

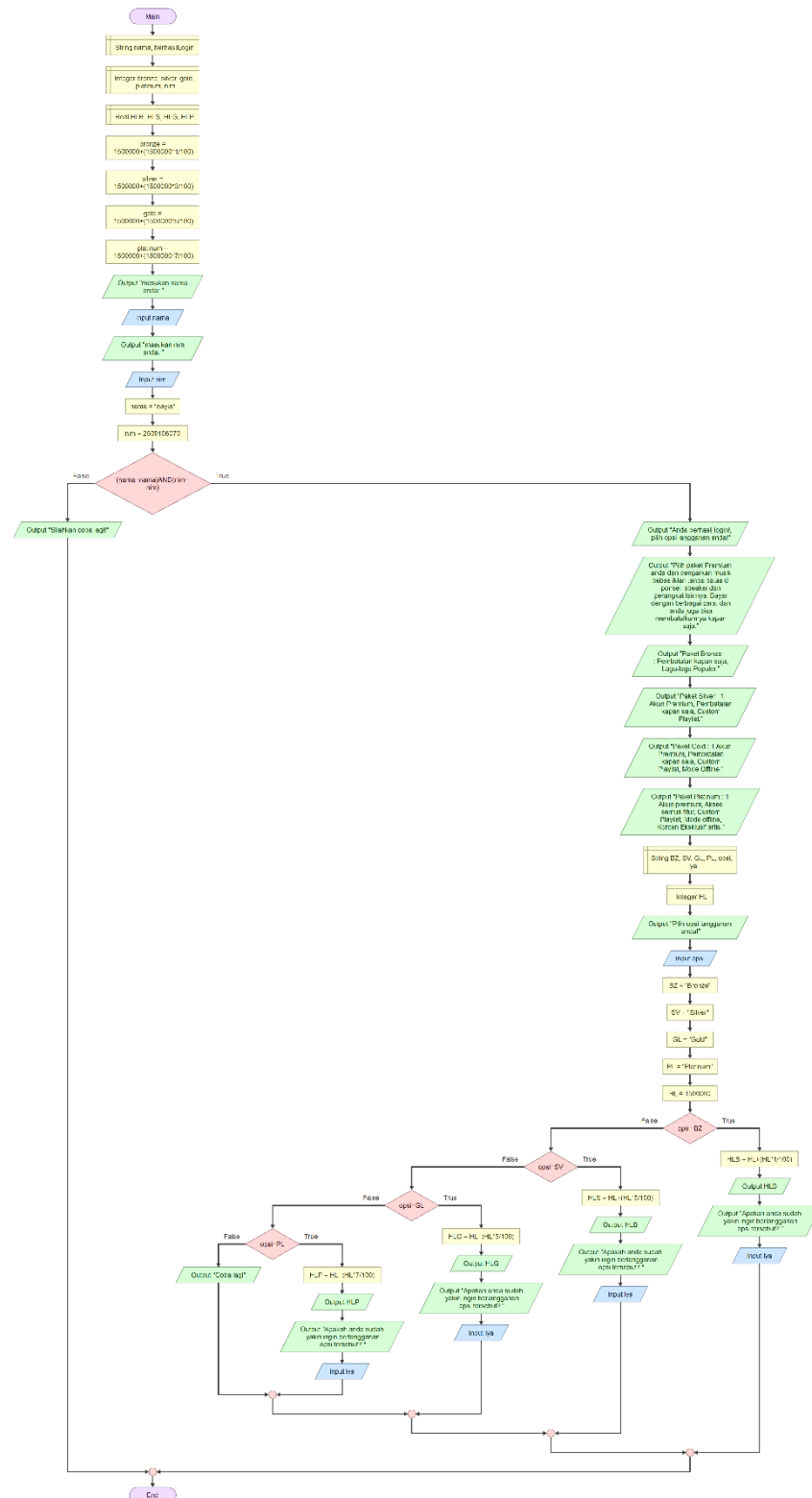
LAPORAN PRAKTIKUM
POSTTEST (3)
ALGORITMA PEMROGRAMAN DASAR



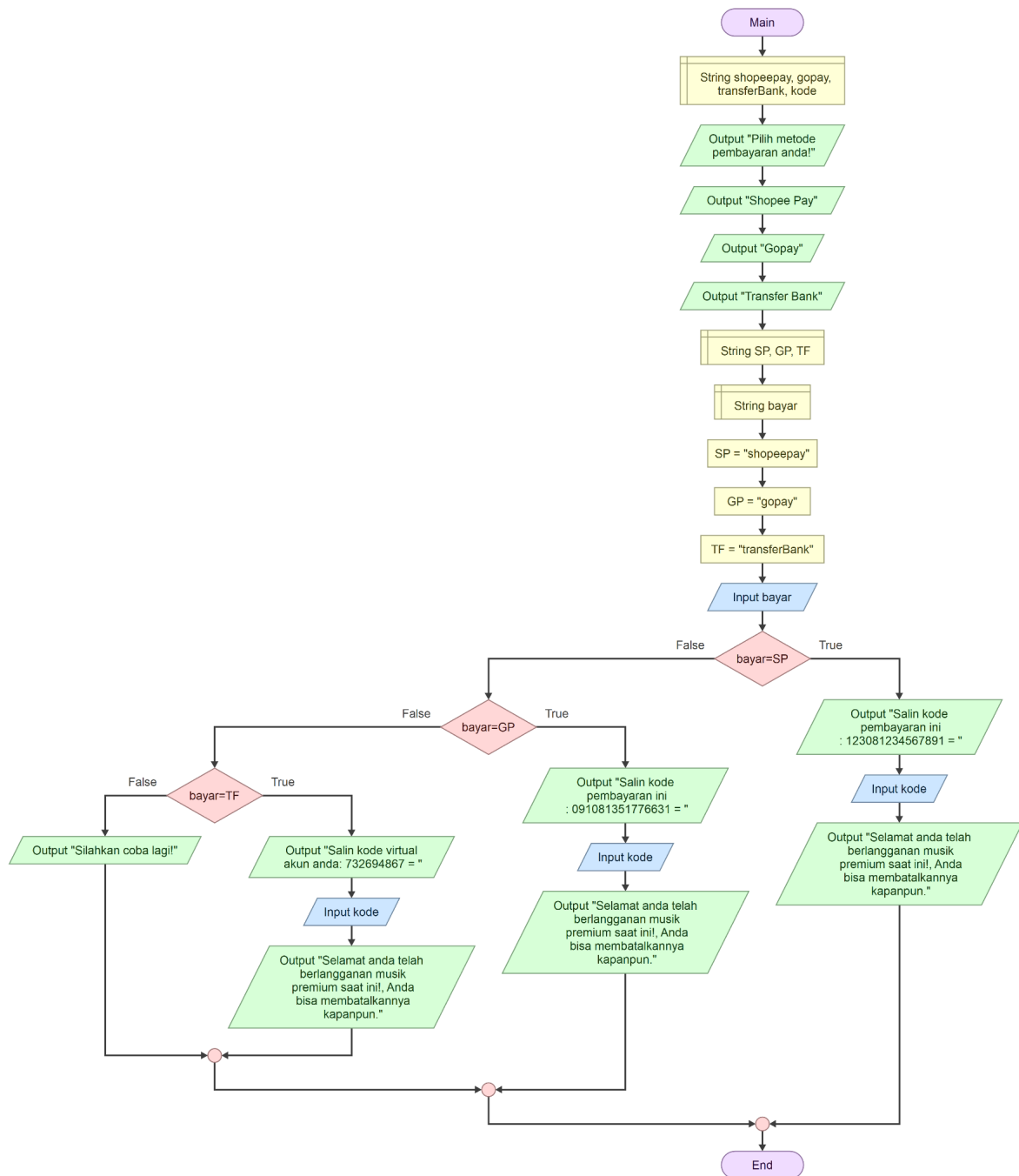
Disusun oleh:
Nayla Nur Elsyada (2509106070)
Kelas (B2'25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Gambar 1.1 Flowchart



