

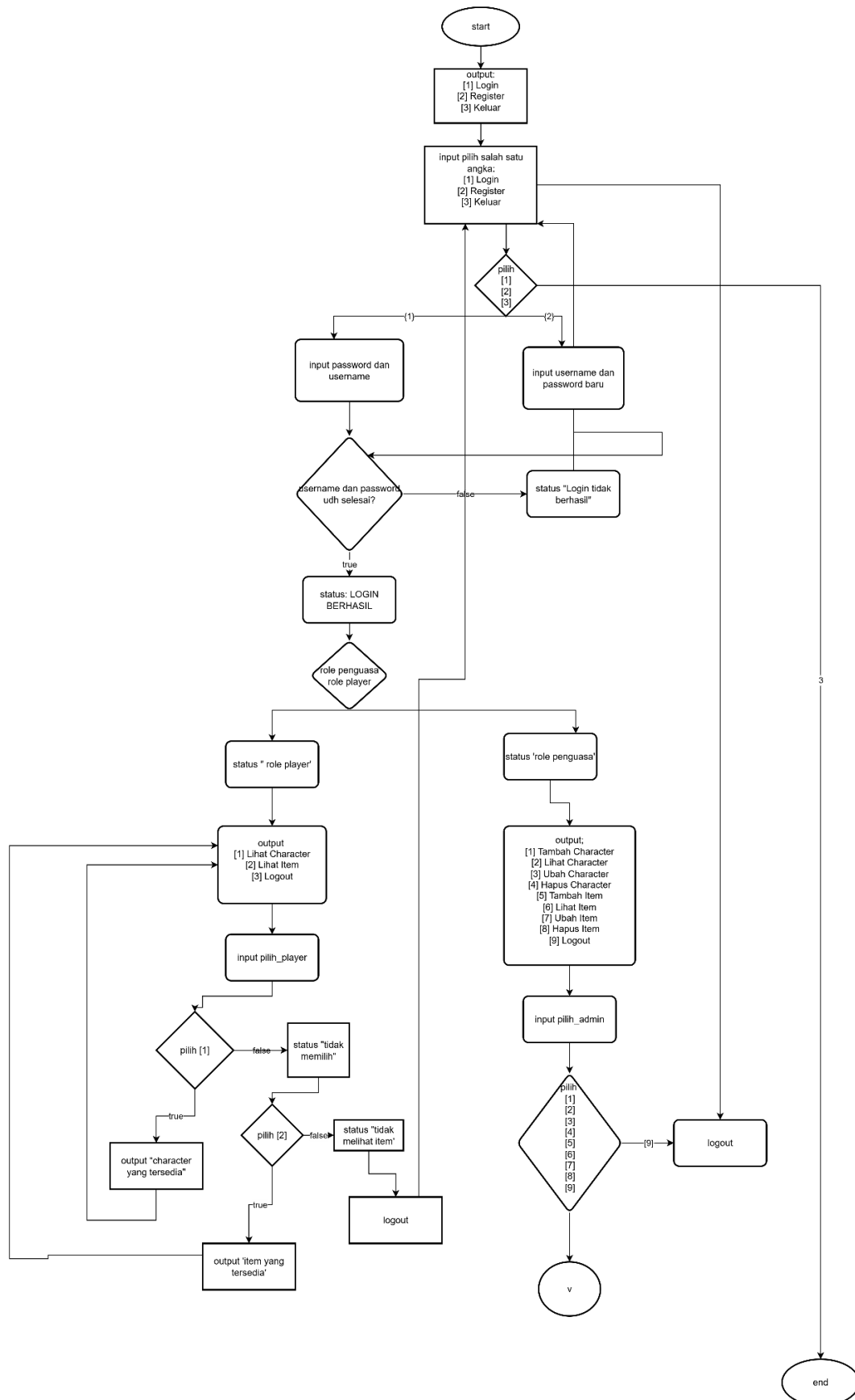
LAPORAN PRAKTIKUM
POSTTEST 5
ALGORITMA PEMROGRAMAN DASAR

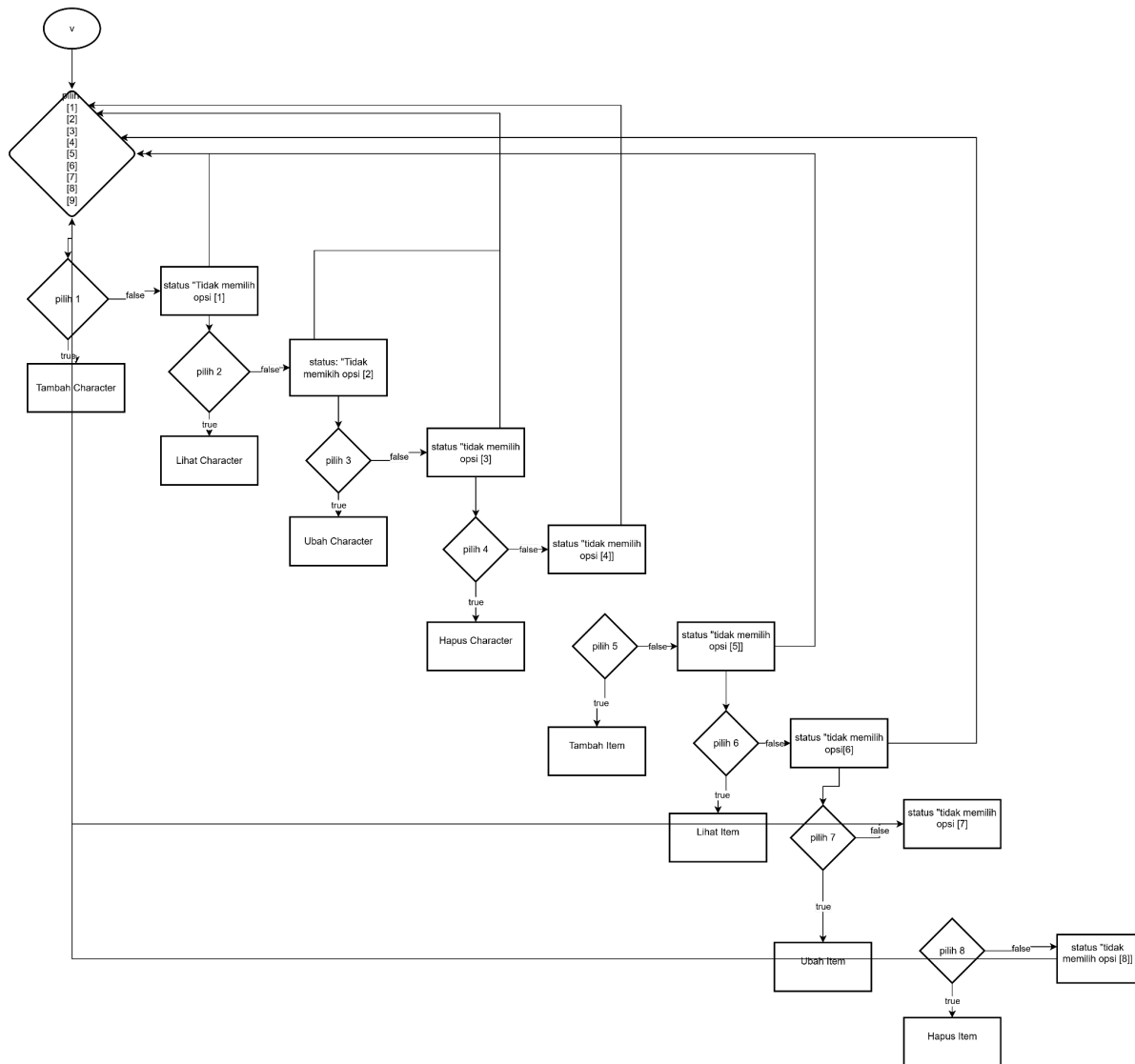


Disusun oleh:
Luvita Khairana Salwa
2509106041
A'25

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart





2. Deskripsi Singkat Program

berfungsi untuk:

1. Untuk Admin:

- Mengelola Database Character - Tambah, lihat, ubah, dan hapus data karakter Genshin
- Mengelola Database Item - Tambah, lihat, ubah, dan hapus data item/barang
- Full Control - Punya akses penuh untuk mengatur semua data

2. Untuk Player (User Biasa):

- Lihat Character - Cek daftar karakter yang tersedia
- Lihat Item - Cek daftar item yang ada beserta harga dan efeknya

- *Limited Access* - Cuma bisa lihat aja, gak bisa ubah data

3. Fitur Umum:

- *Login & Register* - Sistem autentikasi user
- *Role-Based Access* - Ada 2 role (*Admin & Player*) dengan hak akses berbeda
- *Database Sederhana* - Nyimpan data user, character, dan item pakai list

Contoh Penggunaan Real:

Misalnya kamu bikin komunitas Genshin atau website fan-made:

- Admin bisa update data character baru pas ada update game
- Member bisa cek info character dan item tanpa bisa ngubah-ngubah
- Bisa jadi referensi database untuk pemain Genshin Impact

3. Source Code

```
import os

# format user → [username, password, role]

users = [["akito yamada", "akane", "penguasa"]]

characters = []

items = []

# Program Utama

while True:

    os.system("cls" if os.name == "nt" else "clear")

    print("=== SISTEM GENSHIN IMPACT ===")

    print("[1] Login")

    print("[2] Register")

    print("[3] Keluar")

    print("=====")

    pilih = input("Pilih menu diatas: ")
```

```

# === LOGIN ===

if pilih == "1":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== LOGIN ===")

    uname = input("Username: ")

    pw = input("Password: ")

    login_berhasil = False

    role = ""

    for u in users:

        if u[0] == uname and u[1] == pw:

            login_berhasil = True

            role = u[2]

            break

    if not login_berhasil:

        print("Login gagal! Username atau password salah.")

        input("Tekan Enter...")

        continue

    os.system("cls" if os.name == "nt" else "clear")

```

```

print(f"Selamat datang, {uname.title()}!")

print(f"Role: {role.upper()}")

input("Tekan Enter...")

# === MENU ADMIN ===

if role == "penguasa":

    while True:

        os.system("cls" if os.name == "nt" else "clear")

        print("    欢迎来到游戏管理菜单 !!!! ")

        print("=== MENU PENGUASA GAME ===")

        print("[1] Tambah Character")

        print("[2] Lihat Character")

        print("[3] Ubah Character")

        print("[4] Hapus Character")

        print("[5] Tambah Item")

        print("[6] Lihat Item")

        print("[7] Ubah Item")

        print("[8] Hapus Item")

        print("[9] Logout")

        print("=====")

        pilih_admin = input("Pilih menu: ")

# [1] Tambah Character

```

```

        if pilih_admin == "1":

            os.system("cls" if os.name == "nt" else "clear")

            print("=== TAMBAH CHARACTER ===")

            nama = input("Nama Karakter: ")

            elemen = input("Elemen (Pyro/Hydro/Anemo/Electro/Cryo/Geo/Dendro): ")

            rarity = input("Rarity (4★/5★): ")

            senjata = input("Senjata (Sword/Bow/Polearm/Catalyst/Claymore): ")

            region = input("Region: ")

            role = input("Role (DPS/Sub-DPS/Support/Healer): ")

            skill = input("Ultimate Skill: ")

            characters.append([nama, elemen, rarity, senjata, region, role, skill])

            print(f"Character {nama} berhasil ditambahkan!")

# [2] Lihat Character

        elif pilih_admin == "2":

            os.system("cls" if os.name == "nt" else "clear")

            print("=== DAFTAR CHARACTER ===")

            if len(characters) == 0:

                print("Belum ada character.")

            else:

                for c in characters:

```

```

        print(f"- {c[0]} | {c[1]} | {c[2]} | {c[3]}")

# Ubah Character

elif pilih_admin == "3":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== UBAH CHARACTER ===")

    nama_cari = input("Nama character yang ingin diubah: ")

    ditemukan = False

    for c in characters:

        if c[0].lower() == nama_cari.lower():

            print(f"Ketemu: {c[0]}")

            c[3] = input("Ubah senjata jadi: ")

            print("Data berhasil diubah!")

            ditemukan = True

            break

    if not ditemukan:

        print("Character tidak ditemukan.")

# Hapus Character

elif pilih_admin == "4":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== HAPUS CHARACTER ===")

