

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

**SKRIPSI**

**SISTEM INFORMASI *MANAGEMENT* PADA LAYANAN  
KLINIK MOULYA BERBASIS *WEBSITE***



**Diajukan oleh :**

**INDI FEBRINA LATIFAH SIDIK**

**021190010**

**Untuk Memenuhi Sebagian dari Syarat  
Mencapai Gelar Sarjana Komputer**

**PALEMBANG**

**2023**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

**SKRIPSI**

**SISTEM INFORMASI *MANAGEMENT* PADA LAYANAN  
KLINIK MOULYA BERBASIS *WEBSITE***



**Diajukan oleh :**

**INDI FEBRINA LATIFAH SIDIK**

**021190010**

**Untuk Memenuhi Sebagian dari Syarat  
Mencapai Gelar Sarjana Komputer**

**PALEMBANG**

**2023**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

---

**HALAMAN PENGESAHAN PEMBIMBING SKRIPSI**

**NAMA : INDI FEBRINA LATIFAH SIDIK**

**NOMOR POKOK : 021190010**

**PROGRAM STUDI : SISTEM INFORMASI**

**JENJANG PENDIDIKAN : STRATA SATU**

**JUDUL : SISTEM INFORMASI *MANAGEMENT* PADA  
LAYANAN KLINIK MOULYA BERBASIS  
*WEBSITE***

**Tanggal : 23 Agustus 2023**  
**Pembimbing**

**Mengetahui,**  
**Rektor**

**Jaka Purnama, S.Kom., M.Kom.**  
**NIDN : 0219089401**

**Benedictus Effendi, S.T., M.T.**  
**NIP : 09.PCT.13**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

---

**HALAMAN PENGESAHAN PENGUJI SKRIPSI**

**NAMA : INDI FEBRINA LATIFAH SIDIK**

**NOMOR POKOK : 021190010**

**PROGRAM STUDI : SISTEM INFORMASI**

**JENJANG PENDIDIKAN : STRATA SATU**

**JUDUL : SISTEM INFORMASI *MANAGEMENT* PADA  
LAYANAN KLINIK MOULYA BERBASIS  
*WEBSITE***

**Tanggal : 23 Agustus 2023**  
**Penguji 1**

**Tanggal : 22 Agustus 2023**  
**Penguji 2**

**Yayuk Ike Meilani, S.Kom., M.Kom.**  
**NIDN : 0224059102**

**Yarza Aprizal, S.Kom., M.Kom.**  
**NIDN : 0212049302**

**Menyetujui,**  
**Rektor**

**Benedictus Effendi, S.T., M.T.**  
**NIP : 09.PCT.13**

## **MOTTO :**

*“Jadilah lebih baik dari dirimu yang sebelumnya, dan tunjukkan pada mereka yang merendahkanmu bahwa kita juga bisa bahkan lebih baik dari mereka”*

## **Kupersembahkan Kepada :**

- *Kedua orang tua saya yang saya cintai yang telah melahirkan, dan membesarkan saya. Khususnya untuk alm. ayah saya, semoga beliau dapat terus membimbing saya dari atas sana.*
- *Keluarga besar saya yang selalu membantu saya*
- *Sahabat saya Silvia Ratnasari yang selalu membantu saya dikala saya kesulitan, mendukung saya, menyemangati saya dan selalu bersama saya baik ketika senang maupun susah.*
- *Bengek Keleb (Istiqomatul Fadilah dan Putri Natasya Deani) yang selalu mau berjuang bersama saya, dan mengingatkan saya jika saya melakukan kesalahan.*
- *Teman-teman saya yang tidak bisa saya sebutkan satu persatu*
- *Kepada Dosen Pembimbing saya Jaka Purnama, S.Kom., M.Kom yang telah membantu dan membimbing saya.*

## KATA PENGANTAR

Dengan mengucapkan Alhamdulillah peneliti panjatkan puji dan syukur atas kehadiran Allah SWT, yang telah melimpahkan berkat dan rahmat-Nya kepada dengan kelancaran dalam menyelesaikan penulisan skripsi yang berjudul **“Sistem Informasi Management Pada Layanan Klinik Moulya Berbasis Website”** ini dapat diselesaikan guna memenuhi salah satu syarat dalam menyelesaikan program studi Sistem Informasi Program Sarjana tepat pada waktunya.

Dalam penulisan skripsi ini, peneliti sadari telah mendapatkan banyak bantuan baik berupa bimbingan, petunjuk, saran, dan arahan dari dosen pembimbing serta dari berbagai pihak. Oleh sebab itu, pada kesempatan ini peneliti ingin menyampaikan rasa hormat dan terimakasih kepada :

1. Kedua orang tua yang saya cintai
2. Rektor Institut Teknologi dan Bisnis PalComTech Palembang Bapak Benedictus Effendi, S.T., M.T.
3. Wakil Rektor 1 Institut Teknologi dan Bisnis PalComTech Palembang Ibu Adelin, S.T., M.Kom.
4. Kepala Program Studi S1 Sistem Informasi, Ibu Dini Hari Pertiwi, S.Kom., M.Kom.
5. Dosen Pembimbing, Bapak Jaka Purnama, S.Kom., M.Kom.
6. Dosen Penguji 1, Yayuk Ike Meilani, S.Kom., M.Kom.
7. Dosen Penguji 2, Yarza Aprizal, S.Kom., M.Kom.
8. Kepada seluruh keluarga besar, teman, dan sahabat seperjuangan,

Yang telah banyak membantu dan mendukung peneliti sehingga dapat menyelesaikan penulisan skripsi ini.

Demikian kata pengantar dari peneliti, dengan harapan semoga skripsi ini berguna dan bermanfaat bagi semua pihak yang membutuhkan, peneliti menyadari bahwa penulisan pada skripsi ini masih memiliki banyak kekurangan baik dari

segi kalimat maupun tata bahasanya sehingga membutuhkan kritik maupun saran yang membangun untuk menghasilkan sesuatu yang lebih baik.

Akhir kata, atas perhatiannya peneliti ucapkan terima kasih.

Palembang, Agustus 2023

Peneliti

## DAFTAR ISI

<b>HALAMAN JUDUL.....</b>	<b>i</b>
<b>HALAMAN PENGESAHAN PEMBIMBING.....</b>	<b>ii</b>
<b>HALAMAN PENGESAHAN PENGUJI.....</b>	<b>iii</b>
<b>HALAMAN MOTTO DAN PERSEMBAHAN.....</b>	<b>iv</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>DAFTAR ISI.....</b>	<b>vii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xi</b>
<b>DAFTAR TABEL.....</b>	<b>xiv</b>
<b>DAFTAR LAMPIRAN.....</b>	<b>xvi</b>
<b>ABSTRAK.....</b>	<b>xvii</b>
<b>BAB 1       PENDAHULUAN</b>	
1.1..Latar Belakang.....	1
1.2..Rumusan Masalah Penelitian.....	2
1.3..Ruang Lingkup Penelitian.....	2
1.4..Tujuan Penelitian.....	4
1.5..Manfaat Penelitian.....	5
1.5.1. Manfaat Bagi Peneliti.....	5
1.5.2. Manfaat Bagi Pihak Klinik Moulya.....	5
1.5.3. Manfaat Bagi Pihak Akademik.....	5
1.6. Sistematika Penulisan.....	5

## **BAB II            GAMBARAN UMUM PERUSAHAAN**

2.1..Profil Perusahaan.....	8
2.1.1. Sejarah Perusahaan.....	8
2.2. Visi dan Misi.....	8
2.2.1. Visi.....	8
2.2.2. Misi.....	9
2.3. Struktur Organisasi.....	9
2.4. Tugas dan Wewenang.....	10

## **BAB III            TINJAUAN PUSTAKA**

3.1..Teori Pendukung.....	12
3.1.1. <i>Database</i> .....	12
3.1.2. PHP.....	12
3.1.3. <i>Web</i> .....	12
3.1.4. Sistem.....	13
3.1.5. Informasi.....	13
3.2. Penelitian Terdahulu.....	13

## **BAB IV            METODOLOGI PENELITIAN**

4.1..Lokasi dan Jadwal Penelitian.....	18
4.1.1. Lokasi.....	18
4.1.2. Jadwal Penelitian.....	18
4.2. Jenis Data.....	19
4.2.1. Data Primer.....	19
4.2.2. Data Sekunder.....	19
4.3. Teknik Pengumpulan Data.....	20
4.3.1. Wawancara.....	20
4.3.2. Observasi.....	20
4.3.3. Dokumentasi.....	20
4.3.4. Studi Pustaka.....	21
4.4. Alat dan Teknik Pengembangan Sistem.....	21

4.4.1. Alat Pengembangan Sistem.....	21
4.4.1.1. <i>Flowchart</i> .....	21
4.4.1.2. <i>Data Flow Diagram (DFD)</i> .....	24
4.4.1.3. <i>Entity Relationship Diagram (ERD)</i> .....	26
4.4.2. Teknik Pengembangan Sistem.....	29
4.4.2.1. Metode <i>Prototype</i> .....	29
4.5. Alat dan Teknik Pengujian.....	31
4.5.1. <i>Black-Box Testing</i> .....	32

## **BAB V HASIL DAN PEMBAHASAN**

5.1..Hasil.....	34
5.1.1. Pengumpulan Kebutuhan.....	34
5.1.2. Membangun <i>Prototyping</i> .....	34
5.1.2.1. <i>Flowchart</i> yang Berjalan.....	35
5.1.2.1.1. Prosedur Berjalan Pasien Baru.....	35
5.1.2.1.2. Prosedur Berjalan Pasien Lama.....	38
5.1.2.2. <i>Flowchart</i> Yang Diusulkan.....	41
5.1.2.2.1. Tata Usaha/Admin.....	41
5.1.2.2.2. Penanggung Jawab Klinik.....	44
5.1.2.2.3. Rekam Medis.....	47
5.1.2.2.4. Farmasi.....	49
5.1.2.2.5. Dokter.....	52
5.1.2.2.6. Pasien.....	54
5.1.2.3. Diagram Alir Data.....	56
5.1.2.3.1. Diagram Konteks.....	56
5.1.2.3.2. Diagram Level 0.....	57
5.1.2.3.3. Diagram Rinci.....	63
5.1.2.3.3.1. Level 1 Proses 1.....	63
5.1.2.3.3.2. Level 1 Proses 8.....	65
5.1.2.3.3.3. Level 1 Proses 11.....	66
5.1.2.3.3.4. Level 1 Proses 12.....	67

5.1.2.4. <i>Entity Relationship Diagram (ERD)</i> .....	69
5.1.2.5. Struktur Tabel.....	75
5.1.2.6. Desain <i>Interface</i> .....	84
5.1.3. Evaluasi <i>Prototyping</i> .....	96
5.1.4. Mengkodekan Sistem.....	97
5.1.5. Menguji Sistem.....	111
5.1.5.1. <i>Black-Box Testing</i> .....	111
5.1.5.1.1. Skenario Pengujian.....	112
5.1.5.1.2. Hasil Pengujian.....	119
5.1.6. Penggunaan Sistem.....	131

## **BAB VI      PENUTUP**

6.1..Kesimpulan.....	132
6.2..Saran.....	132

<b>DAFTAR PUSTAKA.....</b>	<b>xix</b>
----------------------------	------------

<b>HALAMAN LAMPIRAN.....</b>	<b>xxii</b>
------------------------------	-------------

## DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi.....	9
Gambar 4.1 Tahapan Dalam Metode <i>Prototype</i> .....	29
Gambar 5.1 Prosedur Berjalan Pasien Baru.....	35
Gambar 5.2 Prosedur Berjalan Pasien Lama.....	38
Gambar 5.3 <i>Flowchart</i> yang Diusulkan Tata Usaha.....	41
Gambar 5.4 <i>Flowchart</i> yang Diusulkan Penanggung Jawab.....	45
Gambar 5.5 <i>Flowchart</i> yang Diusulkan Rekam Medis.....	48
Gambar 5.6 <i>Flowchart</i> yang Diusulkan Farmasi.....	50
Gambar 5.7 <i>Flowchart</i> yang Diusulkan Dokter.....	53
Gambar 5.8 <i>Flowchart</i> yang Diusulkan Pasien.....	55
Gambar 5.9 Diagram Konteks.....	56
Gambar 5.10 Diagram Level 0.....	58
Gambar 5.11 Level 1 Proses 1.....	64
Gambar 5.12 Diagram Level 1 Proses 8.....	65
Gambar 5.13 Diagram Level 1 Proses 11.....	66
Gambar 5.14 Diagram Level 1 Proses 12.....	67
Gambar 5.15 <i>Entity Relationship Diagram</i> .....	70
Gambar 5.16 Tampilan Daftar <i>Online</i> .....	85
Gambar 5.17 Pasien Baru.....	86

Gambar 5.18 Pasien Lama.....	87
Gambar 5.19 <i>Login</i> .....	87
Gambar 5.20 <i>Dashboard</i> .....	88
Gambar 5.21 Tampilan Data Dokter.....	89
Gambar 5.22 Tambah Data Poli.....	89
Gambar 5.23 Tambah Data Dokter.....	90
Gambar 5.24 Tampilan Data Rekam Medis (Rekam Medis).....	91
Gambar 5.25 Detail Rekam Medis (Rekam Medis dan Dokter).....	92
Gambar 5.26 Tambah Isi Rekam Medis (Dokter).....	93
Gambar 5.27 Tambah Obat Pasien.....	94
Gambar 5.28 Perbarui Obat Pasien.....	95
Gambar 5.29 Laporan Pasien.....	96
Gambar 5.30 Hasil Tampilan Daftar <i>Online</i> .....	98
Gambar 5.31 Hasil Pasien Baru.....	99
Gambar 5.32 Hasil Pasien Lama.....	100
Gambar 5.33 Hasil Nomor Antrian.....	101
Gambar 5.34 Hasil <i>Login User</i> .....	102
Gambar 5.35 Hasil <i>Dashboard</i> .....	103
Gambar 5.36 Hasil Pemanggilan Pasien.....	104
Gambar 5.37 Hasil Rekam Medis (Dokter).....	105
Gambar 5.38 Hasil Isi Rekam Medis (Dokter).....	106

Gambar 5.39 Hasil Tambah Obat Pasien.....	107
Gambar 5.40 Hasil Perbarui Obat Pasien.....	108
Gambar 5.41 Hasil Cetak Resep.....	108
Gambar 5.42 Lanjutan Hasil Cetak Resep.....	109
Gambar 5.43 Hasil Tambah Pembayaran Pasien.....	110
Gambar 5.44 Hasil Laporan Pasien.....	111

## DAFTAR TABEL

Tabel 3.1 Penelitian Terdahulu.....	13
Tabel 4.1 Jadwal Penelitian.....	19
Tabel 4.2 Simbol-Simbol <i>Flowchart</i> .....	22
Tabel 4.3 Simbol <i>Data Flow Diagram</i> .....	24
Tabel 4.4 Simbol <i>Entity Relationship Diagram</i> .....	26
Tabel 5.1 Tabel Data <i>User</i> .....	75
Tabel 5.2 Tabel Data Dokter.....	76
Tabel 5.3 Tabel Data Bayar.....	76
Tabel 5.4 Tabel Data Obat.....	77
Tabel 5.5 Tabel Data Pasien.....	78
Tabel 5.6 Tabel Data PBF.....	78
Tabel 5.7 Tabel Data Poli.....	79
Tabel 5.8 Tabel Jadwal Dokter.....	79
Tabel 5.9 Tabel Obat Keluar.....	80
Tabel 5.10 Tabel Obat Masuk.....	81
Tabel 5.11 Tabel Obat Pasien.....	82
Tabel 5.12 Tabel Pembayaran.....	82
Tabel 5.13 Tabel Pembayaran Pasien.....	83
Tabel 5.14 Tabel Rekam Medis Pasien.....	83

