

**LAPORAN PROYEK AKHIR PRAKTIKUM**  
**MATA KULIAH ALGORITMA DAN PEMROGRAMAN**  
**DASAR**



**JUDUL: RENTAL PLAYSTATION**

**Oleh:**

**Kelompok 1 C24**

<b>RIFQI AL BUKHARI (KETUA)</b>	<b>2409106112</b>
<b>RIDUAN ALI</b>	<b>2409106114</b>
<b>LANGGENG DIMAS SAPUTRA</b>	<b>2409106121</b>

**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA 2024**

## **KATA PENGANTAR**

Puji syukur ke hadirat Allah SWT atas rahmat dan karunia-Nya sehingga laporan proyek akhir praktikum ini dapat diselesaikan dengan baik. Laporan ini disusun untuk memenuhi salah satu tugas akhir dari mata kuliah praktikum, dengan tujuan memberikan dokumentasi dari program yang telah dirancang dan dibuat.

Program yang disusun dalam proyek ini adalah Sistem Pengelolaan Menu dan Pesanan berbasis terminal, yang berfungsi untuk mengelola menu makanan, minuman, ruangan PlayStation, dan pesanan pelanggan. Program ini dilengkapi dengan berbagai fitur seperti melihat, menambah, menghapus menu, serta membuat dan mengelola pesanan.

Dalam proses penyusunan dan pembuatan program ini, terdapat beberapa kendala, seperti debugging kode dan penyesuaian fitur sesuai kebutuhan. Namun, berkat bimbingan dari dosen pembimbing, asisten praktikum, dan dukungan dari rekan-rekan, kendala tersebut dapat diatasi dengan baik. Ucapan terima kasih yang sebesar-besarnya kami sampaikan kepada pihak-pihak yang telah membantu penyelesaian program dan laporan ini.

Akhir kata, kami berharap laporan ini dapat bermanfaat bagi pembaca dan menjadi referensi dalam pengembangan sistem serupa di masa depan. Segala kritik dan saran yang membangun sangat kami harapkan untuk perbaikan lebih lanjut.

Samarinda, 22 November 2024

Kelompok 1

Kelompok 1 C2 2024 | 2

## **TAKARIR**

<i>Database</i>	Basis Data
<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
<i>Gui</i>	Grafik User Intervace
<i>Output</i>	Keluaran
<i>If</i>	Jika
<i>Else</i>	Lainnya
<i>While</i>	Ketika
<i>Exit</i>	Keluar
<i>Loop</i>	Perulangan

## DAFTAR ISI

KATA PENGANTAR .....	ii
TAKARIR .....	iii
DAFTAR ISI .....	iv
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.....	2
2.3 Konsep/Materi Praktikum yang dipakai .....	2
BAB III HASIL DAN PEMBAHASAN.....	3
3.1 Tampilan Program.....	3
1. Tampilan Menu.....	3
2. Ss lainnya.....	3
3.2 Source Code.....	3
BAB IV PENUTUP .....	4
4.1 Kesimpulan.....	4
4.2 Saran .....	4
DAFTAR PUSTAKA .....	5

LAMPIRAN .....	6
----------------	---

## **DAFTAR GAMBAR**

3.1 Tampilan Menu Login	9
3.2 Tampilan Admin dan Pelanggan	10
3.3 Source Code	10
LAMPIRAN	4

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi informasi saat ini telah berkembang pesat dan digunakan untuk mempermudah berbagai aspek kehidupan, termasuk dalam hal pengelolaan bisnis dan layanan. Salah satu tantangan yang sering dihadapi oleh pengelola usaha, seperti kafe atau rental PlayStation, adalah bagaimana mengatur menu makanan, minuman, dan fasilitas yang tersedia, serta mengelola pesanan pelanggan secara efisien.

Sistem manual seringkali menyebabkan kesalahan dalam pencatatan pesanan atau pengelolaan menu, yang dapat mengurangi tingkat kepuasan pelanggan. Oleh karena itu, diperlukan sebuah sistem berbasis komputer yang mampu memudahkan pengelolaan data dan proses pemesanan, sehingga pengelola dapat memberikan pelayanan yang lebih baik.

Program ini dirancang untuk membantu pengelolaan menu dan pesanan secara terorganisir, mulai dari penyimpanan data menu hingga pengaturan pesanan pelanggan, dengan antarmuka berbasis terminal yang sederhana namun efektif.

### **1.2 Kebutuhan Fungsional**

1. Menyediakan fitur untuk melihat daftar menu makanan, minuman, dan ruangan PlayStation.
2. Menambahkan dan menghapus menu sesuai kebutuhan pengelola.
3. Menerima pesanan makanan, minuman, dan ruangan dari pelanggan.
4. Mengelola daftar pesanan pelanggan serta menghapus pesanan yang selesai.
5. Memberikan laporan sederhana mengenai pengelolaan menu dan pesanan.

### **1.3 Rumusan Masalah**

1. Bagaimana cara mempermudah pengelolaan menu makanan, minuman, dan ruangan PlayStation?
2. Bagaimana sistem dapat mencatat dan mengelola pesanan pelanggan secara efektif?
3. Bagaimana sistem dapat membantu pengelola dalam menambahkan, melihat, dan menghapus menu dengan mudah?

### **1.4 Batasan Masalah**

1. Program hanya berfungsi pada antarmuka terminal dan tidak menggunakan GUI (Graphical User Interface).
2. Data menu dan pesanan hanya disimpan selama program berjalan (tidak menggunakan database permanen).
3. Sistem dirancang untuk skala kecil dan tidak menangani kebutuhan bisnis besar atau kompleks.

### **1.5 Tujuan**

1. Meningkatkan efisiensi dalam pengelolaan menu dan pesanan pada usaha kecil seperti kafe atau rental PlayStation.
2. Mengurangi kesalahan pencatatan dan pengelolaan data secara manual.
3. Memberikan alat bantu yang sederhana namun efektif bagi pengelola untuk meningkatkan pelayanan kepada pelanggan.



