



Coffee Shop

Kafe Kita

PRESENTED BY: KELOMPOK2

- ZAQI
- ADINDA
- DINA MEI LESTARI (09)

Topic Outline

- 1 Tujuan Pembuatan:
- 2 Masalah yang dihadapi
- 3 Solusi Aplikasi
- 4 Proses Pengembangan
- 5 Kesimpulan





Tujuan Pembuatan:

Aplikasi Kafe Pilihan Sederhana ini dibuat dengan tujuan untuk mempermudah pelanggan dalam memilih dan memesan makanan serta minuman di kafe secara praktis. Dengan adanya aplikasi ini, pelanggan dapat melihat daftar menu, memilih makanan dan minuman yang diinginkan, serta menghitung total harga secara otomatis. Aplikasi juga memberikan kemudahan bagi pemilik kafe dalam mengelola pesanan dan memberikan diskon untuk anggota.

Masalah yang dihadapi

- **Proses Pemesanan Manual:** Sebelumnya, pelanggan diharuskan untuk memesan secara manual melalui pelayan. Proses ini memakan waktu dan rentan terhadap kesalahan input.
- **Keterbatasan Pengelolaan Menu dan Harga:** Kafe tidak memiliki sistem yang efektif untuk menampilkan menu dan harga secara otomatis, yang membuat pelanggan harus bertanya atau memeriksa manual.
- **Pengelolaan Diskon Anggota:** Proses pemberian diskon bagi anggota masih dilakukan secara manual, yang bisa menyebabkan kesalahan perhitungan.



Solusi oleh Aplikasi:

- 1. Menu Digital yang Interaktif:** Aplikasi ini menampilkan menu makanan dan minuman secara interaktif, yang memungkinkan pelanggan untuk memilih sesuai preferensi mereka tanpa perlu bertanya kepada pelayan.
- 2. Pencatatan Pemesanan yang Akurat:** Dengan aplikasi ini, setiap pemesanan tercatat dengan jelas dan otomatis, menghindari kesalahan dalam proses pemesanan.
- 3. Perhitungan Total Harga dan Diskon:** Aplikasi ini otomatis menghitung total harga berdasarkan pilihan menu dan memberi diskon bagi anggota, sehingga mempermudah pelanggan dalam melakukan pembayaran.