Tabel 5.15 Tabel Stok Obat.....	84
Tabel 5.16 Rencana <i>Test Case</i> dan Butir Uji.....	111
Tabel 5.17 Skenario Pasien Baru.....	112
Tabel 5.18 Skenario Pasien Lama.....	113
Tabel 5.19 Skenario Halaman <i>Login</i> .....	114
Tabel 5.20 Skenario Tambah Data Pengguna.....	115
Tabel 5.21 Skenario <i>Edit</i> Pengguna.....	116
Tabel 5.22 Skenario Tambah Bayar.....	117
Tabel 5.23 Skenario Pembayaran Pasien.....	118
Tabel 5.24 Skenario Tambah Pengobatan.....	118
Tabel 5.25 Hasil Pengujian Pasien Baru.....	119
Tabel 5.26 Hasil Pengujian Pasien Lama.....	121
Tabel 5.27 Hasil Pengujian <i>Login</i> .....	122
Tabel 5.28 Hasil Pengujian Tambah Data Pengguna.....	123
Tabel 5.29 Hasil Pengujian <i>Edit</i> Pengguna.....	125
Tabel 5.30 Hasil Pengujian Tambah Bayar.....	127
Tabel 5.31 Hasil Pengujian Pembayaran Pasien.....	128
Tabel 5.32 Hasil Pengujian Tambah Pengobatan.....	129

## DAFTAR LAMPIRAN

- 1....Lampiran 1. *Form* Topik dan Judul (*Fotocopy*)
- 2....Lampiran 2. Surat Balasan dari Perusahaan (*Fotocopy*)
- 3....Lampiran 3. *Form* Konsultasi (*Fotocopy*)
- 4....Lampiran 4. Surat Pernyataan (*Fotocopy*)
- 5....Lampiran 5. *Form* Revisi Ujian Pra Sidang (*Fotocopy*)
- 6....Lampiran 6. *Form* Revisi Ujian Kompre (Asli)
- 7....Lampiran 7. *Listing Code*

## ***ABSTRACT***

INDI FEBRINA LATIFAH SIDIK. *Management Information System for Website-Based Moulya Clinic Services.*

*The Moulya Clinic is an institution which provides basic medical care. Moulya Clinic is located at Jalan Peltu Tulus Yahya Lorong Kenanga No. 17A, 2 Ilir, Palembang City and has 3 general practitioners and 1 dentist with an open schedule from Monday to Saturday. Moulya Clinic still uses pen and paper for data collection, so it costs money every month for data collection. For example, the cost of printing a medical chart is about Rp 500,000. After printing, the medical record cards need to be stored properly. That's why Moulya Clinic provides a special room to store the medical record cards. From this example, Moulya Clinic needs a system that can assist in data management. Therefore, a web-based management information system for Moulya Clinic services is created. This system will also be used by administrators played by administration, clinic manager, pharmacy, medical records, patients and doctors and will be based on a website using PHP programming language and MySQL database. Flowchart, Data Flow Diagram (DFD) and Entity Relationship Diagram (ERD) are used for data modeling in building this system. As well as using the Prototype method and using Black Box system testing. The test results show the success rate of the system to be 97.22%.*

***Keywords : Information Systems, Prototype Method, Black-Box Testing***

## ABSTRAK

INDI FEBRINA LATIFAH SIDIK. Sistem Informasi *Management* Pada Layanan Klinik Moulya Berbasis *Website*.

Klinik Moulya merupakan layanan kesehatan yang menyediakan pelayanan medis dasar. Klinik Moulya ini berada di Jalan Peltu Tulus Yahya Lorong Kenanga No. 17A, 2 Ilir, Kota Palembang, dan memiliki 3 dokter umum dan 1 dokter gigi dengan jadwal buka dari hari Senin hingga hari Sabtu. Dalam melakukan pendataan Klinik Moulya masih menggunakan pena dan kertas sehingga memerlukan biaya setiap bulannya dalam melakukan pendataan. Salah satu contohnya adalah rekam medis, dalam sekali pencetakan kartu rekam medis memerlukan biaya kurang lebih Rp500.000. Setelah di cetakpun kartu rekam medis harus disimpan dengan baik, sehingga Klinik Moulya menyediakan satu ruangan khusus untuk menyimpan kartu rekam medis tersebut. Dari contoh tersebut Klinik Moulya memerlukan sistem yang dapat membantu dalam pengelolaan data sehingga terciptalah Sistem Informasi *Management* Pada Layanan Klinik Moulya Berbasis *Website*. Sistem ini dibuat berbasis *website* dan menggunakan bahasa pemrograman PHP serta basis data MySQL, dan sistem ini juga akan digunakan oleh admin diperankan Tata Usaha, Penanggung Jawab Klinik, Farmasi, Rekam Medis, Pasien dan Dokter. Pemodelan data yang digunakan dalam membangun sistem ini menggunakan *Flowchart* (Diagram Alir), *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD). Serta menggunakan metode *Prototype* dan menggunakan pengujian sistem *Black-Box*. Dari hasil pengujian yang telah dilakukan, diketahui tingkat presentase keberhasilan dari sistem yang telah dibuat adalah 97,22%.

**Kata Kunci :** Sistem Informasi, Metode *Prototype*, *Black-Box Testing*

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

*Web* merupakan bagian dari layanan yang dapat berjalan di atas teknologi internet. Halaman *web* dapat diakses dengan adanya teknologi *web server* sebagai penyedia halaman web, HTML sebagai bahasa baku dan HTTP sebagai jalur pengiriman dokumen *web* (Rochman *et al.*, 2020).

Klinik Moulya merupakan salah satu klinik yang memiliki fasilitas untuk melayani kesehatan perorangan dan menyediakan pelayanan medis dasar. Klinik Moulya ini berada di Jalan Peltu Tulus Yahya Lorong Kenanga No. 17A, 2 Ilir, Kota Palembang dan memiliki 3 dokter umum dan 1 dokter gigi dengan jadwal buka dari hari Senin hingga hari Sabtu.

Berdasarkan hasil wawancara yang peneliti lakukan dengan dr. Mohammad Hazem selaku penanggung jawab klinik dan merupakan salah satu dokter yang ada di Klinik Moulya, proses pencatatan rekam medis hingga pendataan obat masih dilakukan dengan menggunakan pena dan kertas. Dalam melakukan pencetakan kartu rekam medis diperlukan biaya sebesar kurang lebih Rp.500.000 untuk sekali pencetakan. Setelah di cetak, kartu tersebut harus disimpan dengan baik sehingga Klinik Moulya juga menyediakan satu ruangan khusus untuk menyimpan data pasien yang jumlah keseluruhannya kurang lebih ada 6.000 data. Kartu rekam medis yang sudah diisi oleh dokter juga harus disimpan dengan baik, agar tidak

terjadi hal-hal yang tidak diinginkan seperti hilang ataupun terselip dan dalam mencari kartu rekam medis juga memerlukan waktu yang lebih lama karena beberapa pasien memiliki nama yang sama. Jika kartu rekam medis tersebut tidak ada, klinik tidak memiliki arsip data dalam bentuk elektronik. Sehingga jika pasien memiliki riwayat alergi obat, dokter yang akan menangani pasien tersebut nantinya bisa saja memberikan obat yang dapat memicu alergi pasien.

Dengan masalah yang sudah dijabarkan di atas maka Klinik Moulya memerlukan suatu aplikasi yang dapat membantu klinik tersebut dalam melakukan pengelolaan data yang ada. Maka dari itu peneliti tertarik melakukan penelitian dengan judul **“Sistem Informasi Management Pada Layanan Klinik Moulya Berbasis Website”**.

## **1.2. Rumusan Masalah Penelitian**

Berdasarkan latar belakang yang telah peneliti paparkan di atas maka rumusan masalah yang diangkat dalam penelitian ini adalah “Bagaimana Klinik Moulya dapat memanfaatkan teknologi menggunakan komputer sehingga dapat melakukan penghematan biaya percetakan, dan tempat penyimpanan dalam mengelola datanya?”.

## **1.3. Ruang Lingkup Penelitian**

Dalam laporan penelitian ini, peneliti telah merangkum dan membahas tentang pembuatan sistem informasi *management* pada

layanan Klinik Moulya berbasis *website*. Berikut adalah ruang lingkup yang peneliti rangkum :

- a. Sistem informasi ini dibuat berbasis *website* dan menggunakan bahasa pemrograman PHP serta basis data MySQL.
- b. Sistem informasi klinik ini akan digunakan oleh admin yang diperankan Tata Usaha, Penanggung Jawab Klinik, Farmasi, Rekam Medis, Pasien dan Dokter.
- c. *Output* sistem informasi klinik berupa :
  - 1) Tata usaha dapat mengelola data pasien (menambahkan data pasien, menghapus data pasien, dan mengedit data pasien), mengelola data dokter (menambahkan data dokter baru, menghapus dokter, dan mengedit dokter), mengelola pengguna (menambahkan pengguna baru, menghapus pengguna, dan mengedit pengguna), mengatur jadwal dokter, dan data pembayaran.
  - 2) Penanggung jawab klinik dapat melihat laporan pasien, melihat laporan pembayaran, dan melihat laporan farmasi (laporan obat masuk, laporan obat keluar, laporan stok obat).
  - 3) Rekam medis dapat mengelola data rekam medis (menambahkan data rekam medis, mengisi data rekam medis, mengedit data rekam medis)

- 4) Farmasi dapat mengelola data obat masuk (menambahkan data obat masuk, menghapus data obat masuk, mengedit data obat masuk), melihat rekam medis pasien bagian obatnya saja, dan mengelola laporan farmasi (data obat masuk, data obat keluar).
  - 5) Pasien dapat melakukan daftar online baik pasien baru maupun pasien lama (pasien yang sudah terdaftar sebagai anggota di Klinik Moulya).
  - 6) Dokter dapat melihat rekam medis pasien, mengisi rekam medis pasien dan mengisi obat pasien.
- d. Pemodelan data : *Flowchart* (Diagram Alir), *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD).
  - e. Metode yang akan digunakan adalah Metode *Prototype*.

#### **1.4. Tujuan Penelitian**

Tujuan dari penelitian ini adalah membangun sebuah sistem informasi *management* pada layanan Klinik Moulya berbasis *website* agar dapat membantu Klinik Moulya dalam melakukan pengelolaan data baik data pasien, farmasi, hingga rekam medis, dan dapat mempermudah dalam melihat laporan.

## **1.5. Manfaat Penelitian**

### **1.5.1. Manfaat Bagi Peneliti**

Peneliti dapat menambah wawasan dan pengalaman untuk belajar tentang pengelolaan sistem informasi, serta dapat meningkatkan rasa tanggung jawab peneliti dalam membuat laporan penelitian maupun membuat sistem informasi *management* pada layanan Klinik Moulya berbasis *website*.

### **1.5.2. Manfaat Bagi Pihak Klinik Moulya**

Dengan adanya sistem informasi *management* pada layanan Klinik Moulya berbasis *website* ini, peneliti mengharapkan sistem ini bisa mempermudah tim medis dalam melakukan pengelolaan data dan dapat mempermudah dalam melihat laporan.

### **1.5.3. Manfaat Bagi Pihak Akademik**

Pihak akademik dapat mengetahui kemampuan mahasiswa/i nya dalam membuat laporan maupun sistem informasi, dan dapat dijadikan acuan referensi untuk membuat laporan dengan tema yang sama.

## **1.6. Sistematika Penulisan**

Sistematika penulisan menggunakan pembahasan yang sesuai dengan ketentuan yang telah diberikan, yaitu sebagai berikut :

**BAB I            PENDAHULUAN**

Bab ini menguraikan mengenai latar belakang, rumusan masalah, ruang lingkup, tujuan penelitian dan manfaat penelitian, serta sistematika penulisan dari laporan skripsi.

**BAB II            GAMBARAN UMUM**

Bab ini menguraikan mengenai gambaran umum perusahaan seperti profil Klinik Moulya, Struktur organisasi, serta tugas dan wewenang

**BAB III           TINJAUAN PUSTAKA**

Bab ini menguraikan tinjauan pustaka tentang teori-teori yang berdasarkan pembuatan skripsi, yang terdiri dari teori pendukung, hasil penelitian terdahulu yang berkaitan dengan penelitian

**BAB IV           METODE PENELITIAN**

Bab ini menguraikan metode penelitian, lokasi dan waktu penelitian, jenis data, teknik pengumpulan data dan alat perancangan sistem.

## **BAB V HASIL DAN PEMBAHASAN**

Bab ini menguraikan hasil pembahasan pembuatan sistem. Pada bab ini juga akan dibahas mengenai pengujian sistem.

## **BAB VI PENUTUP**

Bab akhir ini memuat beberapa saran dan kesimpulan dari pembahasan dalam menerapkan sistem ini.

## **BAB II**

### **GAMBARAN UMUM PERUSAHAAN**

#### **2.1. Profil Perusahaan**

##### **2.1.1. Sejarah Perusahaan**

Pada tanggal 20 Juli 2016, dr. Darmansyah Raya mendirikan sebuah fasilitas kesehatan sederhana. Pendirian fasilitas pelayanan kesehatan ini masih berbentuk praktek mandiri dokter yang sumber dananya bersumber dari modal perorangan milik pribadi.

Seiring berjalannya waktu dan kunjungan pasien ke praktek mandiri dr. Darmansyah Raya terus bertambah jumlahnya sehingga dr. Darmansyah Raya memiliki keinginan untuk memperluas pelayanan tersebut agar lebih banyak lagi pasien yang dapat dibantu olehnya. Akhirnya dr. Darmansyah Raya mendirikan sebuah klinik yang diberi nama sesuai dengan nama anak perempuan pertamanya, yaitu Klinik Moulya dengan menambahkan fasilitas pelayanan kesehatan gigi dan mulut.