## **BAB II**

### **PERANCANGAN**

#### **2.1 Analisis Program**

##### **1. Menu Utama:**

- a. Pengguna (admin, pelanggan dan exit) memilih apakah ingin masuk ke Menu Admin dan Menu Pelanggan atau keluar.

##### **2. Menu Admin:**

- a. Admin dapat melakukan berbagai operasi manajemen, seperti:
- b. Melihat menu makanan, minuman, dan ruangan PlayStation.
- c. Menambah atau menghapus item di menu.
- d. Melihat dan menghapus pesanan pelanggan.
- e. Semua operasi dilakukan melalui input pengguna di terminal.

##### **3. Menu Pelanggan:**

- a. Pelanggan dapat:
- b. Memesan makanan atau minuman dari menu yang tersedia.
- c. Memesan ruangan PlayStation berdasarkan kategori (No Smoking, Smoking, atau VIP).

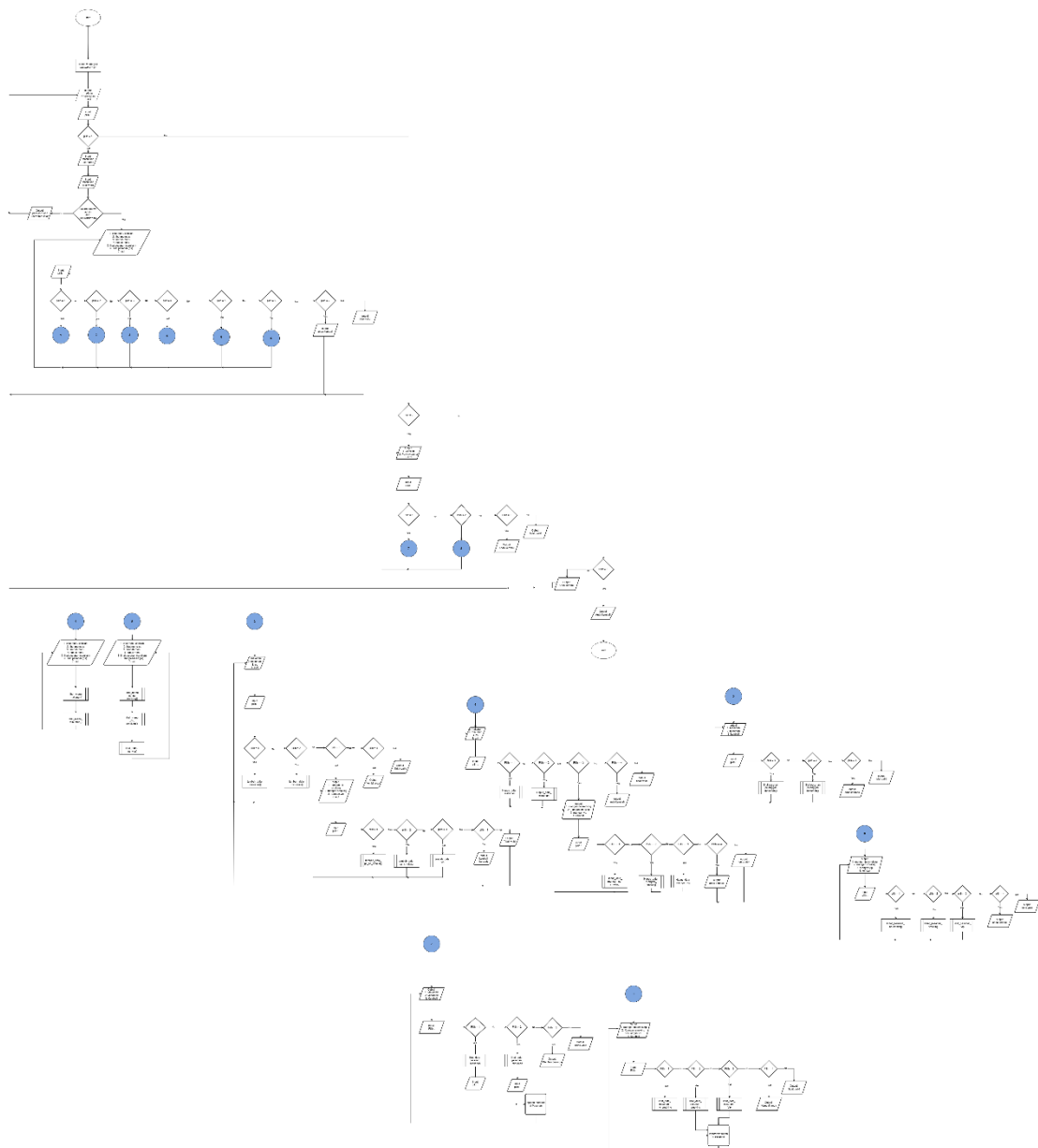
##### **4. Pengelolaan Pesanan:**

- a. Admin dapat memantau dan mengelola semua pesanan pelanggan melalui menu admin.

#### **5. Program Berjalan Secara Berulang:**

- a. Setelah menyelesaikan satu operasi, pengguna dapat kembali ke menu sebelumnya atau keluar dari program.