berikut flowchart dan codingan kami ->



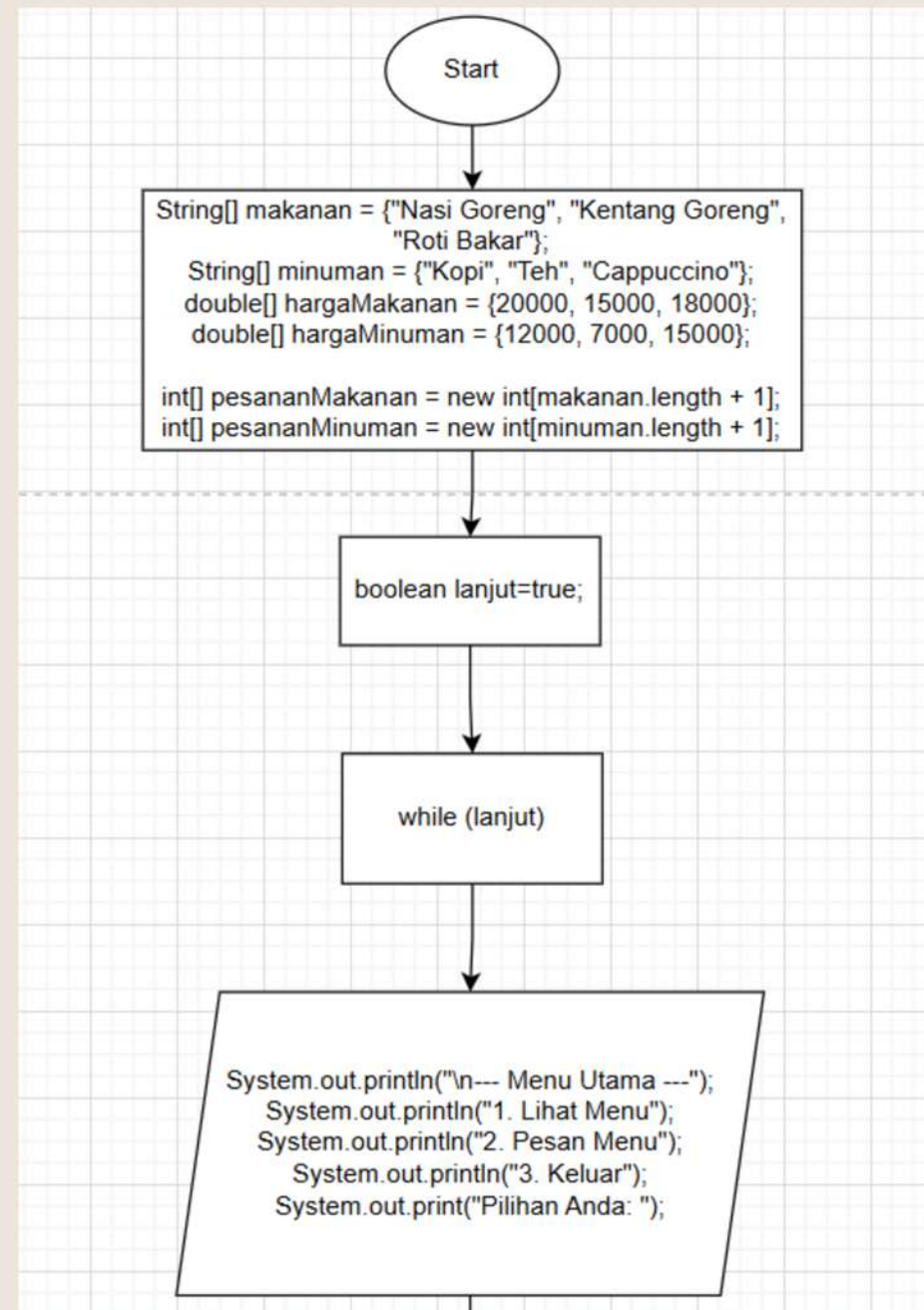

```
import java.util.Scanner;

public class KafePilihanSederhana {
    public static void main(String[] args) {
        String[] makanan = {"Nasi Goreng", "Kentang Goreng", "Roti Bakar"};
        String[] minuman = {"Kopi", "Teh", "Cappuccino"};
        double[] hargaMakanan = {20000, 15000, 18000};
        double[] hargaMinuman = {12000, 7000, 15000};

        int[] pesananMakanan = new int[makanan.length + 1];
        int[] pesananMinuman = new int[minuman.length + 1];

        Scanner sc = new Scanner(System.in);

        boolean lanjut = true;
        while (lanjut) {
            System.out.println("\n--- Menu Utama ---");
            System.out.println("1. Lihat Menu");
            System.out.println("2. Pesan Menu");
            System.out.println("3. Keluar");
            System.out.print("Pilihan Anda: ");
            int pilihan = sc.nextInt();
        }
    }
}
```



--- Menu Utama ---
1. Lihat Menu
2. Pesan Menu
3. Keluar
Pilihan Anda:

Kode ini adalah bagian awal dari program kafe kita.

- Tujuan: Mempermudah pengguna untuk memilih tindakan (lihat menu, pesan menu, keluar).
- Proses dimana Menu Utama ditampilkan.



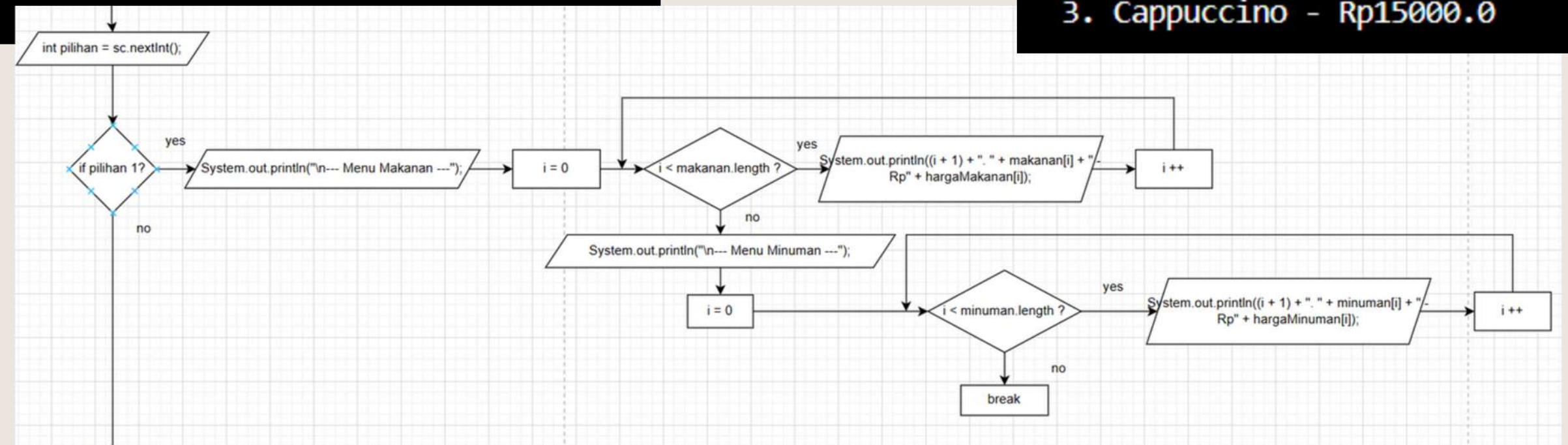

```
switch (pilihan) {
    case 1:
        System.out.println(x:"\n--- Menu Makanan ---");
        for (int i = 0; i < makanan.length; i++) {
            System.out.println((i + 1) + ". " + makanan[i] + " - Rp" + hargaMakanan[i]);
        }
        System.out.println(x:"\n--- Menu Minuman ---");
        for (int i = 0; i < minuman.length; i++) {
            System.out.println((i + 1) + ". " + minuman[i] + " - Rp" + hargaMinuman[i]);
        }
        break;
}
```

