

LAPORAN PROYEK AKHIR PRAKTIKUM
MATA KULIAH ALGORITMA DAN PEMROGRAMAN
DASAR



MANAJEMEN PRODUK DAN PEMESANAN
WARUNG MAKAN ABL

Oleh:

Kelompok 7

MUHAMMAD FAIZ LAZUARDI	2409106031
MUHAMMAD NAUFAL	2409106034
RIFYAN ILHAM	
RANGGA ADITYA RAHMAN	2409106044

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MULAWARMAN
SAMARINDA 2024

KATA PENGANTAR

Puji syukur kami panjatkan kepada Allah SWT atas rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan proyek akhir praktikum ini. Laporan ini disusun untuk memenuhi tugas mata kuliah Algoritma dan Pemrograman Dasar.

Proyek yang kami buat adalah aplikasi manajemen pemesanan produk di Warung Makan ABL. Aplikasi ini dirancang untuk mengelola menu, pesanan, dan data pengguna baik untuk pelanggan maupun admin.

Dalam proses penyusunan laporan dan pembuatan program ini, kami menghadapi beberapa kendala seperti debugging kode dan validasi data, tetapi semua berhasil diatasi berkat bimbingan dari dosen, asisten, dan rekan-rekan satu kelompok.

Kami berharap laporan ini dapat bermanfaat bagi pembaca sebagai referensi di masa mendatang.

Samarinda, 22 November 2024

Kelompok 7

TAKARIR

<i>Database</i>	Basis Data
<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
<i>Output</i>	Keluaran
<i>Admin</i>	Pengelola
<i>Console</i>	Terminal
<i>Register</i>	Daftar
<i>Login</i>	Masuk
<i>Menu</i>	Daftar Pilihan
<i>Error Handling</i>	Penanganan Kesalahan
<i>Validation</i>	Validasi
<i>Index</i>	Indeks
<i>Function</i>	Fungsi
<i>Loop</i>	Perulangan
<i>Random</i>	Acak
<i>Table</i>	Tabel
<i>Progress</i>	Kemajuan
<i>Password</i>	Kata Sandi
<i>User</i>	Pengguna
<i>Role</i>	Peran
<i>Data Structure</i>	Struktur Data
<i>Command</i>	Perintah
<i>Error Message</i>	Pesan Kesalahan
<i>CSV</i>	Nilai Yang Dipisahkan Koma
<i>Module</i>	Modul
<i>Library</i>	Pustaka
<i>Exception</i>	Pengecualian
<i>Authentication</i>	Autentikasi
<i>Authorization</i>	Otorisasi
<i>Syntax</i>	Sintaks

DAFTAR ISI

KATA PENGANTAR	i
TAKARIR	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR	v
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Kebutuhan Fungsional.....	1
1.3 Rumusan Masalah	1
1.4 Batasan Masalah.....	1
1.5 Tujuan.....	1
BAB II PERANCANGAN.....	2
2.1 Analisis Program	2
2.2 Flowchart.....	3
2.3 Konsep/Materi Praktikum yang dipakai	8
BAB III HASIL DAN PEMBAHASAN.....	9
3.1 Tampilan Program	9
3.1.1 Tampilan Menu	9
3.1.2 Ss lainnya.	11
3.2 Source Code	18
BAB IV PENUTUP	38
4.1 Kesimpulan.....	38
4.2 Saran	38
DAFTAR PUSTAKA	39

DAFTAR GAMBAR

Gambar 1 Menu Login/Register	3
Gambar 2 Pengecekan Role Pada Akun & Menu Admin	4
Gambar 3 Menu Admin Mengatur Menu.....	5
Gambar 4 Menu Admin Mengatur Data Pengguna.....	6
Gambar 5 Menu Pengguna.....	7
Gambar 6 END.....	8
Gambar 3. 1.1 Menu awal	9
Gambar 3. 1.2 Menu Pengguna.....	9
Gambar 3. 1.3 Menu Admin Awal.....	10
Gambar 3. 1.4 Menu Admin Memilih Mengatur Menu	10
Gambar 3.2. 1 Loadin.....	11
Gambar 3.2. 2 Register Pengguna.....	11
Gambar 3.2. 3 Login Sebagai Pengguna	11
Gambar 3.2. 4 Melihat Menu	11
Gambar 3.2. 5 Menambahkan Pesanan	12
Gambar 3.2. 6 Mengubah Pesanan.....	12
Gambar 3.2. 7 Menghapus Pesanan	13
Gambar 3.2. 8 Melihat Pesanan	13
Gambar 3.2. 9 Login Sebagai Admin.....	13
Gambar 3.2. 10 Memilih Mengatur Menu Dan Memilih Liat Menu	14
Gambar 3.2. 11Memilih Menambahkan Menu	14
Gambar 3.2. 12 Mengubah Menu	15
Gambar 3.2. 13 Menghapus Menu	15
Gambar 3.2. 14 Melihat Menu Lagi Setelah Dihapus.....	15
Gambar 3.2. 15 Admin Memilih Mengatur Data Pengguna Dan Meliat Data Pengguna.....	16
Gambar 3.2. 16 Menambah Data Pengguna	16
Gambar 3.2. 17 Mengubah Data Pengguna	17
Gambar 3.2. 18 Menghapus Data Pengguna.....	17
Gambar 3.2. 19 Melihat Lagi Data Pengguna.....	17

Gambar 3.2. 20 Keluar Dari Program	17
Gambar 3. 1.5 Menu Admin Memilih Mengatur Data Pengguna.....	10
Gambar 7 Kartu Konsul	40
Gambar 8 Halaman Konsul Pertama.....	41
Gambar 9 Halaman Konsul Kedua	42

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi memberikan peluang untuk mengembangkan aplikasi berbasis console guna mendukung aktivitas sehari-hari. Proyek ini bertujuan mempermudah pengelolaan pemesanan di Warung Makan ABL dengan pendekatan otomatisasi yang sederhana.

1.2 Kebutuhan Fungsional

Sistem registrasi dan login.

Manajemen data menu oleh admin.

Pengelolaan pesanan oleh pelanggan.

Penyajian data dalam format tabel.

1.3 Rumusan Masalah

Bagaimana merancang aplikasi sederhana yang dapat memenuhi kebutuhan manajemen pemesanan dengan efisien?

1.4 Batasan Masalah

Aplikasi berbasis teks tanpa antarmuka grafis.

Data disimpan dalam file CSV..

1.5 Tujuan

Menghasilkan aplikasi manajemen pesanan berbasis console yang memudahkan pengelolaan menu dan pesanan.

BAB II

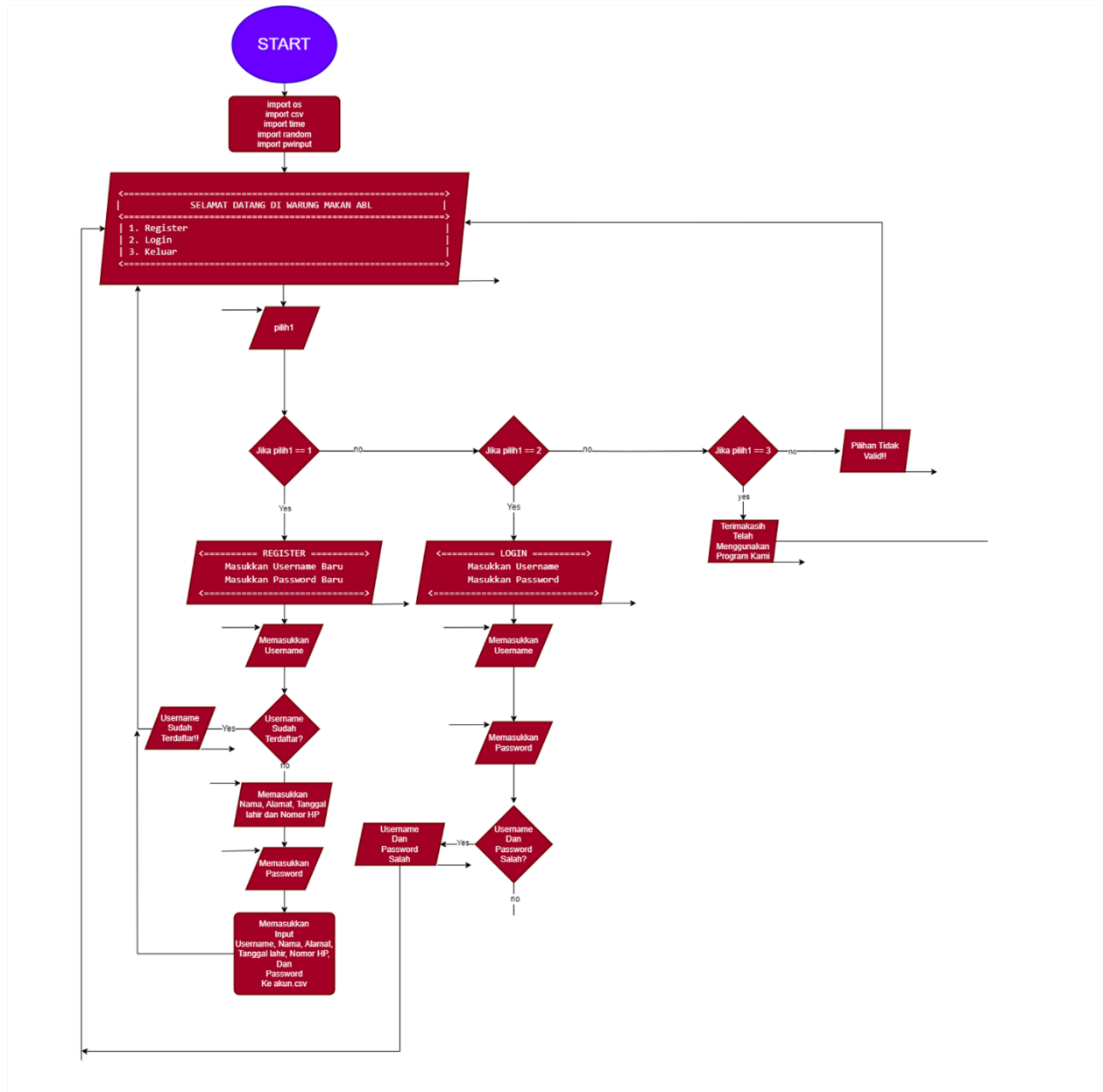
PERANCANGAN

2.1 Analisis Program

Aplikasi manajemen Warung Makan ABL dirancang untuk mengelola data pelanggan, menu, dan pesanan. Berikut adalah alur kerja program:

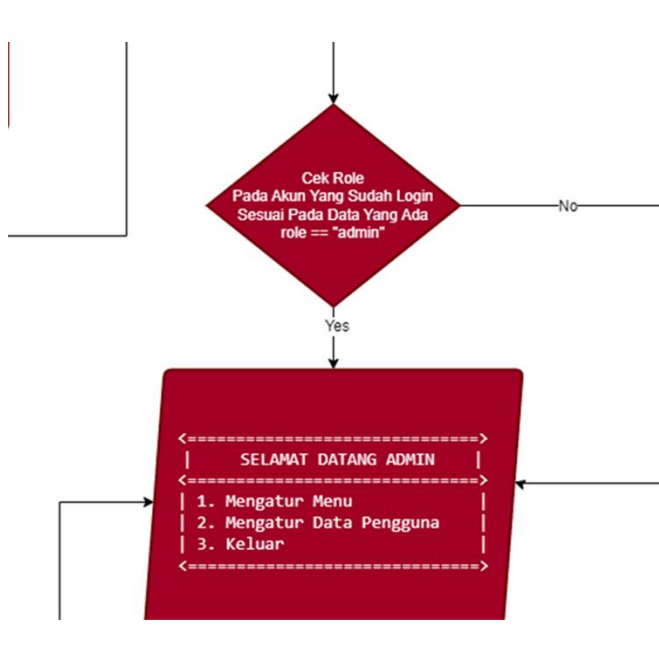
- Pengguna dapat mendaftar atau masuk sebagai admin atau pelanggan.
- Admin memiliki hak untuk menambah, mengubah, dan menghapus data menu serta pengguna.
- Pelanggan dapat melihat menu, membuat pesanan, mengubah, atau menghapus pesanan.
- Data disimpan dan diakses dari file CSV, memastikan integritas data saat program dihentikan.