### **2.2 Flowchart**



### 2.3 Konsep/Materi Praktikum yang dipakai

Dalam pembuatan rental playstation, kami menggunakan berbagai konsep pemrograman yang mendasari fungsionalitas dari perogram tersebut. Berikut adalah beberapa konsep/materi praktikum yang kami terapkan:

## 1. Fungsi Dasar

Fungsi dasar yang digunakan dalam aplikasi ini adalah fungsi **print()**. Fungsi ini digunakan untuk menampilkan informasi kepada pengguna, seperti menu, pesan konfirmasi, dan hasil dari berbagai operasi yang dilakukan dalam aplikasi. Contohnya:

python

VerifyOpen In EditorEditCopy code

```
1 print("=====Selamat Datang Gamers=====")
```

2. Output("=====Selamat Datang Gamers=====")

**Dictionary:** Untuk menu makanan dan minuman

python

VerifyOpen In EditorEditCopy code

```
1 akun_pengguna = {
```

```
2   "menu minuman": 1:{ "Minuman" : "es teh panas/dingin", "Harga" : 4000},
```

```
3   "menu makanan": {1:{ "Makanan" : 'indomie goreng/rebus', 'Harga' : 6000},
```

## 3. Pengendalian Alur Program

Kami menggunakan struktur pengendalian alur seperti **if**, **else**, dan **while** untuk mengatur alur program. Ini memungkinkan aplikasi untuk merespons input pengguna dan menjalankan fungsi yang sesuai. Contoh:

python

VerifyOpen In EditorEditCopy code

```
1 if opsi == "1":
```

```
2     lihat_menu_makanan()
```

#### 4. Fungsi

Kami mendefinisikan beberapa fungsi untuk memecah program menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola. Setiap fungsi memiliki tanggung jawab tertentu,

seperti **lihat\_menu\_minumanan()**, **login\_akun()**, **tambah\_menu\_makanan()**, dan lain-lain. Contoh:

```
python
```

```
VerifyOpen In EditorEditCopy code
```

```
1 menu_makanan():
```

```
2 menu_minuman
```

#### 6. Penggunaan Library Eksternal

Kami menggunakan library eksternal PrettyTable untuk menampilkan data dalam format tabel yang lebih terstruktur dan mudah dibaca. Ini meningkatkan pengalaman pengguna saat melihat menu makanan.

#### 7. Penggunaan Variabel Global

Kami menggunakan variabel global untuk menyimpan pesanan , pesanan ps. Ini memungkinkan kami untuk mengakses dan memodifikasi nilai-nilai ini di seluruh fungsi dalam program.

#### 8. Penggunaan Warna

Kami mendefinisikan kelas Fore untuk mewarnai kode pada teminanl yang digunakan untuk mempercantik tampilan output di terminal. Ini memberikan pengalaman pengguna yang lebih baik dengan memberikan visual yang menarik.

## 9. Modularisasi

Program ini dirancang dengan cara yang modular, sehingga setiap bagian dari aplikasi dapat dikembangkan dan diuji secara terpisah. Ini membuat pemeliharaan dan pengembangan aplikasi lebih mudah.

## 10. Penggunaan Loop

Kami menggunakan loop **while** untuk menjaga aplikasi tetap berjalan hingga pengguna memilih untuk keluar. Ini memungkinkan pengguna untuk melakukan beberapa operasi tanpa harus memulai aplikasi dari awal setiap kali

## BAB III

### HASIL DAN PEMBAHASAN

#### 3.1 Tampilan Menu Login

```
===== SELAMATT DATANGG =====
*****
MENU
[1] Admin
[2] Pelanggan
[3] keluar
=====

silahkan di pilih : 1

Masukkan username anda : cepecial
Masukkan password anda : 24
```

**Admin:** Kemungkinan digunakan untuk masuk ke menu manajemen sistem yang hanya bisa diakses oleh admin.

**Pelanggan:** Pilihan ini biasanya digunakan oleh pelanggan untuk memesan atau melihat menu.

**Keluar:** Untuk keluar dari aplikasi.

### 3.2 Tampilan Admin dan Pelanggan

#### 1. Admin

```
=====
===== SELAMATT DATANGG GAMERSS =====
=====
MENU
[1] Lihat Menu Cemilan
[2] Lihat Menu Ps
[3] Tambah Menu
[4] Hapus Menu
[5] Lihat Pesanan (Cemilan)
[6] Lihat Pesanan (Ps)
[7] Kembali Ke halaman Sebelumnya
=====
silahkan pilih tuan : |
```

#### 2. Pelanggan

```
=====
===== SELAMATT DATANGG GAMERSS =====
=====
MENU
[1] Cemilan
[2] Ps
[3] keluar
=====
pilih : |
```

