

**LAPORAN TUGAS AKHIR SISTEM OPERASI**  
**APLIKASI PEMROGRAMAN SHELL**



Disusun Oleh :

Innama Maesa Putri

TI 1B

2341720235

**POLITEKNIK NEGERI MALANG**  
**JURUSAN TEKNOLOGI INFORMASI**

**2024**

## **Bab I Pendahuluan**

### **1. Pengertian Pemrograman Shell**

Pemrograman shell adalah proses menulis skrip atau program yang dijalankan oleh shell, yaitu antarmuka baris perintah yang memungkinkan pengguna untuk berinteraksi dengan sistem operasi. Shell bertindak sebagai penerjemah perintah, menghubungkan pengguna dengan sistem operasi untuk menjalankan perintah dan skrip.

### **2. Kegunaan Pemrograman Shell**

#### **a) Otomatisasi Tugas:**

Mengotomatisasi tugas-tugas rutin seperti backup, pembersihan sistem, dan pengelolaan log.

#### **b) Scripting untuk Administrasi Sistem:**

Menulis skrip untuk mengelola pengguna, proses, dan layanan pada sistem operasi.

#### **c) Pengembangan dan Uji Coba:**

Digunakan dalam siklus pengembangan perangkat lunak untuk membangun, menguji, dan mengemas aplikasi.

#### **d) Pengolahan Data:**

Menggunakan skrip shell untuk mengumpulkan, memproses, dan menganalisis data.

### **3. Fitur-fitur Umum Pemrograman Shell**

#### **a) Pengendalian Aliran:**

Mendukung konstruksi pengendalian aliran seperti perulangan (for, while, until) dan percabangan (if, case).

#### **b) Pemrosesan Teks:**

Menyediakan alat untuk memanipulasi teks seperti sed, awk, dan perintah bawaan untuk penggantian teks, pencarian, dan pemrosesan pola.

#### **c) Manajemen Proses:**

Dapat memulai, menghentikan, dan memantau proses menggunakan perintah seperti ps, kill, bg, fg, dan jobs.

#### **d) Manipulasi File:**

Memungkinkan operasi pada file dan direktori seperti membuat, menghapus, mengubah nama, dan mengubah izin file menggunakan perintah seperti touch, rm, mv, dan chmod.

e) Penanganan Input/Output:

Mendukung pengalihan input dan output (redirection) dan piping untuk menghubungkan output dari satu perintah ke input perintah lain.

## Bab II Aplikasi

### 1. Penjelasan Fitur

#### a) Membuat direktori

Fitur untuk membuat direktori baru menggunakan perintah *mkdir*, ketika pengguna menjalankan fitur ini pengguna diminta untuk memasukkan nama direktori yang akan dibuat. Kemudian sistem akan membuat direktori baru dengan nama yang telah diinputkan.

#### b) Melihat direktori dan file

Fitur untuk melihat direktori dan file menggunakan perintah *dir*, sistem akan menampilkan file dan direktori yang ada di dalam direktori dimana aplikasi disimpan.

#### c) Download repository github

Fitur untuk mengunduh repository github menggunakan perintah *clone*. Sebelum menjalankan fitur ini, pengguna sebaiknya menginstall package git terlebih dahulu di sistem operasi. Saat menjalankan fitur ini, pengguna akan memasukkan link dari repository yang akan diunduh kemudian sistem akan meng-cloning repository tersebut dan akan terdapat pilihan penempatan hasil clone. Jika pengguna memilih penempatan direktori asal maka hasil clone akan langsung tersimpan. Namun jika memilih penempatan di direktori lain maka hasil clone akan berada pada direktori yang dipilih.

#### d) Cek konektivitas

Fitur ini digunakan untuk memeriksa konektivitas jaringan ke sebuah host atau server menggunakan perintah *ping*. Ketika pengguna menjalankan fitur ini, pengguna akan diminta untuk memasukkan alamat IP atau nama host yang ingin diuji konektivitasnya. Sistem juga akan meminta pengguna memasukkan banyak package yang ingin dikirimkan. Sistem akan mengirim paket ping ke alamat yang ditentukan dan menampilkan hasilnya kepada pengguna.

#### e) Hapus repository github

Fitur ini digunakan untuk menghapus direktori yang berisi repository GitHub yang telah di-clone menggunakan perintah *rm -rf*. Pengguna akan diminta untuk memasukkan nama direktori yang akan dihapus, dan sistem akan menghapus direktori tersebut beserta isinya.

#### f) Hapus folder

Fitur ini digunakan untuk menghapus folder tertentu menggunakan perintah `rm -r`. Pengguna akan diminta untuk memasukkan nama folder yang ingin dihapus, dan sistem akan menghapus folder tersebut.

g) Copy file

Fitur ini digunakan untuk menyalin file dari satu lokasi ke lokasi lain menggunakan perintah `cp`. Pengguna akan diminta untuk memasukkan nama file sumber dan nama file tujuan (beserta path jika berada di direktori yang berbeda).

h) Move file

Fitur ini digunakan untuk memindahkan file dari satu lokasi ke lokasi lain. Pengguna akan diminta untuk memasukkan nama file sumber dan nama file tujuan (beserta path jika berada di direktori yang berbeda).

i) Read file

Fitur ini digunakan untuk membaca dan menampilkan isi file. Pengguna akan diminta untuk memasukkan nama file yang ingin dibaca, dan sistem akan menampilkan isinya.

