

LAPORAN PRAKTIKUM
POSTTEST (8)
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:

Nama (2509106050)

Kelas (B1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

```

graph TD
    Start([start]) --> Init[variabel Dictionary hero_meta  
variabel pengguna baru =0  
variabel username_admin=admin  
variabel password_admin=admin123]
    Init --> Def[def cek_login  
def cari_hero  
def tampilkan_statistik  
def tampilkan_hero  
def daftar_user]
    Def --> Menu[1.daftar pengguna baru  
2.login sebagai user  
3.login sebagai admin  
4.keluar]
    Menu --> P1{if pilihan == 1}
    P1 -- true --> DUser[daftar_user()]
    P1 -- false --> P2{if pilihan == 2}
    P2 -- true --> CLogin[cek_login()]
    CLogin --> LBS[/login berhasil/]
    LBS --> Menu
    P2 -- false --> P3{if pilihan == 3}
    P3 -- true --> Input[/input username dan password admin/]
    Input --> CLoginAdmin[cek_login]
    CLoginAdmin --> Menu
    P3 -- false --> P4{if pilihan == 4}
    P4 --> Break1[break]
    Break1 --> Thank[/terima kasih telah menggunakan program kami/]
    Thank --> End([end])

    Menu --> MMenu[/masuk ke program menu user/]
    MMenu --> MMenu1[1.lihat daftar hero  
2.cari hero  
3.statistik  
4.logout]
    MMenu1 --> P1_1{if pilihan == 1}
    P1_1 -- true --> TampilkanHero1[tampilkan_hero()]
    P1_1 -- false --> P2_1{if pilihan == 2}
    P2_1 -- true --> CariHero1[cari_hero()]
    P2_1 -- false --> P3_1{if pilihan == 3}
    P3_1 -- true --> TampilkanStatistik1[tampilkan_statistik()]
    P3_1 -- false --> P4_1{if pilihan == 4}
    P4_1 -- true --> Break2[break]
    Break2 --> End

    Menu --> MMenu2[/masuk ke menu admin/]
    MMenu2 --> MMenu3[1.lihat daftar hero  
2.tambah hero  
3.hapus hero  
4.cari hero  
5.statistik  
6.logout]
    MMenu3 --> P1_2{if pilihan == 1}
    P1_2 -- true --> TampilkanHero2[tampilkan_hero()]
    P1_2 -- false --> P2_2{if pilihan == 2}
    P2_2 -- true --> ProgramTambah[program tambah hero]
    P2_2 -- false --> P3_2{if pilihan == 3}
    P3_2 -- true --> ProgramHapus[program hapus hero]
    P3_2 -- false --> P4_2{if pilihan == 4}
    P4_2 -- true --> CariHero2[cari_hero()]
    P4_2 -- false --> P5_2{if pilihan == 5}
    P5_2 -- true --> TampilkanStatistik2[tampilkan_statistik()]
    P5_2 -- false --> P6_2{if pilihan == 6}
    P6_2 -- true --> Break3[break]
    Break3 --> End
  
```

2. Deskripsi Program

Pertama dimulai dengan membuat variable dictionary list hero dan role masing masing. selanjutnya membuat variable def yaitu:

1. def cek_login yaitu untuk mengecek anda login sebagai apa.
2. def cari_hero yaitu untuk program cari hero yang inginkan
3. def hitung_hero untuk menghitung ada berapa hero yang ada di list
4. def tampilkan_statistik untuk melihat ada berapa jumlah hero yang ada di list dengan memanggil def hitung_hero menampilkan jumlah hero yang ada di role masing masing
5. def daftar_user untuk program daftar pengguna baru atau user baru

Selanjutnya masuk ke program jika memilih nomor 1 maka akan masuk ke program daftar pengguna baru def_daftar_user yaitu anda masukkan username baru dan password minimal 6 karakter baru dan data nya akan di simpan di variabel dictionary pengguna baru yang sudah di buat tadi. jika sudah membuat akun maka bisa login sebagai user dan akan masuk ke program login sebagai user, di dalam program user anda bisa melihat list hero hero meta dan tidak bisa mengubah nya dan juga anda bisa melihat statistic program, dan juga bisa mencari hero yang ada di variable dictionary. jika memilih nomor 3 yaitu login sebagai Admin anda akan masuk ke program admin yaitu anda bisa menambahkan atau menghapus hero hero yang ada di list dictionary hero meta dan juga anda akan mendapatkan pilihan yang sama yang ada di menu user. dan jika anda memilih nomor 4 anda akan keluar dari program.