### 3.3 Source Code

**from prettytable import PrettyTable # untuk membuat tabel**

**from colorama import init, Fore # untuk mewarnai terminal**



**init()**

**# username dan password admin**

**admin = "cepecial"**

**password = "24"**

**pesanan\_ps = []**

**menu\_makanan = {**

**1:{ "Makanan" : 'indomie goreng/rebus', 'Harga' : 6000},**

**2:{ "Makanan" : 'nasi', 'Harga' : 5000},**

**3:{ "Makanan" : 'telur goreng/rebus', 'Harga' : 3000 },**

**4:{ "Makanan" : 'gorengan', 'Harga' : 5000}**

**}**

**menu\_minuman = {**

**1:{ "Minuman" : "es teh panas/dingin", "Harga" : 4000},**

**2:{ "Minuman" : "Nutrisari", "Harga" : 3000},**

**3:{ "Minuman" : "kopi panas", "Harga" : 3000},**

**4:{ "Minuman" : "kopi dingin", "Harga" : 5000},**

**5:{ "Minuman" : "extrajoss", "Harga" : 4000},**

**6:{ "Minuman" : "extrajoss + susu", "Harga" : 6000},**

```
7:{"Minuman" : "kukubima", "Harga" : 4000},  
8:{"Minuman" : "kukubima + susu", "Harga" : 6000},  
9:{"Minuman" : "pop ice", "Harga" : 4000}  
}
```

```
lihat_menu_ps_no_smoking = [  
    {"No": 1, "Ruangan": 1, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 2, "Ruangan": 2, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 3, "Ruangan": 3, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 4, "Ruangan": 4, "jenis": "ps 5", "Harga": 15000 },  
    {"No": 5, "Ruangan": 5, "jenis": "ps 5", "Harga": 15000 }  
]
```

```
lihat_menu_ps_smoking = [  
    {"No": 1, "Ruangan": 11, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 2, "Ruangan": 12, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 3, "Ruangan": 13, "jenis": "ps 4", "Harga": 10000 },  
    {"No": 4, "Ruangan": 14, "jenis": "ps 5", "Harga": 15000 },  
    {"No": 5, "Ruangan": 15, "jenis": "ps 5", "Harga": 15000 }  
]
```

```
lihat_menu_ps_vip = [  
    {"No": 1,"Ruangan": 21, "jenis": "ps 5", "Harga": 20000 },  
    {"No": 2,"Ruangan": 22, "jenis": "ps 5", "Harga": 20000 },  
    {"No": 3,"Ruangan": 23, "jenis": "ps 5", "Harga": 20000 },  
    {"No": 4,"Ruangan": 24, "jenis": "ps 5", "Harga": 20000 },  
    {"No": 5,"Ruangan": 25, "jenis": "ps 5", "Harga": 20000 }  
]
```

```
def lihat_menu_ps_no_smokingg():  
    print(Fore.BLUE + "===Lihat menu ruangan no smoking===")  
    table = PrettyTable(["No", "Ruangan", "Jenis", "Harga"])  
    for info in lihat_menu_ps_no_smoking:  
        table.add_row([info["No"], info["Ruangan"], info["jenis"], info["Harga"],  
    ])  
    print(table)
```

```
def lihat_menu_ps_smokingg():  
    print(Fore.BLUE + "===Lihat menu ruangan smoking===")  
    table = PrettyTable(["No", "Ruangan", "Jenis", "Harga" ])  
    for info in lihat_menu_ps_smoking:
```

```

        table.add_row([info["No"], info["Ruangan"], info["jenis"], info["Harga"],
    ])

    print(table)

```

```

def lihat_menu_ps_vipp():

    print(Fore.BLUE + "===Lihat menu ruangan Vip===")

    table = PrettyTable(["No", "Ruangan", "Jenis", "Harga" ])

    for info in lihat_menu_ps_vip:

        table.add_row([info["No"], info["Ruangan"], info["jenis"],
info["Harga"]])

    print(table)

```

```

def lihat_menu_makanan():

    print(Fore.GREEN + "===Lihat menu makanan===")

    table = PrettyTable([ "NO", "Makanan", "Harga"])

    for No, info in menu_makanan.items():

        if isinstance(info, dict):

            table.add_row([No, info["Makanan"], info["Harga"]])

    print(table)

```

```

def lihat_menu_minuman():

    print(Fore.GREEN + "===Lihat menu minuman===")

    table = PrettyTable([ "NO","Minuman", "Harga"])

    for No, info in menu_minuman.items():

        table.add_row([ No, info["Minuman"], info["Harga"], ])

    print(table)

```

```

def tambah_data_makanan():

    print(Fore.GREEN + "== Tambah Data Makanan ==")

    print("=" * 10)

    input_nama_makanan = input("Nama makanan\t: ")

    if input_nama_makanan.isalpha():

        try:

            input_Harga= int(input("Harga\t: "))

            next_key = max(menu_makanan.keys()) + 1

            menu_makanan[next_key] = {

                "Makanan": input_nama_makanan,

                "Harga": input_Harga

            }

        print("Data makanan berhasil ditambahkan!")

```

```

        return menu_makanan

    except ValueError:

        print("Inputan Tidak Valid")

    else:

        print("Inputan Tidak Valid")


def tambah_data_minuman():

    print(Fore.GREEN + "== Tambah Data Minuman ==")

    print("=" * 10)

    input_nama_minuman = input("Nama minuman\t: ")

    if input_nama_minuman.isalpha():

        try:

            input_Harga= int(input("Harga\t: "))

            next_key = max(menu_makanan.keys()) + 1

            menu_minuman[next_key]({

                "Minuman" : input_nama_minuman,

                "Harga" : input_Harga

            })

            return menu_minuman

        except ValueError:

            print("Inputan Tidak Valid")

```

**else:**

**print(" Inputan Tidak Valid")**

**def tambah\_data\_ps\_no\_smoking():**

**print(Fore.GREEN + "== Tambah Data Ps No Smoking==")**

**print("=" \* 10)**

**lihat\_menu\_ps\_no\_smokingg()**

**try:**

**input\_nomor = int(input("No: "))**

**input\_nama\_ruangan = int(input("No ruangan\t: "))**

**input\_jenis\_ps = input("jenis\t: ")**

**input\_Harga= int(input("Harga\t: "))**

**lihat\_menu\_ps\_no\_smoking.append({**

**"No" : input\_nomor,**

**"Ruangan" : input\_nama\_ruangan,**

**"jenis" : input\_jenis\_ps,**

**"Harga" : input\_Harga**

**})**

**return input\_nomor,input\_nama\_ruangan,input\_jenis\_ps,input\_Harga**

**except ValueError:**

**print("Inputan Tidak Valid")**

```

def tambah_data_ps_smoking():

    print(Fore.GREEN + "== Tambah Data Ps Smoking==")

    print("=" * 10)

    try:

        input_nomor = int(input("No: "))

        input_nama_ruangan = int(input("No ruangan\t: "))

        input_jenis_ps = input("jenis\t: ")

        input_Harga= int(input("Harga\t: "))

        lihat_menu_ps_smoking.append({

            "No" : input_nomor,

            "Ruangan" : input_nama_ruangan,

            "jenis" : input_jenis_ps,

            "Harga" : input_Harga

        })

        return input_nomor,input_nama_ruangan,input_jenis_ps,input_Harga

    except ValueError:

        print("Inputan Tidak Valid")

```

```

def tambah_data_ps_vip():

    print(Fore.GREEN + "== Tambah Data Ps Vip ==")

```



```

print("=" * 10)

try:

    input_nomor = int(input("No: "))

    input_nama_ruangan = int(input("No ruangan\t: "))

    input_Harga= int(input("Harga\t: "))

    input_jenis_ps = input("jenis Ps\t: ")

    lihat_menu_ps_vip.append({

        "No" : input_nomor,

        "Ruangan" : input_nama_ruangan,

        "jenis" : input_jenis_ps,

        "Harga" : input_Harga

    })

    return input_nomor,input_nama_ruangan,input_jenis_ps,input_Harga


except ValueError:

    print("Inputan Tidak Valid")


def hapus_data_makanan():

    print(Fore.GREEN + "== Hapus Data Makanan ==")

    lihat_menu_makanan()

    print(" ")