```



```

nama_cari = input("Nama character yang ingin dihapus: ")

ditemukan = False

for c in characters:

    if c[0].lower() == nama_cari.lower():

        konfirm = input(f"Hapus {c[0]}? (y/n): ")

        if konfirm == "y":

            characters.remove(c)

            print("Character berhasil dihapus!")

        else:

            print("Batal hapus.")

        ditemukan = True

        break

if not ditemukan:

    print("Character tidak ditemukan.")

# [5] Tambah Item

elif pilih_admin == "5":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== TAMBAH ITEM ===")

    id_item = input("ID Item: ")

    nama_item = input("Nama Item: ")

    kategori = input("Kategori: ")

```

```

        rarity = input("Rarity: ")

        efek = input("Efek: ")

        harga = input("Harga: ")

        items.append([id_item, nama_item, kategori, rarity, efek,
harga])

        print(f"Item {nama_item} berhasil ditambahkan!")

# [6] Lihat Item

elif pilih_admin == "6":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== DAFTAR ITEM ===")

    if len(items) == 0:

        print("Belum ada item.")

    else:

        for i in items:

            print(f"- {i[0]} | {i[1]} | {i[2]} | {i[3]} |
{i[4]} | Rp{i[5]}")

# [7] Ubah Item

elif pilih_admin == "7":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== UBAH ITEM ===")

    id_cari = input("ID item yang ingin diubah: ")

```

```

        ditemukan = False

        for i in items:

            if i[0].lower() == id_cari.lower():

                print(f"Ketemu: {i[1]}")

                i[4] = input("Ubah efek jadi: ")

                print("Efek berhasil diubah!")

                ditemukan = True

                break

        if not ditemukan:

            print("Item tidak ditemukan.")

# [8] Hapus Item

elif pilih_admin == "8":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== HAPUS ITEM ===")

    id_cari = input("ID item yang ingin dihapus: ")

    ditemukan = False

    for i in items:

        if i[0].lower() == id_cari.lower():

            konfirm = input(f"Hapus {i[1]}? (y/n): ")

            if konfirm == "y":

                items.remove(i)

```

```

        print("Item berhasil dihapus!")

    else:

        print("Batal hapus.")

        ditemukan = True

        break

    if not ditemukan:

        print("Item tidak ditemukan.")

# [9] Logout

elif pilih_admin == "9":

    print("Logout berhasil.")

    input("Tekan Enter...")

    break

else:

    print("Pilihan tidak valid.")

    input("Tekan Enter...")

# === MENU PLAYER ===

else:

    while True:

        os.system("cls" if os.name == "nt" else "clear")

```

```

print("=== MENU PLAYER ===")

print("[1] Lihat Character")

print("[2] Lihat Item")

print("[3] Logout")


pilih_user = input("Pilih menu: ")


# [1] Lihat Character

if pilih_user == "1":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== DAFTAR CHARACTER ===")

    if len(characters) == 0:

        print("Belum ada character.")

    else:

        for c in characters:

            print(f"- {c[0]} | {c[1]} | {c[2]} | {c[3]}")

# [2] Lihat Item

elif pilih_user == "2":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== DAFTAR ITEM ===")

    if len(items) == 0:

        print("Belum ada item.")

    else:

        for i in items:

```

```

        print(f"- {i[0]} | {i[1]} | {i[2]} | {i[3]} | {i[4]} |
Rp{i[5]}")

    # [3] Logout

    elif pilih_user == "3":

        print("Logout berhasil.")

        input("Tekan Enter...")

        break

    else:

        print("Pilihan tidak valid.")

        input("Tekan Enter...")

# === REGISTER ===

elif pilih == "2":

    os.system("cls" if os.name == "nt" else "clear")

    print("=== REGISTER ===")

    new_username = input("Username baru: ")

    sudah_ada = False

    for u in users:

        if u[0] == new_username:

            sudah_ada = True

            break

    if sudah_ada:

        print("Username sudah dipakai!")

    else:

        new_password = input("Password: ")

```

```

        users.append([new_uname, new_password, "player"])

        print("Akun berhasil dibuat!")

        input("Tekan Enter...")

# === KELUAR ===

elif pilih == "3":

    print("Keluar dari program.")

    break

else:

    print("Pilihan tidak valid.")

    input("Tekan Enter...")

```

4. HASIL OUTPUT

- output awal

```

=== SISTEM GENSIN IMPACT ===
[1] Login
[2] Register
[3] Keluar
=====
Pilih menu: 

```

- output login sebagai admin

```

    欢迎来到游戏管理菜单 !!!!
=== MENU PENGUASA GAME ===
[1] Tambah Character
[2] Lihat Character
[3] Ubah Character
[4] Hapus Character
[5] Tambah Item
[6] Lihat Item
[7] Ubah Item
[8] Hapus Item
[9] Logout
=====
Pilih menu: 

```

- output sebagai pengguna baru harus register dulu

```

=== REGISTER ===
Username baru: miaw
Password: 20

```

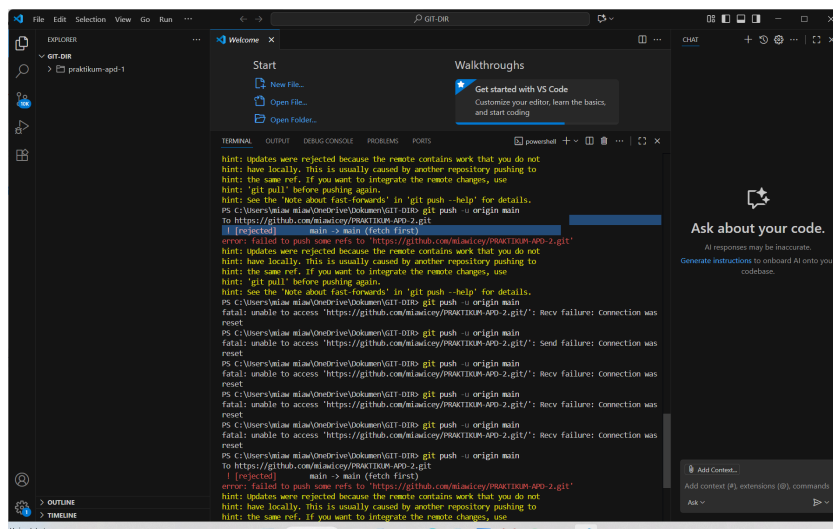
- output menu player sehabis register langsung login sesuai username dan password yang di register

```

=== MENU PLAYER ===
[1] Lihat Character
[2] Lihat Item
[3] Logout
Pilih menu: 

```

5. Langkah-langkah GIT



5.1 GIT Init

git init → Perintah untuk membuat repository Git baru di folder lokal.

- Fungsinya: Menginisialisasi folder agar bisa digunakan Git (ada folder **.git** di dalamnya). Contoh: “git init”

5.2 GIT Add

git add → Perintah untuk menambahkan file atau perubahan file ke staging area.

- Fungsinya: Menandai file mana yang siap untuk disimpan (commit).
Contoh: “git add index.html”(menambahkan satu file) “git add .” (menambahkan semua file)

5.3 GIT Commit

git commit → Perintah untuk menyimpan perubahan dari staging area ke repository lokal.

- Fungsinya: Mencatat snapshot dari kode dengan pesan deskriptif.
Contoh: "git commit -m" (Menambahkan halaman index)

5.4 GIT Remote

git remote → Perintah untuk menghubungkan repository lokal dengan repository di server (misalnya GitHub/GitLab).

- Fungsinya: Agar repository lokal bisa disinkronkan dengan repository online.
Contoh: "git remote add origin" <https://github.com/user/nama-repo.git>

5.5 GIT Push

git push → Perintah untuk mengirim commit dari repository lokal ke repository remote (GitHub, GitLab, dll).

- Fungsinya: Mengupload kode yang sudah di-commit ke server.
Contoh: "git push -u origin main"