3. deskripsi singkat program

Program ini di buat untuk mengetahui apa saja list hero meta ter-update yang ada di game mobile legend. fungsi utamanya yaitu agar player mobile legend bias cepat beradaptasi dengan meta yang ter-update di game.

4. Source Code

```
hero_meta = {
    "Mage": ["Yve", "Zhuxin", "Pharsa", "Zetian", "Novaria", "Kadita"],
    "Tank/Support": ["Floryn", "Mathilda", "Kalea", "Angela", "Badang", "Groock",
"Gatot"],
    "Assassin": ["Hayabusa", "Fanny", "Yi_sun_shin", "Lancelot", "Fredrin"],
    "Fighter": ["Alice", "Cici", "Arlot", "Phoveus", "Gloo"],
    "Marksman": ["Granger", "Nathan", "Wanwan", "Obsidia"]
}
```

```

pengguna_baru = {}
from data import hero_meta, pengguna_baru

def cek_login(usser, pw, tipe):
    if tipe == "user":
        return usser in pengguna_baru and pengguna_baru[usser] == pw
    elif tipe == "admin":
        return usser == "admin" and pw == "admin123"
    return False

def daftar_usser():
    print("\n=== DAFTAR USER BARU ===")
    user_baru = input("Username: ")

    if user_baru == "":
        print("Username tidak boleh kosong!")
        return

    if user_baru in pengguna_baru:
        print("Username sudah ada!")
        return

    pw_baru = input("Password: ")

    if len(pw_baru) < 6:
        print("Password minimal 6 karakter!")
        return

    pengguna_baru[user_baru] = pw_baru
    print("Berhasil daftar!")

def tampilkan_hero():
    print("\n" + "="*50)
    print("DAFTAR HERO META")
    print("="*50)

    nomor = 1
    for role in hero_meta:
        print(f"\n{nomor}. {role}:")
        i = 1
        for hero in hero_meta[role]:
            print(f"    {i}. {hero}")
            i += 1
        nomor += 1
    print("="*50)

def cari_hero(nama):
    for role in hero_meta:
        if nama in hero_meta[role]:
            return True, role
    return False, None

```

```

def tampilkan_statistik():
    print("\n" + "="*50)
    print("STATISTIK HERO")
    print("="*50)

    total = 0
    for role in hero_meta:
        jumlah = len(hero_meta[role])
        print(f"{role}: {jumlah} hero")
        total += jumlah

    print(f"\nTotal Hero: {total}")
    print(f"Total Role: {len(hero_meta)}")
    print("="*50)

def tambah_hero():
    tampilkan_hero()
    print("\n=== TAMBAH HERO ===")

    role_list = list(hero_meta.keys())
    i = 1
    for role in role_list:
        print(f"{i}. {role}")
        i += 1

    pilih = input(f"\nPilih role (1-{len(role_list)}): ")

    if not pilih.isdigit():
        print("Harus angka!")
        return

    index = int(pilih) - 1
    if index < 0 or index >= len(role_list):
        print("Pilihan salah!")
        return

    role_dipilih = role_list[index]
    nama = input("Nama hero: ")

    if nama == "":
        print("Nama tidak boleh kosong!")
        return

    ada, role_lama = cari_hero(nama)
    if ada:
        print(f"Hero sudah ada di {role_lama}!")
        return

    hero_meta[role_dipilih].append(nama)
    print(f"Hero '{nama}' berhasil ditambahkan!")