2.2 Flowchart



Gambar 1 Menu Login/Register

Pada Awalan Flowchart User Memilih Diantara 3 Opsi Yaitu Register, Login, Dan Keluar. Memilih 1 Maka Register, Memilih 2 Maka Login, Memilih 3 Maka Keluar Dari Program. Pada Login Akun Yang Masuk Ke Program Akan Menuju Menu Tergantung Role-nya Yaitu Pengguna Atau Admin.

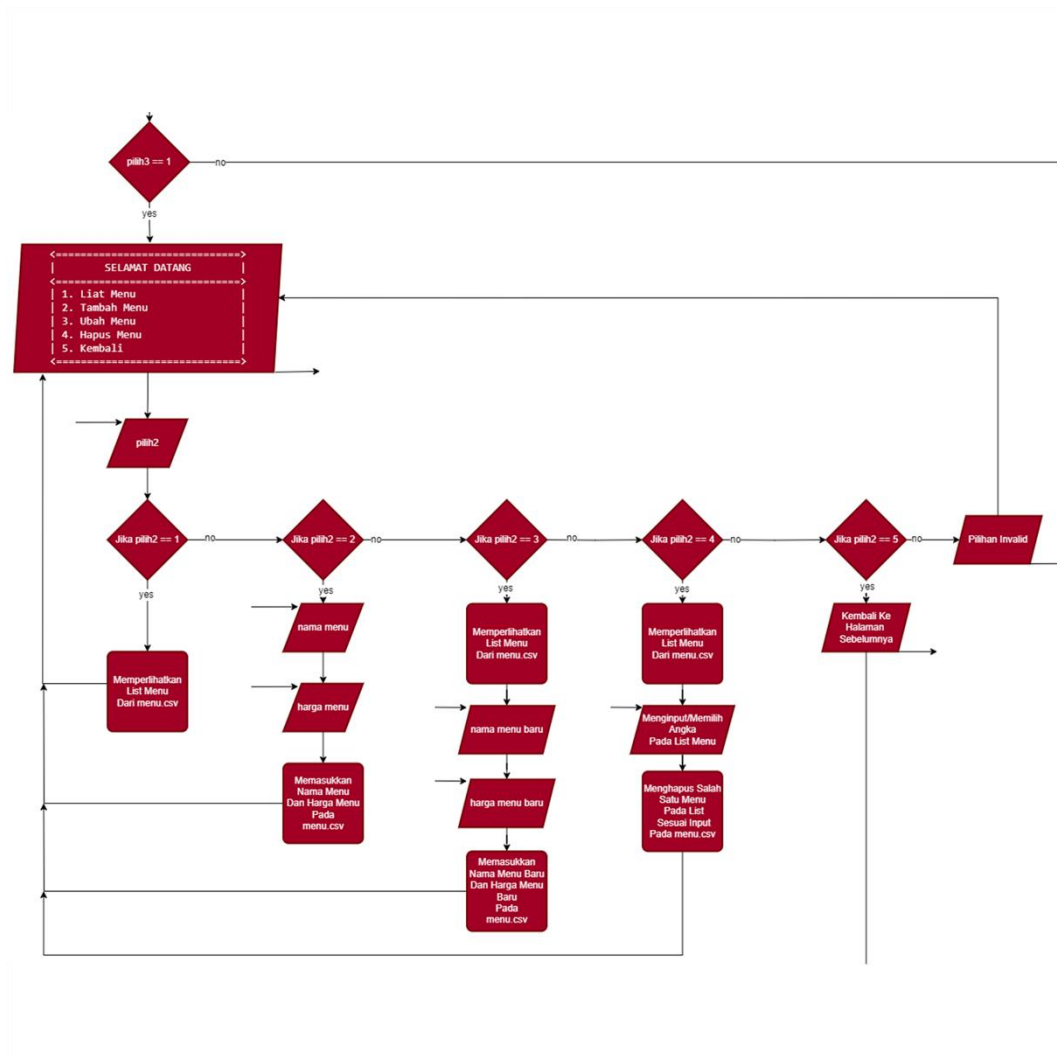


Gambar 2 Pengecekan Role Pada Akun & Menu Admin

Pada Gambar Flowchart Adalah Pengecekan Role Pada Akun Yang Masuk Ke Program, Jika Role Admin Maka Alur Ke Bawah Yaitu Menu Admin. Menu Admin Terdiri Dari 2 Fungsional Yaitu

1. Mengatur Menu = Admin Akan Mengatur Menu Produk Di Program Warung Makan ABL
2. Mengatur Data Pengguna = Admin Akan Mengatur Data Pengguna Yang Sudah Register
3. Keluar = Program Akan Kembali Ke Gambar 1

Pada Alur Selanjutnya Akan Memilih Opsi 1

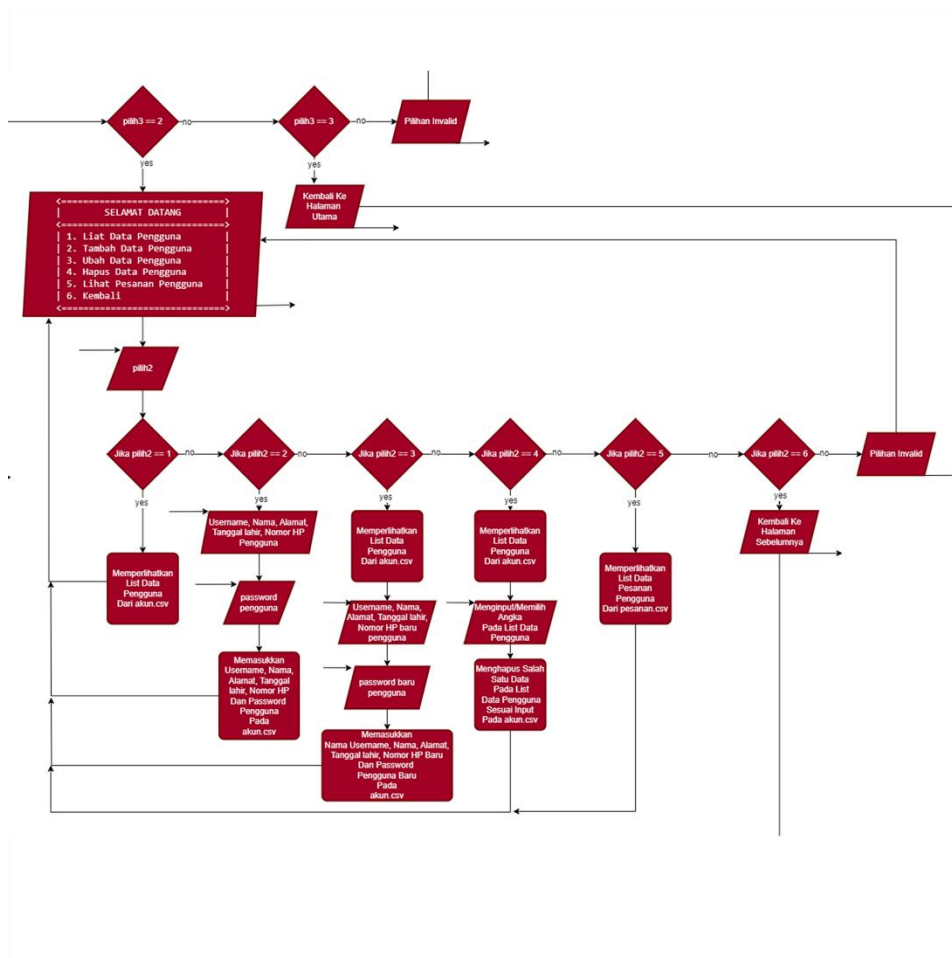


Gambar 3 Menu Admin Mengatur Menu

Pada Opsi Satu Untuk Memlih Menu Admin Yaitu Mengatur Menu.

Didalam Program Ini Admin Bisa Memilih Opsi Sebagai Berikut:

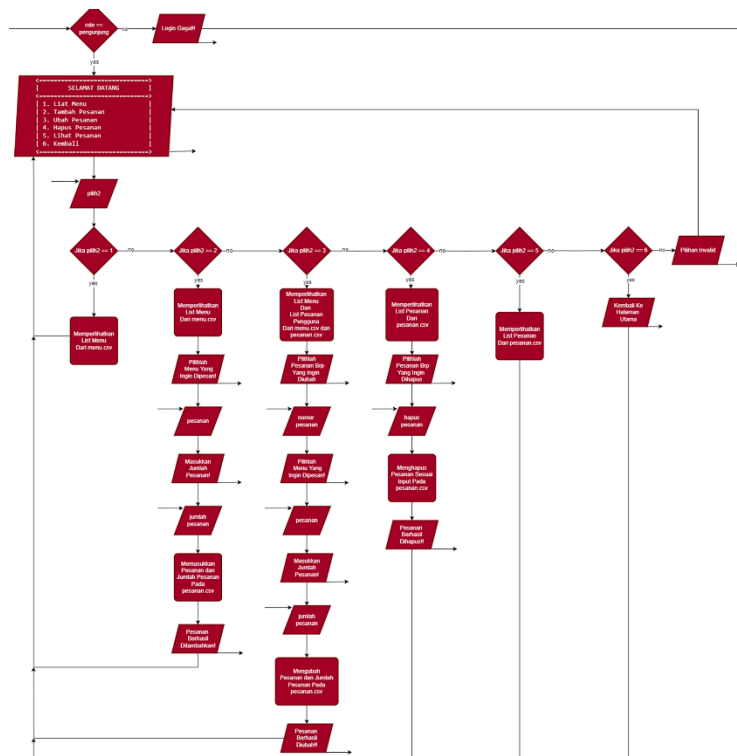
1. Liat Menu = Dapat Melihat Menu Yang Ada
2. Tambah Menu = Dapat Menambahkan Menu Baru
3. Ubah Menu = Dapat Mengubah Menu Yang Sudah Ada
4. Hapus Menu = Dapat Menghapus Salah Satu Menu Yang Sudah Ada
5. Kembali = Program Akan Kembali Ke Gambar 2



Gambar 4 Menu Admin Mengatur Data Pengguna

Pada Opsi Dua Untuk Memilih Menu Admin Yaitu Mengatur Data Pengguna, Didalam Program Ini Admin Bisa Memilih Opsi Sebagai Berikut:

1. Liat Data Pengguna = Dapat Melihat List Pengguna Yang Sudah Register
2. Tambah Data Pengguna = Dapat Menambah Data Pengguna Baru Tanpa Lewat Register
3. Ubah Data Pengguna = Dapat Mengubah Data Pengguna Yang Sudah Ada Sesuai Kebutuhan
4. Hapus Data Pengguna = Dapat Menghapus Salah Satu Data Pengguna Yang Sudah Ada Atau Register
5. Lihat Pesanan Pengguna = Dapat Melihat List Pesanan Pengguna
6. Kembali = Program Akan Kembali Ke Gambar 2



