

# **SOAL**

**LOMBA KOMPETENSI SISWA (LKS) SMK  
Tingkat Provinsi Daerah Istimewa Yogyakarta  
Tahun 2025**



# **LKS SMK**

**Bidang Lomba  
IT Software Solution for Business**

**Tempat Lomba:  
Balai Latihan Pendidikan Teknik Yogyakarta  
Jl. Kyai Mojo No.70, Bener, Kec. Tegalrejo, Kota Yogyakarta,  
Daerah Istimewa Yogyakarta.**

## SKEMA PERSKORAN

Persentase modul penskoran setiap ditunjukkan pada tabel di bawah ini:

MODUL	DESKRIPSI	SKOR (%)
1	Database & Backend Development	25%
2	Admin Panel Development (Web based)	30%
3	Mobile Development	20%
4	Data Access Security and Payment Gateway Integration	10%
5	Business Intelligence & Reporting	15%

Tabel berikut ini menjelaskan spesifikasi yang dibutuhkan dan rincian yang dinilai :

Modul 1		
1.	Membuat ( <i>create</i> ) model fisik <i>database</i>	
	a.	Semua entitas diubah ke dalam format <i>database</i> objek seperti yang diinstruksikan pada <i>test project</i>
	b.	Mendefinisikan hubungan antar entitas
	c.	Semua <i>constraint</i> yang menjadi <i>requirements</i> diimplementasikan agar mendukung sistem
	d.	Membuat seeder insert data admin
2.	Membuat REST API (JSON File) dengan teknologi backend (PHP/Python)	
	a.	Membuat API Create, Read, Update, dan Delete (CRUD) untuk master Meja
	b.	Membuat API CRUD untuk master Menu
	c.	Membuat API untuk register pelanggan di mobile apps
	d.	Membuat API untuk login pelanggan di mobile apps (Integrasi JWT Auth Token)
	e.	Membuat API untuk login admin di web based admin panel (JWT Auth Token)
	f.	Membuat API logout untuk pelanggan (mobile apps)
	g.	Membuat API logout untuk admin (web based)
	h.	Membuat API untuk manajemen pelanggan (CRUD untuk admin)
	i.	Membuat API untuk transaksi reservasi tempat dan menu (update status meja)

	j.	Membuat API Read, Update dan Delete (RUD) transaksi reservasi dari sisi admin
	k.	Membuat API update status transaksi
	l.	Membuat API pembayaran dengan payment gateway (untuk DP booking/reservasi);
	m.	Membuat API Callback URL ketika pelanggan berhasil melakukan pembayaran DP (update status reservasi, status meja dan status transaksi)
	n.	Membuat CRUD untuk transaksi dan detail transaksi dari sisi admin
	o.	Membuat API menyelesaikan transaksi dari sisi admin (ketika pelanggan selesai makan dan melakukan pembayaran di kasir, update data transaksi dan otomatis status meja akan berubah)
	p.	Membuat API reporting jumlah transaksi (range tanggal)
	q.	Membuat API reporting 5 makanan terlaris
	r.	Membuat API reporting jumlah nominal transaksi (range tanggal)
	s.	Membuat API get data transaksi berdasarkan pelanggan
3.		Membuat notifikasi email
	a.	Membuat notifikasi email untuk admin jika ada transaksi baru
	b.	Membuat notifikasi email pengiriman kode reservasi kepada pelanggan
	c.	Membuat fitur email sending untuk mengirimkan invoice kepada pelanggan
<b>Modul 2</b>		
1.		Pembuatan form dan halaman user interface admin & mengintegrasikan dengan REST API
	a.	Membuat halaman login admin
	b.	Membuat halaman form tambah data meja
	c.	Membuat halaman list tampil data meja
	d.	Membuat halaman form edit data meja
	e.	Membuat aksi hapus data meja
	f.	Membuat halaman list data pelanggan
	g.	Membuat aksi hapus data pelanggan
	h.	Membuat halaman form tambah data menu