--- Menu Makanan ---

1. Nasi Goreng - Rp20000.0
2. Kentang Goreng - Rp15000.0
3. Roti Bakar - Rp18000.0

--- Menu Minuman ---

1. Kopi - Rp12000.0
2. Teh - Rp7000.0
3. Cappuccino - Rp15000.0



1. Input Pilihan:

- Pengguna memasukkan angka (1, 2, atau 3) untuk menentukan tindakan yang diinginkan.
- Input disimpan dalam variabel pilihan.

2. Pilihan case 1 (Lihat Menu):

- Menampilkan daftar makanan dan minuman dengan harga masing-masing.
- Menggunakan loop untuk mencetak nama dan harga makanan/minuman secara berurutan.

3. break:

- Menghentikan eksekusi dalam switch setelah menyelesaikan tindakan pilihan.

Tujuan: Menyediakan informasi menu makanan dan minuman beserta harganya secara rapi dan terstruktur.




```

case 2:
boolean pesanLagi = true;
while (pesanLagi) {
    System.out.println(x: "\n--- Menu Makanan ---");
    for (int i = 0; i < makanan.length; i++) {
        System.out.println((i + 1) + ". " + makanan[i] + " - Rp" + hargaMakanan[i]);
    }
    System.out.println(x: "0. Tidak Pesan Makanan");

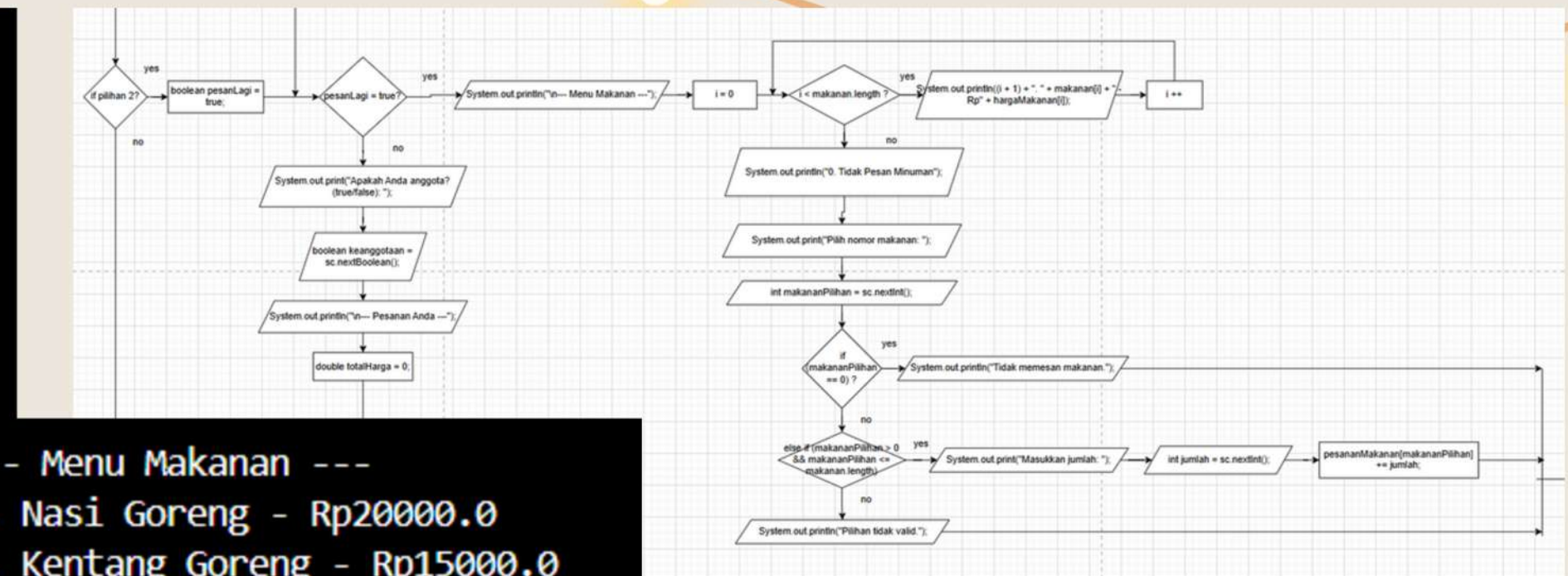
    System.out.print(s: "Pilih nomor makanan: ");
    int makananPilihan = sc.nextInt();
    if (makananPilihan == 0) {
        System.out.println(x: "Tidak memesan makanan.");
    } else if (makananPilihan > 0 && makananPilihan <= makanan.length) {
        System.out.print(s: "Masukkan jumlah: ");
        int jumlah = sc.nextInt();
        pesananMakanan[makananPilihan] += jumlah;
    } else {
        System.out.println(x: "Pilihan tidak valid.");
    }
}