Gambar 5 Menu Pengguna

Ketika Pada Gambar ... Mempunyai Role Pengguna, Maka Program Akan Menuju Ke Menu Pengguna. Isi Program Pada Menu Pengguna Adalah Sebagai Berikut

1. Liat Menu = Pengguna Dapat Melihat List Menu Yang Sudah Dibuat Admin
2. Tambah Pesanan = Pengguna Dapat Menambah Atau Membuat Pesanan Sesuai Pada Menu Yang Ada
3. Ubah Pesanan = Pengguna Dapat Mengubah Pesanan Yang Sudah Dibuat
4. Hapus Pesanan = Pengguna Dapat Menghapus Pesanan Yang Ada
5. Lihat Pesanan = Pengguna Dapat Melihat Pesanan Yang Sudah Ada Atau Dibuat
6. Kembali = Program Akan Kembali Ke Gambar 1



Gambar 6 END

Ketika Pada Gambar 2.2.1 User Memilih Keluar

2.3 Konsep/Materi Praktikum yang dipakai

Konsep yang kami gunakan dalam pembuatan laporan ini sesuai dengan persyaratan yang terdapat pada modul diantaranya:

1. Fungsi dasar: fungsi yang kami gunakan adalah fungsi print.

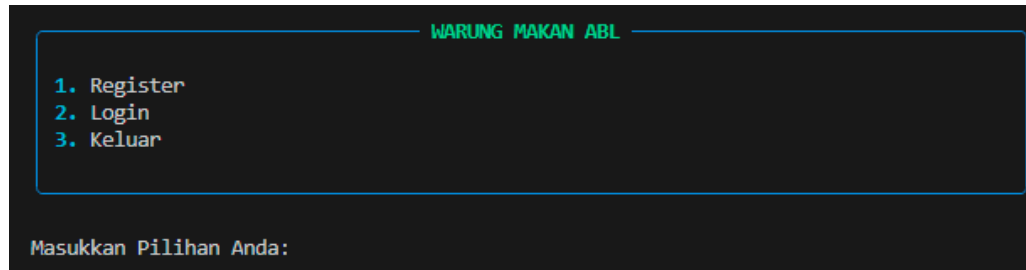
Fungsi print pada program merupakan fungsi yang umum dipakai untuk menampilkan suatu keluaran pada layar peraga.

BAB III

HASIL DAN PEMBAHASAN

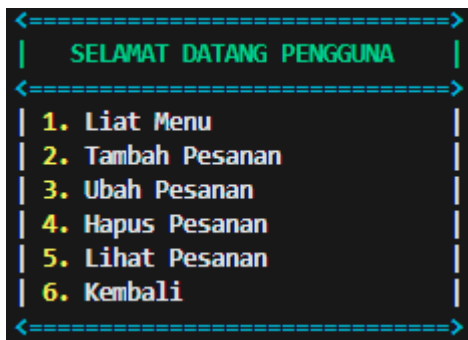
3.1 Tampilan Program

3.1.1 Tampilan Menu



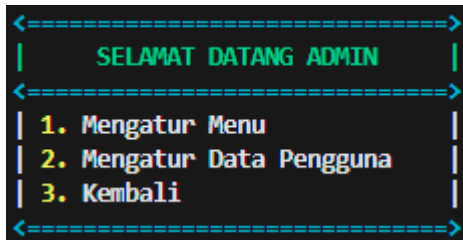
Gambar 3. 0.1 Menu awal

Pada Menu Ini Pengguna Memilih Opsi Register, Login Atau Keluar. Untuk Register Akun Yang Terbuat Mempunyai Role Pengguna. Ketika Login Menggunakan Nama Dan Password Admin Maka Akun Yang Masuk Akan Mempunyai Role Admin



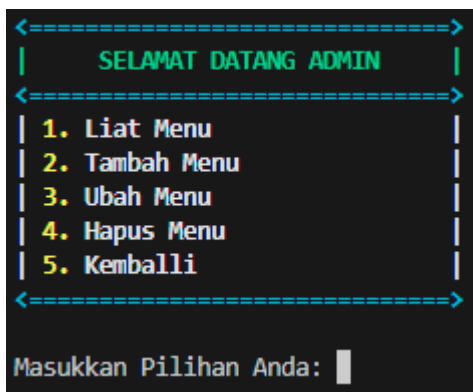
Gambar 3. 0.2 Menu Pengguna

Ketika Login Dan Rolenya Adalah Pengguna, Maka Akan Melihat Menu Ini Yaitu Menu Pengguna. Pada Menu Ini Pengguna Dapat Memakai Menu Ini Sesuai Kebutuhan.



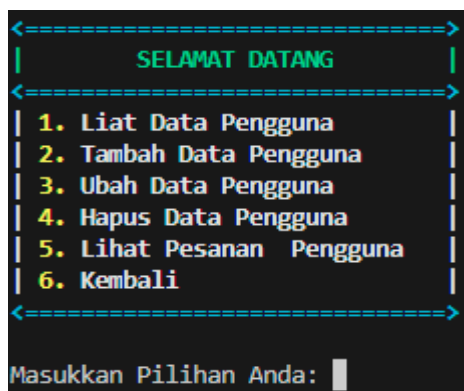
Gambar 3. 0.3 Menu Admin Awal

Ketika Login Dan Rolenya Adalah Admin, Maka Akan Melihat Menu Ini Yaitu Menu Admin. Pada Menu Ini Admin Dapat Mengatur 2 Menu Yaitu Mengatur Menu Dan Data Pengguna.



Gambar 3. 0.4 Menu Admin Memilih Mengatur Menu

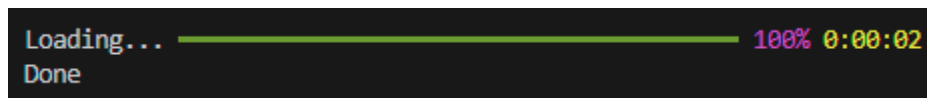
Admin Dapat Mengatur Menu Sesuai Kebutuhan Dengan Opsi Yang Sudah Ada.



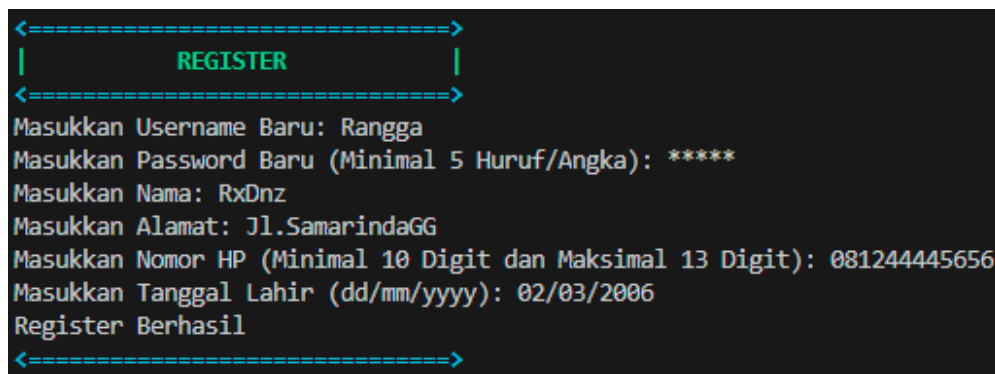
Gambar 3. 0.5 Menu Admin Memilih Mengatur Data Pengguna

Admin Dapat Mengatur Data Pengguna Sesuai Kebutuhan Dengan Opsi Yang Sudah Ada.

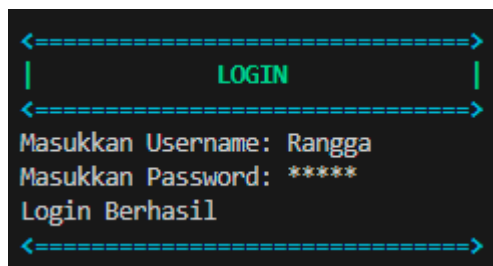
3.1.2 Ss lainnya.



Gambar 3.2. 1 Loadin



Gambar 3.2. 2 Register Pengguna



Gambar 3.2. 3 Login Sebagai Pengguna



Gambar 3.2. 4 Melihat Menu

```

<=====>
|  MENAMBAHKAN PESANAN  |
<=====>

  Menu Makanan



| No | Nama Makanan | Harga     |
|----|--------------|-----------|
| 1  | lele         | Rp.10.000 |
| 2  | bebek        | Rp.25.000 |
| 3  | ayam         | Rp.12.000 |
| 4  | nila         | Rp.20.000 |



Masukkan Nomor Menu Yang Ingin Dipesan: 1
Masukkan Jumlah Pesanan: 3
Pesanan Berhasil Ditambahkan
  
```

Gambar 3.2. 5 Menambahkan Pesanan

```

<=====>
|  MENGUBAH PESANAN  |
<=====>

  List Pesanan



| No | Nama   | Pesanan | Jumlah Pesanan |
|----|--------|---------|----------------|
| 1  | Rangga | lele    | 3              |



Masukkan Nomor Menu Yang Ingin Diubah: 1
  Menu Makanan

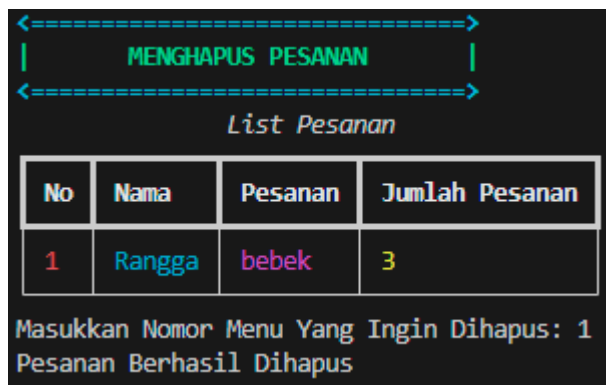


| No | Nama Makanan | Harga     |
|----|--------------|-----------|
| 1  | lele         | Rp.10.000 |
| 2  | bebek        | Rp.25.000 |
| 3  | ayam         | Rp.12.000 |
| 4  | nila         | Rp.20.000 |



Masukkan Nomor Menu Baru: 2
Masukkan Jumlah Pesanan: 3
Pesanan Berhasil Diubah
  
```

Gambar 3.2. 6 Mengubah Pesanan

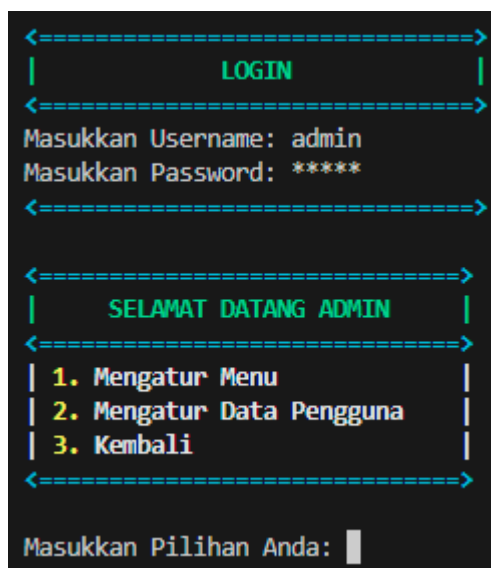


Gambar 3.2. 7 Menghapus Pesanan

List Pesanan

No	Nama	Pesanan	Jumlah Pesanan
-	Tidak ada data	-	-

Gambar 3.2. 8 Melihat Pesanan



Gambar 3.2. 9 Login Sebagai Admin

Menu Makanan		
No	Nama Makanan	Harga
1	lele	Rp.10.000
2	bebek	Rp.25.000
3	ayam	Rp.12.000
4	nila	Rp.20.000


```

<=====>
|  SELAMAT DATANG ADMIN  |
<=====>
|  1. Liat Menu          |
|  2. Tambah Menu       |
|  3. Ubah Menu         |
|  4. Hapus Menu        |
|  5. Kembali           |
|                        |
<=====>
Masukkan Pilihan Anda: 

