

LAPORAN FINAL PROJECT

SISTEM OPERASI STUDI KASUS

MERCHANT MANAGEMENT SYSTEM

Mata Kuliah Sistem Operasi

Dosen Pengampu :

Hendra Maulana, S.Kom, M.Kom



Disusun Oleh :

Mochammad Arya Salsabila (19081010001)

Muhammad Lutfi Pratama (19081010049)

Kelas D

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UPN "VETERAN" JAWA TIMUR
2020

STUDI KASUS

Dalam pengerjaan Final Project mata kuliah Sistem Operasi ini kelompok kami memilih studi kasus Merchant Management System. Inti dari program ini dalam menjalankan operasi CRUD (Create, Read, Update, Delete). Sesuai kepanjangannya CRUD ini digunakan untuk membuat, membaca, mengupdate, dan menghapus sebuah data. Dalam program yang kami kembangkan ini kami melakukan CRUD terhadap data barang. Operasi CRUD ini akan memanipulasi data barang yang berupa array kode_barang, nama_barang, dan kode_barang. Data barang yang digunakan ialah kode barang, nama barang, dan harga barang. Program ini tidak hanya bisa membaca semua data yang ada tetapi juga bisa membaca suatu data tertentu. Selain operasi CRUD program ini juga menerapkan beberapa perintah bash seperti menyisipkan suatu teks ke dalam file berekstensi .txt, membuat folder, dan mengecek apakah suatu file dan folder sudah ada atau belum.

PEMBAHASAN SOURCE CODE

A. DEKLARASI ARRAY

```
declare -a kode_barang  
declare -a nama_barang  
declare -a harga_barang
```

Deklarasi array `kode_barang`, `nama_barang`, `harga_barang` secara explicit. Array tersebut dideklarasikan secara explicit karena kita akan menetapkan nilainya di proses setelahnya.

B. DEKLARASI FUNGSI

```
function is_laporan_exists() {  
    if [[ ! -e ./laporan/$kode_laporan.txt ]]  
    then  
        status=true  
    fi  
}
```

Fungsi `is_laporan_exists()` digunakan untuk mengecek apakah file `$kode_laporan.txt` sudah ada di dalam folder laporan atau belum, jika belum maka akan mengisi nilai variabel `status` menjadi `true`.

```
function is_folder_exists() {  
    if [[ ! -d ./laporan ]]  
    then  
        mkdir laporan  
    fi  
}
```

Fungsi `is_folder_exists()` digunakan untuk mengecek apakah folder laporan sudah ada atau belum, jika belum maka akan dibuat folder baru dengan perintah `mkdir` dan nama laporan.

```
function is_barang_exists() {
    for (( i=0;i<${#kode_barang[@]};i++ ))
    do
        if [[ $kode_barang_in == ${kode_barang[$i]} ]]
        then
            status=true
            index=$i
            break
        fi
    done
}
```

Fungsi is_barang_exists() melakukan looping sebanyak jumlah elemen array kode_barang. Jumlah elemen kode barang bisa didapat dengan syntax \${#kode_barang[@]}. Jika dalam perjalanan looping kode_barang_in nilainya sama dengan kode_barang[\$i] maka looping akan dihentikan dengan break lalu mengisi variabel status dengan true dan variabel index dengan \$i.

```
function insert() {
    status=false
    n=${#kode_barang[@]}

    read -p "Kode barang yang ingin ditambahkan : " kode_barang_in
    echo ""

    is_barang_exists

    if [[ $status == true ]]
    then
        echo -e "Barang sudah ada..\n"
    else
        echo -e "Barang belum ada, lanjut..\n"

        kode_barang[$n]=$kode_barang_in
        echo "Kode Barang : ${kode_barang[$n]}"

        read -p "Nama Barang : " nama_barang[$n]
        read -p "Harga Barang : " harga_barang[$n]

        echo -e "\nBarang berhasil ditambahkan..\n"
    fi
}
```

