

LAPORAN PRAKTIKUM
POSTTEST 8
ALGORITMA PEMROGRAMAN DASAR

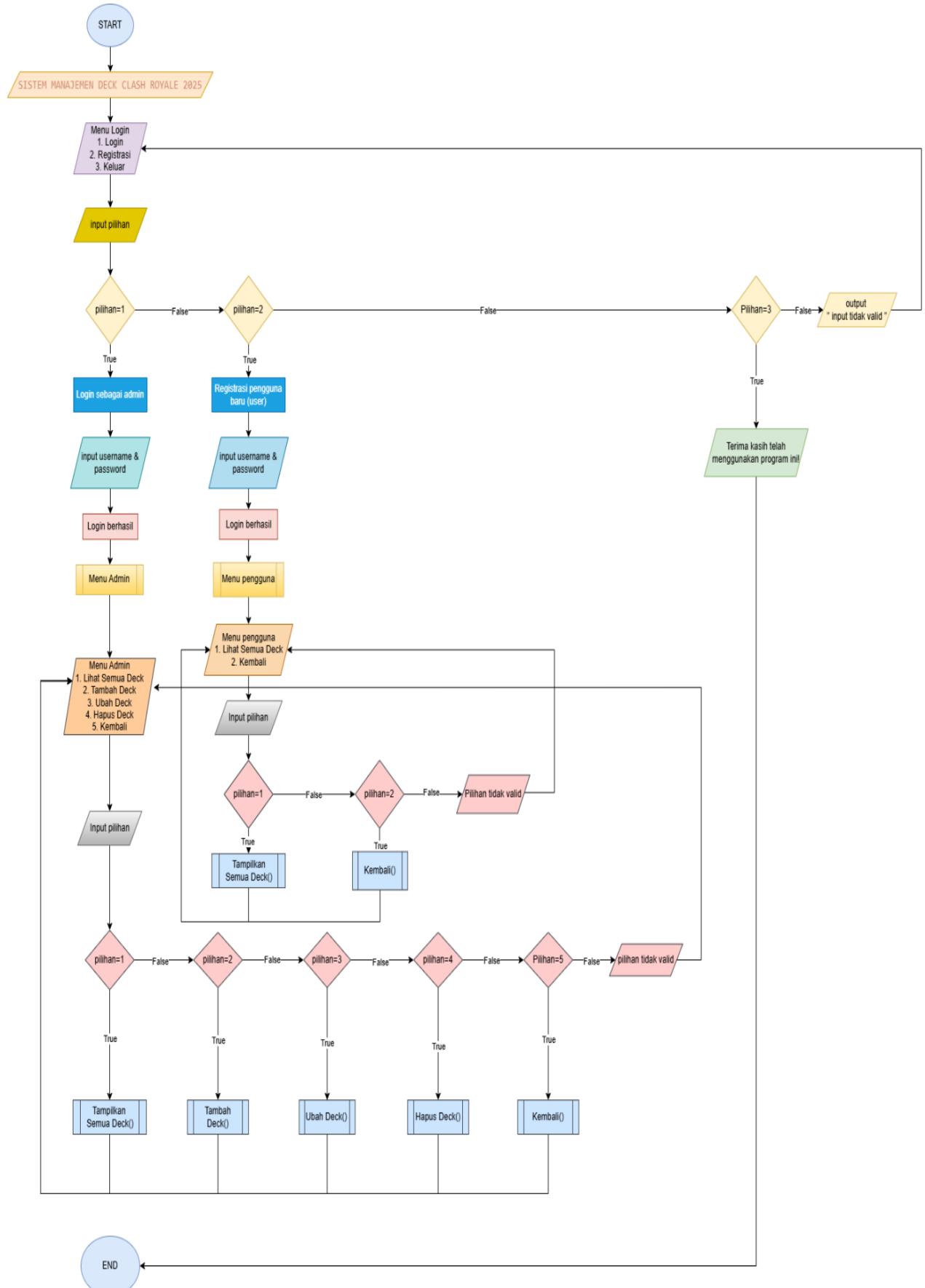


Disusun oleh:
Navtaly Juman (2509106109)
Kelas (C1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA

2025

1. Flowchart



Gambar 1.1 Flowchart

2.Deskripsi Singkat Program

Program ini mengelola deck Clash Royale dengan fitur login dan register. Pengguna dengan peran Admin dapat menambah, mengubah, menghapus, dan melihat data deck, sedangkan User hanya bisa melihat daftar deck yang tersedia. Setiap deck berisi informasi nama deck, pemain, strategi, dan daftar kartu. Program ini menggunakan library PrettyTable agar data tampil dalam bentuk tabel yang rapi di terminal serta dibuat secara modular dengan beberapa file terpisah agar lebih mudah dipahami dan dikelola.

3. Source Code

A. fungsi login dan register (auth.py)



```
1 users = {  
2     "Navtaly Juman": {"password": "2509106109", "role": "Admin"}  
3 }  
4  
5 def login():  
6     username = input("Username: ")  
7     password = input("Password: ")  
8     if username in users and users[username]["password"] == password:  
9         print(f"\nSelamat datang {username}!\n")  
10        return users[username]["role"], username  
11    else:  
12        print("Login gagal! Username atau password salah.")  
13        return None, None  
14  
15 def register():  
16     username = input("Masukkan Username: ")  
17     password = input("Masukkan Password: ")  
18     if username in users:  
19         print("Username sudah digunakan!")  
20     else:  
21         users[username] = {"password": password, "role": "User"}  
22         print("Akun berhasil dibuat!")
```

Gambar 3.1 auth.py

B. fungsi menampilkan, menambah, mengubah, dan menghapus deck (deck.py)

```
1  from prettytable import PrettyTable
2
3  decks = {
4      1: {
5          "nama": "Miner Control",
6          "pemain": "Morten",
7          "strategi": "Control",
8          "kartu": ["Miner", "Poison", "Wall Breaker", "Skeletons", "Bats", "Valkyrie", "Bomb Tower", "Log"]
9      },
10     2: {
11         "nama": "X-Bow 2.9 Cycle",
12         "pemain": "Aragon",
13         "strategi": "Siege",
14         "kartu": ["X-Bow", "Tesla", "Ice Spirit", "Skeletons", "Ice Golem", "Fireball", "Log", "Archers"]
15     },
16     3: {
17         "nama": "LavaLoon",
18         "pemain": "Surgical Goblin",
19         "strategi": "Air Beatdown",
20         "kartu": ["Lava Hound", "Balloon", "Mega Minion", "Tombstone", "Minions", "Fireball", "Zap", "Miner"]
21     },
22     4: {
23         "nama": "Royal Giant Lightning",
24         "pemain": "Mohamed Light",
25         "strategi": "Beatdown",
26         "kartu": ["Royal Giant", "Fisherman", "Lightning", "Mother Witch", "Electro Spirit", "Skeletons", "Hunter", "Log"]
27     },
28     5: {
29         "nama": "Bridge Spam Pekka",
30         "pemain": "Ruben",
31         "strategi": "Control",
32         "kartu": ["Bandit", "Battle Ram", "Royal Ghost", "Dark Prince", "Magic Archer", "Pekka", "Poison", "Zap"]
33     }
34 }
35
36 from prettytable import PrettyTable
37
38 def tampilan_deck():
39     if not decks:
40         print("Belum ada deck yang ditambahkan.")
41         return
42
43     table = PrettyTable(["ID", "Nama Deck", "Pemain", "Strategi", "Kartu"])
44     for key, deck in decks.items():
45         table.add_row([key, deck["nama"], deck["pemain"], deck["strategi"], ", ".join(deck["kartu"])])
46     table.align = "l"
47     table.max_width = 50
48     print(table)
49
50 def tambah_deck(jumlah_kartu):
51     nama = input("Nama Deck: ")
52     pemain = input("Nama Pemain: ")
53     strategi = input("Jenis Strategi: ")
54
55     kartu = []
56     print(f"Masukkan {jumlah_kartu} kartu:")
57     for i in range(jumlah_kartu):
58         kartu.append(input(f"Kartu ke-{i+1}: "))
59
60     new_id = len(decks) + 1
61     decks[new_id] = {"nama": nama, "pemain": pemain, "strategi": strategi, "kartu": kartu}
62     print("Deck berhasil ditambahkan!")
63
```

