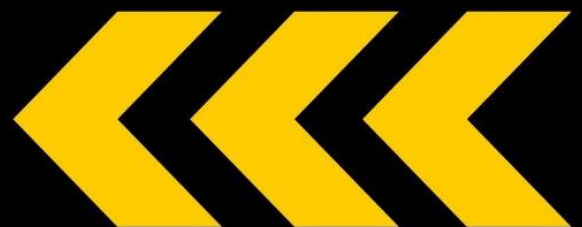




PRAKTIKUM PROGRAMA KOMPUTER

**KELOMPOK 10
2023**



BAB I

DESKRIPSI MASALAH

Bab ini menjelaskan mengenai latar belakang masalah, rumusan masalah, dan tujuan penelitian pada Program MyuList.

1.1 Latar Belakang

Subbab ini menjelaskan mengenai latar belakang dibuatnya Program MyuList.

Lagu menjadi salah satu kegiatan yang tidak pernah lepas menemani hari-hari khayalak umum. Lagu didengarkan untuk menghilangkan stres, relaksasi, meningkatkan motivasi, mengungkapkan emosi, dan masih banyak lagi. Pada zaman sekarang, banyak toko yang menjual CD lagu secara konvensional mulai tergusur oleh aplikasi maupun platform musik online. Maka dari itu melalui praktikum program komputer, kami mencoba untuk menghidupkan kembali penjualan CD dengan memberikan kemudahan yaitu pembelian secara online dengan membuat sebuah program *online shop* sederhana dengan produk yang dijual berupa CD lagu.

1.2 Rumusan Masalah

Subbab ini menjelaskan tentang rumusan masalah dari hasil praktikum program komputer berupa program MyuList. Berikut merupakan beberapa rumusan masalah dalam program *online shop*.

1. Bagaimana cara membangun sistem program yang aman dan terpercaya untuk melakukan transaksi jual beli CD secara online?
2. Bagaimana cara mempermudah pengguna agar dapat dengan cepat menemukan dan membeli CD yang mereka inginkan?
3. Bagaimana mengelola struk secara otomatis dan *paperless* agar pengguna dapat melihat hasil transaksi dan mengurangi penggunaan kertas?.
4. Bagaimana menyediakan fitur pemilihan dan ulasan produk berupa pemisahan genre lagu, penyanyi, dan tahun rilis sebuah lagu yang membantu pengguna dalam memilih CD?



1.3 Tujuan Penelitian

Subbab ini menjelaskan tentang tujuan dilakukannya pembuatan program MyuList. Berikut merupakan beberapa tujuan dalam program *online shop*.

1. Dapat menerapkan dengan baik ilmu program komputer dalam menyusun sebuah program sehingga menghasilkan sebuah program yang efisien, aman, dan terpercaya.
2. Memahami susunan program sehingga program yang dibuat dapat memberikan kemudahan bagi penggunanya untuk mendapatkan hal yang mereka inginkan.
3. Mengetahui cara menghasilkan output berupa struk yang membantu pengguna mengetahui jejak transaksi dan mengurangi penggunaan kertas struk.
4. Mengetahui cara untuk membuat fitur-fitur pada program yang efisien dan efektif dalam membantu kegiatan sehari-hari.

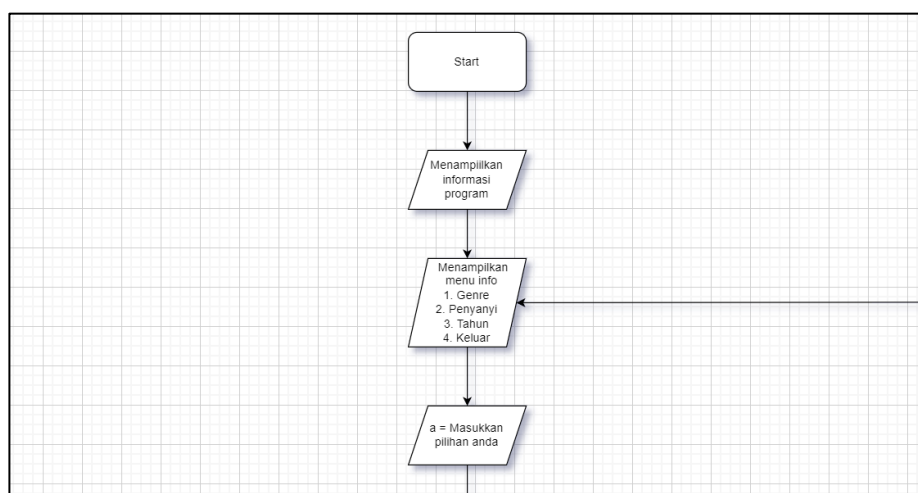
BAB II

FLOWCHART

Bab ini menjelaskan mengenai Flowchart pada program MyuList yaitu program jual beli lagu secara online.

2.1 Flowchart Program

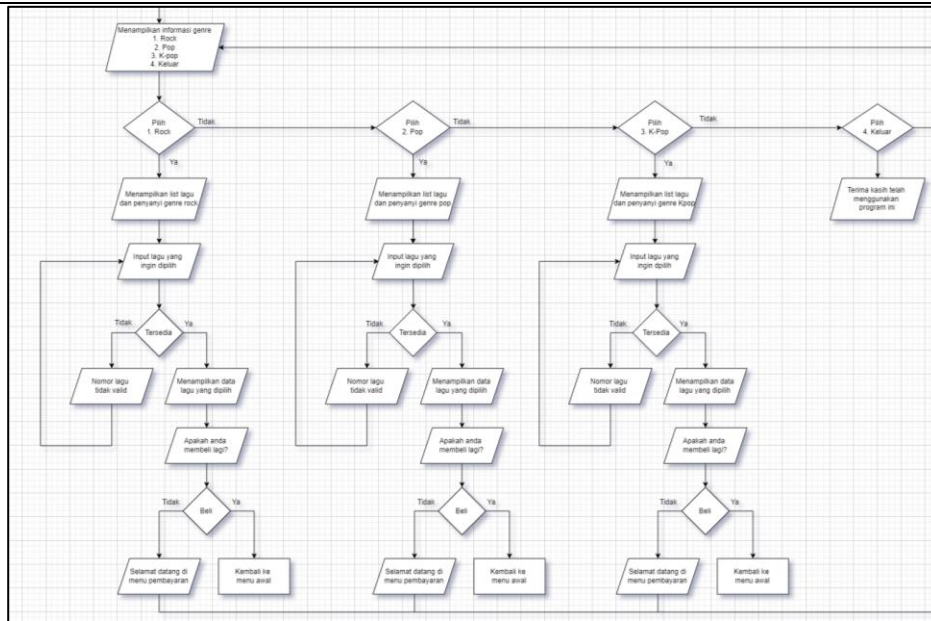
Subbab ini menjelaskan mengenai flowchart program awal dari program MyuList. Pada bagian ini, program berisikan menu awal yang dimana menampilkan informasi mengenai MyuList yang dilanjutkan dengan tampilan menu filter mulai dari genre, penyanyi dan tahunnya. Filter dapat dipilih dengan melakukan *input* angka sesuai dengan filter yang kita inginkan. Ketika pembeli menginput angka 1, maka pembeli akan diarahkan ke halaman filter genre lagu. Ketika pembeli menginput angka 2, maka pembeli akan diarahkan ke halaman filter berdasarkan penyanyi. Begitu juga ketika melakukan input dengan angka 3, maka pembeli akan diarahkan ke halaman filter berdasarkan tahun dari lagu yang diinginkan.