```

```

--- Menu Makanan ---
1. Nasi Goreng - Rp20000.0
2. Kentang Goreng - Rp15000.0
3. Roti Bakar - Rp18000.0
0. Tidak Pesan Makanan
Pilih nomor makanan: 1
Masukkan jumlah: 1

```



1. Pilihan Pesan Menu:

- Ketika pengguna memilih 2, program masuk ke proses pemesanan makanan dan minuman.

2. Loop Pemesanan:

- Variabel pesanLagi digunakan untuk mengontrol apakah pengguna ingin terus memesan atau berhenti.
- Selama pesanLagi bernilai true, program akan terus menampilkan daftar menu makanan dan memproses pemesanan.

3. Logika Pemesanan Makanan:

- Program mencetak daftar makanan beserta harganya, dan opsi 0 untuk tidak memesan makanan.
- Pengguna memasukkan nomor makanan yang ingin dipesan.
 - Jika 0, berarti tidak memesan makanan.
 - Jika nomor valid (dalam rentang daftar makanan), pengguna memasukkan jumlah pesanan, yang kemudian ditambahkan ke array pesananMakanan.
 - Jika nomor tidak valid, program menampilkan pesan kesalahan.

Tujuan: Memfasilitasi pengguna dalam memilih makanan yang ingin dipesan, dengan kemampuan untuk memesan lebih dari satu item atau tidak memesan sama sekali.




```

System.out.println(x:"\n--- Menu Minuman ---");
for (int i = 0; i < minuman.length; i++) {
    System.out.println((i + 1) + ". " + minuman[i] + " - Rp" + hargaMinuman[i]);
}
System.out.println(x:"0. Tidak Pesan Minuman");

System.out.print(s:"Pilih nomor minuman: ");
int minumanPilihan = sc.nextInt();
if (minumanPilihan == 0) {
    System.out.println(x:"Tidak memesan minuman.");
} else if (minumanPilihan > 0 && minumanPilihan <= minuman.length) {
    System.out.print(s:"Masukkan jumlah: ");
    int jumlah = sc.nextInt();
    pesananMinuman[minumanPilihan] += jumlah;
} else {
    System.out.println(x:"Pilihan tidak valid.");
}

System.out.print(s:"Apakah ingin memesan lagi? (true/false): ");
pesanLagi = sc.nextBoolean();