```

```

def hapus_hero():
    tampilkan_hero()
    print("\n=== HAPUS HERO ===")

    role_list = list(hero_meta.keys())
    i = 1
    for role in role_list:
        print(f"{i}. {role}")
        i += 1

    pilih = input(f"\nPilih role (1-{len(role_list)}): ")

    if not pilih.isdigit():
        print("Harus angka!")
        return

    index = int(pilih) - 1
    if index < 0 or index >= len(role_list):
        print("Pilihan salah!")
        return

    role_dipilih = role_list[index]

    if len(hero_meta[role_dipilih]) == 0:
        print("Tidak ada hero!")
        return

    print(f"\nHero di {role_dipilih}:")
    i = 1
    for hero in hero_meta[role_dipilih]:
        print(f"{i}. {hero}")
        i += 1

    pilih_hero = input(f"\nPilih hero (1-{len(hero_meta[role_dipilih])}): ")

    if not pilih_hero.isdigit():
        print("Harus angka!")
        return

    index_hero = int(pilih_hero) - 1
    if index_hero < 0 or index_hero >= len(hero_meta[role_dipilih]):
        print("Pilihan salah!")
        return

    hero_hapus = hero_meta[role_dipilih].pop(index_hero)
    print(f"Hero '{hero_hapus}' berhasil dihapus!")

import os
from fungsi import *

def main():

```

```

os.system('clear')

while True:
    print("\n" + "="*50)
    print("SISTEM MANAGEMENT HERO META")
    print("="*50)
    print("1. Daftar pengguna baru")
    print("2. Login sebagai user")
    print("3. Login sebagai admin")
    print("4. Keluar")
    print("="*50)

    pilih = input("Pilihan (1-4): ")

    if pilih == "1":
        daftar_usser()
        input("\nTekan enter...")

    elif pilih == "2":
        print("\n=== LOGIN USER ===")
        user = input("Username: ")
        pw = input("Password: ")

        if cek_login(user, pw, "user"):
            print("\nLogin berhasil!")
            input("Tekan enter...")

            while True:
                print("\n" + "="*50)
                print("MENU USER")
                print("="*50)
                print("1. Lihat daftar hero")
                print("2. Cari hero")
                print("3. Statistik")
                print("4. Logout")
                print("="*50)

                pilih_user = input("Pilih (1-4): ")

                if pilih_user == "1":
                    tampilkan_hero()
                    input("\nTekan enter...")

                elif pilih_user == "2":
                    nama = input("\nNama hero: ")
                    ketemu, role = cari_hero(nama)
                    if ketemu:
                        print(f"Hero '{nama}' ada di role {role}")
                    else:
                        print("Hero tidak ditemukan!")
                    input("\nTekan enter...")

```

```

        elif pilih_user == "3":
            tampilkan_statistik()
            input("\nTekan enter...")

        elif pilih_user == "4":
            print("\nLogout berhasil!")
            break

        else:
            print("Pilihan salah!")
    else:
        print("\nLogin gagal!")
        input("\nTekan enter...")

elif pilih == "3":
    print("\n=== LOGIN ADMIN ===")
    user = input("Username: ")
    pw = input("Password: ")

    if cek_login(user, pw, "admin"):
        print("\nSelamat datang, Admin!")
        input("Tekan enter...")

    while True:
        print("\n" + "="*50)
        print("MENU ADMIN")
        print("="*50)
        print("1. Lihat daftar hero")
        print("2. Tambah hero")
        print("3. Hapus hero")
        print("4. Cari hero")
        print("5. Statistik")
        print("6. Logout")
        print("="*50)

        pilih_admin = input("Pilih (1-6): ")

        if pilih_admin == "1":
            tampilkan_hero()
            input("\nTekan enter...")

        elif pilih_admin == "2":
            tambah_hero()
            input("\nTekan enter...")

        elif pilih_admin == "3":
            hapus_hero()
            input("\nTekan enter...")

        elif pilih_admin == "4":

```

```

        nama = input("\nNama hero: ")
        ketemu, role = cari_hero(nama)
        if ketemu:
            print(f"Hero '{nama}' ada di role {role}")
        else:
            print("Hero tidak ditemukan!")
            input("\nTekan enter...")

    elif pilih_admin == "5":
        tampilkan_statistik()
        input("\nTekan enter...")

    elif pilih_admin == "6":
        print("\nLogout berhasil!")
        break

    else:
        print("Pilihan salah!")
else:
    print("\nLogin gagal!")
    input("\nTekan enter...")

elif pilih == "4":
    print("\n" + "="*50)
    print("Terima kasih!")
    print("="*50)
    break

else:
    print("Pilihan salah!")

main()

