

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Pelaksanaan kegiatan magang pada *PT Catering Aja Indonesia* dimulai pada saat melalui interview oleh pemilik perusahaan secara langsung. Penempatan posisi magang sebagai backend developer yang berfokus pada pengembangan sistem operasional perusahaan secara keseluruhan berbasis teknologi website administrasi. Berhubungan *PT Catering Aja Indonesia* merupakan perusahaan *start up*, maka saya diharuskan untuk mempelajari dan menguasai ilmu perkodingan itu sendiri sesuai dengan project yang diberikan. Pendamping kerja magang adalah Bapak Steven Djingga selaku Chief Executive Officer (CEO) dalam membahas sistem kerja website perusahaan dan seorang developer pada project sebelumnya untuk mendampingi saya pada waktu tertentu.

Selama kegiatan magang berjalan terdapat beberapa tanggung jawab yang diberikan, yaitu develop back-end (admin portal berbasis PHP Laravel 11)[1] dari awal. Koordinasi dilakukan melalui pertemuan secara non-formal dengan CEO, Zoom Meeting dengan developer project sebelumnya, serta penggunaan Github dan Gitlab yang memudahkan dalam penyusunan hasil pekerjaan, serta memonitor setiap update yang telah dilakukan.

3.2 Tugas yang Dilakukan

Selama menjalani proses kegiatan magang di *PT Catering Aja Indonesia*, terdapat tugas-tugas yang telah ditetapkan saat proses meeting pertama, berupa pemberian workflow atau alur kerja suatu sistem administrasi catering atau fitur secara keseluruhan, serta tampilan desain website back-end. Inti dari tugas yang diberikan ialah perancangan dan pembangunan sistem portal admin & partner dashboard dari awal. Mencakup beberapa modul, seperti User Management (partner), Customer Management, End-point & API Control, dan Sistem Kalkulasi.

Tugas ini dikerjakan menggunakan *framework Laravel 11* [1] dengan metode MVC (Model-View-Controller)[2], yang menghasilkan aplikasi yang terstruktur dan efisien. Implementasi fitur yang mencakup autentikasi multirole untuk admin dan partner dalam melakukan manajemen data, hingga sistem pemesanan. Pengelolaan database, digunakan aplikasi PhpMyAdmin [3]

sebagai sarana untuk mengelola dan memanipulasi data yang diperlukan dalam pengembangan sistem.

Selama pelaksanaan kegiatan magang, tanggung jawab yang diberikan kepada saya adalah sebagai berikut:

1. Mempelajari framework Laravel 11 [1]& arsitektur MVC (Model-View-Controller)[4] yang digunakan oleh perusahaan untuk pembuatan sistem administrasi catering.
2. Menguasai fitur migration dalam laravel 11 yang memberi kemudahan dalam mengelola database *PhpMyAdmin* [3].
3. Melakukan pengembangan website admin stg-pos.cateringaja.com.
4. Pembuatan halaman-halaman pada website admin yang mencakup Kitchen Partner Management, Orders, Payment Gateway, Market Area.

Adapun tujuan akhir dari kegiatan magang selama 5 (lima) bulan ini adalah meningkatkan kualitas sistem adminstrasi pada website stg-pos.cateringaja.com, seperti sistem pembayaran, sistem kerja sama kitchen partner, sistem kalkulasi pembelian produk, sistem kontrol data menu, dan sebagainya.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan kerja magang diuraikan seperti pada Tabel III.1.



Tabel III.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	<ul style="list-style-type: none"> - Perkenalan sistem perusahaan <i>PT Catering Aja Indonesia</i> dan project yang akan menerapkan sistem catering di website. - Melakukan pembicaraan antarmuka dengan CEO <i>PT Catering Aja Indonesia</i> untuk membahas project yang akan dibangun.
2	<ul style="list-style-type: none"> - Mempelajari basis PHP dengan framework Laravel 11 dan arsitektur MVC[4] secara pribadi dan sharing dengan developer sebelumnya - Mempelajari struktur data pada PhpMyAdmin[3] dan apa saja relasi antar tabel.
3	<ul style="list-style-type: none"> - Membahas sistem kerja perusahaan <i>PT Catering Aja Indonesia</i> dengan detail. - Membahas secara detail sistem alur flow administrasi dan fungsionalitas fitur pada website.
4	<ul style="list-style-type: none"> - Berdiskusi dengan developer sebelumnya mengenai repository dan project yang akan dilakukan - Memasukkan segala struktur data menggunakan MySQL[5] di localhost data. - Mengeksplor aplikasi DBeaver[6] untuk mengelola dan mengakses berbagai jenis basis data dengan mudah.
5	<ul style="list-style-type: none"> - Mengubah Fitur Market Area dari sebelumnya menggunakan penentuan longitude dan latitude menjadi input manual dalam memudahkan admin. - Membuat halaman Market Area untuk membuat Area dan Sub Area baru.
6	<ul style="list-style-type: none"> - Pembuatan Halaman Kitchen Partner pada bagian Add Kitchen dan Halaman index Kitchen List yang menampilkan tabel Grid.JS [7]. Penggunaan Grid.JS[7] bertujuan untuk memiliki tampilan yang rapi dan nyaman digunakan pengguna.
7	<ul style="list-style-type: none"> - Pembuatan halaman Edit data partner serta fungsi mengaktifkan dan menonaktifkan Kitchen Partner.
8	<ul style="list-style-type: none"> - Menambah tabel data baru untuk bisa melakukan relasi many to many. Sehingga 1 kitchen partner bisa memiliki 1 market area dan banyak sub area.

Minggu Ke -	Pekerjaan yang dilakukan
9	<ul style="list-style-type: none"> - Pembuatan halaman untuk membuat produk Event dan Snackbox berupa validasi formulir. Produk akan masuk ke database tabel menu. - Mempelajari fitur SweetAlert [8] dan Select2 [5] untuk diterapkan pada website Admin.
10	<ul style="list-style-type: none"> - Membuat halaman Payment menggunakan sistem secara manual yaitu Accept dan Reject. - Mempelajari fitur-fitur penting yang mendukung jalannya flow website, seperti penggunaan Select2 untuk bisa memilih banyak opsi, melemparkan route menggunakan <i>Axios</i>[9], dll. - Mengubah fitur tombol delete, accept dan reject dengan menggunakan SweetAlert.
11	<ul style="list-style-type: none"> - Membuat validasi pada Add Kitchen Product. Admin dapat memasukkan data penting mengenai produk. Pada input timeslot, admin hanya dapat memilih waktu end_time saat start_time telah dipilih. Hal ini merepresentasikan jangka waktu ketersediaan produk.
12	<ul style="list-style-type: none"> - Mengintegrasikan fitur pencarian produk menggunakan AJAX untuk halaman admin. - Mengimplementasikan Select2 ke dalam form pencarian untuk mendukung multiple select yang lebih user-friendly. - Membahas validasi tambahan untuk sistem bersama CEO <i>PT Catering Aja Indonesia</i>.
13	<ul style="list-style-type: none"> - Membangun fitur pelaporan pada halaman admin, termasuk laporan penjualan, pesanan, dan market area..
14	<ul style="list-style-type: none"> - Mengembangkan modul pengelolaan pengguna, seperti halaman untuk membuat, mengedit, dan menghapus kitchen partner, menu, payment. - Melakukan uji coba sistem dengan developer sebelumnya untuk memastikan fitur berjalan sesuai harapan.

Minggu Ke -	Pekerjaan yang dilakukan
15	<ul style="list-style-type: none"> - Melanjutkan pengembangan halaman untuk pengelolaan pembayaran pelanggan, termasuk filter pembayaran berdasarkan status (Pending, Accepted, Rejected). - Mengoptimalkan desain halaman admin dengan memperbarui UI menggunakan library CSS seperti Bootstrap atau TailwindCSS.
16	<ul style="list-style-type: none"> - Melakukan perbaikan pada website CPanel Manager untuk bisa menjalankan hasil pemograman website pada url https://stg-pos.cateringaja.com - Melakukan pengujian sistem secara keseluruhan untuk memastikan integrasi antar fitur berjalan lancar. - Menyusun panduan penggunaan untuk admin, termasuk tutorial cara mengelola data, fitur, dan alur sistem baru.