```

--- Menu Minuman ---

1. Kopi - Rp12000.0

2. Teh - Rp7000.0

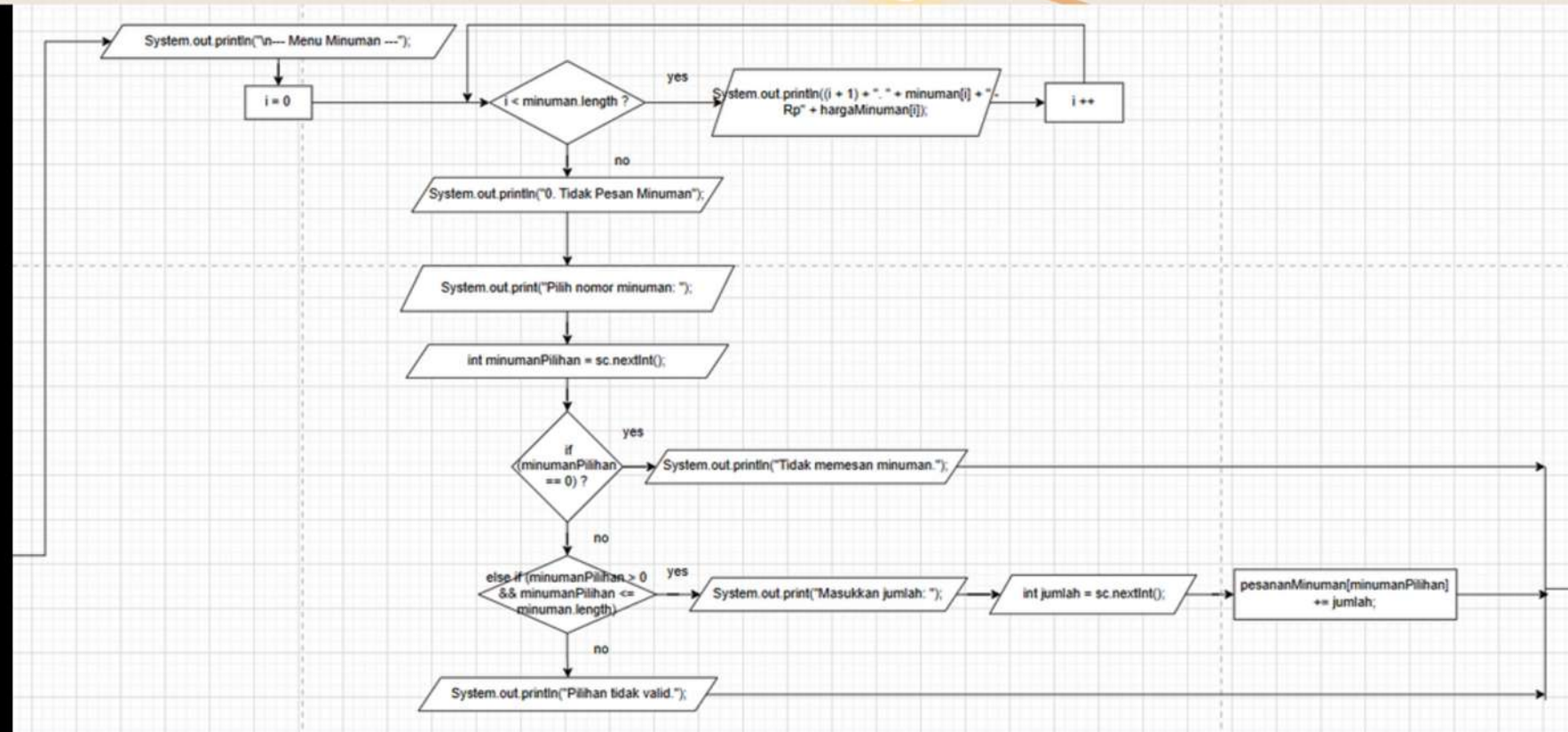
3. Cappuccino - Rp15000.0

0. Tidak Pesan Minuman

Pilih nomor minuman: 1

Masukkan jumlah: 1

Apakah ingin memesan lagi? (true/false): false



1. Menampilkan Menu Minuman:

- Program mencetak daftar minuman beserta harganya.
- Opsi 0 disediakan untuk pengguna yang tidak ingin memesan minuman.

2. Memproses Pilihan Minuman:

- Pengguna memasukkan nomor pilihan:
 - Jika 0, artinya tidak ada minuman yang dipesan, dan program memberikan konfirmasi.
 - Jika nomor valid (dalam rentang daftar minuman), pengguna memasukkan jumlah pesanan. Jumlah ini kemudian ditambahkan ke array pesananMinuman sesuai dengan nomor minuman yang dipilih.
 - Jika nomor tidak valid, program menampilkan pesan kesalahan.

3. Konfirmasi Pemesanan Ulang:

- Setelah proses pemesanan minuman selesai, program bertanya kepada pengguna apakah mereka ingin memesan lagi.
- Jawaban berupa true (untuk lanjut memesan) atau false (untuk berhenti) memengaruhi nilai variabel pesanLagi.