```

```

try:

    hapus = int(input("masukkan data ke berapa yang ingin dihapus: "))

    if hapus > len(menu_makanan) or hapus < 1:

        print("data tidak ditemukan")

    else:

        del_data = menu_makanan.pop(hapus-1)

        print(f'Menu Nomor {del_data['No']} {del_data['Makanan']} dengan
Harga {del_data['Harga']} telah dihapus')

    except ValueError:

        print("Masukkan nomor ruangan yang valid.")


def clear_pesanan_cemilan():

    try:

        hapus = int(input("masukkan pesanan Ke Berapa Yang Sudah Selesai Di
Antar: "))

        if hapus > len(pesanan) or hapus < 0:

            print("data tidak ditemukan")

        else:

            del_data = pesanan.pop(hapus-1)

            print(f'{del_data['item']} dengan Harga {del_data['Harga']} telah di
Selesaikan')

    except ValueError:

```

```

    print("Masukkan nomor ruangan yang valid.")

def clear_pesanan_ps():

    try:

        hapus = int(input("masukkan pesanan Ke Berapa Yang Sudah Selesai Di
Antar: "))

        if hapus > len(pesanan_ps) or hapus < 0:

            print("data tidak ditemukan")

        else:

            del_data = pesanan_ps.pop(hapus-1)

            print(f'{del_data['Ruangan']}' dengan Harga {del_data['Harga']} telah di
Selesaikan")

        except ValueError:

            print("Masukkan nomor ruangan yang valid.")


def hapus_data_minuman():

    print(Fore.BLUE + "== Hapus Data Minuman ==")

    lihat_menu_minuman()

    try:

```

```

hapus = int(input('masukkan data ke berapa yang ingin dihapus: '))

if hapus > len(menu_minuman) or hapus < 1:

    print("data tidak ditemukan")

else:

    del_data = menu_minuman.pop(hapus-1)

    print(f'Menu Nomor{del_data['No']} {del_data['minuman']} dengan
    Harga {del_data['Harga']} telah dihapus')

except ValueError:

    print("Masukkan nomor ruangan yang valid.")

def hapus_data_ps_no_smoking():

    print(Fore.BLUE + "== Hapus Data Ps No Smoking ==")

    lihat_menu_ps_no_smokingg()

    try:

        hapus = int(input('masukkan Ruangan ke berapa yang ingin dihapus: '))

        if hapus > len(lihat_menu_ps_no_smoking) or hapus < 1:

            print("data tidak ditemukan")

        else:

            del_data = lihat_menu_ps_no_smoking.pop(hapus-1)

            print(f'{del_data['Ruangan']} dengan jenis {del_data['jenis']} dan
            dengan Harga {del_data['Harga']} telah dihapus')

```

```

except ValueError:

    print("Masukkan nomor ruangan yang valid.")


def hapus_data_ps_smoking():

    print(Fore.BLUE + "== Hapus Data Ps Smoking ==")

    lihat_menu_ps_smokingg()

    try:

        hapus = int(input("masukkan Ruangan ke berapa yang ingin dihapus: "))

        if hapus > len(lihat_menu_ps_smoking) or hapus < 1:

            print("data tidak ditemukan")

        else:

            del_data = lihat_menu_ps_smoking.pop(hapus-1)

            print(f'{del_data['Ruangan']} dengan jenis {del_data['jenis']} dan
dengan Harga {del_data['Harga']} telah dihapus')

        except ValueError:

            print("Masukkan nomor ruangan yang valid.")


def hapus_data_vip():

    print(Fore.BLUE + "== Hapus Data Ps Vip ==")

    lihat_menu_ps_vipp()

    try:

```

```

hapus = int(input('masukkan Ruangan yang ingin dihapus: '))

if hapus > len(lihat_menu_ps_vip) or hapus < 0:

    print("data tidak ditemukan")

else:

    del_data = lihat_menu_ps_vip.pop(hapus-1)

    print(f'{del_data['Ruangan']} dengan jenis {del_data['jenis']} dan
dengan Harga {del_data['Harga']} telah dihapus')

except ValueError:

    print("Masukkan nomor ruangan yang valid.")


# pelanggan

pesanan = []

def buat_pesanan_cemilan():

    total_pesanan = 0

    while True:

        print(Fore.GREEN + "=" * 50)

        print(Fore.BLUE + "===== MENU CEMILAN
=====")

        print(Fore.GREEN + "=" * 50)

```

```
print("MENU")
```

```
print("[1] Makanan")
```

```
print("[2] Minuman")
```

```
print("[3] keluar")
```

```
print("=" * 50)
```

```
print(" ")
```

```
kategori = input("Masukkan Kategori Cemilan Yang Ingin Dibeli : ")
```

```
if kategori == '1':
```

```
    lihat_menu_makanan()
```

```
    try:
```

```
        pilihan = input("Pilih makanan yang ingin dipesan atau tekan 'k' jika  
sudah selesai memilih: ")
```

```
        if pilihan == 'k':
```

```
            continue
```

```
    try:
```

```
        pilihan = int(pilihan)
```

```
        if 0 <= pilihan <= len(menu_makanan):
```

```
            item = menu_makanan[pilihan ]
```

```

        jumlah = int(input(f"Berapa jumlah {item['Makanan']} yang
dipesan? "))

        pesanan.append({"item": item['Makanan'], "Harga":
item['Harga'], "jumlah": jumlah})

        total_pesanan += item['Harga'] * jumlah

        print(f"{jumlah} {item['Makanan']} ditambahkan ke pesanan.")


    print("\nPesanan Anda:")

    for item in pesanan:

        print(f"- {item['jumlah']} x {item['item']} (Rp
{item['Harga']}/item) = Rp {item['Harga'] * item['jumlah']}")

    print(f"\nTotal Harga: Rp {total_pesanan}")

    lanjut = input("Apakah ingin memesan lagi? (iya/tidak): ")

    if lanjut.lower() != 'iya':

        break

except ValueError:

    print(" Inputan tidak valid")


except TypeError:

    print("Masukkan angka yang valid.")


elif kategori == '2':