#### **2.2. Visi dan Misi**

##### **2.2.1. Visi**

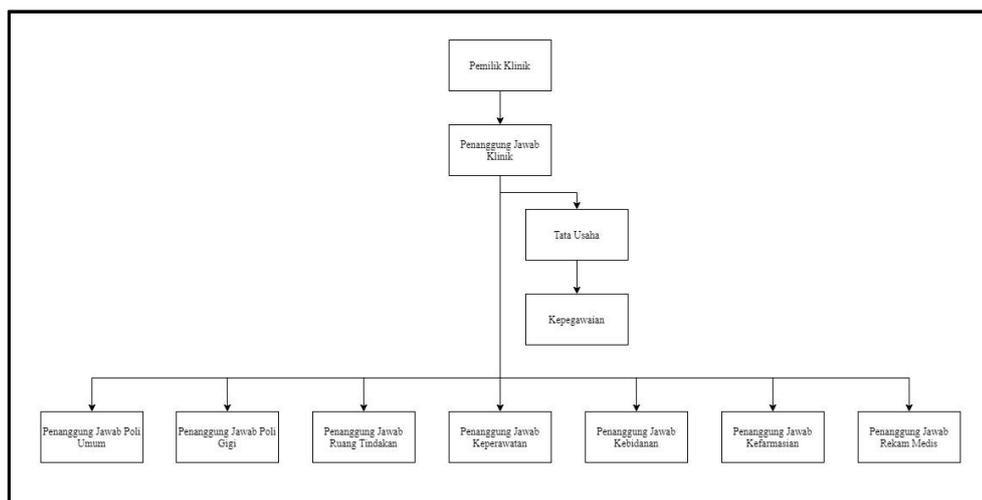
Menjadi klinik pilihan yang unggul dan terpercaya

### 2.2.2. Misi

1. Menjadi fasilitas kesehatan yang standar di bidang pelayanan primer;
2. Memberikan pelayanan kesehatan yang bermutu terjangkau dan mengutamakan keselamatan pasien;
3. Menyediakan sarana dan prasarana kesehatan sesuai dengan perkembangan ilmu pengetahuan dan teknologi;
4. Memiliki sumber daya manusia yang profesional dan kompeten;
5. Melaksanakan program pemerintah dalam peningkatan derajat kesehatan.

### 2.3. Struktur Organisasi

Klinik Moulya memiliki struktur organisasi yang dapat dilihat pada gambar 2.1.



Sumber : Klinik Moulya

**Gambar 2.1 Struktur Organisasi**

## 2.4. Tugas dan Wewenang

1. Penanggung Jawab Klinik
  - Mengkoordinasikan pengembangan klinik
  - Mengawasi dan mengevaluasi kinerja pegawai klinik dalam pelaksanaan tugas sehari-hari
  - Menyusun perencanaan kegiatan klinik
  - Mengawasi dan mengevaluasi kegiatan klinik yang berjalan
  - Mengevaluasi pendapatan dan pengeluaran klinik
2. Tata Usaha
  - Bertanggung jawab terhadap data pribadi pasien
  - Bertanggung jawab kepada pasien yang melakukan administrasi
  - Bertanggung jawab terhadap surat masuk dan surat keluar
  - Bertanggung jawab terhadap data diri pasien
  - Memberikan informasi yang akurat kepada pasien jika sedang ada kegiatan klinik
3. Penanggung Jawab Rekam Medis
  - Bertanggung jawab terhadap data rekam medis pasien
  - Bertanggung jawab terhadap informasi yang ada pada bagian rekam medis pasien
4. Penanggung Jawab Kefarmasian
  - Bertanggung jawab terhadap obat-obatan yang ada
  - Mengecek ketersediaan stok obat

- Bertanggung jawab terhadap obat masuk dan obat keluar
- Bertanggung jawab terhadap nota-nota pembelian obat
- Bertanggung jawab terhadap resep obat
- Membuat pembukuan obat masuk, obat keluar, stok obat.

## **BAB III**

### **TINJAUAN PUSTAKA**

#### **3.1. Teori Pendukung**

##### **3.1.1. Database**

*Database* atau basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut (Andaru, 2018).

##### **3.1.2. PHP**

PHP adalah sebuah bahasa pemrograman yang perintahnya dilaksanakan *server* dan kemudian hasilnya ditampilkan pada komputer *client* (Wijaya *et al.*, 2022).

##### **3.1.3. Web**

*Web* merupakan bagian dari layanan yang dapat berjalan di atas teknologi internet. Halaman *web* dapat diakses dengan adanya teknologi *web server* sebagai penyedia halaman web, HTML sebagai bahasa baku dan HTTP sebagai jalur pengiriman dokumen *web* (Rochman *et al.*, 2020).

### 3.1.4. Sistem

Sistem adalah seperangkat komponen yang saling berhubungan dan saling bekerja sama untuk mencapai beberapa tujuan (Agustin, 2018).

### 3.1.5. Informasi

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang (Agustin, 2018).

## 3.2. Penelitian Terdahulu

Sebagai bahan pertimbangan dalam penelitian ini akan dicantumkan beberapa hasil penelitian terdahulu, berikut penelitian terdahulu yang peneliti gunakan sebagai inspirasi dapat dilihat pada tabel 3.1.

**Tabel 3.1 Penelitian Terdahulu**

No.	Judul Penelitian	Penulis/Tahun	Hasil
1.	Sistem Informasi Pengolahan Data Pasien pada Klinik Essiva Berbasis Web dengan Metode	Murni Nancy Margareta S, Surya Hendra Putra (2022)	- Pengujian sistem yang telah diuji telah menghasilkan sistem pengelolaan yang efektif dan efisien.  - Aplikasi ini dibangun dengan menggunakan bahasa

No.	Judul Penelitian	Penulis/Tahun	Hasil
	Prototype		<p>pemrograman PHP dan database Mysql</p> <ul style="list-style-type: none"> <li>- Sistem ini dapat memberikan informasi kepada masyarakat luas serta mempermudah admin dalam proses pendataan pasien, pemeriksaan, transaksi, dan pembuatan laporan.</li> </ul>
2.	Sistem Informasi Layanan Pada Klinik Dokter Rosdiana Berbasis Web	Halimah dan Amnah (2021)	<ul style="list-style-type: none"> <li>- Admin/user mudah dalam mengelola data pasie, pendaftaran, pembayaran, jadwal petugas pada klinik</li> <li>- Diharapkan mampu meningkatkan kinerja petugas klinik dalam mengelola data yang ada pada Klinik Dokter Rosdiana</li> <li>- Memudahkan pasien untuk melakukan pendaftaran dan berobat di klinik.</li> </ul>

No.	Judul Penelitian	Penulis/Tahun	Hasil
3.	Penerapan Metode Prototype Pada Pengembangan Sistem Informasi Tugas Akhir Mahasiswa	Nurul Renaningtias dan Dyah Apriliani (2021)	<ul style="list-style-type: none"> <li>- Pengembangan sistem informasi ini memberikan banyak fitur baru yang lebih lengkap dari sebelumnya</li> <li>- Sistem ini mampu memberikan kemudahan kepada mahasiswa, admin, dan dosen dalam pengerjaan tugas akhir</li> <li>- Sistem ini mempermudah admin dalam memberikan informasi terkait dengan pengerjaan tugas akhir oleh mahasiswa.</li> </ul>
4.	Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype	Rizky Aditya, Viktor Handrianus Pranatawijaya, Putu Bagus Adidyana Anugrah Putra (2021)	<ul style="list-style-type: none"> <li>- Sebagai media membuat map monitoring yang ada pada aplikasi</li> <li>- Sistem ini dirancang dengan memanfaatkan Google Map API untuk alat sistem yang berfungsi sebagai media pembuat map monitoring</li> </ul>

No.	Judul Penelitian	Penulis/Tahun	Hasil
			<p>yang ada pada aplikasi</p> <ul style="list-style-type: none"> <li>- Aplikasi ini berhasil membuat fitur-fitur seperti fitur login pengguna, fitur beranda, fitur tambah kelompok, fitur map monitoring, fitur absensi kelompok, fitur daftar hadir anggota, dan fitur profil pengguna.</li> </ul>
5.	<p>Penerapan Metode Prototype Pada Perancangan Sistem Pengarsipan Dokumen Kantor Kecamatan Lais</p>	<p>Kurniati (2021)</p>	<ul style="list-style-type: none"> <li>- Sistem pengarsipan dokumen mampu memberikan kemudahan kepada petugas dan admin kantor Kecamatan Lais dalam mengelola dokumen arsip dengan baik serta lebih mudah dalam proses pencarian dan keamanan data lebih terjamin</li> <li>- Berdasarkan hasil uji blackbox semua menu yang ada pada sistem berfungsi</li> </ul>

No.	Judul Penelitian	Penulis/Tahun	Hasil
			<p>dengan baik sesuai dengan fungsionalitas menu tersebut</p> <ul style="list-style-type: none"><li>- Berdasarkan hasil pengujian usability terhadap 30 responden menyatakan bahwa secara keseluruhan sistem dapat diterima pengguna dengan sangat baik, melalui indikasi rata-rata persentase 89% pada pernyataan responden yang meliputi fungsi, menu, mudah dimengerti, mudah untuk digunakan, tampilan yang menarik, membantu pengolahan data, dan dapat mempercepat pekerjaan</li></ul>

## **BAB IV**

### **METODOLOGI PENELITIAN**

#### **4.1. Lokasi dan Jadwal Penelitian**

##### **4.1.1. Lokasi**

Penelitian ini dilakukan di Klinik Moulya yang berada di Jalan Peltu Tulus Yahya Lorong Kenanga No. 17A, 2 Ilir, Kota Palembang. Klinik Moulya ini buka dari hari Senin hingga Sabtu dengan jadwal buka sebagai berikut :

→ Senin – Sabtu :

- Pagi : 08.00-12.00
- Sore : 16.00-18.00

→ Jum'at :

- Pagi : 08.00-11.00
- Sore : 16.00-18.00

##### **4.1.2. Jadwal Penelitian**

Adapun jadwal yang sudah peneliti rangkum dan akan digunakan sebagai acuan dalam pengerjaan sistem informasi klinik ini dapat dilihat pada tabel 4.1.

**Tabel 4.1 Jadwal Penelitian**

No	Kegiatan	Tahun 2023																				
		Maret				April				Mei				Juni				Juli				
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
1	Pengumpulan Kebutuhan																					
	Wawancara	■	■	■	■																	
	Observasi	■	■	■	■																	
	Dokumentasi			■	■																	
	Studi Pustaka	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
2	Membangun Prototyping														■							
3	Evaluasi Prototyping														■	■						
4	Mengkodekan Sistem															■	■	■	■	■		
5	Menguji Sistem																	■	■	■		
6	Evaluasi Sistem																		■	■	■	
7	Penggunaan Sistem																					■

Sumber : Data diolah oleh peneliti (2023)

## 4.2. Jenis Data

### 4.2.1. Data Primer

Data primer yaitu data yang dikumpulkan oleh peneliti langsung dari lapangan. Untuk memperoleh data primer maka penulis langsung datang kesumbernya atau diperoleh dengan menggunakan metode wawancara (Indrasari, 2020).

### 4.2.2. Data Sekunder

Data sekunder adalah data yang bukan diusahakan sendiri pengumpulannya oleh peneliti, dan data sekunder biasanya terwujud data dokumentasi yang berupa data yang diperoleh dari

sumber tidak langsung seperti dari majalah, keterangan-keterangan atau publikasi lainnya (Indrasari, 2020).

### **4.3. Teknik Pengumpulan Data**

#### **4.3.1. Wawancara**

Wawancara adalah merupakan pertemuan dua orang untuk bertukar informasi dan ide melalui tanya jawab, sehingga dapat dikonstruksikan makna dalam suatu topik tertentu (Kurniawan, 2020).

Wawancara ini melibatkan antara peneliti dan narasumber yaitu dr. Mohammad Hazem selaku penanggung jawab klinik yang dilakukan di Klinik Moulya pada tanggal 31 Maret 2023.

#### **4.3.2. Observasi**

Observasi adalah teknik pengumpulan data atau informasi dengan cara pengamatan terhadap kegiatan yang sedang berlangsung (Mahardini, 2020).

Observasi ini peneliti lakukan dengan melihat alur kerja pegawai di Klinik Moulya dalam melakukan kegiatan sehari-hari.

#### **4.3.3. Dokumentasi**

Dokumentasi adalah salah satu teknik pengumpulan data melalui dokumen atau catatan-catatan tertulis yang ada (Ulfah, 2022 dalam Tanjung *et al.*, 2022).

Peneliti mengumpulkan data-data dari dokumen yang diperoleh secara langsung di Klinik Moulya dalam bentuk *file* seperti kartu rekam medis, data pasien, dll.

#### **4.3.4. Studi Pustaka**

Studi Pustaka adalah mencari referensi teori yang relevan dengan kasus atau permasalahan yang ditemukan (Nisaunnajah, 2021).

Peneliti juga membaca jurnal-jurnal terdahulu yang berkaitan dengan judul peneliti, guna untuk mendapatkan informasi serta menjadi referensi yang peneliti butuhkan dalam membuat laporan ini.

### **4.4. Alat dan Teknik Pengembangan Sistem**

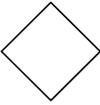
#### **4.4.1. Alat Pengembangan Sistem**

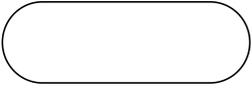
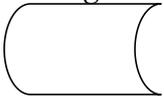
##### **4.4.1.1. *Flowchart***

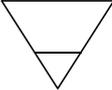
*Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya (Fauzi, 2020).

Menurut Fauzi (2020) simbol-simbol *Flowchart* dapat dilihat pada tabel 4.2.

Tabel 4.2 Simbol-Simbol *Flowchart*

No.	Simbol	Keterangan
1.	<p><i>Input/Output</i></p> 	Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya
2.	<p><i>Processing Symbol</i></p> 	Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer
3.	<p><i>Connector Symbol</i></p> 	Yaitu simbol untuk keluar-masuk atau penyambungan proses dalam lembar /halaman yang sama
4.	<p><i>Flow Direction Symbol</i></p> 	Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga <i>connecting line</i>
5.	<p>Simbol <i>Decision</i></p> 	Simbol pemilihan proses berdasarkan kondisi yang ada
6.	<p><i>Predefined Process</i></p> 	Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure

No.	Simbol	Keterangan
7.	<p><i>Preparation</i></p> 	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam <i>storage</i>
8.	<p><i>Terminator Symbol</i></p> 	Yaitu simbol untuk permulaan ( <i>Start</i> ) atau akhir ( <i>Stop</i> ) dari suatu kegiatan
9.	<p>Dokumen</p> 	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas
10.	<p>Simbol <i>disk and On-line storage</i></p> 	Simbol yang menyatakan <i>input</i> yang berasal dari <i>disk</i> atau disimpan ke <i>disk</i>
11.	<p>Manual Input</p> 	Simbol untuk pemasukan data secara manual <i>on-line keyboard</i>
12.	<p><i>Display</i></p> 	Simbol yang menyatakan peralatan <i>output</i> yang digunakan yaitu layar, <i>plotter</i> , <i>printer</i> dan sebagainya
13.	<p><i>Manual Operation</i></p> 	Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer

No.	Simbol	Keterangan
14.	<i>Off-line Storage</i> 	Penyimpanan yang tidak dapat diakses oleh komputer secara langsung

Sumber : Fauzi (2020)

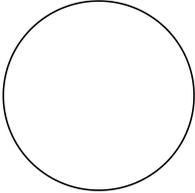
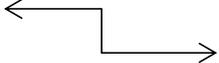
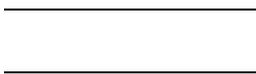
#### 4.4.1.2. *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*) (Harsuda, 2018).

Menurut Harsuda (2018) simbol-simbol *data flow diagram* menggunakan teknik Yourdon and De Marco dapat dilihat pada tabel 4.3.

