

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



MANAJEMENT PENDATAAN RUMAH

Oleh:

Kelompok 5

M. Tedy Azhari	2409106003
Angelina Augustin Umami	2409106014
Nadia Rahmah	2409106018

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

KATA PENGANTAR

Segala puji dan syukur kami panjatkan ke hadirat Tuhan yang Maha Esa, yang telah memberikan rahmat, karunia, dan hidayah-Nya sehingga kami dapat menyelesaikan laporan Proyek Akhir Praktikum Algoritma Pemrograman dasar dengan baik dan tepat pada waktunya. Penyusunan laporan ini bertujuan untuk mendokumentasikan proses mengaplikasikan pemahaman yang sudah diberikan kepada kami selama diadakannya Praktikum Algoritma Pemrograman Dasar.

Program Manajemen Pendataan Rumah yang kami buat merupakan program dengan fitur CRUD(Create, Read, Update, dan Delete) untuk pendataan informasi seperti nomor rumah, jenis rumah, pemilik, tahun berdiri, alamat, dan status rumah. Terdapat juga data akun registrasi yang berisi data username dan peran user serta satu akun admin. Fitur dioperasikan sesuai peran, yaitu sebagai admin atau user. Program ini juga dilengkapi dengan sistem login dan registrasi, serta penyimpanan data menggunakan file JSON.

Kendala yang kami dapatkan selama penyusunan dan pembuatan program adalah saat menggunakan *library* standar *python* yaitu *os* dan file eksternal JSON seperti penghapusan terminal yang tidak keseluruhan. Kami mengucapkan terima kasih kepada pihak-pihak yang telah membantu mengatasi kendala yang kami dapatkan. Kami juga mengucapkan terima kasih kepada Dosen, Asisten Lab yang telah memberikan bimbingan dan arahan pembuatan program dan laporan kami.

Demikian Laporan akhir ini kami buat. Kami menyadari bahwa laporan yang kami buat masih jauh dari sempurna, namun kami berharap agar dapat terus mengembangkan potensi kemampuan yang kami miliki setelah membuat laporan ini. Kami juga berharap laporan ini dapat bermanfaat bagi semua pihak yang memerlukan

Samarinda, 10 November 2024

Kelompok 5

TAKARIR

Daftar padanan kata bahasa asing dalam bahasa Indonesia yang digunakan adalah sebagai berikut:

<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
<i>Output</i>	Keluaran
<i>Clear</i>	Bersihkan
<i>Login</i>	Masuk
<i>Logout</i>	Keluar
<i>Register</i>	Daftar
<i>User</i>	Pengguna
<i>Role</i>	Peran
<i>Validasi</i>	Pengecekan
<i>PrettyTable</i>	Tabel Cantik
<i>Password</i>	Kata Sandi
<i>Username</i>	Nama Pengguna
<i>Role</i>	Peran
<i>Error</i>	Kesalahan
<i>Break</i>	Hentikan
<i>Print</i>	Cetak

DAFTAR ISI

KATA PENGANTAR	2
TAKARIR.....	4
DAFTAR ISI.....	5
DAFTAR GAMBAR	7
BAB I PENDAHULUAN.....	9
1.1 Latar Belakang.....	9
1.2 Kebutuhan Fungsional	9
1.3 Rumusan Masalah.....	12
1.4 Batasan Masalah	12
1.5 Tujuan	13
BAB II PERANCANGAN	14
2.1 Analisis Program	14
2.2 Flowchart	16
2.3 Konsep/Materi Praktikum yang dipakai	30
BAB III	32
3.1 Tampilan Program	32
1. Tampilan Menu	32
a. Source Code	40
.....	41
BAB IV PENUTUP	50
4.1 Kesimpulan	50

4.2	Saran	51
	DAFTAR PUSTAKA	52
	LAMPIRAN.....	53

DAFTAR GAMBAR

Gambar 2. 1 Flowchart Menu Utama	<u>17</u>
<u>Gambar 2. 3 Flowchart Menu Login</u>	<u>18</u>
<u>Gambar 2. 4 Flowchart Menu Register.....</u>	<u>19</u>
Gambar 2. 5 Flowchart Menu Admin.....	20
Gambar 2. 6 Flowchart Menu User	<u>22</u>
<u>Gambar 2. 7 Tambah Data Rumah</u>	<u>23</u>
<u>Gambar 2. 8 Tampilkan Data Rumah</u>	<u>25</u>
<u>Gambar 2. 9 Menu Hapus Data Ruma.....</u>	<u>26</u>
Gambar 2. 10 Menu Tampilkan Data Registras	<u>28</u>
Gambar 3. 1 Tampilkan Menu	29
Gambar 3. 2 Tampilan Menu Admin.....	32
Gambar 3. 3 Tampilan Tambah Data Rumah	Kesalahan! Bookmark tidak ditentukan. <u>3</u>
<u>Gambar 3. 4 Tampilkan Data Rumah</u>	<u>34</u>
Gambar 3. 5 Tampilan Ubah Data Rumah	34
Gambar 3. 6 Tampilan hapus Data Rumah.....	35
Gambar 3. 7 Tampilan Data Registrasi.....	35
<u>Gambar 3. 8 Tampilan Menu User</u>	<u>36</u>
<u>Gambar 3. 9 Tampilan Tambahkan Data Rumah</u>	<u>38</u>
<u>Gambar 3. 10 Tampilan Menu Keluar.....</u>	<u>39</u>
<u>Gambar 4.1 Tampilan Kartu Konsul.....</u>	<u>52</u>

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan berkembangnya zaman, kemajuan teknologi menjadi sebuah solusi yang sangat berdampak pada beberapa aspek kehidupan. Seperti misalnya pada aspek pendataan dan pengelolaan properti. pendataan properti pada sebelumnya cenderung menggunakan proses manual. Proses tersebut cenderung kurang efisien dan rentan terhadap kesalahan. Proses pengelolaan pun dapat menjadi lebih rumit dan memakan waktu yang lama.

Oleh karena itu, Program Manajemen Pendataaan Rumah yang kami buat dirancang untuk memberikan solusi efektif agar memudahkan admin dalam meningkatkan efisiensi pengelolaan data property, seperti kemudahan pengelolaan data, pengelolaan data menjadi lebih cepat, mempermudah pemantauan aset properti, meningkatkan keamanan data dan memudahkan admin atau pemilik asset menentukan tindakan seperti menjual, menyewakan atau merawat properti.

1.2 Kebutuhan Fungsional

1. Menu Pengguna

- Menu Login: pengguna dapat masuk ke program dengan username dan password yang telah terdaftar.
- Menu Registrasi: Pengguna baru dapat masuk ke program dengan memasukkan username dan password. Pengguna bisa memiliki peran sebagai user.

2. Manajemen Akun

- Simpan Data Akun: Data akun disimpan dalam file data_akun.json
- Muat Data Akun: Data akun dapat dimuat dari file data_akun.json
- Tampilkan Data Registrasi (admin): Hanya admin yang dapat melihat semua daftar pengguna yang sudah terdaftar

3. Manajemen Data Rumah

➤ **Tambah Data Rumah:**

Admin dan user dapat menambah data rumah baru:

- Nomor rumah
- Jenis rumah (pilihan: Beton, kayu)
- Nama pemilik
- Tahun berdiri (antara 1980 – 2024)
- Alamat rumah
- Status rumah (pilihan: kosong, dijual, ditempati)
- Data yang telah dimasukkan disimpan kedalam file `data_rumah.json`

➤ **Tampilkan Data Rumah:**

Admin dan user dapat melihat rumah yang terdaftar dalam bentuk tabel dengan kolom:

- No, No rumah, Jenis, Pemilik, Tahun, Alamat, Status.

➤ **Ubah Data Rumah (admin):**

Admin dapat memperbarui data rumah yang sudah terdaftar berdasarkan nomor rumah.

- Admin dapat memperbarui:
- Nomor rumah, Jenis rumah, Nama pemilik, Tahun berdiri, Alamat, dan Status rumah.

➤ **Hapus Data Rumah (Admin):**

- Admin dapat menghapus data rumah berdasarkan nomor rumah yang dipilih.

4. Menu Navigasi

➤ **Menu Utama:**

- Pengguna dapat memilah login, registrasi, atau keluar dari program.

➤ **Menu Admin:**

- Opsi Tersedia:
 1. Tambah Data Rumah
 2. Tampilkan Data rumah
 3. Ubah Data Rumah
 4. Hapus Data Rumah
 5. Tampilkan Data Registrasi
 6. Keluar

➤ **Menu User:**

- Opsi Tersedia:
 1. Tampilkan Data Rumah
 2. Tambah Data Rumah
 3. Tampilkan Data Registrasi
 4. Keluar

5. Fungsi Pendukung Lainnya

➤ **Validasi input:**

- Program ini memiliki validasi input untuk memastikan data yang dimasukkan oleh pengguna sesuai tipe yang diharapkan, seperti angka atau input integer.

➤ **Pembersihan layar:**

- Program akan membersihkan layar secara otomatis ketika berpindah menu yang membuat tampilan lebih rapi.

➤ **Penyimpanan Data:**

- Data rumah dan data akun disimpan secara lokal dengan menggunakan file JSON (data_rumah.json dan data_akun.json)

➤ **Tabel Data:**

- Data rumah dan registrasi pengguna ditampilkan dalam format tabel agar mempermudah dibaca dengan bantuan library PrettyTable.

1.3 Rumusan Masalah

Berikut rumusan masalah yang terdapat pada laporan ini, rumusan masalah dibuat dengan tujuan untuk membantu kami dalam merumuskan permasalahan yang akan diselesaikan.