```

Gambar 3.2. 10 Memilih Mengatur Menu Dan Memilih Liat Menu

```

<=====>
|  MENAMBAHKAN MENU      |
<=====>

Masukkan Nama Menu: Nasi
Masukkan Harga Menu: Rp.20000
Menu Berhasil Ditambahkan

<=====>

<=====>
|  SELAMAT DATANG ADMIN  |
<=====>
|  1. Liat Menu          |
|  2. Tambah Menu       |
|  3. Ubah Menu         |
|  4. Hapus Menu        |
|  5. Kembali           |
|                        |
<=====>
Masukkan Pilihan Anda: 

```

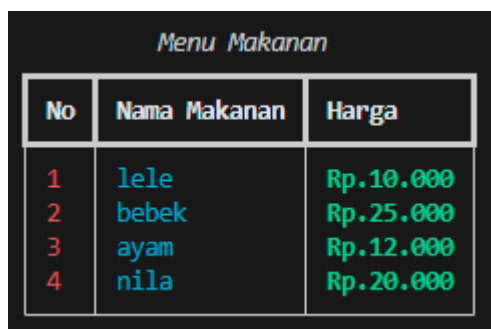
Gambar 3.2. 11Memilih Menambahkan Menu



Gambar 3.2. 12 Mengubah Menu



Gambar 3.2. 13 Menghapus Menu



Gambar 3.2. 14 Melihat Menu Lagi Setelah Dihapus

List Akun						
No	Username	Password Akun	Nama	Alamat	No.Hp	Tanggal Lahir
1	rangga	pwrangga	namaadit	jl.samarin...	081244444444	02/03/2006
2	Rangga	12345	RxDnz	Jl.Samarin...	081244445656	02/03/2006


```

<=====>
|          SELAMAT DATANG          |
<=====>
| 1. Liat Data Pengguna             |
| 2. Tambah Data Pengguna           |
| 3. Ubah Data Pengguna             |
| 4. Hapus Data Pengguna            |
| 5. Lihat Pesanan Pengguna         |
| 6. Kembali                       |
<=====>
Masukkan Pilihan Anda: █

```

Gambar 3.2. 15 Admin Memilih Mengatur Data Pengguna Dan Meliat Data Pengguna

```

<=====>
|          MENAMBAHKAN DATA          |
<=====>

Masukkan Username Baru Pengguna: Faiz
Masukkan Password Baru Pengguna: *****
Masukkan Nama Baru Pengguna: Faized
Masukkan Alamat Baru Pengguna : jl...
Masukkan No.HP Baru Pengguna (Minimal 10 Digit Dan Maksimal 13 Digit): 08241342156
Masukkan Tanggal Lahir Baru Pengguna (dd/mm/yyyy): 22/06/1999
Data Berhasil Ditambahkan

```

Gambar 3.2. 16 Menambah Data Pengguna

MENGUBAH DATA

List Akun

No	Username	Password Akun	Nama	Alamat	No.Hp	Tanggal Lahir
1	rangga	pwrangga	namaadit	jl.samarin...	081244444444	02/03/2006
2	Rangga	12345	RxDnz	Jl.Samarin...	081244445656	02/03/2006
3	Faiz	12345	Faized	jl...	08241342156	22/06/1999

Masukkan Nomor Data Yang Ingin Diubah: 3

Masukkan Username Baru: faiz

Masukkan Password Baru: *****

Masukkan Nama Baru: faized

Masukkan Alamat Baru: jl....

Masukkan No.HP Baru (Minimal 10 Digit Dan Maksimal 13 Digit): 089826265353

Masukkan Tanggal Lahir Baru (dd/mm/yyyy): 02/12/1999

Data Berhasil Diubah

Gambar 3.2. 17 Mengubah Data Pengguna

MENGHAPUS DATA

List Akun

No	Username	Password Akun	Nama	Alamat	No.Hp	Tanggal Lahir
1	rangga	pwrangga	namaadit	jl.samarin...	081244444444	02/03/2006
2	Rangga	12345	RxDnz	Jl.Samarin...	081244445656	02/03/2006
3	faiz	54321	faized	jl....	089826265353	02/12/1999

Masukkan Nomor Data Yang Ingin Dihapus: 3

Data Berhasil Dihapus

Gambar 3.2. 18 Menghapus Data Pengguna

List Akun

No	Username	Password Akun	Nama	Alamat	No.Hp	Tanggal Lahir
1	rangga	pwrangga	namaadit	jl.samarin...	081244444444	02/03/2006
2	Rangga	12345	RxDnz	Jl.Samarin...	081244445656	02/03/2006

Gambar 3.2. 19 Melihat Lagi Data Pengguna

TERIMA KASIH TELAH BERKUNJUNG

Gambar 3.2. 20 Keluar Dari Program

3.2 Source Code

```
import os # import os digunakan untuk menggunakan system operasi
import csv # import csv digunakan untuk membaca, membuat, dan menulis file csv
import time # import time digunakan untuk memberi waktu/jeda pada fungsi loading
import random # import random digunakan untuk menghasilkan nilai random
import pwinput # import pwinput digunakan untuk menyembunyikan input pada password
from rich.table import Table # import Table dari library Rich untuk membantu pembuatan Table
from rich.console import Console # import Console dari library Rich digunakan untuk membantu memproses dan
menampilkan prosedur dari library rich ke terminal
from rich.progress import track # import track dari Library Rich digunakan untuk membantu proses loading
from rich.text import Text # import Text dari Library Rich digunakan untuk memberikan warna pada teks di terminal
from rich.panel import Panel # import Panel dari Library Rich digunakan untuk membungkus Teks didalam border
from rich.align import Align # import Align Dari Library Rich digunakan untuk mengatur posisi Teks

console = Console()

admin = {
    "username": "admin",
    "password": "admin",
}

# ===== [ ERROR HANDLING ] =====

# Merupakan Fungsi untuk menghapus tampilan pada Terminal
def clear():
    os.system("cls" if os.name == "nt" else "clear")

# Fungsi ini digunakan untuk memeriksa error pada variable yang mengandung teks
def error_teks(teks):
    while True:
        try: # dicoba terlebih dahulu
            input_user = teks
            if not input_user:
                raise ValueError("Input tidak boleh kosong") # 'raise ValueError' digunakan untuk mengganti output
yang muncul ketika terjadi error pada value

            elif input_user.isspace():
                raise ValueError("Input tidak boleh termasuk spasi") # 'raise ValueError' digunakan untuk
mengganti output yang muncul ketika terjadi error pada value

            elif input_user.isdigit():
                raise ValueError("Input tidak boleh termasuk angka") # 'raise ValueError' digunakan untuk
mengganti output yang muncul ketika terjadi error pada value

        else:
            break # untuk memberhentikan perulangan

    except ValueError as e: # terjadi jika value tidak tepat
        console.print(f"[red]\nTerjadi Error pada {e}")
        error = "404"
        return error

# Fungsi ini digunakan untuk memeriksa error pada variable yang mengandung angka
def error_nomor(teks):
    while True:
        try: # dicoba terlebih dahulu
            input_user = teks
            if not input_user:
```



```

        raise ValueError("Input tidak boleh kosong") # 'raise ValueError' digunakan untuk mengganti output
yang muncul ketika terjadi error pada value

        elif input_user.isspace():
            raise ValueError("Input tidak boleh termasuk spasi") # 'raise ValueError' digunakan untuk
mengganti output yang muncul ketika terjadi error pada value

        elif input_user.isalpha():
            raise ValueError("Input tidak boleh termasuk teks") # 'raise ValueError' digunakan untuk mengganti
output yang muncul ketika terjadi error pada value

        else:
            break # untuk memberhentikan perulangan

    except ValueError as e: # terjadi jika value tidak tepat
        console.print(f"[red]\nTerjadi Error pada {e}")
        error = "404"
        return error

# ===== [ RICH TABLE ] =====

# Fungsi ini digunakan untuk memberikan efek blink pada teks
def blink(text):
    colour = Text(text)

    for i in range(2):
        print((colour), end="\r", flush=True)
        time.sleep(0.5)
        print(f" " * len(colour), end="\r", flush=True)
        time.sleep(0.5)
    print(f"\r{colour}  ")

# Prosedur ini digunakan pada fungsi table menu untuk menentukan format jumlah harga
def format_titik(angka):
    angka_str = str(angka)
    hasil = ""

    for i, digit in enumerate(reversed(angka_str)):
        if i > 0 and i % 3 == 0:
            hasil = "." + hasil
        hasil = digit + hasil
    return hasil

# Fungsi ini digunakan untuk menampilkan data pada pesanan.csv untuk dijadikan dalam sebuah table berisi daftar
pesanan
def table_pesanan():
    table = Table(title="List Pesanan")
    table.add_column("No", style="red")
    table.add_column("Nama", style="cyan")
    table.add_column("Pesanan", style="magenta")
    table.add_column("Jumlah Pesanan", style="yellow")

    try: # dicoba terlebih dahulu
        with open("../APD/pesanan.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            mark = False
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

            for index, line in enumerate(lines):
                mark = True
                nomor = index+1

```

```

        table.add_row(str(nomor) , line[0], line[1], line[2])

    if not mark:
        table.add_row("-", "Tidak ada data", "-", "-")

except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
    table.add_row("-", "File tidak ditemukan", "-", "-")

except ValueError as e: # terjadi jika value tidak tepat
    table.add_row(f"-", "Terjadi kesalahan pada {e}" , "-", "-")

try: # dicoba terlebih dahulu
    console = Console()
    console.print(table)

except UnboundLocalError:
    console.print("[red]Tidak Ada Pesanan!!")

# Fungsi ini digunakan untuk menampilkan data pada akun.csv untuk dijadikan dalam sebuah table berisi daftar akun
def table_akun() :
    table = Table(title="List Akun")
    table.add_column("No", style="white")
    table.add_column("Username", style="cyan")
    table.add_column("Password Akun", style="blue")
    table.add_column("Nama", style="magenta")
    table.add_column("Alamat", style="green")
    table.add_column("No.Hp", style="yellow")
    table.add_column("Tanggal Lahir", style="red")
    try: # dicoba terlebih dahulu
        with open("../APD/akun.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            mark = False
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

            for index, line in enumerate(lines):
                mark = True
                nomor = index+1
                table.add_row(str(nomor) ,line[0], line[1], line[2], line[3], line[4], line[5],)

                if not mark:
                    table.add_row("-", "Tidak ada data", "-", "-", "-", "-",)

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        table.add_row("-", "File tidak ditemukan", "-", "-", "-", "-",)

    except ValueError as e: # terjadi jika value tidak tepat
        table.add_row(f"-", "Terjadi kesalahan pada {e}" , "-", "-", "-", "-",)

    try: # dicoba terlebih dahulu
        console = Console()
        console.print(table)

    except UnboundLocalError:
        console.print("[red]Tidak Ada Pesanan!!")

# Fungsi ini digunakan untuk menampilkan data pada menu.csv untuk dijadikan dalam sebuah table berisi daftar menu
def table_menu():

    table = Table(title="Menu Makanan")
    table.add_column("No", style="red")
    table.add_column("Nama Makanan", style="cyan")