**Tabel 4.3 Simbol *Data Flow Diagram***

No.	Simbol	Keterangan
1.		Entitas luar ( <i>external entity</i> ) atau masukan ( <i>input</i> ) atau keluaran ( <i>output</i> ) atau biasa disebut juga orang yang memakai/berinteraksi dengan perangkat lunak.  Nama yang digunakan untuk <i>external entity</i>

No.	Simbol	Keterangan
		biasanya berupa kata benda
2.		<p>Proses atau fungsi atau prosedur pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi ilmiah yang harusnya menjadi fungsi atau prosedur didalam kode program.</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
3.		<p>Aliran data merupakan data yang dikirim antar proses, dari penyimpanan ke proses atau dari proses ke <i>external entity</i>.</p> <p>Nama yang digunakan pada aliran data biasanya berupa kata benda, seperti “data siswa”</p>
4.		<p>Penyimpanan (<i>storage</i>) merupakan tempat menampung <i>file-file</i> yang telah diproses sebelumnya.</p> <p>Nama untuk <i>storage</i> biasanya berupa kata benda, seperti “file siswa”</p>

Sumber : Harsuda (2018)

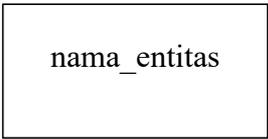
#### 4.4.1.3. *Entity Relationship Diagram (ERD)*

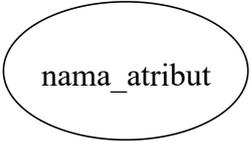
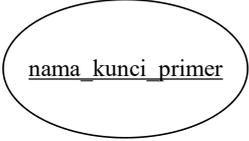
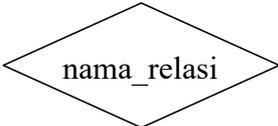
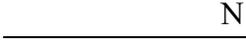
*Entity Relationship Diagram* atau ERD adalah sebuah diagram struktural yang digunakan untuk merancang sebuah database. Sebuah ERD mendeskripsikan data yang akan disimpan dalam sebuah sistem maupun batasannya (Latukolan *et al.*, 2019).

ERD (*Entity Relationship Diagram*) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model (Ropianto dan Indra, 2020).

Menurut Ropianto dan Indra (2020) simbol-simbol *entity relationship diagram* dapat dilihat pada tabel 4.4.

**Tabel 4.4 Simbol *Entity Relationship Diagram***

No.	Simbol	Keterangan
1.	<p><i>Entitas/entity</i></p> 	<p>Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer, penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.</p>

No.	Simbol	Keterangan
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan, biasanya berupa <i>id</i> . Kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.
5.	Asosiasi/ <i>association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multipicity</i> kemungkinan jumlah pemakaian, kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas.

Sumber : Ropianto dan Indra (2020)

Menurut Ropianto dan Indra (2020) ada beberapa kardinalitas relasi yang ada pada *Entity Relationship Diagram* (ERD) adalah sebagai berikut :

a. Satu ke satu (*One to One*)

Setiap elemen dari entitas A berhubungan paling banyak satu dengan elemen pada entitas B. demikian juga sebaliknya setiap elemen B berhubungan paling banyak satu elemen pada entitas A.

b. Satu ke banyak (*One to Many*)

Setiap elemen dari entitas A berhubungan dengan maksimal banyak elemen pada entitas B. dan sebaliknya setiap elemen dari entitas B berhubungan dengan paling banyak satu elemen di entitas A.

c. Banyak ke satu (*Many to One*)

Setiap elemen dari entitas A berhubungan paling banyak dengan satu elemen pada entitas B. dan sebaliknya setiap elemen dari entitas B berhubungan dengan maksimal banyak elemen di entitas A.

d. Banyak ke banyak (*Many to Many*)

Setiap elemen dari entitas A berhubungan maksimal banyak elemen pada entitas B demikian sebaliknya.

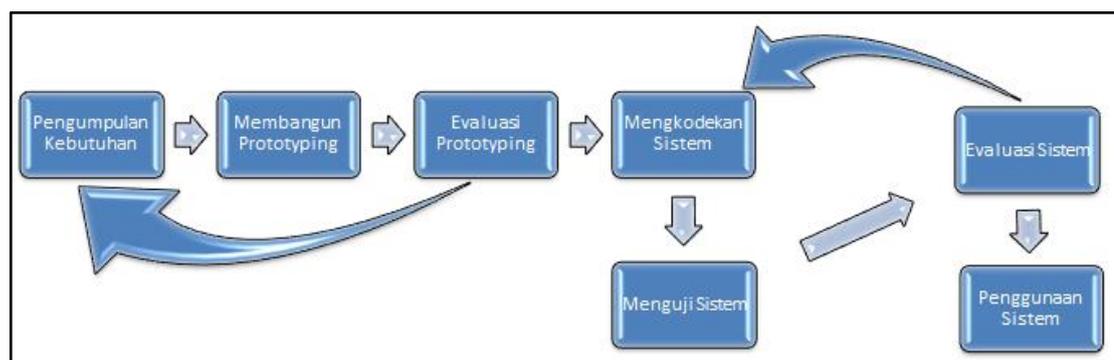
## 4.4.2. Teknik Pengembangan Sistem

### 4.4.2.1. Metode *Prototype*

*Prototyping* model adalah proses pengembangan perangkat lunak yang diawali dengan pengumpulan kebutuhan-kebutuhan dari sistem, yang dilanjutkan dengan pembuatan *prototipe* dan evaluasi dari pengguna (Meilani, 2019).

Metode *Prototyping* adalah metode untuk membangun sebuah perangkat lunak yang tidak terlalu besar dan sumber manusia yang terlibat dalam jumlah terbatas (Oktaviani *et al*, 2020).

Menurut Oktaviani *et al* (2020) metode *Prototype* memiliki beberapa tahapan yang dapat digambarkan seperti pada gambar 4.1.



Sumber : Oktaviani *et al* (2020)

**Gambar 4.1 Tahapan Dalam Metode *Prototype***

Menurut Oktaviani *et al* (2020) ada beberapa tahapan pada metode *prototype* ini ialah sebagai berikut :

### 1. Pengumpulan Kebutuhan

Klien dan pengembang bersama-sama mendefinisikan format *software/perangkat lunak*, mengidentifikasi kebutuhan dan sistem yang dibuat.

### 2. Membangun *Prototyping*

Membangun *prototyping* dengan membuat perancangan sementara yang berfokus penyajian kepada pelanggan (contoh membuat *input* dan format *output*).

### 3. Evaluasi *Prototyping*

Tahap ini dilakukan oleh pelanggan / klien, apakah *prototyping* yang dibuat / dibangun, sudah sesuai dengan keinginan dan kebutuhan pelanggan atau belum. Jika tidak sesuai, *prototyping* akan direvisi dengan mengulangi langkah-langkah sebelumnya. Tapi jika sudah sesuai, maka langkah selanjutnya akan dilaksanakan.

### 4. Mengkodekan Sistem

Di tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

## 5. Menguji Sistem

Setelah sistem sudah menjadi suatu *software* yang siap pakai, maka *software* harus di tes dahulu sebelum digunakan. Hal ini bertujuan untuk meminimalisir kesalahan *software* tersebut. Pengujian dilakukan dengan *Black Box*, *White box*, Pengujian *arsitektur*, *Basis path* dan lain-lain.

## 6. Evaluasi Sistem

Di tahap ini pelanggan / klien mengevaluasi sistem yang sudah dibuat sudah sesuai yang diinginkan. Jika tidak, maka pengembang akan mengulangi langkah ke 4 dan 5. Tapi jika iya, maka langkah ke 7 akan dilakukan.

## 7. Penggunaan Sistem

Prangkat Lunak / *Software* yang telah diuji dan diterima klien / pelanggan siap digunakan.

### **4.5. Alat dan Teknik Pengujian**

Pengujian yang dilakukan peneliti bertujuan untuk menemukan dan mengatasi error pada aplikasi yang dirancang. Peneliti menggunakan *black-box testing* sebagai teknik pengujian.

#### 4.5.1. *Black-Box Testing*

Pada tahap pengujian, peneliti akan menggunakan metode *Black-box Testing*. *Black-Box Testing* merupakan Teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak (Jaya, 2018). *Black-box Testing* memiliki keuntungan dan kerugian sebagai berikut :

1. Keuntungan *Black-box Testing* :
  - a. Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu
  - b. Pengujian dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan
  - c. *Programmer* dan tester keduanya saling bergantung satu sama lain
2. Kekurangan *Black-box Testing* :
  - a. Uji kasus sulit disain tanpa spesifikasi yang jelas
  - b. Kemungkinan memiliki pengulangan tes yang sudah dilakukan oleh *programmer*
  - c. Beberapa bagian *back end* tidak diuji sama sekali

*Black-box Testing* juga memiliki beberapa macam jenis teknik untuk melakukan pengujiannya. Pada penelitian ini teknik yang akan digunakan oleh peneliti adalah Teknik *Equivalence Partitioning*. Teknik *Equivalence Partitioning* yaitu dengan cara

melakukan partition atau pembagian menjadi beberapa partisi dari input data (Ichsanudin *et al*, 2022).

## **BAB V**

### **HASIL DAN PEMBAHASAN**

#### **5.1. Hasil**

Berdasarkan dari pengamatan yang telah peneliti lakukan di Klinik Moulya yang mengacu dengan tahapan-tahapan pada metode *prototype* dalam pembuatan Sistem Informasi *Management* Pada Layanan Klinik Moulya Berbasis *Website*.

##### **5.1.1. Pengumpulan Kebutuhan**

Pada tahapan ini peneliti mengumpulkan data-data secara lengkap dengan melakukan wawancara, dokumentasi, observasi, dan studi pustaka yang kemudian dianalisis guna mengidentifikasi kebutuhan Sistem Informasi *Management* Pada Layanan Klinik Moulya Berbasis *Website*.

Berdasarkan hasil pengumpulan data yang telah dianalisis, maka Klinik Moulya membutuhkan data pasien, jadwal dokter, info dokter, pembayaran, data rekam medis, data obat, daftar online, laporan pasien, laporan pembayaran, laporan farmasi.

##### **5.1.2. Membangun *Prototyping***

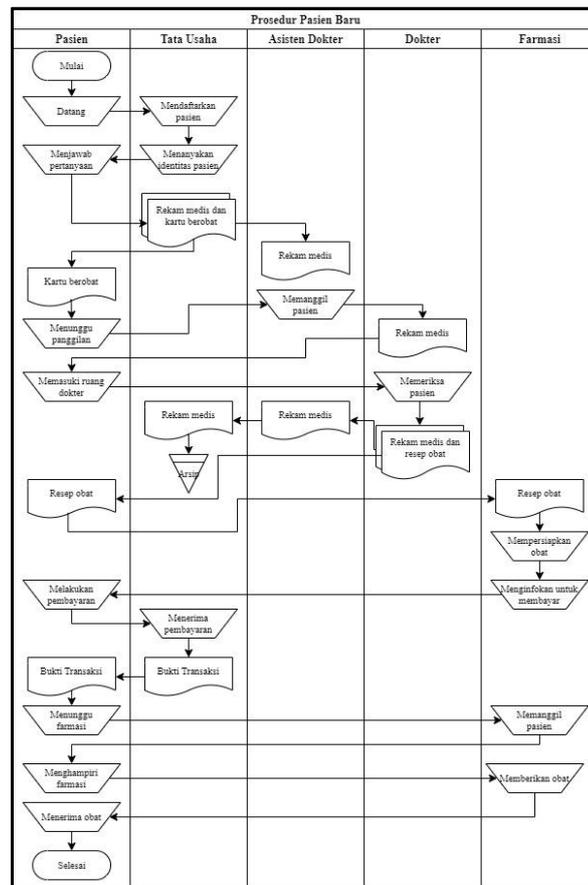
Pada tahapan ini peneliti melakukan tahapan membangun *prototyping* terhadap aplikasi yang dirancang.

### 5.1.2.1. *Flowchart* yang Berjalan

Prosedur yang berjalan masih banyak menggunakan pena dan kertas. Maka peneliti dapat membuat *flowchart* dari alur yang berjalan seperti berikut :

#### 5.1.2.1.1. **Prosedur Berjalan Pasien Baru**

Pada prosedur berjalan untuk bagian pasien baru yang dimulai dari pendaftaran hingga penerimaan obat, peneliti telah membuat *flowchart* yang dapat dilihat pada gambar 5.1.



**Gambar 5.1** Prosedur Berjalan Pasien Baru

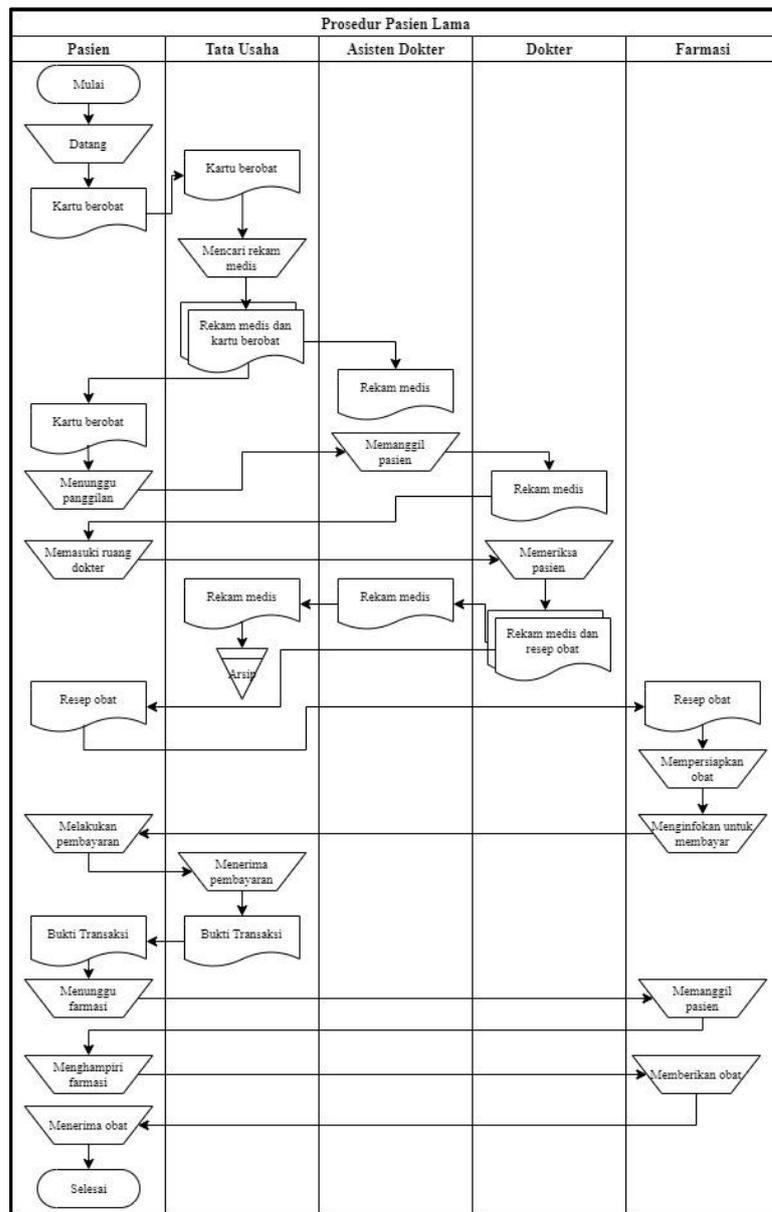
Dari gambar 5.1 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. Pasien datang ke Klinik Moulya
2. Tata usaha mendaftarkan pasien
3. Tata usaha menanyakan identitas pasien berupa nama, alamat, nomor telepon, dll
4. Pasien menjawab pertanyaan dari tata usaha untuk melengkapi identitas
5. Tata usaha membuatkan rekam medis serta kartu berobat pasien
6. Tata usaha memberikan rekam medis ke asisten dokter dan kartu berobat di berikan ke pasien lalu pasien di persilahkan untuk mengantri terlebih dahulu
7. Pasien menunggu panggilan
8. Asisten dokter memanggil pasien
9. Asisten dokter memberikan rekam medis pasien ke dokter
10. Pasien memasuki ruangan dokter
11. Dokter memeriksa pasien
12. Dokter menuliskan resep obat dan mengisi kartu rekam medis pasien

13. Dokter memberikan rekam medis ke asisten dokter lalu asisten dokter memberikan rekam medis tersebut ke tata usaha untuk disimpan ke ruang rekam medis dan resep obat di berikan ke pasien
14. Pasien menyerahkan resep obat ke farmasi
15. Farmasi menerima resep obat dari pasien
16. Farmasi mempersiapkan obat untuk pasien
17. Farmasi memberikan informasi untuk melakukan pembayaran ke tata usaha
18. Pasien melakukan pembayaran
19. Tata usaha menerima pembayaran
20. Tata usaha memberikan bukti pembayaran lalu memberitahukan kepada pasien untuk menunggu
21. Pasien menerima bukti pembayaran
22. Pasien menunggu panggilan farmasi
23. Farmasi memanggil pasien
24. Pasien menghampiri farmasi
25. Farmasi memberikan obat
26. Pasien menerima obat dari farmasi
27. Selesai.