Gambar 2.1.1 Flowchart program awal

2.2 Flowchart Filter Genre

Subbab ini menjelaskan mengenai *flowchart* bagian filter berdasarkan genre untuk memunculkan menu genre dengan pilihan genre Rock, Pop dan K-Pop.



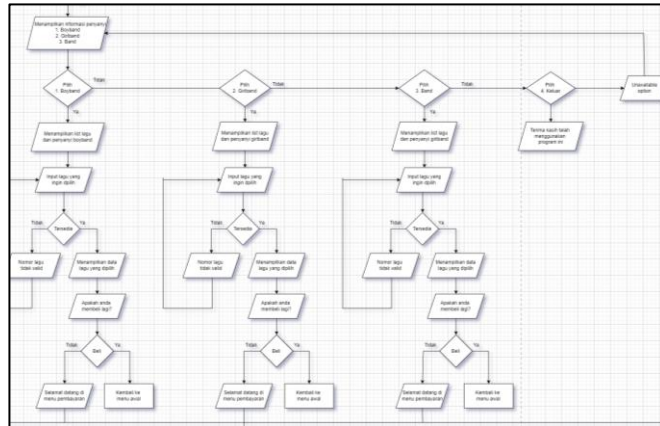
Gambar 2.2.1 Flowchart filter genre

Pada bagian ini, program dimaksudkan untuk dapat memudahkan pembeli mencari dan memilih lagu sesuai dengan genre lagu yang diinginkan dengan cepat. Dalam hal ini, pembeli akan diarahkan ke halaman yang berisi pilihan genre lagu. Kemudian pembeli akan kembali memilih genre dengan menginput angka sesuai dengan keterangan lagu. Ketika pembeli menginput angka 1, maka pembeli akan diarahkan pada halaman yang berisikan list lagu dengan genre lagu Rock. Kemudian, pembeli menginput nomor dengan judul lagu yang ingin dibeli. Ketika pembeli bersedia untuk membeli lagu yang dipilih, maka lagu tersebut akan langsung masuk ke dalam keranjang belanja. Dan apabila, pembeli tidak jadi membeli lagu tersebut akan ada pilihan kembali ke menu awal. Begitu juga dengan genre lagu lainnya yaitu genre Pop dan K-Pop.

2.3 Flowchart Filter Penyanyi

Subbab ini menjelaskan mengenai flowchart bagian filter berdasarkan penyanyi untuk memunculkan menu penyanyi dengan pilihan *girlband*, *boyband* dan *band*. Sama seperti flowchart sebelumnya, flowchart ini akan menampilkan halaman yang berisikan list lagu dengan pembagian berdasarkannya jenis penyanyinya. Ketika pembeli menginput angka 1, maka pembeli akan diarahkan pada halaman yang berisikan list lagu dengan penyanyi *boyband*. Kemudian, pembeli lanjut untuk menginput nomor dengan judul lagu yang ingin dibeli. Ketika pembeli bersedia untuk membeli lagu yang dipilih, maka lagu tersebut akan

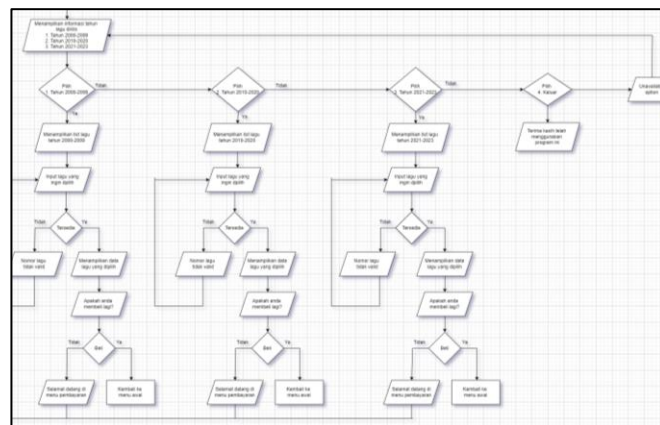
langsung masuk ke dalam keranjang belanja. Dan apabila pembeli tidak jadi membeli lagu tersebut, maka akan ada pilihan kembali ke menu awal. Begitu juga dengan penyanyi lagu lainnya yaitu *girlband* dan *band*.



Gambar 2.3.1 Flowchart Filter Penyanyi

2.4 Flowchart Filter Tahun

Subbab ini menjelaskan mengenai *flowchart* bagian filter berdasarkan tahun lagu untuk memunculkan menu genre dengan pilihan rentang 10 tahun yaitu 2000-2010, 2011-2020, dan 2021 sampai sekarang.