3.3.1 Uraian Proses Pelaksanaan Magang

A. Perancangan dan pembangunan portal admin website back-end stg-pos.cateringaja.com

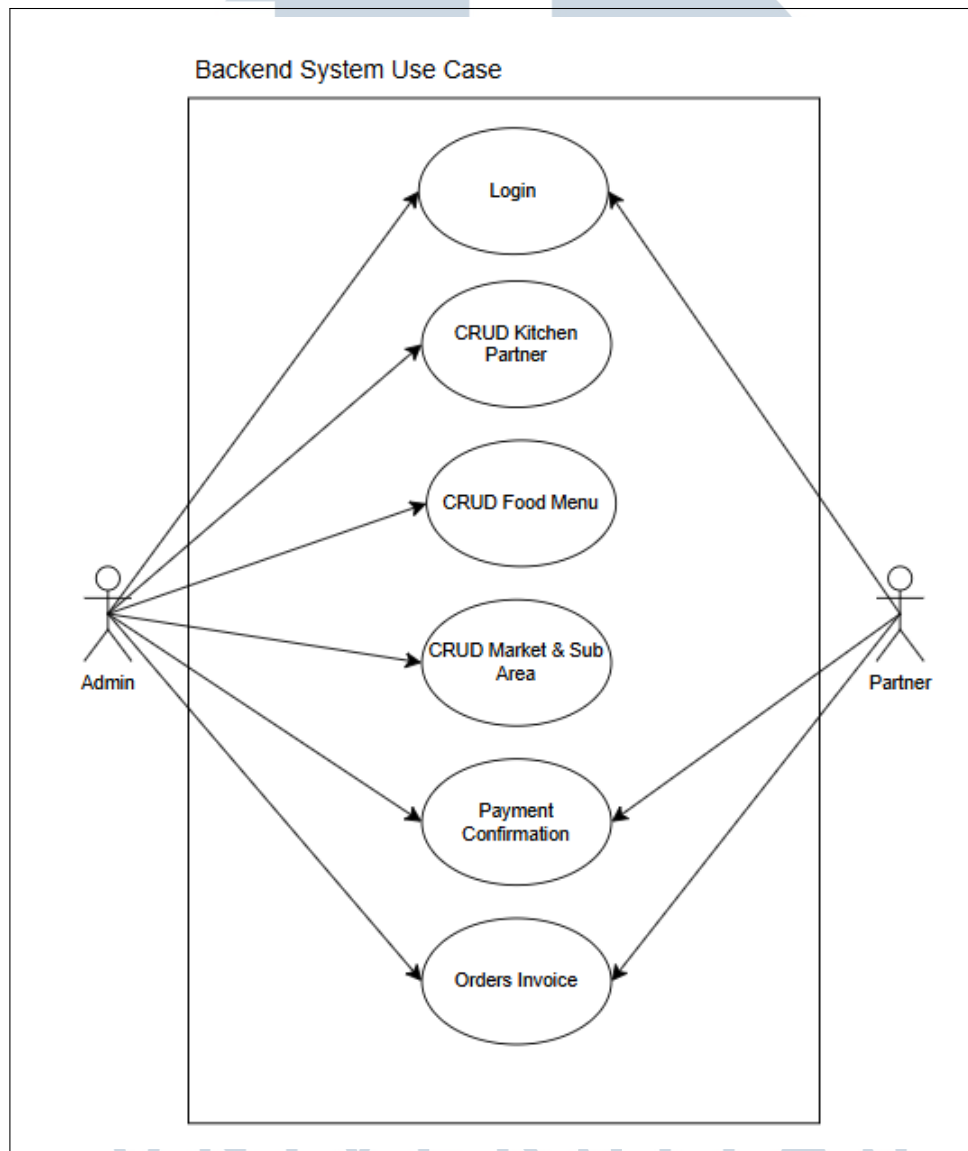
Metode MVC (Model-View-Controller) adalah sebuah arsitektur perangkat lunak yang digunakan untuk memisahkan logika aplikasi menjadi tiga bagian utama, yaitu Model, View, dan Controller. Pendekatan ini membantu pengembang memisahkan data, logika bisnis, dan antarmuka pengguna untuk menciptakan aplikasi yang terstruktur, mudah dikembangkan, dan lebih terjaga pemeliharaannya, serta dapat mempermudah pengembangan oleh developer selanjutnya.

Tahapan awal dalam perancangan website backend stg-pos.cateringaja.com ialah menganalisis kebutuhan pengguna dan alur bisnis dari jasa Catering Aja. Dalam sistem ini, terdapat dua aktor utama, yaitu:

- Admin: Mengelola data, termasuk menu makanan, pesanan pelanggan, dan manajemen kitchen partner.
- Kitchen Partner (Mitra Catering): Bertanggung jawab untuk menyediakan layanan catering seperti persediaan, masak, persiapan, dan pengiriman.

Analisis ini dilakukan bersama CEO *PT Catering Aja Indonesia* untuk memastikan bahwa sistem yang dikembangkan mampu memenuhi kebutuhan setiap aktor dengan tepat dan efisien. Proses ini bertujuan untuk meningkatkan kemudahan

pengguna dalam mengoperasikan sistem melalui platform berbasis web. Berikut adalah ringkasan beberapa bagian utama yang saya kerjakan selama kegiatan magang di *PT Catering Aja Indonesia*, beserta penjelasan mengenai halaman pada website backend.



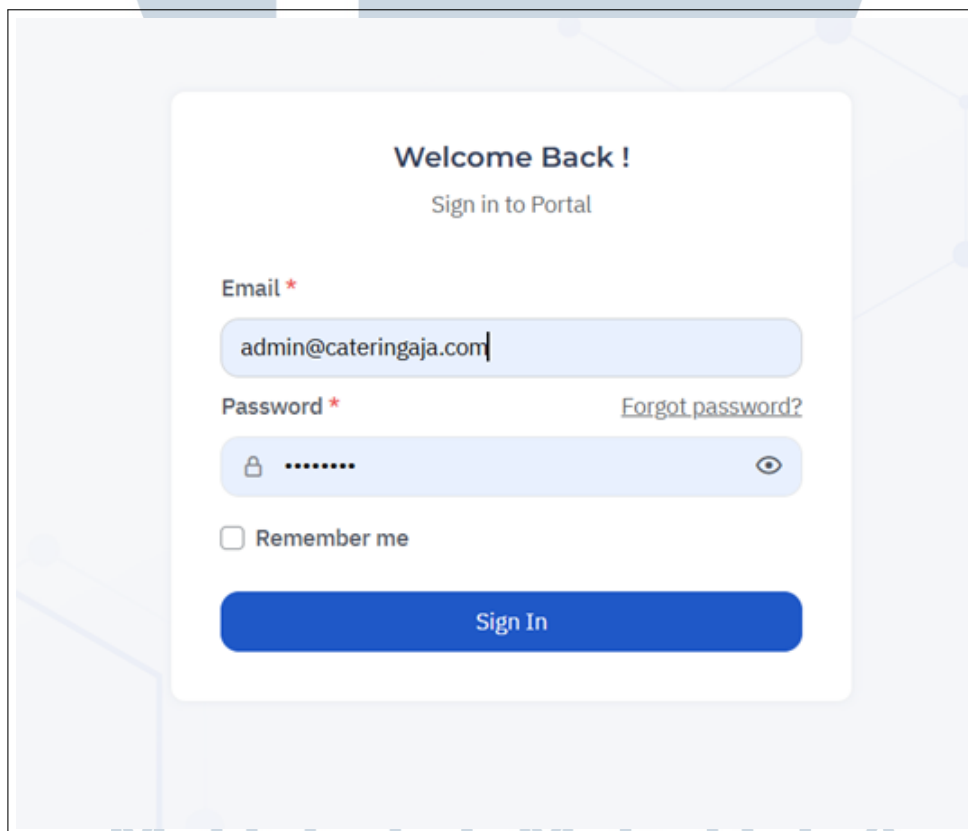
Gambar III.1. Use-case Diagram Sistem Backend

A.1 Halaman Login

Halaman Login adalah halaman antarmuka awal yang digunakan untuk mengautentikasi pengguna berdasarkan kredensial email dan kata sandi sebelum diberikan akses ke sistem. Hal ini bertujuan untuk membatasi akses pengguna hanya pada fitur yang sesuai dengan hak aksesnya. Login dilakukan pada saat pengguna

pertama kali masuk ke dalam website atau saat session pengguna telah kadaluarsa atau hilang karena logout.

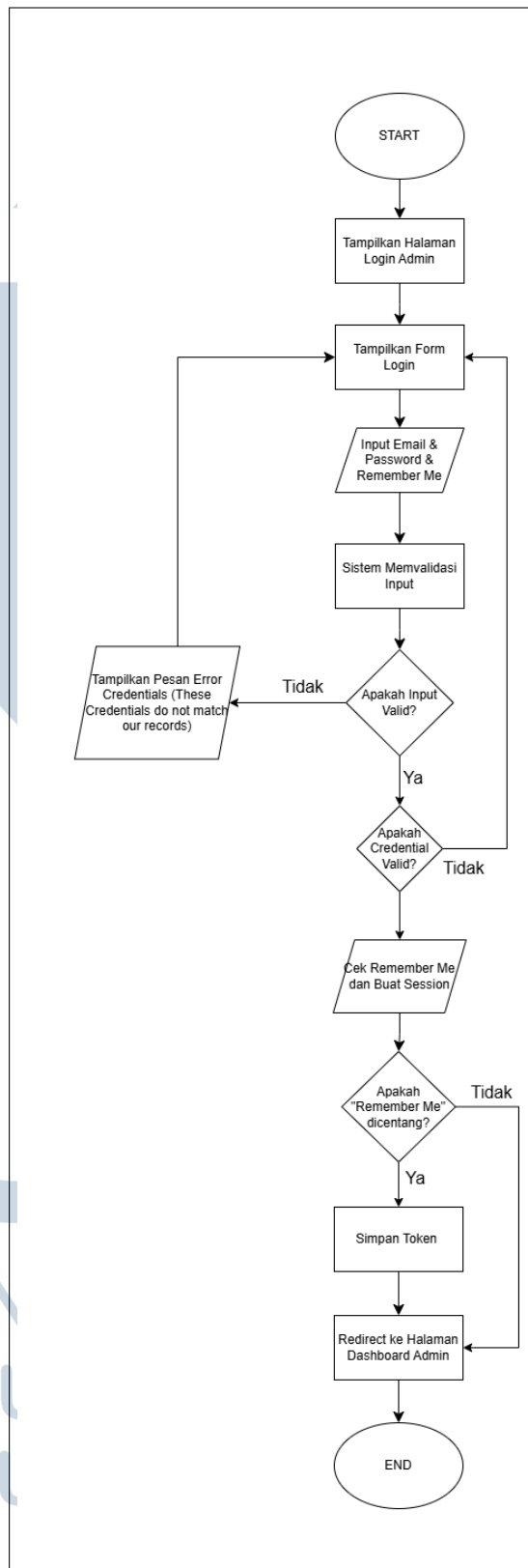
Dalam pengembangan aplikasi website berbasis framework laravel [1], mekanisme login menggunakan autentikasi bawaan laravel yang memanfaatkan middleware, guard, dan validasi di controller. Mekanisme ini memungkinkan sistem untuk memisahkan peran aktor atau pengguna seperti Admin dan Kitchen Partner, agar memiliki hak akses yang aman dan terstruktur dan menjaga keamanan data yang penting. Halaman Login antara Admin dan Kitchen Partner dibuat Route secara terpisah untuk dengan mempertimbangkan keamanan akses. Halaman Login Kitchen Partner memiliki route tambahan berupa “/user/login”. Mekanisme Login Admin dan Kitchen Partner pada website stg-pos.cateringaja.com adalah seperti diagram dibawah ini.