	i.	Membuat halaman list tampil data menu
	j.	Membuat halaman form edit data menu
	k.	Membuat aksi hapus data menu
	l.	Membuat halaman list tampil data transaksi
	m.	Membuat halaman tampil data detail transaksi
	n.	Membuat halaman edit data transaksi (jika ada kemungkinan pelanggan memesan menu tambahan)
	o.	Membuat halaman interface pembayaran (pelunasan) setelah pelanggan selesai makan
Modul 3		
1.	Membuat interface Mobile UI & mengintegrasikan dengan REST API	
	a.	Membuat halaman register pengguna
	b.	Membuat halaman login pengguna
	c.	Membuat component menu (Home, Meja, Menu Makanan/Minuman, Transaksi, Logout)
	d.	Membuat halaman interface Home (Static Content)
	e.	Membuat halaman interface List Meja
	f.	Membuat halaman interface List Menu Makanan/Minuman
	g.	Membuat interface proses reservasi (Memilih meja lalu memilih menu makanan/minuman)
	h.	Membuat interface proses pembayaran DP dengan integrasi Payment Gateway
	i.	Membuat interface list transaksi (history + ongoing)
	j.	Membuat interface halaman detail transaksi
Modul 4		
1.	Mengintegrasikan JWT Auth Token untuk Autentikasi pada aplikasi Web Based dan Mobile	
2.	Mengintegrasikan pembayaran berbasis third party (Payment Gateway) dengan Xendit/Midtrans/dll.	
Modul 5		

1.	Membuat halaman dashboard (menampilkan report jumlah transaksi berdasarkan rentang tanggal, report 5 makanan terlaris, report jumlah nominal transaksi per rentang tanggal) untuk admin Panel
----	---

### WAKTU JADWAL

No	Hari, Tanggal	Waktu	Kegiatan
1	Kamis, 10 April 2025	08.00-10.30	Pengkondisian dan pemeriksaan perangkat (PC atau Laptop). dimana sudah terinstal <i>software</i> atau program yang diperlukan serta akun payment gateway dan email client sender dalam pengembangan aplikasi bisnis sesuai ketentuan perlombaan.  Kontak: Rio Aska: +62 877-3899-8884
		10.30-11.00	Pengundian nomor peserta
		11.00-12.00	Penjelasan teknis
		12.00-13.00	ISHOMA
		13.00-16.00	Proses seleksi hari ke-1 ( <i>Database &amp; Backend Development</i> )
2	Jum'at, 11 April 2025	08.00-08.30	Penjelasan teknis
		08.30-11.30	Proses seleksi hari ke-2 ( <i>Admin Panel Development (Web based)</i> )
		11.30-13.00	ISHOMA
		13.00-16.00	Proses seleksi hari ke-2 ( <i>Admin Panel Development (Web based)</i> ) & <i>Mobile Development</i>
3	Sabtu, 12 April 2025	08.00-08.30	Penjelasan teknis

		08.30-12.00	Proses seleksi hari ke-3 ( <i>Mobile Development, Data Access Security and Payment Gateway Integration &amp; Business Intelligence &amp; Reporting</i> )
		12.00-13.00	ISHOMA
		13.00-14.00	Penilaian juri

### ULASAN PROYEK

Dalam kompetisi ini, peserta diminta untuk mengembangkan sistem reservasi restoran berbasis digital, yang mencakup reservasi tempat duduk, pemesanan makanan, dan integrasi pembayaran online.

Peserta akan mendapatkan empat sumber referensi utama dalam proyek ini:

#### 1. ERD dan Kamus Data

Peserta akan mendapatkan Entity-Relationship Diagram (ERD) dan Kamus Data sebagai panduan utama dalam perancangan database.

- a. ERD digunakan untuk memastikan bahwa semua entitas yang dibuat dalam database memiliki hubungan dan relasi yang tepat sesuai dengan kebutuhan sistem.
- b. Kamus Data berisi detail atribut setiap entitas, tipe data, serta batasan yang harus diterapkan.
- c. Peserta harus memastikan bahwa seluruh struktur database yang dikembangkan sesuai dengan standar normalisasi dan tidak mengandung data redundan.