Gambar 3.2 deck.py

```

1  def ubah_deck(id_ubah):
2      if id_ubah not in decks:
3          print("ID deck tidak ditemukan.")
4          return
5      deck = decks[id_ubah]
6      print(f"\nMengubah deck: {deck['nama']}")

7
8      deck["nama"] = input(f"Nama baru ({deck['nama']}): ") or deck["nama"]
9      deck["emain"] = input(f"emain baru ({deck['emain']}): ") or deck["emain"]
10     deck["strategi"] = input(f"Strategi baru ({deck['strategi']}): ") or deck["strategi"]
11
12     print("Masukkan kartu baru (kosongkan jika tidak ingin mengubah):")
13     kartu_baru = []
14     for i in range(8):
15         k = input(f"Kartu ke-{i+1} ({deck['kartu'][i]}): ")
16         kartu_baru.append(k if k else deck["kartu"][i])
17     deck["kartu"] = kartu_baru
18     print("Deck berhasil diubah!")

19
20 def hapus_deck(id_hapus):
21     if id_hapus in decks:
22         del decks[id_hapus]
23         print("Deck berhasil dihapus!")
24     else:
25         print("ID deck tidak ditemukan.")

26
27 def menu_admin():
28     while True:
29         print("\n==== MENU ADMIN ===")
30         print("1. Lihat Semua Deck")
31         print("2. Tambah Deck")
32         print("3. Ubah Deck")
33         print("4. Hapus Deck")
34         print("5. Kembali ke Menu Utama")
35
36         pilihan = input("Pilih menu: ")
37         if pilihan == "1":
38             tampilan_deck()
39         elif pilihan == "2":
40             tambah_deck(8)
41         elif pilihan == "3":
42             try:
43                 id_ubah = int(input("Masukkan ID deck yang ingin diubah: "))
44                 ubah_deck(id_ubah)
45             except ValueError:
46                 print("ID harus berupa angka!")
47         elif pilihan == "4":
48             try:
49                 id_hapus = int(input("Masukkan ID deck yang ingin dihapus: "))
50                 hapus_deck(id_hapus)
51             except ValueError:
52                 print("ID harus berupa angka!")
53         elif pilihan == "5":
54             break
55         else:
56             print("Pilihan tidak valid!")

57
58 def menu_user():
59     while True:
60         print("\n==== MENU USER ===")
61         print("1. Lihat Semua Deck")
62         print("2. Kembali ke Menu Utama")
63         pilihan = input("Pilih menu: ")
64         if pilihan == "1":
65             tampilan_deck()
66         elif pilihan == "2":
67             print("Logout berhasil.")
68             break
69         else:
70             print("Pilihan tidak valid!")

```

Gambar 3.2 deck.py (lanjutan)

C. alur utama program menu login, pilihan menu (main.py)

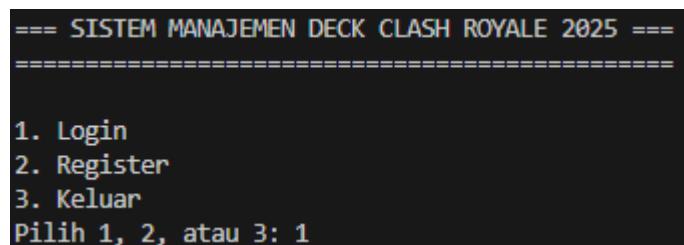


```
● ● ●

1  from auth import login, register
2  from deck import menu_admin, menu_user
3
4  def main():
5      print("\n==== SISTEM MANAJEMEN DECK CLASH ROYALE 2025 ===")
6      print("=====")
7
8      while True:
9          print("\n1. Login")
10         print("2. Register")
11         print("3. Keluar")
12         pilihan = input("Pilih 1, 2, atau 3: ")
13
14         if pilihan == "1":
15             role, nama_user = login()
16             if role == "Admin":
17                 menu_admin()
18             elif role == "User":
19                 menu_user()
20         elif pilihan == "2":
21             register()
22         elif pilihan == "3":
23             print("Terima kasih telah menggunakan program ini!")
24             break
25         else:
26             print("Pilihan tidak valid!")
27
28     if __name__ == "__main__":
29         main()
```

Gambar 3.3 main.py

4. Hasil Output



```
==== SISTEM MANAJEMEN DECK CLASH ROYALE 2025 ===
=====
1. Login
2. Register
3. Keluar
Pilih 1, 2, atau 3: 1
```

Gambar 4.1 Output Awal

```
Pilih 1, 2, atau 3: 1
Username: Navtaly Juman
Password: 2509106109

Selamat datang Navtaly Juman!
```

Gambar 4.2 Output Login Admin

```
==== MENU ADMIN ====
1. Lihat Semua Deck
2. Tambah Deck
3. Ubah Deck
4. Hapus Deck
5. Kembali ke Menu Utama
Pilih menu: 1
```

Gambar 4.3 Output Menu Admin

Pilih menu: 1				
ID	Nama Deck	Pemain	Strategi	Kartu
1	Miner Control	Morten	Control	Miner, Poison, Wall Breaker, Skeletons, Bats, Valkyrie, Bomb Tower, Log
2	X-Bow 2.9 Cycle	Aragon	Siege	X-Bow, Tesla, Ice Spirit, Skeletons, Ice Golem, Fireball, Log, Archers
3	LavaLoon	Surgical Goblin	Air Beatdown	Lava Hound, Balloon, Mega Minion, Tombstone, Minions, Fireball, Zap, Miner
4	Royal Giant Lightning	Mohamed Light	Beatdown	Royal Giant, Fisherman, Lightning, Mother Witch, Electro Spirit, Skeletons, Hunter, Log
5	Bridge Spam Pekka	Ruben	Control	Bandit, Battle Ram, Royal Ghost, Dark Prince, Magic Archer, Pekka, Poison, Zap

Gambar 4.4 Output Lihat Semua Deck

```
Pilih menu: 2
Nama Deck: Auto Win
Nama Pemain: No Name
Jenis Strategi: 100% WIN
Masukkan 8 kartu:
Kartu ke-1: Pekka
Kartu ke-2: Fireball
Kartu ke-3: Goblin
Kartu ke-4: Tesla
Kartu ke-5: Miner
Kartu ke-6: Skeletons
Kartu ke-7: Royal Giant
Kartu ke-8: Wizzart
Deck berhasil ditambahkan!
```

Gambar 4.5 Output Tambah Deck

```
Pilih menu: 3
Masukkan ID deck yang ingin diubah: 6
Nama Deck baru: Pasti Kalah
Nama Pemain baru: Malu
Jenis Strategi baru: Nanggis
Masukkan 8 kartu baru:
Kartu ke-1: Goblin
Kartu ke-2: Archer
Kartu ke-3: Mini Pekka
Kartu ke-4: Ballon
Kartu ke-5: Minions
Kartu ke-6: Ice Golem
Kartu ke-7: Log
Kartu ke-8: Skelatons
Deck berhasil diubah!
```

Gambar 4.6 Output Ubah Deck

```
Pilih menu: 4
Masukkan ID deck yang ingin dihapus: 6
Deck berhasil dihapus!
```

Gambar 4.7 Output Hapus Deck

```
1. Login
2. Register
3. Keluar
Pilih 1, 2, atau 3: 2
Masukkan Username: Upin & Ipin
Masukkan Password: 22
Akun berhasil dibuat!
```

Gambar 4.8 Output Register User

```
1. Login
2. Register
3. Keluar
Pilih 1, 2, atau 3: 1
Username: Upin & Ipin
Password: 22

Selamat datang Upin & Ipin!
```

Gambar 4.9 Output Login User

```
==== MENU USER ====
1. Lihat Semua Deck
2. Kembali ke Menu Utama
Pilih menu: 1
```

Gambar 4.10 Output Menu User

Pilih menu: 1				
ID	Nama Deck	Pemain	Strategi	Kartu
1	Miner Control	Morten	Control	Miner, Poison, Wall Breaker, Skeletons, Bats, Valkyrie, Bomb Tower, Log
2	X-Bow 2.9 Cycle	Aragon	Siege	X-Bow, Tesla, Ice Spirit, Skeletons, Ice Golem, Fireball, Log, Archers
3	LavaLoon	Surgical Goblin	Air Beatdown	Lava Hound, Balloon, Mega Minion, Tombstone, Minions, Fireball, Zap, Miner
4	Royal Giant Lightning	Mohamed Light	Beatdown	Royal Giant, Fisherman, Lightning, Mother Witch, Electro Spirit, Skeletons, Hunter, Log
5	Bridge Spam Pekka	Ruben	Control	Bandit, Battle Ram, Royal Ghost, Dark Prince, Magic Archer, Pekka, Poison, Zap

Gambar 4.11 Output Lihat Semua Deck

```

1. Login
2. Register
3. Keluar
Pilih 1, 2, atau 3: 3
Terima kasih telah menggunakan program ini!

```

Gambar 4.12 Output Keluar

3. Langkah-langkah GIT

5.1 GIT Add

```
PS C:\Users\LENOVO\Documents\praktikum-apd> git add .
```

Gambar 5.1 Git Add

Git Add . ini digunakan untuk menambahkan semua file yang ada di folder saat ini ke staging area.

5.2 GIT Commit

```
PS C:\Users\LENOVO\Documents\praktikum-apd> git commit -m "post-test-apd-8"
[main 5778ced] post-test-apd-8
 9 files changed, 247 insertions(+), 6 deletions(-)
 create mode 100644 kelas/pertemuan 8/pertemuan 8.py
 create mode 100644 post-test/post-test-apd-7/2509106109-NavtalyJuman-PT-7.pdf
 create mode 100644 post-test/post-test-apd-8/Flowchart Navtaly.png
 create mode 100644 post-test/post-test-apd-8/__pycache__/_auth.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/__pycache__/_deck.cpython-313.pyc
 create mode 100644 post-test/post-test-apd-8/auth.py
 create mode 100644 post-test/post-test-apd-8/deck.py
 create mode 100644 post-test/post-test-apd-8/main.py
```

Gambar 5.2 Git Commit

Git commit –m “post-test-apd-8” ini digunakan untuk menyimpan file yang ada di staging area ke dalam repository lokal, dan –m “post-test-apd-8” untuk menambahkan komentar, pada tahap ini filemasih berada di repository lokal komputer belum terupload ke GitHub.

5.3 GIT Push

```
PS C:\Users\LENOVO\Documents\praktikum-apd> git push -u origin main
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 2 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (16/16), 125.44 KiB | 1.99 MiB/s, done.
Total 16 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/naftalijuman/praktikum-apd.git
  3aea444..9821f3b  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 Git Push

Git push ini dipakai untuk mengupload commit dari repository lokal ke repository GitHub di branch main.