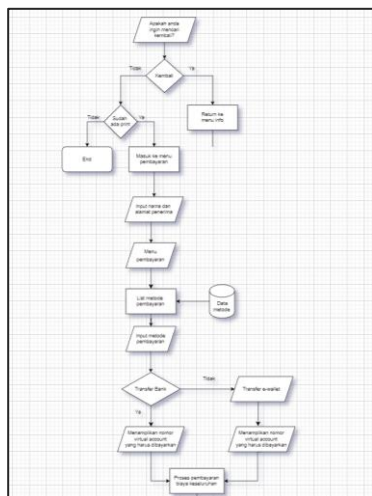


Gambar 2.4.1 Flowchart filter tahun

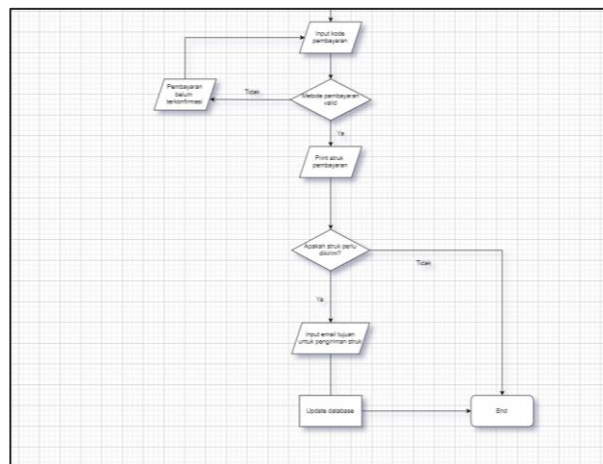
2.5 Flowchart Pembayaran

Subbab ini menjelaskan mengenai flowchart pembayaran pada program MyuList. Pada flowchart bagian pembayaran ini pembeli diharapkan untuk melakukan input data diri seperti nama dan alamat penerima (user). Pada bagian ini juga terdapat menu pembayaran yang terdiri atas metode pembayaran berupa *Gopay*, Mandiri dan BNI (seperti yang tertera pada gambar 2.5.1).

Pada flowchart ini juga terdapat program yang memunculkan total biaya yang harus dibayarkan beserta dengan biaya pengiriman CD. Setelah proses pembayaran selesai dan berhasil, program akan melakukan print terhadap struk pembelian. Dan setelah semua selesai, akan terdapat *end* untuk mengakhiri proses pembelian pada program ini.



Gambar 2.5.1 Flowchart pembayaran (input data pembeli dan metode pembayaran)



Gambar 2.5.2 Flowchart pembayaran (proses)

BAB III

KODE PROGRAM

3.1 Program Data

Subbab ini menjelaskan mengenai program untuk meng-*import library*, yaitu csv untuk membaca file yang berbentuk csv, pandas untuk menganalisis data csv, dan random untuk menghasilkan angka acak saat pembayaran. Pada program ini juga terdapat program untuk membaca data csv yang sudah dibuat dengan kode `pd.read_csv`.

```
1 import pandas as pd
2 import csv
3 import random
4
5 # Define the shopping cart as a global variable
6 cart = []
7
8 # Load song data from CSV files
9 data_lagu_rock = pd.read_csv('data_rock.csv')
10 data_lagu_pop = pd.read_csv('data_pop.csv')
11 data_lagu_kpop = pd.read_csv('data_kpop.csv')
12 data_lagu_boyband = pd.read_csv('data_boyband.csv')
13 data_lagu_girlband = pd.read_csv('data_girlband.csv')
14 data_lagu_band = pd.read_csv('data_band.csv')
15 data_lagu_2000_2010 = pd.read_csv('data_2000-2010.csv')
16 data_lagu_2011_2020 = pd.read_csv('data_2011-2020.csv')
17 data_lagu_2021_2023 = pd.read_csv('data_2021-keatas.csv')
18
```

Gambar 3.1.1 Program bagian database

3.2 Fungsi Menu Awal

Subbab ini menjelaskan mengenai menu awal. Terdapat ucapan selamat datang sebelum menjalankan program dan terdapat menu untuk melakukan filter lagu berdasarkan genre, penyanyi, tahun, dan keluar dari program. Selain itu, pada program bagian ini digunakan percabangan `if-elif-else` dan eksepsi agar saat user salah melakukan input yang diperintahkan, maka akan kembali ke menu awal.


```
# Fungsi Menu Awal
def menu_awal():
    print("Selamat Datang di Program Myulist".center(115))
    print("=====".center(115))
    print("Program ini merupakan program jual beli CD lagu secara online".center(115))

    print("1. Genre")
    print("2. Penyanyi")
    print("3. Tahun")
    print("4. Keluar")

    try:
        pilihan = int(input("Masukkan pilihan anda: "))
        if pilihan == 1:
            genre()
            membeli_kembali()
        elif pilihan == 2:
            penyanyi()
            membeli_kembali()
        elif pilihan == 3:
            tahun()
            membeli_kembali()
        elif pilihan == 4:
            print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
            pass
        else:
            print("Pilihan yang anda masukkan tidak valid!")
            menu_awal()
    except ValueError as v:
        print("Terjadi kesalahan: ", v)
        menu_awal()
```

Gambar 3.2.1 Program bagian menu awal

3.3 Fungsi Genre

Subbab ini menjelaskan mengenai fungsi genre. Pada bagian ini terdapat menu berupa filter lagu berdasarkan genre Rock, Pop, dan Kpop. Pada bagian ini digunakan percabangan if-elif-else untuk user memilih ingin melakukan filter lagu berdasarkan genre yang diinginkan serta terdapat informasi mengenai genre yang dipilih. Pada program bagian fungsi genre ini juga digunakan eksepsi agar saat user melakukan kesalahan saat menginput dapat kembali ke menu pemilihan filter genre.

```
# Fungsi Genre
def genre():
    print("1. Rock")
    print("2. Pop")
    print("3. K-Pop")
    print("4. Keluar")

    try:
        g = int(input("Masukkan pilihan anda: "))
        if g == 1:
            print("Genre musik rock adalah suatu genre musik populer yang mulai")
            print("tumbuh sejak era 50an. Musik rock terbentuk karena pengaruh musik R&B dan country di era 40an.")
            tabel_rock()
            add_to_cart(data_lagu_rock)
        elif g == 2:
            print("Genre musik pop adalah salah satu genre musik yang terkenal. Bahkan genre musik pop bisa")
            print("dibilang sebagai aliran musik paling populer dan paling banyak dinyanyikan oleh penyanyi dan band di seluruh dunia.")
            add_to_cart(data_lagu_pop)
        elif g == 3:
            print("Genre musik k-pop adalah genre musik populer yang berasal dari Korea Selatan.")
            print("Lagu-lagu K-pop banyak dipengaruhi oleh berbagai genre musik, seperti hip-hop, electronic dance, jazz, dan rock.")
            tabel_kpop()
            add_to_cart(data_lagu_kpop)
        elif g == 4:
            print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
            pass
        else:
            print("INPUT SALAH")
            genre()
    except ValueError as v:
        print("Terjadi kesalahan: ", v)
        genre()
```