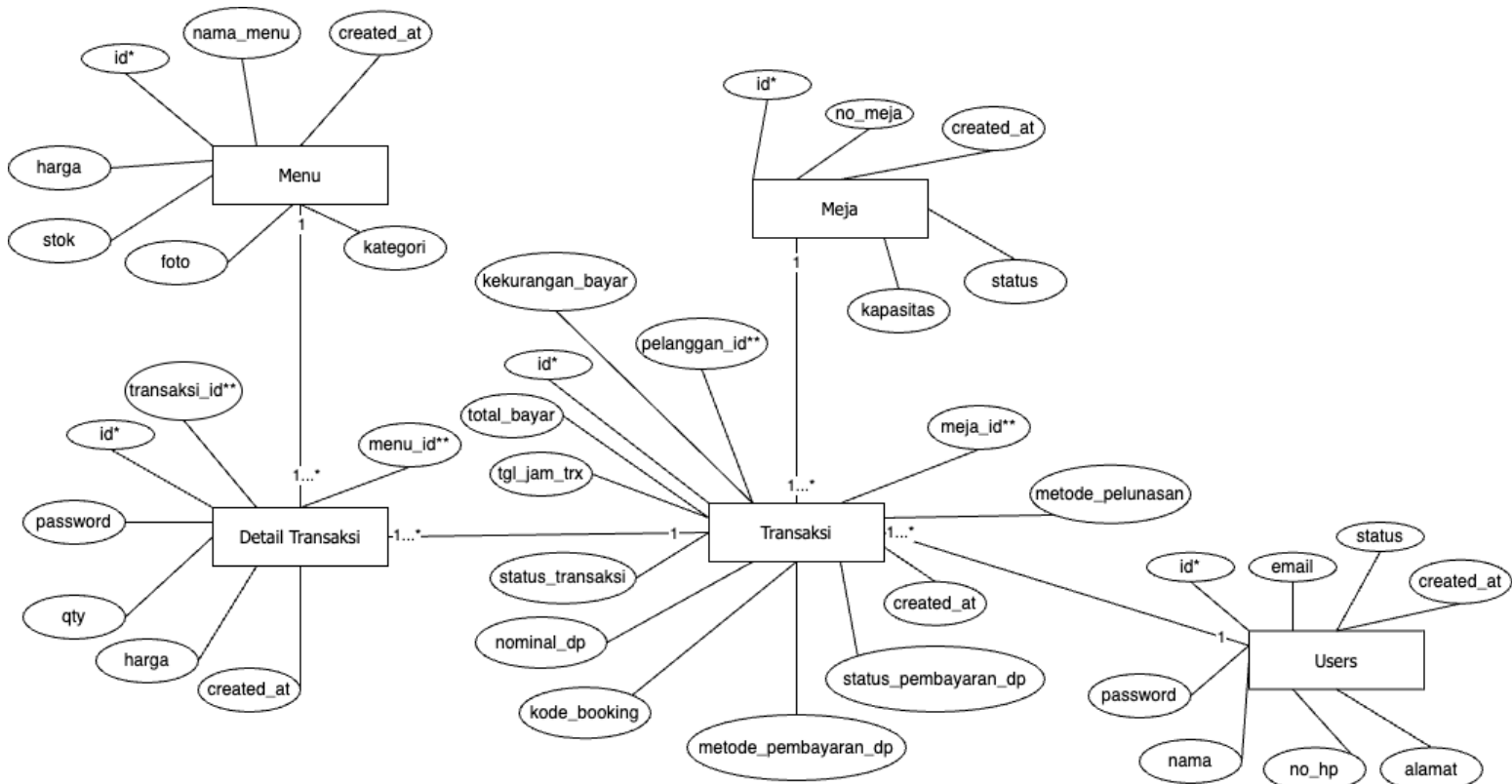
#### 2. Proses Bisnis & Alur Kerja Sistem

- a. Peserta akan mengembangkan sistem Smart Restaurant Reservation & Ordering System untuk Restoran ESEMKA - DIY. Sistem ini mencakup Admin Panel (Web) untuk restoran dan Aplikasi Mobile (Pelanggan) untuk pengguna.