```



```

    lihat_menu_minuman()

    try:

        pilihan = input('Pilih minuman yang ingin dibeli atau ketik 'k' jika
sudah selesai memilih: ')

        if pilihan == 'k':

            continue

    try:

        pilihan = int(pilihan)

        if 0 <= pilihan <= len(menu_minuman):

            item = menu_minuman[pilihan ]

            jumlah = int(input(f'Berapa jumlah {item['Minuman']} yang
dipesan? '))

            pesanan.append({'item': item['Minuman'], 'Harga':
item['Harga'], 'jumlah': jumlah})

            total_pesanan += item['Harga'] * jumlah

            print(f'{jumlah} {item['Minuman']} ditambahkan ke
pesanan.")

        print("\nPesanan Anda:")

        for item in pesanan:

            print(f'- {item['jumlah']} x {item['item']} (Rp
{item['Harga']}/item) = Rp {item['Harga'] * item['jumlah']}")

```

```

        print(f"\nTotal Harga: Rp {total_pesanan}")

        lanjut = input("Apakah ingin memesan lagi? (iya/tidak): ")

        if lanjut.lower() != 'iya':

            break

    except ValueError:

        print("Masukkan angka yang valid.")

    except TypeError:

        print("Masukkan angka yang valid.")

elif kategori == '3':

    print("\nPesanan Anda:")

    for item in pesanan:

        print(f"- {item['jumlah']} x {item['item']} (Rp {item['Harga']}/item) =  
Rp {item['Harga'] * item['jumlah']}")

        print(f"\nTotal Harga: Rp {total_pesanan}")

        break

else:

    print("anda tidak jadi memesan.")

```

```

def lihat_pesanan_cemilan():

    if pesanan:

        print("\nPesanan Anda:")

        for item in pesanan:

            print(f'- {item['jumlah']} x {item['item']} (Rp {item['Harga']}/item) = Rp {item['Harga'] * item['jumlah']}')

def buat_pesanan_ps():

    total_pesanan = 0

    while True:

        print(Fore.GREEN + "=" * 50)

        print(Fore.BLUE + "===== MENU RUANGAN PLAYSTATION =====")

        print(Fore.GREEN + "=" * 50)

        print("MENU")

        print("[1] Ruangan No Smoking")

        print("[2] Ruangan Smoking")

        print("[3] Ruangan Vip")

```

```

print("[4] keluar")

print("=" * 50)

print(" ")

kategori_ruangan = input("Pilih kategori ruangan : ")

if kategori_ruangan == '1':

    lihat_menu_ps_no_smokingg()

    try:

        pilihan = input('pilih Ruangan yang ingin dipesan atau tekan 'k'
jika sudah selesai memilih: ')

        if pilihan == 'k':

            continue

        try:

            pilihan = int(pilihan)

            if 1 <= pilihan <= len(lihat_menu_ps_no_smoking):

                item = lihat_menu_ps_no_smoking[pilihan - 1]

                pesanan_ps.append({"Ruangan": item, "Harga":
item['Harga'],})

                total_pesanan += item['Harga']

                print(f'{item['Ruangan']} ditambahkan ke pesanan.')

```

```

for item in pesanan_ps:

    print(f'- {item['Ruangan']} = (Rp {item['Harga']}/item)')

    print(f'\nTotal Harga: Rp {total_pesanan}')
```

```

if pilihan > len(lihat_menu_ps_no_smoking) or pilihan < 1:

    print("data tidak ditemukan")

else:

    del_data = lihat_menu_ps_no_smoking.pop(pilihan-1)

    print(f'{del_data['Ruangan']} dengan jenis
{del_data['jenis']} dan dengan Harga {del_data['Harga']} telah dihapus')

else:

    print("Nomor makanan tidak valid.")

except ValueError:

    print("Masukkan angka yang valid.")

except TypeError:

    print("Masukkan angka yang valid.")

elif kategori_ruangan == '2':

    lihat_menu_ps_smokingg()

```

```

        pilihan = input('pilih Ruangan yang ingin dipesan atau tekan 'k' jika
sudah selesai memilih: ')

        if pilihan in pesanan_ps:

            print("Ruangan Sudah Di pesan")

        elif pilihan == 'k':

            continue

    try:

        pilihan = int(pilihan)

        if 1 <= pilihan <= len(lihat_menu_ps_smoking):

            item = lihat_menu_ps_smoking[pilihan - 1]

            pesanan_ps.append({"Ruangan": item, "Harga": item['Harga'],})

            total_pesanan += item['Harga']

            print(" ")

            print(f" Ruangan {item['Ruangan']} ditambahkan ke pesanan.")

            print("\nPesanan Anda:")

            for item in pesanan_ps:

                print(f"- {item['Ruangan']} = (Rp {item['Harga']}/item)")

                print(f"\nTotal Harga: Rp {total_pesanan}")

            if pilihan > len(lihat_menu_ps_smoking) or pilihan < 1:

                print("data tidak ditemukan")

            else:

```

```

        del_data = lihat_menu_ps_smoking.pop(pilihan-1)

        print(f'{del_data['Ruangan']} dengan jenis {del_data['jenis']}
dan dengan Harga {del_data['Harga']} telah dihapus')

    except ValueError:

        print("Masukkan angka yang valid.")


elif kategori_ruangan == '3':

    lihat_menu_ps_vipp()

    try:

        pilihan = input("pilih Ruangan yang ingin dipesan atau tekan 'k'
jika sudah selesai memilih: ")

        if pilihan == 'k':

            continue

        try:

            pilihan = int(pilihan)

            if 1 <= pilihan <= len(lihat_menu_ps_vip):

                item = lihat_menu_ps_vip[pilihan - 1]

                pesanan_ps.append({'Ruangan': item, "Harga":
item['Harga'],})

                total_pesanan += item['Harga']

