

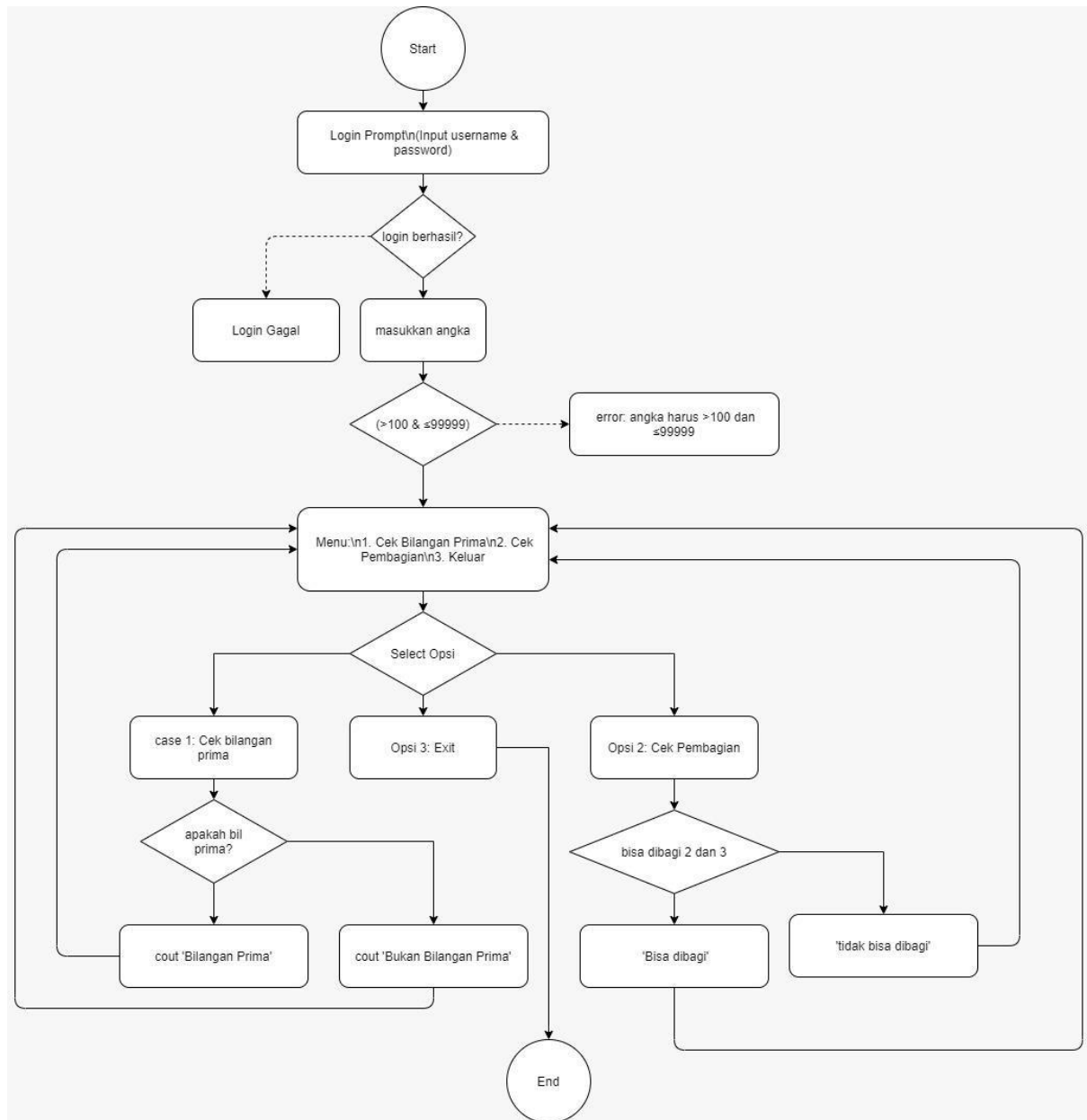
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:
Langgeng Dimas
Saputra
(2409106121)
Kelas (C2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



2. Analisis Program

2.1 Deskripsi Singkat Program

Tujuan:

Program ini dibuat untuk **mensimulasikan transaksi pembelian Unknown Cash (UC) dalam game PUBG**. Program ini memiliki dua jenis pengguna:

1. **Admin** → Dapat melihat daftar harga dan pesanan pembeli.
2. **Pembeli** → Dapat melihat harga UC, melakukan pembelian, dan melihat total pembayaran.

Program ini juga menerapkan sistem login untuk **admin** (username = "dimas", password = "121"), sedangkan **pembeli tidak perlu login**.

2.2 Penjelasan Alur & Algoritma

A. Penjelasan Alur Program

Login

- Program meminta pengguna memilih peran: **Admin** atau **User**.
- Jika memilih **Admin**, pengguna harus memasukkan **username** dan **password**.
- Jika memilih **User**, langsung masuk ke menu pembelian UC tanpa login.

Menu Admin

- Dapat melihat daftar harga UC.
- Dapat melihat pesanan yang telah dibuat oleh pembeli.
- Bisa logout untuk kembali ke menu utama.