Fungsi insert() digunakan untuk menambahkan barang. Diawali dengan input kode barang lalu dicek apakah kode barang tersebut belum ada dengan memanggil fungsi is_barang_exists(). Jika status bernilai true maka kode barang sudah ada, sebaliknya jika status bernilai false maka

kode barang belum ada dan akan dilanjutkan dengan input nama barang dan harga barang. Kode barang, nama barang, harga barang masing masing akan ditambahkan ke array `kode_barang[$n]`, `nama_barang[$n]`, `harga_barang[$n]`. `n` adalah index yang diambil dari panjang array `kode_barang`.

```
function search() {
    status=false

    read -p "Kode barang yang ingin dicari : " kode_barang_in
    echo ""

    is_barang_exists

    if [[ $status == true ]]
    then
        echo "Kode Barang      : ${kode_barang[$index]}"
        echo "Nama Barang       : ${nama_barang[$index]}"
        echo "Harga Barang        : ${harga_barang[$index]}"

        echo -e "\nBarang berhasil ditemukan..\n"
    else
        echo -e "Barang tidak ditemukan..\n"
    fi
}
```

Fungsi `search()` digunakan untuk mencari barang dengan menginputkan kode barang terlebih dahulu lalu dicek apakah kode barang tersebut sudah ada atau belum dengan memanggil fungsi `is_barang_exists()`. Jika kode barang ada yang berarti status bernilai `true` maka program akan menampilkan kode barang, nama barang, dan harga barang yang dicari. Jika status bernilai `false` maka barang yang dicari tidak ada atau tidak ditemukan.

```
function show() {
    echo -e "Barang yang tersedia..\n"

    for (( i=0;i<${#kode_barang[@]};i++ ))
    do
        echo "Kode Barang      : ${kode_barang[$i]}"
        echo "Nama Barang       : ${nama_barang[$i]}"
        echo "Harga Barang        : ${harga_barang[$i]}"
        echo ""
    done
}
```

Fungsi show() digunakan untuk menampilkan semua barang yang telah diinputkan oleh user sebelumnya. Fungsi ini diawali menggunakan looping untuk mengambil nilai kode barang, nama barang, dan harga barang yang dimulai dari indeks ke nol sampai panjang array kode_barang.

```
function delete() {
    status=false

    read -p "Kode barang yang ingin dihapus : " kode_barang_in
    echo ""

    is_barang_exists

    if [[ $status == true ]]
    then
        unset kode_barang[$index]
        unset nama_barang[$index]
        unset harga_barang[$index]

        kode_barang=( "${kode_barang[@]}" )
        nama_barang=( "${nama_barang[@]}" )
        harga_barang=( "${harga_barang[@]}" )

        echo -e "Barang telah dihapus..\n"
    else
        echo -e "Barang tidak ditemukan..\n"
    fi
}
```

Fungsi delete digunakan untuk menghapus barang dari array. Sama seperti fungsi fungsi sebelumnya fungsi ini akan mengecek apakah kode barang yang diinput oleh user untuk dihapus sudah ada atau belum dengan memanggil fungsi is_barang_exists(). Jika kode barang tersebut ada yang artinya status bernilai true maka kode barang, nama barang, dan harga barang pada indeks tersebut akan dihapus dengan perintah unset nama_array[\$i]. setelah itu akan dilakukan reindex array dengan perintah nama_array=("\${nama_array[@]}"). Reindex ini digunakan untuk menggeser indeks array yang kosong (akibat ada data yang dihapus pada indeks ke-i) dengan indeks array yang berikutnya. Sebaliknya jika status bernilai false maka artinya barang yang dicari tidak ada.

```
function update() {
    status=false
```

```

read -p "Kode barang yang ingin diperbarui : " kode_barang_in
echo ""

is_barang_exists

if [[ $status == true ]]
then
    read -p "Kode Barang : " kode_barang[$index]
    read -p "Nama Barang : " nama_barang[$index]
    read -p "Harga Barang : " harga_barang[$index]

    echo -e "\nBarang berhasil diperbarui..\n"
else
    echo -e "Barang tidak ditemukan..\n"
fi
}

