

**LAPORAN RESMI  
UJIAN AKHIR SEMESTER  
PRAKTIKUM SISTEM OPERASI**

**VArchiver**



**NIM : 21104410049  
NAMA : Viery Nugroho  
JURUSAN : Teknik Informatika  
KELAS : TI 4B 2023**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ISLAM BALITAR  
2023**

# **LEMBAR PERSETUJUAN**

## **VArchiver**

**NIM** : 21104410049  
**NAMA** : Viery Nugroho  
**JURUSAN** : Teknik Informatika  
**KELAS** : TI 4B 2023

Disetujui,  
Blitar, ... Juli 2023  
Dosen

Mohammad Faried Rahmat, S.ST., M.Tr.T  
NIDN.

## DESKRIPSI PROGRAM

```
-----
Selamat Datang di VArchiver

VArchiver
Aplikasi Pengelola File meliputi Melihat Daftar File,
Menghapus File, Mengcopy File, Memotong File,
Kompres dan Ekstrak File

--- Dibuat Oleh ---
Nama : Viery Nugroho
NIM  : 21104410049
Kelas: Teknik Informatika 4B
-----

-- Pilih Menu VArchiver --
1. Lihat File
2. Hapus File/Folder
3. Copy File
4. Move File
5. Kompres File
6. Ekstrak File
7. Kompres Folder
8. Ekstrak Folder
-----
Masukkan Pilihan:
```

Gambar 1 VArchiver

VArchiver adalah sebuah program yang terinspirasi dari sebuah aplikasi android bernama “ZArchiver” yang berfungsi untuk mengelola sebuah file dan folder atau biasa disebut sebagai file manager. Nama VArchiver diambil dari nama aplikasi ZArchiver dimana penggunaan huruf V diambil dari nama saya, yaitu Viery.

Terdapat total 8 fitur pada program VArchiver, yaitu:

1. Lihat File

Untuk melihat daftar file dengan perintah linux yang dijalankan adalah **ls -l** yang berfungsi untuk menampilkan informasi tentang sebuah direktori. Informasi yang akan ditampilkan meliputi permission, jumlah hardlink (mengacu pada elemen induknya), owner, group, size, last modification time, directory name.

Dalam penggunaannya user hanya perlu memasukkan path direktori untuk memeriksa isi dari direktori tersebut.

2. Hapus File / Folder

Untuk menghapus File menggunakan perintah linux **rm -i**, yang ketika perintah dijalankan maka akan mengeluarkan semacam alert sebelum perintah hapus dieksekusi. Dan untuk menghapus folder menggunakan perintah linux **rm -r** yang berfungsi untuk menghapus direktori dan seluruh isinya.

Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file dan menghapus file.

### 3. Copy File

Untuk menyalin file menggunakan perintah linux **cp**, Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file dan menyalin file. Dan input kedua adalah nama file yang akan dieksekusi. Dan ketika program copy dijalankan, maka file otomatis tersalin dengan format nama "copy-(nama\_file)".

### 4. Move File

Untuk menyalin file menggunakan perintah linux **cp**, Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file dan menyalin file. Dan input kedua adalah nama file yang akan dieksekusi. Dan ketika program move dijalankan, maka file otomatis dipindahkan dalam folder move\_folder.

### 5. Kompres File

Untuk mengompres file menggunakan perintah linux **gzip -c**. Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file dan mengompres file. Dan input kedua adalah nama file yang akan dieksekusi. Dan ketika program kompres file dijalankan, maka file otomatis dikompresi dalam dengan format nama (file).gz.

### 6. Ekstrak File

Untuk mengekstrak file menggunakan perintah linux **gunzip**. Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file dan mengekstrak file. Dan input kedua adalah nama file yang akan dieksekusi. Dan ketika program ekstrak file dijalankan, maka file otomatis diekstrak.

