item)
        <div class="menu-item">
          <h3>{{ $item->name }}</h3>
          <p>{{ number_format($item->price, 0, ',', '.') }}k</p>
        </div>
      @endforeach
      <div class="form-container">
        <label for="food">Choose Your Food or drink:</label>
        <select id="food1">
          <option value="">Select Food or drink</option>
          @foreach ($foodItems as $item)
            <option value="{{ $item->name }}">{{ $item->name }}</option>
          @endforeach
        </select>
        <label for="food">Choose Your Food or drink:</label>
        <select id="food2">
          <option value="">Select Food or drink</option>
          @foreach ($foodItems as $item)
            <option value="{{ $item->name }}">{{ $item->name }}</option>
          @endforeach
        </select>
        <button onclick="addToCart()">Add</button>
      </div>
    </div>
  </div>

```

Penjelasan: Ini adalah penggalan kode dari view cartuser.

```

<body>
  <div class="container-fluid">
    <div class="row">
      <!-- Sidebar -->
      @include('layout.sidebar')

      <!-- Main Content -->
      <main class="col-md-9 ms-sm-auto col-lg-10 px-md-4">
        <div class="welcome d-flex align-items-center">
          
          <h2>HALLO WELCOME!<br>{{ Auth::user()->full_name }}</h2>
        </div>
        <div class="row g-4">
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>FOOD AND DRINK</h3>
              <p>Explore a variety of delicious food and drinks.</p>
            </div>
          </div>
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>CART</h3>
              <p>Review your selections and proceed with the order.</p>
            </div>
          </div>
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>PROFILE</h3>
              <p>Update your personal information.</p>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

Penjelasan: Ini adalah penggalan kode dari view dashboard.

```

<body>
  @include('layout.sidebar')

  <div class="main-content">
    <div class="profile-card">
      
      <form id="user-form">
        <div class="form-group">
          <label for="full-name">Full Name</label>
          <input type="text" id="full-name" name="full_name">
          <div class="error-message" id="full-name-error"></div>
        </div>
        <div class="form-group">
          <label for="nip-nim-id">NIP/NIM ID</label>
          <input type="text" id="nip-nim-id" name="nip_nim_id">
          <div class="error-message" id="nip-nim-id-error"></div>
        </div>
        <div class="form-group">
          <label for="phone-number">Phone Number</label>
          <input type="text" id="phone-number" name="phone">
          <div class="error-message" id="phone-number-error"></div>
        </div>
        <div class="form-group">
          <label for="password">Password</label>
          <input type="password" id="password" name="password">
          <div class="error-message" id="password-error"></div>
        </div>
        <button type="submit" class="btn-save">Edit Profile</button>
      </form>
    </div>
  </div>

```

Penjelasan: Ini adalah penggalan kode dari view edit profile.

```

<div class="main-content">
  <div class="menu-section">
    <h2>Makanan</h2>
    <div class="menu-container">
      <div class="menu-item">
        
        <h3>Ayam Geprek</h3>
        <p>15k</p>
      </div>
      <div class="menu-item">
        
        <h3>Soto Ayam</h3>
        <p>15k</p>
      </div>
      <div class="menu-item">
        
        <h3>Dimsum Mentai</h3>
        <p>15k</p>
      </div>
    </div>
  </div>
  <div class="menu-section">
    <h2>Minuman</h2>
    <div class="menu-container">
      <div class="menu-item">
        
        <h3>Esteh Manis</h3>
        <p>10k</p>
      </div>
      <div class="menu-item">
        
        <h3>Es Teh Tawar</h3>
        <p>5k</p>
      </div>
    </div>
  </div>
</div>

```

Penjelasan: Ini adalah penggalan kode dari view drink and food.

```

    .btn:hover {
      background-color: #ffb74d;
    }
  }
</style>
</head>

<body>
  <div class="container d-flex justify-content-center align-items-center">
    <div class="card">
      <div class="logo">
        
      </div>
      <h1>Welcome</h1>
      <p>Let's make ordering food easier, faster, and more delightful than ever before.</p>
      <button class="btn" onclick="redirectToSignup()">GET STARTED</button>
    </div>
  </div>

  <script>
    function redirectToSignup() {
      window.location.href = "{{ route('signup') }}";
    }
  </script>
  <!-- Bootstrap JS -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>

</html>

```

Penjelasan: Ini adalah penggalan kode dari view landing page.

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
  <link rel="stylesheet" href="style.css">
</head>

<body>
  <div class="container">
    <div class="card">
      <h1>Login</h1>
      <form class="login-form" id="login-form" action="{{ route('login') }}" method="POST">
        @csrf
        <div class="form-group">
          <label for="nip">NIP/NIM/ID</label>
          <input type="text" id="nip" name="nip" placeholder="Enter Your NIP/NIM/ID" required>
        </div>
        <div class="password-container">
          <label for="password">Password</label>
          <input type="password" id="password-field" name="password" placeholder="Enter Your Password" required>
          <span id="toggle-password" class="toggle-password">👁️</span>
        </div>
        <button type="submit">Login</button>
        <p>Don't have an account? <a href="{{ route('signup') }}">Sign Up</a></p>
      </form>
    </div>
  </div>
  <script src="script.js"></script>
</body>

```

Penjelasan: Ini adalah penggalan kode dari view login.

