

LAPORAN PRAKTIKUM
POSTTEST 4
ALGORITMA PEMROGRAMAN DASAR

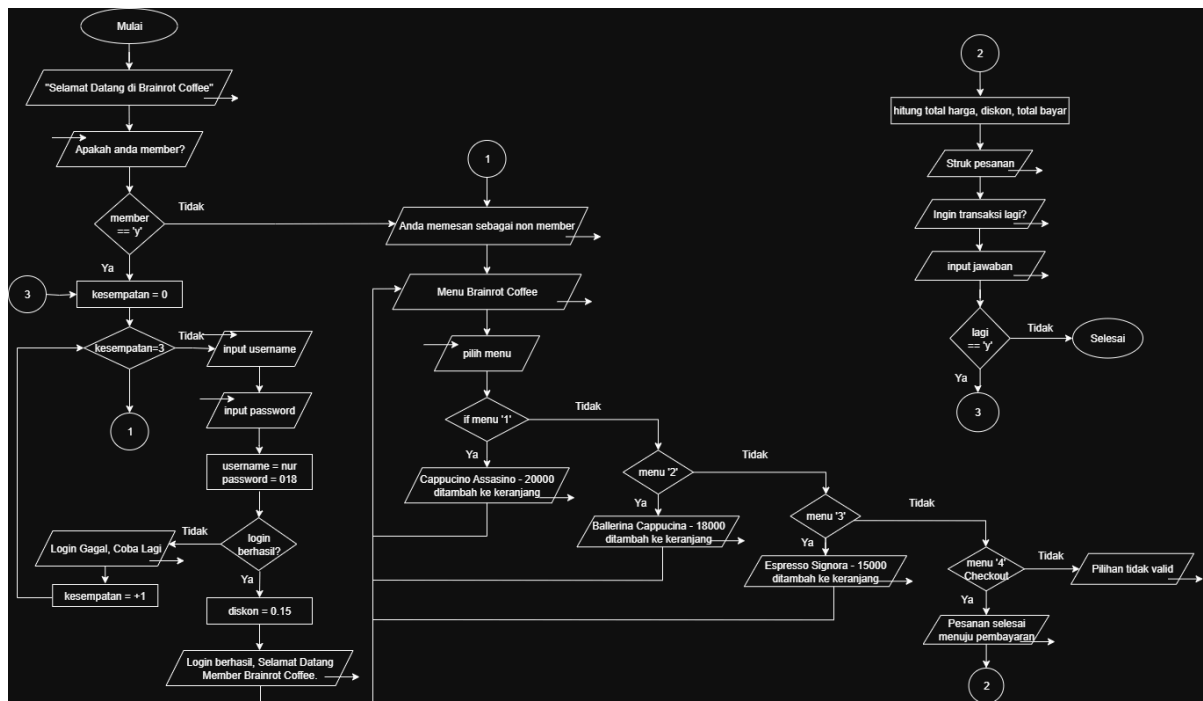


Disusun oleh:
Nur Azizah Islamiyah (2509106018)
Kelas (A1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart

Pertama, program akan menampilkan tulisan selamat datang. Lalu, program akan menanyakan status keanggotaan pembeli. Jika pembeli adalah member, program akan mengarahkan untuk login terlebih dahulu dengan menginput username dan password yang valid. Pembeli akan diberi 3 kesempatan untuk login, jika login berhasil pembeli akan mendapatkan potongan 15% dan program menampilkan "Login berhasil, Selamat Datang Member Brainrot Coffee". Lalu, jika login gagal, program akan langsung meneruskan ke pemesanan non member. Setelah proses login member maupun non member, sistem akan menampilkan menu dari Brainrot Coffee. Kemudian, pembeli bisa menginput apa saja menu yang ingin dibeli dan dimasukkan ke keranjang. Jika sudah, pembeli bisa memilih menu checkout untuk lanjut ke pembayaran. Terakhir, program menghitung total harga, diskon (member 15%, non member 0), dan menampilkan struk pembayarannya.



Gambar 1.1 Flowchart Program

2. Deskripsi Singkat Program

Program ini adalah sistem sederhana yang digunakan untuk melayani pembeli. Program ini bisa membedakan alur layanan untuk pembeli menjadi dua, yaitu member dan non member. Jika pembeli adalah seorang member, pembeli akan diarahkan untuk login terlebih dahulu. Lalu, jika login berhasil pembeli akan mendapat potongan 15% saat berbelanja di kedai. Dan untuk pembeli non member tidak akan mendapat diskon dalam pemesanannya.

