

# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Perkembangan teknologi informasi yang pesat telah mengubah perilaku konsumen dalam berbagai aspek kehidupan, termasuk dalam cara mereka memesan makanan dan produk kuliner. Di tengah persaingan bisnis yang semakin ketat, pelaku usaha di sektor kuliner seperti toko kue dituntut untuk mampu beradaptasi dan memanfaatkan teknologi digital agar tetap kompetitif dan dapat memenuhi kebutuhan pelanggan secara lebih cepat dan efisien.

Namun, kenyataannya masih banyak toko kue yang melakukan proses pemesanan secara manual, baik melalui pesan singkat, media sosial, maupun komunikasi langsung di toko. Metode ini sering menimbulkan sejumlah kendala, seperti pencatatan pesanan yang tidak terorganisir, keterlambatan dalam merespon pelanggan, kesalahan informasi produk, hingga keterbatasan dalam menjangkau pasar yang lebih luas. Kondisi ini tentunya dapat memengaruhi kepuasan pelanggan dan menghambat pertumbuhan bisnis.

Melihat kebutuhan tersebut, Ifartech hadir sebagai penyedia layanan teknologi informasi yang menawarkan solusi berupa pembuatan website khusus untuk pemesanan kue. Website ini dirancang tidak hanya sebagai etalase digital yang menampilkan produk-produk kue secara menarik, tetapi juga sebagai sistem yang memudahkan pelanggan dalam melakukan pemesanan, memilih tanggal pengambilan, hingga melakukan pembayaran secara online. Dengan demikian, toko kue dapat mengelola pesanan secara lebih terstruktur, mengurangi beban administrasi manual, serta meningkatkan efisiensi operasional.

Dalam tahap awal pengembangan proyek ini, tim Ifartech melakukan proses identifikasi kebutuhan melalui survei dan wawancara

langsung dengan mitra bento cake sebagai calon pengguna. Dari proses tersebut, diperoleh berbagai masukan dan permasalahan yang mereka hadapi dalam sistem pemesanan saat ini. Pemilik usaha ini pun menyambut baik inisiatif digitalisasi yang ditawarkan oleh Ifartech, karena mereka menyadari pentingnya inovasi teknologi dalam meningkatkan daya saing di era digital.

Dengan adanya kolaborasi ini, stakeholder utama proyek – yaitu mitra bento cake – terlibat secara aktif dalam proses pengembangan, mulai dari penyusunan fitur, tampilan website, hingga uji coba sistem. Hal ini diharapkan dapat menghasilkan solusi yang benar-benar sesuai dengan kebutuhan pengguna, sekaligus mendukung transformasi digital di sektor usaha mikro dan kecil secara lebih luas.

## **B. Project Charter**

### **1. Tujuan**

Tujuan dari proyek ini adalah untuk mengembangkan dan menghadirkan solusi digital yang inovatif dan berkualitas tinggi berupa website dan/atau aplikasi yang disesuaikan dengan kebutuhan klien. Proyek ini bertujuan untuk mendukung transformasi digital klien, meningkatkan efisiensi operasional, dan memperkuat kehadiran digital mereka di pasar.

### **2. Ruang Lingkup**

Ruang lingkup proyek meliputi:

- Analisis kebutuhan dan studi kelayakan proyek digital untuk klien.
- Desain UI/UX untuk platform web.
- Pengembangan front-end dan back-end sistem sesuai spesifikasi.

- Uji coba fungsionalitas dan keamanan sistem.
- Pelatihan dasar untuk pengguna akhir (user manual/training).
- Layanan pemeliharaan dan dukungan teknis pasca peluncuran untuk jangka waktu yang disepakati.

### 3. Stakeholder

Berikut adalah pihak-pihak yang terlibat dan berkepentingan dalam proyek ini:

- Klien : Penerima solusi digital, berperan sebagai pemberi kebutuhan dan penentu persetujuan akhir.
- Tim IFARTech Company:
  - *Project Manager* : Bertanggung jawab atas keseluruhan manajemen proyek.
  - *Business Analyst* : Menganalisis kebutuhan bisnis klien dan menyusunnya menjadi spesifikasi teknis.
  - *UI/UX Designer* : Mendesain antarmuka yang user-friendly dan menarik.
  - *Quality Assurance (QA)* : Melakukan sistem berdasarkan desain dan spesifikasi teknis.
  - *Web Developer* : Mengembangkan sistem berdasarkan desain dan spesifikasi teknis.

## **BAB II**

### **PENDAHULUAN**

#### **A. Analisis Kelayakan**

Pembuatan website penjualan cammeo merupakan project penting untuk meningkatkan visibilitas dan kredibilitas toko di mata pelanggan dan stakeholder. Website ini akan menjadi platform online informatif dan menarik untuk memperkenalkan Cammeo sebagai toko cake bento korea terpercaya dan berkualitas.

Analisis SWOT dilakukan untuk menilai kelayakan project berdasarkan kekuatan, kelemahan, peluang, dan ancaman yang mungkin dihadapi:

##### **1. Strength**

###### **a. Pemanfaatan Teknologi Digital**

Website dirancang sebagai solusi end-to-end untuk pemesanan kue, mulai dari katalog produk, pemilihan tanggal, hingga pembayaran online.

###### **b. Pengelolaan Pemesanan yang Terstruktur**

Sistem digital membantu toko bento cake mencatat pesanan secara otomatis dan rapi, mengurangi kesalahan dan meningkatkan efisiensi.

###### **c. Kolaborasi Aktif dengan Stakeholder**

Mitra bento cake terlibat langsung dalam pengembangan, memastikan fitur dan tampilan sesuai kebutuhan pengguna.

###### **d. Peningkatan Kepuasan Pelanggan**

Pelanggan dapat memesan kapan saja tanpa harus datang langsung, dengan informasi produk yang jelas dan proses pembayaran yang mudah.

###### **e. Pengurangan Beban Adinistratif Manual**

Digitalisasi mengurangi ketergantungan pada pencatatan manual dan interaksi satu-per-satu dengan pelanggan.

## **2. Weakness**

- a. Keterbatasan Sumber Daya dan Pendanaan di Awal Proyek  
Sebagai proyek tahap awal, bisa terdapat kendala dalam pendanaan, tenaga pengembang, atau waktu untuk iterasi sistem.
- b. Ketergantungan pada Kesiapan Mitra Usaha  
Tidak semua mitra usaha bento cake siap atau familiar dengan teknologi digital, yang bisa memperlambat adopsi.
- c. Kebutuhan Maintenance dan Update Berkala  
Website perlu diperbarui secara rutin agar tetap optimal dan bebas dari bug, yang bisa menjadi beban teknis dan finansial.

## **3. Opportunities**

- a. Trend Digitalisasi UMKM  
Pemerintah dan pasar mendorong adopsi teknologi oleh UMKM, termasuk sektor kuliner.
- b. Perluasan Jangkauan Pasar  
Dengan sistem online, toko bento cake dapat menjangkau konsumen di luar wilayah sekitarnya dan menjual produk dengan jangkauan lebih luas.
- c. Integrasi Layanan Tambahan  
Website bisa dikembangkan lebih lanjut dengan fitur-fitur seperti loyalty program, notifikasi pesanan, atau integrasi dengan layanan Pengiriman.
- d. Potensi Kolaborasi dengan Mitra Serupa  
Keberhasilan proyek ini bisa membuka peluang kerja sama dengan toko kue lain atau usaha sejenis yang juga membutuhkan digitalisasi.

#### **4. Threats**

- a. Persaingan dengan Platform Besar  
Website mandiri harus bersaing dengan marketplace makanan besar (seperti GoFood, GrabFood, atau Tokopedia) yang sudah punya infrastruktur kuat.
- b. Resistensi Perubahan dari Pelaku Usaha Tradisional  
Beberapa pemilik usaha masih nyaman dengan cara manual dan skeptis terhadap investasi digital.
- c. Risiko Keamanan Data dan Transaksi  
Sistem online membawa tantangan baru dalam menjaga keamanan data pelanggan dan transaksi Pembayaran.
- d. Perubahan Perilaku Konsumen yang Cepat  
Preferensi konsumen terhadap platform dan teknologi dapat berubah cepat, sehingga website perlu adaptif dan fleksibel

#### **B. Work Breakdown Structure**

1. Perancangan Sistem
  - 1.1 Perencanaan Proyek
  - 1.2 Menghubungi Mitra
  - 1.3 Identifikasi Fitur Sistem
  - 1.4 Perancangan UI/UX
  - 1.5 Perancangan Database
2. Pengembangan Sistem
  - 2.1 Pengembangan Front-End
  - 2.2 Pengembangan Back-End
  - 2.3 Integrasi Sistem
3. Pengujian & Implementasi
  - 3.1 Pengujian Fungsional
  - 3.2 Perbaikan Bug
  - 3.3 Hosting

### 3.4 Finalisasi dan Laporan Hasil

#### C. Sumber Kebutuhan Daya

##### 1. Sumber Daya Manusia (SDM)

Tim kami terdiri dari empat anggota yang berperan penting dalam kesuksesan proyek.

###### a. Project Manager/Ketua Kelompok (Finna Luthfiah Hafizh)

(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)

- Bertanggung jawab mengoordinasi seluruh proses pengembangan.
- Menyusun timeline, mengatur rapat koordinasi, dan menjembatani komunikasi antara tim pengembang dan mitra usaha.
- Memastikan proyek berjalan sesuai rencana dan target.
- Mencatat hasil setiap rapat.
- Menghubungi mitra dan negosiasi secara offline
- Membagi jobdesk masing-masing kelompok
- Koordinasi ke dosen pembimbing
- Memantau dan terus evaluasi tiap jobdesk kelompok
- Membuat jadwal pelaksanaan kerja kelompok

###### b. Sekretaris (Siti Nur Rohmah)

(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)

- Mendokumentasikan seluruh kegiatan proyek, termasuk risalah rapat, keputusan penting, dan perubahan rencana.
- Mengelola dokumen administratif proyek, seperti surat-menyerat, dan laporan kemajuan.
- Mengatur jadwal rapat dan mengingatkan anggota tim mengenai agenda proyek.

- Membantu Project Manager dalam menyusun laporan berkala dan pelaporan ke mitra usaha atau dosen pembimbing.
  - Menyimpan dan mengarsipkan file penting proyek di repository atau cloud storage yang disepakati tim.
- c. UI/UX Designer (Ipmawati Dwi Refinda)  
(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)
- Mendesain tampilan dan pengalaman pengguna (user interface dan user experience) website.
  - Membuat mockup yang user-friendly berdasarkan masukan dari mitra bento cake.
  - Memastikan tampilan website menarik, intuitif, dan sesuai identitas brand.
- d. Frontend Developer (Finna Luthfiah Hafizh)  
(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)
- Mengubah desain UI menjadi halaman web interaktif menggunakan HTML, CSS, dan JavaScript.
  - Mengintegrasikan tampilan dengan sistem backend.
- e. Backend Developer (Alya Nurhasanah)  
(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)
- Membangun sistem logika dan fungsionalitas website seperti pemrosesan pemesanan, manajemen data produk, dan sistem pembayaran.
  - Memastikan keamanan serta keandalan server.
- f. Quality Assurance (QA)/Tester (Siti Nur Rohmah)  
(Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)
- Menguji fungsi-fungsi website untuk memastikan tidak ada bug atau error.
  - Melakukan uji coba dan mencatat feedback pengguna.

g. Content Writer / Admin Produk (Ipmawati Dwi Refinda)  
 (Pengerjaan = offline +- 4jam , online (kerja di kos masing-masing) = fleksibel)

- Menulis deskripsi produk, informasi toko, dan konten halaman lainnya.
- Mengelola data produk seperti foto, harga, dan keterangan pada sistem website

## 2. Sumber Daya Fisik

### a. Software

- VSCode
- Composer
- Xampp
- Figma
- Draw I/O
- Spreadsheet

### b. Hardware

- Laptop
- Wi-Fi
- Smartphone

## D. Rencana Jadwal Pelaksanaan Proyek

Table 1 Rencana Jadwal Pelaksanaan Proyek

Kode WBS	Nama Pekerjaan	Mgg	PJ	Biaya											
		1	2	3	4	5	6	7	8	9	10	11	12		
1.	Perencanaan Proyek													F,R,I,A	Rp 120.000,00
2.	Menghubungi Mitra													F	Rp 0,-
3.	Identifikasi Fitur													I	Rp 0,-
4.	Perancangan ui/ux													I,A	Rp 60.000,00
5.	Perancangan Database													A	Rp 20.000,00
6.	Pengembangan Front-end													F	Rp 150.000,00
7.	Pengembangan Back-end													A	Rp 120.000,00

8.	Integrasi Sistem													F,A	Rp 100.000,00
9.	Pengujian Fungsional													R	Rp 20.000,00
10.	Perbaiki Bug													F,A	Rp 10.000,00
11	Hosting													A	Rp 55.000,00
12	Finalisasi dan Laporan Hasil													R	Rp 0,-

**Keterangan:**

F = Fina

R = Rahma

I = Ipma

A = Alya

### E. Rencana Nilai Proyek

No	Akun	Durasi	Harga	Jumlah
1	Pembuatan Proposal	4 hari	Rp 10.000/jam	Rp 40.000,00
2	Pembuatan Rekapitulasi Dana	2 hari	Rp 10.000/jam	Rp 20.000,00
3	Pedoman/Tutorial	2 hari	Rp 15.000/jam	Rp 60.000,00
	Total Project Manager			Rp 120.000,00
4	Desain UI	3 hari	Rp 20.000/jam	Rp 60.000,00
	Total Desain UI/UX			Rp 60.000,00
5	Rancangan Database	1 hari	Rp 20.000/jam	Rp 20.000,00
6	Pengembangan Front End	7 hari	Rp 25.000/jam	Rp 175.000,00
7	Integrasi Dengan Back End	5 hari	Rp 25.000/jam	Rp 125.000,00
8	Pengembangan Back End	6 hari	Rp 20.000/jam	Rp 120.000,00
	Total Full Stack Programmer			Rp 740.000,00
9	Perencanaan Pengujian	1 hari	Rp 15.000/jam	Rp 15.000,00
10	Pengujian Fungsi	1 hari	Rp 20.000/jam	Rp 20.000,00
	Total Quality Assurance			Rp 35.000,00
	Total Biaya			Rp 955.000,00

## BAB III

### PELAKSANAAN PROYEK

#### A. Realisasi Jadwal Pelaksanaan

##### 1. Perencanaan Proyek

Perencanaan proyek dilakukan sebagai tahap awal untuk memastikan arah dan pelaksanaan proyek berjalan sesuai target. Tim memulai dengan menyusun Work Breakdown Structure (WBS) untuk membagi kegiatan ke dalam tahapan yang jelas, mulai dari perencanaan, pengembangan sistem, pengujian, hingga implementasi dan pelaporan hasil. Kegiatan awal mencakup menghubungi mitra usaha Cammeo guna menggali kebutuhan sistem, lalu dilanjutkan dengan identifikasi fitur utama yang akan dikembangkan. Tim juga menyusun jadwal mingguan untuk memantau progres proyek dan memastikan semua pekerjaan dilakukan sesuai timeline yang telah ditetapkan. Berikut table realisasi jadwal perencanaan proyek :

No	Nama Pekerjaan	PJ	Minggu Pelaksanaan (1–12)
1	Perencanaan Proyek	F,R,I,A	Minggu 1
2	Menghubungi Mitra	F	Minggu 1
3	Identifikasi Fitur	I	Minggu 2
4	Perancangan UI/UX	I,A	Minggu 3
5	Perancangan Database	A	Minggu 3
6	Pengembangan Front-End	F	Minggu 4 – 6
7	Pengembangan Back-End	A	Minggu 5 – 7
8	Integrasi Sistem	F,A	Minggu 8
9	Pengujian Fungsional	R	Minggu 9
10	Perbaikan Bug	F,A	Minggu 10
11	Hosting	A	Minggu 11
12	Finalisasi dan Laporan	R	Minggu 12

##### 2. Menghubungi Mitra

Sebagai langkah awal pelaksanaan proyek, tim melakukan pendekatan langsung kepada pemilik toko Cammeo untuk menawarkan solusi digital berupa website pemesanan kue. Proses ini diawali dengan perkenalan tim dan pemaparan singkat mengenai tujuan proyek serta manfaat digitalisasi bagi bisnis kuliner, khususnya dalam hal efisiensi pemesanan dan pengelolaan data. Mitra menyambut baik tawaran tersebut karena melihat peluang peningkatan layanan dan jangkauan pasar. Dalam pertemuan tersebut, kami juga menggali kebutuhan dan kendala yang

selama ini dihadapi oleh mitra dalam proses pemesanan manual. Hasil diskusi ini menjadi dasar dalam merancang fitur dan alur sistem yang sesuai dengan kebutuhan nyata pengguna, sehingga proyek benar-benar memberikan nilai tambah bagi mitra usaha.

### 3. Ruang Lingkup dan Identifikasi Fitur

Fitur yang akan dimasukkan pada sistem akan disesuaikan dengan kebutuhan pengguna atau owner yaitu:

#### a. Halaman Website

- **Home** – menampilkan profil singkat CAMMEO, promo, dan CTA pemesanan.
- **Menu** – menampilkan daftar produk cake beserta harga dan varian.
- **About** – menjelaskan sejarah, visi, dan kontak CAMMEO.
- **Checkout** – form pemesanan (nama, no WA, pilihan produk, varian, jumlah).

#### b. Proses Pemesanan

- User mengisi form pemesanan pada halaman checkout.
- Sistem menyimpan data pesanan ke database.
- Sistem menampilkan ringkasan pesanan.
- Tombol "**Konfirmasi via WhatsApp**" otomatis membuka WA admin dengan pesan terformat.

#### c. Manajemen Produk (Backend)

- Admin dapat menambah, mengubah, menghapus produk & varian.
- Admin dapat melihat daftar pesanan yang masuk.

#### d. Teknologi yang Digunakan

- **Framework:** Laravel (tanpa sistem login user).
- **Database:** MySQL/MariaDB.
- **Server Lokal:** Laragon (untuk pengembangan).

#### e. Batasan Sistem

- Tidak ada login/registrasi pelanggan.
- Pembayaran dilakukan manual melalui transfer/konfirmasi

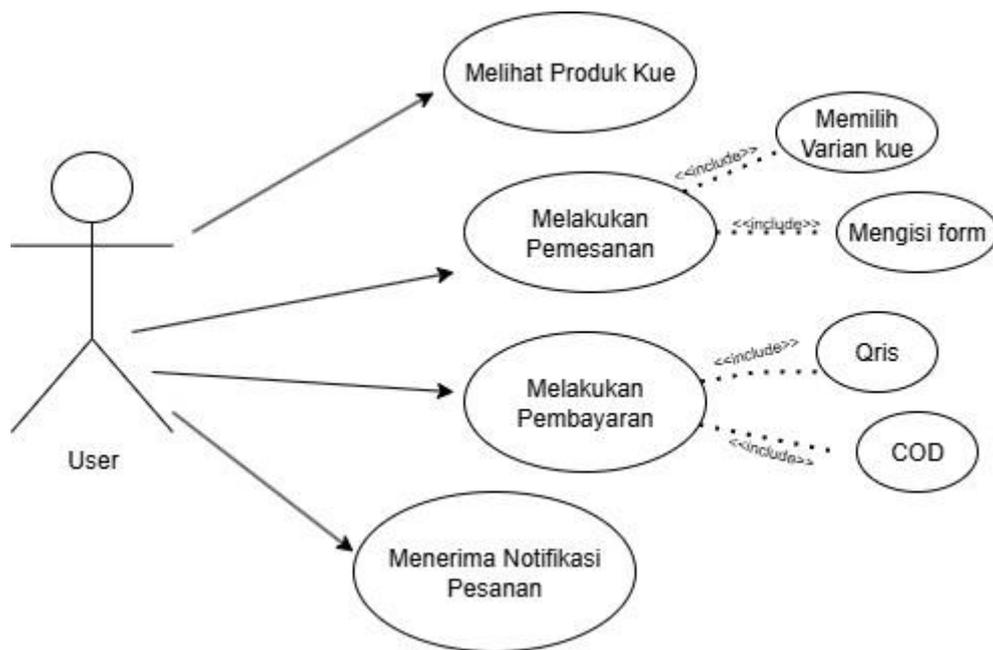
WA.