```

Fungsi update digunakan untuk meng-update data barang. Fungsi ini akan mengecek apakah kode barang yang dicari sudah ada atau tidak dengan fungsi `is_barang_exists()`. Jika ternyata kode barang ada atau status bernilai `true` maka akan dilakukan update kode barang, nama barang, dan harga barang pada indeks tersebut. Namun jika kode barang tidak ada maka akan menampilkan teks barang tidak ditemukan.

```

function print_out() {
    status=false

    is_folder_exists

    read -p "Kode laporan : " kode_laporan
    echo ""

    is_laporan_exists

    if [[ $status == true ]]
    then
        echo "Laporan Penjualan $kode_laporan" >> ./laporan/$kode_laporan.txt
        echo $date >> ./laporan/$kode_laporan.txt
        echo "" >> ./laporan/$kode_laporan.txt

        total=0

        for (( i=0;i<${#kode_barang[@]};i++ ))
        do
            echo "Kode Barang : ${kode_barang[$i]}" >> ./laporan/$kode_lapor
an.txt

```

```

        echo "Nama Barang   : ${nama_barang[$i]}" >> ./laporan/$kode_lapor
an.txt
        echo "Harga Barang  : ${harga_barang[$i]}" >> ./laporan/$kode_lapo
ran.txt
        echo "" >> ./laporan/$kode_laporan.txt
        let total+=${harga_barang[$i]}
    done

    echo "Total Harga     : $total" >> ./laporan/$kode_laporan.txt
    echo "" >> ./laporan/$kode_laporan.txt

    cat ./laporan/$kode_laporan.txt

    unset kode_barang
    unset nama_barang
    unset harga_barang
else
    echo -e "Laporan sudah ada..\n"
fi
}

```

Fungsi print_out() akan menghasilkan file.txt yang berupa laporan penjualan dengan nama kode_laporan.txt di dalam folder laporan. Fungsi ini diawali dengan mengecek folder laporan sudah ada atau belum dengan fungsi is_folder_exists(). Jika folder laporan belum ada maka akan dibuat folder tersebut. Kemudian akan meminta input berupa kode laporan untuk nama file.txt-nya. Setelah itu mengecek apakah file.txt dengan nama kode laporan yang telah diinputkan sebelumnya sudah ada atau belum. Jika belum ada yang artinya status bernilai true maka akan memasukkan teks “Laporan penjualan \$kode_laporan” dan “\$date” (tanggal hari saat itu) ke dalam file kode_laporan.txt yang berada di dalam folder laporan. Lalu ada inisialisasi total dengan nol dan dilakukan looping sebanyak array kode_barang untuk memasukkan data kode barang, nama barang, harga barang secara urut mulai indeks ke nol ke dalam file kode_laporan.txt, jika loop sudah berakhir maka akan memasukkan teks “Total harga : \$total” ke dalam file kode_laporan.txt juga. Yang terakhir ialah menampilkan isi dari laporan yang bernama kode_laporan.txt dengan perintah cat ./laporan/\$kode_laporan.txt dan menghapus array kode_barang, nama_barang, dan harga_barang dengan perintah unset nama_array.

```

function menu() {
    date=`date +"%b %d, %Y %I:%M %p"`
    echo ""
    echo "=====

```



```

echo "| Merchant Management System |"
echo "|   $date   |"
echo "=====|
echo "| Menu :                |"
echo "| 1. Tambah Barang      |"
echo "| 2. Cari Barang        |"
echo "| 3. Lihat Barang       |"
echo "| 4. Hapus Barang       |"
echo "| 5. Perbarui Barang    |"
echo "| 6. Cetak Laporan      |"
echo "| 7. Keluar             |"
echo "=====|
echo ""
read -p "Masukkan pilihan : " pilihan
echo ""

case "$pilihan" in
    1)
        insert
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    2)
        search
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    3)
        show
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    4)
        delete
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    5)
        update
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    6)
        print_out
        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
        ;;
    7)
        exit 0
        ;;
    *)
        echo -e "Pilihan salah\n"

```

```

        read -p "Tekan tombol apa saja untuk melanjutkan.." hold
        clear; menu
    ;;
esac
}