```

```

        print(f'{item['Ruangan']} ditambahkan ke pesanan.')

    print("\nPesanan Anda:")

    for item in pesanan_ps:

        print(f'- {item['Ruangan']} = (Rp {item['Harga']}/item)')

        print(f"\nTotal Harga: Rp {total_pesanan}")

    if pilihan > len(lihat_menu_ps_vip) or pilihan < 1:

        print("data tidak ditemukan")

    else:

        del_data = lihat_menu_ps_vip.pop(pilihan-1)

        print(f'{del_data['Ruangan']} dengan jenis {del_data['jenis']} dan dengan Harga {del_data['Harga']} telah dihapus')

    except ValueError:

        print("Masukkan angka yang valid.")

    except TypeError:

        print("Masukkan angka yang valid.")

elif kategori_ruangan == "4":

    break

else:

    print("inputan tidak valid")

```



```

def lihat_pesanan_ps():

    if pesanan_ps:

        print("\nPesanan Anda:")

        for item in pesanan_ps:

            print(f'- {item['Ruangan']} = (Rp {item['Harga']}) ')

# program inti

while True:

    print(Fore.BLUE + "*" * 50)

    print(Fore.GREEN + "===== SELAMATT DATANGG
=====")

    print(Fore.BLUE + "*" * 50)

    print("MENU")

    print("[1] Admin")

    print("[2] Pelanggan")

    print("[3] keluar")

    print("=" * 50)

    print(" ")

```

```

memilih = input('silahkan di pilih : ')

print(" ")

if memilih == "1":

    username_admin = input(Fore.YELLOW + "Masukkan username anda : ")

    password_admin = input("Masukkan password anda : ")

    if username_admin == admin and password_admin == password :

        while True:

            print(Fore.GREEN + "=" * 50)

            print(Fore.BLUE + "===== SELAMATT DATANGG  
GAMERSS =====")

            print(Fore.GREEN + "=" * 50)

            print("MENU")

            print("[1] Lihat Menu Cemilan")

            print("[2] Lihat Menu Ps")

            print("[3] Tambah Menu")

            print("[4] Hapus Menu")

            print("[5] LIhat Pesanan (Cemilan)")

            print("[6] Lihat Pesanan (Ps)")

            print("[7] Kembali Ke halaman Sebelumnya")

```

```
print("=" * 50)
```

```
print(" ")
```

```
pilih = input("silahkan pilih tuan : ")
```

```
print(" ")
```

```
if pilih == "1":
```

```
    lihat_menu_makanan()
```

```
    print(" ")
```

```
    lihat_menu_minuman()
```

```
elif pilih == "2":
```

```
    lihat_menu_ps_no_smokingg()
```

```
    print(" ")
```

```
    lihat_menu_ps_smokingg()
```

```
    print(" ")
```

```
    lihat_menu_ps_vipp()
```

```
    print(" ")
```

```

elif pilih == "3":

    while True:

        print(Fore.GREEN + "=" * 50)

        print(Fore.BLUE + "===== MENU YANG INGIN DI
TAMBAH =====")

        print(Fore.GREEN + "=" * 50)

        print("MENU")

        print("[1] Menu Makanan")

        print("[2] Menu Minuman")

        print("[3] Ruangan Ps")

        print("[4] keluar")

        print("=" * 50)

        print(" ")

        pilih_ganti = input("masukkan yang ingin menu mana yang
anda ingin ganti : ")

        print(" ")

        if pilih_ganti == "1":

            tambah_data_makanan()

```

```
elif pilih_ganti == "2":
```

```
tambah_data_minuman()
```

```
elif pilih_ganti == "3":
```

```
while True:
```

```
print(Fore.GREEN + "=" * 50)
```

```
print(Fore.BLUE + "=====MENU RUANGAN YANG  
INGIN DI TAMBAH =====")
```

```
print(Fore.GREEN + "=" * 50)
```

```
print("MENU")
```

```
print("[1] Ruangan No Smoking")
```

```
print("[2] Ruangan Smoking")
```

```
print("[3] Ruangan Vip")
```

```
print("[4] keluar")
```

```
print("=" * 50)
```

```
print(" ")
```

```
pilih_gantii = input("pilih yang mana : ")
```

```
if pilih_gantii == "1":
```

```
tambah_data_ps_no_smoking()
```

```
elif pilih_gantii == "2":
```

```
tambah_data_ps_smoking()
```

```
elif pilih_gantii == "3":
```

```
tambah_data_ps_vip()
```

```
elif pilih_gantii == "4":
```

```
print("kembali ke beranda")
```

```
break
```

```
else:
```

```
print("inputan anda tidak valid,silahkan pilih 1,2,3,dan
```

```
4")
```

```
elif pilih_ganti == "4":
```

```
print("kembali ke beranda")
```

**break**

**else:**

**print("inputan anda tidak valid,silahkan pilih 1,2,3,dan 4")**

**elif pilih == "4":**

**while True:**

**print(Fore.GREEN + "=" \* 50)**

**print(Fore.BLUE + "===== MENU YANG INGIN DI  
HAPUS =====")**

**print(Fore.GREEN + "=" \* 50)**

**print("MENU")**

**print("[1] Menu Makanan")**

**print("[2] Menu Minuman")**

**print("[3] Ruangan Ps")**

**print("[4] keluar")**

**print("=" \* 50)**

**print(" ")**

```

        pilih_hapus = input('masukkan yang ingin menu mana yang
anda ingin ganti : ')

        print(" ")

        if pilih_hapus == "1":

            hapus_data_makanan()

        elif pilih_hapus == "2":

            hapus_data_minuman()

        elif pilih_hapus == "3":

            while True:

                print(Fore.GREEN + "=" * 50)

                print(Fore.BLUE + "===== MENU RUANGAN YANG
INGIN DI HAPUS =====")

                print(Fore.GREEN + "=" * 50)

                print("MENU")

                print("[1] Ruangan No Smoking")

                print("[2] Ruangan Smoking")