```

```

table.add_column("Harga", style="bold green")

try: # dicoba terlebih dahulu
    with open("../APD/menu.csv", "r") as file: # 'with open' digunakan untuk membuka file external
        mark = False
        lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        for index, line in enumerate(lines):
            mark = True
            nomor = index+1
            x = line[1]
            akhir = "Rp." + format_titik(x)
            table.add_row(str(nomor), line[0], akhir)

            if not mark:
                table.add_row("-", "Tidak ada data", "-")

except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
    table.add_row("-", "File tidak ditemukan", "-")

except ValueError as e: # terjadi jika value tidak tepat
    table.add_row(f"-", "Terjadi kesalahan pada {e}", "-")

try: # dicoba terlebih dahulu
    console = Console()
    console.print(table)

except UnboundLocalError:
    console.print("[red]Tidak Ada Pesanan!!")

# Fungsi ini digunakan untuk menampilkan loading dalam Terminal
def loading():
    clear()

    for i in track(range (random.randint(5, 12)), description="Loading..."):
        time.sleep(0.25)

    blink("Done")
    clear()

# ===== [ LOGIN/REGISTER ] =====

# Fungsi ini digunakan untuk mengecek username untuk menentukan username apakah sudah terdaftar atau belum
def cek_username(username):
    try: # dicoba terlebih dahulu
        with open("../APD/akun.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            reader = csv.reader(file) # sebagai variable untuk membaca file external

            for row in reader:
                if row[0] == username:
                    console.print("[red]Username Sudah Tendaftar")
                    return True

            return False

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")
        return False

# Fungsi ini digunakan untuk melakukan Register Pengunjung pada program

```

```

def register(username, password, nama, alamat, no_hp, tanggal_lahir, role = "Pengunjung"):
    try: # dicoba terlebih dahulu
        with open("./APD/akun.csv", "a", newline='') as file: # 'with open' digunakan untuk membuka file external
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerow([username, password, nama, alamat, no_hp, tanggal_lahir, role])
            print("Register Berhasil")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk melakukan Login pada program
def login(username, password):
    try: # dicoba terlebih dahulu
        if username == admin["username"] and password == admin["password"]: # pengecekan login
            aksesadmin = "ADMIN"
            return aksesadmin

        elif username == "username" and password == "password": # error handling
            return

        else:
            with open("./APD/akun.csv", "r") as file: # 'with open' digunakan untuk membuka file external
                reader = csv.reader(file) # 'reader' sebagai variable untuk membaca file external

                for row in reader:
                    if row[0] == username and row[1] == password:
                        print("Login Berhasil")
                        return row[6]

            console.print("[red]Username atau Password Salah")
            return None

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")
        return None

# Fungsi ini digunakan untuk memberhentikan program dan menampilkan pesan terimakasih menggunakan Rich Library
def keluar_dari_program():
    # Isi Teks Saat Memilih Keluar
    teks_keluar = "[bold white]TERIMA KASIH TELAH BERKUNJUNG"

    # Membuat Teks Di Posisi Tengah menggunakan Align Dari Library Rich
    tengah = Align(teks_keluar, align="center")

    # Panel untuk membungkus teks di dalam border
    panel = Panel(tengah, border_style="bold blue")

    # Cetak panel Yang Sudah Dibuat
    console.print(panel)

    # Keluar dari program
    exit()

# ===== [ PENGGUNA PESAN MANAGEMENT ] =====

# Fungsi ini digunakan untuk menambahkan pesanan pada program Role Pengunjung
def tambah_pesanan(username, nama_menu, jumlah_pesanan):
    try: # dicoba terlebih dahulu
        with open("./APD/menu.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            reader = list(csv.reader(file))

```

```

        if 0 <= nama_menu < len(reader):
            nama_menu = reader[nama_menu][0]

        else:
            console.print("[red]Menu yang dipilih tidak valid.")
            return

    pesanan_ada = False
    pesanan = []
    try: # dicoba terlebih dahulu
        with open("./APD/pesanan.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            reader = csv.reader(file) # sebagai variable untuk membaca file external

            for row in reader:
                if row[0] == username and row[1] == nama_menu:
                    new_jumlah = str(int(row[2]) + int(jumlah_pesanan))
                    pesanan.append([username, nama_menu, new_jumlah])
                    pesanan_ada = True

                else:
                    pesanan.append(row)

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        pass

    if not pesanan_ada:
        pesanan.append([username, nama_menu, str(jumlah_pesanan)])

    with open("./APD/pesanan.csv", "w", newline='') as file:
        writer = csv.writer(file) # 'writer' digunakan sebagai variable
        writer.writerows(pesanan)

    print("Pesanan Berhasil Ditambahkan")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Menu Tidak Ditemukan")

    except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
        console.print("[red]Terjadi kesalahan dalam memproses jumlah pesanan")

# Fungsi ini digunakan untuk mengubah pesanan pada program Khusus Role Pengunjung
def ubah_pesanan(index, nama_menu, jumlah_pesanan):
    try: # dicoba terlebih dahulu
        with open("./APD/pesanan.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

            if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
                lines[index][2] = jumlah_pesanan

            with open("./APD/menu.csv", "r") as file: # 'with open' digunakan untuk membuka file external
                lines2 = list(csv.reader(file))
                menubar = lines2[nama_menu][0]
                lines[index][1] = menubar

            with open("./APD/pesanan.csv", "w", newline='') as file:
                writer = csv.writer(file) # 'writer' digunakan sebagai variable
                writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines
    yang dipilih

```

```

        print("Pesanan Berhasil Diubah")

    else:
        console.print("[red]Pilihan Tidak Valid")

except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
    console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk menghapus pesanan pada program Khusus Role Pengunjung
def hapus_pesanan(index):
    try: # dicoba terlebih dahulu
        with open("../APD/pesanan.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
            del lines[index]

            with open("../APD/pesanan.csv", "w", newline='') as file:
                writer = csv.writer(file) # 'writer' digunakan sebagai variable
                writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines yang
dipilih

            print("Pesanan Berhasil Dihapus")
        else:
            console.print("[red]Pilihan Tidak Valid")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# ===== [ ADMIN MENU MANAGEMENT ] =====

# Fungsi ini digunakan untuk menambahkan menu baru pada table menu. Fitur ini hanya khusus digunakan oleh Role
Admin
def tambah_menu(nama_menu, harga_menu):
    try: # dicoba terlebih dahulu
        with open("../APD/menu.csv", "a", newline='') as file: # 'with open' digunakan untuk membuka file external
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerow([nama_menu, harga_menu]) # 'writerow' digunakan untuk menulis row baru pada file csv
            print("Menu Berhasil Ditambahkan")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk mengubah menu pada table menu yang sudah ada. Fitur ini hanya khusus digunakan oleh
Role Admin
def ubah_menu(index, menu_baru, harga_baru):
    try: # dicoba terlebih dahulu
        with open("../APD/menu.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
            lines[index][0] = menu_baru
            lines[index][1] = harga_baru

            with open("../APD/menu.csv", "w", newline='') as file: # 'with open' digunakan untuk membuka file
external
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines yang
dipilih

            print("Menu Berhasil Diubah")

```

```

        else:
            console.print("[red]Pilihan Tidak Valid")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk menghapus salah satu menu pada table menu yang sudah ada. Fitur ini hanya khusus
digunakan oleh Role Admin
def hapus_menu(index):
    try: # dicoba terlebih dahulu
        with open("../APD/menu.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
            del lines[index]

        with open("../APD/menu.csv", "w", newline='') as file: # 'with open' digunakan untuk membuka file
external
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines yang
dipilih
        print("Menu Berhasil Dihapus")

    else:
        console.print("[red]Pilihan Tidak Valid")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# ===== [ ADMIN DATA MANAGEMENT ] =====

# Fungsi ini digunakan untuk menambah berbagai data sebagai akun baru pada table akun. Fitur ini hanya khusus
digunakan oleh Role Admin
def tambah_data(username_baru, password_baru, nama_baru, alamat_baru, no_hp_baru, tanggal_lahir_baru, role_baru =
"Pengunjung"):
    try: # dicoba terlebih dahulu
        with open("../APD/akun.csv", "a", newline='') as file: # 'with open' digunakan untuk membuka file external
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerow([username_baru, password_baru, nama_baru, alamat_baru, no_hp_baru, tanggal_lahir_baru,
role_baru])
        print("Data Berhasil Ditambahkan")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk mengubah berbagai data pada salah satu akun yang ada di dalam table akun. Fitur ini
hanya khusus digunakan oleh Role Admin
def ubah_data(index, username_baru, password_baru, nama_baru, alamat_baru, no_hp_baru, tanggal_lahir_baru,
role_baru = "Pengunjung"):
    try: # dicoba terlebih dahulu
        with open("../APD/akun.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
            lines[index][0] = username_baru
            lines[index][1] = password_baru
            lines[index][2] = nama_baru
            lines[index][3] = alamat_baru
            lines[index][4] = no_hp_baru
            lines[index][5] = tanggal_lahir_baru

```

```

        lines[index][6] = role_baru

        with open("../APD/akun.csv", "w", newline='') as file:
            writer = csv.writer(file) # 'writer' digunakan sebagai variable
            writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines yang
dipilih
            print("Data Berhasil Diubah")

        else:
            console.print("[red]Pilihan Tidak Valid")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk menghapus salah satu akun yang ada di dalam table akun. Fitur ini hanya khusus
digunakan oleh Role Admin
def hapus_data(index):
    try: # dicoba terlebih dahulu
        with open("../APD/akun.csv", "r") as file: # 'with open' digunakan untuk membuka file external
            lines = list(csv.reader(file)) # 'lines' digunakan sebagai variable

        if 0 <= index < len(lines): # Fungsi ini digunakan untuk mengecek index yang di input user
            del lines[index]
            with open("../APD/akun.csv", "w", newline='') as file:
                writer = csv.writer(file) # 'writer' digunakan sebagai variable
                writer.writerows(lines) # 'writer.writerow(lines)' digunakan untuk mengganti data pada lines yang
dipilih
                print("Data Berhasil Dihapus")

        else:
            console.print("[red]Pilihan Tidak Valid")

    except FileNotFoundError: # terjadi ketika file external tidak dapat ditemukan
        console.print("[red]File Tidak Ditemukan")

# Fungsi ini digunakan untuk melakukan validasi pada berbagai input yang berhubungan dengan Tanggal Lahir pada
program
def validasi_tanggallahir(tanggallahir):
    try: # dicoba terlebih dahulu
        dd, mm, yy = map(int, tanggallahir.split('/')) #Mengesplit input tanggallahir dari contoh:02/03/06 menjadi
02,03,06

        if dd < 1 or dd > 31:
            print(f"Tanggal Pada Hari Tidak Valid ({dd}). Maksimal 31.")
            return "404"

        if mm < 1 or mm > 12:
            print(f"Tanggal Pada Bulan Tidak Valid ({mm}). Maksimal 12.")
            return "404"

        if yy < 1900 or yy > 2024:
            print(f"Tanggal Pada Tahun Tidak Valid ({yy}). Minimal Tahun 1900 Dan Maksimal Tahun 2024.")
            return "404"

        return "Tanggal valid."

    except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
        console.print("[red]Format Tanggal Salah. Gunakan format dd/mm/yyyy.")
        return "404"