## ERD dan Kamus Data

### 1. Entity Relational Database (ERD)

#### Smart Restaurant Reservation & Ordering System



### 2. Kamus Data

Table	PK	FK	Column	Required	Type	Length	Notes
users	PK		id	Yes	int		Auto Increment/Identity
			email	Yes	varchar	100	
			password	Yes	varchar	255	
			status	Yes	enum		['admin','pelanggan']
			nama	Yes	varchar	100	
			no_hp	Yes	varchar	20	
			alamat	Yes	text		
			created_at	No	timestamp		current_timestamp
meja	PK		id	Yes	int		Auto Increment/Identity
			no_meja	Yes	int		unique
			kapasitas	Yes	int		default:2
			status	Yes	enum		['kosong','dipesan','terisi']

			created_at	No	timestamp		current_timestamp
menu	PK		id	Yes	int		Auto Increment/Identity
			nama_menu	Yes	varchar	50	
			harga	Yes	double		
			stok	Yes	int		
			foto	No	text		
			created_at	No	timestamp		current_timestamp
transaksi	PK		id	Yes	int		Auto Increment/Identity
		FK	pelanggan_id	Yes	int		Foreign Key (pelanggan)
		FK	meja_id	Yes	int		Foreign key (meja)
			kode_booking	Yes	varchar	20	unique
			tgl_jam_trx	No	datetime		current_timestamp
			status_transaksi	Yes	enum		['pending',' reserved', 'checkin', 'done','failed']
			nominal_dp	Yes	double		
			metode_pembayaran_dp	Yes	varchar	50	
			status_pembayaran_dp	Yes	enum		['deny ' , 'pending', 'cancel', 'settlement', 'expired', 'refund']
			total_bayar	Yes	double		
			kekurangan	No	double		
			metode_pembayaran_trx	Yes	varchar		
			created_at	No	timestamp		current_timestamp
detail_transaksi	PK		id	Yes			Auto Increment/Identity
		FK	transaksi_id	Yes	int		
		FK	menu_id				
			qty	Yes	int		
			harga	Yes	double		
			created_at	No	timestamp		current_timestamp

## DEVELOPMENT SISTEM INFORMASI RESERVASI & PEMESANAN RESTORAN ESEMKA - DIY

Restoran ESEMKA, yang berlokasi di Provinsi Daerah Istimewa Yogyakarta (DIY), merupakan salah satu restoran yang terkenal dengan konsep modern dan layanan berkualitas tinggi. Dengan meningkatnya jumlah pelanggan, restoran ini menghadapi tantangan dalam pengelolaan reservasi tempat duduk dan pemesanan makanan. Banyak pelanggan yang datang langsung ke restoran hanya untuk mengetahui bahwa tempat sudah penuh atau harus menunggu lama tanpa kepastian.



Dewan manajemen ESEMKA menyadari bahwa mereka membutuhkan solusi digital yang terintegrasi untuk mengelola reservasi meja, pemesanan makanan, serta pembayaran secara online. Sistem ini diharapkan dapat membantu restoran mengoptimalkan operasional mereka, meningkatkan kepuasan pelanggan, serta meminimalkan antrian dan ketidakpastian dalam mendapatkan tempat duduk.

Sebagai seorang konsultan dan programmer, Anda ditugaskan untuk menganalisis, merancang, dan membangun sistem reservasi restoran berbasis digital. Sistem ini harus mencakup backend API, web admin panel untuk pengelolaan restoran, aplikasi mobile untuk pelanggan, serta fitur integrasi pembayaran digital dan laporan bisnis.

Dalam pengembangan sistem ini, Anda harus memperhatikan instruksi teknis dan kebutuhan bisnis dengan cermat, sehingga sistem yang dikembangkan dapat berfungsi secara efisien, aman, dan sesuai dengan praktik terbaik industri restoran.

Tantangan utama dalam proyek ini:

1. Mengatasi ketidakpastian reservasi tempat dengan sistem monitoring real-time yang menampilkan ketersediaan meja.
2. Meningkatkan efisiensi layanan dengan aplikasi mobile yang memungkinkan pelanggan memilih meja, memesan makanan, dan membayar langsung.
3. Membantu restoran dalam analisis bisnis dengan fitur laporan transaksi, jumlah reservasi per periode, serta menu favorit pelanggan.

Anda diharapkan dapat mengembangkan sistem ini dengan teknologi modern dan memastikan bahwa solusi yang Anda buat mampu meningkatkan efektivitas operasional restoran ESEMKA di DIY secara signifikan!

## **Modul 1: Database & Backend Development**

### **1. Database and Table Creation**

Buat database menggunakan MySQL atau PostgreSQL di server database lokal Anda dengan nama "RestaurantESEMKA\_DB". Setelah itu, buat tabel yang dibutuhkan sesuai dengan ERD dan Kamus Data yang telah diberikan.

Pastikan setiap tabel memiliki:

- a. Primary Key (PK)
- b. Foreign Key (FK)

- c. Constraint (NOT NULL, UNIQUE, DEFAULT, CHECK)
- d. Tipe data yang sesuai

**Tantangan bagi peserta:**

Menambahkan tabel (di luar tabel yang sudah diberikan di atas pada ERD dan kamus data) yang mampu mengakomodir proses pembayaran menggunakan Payment Gateway, yaitu dapat otomatis mengupdate status transaksi ketika pelanggan gagal/berhasil melakukan pembayaran DP (Callback URL).

2. Develop REST API (Backend Development)

Membangun REST API menggunakan PHP Laravel atau Python Flask/Django untuk menangani operasi CRUD (Create, Read, Update, Delete) dan transaksi bisnis sesuai kebutuhan dari proses bisnis yang berjalan.

**Modul 2: Admin Panel Development (Web based)**

1. Halaman Login Admin

- a. Admin mengakses halaman login.
- b. Admin memasukkan username dan password.
- c. Sistem memvalidasi kredensial melalui REST API (JWT Authentication).
- d. Jika valid, admin diarahkan ke dashboard utama.
- e. Jika gagal, sistem menampilkan pesan error.

2. Halaman Form Tambah Data Meja

- a. Admin mengisi data meja.
- b. Klik "Simpan", sistem memvalidasi input.
- c. Jika valid, data dikirim ke backend untuk disimpan dalam database.
- d. Jika sukses, sistem menampilkan pesan sukses dan membersihkan form.
- e. Jika gagal, sistem menampilkan pesan gagal dan kembali pada form inputan sebelumnya

3. Halaman List Data Meja

- a. Admin melihat daftar semua meja dalam bentuk tabel.
- b. Sistem mengambil data meja dari REST API.

- c. Admin dapat mengklik tombol "Edit" atau "Hapus" untuk mengelola meja.
- 4. Halaman Form Edit Data Meja
  - a. Admin memilih meja yang ingin diedit dari daftar.
  - b. Sistem menampilkan detail data meja dalam form.
  - c. Admin mengedit data dan menekan tombol "Update".
  - d. Sistem mengirim data yang diperbarui ke backend.
- 5. Aksi Hapus Data Meja
  - a. Admin memilih meja yang ingin dihapus.
  - b. Sistem menampilkan konfirmasi penghapusan.
  - c. Jika admin menekan "Ya", data meja dihapus dari database.
  - d. Jika admin menekan "Batal", proses dibatalkan.
- 6. Halaman List Data Pelanggan
  - a. Admin melihat daftar pelanggan yang sudah mendaftar.
  - b. Sistem mengambil data pelanggan dari REST API.
  - c. Admin bisa menghapus pelanggan yang bermasalah.
- 7. Aksi Hapus Data Pelanggan
  - a. Admin memilih pelanggan yang ingin dihapus.
  - b. Sistem menampilkan konfirmasi sebelum menghapus.
  - c. Jika admin menekan "Ya", pelanggan dihapus dari database.
- 8. Halaman Form Tambah Data Menu
  - a. Admin mengisi data menu.
  - b. Klik "Simpan", sistem memvalidasi input.
  - c. Data dikirim ke backend untuk disimpan.
  - d. Jika sukses, sistem menampilkan pesan sukses dan membersihkan form.
  - e. Jika gagal, sistem menampilkan pesan gagal dan kembali pada form inputan sebelumnya,
- 9. Halaman List Data Menu
  - a. Admin melihat daftar semua menu dalam bentuk tabel.
  - b. Sistem mengambil data menu dari REST API.
  - c. Admin bisa mengedit atau menghapus menu.
- 10. Halaman Form Edit Data Menu

- a. Admin memilih menu yang ingin diperbarui.
- b. Sistem menampilkan detail menu dalam form.
- c. Admin mengubah data menu dan menekan "Update".
- d. Data dikirim ke backend untuk diperbarui.
- e. Jika sukses, sistem menampilkan pesan sukses dan mempertahankan form dengan isian yang terbaru.
- f. Jika gagal, sistem menampilkan pesan gagal dan kembali pada form edit data.

#### 11. Aksi Hapus Data Menu

- a. Admin memilih menu yang ingin dihapus.
- b. Sistem menampilkan konfirmasi sebelum menghapus.
- c. Jika admin menekan "Ya", menu dihapus dari database dan menampilkan pesan data berhasil dihapus.

#### 12. Halaman List Data Transaksi

- a. Admin melihat daftar transaksi yang telah dilakukan pelanggan.
- b. Sistem mengambil data transaksi dari REST API.
- c. Admin bisa mengklik "Lihat Detail" untuk melihat informasi lengkap, mengedit atau menghapus data transaksi.

#### 13. Halaman Detail Transaksi

- a. Admin memilih transaksi yang ingin dilihat lebih detail.
- b. Sistem menampilkan informasi pelanggan, pesanan, status pembayaran.

#### 14. Halaman Edit Data Transaksi

- a. Admin bisa menambahkan pesanan baru ke transaksi pelanggan (misalnya pelanggan menambah pesanan).
- b. Admin memilih menu tambahan dan jumlahnya.
- c. Data transaksi diperbarui dan disimpan ke backend.
- d. Jika sukses, sistem menampilkan pesan sukses dan mempertahankan form dengan isian yang terbaru.
- e. Jika gagal, sistem menampilkan pesan gagal dan kembali pada form edit data.

#### 15. Halaman Interface Pembayaran (Pelunasan Setelah Makan)

- a. Admin memilih transaksi pelanggan yang akan dilunasi/melakukan pembayaran di kasir (pelanggan selesai makan).

- b. Sistem menampilkan total tagihan dan detail pesanan pelanggan.
- c. Admin memilih metode pembayaran (Tunai, QRIS, Kartu Debit/Kredit => pilihan static dan tidak menggunakan payment gateway, hanya untuk proses pencatatan saja).
- d. Sistem mencatat pembayaran dan memperbarui status transaksi menjadi "done" serta sistem melakukan update status meja tempat makan dari pelanggan menjadi "kosong".
- e. Jika pelanggan menginginkan struk/invoice, maka sistem dapat mengirimkan struk/invoice kepada pelanggan melalui email.

### **Modul 3: Mobile Development**

Membuat Interface Mobile UI & Integrasi dengan REST API. Aplikasi ini akan digunakan oleh pelanggan untuk melakukan reservasi meja, memesan makanan, serta melakukan pembayaran secara online. Semua fitur harus diintegrasikan dengan REST API.