Gambar 3.3.1 Program bagian fungsi genre

3.4 Fungsi Penyanyi

Subbab ini menjelaskan mengenai fungsi penyanyi. Pada bagian ini terdapat menu berupa filter lagu berdasarkan penyanyi boyband, girlband, dan band. Pada bagian ini digunakan percabangan if-elif-else untuk user memilih ingin melakukan filter lagu berdasarkan penyanyi yang diinginkan serta terdapat informasi mengenai jenis penyanyi yang dipilih. Pada program bagian fungsi penyanyi ini juga digunakan eksepsi agar saat user melakukan kesalahan saat menginput dapat kembali ke menu pemilihan filter penyanyi.

```
# Fungsi Penyanyi
def penyanyi():
    print("1. Boyband")
    print("2. Girlband")
    print("3. Band")
    print("4. Keluar")

    p = int(input("Masukkan pilihan anda: "))
    # looping
    if p == 1:
        print("Boy band adalah sejenis kelompok musik pop atau R&B yang terdiri dari tiga anggota atau lebih.")
        print("semuanya penyanyi laki-laki muda.")
        tabel_boyband()
        add_to_cart(data_lagu_boyband)
    elif p == 2:
        print("Girl band/Grup vokal wanita adalah kelompok musik populer yang terdiri atas")
        print("beberapa penyanyi wanita muda yang biasanya melakukan penampilan musik yang selaras.")
        tabel_girlband()
        add_to_cart(data_lagu_girlband)
    elif p == 3:
        print("Band atau ansambel musik merupakan kumpulan yang terdiri atas dua atau lebih musisi")
        print("yang memainkan alat musik ataupun bernyanyi.")
        tabel_band()
        add_to_cart(data_lagu_band)
    elif p == 4:
        print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
        pass
    else:
        print("INPUT SALAH")
        penyanyi()
```

Gambar 3.4.1 Program bagian fungsi penyanyi

3.5 Fungsi Tahun

Subbab ini menjelaskan mengenai fungsi tahun. Pada bagian ini terdapat menu berupa filter lagu berdasarkan rentang tahun 2000-2010, tahun 2011-2020, dan 2021 ke atas. Pada bagian ini digunakan percabangan if-elif-else untuk user memilih ingin melakukan filter lagu berdasarkan rentang tahun yang diinginkan serta terdapat informasi mengenai tahun yang dipilih. Pada program bagian fungsi penyanyi ini juga digunakan eksepsi agar saat user melakukan kesalahan saat menginput dapat kembali ke menu pemilihan filter tahun.

```
# Fungsi Tahun
def tahun():
    print("1. Tahun 2000-2010")
    print("2. Tahun 2011-2020")
    print("3. >=Tahun 2021")
    print("4. Keluar")

    t = int(input("Masukkan pilihan anda: "))
    # Looping
    if t == 1:
        print("Musik ini adalah musik pada era 2000-2010 yang terdiri dari genre Rock, Pop dan Kpop")
        tabel_lagu2000()
        add_to_cart(data_lagu_2000_2010)
    elif t == 2:
        print("Musik ini adalah musik pada era 2011-2020 yang terdiri dari genre Rock, Pop dan Kpop")
        tabel_lagu2011()
        add_to_cart(data_lagu_2011_2020)
    elif t == 3:
        print("Musik ini adalah musik pada era lebih dari 2021 yang terdiri dari genre Rock, Pop dan Kpop")
        tabel_lagu2021()
        add_to_cart(data_lagu_2021_2023)
    elif t == 4:
        print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
        pass
    else:
        print("INPUT SALAH")
        tahun()
```

Gambar 3.5.1 Program bagian fungsi tahun

3.6 Fungsi Tabel Genre Rock

Subbab ini menjelaskan mengenai fungsi untuk membuat tabel genre Rock dan menampilkan data lagu rock yang dimiliki dalam bentuk tabel dengan menggunakan iterasi for yang berisi nomor list, genre, nama penyanyi dan judul, tahun lagu dirilis, harga lagu, serta *link* Youtube agar pengguna dapat mendengarkan lagu sebelum memilih dan membeli lagunya. Pada bagian ini, data lagu Rock dapat diakses dengan membaca csv. Program ini sama seperti pada program fungsi tabel genre Pop dan Kpop.

```
def tabel_rock():
    data_lagu_rock = []
    with open('data_rock.csv', 'r') as file:
        reader = csv.reader(file)
        # Mem baca baris header (kolom)
        header = next(reader)
        data_lagu_rock.append(header)

        # Mem baca baris data
        for row in reader:
            data_lagu_rock.append(row)

    print('\n')
    print('TABEL'.center(113))
    print('DATA LAGU ROCK'.center(113))
    print('\n')
    print(' '*113).center(113)

    # Header
    print('{' + '{:~4}' | (header[0]:~8) | (header[1]:~45) | (header[2]:~8) | (header[3]:~8) | (header[4]:~28) |'.center(113))
    print('{' + '{:~4}' | (header[0]:~8) | (header[1]:~45) | (header[2]:~8) | (header[3]:~8) | (header[4]:~28) |'.center(113))

    # Data lagu rock
    for i in range(1, len(data_lagu_rock)):
        print('{' + '{:~4}' | (data_lagu_rock[i][0]:~8) | (data_lagu_rock[i][1]:~45) | (data_lagu_rock[i][2]:~8) | (data_lagu_rock[i][3]:~8) | (data_lagu_rock[i][4]:~28) |'.center(113))
        print('{' + '{:~4}' | (data_lagu_rock[i][0]:~8) | (data_lagu_rock[i][1]:~45) | (data_lagu_rock[i][2]:~8) | (data_lagu_rock[i][3]:~8) | (data_lagu_rock[i][4]:~28) |'.center(113))
```

Gambar 3.6.1 Program bagian fungsi tabel genre Rock

3.7 Fungsi Tabel Penyanyi Boyband

Subbab ini menjelaskan mengenai fungsi untuk membuat tabel penyanyi boyband dan menampilkan data lagu penyanyi boyband yang dimiliki dalam bentuk tabel dengan menggunakan iterasi for yang berisi nomor list, genre, nama penyanyi dan judul, tahun lagu dirilis, harga lagu, serta *link* Youtube agar pengguna dapat mendengarkan lagu sebelum memilih dan membeli lagunya. Pada bagian ini, data

penyanyi boyband dapat diakses dengan membaca csv. Program ini sama seperti pada program fungsi tabel penyanyi girlband dan band.

```
#TABEL BOYBAND
def tabel_boyband():
    data_lagu_boyband = []
    with open('data_boyband.csv', 'r') as file:
        reader = csv.reader(file)
        # Membaca baris header (kolom)
        header = next(reader)
        data_lagu_boyband.append(header)

        # Membaca baris data
        for row in reader:
            data_lagu_boyband.append(row)

    print("")
    print("TABEL".center(113))
    print("DATA LAGU BOYBAND".center(113))
    print("")
    print(":".center(113))

    # Header
    print(f"| {header[0]:^45} | {header[1]:^8} | {header[2]:^8} | {header[3]:^8} | {header[4]:^28} |".center(113))
    print(":".center(113))

    # Data lagu boyband
    for i in range(1, len(data_lagu_boyband)):
        print(f"| {i:^4} | {data_lagu_boyband[i][0]:^45} | {data_lagu_boyband[i][1]:^8} | {data_lagu_boyband[i][2]:^8} | {data_lagu_boyband[i][3]:^8} | {data_lagu_boyband[i][4]:^28} |".center(113))
        print(":".center(113))
```

