

LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN DASAR

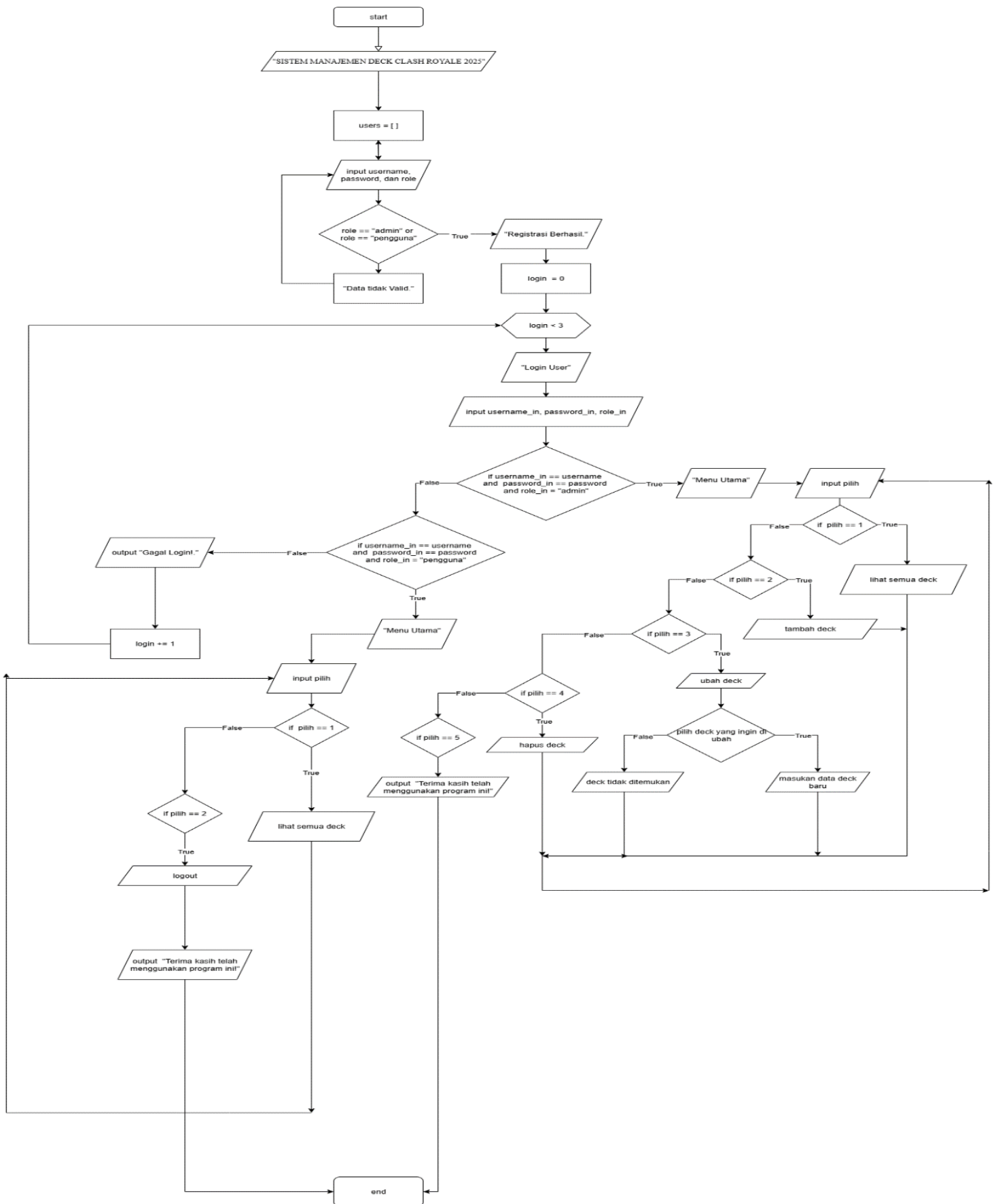


Disusun oleh:
Navtaly Juman (2509106109)
Kelas (C1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA

2025

1.Flowchart



Gambar 1.1 Flowchart

2.Deskripsi Singkat Program

Program ini adalah sistem manajemen deck Clash Royale yang menggunakan dictionary untuk menyimpan data deck dan akun pengguna. Program dapat melakukan operasi CRUD (Create, Read, Update, Delete), memiliki fitur login dan register, serta mendukung dua jenis pengguna yaitu admin dan user. Admin dapat menambah, melihat, mengubah, dan menghapus deck, sedangkan user hanya bisa melihat daftar deck yang tersedia.

3. Source Code

A. Dictionary Awal

```
1  users = {
2      "Navtaly Juman": {"password": "2509106109", "role": "Admin"}
3  }
4
5  decks = {
6      1: {
7          "nama": "Miner Control",
8          "pemain": "Morten",
9          "strategi": "Control",
10         "kartu": ["Miner", "Poison", "Wall Breaker", "Skeletons", "Bats", "Valkyrie", "Bomb Tower", "Log"]
11     },
12     2: {
13         "nama": "X-Bow 2.9 Cycle",
14         "pemain": "Aragon",
15         "strategi": "Siege",
16         "kartu": ["X-Bow", "Tesla", "Ice Spirit", "Skeletons", "Ice Golem", "Fireball", "Log", "Archers"]
17     },
18     3: {
19         "nama": "LavaLoon",
20         "pemain": "Surgical Goblin",
21         "strategi": "Air Beatdown",
22         "kartu": ["Lava Hound", "Balloon", "Mega Minion", "Tombstone", "Minions", "Fireball", "Zap", "Miner"]
23     },
24     4: {
25         "nama": "Royal Giant Lightning",
26         "pemain": "Mohamed Light",
27         "strategi": "Beatdown",
28         "kartu": ["Royal Giant", "Fisherman", "Lightning", "Mother Witch", "Electro Spirit", "Skeletons", "Hunter", "Log"]
29     },
30     5: {
31         "nama": "Bridge Spam Pekka",
32         "pemain": "Ruben",
33         "strategi": "Control",
34         "kartu": ["Bandit", "Battle Ram", "Royal Ghost", "Dark Prince", "Magic Archer", "Pekka", "Poison", "Zap"]
35     }
36 }
```

B. Menu Utama (Login, Register, Keluar)

```
1  print("=== SISTEM MANAJEMEN DECK CLASH ROYALE 2025 ===")
2
3  while True:
4      print("1. Login")
5      print("2. Register")
6      print("3. Keluar")
7      pilih = input("Pilih 1, 2, atau 3: ")
```

C. Login dan Role

```
1  if pilih == "1":
2      username = input("Username: ")
3      password = input("Password: ")
4
5      if username in users and users[username]["password"] == password:
6          print("Selamat datang ya", username)
7
```

D. Menu Admin

```
1  if users[username]["role"] == "Admin":
2      while True:
3          print("=== MENU KHUSUS ADMIN ===")
4          print("1. Lihat Semua Deck")
5          print("2. Tambah Deck")
6          print("3. Ubah Deck")
7          print("4. Hapus Deck")
8          print("5. Logout")
9          pilih_admin = input("Pilih menu: ")
10         if pilih_admin == "1":
11             if len(decks) == 0:
12                 print("Belum ada deck yang ditambahkan.")
```

E. Fitur Admin (CRUD Deck)

```
1  else:
2      for key in decks:
3          deck = decks[key]
4          print(f"ID: {key}")
5          print(f>Nama Deck: {deck['nama']}")
6          print(f>Nama Pemain: {deck['pemain']}")
7          print(f>Strategi: {deck['strategi']}")
8          print(f>Kartu: {deck['kartu']}")
9
10     elif pilih_admin == "2":
11         nama = input("Nama Deck: ")
12         pemain = input("Nama Pemain: ")
13         strategi = input("Jenis Strategi: ")
14         kartu = []
15         print("Masukkan 8 kartu:")
16         for i in range(8):
17             kartu_baru = input("Kartu ke-" + str(i+1) + ": ")
18             kartu.append(kartu_baru)
19         new_id = len(decks) + 1
20         decks[new_id] = {"nama": nama, "pemain": pemain, "strategi": strategi, "kartu": kartu}
21         print("Deck berhasil ditambahkan!")
22
23     elif pilih_admin == "3":
24         ubah = int(input("Masukkan ID deck yang ingin diubah: "))
25         if ubah in decks:
26             nama = input("Nama Deck baru: ")
27             pemain = input("Nama Pemain baru: ")
28             strategi = input("Jenis Strategi baru: ")
29             kartu = []
30             print("Masukkan 8 kartu baru:")
31             for i in range(8):
32                 kartu_baru = input("Kartu ke-" + str(i+1) + ": ")
33                 kartu.append(kartu_baru)
34             decks[ubah].update({
35                 "nama": nama,
36                 "pemain": pemain,
37                 "strategi": strategi,
38                 "kartu": kartu
39             })
40             print("Deck berhasil diubah!")
41         else:
42             print("ID deck tidak ditemukan.")
43
44     elif pilih_admin == "4":
45         hapus = int(input("Masukkan ID deck yang ingin dihapus: "))
46         if hapus in decks:
47             konfirmasi = input("Yakin ingin menghapus? (ya/tidak): ")
48             if konfirmasi == "ya":
49                 del decks[hapus]
50                 print("Deck berhasil dihapus!")
51             else:
52                 print("ID tidak ditemukan.")
53
54     elif pilih_admin == "5":
55         break
56     else:
57         print("Pilihan tidak valid!")
58
```

F. Menu User

```
1  else:
2      while True:
3          print("=== MENU USER ===")
4          print("1. Lihat Semua Deck")
5          print("2. Logout")
6          pilih_user = input("Pilih menu: ")
7
8          if pilih_user == "1":
9              if len(decks) == 0:
10                 print("Belum ada deck yang ditambahkan.")
11             else:
12                 for key in decks:
13                     deck = decks[key]
14                     print(f"ID: {key}")
15                     print(f>Nama Deck: {deck['nama']}")
16                     print(f>Pemain: {deck['pemain']}")
17                     print(f>Strategi: {deck['strategi']}")
18                     print(f>Kartu: {deck['kartu']}")
19             elif pilih_user == "2":
20                 break
21             else:
22                 print("Pilihan tidak valid!")
23
24     else:
25         print("Login gagal! Username atau password salah.")
```

G. Register & Keluar

```
1  elif pilih == "2":
2      username = input("Masukkan Username: ")
3      password = input("Masukkan Password: ")
4      if username in users:
5          print("Username sudah digunakan!")
6      else:
7          users[username] = {"password": password, "role": "User"}
8          print("Akun berhasil dibuat!")
9
10     elif pilih == "3":
11         print("Terima kasih sudah menggunakan program ini!")
12         break
13
14     else:
15         print("Pilihan tidak ada!")
16
```

4. Hasil Output

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

Gambar 1.2 Output Awal

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

Gambar 1.3 Output Login Admin

```
=== MENU KHUSUS ADMIN ===
1. Lihat Semua Deck
2. Tambah Deck
3. Ubah Deck
4. Hapus Deck
5. Logout
Pilih menu: █
```

Gambar 1.4 Output Menu Admin

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

Gambar 1.5 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: Wizzard
Deck berhasil ditambahkan!

```

Gambar 1.6 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 1.7 Output Ubah Deck

```

Pilih menu: 4
Masukkan ID deck yang ingin dihapus: 6
Yakin ingin menghapus? (ya/tidak): ya
Deck berhasil dihapus!

```

Gambar 1.8 Output Hapus Deck

```

Pilih 1, 2, atau 3: 2
Masukkan Username: Mama ku
Masukkan Password: Love you
Akun berhasil dibuat!

```

Gambar 1.9 Output Register User

```

Pilih 1, 2, atau 3: 1
Username: Mama ku
Password: Love you
Selamat datang ya Mama ku
=== MENU USER ===
1. Lihat Semua Deck
2. Logout
Pilih menu:

```

Gambar 1.10 Output Register User


```

=== MENU USER ===
1. Lihat Semua Deck
2. Logout
Pilih menu: 2
1. Login
2. Register
3. Keluar
Pilih 1, 2, atau 3: 3
Terima kasih sudah menggunakan program ini!

```

Gambar 1.11 Output Selesai

3. Langkah-langkah GIT

5.1 GIT Add

```

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

```

Gambar 2.1 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-6"
[main 7152890] post-test-apd-6
3 files changed, 285 insertions(+)
create mode 100644 kelas/pertemuan 6/pertemuan6.py
create mode 100644 post-test/post-test-apd-6/2509106109-NavtalyJuman-Flowchart.drawio.png
create mode 100644 post-test/post-test-apd-6/2509106109-NavtalyJuman-PT-6.py

```

Gambar 2.2 Git Commit

Git commit -m "post-test-apd-6" ini digunakan untuk menyimpan file yang ada di staging area ke dalam repository lokal, dan -m "post-test-apd-6" untuk menambahkan komentar, pada tahap ini file masih 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: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 2 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 127.34 KiB | 2.96 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/naftalijuman/praktikum-apd.git
 13738c5..7152890  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 2.3 Git Push

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