3. Source Code

A. Fitur Login

Fitur ini digunakan untuk membedakan alur layanan pembeli (member, non member). Pada fitur ini, user diberi kesempatan 3 kali untuk login. Jika gagal login 3 kali maka sistem akan menganggapnya sebagai non-member dan langsung melanjutkan ke menu belanja tanpa mendapatkan diskon. Untuk member yang berhasil login akan diberikan diskon 15% di akhir transaksi.

Source Code:

```
#Login member
if status_member == 'y':
    print('\n--- Silakan Login Terlebih Dahulu ---')
    for percobaan in range(3):
        print(f'--- (Percobaan {percobaan + 1} dari 3) ---')
        input_username = input('Masukkan Username: ')
        input_password = input('Masukkan Password: ')

        if input_username == '' or input_password == '':
            print('Username atau Password tidak boleh kosong')
            if percobaan < 2:
                print(f'Sisa percobaan: {2 - percobaan}')
                continue

        if input_username == username_valid and input_password == password_valid:
            login_berhasil = True
            diskon_rate = 0.15
            print('Login Berhasil! Selamat Datang Member Brainrot Coffee')
```

```

        break
    else:
        print('Username atau Password salah, Coba Lagi!')
        if percobaan < 2:
            print(f'Sisa percobaan: {2 - percobaan}')

    if not login_berhasil:
        print('\nLogin Gagal, Transaksi Non Member')

if not login_berhasil:
    print('\n--- Anda memesan sebagai non member ---')

```

B. Pemesanan

Fitur ini digunakan untuk menampilkan menu dan memproses pemesanan. Di fitur ini, pembeli bisa menambahkan menu yg ingin dibeli ke dalam keranjang. Jika sudah, pembeli bisa memilih menu checkout untuk lanjut ke pembayaran.

Source Code:

```

#pemesanan
while True:
    print('\nMenu Brainrot Coffee')
    print('1. Cappucino Assasino - Rp. 20.000 ')
    print('2. Ballerina Cappucina - Rp. 18.000 ')
    print('3. Espresso Signora - Rp. 15.000 ')
    print('4. Checkout')
    print('-----')
    if total_harga > 0:
        print(f'Total Belanja Sementara: Rp {total_harga:,}')

    pilihan = input('\nMasukkan kode menu pesanan (1/2/3): ')

    if pilihan == '1':
        total_harga += 20000
        keranjang += '- Cappucino Assasino   Rp 20,000\n'
        print('-> "Cappucino Assasino" telah ditambahkan ke keranjang.')
    elif pilihan == '2':
        total_harga += 18000
        keranjang += '- Ballerina Cappucina   Rp 18,000\n'
        print('-> "Ballerina Cappucina" telah ditambahkan ke
keranjang.')
    elif pilihan == '3':
        total_harga += 15000
        keranjang += '- Espresso Signora     Rp 15,000\n'

```

```

        print('-> "Espresso Signora" telah ditambahkan ke keranjang.')
    elif pilihan == '4':
        print('Pesanan selesai, siap untuk pembayaran')
        break
    else:
        print('Pilihan tidak valid, coba lagi')

```

C. Struk Pembayaran

Fitur ini digunakan untuk menampilkan struk/rincian pesanan yang berisi daftar minuman yang dibeli, total harga, diskon (jika member), dan total bayar.

Source Code:

```

#struk
os.system('cls')
print('\n\n=====')
print('                STRUK PEMBAYARAN                ')
print('=====')

if keranjang == "":
    print('Anda tidak membeli apa pun')
else:
    print('Produk yang dibeli: ')
    print(keranjang)
    print('-----')

    jumlah_diskon = int(total_harga *diskon_rate)
    total_bayar = total_harga - jumlah_diskon

    if login_berhasil:
        print(f'Total Harga          : Rp {total_harga}')
        print(f'Diskon Member (15%)    : Rp {jumlah_diskon}')
        print('-----')
        print(f'Total yang Harus Dibayar : Rp {total_bayar}')
    else:
        print(f'Total yang Harus Dibayar : Rp {total_bayar}')

print('=====')
print('                Terima kasih telah membeli!                ')
print('=====')

```

D. Fitur Ingin Transaksi Lagi

Fitur ini digunakan untuk menanyakan pembeli ingin memulai transaksi lagi atau tidak. Jika pembeli menjawab 'y' maka program akan mengulangnya ke menu login, jika 'n' program akan berhenti dan menampilkan kata terimakasih.