1. Halaman Register Pengguna
  - a. Pengguna mengakses halaman register.

- b. Mengisi nama, email, nomor HP, alamat dan password.
  - c. Klik "Daftar", sistem memvalidasi input.
  - d. Jika valid, data dikirim ke backend untuk disimpan.
  - e. Jika valid dan berhasil, maka muncul pesan sukses dan diarahkan pada halaman login
  - f. Jika tidak valid dan gagal, maka muncul pesan error yang mengakibatkan gagal dan pengguna diarahkan pada halaman register dengan data yang diinputkan sebelumnya agar dapat diperbaiki
2. Halaman Login Pengguna
- a. Pengguna mengakses halaman login.
  - b. Memasukkan email & password.
  - c. Klik "Login", sistem memvalidasi kredensial melalui REST API.
  - d. Jika berhasil, backend mengirim token JWT dan user diarahkan ke halaman utama.
3. Membuat Component Menu (Navigasi)
- Component ini berisi navigasi utama dalam aplikasi:
- a. Home → Halaman utama
  - b. Meja → Daftar meja yang tersedia
  - c. Menu Makanan/Minuman → List menu restoran
  - d. Transaksi → Daftar transaksi pelanggan
  - e. Logout → Keluar dari akun
- Tidak ada API langsung, hanya pengelolaan state dan navigasi antar halaman.
4. Halaman Interface Home (Static Content)
- Halaman ini berisi informasi restoran, promo terbaru, serta layanan yang tersedia.
- a. Menggunakan static content untuk tampilan yang menarik.
  - b. Bisa menampilkan carousel promo atau banner diskon.
- Tidak ada API langsung, hanya untuk tampilan informasi.
5. Halaman Interface List Meja
- a. Pengguna melihat daftar meja yang tersedia dalam bentuk grid atau list yang memperlihatkan informasi kapasitas orang untuk setiap meja.
  - b. Sistem mengambil data meja dari backend.
  - c. Pengguna bisa memilih meja untuk reservasi.

- d. Setelah pengguna memilih meja, kemudian akan diarahkan untuk memilih menu makanan dan minuman
6. Halaman Interface List Menu Makanan/Minuman
- a. Pengguna dapat melihat daftar menu yang tersedia dengan informasi nama menu, foto/gambar, harga serta stok yang tersedia.
  - b. Pengguna bisa menambahkan menu ke keranjang pesanan.
  - c. Sistem mengambil data menu dari backend (REST API).
  - d. Setelah pengguna selesai memilih menu makanan dan minuman, selanjutnya dapat menyelesaikan proses reservasi dengan menekan tombol lanjutkan untuk memilih metode pembayaran.
7. Interface Proses Reservasi
- a. Pengguna memilih meja yang tersedia.
  - b. Sistem menampilkan opsi pemilihan menu makanan/minuman.
  - c. Pengguna memilih menu dan menambahkan ke pesanan.
  - d. Sistem dapat menghitung nominal DP yaitu 50% dari total pesanan.
  - e. Pengguna memilih metode pembayaran untuk membayar DP.
  - f. Klik "Konfirmasi Reservasi", sistem memvalidasi pesanan.
  - g. Jika sukses, sistem menyimpan reservasi dan menampilkan nomor booking serta sistem melakukan proses update status meja menjadi "dipesan" agar tidak dapat dipesan oleh orang lain. Setelah itu, sistem akan memunculkan interface untuk melakukan pembayaran yang sudah terintegrasi dengan payment gateway.
  - h. Pada tahap ini juga sistem akan menginsertkan data pada tabel transaksi dan detail transaksi. Pada tabel transaksi status transaksinya adalah "**pending**", status pembayaran DP juga "**pending**". Serta impact lain juga terhadap menu-menu yang dipesan akan berkurang stoknya sesuai dengan jumlah yang dipesan.
8. Interface Proses Pembayaran DP dengan Payment Gateway
- a. Setelah konfirmasi reservasi, sistem menampilkan total pembayaran DP.
  - b. Pengguna memilih metode pembayaran (QRIS, e-Wallet, VA, transfer, dll).
  - c. Sistem menghubungkan ke payment gateway untuk proses transaksi (hanya diberikan waktu 10 menit).
  - d. Pelanggan melakukan pembayaran