```



```

except Exception as e:
    console.print(f"[red]Error tak terduga: {e}")
    return "404"

# ===== [ MENU ] =====

# Fungsi ini digunakan untuk menampilkan teks pada Terminal jika pengguna input "kembali" pada program
def kembali_ke_menu():
    print("\nKembali Ke Halaman Utama")

# Fungsi ini digunakan untuk menampilkan menu utama pada Terminal di program
def tampilan_menu_utama():
    menu_text = """
[bold cyan]1.[/bold cyan] Register
[bold cyan]2.[/bold cyan] Login
[bold cyan]3.[/bold cyan] Keluar
"""
    panel = Panel(menu_text, title="[bold green]WARUNG MAKAN ABL[/bold green]", border_style="bold blue")
    console.print(panel)

# Fungsi ini digunakan untuk menampilkan menu admin yang pertama pada Terminal di program
def menu_admin1():
    console.print("""
[bold cyan]<=====>
[bold green]|      SELAMAT DATANG ADMIN      |
[bold cyan]<=====>
[bold white]| [bold yellow]1.[/bold yellow] Mengatur Menu           |
| [bold yellow]2.[/bold yellow] Mengatur Data Pengguna      |
| [bold yellow]3.[/bold yellow] Kembali                      |
[bold cyan]<=====>
""")

# Fungsi ini digunakan untuk menampilkan menu admin yang kedua pada Terminal di program
def menu_admin2():
    console.print("""
[bold cyan]<=====>
[bold green]|      SELAMAT DATANG ADMIN      |
[bold cyan]<=====>
[bold white]| [bold yellow]1.[/bold yellow] Liat Menu           |
| [bold yellow]2.[/bold yellow] Tambah Menu          |
| [bold yellow]3.[/bold yellow] Ubah Menu              |
| [bold yellow]4.[/bold yellow] Hapus Menu              |
| [bold yellow]5.[/bold yellow] Kemballi              |
[bold cyan]<=====>
""")

# Fungsi ini digunakan untuk menampilkan menu pengunjung pada Terminal di program
def menu_pengunjung():
    console.print("""
[bold cyan]<=====>
[bold green]|      SELAMAT DATANG PENGGUNA    |
[bold cyan]<=====>
[bold white]| [bold yellow]1.[/bold yellow] Liat Menu           |
| [bold yellow]2.[/bold yellow] Tambah Pesanan        |
| [bold yellow]3.[/bold yellow] Ubah Pesanan          |
| [bold yellow]4.[/bold yellow] Hapus Pesanan          |
| [bold yellow]5.[/bold yellow] Lihat Pesanan          |
| [bold yellow]6.[/bold yellow] Kembali              |
[bold cyan]<=====>
""")

```

```

# Fungsi ini digunakan untuk menampilkan menu data pengguna pada Terminal di program. Fitur ini hanya khusus untuk
Role Admin
def menu_data():
    console.print("""
[bold cyan]<=====>
[bold green]|          SELAMAT DATANG          |
[bold cyan]<=====>
[bold white]| [bold yellow]1.[/bold yellow] Liat Data Pengguna          |
| [bold yellow]2.[/bold yellow] Tambah Data Pengguna          |
| [bold yellow]3.[/bold yellow] Ubah Data Pengguna          |
| [bold yellow]4.[/bold yellow] Hapus Data Pengguna          |
| [bold yellow]5.[/bold yellow] Lihat Pesanan Pengguna          |
| [bold yellow]6.[/bold yellow] Kembali          |
[bold cyan]<=====>
""")

# ===== [ PROGRAM ] =====

# Memanggil Fungsi Loading untuk menampilkan fitur loading pada Terminal
loading()

# Fungsi ini digunakan untuk memulai sebuah program
def program():

    # Memanggil Fungsi tampilan_menu_utama
    tampilan_menu_utama()

    # Digunakan agar pengguna dapat menginput angka yang sesuai pada menu yang di tampilkan
    pilih1 = input("\nMasukkan Pilihan Anda: ")

    # Menjalankan sebuah prosedur sesuai inputan pengguna sebelumnya
    if pilih1 == "1": # Melakukan prosedur Register jika input pengguna merupakan "1"
        clear()
        console.print("[bold cyan]<=====>")
        console.print("[bold green]|          REGISTER          |")
        console.print("[bold cyan]<=====>")

        # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat menginput kembali
        while True:

            # Digunakan agar pengguna dapat menginput username
            username_baru = input("Masukkan Username Baru: ").strip()

            # Mengecek variabel username_baru apakah valid atau tidak. proses akan berjalan jika input username
valid
            if error_teks(username_baru) == "404":
                break # untuk memberhentikan perulangan

            # Mengecek apakah username yang di input terdaftar dalam akun.csv atau tidak. proses akan berjalan
jika username tidak terdaftar dalam akun.csv
            if not cek_username(username_baru):

                # Digunakan agar pengguna dapat menginput password
                password_baru = pwinput.pwinput("Masukkan Password Baru (Minimal 5 Huruf/Angka): ").strip()

                # Mengecek apakah input password tidak kosong atau jumlah karakter kurang lebih dari 4, jika valid
maka proses akan dilanjutkan
                if password_baru == '' or len(password_baru) <= 4:
                    console.print("[red>Password Minimal 5 Huruf/angka dan Tidak Boleh Kosong!!")

```

```

        break # untuk memberhentikan perulangan

        # Digunakan agar pengguna dapat menginput nama
        nama_baru = input("Masukkan Nama: ").strip()

        # Mengecek apakah input nama apakah valid atau tidak, proses akan berjalan jika input nama valid
        if error_teks(nama_baru) == "404":
            break # untuk memberhentikan perulangan

        # Digunakan agar pengguna dapat menginput alamat pengguna
        alamat_baru = input("Masukkan Alamat: ").strip()

        # Mengecek apakah input Alamat Pengguna termasuk valid atau tidak, jika valid maka proses akan
berlanjut
        if not alamat_baru:
            console.print("[red]Alamat Tidak Boleh Kosong!!")
            break # untuk memberhentikan perulangan

        # Digunakan agar pengguna dapat menginput No Hp Pengguna
        no_hp_baru = input("Masukkan Nomor HP (Minimal 10 Digit dan Maksimal 13 Digit): ").strip()

        # Mengecek apakah input No Hp Pengguna termasuk valid atau jumlah karakter No Hp kurang dari 14
dan lebih dari 9. Jika valid maka proses akan berlanjut
        if error_nomor(no_hp_baru) == "404" or len(no_hp_baru) <= 9 or len(no_hp_baru) >= 14:
            console.print("[red]Nomor Hape Minimal 10 Digit Dengan Maksimal 13 Digit!!")
            break # untuk memberhentikan perulangan

        # Digunakan agar pengguna dapat menginput Tanggal Lahir Pengguna
        tanggal_lahir_baru = str(input("Masukkan Tanggal Lahir (dd/mm/yyyy): ").strip())

        # Mengecek apakah Tanggal Lahir Pengguna Termasuk Valid atau jumlah karakter input pengguna harus
berjumlah 10 karakter, jika valid maka proses akan berlanjut
        if validasi_tanggallahir(tanggal_lahir_baru) == "404" or not len(tanggal_lahir_baru) == 10:
            break # untuk memberhentikan perulangan

        # Menambahkan berbagai inputan pengguna sebelumnya ke akun.csv
        register(username_baru, password_baru, nama_baru, alamat_baru, no_hp_baru, tanggal_lahir_baru)
        break # untuk memberhentikan perulangan

    else:
        break # untuk memberhentikan perulangan
    console.print("[bold cyan]<=====>")

# Melakukan prosedur Login jika input pengguna merupakan "2"
elif pilih1 == "2":
    clear()
    console.print("[bold cyan]<=====>")
    console.print("[bold green]|          LOGIN          |")
    console.print("[bold cyan]<=====>")
    # Digunakan agar pengguna dapat menginput username dan password pengguna
    username = input("Masukkan Username: ")
    password = pwinput.pwinput("Masukkan Password: ")
    role = login(username,password) # Memberikan Role sesuai inputan pengguna
    console.print("[bold cyan]<=====>")

# ===== [ PILIHAN ADMIN MENGATUR MENU ] =====

# Prosedur akan berjalan jika Role pengguna Merupakan Admin
if role == "ADMIN":
    # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat menginput kembali

```

```

while True:

    # Memanggil Fungsi menu admin yang pertama
    menu_admin1()

    # Digunakan agar pengguna dapat menginput angka sesuai daftar menu yang di tampilkan
    pilih2 = input("Masukkan Pilihan Anda: ")
    clear()

    # Prosedur akan berjalan jika pengguna menginput "1"
    if pilih2 == "1":
        # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat menginput
        kembali

        while True:
            # Memanggil Fungsi Menu admin ke dua untuk di tampilkan ke Terminal
            menu_admin2()

            # Digunakan agar pengguna dapat menginput angka sesuai menu yang di tampilkan
            pilih3 = input("Masukkan Pilihan Anda: ")
            clear()

            # Prosedur akan berjalan jika pengguna menginput angka "1", proses ini ditujukan untuk
            menampilkan menu

            if pilih3 == "1":
                clear()
                table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal # Memanggil
                fungsi Table menu untuk ditampilkan di Terminal

            # Prosedur akan berjalan jika pengguna menginput angka "2", proses ini ditujukan untuk
            menambahkan menu

            elif pilih3 == "2":
                clear()
                console.print("[bold cyan]<=====>")
                console.print("[bold green]|          MENAMBAHKAN MENU          |")
                console.print("[bold cyan]<=====>")

            # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat
            menginput kembali

            while True:
                # Digunakan agar pengguna dapat menginput nama menu pada table menu
                nama_menu = input("\nMasukkan Nama Menu: ")

                # Mengecek apakah nama menu sudah valid atau tidak. Proses akan berjalan jika nama
                menu dinyatakan valid

                if error_teks(nama_menu) == "404":
                    break # untuk memberhentikan perulangan

                # Digunakan agar pengguna dapat menginput harga menu pada table menu
                harga_menu = input("Masukkan Harga Menu: Rp.")

                # Mengecek apakah input harga menu valid atau tidak. Proses akan berjalan jika
                harga menu dinyatakan valid

                if error_nomor(harga_menu) == "404":
                    break # untuk memberhentikan perulangan

                try: # dicoba terlebih dahulu
                    harga_menu = int(harga_menu) # Mengubah harga menu menjadi integer
                    tambah_menu(nama_menu, harga_menu) # Menambahkan nama menu dan harga menu ke
                    menu.csv

                    break # untuk memberhentikan perulangan

```

```

except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
    console.print("[red]Harga harus berupa angka")
    console.print("[bold cyan]\n<=====>")

# Prosedur akan berjalan jika pengguna menginput angka "3", proses ini ditujukan untuk
mengubah salah satu menu
elif pilih3 == "3":
    clear()
    table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal # Memanggil
fungsi tabel menu untuk ditampilkan di Terminal
    # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat
menginput kembali

    while True:
        try: # dicoba terlebih dahulu
            # Digunakan agar pengguna dapat input Nomor menu, menu baru, dan harga baru
            untuk mengubah data yang lama

            index_baru = int(input("\nMasukkan Nomor Menu Yang Ingin Diubah: ")) - 1
            menu_baru = input("Masukkan Menu Baru: ")
            harga_baru = input("Masukkan Harga Baru: Rp.")

            # Mengecek apakah input harga menu valid atau tidak. Proses akan berjalan jika
            harga menu dinyatakan valid

            if error_nomor(harga_baru) == "404":
                break # untuk memberhentikan perulangan

            # Mengubah data berdasarkan inputan pengguna sebelumnya ke
            menu.csv

            ubah_menu(index_baru, menu_baru, harga_baru)
            break # untuk memberhentikan perulangan

        except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
            console.print("[red]Input tidak valid")

# Prosedur akan berjalan jika pengguna menginput angka "4", proses ini ditujukan untuk
menghapus salah satu menu
elif pilih3 == "4":
    clear()
    table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal # Memanggil
fungsi tabel menu untuk ditampilkan di Terminal
    try: # dicoba terlebih dahulu
        # Digunakan agar pengguna dapat menginput nomor menu yang ingin dihapus dalam
        tabel menu

        index_hapus = int(input("\nMasukkan Nomor Menu Yang Ingin Dihapus: ")) - 1
        hapus_menu(index_hapus) # Menghapus data berdasarkan inputan pengguna sebelumnya

    except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
        console.print("[red]Input tidak valid")

# Prosedur akan berjalan jika pengguna menginput angka "5", proses ini akan mengarahkan
pengguna ke menu sebelumnya
elif pilih3 == "5":
    clear()
    kembali_ke_menu() # Memanggil fungsi kembali ke menu
    break # untuk memberhentikan perulangan

else:
    console.print("[red]\nPilihan Invalid")

# ===== [ PILIHAN ADMIN MENGATUR DATA ] =====

