

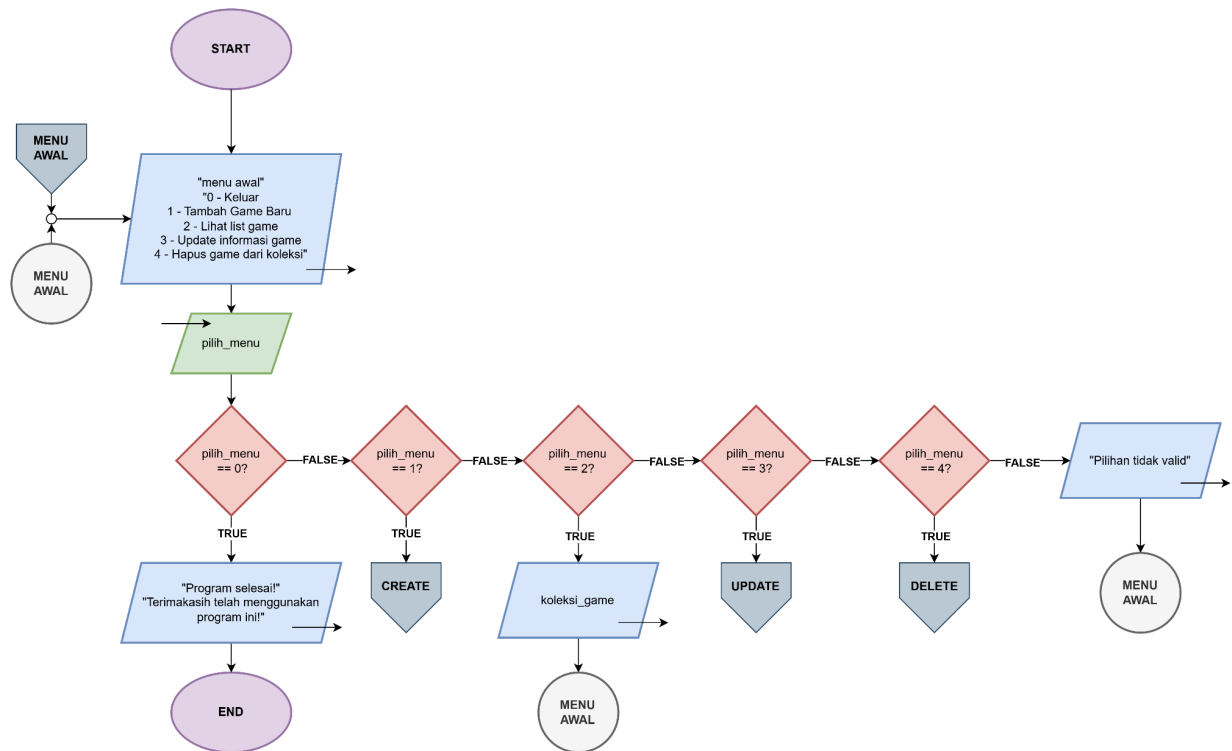
LAPORAN PRAKTIKUM
POSTTEST 5
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Yoga Ananda Prasetya (2509106017)
Kelas (A1'25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

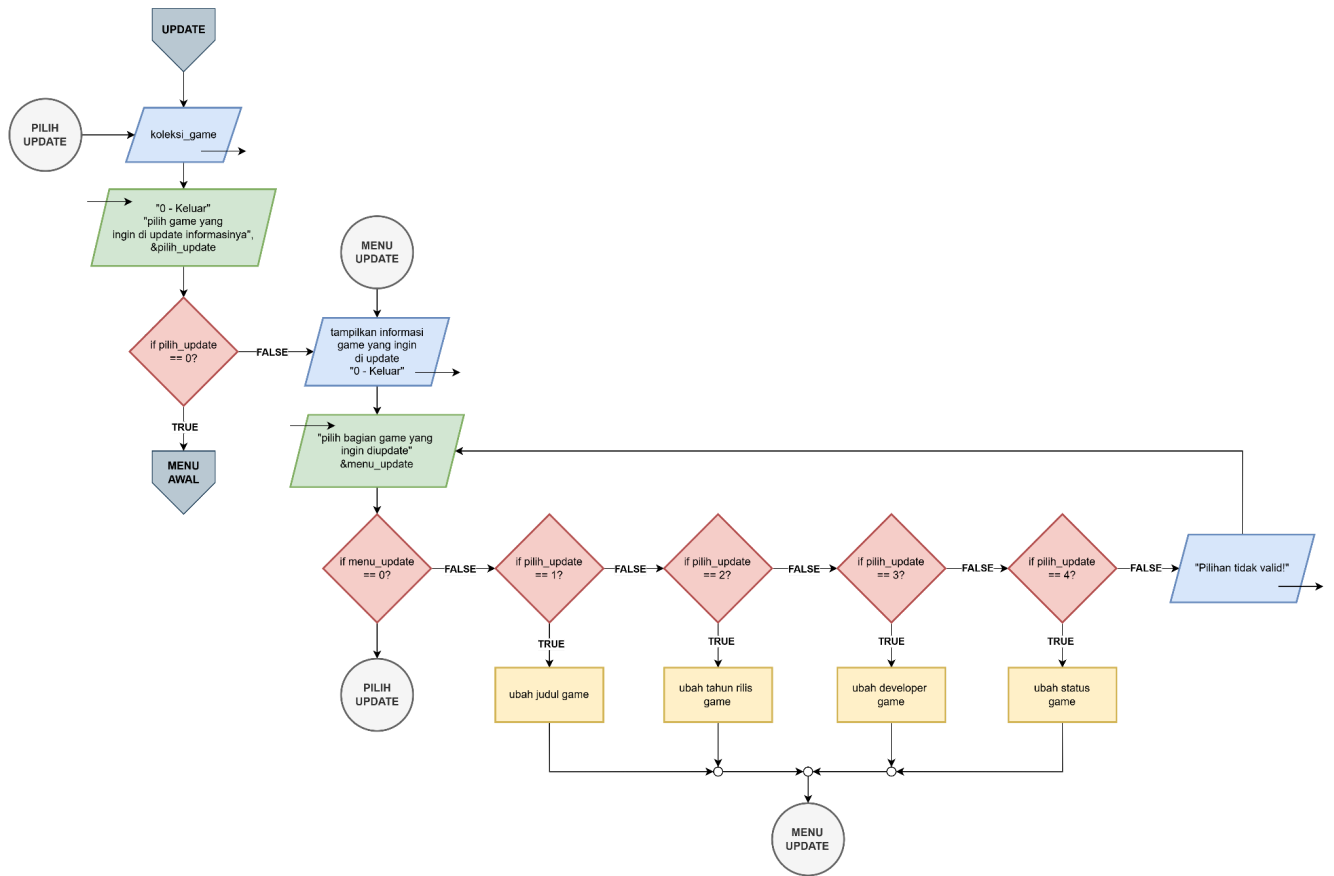
1. Flowchart



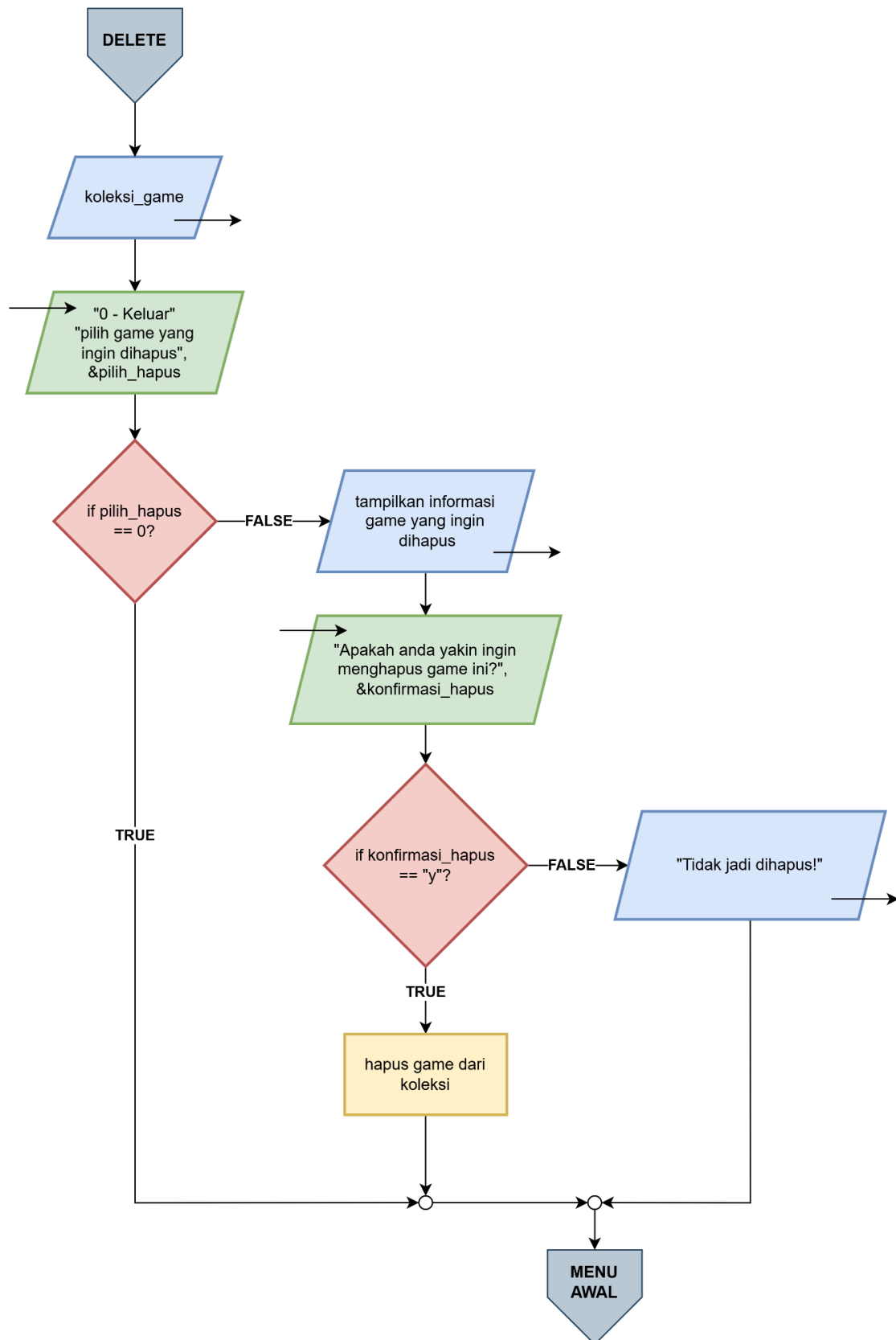
Gambar 1 Flowchart – Menu Awal



Gambar 2 Flowchart – Menu Create



Gambar 3 Flowchart – Menu Update



Gambar 4 Flowchart – Menu Delete

2. Deskripsi Singkat Program

Program ini merupakan program yang menerapkan prinsip CRUD (*Create*, *Read*, *Update*, dan *Delete*), yang dimana prinsip CRUD ini diterapkan untuk membuat sebuah sistem direktori sederhana yang dapat menyimpan informasi game yang dimiliki oleh seseorang. Berikut ini merupakan implementasi dari setiap prinsip CRUD pada sistem direktori penyimpanan game.

PRINSIP CRUD	IMPLEMENTASI PRINSIP CRUD
<i>Create</i>	: Pengguna dapat menambahkan game yang diinginkan ke dalam direktori. Disini ada 4 data yang harus dimasukkan pengguna ketika ingin menambah game baru ke direktori. 4 Data tersebut yaitu sebagai berikut : <ol style="list-style-type: none">1. Nama game2. Tahun rilis game3. Pengembang/<i>Developer</i> game4. Status game (sudah tamat atau belum)
<i>Read</i>	: Pengguna dapat melihat daftar game yang sudah ditambahkan ke direktori. Pengguna dapat melihat keempat data yang telah dimasukkan pada proses <i>create</i> .
<i>Update</i>	: Pengguna dapat mengganti data game yang ada di direktori jika seandainya pengguna salah memasukkannya ketika sedang dalam proses <i>create</i> . Program juga dilengkapi fitur memilih bagian data mana yang akan diganti, sehingga pengguna tidak perlu repot memasukkan keempat data tersebut jikalau hanya satu data saja yang bermasalah.
<i>Delete</i>	: Pengguna dapat menghapus game yang terdapat di dalam direktori. Proses <i>Delete</i> berlaku secara permanen, artinya data game akan hilang selamanya jika pengguna memilih untuk menghapusnya. Namun, program dilengkapi mekanisme pengaman yang memastikan kembali apakah pengguna ingin menghapus game yang

	dipilih.
--	----------

3. Source Code

A. Import Library dan Deklarasi List

Program ini diawali dengan *import* library serta deklarasi list yang akan digunakan. Library yang digunakan adalah *os* dan *time*, keduanya berperan pada proses membersihkan terminal agar hasil *output* terlihat bersih dan rapi. Untuk list, ada dua list yang digunakan, yaitu list yang menyimpan menu utama (menu_awal), dan list yang menyimpan daftar game (koleksi_game)

Source Code:

```
# Import library yang diperlukan
import os
import time

# Deklarasi List yang diperlukan
# List menu_awal digunakan untuk memanggil menu utama tanpa harus mencetak nilai
# satu persatu-persatu
menu_awal = [
    ["Tambah Game Baru", "[CREATE]"], # Formatnya yaitu : ["Nama Menu",
    "Prinsip CRUD yang diterapkan"]
    ["Lihat list game", "[READ]"],
    ["Update Informasi Game", "[UPDATE]"],
    ["Hapus Game dari Koleksi", "[DELETE]"]
]

# List ini menyimpan semua informasi / data game yang kita masukkan ke direktori
# Formatnya yaitu : ["Nama Game", "Tahun rilis game", "Developer Game", "Status
game"]
koleksi_game = [
    ["Armored Core V", "2008", "Fromsoftware", "Belum Tamat"],
    ["Armored Core Verdict Day", "2009", "Fromsoftware", "Tamat"],
    ["Armored Core 4: For Answer", "2008", "Fromsoftware", "Tamat"],
    ["Armored Core 4", "2007", "Fromsoftware", "Belum Tamat"],
]
```

B. Menu Utama

Bagian ini merupakan penghubung dari setiap proses yang ada pada CRUD. Di bagian menu utama ini pengguna dapat memilih menu yang diinginkan.

Source Code:

```
jumlah_baris = 1 # Me-reset ulang jumlah baris yang akan digunakan ketika list
menu_awal dipanggil
indeks_global = 0 # Indeks utama yang digunakan diseluruh bagian program

# Bagian header menu utama
print("=" * 25, "SELAMAT DATANG DI DIREKTORI GAME SEDERHANA", "=" * 25)
print(f"{'No':5}{'Nama Menu':69}{'Fungsi'}")
print("=" * 94)

# Perulangan ini bertujuan untuk memanggil menu utama lewat list yang sudah
dideklarasikan sebelumnya
for i in menu_awal:

print(f"{str(jumlah_baris):5}{menu_awal[indeks_global][0]:69}{menu_awal[indeks_g
lobal][1]}")
    jumlah_baris += 1
    indeks_global += 1

# Bagian ini menampilkan opsi keluar / menghentikan proram
print("=" * 94)
print(f"{'0':5}Keluar")
print("=" * 94)

# Bagian input pilihan yang diinginkan oleh pengguna
pilih_menu = int(input("Masukkan menu yang ingin dipilih : "))
print("=" * 94)

# Bagian ini bertujuan untuk membersihkan terminal
time.sleep(1)
os.system("clear")
os.system("cls")
```

C. Fitur menambahkan game baru ke dalam direktori (Create)

Seperti yang sudah dijelaskan sebelumnya, pengguna dapat menambahkan game baru ke dalam direktori. Di bagian ini terdapat sebuah mekanisme yang memastikan agar data yang dimasukkan oleh pengguna tidak kosong. Berikut merupakan cuplikan singkat dari fitur ini beserta mekanisme pengaman agar data yang dimasukkan tidak kosong.

Source Code:

```
# Bagian header serta input data game yang ingin dimasukkan ke direktori
print("=" * 25, "MENAMBAHKAN GAME BARU KE DIREKTORI", "=" * 25)
game_baru = input(f"{'Masukkan Judul Game':30}: ")
tahun_rilis = input(f"{'Masukkan Tahun Rilis Game':30}: ")
```



```

        developer_game = input(f"{'Masukkan Developer Game':30}: ")
        status_game = input(f"{'Masukkan Status Game':30}: ")
        print("=" * 86)
        # (Terdapat blok kode diantara kedua kode ini)

# Memeriksa apakah ada dari keempat data tersebut apakah ada yang kosong
    if game_baru and tahun_rilis and developer_game and status_game !=
0: # Jika semua data terisi, program lanjut ke bagian ini
        # Bagian ini menampilkan informasi terkait game baru yang akan
        dimasukkan ke direktori
        print("=" * 45, "INFORMASI GAME BARU", "=" * 45)
        print(f"{'Judul Game':45} {'Tahun Rilis':20} {'Developer':25}
{'Status'}")
        print(f"{game_baru:45} {tahun_rilis:20} {developer_game:25}
{status_game}")
        print("=" * 111)
        input("Tekan 'enter' untuk melanjutkan")
        print("=" * 111)

```

D. Fitur melihat game yang terdapat pada direktori (Read)

Fitur ini cukup sederhana. Pengguna dapat melihat game apa saja yang terdapat di direktori.

Source code:

```

# Header Program
print("=" * 49, "KOLEKSI GAME", "=" * 48)
print(f"{'No':5}{'Judul Game':45} {'Tahun Rilis':20} {'Developer':25}
{'Status'}")
print("=" * 111)

# Digunakan untuk membuat daftar game yang ada di direktori
indeks_global = 0
for i in koleksi_game:
    print(f"{str(indeks_global +
1):5}{koleksi_game[indeks_global][0]:45} {koleksi_game[indeks_global][1]:20}
{koleksi_game[indeks_global][2]:25} {koleksi_game[indeks_global][3]}")
    indeks_global += 1

print("=" * 111)
input("Tekan 'enter' untuk kembali ke menu utama ")
print("=" * 111)

time.sleep(1)
os.system("clear")
os.system("cls")

```

E. Fitur Mengganti data game yang diinginkan (Update)

Bagian ini memungkinkan pengguna untuk mengganti data game yang ada di direktori jika seandainya pengguna melakukan kesalahan pada saat proses *Create*. Sama seperti proses *Create*, bagian ini dilengkapi dengan mekanisme yang dapat memeriksa apakah pengguna memasukkan data kosong atau tidak. Berikut ini merupakan cuplikan kode yang berisi fitur tersebut.

Source code:

```
# Bagian ini memeriksa judul game yang dimasukkan kosong atau tidak
if len(ganti_judul) != 0: # Program akan lanjut kesini jika judul
game yang dimasukkan tidak kosong
    koleksi_game[pilih_menu_update][0] = ganti_judul # Mengganti
judul game dengan yang baru

    print("=" * 61)
    print("Judul Game Sekarang :",
    koleksi_game[pilih_menu_update][0]) # Menampilkan judul game
setelah penggantian
    input("Tekan 'enter' untuk kembali")
    print("=" * 61)

    time.sleep(1)
    os.system("clear")
    os.system("cls")
    break
else: # Program akan menjalankan bagian ini jika judul game
kosong

    print("=" * 111)
    print("Judul Game Tidak Boleh kosong!")
    print("=" * 111)

    time.sleep(1)
    os.system("clear")
    os.system("cls")
```

F. Fitur Menghapus Game yang Sudah tidak diinginkan lagi (Delete)

Melalui fitur ini pengguna dapat menghapus game yang tidak diinginkan lagi dari direktori. Sifat dari proses *delete* ini adalah permanen, yang artinya data game akan terhapus selamanya jika pengguna memilih untuk menghapus data game tersebut. Oleh karena itu, program dilengkapi dengan sebuah pertanyaan yang bertujuan untuk memastikan kembali apakah pengguna yakin ingin menghapus game dari direktori.

Source Code:

```

        # Bagian yang memeriksa jawaban dari pengguna
        if konfirmasi_hapus == "y": # Jika menjawab ya, maka program
            akan lanjut kesini
            del koleksi_game[pilih_menu_hapus] # Bagian ini menghapus
            game yang dipilih oleh pengguna

            # Bagian ini menampilkan sisa game yang ada pada direktori
            print("=" * 110)
            print("GAME YANG TERSISA")
            print("=" * 110)

            print(f"{'No':5}{'Judul Game':45} {'Tahun Rilis':20}
{'Developer':25} {'Status'}")
            print("=" * 110)

            indeks_global = 0
            for i in koleksi_game:
                print(f"{'No':5}{'Judul Game':45} {'Tahun Rilis':20}
{'Developer':25} {'Status'}")
                indeks_global += 1
            print("=" * 110)
            input("Tekan 'enter' untuk kembali")
            print("=" * 110)

```

G. Fitur Membersihkan Terminal

Fitur ini tersebar di seluruh kode program. Fitur ini berfungsi untuk membersihkan hasil output yang ada di terminal. Tujuannya adalah Terminal menjadi rapi dan memuat informasi yang benar-benar diperlukan oleh pengguna. Fitur ini hanya terdiri dari dua baris kode, yaitu `time.sleep()` dan `os.system()`. Kode `time.sleep()` berfungsi untuk menjeda program sesuai dengan waktu yang telah ditentukan sebelumnya, sedangkan kode `os.system()` berfungsi untuk menghapus semua output yang terdapat di terminal.

Tidak seperti di program sebelum, fitur ini ditambah satu baris kode dengan tujuan agar fitur ini bisa berjalan di *Operating System* yang berbeda.

Source Code:

```

time.sleep(1)
os.system("clear")
os.system("cls")

```

4. Hasil Output

```
===== SELAMAT DATANG DI DIREKTORI GAME SEDERHANA =====
No    Nama Menu                                     Fungsi
=====
1     Tambah Game Baru                               [CREATE]
2     Lihat list game                               [READ]
3     Update Informasi Game                         [UPDATE]
4     Hapus Game dari Koleksi                       [DELETE]
=====
0     Keluar
=====
Masukkan menu yang ingin dipilih : █
```

Gambar 4.1 Tampilan menu utama program

```
===== INFORMASI GAME BARU =====
Judul Game      Tahun Rilis      Developer      Status
GTA V           2013            Rockstar Games Tamat
=====
Tekan 'enter' untuk melanjutkan █
```

Gambar 4.2 Tampilan setelah memasukkan game baru

```
===== LIST GAME SEKARANG =====
No    Judul Game      Tahun Rilis      Developer      Status
=====
1     Armored Core V   2008            Fromsoftware   Belum Tamat
2     Armored Core Verdict Day 2009            Fromsoftware   Tamat
3     Armored Core 4: For Answer 2008            Fromsoftware   Tamat
4     Armored Core 4   2007            Fromsoftware   Belum Tamat
5     tes              tes             tes            tes
6     GTA V           2013            Rockstar Games Tamat
=====
Tekan 'enter' untuk kembali █
```

Gambar 4.3 Tampilan daftar game setelah game baru dimasukkan

```
===== KOLEKSI GAME =====
No    Judul Game      Tahun Rilis      Developer      Status
=====
1     Armored Core V   2008            Fromsoftware   Belum Tamat
2     Armored Core Verdict Day 2009            Fromsoftware   Tamat
3     Armored Core 4: For Answer 2008            Fromsoftware   Tamat
4     Armored Core 4   2007            Fromsoftware   Belum Tamat
5     tes              tes             tes            tes
6     GTA V           2013            Rockstar Games Tamat
=====
Tekan 'enter' untuk kembali ke menu utama █
```

Gambar 4.4 Tampilan menu kedua (Read) yang berisi daftar game

===== MENU UPDATE =====				
No	Judul Game	Tahun Rilis	Developer	Status
=====				
1	Armored Core V	2008	Fromsoftware	Belum Tamat
2	Armored Core Verdict Day	2009	Fromsoftware	Tamat
3	Armored Core 4: For Answer	2008	Fromsoftware	Tamat
4	Armored Core 4	2007	Fromsoftware	Belum Tamat
5	tes	tes	tes	tes
6	GTA V	2013	Rockstar Games	Tamat
=====				
0	Keluar			
=====				
Masukkan nomor game yang ingin diubah : █				

Gambar 4.5 Tampilan menu ketika ingin mengubah data game

===== GAME YANG INGIN DIUPDATE =====	
1 - Judul Game	: tes
2 - Tahun Rilis	: tes
3 - Developer	: tes
4 - Status	: tes
=====	
0 - Keluar	
=====	
Masukkan bagian yang ingin diubah : █	

Gambar 4.6 Tampilan pilihan data game yang ingin diupdate

===== MENU UPDATE =====				
No	Judul Game	Tahun Rilis	Developer	Status
=====				
1	Armored Core V	2008	Fromsoftware	Belum Tamat
2	Armored Core Verdict Day	2009	Fromsoftware	Tamat
3	Armored Core 4: For Answer	2008	Fromsoftware	Tamat
4	Armored Core 4	2007	Fromsoftware	Belum Tamat
5	Hollow knight: Silksong	2025	Team Cherry	Belum Tamat
6	GTA V	2013	Rockstar Games	Tamat
=====				
0	Keluar			
=====				
Masukkan nomor game yang ingin diubah : <input type="text"/>				

Gambar 4.7 Tampilan data game setelah di update

===== MENU HAPUS =====				
No	Judul Game	Tahun Rilis	Developer	Status
=====				
1	Armored Core V	2008	Fromsoftware	Belum Tamat
2	Armored Core Verdict Day	2009	Fromsoftware	Tamat
3	Armored Core 4: For Answer	2008	Fromsoftware	Tamat
4	Armored Core 4	2007	Fromsoftware	Belum Tamat
5	Hollow knight: Silksong	2025	Team Cherry	Belum Tamat
6	GTA V	2013	Rockstar Games	Tamat
=====				
0	Keluar			
=====				
Masukkan nomor game yang ingin dihapus : <input type="text"/>				

Gambar 4.8 Tampilan menu jika pengguna ingin menghapus game

```
=====
GAME YANG INGIN DIHAPUS
=====
1 - Judul Game      : Armored Core V
2 - Tahun Rilis     : 2008
3 - Developer       : Fromsoftware
4 - Status          : Belum Tamat
=====
Apakah anda yakin ingin menghapus game ini dari direktori (y/n)? █
```

Gambar 4.9 Tampilan game yang ingin dihapus dari direktori

```
=====
GAME YANG TERSISA
=====
No  Judul Game                Tahun Rilis  Developer      Status
=====
1   Armored Core Verdict Day   2009        Fromsoftware   Tamat
2   Armored Core 4: For Answer 2008        Fromsoftware   Tamat
3   Hollow knight: Silksong    2025        Team Cherry    Belum Tamat
4   GTA V                     2013        Rockstar Games Tamat
=====
Tekan 'enter' untuk kembali █
```

Gambar 4.10 Tampilan daftar game setelah beberapa game dihapus

```
Program Selesai!
Terimakasih telah menggunakan program ini!
```

Gambar 4.11 Tampilan program setelah selesai digunakan

5. Langkah-langkah GIT

5.1 GIT Add

```
PS E:\Git\praktikum-apd\post-test\post-test-apd-5> git add 2509106017-Yoga-Ananda-Prasetya-PT-5.py
PS E:\Git\praktikum-apd\post-test\post-test-apd-5> git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   2509106017-Yoga-Ananda-Prasetya-PT-5.py
```

Gambar 5.1 Git Add

“Git add” disini berfungsi untuk memasukkan file yang sudah kita kerjakan ke dalam staging area. Staging area disini berfungsi sebagai tempat singgah sementara file-file yang ingin kita commit di git nantinya. “Git status” disini hanya berfungsi untuk memeriksa apakah file yang ingin di-commit sudah ada di staging area atau belum.