- e. Jika sukses, sistem memperbarui status transaksi dari "Pending" menjadi "reserved". Selain itu status pembayaran DP dari "Pending" menjadi "Settlement". Hal tersebut dapat dilakukan otomatis dengan membuat sebuah Callback URL yang dikoneksikan dengan payment gateway. Selanjutnya pengguna akan mendapatkan kode booking di email (bukti reservasi) yang nanti akan ditunjukkan kepada pegawai restaurant. Selain itu, sistem akan menampilkan pesan bahwa pelanggan harus datang maksimal 30 menit dari proses transaksi yang dilakukan.
  - f. Akan jika sampai batas waktu yang ditentukan pelanggan tidak kunjung melakukan pembayaran, maka admin akan membatalkan pemesanan dengan impactnya adalah, meja diset menjadi "kosong" lagi, stok menu akan kembali, sert data transaksi juga dihapus.
  - g. Pelanggan yang melakukan 3x pembatalan proses pemesanan, maka admin dapat menghapus data pelanggan tersebut.
9. Interface List Transaksi (History & Ongoing)
- a. Pengguna melihat daftar transaksi yang pernah dilakukan.
  - b. Sistem mengambil data transaksi dari backend (REST API).
  - c. Status transaksi bisa "Pending" (belum melakukan pembayaran), "Reserved" (sudah melakukan pembayaran), atau "Selesai" (transaksi sudah selesai).
10. Interface Halaman Detail Transaksi
- a. Pengguna memilih transaksi untuk melihat detailnya.
  - b. Sistem menampilkan meja yang dipesan, menu yang dipesan, total pembayaran.
  - c. Jika transaksi belum selesai, pengguna bisa melihat opsi pembayaran akhir.

### **Data Access Security and Payment Gateway Integration**

1. Mengintegrasikan JWT Auth Token untuk Autentikasi Web dan Mobile
- Tujuan
- a. Mengamankan akses ke API dengan JSON Web Token (JWT).
  - b. Mencegah akses tidak sah ke halaman admin dan pelanggan.
- Alur Implementasi:
- a. Registrasi & Login



- 1) Saat pelanggan atau admin login, backend akan menghasilkan JWT Token.
- 2) Token ini dikirim sebagai respons dan disimpan di Local Storage (Web) atau Secure Storage (Mobile).
- b. Menyertakan JWT dalam Setiap Request
  - 1) Saat pengguna mengakses API yang membutuhkan autentikasi, token dikirim dalam Authorization Header.
  - 2) Backend akan memvalidasi token sebelum memproses permintaan.
- c. Memvalidasi dan Mengecek Token Expired
 

Jika token expired, sistem meminta pengguna untuk login kembali.
2. Mengintegrasikan Pembayaran dengan Third-Party Payment Gateway
 

Metode yang digunakan: Xendit / Midtrans/dll

Tujuan:

  - a. Memfasilitasi pembayaran DP saat reservasi.
  - b. Memastikan reservasi valid hanya untuk pelanggan yang sudah membayar.

Alur Implementasi:

  - a. Pembayaran DP (Down Payment) saat Reservasi
    - 1) Setelah pengguna melakukan reservasi meja dan memilih menu, sistem menghitung total pembayaran DP.
    - 2) Pelanggan memilih metode pembayaran (QRIS, VA Bank, e-Wallet, Gopay, dll).
    - 3) Sistem menghubungkan pelanggan ke halaman pembayaran dari Payment Gateway (Xendit/Midtrans).
    - 4) Setelah pembayaran sukses, status transaksi diperbarui menjadi "reserved" dengan Callback URL.

### **Business Intelligence & Reporting**

Membuat Halaman Dashboard untuk Admin Panel

Halaman ini akan menampilkan laporan bisnis penting untuk membantu admin dalam pengambilan keputusan berdasarkan data transaksi dan reservasi.

Fitur yang Ditampilkan di Dashboard:

1. Jumlah Transaksi Berdasarkan Rentang Tanggal
  - a. Menampilkan grafik jumlah transaksi yang dilakukan dalam periode tertentu.

- b. Admin bisa memilih rentang tanggal (misal: hari ini, minggu ini, bulan ini).
  - c. Grafik line chart atau bar chart untuk memvisualisasikan jumlah transaksi.
- 2. Report 5 Menu Makanan Terlaris
  - a. Menampilkan daftar 5 makanan/minuman paling banyak dipesan dalam periode tertentu.
  - b. Bisa disajikan dalam bentuk ranking list atau pie chart.
  - c. Admin bisa melihat pola konsumsi pelanggan untuk strategi menu dan promosi.
- 3. Report Nominal Transaksi per Rentang Tanggal
  - a. Menampilkan total pendapatan restoran dalam rentang tanggal tertentu.
  - b. Bisa dalam bentuk bar chart atau summary card.
  - c. Admin dapat melihat bagaimana performa keuangan restoran berdasarkan periode waktu tertentu.