```

```

        # Prosedur akan berjalan jika pengguna menginput angka "2", proses ini berfungsi untuk mengatur
data pengguna (Khusus Role Admin)
        elif pilih2 == "2":
            # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat menginput
kembali

            while True:
                menu_data() # Memanggil fungsi menu data untuk ditampilkan di terminal

                # Digunakan agar pengguna dapat menginput angka sesuai dengan menu yang ditampilkan
                pilih4 = input("Masukkan Pilihan Anda: ")
                clear()

                # Prosedur akan berjalan jika pengguna menginput angka "1", proses ini berfungsi untuk
menampilkan data dalam tabel
                if pilih4 == "1":
                    clear()
                    table_akun() # Memanggil Fungsi table akun untuk ditampilkan di Terminal # Memanggil
fungsi tabel akun untuk ditampilkan di Terminal

                # Prosedur akan berjalan jika pengguna menginput angka "2", proses ini berfungsi untuk
menampilkan data dalam tabel
                elif pilih4 == "2":
                    clear()
                    console.print("[bold cyan]<=====>")
                    console.print("[bold green]|          MENAMBAHKAN DATA          |")
                    console.print("[bold cyan]<=====>")

                    # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat
menginput kembali

                    while True:
                        try: # dicoba terlebih dahulu

                            # Digunakan agar pengguna dapat menginput username baru
                            nambah_username = input("\nMasukkan Username Baru Pengguna: ").strip()

                            # Memeriksa apakah input username pengguna valid atau tidak. proses akan
berjalan jika username valid

                            if not cek_username(nambah_username):
                                if error_teks(nambah_username) == "404":
                                    break # untuk memberhentikan perulangan

                            # Digunakan agar pengguna dapat menginput password baru
                            nambah_password = pwinput.pwinput("Masukkan Password Baru Pengguna:
").strip()

                            # Memeriksa apakah input password pengguna tidak kosong atau jumlah
karakter password kurang dari 5

                            if nambah_password == '' or len(nambah_password) <= 4:
                                console.print("[red>Password Minimal 5 Huruf/angka dan Tidak Boleh
Kosong!!")

                                break # untuk memberhentikan perulangan

                            # Digunakan agar pengguna dapat menginput Nama baru
                            nambah_nama = input("Masukkan Nama Baru Pengguna: ").strip()
                            # Memeriksa apakah input Nama pengguna valid atau tidak. proses akan
berjalan jika Nama pengguna termasuk valid

                            if error_teks(nambah_nama) == "404":
                                break # untuk memberhentikan perulangan

```

```

# Digunakan agar pengguna dapat menginput Alamat Pengguna baru
nambah_alamat = input("Masukkan Alamat Baru Pengguna : ").strip()
# Memeriksa apakah input Alamat pengguna valid atau tidak. proses akan
berjalan jika Alamat pengguna termasuk valid
if not nambah_alamat:
    console.print("[red]Alamat Tidak Boleh Kosong!!")
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput No Hp Pengguna baru
nambah_NoHP = input("Masukkan No.HP Baru Pengguna (Minimal 10 Digit Dan
Maksimal 13 Digit): ").strip()
# Memeriksa apakah input No Hp Termasuk valid atau jumlah karakter pada No
Hp Pengguna kurang dari 14 dan lebih dari 9. proses akan berjalan jika Alamat pengguna termasuk
valid
if error_nomor(nambah_NoHP) == "404" or len(nambah_NoHP) <= 9 or
len(nambah_NoHP) >= 14:
    console.print("[red]Nomor Hape Minimal 10 Digit Dengan Maksimal 13
Digit!!")
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput Tanggal Lahir Pengguna baru
dengan prompt yang ditentukan
nambah_tanggallahir = str(input("Masukkan Tanggal Lahir Baru Pengguna
(dd/mm/yyyy): ").strip())
# Memeriksa apakah input Tanggal Lahir pengguna valid atau jumlah karakter
pada input tanggal lahir pengguna merupakan 10 karakter. proses akan berjalan jika Tanggal Lahir pengguna termasuk
valid
if validasi_tanggallahir(nambah_tanggallahir) == "404" or not
len(nambah_tanggallahir) == 10:
    break # untuk memberhentikan perulangan

# Menambahkan data pengguna ke akun.csv sesuai berbagai input pengguna
sebelumnya
tambah_data(nambah_username, nambah_password, nambah_nama, nambah_alamat,
nambah_NoHP, nambah_tanggallahir)
break # untuk memberhentikan perulangan

else:
    break # untuk memberhentikan perulangan

except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
    console.print("[red]Harga harus berupa angka")
    console.print("[bold cyan]\n<=====>")

# Prosedur akan berjalan jika pengguna menginput angka "3", proses ini berfungsi untuk
mengubah data pengguna pada salah satu data di dalam tabel data
elif pilih4 == "3":
    clear()
    console.print("[bold cyan]<=====>")
    console.print("[bold green]|          MENUBAH DATA          |")
    console.print("[bold cyan]<=====>")
    table_akun() # Memanggil Fungsi table akun untuk ditampilkan di Terminal

    try: # dicoba terlebih dahulu
        index_baru = int(input("Masukkan Nomor Data Yang Ingin Diubah: ")) - 1 # habis
        memasukkan index nanti dia milih mau data bagian mana yang diubah
        username_baru = input("Masukkan Username Baru: ").strip() # Digunakan agar
        pengguna dapat menginput Username untuk mengubah data yang lama
        if error_teks(nambah_username) == "404": # Memeriksa apakah input Username
        pengguna sudah valid atau tidak. proses akan berjalan Username pengguna termasuk valid

```

```

break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput password baru untuk mengubah data yang
lama
password_baru = pwininput.pwininput("Masukkan Password Baru: ").strip()
# Memeriksa apakah input password pengguna tidak kosong atau jumlah karakter
password kurang dari 5
if password_baru == '' or len(password_baru) <= 4:
    console.print("[red>Password Minimal 5 Huruf/angka dan Tidak Boleh Kosong!!")
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput nama baru untuk mengubah data yang
lama
nama_baru = input("Masukkan Nama Baru: ").strip()
# Memeriksa apakah input Nama pengguna valid atau tidak. proses akan berjalan jika
Nama pengguna termasuk valid
if error_teks(nama_baru) == "404":
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput Alamat baru untuk mengubah data yang
lama
alamat_baru = input("Masukkan Alamat Baru: ").strip()
# Memeriksa apakah input Alamat pengguna valid atau tidak. proses akan berjalan
jika Alamat pengguna termasuk valid
if not alamat_baru:
    console.print("[red]Alamat Tidak Boleh Kosong!!")
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput No Hp baru untuk mengubah data yang lama
no_hp_baru = input("Masukkan No.HP Baru (Minimal 10 Digit Dan Maksimal 13 Digit):
").strip()
# Memeriksa apakah input No Hp Termasuk valid atau jumlah karakter pada No Hp
Pengguna kurang dari 14 dan lebih dari 9. proses akan berjalan jika Alamat pengguna termasuk valid
if error_nomor(no_hp_baru) == "404" or len(no_hp_baru) <= 9 or len(no_hp_baru) >=
14:
    console.print("[red]Nomor Hape Minimal 10 Digit Dengan Maksimal 13 Digit!!")
    break # untuk memberhentikan perulangan

# Digunakan agar pengguna dapat menginput Tanggal Lahir baru untuk mengubah data
yang lama
tanggal_lahir_baru = str(input("Masukkan Tanggal Lahir Baru (dd/mm/yyyy):
").strip())
# Memeriksa apakah input Tanggal Lahir pengguna valid atau jumlah karakter pada
input tanggal lahir pengguna merupakan 10 karakter. proses akan berjalan jika Tanggal Lahir pengguna termasuk
valid
if validasi_tanggallahir(tanggal_lahir_baru) == "404" or not
len(tanggal_lahir_baru) == 10:
    break # untuk memberhentikan perulangan

# Mengubah data yang lama dengan berbagai input pengguna sebelumnya sebagai data
yang baru di akun.csv
ubah_data(index_baru, username_baru, password_baru, nama_baru, alamat_baru,
no_hp_baru, tanggal_lahir_baru) #role_baru dihapus aja nanti

except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
    console.print("[red]Input tidak valid")
    console.print("[bold cyan]\n<=====>")

# Prosedur akan berjalan jika pengguna menginput angka "3", proses ini berfungsi untuk
menghapus salah satu data pengguna di dalam tabel data / akun.csv

```



```

elif pilih4 == "4":
    clear()
    console.print("[bold cyan]<=====>")
    console.print("[bold green]|          MENGHAPUS DATA          |")
    console.print("[bold cyan]<=====>")
    table_akun() # Memanggil Fungsi table akun untuk ditampilkan di Terminal

    try: # dicoba terlebih dahulu
        # Berfungsi agar pengguna dapat menginput nomor data mana yang ingin di hapus
        index_hapus = int(input("\nMasukkan Nomor Data Yang Ingin Dihapus: ")) - 1
        # Menghapus salah satu data di akun.csv sesuai dengan input pengguna
        hapus_data(index_hapus)

    except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
        console.print("[red]Input tidak valid")
        console.print("[bold cyan]\n<=====>")

elif pilih4 == "5":
    clear()
    table_pesanan() # Memanggil Fungsi tabel pesanan untuk ditampilkan di Terminal #
Memanggil Fungsi tabel pesanan untuk ditampilkan di Terminal

elif pilih4 == "6":
    clear()
    kembali_ke_menu()
    break # untuk memberhentikan perulangan

else:
    console.print("[red]\nPilihan Invalid")

elif pilih2 == "3":
    kembali_ke_menu()
    break # untuk memberhentikan perulangan

else:
    console.print("[red]Pilihan Tidak Valid")

# ===== [ PROGRAM PENGGUNA ] =====

# Prosedur akan berjalan jika Role pengguna Merupakan Pengunjung
elif role == "Pengunjung":
    # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat menginput kembali
    while True:

        menu_pengunjung() # Memanggil Fungsi menu pengunjung untuk ditampilkan di Terminal

        # Digunakan agar pengguna dapat menginput angka sesuai dengan menu yang ditampilkan
        pilih2 = input("\nMasukkan Pilihan Anda: ")
        clear()

        # Prosedur akan berjalan jika pengguna menginput angka "1", proses ini berfungsi untuk menampilkan
menu dalam tabel
        if pilih2 == "1":
            clear()
            table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal

            # Prosedur akan berjalan jika pengguna menginput angka "2", proses ini berfungsi untuk menambahkan
pesanan dalam tabel pesanan
            elif pilih2 == "2":