The image shows a login form with the following elements:

- Title:** Welcome Back !
- Subtitle:** Sign in to Portal
- Email Field:** Labeled "Email *", containing the text "admin@cateringaja.com".
- Password Field:** Labeled "Password *", containing masked characters ".....". It includes a "Forgot password?" link and a toggle icon for visibility.
- Remember me:** A checkbox labeled "Remember me" which is currently unchecked.
- Sign In Button:** A prominent blue button labeled "Sign In".

Gambar III.2. Use-case Diagram Sistem Backend



Gambar III.3. Diagram Flowchart proses Login Admin dan Kitchen Partner.

Gambar diatas menunjukkan flowchart untuk proses Login Admin dan Kitchen Partner. Proses dimulai saat link website di klik dan akan muncul halaman login. Pada halaman ini pengguna akan ditampilkan formulir login berupa email dan kata sandi. Pengisian formulir login harus berdasarkan data yang telah dibuat oleh admin. Apabila email dan kata sandi tidak terdaftar dalam database, maka laravel akan mengecek kredensial login. Jika email dan kata sandi tidak terdaftar dalam database, maka sistem akan mengembalikan pesan error dan halaman akan reload ke pengisian formulir login. Admin dan Kitchen Partner juga dapat menyimpan session email dan kata sandi mereka untuk automatisasi login. Apabila kata sandi salah maka hanya error credentials dibawah form email. Email dan kata sandi terdaftar akan divalidasi dan diarah ke halaman Dashboard.

Terdapat dua link url portal website yang berbeda antara Admin dengan Kitchen Partner. Akan tetapi kedua halaman memiliki flow yang sama, yang membedakan hanyalah pengambilan data dari tabel yang berbeda. Sehingga terdapat tabel yang berisi data Admin pada tabel users dan Kitchen Partner pada tabel user.

A.2 Halaman Kitchen Partner

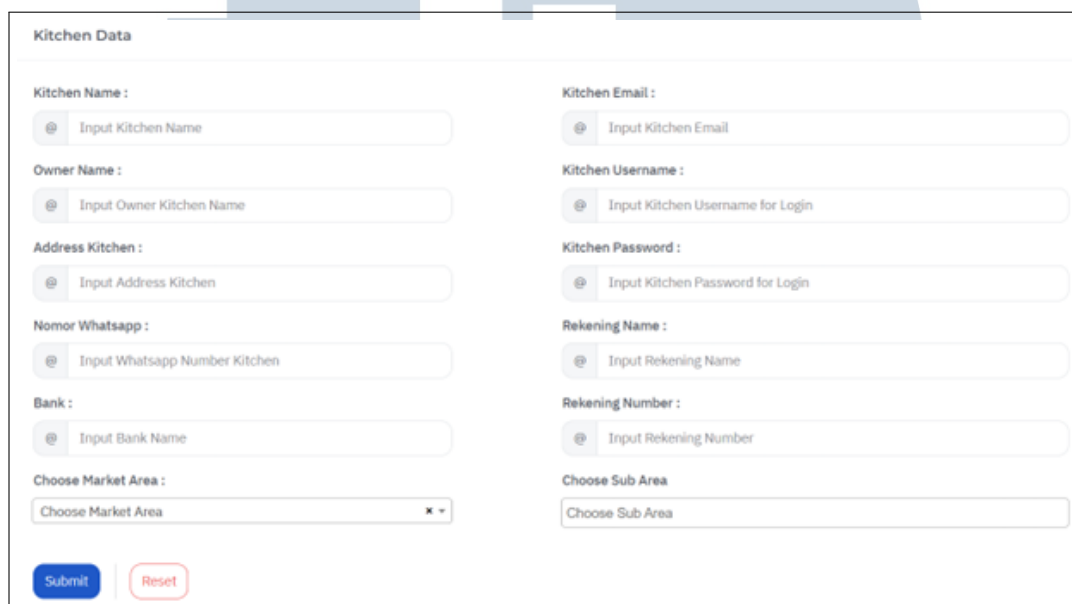
Halaman Kitchen Partner merupakan halaman antarmuka yang berperan penting sebagai pusat pengelolaan dan pengawasan terhadap *mitra* penyedia jasa catering (kitchen partner). Halaman kitchen partner terdapat dua halaman penting, halaman pendaftaran/pengisian data *partner* (Add Kitchen), dan halaman untuk melihat data partner (Kitchen List).

- Halaman Add Kitchen

Halaman Add Kitchen ini dapat melakukan pengisian data partner yang bersedia bekerja sama. Admin akan membuat data kepada partner berupa email dan kata sandi kepada partner untuk mengakses website khusus partner. Mengelola data milik setiap partner, seperti mengaktif dan menonaktifkan restoran, melakukan sembunyi atau hide dan menghapus produk. Halaman Kitchen List memiliki tampilan tabel yang berisi data setiap partner/mitra mulai dari Kitchen Name, Owner Name, Phone Number, Operational, Status, Email, Address, dan Action. Tampilan tabel menggunakan *Grid.JS*[7] agar memiliki tampilan yang menarik dan rapi. Segala sistem untuk mengatur database terdapat pada halaman Kitchen List ini. Berikut ini adalah penjelasan mengenai halaman Kitchen List.

Penjelasan pengisian Formulir Add Kitchen:

1. Pada halaman ini admin dapat melakukan pengisian data penting untuk kitchen partner.
2. Admin mengisi *Kitchen Name*, *Owner Name*, *Nomor WhatsApp*, *Address Kitchen*, *Bank*, *Choose Market Area*, *Kitchen Email*, *Kitchen Username*, *Kitchen Password*, *Rekening Name*, *Rekening Number*, *Choose Sub Area*.



The image shows a web form titled "Kitchen Data" with two columns of input fields. The left column includes: "Kitchen Name" (text input), "Owner Name" (text input), "Address Kitchen" (text input), "Nomor Whatsapp" (text input), "Bank" (text input), and "Choose Market Area" (dropdown menu). The right column includes: "Kitchen Email" (text input), "Kitchen Username" (text input), "Kitchen Password" (text input), "Rekening Name" (text input), "Rekening Number" (text input), and "Choose Sub Area" (text input). At the bottom left, there are two buttons: "Submit" (blue) and "Reset" (red).

Gambar III.4. Halaman Pengisian Data Kitchen Partner.

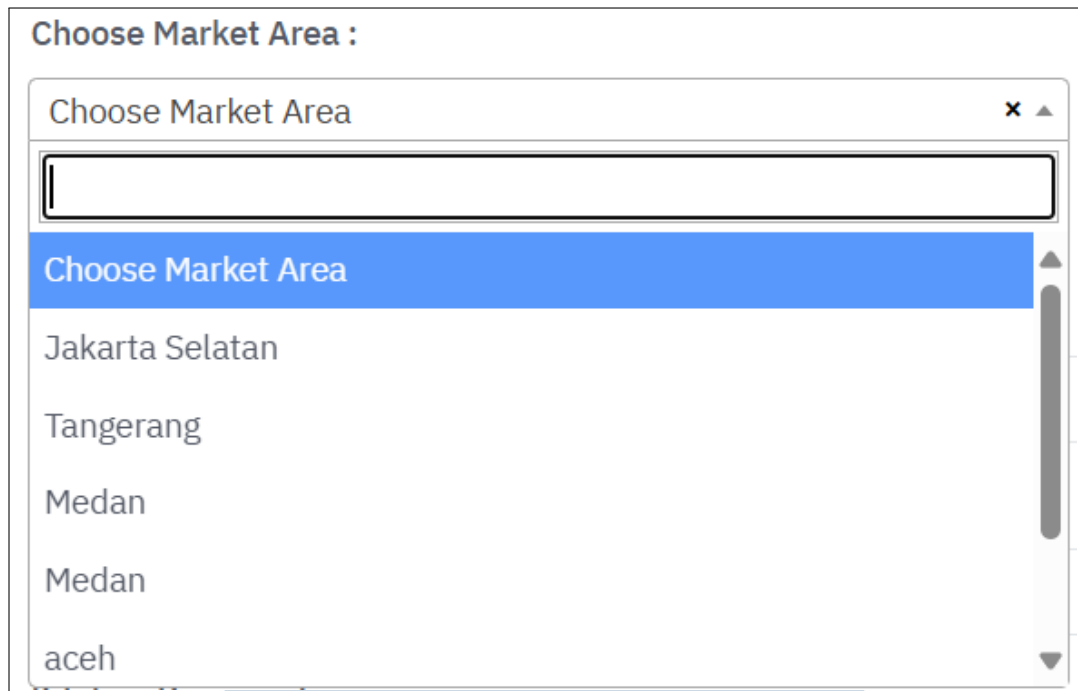
3. Pengisian data perbankan, seperti Nomor Rekening, Nama Rekening, dan Jenis Bank, memiliki peran penting dalam mendukung efisiensi administrasi transaksi antara *PT Catering Aja Indonesia* dan Kitchen Partner. Data ini memudahkan admin atau pihak pemasaran dalam memproses pembayaran secara akurat dan tepat waktu. Proses pembayaran dilakukan setelah data pembelian pelanggan diverifikasi untuk memastikan keabsahan transaksi. Sistem ini dirancang sesuai dengan mekanisme pemesanan yang berlaku, di mana pelanggan hanya dapat memesan makanan untuk hari berikutnya (H-1) atau untuk tanggal lain di masa mendatang. Pembatasan ini memberikan waktu yang cukup bagi tim operasional untuk memvalidasi data, mengelola pembayaran, dan memastikan layanan berjalan lancar sesuai standar perusahaan.
4. Pengisian Username, Kitchen Email, dan Kitchen Password dirancang

untuk membuat akun bagi Kitchen Partner yang telah menjalin kontrak kerja sama dengan perusahaan. Akun ini memberikan akses ke portal khusus yang memungkinkan Kitchen Partner mengelola berbagai aspek operasional mereka. Kata sandi yang dibuat menggunakan metode bcrypt untuk menghasilkan hashing, sehingga kata sandi tidak disimpan dalam bentuk teks asli di database, menjaga keamanan data dari potensi pelanggaran.