1. Bagaimana menciptakan sistem pendataan rumah yang terstruktur sehingga mempermudah pencatatan dan pengelolaan informasi?
2. Bagaimana memastikan keamanan data, terutama untuk akun pengguna dan data rumah yang sensitif?
3. Bagaimana sistem ini dapat menyediakan fitur yang relevan, seperti tambah, ubah, hapus, dan melihat data rumah?
4. Bagaimana membedakan akses antara pengguna biasa (user) dan admin untuk menjaga integritas data?

1.4 Batasan Masalah

Berikut batasan masalah yang ada pada laporan ini, batasan masalah ini dibuat agar penyelesaian masalah tidak terlalu meluas.

1. Sistem ini dirancang hanya untuk pengelolaan data rumah dan akun pengguna
2. Sistem berbasis CLI (Command Line Interface) tanpa integrasi GUI atau web
3. Data disimpan secara lokal dalam format file JSON, tanpa dukungan database relasional atau berbasis cloud
4. Hanya dua jenis akun yang didukung yaitu admin dan user
5. Validasi data terbatas pada input pengguna, tanpa verifikasi lebih lanjut untuk data seperti alamat atau nama.

1.5 Tujuan

Adapun tujuan yang akan dicapai setelah ditemukannya masalah.

1. Efisiensi: Meningkatkan efisiensi dalam pengelolaan data rumah dengan menyediakan fitur tambah, ubah, hapus, dan melihat data secara langsung.
2. Keamanan: Menyediakan sistem login dengan akses berbasis peran (role-based access control) untuk admin dan user.
3. Struktur Data: Mengelola data rumah dalam format terstruktur sehingga mudah diakses dan diolah.
4. Kemudahan Penggunaan: Membuat antarmuka berbasis teks yang sederhana dan mudah digunakan oleh berbagai kalangan pengguna.
5. Modularitas: Memberikan desain program modular yang dapat dikembangkan lebih lanjut di masa depan.

BAB II

PERANCANGAN

2.1 Analisis Program

Program “Manajemen Pendataan Rumah” ini dirancang untuk mempermudah pendataan rumah, dengan dua peran utama yaitu Admin dan User. Pada program ini memungkinkan pengguna untuk bisa login, registrasi, menambah, mengubah, menghapus, dan melihat data rumah.

Alur Program

1. Persiapan Awal

- Pada program akan dilakukan pemuatan data dari file JSON (data_rumah.json dan data_akun.json) saay pertama kali dijalankan. Jika file tidak ditemukan maka program akan memuat data kosong.

2. Tampilan Menu Utama

- Program akan menampilkan menu utama dan pengguna diminta untuk memilih opsi dengan memasukkan angka 1, 2, atau 3:
- 1 (Login): Masuk
- 2 (Register): Membuat akun baru
- 3 (Keluar): Menutup program

3. Proses Login

- Jika pengguna memilih login maka sistem akan meminta memasukkan username dan password
- Kemudian sistem akan mengecek data akun:
- Jika valid, maka pengguna akan diarahkan ke menu sesuai perannya (admin atau user)
- Jika tidak valid, maka akan muncul pesan error.

4. Proses Registrasi

- Jika pengguna memilih register maka sistem akan meminta username

dan password baru.

- Jika username belum terdaftar, maka akun akan disimpan dengan role 'user'
- Kemudian akun baru akan disimpan di file data_akun.json.

5. Menu Admin

- Setelah login sebagai admin maka akan muncul menu admin
- Fungsi masing-masing menu:
- Tambah data rumah: menambahkan data rumah yang baru
- Tampilkan data rumah: Menampilkan semua data rumah dalam bentuk tabel
- Ubah data rumah: mengubah data rumah berdasarkan nomor rumah
- Hapus data rumah: Menghapus data rumah berdasarkan nomor rumah
- Tampilkan data registrasi: Menampilkan seluruh akun yang terdaftar
- Keluar: Kembali ke menu utama.

6. Menu User

- Setelah login sebagai User maka akan muncul menu User.
- Fungsi masing-masing menu:
- Tampilkan data rumah: Menampilkan data rumah
- Tambahkan data rumah: Menambahkan data rumah baru
- Tampilkan data registrasi: Menampilkan semua akun yang sudah terdaftar
- Keluar: Kembali ke menu utama.

7. Operasi Data Rumah

- Pada fitur, Tambah, Ubah dan Hapus Data Rumah, Data Rumah akan disimpan dalam file JSON (data_rumah.json)
- Data rumah yang dimasukkan:
- Nomor rumah
- Jenis rumah (kayu/beton)
- Nama pemilik
- Tahun berdiri (antara 1980 – 2024)
- Alamat rumah

- Status rumah (kosong, dijual, ditempati)

8. Keluar Program

- Pengguna dapat memilih opsi “Keluar” dari menu utama atau bisa juga dari menu admin/user untuk menutup program.

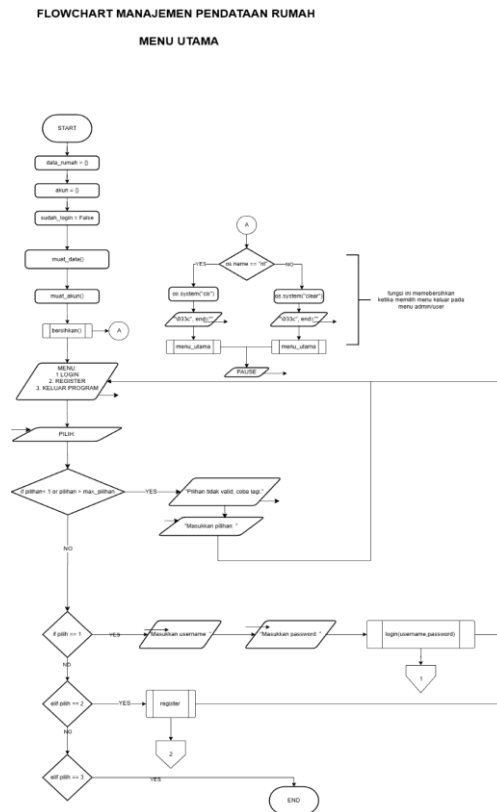
2.2 Flowchart

Flowchart atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem (Rosaly, 2019). Flowchart digunakan sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada program. Dengan adanya flowchart, setiap urutan proses dapat digambarkan menjadi lebih jelas.

1. Flowchart menu utama

- 1) Start.
- 2) Muat data akun dan data rumah
- 3) Masuk ke menu utama lalu input pilihan
 1. Login
 2. Register
 3. Keluar Program
- 4) Jika pilihan == 1, maka program meminta input username dan password kepada pengguna, lalu akan di arahkan menuju ke menu login,
- 5) jika pilihan == 2, maka akan diarahkan menuju ke menu register
- 6) jika pilihan == 3, maka pengguna akan keluar dari program.

7) End.

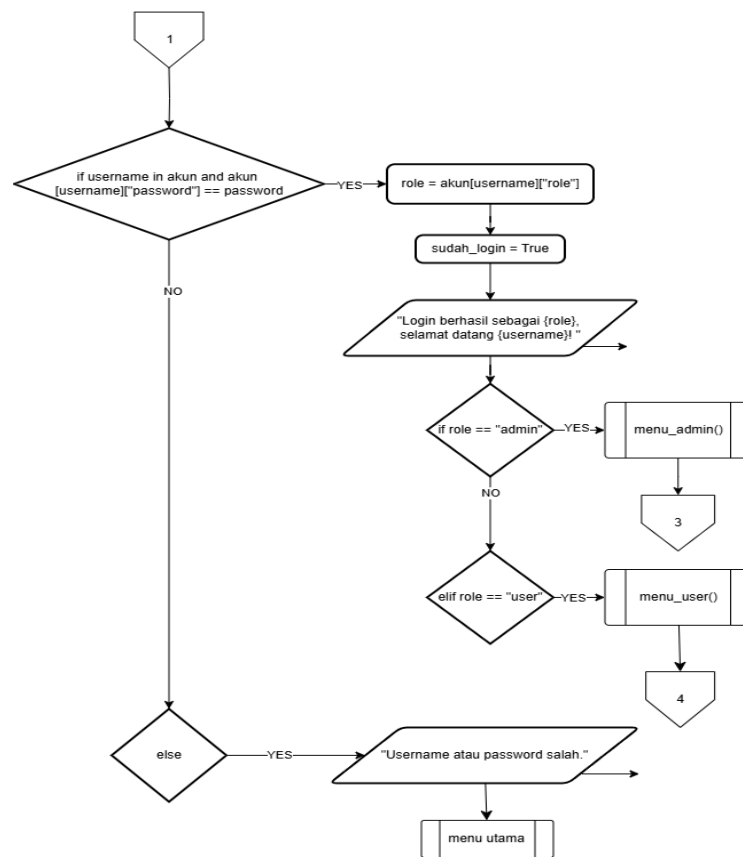


Gambar 2. 1 Flowchart Menu Utama

2. Menu Login

- 1.) Memeriksa apakah Username dan password valid
- 2.) Jika YES maka program memeriksa role pengguna.
- 3.) Jika username dan password adalah tedy dan 123, maka role adalah sebagai admin.
Jika selain username dan password tersebut maka role adalah user
- 4.) Jika NO maka program akan menampilkan output “Username atau password salah.”
Dan diarahkan menuju menu utama.