Tujuan: Memudahkan pengguna dalam memesan minuman, dengan fleksibilitas untuk memesan beberapa kali atau tidak memesan sama sekali.




```

System.out.print(s:"Apakah Anda anggota? (true/false): ");
boolean keanggotaan = sc.nextBoolean();

System.out.println(x:"\n--- Pesanan Anda ---");
double totalHarga = 0;

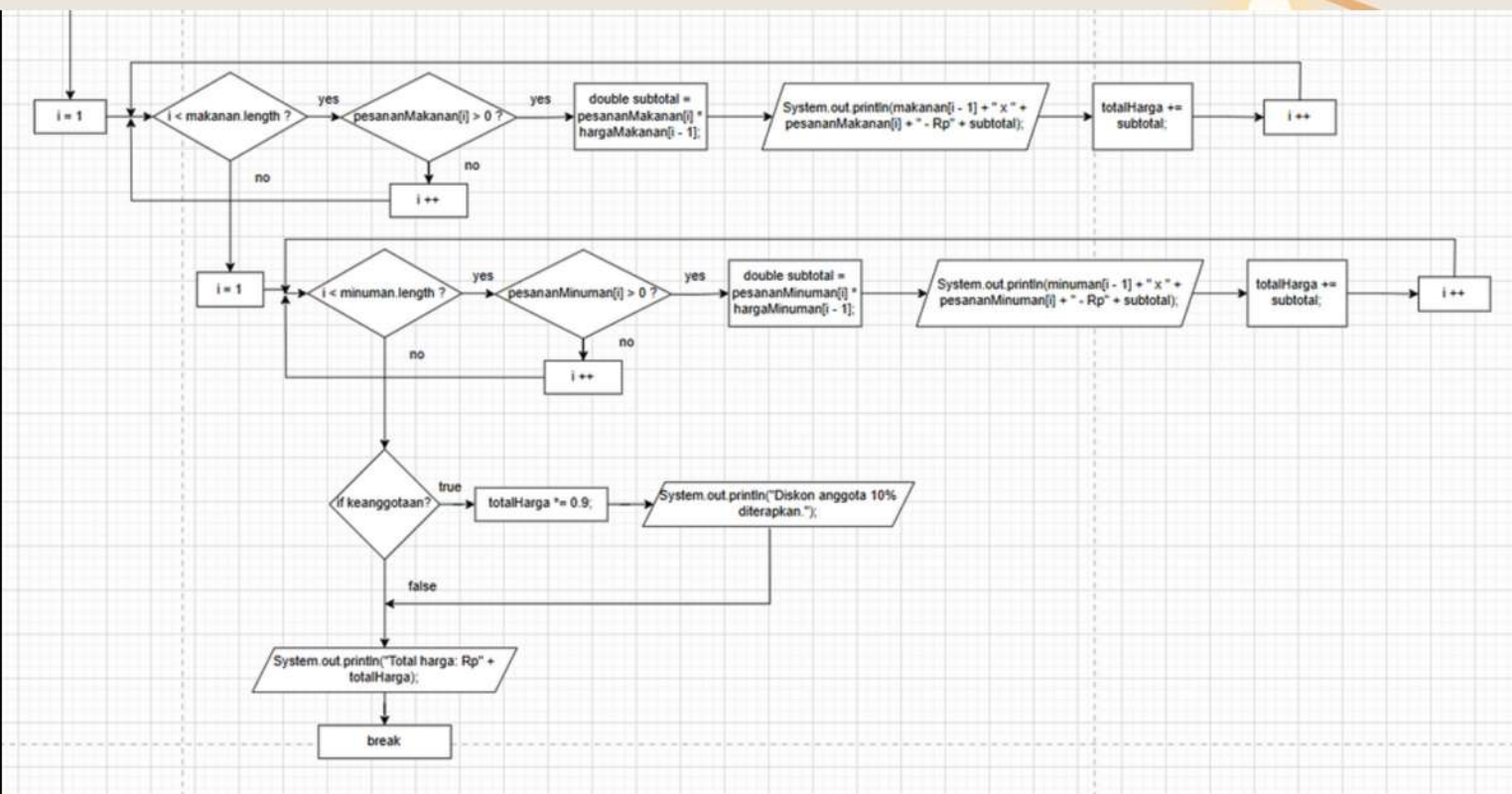
for (int i = 1; i <= makanan.length; i++) {
    if (pesananMakanan[i] > 0) {
        double subtotal = pesananMakanan[i] * hargaMakanan[i - 1];
        System.out.println(makanan[i - 1] + " x " + pesananMakanan[i] + " - Rp" + subtotal);
        totalHarga += subtotal;
    }
}

for (int i = 1; i <= minuman.length; i++) {
    if (pesananMinuman[i] > 0) {
        double subtotal = pesananMinuman[i] * hargaMinuman[i - 1];
        System.out.println(minuman[i - 1] + " x " + pesananMinuman[i] + " - Rp" + subtotal);
        totalHarga += subtotal;
    }
}

if (keanggotaan) {
    totalHarga *= 0.9;
    System.out.println(x:"Diskon anggota 10% diterapkan.");
}

System.out.println("Total harga: Rp" + totalHarga);
break;

```



Apakah Anda anggota? (true/false): true

--- Pesanan Anda ---

Nasi Goreng x 1 - Rp20000.0

Kopi x 1 - Rp12000.0

Diskon anggota 10% diterapkan.