- o Tidak ada integrasi dengan sistem pembayaran otomatis (payment gateway).

Setelah fitur-fitur diidentifikasi, langkah selanjutnya adalah membuat use case diagram serta struktur menu yang jelas dan terorganisir. Use case diagram ini akan memvisualisasikan berbagai interaksi antara pengguna dan sistem, sehingga membantu dalam memahami kebutuhan dan alur kerja aplikasi secara komprehensif.

Diagram :

### 1. Use Case User



Gambar 1 Use Case Diagram User

Pengguna dapat melihat daftar produk kue yang tersedia dan memilih varian yang diinginkan. Setelah itu, pengguna dapat melakukan pemesanan dengan mengisi form yang mencakup informasi pribadi dan detail pesanan. Untuk pembayaran, sistem menyediakan dua metode, yaitu QRIS dan pembayaran tunai di tempat (COD). Setelah melakukan pemesanan dan pembayaran, pengguna akan menerima notifikasi terkait status pesanan mereka. Relasi “include” digunakan dalam diagram ini untuk menunjukkan bahwa proses seperti memilih varian dan mengisi form merupakan bagian integral dari kegiatan pemesanan, serta bahwa pilihan metode pembayaran merupakan bagian dari proses pembayaran.

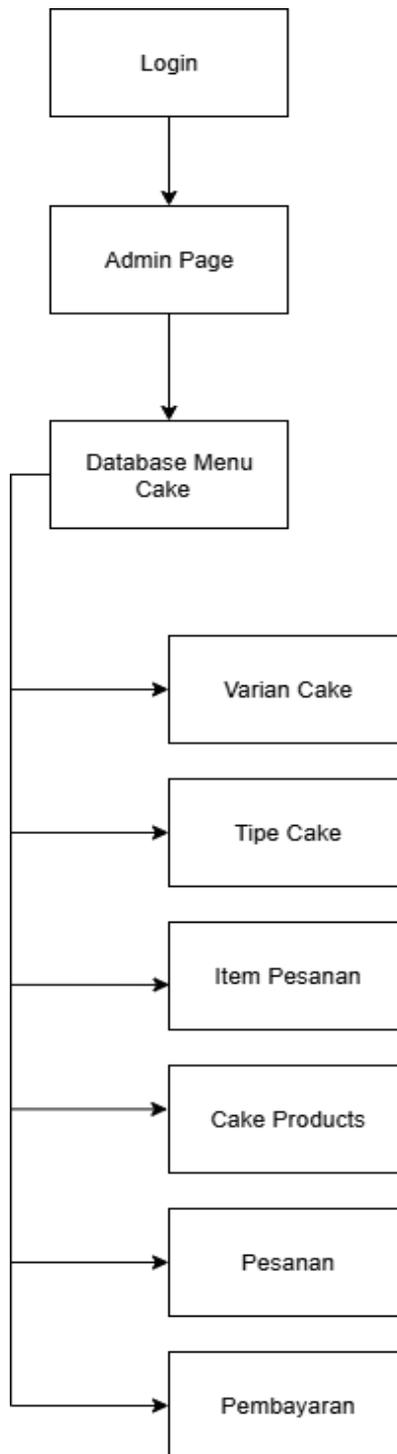
## 2. Use Case Admin



## Gambar 2 Use Case Diagram Admin

Admin memiliki hak akses untuk melakukan login ke dalam sistem dengan memasukkan email dan password. Setelah login, admin dapat melakukan pengelolaan terhadap pesanan pelanggan, termasuk melihat daftar pesanan, menghapus pesanan, dan mengedit data pesanan. Selain itu, admin juga dapat mengelola data produk kue, seperti menambahkan produk baru, mengedit informasi produk yang sudah ada, dan menghapus produk yang tidak lagi tersedia. Hubungan antara setiap fungsi divisualisasikan dengan relasi “extend”, yang menunjukkan bahwa fitur-fitur seperti tambah, edit, atau hapus merupakan bagian dari proses utama pengelolaan data.

### 3. Struktur Menu

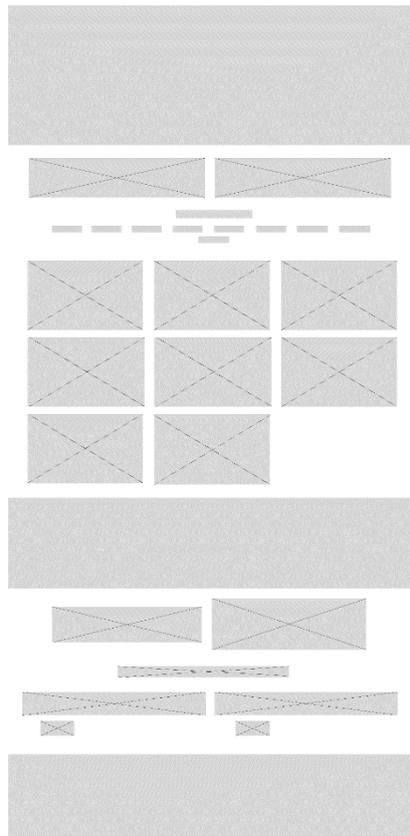


Gambar 3 Struktur Menu Cammeo

Proses diagram struktur menu dimulai dari fitur login, di mana admin harus melakukan autentikasi terlebih dahulu menggunakan email dan password. Setelah berhasil masuk, admin diarahkan ke halaman utama admin (Admin Page) yang menjadi pusat pengelolaan seluruh data dalam sistem. Dari halaman ini, admin dapat mengakses menu database kue yang terdiri dari beberapa komponen penting, seperti varian kue, tipe kue, item pesanan, daftar produk kue, data pesanan pelanggan, dan informasi pembayaran. Setiap menu tersebut berfungsi untuk membantu admin dalam melakukan pengelolaan data secara terstruktur dan efisien.

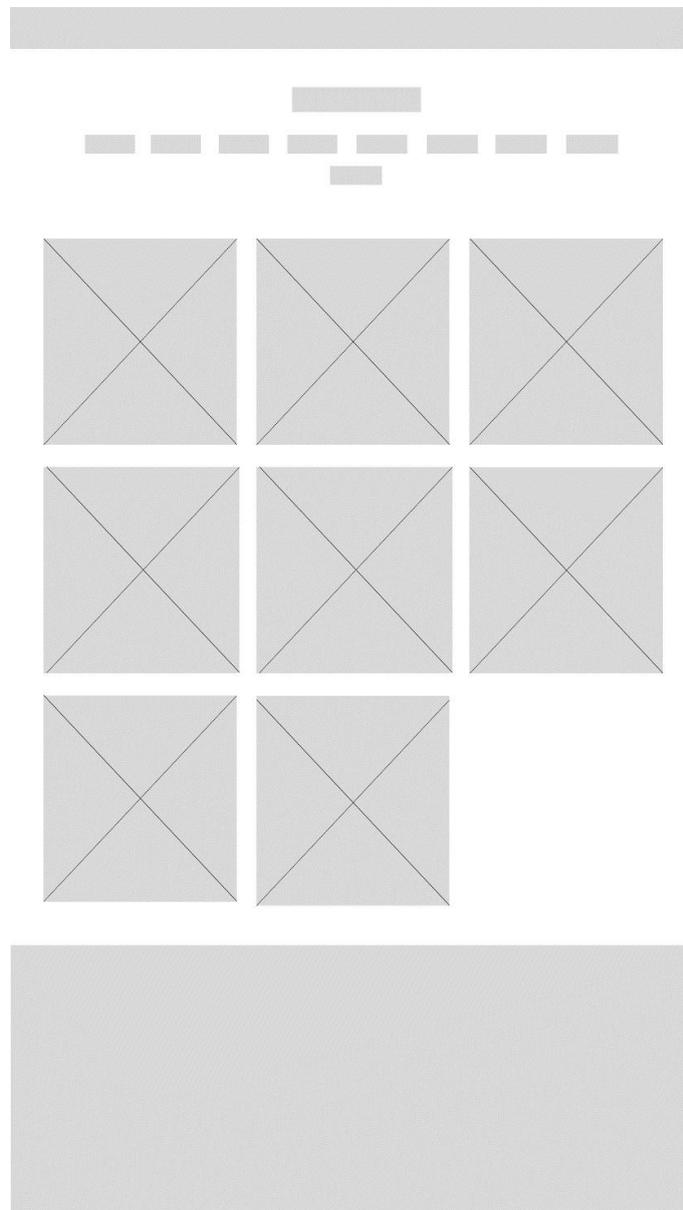
#### 4. Perancangan UI/UX

##### a. Homepage



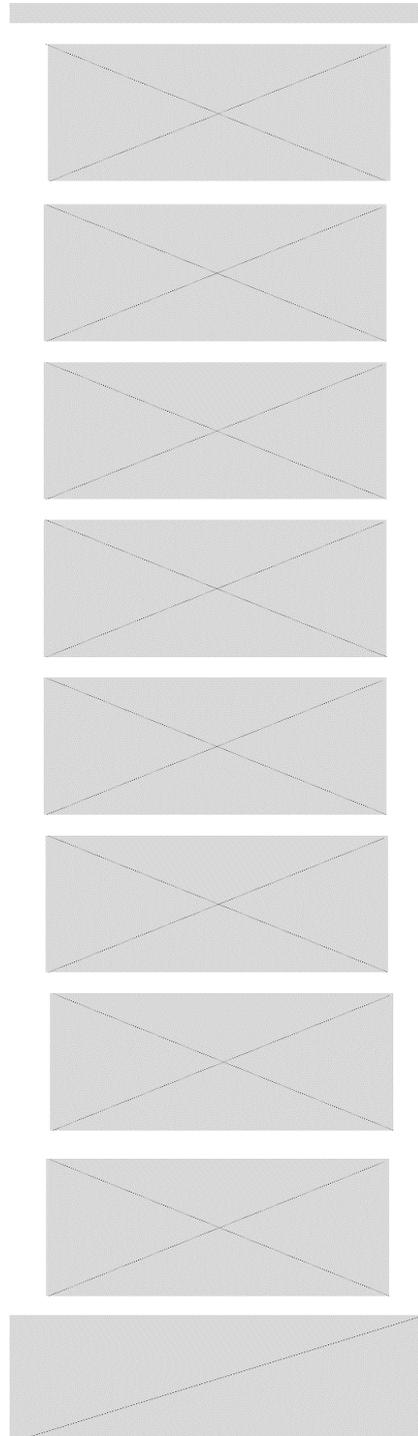
Gambar 4 Rancangan Home Page Website

b. Menu



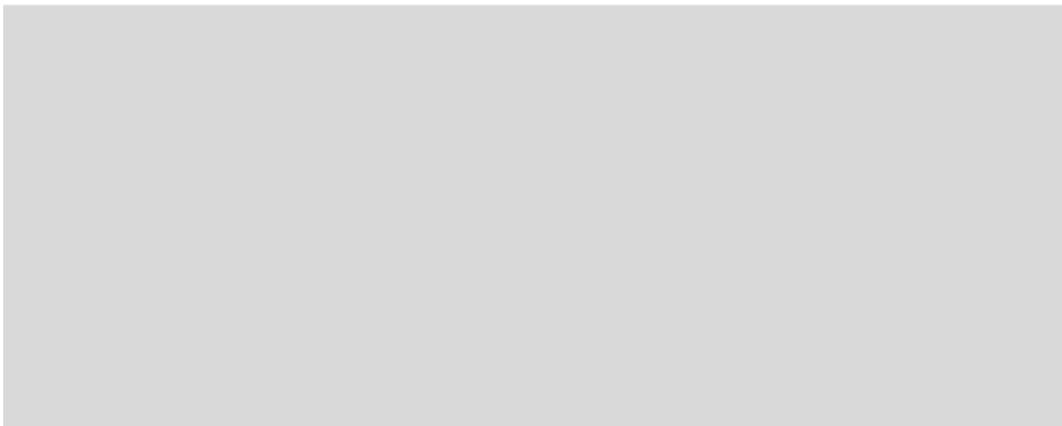
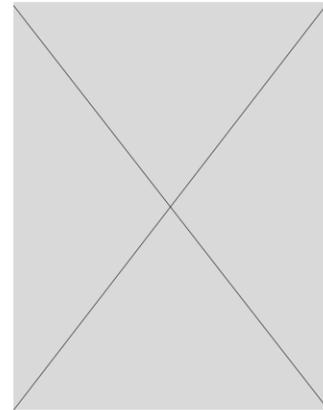
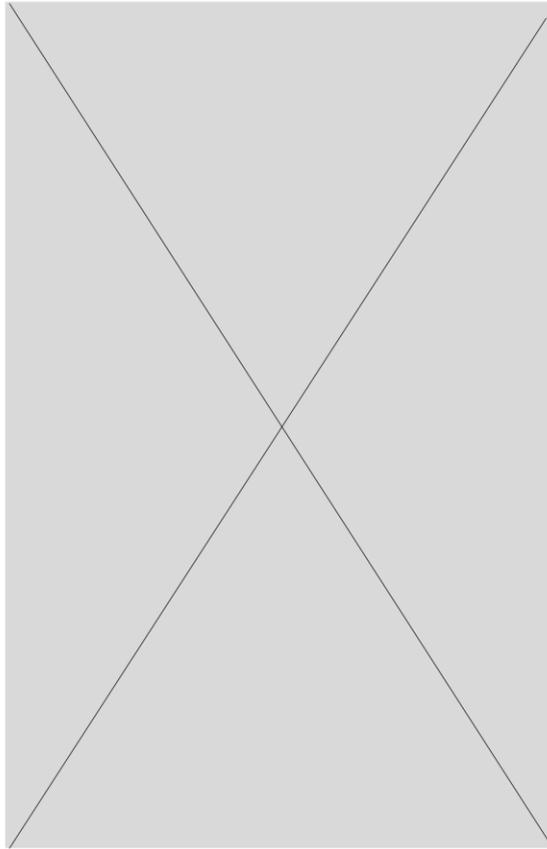
Gambar 5 Rancangan Daftar Menu

c. Detail Menu



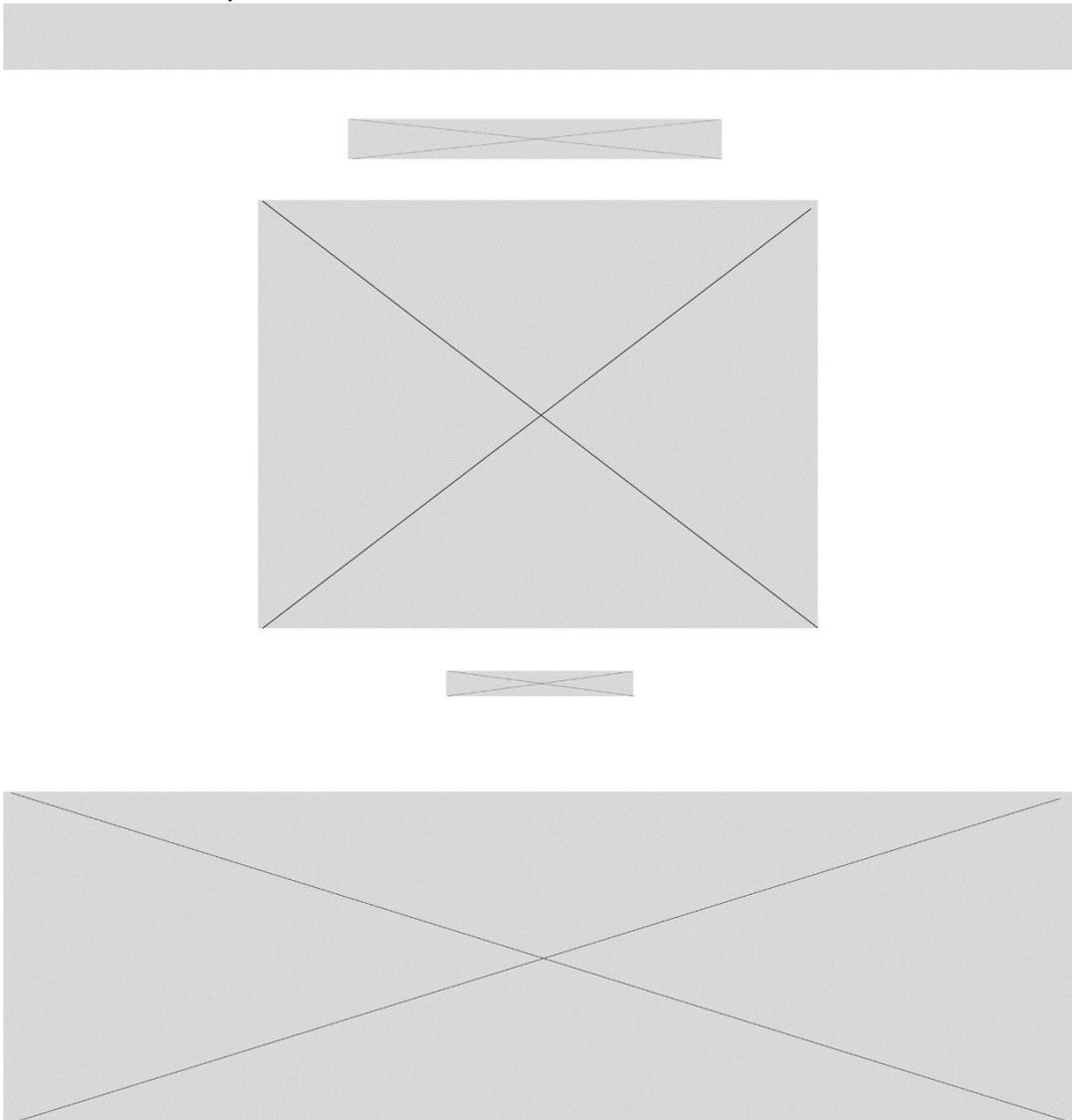
Gambar 6 Rancangan Detail Menu

d. Order Now



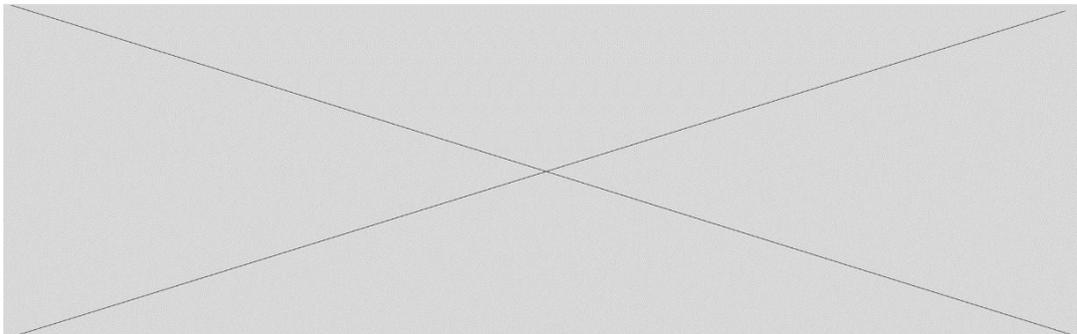
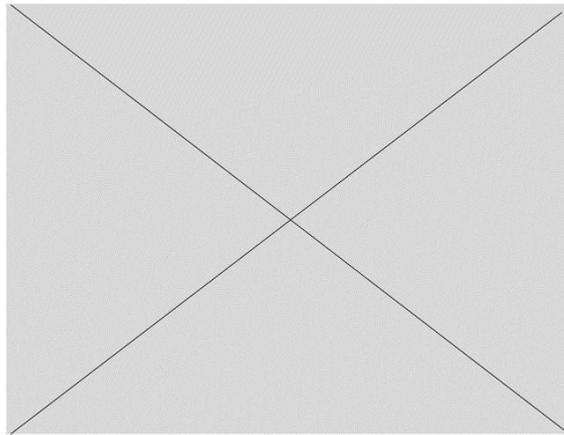
Gambar 7 Rancangan Bagian Order