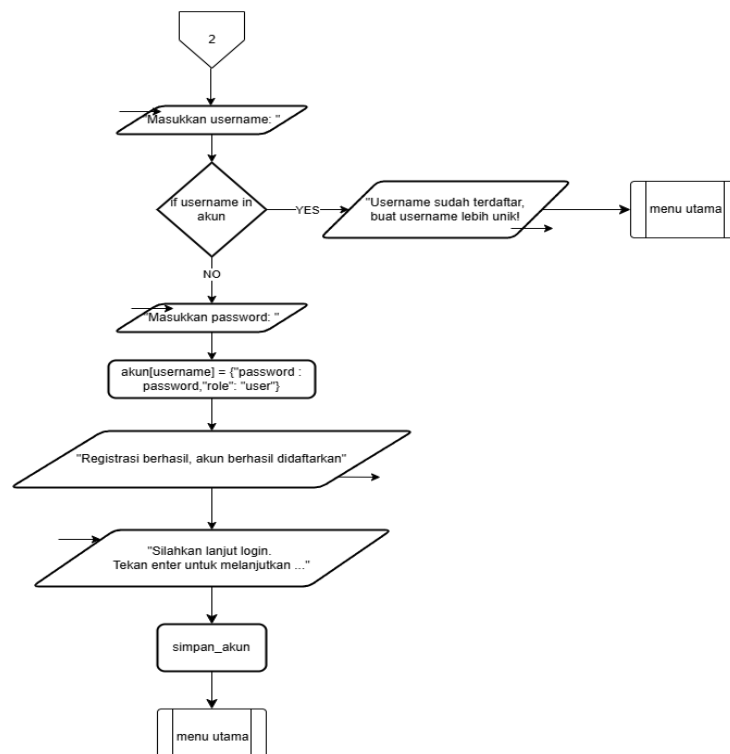
MENU LOGIN



Gambar 2. 2 Flowchart Menu Login

3. Menu Register

MENU REGISTER

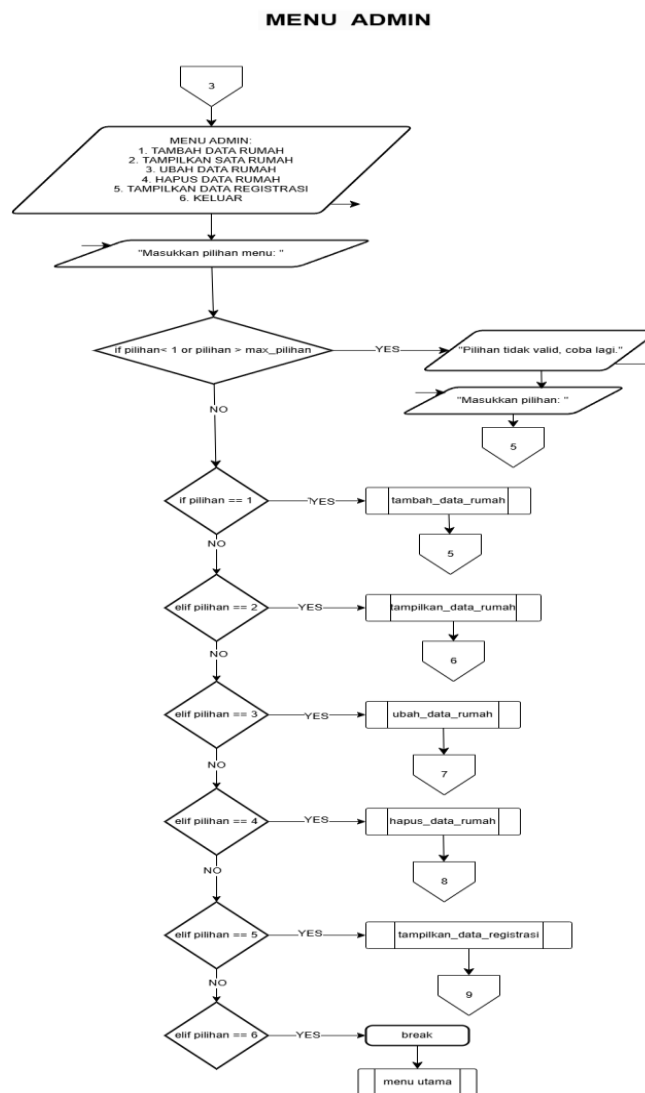


Gambar 2. 3 Flowchart Menu Register

Flowchart menu register

- 1) Masukkan username dan password
- 2) Memeriksa apakah username ada dalam data akun
- 3) Jika ada(YES), maka program akan menampilkan output “ Username sudah terdaftar, buat username lebih unik!”, lalu di arahkan ke menu utama
- 4) Jika tidak ada(NO),maka program akan meminta masukkan password
- 5) Role pengguna akan otomatis menjadi user
- 6) Program menampilkan output “Registrasi berhasil, silahkan lanjut login.”

4. Menu Admin



Gambar 2. 4 Flowchart Menu Admin

Flowchart menu admin

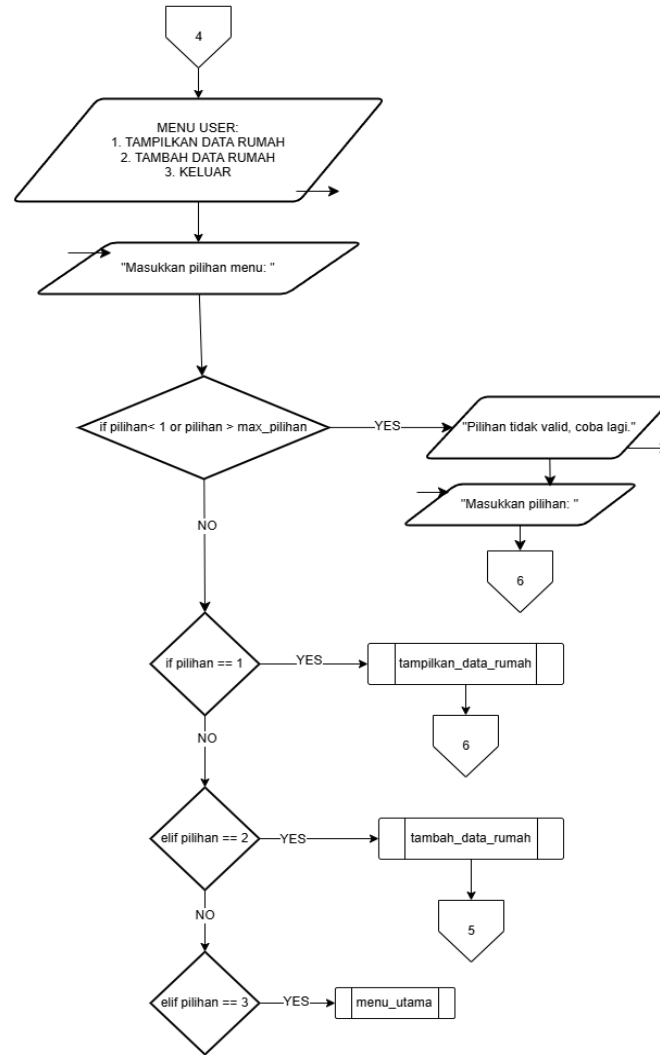
- 1) Program meminta input pilihan menu
- 2) Jika input pilihan salah, program akan menampilkan output “ Pilihan tidak valid, coba lagi.”.Program akan terus meminta inputan pilihan sampai inputan benar
- 3) Jika inputan sudah benar maka akan diarahkan ke menu yang dipilih

- 4) Terdapat 6 menu admin, yaitu
1. Tambah data rumah
 2. Tampilkan data rumah
 3. Ubah data rumah
 4. Hapus data rumah
 5. Tampilakan data registrasi
 6. Keluar

MENU USER

5. Menu

User



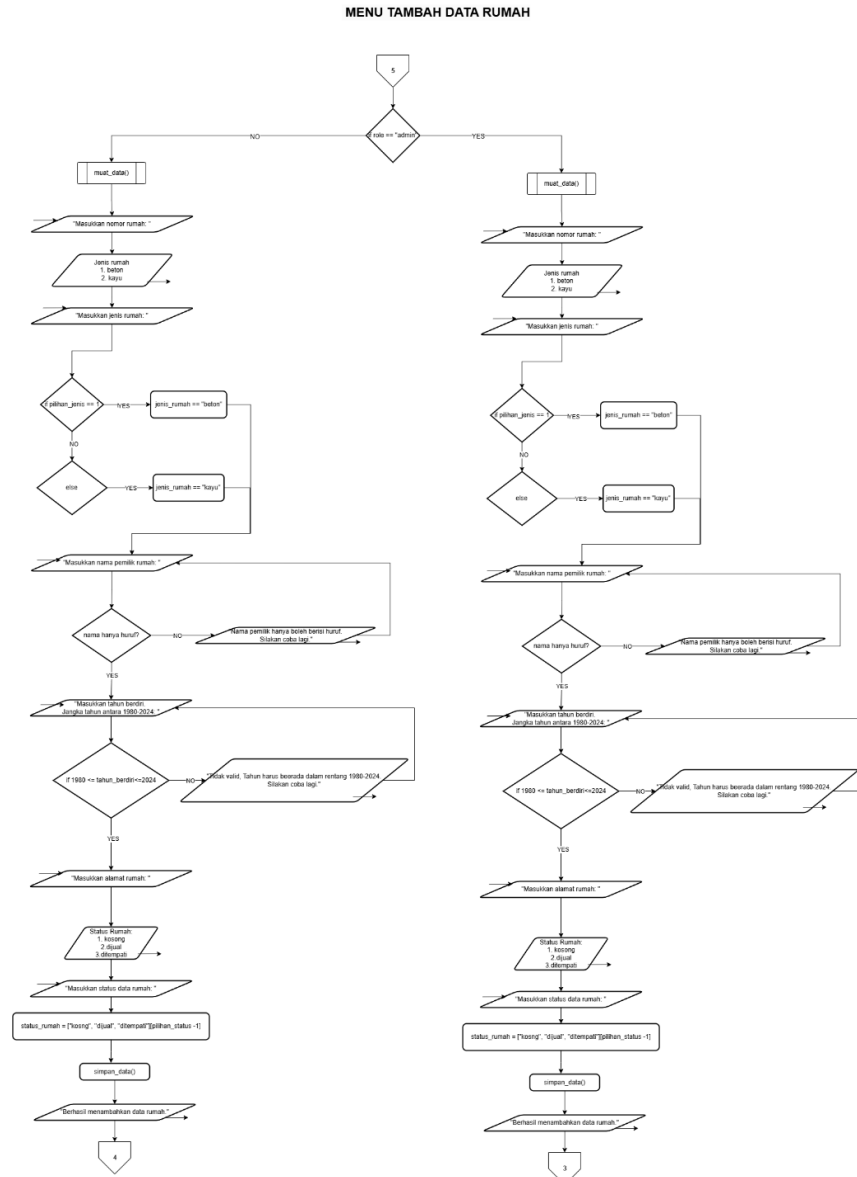
Gambar 2. 5 Flowchart Menu User

Flowchart menu user

- 1) Program meminta input pilihan menu
- 2) Jika input pilihan salah, program akan menampilkan output “ Pilihan tidak valid, coba lagi.” program akan terus meminta inputan pilihan sampai inputan benar
- 3) Jika inputan sudah benar maka akan diarahkan ke menu yang dipilih
- 4) Terdapat 3 menu pada menu user yaitu:
 1. Tampilkan data rumah

2. Tambah data rumah
3. Keluar

6. Tambah Data Rumah



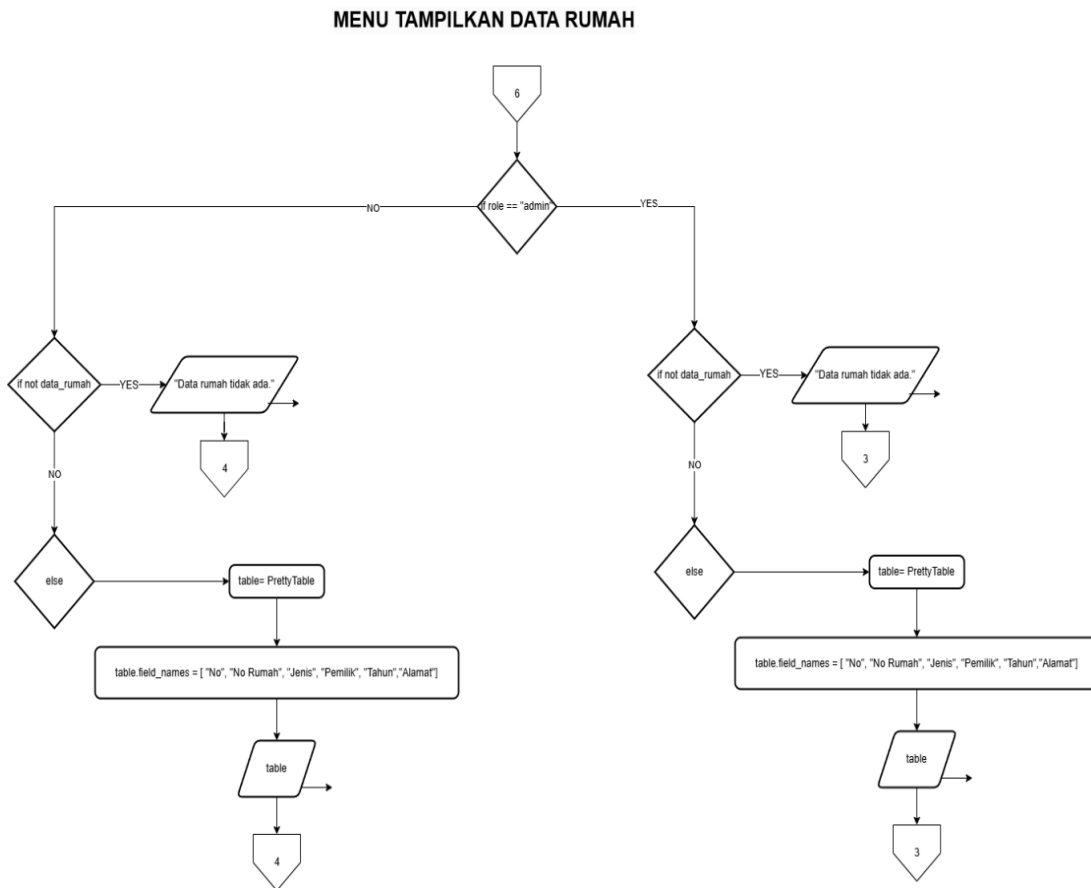
Gambar 2. 6 Tambah Data Rumah

Flowchart menu tambah data rumah

Pada menu ini admin dan user bisa menggunakannya.

- 1) Muat data rumah
- 2) Masukkan nomer rumah
- 3) Masukkan jenis rumah
Jika jenis rumah 1, maka jenis rumah adalah beton
Jika jenis rumah 2, maka jenis rumah adalah kayu
- 4) Masukkan nama pemilik rumah
- 5) Memeriksa nama pemilik rumah harus berupa hurup
- 6) Jika tidak berupa huruf maka program akan menampilkan pesan output “
Nama pemilik hanya boleh berisi huruf, silakan coba lagi.”
- 7) Jika benar berupa huruf maka program meminta input masukkan tahun
berdiri
- 8) Memeriksa jangka tahun apakah diantara tahun 1980 sampai 2024
- 9) Jika tidak maka program akan menampilkan output “Tidak valid, Tahun
harus berada dalam rentang 1980-2024. Silakan coba lagi
- 10) Masukkan alamat rumah
- 11) Masukkan status rumah
 1. Kosong
 2. Dijual
 3. Ditempati
- 12) Simpan data rumah
- 13) Data rumah berhasil di tambahkan
- 14) Kembali ke menu sesuai role (admin/user)

7. Menu Tampilkan Data Rumah

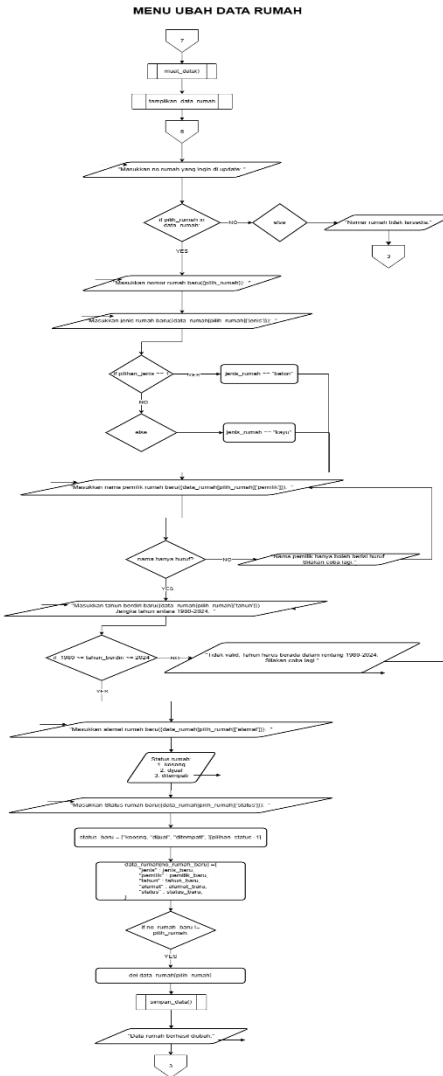


Gambar 2. 7 Tampilkan Data Rumah

Flowchart menu tampilkan data rumah

- 1) Memeriksa apakah data rumah tidak ada
- 2) Jika data rumah tidak ada maka program akan menampilkan pesan “ data rumah tidak ada, lalu akan diarahkan ke menu sesuai role
- 3) Jika data rumah ada, maka data rumah akan dibuat dengan menggunakan PrettyTable
- 4) Tampilkan table
- 5) Kembali ke menu sesuai role(admin/user)

8. Menu Ubah Data Rumah



Gambar 2. 7 Menu Ubah Data Rumah

Flowchart menu ubah data rumah

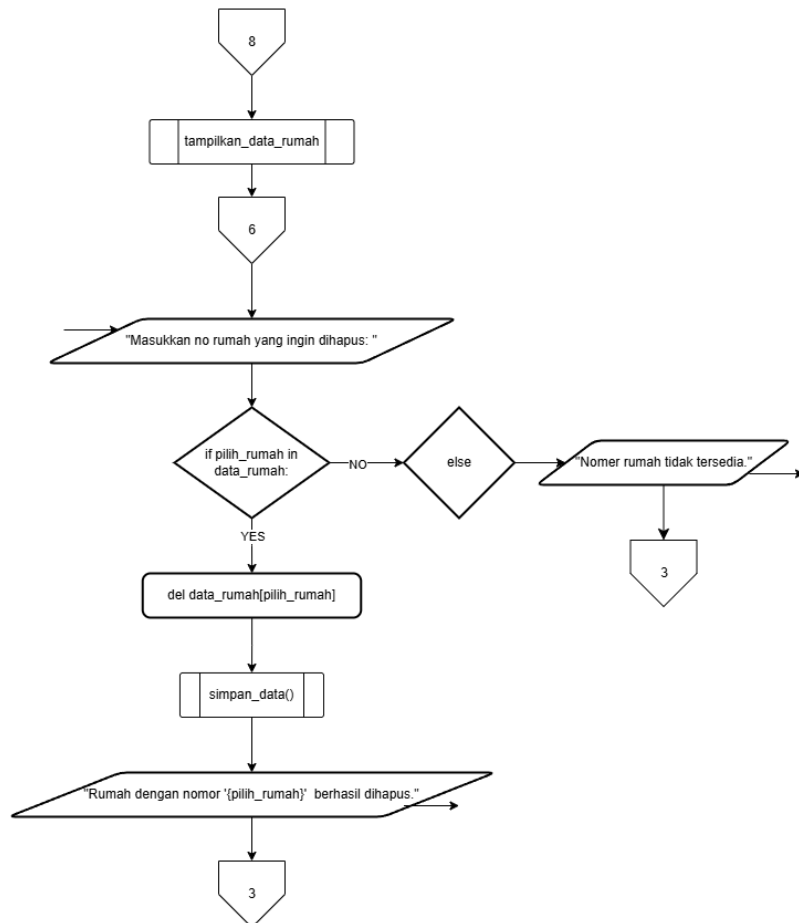
- 1) Muat data rumah
- 2) Tampilkan data rumah
- 3) Masukkan nomer rumah yang ingin di update

- 4) Memeriksa apakah nomer rumah ada di dalam data rumah
- 5) Jika tidak ada maka program akan menampilkan output “Nomer rumah tidak tersedia.” Lalu kembali ke menu admin.
- 6) Jika nomer rumah ada di dalam data rumah, maka program akan dilanjutkan dengan input jenis rumah
- 7) Masukkan jenis rumah
- 8) Memeriksa inputan jenis rumah
- 9) Jika jenis rumah 1, maka jenis rumah adalah beton
- 10) Jika jenis rumah 2, maka jenis rumah adalah kayu
- 11) Masukkan nama pemilik rumah
- 12) Memeriksa nama pemilik rumah harus berupa huruf
- 13) Jika tidak berupa huruf maka program akan menampilkan pesan output
- 14) Nama pemilik hanya boleh berisi huruf, silakan coba lagi.”
- 15) Jika benar berupa huruf maka program meminta input masukkan tahun berdiri
- 16) Memeriksa jangka tahun apakah diantara tahun 1980 sampai 2024
- 17) Jika tidak maka program akan menampilkan output “Tidak valid, Tahun harus berada dalam rentang 1980-2024. Silakan coba lagi
- 18) Masukkan alamat rumah
- 19) Masukkan status rumah
 1. Kosong
 2. Dijual
 3. Ditempati
- 15) Simpan datanrumah
- 16) Data rumah berhasil di tambahkan
- 17) Kembali ke menu sesuai role (admin/user)

9. Menu Hapus data Rumah

Flowchart menu hapus data rumah

MENU HAPUS DATA RUMAH

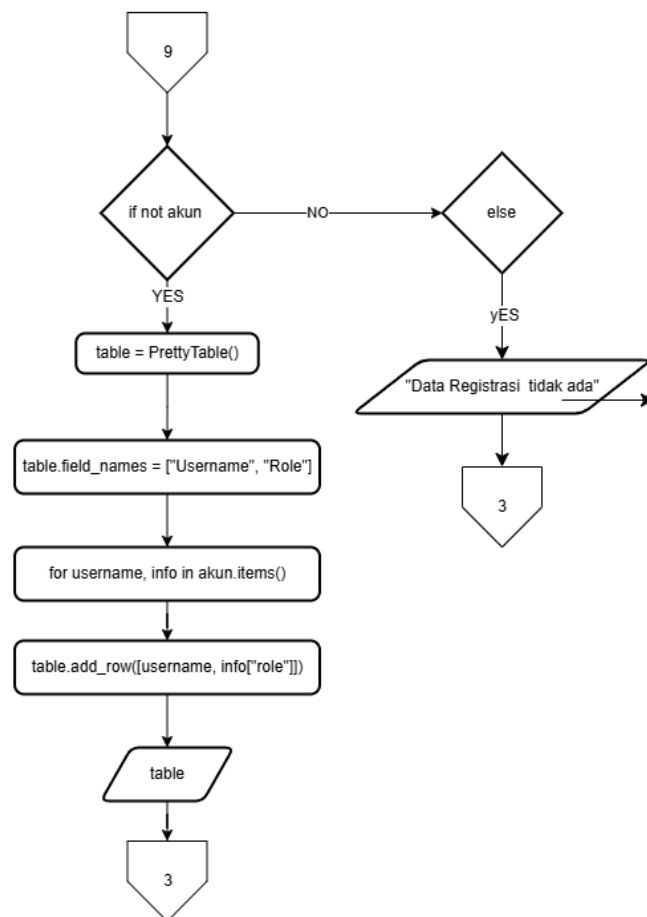


Gambar 2. 8 Menu Hapus Data Rumah

- 1) Tampilkan data rumah
- 2) Masukkan nomer rumah yang ingin dihapus
- 3) Periksa apakah nomer rumah ada dalam data rumah
- 4) Jika tidak ada maka program akan menampilkan output “Nomer rumah tidak tersedia.” Lalu kembali ke menu admin.
- 5) Jika nomer rumah ada di dalam data rumah, maka program akan dilanjutkan dengan input jenis rumah

10. Menu Tampilkan Data Registrasi

MENU TAMPILKAN DATA REGISTRASI



Gambar 2. 9 Menu Tampilkan Data Registrasi

Flowchart menu tampilkan data registrasi

1) Memeriksa apakah akun tidak ada

Jika benar tidak ada maka akan menampilkan pesan “data registrasi tidak ada”

Jika salah tidak ada maka table akan dibuat menggunakan PretttyTable

2) Kembali ke menu admin

2.3 Konsep/Materi Praktikum yang dipakai

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

1.) Fungsi dasar:

1. Fungsi print pada program merupakan fungsi yang umum dipakai untuk menampilkan suatu keluaran pada layar terminal. Fungsi ini berguna untuk menampilkan informasi kepada pengguna, seperti hasil input, pesan kesalahan, debugging dengan mencetak nilai variabel. Misalnya, print (“data rumah berhasil diubah”) digunakan untuk menampilkan data rmah yang baru di ubah.
2. Fungsi input digunakan untuk membaca input dari pengguna melalui keyboard. Misalnya, Username = input (“Masukkan Username: ”), fungsi input menerima nama user untuk login ke program.

2.) Struktur Kontrol:

1. Percabangan (if-else): struktur ini digunakan untuk Percabangan mengambil

keputusan berdasarkan kondisi tertentu. Misalnya, pada simpan data rumah yang tersimpan di dalam file JSON.

2. Perulangan (while-for): struktur ini digunakan untuk program dapat mengatur perintah-perintah dalam programnya agar berjalan secara berulang-ulang hingga pada suatu kondisi dimana syarat untuk perulangannya sudah tidak terpenuhi kembali. Misalnya, menampilkan menu admin atau user sampai pengguna memilih pengguna untuk keluar.

3.) Manipulasi Data

Program ini menggunakan JSON untuk menyimpan data rumah dan data akun, juga menggunakan OS untuk membersihkan layar dan juga menggunakan PrettyTable untuk mempercantik tabel.

4.) Library Eksternal

1. PrettyTable

Digunakan untuk membuat dan menampilkan tabel menjadi terstruktur atau menjadi lebih rapi sehingga mudah dibaca.

5.) Role – Based Access Control (RBAC)

Program ini membedakan akses dan kemampuan pengguna berdasarkan peran (role):

1. Admin : Memiliki akses penuh untuk menambahkan, mengubah, menghapus, dan melihat data rumah, serta mengelola data registrasi pengguna
2. User: Hanya memiliki akses untuk melihat data rumah dan menambahkan data baru.

BAB III

HASIL DAN PEMBAHASAN

3.1 Tampilan Program

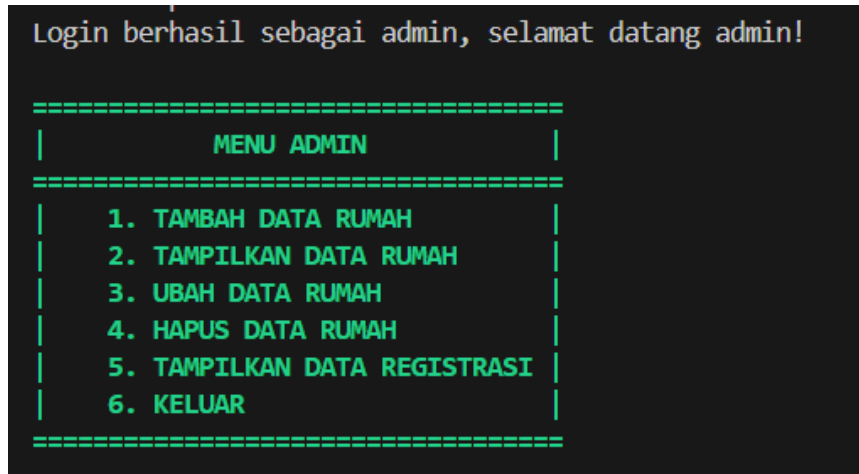
1. Tampilan Menu



Gambar 3. 1 Tampilan Menu

Pada tampilan menu jika pengguna ingin masuk sebagai admin maka akan memilih menu login kemudian program akan meminta input username dan password yang sudah terdaftar pada sistem. Jika pengguna adalah user maka memilih menu register terlebih dahulu dengan memasukkan username dan password yang baru kemudian jika sudah terdaftar baru memilih menu login dan masuk ke program. Dan jika ingin keluar pilih menu keluar program.

2. Tampilan Menu Admin



Gambar 3. 2 Tampilan Menu Admin

Pada menu admin terdapat 6 menu pilihan yaitu:

1. Tambah Data Rumah: Digunakan untuk menambahkan data rumah ke dalam sistem.
2. Tampilkan Data Rumah: Digunakan untuk menampilkan data-data rumah yang sudah tersimpan didalam program.
3. Ubah Data Rumah: Digunakan untuk mengubah data rumah yang ingin diubah.
4. Hapus Data Rumah: Digunakan untuk menghapus data rumah yang ingin kita hapus.
5. Tampilkan Data Registrasi: Digunakan untuk menampilkan data siapa saja yang masuk ke program dan hanya admin yang dapat melihat.
6. Keluar: Digunakan jika ingin keluar dari menu admin.

1.) Tampilan Tambah Data Rumah

```
=====
|           MENU ADMIN           |
=====
|  1. TAMBAH DATA RUMAH        |
|  2. TAMPILKAN DATA RUMAH     |
|  3. UBAH DATA RUMAH         |
|  4. HAPUS DATA RUMAH        |
|  5. TAMPILKAN DATA REGISTRASI |
|  6. KELUAR                   |
=====

Masukkan pilihan menu: 1
Masukkan nomor rumah: 21

=====
| Jenis Rumah |
=====
|  1. beton   |
|  2. kayu    |
=====

Pilih jenis rumah: 1
Masukkan nama pemilik rumah: tedy
Masukkan tahun berdiri.
Jangka tahun antara 1980-2024: 2020
Masukkan alamat rumah: jl.gelatik

=====
| Status Rumah |
=====
|  1. kosong   |
|  2. dijual   |
|  3. ditempati |
=====

Pilih status rumah: 1
Berhasil menambahkan data rumah.
```

Gambar 3. 3 Tampilan Tambah Data Rumah

Tampilan menu tersebut ketika pengguna memilih menu tambah data rumah, pengguna diminta untuk menginput nomor rumah, kemudian akan muncul pilih jenis rumah, jika sudah pengguna diminta memasukkan nama pengguna rumah, masukkan tahun berdiri dengan jangka tahun yang ada, setelah itu masukkan alamat rumah, dan pengguna memilih status rumah jika sudah data rumah berhasil ditambahkan.

2.) Tampilkan Data Rumah

Masukkan pilihan menu: 2

No	No Rumah	Jenis	Pemilik	Tahun	Alamat	status
1	21	beton	tedy	2020	jl.gelatik	kosong
2	22	kayu	nadia	2021	jl.perjuangan	dijual
3	23	beton	angel	2023	jl.pramuka	ditempati

MENU ADMIN	
1.	TAMBAH DATA RUMAH
2.	TAMPILKAN DATA RUMAH
3.	UBAH DATA RUMAH
4.	HAPUS DATA RUMAH
5.	TAMPILKAN DATA REGISTRASI
6.	KELUAR

Gambar 3. 4 Tampilkan Data Rumah

Pada menu ini menampilkan tampilan data-data rumah yang tersimpan di sistem.

3.) Tampilan Ubah Data Rumah

```
=====
|          MENU ADMIN          |
=====
|  1. TAMBAH DATA RUMAH      |
|  2. TAMPILKAN DATA RUMAH   |
|  3. UBAH DATA RUMAH       |
|  4. HAPUS DATA RUMAH      |
|  5. TAMPILKAN DATA REGISTRASI |
|  6. KELUAR                  |
=====

Masukkan pilihan menu: 3
+---+---+---+---+---+---+---+
| No | No Rumah | Jenis | Pemilik | Tahun | Alamat | status |
+---+---+---+---+---+---+---+
| 1 | 21 | beton | tedy | 2020 | jl.gelatik | kosong |
| 2 | 22 | kayu | nadia | 2021 | jl.perjuangan | dijual |
| 3 | 23 | beton | angel | 2023 | jl.pramuka | ditempati |
+---+---+---+---+---+---+---+

Masukkan nomor rumah yang ingin diupdate: 22
Masukkan nomor rumah baru (22): 24

=====
| Jenis Rumah |
=====
| 1. beton |
| 2. kayu |
=====

Masukkan jenis rumah baru (kayu): 2
Masukkan nama pemilik rumah baru (nadia): nadia rahmah
Masukkan tahun berdiri baru (2021):
Jangka tahun antara 1980-2024: 2024
Masukkan alamat rumah baru (jl.perjuangan): jl.jakarta

=====
| Status Rumah |
=====
| 1. kosong |
| 2. dijual |
| 3. ditempati |
=====

Masukkan status rumah baru (dijual): 1
Data rumah berhasil diubah.
```

Gambar 3. 5 Tampilan Ubah Data Rumah

Pada menu ubah data rumah pengguna diminta memasukkan nomor rumah yang ingin diupdate, kemudian masukkan nomor rumah baru, pengguna diminta untuk memilih jenis rumah, masukkan nama pemilik rumah baru, masukkan tahun berdiri rumah baru, masukkan alamat rumah baru, pengguna diminta untuk memilih status rumah jika sudah

maka data rumah berhasil diubah.

4.) Tampilan Hapus Data Rumah

```
=====
|           MENU ADMIN           |
=====
|  1. TAMBAH DATA RUMAH        |
|  2. TAMPILKAN DATA RUMAH     |
|  3. UBAH DATA RUMAH         |
|  4. HAPUS DATA RUMAH        |
|  5. TAMPILKAN DATA REGISTRASI|
|  6. KELUAR                   |
=====

Masukkan pilihan menu: 4
+---+---+---+---+---+---+---+
| No | No Rumah | Jenis | Pemilik | Tahun | Alamat | status |
+---+---+---+---+---+---+---+
| 1  | 21       | beton | tedy    | 2020  | jl.gelatik | kosong |
| 2  | 23       | beton | angel   | 2023  | jl.pramuka | ditempati |
| 3  | 24       | kayu  | nadia rahmah | 2024  | jl.jakarta | kosong |
+---+---+---+---+---+---+---+

Masukkan nomor rumah yang ingin dihapus: 24
Rumah dengan nomor '24' berhasil dihapus.

=====
|           MENU ADMIN           |
=====
|  1. TAMBAH DATA RUMAH        |
|  2. TAMPILKAN DATA RUMAH     |
|  3. UBAH DATA RUMAH         |
|  4. HAPUS DATA RUMAH        |
|  5. TAMPILKAN DATA REGISTRASI|
|  6. KELUAR                   |
=====

Masukkan pilihan menu: 2
+---+---+---+---+---+---+---+
| No | No Rumah | Jenis | Pemilik | Tahun | Alamat | status |
+---+---+---+---+---+---+---+
| 1  | 21       | beton | tedy    | 2020  | jl.gelatik | kosong |
| 2  | 23       | beton | angel   | 2023  | jl.pramuka | ditempati |
+---+---+---+---+---+---+---+
```

Gambar 3. 6 Tampilan hapus Data Rumah

Pada menu ini pengguna diminta untuk memasukkan nomor rumah yang ingin dihapus, setelah itu data rumah dengan nomor rumah tersebut berhasil dihapus.

5.) Tampilkan Data Registrasi

```

=====
|           MENU ADMIN           |
=====
|  1. TAMBAH DATA RUMAH        |
|  2. TAMPILKAN DATA RUMAH     |
|  3. UBAH DATA RUMAH         |
|  4. HAPUS DATA RUMAH        |
|  5. TAMPILKAN DATA REGISTRASI |
|  6. KELUAR                    |
=====

Masukkan pilihan menu: 5

+-----+-----+
| Username | Role |
+-----+-----+
| admin    | admin |
| tedy     | user  |
| angel    | user  |
| nadia    | user  |
+-----+-----+

```

Gambar 3. 7 Tampilan Data Registrasi

Gambar 3.7 Tampilan Data Registrasi

Pada tampilan data registrasi akan menampilkan siapa saja yang telah registrasi ke program ini dan hanya admin yang dapat melihat.

3. Tampilan Menu User

```

=====
|           MENU USER           |
=====
|  1. TAMPILKAN DATA RUMAH     |
|  2. TAMBAH DATA RUMAH       |
|  3. KELUAR                    |
=====

Masukkan pilihan menu: 1

+---+---+---+---+---+---+---+
| No | No Rumah | Jenis | Pemilik | Tahun | Alamat | status |
+---+---+---+---+---+---+---+
| 1  | 21       | beton | tedy    | 2020  | jl.gelatik | kosong |
| 2  | 23       | beton | angel   | 2023  | jl.pramuka | ditempati |
+---+---+---+---+---+---+---+

```

Gambar 3. 8 Tampilan Menu User

Pada Menu User ada 3 pilihan:

1. Tampilan Data Rumah : Menampilkan data-data rumah yang sudah tersimpan di sistem.

2. Tambahkan Data Rumah : Menambahkan data rumah yang ingin ditambah
3. Keluar : Digunakan jika ingin keluar dari menu user.