Total harga: Rp28800.0

1. Konfirmasi Keanggotaan:

- Program menanyakan apakah pengguna adalah anggota.
- Jawaban berupa true atau false disimpan dalam variabel keanggotaan.

2. Menampilkan Pesanan dan Menghitung Total Harga:

- Makanan:
 - Untuk setiap jenis makanan yang dipesan ($\text{pesananMakanan}[i] > 0$), program menghitung subtotal (jumlah makanan dikalikan harga makanan).
 - Subtotal ditambahkan ke totalHarga dan ditampilkan di layar.
- Minuman:
 - Proses yang sama dilakukan untuk minuman.

3. Pemberian Diskon Keanggotaan:

- Jika pengguna adalah anggota ($\text{keanggotaan} == \text{true}$), total harga dikalikan 0.9, memberikan diskon sebesar 10%.
- Pesan konfirmasi diskon ditampilkan.

4. Menampilkan Total Harga Akhir:

- Total harga keseluruhan (setelah diskon jika ada) ditampilkan ke layar.

Tujuan:

- Memproses daftar pesanan dengan menghitung total harga secara rinci.
- Memberikan diskon khusus bagi anggota untuk meningkatkan loyalitas pelanggan.




```

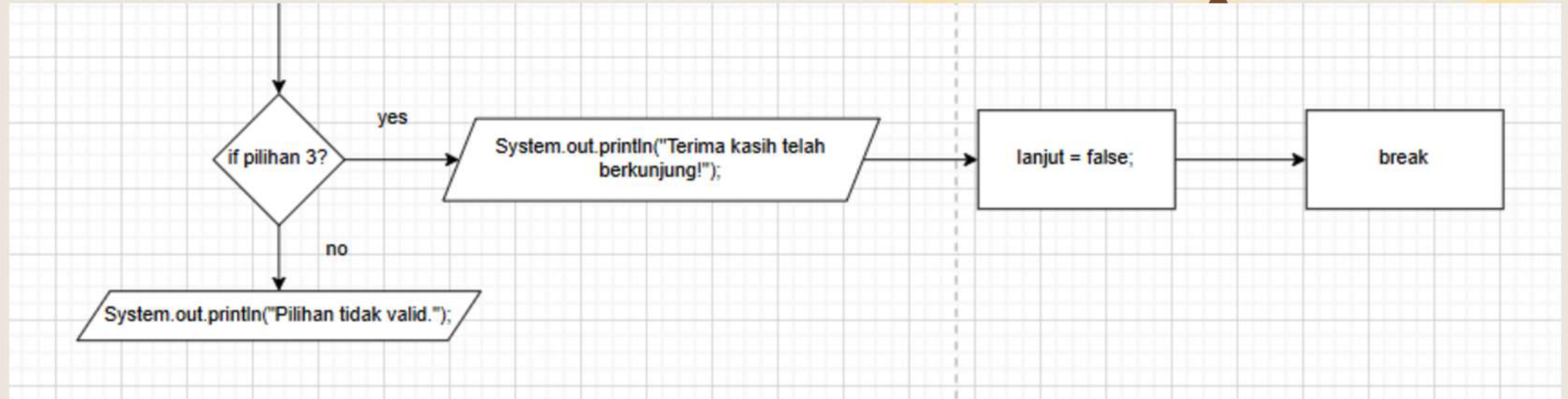
    case 3:
        System.out.println(x:"Terima kasih telah berkunjung!");
        lanjut = false;
        break;

    default:
        System.out.println(x:"Pilihan tidak valid.");

}

sc.close();
}
}

```



```

--- Menu Utama ---
1. Lihat Menu
2. Pesan Menu
3. Keluar
Pilihan Anda: 3
Terima kasih telah berkunjung!

```

Case 3:

1. Pesan Penutup:

- Menampilkan pesan "Terima kasih telah berkunjung!" kepada pengguna untuk memberi kesan akhir yang ramah.

2. Menghentikan Program:

- Variabel lanjut diubah menjadi false, yang akan keluar dari loop while, sehingga program berhenti.