```

Fungsi menu() adalah fungsi terakhir yang dideklarasikan pada program merchant management system ini. Fungsi menu ini digunakan untuk menampilkan menu yang tersedia di dalam program, menu yang tersedia diantaranya :

- | | |
|------------------|--------------------|
| 1. Tambah barang | 5. Perbarui barang |
| 2. Cari barang | 6. Cetak laporan |
| 3. Lihat barang | 7. Keluar |
| 4. Hapus barang | |

Kemudian mengambil input dari user berupa no menu yang dipilih dan dilakukan pengecekan dengan switch case. Dengan begitu program akan menyeleksi no menu yang dipilih untuk dilakukan proses selanjutnya yaitu memanggil fungsi sesuai dengan yang tertera di source code di atas. Jika user memilih menu no 1, 2, 3, 4, 5, 6, dan 7 diakhir pemanggilan fungsi program akan selalu memanggil fungsi menu() lagi. Jika user memasukkan angka selain 1, 2, 3, 4, 5, 6, dan 7 maka akan masuk ke default yang tetap akan menampilkan teks “Pilihan salah” lalu memanggil fungsi menu() lagi untuk input menu. Jika user memasukkan angka 7 maka program akan berhenti atau keluar karena di dalam case 7 itu dideklarasikan perintah exit 0. Perintah clear digunakan untuk membersihkan output sebelumnya pada terminal sehingga saat memanggil fungsi menu() lagi tampilannya terlihat bersih atau clear.

C. PEMANGGILAN FUNGI MENU

```

menu

```

Fungsi menu() dipanggil agar program menampilkan menu yang tersedia dari program merchant management system ini saat program ini pertama kali dijalankan sebenarnya pemanggilan fungsi menu ini mirip seperti fungsi main pada kebanyakan program karena ini merupakan inti dari jalannya program ini.

OUTPUT PROGRAM

Menjalankan program menggunakan perintah bash nama_file.sh pada terminal Linux.

```
=====
| Merchant Management System |
|   Des 19, 2020 09:44   |
=====
| Menu :                  |
| 1. Tambah Barang      |
| 2. Cari Barang        |
| 3. Lihat Barang       |
| 4. Hapus Barang       |
| 5. Perbarui Barang    |
| 6. Cetak Laporan      |
| 7. Keluar             |
=====

Masukkan pilihan :
```

Gambar 1 : Tampilan menu utama

Menu Tambah Barang. Sebagai contoh, ditambahkan 3 barang yang berbeda.

```
=====
| Merchant Management System |
|   Des 19, 2020 09:44   |
=====
| Menu :                  |
| 1. Tambah Barang      |
| 2. Cari Barang        |
| 3. Lihat Barang       |
| 4. Hapus Barang       |
| 5. Perbarui Barang    |
| 6. Cetak Laporan      |
| 7. Keluar             |
=====

Masukkan pilihan : 1

Kode barang yang ingin ditambahkan : 001

Barang belum ada, lanjut..

Kode Barang : 001
Nama Barang : Sabun Mandi
Harga Barang : 5000

Barang berhasil ditambahkan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 2 : Menambahkan barang ke-1

```
=====
| Merchant Management System |
|   Des 19, 2020 09:49   |
=====
| Menu :                  |
| 1. Tambah Barang       |
| 2. Cari Barang         |
| 3. Lihat Barang        |
| 4. Hapus Barang        |
| 5. Perbarui Barang     |
| 6. Cetak Laporan       |
| 7. Keluar              |
=====

Masukkan pilihan : 1

Kode barang yang ingin ditambahkan : 002

Barang belum ada, lanjut..

Kode Barang   : 002
Nama Barang   : Sikat Gigi
Harga Barang  : 15000

Barang berhasil ditambahkan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 3 : Menambahkan barang ke-2

```
=====
| Merchant Management System |
|   Des 19, 2020 09:53   |
=====
| Menu :                  |
| 1. Tambah Barang       |
| 2. Cari Barang         |
| 3. Lihat Barang        |
| 4. Hapus Barang        |
| 5. Perbarui Barang     |
| 6. Cetak Laporan       |
| 7. Keluar              |
=====

Masukkan pilihan : 1

Kode barang yang ingin ditambahkan : 003

Barang belum ada, lanjut..

Kode Barang   : 003
Nama Barang   : Beras
Harga Barang  : 55000

Barang berhasil ditambahkan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 4 : Menambahkan barang ke-3

Menu Cari Barang.