1.) Tampilan Tambahkan Data Rumah

```
=====
|           MENU USER           |
=====
|  1. TAMPILKAN DATA RUMAH    |
|  2. TAMBAHKAN DATA RUMAH    |
|  3. KELUAR                    |
=====

Masukkan pilihan menu: 2
Masukkan nomor rumah: 25

=====
| Jenis Rumah |
=====
|  1. beton   |
|  2. kayu    |
=====

Pilih jenis rumah: 2
Masukkan nama pemilik rumah: angelina
Masukkan tahun berdiri.
Jangka tahun antara 1980-2024: 2009
Masukkan alamat rumah: jl.sambaliung

=====
| Status Rumah |
=====
|  1. kosong   |
|  2. dijual   |
|  3. ditempati |
=====

Pilih status rumah: 2
Berhasil menambahkan data rumah.
```

Gambar 3. 9 Tampilan Tambahkan Data Rumah

Pada menu ini pengguna diminta memasukkan nomor rumah kemudian memilih jenis rumah, jika sudah memasukkan nama pemilik rumah, memasukkan tahun berdiri, masukkan alamat rumah, selanjutnya pilih status rumah, jika sudah data rumah berhasil ditambahkan.

4. Tampilan Menu Keluar



Gambar 3. 10 Keluar Program

Pada menu keluar program digunakan jika pengguna ingin keluar dari program.

a. Source Code

```

1  # MANAJEMEN PENDATAAN RUMAH
2  import os
3  import json
4  from prettytable import PrettyTable
5
6
7  # Data utama untuk rumah dan akun
8  data_rumah = {}
9  akun = {}
10 sudah_login = False
11
12 # Fungsi pembersihan layar
13 def bersihkan():
14     os.system("cls" if os.name == "nt" else "clear")
15     print("\033c", end="")
16 # Fungsi penyimpanan dan pemuatan data rumah dari file JSON
17 def simpan_data(filename="data_rumah.json"):
18     with open(filename, "w") as file:
19         json.dump(data_rumah, file)
20
21 def muat_data(filename="data_rumah.json"):
22     global data_rumah
23     if os.path.exists(filename):
24         with open(filename, "r") as file:
25             data_rumah = json.load(file)
26     else:
27         data_rumah = {}
28
29 # Fungsi penyimpanan dan pemuatan data akun dari file JSON
30 def simpan_akun(filename="data_akun.json"):
31     with open(filename, "w") as file:
32         json.dump(akun, file)
33
34 def muat_akun(filename="data_akun.json"):
35     global akun
36     if os.path.exists(filename):
37         with open(filename, "r") as file:
38             akun = json.load(file)
39     else:
40         akun = {}

```

```

43     return {"tedy": {"password": "123", "role": "admin"}} # 1 akun admin yang disiapkan
44 # Fungsi input integer dengan validasi
45 def input_int(pesan):
46     try:
47         return int(input(pesan))
48     except ValueError:
49         print("Input harus berupa angka.")
50         return input_int(pesan)
51
52 # Fungsi validasi pilihan menu
53 def validasi_pilihan_menu(pilihan, max_pilihan):
54     try:
55         if pilihan < 1 or pilihan > max_pilihan:
56             raise ValueError("Pilihan tidak valid, coba lagi.")
57         return pilihan
58     except ValueError as e:
59         print(e)
60         return validasi_pilihan_menu(input_int("Masukkan pilihan: "), max_pilihan)
61
62 # Fungsi tampilan menu utama
63 def tampilkan_menu_utama():
64     bersihkan()
65     print("""\
66     \033[1;32m
67     | MANAJEMEN PENDATAAN RUMAH
68
69     |
70     |
71     |
72     |
73     |
74     |
75     | •—————•( )•—————•
76     | • 1. Login
77     | • 2. Register
78     | • 3. Keluar Program
79     | \033[0m
80     | """)

```

```

82 # Fungsi login
83 def login(username, password):
84     global sudah_login
85     try:
86         if username in akun and akun[username]["password"] == password:
87             role = akun[username]["role"]
88             sudah_login = True
89             print(f"Login berhasil sebagai {role}, selamat datang {username}!")
90             if role == "admin":
91                 menu_admin()
92             elif role == "user":
93                 menu_user()
94         else:
95             raise ValueError("Username atau password salah")
96     except ValueError as e:
97         print(e)
98         os.system("pause")
99
100 # Fungsi registrasi pengguna
101 def register():
102     global akun
103     try:
104         username = input("Masukkan username: ")
105         if username in akun:
106             raise ValueError("Username sudah terdaftar")
107
108         password = input("Masukkan password: ")
109         akun[username] = {"password": password, "role": "user"} # Semua pengguna baru menjadi 'user'
110         simpan_akun() # Simpan ke file data akun
111         print("Registrasi berhasil, akun berhasil didaftarkan")
112         input("Silakan lanjut login. Tekan Enter untuk melanjutkan...")
113     except ValueError as e:
114         print(e)
115         os.system("pause")
116
117 # Fungsi untuk menampilkan data rumah dalam format tabel
118 def tampilkan_data_rumah():
119     muat_data()
120     if not data_rumah:
121         print("Data rumah tidak ada.")
122     else:

```

```

122     else:
123         table = PrettyTable()
124         table.field_names = ["No", "No Rumah", "Jenis", "Pemilik", "Tahun", "Alamat", "status"]
125         for i, (no, info) in enumerate(data_rumah.items(), start=1):
126             table.add_row([i, no, info["jenis"], info["pemilik"], info["tahun"], info["alamat"], info["status"]])
127         print(table)
128
129 # Fungsi tampilkan data registrasi untuk admin
130 def tampilkan_data_registrasi():
131     if not akun:
132         print("Data registrasi tidak ada.")
133     else:
134         table = PrettyTable()
135         table.field_names = ["Username", "Role"]
136         for username, info in akun.items():
137             table.add_row([username, info["role"]])
138         print(table)
139
140 # Fungsi tambah data rumah dengan menu jenis dan status rumah
141 def tambah_data_rumah():
142     muat_data()
143
144     while True:
145         no_rumah = str(input_int("Masukkan nomor rumah: "))
146         if no_rumah in data_rumah:
147             print("Nomor rumah sudah ada. Silakan coba dengan nomor lain.")
148         else:
149             break
150
151
152
153     # Menampilkan menu jenis rumah
154     print("""\
155 =====
156 | Jenis Rumah |
157 =====
158 | 1. beton   |
159 | 2. kayu    |
160 =====
161 | """)

```

```

162     pilihan_jenis = validasi_pilihan_menu(input_int("Pilih jenis rumah: "), 2)
163     jenis_rumah = "beton" if pilihan_jenis == 1 else "kayu"
164
165     # Validasi nama pemilik hanya berisi huruf
166     while True:
167         nama_pemilik_rumah = input("Masukkan nama pemilik rumah: ").strip() # " nadia "
168         if nama_pemilik_rumah.replace(" ", "").isalpha(): # "nadia123"
169             break
170         print("Nama pemilik hanya boleh berisi huruf. Silakan coba lagi.")
171
172     # Validasi tahun berdiri
173     while True:
174         tahun_berdiri = input_int("Masukkan tahun berdiri.\nJangka tahun antara 1980-2024: ")
175         if 1980 <= tahun_berdiri <= 2024:
176             break
177         print("Tidak valid, Tahun harus berada dalam rentang 1980-2024.\nSilahkan input ulang dan coba lagi")
178
179     alamat_rumah = input("Masukkan alamat rumah: ")
180
181     # Menampilkan menu status rumah
182     print("""\
183     =====
184     | Status Rumah |
185     =====
186     | 1. kosong    |
187     | 2. dijual    |
188     | 3. ditempati |
189     =====
190     """)
191     pilihan_status = validasi_pilihan_menu(input_int("Pilih status rumah: "), 3)
192     status_rumah = ["kosong", "dijual", "ditempati"][pilihan_status - 1]
193
194     data_rumah[no_rumah] = {
195         "jenis": jenis_rumah,
196         "pemilik": nama_pemilik_rumah,
197         "tahun": tahun_berdiri,
198         "alamat": alamat_rumah,
199         "status": status_rumah
200     }
201     simpan_data()
202     print("Berhasil menambahkan data rumah.")

```

```

205 # Fungsi ubah data rumah
206 def ubah_data_rumah():
207     muat_data()
208     tampilkan_data_rumah()
209     pilih_rumah = str(input("Masukkan nomor rumah yang ingin diupdate: ")).strip()
210
211     if pilih_rumah in data_rumah:
212         no_rumah_baru = str(input_int(f"Masukkan nomor rumah baru ({pilih_rumah}): "))
213
214         # Pilih jenis rumah
215         print("""\
216 =====
217 | Jenis Rumah |
218 =====
219 | 1. beton    |
220 | 2. kayu     |
221 =====
222 """)
223         pilihan_jenis = validasi_pilihan_menu(input_int(f"Masukkan jenis rumah baru ({data_rumah[pilih_rumah]['jenis']}): "), 2)
224         jenis_baru = "beton" if pilihan_jenis == 1 else "kayu"
225
226         # Validasi nama pemilik baru
227         while True:
228             pemilik_baru = input(f"Masukkan nama pemilik rumah baru ({data_rumah[pilih_rumah]['pemilik']}): ").strip()
229             if pemilik_baru.replace(" ", "").isalpha():
230                 break
231             print("Nama pemilik hanya boleh berisi huruf. Silakan coba lagi.")
232
233         # Validasi tahun berdiri baru
234         while True:
235             tahun_baru = input_int(f"Masukkan tahun berdiri baru ({data_rumah[pilih_rumah]['tahun']}).\nJangka tahun antara 1980-2024: ")
236             if 1980 <= tahun_baru <= 2024:
237                 break
238             print("Tidak valid, Tahun harus berada dalam rentang 1980-2024.\nSilahkan input ulang dan coba lagi")
239
240         alamat_baru = input(f"Masukkan alamat rumah baru ({data_rumah[pilih_rumah]['alamat']}): ")
241
242         # Pilih status rumah
243         print("""\
244