### 7. Kompres Folder

Untuk mengompres folder menggunakan perintah linux **tar -czf**. Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file/folder dan mengompres folder. Dan input kedua adalah nama folder yang akan

dieksekusi. Dan ketika program kompres folder dijalankan, maka folder otomatis dikompres dengan format nama “(nama\_folder).tar.gz”.

#### 8. Ekstrak Folder

Untuk mengekstrak folder menggunakan perintah linux **tar -xf**. Dalam penggunaannya user menginputkan 2 buah data, yaitu path direktori untuk melihat isi direktori, hal ini bertujuan untuk memudahkan user dalam mengingat nama file/folder dan mengekstrak folder. Dan input kedua adalah nama folder yang akan dieksekusi. Dan ketika program ekstrak folder dijalankan, maka folder otomatis diekstrak.

Sebelum perintah pada masing-masing program dijalankan akan terjadi pengecekan apakah data yang dimasukkan user sesuai dengan syarat untuk menjalankan program diatas, hal ini bertujuan untuk meminimalisir terjadinya eror pada program. Dan jika syarat tidak terpenuhi maka program akan menjalankan perintah untuk keluar dari program yang dimana berarti program gagal atau batal dijalankan.

## PENJELASAN PROGRAM

```
root@viery:/home/viery/prak-sisop/UAS# ls
copy.sh      hapus.sh      lihat_file.sh  move.sh      VArchiver.sh
ekstrak_folder.sh  kompres_folder.sh  menu.sh      opening.sh
ekstrak.sh    kompres.sh    move_folder    tes.txt
root@viery:/home/viery/prak-sisop/UAS#
```

Gambar 2 Struktur Folder dan File Program

Program VArchiver ini terdiri dari beberapa file fitur seperti yang dijelaskan pada deskripsi program di atas, dan juga terdapat 3 file utama yaitu opening.sh, menu.sh dan VArchiver.sh. Selain itu terdapat folder dengan nama move\_folder untuk menyimpan data yang dipindah dan 1 buah file tes.txt untuk menguji program.

### 1. VArchiver.sh

```
. opening.sh
. menu.sh
```

Gambar 3 VArchiver.sh

File VArchiver.sh ini berisikan 2 buah baris kode saja yaitu import file opening.sh dan juga file menu.sh. File tersebut diimport dengan keyword titik (.) kemudian diikuti dengan nama file. File VArchiver.sh inilah nanti yang akan dijalankan ketika kita ingin menjalankan program VArchiver.

### 2. opening.sh

```
# Viery Nugroho
# 21104410049
# UAS - Praktikum Sistem Operasi

clear

echo "-----"
echo "Selamat Datang di VArchiver"
echo ""
echo "VArchiver"
echo "Aplikasi Pengelola File meliputi Melihat Daftar File, "
echo "Menghapus File, Mengcopy File, Memotong File,"
echo "Kompres dan Ekstrak File"
echo ""
echo "--- Dibuat Oleh ---"
echo "Nama : Viery Nugroho"
echo "NIM : 21104410049"
echo "Kelas: Teknik Informatika 4B"
echo "-----"
```

Gambar 4 opening.sh

File opening.sh berisikan beberapa kalimat seperti penjelasan program dan sedikit identitas saya sebagai pembuat program VArchiver.

Perintah clear berfungsi untuk menghapus layar terminal menjadi bersih lagi dan echo berfungsi untuk menampilkan sebuah kalimat. Dan untuk kalimat berwarna biru merupakan sebuah komentar yang dimana hal itu tidak akan tereksekusi oleh program.

### 3. menu.sh

```
# program
echo "-- Pilih Menu VArchiver --"
echo "1. Lihat File"
echo "2. Hapus File/Folder"
echo "3. Copy File"
echo "4. Move File"
echo "5. Kompres File"
echo "6. Ekstrak File"
echo "7. Kompres Folder"
echo "8. Ekstrak Folder"
echo "-----"
read -p "Masukkan Pilihan: " pilihan

case "$pilihan" in
    1)
        clear
        echo "---- VArchiver ----"
        echo "--- Lihat File ---"
        . lihat_file.sh
        ;;
    2)
        clear
        echo "---- VArchiver ----"
        echo "--- Hapus File/Folder ---"
        . hapus.sh
        ;;
    3)
        clear
        echo "---- VArchiver ----"
        echo "--- Copy File ---"
        . copy.sh
        ;;
```

Gambar 5 menu.sh - 1

```
4)
    clear
    echo "---- VArchiver ----"
    echo "--- Cut File ---"
    . move.sh
    ;;
5)
    clear
    echo "---- VArchiver ----"
    echo "--- Kompres File ---"
    . kompres.sh
    ;;
6)
    clear
    echo "---- VArchiver ----"
    echo "--- Ekstrak File ---"
    . ekstrak.sh
    ;;
7)
    clear
    echo "---- VArchiver ----"
    echo "--- Kompres Folder ---"
    . kompres_folder.sh
    ;;
8)
    clear
    echo "---- VArchiver ----"
    echo "--- Ekstrak Folder ---"
    . ekstrak_folder.sh
    ;;
*)
    echo "Pilihan Tidak Ada"
    . VArchiver.sh
    ;;
esac
```

Gambar 6 menu.sh - 2

Pada file menu.sh ini berisikan tampilan menu dan pengondisian untuk pemilihan menu menggunakan case. Terdapat 2 perintah baru di halaman ini yaitu read dan case.

#### a) Read

Perintah ini digunakan untuk membaca masukan dari user dan disimpan ke sebuah variable. Terdapat opsi **-p** yang digunakan untuk menampilkan pesan sebelum user memberikan masukan.

#### b) Case

Perintah ini bersifat sama seperti pengondisian switch case pada bahasa pemrograman lain seperti PHP, Javascript, C#, dan lain-lain. Namun, pada shell perintah case dibuat lebih simple dimana kita hanya perlu menuliskan case diikuti nama variable dan in sebagai opsi untuk membandingkan nilai variable kepada opsi yang ada dalam blok kode case. Setiap pembandingan akan diletakkan sebelum tanda tutup kurung yang kemudian diakhiri dengan 2 titik koma / semicolon.

Dan untuk memberikan nilai default pada case di shell, blok kode ditaruh pada nilai pembandingan \* yang dimana ini akan dieksekusi ketika masukan user tidak menemukan kecocokan dengan nilai pembandingan yang ada pada case.

Ketika pembandingan memiliki nilai yang sama dengan masukan dari user maka blok kode dengan opsi tertentu akan dieksekusi, dimana pada setiap blok kode memiliki fungsi untuk menghapus tampilan layar terminal, menampilkan kalimat, dan mengimport atau memanggil halaman shell lain untuk dijalankan.

#### 4. Lihat\_file.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo ""

read -p "Masukkan Path Direktori: " direktori

#program

#cek
if [ -n "$direktori" ] && [ -d "$direktori" ]; then
    ls -l "$direktori"
else
    echo "$direktori bukan sebuah Direktori!"
    exit 1
fi
```

Gambar 7 lihat\_file.sh

Pada file ini terdapat program untuk melihat daftar isi dari sebuah direktori/folder dimana sebelum kondisi dijalankan terdapat sebuah perintah untuk user agar memasukkan path direktori atau lokasi direktori yang akan dicek isi dari direktori tersebut. Jika user sudah memasukkan nilai pada variable direktori, selanjutnya data masukan akan dicek dengan pengondisian seperti pada gambar 7 dengan kondisi apakah **-n direktori** yang berarti apakah variable direktori bernilai kosong atau tidak. Dan kondisi **-d direktori** yang berfungsi untuk melakukan pengecekan apakah variable direktori merupakan sebuah direktori atau bukan.

Jika kondisi terpenuhi maka perintah **ls -l** akan dijalankan dimana perintah ini berfungsi untuk melihat daftar isi dari suatu direktori secara mendetail meliputi permission, jumlah hardlink (mengacu pada elemen induknya), owner, group, size, last modification time, directory name.



## 5. copy.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah file yang akan di-copy"
echo ""

read -p "-> Masukkan Path Direktori Dari File: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

read -p "-> Masukkan File Yang Akan Dicopy: " file
#cek file
if [ -f "$file" ] && [ -e "$file" ]; then
    cp "$file" "copy-$file"
    echo "file Berhasil Di Copy!"
else
    echo "$file - Bukan Sebuah File!"
    exit 1
fi
```