```



```
print("[3] Ruangan Vip")

print("[4] keluar")

print("=" * 50)

print(" ")

pilih_hapuss= input("pilih yang mana : ")

print(" ")


if pilih_hapuss == "1":

    hapus_data_ps_no_smoking()


elif pilih_hapuss == "2":

    hapus_data_ps_smoking()


elif pilih_hapuss == "3":

    hapus_data_vip()


elif pilih_hapuss == "4":

    print("Kembali ke menu sebelumnya")

    break

else:

    print("inputan tidak valid")
```

```
elif pilih_hapus == "4":  
    print("kembali ke menu sebelumnya")  
    print(" ")  
    break  
else:  
    print("inputan tidak valid")
```

```
elif pilih == "5":  
    lihat_pesanan_cemilan()  
    print(" ")  
    clear_pesanan_cemilan()  
    print(" ")
```

```
elif pilih == "6":  
    lihat_pesanan_ps()  
    print(" ")  
    clear_pesanan_ps()
```

```
elif pilih == "7":
```

```
    print("Kembali Ke Beranda")
```

```
    break
```

```
elif memilih == "2":
```

```
    while True:
```

```
        print(Fore.GREEN + "=" * 50)
```

```
        print(Fore.BLUE + "===== SELAMATT DATANGG  
GAMERSS =====")
```

```
        print(Fore.GREEN + "=" * 50)
```

```
        print("MENU")
```

```
        print("[1] Cemilan")
```

```
        print("[2] Ps")
```

```
        print("[3] keluar")
```

```
        print("=" * 50)
```

```
        print(" ")
```

```
    memilih = input("pilih : ")
```

```
if memilih == "1":
```

```
buat_pesanan_cemilan()
```

```
print(" ")
```

```
elif memilih == "2":
```

```
    buat_pesanan_ps()
```

```
    print(" ")
```

```
elif memilih == "3":
```

```
    break
```

```
else:
```

```
    print("inputan tidak valid")
```

```
elif memilih == "3":
```

```
    print(Fore.CYAN + "TERIMAKASIH DAN SEMANGAT")
```

```
    break
```

```
else:
```

```
    print("inputan tidak valid")
```

## **BAB IV**

### **PENUTUP**

#### **4.1 Kesimpulan**

Program yang telah dibuat merupakan sebuah sistem sederhana berbasis terminal yang dapat digunakan untuk mengelola usaha kecil seperti kafe atau rental PlayStation. Program ini memungkinkan admin untuk:

- a. Mengelola menu makanan, minuman, dan ruangan PlayStation dengan fitur penambahan, penghapusan, dan peninjauan data.
- b. Mengelola pesanan pelanggan secara real-time, sehingga memudahkan operasional harian.

Sedangkan pelanggan dapat dengan mudah memesan makanan, minuman, atau ruangan PlayStation melalui menu yang tersedia. Program ini membantu mengotomasi proses pencatatan pesanan yang sebelumnya mungkin dilakukan secara manual, sehingga lebih cepat, akurat, dan efisien.

Namun, program ini masih memiliki keterbatasan, seperti tidak adanya penyimpanan data permanen dan antarmuka yang terbatas pada terminal.

#### **4.2 Saran**

Untuk pengembangan lebih lanjut, berikut beberapa saran yang dapat diterapkan:

### **1. Penambahan Penyimpanan Data**

- a. Menggunakan file eksternal (CSV/JSON) atau database sederhana (SQLite) untuk menyimpan data menu dan pesanan secara permanen.

### **2. Pengembangan Antarmuka Pengguna**

- b. Meningkatkan pengalaman pengguna dengan menggunakan GUI (Graphical User Interface) berbasis library seperti Tkinter atau PyQt, sehingga lebih mudah digunakan.

### **3. Validasi Input yang Lebih Baik**

- c. Menambahkan validasi input yang lebih ketat untuk mencegah kesalahan input data oleh pengguna.

### **4. Multi-User Support**

- d. Menambahkan fitur untuk mendukung penggunaan oleh beberapa pengguna secara bersamaan, seperti menggunakan teknologi berbasis jaringan.

### **5. Laporan Penjualan**

- e. Menambahkan fitur laporan harian atau bulanan untuk membantu admin menganalisis pendapatan dan pesanan pelanggan.

## DAFTAR PUSTAKA

- Python Software Foundation. (2024). *Python Documentation*. Diakses dari <https://docs.python.org>.
- Lundh, F. (2001). *An Introduction to Tkinter*. Diakses dari <https://tkinter-python.readthedocs.io>.
- Beazley, D. M. (2009). *Python Essential Reference (4th Edition)*. Boston: Addison-Wesley.
- PrettyTable Documentation. (2024). Diakses dari <https://pypi.org/project/PrettyTable/>.
- McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Sebastopol: O'Reilly Media.
- Yulianto, A. S., & Darmawan, R. (2022). *Pengembangan Aplikasi Berbasis Terminal untuk Usaha Kecil*. Bandung: Alfabeta.

## **LAMPIRAN**

### **Rental Playstation Algoritma dan Pemrograman Dasar**

**KELAS Informatika C2'24:**

**KELOMPOK 1:**

**Rifqi Al Bukhari / 2409106112(ketua)**

**Riduan Ali / 2409106114**

**Langgeng Dimas Saputra/ 2409106121**

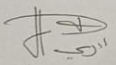



**LABORATORIUM FAKULTAS TEKNIK**

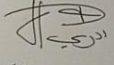
**UNIVERSITAS MULAWARMAN**

**2024**



Tanggal Konsultasi : 03 November 2021	
Uraian / Pembahasan : Perbaikan Flowchart • penggunaan DEF pada Flowchart • garis yang bertitik-titik • kotak yang pasti	
Asisten Lab	Ketua Kelompok
 Nama: Adriani Mawak Allo	 Nama: RIFI AL-BUMITARI

Tanggal Konsultasi :	
Uraian / Pembahasan :	
Asisten Lab	Ketua Kelompok
 Nama: Adriani Mawak Allo	Nama: