# Tugas Project Pemrograman Berorientasi Objek



**DISUSUN OLEH :**

**Nama : M Raihan Frakasya  
Nim : 2200018216  
Kelas : E  
UserName : Tolekstar (Raihan Frakasya)  
Link GitHub :** [Tolekstar’s gists (github.com)](https://gist.github.com/Tolekstar)

# Link GitHub : [Tolekstar’s gists (github.com)](https://gist.github.com/Tolekstar) Deskirpsi Aplikasi Karena semaking berkembangnya teknologi sangat banyak hal yang bisa di lakukan melalui teknologi, oleh karena itu aplikasi ini memberi kemudahan untuk customer dalam memesan makanan seperti efisiensi waktu dan penghematan energi, aplikasi ini juga bisa di gunakan untuk memesan makanan dari rumah ataupun saat ingin makan di tempatnya langsung Fitur utama : user akan menginputkan data diri dan memilih menu makanan sesuai keinginnan serta memesan makanan melalui aplikasi Alur Kerja CUSTOMER memesan makanan melalui aplikasi APLIKASI pesanan masuk ke kasir KASIR memberitahu pesanan ke koki KOKI koki akan menyelesaikan masakan PELAYAN dan jika selesai pelayan akan mengambil makanan dan mengantarkan ke customer, jika memesan melalui rumah pelayan akan memberikan makanan ke kurir untuk di antar ke customer yang sedang berada di rumah KURIR kurir akan mengambil makanan yang disediakan oleh pelayan Diagram Class Customer : Nama : String/Varchar alamat : String/Varchar Aplikasi : Menu : String/Varchar Harga :INT (integer) Total :INT (integer) Nama : String/Varchar alamat : String/Varchar Restoran : Menu : String/Varchar

# Harga :INT (integer) Total :INT (integer) Nama : String/Varchar alamat : String/Varchar Pelayan : Menu : String/Varchar Nama : String/Varchar alamat : String/Varchar Rancangan UI

# Customer akan langsung melihat beberapa menu yang di sediakan, jika customer ingin menambahkan pesanan customer tinggal menklik tombol “+” pada menu dan jika ingin memesan lebih dari 2 customer tinggal menklik “+” pada menu sebanyak yang ingin di pesan customer dan jika ingin mengurangi pesanan customer tinggal menklik “-“ pada menu, dan jika sudah maka pada total akan muncul jumlah yang harus di bayar, dan jika ingin menyudahi memesan customer tinggal klik pesan Implementasi Kodingan Java Membuat Class Itemmakanan serta variabel di dalam nya Membuat Class keranjangbelanja serta tambah item dan tampilankeranjang agar bisa menambah item atau menampilkan keranjang saat program di jalankan Membuat class Restoran dan list makanan agar saat program di jalankan dapat menampilkan menu makanan membuat class SistemPemesananMakananOnline dan menampilkan output” yang di perlukan serta agar user dapat menginputkan pilihan Screenshot Tampilan Luaran Program Uji Coba Program Error/Sukses Source Kode: import java.util.ArrayList;