Gambar 8 copy.sh

Pada file ini berisikan perintah untuk mencopy file yang dimana nama file telah diinputkan oleh user. Kemudian perintah **cp** untuk menyalin file akan dieksekusi dimana file akan dicopy dan secara default, file yang telah dicopy mempunyai nama **copy-nama\_file**. Tetapi sebelum melakukan eksekusi copy file tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa file atau bukan. Validasi dilakukan dengan menjalankan perintah **-f** yang merupakan perintah untuk mengecek apakah file tersebut merupakan sebuah file atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah file ada atau tidak. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam penyalinan file user diharuskan untuk memasukkan path file dimana file tersebut ada untuk ditampilkan guna mempermudah user untuk menyalin sebuah file dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

## 6. move.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah file yang akan di-pindah"
echo ""

read -p "-> Masukkan Path Direktori Dari File: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

read -p "-> Masukkan File Yang Akan Dipindah: " file
#cek file
if [ -f "$file" ] && [ -e "$file" ]; then
    mv "$file" "move_folder/$file"
    echo "file Berhasil Di Pindah!"
else
    echo "$file - Bukan Sebuah File!"
    exit 1
fi
```

Gambar 9 move.sh

Pada file ini berisikan perintah untuk mencopy file yang dimana nama file telah diinputkan oleh user. Kemudian perintah **mv** untuk memindah file akan dieksekusi dimana file akan dipindah dan secara default ke folder `move_folder`. Tetapi sebelum melakukan eksekusi move file tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa file atau bukan. Validasi dilakukan dengan menjalankan perintah **-f** yang merupakan perintah untuk mengecek apakah file tersebut merupakan sebuah file atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah file ada atau tidak. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam pemindahan file user diharuskan untuk memasukkan path file dimana file tersebut ada untuk ditampilkan guna mempermudah user untuk memindah sebuah file dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

## 7. kompres.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah file yang akan di-kompres"
echo ""

read -p "-> Masukkan Path Direktori Dari File: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

read -p "-> Masukkan File Yang Akan Dikompres: " file
#cek file
if [ -f "$file" ] && [ -e "$file" ]; then
    gzip -c "$file" > "$file.gz"
    echo "file Berhasil Di Kompres!"
else
    echo "$file - Bukan Sebuah File!"
    exit 1
fi
```

Gambar 10 kompres.sh

Pada file ini berisikan perintah untuk mengompres file yang dimana nama file telah diinputkan oleh user. Kemudian perintah **gzip -c** untuk mengompres file akan dieksekusi dimana file yang telah dikompres akan ditempatkan ke folder yang sama. Tetapi sebelum melakukan eksekusi kompres file tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa file atau bukan. Validasi dilakukan dengan menjalankan perintah **-f** yang merupakan perintah untuk mengecek apakah file tersebut merupakan sebuah file atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah file ada atau tidak. Dimana. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam pengompresan file user diharuskan untuk memasukkan path file dimana file tersebut ada untuk ditampilkan guna mempermudah user untuk mengompres sebuah file dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

## 8. ekstrak.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah file yang akan di-ekstrak"
echo ""

read -p "-> Masukkan Path Direktori Dari File: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

read -p "-> Masukkan File Yang Akan Diekstrak: " file
#cek file
if [ -f "$file" ] && [ -e "$file" ]; then
    gunzip "$file"
    echo "file Berhasil Di Ekstrak!"
else
    echo "$file - Bukan Sebuah File / File Bukan Berekstensi .gz!"
    exit 1
fi
```

Gambar 11 Ekstrak.sh

Pada file ini berisikan perintah untuk mengekstrak file yang dimana nama file telah diinputkan oleh user. Kemudian perintah **gunzip** untuk mengekstrak file akan dieksekusi dimana file yang telah diekstrak akan ditempatkan ke folder yang sama. Tetapi sebelum melakukan eksekusi ekstrak file tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa file atau bukan. Validasi dilakukan dengan menjalankan perintah **-f** yang merupakan perintah untuk mengecek apakah file tersebut merupakan sebuah file atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah file ada atau tidak. Dimana. Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam pengestrakan file user diharuskan untuk memasukkan path file dimana file tersebut ada untuk ditampilkan guna mempermudah user untuk mengekstrak sebuah file dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

## 9. kompres\_folder.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah folder yang akan di-kompres"
echo ""

read -p "-> Masukkan Path Direktori Dari Folder: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

pwd

read -p "-> Masukkan Folder Yang Akan Dikompres: " folder
#cek file
if [ -n "$folder" ] && [ -d "$folder" ]; then
    tar -czf "$folder.tar.gz" "$folder"
    echo "folder Berhasil Di Kompres!"
else
    echo "$folder - Bukan Sebuah Folder!"
    echo "Kompres Gagal"
    exit 1
fi
```

Gambar 12 kompres\_folder.sh

Pada file ini berisikan perintah untuk mengompres folder yang dimana nama folder telah diinputkan oleh user. Kemudian perintah **tar -czf** untuk mengompres folder dan akan dieksekusi dimana folder yang telah dikompres akan ditempatkan ke folder yang sama. **Tar** merupakan perintah untuk mengompresi arsip file. **-c** adalah opsi untuk menunjukkan Tindakan "create". **-z** adalah opsi untuk menambahkan kompresi menggunakan gzip. Dan **-f** adalah opsi untuk menentukan nama file arsip yang dihasilkan. Tetapi sebelum melakukan eksekusi kompres folder tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa folder atau bukan. Validasi dilakukan dengan menjalankan perintah **-d** yang merupakan perintah untuk mengecek apakah folder tersebut merupakan sebuah folder/direktori atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah folder ada atau tidak. Dimana. Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam pengompresan folder user diharuskan untuk memasukkan path folder dimana folder dan file tersebut ada untuk ditampilkan guna mempermudah user untuk mengompres sebuah folder dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

#### 10.ekstrak\_folder.sh

```
echo ""
echo "NOTE: "
echo "- input sebuah path direktori"
echo "- input sebuah folder yang akan di-ekstrak"
echo ""

read -p "-> Masukkan Path Direktori Dari Folder: " path

#cek path
if [ -n "$path" ] && [ -d "$path" ]; then
    ls "$path"
else
    echo "$path - Tidak Tersedia"
    echo "-----"
    exit 1
fi

read -p "-> Masukkan Folder Yang Akan Diekstrak: " folder
#cek file
if [ -n "$folder" ] && [ -f "$folder" ]; then
    tar -xf "$folder"
    echo "file_folder Berhasil Di Ekstrak!"
else
    echo "$folder - Bukan Sebuah File!"
    echo "Ekstrak Gagal"
    exit 1
fi
```

Gambar 13 ekstrak\_folder.sh

Pada file ini berisikan perintah untuk mengekstrak folder yang dimana nama folder telah diinputkan oleh user. Kemudian perintah **tar -xf** untuk mengekstrak folder akan dieksekusi dimana folder yang telah diekstrak akan ditempatkan ke folder yang sama. **Tar** merupakan perintah untuk mengompresi arsip file. **-x** digunakan untuk menunjukkan Tindakan ekstraksi dan **-f** adalah opsi untuk menentukan nama file arsip yang akan diolah. Tetapi sebelum melakukan eksekusi ekstrak folder tersebut, inputan/masukan user akan lebih dahulu masuk ke tahap validasi atau pengecekan apakah inputan user berupa folder atau bukan. Validasi dilakukan dengan menjalankan perintah **-d** yang merupakan perintah untuk mengecek apakah folder tersebut merupakan sebuah folder/direktori atau bukan dan **-e** berfungsi untuk melakukan pengecekan apakah folder ada atau tidak. Dimana. Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

Tetapi sebelum itu, untuk mempermudah dalam pengekstrakan folder user diharuskan untuk memasukkan path folder dimana folder dan file tersebut ada untuk ditampilkan guna mempermudah user untuk mengekstrak sebuah folder dengan menjalankan perintah **ls** yang berfungsi untuk menampilkan daftar isi dari sebuah direktori. Program ini dijalankan setelah melakukan validasi dengan perintah **-n** yang berfungsi untuk mengecek apakah variable bernilai kosong atau tidak dan juga perintah **-d** untuk mengecek apakah nilai dari sebuah variable merupakan sebuah direktori atau bukan. Dan Jika kondisi tidak terpenuhi maka program akan dikeluarkan dan program batal dijalankan.

## PENGUJIAN PROGRAM

### 1. VArchiver.sh

```
-----
Selamat Datang di VArchiver

VArchiver
Aplikasi Pengelola File meliputi Melihat Daftar File,
Menghapus File, Mengcopy File, Memotong File,
Kompres dan Ekstrak File

--- Dibuat Oleh ---
Nama : Viery Nugroho
NIM : 21104410049
Kelas: Teknik Informatika 4B
-----

-- Pilih Menu VArchiver --
1. Lihat File
2. Hapus File/Folder
3. Copy File
4. Move File
5. Kompres File
6. Ekstrak File
7. Kompres Folder
8. Ekstrak Folder
-----

Masukkan Pilihan:
```

Gambar 14 VArchiver

Pada VArchive.sh menampilkan data hasil program dari file opening.sh dan menu.sh. File menu.sh digunakan untuk melakukan pemilihan program yang akan dijalankan. Pada halaman ini user menginputkan pilihan untuk memilih program mana yang akan dijalankan.

### 2. Lihat\_file.sh

```
---- VArchiver ----
--- Lihat File ---

NOTE:
- input sebuah path direktori

Masukkan Path Direktori: /home/viery/prak-sisop/UAS
total 52
-rwxr-xr-x 1 root root 523 Jul 21 02:50 copy.sh
-rwxr-xr-x 1 root root 569 Jul 21 04:06 ekstrak_folder.sh
-rwxr-xr-x 1 root root 552 Jul 21 03:32 ekstrak.sh
-rwxr-xr-x 1 root root 590 Jul 21 04:13 hapus.sh
-rwxr-xr-x 1 root root 588 Jul 21 04:03 kompres_folder.sh
-rwxr-xr-x 1 root root 537 Jul 21 03:01 kompres.sh
-rwxr-xr-x 1 root root 261 Jul 21 15:23 lihat_file.sh
-rwxr-xr-x 1 root root 1116 Jul 21 04:09 menu.sh
drwxr-xr-x 2 root root 4096 Jul 21 06:29 move_folder
-rwxr-xr-x 1 root root 536 Jul 21 02:56 move.sh
-rwxr-xr-x 1 root root 519 Jul 21 01:43 opening.sh
-rw-r--r-- 1 root root 11 Jul 21 03:41 tes.txt
-rwxr-xr-x 1 root root 23 Jul 21 01:59 VArchiver.sh
root@viery:/home/viery/prak-sisop/UAS# _
```

Gambar 15 lihat\_file.sh

Pada program lihat\_file.sh menampilkan daftar isi dari sebuah direktori meliputi permission, jumlah hardlink (mengacu pada elemen induknya), owner, group, size, last modification time, directory name. Pada halaman ini user menginputkan path direktori untuk menampilkan daftar isi dari sebuah direktori.

### 3. Hapus.sh

```
---- VArchiver ----
--- Hapus File/Folder ---

NOTE:
- input sebuah path direktori
- input sebuah file yang akan dihapus

-> Masukkan Path Direktori Dari File: /home/viery/prak-sisop/UAS
copy.sh      hapus.sh      lihat_file.sh  move.sh      VArchiver.sh
ekstrak_folder.sh  kompres_folder.sh  menu.sh      opening.sh
ekstrak.sh     kompres.sh      move_folder   tes.txt
-> Masukkan File Yang Akan Dihapus: tes.txt
rm: remove regular file 'tes.txt'? y
file Berhasil Dihapus!
root@viery:/home/viery/prak-sisop/UAS#
```

**Gambar 16 hapus.sh**

Pada program hapus.sh berfungsi untuk menghapus sebuah file/folder. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama file/folder yang akan dihapus.

```
root@viery:/home/viery/prak-sisop/UAS# ls
copy.sh      ekstrak.sh  kompres_folder.sh  lihat_file.sh  move_folder  opening.sh
ekstrak_folder.sh  hapus.sh    kompres.sh        menu.sh        move.sh      VArchiver.sh
```

**Gambar 17 File Terhapus**

### 4. Copy.sh

```
---- VArchiver ----
--- Copy File ---

NOTE:
- input sebuah path direktori
- input sebuah file yang akan di-copy

-> Masukkan Path Direktori Dari File: /home/viery/prak-sisop/UAS
copy.sh      hapus.sh      lihat_file.sh  move.sh      VArchiver.sh
ekstrak_folder.sh  kompres_folder.sh  menu.sh      opening.sh
ekstrak.sh     kompres.sh      move_folder   tes.txt
-> Masukkan File Yang Akan Dicopy: tes.txt
file Berhasil Di Copy!
```

**Gambar 18 copy.sh**

Pada program copy.sh berfungsi untuk menyalin sebuah file. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama file yang akan dicopy dengan nama copy-nama\_file.



## 5. Move.sh

```
---- VArchiver ----
--- Cut File ---

NOTE:
- input sebuah path direktori
- input sebuah file yang akan di-pindah

-> Masukkan Path Direktori Dari File: /home/viery/prak-sisop/UAS
copy.sh          ekstrak.sh          kompres.sh      move_folder    tes.txt
copy-tes.txt     hapus.sh          lihat_file.sh   move.sh        VArchiver.sh
ekstrak_folder.sh kompres_folder.sh menu.sh         opening.sh
-> Masukkan File Yang Akan Dipindah: copy-tes.txt
file Berhasil Di Pindah!
root@viery:/home/viery/prak-sisop/UAS# _
```

Gambar 19 move.sh

Pada program move.sh berfungsi untuk memindah sebuah file. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama file yang akan dipindah pada folder move\_folder.

```
root@viery:/home/viery/prak-sisop/UAS# ls
copy.sh          hapus.sh          lihat_file.sh   move.sh        VArchiver.sh
ekstrak_folder.sh kompres_folder.sh menu.sh         opening.sh
ekstrak.sh       kompres.sh        move_folder     tes.txt
root@viery:/home/viery/prak-sisop/UAS# ls move_folder
copy-tes.txt
root@viery:/home/viery/prak-sisop/UAS# _
```

Gambar 20 hasil move.sh

## 6. Kompres.sh

```
---- VArchiver ----
--- Kompres File ---

NOTE:
- input sebuah path direktori
- input sebuah file yang akan di-kompres

-> Masukkan Path Direktori Dari File: /home/viery/prak-sisop/UAS
copy.sh          hapus.sh          lihat_file.sh   move.sh        VArchiver.sh
ekstrak_folder.sh kompres_folder.sh menu.sh         opening.sh
ekstrak.sh       kompres.sh        move_folder     tes.txt
-> Masukkan File Yang Akan Dikompres: tes.txt
file Berhasil Di Kompres!
```

Gambar 21 kompres.sh

Pada program kompres.sh berfungsi untuk mengompres sebuah file. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama file yang akan dikompres pada folder yang sama dengan format .gz.

## 7. Ekstrak.sh

```
---- VArchiver ----
--- Ekstrak File ---

NOTE:
- input sebuah path direktori
- input sebuah file yang akan di-ekstrak

-> Masukkan Path Direktori Dari File: /home/viery/prak-sisop/UAS
copy.sh      hapus.sh      lihat_file.sh  move.sh      VArchiver.sh
ekstrak_folder.sh  kompres_folder.sh  menu.sh      opening.sh
ekstrak.sh    kompres.sh      move_folder   tes.txt.gz
-> Masukkan File Yang Akan Diekstrak: tes.txt.gz
file Berhasil Di Ekstrak!
```

Gambar 22 ekstrak.sh

Pada program ekstrak.sh berfungsi untuk mengekstrak sebuah file. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama file yang akan diekstrak. Dan file yang telah diekstrak akan terhapus.

## 8. Kompres\_folder.sh

```
---- VArchiver ----
--- Kompres Folder ---

NOTE:
- input sebuah path direktori
- input sebuah folder yang akan di-kompres

-> Masukkan Path Direktori Dari Folder: /home/viery/prak-sisop/UAS
copy.sh      hapus.sh      lihat_file.sh  move.sh      VArchiver.sh
ekstrak_folder.sh  kompres_folder.sh  menu.sh      opening.sh
ekstrak.sh    kompres.sh      move_folder   tes.txt
/home/viery/prak-sisop/UAS
-> Masukkan Folder Yang Akan Dikompres: move_folder
folder Berhasil Di Kompres!
```

Gambar 23 kompres folder

Pada program kompres.sh berfungsi untuk mengompres sebuah folder. Pada halaman ini user memasukkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user memasukkan nama folder yang akan dikompres pada folder yang sama dengan format .tar.gz.

```
root@viery:/home/viery/prak-sisop/UAS# ls
copy.sh      hapus.sh      lihat_file.sh  move_folder.tar.gz  tes.txt
ekstrak_folder.sh  kompres_folder.sh  menu.sh      move.sh             VArchiver.sh
ekstrak.sh    kompres.sh      move_folder    opening.sh
```

Gambar 24 hasil kompres folder

## 9. Ekstrak\_folder.sh

```
---- VArchiver ----  
--- Ekstrak Folder ---  
  
NOTE:  
- input sebuah path direktori  
- input sebuah folder yang akan di-ekstrak  
  
-> Masukkan Path Direktori Dari Folder: /home/viery/prak-sisop/UAS  
copy.sh          hapus.sh          lihat_file.sh      move.sh          VArchiver.sh  
ekstrak_folder.sh  kompres_folder.sh  menu.sh            opening.sh  
ekstrak.sh        kompres.sh          move_folder.tar.gz  tes.txt  
-> Masukkan Folder Yang Akan Diekstrak: move_folder.tar.gz  
file folder Berhasil Di Ekstrak!
```

**Gambar 25 ekstrak folder**

Pada program ekstrak.sh berfungsi untuk mengekstrak sebuah folder. Pada halaman ini user menginputkan path direktori untuk menampilkan daftar isi dari sebuah direktori. Dan user menginputkan nama folder terkompres yang akan diekstrak.

```
root@viery:/home/viery/prak-sisop/UAS# ls  
copy.sh          hapus.sh          lihat_file.sh      move_folder.tar.gz  tes.txt  
ekstrak_folder.sh  kompres_folder.sh  menu.sh            move.sh             VArchiver.sh  
ekstrak.sh        kompres.sh          move_folder         opening.sh  
root@viery:/home/viery/prak-sisop/UAS#
```

**Gambar 26 Hasil Ekstrak**