## 2. Hasil

a) Membuat direktori

```
=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 1
=====
Membuat direktori baru
Masukkan nama direktori : direktori baru
=====
```

b) Melihat direktori dan file

```

=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 2
=====
Melihat daftar direktori dan file
Daftar direktori dan file :
baru coba direktori ppp uas.sh wowo.txt
=====

```

c) Download repository github

```

=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 3
=====
Download repository github
Masukkan link github : https://github.com/sukinnamz/UAS-SISTEMOPERASI.git
Pilih direktori :
1. Direktori ini
2. Pilih direktori lain
==> 1
Cloning into 'UAS-SISTEMOPERASI'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 5), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (16/16), 442.42 KiB | 1022.00 KiB/s, done.
Resolving deltas: 100% (5/5), done.
=====

```

d) Cek konektivitas

```

=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 4
=====
Cek konektivitas
Masukkan server : youtube.com
Masukkan jumlah paket yang ingin dikirim : 4
PING youtube.com (142.251.12.136) 56(84) bytes of data.
64 bytes from se-in-f136.1e100.net (142.251.12.136): icmp_seq=1 ttl=128 time=47.2 ms
64 bytes from se-in-f136.1e100.net (142.251.12.136): icmp_seq=2 ttl=128 time=34.2 ms
64 bytes from se-in-f136.1e100.net (142.251.12.136): icmp_seq=3 ttl=128 time=29.1 ms
64 bytes from se-in-f136.1e100.net (142.251.12.136): icmp_seq=4 ttl=128 time=39.5 ms

--- youtube.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 29.144/37.519/47.217/6.693 ms
=====

```

e) Hapus repository github

```
=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 5
=====
Menghapus repository githun
Masukkan nama repository : UAS-SISTEMOPERASI
=====
```

f) Hapus folder

```
=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 6
=====
Menghapus folder
Masukkan nama folder : direktori
=====
```

g) Copy file

```

=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 7
=====
Menyalin file
Masukkan nama file : wowo.txt
Masukkan direktori tujuan : baru
=====

```

h) Move file

```

=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 8
=====
Memindahkan file
Masukkan nama file : wowo.txt
Masukkan direktori tujuan : coba
=====

```

i) Read file



```
=====
Daftar menu
=====
1. Membuat direktori
2. Lihat direktori dan file
3. Download repository github
4. Cek konektivitas
5. Hapus repository github
6. Hapus folder
7. Copy file
8. Move file
9. Read file
10. Keluar
Masukkan menu : 9
=====
Membaca isi file
Masukkan nama file : uas.sh
echo "=====
echo "Selamat Datang Di Aplikasi Innam"
echo "=====
while true; do
echo "=====
echo "Daftar menu"
echo "=====
echo "1. Membuat direktori"
echo "2. Lihat direktori dan file"
echo "3. Download repository github"
echo "4. Cek konektivitas"
echo "5. Hapus repository github"
echo "6. Hapus folder"
echo "7. Copy file"
echo "8. Move file"
echo "9. Read file"
echo "10. Keluar"
echo -n "Masukkan menu : "
read menu

case $menu in
```

### **Bab III Penutup**

#### **1. Kesimpulan**

Dalam pembelajaran sistem operasi, praktek untuk membuat aplikasi sebagai tugas akhir dirasa cukup efektif untuk pemahaman mahasiswa mengenai sistem operasi dan perintah-perintah di dalamnya. Dengan tugas ini penulis sendiri dapat memahami pengertian aplikasi shell, kegunaan, fitur-fitur umum hingga penerapannya pada sistem operasi.

#### **2. Saran**

Berdasarkan hasil implementasi dan pengujian fitur-fitur di atas, berikut adalah beberapa saran untuk pengembangan dan penggunaan lebih lanjut:

##### **a) Keamanan dan Validasi Input:**

- Tambahkan validasi input untuk memastikan bahwa pengguna memasukkan data yang benar dan valid. Misalnya, memeriksa apakah direktori atau file sudah ada sebelum membuat atau menghapusnya.
- Implementasikan mekanisme keamanan untuk mencegah operasi berbahaya, seperti penghapusan direktori penting secara tidak sengaja.

##### **b) Logging dan Pelaporan Kesalahan:**

- Tambahkan fitur logging untuk mencatat semua operasi yang dilakukan oleh skrip. Ini akan membantu dalam pelacakan dan debugging jika terjadi masalah.
- Buat mekanisme pelaporan kesalahan yang lebih baik agar pengguna dapat memahami dan menangani kesalahan dengan mudah.

##### **c) Antarmuka Pengguna yang Lebih Baik:**

- Pertimbangkan untuk mengembangkan antarmuka pengguna grafis (GUI) untuk membuat skrip lebih user-friendly, terutama bagi pengguna yang kurang familiar dengan antarmuka baris perintah.
- Gunakan warna dan format teks yang berbeda untuk output yang lebih jelas dan mudah dibaca.