**5.1.2.1.2. Prosedur Berjalan Pasien Lama**

Pada prosedur berjalan untuk bagian pasien lama yang dimulai dari pendataan hingga penerimaan obat peneliti telah membuat *flowchart* yang dapat dilihat pada gambar 5.2.



**Gambar 5.2 Prosedur Berjalan Pasien Lama**

Dari gambar 5.2 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. Pasien datang ke Klinik Moulya
2. Pasien memberikan kartu berobat
3. Tata usaha Menerima kartu berobat yang diberikan oleh pasien
4. Tata usaha mencari rekam medis yang sesuai dengan nomor rekam medis pasien ataupun nama pasien sesuai dengan data yang tertera pada kartu berobat
5. Tata usaha memberikan rekam medis ke asisten dokter dan kartu berobat di berikan kembali kepada pasien, lalu pasien di persilahkan untuk melakukan antri terlebih dahulu
6. Pasien menunggu panggilan dari asisten dokter di ruang tunggu
7. Asisten dokter memanggil pasien yang sedang melakukan antri di ruang tunggu
8. Asisten dokter memberikan rekam medis pasien ke dokter
9. Pasien memasuki ruangan dokter

10. Dokter memeriksa pasien sesuai dengan keluhan yang diberikan oleh pasien saat dokter menanyai pasien tersebut
11. Dokter menuliskan resep obat dan mengisi kartu rekam medis pasien
12. Dokter memberikan rekam medis ke asisten dokter lalu asisten dokter memberikan rekam medis tersebut ke tata usaha untuk disimpan ke ruang rekam medis dan resep obat di berikan ke pasien
13. Pasien menyerahkan resep obat ke farmasi
14. Farmasi menerima resep obat dari pasien
15. Farmasi mempersiapkan obat untuk pasien
16. Farmasi memberikan informasi untuk melakukan pembayaran ke tata usaha
17. Pasien melakukan pembayaran
18. Tata usaha menerima pembayaran
19. Tata usaha memberikan bukti pembayaran lalu memberitahukan kepada pasien untuk menunggu
20. Pasien menerima bukti pembayaran
21. Pasien menunggu panggilan farmasi
22. Farmasi memanggil pasien

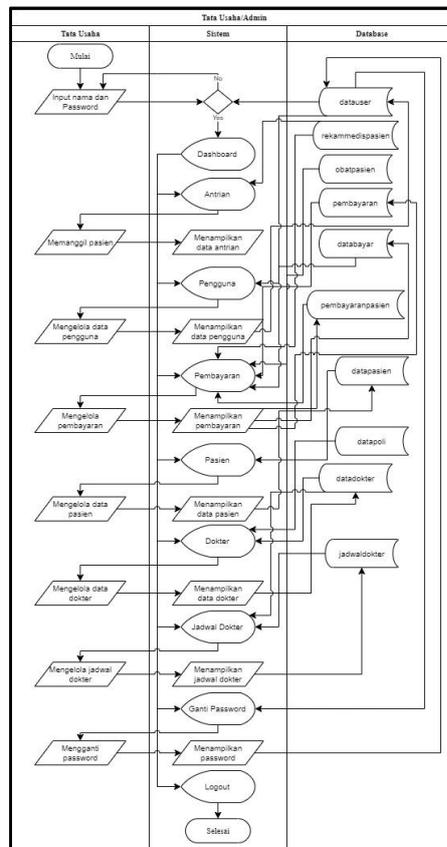
23. Pasien menghampiri farmasi
24. Farmasi memberikan obat
25. Pasien menerima obat dari farmasi
26. Selesai.

### 5.1.2.2. Flowchart yang Diusulkan

#### 5.1.2.2.1. Tata Usaha/Admin

*Flowchart* sistem diusulkan untuk bagian tata usaha/admin yang dimulai dari melakukan *login* hingga melakukan *logout*, dapat dilihat pada gambar

5.3.



**Gambar 5.3** Flowchart yang Diusulkan Tata Usaha

Dari gambar 5.3 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. *User* memasukkan nama dan *password* yang sudah terdaftar sebagai tata usaha pada *form login*.
2. Sistem akan melakukan validasi terhadap *inputan* dari tabel *datauser* yang ada di *database*, jika iya maka akan berpindah ke tampilan *dashboard*, jika tidak maka akan kembali menampilkan *form login*.
3. Tata Usaha memilih menu antrian untuk memanggil pasien. Pada menu ini, sistem akan menampilkan data-data nama pasien, poli yang dituju dan nomor urutan antri, pemanggilan pasien dapat menggunakan pemanggilan biasa dengan memanggil sesuai nomor antrian yang tertera di data. Pengambilan data antrian pasien tersebut diambil dari dalam *database* pada tabel *rekammedispasien*.
4. Tata usaha memilih menu pengguna untuk mengelola data pengguna yang kemudian datanya akan disimpan ke dalam *database datauser* lalu sistem akan menampilkan data

pengguna yang datanya diambil dari dalam *database datauser*.

5. Tata usaha memilih menu pembayaran untuk mengelola data pembayaran yang kemudian datanya akan disimpan ke dalam *database* pembayaran, *databayar*, dan *pembayaranpasien* lalu sistem akan menampilkan data pembayaran yang datanya diambil dari dalam *database* pembayaran, *databayar*, *pembayaranpasien*, *rekammedispasien* dan *obatpasien*.
6. Tata usaha memilih menu pasien untuk mengelola data pasien seperti menambahkan data pasien baru, menghapus data pasien lama, ataupun melakukan edit data pada data pasien yang sudah terdata, kemudian data tersebut akan disimpan ke dalam *database* *datapatient* lalu sistem akan menampilkan data pasien yang datanya diambil dari dalam *database* *datapatient*.
7. Tata usaha memilih menu dokter untuk mengelola data dokter yang kemudian datanya akan disimpan ke dalam *database* *datadoctor* lalu sistem akan menampilkan data dokter yang

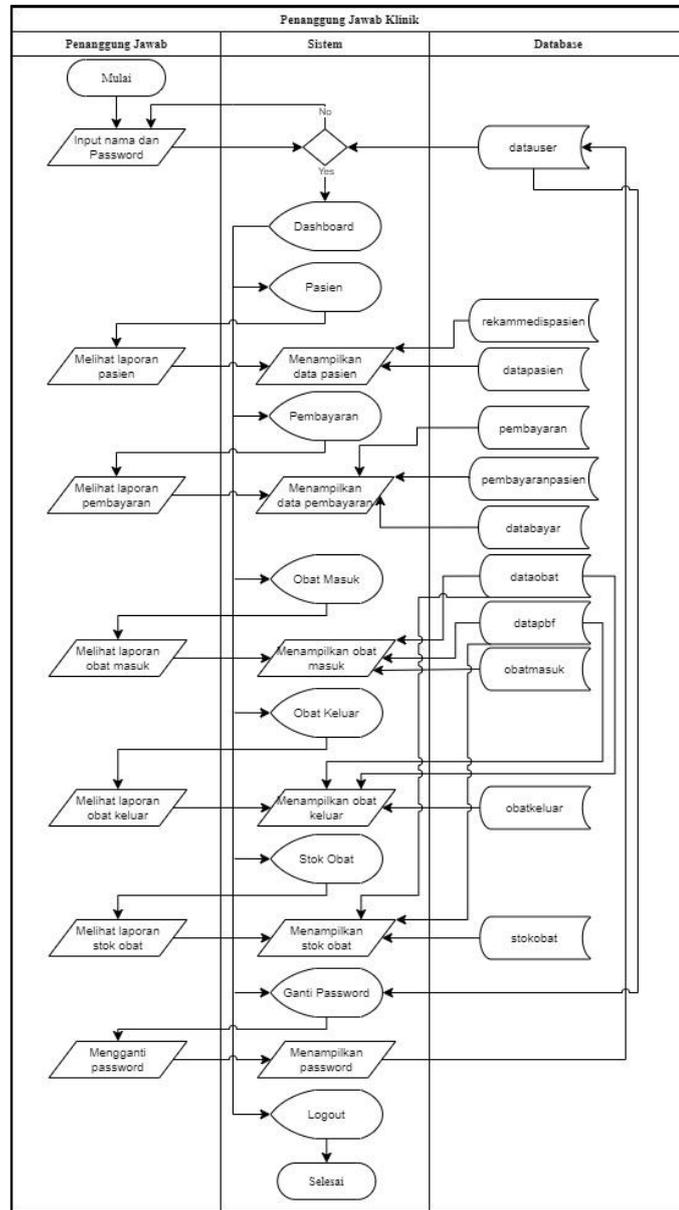
datanya diambil dari dalam *database* datadokter dan datapoli.

8. Tata usaha memilih menu jadwal dokter untuk mengelola jadwal dokter. Penggantian jadwal ini berguna untuk mengganti jadwal dokter yang ada. Karena di Klinik Moulya sendiri, dokter memiliki jadwal yang berbeda-beda setiap minggunya. Data yang sudah ada kemudian akan disimpan ke dalam *database* pada tabel *jadwaldokter*, lalu sistem akan menampilkan jadwal dokter yang datanya diambil dari dalam *database* datadokter dan *jadwaldokter*.
9. Tata Usaha memilih menu ganti *password* untuk mengganti kata sandi yang kemudian datanya akan disimpan ke dalam *database* *datauser* lalu sistem akan menampilkan ganti *password* yang datanya diambil dari dalam *database* *datauser*.
10. Tata usaha memilih menu *logout*.
11. Selesai.

#### **5.1.2.2.2. Penanggung Jawab Klinik**

*Flowchart* sistem diusulkan untuk bagian penanggung jawab klinik yang dimulai dari

melakukan *login* hingga melakukan *logout*, dapat dilihat pada gambar 5.4.



**Gambar 5.4 Flowchart yang Diusulkan Penanggung Jawab**

Dari gambar 5.4 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. *User* memasukkan nama dan *password* yang sudah terdaftar sebagai penanggung jawab.
2. Sistem akan melakukan validasi terhadap *inputan* dari tabel *datauser* yang ada di *database*, jika iya maka akan berpindah ke tampilan *dashboard*, jika tidak maka akan kembali menampilkan *form login*.
3. Penanggung jawab memilih menu pasien untuk melihat data pasien. Pada menu ini terdapat grafik yang bisa dijadikan acuan kegiatan klinik untuk kedepannya. Kemudian datanya diambil dari dalam *database* rekammedispasien dan *datap pasien*.
4. Penanggung jawab memilih menu pembayaran untuk melihat data pembayaran yang kemudian datanya diambil dari dalam *database* rekammedispasien, *datap pasien* dan *pembayaran pasien*.
5. Penanggung jawab memilih menu obat masuk untuk melihat data obat masuk. Pada menu ini terdapat grafik yang bisa dijadikan acuan kegiatan klinik untuk kedepannya. Kemudian

datanya diambil dari dalam *database* dataobat, datapbf, dan obatmasuk.

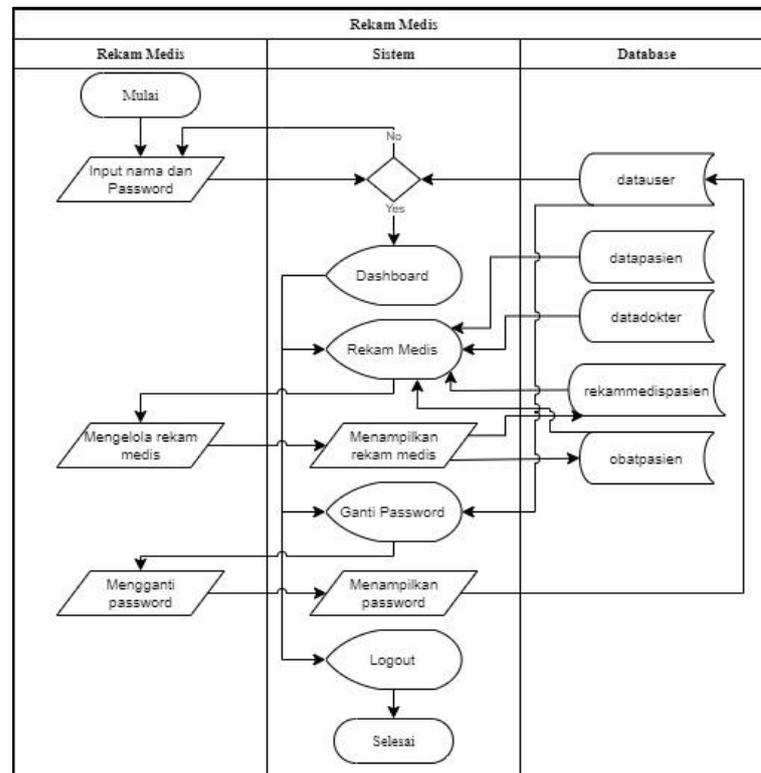
6. Penanggung jawab memilih menu obat keluar untuk melihat data obat keluar . Pada menu ini terdapat grafik yang bisa dijadikan acuan kegiatan klinik untuk kedepannya. Kemudian datanya diambil dari dalam *database* dataobat, datapbf, dan obatkeluar.
7. Penanggung jawab memilih menu stok obat untuk melihat data stok obat yang kemudian datanya diambil dari dalam *database* dataobat, datapbf, dan stokobat.
8. Penanggung Jawab memilih menu ganti *password* untuk mengganti kata sandi yang kemudian datanya akan disimpan ke dalam *database* datauser lalu sistem akan menampilkan ganti *password* yang datanya diambil dari dalam *database* datauser.
9. Penanggung jawab memilih menu *logout*.
10. Selesai.

#### **5.1.2.2.3. Rekam Medis**

*Flowchart* sistem diusulkan untuk bagian rekam medis yang dimulai dari melakukan *login*

hingga melakukan *logout*, dapat dilihat pada gambar

5.5.