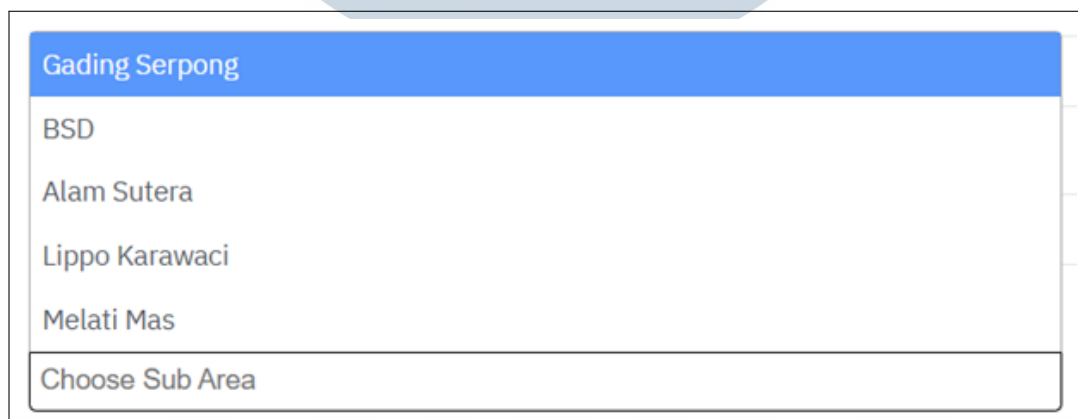
Meskipun admin memiliki kemampuan untuk membuat atau mengatur ulang kata sandi bagi Kitchen Partner, kata sandi yang sudah dibuat tidak dapat dilihat kembali karena sistem menyimpannya dalam bentuk hash. Jika Kitchen Partner lupa kata sandinya, admin dapat mereset kata sandi dengan mengatur ulang data di sistem, atau Kitchen Partner dapat menggunakan fitur forgot password yang memungkinkan mereka mengganti kata sandi lama dengan yang baru. Sistem ini dirancang untuk memberikan tingkat keamanan tinggi sekaligus fleksibilitas dalam pengelolaan akun, baik oleh admin maupun Kitchen Partner.

5. Choose *Market Area* dan *Sub Area* ini dapat dilakukan dengan format dropdown select. Admin dapat memilih Market Area terlebih dahulu untuk bisa memunculkan data sesuai id_sub pada Choose Sub Area. Sehingga terdapat relasi antara Market Area dengan Sub Area. Contohnya adalah apabila admin memilih Market Area Tangerang, maka pada Sub Area akan menampilkan opsi Gading Serpong, Alam Sutera, Karawaci, BSD, dsb. Fitur ini memanfaatkan fitur Select2, Hal ini dapat memudahkan admin untuk mencari Market Area yang diinginkan dan dapat memilih lebih dari satu opsi dengan *tagging*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar III.5. Fitur Select2 Choose Market Area.



Gambar III.6. Fitur Select2 Choose Sub Area.

- Halaman Kitchen List

Pada halaman ini admin dapat mengubah dan memanipulasi data yang diinginkan. Penjelasan segala fitur dan alur yang dapat dilakukan akan ditampilkan dalam bentuk Use Case Diagram sebagai berikut.

Penjelasan Tahapan Alur dan Fungsionalitas Fitur pada Halaman Kitchen List.

1. Pada awal halaman Kitchen List, Admin akan ditampilkan *Grid.js*[7] yang merupakan fitur untuk styling tabel agar memiliki tampilan yang

rapi. Pada kolom bagian kanan tabel, terdapat tombol button action berupa "View Details". Apabila admin menekan tombol, maka akan redirect ke halaman lain yaitu Informations yang berisi kumpulan data milik Kitchen Partner tersebut. Halaman ini terdapat sub halaman yaitu *Informations, Products, Setting* dalam bentuk button yang dapat di klik.

Kitchen Name	Owner	Phone Number	Operatio...	Status	email	Address	Action
Goyang Lidah	Veronicaa	081717117444	Open	Active	stevenpbb@gmail.com	Ruko Sentra Gading Blok SG 3 No. 5	View Details
Warung Bu Anik	Bu Anik	082432525	Close	Inactive	anik@gmail.com	Jl. BSD No. 5	View Details
Alan Kitchen	Alan	+62 822-8917-2506	Open	Active	alankitchen@gmail.com	Jalan wahidin, medan	View Details

Gambar III.7. Fitur Grid.js untuk menampilkan data Kitchen Partner.

- Halaman *Informations* berisi data yang telah dibuat di Add Kitchen dalam bentuk *Disabled Form*. Tujuan utama dalam bentuk form ialah agar admin dapat mengubah data milik kitchen partner. Terdapat tombol "Edit Kitchen Data" pada bagian bawah form. Saat tombol ditekan, form *disabled* akan berubah menjadi *enabled* untuk dilakukan perubahan data Kitchen Partner.

Kitchen Partner : Goyang Lidah

[Informations](#)
[Products](#)
[Setting](#)

Kitchen Name : Goyang Lidah
 Owner Name : Veronicaa
 Address Kitchen : Ruko Sentra Gading Blok SG 3 No. 5
 Nomor Whatsapp : 081717117444
 Bank : BCA
 Choose Market Area : Tangerang

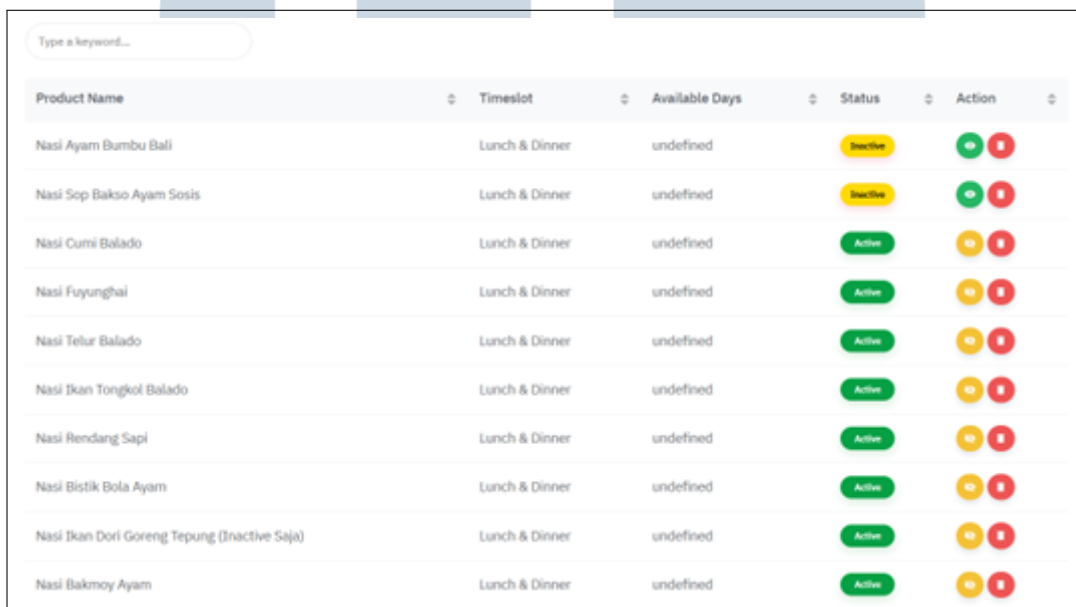
Kitchen Email : stevenpbb@gmail.com
 Kitchen Username : goyanglidah01
 Kitchen Password : \$2y\$10\$qPFxc2r4H0r79FjNadxe.Or.qCuICFNvmaqLoYob0eucX4kLydYq
 Rekening Name : Goyang Lidah
 Rekening Number : 123456
 Choose Sub Area : Gading Serpong

Status Kitchen : Open

[Edit Kitchen Data](#)
[Back To List](#)

Gambar III.8. Halaman Informations yang berisi Disabled form.

3. Pada halaman ini Admin juga dapat mengubah status kitchen dengan tombol *checkbox*, apabila *checkbox* dimatikan maka status kitchen akan berubah menjadi “*Inactive*” untuk menyatakan bahwa kitchen tidak aktif atau sudah tidak ada kontrak dengan catering aja. Hal ini bertujuan untuk mempermudah admin agar tidak kehilangan data partner. Apabila suatu saat melakukan partner kontrak lagi, maka admin hanya perlu mengaktifkan kembali status Kitchen Partner. Setelah selesai mengubah data maka admin dapat menyimpan data dengan klik tombol save. Data akan dilakukan validasi lagi dan mengubah data yang diubah.



Product Name	Timeslot	Available Days	Status	Action
Nasi Ayam Bumbu Bali	Lunch & Dinner	undefined	Inactive	🟢 🔴
Nasi Sop Bakso Ayam Sosis	Lunch & Dinner	undefined	Inactive	🟢 🔴
Nasi Cumi Balado	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Fuyunghai	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Telur Balado	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Ikan Tongkol Balado	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Rendang Sapi	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Bistik Bola Ayam	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Ikan Dori Goreng Tepung (Inactive Saja)	Lunch & Dinner	undefined	Active	🟡 🔴
Nasi Bakmoy Ayam	Lunch & Dinner	undefined	Active	🟡 🔴

Gambar III.9. Halaman Products yang berisi tabel data menu.