Gambar 1.2 Flowchart

Penjelasan:

1. Mulai, Memulai program.
2. Deklarasi variabel
 - Nama, berhasilLogin (string), guna menyimpan variabel
 - bronze, silver, gold, platinum (interger), guna menyimpan pilihan paket premium
 - HLB, HLS, HLG, HLP (Real), guna menyimpan data

3. input
 - "Silahkan isi nama anda", User mengisi nama
 - "Silahkan isi NIM anda", User mengisi NIM
4. If
 - Jika benar
 - Diberi 4 pilihan (Bronze,Silver,Gold, dan Platinum) dengan feedback masing-masing
 - metode pembayaran yang diberi 3 pilihan (ShopeePay,Gopay,dan Transfer Bank), user memilih metode pembayaran
5. Else
 - Tampilkan "Silahkan coba lagi!"
6. End

2. Deskripsi Singkat Program

Program ini dibuat guna pembayaran langganan pada aplikasi musik agar user bisa mendengarkan musik sepuasnya tanpa gangguan apapun.

3. Source Code

3.1 Fitur Input Data

Fitur ini digunakan untuk mengisi data pengguna

```
# user mengisi data
nama = input("Masukan nama anda :")
nim = input("Masukan NIM anda :")
```

3.2 Fitur Deklarasi Variabel

Fitur ini digunakan untuk menyimpan data paket premium serta harganya

```
# hitung harga akun premium + biaya admin
Bronze = 1500000 + (1500000*1/100) # harga paket bronze
Silver = 1500000 + (1500000*3/100) # harga paket silver
Gold = 1500000 + (1500000*5/100) # harga paket gold
Platinum = 1500000 + (1500000*7/100) #harga paket silver
```

3.3 Fitur If - elif - else

Fitur ini digunakan untuk memberi pilihan pada user guna memilih langganan akun premium.

```

if nama == "nayla" and nim == "2509106070" :
    print ("Anda berhasil login!, pilih opsi langganan anda!")

    # pilihan opsi biaya langganan aplikasi musik
    print ("Pilih paket Premium anda dan dengarkan musik bebas iklan tanpa
batas di ponsel, speaker,dan perangkat lainnya. Bayar dengan berbagai cara,
dan anda juga bisa membatalkannya kapan saja.")
    print
    ("=====
=====")
    print ("Paket Bronze : Pembatalan kapan saja, Lagu-lagu Populer.")
    print ("Paket Silver : 1 Akun Premium, Pembatalan kapan saja, Custom
Playlist.")
    print ("Paket Gold : 1 Akun Premium, Pembatalan kapan saja, Custom
Playlist, Mode Offline. ")
    print ("Paket Platinum : 1 Akun premium, Akses semua fitur, Custom
Playlist, Mode offline, Konten Eksklusif artis.")
    opsi = input("pilih opsi langganan anda! ")

    # tampilkan harga
    if opsi == "bronze":
        print (f"Harga Paket Bronze : Rp {int(Bronze)}")
    elif opsi == "silver":
        print (f"Harga Paket Silver : Rp {int(Silver)}")
    elif opsi == "gold":
        print (f"Harga Paket Gold : Rp {int(Gold)}")
    elif opsi == "platinum":
        print (f"Harga Paket Platinum : Rp {int(Platinum)}")
    input ("Apakah anda sudah yakin ingin berlangganan opsi tersebut? ")
    print ("Berikut adalah opsi pembayaran yang bisa anda lakukan: ")
    print ("-ShopeePay")
    print ("-Gopay")
    print ("-Transfer Bank")

    bayar = input ("Pilih opsi pembayaran anda: ")
    if bayar == "shopeepay":
        input ("Salin kode pembayaran ini : 123081234567891 = ")
        print ("Selamat anda telah berlangganan musik premium saat ini!,
Anda bisa membatalkannya kapanpun.")
    elif bayar == "gopay":
        input ("Salin kode pembayaran ini : 091081351776631 = ")
        print ("Selamat anda telah berlangganan musik premium saat ini!,
Anda bisa membatalkannya kapanpun.")
    elif bayar == "transfer bank":
        input ("Salin kode virtual akun anda: 732694867 = ")
        print ("Selamat anda telah berlangganan musik premium saat ini!,
Anda bisa membatalkannya kapanpun.")
    else:

```

```
print ("Silahkan coba lagi!")
```

4. Hasil Output

```
Masukan nama anda :nayla
Masukan NIM anda :2509106070
Anda berhasil login!, pilih opsi langganan anda!
Pilih paket Premium anda dan dengarkan musik bebas iklan tanpa batas di ponsel, speaker,dan perangkat lainnya. Bayar dengan berbagai cara, dan anda juga bisa
membatalkannya kapan saja.
=====
Paket Bronze : Pembatalan kapan saja, Lagu-lagu Populer.
Paket Silver : 1 Akun Premium, Pembatalan kapan saja, Custom Playlist.
Paket Platinum : 1 Akun premium, Akses semua fitur, Custom Playlist, Mode offline, Konten Eksklusif artis.
pilih opsi langganan anda! gold
Harga Paket Gold : Rp 1575000
Apakah anda sudah yakin ingin berlangganan opsi tersebut? yakin
Berikut adalah opsi pembayaran yang bisa anda lakukan:
-ShopeePay
-Gopay
-Transfer Bank
Pilih opsi pembayaran anda: gopay
Salin kode pembayaran ini : 091081351776631 = 091081351776631
Selamat anda telah berlangganan musik premium saat ini!, Anda bisa membatalkannya kapanpun.
```

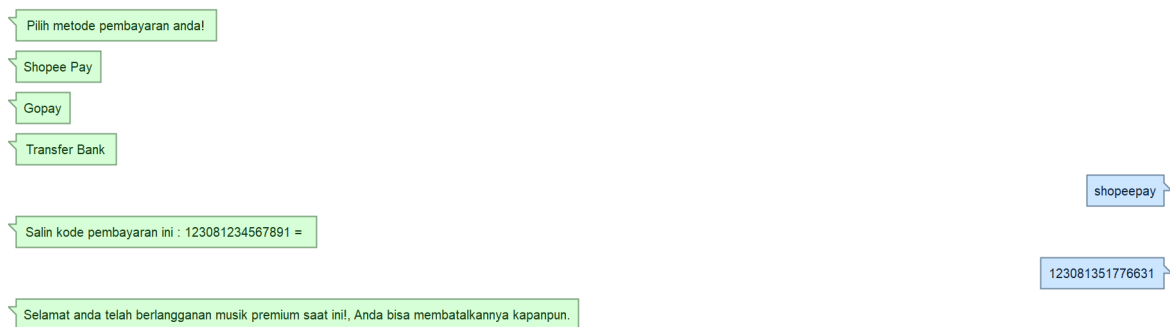
Gambar 4.1 Hasil Output Python

```
graph TD
    A[masukan nama anda:] --> B[nayla]
    B --> C[masukan nim anda:]
    C --> D[2509106070]
    D --> E[Anda berhasil login!, pilih opsi langganan anda!]
    E --> F[Pilih paket Premium anda dan dengarkan musik bebas iklan tanpa batas di ponsel, speaker,dan perangkat lainnya. Bayar dengan berbagai cara, dan anda juga bisa membatalkannya kapan saja.]
    F --> G[Paket Bronze : Pembatalan kapan saja, Lagu-lagu Populer.]
    G --> H[Paket Silver : 1 Akun Premium, Pembatalan kapan saja, Custom Playlist.]
    H --> I[Paket Gold : 1 Akun Premium, Pembatalan kapan saja, Custom Playlist, Mode Offline.]
    I --> J[ ]
    J --> K[ ]
```