Gambar 3.7.1 Program bagian fungsi tabel penyanyi boyband

3.8 Fungsi Tabel Tahun 2000-2010

Subbab ini menjelaskan mengenai fungsi untuk membuat tabel tahun 2000 hingga 2010 dan menampilkan data lagu pada rentang tahun tersebut yang dimiliki dalam bentuk tabel dengan menggunakan iterasi for yang berisi nomor list, genre, nama penyanyi dan judul, tahun lagu dirilis, harga lagu, serta *link* Youtube agar pengguna dapat mendengarkan lagu sebelum memilih dan membeli lagunya. Pada bagian ini, data tahun dapat diakses dengan membaca csv. Program ini sama seperti pada program fungsi tahun 2011-2020 dan 2021 ke atas.

```
#TABEL Tahun 2000-2010
def tabel_lagu2000():
    data_lagu_2000_2010 = []
    with open('data_2000-2010.csv', 'r') as file:
        reader = csv.reader(file)
        # Membaca baris header (kolom)
        header = next(reader)
        data_lagu_2000_2010.append(header)

        # Membaca baris data
        for row in reader:
            data_lagu_2000_2010.append(row)

    print("")
    print("TABEL".center(113))
    print("DATA LAGU Tahun 2000-2010".center(113))
    print("")
    print(":".center(113))

    # Header
    print(f"| {header[0]:^45} | {header[1]:^8} | {header[2]:^8} | {header[3]:^8} | {header[4]:^28} |".center(113))
    print(":".center(113))

    # Data lagu tahun 2000-2010
    for i in range(1, len(data_lagu_2000_2010)):
        print(f"| {i:^4} | {data_lagu_2000_2010[i][0]:^45} | {data_lagu_2000_2010[i][1]:^8} | {data_lagu_2000_2010[i][2]:^8} | {data_lagu_2000_2010[i][3]:^8} | {data_lagu_2000_2010[i][4]:^28} |".center(113))
        print(":".center(113))
```

Gambar 3.8.1 Program bagian fungsi tabel tahun 2000-2010

3.9 Fungsi Menambahkan Lagu ke Keranjang

Subbab ini menjelaskan mengenai fungsi menambahkan lagu ke keranjang. Pada program bagian ini, terdapat percabangan if-else serta pengguna memasukkan nomor lagu yang ingin dibeli agar lagu yang dibeli dapat ditambahkan ke keranjang. Saat pengguna memasukkan nomor yang lebih kecil dari satu atau melebihi jumlah lagu yang berada di data, maka akan tercetak "Nomor lagu tidak valid" atau ketika pengguna memasukkan nomor yang benar, maka lagu akan dimasukkan ke keranjang menggunakan salah satu metode pandas, yaitu *append*.

```
# Fungsi untuk menambahkan lagu ke keranjang
def add_to_cart(products):
    pilihan = int(input("Masukkan nomor lagu yang ingin ditambahkan ke keranjang: "))
    if pilihan < 1 or pilihan > len(products):
        print("Nomor lagu tidak valid!")
        add_to_cart(products)
    else:
        song = products.loc[pilihan - 1, :]
        cart.append(song)
        print(f"Lagu {song['name']} telah ditambahkan ke keranjang!")
        input("Tekan ENTER untuk lanjut \n")
```

Gambar 3.9.1 Program bagian fungsi tabel tahun 2000-2010

3.10 Fungsi Menghapus Item dalam Keranjang.

Subbab ini menjelaskan mengenai fungsi menghapus item dalam keranjang menggunakan percabangan if-else dan iterasi for. Pengguna harus memasukkan nomor lagu yang ingin dihapus agar lagu dapat dihapus menggunakan salah satu metode pandas, yaitu pop.

```
# Fungsi untuk menghapus item dari keranjang
def remove_from_cart():
    if not cart:
        print("Keranjang masih kosong!")
        input("Tekan ENTER untuk lanjut \n")
    else:
        print("== Isi Keranjang ==")
        print("-----")
        for i, song in enumerate(cart):
            print(f"{i+1}. {song['genre']}, {song['name']}, {song['year']}, {song['link']}: Rp{song['price']}")
        pilihan = int(input("Masukkan nomor lagu yang ingin dihapus dari keranjang: "))
        if pilihan < 1 or pilihan > len(cart):
            print("Nomor lagu tidak valid!")
            remove_from_cart()
        else:
            song = cart.pop(pilihan - 1)
            print(f"Lagu {song['name']} telah dihapus dari keranjang!")
            input("Tekan ENTER untuk lanjut \n")
```

Gambar 3.10.1 Program bagian fungsi menghapus isi dalam keranjang

3.11 Fungsi Menampilkan Isi Keranjang

Subbab ini menjelaskan bagian fungsi menampilkan isi keranjang. Pada program ini digunakan percabangan if-else dan iterasi for. Pengguna dapat melihat lagu apa saja yang sudah dipilih, apabila sebelumnya pengguna menghapus semua isi keranjang, maka keranjang akan menampilkan bahwa keranjang masih kosong.

```
# Fungsi untuk menampilkan isi keranjang
def display_cart():
    if not cart:
        print("Keranjang masih kosong!")
    else:
        print("== Isi Keranjang ==")
        print("-----")
        for i, song in enumerate(cart):
            print(f"{i+1}. {song['genre']}, {song['name']}, {song['year']}, {song['link']}: Rp{song['price']}")
        input("Tekan ENTER untuk lanjut \n")
```

Gambar 3.11.1 Program bagian fungsi menghapus isi dalam keranjang

3.12 Fungsi Menu Pembelian

Subbab ini menjelaskan bagian fungsi menu pembelian. Terdapat empat menu, yaitu menampilkan isi keranjang, menghapus isi keranjang, melanjutkan pembayaran, dan keluar dari menu. Pada program ini, menggunakan percabangan *if-elif-else* dan *nested if*. Pada fungsi ini juga dapat menambahkan kembali isi keranjang dengan memasukan *yes* atau masuk ke menu pembayaran dengan pilihan *no*.

```
#Menu Pembelian
def kembali_pembelian():
    pilihan_kembali = str(input("Apakah anda ingin membeli lagi? (y/n): "))
    while True:
        if pilihan_kembali == "y":
            menu_awal()
        elif pilihan_kembali == "n":
            print("1. Keranjang")
            print("2. Hapus dari Keranjang")
            print("3. Lanjutkan pembayaran")
            print("4. Keluar")
            pilihan_pembelian = int(input("Masukkan pilihan anda: "))
            if pilihan_pembelian == 1:
                display_cart()
            elif pilihan_pembelian == 2:
                remove_from_cart()
            elif pilihan_pembelian == 3:
                checkout()
                break
            elif pilihan_pembelian == 4:
                print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
                break
            else:
                print("INPUT SALAH")
                kembali_pembelian()
        else:
            kembali_pembelian()
```