Default:

1. Menangani Input Tidak Valid:

- Jika pengguna memasukkan pilihan yang tidak sesuai dengan opsi yang tersedia (1, 2, atau 3), program akan menampilkan pesan "Pilihan tidak valid."
- Hal ini memastikan bahwa program tidak crash dan tetap memberikan umpan balik kepada pengguna.

Tujuan:

- Case 3: Mengakhiri program dengan baik setelah pengguna memilih untuk keluar.
- Default: Menjaga program tetap berjalan meskipun pengguna salah memasukkan input, serta memberikan pengalaman pengguna yang lebih baik dengan pesan yang jelas.



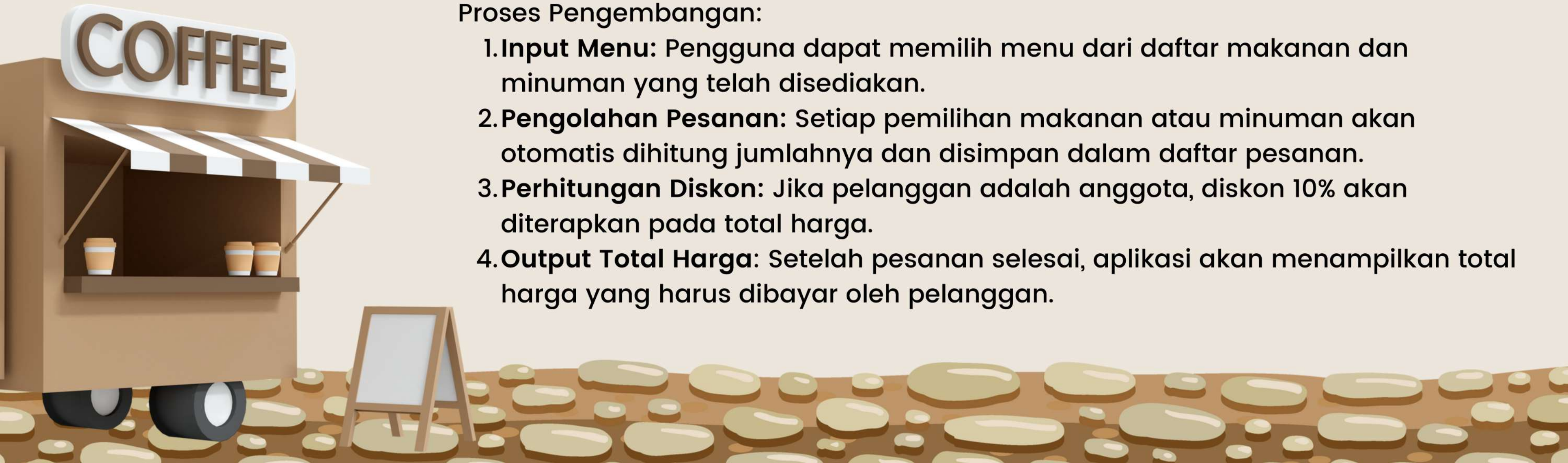
pendekatan : aplikasi berbasis program komputer

Mengapa Menggunakan Pendekatan Ini?

Pendekatan ini dipilih karena dapat meningkatkan efisiensi operasional kafe, memberikan pengalaman yang lebih nyaman bagi pelanggan, serta meminimalkan kesalahan manusia dalam proses pemesanan dan penghitungan harga. Selain itu, aplikasi ini juga dapat diimplementasikan dengan mudah tanpa memerlukan perangkat keras tambahan, cukup dengan menggunakan perangkat yang ada seperti laptop atau smartphone.

Proses Pengembangan:

1. **Input Menu:** Pengguna dapat memilih menu dari daftar makanan dan minuman yang telah disediakan.
2. **Pengolahan Pesanan:** Setiap pemilihan makanan atau minuman akan otomatis dihitung jumlahnya dan disimpan dalam daftar pesanan.
3. **Perhitungan Diskon:** Jika pelanggan adalah anggota, diskon 10% akan diterapkan pada total harga.
4. **Output Total Harga:** Setelah pesanan selesai, aplikasi akan menampilkan total harga yang harus dibayar oleh pelanggan.





Aplikasi Kafe Kita ini merupakan solusi yang efisien untuk meningkatkan kenyamanan pelanggan dalam melakukan pemesanan, serta membantu kafe dalam mengelola pesanan dan diskon anggota dengan lebih efektif. Dengan sistem yang otomatis dan mudah digunakan, aplikasi ini dapat mempercepat proses pelayanan dan mengurangi risiko kesalahan dalam penghitungan harga.





Thank you for listening!

DON'T HESITATE TO ASK ANY QUESTIONS!