# import java.util.HashMap;

# import java.util.List;

# import java.util.Map;

# import java.util.Scanner;

# class ItemMakanan {

# private String nama;

# private double harga;

# public ItemMakanan(String nama, double harga) {

# this.nama = nama;

# this.harga = harga;

# }

# public String getNama() {

# return nama;

# }

# public double getHarga() {

# return harga;

# }

# }

# class KeranjangBelanja {

# private Map<ItemMakanan, Integer> items;

# public KeranjangBelanja() {

# this.items = new HashMap<>();

# }

# public void tambahItem(ItemMakanan item, int jumlah) {

# items.put(item, items.getOrDefault(item, 0) + jumlah);

# }

# public void tampilkanKeranjang() {

# System.out.println("Keranjang Belanja:");

# double total = 0;

# for (Map.Entry<ItemMakanan, Integer> entry : items.entrySet()) {

# ItemMakanan item = entry.getKey();

# int jumlah = entry.getValue();

# double subtotal = item.getHarga() \* jumlah;

# total += subtotal;

# System.out.println(item.getNama() + " - Jumlah: " + jumlah + " - Subtotal: Rp" + subtotal);

# }

# System.out.println("Total: Rp" + total);

# }

# }

# class Restoran {

# private List<ItemMakanan> menu;

# public Restoran() {

# this.menu = new ArrayList<>();

# }

# public void tambahKeMenu(ItemMakanan item) {

# menu.add(item);

# }

# public void tampilkanMenu() {

# System.out.println("Menu:");

# for (int i = 0; i < menu.size(); i++) {

# ItemMakanan item = menu.get(i);

# System.out.println((i + 1) + ". " + item.getNama() + " - Rp" + item.getHarga());

# }

# }

# public void stokmakanan(){

# }

# public List<ItemMakanan> getMenu() {

# return menu;

# }

# }

# public class SistemPemesananMakananOnline {

# public static void main(String[] args) {

# Scanner scanner = new Scanner(System.in);

# System.out.print("Masukkan Nama Anda: ");

# String namaPelanggan = scanner.nextLine();

# System.out.print("Masukkan Alamat Anda / no meja : ");

# String alamatPelanggan = scanner.nextLine();

# Restoran restoran = new Restoran();

# restoran.tambahKeMenu(new ItemMakanan("Paket Nasi & Ayam Goreng + Esteh", 20.000));

# restoran.tambahKeMenu(new ItemMakanan("Paket Nasi & Ayam Geprek + Esteh", 25.000));

# restoran.tambahKeMenu(new ItemMakanan("Paket Lalapan & cemilan", 10.000));

# KeranjangBelanja keranjang = new KeranjangBelanja();

# int pilihan;

# do {

# System.out.println("\n1. Tampilkan Menu");

# System.out.println("2. Tambah Item ke Keranjang");

# System.out.println("3. Lihat Keranjang");

# System.out.println("4. Pesan");

# System.out.print("Masukkan pilihan Anda: ");

# pilihan = scanner.nextInt();

# switch (pilihan) {

# case 1:

# restoran.tampilkanMenu();

# break;

# case 2:

# restoran.tampilkanMenu();

# System.out.print("Masukkan nomor item untuk ditambahkan ke keranjang: ");

# int nomorItem = scanner.nextInt();

# if (nomorItem >= 1 && nomorItem <= restoran.getMenu().size()) {

# ItemMakanan itemDipilih = restoran.getMenu().get(nomorItem - 1);

# System.out.print("Masukkan jumlah: ");

# int jumlah = scanner.nextInt();

# keranjang.tambahItem(itemDipilih, jumlah);

# System.out.println("Item ditambahkan ke keranjang!");

# } else {

# System.out.println("Nomor item tidak valid.");

# }

# break;

# case 3:

# keranjang.tampilkanKeranjang();

# break;

# case 4:

# System.out.println("Terima kasih, " + namaPelanggan + "! Pesanan Anda akan dikirim ke alamat/no meja: " + alamatPelanggan);

# break;

# default:

# System.out.println("Pilihan tidak valid. Silakan coba lagi.");

# break;

# }

# } while (pilihan != 4);

# }

# } Bukti Unggah Final Projek di Github

# Bukti Pengumpulan Tugas Bukti Unggah di Youtube Analisi Pengerjaan Tinjauan dari sisi waktu Dalam pembuatan projek ini (desain rancangan ui/ux, desain sampul, pembuatan alur, diagram, program dan lain lainnya) di butuhkan waktu beberapa hari dalam bahkan hampir seminggu Ketercapaian spesifikasi ketercapaian spesifikasi untuk program yang sederhana ini masih ada beberapa fitur yang kurang seperti rekomendasi makanan dan terutama untuk menu yang terlalu sedikit kendala di karenakan saya masih belajar, jadi saya kesulitan dalam menginstal intellij Idea dan dalam menguplod file ke github serta saya mengalami beberapa kali error dalam pembuatan program sehingga saya membutuhkan bantuan google Kesimpulan Aplikasi pesan makan online telah menjadi inovasi yang sangat berpengaruh dalam industri kuliner modern. Dengan menyediakan layanan yang mudah digunakan, cepat, dan efisien, aplikasi ini telah mengubah cara orang memesan dan menikmati makanan