Source Code:

```
#apakah ingin traksaksi
transaksi_lagi = input('\nApakah Anda ingin memulai transaksi baru?
(y/n): ')
if transaksi_lagi != 'y':
    break

print('\nTerimakasih, Jangan lupa untuk berkunjung lagi!')
```

E. Fitur Clear Terminal

Fitur ini digunakan untuk membersihkan terminal agar output program terlihat rapi. Fitur ini diletakkan sebelum pencetakan struk agar pembeli bisa fokus pada struk pembayarannya saja.

Source Code:

```
os.system('cls')
```

4. Hasil Output

```
=====
      SELAMAT DATANG DI BRAINROT COFFEE
=====
Apakah Anda member? (y/n) : y

--- Silakan Login Terlebih Dahulu ---
--- (Percobaan 1 dari 3) ---
Masukkan Username: nur
Masukkan Password: 018
Login Berhasil! Selamat Datang Member Brainrot Coffee

Menu Brainrot Coffee
1. Cappucino Assasino - Rp. 20.000
2. Ballerina Cappucina - Rp. 18.000
3. Espresso Signora - Rp. 15.000
4. Checkout
-----

Masukkan kode menu pesanan (1/2/3): 1
-> "Cappucino Assasino" telah ditambahkan ke keranjang.

Menu Brainrot Coffee
1. Cappucino Assasino - Rp. 20.000
2. Ballerina Cappucina - Rp. 18.000
3. Espresso Signora - Rp. 15.000
4. Checkout
-----

Total Belanja Sementara: Rp 20,000

Masukkan kode menu pesanan (1/2/3):
```

Gambar 5.1 Output Proses Pemesanan

```
=====
      STRUK PEMBAYARAN
=====
Produk yang dibeli:
- Cappucino Assasino    Rp 20,000

-----

Total Harga           : Rp 20000
Diskon Member (15%)   : Rp 3000
-----

Total yang Harus Dibayar : Rp 17000
=====

      Terima kasih telah membeli!
=====

Apakah Anda ingin memulai transaksi baru? (y/n): n

Terimakasih, Jangan lupa untuk berkunjung lagi!
```

Gambar 5.2 Struk Pembayaran

5. Langkah-langkah GIT

5.1 GIT Add

```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/praktikum-apd (main)
$ git add .
```

Gambar 5.1 Git add

Git add berfungsi untuk menambah file dan folder.

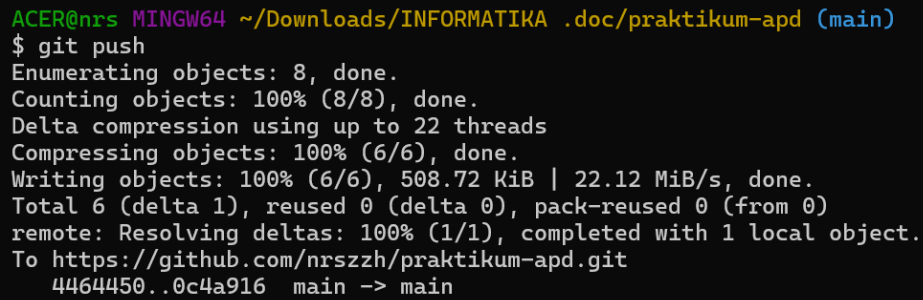
5.2 GIT Commit

```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/praktikum-apd (main)
$ git commit -m "PT-3"
[main 0c4a916] PT-3
2 files changed, 72 insertions(+)
create mode 100644 post-test/post-test-apd-3/2509106018-NUR AZIZAH ISLAMIYAH-PT-3.pdf
create mode 100644 post-test/post-test-apd-3/2509106018_NUR AZIZAH ISLAMIYAH_PT-3.py
```

Gambar 5.2 Git Commit

Git commit berfungsi untuk menyimpan perubahan di repository, dan pesan commit untuk menjelaskan perubahan.

5.3 GIT Push



```
ACER@nrs MINGW64 ~/Downloads/INFORMATIKA .doc/praktikum-apd (main)
$ git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 22 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 508.72 KiB | 22.12 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/nrszzh/praktikum-apd.git
 4464450..0c4a916  main -> main
```

Gambar 5.3 Git Push

Git push berfungsi untuk mengupload file yang ada di repository komputer ke repository Github.