```

```

245 =====
246 | Status Rumah |
247 =====
248 | 1. kosong |
249 | 2. dijual |
250 | 3. ditempati |
251 =====
252 """
253 pilihan_status = validasi_pilihan_menu(input_int(f"Masukkan status rumah baru ({data_rumah[pilih_rumah]['status']): "), 3)
254 status_baru = ["kosong", "dijual", "ditempati"][pilihan_status - 1]
255
256 # Update data rumah
257 data_rumah[no_rumah_baru] = {
258     "jenis": jenis_baru,
259     "pemilik": pemilik_baru,
260     "tahun": tahun_baru,
261     "alamat": alamat_baru,
262     "status": status_baru
263 }
264 if no_rumah_baru != pilih_rumah:
265     del data_rumah[pilih_rumah]
266     simpan_data()
267     print("Data rumah berhasil diubah.")
268 else:
269     print("Nomor rumah tidak tersedia.")
270
271
272 # Fungsi hapus data rumah
273 def hapus_data_rumah():
274     muat_data()
275     tampilkan_data_rumah()
276     pilih_rumah = str(input("Masukkan nomor rumah yang ingin dihapus: "))
277     if pilih_rumah in data_rumah:
278         del data_rumah[pilih_rumah]
279         simpan_data()
280         print(f"Rumah dengan nomor '{pilih_rumah}' berhasil dihapus.")
281     else:
282         print("Nomor rumah tidak valid.")
283

```



```

284 # Menu admin
285 def menu_admin():
286     while True:
287         print("""\
288 =====
289 |           MENU ADMIN           |
290 =====
291 | 1. TAMBAH DATA RUMAH          |
292 | 2. TAMPILKAN DATA RUMAH       |
293 | 3. UBAH DATA RUMAH            |
294 | 4. HAPUS DATA RUMAH           |
295 | 5. TAMPILKAN DATA REGISTRASI  |
296 | 6. KELUAR                       |
297 =====
298 """)
299     pilihan = validasi_pilihan_menu(input_int("Masukkan pilihan menu: "), 6)
300     if pilihan == 1:
301         tambah_data_rumah()
302     elif pilihan == 2:
303         tampilkan_data_rumah()
304     elif pilihan == 3:
305         ubah_data_rumah()
306     elif pilihan == 4:
307         hapus_data_rumah()
308     elif pilihan == 5:
309         tampilkan_data_registrasi()
310     elif pilihan == 6:
311         break
312
313 # Menu user
314 def menu_user():
315     while True:
316         print("""\
317 =====
318 |           MENU USER            |
319 =====
320 | 1. TAMPILKAN DATA RUMAH       |
321 | 2. TAMBAHKAN DATA RUMAH       |
322 | 3. KELUAR                       |
323 =====
324 """)

```

```

325     pilih = validasi_pilihan_menu(input_int("Masukkan pilihan menu: "), 4)
326     if pilih == 1:
327         tampilkan_data_rumah()
328     elif pilih == 2:
329         tambah_data_rumah()
330     elif pilih == 3:
331         break
332
333 # Fungsi utama program
334 def main():
335     global data_rumah, akun
336     muat_data()
337     muat_akun()
338     while True:
339         tampilkan_menu_utama()
340         pilihan = validasi_pilihan_menu(input_int("PILIH: "), 3)
341         if pilihan == 1:
342             username = input("Masukkan username: ")
343             password = input("Masukkan password: ")
344             login(username, password)
345         elif pilihan == 2:
346             register() # Memanggil fungsi register
347         elif pilihan == 3:
348             print("Keluar dari program.")
349             break
350
351 if __name__ == "__main__":
352     main()
353

```

BAB IV

PENUTUP

4.1 Kesimpulan

Berdasarkan hasil pengembangan dan pengujian program Manajemen Pendataan Rumah, dapat disimpulkan beberapa hal berikut:

1. Tujuan Program:
Program ini berhasil mencapai tujuannya sebagai aplikasi berbasis terminal untuk mempermudah manajemen data rumah dan akun pengguna. Fitur-fiturnya mendukung pengelolaan data dengan proses login dan otorisasi yang sesuai peran (admin dan user).
2. Pencapaian Utama:
 - Manajemen Data Rumah:
Program memungkinkan pengguna untuk menambah, melihat, mengubah, dan menghapus data rumah dengan validasi input yang baik, seperti:
 - Nomor rumah unik.
 - Validasi nama pemilik hanya berupa huruf.
 - Tahun rumah berada dalam rentang yang ditentukan.
 - ManajemenAkun:
Pengguna dapat mendaftarkan akun baru dan login menggunakan data yang tersimpan dengan persisten di file JSON.
 - Tampilan Data yang Informatif:
Data rumah dan akun disajikan dalam bentuk tabel yang rapi menggunakan library PrettyTable, sehingga mempermudah pengguna dalam membaca informasi.
3. Kelebihan Program:
 - Antar muka sederhana dan mudah digunakan.
 - Validasi input yang ketat membantu mengurangi kesalahan data.
 - Penyimpanan data menggunakan file JSON memungkinkan data dapat diakses kembali setelah program ditutup.
4. Kendala dan Kekurangan:
 - Keamanan Data:
Tidak ada enkripsi pada penyimpanan data akun, sehingga rentan jika file JSON diakses secara langsung.
 - Keterbatasan
Penyimpanan berbasis JSON kurang cocok untuk menangani data dalam jumlah besar dibandingkan dengan database relasional.
 - Minimnya Fitur Pengelolaan Data Lanjutan:
Belum tersedia fitur pencarian, filter, atau laporan terperinci terkait data rumah.

5. Rekomendasi Pengembangan:

- Implementasikan enkripsi untuk kata sandi dan tingkatkan keamanan data pengguna.
- Gunakan database relasional seperti SQLite atau MySQL untuk penyimpanan data agar lebih efisien dan skalabel.
- Tambahkan fitur pencarian, filter, dan laporan ringkasan data rumah untuk meningkatkan fungsionalitas program.
- Beralih ke antarmuka berbasis GUI untuk meningkatkan pengalaman pengguna.

4.2 Saran

Berikut beberapa saran untuk program yang dibuat:



1. Keamanan:
 - Gunakan algoritma enkripsi seperti bcrypt untuk menyimpan kata sandi dengan aman.
 - Tambahkan fitur timeout pada login untuk mencegah brute-force attack.
2. Fitur Tambahan:
 - Filter dan Pencarian: Tambahkan opsi pencarian data rumah berdasarkan kriteria tertentu (misalnya status, jenis rumah, tahun berdiri, atau lokasi).
 - Histori Perubahan Data: Simpan riwayat perubahan data untuk pelacakan dan audit.
3. Fitur Logout: Berikan opsi logout untuk mengamankan sesi pengguna.
- Peningkatan Efisiensi Data:
 - Gunakan database relasional seperti SQLite atau MySQL untuk menyimpan data rumah dan akun. Hal ini akan mempermudah pengelolaan data yang lebih besar.
4. Antarmuka Pengguna:
 - Migrasikan program ke antarmuka berbasis GUI menggunakan library seperti Tkinter atau PyQt untuk meningkatkan pengalaman pengguna.
 - Tambahkan fitur pagination pada tabel untuk menampilkan data dalam jumlah besar.
5. Uji Coba dan Validasi:
 - Terapkan unit testing untuk memvalidasi fungsi-fungsi utama program.
 - Pastikan validasi input mencakup lebih banyak skenario untuk menghindari edge cases.



DAFTAR PUSTAKA

Rosaly, R., & Prasetyo, A. (2019). Pengertian Flowchart Beserta Fungsi dan Simbol-simbol Flowchart yang Paling Umum Digunakan.

Romzi, M., & Kurniawan, B. (2020). Pembelajaran Pemrograman Python Dengan Pendekatan Logika Algoritma. JTIM: Jurnal Teknik Informatika Mahakarya, 3(2), 37-44.

LAMPIRAN

Tanggal Konsultasi : Kamis, 31 Oktober 2024	
<p>Uraian / Pembahasan :</p> <ol style="list-style-type: none"> 1. Tambahkan menu "keluar/logout" pada flowchart di bagian menu admin. 2. Tambahkan Tipe data list 3. Perjelas dengan adanya jangka tahun (rentang tahun) 4. Buat satu file eksternal untuk menyimpan data registrasi akun. 5. point + terdapat (creat/menambah) di menu user 6. Perbaiki fungsi CLS agar dapat menghapus semua tampilan dengan benar. 7. Diperhatikan, penggunaan dictionary pada nomor rumah, jika nomor rumah sudah ada, maka program error karena menggunakan dictionary. 	
Asisten Lab	Ketua Kelompok
	
Nama: Christen Fom	Nama: M. Tedy Azhari

Tanggal Konsultasi : Rabu, 13 November 2024.	
<p>Uraian / Pembahasan :</p> <ol style="list-style-type: none"> 1. Flowchart lebih dirapikan lagi 2. Program ditambahkan komen penjelasan. 	
Asisten Lab	Ketua Kelompok
	
Nama: Christen Farrey	Nama: M. Tedy Azhari

Gambar 4. 1 Kartu Konsul