Menu User (Pembeli)

- Bisa melihat daftar harga UC.
- Bisa melakukan pembelian UC (hanya dalam kelipatan 20 UC).
- Setelah pembelian, sistem akan menampilkan total harga dan menyimpan pesanan.
- Setelah selesai membeli, langsung logout kembali ke menu utama.

B. Algoritma

• Mulai Program

• Tampilkan menu:

- 1. Login sebagai Admin
- 2. Login sebagai User
- 3. Keluar

• Jika pengguna memilih **Admin**:

- Meminta input **username** dan **password**.
- Jika benar, masuk ke **Menu Admin**.
- Jika salah 3 kali, program akan berhenti.

• Jika memilih **User**, langsung masuk ke **Menu User**.

• Menu Admin:

- Pilih 1 → Menampilkan daftar harga UC.
- Pilih 2 → Menampilkan daftar pesanan pembeli.
- Pilih 3 → Logout kembali ke menu utama.

• Menu User:

- Pilih 1 → Menampilkan daftar harga UC.
- Pilih 2 → Memasukkan jumlah UC yang ingin dibeli.
 - Jika jumlah UC **tidak kelipatan 20**, tampilkan error.
 - Jika valid, simpan pesanan ke dalam array dan tampilkan total harga.

- Setelah selesai membeli, kembali ke menu utama.
- Pilih **3** → Logout kembali ke menu utama.
- Jika pengguna memilih **Keluar**, tampilkan pesan terima kasih dan hentikan program.
- Selesai.

3 Source Code

- Admin harus login dengan **username: "dimas"** dan **password: "121"**.
- Jika salah **3 kali**, akan kembali ke menu utama.
- User (Pembeli) bisa langsung masuk tanpa login.
- **Menu Admin**
 - Bisa melihat daftar harga UC.
 - Bisa melihat daftar pesanan pembeli.
 - Bisa logout untuk kembali ke menu utama.
- **Menu User (Pembeli)**
 - Bisa melihat daftar harga UC.
 - Bisa membeli UC dengan **ketentuan kelipatan 20**.
 - Jika jumlah UC valid, data akan disimpan ke dalam array **vector**.
 - Bisa logout untuk kembali ke menu utama.
- **CRUD Data**
 - **Create:** Pembeli membuat pesanan dan data disimpan dalam vector.
 - **Read:** Admin bisa melihat daftar pesanan pembeli.

```
bool cekPembagian(int angka) {
    return (angka % 6 == 0 && angka % 5 != 0);
}
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

- **Admin login** *Berhasil masuk setelah memasukkan username & password*
- **User login** *Langsung masuk tanpa memasukkan data*
- **User beli 40 UC** *Pesanan berhasil, total Rp20.000*
 - Admin melihat pesanan yang sudah dipesan pembeli

4.2 Hasil Output

5. Git

1. Konfigurasi User Git

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git config --global user.email "langgengdimas121@gmail.com"
```

- Perintah ini digunakan untuk mengatur alamat email yang akan digunakan dalam commit Git.
- Opsi `--global` berarti pengaturan ini berlaku untuk semua repository di komputer.

2. Inisialisasi Repository Git

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git init
Reinitialized existing Git repository in C:/Users/ASUS/Documents/praktikum-apl/.git/
```

- Perintah ini digunakan untuk menginisialisasi repository Git di dalam folder saat ini.
- Jika repository Git sudah ada, Git akan menampilkan pesan **"Reinitialized existing Git repository"**.

3. Menambahkan Remote Repository

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git remote add origin https://github.com/2409106121langgengdimassaputra/praktikum-apl.git
error: remote origin already exists.
```

- Perintah ini menghubungkan repository lokal dengan repository online di GitHub.
- **origin** adalah nama default untuk remote repository.
- Jika sebelumnya sudah ada remote dengan nama **origin**, maka akan muncul pesan **"remote origin already exists"**.

4. Menambahkan File ke Staging Area

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git add .
```

- Perintah ini digunakan untuk menambahkan semua file yang telah diubah atau ditambahkan ke dalam **staging area** sebelum dilakukan commit.

5. Membuat Commit

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git commit -m "Langgeng Posttest"
On branch master
nothing to commit, working tree clean
```

- Commit digunakan untuk menyimpan perubahan ke dalam repository lokal.
- Opsi -m "Langgeng Posttest" menambahkan pesan deskriptif tentang perubahan yang dilakukan.
- Jika tidak ada perubahan, Git akan menampilkan "**nothing to commit, working tree clean**".

6. Mengunggah ke Repository GitHub (Push)

```
ASUS@LAPTOP-7JK1V59G MINGW64 ~/Documents/praktikum-apl (master)
$ git push -u origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 446.12 KiB | 808.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/2409106121langgengdimassaputra/praktikum-apl.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

- Perintah ini mengunggah perubahan dari branch **master** di lokal ke branch **master** di remote **origin** (GitHub).
- Opsi -u (--set-upstream) menghubungkan branch lokal dengan branch di remote, sehingga ke depannya cukup menggunakan git push tanpa parameter tambahan.
- Pesan "**Please complete authentication in your browser...**" muncul karena GitHub membutuhkan autentikasi sebelum bisa mengunggah perubahan.
- Setelah autentikasi berhasil, proses *push* berjalan hingga selesai.
- Menunjukkan bahwa file berhasil dikompresi dan dikirim ke GitHub.