Gambar 3.12.1 Program bagian fungsi menu pembelian

3.13 Fungsi Melanjutkan Pembayaran

Subbab ini menjelaskan bagian fungsi melanjutkan pembayaran. Program ini menggunakan percabangan *if-else*, *nested if*, dan iterasi *for*. Pada saat keranjang kosong atau pengguna menghapus semua isi keranjang, maka keranjang akan bertanya apakah pengguna ingin kembali ke menu awal untuk menambahkan lagu atau berhenti menggunakan program. Pada saat keranjang memiliki isi, maka akan muncul lagu apa saja yang dipilih dan berapa total harga yang harus dibayarkan, tetapi belum termasuk biaya pengiriman CD.

```
# Fungsi untuk melanjutkan pembayaran
def checkout():
    global total_price
    if not cart:
        print("Keranjang masih kosong!")
        input("Tekan ENTER untuk lanjut \n")
        pilihan_checkout_kembali = input("Apakah anda ingin kembali ke menu awal? (y/n)")
        if pilihan_checkout_kembali == "y":
            menu_awal()
        elif pilihan_checkout_kembali == "n":
            print("\nTERIMA KASIH TELAH MENGGUNAKAN PROGRAM INI\n")
        else:
            print("Input yang anda masukkan salah")
            checkout()

    else:
        print("")
        print("=====MY LIST=====")
        print("-----")
        total_price = sum(song['price'] for song in cart)
        for song in cart:
            print(f"{song['name']}: Rp{song['price']}")
        print(f"Total Harga: Rp{total_price}")
        input("Tekan ENTER untuk lanjut \n")
        datapengguna()
        strukbeli()
        metodebayar()
        # Reset the cart
        cart.clear()
```

Gambar 3.13.1 Program bagian fungsi melanjutkan pembayaran

3.14 Fungsi Data Pengguna

Subbab ini menjelaskan mengenai fungsi data pengguna. Program ini menggunakan iterasi for dan mengharuskan pengguna untuk memasukkan data diri berupa nama, nomor ponsel, alamat rumah, dan kota domisili. Global digunakan agar data diri dapat dipanggil dalam fungsi lain.

```
# Input Data User
def datapengguna():
    global nama
    global phone
    global rumah
    global kota

    print('          DATA USER          ')
    nama = input('Masukkan Nama Anda          : ')
    for name in nama:
        if name.isdigit():
            print("Tolong Masukan Dengan Huruf")
            datapengguna()
    phone = int(input('Masukkan No HP Anda          :+62 '))
    rumah = input('Masukkan Alamat Anda          : ')
    kota = str(input('Masukkan Kota Anda          : '))
    print()
    print('=====')
    print()
```

Gambar 3.14.1 Program bagian fungsi data pengguna

3.15 Fungsi Struk Pembelian

Subbab ini menjelaskan bagian struk pembelian. Pada program ini, terdapat modul random yang digunakan untuk menghasilkan kode acak dalam pembayaran. Selain itu, di dalam program ini menggunakan pengulangan while true, percabangan if-else, dan eksepsi. Pengguna harus memasukkan kode pembayaran yang tepat agar struk dapat dicetak, jika kode pembayaran yang dimasukkan salah, maka akan kembali ke kode pelunasan.

```
def struk_beli():
    # kode pelunasan
    print()
    lunas = random.randrange(55555, 77777, 10)
    print("Kode Pembayaran anda adalah", lunas)
    print("Silahkan membayar ke No. Akun tersebut")
    while True:
        try:
            print()
            bayarr = int(input("Masukan Kode Pelunasan: "))
            if bayarr == lunas:
                print("Pembayaran Dengan No Akun", bayarr, "Berhasil. \n Hubungi Admin Jika Ada Kendala")
                break
            else:
                print("Maaf, Kode Pelunasan Salah!")
        except ValueError as v:
            print("Terjadi kesalahan: ", v)
            kode_pelunasan()

    # Menampilkan struk
    print()
    print('-----MENGENAL DATA PENERIMA BARANG-----')
    nama_penerima = str(input('Masukkan nama penerima          : '))
    nohp_penerima = int(input('Masukkan no HP penerima          : +62 '))
    alamat_penerima = str(input('Masukkan alamat penerima      : '))

    # Template Struk
    print()
    print('=====')
    print("STRUK PEMBELIAN")
    print('=====')
    print()
    print("Nama          : ", nama_penerima)
    print("Nomor HP       : +62", nohp_penerima)
    print("Alamat        : ", alamat_penerima)
    print("Kode Pembayaran : ", bayarr)
    print("total_pembayaran()")
    print("=====")
    print("TERIMA KASIH")
    print("=====")
    print()
    print("Terimakasih pesanan akan dikirim\n")
    print("Apabila belum terkirim dapat hubungi admin.")
    menu_awal()
```

Gambar 3.15.1 Program bagian fungsi Struk Pembelian

3.16 Fungsi Biaya Pengiriman

Subbab ini menjelaskan bagian Biaya Ppngiriman. Pada program ini, terdapat variabel biaya-biaya pengiriman di beberapa kota seperti Kota Semarang, Salatiga, Malang, dan kota lainnya. Program ini menggunakan percabangan if-elif-else dan menggunakan eksepsi, sehingga saat pengguna salah memasukkan kota akan kembali ke pemilihan kota yang dituju.

```
#Ongkos Kirim
# Kota Jogja
ongkir_jgj = 5000
# Kota Semaarang
ongkir_smg = 7000
# Kota Surabaya
ongkir_sby = 9000
# Kota Malang
ongkir_mlg = 11000
# Kota Salatiga
ongkir_slt = 6000
# alamat ongkir
def ongkir():
    global pay
    print('Barang anda akan diantarkan jika anda berada pada kota jangkauan')
    try:
        kota = input("Kota (Jogja/Semarang/Surabaya/Malang/Salatiga): ")
        if kota == "Jogja":
            pay = ongkir_jgj + total_price
            print("Total yang harus dibayarkan: Rp {}".format(pay))
        elif kota == "Semarang":
            pay = ongkir_smg + total_price
            print("Total yang harus dibayarkan: Rp {}".format(pay))
        elif kota == "Surabaya":
            pay = ongkir_slt + total_price
            print("Total yang harus dibayarkan: Rp {}".format(pay))
        elif kota == "Malang":
            pay = ongkir_mlg + total_price
            print("Total yang harus dibayarkan: Rp {}".format(pay))
        elif kota == "Salatiga":
            pay = ongkir_slt + total_price
            print("Total yang harus dibayarkan: Rp {}".format(pay))
        else:
            print("Kota yang dicantumkan tidak terdeteksi.")
            ongkir()
    except ValueError as v:
        print("Terjadi kesalahan: ", v)
        ongkir()
```

