

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



PEMESANAN KAMAR HOTEL

Oleh:

Kelompok III

NAMA KETUA KELOMPOK	NIM KETUA KELOMPOK
Muhammad Kurniawan	2409106091
Muhammad Taufiqul Hafizh	2409106099
Muttaqin	
Muhammad Fizriannur	2409106105

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

KATA PENGANTAR

1. Paragraf 1:

Segala puji dan syukur kami panjatkan ke hadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan ini dengan baik. Laporan ini disusun sebagai bagian dari pelaksanaan tugas akhir dalam rangka memenuhi salah satu persyaratan akademik.

2. Paragraf 2:

Program yang kami buat bertujuan untuk meningkatkan efisiensi dalam Program Pemesanan Kamar Hotel, dengan fitur-fitur yang dirancang sesuai kebutuhan pengguna.

3. Paragraf 3:

Selama proses penyusunan laporan dan pembuatan program, kami menghadapi beberapa kendala, seperti, contoh: pwinput tidak ada, kurang error handling availability diubah menjadi enable / disable, waktu booking, tambahan deskripsi fasilitas di setiap jenis kamar dan diskon hari penting, namun berkat bimbingan dan bantuan dari berbagai pihak, kendala tersebut dapat teratasi. Ucapan terima kasih kami sampaikan kepada Mba Yuyun Nabila atas ilmu, dukungan, dan arahannya selama proses ini.

4. Paragraf 4:

Akhir kata, kami berharap laporan ini dapat bermanfaat bagi pembaca dan memberikan kontribusi dalam bidang PA APD Semester I. Kritik dan saran sangat kami harapkan untuk penyempurnaan di masa mendatang.

Samarinda, 22 November 2024

TAKARIR

Dalam pembuatan sistem informasi ini, sejumlah istilah bahasa asing digunakan. Sebagai bagian dari upaya menjaga penggunaan bahasa Indonesia yang baik dan benar. Daftar padanan kata bahasa asing dalam bahasa Indonesia yang digunakan adalah:
sebagai berikut:

<i>Database</i>	Basis Data
<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
<i>Output</i>	Keluaran
<i>Disable</i>	Tidak Aktif
<i>Close</i>	Menutup
Dictionary	Kamus
If	Jika
Else	Maka
Password	Kata Sandi

DAFTAR ISI

KATA PENGANTAR	ii
TAKARIR	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	vi
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.....	6
3.1 Tampilan Program.....	6
3.1.1 Tampilan Menu Utama.....	6
3.1.2 Tampilan Login.	3
3.1.3 Tampilan Pembuatan Akun.	4
3.1.4 Tampilan Keluar di Menu Utama.	4
3.1.5 Tampilan Menu Admin	4
3.1.6 Tampilan Tambah Kamar di Menu Admin.	5
3.1.7 Tampilan Lihat Kamar di Menu Admin.....	5
3.1.8 Tampilan Ubah Kamar di Menu Admin	6
3.1.9 Tampilan Hapus Kamar di Menu Admin.	6
3.1.10 Tampilan Menu User.....	7
3.1.11 Tampilan Lihat Kamar Tersedia di Menu User.	7
3.1.12 Tampilan Booking Kamar di Menu User.	7
3.2 Source Code	11
BAB IV PENUTUP	22
4.1 Kesimpulan	22
4.2 Saran	22
DAFTAR PUSTAKA	23
LAMPIRAN	24

DAFTAR GAMBAR

Gambar 3.1.1 Tampilan Menu Utama.....	3
Gambar 3.1.2 Tampilan Login.....	3
Gambar 3.1.3 Tampilan Pembuatan Akun.	4
Gambar 3.1.4 Tampilan Keluar di Menu Utama.....	4
Gambar 3.1.5 Tampilan Menu Admin.	4
Gambar 3.1.6 Tampilan Tambah Kamar di Menu Admin.....	5
Gambar 3.1.7 Tampilan Lihat Kamar di Menu Admin.	5
Gambar 3.1.8 Tampilan Ubah Kamar di Menu Admin.	6
Gambar 3.1.9 Tampilan Hapus Kamar di Menu Admin.	6
Gambar 3.1.10 Tampilan Menu User.....	7
Gambar 3.1.11 Tampilan Lihat Kamar Tersedia di Menu User.....	7
Gambar 3.1.12 Tampilan Booking Kamar di Menu User.....	7

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pengelolaan data hotel sering kali dilakukan secara manual, menyebabkan kesalahan dalam pencatatan dan pengelolaan informasi. Oleh karena itu, dibutuhkan sistem yang terkomputerisasi untuk mengatasi masalah ini.

1.2 Kebutuhan Fungsional

1. Admin dapat menambah, melihat, mengubah, dan menghapus data kamar.
2. User dapat melihat kamar tersedia dan melakukan booking.

1.3 Rumusan Masalah

Bagaimana membangun sistem informasi berbasis Python yang memudahkan manajemen data hotel?

1.4 Batasan Masalah

- Data hanya disimpan dalam file CSV.
- Sistem ini bersifat lokal (tidak menggunakan basis data server).

1.5 Tujuan

Membuat sistem manajemen hotel yang sederhana dan mudah digunakan dengan Python.

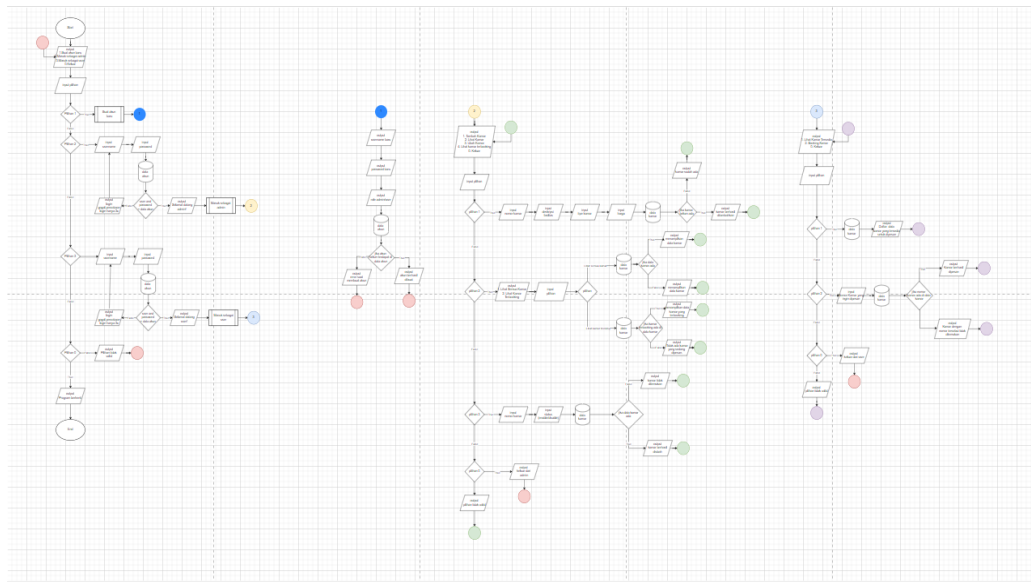
BAB II

PERANCANGAN

2.1 Analisis Program

Program ini memungkinkan admin untuk mengelola data kamar, serta user untuk memesan kamar. Data disimpan menggunakan file CSV dan ditampilkan dengan tabel yang rapi menggunakan PrettyTable.

2.2 Flowchart



1. Awal Program

- Program dimulai dengan menampilkan output pilihan:
 1. Buat akun baru
 2. Masuk sebagai admin
 3. Masuk sebagai user

4. Keluar

2. Memilih Opsi

1. User diminta untuk memilih salah satu opsi dengan menginput pilihan.

3. Memilih Buat Akun Baru

1. Jika user memilih opsi buat akun baru, maka program akan meminta input username dan password.
2. Data username dan password akan disimpan di database akun.
3. Program akan menampilkan output username baru dan password baru.

4. Memilih Masuk Sebagai Admin

1. Jika user memilih opsi masuk sebagai admin, maka program akan meminta input username dan password.
2. Data username dan password akan dicocokkan dengan database akun.
3. Jika username dan password benar, program akan menampilkan output "Selamat datang admin!".
4. Jika username dan password salah, program akan menampilkan output "Login gagal, percobaan login hanya 3x".

5. Memilih Masuk Sebagai User

1. Jika user memilih opsi masuk sebagai user, maka program akan meminta input username dan password.
2. Data username dan password akan dicocokkan dengan database akun.
3. Jika username dan password benar, program akan menampilkan output "Selamat datang user!".

4. Jika username dan password salah, program akan menampilkan output "Login gagal, percobaan login hanya 3x".

6. Memilih Opsi Sebagai Admin

Jika user sudah login sebagai admin, program akan menampilkan output pilihan:

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Kamar
4. Lihat kamar terbooking
5. Keluar

7. Memilih Opsi Tambah Kamar

1. Jika user memilih opsi tambah kamar, maka program akan meminta input deskripsi fasilitas, tipe kamar, dan harga kamar.
2. Data kamar baru akan disimpan di database kamar.
3. Program akan menampilkan output "Kamar berhasil ditambahkan!".

8. Memilih Opsi Lihat Kamar

1. Jika user memilih opsi lihat kamar, maka program akan menampilkan output semua data kamar.
2. Program akan menampilkan output "Kamar berhasil ditambahkan!".

9. Memilih Opsi Ubah Kamar

1. Jika user memilih opsi ubah kamar, maka program akan meminta input nomor kamar yang ingin diubah.
2. Data kamar yang sesuai dengan nomor kamar yang diinput akan diubah.

3. Program akan menampilkan output "Kamar berhasil diubah!".

10. Memilih Opsi Lihat Kamar Terbooking

1. Jika user memilih opsi lihat kamar terbooking, maka program akan menampilkan semua data kamar yang sudah terbooking.
2. Program akan menampilkan output "Berikut adalah kamar yang terbooking".

11. Memilih Opsi Keluar

1. Jika user memilih opsi keluar, maka program akan menampilkan output "Terima kasih telah menggunakan aplikasi!" dan program akan berhenti.

12. Mengulangi Proses

1. Setelah menyelesaikan salah satu opsi, program akan kembali ke langkah 2 untuk meminta user memilih opsi lainnya hingga user memilih untuk keluar.

2.3 Konsep/Materi Praktikum yang dipakai

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

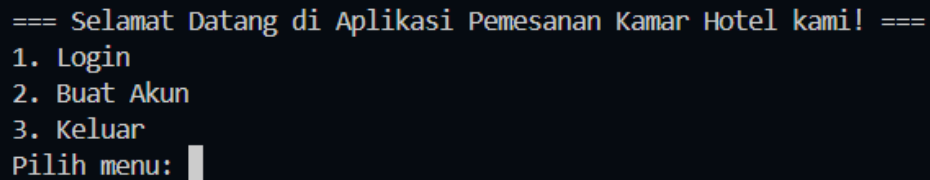
1. **Modul CSV:** Untuk membaca dan menulis file data.
2. **PrettyTable:** Untuk menampilkan data dalam bentuk tabel.
3. **Pwinput:** Untuk input password dengan keamanan lebih baik.

BAB III

HASIL DAN PEMBAHASAN

3.1 Tampilan Program

3.1.1 Tampilan Menu Utama

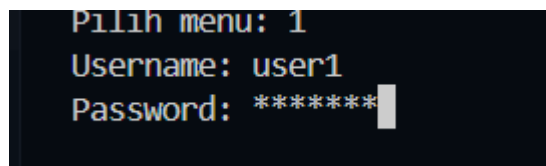


```
=== Selamat Datang di Aplikasi Pemesanan Kamar Hotel kami! ===  
1. Login  
2. Buat Akun  
3. Keluar  
Pilih menu: |
```

Gambar 3.1.1

Ini adalah tampilan menu utama yang berisi Login, Buat Akun, dan Keluar dari program.

3.1.2 Tampilan Login



```
Pilih menu: 1  
Username: user1  
Password: *****|
```

Gambar 3.1.2

Ini adalah tampilan menu login yaitu memasukkan username dan password yang di hide menggunakan pwinput.

3.1.3 Tampilan Pembuatan Akun

```
Pilih menu: 2
Masukkan username baru: user2
Masukkan password baru: *****
Akun berhasil dibuat!
```

Gambar 3.1.3

Ini adalah tampilan Pembuatan akun yaitu memasukkan Username dan Password baru yang ingin dibuat.

3.1.4 Tampilan Keluar di Menu Utama

```
Pilih menu: 3
Terima kasih telah menggunakan aplikasi ini.
```

Gambar 3.1.4

Ini adalah tampilan Keluar yang berarti keluar program/close program di menu utama

3.1.5 Tampilan Menu Admin

```
Username: admin
Password: *****
Login berhasil! Selamat datang, Admin.

=== Menu Admin ===
1. Tambah Kamar
2. Lihat Kamar
3. Ubah Kamar
4. Hapus Kamar
0. Keluar
Pilih menu: █
```

Gambar 3.1.5

Ini adalah tampilan menu admin, yang dimana admin memiliki Username dan Password khusus didalam codingan.

3.1.6 Tampilan Tambah Kamar di Menu Admin

```
Nomor Kamar: 03
Deskripsi Fasilitas: tes
Tipe Kamar: Suite
Harga: rp 1500000
Kamar berhasil ditambahkan!
```

Gambar 3.1.6

Ini adalah tampilan Tambah Kamar yang menginputkan Nomor Kamar, Deskripsi Fasilitas, Tipe Kamar, dan Harga.

3.1.7 Tampilan Lihat Kamar di Menu Admin

```
=== Pilih Opsi ===
1. Lihat Semua Kamar
2. Lihat Kamar Terbooking
Pilih opsi: 1
+-----+-----+-----+-----+-----+
| Nomor Kamar | Deskripsi | Tipe Kamar | Harga | Status |
+-----+-----+-----+-----+-----+
| 01 | tes | standard | rp 500000 | Terbooking |
| 02 | tes | deluxe | rp 750000 | Tersedia |
| 03 | tes | Suite | rp 1500000 | Tersedia |
+-----+-----+-----+-----+-----+

=== Menu Admin ===
1. Tambah Kamar
2. Lihat Kamar
3. Ubah Kamar
4. Hapus Kamar
0. Keluar
Pilih menu: 2

=== Pilih Opsi ===
1. Lihat Semua Kamar
2. Lihat Kamar Terbooking
Pilih opsi: 2
+-----+-----+-----+-----+
| Nomor Kamar | Deskripsi | Tipe Kamar | Harga |
+-----+-----+-----+-----+
| 01 | tes | standard | rp 500000 |
+-----+-----+-----+-----+
```

Gambar 3.1.7

Ini adalah tampilan Lihat kamar di Menu Admin yang terdapat 2 opsi yaitu Lihat Semua Kamar dan Lihat Kamar Terbooking

3.1.8 Tampilan Ubah Kamar di Menu Admin

```
Nomor Kamar yang ingin diubah: 03
Deskripsi Fasilitas (tes): tes
Tipe Kamar (Suite): Superior
Harga (rp 1500000): rp 2000000
Kamar berhasil diubah.
```

Gambar 3.1.8

Ini adalah tampilan Ubah Kamar, untuk penginputan awalnya adalah Nomor Kamar dilanjutkan dengan pengubahan Deskripsi Fasilitas, Tipe Kamar, dan Harga.

3.1.9 Tampilan Hapus Kamar di Menu Admin

```
Nomor Kamar yang ingin dihapus: 03
Kamar berhasil dihapus.
```

Gambar 3.1.9

Ini adalah tampilan Hapus Kamar, untuk penghapusan cukup menginputkan kamar nomor berapa yang ingin dihapus.

3.1.10 Tampilan Menu User

```
Username: user2
Password: *****
Login berhasil! Selamat datang, user2.

=== Menu User ===
1. Lihat Kamar Tersedia
2. Booking Kamar
0. Keluar
Pilih menu: █
```

Gambar 3.1.10

Ini adalah tampilan menu user yang terdapat 2 opsi yaitu Lihat Kamar Tersedia dan Booking Kamar.

3.1.11 Tampilan Lihat Kamar Tersedia di Menu User

```
=== Menu User ===
1. Lihat Kamar Tersedia
2. Booking Kamar
0. Keluar
Pilih menu: 1

+-----+-----+-----+-----+
| Nomor Kamar | Deskripsi | Tipe Kamar | Harga |
+-----+-----+-----+-----+
|      02     |    tes   |   deluxe   | rp 750000 |
+-----+-----+-----+-----+
```

Gambar 3.1.11

Ini adalah tampilan Lihat Kamar Tersedia di Menu user, ini menampilkan kamar yang tersedia atau enable

3.1.12 Tampilan Booking Kamar di Menu Admin

```
=== Menu User ===
1. Lihat Kamar Tersedia
2. Booking Kamar
0. Keluar
Pilih menu: 1

+-----+-----+-----+-----+
| Nomor Kamar | Deskripsi | Tipe Kamar | Harga |
+-----+-----+-----+-----+
|      02     |    tes   |   deluxe   | rp 750000 |
+-----+-----+-----+-----+
```

Gambar 3.1.12

Ini adalah tampilan Booking Kamar, jika Kamar terbooking maka otomatis kamar menjadi disable dan hilang di Lihat Kamar Tersedia.

3.2 Source Code

```
import csv
import os
from prettytable import PrettyTable
import pwininput

# Nama file CSV untuk data kamar dan akun
csv_file = 'kamar.csv'
akun_file = 'akun.csv'

# Akun admin khusus
ADMIN_USERNAME = "admin"
ADMIN_PASSWORD = "admin123"

# Fungsi untuk memastikan file CSV ada
def initialize_csv():
    if not os.path.exists(csv_file):
        with open(csv_file, 'w', newline='') as file:
            writer = csv.writer(file)
            writer.writerow(['nomor_kamar', 'deskripsi_fasilitas',
'tipe_kamar', 'harga', 'disabled'])

# Fungsi untuk memastikan file akun CSV ada
def initialize_akun_csv():
    if not os.path.exists(akun_file):
        with open(akun_file, 'w', newline='') as file:
            writer = csv.writer(file)
            writer.writerow(['username', 'password', 'role'])

# Fungsi untuk membaca data akun dari CSV
```

```

def baca_akun():
    akun = []
    try:
        with open(akun_file, 'r') as file:
            reader = csv.reader(file)
            next(reader) # Skip header
            for row in reader:
                akun.append(row)
    except FileNotFoundError:
        print("File CSV tidak ditemukan. Pastikan file ada atau jalankan 'initialize_akun_csv' untuk membuat file.")
    return akun

# Fungsi untuk membaca data kamar dari CSV
def baca_kamar():
    kamar = []
    try:
        with open(csv_file, 'r') as file:
            reader = csv.DictReader(file)
            for row in reader:
                kamar.append(row)
    except FileNotFoundError:
        print("File CSV kamar tidak ditemukan. Pastikan file ada atau jalankan 'initialize_csv' untuk membuat file.")
    return kamar

# Fungsi untuk menulis data akun ke CSV
def tulis_akun(akun):
    try:
        with open(akun_file, 'w', newline='') as file:
            fieldnames = ['username', 'password', 'role']
            writer = csv.DictWriter(file, fieldnames=fieldnames)

```

```

        writer.writeheader()
        writer.writerows(akun)
    except Exception as e:
        print("Error saat menulis file CSV akun:", e)

# Fungsi untuk menulis data kamar ke CSV
def tulis_kamar(kamar):
    try:
        with open(csv_file, 'w', newline='') as file:
            fieldnames = ['nomor_kamar', 'deskripsi_fasilitas',
            'tipe_kamar', 'harga', 'disabled']
            writer = csv.DictWriter(file, fieldnames=fieldnames)
            writer.writeheader()
            writer.writerows(kamar)
    except Exception as e:
        print("Error saat menulis file CSV kamar:", e)

# Fungsi untuk membuat akun baru dengan pengecekan duplikasi username
def buat_akun():
    username = input("Masukkan username baru: ")

    # Membaca semua username yang ada di akun.csv
    akun = baca_akun()
    usernames = [row[0] for row in akun]

    # Periksa apakah username sudah ada (termasuk admin)
    if username == ADMIN_USERNAME:
        print("Username sudah digunakan untuk admin. Silakan pilih
        username lain.")
        return
    if username in usernames:

```

```

        print("Username sudah digunakan. Silakan pilih username
lain.")

        return

    # Jika username tersedia, Lanjutkan membuat akun baru
    password = pwinput.pwinput("Masukkan password baru: ") #
Menggunakan pwinput
    role = "user" # Role default untuk akun baru
    with open(akun_file, 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow([username, password, role])
    print("Akun berhasil dibuat!")

# Fungsi untuk Login dengan error handling jika password salah (3
percobaan)
def login():
    attempts = 3 # Maksimum percobaan Login
    while attempts > 0:
        username = input("Username: ")
        password = pwinput.pwinput("Password: ") # Menggunakan
pwinput untuk password

        # Login sebagai admin
        if username == ADMIN_USERNAME and password == ADMIN_PASSWORD:
            print("Login berhasil! Selamat datang, Admin.")
            menu_admin()
            return

        # Login sebagai user
        akun = baca_akun()
        for row in akun:

```

```

        if row[0] == username and row[1] == password and row[2]
== "user":

            print(f"Login berhasil! Selamat datang, {username}.")
            menu_user() # Menu untuk user yang sudah Login
            return

        attempts -= 1
        if attempts > 0:
            print(f"Username atau password salah. Percobaan {3 -
attempts} dari 3.")
        else:
            print("Login gagal. Percobaan login habis.")

# Fungsi untuk menu Admin
def menu_admin():
    while True:
        print("\n=== Menu Admin ===")
        print("1. Tambah Kamar")
        print("2. Lihat Kamar")
        print("3. Ubah Kamar")
        print("4. Hapus Kamar")
        print("0. Keluar")
        pilihan = input("Pilih menu: ")

        if pilihan == '1':
            nomor_kamar = input("Nomor Kamar: ")
            deskripsi_fasilitas = input("Deskripsi Fasilitas: ")
            tipe_kamar = input("Tipe Kamar: ")
            harga = input("Harga: ")
            tambah_kamar(nomor_kamar, deskripsi_fasilitas,
tipe_kamar, harga)
            elif pilihan == '2':

```

```

        print("\n=== Pilih Opsi ===")
        print("1. Lihat Semua Kamar")
        print("2. Lihat Kamar Terbooking")
        sub_pilihan = input("Pilih opsi: ")
        if sub_pilihan == '1':
            lihat_kamar()
        elif sub_pilihan == '2':
            lihat_kamar_terbooking()
        else:
            print("Pilihan tidak valid.")
    elif pilihan == '3':
        nomor_kamar = input("Nomor Kamar yang ingin diubah: ")
        ubah_detail_kamar(nomor_kamar)
    elif pilihan == '4':
        nomor_kamar = input("Nomor Kamar yang ingin dihapus: ")
        hapus_kamar(nomor_kamar)
    elif pilihan == '0':
        break
    else:
        print("Pilihan tidak valid.")

# Fungsi untuk menu User
def menu_user():
    while True:
        print("\n=== Menu User ===")
        print("1. Lihat Kamar Tersedia")
        print("2. Booking Kamar")
        print("0. Keluar")
        pilihan = input("Pilih menu: ")

        if pilihan == '1':
            lihat_kamar_tersedia()

```

```

        elif pilihan == '2':
            nomor_kamar = input("Nomor Kamar yang ingin dipesan: ")
            booking_kamar_user(nomor_kamar)
        elif pilihan == '0':
            break
        else:
            print("Pilihan tidak valid.")

# Fungsi untuk melihat kamar yang tersedia (tidak terbooking)
def lihat_kamar_tersedia():
    kamar = baca_kamar()
    tersedia = [k for k in kamar if k['disabled'] == 'enable']
    if tersedia:
        table = PrettyTable()
        table.field_names = ["Nomor Kamar", "Deskripsi", "Tipe Kamar", "Harga"]
        for k in tersedia:
            table.add_row([k['nomor_kamar'],
k['deskripsi_fasilitas'], k['tipe_kamar'], k['harga']])
        print(table)
    else:
        print("Tidak ada kamar yang tersedia.")

# Fungsi untuk booking kamar user
def booking_kamar_user(nomor_kamar):
    kamar = baca_kamar()
    kamar_ditemukan = False
    for k in kamar:
        if k['nomor_kamar'] == nomor_kamar:
            if k['disabled'] == 'disable':
                print(f"Kamar {nomor_kamar} sudah terbooking.")
            else:

```

```

        k['disabled'] = 'disable' # Tandai sebagai
    terbooking

    kamar_ditemukan = True
    print(f"Kamar {nomor_kamar} berhasil dipesan!")
    break
if not kamar_ditemukan:
    print("Mohon maaf kamar tidak tersedia.")
    tulis_kamar(kamar)

# Fungsi untuk melihat semua kamar (termasuk yang tersedia dan
    terbooking)
def lihat_kamar():
    kamar = baca_kamar()
    if kamar:
        table = PrettyTable()
        table.field_names = ["Nomor Kamar", "Deskripsi", "Tipe
Kamar", "Harga", "Status"]
        for k in kamar:
            status = "Tersedia" if k['disabled'] == 'enable' else
"Terbooking"
            table.add_row([k['nomor_kamar'],
k['deskripsi_fasilitas'], k['tipe_kamar'], k['harga'], status])
        print(table)
    else:
        print("Tidak ada kamar yang tersedia.")

# Fungsi untuk melihat kamar yang terbooking
def lihat_kamar_terbooking():
    kamar = baca_kamar()
    terbooking = [k for k in kamar if k['disabled'] == 'disable']
    if terbooking:
        table = PrettyTable()

```



```

        table.field_names = ["Nomor Kamar", "Deskripsi", "Tipe
Kamar", "Harga"]
        for k in terbooking:
            table.add_row([k['nomor_kamar'],
k['deskripsi_fasilitas'], k['tipe_kamar'], k['harga']])
        print(table)
    else:
        print("Belum ada kamar yang terbooking.")

# Fungsi untuk mengubah detail kamar
def ubah_detail_kamar(nomor_kamar):
    kamar = baca_kamar()
    kamar_ditemukan = False
    for k in kamar:
        if k['nomor_kamar'] == nomor_kamar:
            k['deskripsi_fasilitas'] = input(f"Deskripsi Fasilitas
({k['deskripsi_fasilitas']}): ") or k['deskripsi_fasilitas']
            k['tipe_kamar'] = input(f"Tipe Kamar ({k['tipe_kamar']}):
") or k['tipe_kamar']
            k['harga'] = input(f"Harga ({k['harga']}): ") or
k['harga']
            kamar_ditemukan = True
            print("Kamar berhasil diubah.")
            break
    if not kamar_ditemukan:
        print("Kamar tidak ditemukan.")
    tulis_kamar(kamar)

# Fungsi untuk menambah kamar baru
def tambah_kamar(nomor_kamar, deskripsi_fasilitas, tipe_kamar,
harga):
    kamar = baca_kamar()

```

```

    kamar_baru = {
        'nomor_kamar': nomor_kamar,
        'deskripsi_fasilitas': deskripsi_fasilitas,
        'tipe_kamar': tipe_kamar,
        'harga': harga,
        'disabled': 'enable'
    }
    kamar.append(kamar_baru)
    tulis_kamar(kamar)
    print("Kamar berhasil ditambahkan!")

# Fungsi untuk menghapus kamar
def hapus_kamar(nomor_kamar):
    kamar = baca_kamar()
    kamar_baru = [k for k in kamar if k['nomor_kamar'] !=
nomor_kamar]
    if len(kamar_baru) < len(kamar):
        tulis_kamar(kamar_baru)
        print("Kamar berhasil dihapus.")
    else:
        print("Kamar tidak ditemukan.")

# Program utama
def main():
    initialize_csv()
    initialize_akun_csv()

    while True:
        print("\n=== Selamat Datang di Aplikasi Pemesanan Kamar Hotel
kami! ===")
        print("1. Login")
        print("2. Buat Akun")

```

```
print("3. Keluar")
pilihan = input("Pilih menu: ")

if pilihan == '1':
    login()
elif pilihan == '2':
    buat_akun()
elif pilihan == '3':
    print("Terima kasih telah menggunakan aplikasi ini.")
    break
else:
    print("Pilihan tidak valid.")

# Main untuk menjalankan program
if __name__ == "__main__":
    main()
```

BAB IV

PENUTUP

4.1 Kesimpulan

Program ini berhasil menyediakan fitur dasar untuk mengelola data hotel dengan mudah.

4.2 Saran

Pengembangan lebih lanjut dapat mencakup penggunaan basis data relasional dan antarmuka grafis.

DAFTAR PUSTAKA

Referensi buku ataupun website yang dibuat panduan selama pengerjaan proyek dan laporan.

1. Dokumentasi Python (<https://docs.python.org>)
2. Modul PrettyTable (<https://pypi.org/project/PrettyTable/>)
3. Dokumentasi Pwinput. <https://pypi.org/project/pwinput/>

LAMPIRAN