5.2 GIT Commit

```
PS E:\Git\praktikum-apd\post-test\post-test-apd-5> git commit -m "Menambahkan Source Code Post test 5"
[main 7ca1085] Menambahkan Source Code Post test 5
1 file changed, 471 insertions(+)
create mode 100644 post-test/post-test-apd-5/2509106017-Yoga-Ananda-Prasetya-PT-5.py
```

Gambar 5.2 Git Commit

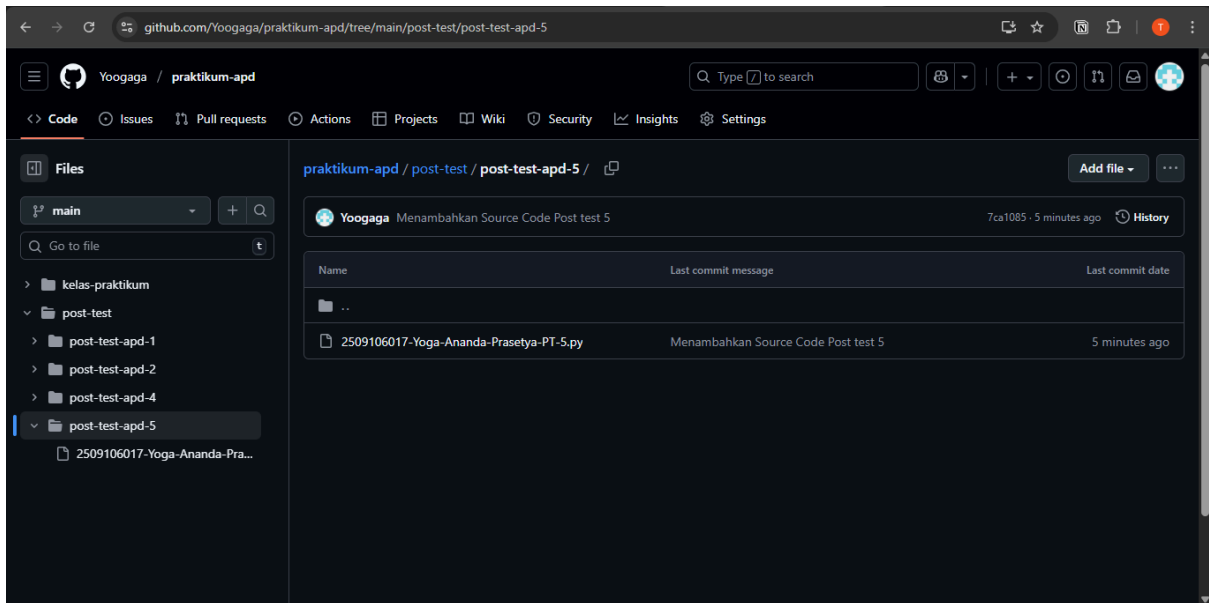
“Git Commit” disini berfungsi untuk menyimpan riwayat dari kode yang telah kita kerjakan ke dalam repository yang ada di dalam komputer kita (repository lokal). Disini kita juga dapat meninggalkan pesan singkat untuk mengingatkan kita atau orang lain terkait apa yang sudah kita kerjakan di kode tersebut.

5.3 GIT Push

```
PS E:\Git\praktikum-apd\post-test\post-test-apd-5> git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 2 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 4.17 KiB | 30.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote: This repository moved. Please use the new location:
remote: https://github.com/Yoogaga/praktikum-apd.git
To https://github.com/Yoogaga/belajargit.git
d274c76..7ca1085 main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.3 Git Push

“Git Push” disini berfungsi untuk mengunggah file-file yang sudah di-commit di repository lokal menuju ke repository yang ada di Github. Tujuannya adalah agar file-file yang sudah dikerjakan dapat diakses secara online dan dapat diakses oleh siapapun selama pengaturan repository-nya tidak di-private.



Gambar 5.4 Tampilan pada Github