4. Halaman products adalah halaman yang berisi kumpulan data menu yang telah dibuat oleh admin. Catering Aja memiliki 3 jenis tipe produk, yaitu Langganan, Event, dan Snackbox.
- Langganan adalah jenis pesanan yang dapat dibeli pelanggan secara harian, mingguan, dan bulanan.
 - Harian adalah jenis pesanan yang dibeli pelanggan dengan satu hari yang ditentukan.
 - Mingguan adalah jenis pesanan yang dapat dibeli pelanggan selama 5 hari dengan jumlah produk bebas.
 - Bulanan adalah jenis pesanan yang dapat dibeli pelanggan dalam satu bulan kedepan.

	Harian	Mingguan	Bulanan
Harga	Rp33.000,-	Rp31.000,-	Rp29.000,-

Tabel III.2. Tabel Harga Harian, Mingguan, dan Bulanan

- Masing-masing jenis langganan memiliki harga yang berbeda.

Product Name	Timeslot	Available Days	Status	Action
Nasi Ayam Bumbu Bali	Lunch & Dinner	undefined	Inactive	🔄 🗑️
Nasi Sop Bakso Ayam Sosis	Lunch & Dinner	undefined	Inactive	🔄 🗑️
Nasi Cumi Bolado	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Fuyunghai	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Telur Balado	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Ikan Tongkol Balado	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Rendang Sapi	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Bistik Bola Ayam	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Ikan Dori Goreng Tepung (Inactive Saja)	Lunch & Dinner	undefined	Active	🔄 🗑️
Nasi Bakmoy Ayam	Lunch & Dinner	undefined	Active	🔄 🗑️

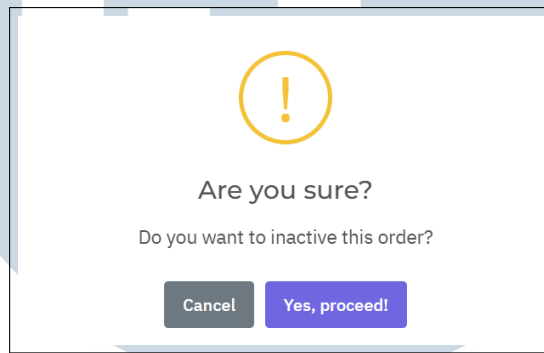
Gambar III.10. Halaman Products yang berisi tabel data menu.

Setiap tombol jenis pesanan akan menampilkan tabel Grid.js untuk menampilkan data menu masing-masing. Sehingga memiliki perbedaan produk antara Langganan, Event, dan Snackbox.

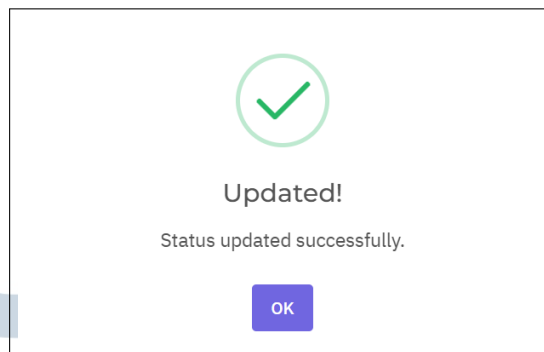
5. Terdapat tombol pada action yang dapat melakukan sembunyi/hide untuk menyembunyikan produk di front-end dan delete untuk menghapus data produk. Apabila tombol hide di klik maka akan mengubah status menjadi inactive, dan dapat dilakukan active kembali. Data akan dilemparkan ke website front-end untuk memunculkan dan menyembunyikan data. Fungsi delete itu telah menggunakan Sweet Alert agar memiliki tampilan yang rapi serta meyakinkan admin untuk menghapus data yang tidak dapat di restore. Terdapat tahapan delete berupa:
 - Meyakinkan admin dengan Are you sure? Dengan tombol cancel dan yes, delete it.
 - Apabila admin menekan yes maka akan menuju ke proses selanjutnya pada SweetAlert yaitu fungsi (**.then**) dengan

memunculkan form untuk mengisi alasan penghapusan menu. Pengisian ini bersifat tidak wajib dan dapat di skip ke langkah selanjutnya.

- Saat menekan tombol “*Submit Reason and Proceed*” maka akan menuju ke proses Alert selanjutnya yaitu *Final Confirmation*.
- Pada final confirmation terdapat deskripsi yang berisi reason yang telah dimasukkan pada form Submit Reason. Admin dapat klik “yes” jika setuju untuk menghapus data dan “cancel” untuk membatalkan proses.

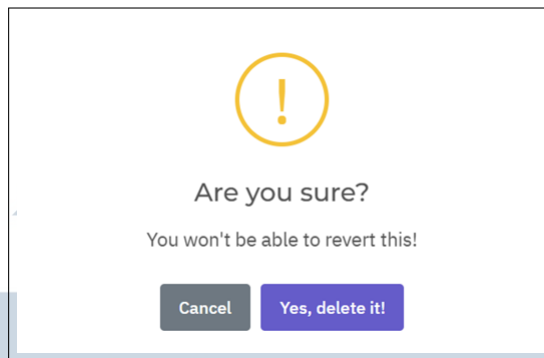


Gambar III.11. Pop Up Sweet Alert Hide Menu.

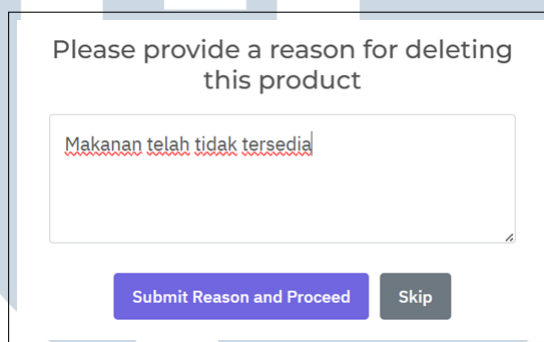


Gambar III.12. Pop Up Sweet Alert Hide Menu.

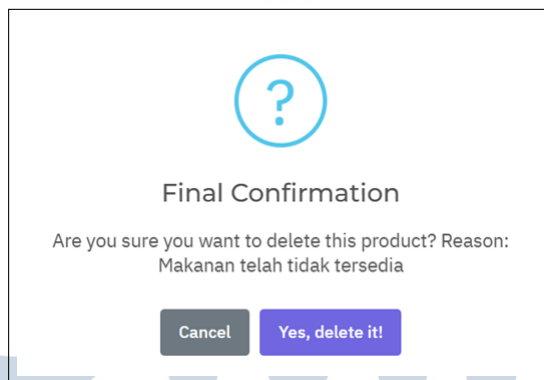
Berikut adalah tampilan dari SweetAlert dalam proses menghapus data menu.



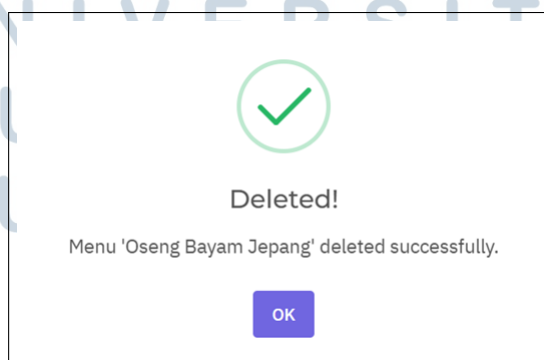
Gambar III.13. Pop Up Sweet Alert Delete Menu.



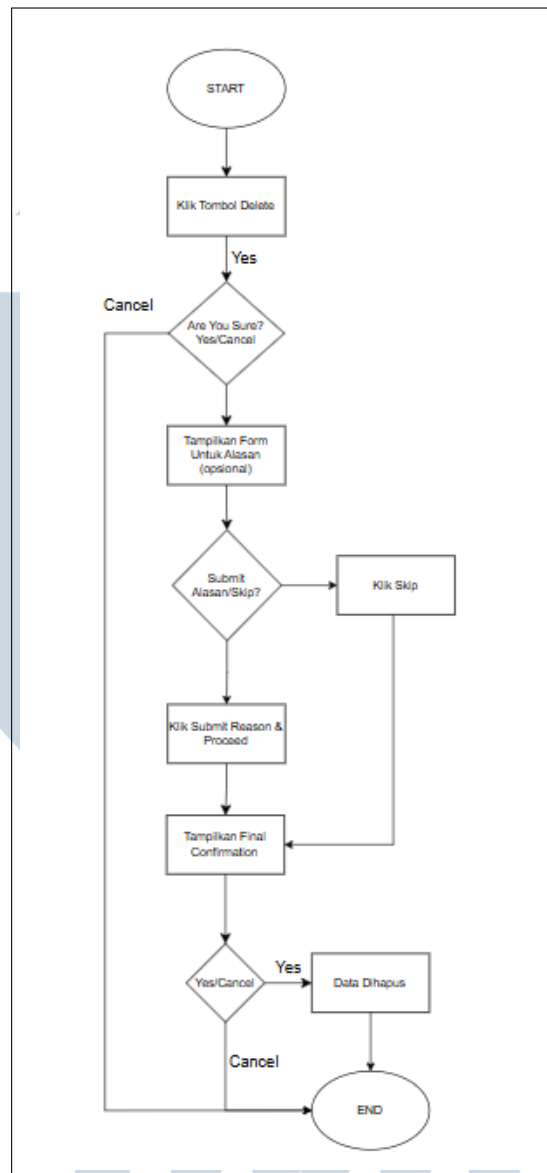
Gambar III.14. Pop Up Sweet Alert Delete Menu.



Gambar III.15. Pop Up Sweet Alert Delete Menu.