Gambar 3.16.1 Program bagian fungsi Biaya Pengiriman

3.17 Fungsi Metode Pembayaran

Subbab ini menjelaskan bagian Metode Pembayaran. Program ini menggunakan percabangan if-elif-else dan eksepsi, sehingga saat pengguna memasukkan angka di luar metode pembayaran yang tersedia, pengguna akan kembali ke pertanyaan mengenai metode pembayaran yang akan dipilih.

```
# Menu Metode Bayar
def metodebayar():
    global bayar
    print("Pilih Metode Pembayaran\n"
          "[1] Gopay\n"
          "[2] Bank Mandiri\n"
          "[3] Bank BNI\n")

    # Pengguna memilih metode pembayaran

    try:
        bayar = int(input('Masukkan pilihan anda: '))
        if bayar == 1:
            print('silahkan lakukan pembayaran ke akun gopay kami')
            print('081563184352 A.N. Gabeta')
            print('lalu diikuti dengan 5 digit kode pembayaran')
            print('contoh 081563184352-55555')
            struk()
        elif bayar == 2:
            print('silahkan lakukan pembayaran ke rekening mandiri kami')
            print('720839214 A.N. Kasturia')
            print('lalu diikuti dengan 5 digit kode pembayaran')
            print('contoh 720839214-55555')
            struk()
        elif bayar == 3:
            print('silahkan lakukan pembayaran ke rekening BNI kami')
            print('081567878985 A.N. Katon')
            print('lalu diikuti dengan 5 digit kode pembayaran')
            print('contoh 081567878985-55555')
            struk()
        else:
            print("Maaf input yang anda masukkan salah")
            metodebayar()
    except ValueError as v:
        print("Terjadi kesalahan: ", v)
        metodebayar()
```

Gambar 3.17.1 Program bagian fungsi Metode Pembayaran

BAB IV

HASIL RUNNING PROGRAM

Bab ini menjelaskan tentang hasil *running* dari program Myulist.

4.1. Memilih Filter Lagu

Subbab ini menjelaskan mengenai hasil *running* dalam pemilihan filter lagu dalam program Myulist, seperti menampilkan pemilihan genre lagu, penyanyi, dan juga tahun rilis dari lagu. Apabila pengguna mengetik angka 1, maka akan muncul beberapa genre lagu yang dapat dipilih, seperti rock, pop, dan kpop. Apabila pengguna mengetik angka 2, maka program akan menampilkan list daftar penyanyi seperti *boyband*, *girlband*, dan juga *band*. Selain itu, apabila pengguna mengetik angka 3, maka program akan menampilkan daftar tahun rilis setiap lagu dari berbagai genre, mulai dari tahun 2000 s.d. 2010, 2011 s.d. 2020, dan lebih dari tahun 2021.

```

Selamat Datang di Program Myulist
=====
Program ini merupakan program jual beli album lagu secara online

1. Genre
2. Penyanyi
3. Tahun
4. Keluar
Masukkan pilihan anda: 1
1. Rock
2. Pop
3. K-Pop
4. Keluar
Masukkan pilihan anda: 1
Genre musik rock adalah suatu genre musik populer yang mulai
tumbuh sejak era 50an. Musik rock terbentuk karena pengaruh musik R&B dan country di era 40an.

TABEL
DATA LAGU ROCK

```

No	genre	name	price	year	link
1	Rock	Linkin Park-In the End	6000 S	2004	https://youtu.be/Soa3g07tl-c
2	Rock	Green Day-Boulevard of Broken Dreams	6000 S	2004	https://youtu.be/Soa3g07tl-c
3	Rock	The White Stripes-Seven Nation Army	6000 S	2003	https://youtu.be/0J2QdDbeImY
4	Rock	Foo Fighters-The Pretender	6000 S	2007	https://youtu.be/SBjQ9tuwTQ
5	Rock	The White Stripes-Seven Nation Army	6000 S	2003	https://youtu.be/0J2QdDbeImY
6	Rock	Foo Fighters-The Pretender	6000 S	2007	https://youtu.be/SBjQ9tuwTQ
7	Rock	Panic! at the Disco-I Write Sins Not Tragedies	6000 S	2006	https://youtu.be/vc6vs-15dkc
8	Rock	Avenged Sevenfold-Bet Country	6000 S	2005	https://youtu.be/OR0F3qtadJg
9	Rock	Black Veil Brides-In the End	6000 S	2012	https://youtu.be/f0EQI1zPowM
10	Rock	Halestorm-I Miss the Misery	6000 S	2012	https://youtu.be/YpJAmlnBxoA
11	Rock	Foo Fighters-Walk	6000 S	2011	https://youtu.be/4PkcFQt1bmU
12	Rock	Skillet-Monster	6000 S	2009	https://youtu.be/1mJ1M_RnsVE
13	Rock	Bullet for My Valentine-Tears Don't Fall	6000 S	2005	https://youtu.be/9sT0Q0dkl3Q
14	Rock	Avenged Sevenfold - Nightmare	6000 S	2010	https://youtu.be/94bGzWYHbu0
15	Rock	The Pretty Reckless-Heaven Knows	6000 S	2014	https://youtu.be/rHbXJcQ99jA

Gambar 4.1. Hasil Memilih Filter Lagu

Dari hasil tersebut diketahui bahwa pengguna dapat memilih lagu sesuai dengan filter yang ada.