```

4.output program

```
=====
MENU USER
=====
```

1. Lihat hero
 2. Cari hero
 3. Statistik
 4. Logout
- ```
=====
```

```
Pilih (1-4): 2
```

```
Nama hero: Yve
Hero 'Yve' ada di role Mage
```

```
Tekan enter unutm melanjutkan...█
```

```
=====
SISTEM MANAGEMENT HERO META
=====
```

- ```
1. Daftar pengguna baru
2. Login sebagai user
3. Login sebagai admin
4. Keluar
=====
```

```
Pilihan (1-4): 1
```

```
=====
== DAFTAR USER BARU ==
=====
```

```
Username baru: nopala
Password baru: nopala
User 'nopala' berhasil terdaftar!
=====
```

```
=====
SISTEM MANAGEMENT HERO META
=====
```

- ```
1. Daftar pengguna baru
2. Login sebagai user
3. Login sebagai admin
4. Keluar
=====
```

```
Pilihan (1-4): 2
```

```
=====
== LOGIN USER ==
=====
```

```
Username: nopala
Password: nopala
```

```
Login berhasil!
Tekan enter untuk melanjutkan...
=====
```

```
=====
MENU USER
=====
```

- ```
1. Lihat hero
2. Cari hero
3. Statistik
4. Logout
=====
```

```
Pilih (1-4): █
```

```
=====
STATISTIK HERO
=====
```

```
Total Hero: 27
```

```
Total Role: 5
```

```
Jumlah per Role:
```

```
  Mage: 6 hero
```

```
  Tank/Support: 7 hero
```

```
  Assassin: 5 hero
```

```
  Fighter: 5 hero
```

```
  Marksman: 4 hero
=====
```

```
Tekan enter untuk melanjutkan...█
```

```
=====
SISTEM MANAGEMENT HERO META
=====
```

1. Daftar pengguna baru
2. Login sebagai user
3. Login sebagai admin
4. Keluar

```
=====
Pilihan (1-4): 3
```

```
=== LOGIN ADMIN ===
```

```
Username: admin
```

```
Password: admin123
```

```
Selamat datang, Admin!
```

```
Tekan enter untuk melanjutkan...
```

```
=====
MENU ADMIN
=====
```

1. Lihat hero
2. Tambah hero
3. Hapus hero
4. Cari hero
5. Statistik
6. Logout

```
=====
Pilih (1-6): █
```

=== TAMBAH HERO BARU ===

Pilih role (1-5): 5

Masukkan nama hero baru untuk Marksman: joy

Hero 'joy' berhasil ditambahkan ke role Marksman!

Tekan enter untuk kembali

=== HAPUS HERO ===

Pilih role (1-5): 5

Daftar hero di Marksman:

1. Granger
2. Nathan
3. Wanwan
4. Obsidia
5. joy

Pilih hero yang ingin dihapus (1-5): 5

Hero 'joy' berhasil dihapus dari role Marksman!

Tekan enter untuk kembali

5. langkah langkah GIT

```
● macbook@macbooks-MacBook-Air praktikum-apd % git commit -m "apd pt 7"
[main 9cbb859] apd pt 7
 4 files changed, 1 insertion(+), 1 deletion(-)
 create mode 100644 post-test/post-test-apd-7/[Template] Template Laporan APD 3+++ docx
 create mode 100644 post-test/post-test-apd-7/~$emplate] Template Laporan APD 3+++ docx
● macbook@macbooks-MacBook-Air praktikum-apd % git push -u origin main
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (8/8), 197.62 KiB | 7.90 MiB/s, done.
Total 8 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/Nopal07A/praktikum-apd.git
 eb41cf2..9cbb859 main -> main
branch 'main' set up to track 'origin/main'.
```

1. GIT Add

git add berfungsi sebagai menyimpan perubahan yang kita lakukan pada code, dan untuk memindahkan perubahan pada berkas dari working directory ke area staging

2. GIT Commit

git commit berfungsi sebagai perubahan yang telah disiapkan (ditambahkan ke staging area)

3. GIT Push

git push berfungsi sebagai untuk mengirim commit atau perubahan lokal ke repository Git