```

```

clear()
console.print("[bold cyan]<=====>")
console.print("[bold green]|      MENAMBAHKAN PESANAN      |")
console.print("[bold cyan]<=====>")
table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal

try: # dicoba terlebih dahulu
    # Digunakan agar pengguna dapat memasukkan angka sesuai menu yang ditampilkan dan ingin
dipesan
    index_pesanan = int(input("Masukkan Nomor Menu Yang Ingin Dipesan: ")) - 1
    # Terdapat Loop agar jika terjadi kesalahan input pada pengguna, maka pengguna dapat
menginput kembali
    while True:
        # Digunakan agar pengguna dapat menginput jumlah pesanan yang diinginkan
        jumlah_pesanan = input("Masukkan Jumlah Pesanan: ")
        # Memeriksa apakah input Jumlah Pesanan valid atau tidak. proses akan berjalan jika
Jumlah Pengguna termasuk valid
        if error_nomor(jumlah_pesanan) == "404":
            break # untuk memberhentikan perulangan

        # Menambahkan pesanan sesuai input pengguna sebelumnya ke dalam pesanan.csv
        tambah_pesanan(username, index_pesanan, jumlah_pesanan)
        break # untuk memberhentikan perulangan

except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
    console.print("[red]Input Tidak Valid")

# Prosedur akan berjalan jika pengguna menginput angka "3", proses ini berfungsi untuk mengubah
salah satu pesanan di dalam tabel pesanan
elif pilih2 == "3":
    clear()
    console.print("[bold cyan]<=====>")
    console.print("[bold green]|      MENGUBAH PESANAN      |")
    console.print("[bold cyan]<=====>")
    table_pesanan() # Memanggil Fungsi tabel pesanan untuk ditampilkan di Terminal

    try: # dicoba terlebih dahulu
        index_pesanan = int(input("Masukkan Nomor Menu Yang Ingin Diubah: ")) - 1 # habis
memasukkan index nanti dia milih mau data bagian mana yang diubah
        table_menu() # Memanggil Fungsi table menu untuk ditampilkan di Terminal
        # Digunakan agar pengguna dapat memasukkan angka sesuai nomor pesanan yang baru
        index_baru = int(input("Masukkan Nomor Menu Baru: ")) - 1
        # Digunakan agar pengguna dapat menginput jumlah pesanan yang diinginkan
        jumlah_pesanan = int(input("Masukkan Jumlah Pesanan: "))
        # Mengubah data pesanan yang lama dengan input pengguna sebelumnya sebagai data pesanan
yang baru
        ubah_pesanan(index_pesanan, index_baru, jumlah_pesanan)

    except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
        console.print("[red]Jumlah Pesanan harus berupa angka")

# Prosedur akan berjalan jika pengguna menginput angka "4", proses ini berfungsi untuk menghapus
salah satu pesanan di dalam tabel pesanan
elif pilih2 == "4":
    clear()
    console.print("[bold cyan]<=====>")
    console.print("[bold green]|      MENGHAPUS PESANAN      |")
    console.print("[bold cyan]<=====>")
    table_pesanan() # Memanggil Fungsi tabel pesanan untuk ditampilkan di Terminal

```

```

        try: # dicoba terlebih dahulu
            # Berfungsi agar pengguna dapat menginput nomor pesanan mana yang ingin di hapus
            index_pesanan = int(input("Masukkan Nomor Menu Yang Ingin Dihapus: ")) - 1
            # Menghapus pesanan sesuai input pengguna sebelumnya
            hapus_pesanan(index_pesanan)

        except ValueError: # Merupakan salah satu Error Handling ketika input tidak sesuai
            console.print("[red]Input Tidak Valid")

        # Prosedur akan berjalan jika pengguna menginput angka "5", proses ini berfungsi untuk melihat
        daftar_pesanan di dalam tabel pesanan
        elif pilih2 == "5":
            clear()
            table_pesanan() # Memanggil Fungsi tabel pesanan untuk ditampilkan di Terminal

        # Prosedur akan berjalan jika pengguna menginput angka "6", proses ini berfungsi untuk mengarahkan
        pengguna ke menu sebelumnya
        elif pilih2 == "6":
            kembali_ke_menu()
            break # untuk memberhentikan perulangan

        else:
            console.print("[red]\nPilihan Invalid")

    else:
        console.print("[bold green]|          Login Gagal!          |")
        console.print("[bold cyan]<=====>")

    # Prosedur akan berjalan jika pengguna menginput angka "3", proses ini berfungsi untuk keluar dari program
    elif pilih1 == "3":
        clear()
        keluar_dari_program() # Memanggil fungsi keluar dari program

    else:
        clear()
        console.print("[red]\nPilihan Tidak Valid!")

while (True):
    program()

```

BAB IV

PENUTUP

4.1 Kesimpulan

Aplikasi manajemen pemesanan Warung Makan ABL ini berhasil memenuhi kebutuhan dasar untuk pengelolaan data menu, pengguna, dan pesanan dengan antarmuka berbasis teks. Program ini memanfaatkan modularitas, validasi input, dan manajemen file yang baik untuk mendukung operasional warung makan.

4.2 Saran

- Mengembangkan antarmuka grafis untuk kemudahan penggunaan.
- Mengintegrasikan sistem ke basis data untuk skalabilitas.
- Menambahkan fitur pembayaran dan laporan transaksi.

DAFTAR PUSTAKA

Lutz, M. (2001). Programming python. " O'Reilly Media, Inc."

Kiusalaas, J. (2010). Numerical methods in engineering with Python.
Cambridge university press

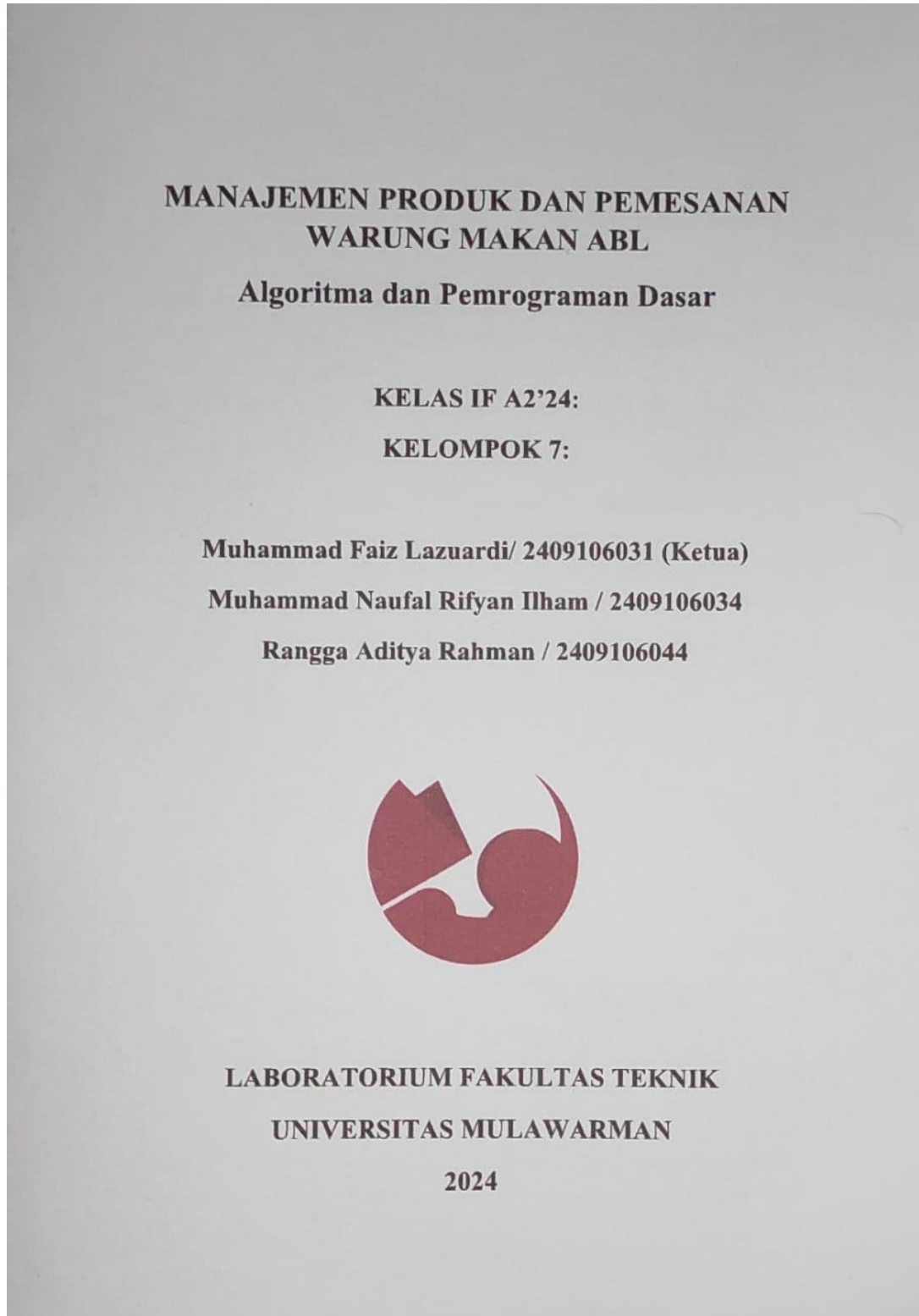
Shafranovich, Y. (2005). Common format and MIME type for comma-separated values (CSV) files (No. rfc4180)

Muqorobin, M., & Rais, N. A. R. (2022). Comparison of PHP programming language with codeigniter framework in project CRUD. International Journal of Computer and Information System (IJCIS), 3(3), 94-98

Safari, A., & Mekhilef, S. (2010). Simulation and hardware implementation of incremental conductance MPPT with direct control method using cuk converter. *IEEE transactions on industrial electronics*, 58(4), 1154-1161.

Shafranovich, Y. (2005). Common format and MIME type for comma-separated values (CSV) files (No. rfc4180).

LAMPIRAN



Gambar 7 Kartu Konsul

Tanggal Konsultasi : 3 NOVEMBER 2024

Uraian / Pembahasan :

- **Konsultasi Flowchart**
Flowchart sudah bagus tetapi masih bisa ditambahkan
Serta beberapa saran seperti:
 1. Admin dapat melihat pesanan user
 2. Menambah detail atribut setiap pengunjung
(Nama, Alamat, Tanggal Lahir)
- **Konsultasi Codingan**
Sudah terlihat alur programnya, tetapi masih banyak
bug dan fitur yang belum ditambah. Saran:
 1. Menambahkan animasi delay pada program
 2. Menggunakan lebih banyak library

Asisten Lab



Nama: Ahmad Nur Fauzan

Ketua Kelompok



Nama: Muhammad Faiz L.

Gambar 8 Halaman Konsul Pertama

Tanggal Konsultasi : 14 NOVEMBER 2024

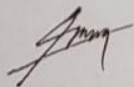
Uraian / Pembahasan :

- Konsultasi Codingan

Progress codingan sudah hampir selesai, tinggal error handling dan memperbaiki tampilan

1. Bagian username tidak menggunakan angka
2. Setelah input selalu dilakukan pengecekan agar tidak terjadi error
3. Fix bug pada bagian ubah pesanan
4. Error handling pada program lebih lengkap

Asisten Lab



Nama: Ahmad Nur Fauzan

Ketua Kelompok



Nama: Muhammad Faiz L.

Gambar 9 Halaman Konsul Kedua