# STUDI KASUS SISTEM PEMESANAN MAKANAN ONLINE

**Deskripsi Kasus:**

Sebuah perusahaan rintisan bernama "FoodGo" ingin mengembangkan sistem pemesanan makanan online. Sistem ini memungkinkan pelanggan untuk memesan makanan dari berbagai restoran yang terdaftar melalui aplikasi web atau mobile. Pelanggan dapat melihat menu, memilih makanan, melakukan pemesanan, dan membayar secara online. Restoran akan menerima pesanan dan mengelola proses pengiriman atau pengambilan.

**Aktor yang terlibat:**

* Pelanggan
* Restoran
* Admin FoodGo

**Fitur utama sistem:**

* Pelanggan:
* Registrasi dan Login
* Melihat daftar restoran dan menu
* Mencari restoran berdasarkan lokasi, kategori atau nama
* Membuat pesanan
* Melihat detail pesanan dan status nya
* Memberikan ulasan dan rating restoran
* Melakukan pembayaran online
* Mengelola profil
* Restoran:
* Login
* Mengelola menu (Tambah, Edit, Hapus)
* Menerima dan mengkonfirmasi pesanan
* Memperbarui status pesanan (misalnya, “Sedang disiapkan”, “Siap diantar”)
* Melihat laporan penjualan
* Mengelola informasi restoran (jam buka, alamat, dll)
* Admin FoodGo:
* Mengelola data restoran (verifikasi, aktivasi, blokir)
* Mengelola kategori makanan
* Melihat laporan keseluruhan
* Mengelola promosi dan diskon
* Mengelola pengguna (admin)

**Kebutuhan:**

Buatlah diagram UML berikut menggunakan Draw.io:

1. Use Case Diagram:

Menampilkan aktor dan fungsionalitas sistem

1. Activity Diagram:

Menjelaskan alur proses pemesanan makanan (dari pelanggan memesan hingga makanan diterima).

1. Class Diagram:

Menampilkan kelas-kelas dalam sistem, atribut, method, dan relasi antar kelas.

# NASKAH VIDEO

1. Opening

Assalamualaikum warahmatullahi wabarakatuh. Selamat datang di tutorial UML Sistem Pemesanan Makanan Online "FoodGo".

Dalam video ini, kita akan membahas studi kasus, membuat Use Case Diagram, Activity Diagram, dan Class Diagram untuk sistem ini.

1. Memahami Studi Kasus

Kita akan mulai dengan memahami studi kasus Sistem Pemesanan Makanan Online "FoodGo". Sistem ini menghubungkan pelanggan dengan berbagai restoran, memungkinkan pemesanan dan pembayaran online.

Aktor yang terlibat adalah Pelanggan, Restoran, dan Admin FoodGo. Fitur-fitur utama mencakup registrasi, pemesanan, pengelolaan menu, dan pembayaran.

1. Membuat Use Case Diagram

Sebelum membuat diagram, mari kita pahami apa itu Use Case Diagram. (Penjelasan singkat tentang Use Case Diagram).

**Langkah-Langkah:**

1. Buka draw.io
2. Tambahkan aktor:

* Pelanggan
* Restoran
* Admin FoodGo

1. Tambahkan Use Case:

* Pelanggan: Registrasi, Login, Lihat restoran, cari restoran, lihat menu, buat pesanan, bayar pesanan, beri ulasan, kelola profil.
* Restoran: Login, kelola menu, terima pesanan, konfiramsi pesanan, Update status pesanan, lihat laporan penjualan, kelola info restoran.
* Admin FoodGo: Kelola restoran, kelola kategori, lihat laporan, kelola promosi, kelola admin.

1. Hubungkan aktor dengan use case yang relevan, misalnya: Pelanggan terhubung ke “Buat pesanan” dan “Bayar Pesanan”.
2. Simpan Use Case Diagram.
3. Membuat Activity Diagram

Sebelum kita membuat activity diagram nya, mari kita pahami terlebih dahulu apa itu Activity Diagram? (Menurut pendapat anda sendiri).

**Langkah-langkah:**

1. Buat Activity Diagram untuk alur pemesanan makanan:

* Mulai (**Start**)
* Pelanggan memilih restoran
* Pelanggan memilih makanan dari menu
* Pelanggan membuat pesanan
* Sistem memvalidasi pesanan
* Sistem menghitung total biaya
* Pelanggan melakukan pembayaran
* Sistem mengirim notifikasi ke restoran
* Restoran menerima pesanan
* Restoran menyiapkan makanan
* Restoran mengupdate status pesanan
* Pelanggan menerima makanan
* Selesai (**END**)

1. Gunakan Swimlane untuk memisahkan aktivitas antara pelanggan, sistem, dan restoran.
2. Tambahkan Decision Node (Misalnya, untuk validasi pembayaran atau ketersediaan makanan)
3. Simpan Activity Diagram.
4. Membuat Class Diagram

Sebelum kita membuat class diagram nya, mari kita pahami terlebih dahulu apa itu Class Diagram? (Menurut pendapat anda sendiri).

**Langkah-langkah:**

1. Buat Class Diagram:
2. Class Pelanggan:

* Atribut: id\_pelanggan, nama, email, password, alamat, no\_telepon.
* Method: registrasi(), login(), buatPesaanan(), bayarPesanan(), beriUlasan()

1. Kelas Restoran:

* Atribut: id\_restora, nama\_restoran, alamat, no\_telepon, jam\_buka, kategori
* Method: kelolaMenu(), terimaPesanan(), updateStatusPesanan()

1. Kelas Menu:

* Atribut: id\_menu, id\_restoran, nama\_makanan, deskripsi, harga
* Method: tambahMakanan(), editMakanan(), hapusMakanan()

1. Kelas Pesanan:

* Atribut: id\_pesanan, id\_pelanggan, id\_restoran, tanggal\_pesan, total\_harga, status\_pesanan, alamat\_pengiriman
* Method: buatPesanan(), batalPesanan()

1. Kelas ItemPesanan:

* Atribut: id\_item, id\_pesanan, id\_menu, jumlah, subtotal

1. Kelas Pembayaran:

* Atribut: id\_pembayaran, id\_pesanan, metode\_pembayaran, tanggal\_bayar, jumlah\_bayar, status\_pembayaran

1. Kelas Admin:

* Atribut: id\_admin, nama, email, password
* Method: kelolaRestoran(), kelolaKategori()

1. Buat relasi antar kelas (asosiasi, agregasi, komposisi). Contoh:

* Pelanggan membuat Pesanan
* Restoran memiliki Menu
* Pesanan terdiri dari ItemPesanan
* Pesanan terkait dengan Pembayaran

1. Simpan Class Diagram
2. Closing

Demikianlah penjelasan tentang pembuatan UML untuk Sistem Pemesanan Makanan Online "FoodGo"

Semoga video ini bermanfaat dan membantu Anda dalam memahami dan membuat diagram UML.

Terima kasih telah menonton. Wassalamu’alaikum warahmatullahi wabarakatuh.