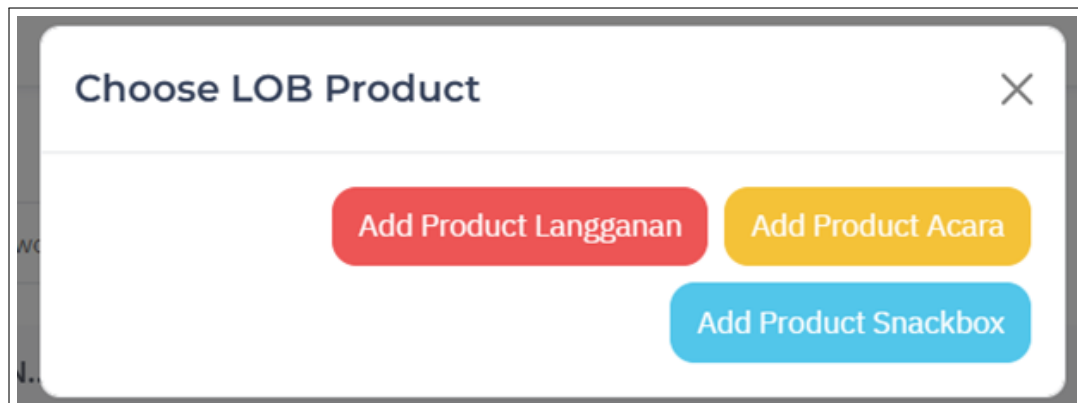
Gambar III.16. Pop Up Sweet Alert Delete Menu.



Gambar III.17. Diagram Flowchart proses *Delete*.

6. Halaman Add Product

Setiap jenis pesanan terdapat halaman Add Product masing-masing. Pada halaman Add Product, admin dapat membuat produk baru masing-masing Kitchen Partner.



Gambar III.18. Tampilan Modal pada halaman *Choose LOB Product*.

Halaman ini berisi formulir untuk mengisi data penting produk. Terdapat perbedaan antara jenis pesanan Langganan dengan Event dan Snackbox. Hal ini dikarenakan Langganan adalah jenis pesanan yang dipesan berdasarkan waktu yang diinginkan pelanggan dalam membeli menu, sehingga terdapat tiga jenis waktu (Lunch, Dinner, dan Lunch & Dinner). Sedangkan jenis pesanan Event dan Snackbox ditunjukkan untuk satu waktu atau acara dan dalam jumlah kuantiti yang besar. Pada jenis Event dan Snackbox ini terdapat start time dan end time untuk menentukan tersedianya produk dan tidak tersedianya produk. Pada jenis Event dan Snack Box juga terdapat form tambahan yang dapat muncul saat menekan tombol plus “+” di bagian bawah form utama. Sehingga dapat memantau menu apa saja yang telah ditambahkan sebelumnya. Setiap ingin menambah produk, admin hanya perlu menekan tombol plus dibawah dan form baru akan muncul sesuai dengan menu milik kitchennya. Sehingga setiap kitchen memiliki masing-masing form pengisian. End Time hanya dapat memilih jam atau waktu diatas dari waktu start time yang telah dipilih.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Add Product Kitchen Data

Package Name:

Package Description:

Price (Kitchen) / Harga Kitchen:

Price (Customer) / Harga Pelanggan:

Pilih Hari (untuk menu tersedia):
 Hari Senin
 Hari Selasa
 Hari Rabu
 Hari Kamis
 Hari Jumat
 Hari Sabtu

Upload Gambar Menu (max 1600px x 1600px & max 2 mb):
 No file chosen

Minimum Quantity

TimeSlot
Start Time:
End Time:

Add Menu:

Menu Title:

Description:

Gambar III.19. Tampilan Halaman Form Add Product Event dan Snackbar.

Add Product Kitchen Data

Menu Name:

Menu Description:

Price (Kitchen) / Harga Kitchen:

Price (Customer) / Harga Pelanggan:

Pilih Hari (untuk menu tersedia):
 Hari Senin
 Hari Selasa
 Hari Rabu
 Hari Kamis
 Hari Jumat
 Hari Sabtu

Upload Gambar Menu (max 1600px x 1600px & max 2 mb):
 No file chosen

Product Cycle (per days):

Choose Time Available:

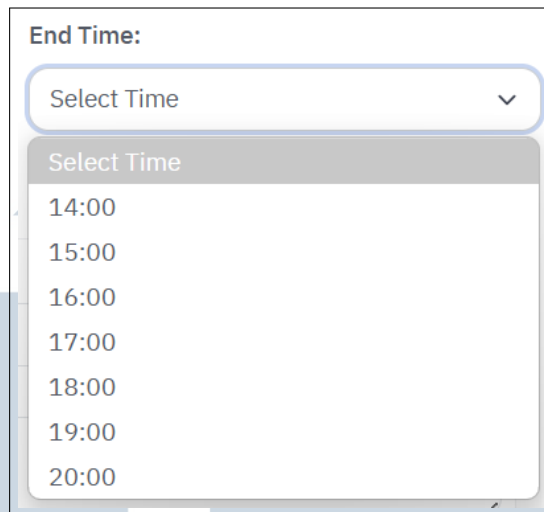
Choose Category Menu:

Pilih Jenis Masakan:

Gambar III.20. Tampilan Halaman Form Add Product Langganan.

Start Time:

Gambar III.21. Tampilan Dropdown *Start Time*.



Gambar III.22. Tampilan Dropdown *End Time*.

A.3 Halaman Payment

Halaman payment merupakan halaman antarmuka yang dirancang untuk menampilkan atau merekap data pemesanan dan pembayaran yang dilakukan oleh pelanggan. Data pembayaran akan secara otomatis tercatat di halaman admin setiap kali transaksi diterima. Untuk memberikan tampilan yang rapi dan terstruktur, halaman ini memanfaatkan Grid.js, sebuah alat yang memungkinkan penyajian data dalam format tabel yang dinamis dan responsif.

Halaman ini menggabungkan berbagai jenis data tabel untuk menampilkan informasi penting terkait pelanggan. Admin menggunakan halaman Payment untuk memverifikasi data pembelian secara manual. Setiap data pembelian yang baru masuk akan memiliki status awal **“Pending”**. Jika pesanan diterima, status akan berubah menjadi **“Accepted”** (ditandai dengan warna hijau), sedangkan jika ditolak, status akan berubah menjadi **“Rejected”** (ditandai dengan warna merah). Pada tabel terdapat *Order ID*, *Customer Name*, *Area*, *Sub Area*, *Phone Number*, *Quantity*, *Total Amount*, *Transfer*, *Status*, dan *Action*. Berikut adalah penjelasan beberapa kolom penting pada halaman payment.

- Order ID merupakan hasil dari Random Number Generate agar bisa menghasilkan id yang berbeda. Penggunaan `Carbon::now()->('mdY')` akan menghasilkan format tanggal *mdY* seperti 02252024 untuk 25 Februari 2024. Kemudian disusul dengan Random Number yang menghasilkan 4 digit angka acak untuk memastikan kode unik. Dikombinasikan prefiks **“CA”** sebagai singkatan dari Catering Aja, sehingga menghasilkan kode seperti

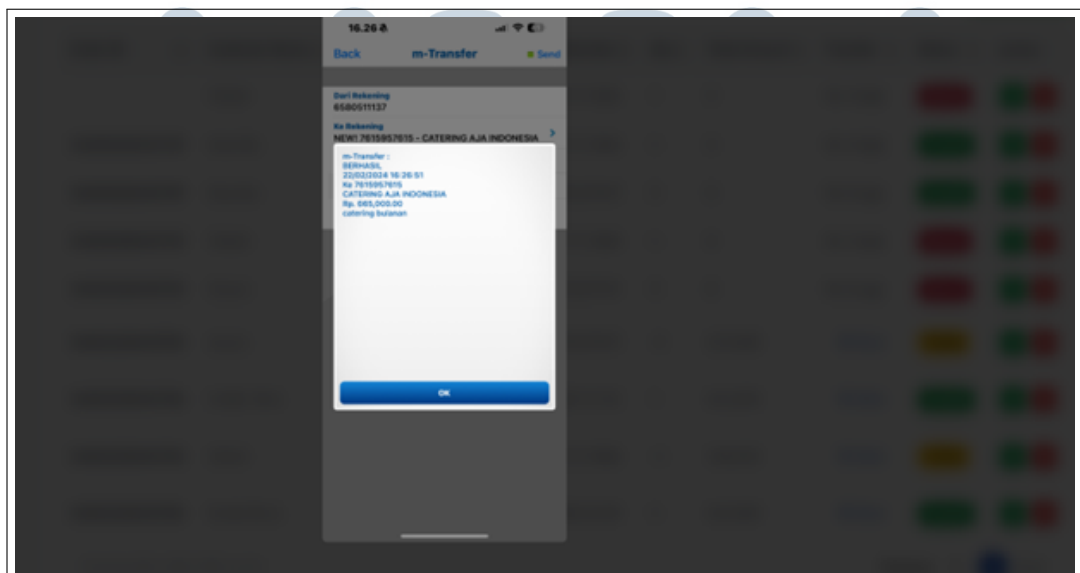
CA022520241797. Order ID akan menjadi kode order yang akan didapatkan pelanggan untuk mengecek status pemesanan mereka pada website.

- Admin dapat melakukan aksi pada bagian action. Apabila data sudah benar sesuai, maka data akan diterima dengan klik tombol centang. Namun jika tidak sesuai, maka data akan di reject dengan menekan tombol silang. Setiap validasi action terdapat alert menggunakan *SweetAlert* sebagai fitur untuk meyakinkan admin dalam melakukan aksi.

Order ID	Customer Name	Area	Sub Area	Phone Number	Qty	Total Amount	Transfer	Status	Action
	Steven	Tangerang	Wahidin	081717117444	1	0	No Image	Rejected	✓ ✕
CA030320241749	Kayudra	Tangerang	Johor	081717117444	5	0	No Image	Accepted	✓ ✕
CA030320241749	Kayudra	Tangerang	Wahidin	081717117444	5	0	No Image	Accepted	✓ ✕
CA022520241719	Steven	Tangerang	Johor	081717117444	5	0	No Image	Rejected	✓ ✕
CA022520241719	Steven	Tangerang	Wahidin	081717117444	5	0	No Image	Rejected	✓ ✕
CA022220241703	steven	Tangerang	Wahidin	081717117444	-3	144,000	View	Pending	✓ ✕

Gambar III.23. Tampilan halaman *Payment Confirmation*.

- Terdapat fitur view pada transfer yang dapat di klik untuk melihat foto transfer dengan jelas.



Gambar III.24. Tampilan modal bukti pembayaran pada *Payment Confirmation*.

A.4 Halaman Orders

Halaman Orders adalah antarmuka yang dirancang untuk menampilkan kumpulan data pesanan yang telah disetujui pada halaman Payment. Data pesanan akan secara otomatis masuk ke halaman Orders setelah status pembayaran berubah menjadi **Accepted** dan memiliki *id_status* 10.

Halaman ini memberikan kemudahan bagi admin untuk memantau detail pesanan yang telah diterima. Salah satu fitur penting pada halaman ini adalah kolom “**View Invoice**”, yang memungkinkan admin untuk membuka tampilan modal berisi detail pesanan pelanggan secara lengkap.

Tabel pada halaman Orders mencakup informasi berikut:

- **Order ID:** Nomor unik yang mengidentifikasi setiap pesanan.
- **Customer Name:** Nama pelanggan yang melakukan pemesanan.
- **Area dan Sub-Area:** Lokasi pelanggan, termasuk wilayah utama dan wilayah spesifik.
- **Phone Number:** Nomor telepon pelanggan untuk keperluan komunikasi.
- **Tabel Produk:** Rincian menu yang dipesan, mencakup:
 - * Harga per menu.
 - * Jumlah (Quantity/Qty) dari setiap menu yang dipesan.
 - * Total harga untuk setiap menu berdasarkan jumlah yang dipesan.
- **Total Harga Pesanan:** Total keseluruhan harga pesanan, dihitung dari hasil kalkulasi seluruh produk yang dipesan oleh pelanggan.

Halaman Orders memastikan data pesanan tersimpan secara terstruktur dan mempermudah admin dalam mengelola detail transaksi, termasuk memantau *produk, harga, dan jumlah pesanan*. Dengan fitur yang intuitif seperti **View Invoice**, admin dapat dengan cepat meninjau dan mencetak dokumen pesanan untuk kebutuhan operasional.

Order ID	Customer N...	Area	Sub Area	Address	Phone Num...	Jenis P...
CA022220241706	Gratia Elena	Tangerang	Wahidin	Jalan alam sutera boulevard komplek ruko the prominence blok 38 G no 10 alam sutera	081808111138	Langganan
CA022220241706	Gratia Elena	Tangerang	Johor	Jalan alam sutera boulevard komplek ruko the prominence blok 38 G no 10 alam sutera	081808111138	Langganan
CA022220241706	Gratia Elena	Tangerang	Titik Kuning	Jalan alam sutera boulevard komplek ruko the prominence blok 38 G no 10 alam sutera	081808111138	Langganan
CA030320241749	Kaydra	Tangerang	Wahidin	Jl. Ansari	081717117444	Langganan
CA030320241749	Kaydra	Tangerang	Johor	Jl. Ansari	081717117444	Langganan
CA030320241749	Kaydra	Tangerang	Titik Kuning	Jl. Ansari	081717117444	Langganan

Gambar III.25. Tampilan Tabel Halaman Orders.

Database yang digunakan dalam pengembangan sistem ini adalah PhpMyAdmin [3]. Pengelolaan database dilakukan melalui PhpMyAdmin [3] sebagai antarmuka grafis yang memudahkan admin dalam memonitor tabel, membuat relasi antar entitas, serta menjalankan kueri untuk manipulasi data.

A.5 Potongan Koding

JavaScript Code for deletePartnerMenu

Berikut adalah kode JavaScript untuk fungsi deletePartnerMenu:

```

1 function deletePartnerMenu(id, button) {
2   // Stage 1: Initial confirmation (Are you sure?)
3   Swal.fire({
4     title: 'Are you sure?',
5     text: "You won't be able to revert this!",
6     icon: 'warning',
7     showCancelButton: true,
8     confirmButtonText: 'Yes, delete it!',
9     cancelButtonText: 'Cancel',
10    reverseButtons: true
11  }).then((result) => {
12    if (result.isConfirmed) {
13      // Stage 2: Ask for a reason (Optional)
14      Swal.fire({
15        title: 'Please provide a reason for deleting
16        this product',
17        input: 'textarea',
18        inputPlaceholder: 'Reason for deletion...',

```



```

18         showCancelButton: true,
19         confirmButtonText: 'Submit Reason and
Proceed',
20         cancelButtonText: 'Skip',
21         preConfirm: (reason) => {
22             if (reason && reason.trim().length > 0)
{
23                 return reason;
24             } else {
25                 return 'No reason provided';
26             }
27         }
28     }).then((result) => {
29         if (result.isConfirmed) {
30             const reason = result.value;
31
32             // Stage 3: Final confirmation (Are you
really sure?)
33             Swal.fire({
34                 title: 'Final Confirmation',
35                 text: `Are you sure you want to
delete this product? Reason: ${reason}`,
36                 icon: 'question',
37                 showCancelButton: true,
38                 confirmButtonText: 'Yes, delete it
!',
39                 cancelButtonText: 'Cancel',
40                 reverseButtons: true
41             }).then((finalResult) => {
42                 if (finalResult.isConfirmed) {
43                     axios.delete(`/product/delete/${
id}`, {
44                         headers: {
45                             'X-CSRF-TOKEN': '{{
csrf_token() }}',
46                             'Content-Type': '
application/json',
47                             'Accept': 'application/

```

```

48         json'
49         })
50         .then(response => {
51             const data = response.data;
52             if (data.success) {
53                 button.closest('tr').
remove ();
54                 Swal.fire({
55                     title: 'Deleted!',
56                     text: data.message,
57                     icon: 'success',
58                     confirmButtonText: '
OK'
59                 });
60             } else {
61                 Swal.fire({
62                     title: 'Error!',
63                     text: 'Failed to
delete product: ' + data.message,
64                     icon: 'error'
65                 });
66             }
67         })
68         .catch(error => {
69             Swal.fire({
70                 title: 'Error!',
71                 text: 'Failed to delete
product due to server error.',
72                 icon: 'error'
73             });
74         });
75         } else if (finalResult.dismiss ===
Swal.DismissReason.cancel) {
76             Swal.fire('Cancelled', 'Your
product was not deleted.', 'info');
77         }
78     });

```

```

79         } else if (result.dismiss === Swal.
DismissReason.cancel) {
80             Swal.fire({
81                 title: 'Final Confirmation',
82                 text: 'Are you sure you want to
delete this product?',
83                 icon: 'question',
84                 showCancelButton: true,
85                 confirmButtonText: 'Yes, delete it
!',
86                 cancelButtonText: 'Cancel',
87                 reverseButtons: true
88             }).then((finalResult) => {
89                 if (finalResult.isConfirmed) {
90                     axios.delete(`/product/delete/${
id}` , {
91                         headers: {
92                             'X-CSRF-TOKEN': '{{
csrf_token() }}',
93                             'Content-Type': '
application/json',
94                             'Accept': 'application/
json'
95                         }
96                     })
97                     .then(response => {
98                         const data = response.data;
99                         if (data.success) {
100                             button.closest('tr').
remove();
101                             Swal.fire({
102                                 title: 'Deleted!',
103                                 text: data.message,
104                                 icon: 'success',
105                                 confirmButtonText: '
OK '
106                             });
107                     } else {

```

```

108         Swal.fire({
109             title: 'Error!',
110             text: 'Failed to
delete product: ' + data.message,
111             icon: 'error'
112         });
113     }
114 })
115     .catch(error => {
116         Swal.fire({
117             title: 'Error!',
118             text: 'Failed to delete
product due to server error.',
119             icon: 'error'
120         });
121     });
122     } else if (finalResult.dismiss ===
Swal.DismissReason.cancel) {
123         Swal.fire('Cancelled', 'Your product
was not deleted.', 'info');
124     }
125     });
126     }
127     });
128     } else {
129         Swal.fire('Cancelled', 'Your product was not
deleted.', 'info');
130     }
131     });
132 }

```

B. Database

Database yang digunakan dalam pengembangan sistem ini adalah PhpMyAdmin [3]. Pengelolaan database dilakukan melalui PhpMyAdmin [3] sebagai antarmuka grafis yang memudahkan admin dalam memonitor tabel, membuat relasi antar entitas, serta menjalankan kueri untuk manipulasi data.

B.1 Struktur Tabel

Dalam Pembuatan sistem Website Portal Administrasi terdapat beberapa tabel yang sangat penting untuk mendukung pengelolaan data dan tampilan antarmuka pengguna.

- Pembuatan Sistem Login

Sistem database Login menggunakan dua tabel yang berbeda. Tabel user yang berisi data kitchen partner digunakan untuk login kitchen partner pada route “/user/login”. Sedangkan untuk tabel users yang berisi data admin terdapat di route “/”.

- Pembuatan Sistem Add Market Area

Sistem Add Market Area memiliki hubungan atau relasi antara tabel daerah dan daerah_sub. Terjadi relasi **One-to-Many** antara tabel daerah dengan daerah_sub. Penggunaan relasi One-to-Many karena satu daerah bisa memiliki banyak sub daerah. Pada tabel dibawah ini, wilayah meliputi nama daerah dalam jangkauan luas. Sedangkan nama pada tabel daerah_sub merupakan kolom dari nama daerah dalam jangkauan kecil. Pada contoh case adalah market area Tangerang memiliki sub area Gading Sepong, Alam Sutera, BSD, Karawaci, dll.