```

<body>
  @include('layout.sidebar')

  <div class="main-content">
    <div class="menu-section">
      <h2>Makanan</h2>
      <div class="menu-container">
        <div class="menu-item">
          
          <h3>Ayam Geprek</h3>
          <p>15k</p>
        </div>
        <div class="menu-item">
          
          <h3>Soto Ayam</h3>
          <p>15k</p>
        </div>
        <div class="menu-item">
          
          <h3>Dimsum Mentai</h3>
          <p>15k</p>
        </div>
      </div>
    </div>
    <div class="menu-section">
      <h2>Minuman</h2>
      <div class="menu-container">
        <div class="menu-item">
          
          <h3>Esteh Manis</h3>
          <p>10k</p>
        </div>
        <div class="menu-item">

```

Penjelasan: Ini adalah penggalan kode dari view logout.

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Sign Up</title>
  <link rel="stylesheet" href="style.css">
</head>

<body>
  <div class="container">
    <div class="card">
      @if (session('success'))
      <div class="alert alert-success">
        {{ session('success') }}
      </div>
    @endif

    <h1>Sign Up</h1>
    <form class="signup-form" action="{{ route('signup') }}" method="POST">
      @csrf

      <label for="full_name">Full Name</label>
      <input type="text" id="full_name" name="full_name" placeholder="Enter Your Full Name" required>

      <label for="nip_nim_id">NIP/NIM/ID</label>
      <input type="text" id="nip_nim_id" name="nip_nim_id" placeholder="Enter Your NIP/NIM/ID" required>

      <label for="phone">Phone Number</label>
      <input type="text" id="phone" name="phone" placeholder="Enter Your Phone Number" required>

      <div class="password-container">
        <label for="password">Password</label>
        <input type="password" id="password" name="password" placeholder="Enter Your Password" required>
        <span class="toggle-password">👁️</span>
      </div>

      <button type="submit">Sign Up</button>
    </form>
  </div>
</body>

```

Penjelasan: Ini adalah penggalan kode dari view signup.

```

<body>
  @include('layout.sidebar')
  <div class="main-content">

    <!-- Menu Container -->
    <div class="menu-container" id="menuContainer">
      <h2>Menu</h2>
      @foreach ($items as $menu)
        <div class="menu-item">
          <span>{{ $menu->name }}</span>
          <span>{{ $menu->price }}k</span>
        </div>
      @endforeach
    </div>

    <!-- Add New Menu Item Section -->
    <div class="add-menu-container" id="addMenuContainer">
      <h1>Welcome Admin, {{ Auth::user()->full_name }}!</h1>
      <h3>Add New Menu Item</h3>
      <form id="addMenuForm" enctype="multipart/form-data">
        @csrf
        <label for="newMenuName">Menu Name:</label>
        <input type="text" id="newMenuName" name="name" placeholder="Enter menu item name" required>

        <label for="newMenuPrice">Price:</label>
        <input type="number" id="newMenuPrice" name="price" placeholder="Enter menu price" required>

        <label for="newMenuImage">Upload Image:</label>
        <input type="file" id="newMenuImage" name="image" accept="image/*" required>

        <button type="button" id="submitMenu">Add Menu Item</button>
      </form>
    </div>
  </div>

```

```

<body>
  <div class="container-fluid">
    <div class="row">
      @include("layout.sidebar")
      <!-- Sidebar -->

      <!-- Main Content -->
      <main class="col-md-9 ms-sm-auto col-lg-10 px-md-4">
        <div class="welcome d-flex align-items-center">
          
          <h2>HALLO WELCOME!<br>{{ Auth::user()->full_name }}</h2>
        </div>
        <div class="row g-4">
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>FOOD AND DRINK</h3>
              <p>Explore a variety of delicious food and drinks.</p>
            </div>
          </div>
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>CART</h3>
              <p>Review your selections and proceed with the order.</p>
            </div>
          </div>
          <div class="col-md-4">
            <div class="card text-center p-4 shadow">
              
              <h3>PROFILE</h3>
              <p>Update your personal information.</p>
            </div>
          </div>
        </div>
      </main>
    </div>
  </div>

```

Penjelasan: Ini adalah penggalan kode dari view admin.

c) Link Aplikasi

- Link Github / Gdrive: <https://github.com/Ellyzahardianty/GO-KANTIN-UAS-PEMWEB-.git>
- Link URL: [Go Kantin](#)