**Gambar 5.5 Flowchart yang Diusulkan Rekam Medis**

Dari gambar 5.5 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

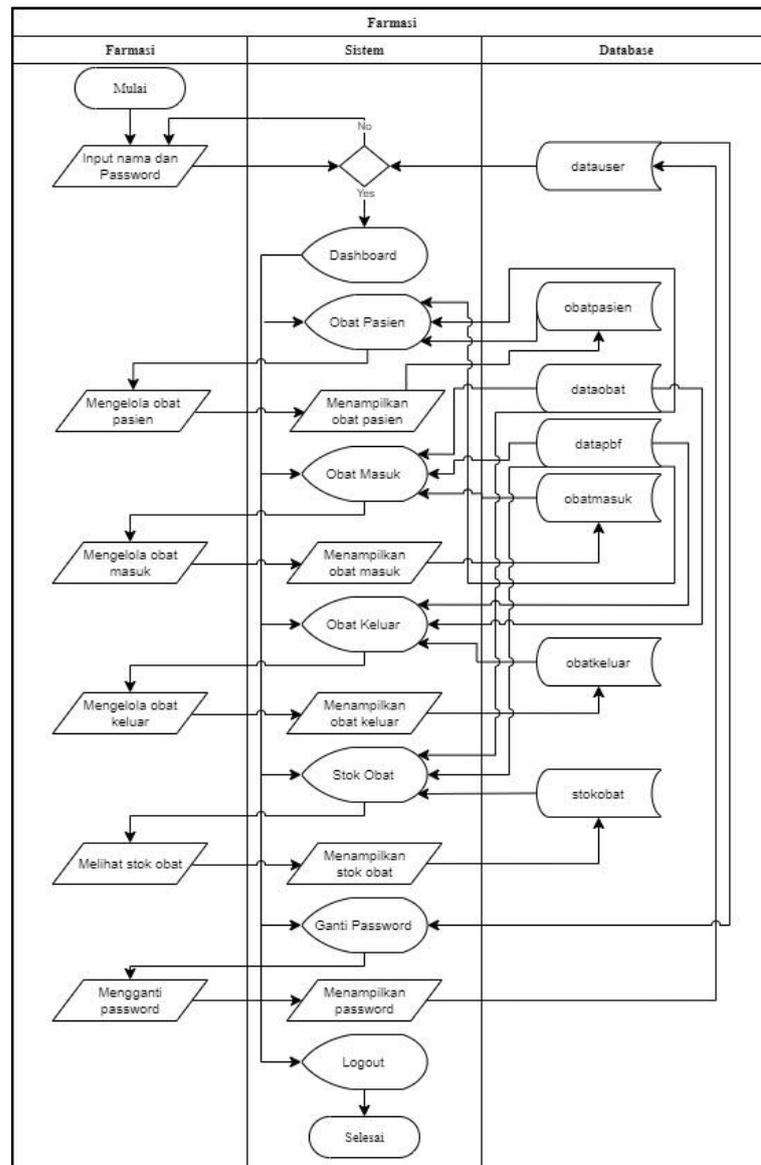
1. *User* memasukkan nama dan *password* yang sudah terdaftar sebagai rekam medis.
2. Sistem akan melakukan validasi terhadap *inputan* dari tabel *datauser* yang ada di *database*, jika iya maka akan berpindah ke tampilan

*dashboard*, jika tidak maka akan kembali menampilkan *form login*.

3. Rekam medis memilih menu rekam medis untuk mengelola data rekam medis yang kemudian datanya akan disimpan ke dalam *database* rekammedispasien dan obatpasien lalu sistem akan menampilkan data rekam medis yang datanya diambil dari dalam *database* rekammedispasien, datapasien, datadokter dan obatpasien
4. Rekam medis memilih menu ganti *password* untuk mengganti kata sandi yang kemudian datanya akan disimpan ke dalam *database* *datauser* lalu sistem akan menampilkan ganti *password* yang datanya diambil dari dalam *database* *datauser*.
5. Rekam medis memilih menu *logout*.
6. Selesai.

#### **5.1.2.2.4. Farmasi**

*Flowchart* sistem diusulkan untuk bagian farmasi yang dimulai dari melakukan *login* hingga melakukan *logout*, dapat dilihat pada gambar 5.6.



**Gambar 5.6** *Flowchart* yang Diusulkan Farmasi

Dari gambar 5.6 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. *User* memasukkan nama dan *password* yang sudah terdaftar sebagai farmasi.

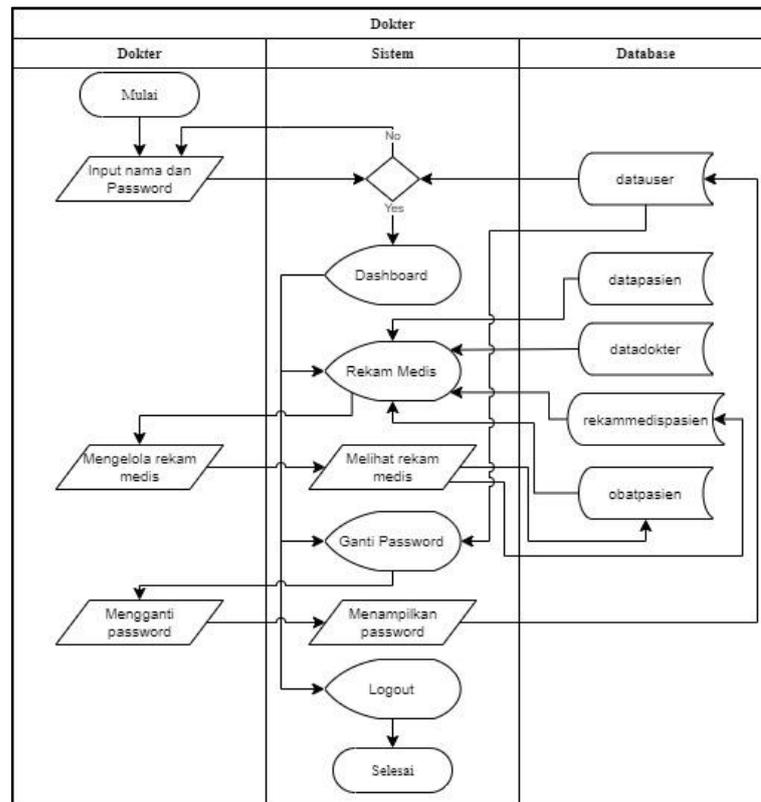
2. Sistem akan melakukan validasi terhadap *inputan* dari tabel *datauser* yang ada di *database*, jika iya maka akan berpindah ke tampilan *dashboard*, jika tidak maka akan kembali menampilkan *form login*.
3. Farmasi memilih menu obat pasien untuk mengelola data obat pasien. Menu ini digunakan saat farmasi ingin melihat resep obat yang telah diberikan dokter untuk pasien yang telah melakukan pemeriksaan sebelumnya. Kemudian data pada obat pasien ini akan disimpan ke dalam *database* obatpasien lalu sistem akan menampilkan obat pasien yang datanya diambil dari dalam *database* obatpasien, dataobat, dan datapbf.
4. Farmasi memilih menu obat masuk untuk mengelola data obat masuk yang kemudian datanya akan disimpan ke dalam *database* obatmasuk lalu sistem akan menampilkan obat masuk yang datanya diambil dari dalam *database* obatmasuk, dataobat, dan datapbf.
5. Farmasi memilih menu obat keluar untuk mengelola data obat keluar. Pada menu ini

memiliki perbedaan antara *user* farmasi dan penanggung jawab, *user* farmasi tidak memiliki grafik yang ada di *user* penanggung jawab. Data pada menu ini akan disimpan ke dalam *database* obatkeluar lalu sistem akan menampilkan obat keluar yang datanya diambil dari dalam *database* obatkeluar, dataobat, dan datapbf.

6. Farmasi memilih menu stok obat untuk mengelola data stok obat yang kemudian datanya menampilkan stok obat yang datanya diambil dari dalam *database* stokobat, dataobat, dan datapbf.
7. Farmasi memilih menu ganti *password* untuk mengganti kata sandi yang kemudian datanya akan disimpan ke dalam *database* datauser lalu sistem akan menampilkan ganti *password* yang datanya diambil dari dalam *database* datauser.
8. Farmasi memilih menu *logout*.
9. Selesai.

#### **5.1.2.2.5. Dokter**

*Flowchart* sistem diusulkan untuk bagian dokter yang dimulai dari melakukan *login* hingga melakukan *logout*, dapat dilihat pada gambar 5.7.



**Gambar 5.7 Flowchart yang Diusulkan Dokter**

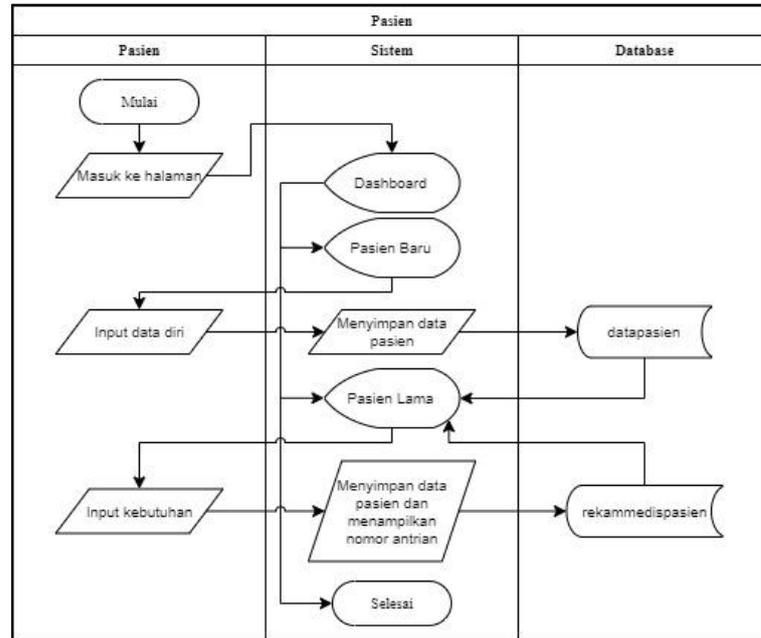
Dari gambar 5.7 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. *User* memasukkan nama dan *password* yang sudah terdaftar sebagai dokter.
2. Sistem akan melakukan validasi terhadap *inputan* dari tabel *datauser* yang ada di *database*, jika iya maka akan berpindah ke tampilan *dashboard*, jika tidak maka akan kembali menampilkan *form login*.

3. Dokter memilih menu rekam medis untuk mengelola data rekam medis. Pada menu ini dokter dapat melihat data-data rekam medis sebelumnya yang dimiliki oleh pasien. Kemudian data pada menu rekam medis ini akan disimpan ke dalam *database* pada tabel rekammedispasien dan obatpasien lalu sistem akan menampilkan rekam medis yang datanya diambil dari dalam *database* datadokter, rekammedispasien, dan datapasien.
4. Dokter memilih menu ganti *password* untuk mengganti kata sandi yang kemudian datanya akan disimpan ke dalam *database* datauser lalu sistem akan menampilkan ganti *password* yang datanya diambil dari dalam *database* datauser.
5. Dokter memilih menu *logout*.
6. Selesai.

#### **5.1.2.2.6. Pasien**

*Flowchart* sistem diusulkan untuk bagian pasien yang dimulai dari melakukan pendaftaran sebagai pasien baru hingga mendapatkan nomor antrian, dapat dilihat pada gambar 5.8.



**Gambar 5.8 Flowchart yang Diusulkan Pasien**

Dari gambar 5.8 peneliti dapat menjelaskan alur dari *flowchart* sebagai berikut :

1. Pasien masuk ke halaman *dashboard*.
2. Sistem menampilkan halaman *dashboard*.
3. Pasien memilih menu pasien baru untuk melakukan pendaftaran bagi pasien yang belum terdaftar yang kemudian datanya akan disimpan ke dalam *database* *datap pasien* lalu pasien akan otomatis berada di halaman pasien lama untuk melanjutkan pendaftaran berobat secara online.
4. Pasien memilih menu pasien lama untuk melakukan pendaftaran berobat secara online

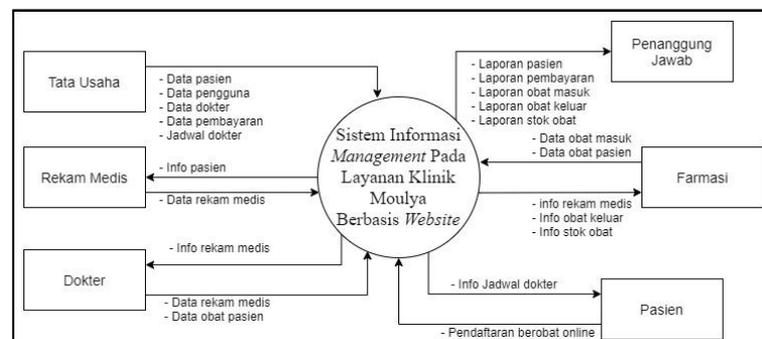
bagi pasien yang sudah terdaftar yang kemudian datanya akan disimpan ke dalam *database* rekam medis pasien dan sistem akan menampilkan nomor antrian pasien.

5. Selesai.

### 5.1.2.3. Diagram Alir Data

#### 5.1.2.3.1. Diagram Konteks

Diagram konteks digambarkan dengan satu simbol lingkaran besar yang mewakili seluruh proses atau kegiatan yang terdapat dalam suatu sistem. Diagram konteks sistem yang peneliti buat dapat dilihat pada gambar 5.9.



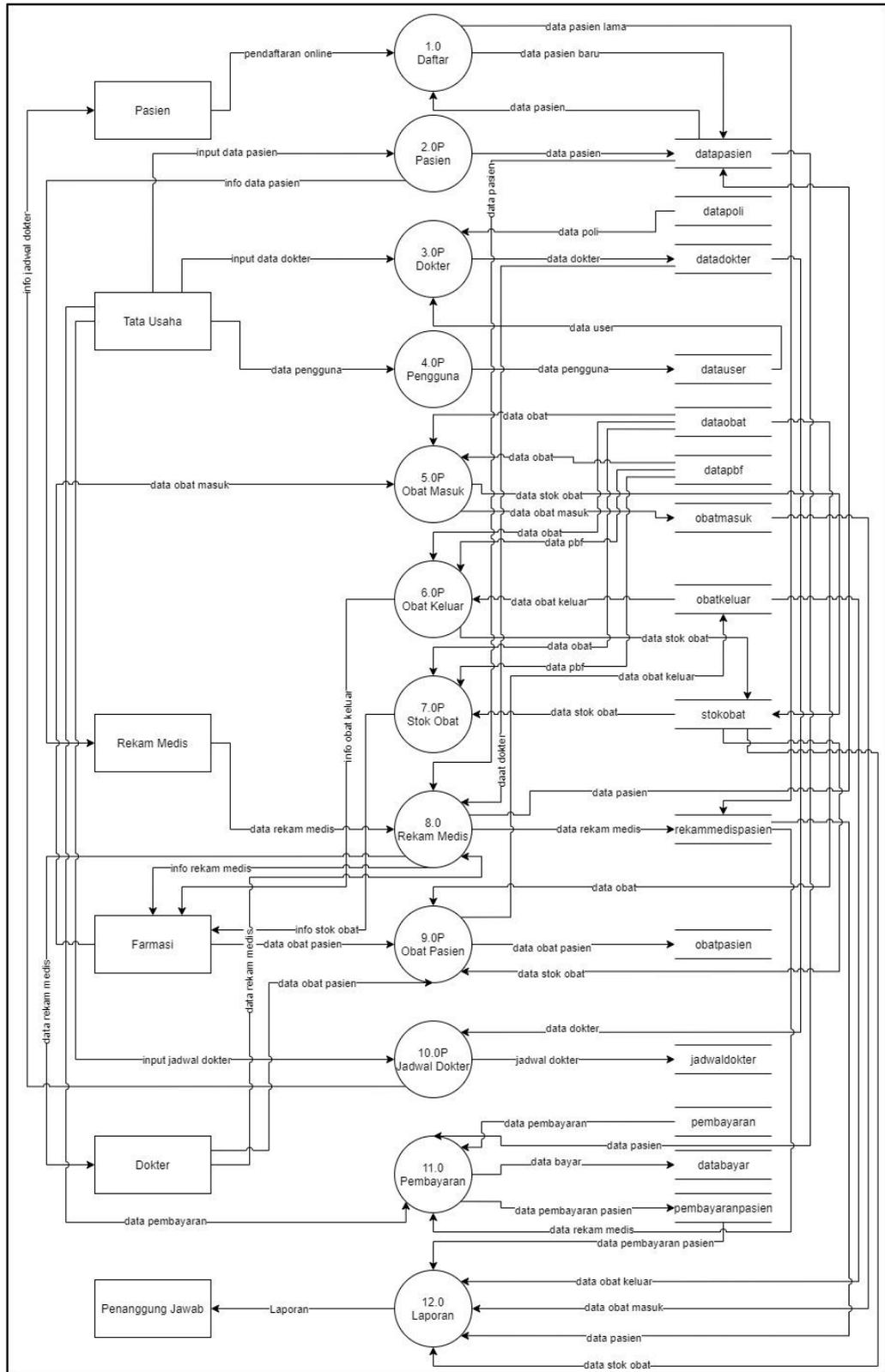
**Gambar 5.9 Diagram Konteks**

Dari gambar 5.9 peneliti dapat menjelaskan alur dari diagram konteks sebagai berikut :

1. Tata usaha melakukan input data pasien, input data pengguna, input data dokter, input data pembayaran dan input jadwal dokter.
2. Penanggung jawab mendapatkan laporan pasien, laporan pembayaran, laporan obat masuk, laporan obat keluar dan laporan stok obat.
3. Rekam medis melakukan input data rekam medis. Sistem akan memberikan info pasien.
4. Farmasi melakukan input data obat masuk dan input data obat pasien. Sistem akan memberikan info rekam medis, info obat keluar, dan info stok obat.
5. Dokter melakukan input data rekam medis dan data obat pasien. Sistem akan memberikan info rekam medis.
6. Pasien melakukan input pendaftaran berobat online. Sistem akan memberikan info jadwal dokter.

#### **5.1.2.3.2. Diagram Level 0**

Berdasarkan *flowchart* sistem yang diusulkan sebelumnya, maka peneliti dapat membuat gambaran arus data berupa diagram level 0 yang dapat dilihat pada gambar 5.10.



Gambar 5.10 Diagram Level 0

Dari gambar 5.10 peneliti dapat menjelaskan alur dari diagram level 0 sebagai berikut :

1. Proses 1.0 daftar merupakan proses yang dilakukan oleh pasien, proses ini berupa pendaftaran online yang datanya akan disimpan menjadi dua bagian sesuai dengan kebutuhan pasien yaitu pasien baru data akan disimpan ke dalam tabel *database* datapasien dan pasien lama data akan disimpan ke dalam tabel *database* rekammedispasien. Ketika melakukan pendaftaran online untuk pasien lama, sistem akan memanggil data pasien untuk memastikan pasien tersebut sebelumnya telah terdaftar di klinik atau belum terdaftar.
2. Proses 2.0P pasien merupakan proses yang dilakukan oleh tata usaha, proses ini berupa pengelolaan data seperti tambah data, edit data, dan hapus data. Pada proses ini data akan disimpan ke dalam tabel *database* datapasien. Pada proses ini pula lah *user* yang telah melakukan *login* dengan level sebagai rekam medis akan mendapatkan informasi data pasien.

3. Proses 3.0P dokter merupakan proses yang dilakukan oleh tata usaha, proses ini berupa pengelolaan data seperti tambah data, *edit* data, dan hapus data. Pada proses ini data akan disimpan ke dalam tabel *database* datadokter dan sistem juga akan memanggil data dari tabel *database* datapoli serta tabel datauser.
4. Proses 4.0P Pengguna merupakan proses yang dilakukan oleh tata usaha, proses ini berupa pengelolaan data seperti tambah data, *edit* data, dan hapus data. Pada proses ini data akan disimpan ke dalam tabel *database* datauser.
5. Proses 5.0P obat masuk merupakan proses yang dilakukan oleh farmasi, proses ini berupa pengelolaan data seperti tambah data, *edit* data, dan hapus data. Data pada proses ini akan disimpan ke dalam tabel *database* obatmasuk dan pada proses ini juga tabel stokobat akan terisi setelah farmasi mengirim data obat masuk. Saat melakukan pengelolaan data, sistem akan memanggil tabel dataobat dan datapbf.
6. Proses 6.0P obat keluar merupakan proses yang dilakukan oleh farmasi setelah melakukan

pengimputan pada obat pasien yang datanya akan disimpan ke dalam tabel *database* obatkeluar. Data pada tabel *database* obatkeluar ini juga akan melakukan pengurangan stok pada tabel *database* stokobat. Pada proses ini sistem akan memanggil tabel dataobat dan datapbf. Pada proses ini juga farmasi akan mendapatkan informasi obat keluar.

7. Proses 7.0P stok obat merupakan proses yang dilakukan oleh farmasi setelah melakukan tambah data ke tabel *database* obatmasuk. Ketika melakukan tambah data, proses tersebut juga menyimpan data yang datanya disimpan ke dalam tabel *database* stokobat dan pada proses ini ketika farmasi melakukan tambah data pada obat pasien, data pada tabel *database* stokobat akan berkurang. Saat akan melihat data dari stok obat, sistem akan memanggil tabel dataobat dan datapbf. Pada proses ini juga farmasi akan mendapatkan informasi stok obat.
8. Proses 8.0 rekam medis merupakan 2 proses yang dilakukan oleh rekam medis dan dokter yang datanya akan disimpan ke dalam tabel

*database* rekammedispasien dan datapasien. Sistem pada proses ini juga akan memanggil tabel datapasien dan data dokter dari dalam *database*. Pada proses ini pula lah *user* yang telah melakukan *login* dengan level sebagai dokter akan mendapatkan informasi rekam medis.

9. Proses 9.0P obat pasien merupakan proses yang dilakukan oleh farmasi dan dokter yang datanya akan disimpan ke dalam tabel *database* obat pasien. Sistem pada proses ini juga akan memanggil tabel dataobat dan stokobat dari dalam *database*. Pada proses ini pulalah farmasi yang telah melakukan *input* data ke dalam tabel *database* obatpasien, data yang ada juga akan tersimpan ke dalam tabel *database* obatkeluar.
10. Proses 10.0P jadwal dokter merupakan proses yang dilakukan oleh tata usaha yang datanya akan disimpan ke dalam tabel *database* jadwaldokter. Sistem pada proses ini juga akan memanggil tabel datadokter dari dalam *database*. Ketika pasien masuk ke dalam halaman

tampilan daftar online maka pasien akan mendapatkan informasi jadwal dokter.

11. Proses 11.0 pembayaran merupakan 2 proses yang dilakukan oleh tata usaha yang datanya akan disimpan ke dalam tabel *database* databayar dan pembayaranpasien. Sistem pada proses ini juga akan memanggil tabel pembayaran, rekammedispasien, dan datapasien dari dalam *database*.

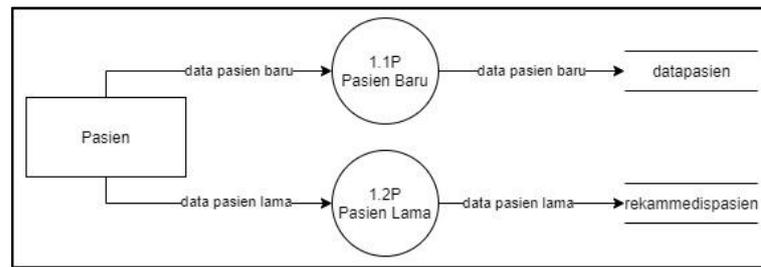
12. Proses 12.0 laporan merupakan proses yang terdiri dari kumpulan-kumpulan laporan yang ada pada beberapa bagian. Laporan tersebut akan di tampilkan ketika *user* telah melakukan *login* sebagai penanggung jawab.

#### **5.1.2.3.3. Diagram Rinci**

Berdasar diagram level 0, terdapat beberapa proses yang perlu di rincikan prosesnya. Beberapa proses yang perlu di rincikan ialah sebagai berikut :

##### **5.1.2.3.3.1. Level 1 Proses 1**

Diagram rinci level 1 proses 1 yang dapat dilihat pada gambar 5.11.



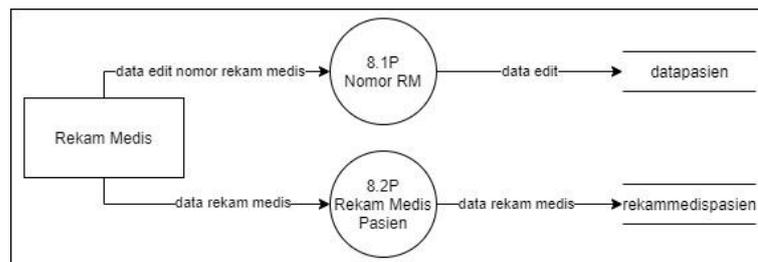
**Gambar 5.11 Level 1 Proses 1**

Pada diagram level 0 di proses 1.0 daftar terbagi menjadi 2 proses yang telah digambarkan pada diagram rinci pada level 1 proses 1 yang dapat dilihat pada gambar 5.11. Dari gambar 5.11 peneliti dapat menjelaskan alur dari diagram level 1 proses 1 sebagai berikut :

1. Proses 1.1P pasien baru merupakan proses yang dilakukan oleh pasien yang datanya belum terdaftar di klinik. Pada proses ini data akan disimpan ke dalam tabel *database* datapasien.
2. Proses 1.2P pasien lama merupakan proses yang dilakukan oleh pasien yang datanya sudah terdaftar di klinik. Pada proses ini data akan disimpan ke dalam tabel *database* rekammedispasien.

### 5.1.2.3.3.2. Level 1 Proses 8

Diagram rinci level 1 proses 8 yang dapat dilihat pada gambar 5.12.



**Gambar 5.12 Diagram Level 1 Proses 8**

Pada diagram level 0 di proses 8.0 pembayaran terbagi menjadi 2 proses yang telah digambarkan pada diagram rinci pada level 1 proses 8 yang dapat dilihat pada gambar 5.12. Dari gambar 5.12 peneliti dapat menjelaskan alur dari diagram level 1 proses 8 sebagai berikut :

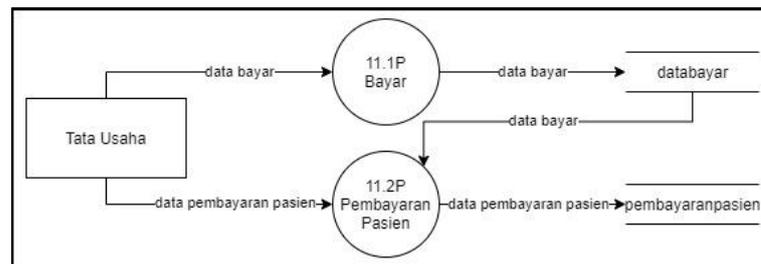
1. Proses 8.1P nomor rm merupakan proses yang dilakukan oleh rekam medis. Pada proses ini adalah data dimana *user* rekam medis ingin menambahkan nomor rekam medis ataupun melakukan pengeditan pada nomor rekam medis tersebut dan

data yang ada akan disimpan ke dalam tabel *database* datapasien.

2. Proses 8.2P rekam medis pasien merupakan proses yang dilakukan oleh rekam medis. Pada proses ini data yang ada akan disimpan ke dalam tabel *database* rekammedispasien.

#### 5.1.2.3.3.3. Level 1 Proses 11

Diagram rinci level 1 proses 11 yang dapat dilihat pada gambar 5.13.



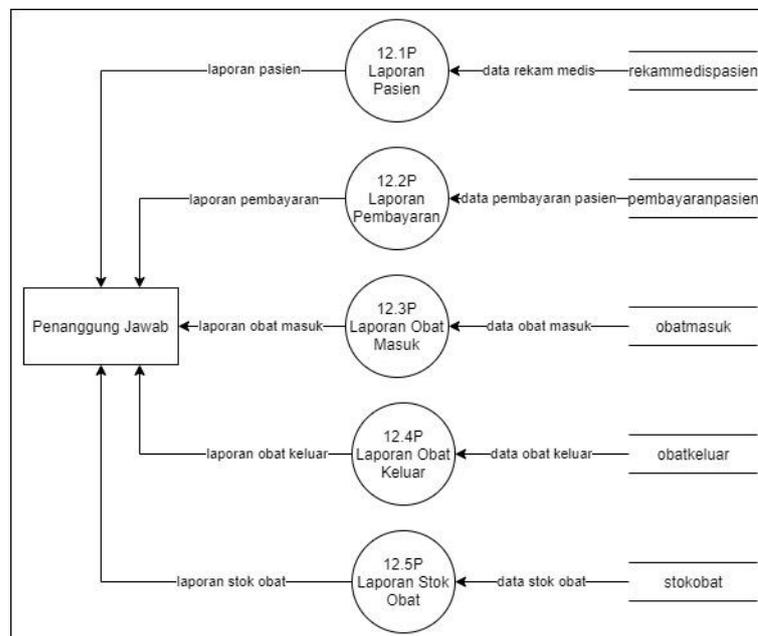
**Gambar 5.13 Diagram Level 1 Proses 11**

Pada diagram level 0 di proses 11.0 pembayaran terbagi menjadi 2 proses yang telah digambarkan pada diagram rinci pada level 1 proses 11 yang dapat dilihat pada gambar 5.13. Dari gambar 5.13 peneliti dapat menjelaskan alur dari diagram level 1 proses 11 sebagai berikut :

1. Proses 11.1P bayar merupakan proses yang dilakukan oleh tata usaha. Pada proses ini data yang ada akan disimpan ke dalam tabel *database* databayar.
2. Proses 11.2P pembayaran pasien merupakan proses yang dilakukan oleh tata usaha. Pada proses ini data yang ada akan disimpan ke dalam tabel *database* pembayaranpasien.

#### 5.1.2.3.3.4. Level 1 Proses 12

Diagram rinci level 1 proses 12 yang dapat dilihat pada gambar 5.14.