Gambar 4.2 Hasil Output Flowgorithm



Gambar 4.3 Hasil Output Flowgorithm



Gambar 4.4 Hasil Output Flowgorithm

5. Langkah-langkah GIT

5.1 GIT Init

perintah dasar Git yang digunakan untuk membuat repositori Git baru atau menginisialisasi direktori yang sudah ada sebagai repositori Git. Perintah ini menciptakan sebuah direktori tersembunyi bernama `.git` di dalam proyek Anda, yang berisi semua data, konfigurasi, dan riwayat yang diperlukan oleh Git untuk melacak perubahan.

```
PS C:\Users\ASUS-GK\Documents\PRAKTIKUM_APD_B2'25> git init
Reinitialized existing Git repository in C:/Users/ASUS-GK/Documents/PRAKTIKUM_APD_B2'25/.git/
```

Gambar 5.1 git init

5.2 GIT Add

Fungsi `git add` adalah untuk memindahkan perubahan dari direktori kerja (working directory) ke area staging (staging area), yang merupakan langkah awal untuk mempersiapkan berkas agar dimasukkan ke dalam commit berikutnya dalam repositori Git.

```
PS C:\Users\ASUS-GK\Documents\PRAKTIKUM_APD_B2'25> git add .
```

Gambar 5.2 git add

5.3 GIT Commit

Fungsi `git commit` adalah untuk menyimpan "snapshot" atau rekam jejak perubahan yang telah Anda buat pada proyek Anda dan menyimpannya ke dalam riwayat repositori Git. Setiap kali Anda melakukan `commit`, Git akan menyimpan perubahan tersebut beserta informasi tentang siapa yang membuat perubahan, kapan, dan catatan singkat mengenai perubahan tersebut, menciptakan titik penyimpanan yang dapat Anda temukan kembali di masa mendatang.

```
PS C:\Users\ASUS-GK\Documents\PRAKTIKUM_APD_B2'25> git commit -m "2509106070_NAYLANURELSYADA_PT_3"
[main 3ce059a] 2509106070_NAYLANURELSYADA_PT_3
```

Gambar 5.3 git commit

5.4 GIT Remote

Fungsi `git remote` adalah perintah Git yang digunakan untuk mengelola koneksi antara repositori lokal (di komputer Anda) dan repositori jarak jauh (di server, seperti GitHub). Ini memungkinkan Anda untuk menampilkan, menambahkan, menghapus, atau mengubah URL repositori jarak jauh yang digunakan untuk berkolaborasi dengan tim, membagikan kode, dan menyinkronkan perubahan dengan repositori pusat.

```
PS C:\Users\ASUS-GK\Documents\PRAKTIKUM_APD_B2'25> git remote add origin https://github.com/elsyadaa-del/Praktikum_APD_B2-25
error: remote origin already exists.
```

Gambar 5.4 git remote

5.5 GIT Push

Fungsi `git push` adalah untuk mengunggah (mendorong) commit atau perubahan yang telah dibuat di repositori lokal Anda ke repositori jarak jauh, seperti repositori di GitHub atau GitLab. Tujuan utamanya adalah membagikan kode Anda kepada tim, menyimpannya sebagai cadangan, dan memungkinkan kolaborasi.


```
PS C:\Users\ASUS-GK\Documents\PRAKTIKUM_APD_B2'25> git push -u origin main
Enumerating objects: 22, done.
Counting objects: 100% (20/20), done.
Delta compression using up to 4 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 2.28 KiB | 233.00 KiB/s, done.
Total 12 (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/elsyadaa-del/Praktikum_APD_B2-25.git
 96cc338..3ce059a  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.5 git push