e. Payment Cash



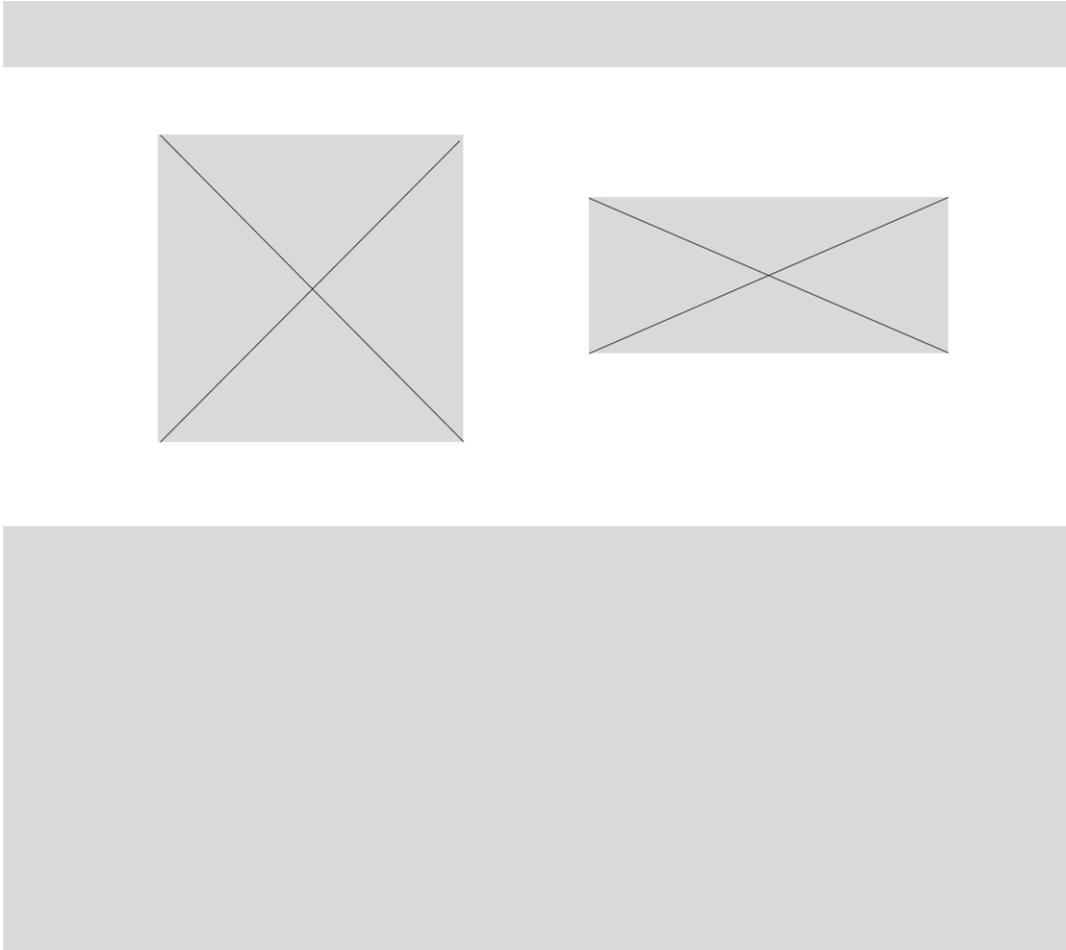
Gambar 8 Rancangan Payment Cash

f. Payment Qris



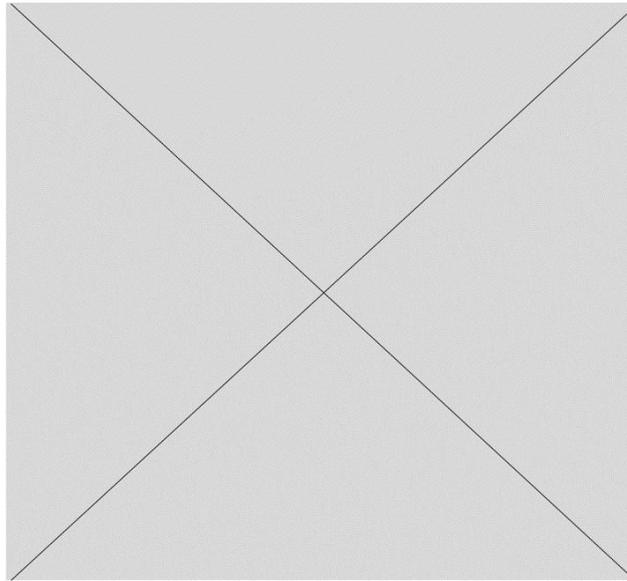
Gambar 9 Rancangan Payment Qris

g. About



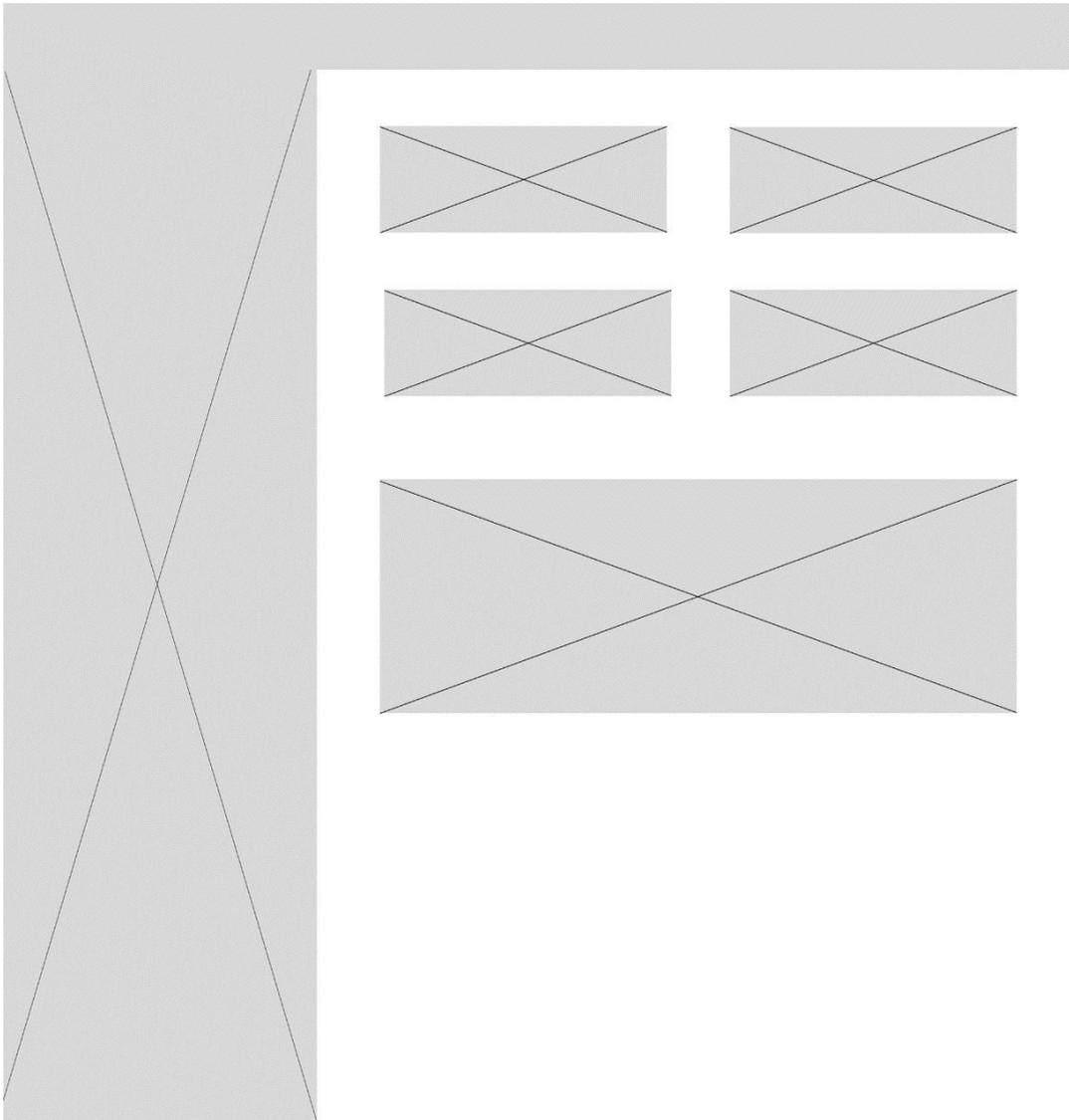
Gambar 10 Rancangan About Cammeo

#### h. Login Admin



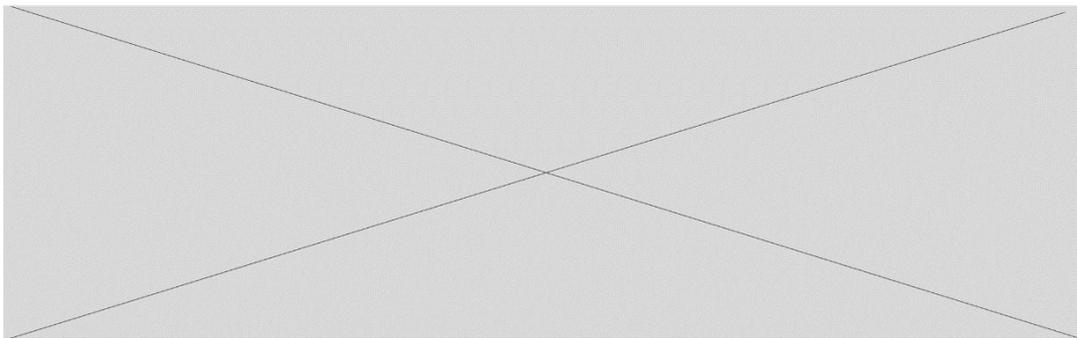
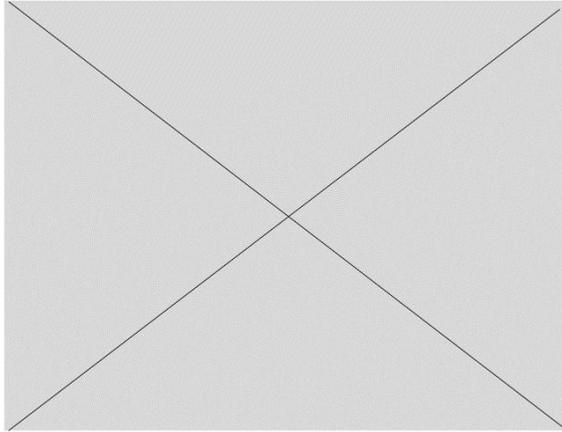
Gambar 11 Rancangan Bagian Admin Login

i. Dashboard Admin



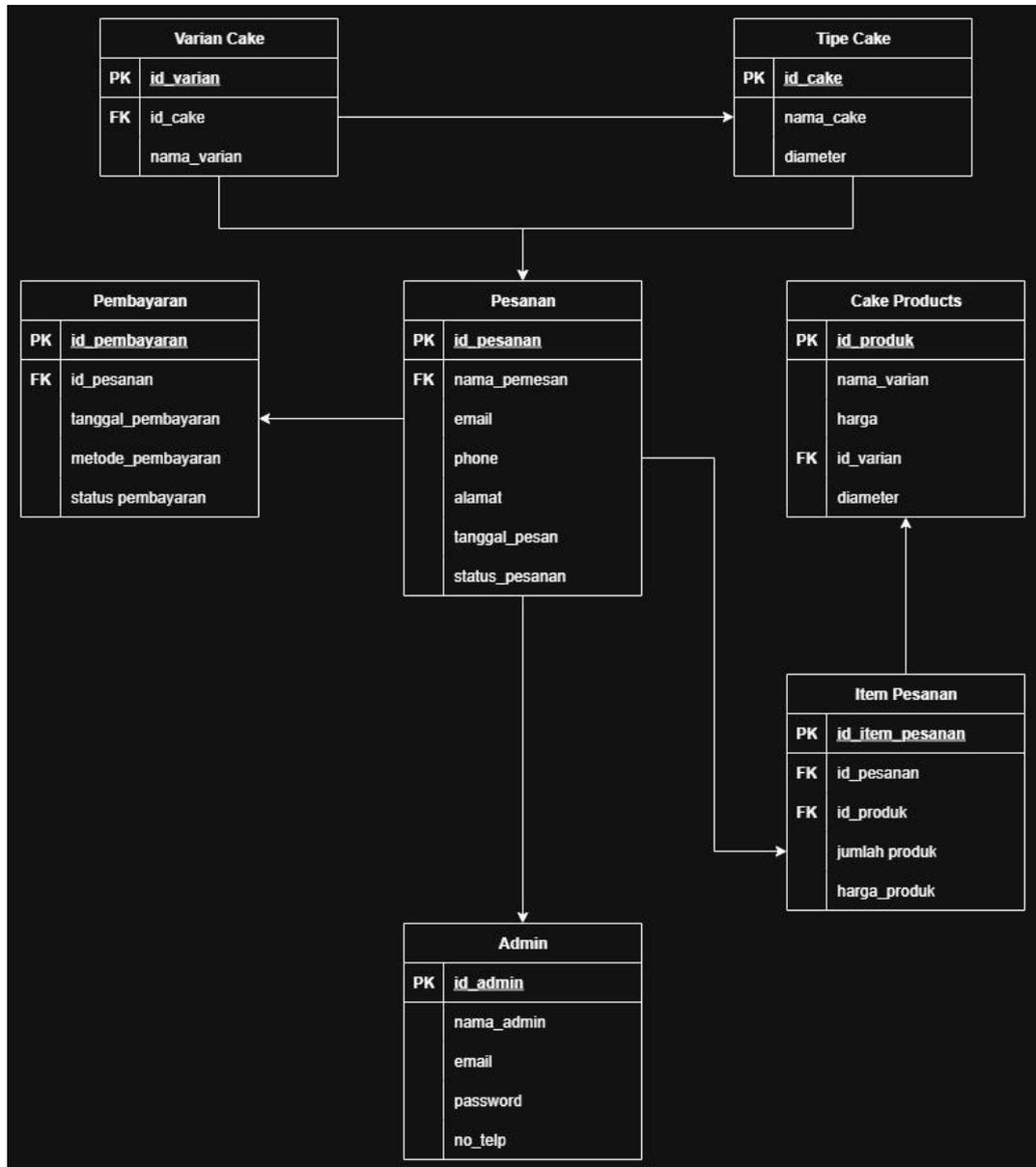
Gambar 12 Rancangan Dashboard Admin

j. Konfirmasi Pesanan



Gambar 13 Rancangan Bagian Konfirmasi Pesanan

## 5. Perancangan Database



Gambar 14 Perancangan Database

### 1. Tipe Cake

- id\_cake (PK): ID unik untuk tiap jenis cake (misalnya Blackforest, Tiramisu, dll).
- nama\_cake: Nama dari jenis cake.
- diameter: Ukuran diameter cake.

Relasi: Satu Tipe Cake bisa memiliki banyak Varian Cake (1 to many).

## 2. Varian Cake

- id\_varian (PK): ID unik untuk tiap varian cake.
- id\_cake (FK): Mengacu ke tabel Tipe Cake.
- nama\_varian: Nama variannya (misal: Cokelat, Stroberi, Keju, dll).

Relasi: Satu Varian Cake bisa memiliki banyak Produk (Cake Products)

## 3. Cake Product

- id\_produk (PK): ID unik untuk produk cake tertentu.
- nama\_varian: Nama varian produk (redundan, tapi bisa untuk deskripsi).
- harga: Harga produk.
- id\_varian (FK): Mengacu ke tabel Varian Cake.
- diameter: Ukuran diameter produk.

Relasi: Satu produk bisa dipesan oleh banyak item pesanan (Item Pesanan)

## 4. Pesanan

- id\_pesanan (PK): ID unik pesanan.
- nama\_pemesan, email, phone, alamat: Info pemesan.
- tanggal\_pesanan: Tanggal pemesanan dilakukan.
- status\_pesanan: Status (misal: pending, diproses, selesai).

Relasi: Satu pesanan dapat memiliki:

- Banyak Item Pesanan
- Satu Pembayaran

## 5. Item Pesanan

- id\_item\_pesanan (PK): ID unik item pesanan.
- id\_pesanan (FK): Mengacu ke Pesanan.
- id\_produk (FK): Mengacu ke Cake Products.
- jumlah\_produk: Berapa banyak produk dipesan.
- harga\_produk: Harga pada saat pemesanan.

## 6. Pembayaran

- id\_pembayaran (PK): ID unik pembayaran.
- id\_pesanan (FK): Mengacu ke pesanan.
- tanggal\_pembayaran: Waktu pembayaran dilakukan.

- metode\_pembayaran: Transfer, e-wallet, dsb.
- status\_pembayaran: Lunas, pending, gagal, dll.

Relasi: Satu pembayaran hanya terkait ke satu pesanan.

## 7. Admin

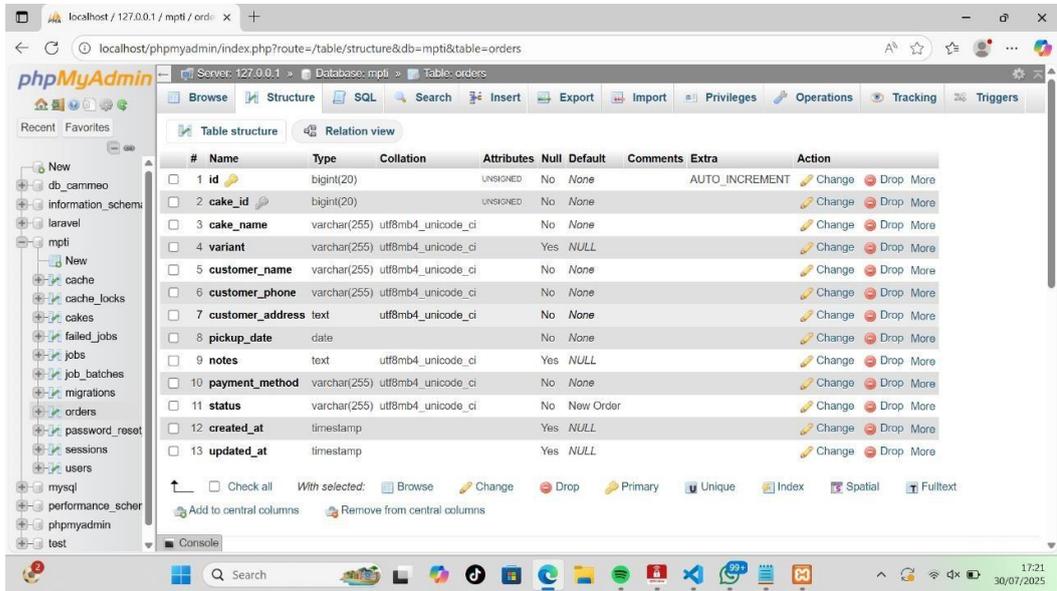
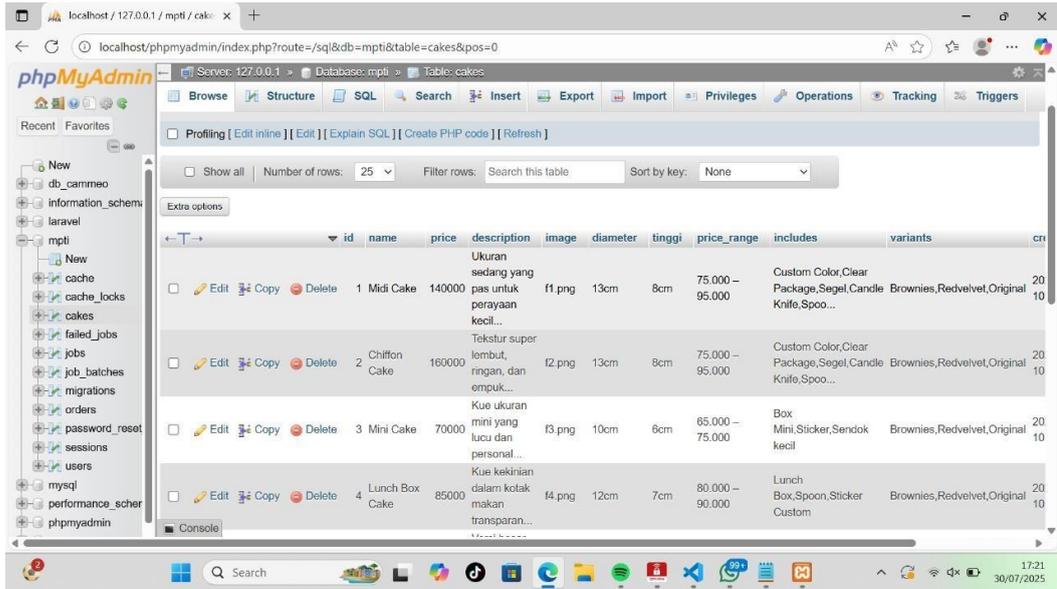
- id\_admin (PK): ID unik admin.
- nama\_admin, email, password, no\_telp: Data admin yang mengelola sistem.

Fungsi: Mungkin digunakan untuk login dan pengelolaan backend sistem (tidak memiliki relasi langsung dalam ERD, tetapi fungsional).

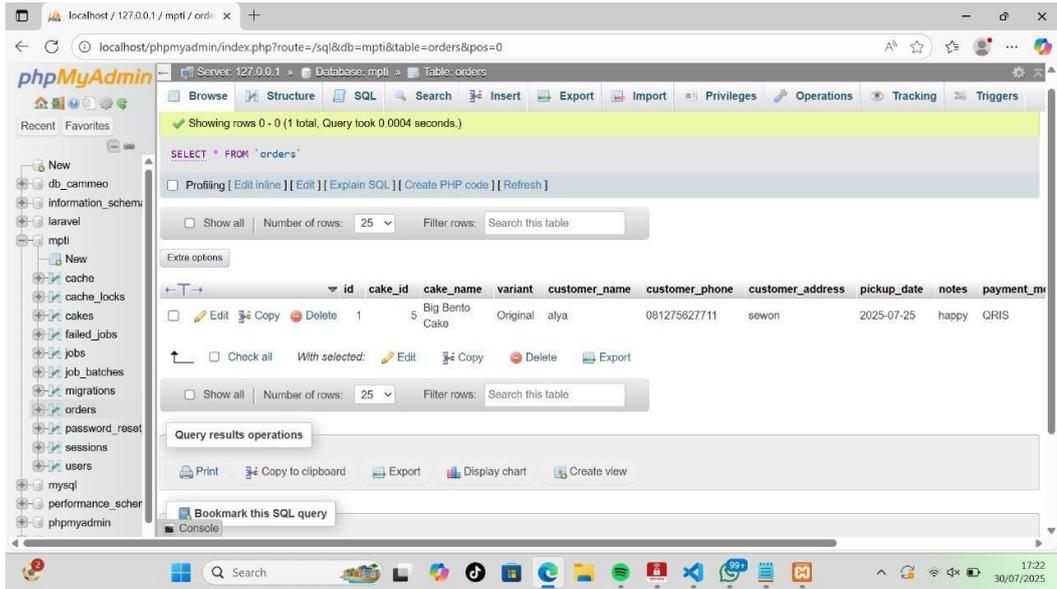
The screenshot shows the phpMyAdmin interface for the 'mpti' database, specifically the 'cakes' table. The table structure is as follows:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
3	price	int(11)			No	None			Change Drop More
4	description	text	utf8mb4_unicode_ci		No	None			Change Drop More
5	image	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
6	diameter	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
7	tinggi	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
8	price_range	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
9	includes	text	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
10	variants	text	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
11	created_at	timestamp			Yes	NULL			Change Drop More
12	updated_at	timestamp			Yes	NULL			Change Drop More

struktur table cakes

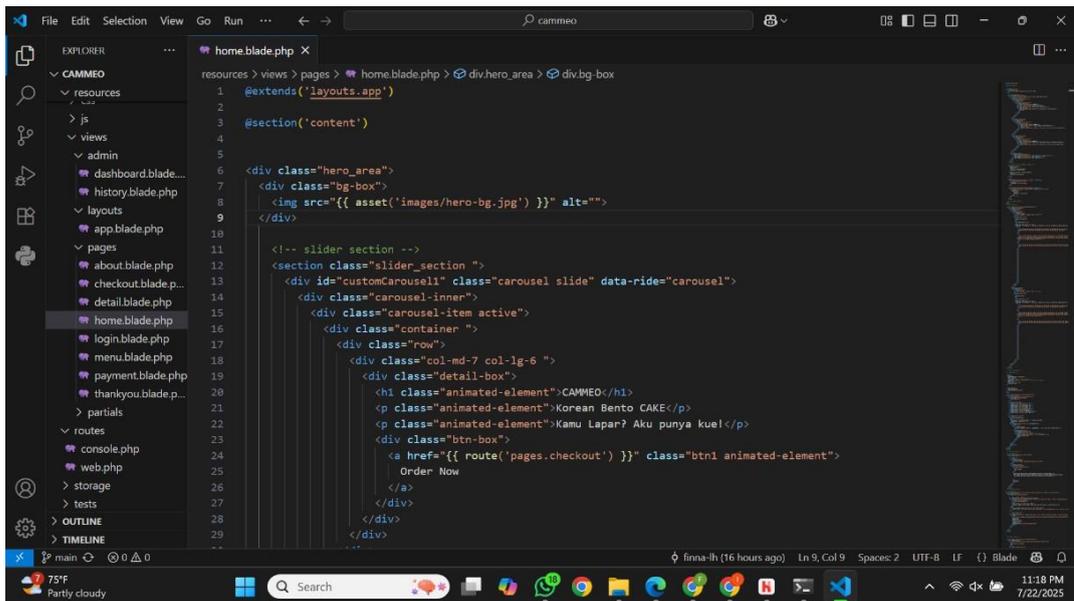


struktur tabel orders



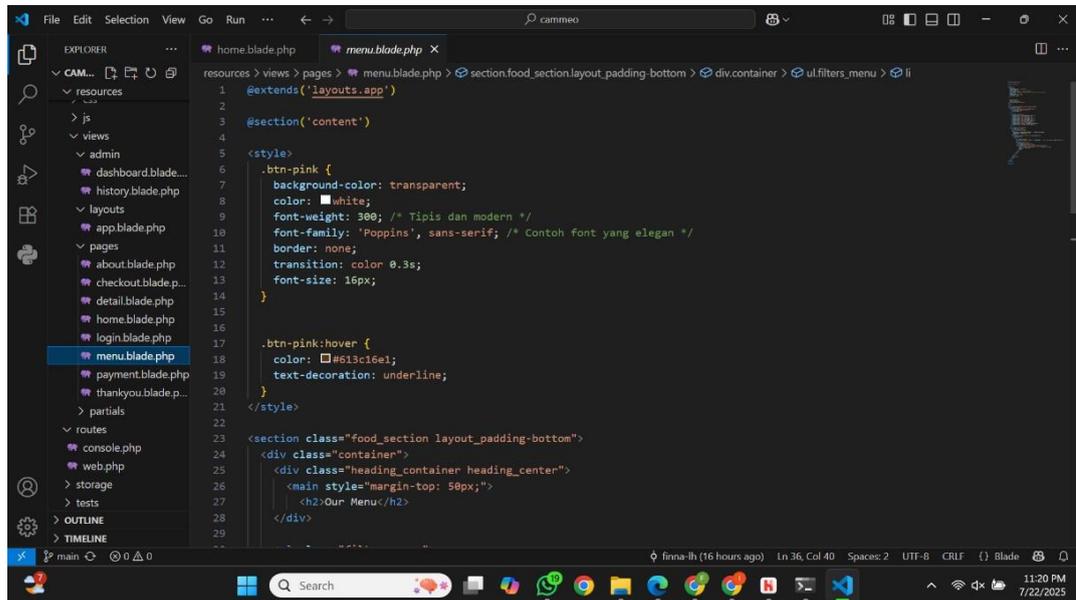
## 6. Pengembangan Front-end

### Home



Gambar 15 Frontend Home

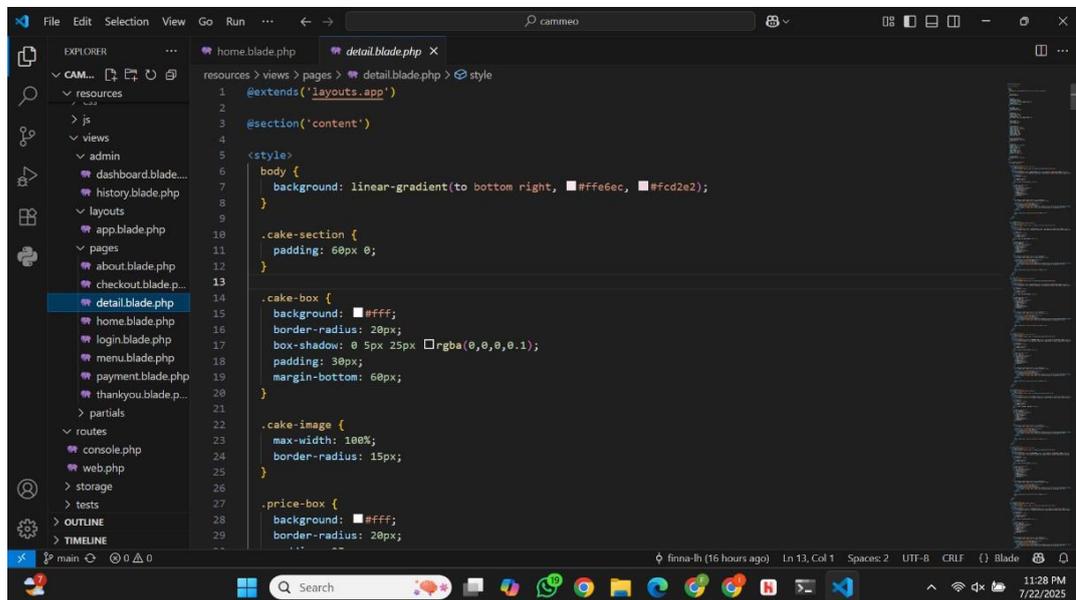
## Menu



```
1 @extends('layouts.app')
2
3 @section('content')
4
5 <style>
6   .btn-pink {
7     background-color: transparent;
8     color: white;
9     font-weight: 300; /* Tipis dan modern */
10    font-family: 'Poppins', sans-serif; /* Contoh font yang elegan */
11    border: none;
12    transition: color 0.3s;
13    font-size: 16px;
14  }
15
16  .btn-pink:hover {
17    color: #613c16e1;
18    text-decoration: underline;
19  }
20 </style>
21
22 <section class="food_section layout_padding-bottom">
23   <div class="container">
24     <div class="heading_container heading_center">
25       <main style="margin-top: 50px;">
26         <h2>Our Menu</h2>
27       </div>
28     </div>
29   </section>
```

Gambar 16 Frontend Menu

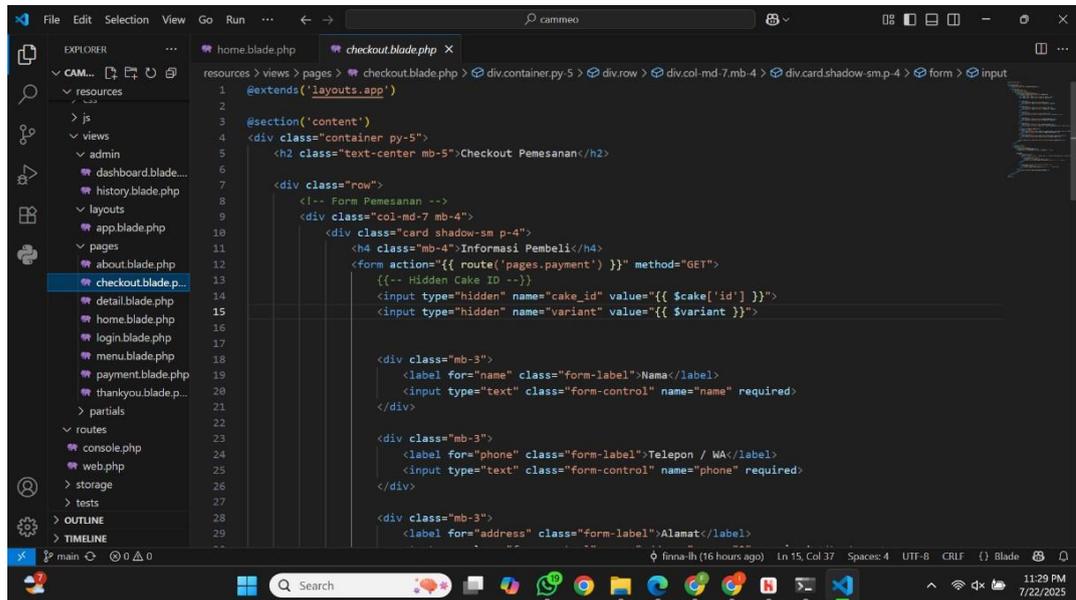
## Detail Menu



```
1 @extends('layouts.app')
2
3 @section('content')
4
5 <style>
6   body {
7     background: linear-gradient(to bottom right, #ffe6ec, #fcd2e2);
8   }
9
10  .cake-section {
11    padding: 60px 0;
12  }
13
14  .cake-box {
15    background: #ffff;
16    border-radius: 20px;
17    box-shadow: 0 5px 25px #000000;
18    padding: 30px;
19    margin-bottom: 60px;
20  }
21
22  .cake-image {
23    max-width: 100%;
24    border-radius: 15px;
25  }
26
27  .price-box {
28    background: #ffff;
29    border-radius: 20px;
30  }
```

Gambar 17 Frontend Detail Menu

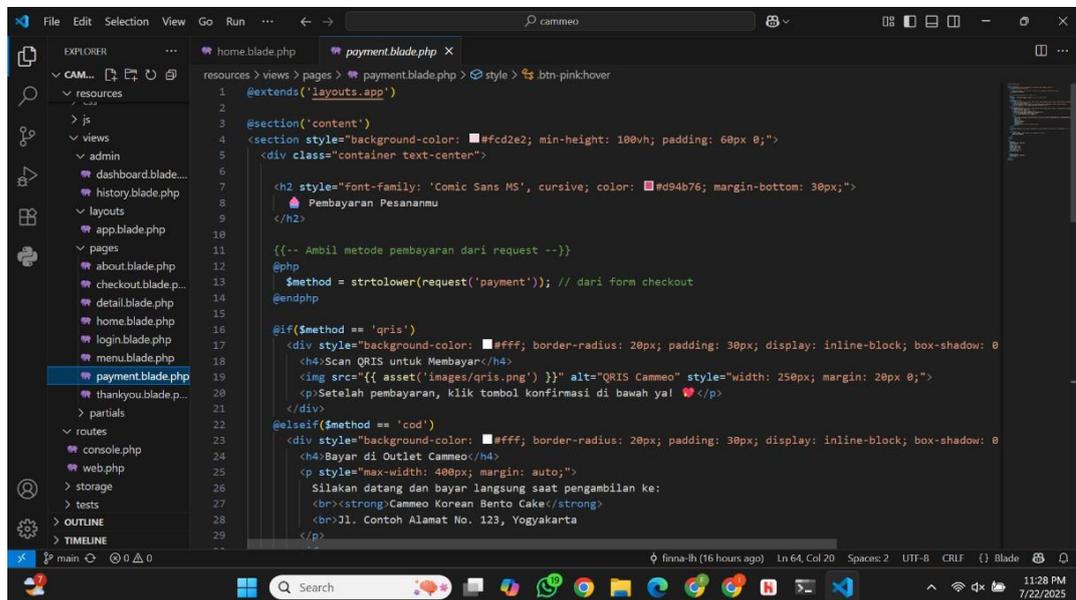
## Order now



```
1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container py-5">
5   <h2 class="text-center mb-5">Checkout Pemesanan</h2>
6
7   <div class="row">
8     <!-- Form Pemesanan -->
9     <div class="col-md-7 mb-4">
10      <div class="card shadow-sm p-4">
11        <h4 class="mb-4">Informasi Pembeli</h4>
12        <form action="{{ route('pages.payment') }}" method="GET">
13          <!-- Hidden Cake ID -->
14          <input type="hidden" name="cake_id" value="{{ $cake['id'] }}">
15          <input type="hidden" name="variant" value="{{ $variant }}">
16
17          <div class="mb-3">
18            <label for="name" class="form-label">Nama</label>
19            <input type="text" class="form-control" name="name" required>
20          </div>
21
22          <div class="mb-3">
23            <label for="phone" class="form-label">Telepon / WA</label>
24            <input type="text" class="form-control" name="phone" required>
25          </div>
26
27          <div class="mb-3">
28            <label for="address" class="form-label">Alamat</label>
29            <input type="text" class="form-control" name="address" required>
30          </div>
31        </form>
32      </div>
33    </div>
34  </div>
35 </div>
```

Gambar 18 Frontend Order Now

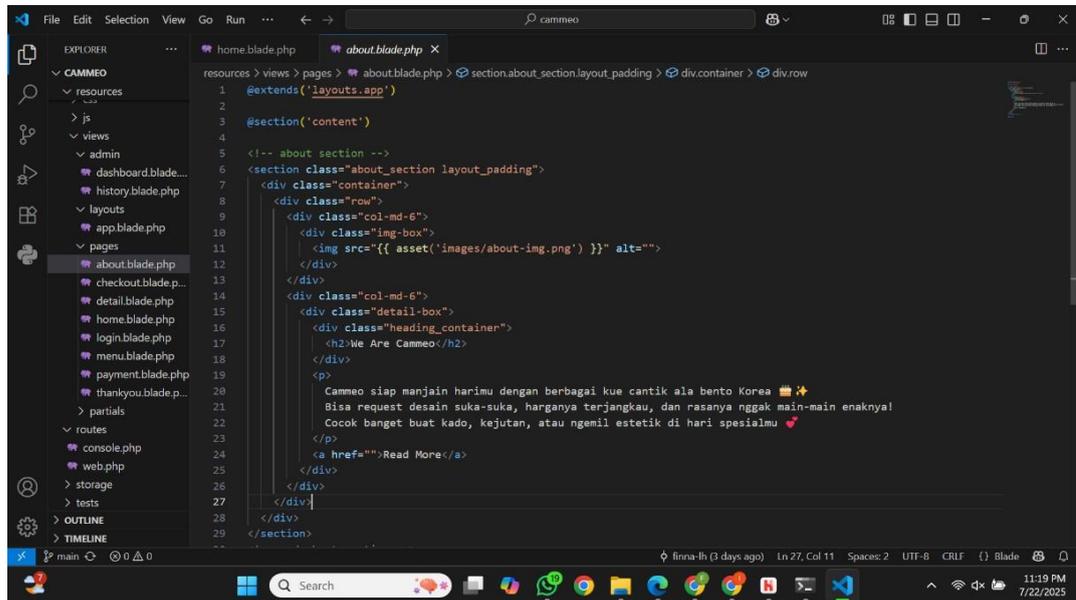
## Payment



```
1 @extends('layouts.app')
2
3 @section('content')
4 <div style="background-color: #fcd2e2; min-height: 100vh; padding: 60px 0;">
5   <div class="container text-center">
6     <h2 style="font-family: 'Comic Sans MS', cursive; color: #d94b76; margin-bottom: 30px;">
7       Pembayaran Pesananmu
8     </h2>
9
10    <!-- Ambil metode pembayaran dari request -->
11    @php
12      $method = strtolower(request('payment')); // dari form checkout
13    @endphp
14
15    @if($method == 'qris')
16      <div style="background-color: #ffff; border-radius: 20px; padding: 30px; display: inline-block; box-shadow: 0 4px 6px #ccc;">
17        <h4>Scan QRIS untuk Membayar</h4>
18        
19        <p>Setelah pembayaran, klik tombol konfirmasi di bawah ya! </p>
20      </div>
21
22    @elseif($method == 'cod')
23      <div style="background-color: #ffff; border-radius: 20px; padding: 30px; display: inline-block; box-shadow: 0 4px 6px #ccc;">
24        <h4>Bayar di Outlet Cammeo</h4>
25        <p style="max-width: 400px; margin: auto;">
26          Silakan datang dan bayar langsung saat pengambilan ke:
27          <br><strong>Cammeo Korean Bento Cake</strong>
28          <br>Jl. Contoh Alamat No. 123, Yogyakarta
29        </p>
30      </div>
31    @endif
32  </div>
33 </div>
```

Gambar 19 Frontend Payment

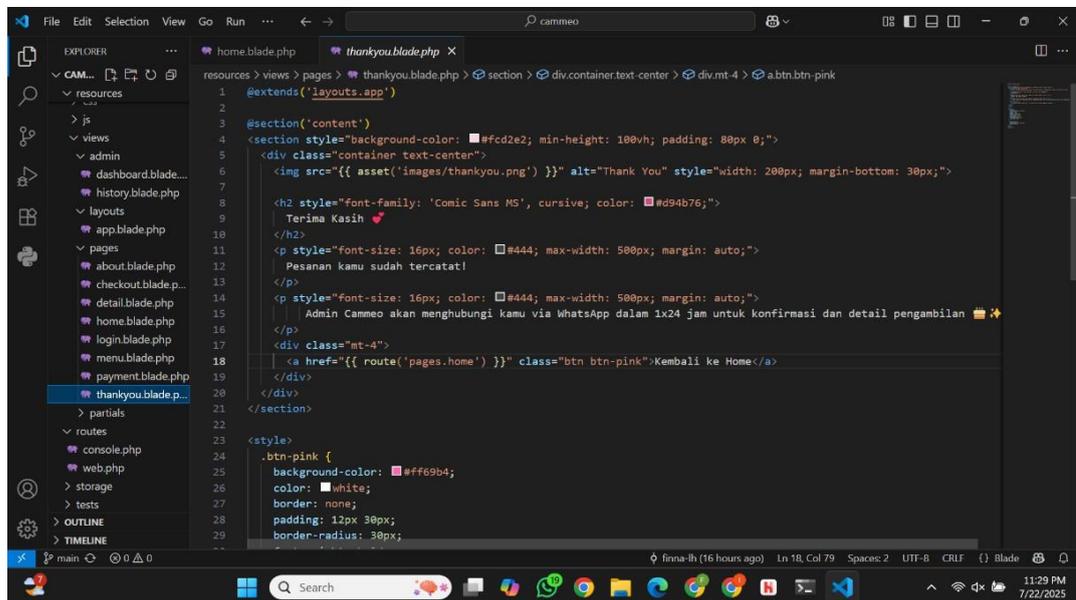
## About



```
resources > views > pages > CAMMEO > about.blade.php > section.about_section.layout_padding > div.container > div.row
1 @extends('layouts.app')
2
3 @section('content')
4
5 <!-- about section -->
6 <section class="about_section layout_padding">
7   <div class="container">
8     <div class="row">
9       <div class="col-md-6">
10        <div class="img-box">
11          
12        </div>
13      </div>
14      <div class="col-md-6">
15        <div class="detail-box">
16          <div class="heading_container">
17            <h2>We Are Cammeo</h2>
18          </div>
19          <p>
20            Cammeo siap manjain harimu dengan berbagai kue cantik ala bento Korea 🍰👉
21            Bisa request desain suka-suka, harganya terjangkau, dan rasanya nggak main-main enaknyal
22            Cocok banget buat kado, kejutan, atau ngemil estetik di hari spesialmu 🥰
23          </p>
24          <a href="">Read More</a>
25        </div>
26      </div>
27    </div>
28  </section>
29
```

Gambar 20 Frontend About Cammeo

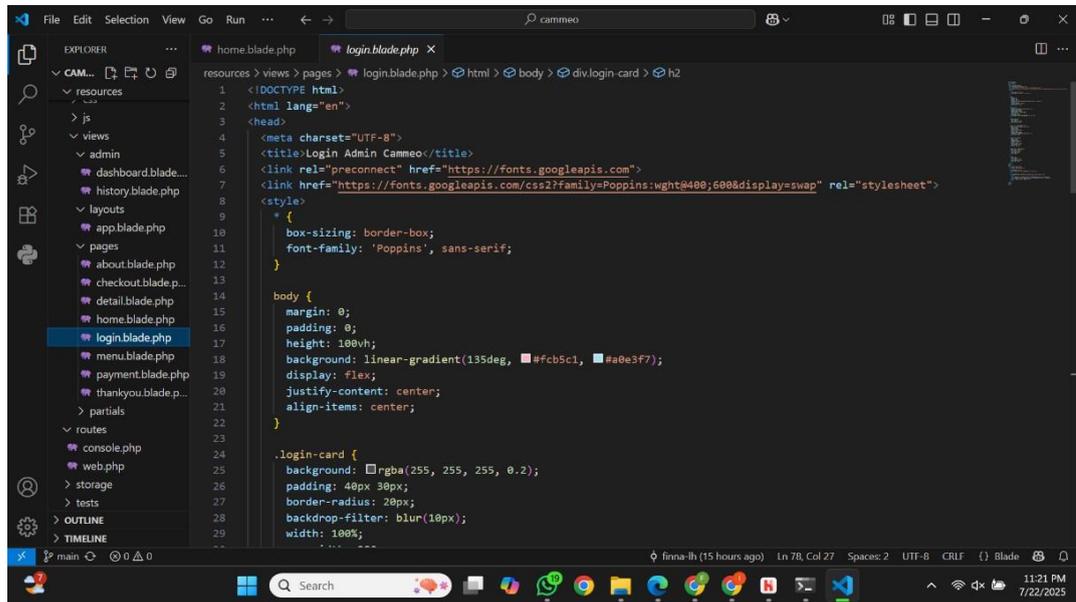
## Konfirmasi Pesanan



```
resources > views > pages > CAMMEO > thankyou.blade.php > section > div.container.text-center > div.mt-4 > a.btn.btn-pink
1 @extends('layouts.app')
2
3 @section('content')
4   <section style="background-color: #fcd2e2; min-height: 100vh; padding: 80px 0;">
5     <div class="container text-center">
6       
7       <h2 style="font-family: 'Comic Sans MS', cursive; color: #d94b76;">
8         Terima Kasih 🥰
9       </h2>
10      <p style="font-size: 16px; color: #444; max-width: 500px; margin: auto;">
11        Pesanan kamu sudah tercatat!
12      </p>
13      <p style="font-size: 16px; color: #444; max-width: 500px; margin: auto;">
14        Admin Cammeo akan menghubungi kamu via WhatsApp dalam 1x24 jam untuk konfirmasi dan detail pengambilan 📦👉
15      </p>
16      <div class="mt-4">
17        <a href="{{ route('pages.home') }}" class="btn btn-pink">Kembali ke Home</a>
18      </div>
19    </div>
20  </section>
21
22 <style>
23   .btn-pink {
24     background-color: #ff69b4;
25     color: white;
26     border: none;
27     padding: 12px 30px;
28     border-radius: 30px;
29
```

Gambar 21 Frontend Konfirmasi Pesanan

## Login admin

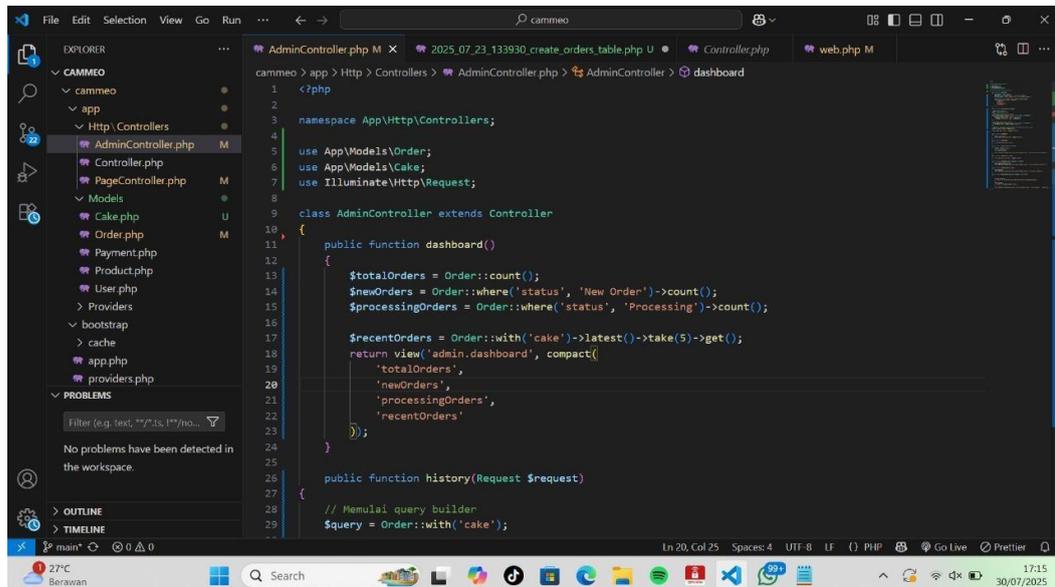


```
resources > views > pages > login.blade.php > html > body > div.login-card > h2
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Login Admin Cammeo</title>
6 <link rel="preconnect" href="https://fonts.googleapis.com">
7 <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap" rel="stylesheet">
8 <style>
9 *
10 {
11   box-sizing: border-box;
12   font-family: 'Poppins', sans-serif;
13 }
14
15 body {
16   margin: 0;
17   padding: 0;
18   height: 100vh;
19   background: linear-gradient(135deg, #fcb5c1, #a0e3f7);
20   display: flex;
21   justify-content: center;
22   align-items: center;
23 }
24
25 .login-card {
26   background: rgba(255, 255, 255, 0.2);
27   padding: 40px 30px;
28   border-radius: 20px;
29   backdrop-filter: blur(10px);
30   width: 100%;
31 }
```

Gambar 22 Frontend Admin Login

## 7. Pengembangan Back-end

### AdminController.php



```
AdminController.php M X
2025.07.23.133930.create_orders.table.php U
Controller.php
web.php M
cammeo > app > Http > Controllers > AdminController.php > AdminController > dashboard
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Order;
6 use App\Models\Cake;
7 use Illuminate\Http\Request;
8
9 class AdminController extends Controller
10 {
11     public function dashboard()
12     {
13         $totalOrders = Order::count();
14         $newOrders = Order::where('status', 'New Order')->count();
15         $processingOrders = Order::where('status', 'Processing')->count();
16
17         $recentOrders = Order::with('cake')->latest()->take(5)->get();
18         return view('admin.dashboard', compact([
19             'totalOrders',
20             'newOrders',
21             'processingOrders',
22             'recentOrders'
23         ]));
24     }
25
26     public function history(Request $request)
27     {
28         // Memulai query builder
29         $query = Order::with('cake');
```

Gambar 23 Backend Admin Controller

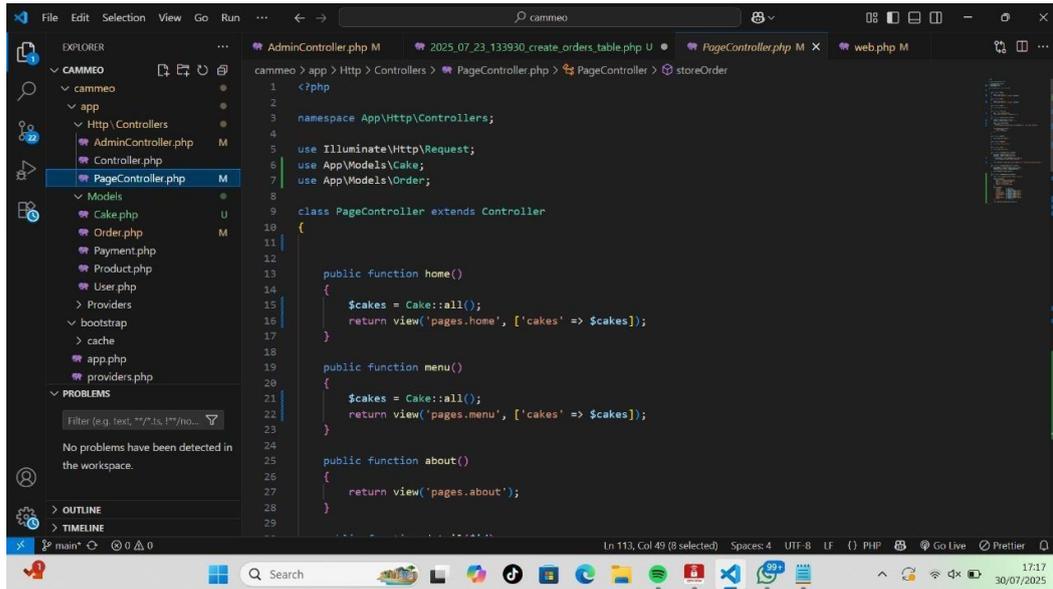
```
AdminController.php M X 2025_07_23_133930_create_orders_table.php U PageController.php M web.php M
cammeo > app > Http > Controllers > AdminController.php > AdminController > dashboard
9 class AdminController extends Controller
60 public function cakesStore(Request $request)
62     $newCake = $request->all();
63
64     Cake::create($newCake);
65
66     return redirect()->route('admin.cakes.index')->with('success', 'Kue baru berhasil ditambahkan!');
67 }
68
69 public function cakesEdit(Cake $cake)
70 {
71     return view('admin.cakes.edit', compact('cake'));
72 }
73
74 public function cakesUpdate(Request $request, Cake $cake)
75 {
76     $cake->update($request->all());
77     return redirect()->route('admin.cakes.index')->with('success', 'Data kue berhasil diperbarui!');
78 }
79
80 public function cakesDestroy(Cake $cake)
81 {
82     $cake->delete();
83     return redirect()->route('admin.cakes.index')->with('success', 'Kue berhasil dihapus!');
84 }
85
86 public function updateOrderStatus(Request $request, Order $order)
87 {
88
```

Gambar 24 Lanjutan 1 Backend AdminController

```
AdminController.php M X 2025_07_23_133930_create_orders_table.php U PageController.php M web.php M
cammeo > app > Http > Controllers > AdminController.php > AdminController > dashboard
9 class AdminController extends Controller
86 public function updateOrderStatus(Request $request, Order $order)
89
90     $request->validate([
91         'status' => 'required|string|in:New Order,Processing,Completed,Cancelled',
92     ]);
93
94     $order->update([
95         'status' => $request->input('status'),
96     ]);
97
98     return redirect()->route('admin.orders.history')->with('success', 'Status pesanan #' . $order->id .
99     );
100 }
101
```

Gambar 25 Lanjutan 2 Backend AdminController

## PagController.php



```
<?php
namespace App\Http\Controllers;

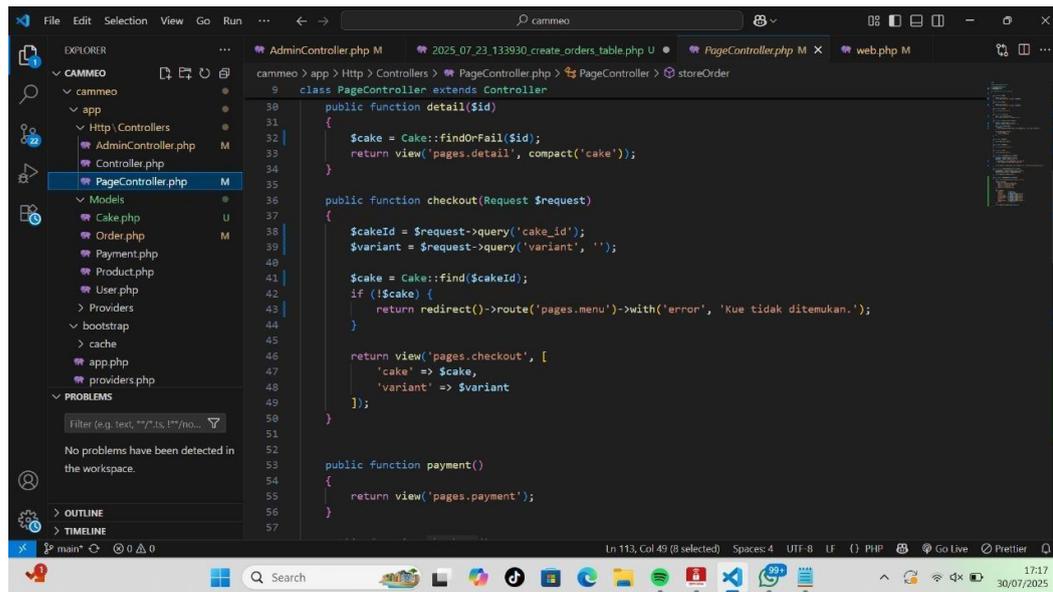
use Illuminate\Http\Request;
use App\Models\Cake;
use App\Models\Order;

class PageController extends Controller
{
    public function home()
    {
        $cakes = Cake::all();
        return view('pages.home', ['cakes' => $cakes]);
    }

    public function menu()
    {
        $cakes = Cake::all();
        return view('pages.menu', ['cakes' => $cakes]);
    }

    public function about()
    {
        return view('pages.about');
    }
}
```

Gambar 26 Backend PagController



```
class PageController extends Controller
{
    public function detail($id)
    {
        $cake = Cake::findOrFail($id);
        return view('pages.detail', compact('cake'));
    }

    public function checkout(Request $request)
    {
        $cakeId = $request->query('cake_id');
        $variant = $request->query('variant', '');

        $cake = Cake::find($cakeId);
        if (!$cake) {
            return redirect()->route('pages.menu')->with('error', 'Kue tidak ditemukan.');
```

Gambar 27 Lanjutan 1 Backend PagController

```

class PageController extends Controller
{
    public function thankyou()
    {
        return view('pages.thankyou');
    }

    public function login()
    {
        return view('pages.login');
    }

    public function handleLogin(Request $request)
    {
        $username = $request->input('username');
        $password = $request->input('password');

        if ($username === 'admin' && $password === 'admin') {
            return redirect()->route('admin.dashboard');
        }

        return redirect()->route('pages.login')->with('error', 'Username atau password salah!');
    }

    public function proceedToPayment(Request $request)
    {
        $paymentMethod = $request->input('payment_method');
        session(['payment_method' => $paymentMethod]);

        return redirect()->route('payment');
    }
}

```

Gambar 28 Lanjutan 2 Backend PagController

```

public function storeOrder(Request $request)
{
    $cake = Cake::findOrFail($request->input('cake_id'));

    $request->validate([
        'name' => 'required|string|max:255',
        'phone' => 'required|string|max:20',
        'address' => 'required|string',
        'date' => 'required|date',
        'payment' => 'required|string',
    ]);

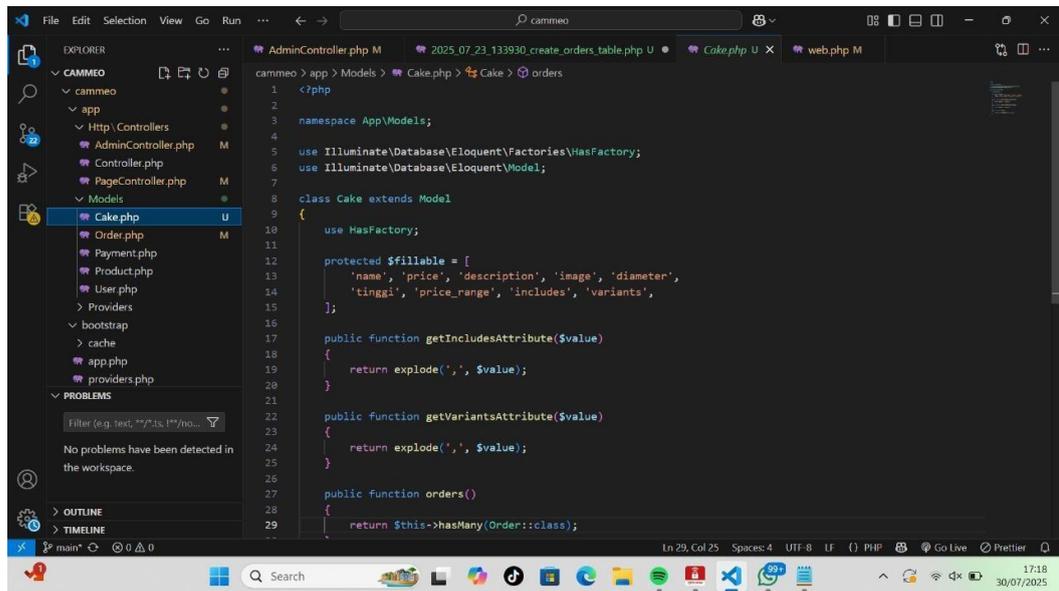
    Order::create([
        'cake_id' => $cake->id,
        'cake_name' => $cake->name,
        'variant' => $request->input('variant'),
        'customer_name' => $request->input('name'),
        'customer_phone' => $request->input('phone'),
        'customer_address' => $request->input('address'),
        'pickup_date' => $request->input('date'),
        'notes' => $request->input('note'),
        'payment_method' => $request->input('payment'),
        'status' => 'New Order'
    ]);

    return redirect()->route('pages.thankyou');
}

```

Gambar 29 Lanjutan 3 Backend PagController

## Cake.php

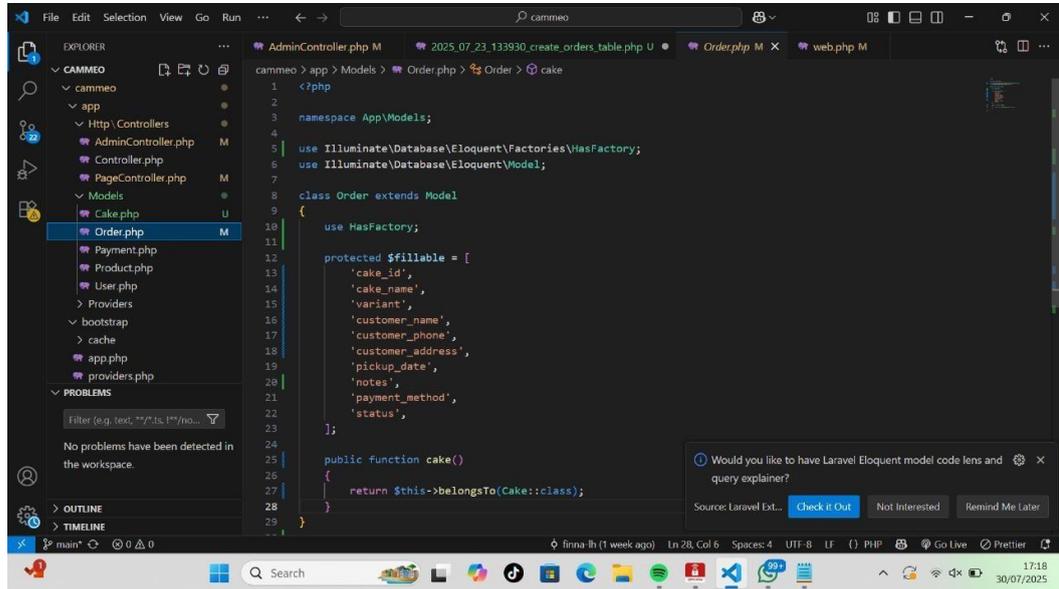


The screenshot shows the Visual Studio Code editor with the file explorer on the left. The 'Models' folder is expanded, and 'Cake.php' is selected. The main editor displays the PHP code for the Cake model, which extends the Illuminate\Database\Eloquent\Model class and uses the HasFactory trait. The fillable attributes include name, price, description, image, diameter, tinggi, price\_range, includes, and variants. The model includes methods for getIncludesAttribute, getVariantsAttribute, and orders.

```
<?php
1
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Cake extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'name', 'price', 'description', 'image', 'diameter',
14         'tinggi', 'price_range', 'includes', 'variants',
15     ];
16
17     public function getIncludesAttribute($value)
18     {
19         return explode(',', $value);
20     }
21
22     public function getVariantsAttribute($value)
23     {
24         return explode(',', $value);
25     }
26
27     public function orders()
28     {
29         return $this->hasMany(Order::class);
30     }
31 }
```

Gambar 30 Backend Menu Cake

## Order.php

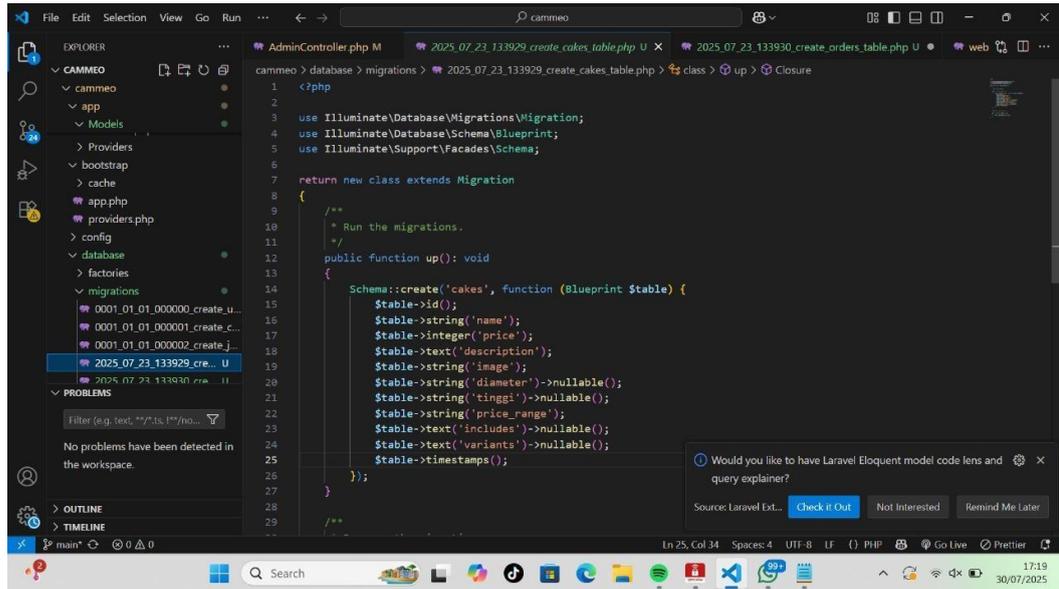


The screenshot shows the Visual Studio Code editor with the file explorer on the left. The 'Models' folder is expanded, and 'Order.php' is selected. The main editor displays the PHP code for the Order model, which extends the Illuminate\Database\Eloquent\Model class and uses the HasFactory trait. The fillable attributes include cake\_id, cake\_name, variant, customer\_name, customer\_phone, customer\_address, pickup\_date, notes, payment\_method, and status. The model includes a cake() method that returns a belongsTo relationship with the Cake model.

```
<?php
1
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Order extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'cake_id',
14         'cake_name',
15         'variant',
16         'customer_name',
17         'customer_phone',
18         'customer_address',
19         'pickup_date',
20         'notes',
21         'payment_method',
22         'status',
23     ];
24
25     public function cake()
26     {
27         return $this->belongsTo(Cake::class);
28     }
29 }
```

Gambar 31 Backend Order

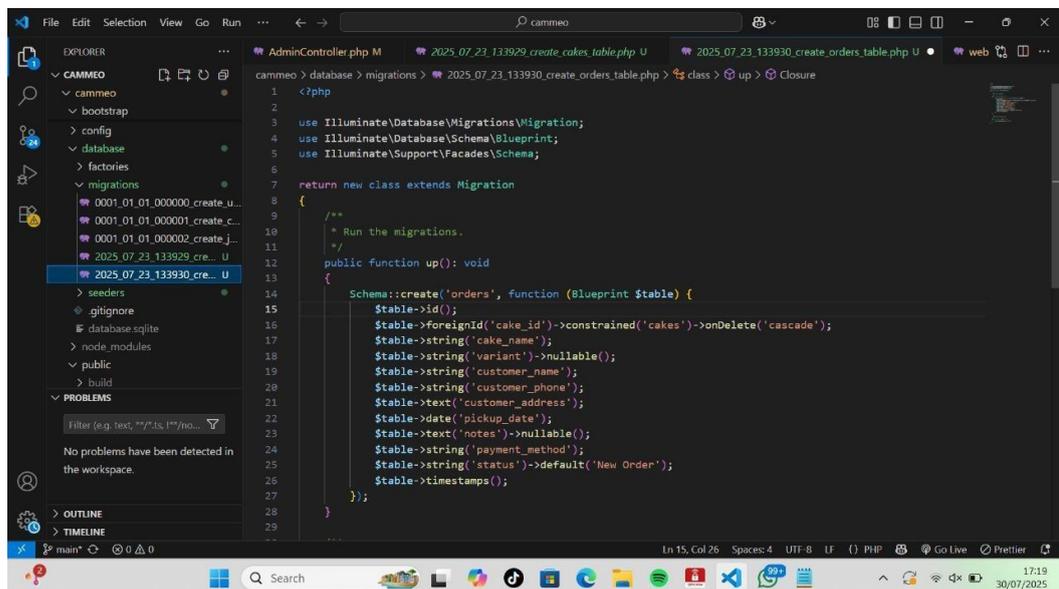
## migration table cake



```
AdminController.php M 2025_07_23_133929_create_cakes_table.php U 2025_07_23_133930_create_orders_table.php U web ...
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('cakes', function (Blueprint $table) {
15             $table->id();
16             $table->string('name');
17             $table->integer('price');
18             $table->text('description');
19             $table->string('image');
20             $table->string('diameter')->nullable();
21             $table->string('tinggi')->nullable();
22             $table->string('price_range');
23             $table->text('includes')->nullable();
24             $table->text('variants')->nullable();
25             $table->timestamps();
26         });
27     }
28 }
29 /**
```

Gambar 32 Backend Migrate table cake

## migration table orders



```
AdminController.php M 2025_07_23_133929_create_cakes_table.php U 2025_07_23_133930_create_orders_table.php U web ...
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('orders', function (Blueprint $table) {
15             $table->id();
16             $table->foreignId('cake_id')->constrained('cakes')->onDelete('cascade');
17             $table->string('cake_name');
18             $table->string('variant')->nullable();
19             $table->string('customer_name');
20             $table->string('customer_phone');
21             $table->text('customer_address');
22             $table->date('pickup_date');
23             $table->text('notes')->nullable();
24             $table->string('payment_method');
25             $table->string('status')->default('New Order');
26             $table->timestamps();
27         });
28     }
29 }
```

Gambar 33 Backend Migrate table order

## 8. Integrasi Sistem

Integrasi sistem dilakukan setelah pengembangan bagian front-end dan back-end selesai. Pada tahap ini, seluruh komponen sistem disatukan agar dapat berfungsi sebagai satu kesatuan yang utuh. Integrasi difokuskan pada penghubungan antarhalaman pengguna (seperti halaman pemesanan, detail produk, dan metode pembayaran) dengan logika bisnis dan basis data yang telah dikembangkan pada back-end. Misalnya, saat pengguna melakukan pemesanan melalui form di halaman front-end, data tersebut dikirim ke server dan disimpan dalam database melalui sistem backend. Selain itu, dilakukan integrasi pada dashboard admin agar admin dapat mengakses dan mengelola data produk serta pesanan secara real-time. Proses ini juga mencakup penyambungan metode pembayaran yang mendukung opsi QRIS dan COD, serta memastikan komunikasi data antara halaman pelanggan dan panel admin berjalan dengan lancar. Skema database yang telah dirancang sebelumnya juga mulai diimplementasikan secara penuh agar semua proses CRUD dapat berjalan tanpa hambatan.

## **9. Pengujian Fungsional**

Pengujian fungsional dilakukan untuk memastikan bahwa setiap fitur dalam sistem bekerja sesuai dengan kebutuhan dan skenario penggunaan yang telah ditentukan. Seluruh alur penggunaan sistem diuji mulai dari proses pemesanan oleh pelanggan, pemilihan metode pembayaran, hingga pengelolaan pesanan oleh admin. Setiap komponen diuji secara menyeluruh untuk memastikan tidak terdapat error atau fungsi yang tidak berjalan. Uji coba dilakukan oleh tim internal serta pihak mitra usaha untuk mendapatkan masukan langsung dari sisi pengguna. Hasil dari pengujian ini digunakan sebagai dasar dalam melakukan perbaikan atau penyempurnaan sistem.

## **10. Perbaikan Bug**

Setelah dilakukan pengujian, ditemukan beberapa bug yang perlu diperbaiki, seperti kesalahan validasi form input, tampilan halaman yang tidak responsif di beberapa perangkat, serta data yang tidak tersimpan dengan benar dalam database. Tim pengembang melakukan proses debugging untuk mengidentifikasi penyebab utama dari setiap bug tersebut. Setelah itu, dilakukan perbaikan di sisi kode program, baik pada backend (Laravel) maupun frontend (HTML, CSS, dan JavaScript). Perbaikan ini diikuti dengan pengujian ulang agar sistem benar-benar stabil dan bebas dari kesalahan sebelum diluncurkan ke pengguna akhir.

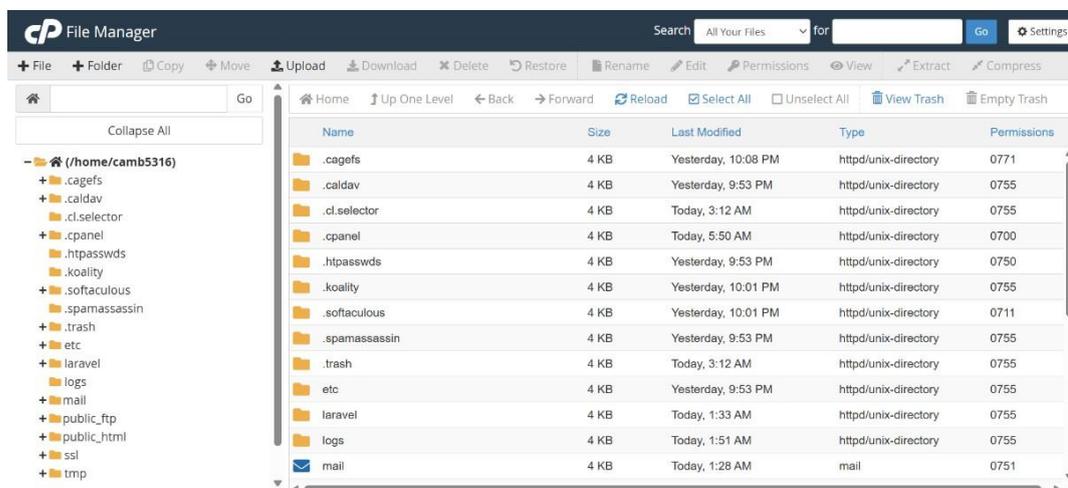
## **11. Hosting**

Proses deploy dilakukan dengan mengunggah semua folder website yang telah selesai dibuat ke server hosting yang dipilih. Website di-deploy melalui layanan Rumahweb, yang dikenal sebagai salah satu penyedia hosting terpercaya di Indonesia. Proses ini melibatkan transfer semua file, termasuk file HTML, CSS,

JavaScript, gambar, dan file lain yang dibutuhkan untuk menjalankan website secara online. Setelah semua dokumen berhasil diunggah, website akan dapat diakses oleh pengguna melalui internet. Biaya yang dikeluarkan untuk melakukan deployment ini adalah sebesar 53.234 rupiah, yang mencakup biaya Register Domain + ID Protection dan Hosting | Entry Hosting yang diperlukan. Dengan demikian, website yang telah selesai dibuat dapat diakses oleh pengguna secara luas, mendukung tujuan bisnis atau proyek yang dijalankan. Proses deployment yang dilakukan dengan baik akan memastikan website bisa diakses semua pengguna dan berfungsi lebih optimal.

**Keranjang Belanja**  
Periksa kembali pesanan anda dan checkout

Keterangan	Satuan	Periode	Total
Register Domain + ID Protection Domain: apoteker-ld.my.id	12,000 IDR ID Protection + 10,000 IDR	1 Tahun	22,000 IDR
Hosting   Entry Hosting Domain: apoteker-ld.my.id	25,000 IDR per Bulan	1 Bulan	25,000 IDR
<b>Total order</b>			<b>47,000 IDR</b>
Subtotal			47,000 IDR
PPN			5,170 IDR
Fee			1,064 IDR
<b>Total</b>			<b>53,234 IDR</b>



## 12. Pemeliharaan

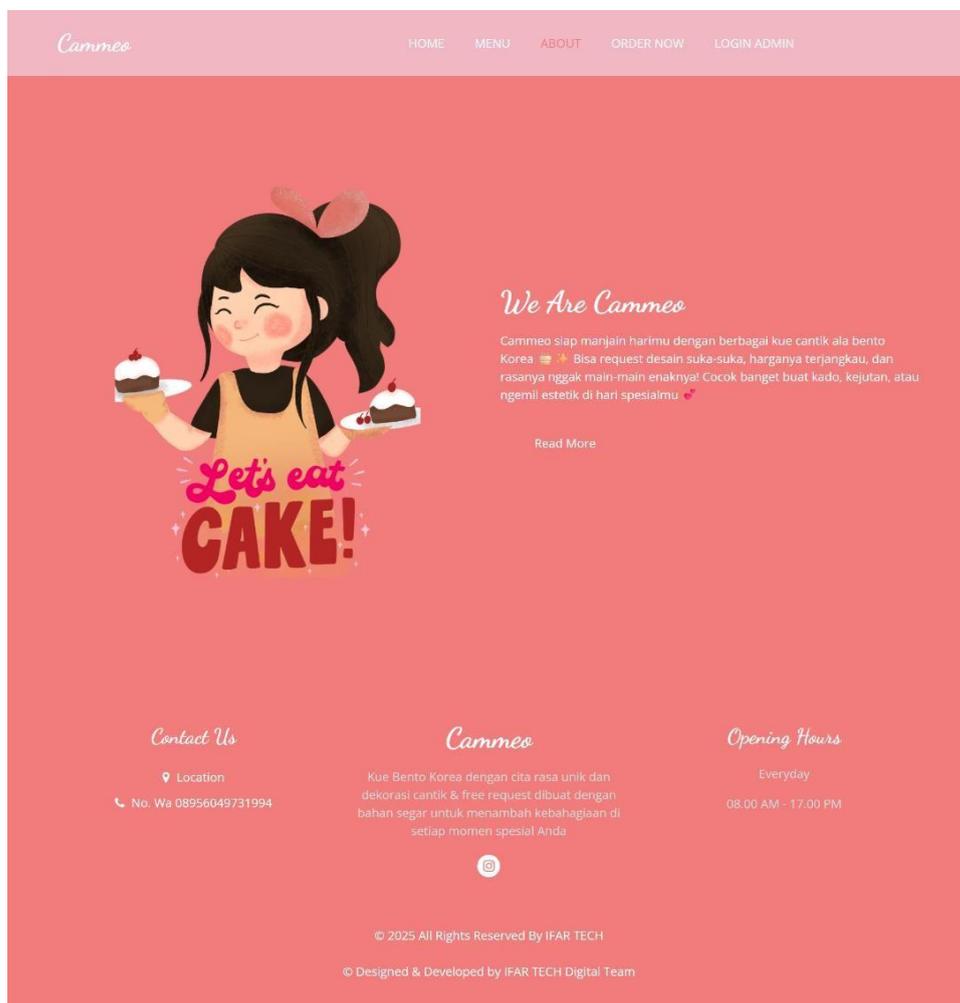
Proses pemeliharaan sistem dilakukan secara berkala untuk memastikan website penjualan Cammeo tetap berjalan dengan optimal setelah diterapkan. Pemantauan dilakukan terhadap fitur utama seperti pemesanan produk, metode pembayaran, serta dashboard admin guna mendeteksi potensi gangguan teknis sejak dini. Tim juga rutin melakukan pengecekan dan perbaikan bug yang mungkin muncul akibat interaksi pengguna atau kondisi teknis lainnya. Selain itu, pembaruan konten dilakukan untuk menyesuaikan informasi produk, harga, dan deskripsi agar tetap relevan. Umpan balik dari mitra toko Cammeo dimanfaatkan

sebagai dasar dalam melakukan perbaikan dan penyesuaian fitur. Seluruh kegiatan pemeliharaan ini dilaksanakan selama tiga bulan setelah sistem diimplementasikan, sesuai dengan kesepakatan awal dalam ruang lingkup dukungan proyek.

## B. Realisasi Hasil Pekerjaan

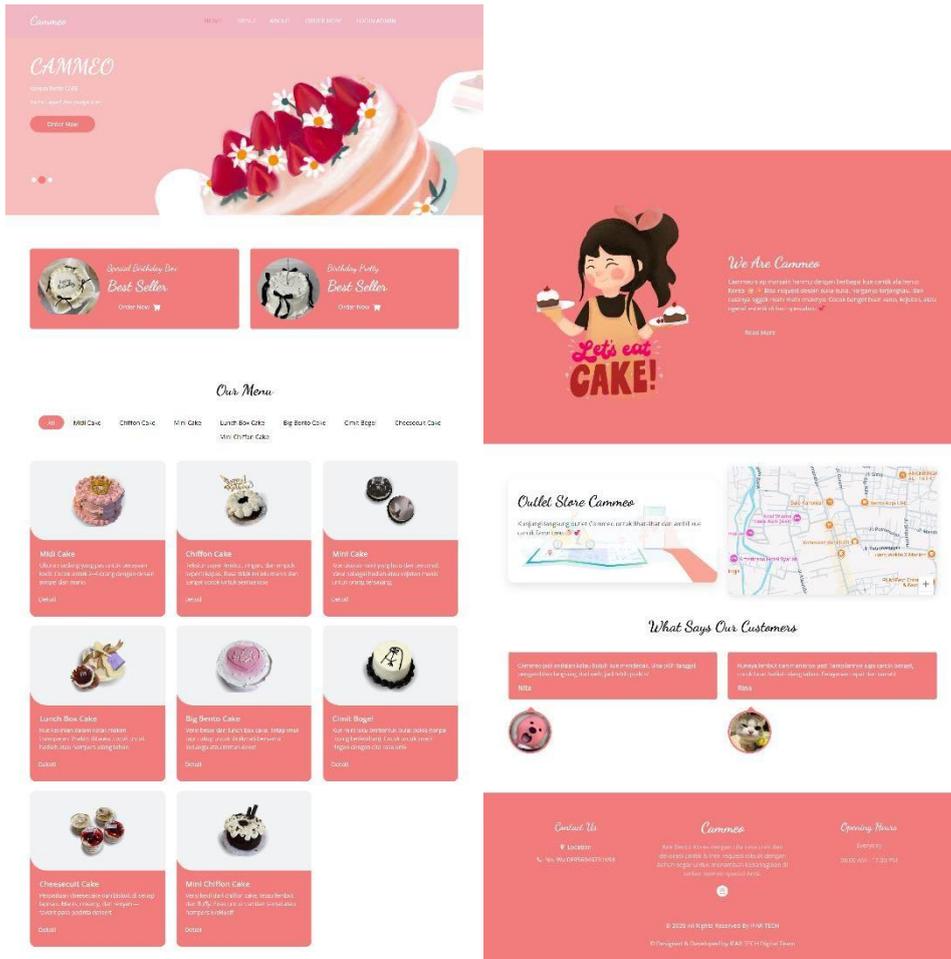
### 1. UI/UX Designer

#### About



Gambar 34 UI/UX About Cammeo

#### Home



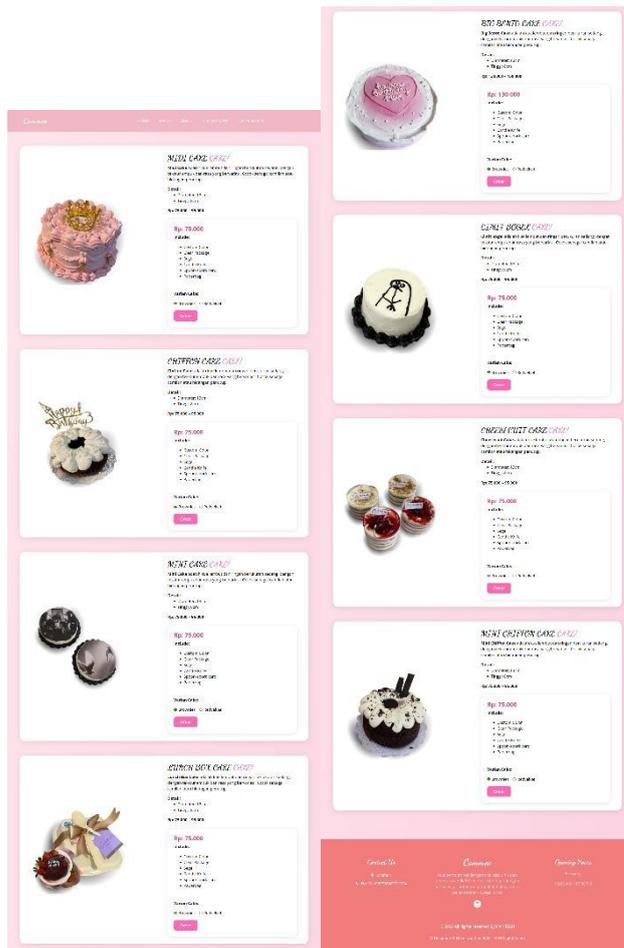
Gambar 35 UI/UX Home

# Menu



Gambar 36 UI/UX Menu

## Detail Menu



Gambar 37 UI/UX Detail Menu

## Order Now

*Cammeo*      HOME   MENU   ABOUT   **ORDER NOW**   LOGIN ADMIN

### Checkout Pemesanan

#### Informasi Pembeli

Nama

Telepon / WA

Alamat

Tanggal Pengambilan

Catatan Tambahan

Metode Pembayaran  
 QRIS (Scan QR Code)  
 Cash on Delivery (COD)

Lanjut ke Pembayaran

#### Ringkasan Pesanan

**Varian:**

 **Chiffon Cake**  
Tekstur super lembut, ringan, dan empuk seperti kapas. Rasa tidak terlalu manis dan sangat cocok untuk semua usia  
Rp 160.000

---

**Total** **Rp 160.000**

### Contact Us

📍 Location  
☎ No. Wa 08956049731994

### Cammeo

Kue Bento Korea dengan cita rasa unik dan dekorasi cantik & free request dibuat dengan bahan segar untuk menambah kebahagiaan di setiap momen spesial Anda

📷

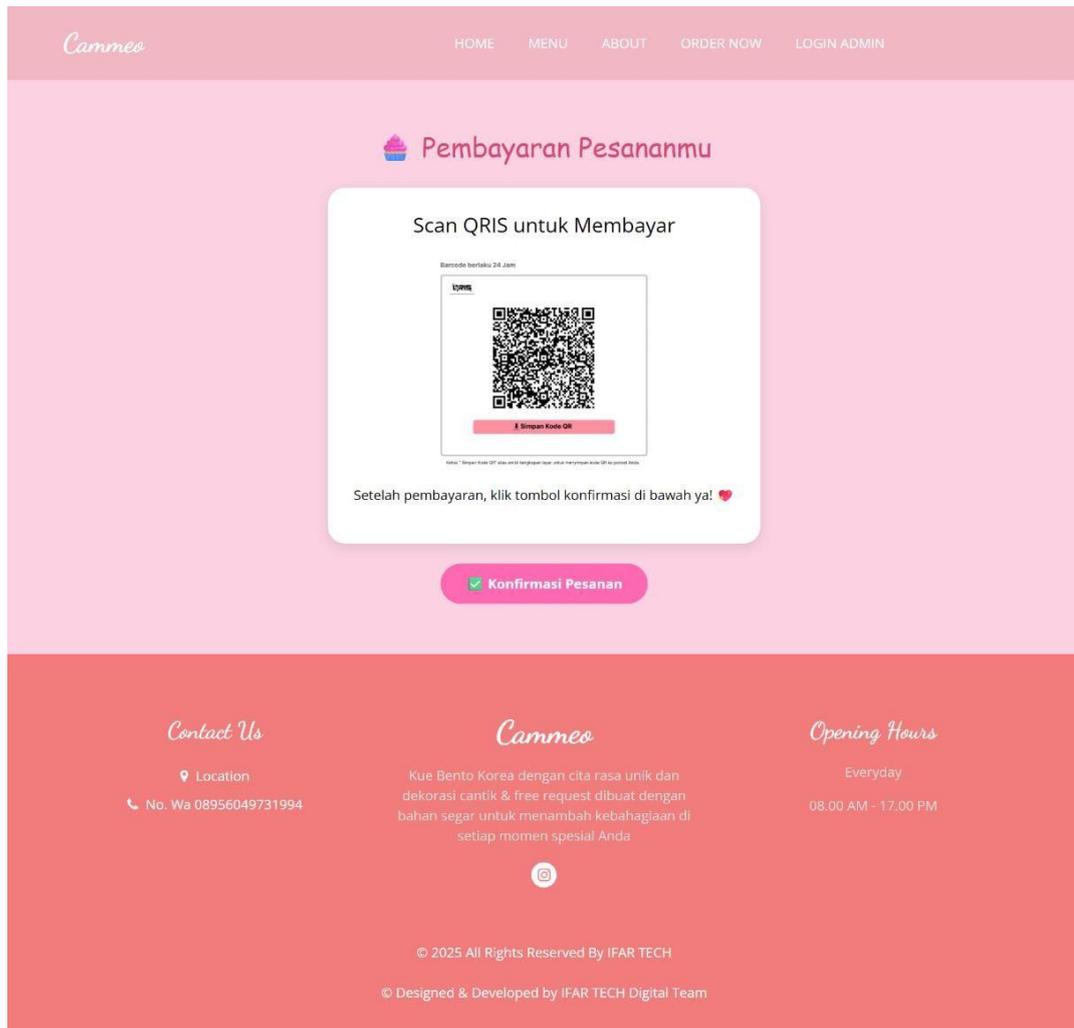
### Opening Hours

Everyday  
08.00 AM - 17.00 PM

© 2025 All Rights Reserved By IFAR TECH  
© Designed & Developed by IFAR TECH Digital Team

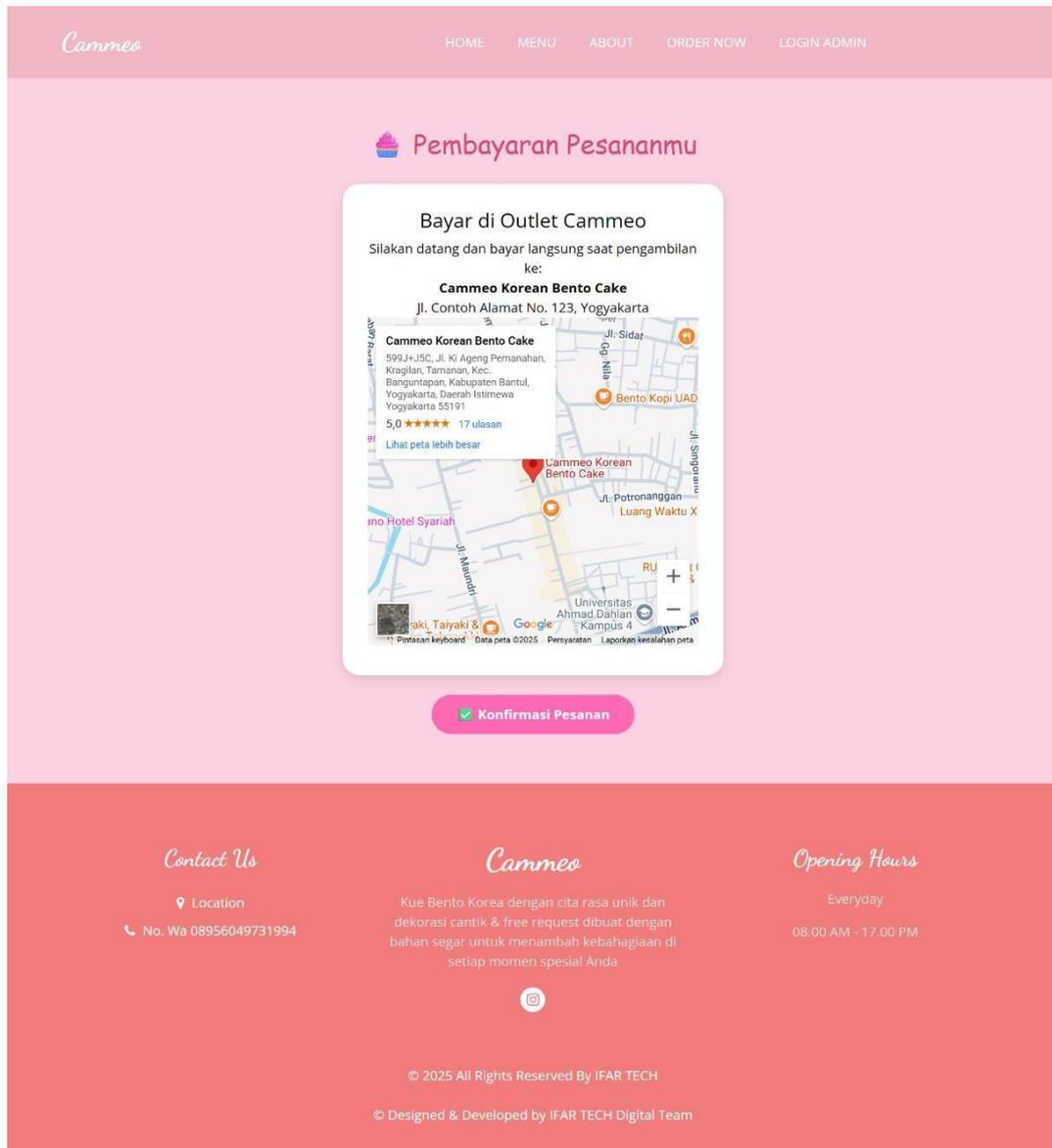
Gambar 38 UI/UX Order Now

## Pyment Qris



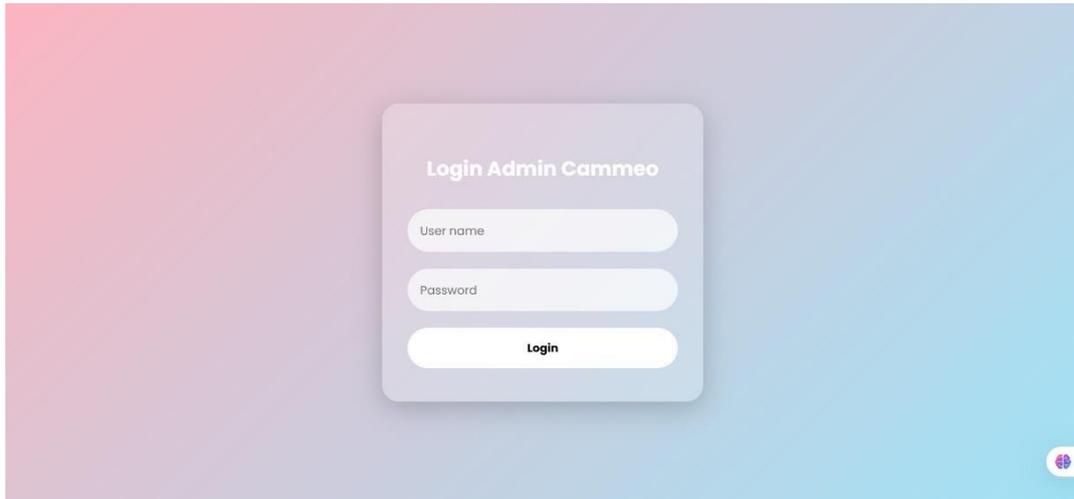
Gambar 39 UI/UX Payment Qris

## Payment Cash



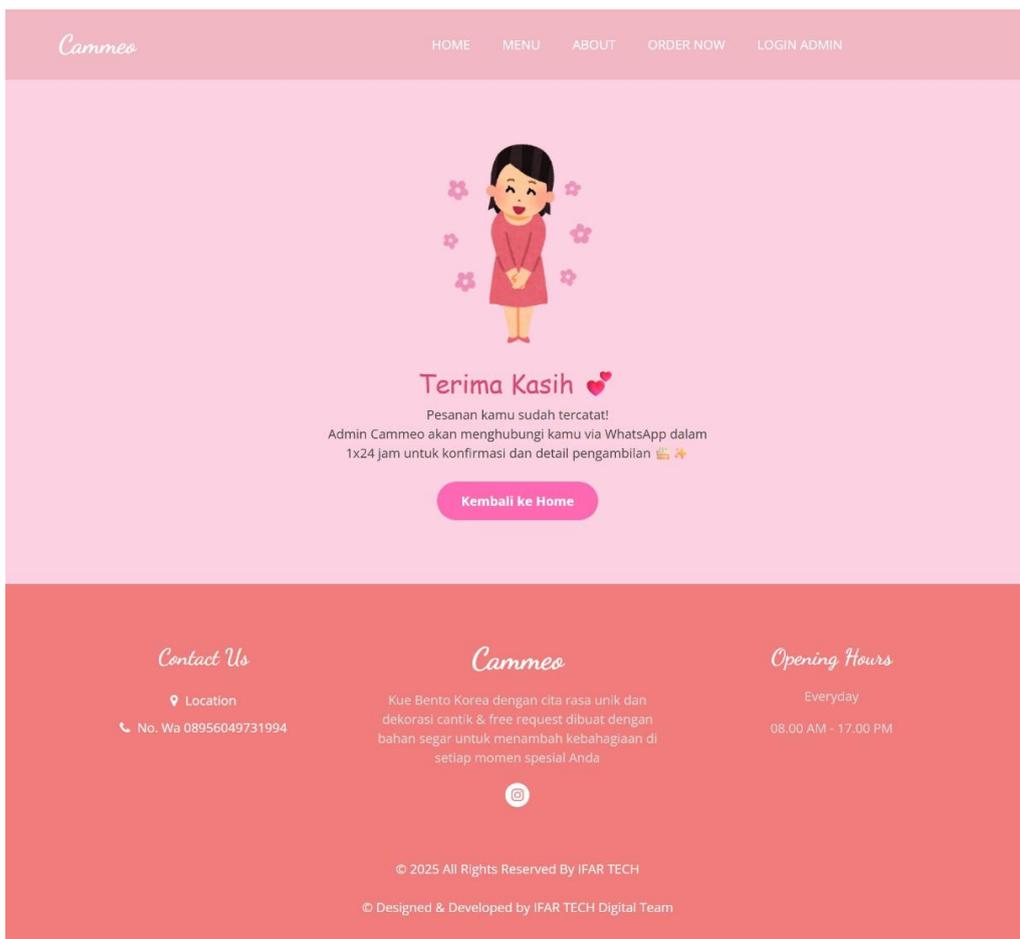
Gambar 40 UI/UX Payment Cash

## Login Admin



Gambar 41 UI/UX Login admin

## Konfirmasi Pesanan



Gambar 42 UI/UX Konfirmasi Pesanan

## 2. Backend

### Admin Dashboard

*Cammeo* HOME MENU ABOUT ORDER NOW LOGIN ADMIN

*Admin Dashboard*

Total Pesanan **6**

Pesanan Baru **2**

Sedang Diproses **1**

**Manajemen Menu**  
Tambah, edit, atau hapus kue dari daftar menu.  
[Kelola Kue](#)

**Pesanan Terbaru**

ID	Pelanggan	Kue	Tanggal Pesan	Status
6	alya	Chiffon Cake	01 Aug 2025, 18:36	<a href="#">New Order</a>
5	alya	Chiffon Cake	01 Aug 2025, 18:34	<a href="#">New Order</a>
4	alya	Midi Cake	01 Aug 2025, 16:21	<a href="#">New Order</a>
3	alya	Midi Cake	01 Aug 2025, 16:21	<a href="#">New Order</a>
2	alya	Midi Cake	01 Aug 2025, 13:57	<a href="#">New Order</a>

[Lihat Semua Riwayat Pesanan](#)

*Contact Us*  
Location  
No. Wa 08956049731994

*Cammeo*  
Kue Bento Korea dengan cita rasa unik dan dekorasi cantik & free request dibuat dengan bahan segar untuk menambah kebahagiaan di setiap momen spesial Anda

*Opening Hours*  
Everyday  
08.00 AM - 17.00 PM

© 2025 All Rights Reserved By IFAR TECH  
© Designed & Developed by IFAR TECH Digital Team

Gambar 43 Backend Dashboard Admin

## Button Kelola Kue

*Cammeo* HOME MENU ABOUT ORDER NOW LOGIN ADMIN

*Kelola Menu Kue* [Tambah Kue Baru](#)

ID	Nama Kue	Harga	Aksi
1	Midi Cake	Rp 140,000	<a href="#">Edit</a> <a href="#">Hapus</a>
2	Chiffon Cake	Rp 160,000	<a href="#">Edit</a> <a href="#">Hapus</a>
3	Mini Cake	Rp 70,000	<a href="#">Edit</a> <a href="#">Hapus</a>
4	Lunch Box Cake	Rp 85,000	<a href="#">Edit</a> <a href="#">Hapus</a>
5	Big Bento Cake	Rp 180,000	<a href="#">Edit</a> <a href="#">Hapus</a>
6	Cimit Bogel	Rp 50,000	<a href="#">Edit</a> <a href="#">Hapus</a>
7	Cheesecuit Cake	Rp 120,000	<a href="#">Edit</a> <a href="#">Hapus</a>
8	Mini Chiffon Cake	Rp 75,000	<a href="#">Edit</a> <a href="#">Hapus</a>

*Contact Us*  
Location  
No. Wa 08956049731994

*Cammeo*  
Kue Bento Korea dengan cita rasa unik dan dekorasi cantik & free request dibuat dengan bahan segar untuk menambah kebahagiaan di setiap momen spesial Anda

*Opening Hours*  
Everyday  
08.00 AM - 17.00 PM

© 2025 All Rights Reserved By IFAR TECH  
© Designed & Developed by IFAR TECH Digital Team

Gambar 44 Backend Kelola Kue

## Button edit pada kelola kue

*Cammeo* HOME MENU ABOUT ORDER NOW LOGIN ADMIN

### Edit Kue: Midi Cake

Nama Kue:

Diameter:

Harga (Angka saja):

Tinggi:

Kisaran Harga (Teks):

Includes (Pisahkan dengan koma):

Nama File Gambar:

Varian (Pisahkan dengan koma):

Deskripsi:

*Contact Us*  
Location  
No. Wa 08956049731994

*Cammeo*  
Kue Bento Korea dengan cita rasa unik dan dekorasi cantik & free request dibuat dengan bahan segar untuk menambah kebahagiaan di setiap momen spesial Anda

*Opening Hours*  
Everyday  
08.00 AM - 17.00 PM

© 2025 All Rights Reserved By IFAR TECH  
© Designed & Developed by IFAR TECH Digital Team

Gambar 45 Backend BUTton Kelola Kue

## Button hapus pada kelola kue

*Cammeo* cammeo.my.id says Anda yakin ingin menghapus kue ini? OK Cancel

*Kelola Menu Kue*

ID	Nama Kue	Harga	Aksi
1	Midi Cake	Rp 140,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
2	Chiffon Cake	Rp 160,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
3	Mini Cake	Rp 70,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
4	Lunch Box Cake	Rp 85,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
5	Big Bento Cake	Rp 180,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
6	Cimit Bogel	Rp 50,000	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 46 Backend Button Hapus

## Button lihat semua riwayat pesanan

*Cammeo* HOME MENU ABOUT ORDER NOW LOGIN ADMIN

*Riwayat Pesanan*

Cari nama pelanggan/kue  Semua Status

No	Pelanggan	Kue	Jumlah	Tanggal Pesan	Ambil	Status	Aksi
1	alya 081273076223	Chiffon Cake	1	01/08/2025 18:36	07 Aug	Pesanan Baru	<input type="button" value="Detail"/> <input type="button" value="Ubah"/>
2	alya 081273076223	Chiffon Cake	1	01/08/2025 18:34	07 Aug	Pesanan Baru	<input type="button" value="Detail"/> <input type="button" value="Ubah"/>
3	alya 081275627711	Midi Cake	1	01/08/2025 16:21	29 Aug	Dibatalkan	<input type="button" value="Detail"/>
4	alya 081275627711	Midi Cake	1	01/08/2025 16:21	02 Aug	Sedang Diproses	<input type="button" value="Detail"/> <input type="button" value="Ubah"/>
5	alya 081273076223	Midi Cake	1	01/08/2025 13:57	09 Aug	Selesai	<input type="button" value="Detail"/>
6	alya 081275627711	Big Bento Cake	1	24/07/2025 11:43	25 Jul	Selesai	<input type="button" value="Detail"/>

*Contact Us*  
Location  
No. Wa 08956049731994

*Cammeo*  
Kue Bento Korea dengan cita rasa unik dan dekorasi cantik & free request dibuat dengan bahan segar untuk menambah kebahagiaan di setiap momen spesial Anda

*Opening Hours*  
Everyday  
08.00 AM - 17.00 PM

© 2025 All Rights Reserved By IFAR TECH  
© Designed & Developed by IFAR TECH Digital Team

Gambar 47 Backend Lihat History

### 3. Testing

ID	Test Case	Deskripsi	Langkah	Input	Output yang diharapkan	Catatan	Hasil		Hasil		Hasil	
							Valid	Tidak	Valid	Tidak	Valid	Tidak
TC-01	Validasi Form Login (Input Valid)	Menguji apakah login berhasil dengan menggunakan kombinasi username dan password yang valid.	1. Masukkan username yang valid (misalnya:user123) 2. Masukkan password yang valid (misalnya>Password123). 3. Klik tombol "Login"	Username: user123, Password: Password123	Pengguna berhasil login dan diarahkan ke halaman utama/dashboard sesuai level user	Pengujian dasar untuk memastikan login berfungsi dengan kombinasi username dan password yang benar. pastikan data sudah ada di DB.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-02	Validasi Form Login (Input Invalid)	Menguji apakah login gagal dengan kombinasi username atau password yang salah.	1. Masukkan username yang valid (misalnya:user123). 2. Masukkan password yang salah (misalnya:Wrong34) 3. Klik tombol "Login".	Username: user123,	Pesan kesalahan yang password salah muncul. Pengguna tidak dapat login.	Pastikan sistem kesalahan yang sesuai tanpa mengungkapkan apakah masalahnya ada pada username atau password.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-03	Validasi Password Minimum Length	Menguji apakah sistem memvalidasi panjang password yang sesuai (misalnya, minimal 8 karakter).	1. Masukkan username yang valid (misalnya: user123). 2. Masukkan password yang kurang dari 8 karakter (misalnya:Pass12). 3. Klik tombol "Login".	Username: user123, Password: Pass12	Pesan kesalahan yang menyatakan bahwa password harus memiliki panjang minimal 8 karakter muncul.	Pastikan sistem tidak menerima password yang tidak memenuhi kriteria panjang minimum.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-04	Validasi Keamanan (Brute Force Protection)	Menguji apakah sistem memiliki perlindungan terhadap percakapan login yang gagal berulang (brute force)	1. Masukkan username yang valid (misalnya:user123) 2. Masukkan password yang salah sebanyak 5 kali berturut-turut.	Username: user123, Password: WrongPass (dimasukkan 5 kali)	Akun terkunci atau diberi waktu tunggu sebelum percakapan login berikutnya diperbolehkan. Pesan keamanan muncul.	Pastikan sistem memiliki proteksi terhadap percakapan login yang gagal berulang.	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-05	Validasi Penggunaan Karakter Khusus dalam Username dan Password	Menguji apakah sistem menerima karakter khusus dalam username dan password.	1. Masukkan username dengan karakter khusus (misalnya: user@123).	Username: user@123, Password: P@sswOrd!	Login berhasil dan pengguna dapat masuk ke sistem tanpa masalah.	Pastikan bahwa karakter khusus diterima jika tidak ada batasan pada input tersebut.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-06	Pilih Kue	Menguji pemilihan jenis kue	1. Buka halaman katalog kue 2. Pilih salah satu kue 3. Klik detail	Klik salah satu kue	Halaman detail kue muncul dengan informasi lengkap	Pastikan data sesuai database	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-07	Pilih Kue Invalid	Menguji pemilihan kue yang sudah dihapus dari database	1. Cari kue yang sudah dihapus dari database 2. Klik detail	Kue tidak tersedia	Pesan error: "Kue tidak tersedia"	Pastikan sistem menampilkan error yang tepat	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC - 08	Tambah ke Keranjang	Menguji menambahkan kue ke keranjang belanja	1. Pilih kue 2. Klik Tambah ke Keranjang	Kue yang dipilih	Kue masuk ke keranjang belanja	Pastikan jumlah item bertambah	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-09	Update Jumlah Pesanan	Menguji perubahan jumlah kue di keranjang	1. Buka keranjang belanja 2. Ubah jumlah kue (misalnya dari 1 ke 3) 3. Klik Update	Jumlah produk diubah	Jumlah total harga otomatis diperbarui	Pastikan stok tersedia cukup	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-10	Hapus Produk dari Keranjang	Menguji penghapusan produk dari keranjang	1. Buka keranjang 2. Klik tombol Hapus di salah satu produk	Klik hapus produk	Produk hilang dari daftar keranjang	Total harga diperbarui	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-11	Checkout Valid	Menguji proses checkout dengan data alamat lengkap	1. Buka keranjang belanja 2. Klik Checkout 3. Masukkan alamat pengiriman lengkap 4. Klik Konfirmasi	Data alamat lengkap	Pesanan berhasil dibuat dan nomor pesanan muncul	Field wajib harus terisi	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-12	Checkout Invalid	Menguji checkout dengan data alamat kosong	1. Buka keranjang belanja 2. Klik Checkout 3. Kosongkan alamat 4. Klik Konfirmasi	Data alamat kosong	Pesan error: "Alamat tidak boleh kosong"	Pastikan validasi input berjalan	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-13	Pembayaran Valid	Menguji pembayaran dengan metode transfer bank	1. Pilih metode transfer bank 2. Upload bukti pembayaran valid 3. Klik Konfirmasi	Bukti transfer	Status pesanan berubah menjadi "Menunggu Konfirmasi Admin"	Pastikan file upload diterima	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TC-14	Pembayaran Invalid	Menguji pembayaran dengan file bukti tidak sesuai format	1. Pilih metode transfer bank 2. Upload file bukan bukti transfer (misalnya file txt kosong) 3. Klik Konfirmasi	File tidak sesuai	Pesan error: "Format file tidak valid"	Pastikan hanya file gambar/PDF diterima	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TC-15	Riwayat Pesanan	Menguji apakah riwayat pesanan tampil dengan benar	1. Login ke akun 2. Buka menu Riwayat Pesanan	Akun dengan minimal 1 pesanan	Daftar pesanan sebelumnya ditampilkan	Pastikan status sesuai database	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

- Total Case = 15
- Valid = 6
- Tidak valid = 9

⇒ Valid =  $(6/15) \times 100\% = 40\%$

⇒ Tidak valid =  $(9/15) \times 100\% = 60\%$

Berdasarkan hasil uji coba yang dilakukan terhadap 15 skenario test case pada sistem pemesanan kue, diperoleh hasil bahwa sebanyak 6 test case atau sebesar 40% berjalan dengan “Valid”, sedangkan 9 test case atau 60% lainnya dinyatakan “Tidak Valid”. Persentase keberhasilan pengujian yang masih berada di angka 40% menunjukkan bahwa sistem masih memerlukan perbaikan lebih lanjut. Beberapa fitur inti, khususnya pada bagian pemesanan, validasi data, serta proses checkout, belum dapat berfungsi sesuai dengan kebutuhan pengguna. Hal ini menandakan bahwa implementasi sistem masih perlu dilakukan debugging, optimalisasi logika, dan penguatan validasi input agar sistem lebih andal digunakan dalam skenario nyata.

#### **A. Keberlanjutan Proyek**

Proyek yang dikembangkan selama tiga bulan lamanya akhirnya mencapai titik penyelesaian. Seluruh tahapan, dari perencanaan hingga eksekusi, telah dilalui dengan dedikasi dan kerja keras dari semua pihak yang terlibat. Penyelesaian ini juga menandai berakhirnya periode garansi yang telah disepakati melalui Memorandum of Understanding (MOU). Dalam masa garansi ini, setiap masalah atau kerusakan yang terjadi telah ditangani dengan cepat dan profesional sesuai dengan ketentuan yang telah ditetapkan. Dengan berakhirnya masa garansi ini, diharapkan proyek ini dapat beroperasi dengan lancar dan memberikan manfaat sesuai dengan tujuan awal pengembangannya.

## **BAB IV**

### **PENUTUP**

#### **A. Kesimpulan**

Proyek pembuatan website penjualan Cammeo berhasil dilaksanakan sesuai dengan rencana yang telah ditetapkan. Website ini dirancang untuk menjawab kebutuhan digitalisasi proses pemesanan pada toko kue Cammeo, khususnya dalam memudahkan pelanggan dalam melihat katalog, melakukan pemesanan, memilih metode pembayaran, serta mempermudah pihak admin dalam mengelola produk dan pesanan.

Seluruh tahapan proyek, mulai dari analisis kebutuhan, desain UI/UX, pengembangan sistem front-end dan back-end, hingga pengujian dan hosting, telah terlaksana dengan baik berkat kerja sama tim yang solid dan keterlibatan aktif dari mitra usaha.

Melalui sistem ini, toko kue Cammeo diharapkan dapat meningkatkan efisiensi operasional, memperluas jangkauan pasar, serta meningkatkan kepuasan pelanggan. Selain itu, proyek ini juga memberikan pengalaman nyata dalam manajemen proyek teknologi informasi, mulai dari tahap perencanaan hingga serah terima produk.

#### **B. Saran**

Untuk pengembangan lebih lanjut, sistem ini dapat ditingkatkan melalui penambahan fitur pembayaran otomatis dengan integrasi payment gateway guna meningkatkan efisiensi transaksi, serta penerapan sistem keamanan yang lebih kuat untuk melindungi data pengguna dan transaksi dari potensi risiko siber. Selain itu, fitur notifikasi real-time melalui email atau WhatsApp dapat diterapkan untuk memberi informasi status pesanan kepada pelanggan secara langsung. Mengingat tingginya penggunaan perangkat mobile, pengembangan versi aplikasi mobile juga

layak dipertimbangkan agar jangkauan pengguna semakin luas. Terakhir, evaluasi dan pemeliharaan berkala perlu dilakukan agar performa website tetap optimal dan terus berkembang sesuai kebutuhan serta masukan dari pengguna.

## DAFTAR PUSTAKA

L. H. Inc., "Instalation," 2024. [Online]. Available: <https://laravel.com/docs/10.x>.

Boostrap, "Boostrap," 2024. [Online]. Available:  
<https://getbootstrap.com/docs/5.0/getting-started/introduction/>.

Laravel, "Eloquent: Getting Started," 2024. [Online]. Available:  
<https://laravel.com/docs/11.x/eloquent>.

L. Indonesia, "laravel," 2024. [Online]. Available: <https://medium.com/laravel-web-id>.

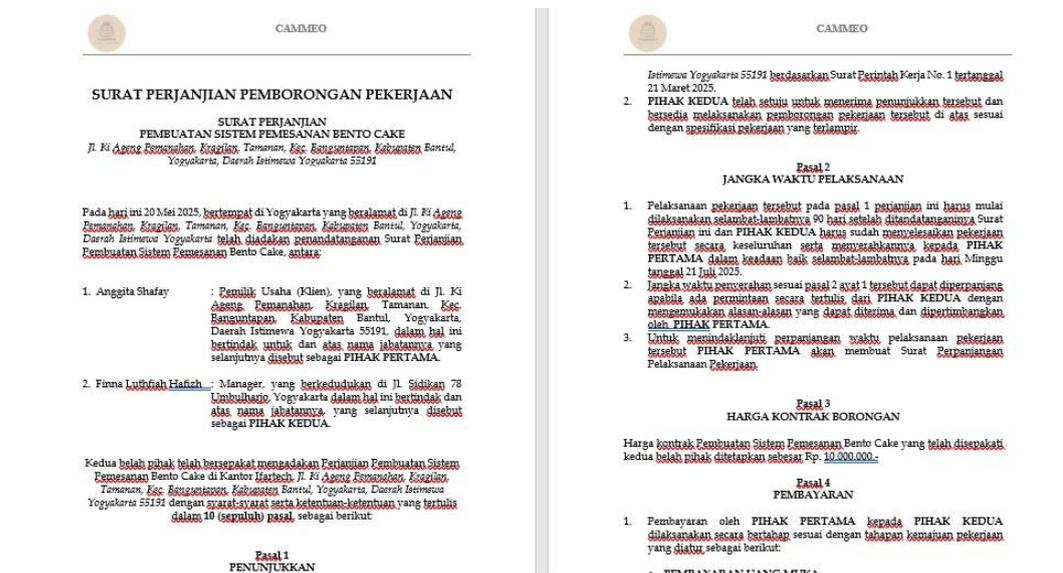
## LAMPIRAN

### 1. Proposal Proyek

<https://drive.google.com/file/d/14lsFCNHG7E0686QYsoVHPMVf-CXQpsAa/view?usp=sharing>

### 2. Surat Perintah Kerja / Kontrak Kerja dengan Mitra/Client

[https://drive.google.com/drive/folders/1\\_ihvfcZLqblBp2JQ\\_eh\\_ylgQx-dlllbr?usp=sharing](https://drive.google.com/drive/folders/1_ihvfcZLqblBp2JQ_eh_ylgQx-dlllbr?usp=sharing)



### 3. Log Book Kelompok sudah terisi minimal 7x

[https://drive.google.com/drive/folders/1Ecvijs9jxQSKqO\\_QHBnIWOPYBJuVDRsa?usp=sharing](https://drive.google.com/drive/folders/1Ecvijs9jxQSKqO_QHBnIWOPYBJuVDRsa?usp=sharing)

### 4. Logbook Individu sudah terisi minimal 7x

[https://drive.google.com/drive/folders/1TChcySPF36oG53bSTFazFDTw0DE\\_kPEG?usp=sharing](https://drive.google.com/drive/folders/1TChcySPF36oG53bSTFazFDTw0DE_kPEG?usp=sharing)

## 5. Foto dokumentasi kegiatan proyek



<https://drive.google.com/drive/folders/1y936U8iFlxfAav-KNTHuZhW-xMx7fmV2?usp=sharing>

## 6. Berita acara / bukti serah terima proyek



## 7. Tools : hosting, pemegang password

Tools:

- VSCode
- Composer
- Laravel Framework
- Figma
- XAMPP
- Draw IO
- Spreadsheet

Pemegang Password :

- Finna Luthfiah Hafizh

Akses web domain : <http://cammeo.my.id/>

## 8. Link video profil produk luaran proyek (durasi 4-7)

<https://drive.google.com/drive/folders/1nEKn8iV8MWcqYgVvL9EigMSrvJdmoSRp?usp=sharing>

## 9. Poster produk luaran projek

[https://drive.google.com/file/d/1j77ZkavC8wMUyQLijNVx5zBsx\\_kMo4bn/view?usp=sharing](https://drive.google.com/file/d/1j77ZkavC8wMUyQLijNVx5zBsx_kMo4bn/view?usp=sharing)

## 10. Slide presentasi projek

[https://www.canva.com/design/DAGuppf2BJc/9cspLNO8BOJfQ5YPXuN9Jg/edit?utm\\_content=DAGuppf2BJc&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAGuppf2BJc/9cspLNO8BOJfQ5YPXuN9Jg/edit?utm_content=DAGuppf2BJc&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

## 11. Bukti Pembayaran