4.2. Melakukan Input lagu

Subbab ini menjelaskan mengenai hasil *running* dalam menginputkan lagu yang diinginkan dalam *display*. Bagian ini menjelaskan mengenai lanjutan dari bagian sebelumnya yaitu memilih filter lagu. Pada saat pengguna telah menentukan genre lagu yang dia inginkan, maka program akan lanjut pada bagian input lagu. Pada bagian ini pengguna dapat menginputkan lagu sesuai dengan judul lagu yang tertera pada *display* sesuai dengan genre yang telah dipilih. Apabila judul lagu yang diinputkan salah atau berbeda dengan display maka program akan kembali ke pilihan lagu.

```

| 42 | Rock | Greta Van Fleet-Highway Tune | 6000 | S | 2017 | https://youtu.be/aJg40Jxp-co |
| 43 | Rock | Imagine Dragon-Radioactive | 6000 | S | 2012 | https://youtu.be/ktvTqknDobU |
| 44 | Rock | Fall Out Boy-Sugar We are GoIn Down | 6000 | S | 2005 | https://youtu.be/uHG-vLZrb-g |
| 45 | Rock | The Strokes-Last Nite | 6000 | S | 2001 | https://youtu.be/TOypSnKFHrE |
| 46 | Rock | Three Days Grace-I Hate Everthing About You | 6000 | S | 2003 | https://youtu.be/d8ekz_CSBVg |
| 47 | Rock | Queens of the Stone Age-No One Knows | 6000 | S | 2002 | https://youtu.be/s88e_q7oufE |
| 48 | Rock | Audioslave-Like a Stone | 6000 | S | 2002 | https://youtu.be/7QU1nvuxaMA |
| 49 | Rock | Muse-Starlight | 6000 | S | 2006 | https://youtu.be/Pgum6OT_VH8 |

Masukkan nomor lagu yang ingin ditambahkan ke keranjang: 1
Lagu Linkin Park-In the End telah ditambahkan ke keranjang!
Tekan ENTER untuk lanjut

Apakah anda ingin membeli lagi? (y/n): y

Selamat Datang di Program Myulist
Program ini merupakan program jual beli album lagu secara online

1. Genre
2. Penyanyi
3. Tahun
4. Keluar
Masukkan pilihan anda: 2
1. Boyband
2. Girlband
3. Band
4. Keluar
Masukkan pilihan anda: 2
Girl bandGrup vokal wanita adalah kelompok musik populer yang terdiri atas
beberapa penyanyi wanita muda yang biasanya melakukan penampilan musik yang selaras.

TABEL
DATA LAGU GIRLBAND

| No | genre | name | price | year | link |
| 1 | Kpop | Red Velvet-Happiness | 6000 | 2014 | https://www.youtube.com/watch?v=JFgv8BfxEs |
| 2 | Kpop | Red Velvet-you bett | 6000 | 2015 | https://www.youtube.com/watch?v=3PgGuJP8_jI |
| 3 | Kpop | Red Velvet-Power Up | 6000 | 2018 | https://www.youtube.com/watch?v=aiHSVQy9xd8 |
| 4 | Kpop | Red Velvet-Dumb Dumb | 6000 | 2015 | https://www.youtube.com/watch?v=XGd8aEDWp0 |

```

Gambar 4.2. Hasil Melakukan Input Lagu

Dari hasil tersebut diketahui bahwa pengguna dapat melakukan input lagu sesuai dengan filter yang telah dipilih.

4.3. Melakukan Pembayaran

Subbab ini menjelaskan mengenai hasil *running* dalam melakukan pembayaran sesuai dengan total nominal yang perlu dibayarkan. Nominal ini didapatkan dari hasil total nominal pada fitur keranjang belanja. Nominal yang perlu dibayarkan oleh setiap pengguna berbeda-beda bergantung pada harga barang dan kuantitasnya serta bergantung pada seberapa besar ongkir yang berlaku. Pada setiap kota diberlakukan ongkir yang berbeda-beda mulai dari Rp 5.000,00 hingga

Rp 11.000,00 tergantung seberapa jauh jarak pengiriman dari pada pengguna. Selain itu, program akan menampilkan data pengguna mulai dari nama, nomor handphone, alamat, dan asal kota. Tidak hanya itu, program juga menampilkan data penerima barang untuk mengantarkan barang ke kota jangkauan. Program ini menyediakan metode pembayaran e-wallet berupa gopay, bank mandiri, dan bank BNI. Setelah pengguna memilih metode yang ingin digunakan, pengguna dapat melanjutkan pembayaran melalui rekening yang telah disediakan beserta lima digit kode pembayarannya. Setelah itu, pengguna dapat mengisi kode pelunasan sesuai kode pembayaran yang tersedia. Apabila pembayaran telah berhasil maka akan tercetak struk pembelian dan resi penerima barang.

```

-----DATA USER-----
Masukkan Nama Anda           : Katon Adiluhung
Masukkan No HP Anda          : +62 8129849292
Masukkan Alamat Anda         : Jl. Surya
Masukkan Kota Anda           : Surakarta

=====

-----DATA PENERIMA BARANG-----
Masukkan nama penerima      : Gabeta Estefan
Masukkan no HP penerima     : +62 999102982
Masukkan alamat penerima    : Tembalang no.10
Barang anda akan diantarkan jika anda berada pada kota jangkauan
Kota (Jogja/Semarang/Surabaya/Malang/Salatiga): Semarang
Total yang harus dibayarkan: Rp 13000
Pilih Metode Pembayaran
[1] Gopay
[2] Bank Mandiri
[3] Bank BNI

Masukkan pilihan anda: 2
silahkan lakukan pembayaran ke rekening mandiri kami
720839214 A.N. Kasturia
lalu diikuti dengan 5 digit kode pembayaran
contoh 720839214-55555
Kode Pembayaran anda adalah 69005
Silahkan membayar ke No. Akun tersebut

Masukan Kode Pelunasan: 69005
Pembayaran Dengan No Akun 69005 Berhasil.
Hubungi Admin Jika Ada Kendala

=====
STRUK PEMBELIAN
=====

Nama           : Katon Adiluhung
Nomor HP       : +62 8129849292
Alamat         : Jl. Surya , Surakarta
Kode Pembayaran : 69005
Nominal Bayar  : Rp. 13000

=====
TERIMA KASIH
=====

=====
RESI PENERIMA BARANG
=====

```

Gambar 4.3. Hasil Proses Melakukan Pembayaran

```
=====
                        RESI PENERIMA BARANG
=====

Nama           : Gabeta Estefan
Nomor HP       : +62 999102982
Alamat        : Tembalang no.10
Kode Pembayaran : 69005
Nominal Bayar  : Rp. 13000
=====

                        TERIMA KASIH
=====

Terimakasih pesanan akan dikirim
Apabila belum terkirim dapat hubungi admin.
```

Gambar 4.4. Hasil Resi Penerima Barang

Dari hasil tersebut diketahui bahwa pengguna dapat melakukan pembayaran lagi sesuai dengan total pembayaran.