Tabel III.3. Struktur Database Tabel *daerah*.

Name	Type	Keterangan
id_daerah	int(11)	Primary Key
wilayah	varchar(70)	Nama wilayah.
latitude_city	varchar(50)	-
longitude_city	varchar(50)	-
status_area	int(11)	Berupa Active atau Inactive

Tabel III.4. Struktur Database Tabel *daerah_sub*.

Name	Type	Keterangan
id_sub	int(11)	Primary Key
daerah	int(11)	Foreign Key lokasi.
nama	int(11)	varchar(50)
status_wilayah	int(11)	Berisi Active dan Inactive

id_daerah	wilayah	id_sub	nama	daerah
		1	Gading Serpong	1
1	Tangerang	2	Alam Sutera	1
2	Jakarta	3	BSD	1
		4	Sudirman	2

- Pembuatan Sistem Multiple Select Area

Sistem Multiple Select Area dirancang untuk memungkinkan pemilihan lebih dari satu sub-area dalam suatu *market area*. Hal ini bertujuan agar sebuah kitchen partner dapat melayani berbagai sub-area dalam cakupan wilayah tertentu. Dalam implementasinya, sistem ini memanfaatkan relasi **Many-to-Many** antara beberapa tabel utama, yaitu tabel daerah (market areas), daerah_sub (sub_areas), user (partner), dan data_area. Tabel data_area berfungsi sebagai penghubung (junction table) yang mencatat relasi antar ketiga tabel tersebut dalam bentuk ID atau nilai integer. Pada tabel di bawah ini, lokasi merupakan representasi dari sub_area yang dipilih secara banyak.

Tabel III.5. Struktur Database Tabel *data_area*.

Name	Type	Keterangan
id_areaKitchen	int(11)	Primary Key
id_user	int(11)	id user kitchen partner.
lokasi	int(11)	Berisi id lokasi

- Pembuatan Sistem Kitchen List

Kitchen List merupakan hasil tampilan data kitchen partner dari kumpulan data tabel user dengan *Primary Key* id_user. Tabel user memiliki relasi One-to-Many dengan tabel menu. Tabel menu merupakan kumpulan data menu masing-masing kitchen partner. Sehingga satu kitchen partner bisa memiliki banyak menu. Untuk menghubungkan relasi tersebut, tabel menu memiliki foreign key merchant yang merepresentasikan kitchen partner.

- Pembuatan sistem Order dan Payment

Pada pembuatan sistem Order dan Payment melibatkan tabel order_customer dan order_menu. Tabel order_customer memiliki relasi dengan tabel status dan order_menu. Pada order_customer terdapat kolom status yang memiliki relasi dengan tabel status digunakan untuk memanggil id_status dan memunculkan nama_status seperti *Pending, Rejected, dan Accepted*.

Penjelasan relasi antar tabel.

- Tabel order_menu berfungsi sebagai penghubung (pivot table) memiliki relasi *Many-to-Many* dengan tabel order_customer, user, status dan menu. Tabel ini berisi id dan integer digunakan untuk memanggil data menu yang dipesan yang telah dikalkulasi pada fungsi calculate.
- Tabel order_customer memiliki relasi One-to-Many dengan order_menu, sehingga dalam penjelasannya adalah satu customer bisa memesan banyak jenis menu dan dapur yang berbeda-beda.

Tabel III.6. Struktur Database Tabel *order_menu*.

Name	Type	Keterangan
id_orderMenu	int(11)	Primary Key
id_order	int(11)	ID order pesanan.
kode_order	varchar(50)	Kode Order pesanan
date	date	Waktu dan tanggal
kitchen	int(11)	Berisi id kitchen partner
menu	int(11)	Berisi id menu
qty	int(11)	Berisi total barang
harga	int(30)	Berisi total harga
paket	int(11)	Berisi id jenis paket
Lanjut pada halaman berikutnya		

Tabel III.6 A sample long table (lanjutan)

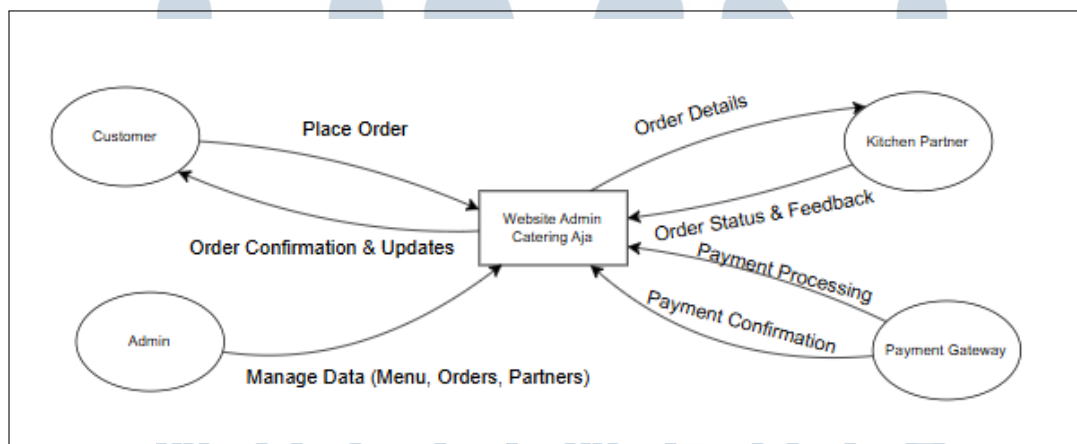
First column	Second column	Third column
kirim	int(11)	-
kurir	int(11)	-
note	varchar(400)	Keterangan
status_order	int(11)	Berisi id status

B.2 Struktur Data Flow Diagram

Data Flow Diagram (DFD) merupakan tampilan representasi visual yang menunjukkan aliran data dalam sebuah sistem informasi. DFD berfungsi membantu menggambarkan dengan jelas bagaimana pergerakan data pada suatu sistem. Membuatnya lebih mudah dipahami baik oleh pengguna teknis maupun non-teknis. DFD menyediakan informasi mengenai input dan output dari setiap entitas dan proses yang ada dalam sistem tersebut[10].

Berikut adalah penerapan Data Flow Diagram (DFD) pada pembuatan website backend.

1. DFD Level-0

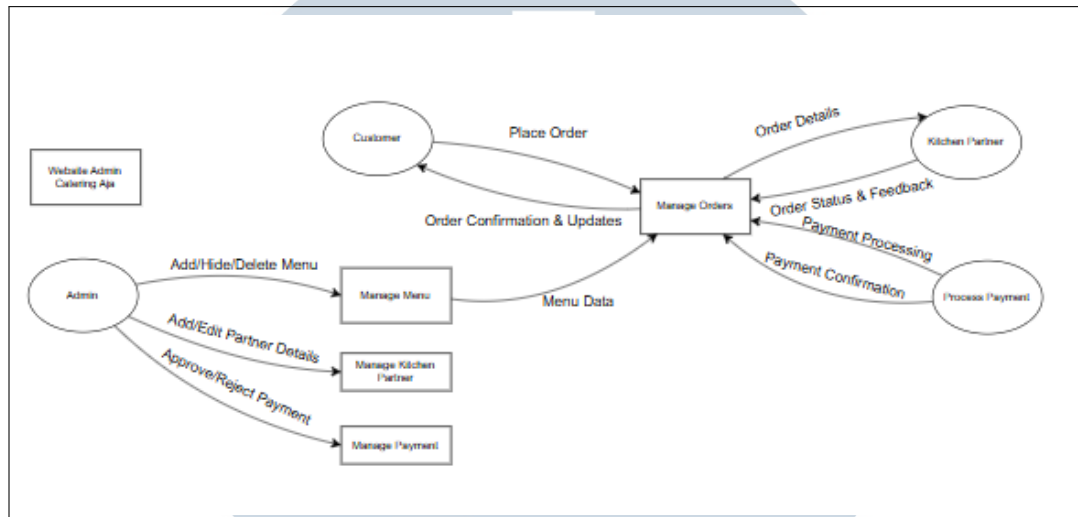


Gambar III.26. Data Flow Diagram Level-0.

Pada DFD Level-0 ini terlihat bahwa Website Admin Catering Aja memiliki banyak aktivitas yang dapat dilakukan. Website admin dapat digunakan Admin dalam manajemen data menu (menambah menu, menghapus menu, dan hide menu), Customer untuk pemesanan menu, perhitungan pesanan,

serta Kitchen Partner untuk mendapatkan pesanan, dan melemparkan status pesanan ke website admin.

2. DFD Level-1



Gambar III.27. Data Flow Diagram Level-1.

DFD Level-1 merupakan level yang lebih rinci dari DFD Level-0, dimana kita mulai memecah sistem utama dari yang digambarkan pada level-0 menjadi beberapa subproses. Level 1 memperlihatkan bagaimana suatu sistem bekerja dengan memecah proses awal menjadi bagian yang lebih spesifik, serta aliran data diantara proses-proses tersebut.

Elemen-elemen dalam DFD Level-1:

- Proses Utama yang Diperinci

Di Level-1, proses yang ada pada DFD Level-0 dipecah menjadi beberapa subproses yang lebih kecil. Misalnya, dalam sistem pemesanan catering, proses utama yang ada pada Level-0 seperti "Sistem Pemesanan Catering" bisa dibagi menjadi subproses seperti "Manajemen Pesanan", "Proses Pembayaran", dan "Manajemen Menu".

- Entitas Eksternal

Entitas yang berada di luar sistem, yang berinteraksi dengan sistem melalui aliran data. Di Level-1, entitas eksternal seperti Customer (Pelanggan), Admin, atau Payment Confirmation akan tetap ada, namun sekarang mereka berinteraksi dengan subproses yang lebih terperinci.