

**LAPORAN PROYEK AKHIR**  
**DASAR DASAR PEMROGRAMAN**  
**SISTEM RESERVASI DAN PELAYANAN PETCARE**



**Disusun oleh:**

**KELOMPOK 11**

Andhika Fhariel Fadhlurrohman	2509116083
Rasya Aditia Ramadani	2509116082
Meilanie	2509116109

**Asisten Laboratorium**

**Taufik Ramadhani**  
**2409116001**

**Dwi Pebrianto Pradana**  
**2409116012**

**PROGRAM STUDI SISTEM INFORMASI**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS MULAWARMAN**  
**2025**

## **KATA PENGANTAR**

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan tugas praktikum mata kuliah Dasar-Dasar Pemrograman dengan judul “Sistem Reservasi dan Pelayanan PetCare” tepat pada waktunya.

Tugas ini disusun sebagai salah satu bentuk penerapan materi yang telah dipelajari dalam kegiatan praktikum, khususnya mengenai pembuatan program berbasis logika, algoritma, dan struktur data sederhana. Melalui tema Sistem Reservasi dan Pelayanan PetCare, penulis berupaya merancang sebuah sistem yang dapat membantu proses pemesanan layanan perawatan hewan peliharaan secara lebih mudah, cepat, dan efisien.

Penyusun menyadari bahwa tugas ini masih jauh dari sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun dari asisten praktikum maupun rekan mahasiswa agar karya ini dapat menjadi lebih baik di masa mendatang.

Akhir kata, penyusun mengucapkan terima kasih kepada dosen pengampu dan asisten praktikum mata kuliah Dasar-Dasar Pemrograman, serta semua pihak yang telah memberikan dukungan, bantuan, dan motivasi dalam penyusunan tugas ini. Semoga laporan ini dapat memberikan manfaat dan wawasan bagi pembaca.

Oktober,2025

Penyusun

## DAFTAR ISI

KATA PENGANTAR .....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR .....	iii
DAFTAR LAMPIRAN.....	iv
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan .....	3
1.5 Manfaat.....	3
BAB II.....	4
PEMBAHASAN .....	4
2.1 Analisis Sistem .....	4
2.2 Flowchart Sistem .....	4
2.2.1 Flowchart Login Utama.....	4
2.2.2 flowchart pengisian saldo .....	6
2.2.3 Flowchart Menu User .....	8
2.2.4 Flowchart Menu Admin.....	10
2.3 Implementasi dan penjelasan Program (Coding).....	12
2.3.1 Kode Utama Program.....	12
BAB III .....	18
KESIMPULAN .....	18

## DAFTAR GAMBAR

Gambar 1 flowchart proses login.....	4
Gambar 2 flowchart menu login 2 .....	5
Gambar 3 flowchart proses pengisian saldo.....	6
Gambar 4 flowchart menu pengguna .....	8
Gambar 5 flowchart menu admin .....	10
Gambar 6 codingan import library .....	12
Gambar 7 codingan deklarasi file dan konstanta .....	12
Gambar 8 codingan data awal pengguna .....	13
Gambar 9 codingan fungsi utilitas(pendukung) .....	13
Gambar 10 codingan fungsi csv.....	13
Gambar 11 codingan manajemen pengguna .....	14
Gambar 12 codingan data hewan (petcare) .....	14
Gambar 13 codingan reverensi dan invoice .....	14
Gambar 14 codingan format tabel (PrettyTable).....	14
Gambar 15 codingan login sistem.....	15
Gambar 16 codingan menu admin .....	15
Gambar 17 codingan menu pengguna.....	16
Gambar 18 codingan fungsi utama (main program) .....	16
Gambar 19 codingan jalankan program .....	17

## **DAFTAR LAMPIRAN**

<b>lampiran 1 dokumentasi proses pengerjaan PA.....</b>	<b>19</b>
<b>lampiran 2 dokumentasi proses pengerjaan PA.....</b>	<b>20</b>
<b>lampiran 3 dokumentasi proses pengerjaan PA.....</b>	<b>20</b>
<b>lampiran 5 dokumentasi konsul 1.....</b>	<b>20</b>
<b>lampiran 6 dokumentasi konsul 2.....</b>	<b>21</b>
<b>lampiran 7 dokumentasi proses pengerjaan PA.....</b>	<b>21</b>

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perkembangan teknologi informasi yang semakin pesat telah membawa perubahan signifikan dalam berbagai aspek kehidupan, termasuk sektor pelayanan jasa. Salah satu sektor yang mulai berkembang pesat adalah layanan PetCare, yaitu layanan perawatan hewan peliharaan seperti grooming, vaksinasi, penitipan, dan konsultasi kesehatan. Peningkatan jumlah pemilik hewan peliharaan di masyarakat menuntut adanya sistem pelayanan yang cepat, efisien, dan terorganisir dengan baik.

Namun, pada kenyataannya masih banyak tempat PetCare yang menjalankan sistem secara manual dalam mencatat data pelanggan, jadwal layanan, serta transaksi pembayaran. Hal tersebut dapat menimbulkan berbagai permasalahan seperti kehilangan data, kesalahan pencatatan, serta pelayanan yang kurang optimal. Untuk mengatasi hal tersebut, dibutuhkan sebuah sistem reservasi dan pelayanan PetCare berbasis komputer yang mampu mengotomatisasi proses tersebut agar lebih efektif dan efisien.

Bahasa pemrograman Python dipilih karena memiliki sintaks yang sederhana dan mudah dipahami, serta didukung oleh berbagai library yang dapat membantu proses pengembangan aplikasi, seperti PrettyTable untuk menampilkan tabel data dan CSV untuk penyimpanan data dinamis. Sistem ini dirancang untuk memiliki dua peran utama, yaitu Admin dan User, dengan penerapan sistem CRUD (Create, Read, Update, Delete), fitur transaksi menggunakan saldo e-money, serta pembuatan invoice otomatis setelah transaksi berhasil.

Dengan adanya sistem ini, diharapkan proses pelayanan di tempat PetCare dapat berjalan lebih cepat, akurat, dan teratur. Selain itu, sistem ini juga dapat membantu pemilik usaha PetCare dalam mengelola data pelanggan, layanan, dan transaksi dengan lebih mudah serta mengurangi kemungkinan terjadinya kesalahan manusia (human error).

## **1.2 Rumusan Masalah**

Berdasarkan uraian latar belakang di atas, maka dapat dirumuskan beberapa permasalahan sebagai berikut:

1. Bagaimana merancang dan mengimplementasikan sistem reservasi dan pelayanan PetCare menggunakan bahasa Python?
2. Bagaimana menerapkan sistem CRUD dengan dua peran pengguna (Admin dan User)?
3. Bagaimana sistem dapat melakukan transaksi menggunakan saldo e-money serta menghasilkan invoice otomatis setelah transaksi dilakukan?
4. Bagaimana cara menyimpan dan menampilkan data pengguna, layanan, serta transaksi secara dinamis menggunakan format CSV?

## **1.3 Batasan Masalah**

Agar pembahasan lebih fokus dan tidak meluas, maka penelitian ini dibatasi pada hal-hal berikut:

1. Sistem hanya dijalankan melalui terminal (CLI), belum menggunakan antarmuka grafis (GUI).
2. Data pengguna, layanan, dan transaksi disimpan dalam format CSV, bukan dalam basis data (database) seperti MySQL atau SQLite.
3. Sistem hanya memiliki dua peran pengguna, yaitu Admin (akses penuh) dan User (akses terbatas).
4. Transaksi hanya menggunakan saldo virtual atau e-money internal sistem, bukan transaksi online sebenarnya.
5. Jumlah layanan yang disediakan minimal terdiri dari lima jenis (misalnya: grooming, vaksin, penitipan, konsultasi, dan pemeriksaan kesehatan).
6. Sistem belum mendukung fitur notifikasi otomatis atau pengingat jadwal.

## **1.4 Tujuan**

Tujuan dari pembuatan sistem ini adalah sebagai berikut:

1. Menerapkan konsep dasar pemrograman Python dalam pembuatan proyek sistem nyata.
2. Merancang sistem reservasi dan pelayanan PetCare yang membantu proses administrasi menjadi lebih efektif dan efisien.
3. Mengimplementasikan sistem CRUD dengan pembagian peran pengguna (Admin dan User) sesuai hak aksesnya.
4. Menyediakan fitur transaksi dengan saldo e-money serta menampilkan invoice setelah transaksi berhasil.
5. Menerapkan penyimpanan data dinamis menggunakan format CSV agar data tetap tersimpan meskipun program ditutup.
6. Menggunakan library tambahan seperti PrettyTable untuk menampilkan tabel data dan getpass untuk keamanan login.

## **1.5 Manfaat**

Pembuatan sistem reservasi dan pelayanan PetCare ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memberikan pengalaman nyata dalam menerapkan konsep dasar pemrograman Python, seperti penggunaan variabel, fungsi, percabangan, perulangan, serta sistem CRUD.
2. Membantu pengelola layanan PetCare dalam mengatur data reservasi, pelanggan, dan transaksi secara lebih efisien dan terorganisir.
3. Menjadi dasar pengembangan sistem yang lebih kompleks di masa mendatang, seperti versi berbasis web atau aplikasi mobile.



## BAB II

### PEMBAHASAN

#### 2.1 Analisis Sistem

Sistem reservasi dan pelayanan PetCare merupakan sistem berbasis program Python yang bertujuan untuk mempermudah proses pemesanan layanan hewan peliharaan, seperti grooming, konsultasi dokter, atau penitipan.

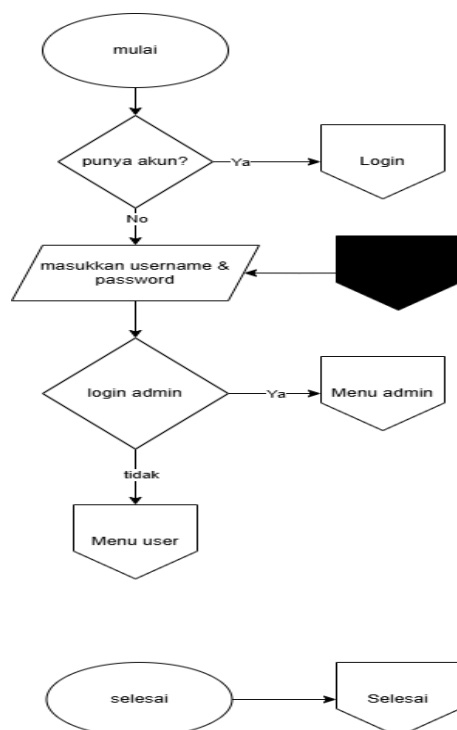
Sistem ini dirancang agar dapat digunakan oleh dua jenis pengguna, yaitu admin dan user (pelanggan). Admin berfungsi mengelola data layanan, sedangkan user dapat melakukan reservasi dan melihat informasi layanan.

Sebelum pembuatan program dilakukan, diperlukan rancangan alur kerja sistem dalam bentuk flowchart, agar proses login, pembuatan akun, hingga pengelolaan data dapat terstruktur dengan jelas.

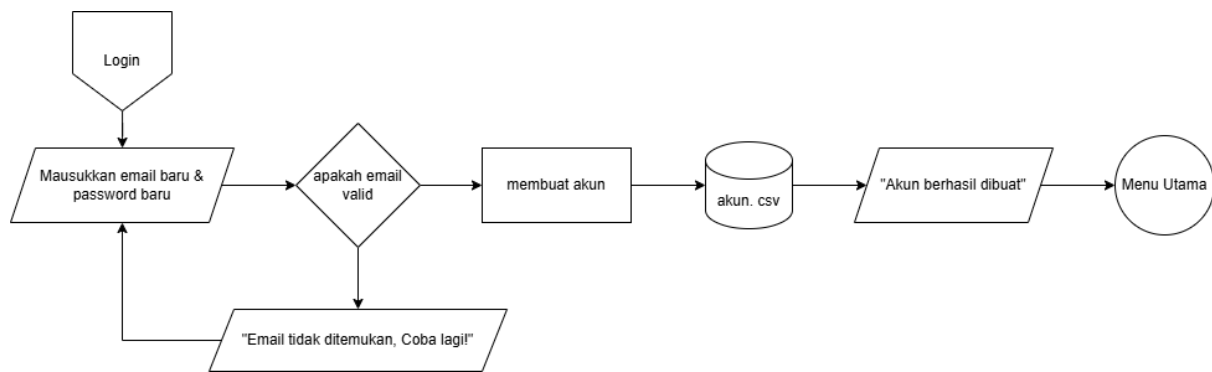
#### 2.2 Flowchart Sistem

Berikut merupakan beberapa flowchart utama dalam sistem reservasi dan pelayanan PetCare:

##### 2.2.1 Flowchart Login Utama



*Gambar 1 flowchart proses login*



Gambar 2 flowchart menu login 2

Penjelasan:

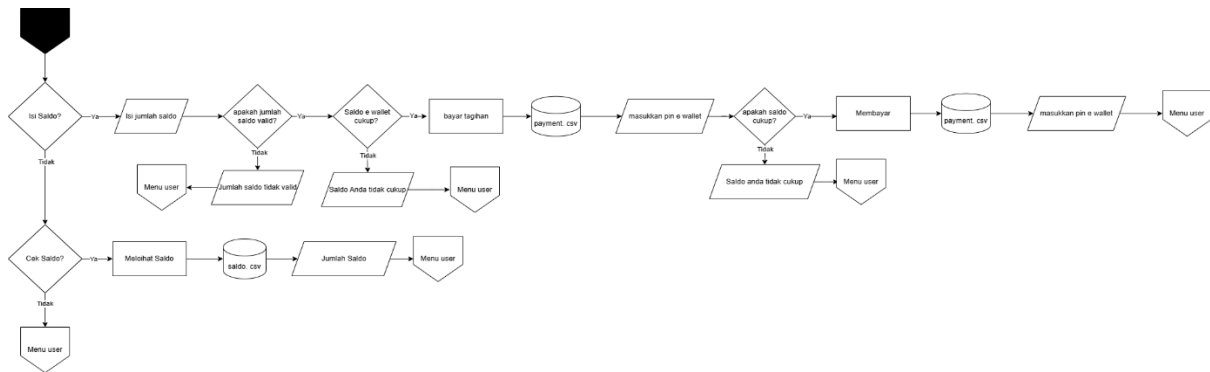
Flowchart ini menggambarkan proses alur login dan akses menu pada sebuah system yang pertama (1)

1. Mulai: Proses dimulai.
2. Punya Akun?: Sistem memeriksa apakah pengguna sudah memiliki akun.
  - Ya: Jika sudah, pengguna diarahkan ke proses Login.
  - Tidak: Jika belum, pengguna diminta untuk memasukkan username dan password.
3. Masukkan Username & Password: Pengguna memasukkan informasi login mereka.
4. Login Admin?: Sistem memeriksa apakah login tersebut adalah login admin.
  - Ya: Jika ya, pengguna diarahkan ke Menu Admin.
  - Tidak: Jika bukan, pengguna diarahkan ke Menu User.
5. Menu Admin/Menu User: Pengguna mengakses menu yang sesuai dengan hak akses mereka.
6. Selesai: Proses selesai. Flowchart ini menggambarkan proses awal ketika pengguna menjalankan program. Sistem akan menanyakan apakah pengguna sudah memiliki akun atau belum.

Flowchart ini menggambarkan proses pembuatan akun baru yang kedua (2) setelah login akun

1. Login: Proses dimulai dengan langkah login.
2. Masukkan Email dan Password Baru: Pengguna diminta memasukkan email dan password baru.
3. Validasi Email: Sistem akan memeriksa apakah email yang dimasukkan valid.
4. Jika Email Tidak Valid: Jika email tidak ditemukan atau tidak valid, sistem akan menampilkan pesan "Email tidak ditemukan, Coba lagi!" dan pengguna akan kembali ke langkah memasukkan email dan password baru.
5. Membuat Akun: Jika email valid, sistem akan membuat akun baru.
6. akun.csv: Data akun baru disimpan dalam file "akun.csv".
7. Akun Berhasil Dibuat: Sistem menampilkan pesan "Akun berhasil dibuat".
8. Menu Utama: Pengguna diarahkan ke menu utama aplikasi.

### 2.2.2 flowchart pengisian saldo



Gambar 3 flowchart proses pengisian saldo

#### Penjelasan:

Flowchart ini menggambarkan alur proses pengisian saldo, pengecekan saldo, dan pembayaran melalui e-wallet

1. Isi Saldo?: Pengguna memilih untuk mengisi saldo atau tidak.
  - Ya: Pengguna memasukkan jumlah saldo yang ingin diisi. Sistem kemudian memvalidasi apakah jumlah saldo tersebut valid atau tidak. Jika tidak valid, pengguna kembali ke Menu User.
  - Tidak: Pengguna melanjutkan ke pilihan Cek Saldo.
2. Cek Saldo?: Pengguna memilih untuk mengecek saldo atau tidak.
  - Ya: Sistem menampilkan jumlah saldo yang tersimpan dalam file saldo.csv, lalu kembali ke Menu User.
  - Tidak: Pengguna tetap berada di Menu User.
3. Saldo e-Wallet Cukup?: Setelah mengisi saldo, jika pengguna ingin melakukan pembayaran, sistem akan memeriksa apakah saldo e-wallet mencukupi untuk membayar tagihan.
  - Ya: Pengguna dapat membayar tagihan. Informasi pembayaran disimpan dalam file payment.csv. Setelah itu, pengguna diminta memasukkan PIN e-wallet untuk verifikasi.
  - Tidak: Jika saldo tidak mencukupi, pengguna akan mendapat pemberitahuan "Saldo Anda tidak cukup" dan kembali ke Menu User.
4. Apakah Saldo Cukup?: Setelah memasukkan PIN, sistem kembali memeriksa apakah saldo mencukupi untuk pembayaran.
  - Ya: Sistem memproses pembayaran dan menyimpan informasi dalam file payment.csv, lalu kembali ke Menu User.
  - Tidak: Jika saldo tetap tidak mencukupi, pengguna akan mendapat pemberitahuan "Saldo Anda tidak cukup" dan kembali ke Menu User.

Flowchart ini menjelaskan proses pembuatan akun baru bagi pengguna yang belum memiliki akun.

Pengguna diminta memasukkan email baru dan password baru.

Sistem kemudian memeriksa apakah email yang dimasukkan valid.

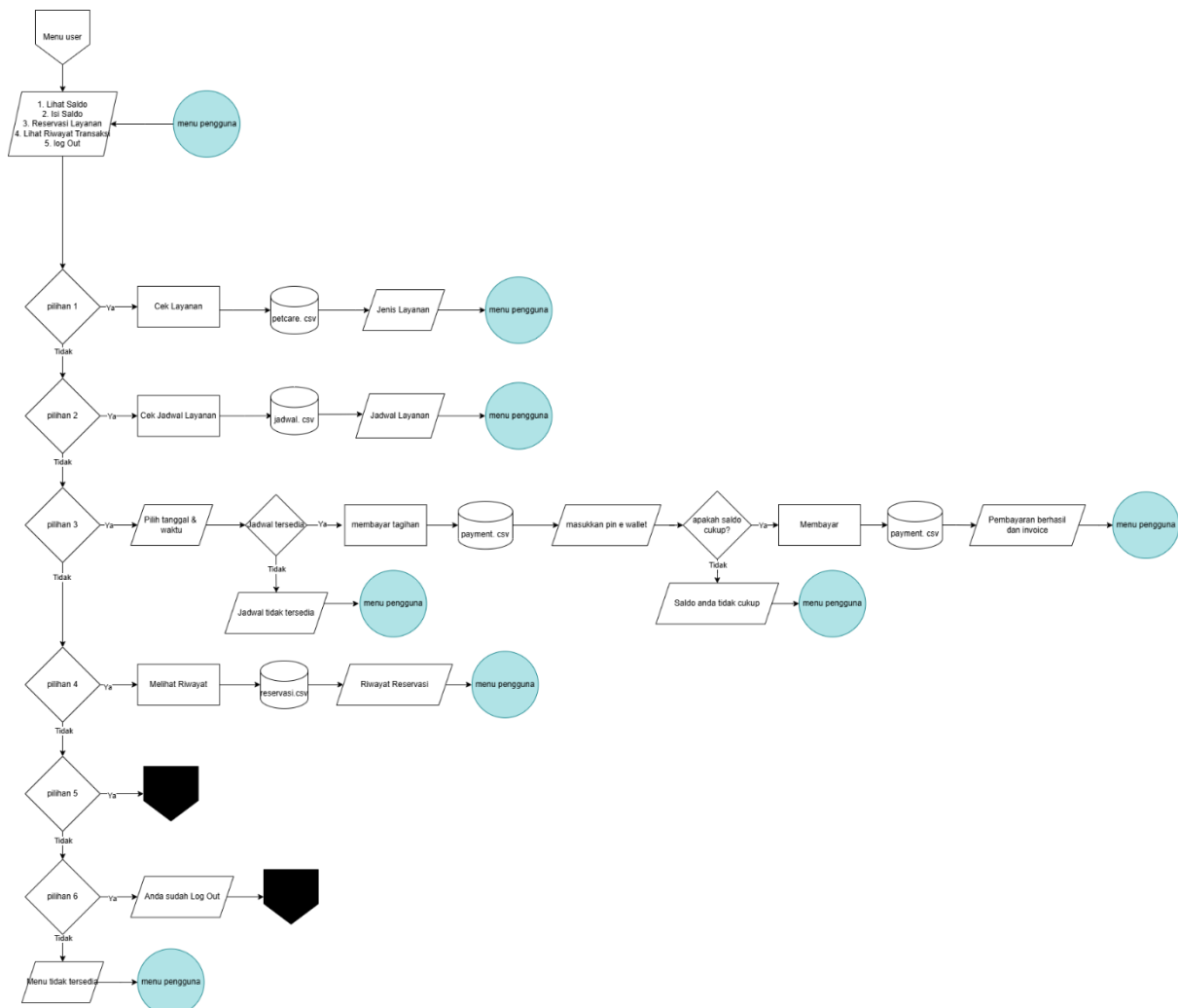
Jika valid, sistem akan menyimpan data akun baru ke file akun.csv dan menampilkan pesan “Akun berhasil dibuat”.

Jika tidak valid, sistem akan menampilkan pesan kesalahan dan meminta pengguna untuk mengulangi proses input email.

Setelah akun berhasil dibuat, pengguna akan diarahkan ke menu utama untuk login kembali.

Flowchart ini memastikan bahwa proses pendaftaran dilakukan dengan validasi data agar tidak terjadi duplikasi atau kesalahan akun.

### 2.2.3 Flowchart Menu User



Gambar 4 flowchart menu pengguna

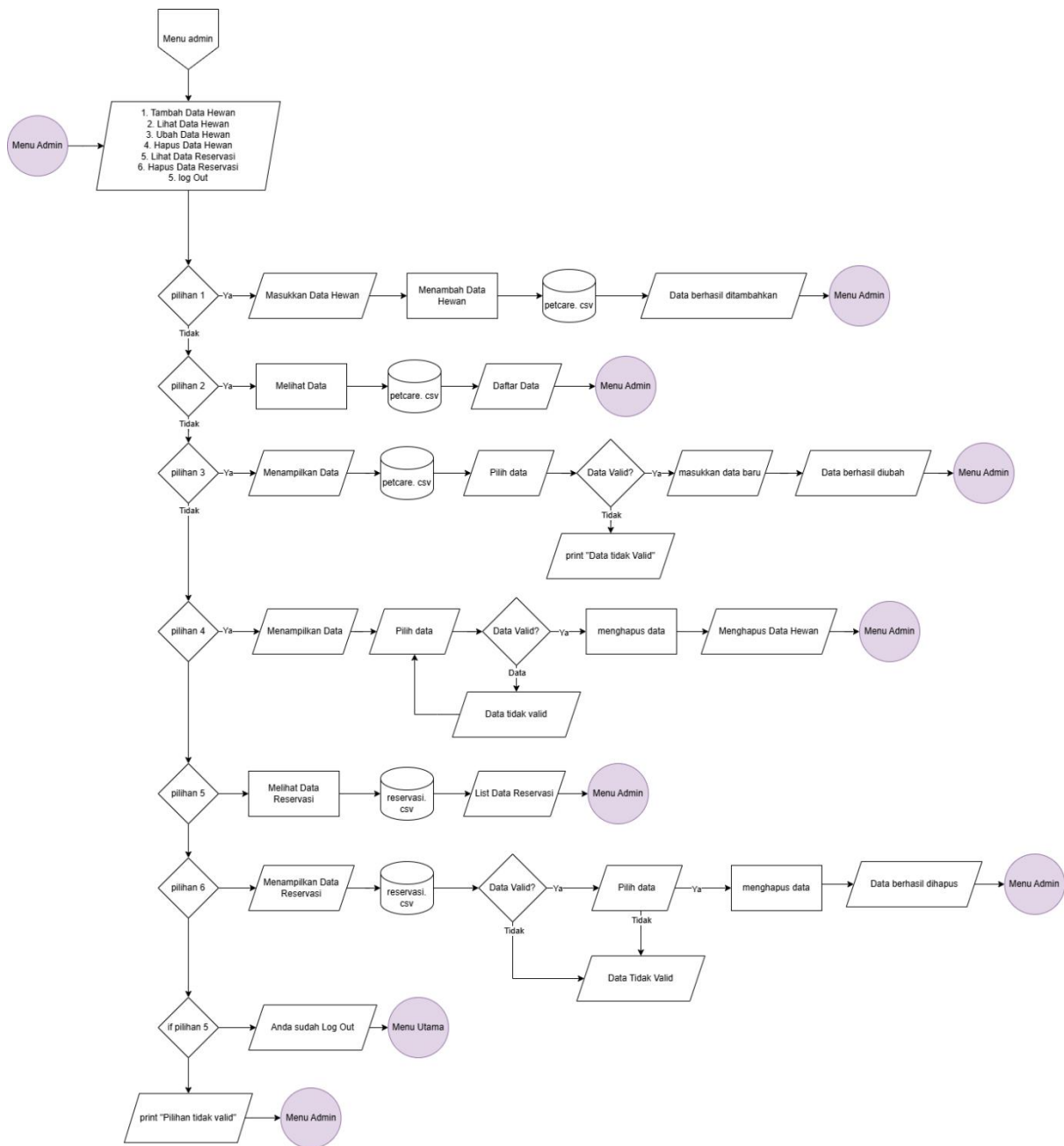
Penjelasan;

Flowchart ini menggambarkan alur penggunaan aplikasi atau sistem yang menyediakan layanan pet care (perawatan hewan peliharaan).

1. Menu User: Pengguna masuk ke menu utama yang menyediakan beberapa opsi:
  - Cek layanan
  - Cek jadwal layanan
  - Reservasi
  - Isi saldo
  - Cek saldo
  - Log out

2. Pilihan 1 (Cek Layanan): Jika pengguna memilih untuk cek layanan, sistem akan menampilkan jenis-jenis layanan dari file `petcare.csv` .
3. Pilihan 2 (Cek Jadwal Layanan): Jika pengguna memilih untuk cek jadwal layanan, sistem akan menampilkan jadwal layanan dari file `jadwal.csv` .
4. Pilihan 3 (Reservasi): Jika pengguna memilih reservasi:
  - Pengguna memilih tanggal dan waktu.
  - Sistem memeriksa apakah jadwal tersebut tersedia. Jika tidak, akan muncul pemberitahuan "Jadwal tidak tersedia" dan kembali ke menu pengguna.
  - Jika jadwal tersedia, pengguna dapat membayar tagihan.
  - Data pembayaran disimpan dalam `payment.csv` .
  - Pengguna memasukkan PIN e-wallet.
  - Sistem memeriksa apakah saldo cukup. Jika tidak, akan muncul pesan "Saldo Anda tidak cukup" dan kembali ke menu pengguna.
  - Jika saldo cukup, pembayaran dilakukan dan data pembayaran disimpan dalam `payment.csv`
  - Setelah pembayaran berhasil, pengguna kembali ke menu pengguna.
5. Pilihan 4: melihat Riwayat pemesanan
6. Pilihan 5: setelah melihat Riwayat pemesanan lalu pilih jika ya lanjut ke menu selanjutnya, jika tidak keluar
7. Pilihan 6: jika memilih ya (Log Out): Jika tidak menampilkan menu tidak tersedia pengguna memilih log out.

## 2.2.4 Flowchart Menu Admin



Gambar 5 flowchart menu admin

Penjelasan:

Flowchart ini menggambarkan alur menu admin pada suatu sistem, yang memungkinkan admin untuk mengelola data hewan dan data reservasi.

1. Menu Admin:
  - Admin memiliki beberapa opsi:
  - Tambah Data Hewan
  - Lihat Data Hewan
  - Ubah Data Hewan
  - Hapus Data Hewan
  - Lihat Data Reservasi
  - Hapus Data Reservasi
  - Log Out
2. Pilihan 1 (Tambah Data Hewan): Admin memasukkan data hewan baru, yang kemudian ditambahkan ke dalam file `petcare.csv` . Setelah data berhasil ditambahkan, admin kembali ke Menu Admin.
3. Pilihan 2 (Lihat Data Hewan): Sistem menampilkan daftar data hewan yang ada di dalam file `petcare.csv` . Admin kemudian kembali ke Menu Admin.
4. Pilihan 3 (Ubah Data Hewan):
  - Sistem menampilkan data hewan dari file `petcare.csv` .
  - Admin memilih data yang ingin diubah.
  - Sistem memvalidasi apakah data yang dipilih valid.
  - Jika valid, admin memasukkan data baru untuk menggantikan data lama. Setelah data berhasil diubah, admin kembali ke Menu Admin.
  - Jika tidak valid, sistem menampilkan pesan "Data Tidak Valid" dan admin kembali ke Menu Admin.
5. Pilihan 4 (Hapus Data Hewan):
  - Sistem menampilkan data hewan.
  - Admin memilih data yang ingin dihapus.
  - Sistem memvalidasi apakah data yang dipilih valid.
  - Jika valid, admin menghapus data hewan tersebut. Setelah data berhasil dihapus, admin kembali ke Menu Admin.
  - Jika tidak valid, sistem menampilkan pesan "Data Tidak Valid".
6. Pilihan 5 (Lihat Data Reservasi): Sistem menampilkan daftar data reservasi yang ada di dalam file `reservasi.csv` . Admin kemudian kembali ke Menu Admin.
7. Pilihan 6 (Hapus Data Reservasi): Sistem menampilkan data reservasi.
  - Admin memilih data yang ingin dihapus.
  - Sistem memvalidasi apakah data yang dipilih valid.
  - Jika valid, admin menghapus data reservasi tersebut. Setelah data berhasil dihapus, admin kembali ke Menu Admin.
  - Jika tidak valid, sistem menampilkan pesan "Data Tidak Valid".



8. Pilihan 7 (Log Out): Admin memilih opsi ini untuk keluar dari Menu Admin dan kembali ke Menu Utama.

Jika ada pilihan yang tidak valid, sistem akan menampilkan pesan "Pilihan tidak valid" dan admin kembali ke Menu Admin.

## 2.3 Implementasi dan penjelasan Program (Coding)

Program ini adalah sistem reservasi dan pelayanan PetCare yang dibangun menggunakan bahasa Python.

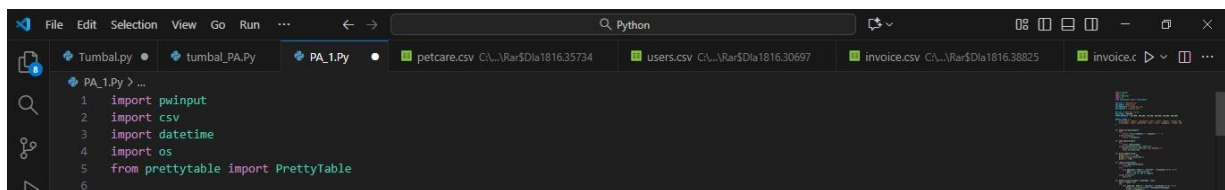
Tujuannya adalah untuk mengelola data pelanggan, layanan, transaksi, dan saldo secara otomatis tanpa perlu mencatat manual.

Program memiliki dua jenis pengguna:

- Admin → dapat mengelola data hewan dan melihat data reservasi.
- Pengguna (User) → dapat melakukan top-up saldo, melakukan reservasi layanan, dan melihat riwayat transaksi.

Data disimpan dalam file CSV agar tetap tersimpan meskipun program ditutup. Program dibuat menggunakan Python dengan library tambahan seperti:

### 2.3.1 Kode Utama Program



Gambar 6 codingan import library

Penjelasan:

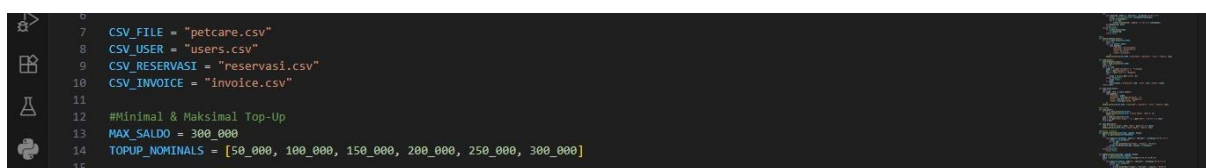
pwinput → untuk input password tanpa menampilkan karakter di layar (keamanan login).

csv → untuk membaca dan menulis data ke file .csv.

datetime → untuk mencatat tanggal & waktu transaksi.

os → untuk pengecekan dan manipulasi file (misal: membuat file baru, mengganti nama file).

PrettyTable → untuk menampilkan data dalam bentuk tabel agar lebih rapi di terminal.



Gambar 7 codingan deklarasi file dan konstanta

Penjelasan:

Program menyimpan data dalam empat file CSV:

petcare.csv → data hewan.

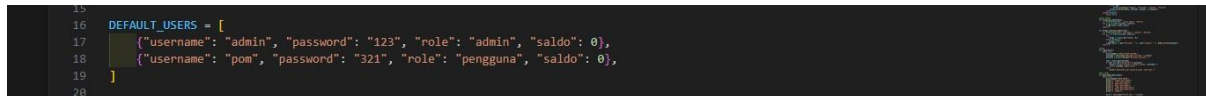
users.csv → data pengguna (username, password, role, saldo).

reservasi.csv → data pemesanan layanan.

invoice.csv → catatan transaksi/invoice.

MAX\_SALDO → batas maksimal saldo yang bisa dimiliki pengguna.

TOPUP\_NOMINALS → daftar nominal top-up yang bisa dipilih pengguna

A screenshot of a code editor showing Python code. Lines 16 to 19 define a list named 'DEFAULT\_USERS' containing two dictionaries. The first dictionary represents an admin user with username 'admin', password '123', role 'admin', and saldo 0. The second dictionary represents a regular user with username 'pom', password '321', role 'pengguna', and saldo 0. Line 20 shows the end of the list definition.

*Gambar 8 codingan data awal pengguna*

Program membuat akun default:

admin / 123 → untuk admin.

pom / 321 → untuk pengguna biasa.

A screenshot of a code editor showing a Python function named 'human\_currency'. The function takes 'amount' as an argument. It uses a try-except block to format the amount as a string with commas as thousands separators and a period as a decimal separator. If an exception occurs, it returns the amount as a string with 'Rp' prefix.

*Gambar 9 codingan fungsi utilitas(pendukung)*

human\_currency(amount)

Mengubah angka menjadi format uang Indonesia, misalnya:

50000 → Rp50,000

safe\_input(prompt)

Membaca input dari pengguna dengan penanganan error jika pengguna menekan Ctrl + C atau Ctrl + D.

print\_header(title)

Menampilkan judul dengan garis pembatas agar tampilan lebih rapi.

A screenshot of a code editor showing the first line of a Python script: 'import csv'.

*Gambar 10 codingan fungsi csv*

read\_csv\_dict(path)

Membaca file CSV dan mengembalikan daftar data (list of dictionary).

atomic\_write\_csv(path, fieldnames, rows)

Menulis ulang file CSV dengan aman:

Menulis ke file sementara .tmp terlebih dahulu.

Mengganti file asli setelah penulisan berhasil.

```
15
16 DEFAULT_USERS = [
17     {"username": "admin", "password": "123", "role": "admin", "saldo": 0},
18     {"username": "pom", "password": "321", "role": "pengguna", "saldo": 0},
19 ]
20
```

*Gambar 11 codingan manajemen pengguna*

`ensure_default_users()`

Jika file `users.csv` belum ada, sistem otomatis membuat akun default.

`load_users()` dan `save_users()`

`load_users()` → memuat semua data pengguna dari CSV ke dictionary Python.

`save_users()` → menyimpan kembali perubahan (misal: top-up saldo) ke CSV.

```
100 def load_pets():
101     if not os.path.exists(CSV_FILE):
102         atomic_write_csv(CSV_FILE, ["Jenis Hewan", "Umur"], [])
103         return []
104     rows = read_csv_dict(CSV_FILE)
105     pets = [{"r.get("Jenis Hewan", ""), r.get("Umur", "")} for r in rows]
106     return pets
107
108 def save_pets(pets):
109     rows = [{"Jenis Hewan": p[0], "Umur": p[1]} for p in pets]
110     atomic_write_csv(CSV_FILE, ["Jenis Hewan", "Umur"], rows)
111
```

*Gambar 12 codingan data hewan (petcare)*

`load_pets()` dan `save_pets()`

Menyimpan dan memuat data jenis hewan dan umur dari file `petcare.csv`.

Admin bisa menambah, mengubah, dan menghapus data hewan

```
131 def append_invoice(username, layanan, harga):
132     ada = os.path.exists(CSV_INVOICE)
133     waktu = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
134     try:
135         with open(CSV_INVOICE, mode="a", newline="", encoding="utf-8") as f:
136             writer = csv.writer(f)
137             if not ada:
138                 writer.writerow(["Tanggal", "Username", "Layanan", "Harga"])
139             writer.writerow([waktu, username, layanan, int(harga)])
140     except Exception:
141         return False
142
```

*Gambar 13 codingan reverensi dan invoice*

`append_reservation(username, layanan, harga)`

Menyimpan data reservasi layanan ke `reservasi.csv`.

`append_invoice(username, layanan, harga)`

Menyimpan data transaksi ke `invoice.csv` beserta tanggal dan waktu.

```
145 def format_pet_table(pets):
146     t = PrettyTable(["No", "Jenis Hewan", "Umur"])
147     for i, p in enumerate(pets, start=1):
148         t.add_row([i, p[0], p[1]])
149     return t
150
```

*Gambar 14 codingan format tabel (PrettyTable)*

`format_pet_table(pets)`

Menampilkan daftar hewan dalam bentuk tabel rapi.

`format_reservasi_table(rows)`

Menampilkan data reservasi beserta harga dengan format uang



```
161 #LOGIN
162 def login(users):
163     while True:
164         print_header("LOGIN SISTEM PETCARE")
165         username = safe_input("Masukkan username: ").strip()
166         password = pwinput("Masukkan password: ")
167
168         user = users.get(username)
169         if user and user["password"] == password:
170             print(f"\nLogin berhasil! Selamat datang, {username}.")
171             return username, user["role"]
172         else:
173             print("\nUsername atau password salah, coba lagi.")
174
```

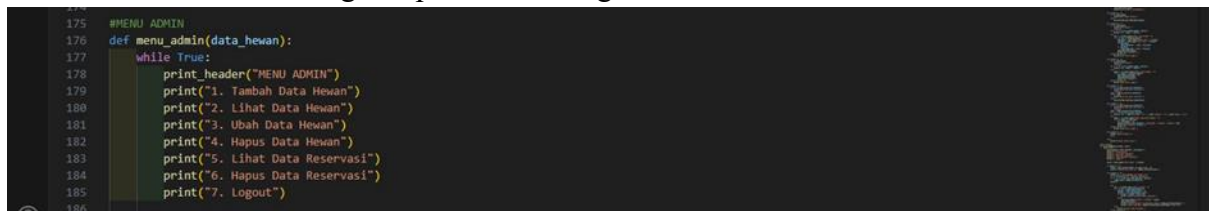
*Gambar 15 codingan login sistem*

`def login(users):`

Fungsi ini meminta username dan password, lalu:

Jika benar → login berhasil dan menampilkan role pengguna.

Jika salah → diminta ulang sampai berhasil login.



```
175 #MENU ADMIN
176 def menu_admin(data_hewan):
177     while True:
178         print_header("MENU ADMIN")
179         print("1. Tambah Data Hewan")
180         print("2. Lihat Data Hewan")
181         print("3. Ubah Data Hewan")
182         print("4. Hapus Data Hewan")
183         print("5. Lihat Data Reservasi")
184         print("6. Hapus Data Reservasi")
185         print("7. Logout")
186
```

*Gambar 16 codingan menu admin*

`def menu_admin(data_hewan):`

Admin memiliki 7 menu:

1. Tambah Data Hewan

→ menambahkan jenis & umur hewan.

2. Lihat Data Hewan

→ menampilkan semua data dalam tabel.

3. Ubah Data Hewan

→ mengedit jenis atau umur hewan.

4. Hapus Data Hewan

→ menghapus data berdasarkan nomor urutan.

5. Lihat Data Reservasi

→ melihat semua pemesanan pengguna.

## 6. Hapus Data Reservasi

→ menghapus pemesanan tertentu.

## 7. Logout

→ keluar ke menu login.

```
271 #MENU PENGGUNA
272 def menu_pengguna(username, users):
273     while True:
274         print_header(f"MENU PENGGUNA ({username})")
275         print("1. Lihat Saldo")
276         print("2. Isi Saldo (Top Up)")
277         print("3. Reservasi Layanan")
278         print("4. Lihat Riwayat Transaksi")
279         print("5. Logout")
280
```

*Gambar 17 codingan menu pengguna*

def menu\_pengguna(username, users):

Pengguna memiliki 5 menu utama:

### 1. Lihat Saldo

Menampilkan saldo e-money saat ini.

### 2. Isi Saldo (Top-Up)

Pilih nominal dari daftar dan tambah ke saldo (tidak boleh melebihi batas maksimal).

### 3. Reservasi Layanan

Pilih layanan seperti Grooming, Vaksin, Penitipan, lalu sistem otomatis:

Mengurangi saldo.

Menyimpan data ke reservasi.csv dan invoice.csv.

### 4. Lihat Riwayat Transaksi

Menampilkan semua invoice pengguna dalam tabel.

### 5. Logout

Keluar ke menu login.

```
360 #MAIN
361 def main():
362     users = load_users()
363     pets = load_pets()
364     username, role = login(users)
365     if role == "admin":
366         menu_admin(pets)
367     else:
368         menu_pengguna(username, users)
369
```

*Gambar 18 codingan fungsi utama (main program)*

Langkah utama saat program dijalankan:

1. Memuat data pengguna dan hewan dari file CSV.

2. Melakukan proses login.

3. Menampilkan menu sesuai role:

Admin → menu\_admin()

Pengguna → menu\_pengguna()



*Gambar 19 codingan jalankan program*

Langkah utama saat program dijalankan:

1. Memuat data pengguna dan hewan dari file CSV.

2. Melakukan proses login.

3. Menampilkan menu sesuai role:

Admin → menu\_admin()

Pengguna → menu\_pengguna()

#### Kesimpulan Penjelasan

Program ini menunjukkan penerapan konsep dasar pemrograman Python:

- Struktur data (list, dict)
- Fungsi dan modularisasi
- Pengelolaan file CSV
- Keamanan input dan login
- Penerapan sistem CRUD
- Tampilan data dengan tabel interaktif
- Program ini sangat cocok sebagai proyek akhir praktikum dasar pemrograman karena sudah menerapkan:
  - Authentication system
  - Role-based access control
  - Persistent storage (CSV)
  - Transaction & balance management
  - Data integrity handling (atomic write)

## **BAB III**

### **KESIMPULAN**

Berdasarkan hasil perancangan dan implementasi sistem reservasi dan pelayanan PetCare berbasis Python, dapat disimpulkan bahwa sistem ini berhasil membantu proses administrasi dan pelayanan pada tempat PetCare menjadi lebih efisien, cepat, dan terorganisir. Sistem yang dikembangkan mampu mengotomatisasi berbagai kegiatan yang sebelumnya dilakukan secara manual, seperti pencatatan data pelanggan, pengelolaan layanan, transaksi, serta pembuatan invoice

Sistem ini menerapkan konsep dasar CRUD (Create, Read, Update, Delete) dengan dua peran pengguna, yaitu Admin dan User, yang masing-masing memiliki hak akses berbeda. Admin dapat mengelola seluruh data seperti layanan, pelanggan, serta transaksi, sementara User dapat melakukan reservasi layanan dan transaksi menggunakan saldo e-money internal sistem. Proses transaksi dilengkapi dengan pembuatan invoice otomatis sebagai bukti pembayaran, serta penyimpanan data secara dinamis menggunakan format CSV, sehingga data tetap tersimpan meskipun program ditutup.

Bahasa pemrograman Python dipilih karena kesederhanaannya serta dukungan library seperti PrettyTable untuk tampilan tabel dan getpass untuk keamanan login. Dengan pemanfaatan library tersebut, sistem mampu memberikan pengalaman penggunaan yang mudah dan informatif, meskipun masih berbasis Command Line Interface (CLI).

Dari hasil pengujian, sistem mampu menjalankan fungsinya dengan baik sesuai dengan batasan yang telah ditentukan. Sistem terbukti dapat membantu pengelola PetCare dalam mengurangi risiko kesalahan pencatatan (human error), mempercepat proses pelayanan, serta meningkatkan ketertiban administrasi data pelanggan dan transaksi.

Secara keseluruhan, sistem ini memberikan kontribusi nyata terhadap digitalisasi layanan PetCare, serta dapat dijadikan pondasi awal untuk pengembangan sistem yang lebih lanjut, seperti versi berbasis web atau aplikasi mobile dengan fitur notifikasi otomatis, integrasi database, serta transaksi online yang sesungguhnya.

Dengan demikian, dapat disimpulkan bahwa sistem reservasi dan pelayanan PetCare berbasis Python ini telah memenuhi tujuan penelitian, yaitu merancang solusi teknologi yang sederhana, efisien, dan bermanfaat dalam mendukung operasional bisnis jasa perawatan hewan peliharaan di era digital.

## LAMPIRAN

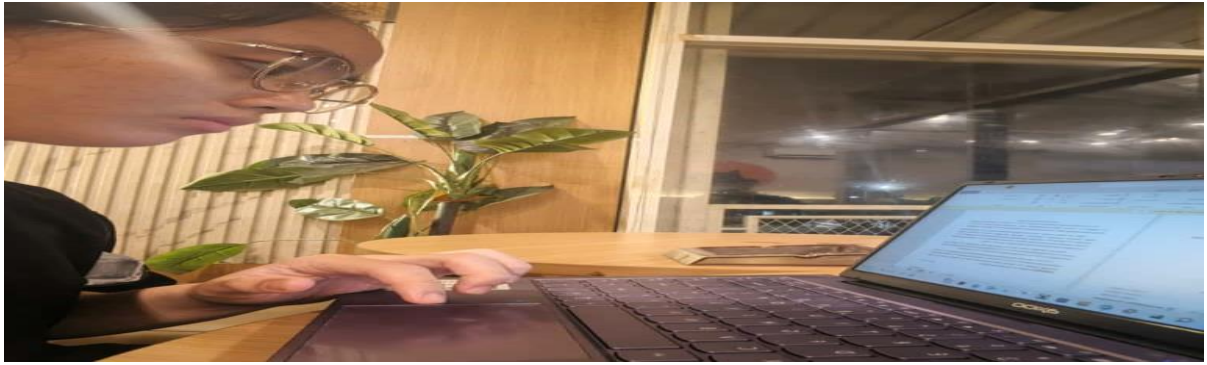
**Table 1** : Table kontribusi

Nama	Kontribusi	Bagian
Andhika Fhariel Fadhlurrohman(2509116083)	Flowchart,github,codingan,konsep	Membuat flowchart dari awal hingga akhir Membantu membuat kode Membuat readme
Rasya Aditia Ramadani (2509116082)	Flowchart,codingan,github,konsep	Membuat kode dari awal hingga selesai Membantu mengisi flowchart Membuat readme
Meilanie (2509116109)	Flowchart,laporan,github,konsep	Menulis laporan dari awal hingga selesai Membantu mengisi flowchart Mengisi readme

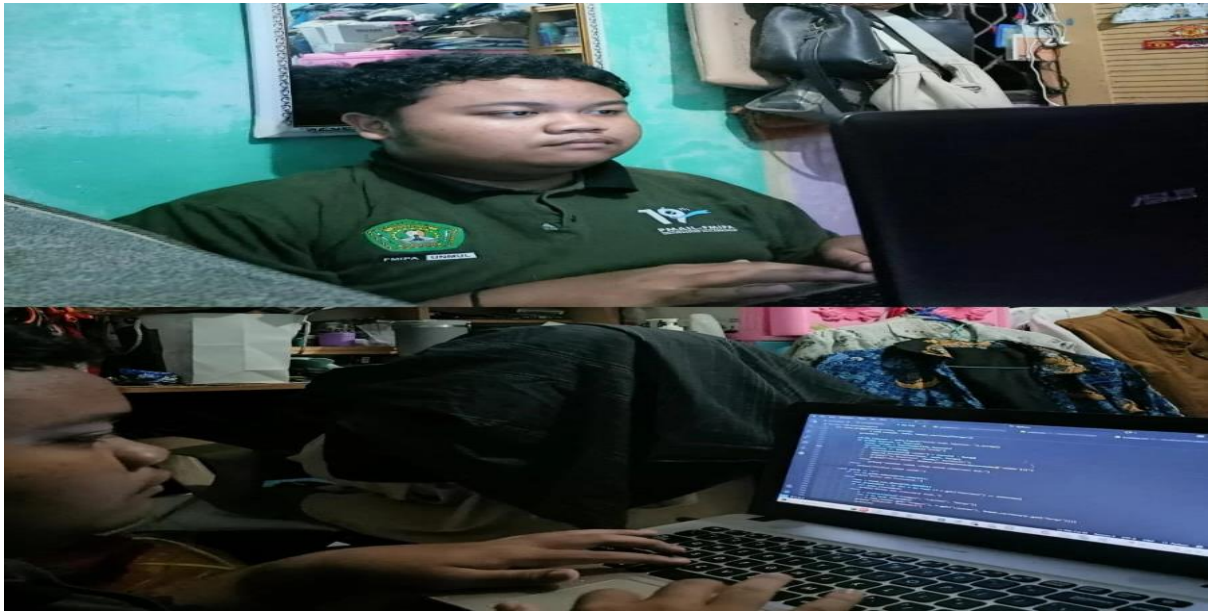


*lampiran 1 dokumentasi proses pengerjaan PA*





*lampiran 2 dokumentasi proses pengerjaan PA*



*lampiran 3 dokumentasi proses pengerjaan PA*



*lampiran 4 dokumentasi konsul 1*



*lampiran 5 dokumentasi konsul 2*



*lampiran 6 dokumentasi proses pengerjaan PA*