```
=====
| Merchant Management System |
|   Des 19, 2020 09:55     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 2

Kode barang yang ingin dicari : 002

Kode Barang   : 002
Nama Barang   : Sikat Gigi
Harga Barang   : 15000

Barang berhasil ditemukan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 5 : Mencari barang dengan kode barang 002 (Input benar atau kode tersebut tersedia)

```
=====
| Merchant Management System |
|   Des 19, 2020 09:57     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 2

Kode barang yang ingin dicari : 005

Barang tidak ditemukan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 6 : Mencari barang dengan kode barang 005 (Input salah atau kode tersebut tidak tersedia)

Menu Lihat Barang.

```
=====
| Menu :                               |
| 1. Tambah Barang                    |
| 2. Cari Barang                     |
| 3. Lihat Barang                     |
| 4. Hapus Barang                     |
| 5. Perbarui Barang                 |
| 6. Cetak Laporan                   |
| 7. Keluar                           |
|                                     |
=====

Masukkan pilihan : 3

Barang yang tersedia..

Kode Barang   : 001
Nama Barang   : Sabun Mandi
Harga Barang  : 5000

Kode Barang   : 002
Nama Barang   : Sikat Gigi
Harga Barang  : 15000

Kode Barang   : 003
Nama Barang   : Beras
Harga Barang  : 55000

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 7 : Menampilkan barang atau lihat barang yang telah ditambahkan sebelumnya

Menu Hapus Barang.

```
=====
| Merchant Management System          |
|      Des 19, 2020 10:01             |
|                                     |
=====
| Menu :                               |
| 1. Tambah Barang                    |
| 2. Cari Barang                     |
| 3. Lihat Barang                     |
| 4. Hapus Barang                     |
| 5. Perbarui Barang                 |
| 6. Cetak Laporan                   |
| 7. Keluar                           |
|                                     |
=====

Masukkan pilihan : 4

Kode barang yang ingin dihapus : 002

Barang telah dihapus..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 8 : Menghapus barang dengan kode barang 002

```
=====
| Merchant Management System |
|   Des 19, 2020 10:02     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 3

Barang yang tersedia..

Kode Barang   : 001
Nama Barang   : Sabun Mandi
Harga Barang  : 5000

Kode Barang   : 003
Nama Barang   : Beras
Harga Barang  : 55000

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 9 : Melihat barang setelah dihapus

```
=====
| Merchant Management System |
|   Des 19, 2020 10:03     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 4

Kode barang yang ingin dihapus : 005

Barang tidak ditemukan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 10 : Menghapus barang dengan kode barang 005 (Input salah atau kode tersebut tidak tersedia)

Menu Perbarui Barang.

```
=====
| Merchant Management System |
|   Des 19, 2020 10:05     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 5

Kode barang yang ingin diperbarui : 003

Kode Barang   : 002
Nama Barang   : Mie Goreng
Harga Barang  : 3000

Barang berhasil diperbarui..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 11 : Memperbarui barang dengan kode barang 003

```
=====
| Merchant Management System |
|   Des 19, 2020 10:06     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 3

Barang yang tersedia..

Kode Barang   : 001
Nama Barang   : Sabun Mandi
Harga Barang  : 5000

Kode Barang   : 002
Nama Barang   : Mie Goreng
Harga Barang  : 3000

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 12 : Melihat barang setelah diperbarui


```
=====
| Merchant Management System |
|   Des 19, 2020 10:08     |
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 5

Kode barang yang ingin diperbarui : 005

Barang tidak ditemukan..

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 13 : Memperbarui barang dengan kode barang 005 (Input salah atau kode tersebut tidak tersedia)

Menu Cetak Laporan.

```
=====
| Menu :                   |
| 1. Tambah Barang        |
| 2. Cari Barang          |
| 3. Lihat Barang         |
| 4. Hapus Barang         |
| 5. Perbarui Barang      |
| 6. Cetak Laporan        |
| 7. Keluar               |
=====

Masukkan pilihan : 6

Kode laporan : 01

Laporan Penjualan 01
Des 19, 2020 10:10

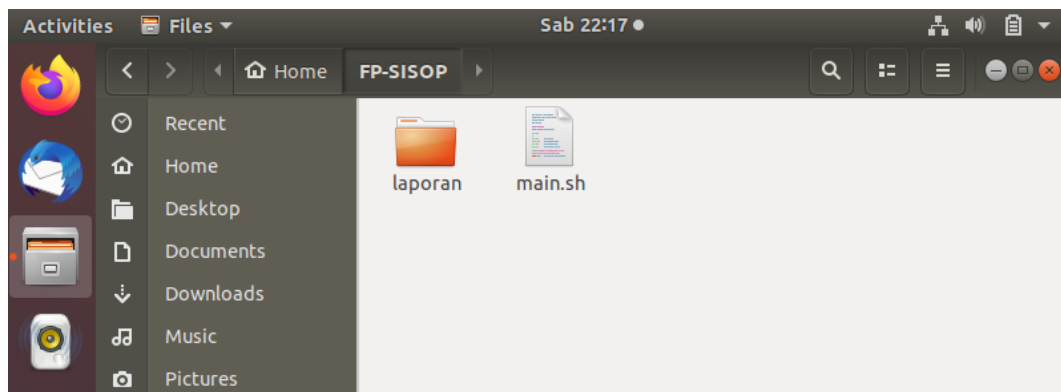
Kode Barang   : 001
Nama Barang   : Sabun Mandi
Harga Barang  : 5000

Kode Barang   : 002
Nama Barang   : Mie Goreng
Harga Barang  : 3000

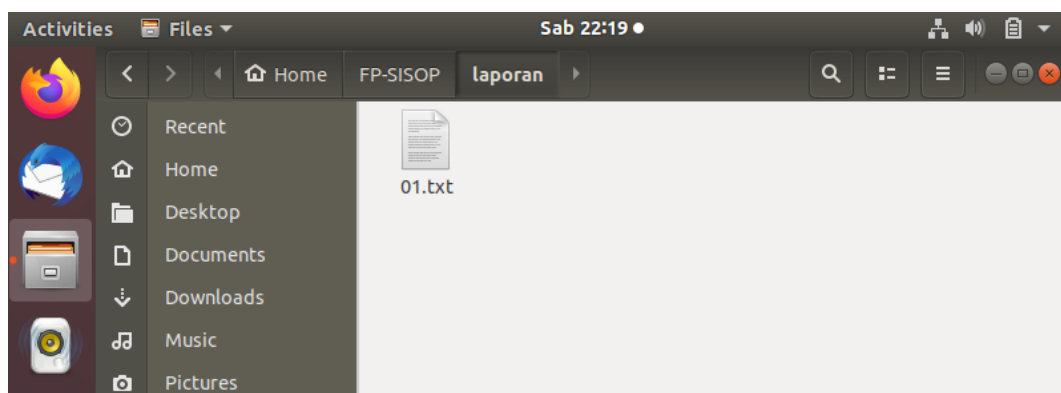
Total Harga   : 8000

Tekan tombol apa saja untuk melanjutkan..
```

Gambar 14 : Mencetak laporan dengan kode laporan 01



Gambar 15 : Dibuat folder baru yang bernama "laporan"



Gambar 16 : Folder "laporan" berisi file "01.txt"



Gambar 17 : Isi dari file yang bernama "01.txt"

Menu Keluar.

```
=====
| Merchant Management System |
|   Des 19, 2020 10:12   |
=====
| Menu :                  |
| 1. Tambah Barang       |
| 2. Cari Barang         |
| 3. Lihat Barang        |
| 4. Hapus Barang        |
| 5. Perbarui Barang     |
| 6. Cetak Laporan       |
| 7. Keluar              |
=====
Masukkan pilihan : 7
```

Gambar 18 : Keluar dari program