**Gambar 5.14 Diagram Level 1 Proses 12**

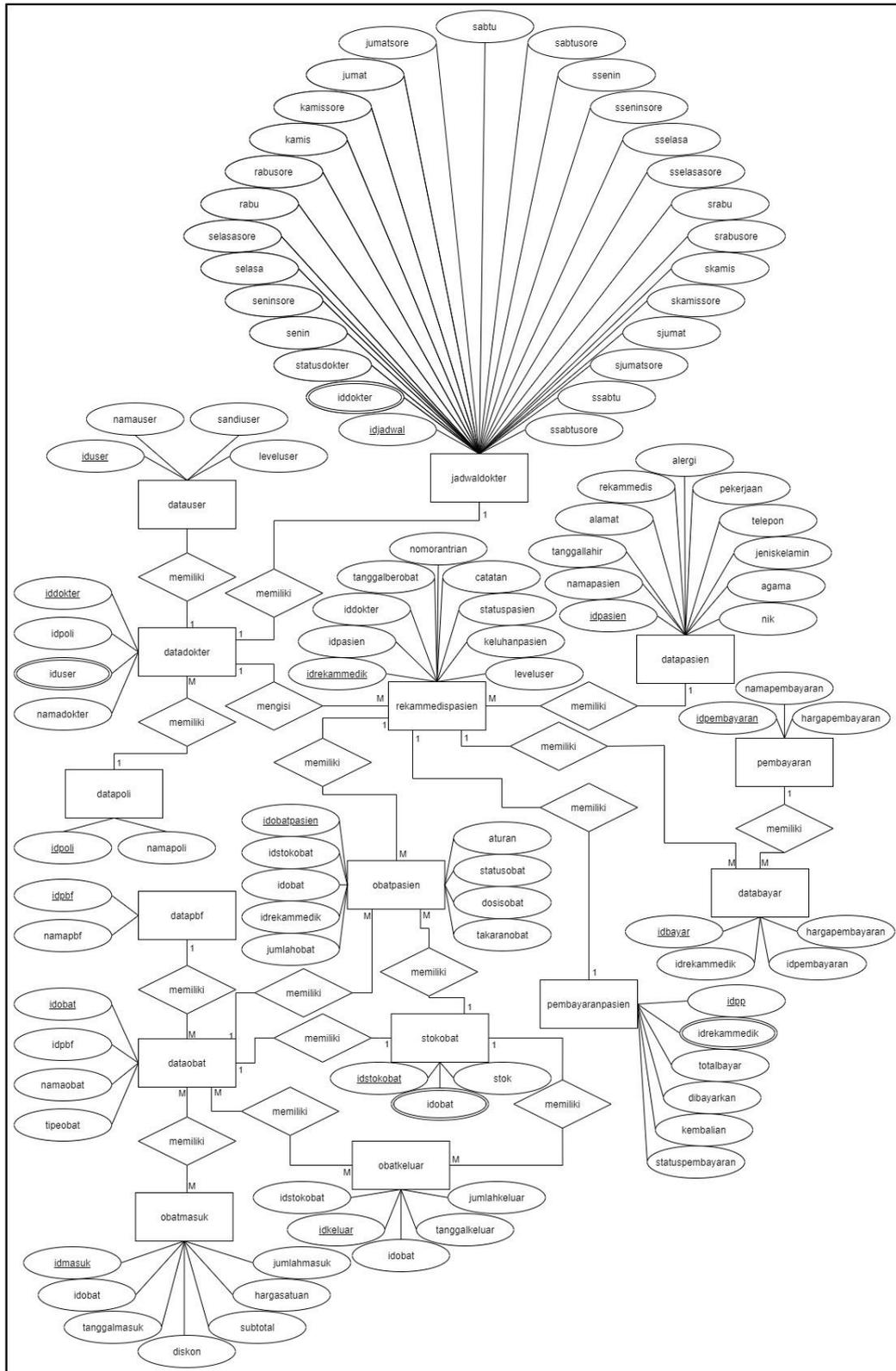
Pada diagram level 0 di proses 12.0 laporan terbagi menjadi 5 proses yang telah digambarkan pada diagram rinci pada level 1 proses 12 yang dapat dilihat pada gambar 5.14. Dari gambar 5.14 peneliti dapat menjelaskan alur dari diagram level 1 proses 12 sebagai berikut :

1. Proses 12.1P laporan pasien merupakan data laporan pasien yang telah datang dan telah melakukan pengobatan di Klinik Moulya, yang datanya akan didapatkan oleh penanggung jawab . Pada proses ini data yang ada diambil dari tabel *database* rekammedispasien.
2. Proses 12.2P laporan pembayaran merupakan data laporan yang didapatkan penanggung jawab agar penanggung jawab dapat mengetahui mana pasien yang telah melakukan pembayaran dan mana yang tidak melakukan pembayaran. Pada proses ini data yang ada diambil dari tabel *database* pembayaranpasien.

3. Proses 12.3P laporan obat masuk merupakan data laporan yang didapatkan penanggung jawab. Pada proses ini data yang ada diambil dari tabel *database* obatmasuk.
4. Proses 12.4P laporan obat keluar merupakan data laporan yang didapatkan penanggung jawab setelah farmasi memberikan obat kepada pasien. Pada proses ini data yang ada diambil dari tabel *database* obatkeluar.
5. Proses 12.5P laporan stok obat merupakan data laporan yang didapatkan penanggung jawab agar penanggung jawab mengetahui obat beserta jumlah stok yang ada di bagian farmasi. Pada proses ini data yang ada diambil dari tabel *database* stokobat.

#### **5.1.2.4. Entity Relationship Diagram (ERD)**

*Entity Relationship Diagram* (ERD) telah digambarkan oleh peneliti yang dapat dilihat pada gambar 5.15.



**Gambar 5.15 Entity Relationship Diagram**

Dari gambar 5.15 peneliti dapat menjelaskan alur dari *entity relationship diagram* sebagai berikut :

1. Pada entitas *datadokter* dan *datauser* memiliki hubungan *one to one*. Hubungan tersebut memiliki arti, satu data pada *datadokter* hanya dapat memiliki satu data pada *datauser* dan satu data pada *datauser* hanya dapat dimiliki oleh satu data pada *datadokter*.
2. Pada entitas *datadokter* dan *datapoli* memiliki hubungan *many to one*. Hubungan tersebut memiliki arti, banyak data pada *datadokter* memiliki satu data pada *datapoli* yang sama dan satu data pada *datapoli* dapat dimiliki oleh banyak data pada *datadokter*.
3. Pada entitas *datadokter* dan *jadwaldokter* memiliki hubungan *one to one*. Hubungan tersebut memiliki arti, satu data pada *datadokter* hanya dapat memiliki satu data pada *jadwaldokter* dan satu data pada *jadwaldokter* hanya dapat dimiliki oleh satu data pada *datadokter*.
4. Pada entitas *datadokter* dan *rekammedispasien* memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada *datadokter* dapat mengisi banyak data pada *rekammedispasien* dan banyak data pada *rekammedispasien* dapat diisi oleh satu data pada *datadokter*.

5. Pada entitas dataobat dan datapbf memiliki hubungan *many to one*. Hubungan tersebut memiliki arti, banyak data pada dataobat dapat memiliki satu data pada datapbf yang sama dan satu data pada datapbf dapat dimiliki oleh banyak data pada dataobat.
6. Pada entitas obatmasuk dan dataobat memiliki hubungan *many to many*. Hubungan tersebut memiliki arti, banyak data pada obatmasuk dapat memiliki banyak data pada dataobat dan banyak data pada dataobat dapat dimiliki oleh banyak data pada obatmasuk.
7. Pada entitas stokobat dan dataobat memiliki hubungan *one to one*. Hubungan tersebut memiliki arti, satu data pada stokobat hanya dapat memiliki satu data pada dataobat dan satu data pada dataobat hanya dapat dimiliki oleh satu data pada stokobat.
8. Pada entitas obatkeluar dan dataobat memiliki hubungan *many to many*. Hubungan tersebut memiliki arti, banyak data pada obatmasuk dapat memiliki banyak data pada dataobat dan banyak data pada dataobat dapat dimiliki oleh banyak data pada obatkeluar.

9. Pada entitas obatpasien dan dataobat memiliki hubungan *many to one*. Hubungan tersebut memiliki arti, banyak data pada obatpasien memiliki satu data pada dataobat yang sama dan satu data pada dataobat dapat dimiliki oleh banyak data pada obatpasien.
10. Pada entitas stokobat dan obatkeluar memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada stokobat dapat memiliki banyak data pada obatkeluar dan banyak data pada obatkeluar dapat dimiliki oleh satu data pada stokobat.
11. Pada entitas stokobat dan obatpasien memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada stokobat dapat memiliki banyak data pada obatpasien dan banyak data pada obatpasien dapat dimiliki oleh satu data pada stokobat.
12. Pada entitas rekammedispasien dan obatpasien memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada rekammedispasien dapat memiliki banyak data pada obatpasien dan banyak data pada obatpasien dapat dimiliki oleh satu data pada rekammedispasien.
13. Pada entitas pembayaran dan databayar memiliki hubungan *one to many*. Hubungan tersebut memiliki

arti, satu data pada pembayaran dapat memiliki banyak data pada databayar dan banyak data pada databayar dapat dimiliki oleh satu data pada pembayaran

14. Pada entitas datapasien dan rekammedispasien memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada datapasien dapat memiliki banyak data pada rekammedispasien dan banyak data pada rekammedispasien dapat dimiliki oleh satu data pada datapasien.
15. Pada entitas rekammedispasien dan databayar memiliki hubungan *one to many*. Hubungan tersebut memiliki arti, satu data pada rekammedispasien dapat memiliki banyak data pada databayar dan banyak data pada databayar dapat dimiliki oleh satu data pada rekammedispasien.
16. Pada entitas rekammedispasien dan pembayaranpasien memiliki hubungan *one to one*. Hubungan tersebut memiliki arti, satu data pada rekammedispasien hanya dapat memiliki satu data pada pembayaranpasien dan satu data pada pembayaranpasien hanya dapat dimiliki oleh satu data pada rekammedispasien.

### 5.1.2.5. Struktur Tabel

Struktur tabel berisikan nama-nama *field*, tipe data, dan ukuran dari tabel yang dibuat berdasarkan *Entity Relationship Diagram* (ERD) yang telah digambarkan pada gambar 5.15. Struktur tabel dapat dilihat sebagai berikut :

#### 1. Tabel Data *User*

Tabel data *user* digunakan untuk menyimpan data *user* atau pengguna yang kemudian tabel ini akan digunakan untuk melakukan verifikasi data saat *user* atau pengguna ingin melakukan *login* untuk masuk ke akun yang telah terdaftar. Struktur tabel dari tabel data *user* dapat dilihat pada tabel 5.1.

Nama Tabel : *datauser*

*Primary Key* : *iduser*

**Tabel 5.1 Tabel Data *User***

No.	Nama <i>Field</i>	Tipe Data	Ukuran	Keterangan
1	<i>iduser</i>	<i>INT</i>	11	<i>Primary Key</i>
2	<i>namauser</i>	<i>Varchar</i>	50	Nama setiap <i>user</i>
3	<i>sandiuser</i>	<i>Varchar</i>	50	Sandi setiap <i>user</i>
4	<i>leveluser</i>	<i>Varchar</i>	25	Level yang diberikan untuk <i>user</i> sesuai dengan kedudukan <i>user</i> di klinik

#### 2. Tabel Data Dokter

Tabel data dokter digunakan untuk menyimpan data dokter berupa data diri dari dokter yang bekerja di klinik, tabel ini akan dipanggil pada saat mengisi jadwal

dokter dengan menggunakan iddokter. Struktur tabel dari tabel data dokter dapat dilihat pada tabel 5.2.

Nama Tabel : datadokter

*Primary Key* : iduser

*Foreign Key* : idpoli

*Index Key* : iduser

**Tabel 5.2 Tabel Data Dokter**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	iddokter	<i>INT</i>	11	<i>Primary Key</i>
2	idpoli	<i>INT</i>	11	<i>Foreign Key</i>
3	iduser	<i>INT</i>	11	<i>Index Key</i>
4	namadokter	<i>Varchar</i>	50	Nama setiap dokter

### 3. Tabel Data Bayar

Tabel data bayar digunakan untuk menyimpan data bayar berupa tipe pengobatan yang telah didapatkan oleh pasien selama pengobatan berlangsung. Struktur tabel dari tabel data bayar dapat dilihat pada tabel 5.3.

Nama Tabel : databayar

*Primary Key* : idbayar

*Index Key* : idrekammedik, idpembayaran

**Tabel 5.3 Tabel Data Bayar**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idbayar	<i>INT</i>	11	<i>Primary Key</i>
2	idrekammedik	<i>INT</i>	11	<i>Index Key</i>
3	idpembayaran	<i>INT</i>	11	<i>Index Key</i>
4	hargapembayaran	<i>INT</i>	50	Harga setiap pembayaran

#### 4. Tabel Data Obat

Tabel data obat digunakan untuk menyimpan data obat berupa informasi atau nama obat yang pernah digunakan di klinik, tabel ini akan dipanggil pada saat menambahkan data pada tabel obat masuk dengan menggunakan idobat. Struktur tabel dari tabel data obat dapat dilihat pada tabel 5.4.

Nama Tabel : dataobat

*Primary Key* : idobat

*Index Key* : idpbf

**Tabel 5.4 Tabel Data Obat**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idobat	<i>INT</i>	11	<i>Primary Key</i>
2	idpbf	<i>INT</i>	11	<i>Index Key</i>
3	namaobat	<i>Varchar</i>	100	Nama setiap obat
4	tipeobat	<i>Varhar</i>	25	Tipe setiap obat

#### 5. Tabel Data Pasien

Tabel data pasien digunakan untuk menyimpan data pasien berupa data diri pasien yang telah terdaftar di klinik, tabel ini akan di. Struktur tabel dari tabel data pasien dapat dilihat pada tabel 5.5.

Nama Tabel : datapasien

*Primary Key* : idpasien

**Tabel 5.5 Tabel Data Pasien**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idpasien	<i>INT</i>	11	<i>Primary Key</i>
2	namapasien	<i>Varchar</i>	50	Nama setiap pasien
3	tanggallahir	<i>Date</i>	-	Tanggal lahir setiap pasien
4	alamat	<i>Varchar</i>	100	Alamat setiap pasien
5	rekammedis	<i>Varchar</i>	10	Nomor rekam medis setiap pasien
6	pekerjaan	<i>Varchar</i>	100	Pekerjaan setiap pasien
7	telepon	<i>Char</i>	25	Nomor telepon setiap pasien
8	jeniskelamin	<i>Varchar</i>	15	Jenis kelamin setiap pasien
9	agama	<i>Varchar</i>	15	Agama setiap pasien
10	nik	<i>INT</i>	20	Nomor Induk Kependudukan setiap pasien
11	alergi	<i>Varchar</i>	1000	Alergi yang diderita pasien

## 6. Tabel Data PBF

Tabel data PBF digunakan untuk menyimpan data dari pedagang besar farmasi berupa informasi dari PBF. Struktur tabel dari tabel data PBF dapat dilihat pada tabel 5.6.

Nama Tabel : datapbf

*Primary Key* : idpbf

**Tabel 5.6 Tabel Data PBF**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idpbf	<i>INT</i>	11	<i>Primary Key</i>
2	namapbf	<i>Varchar</i>	50	Nama setiap Pedagang Besar Farmasi

## 7. Tabel Data Poli

Tabel data poli digunakan untuk menyimpan data poli berupa informasi poli yang akan digunakan oleh

dokter. Struktur tabel dari tabel data poli dapat dilihat pada tabel 5.7.

Nama Tabel : datapoli

*Primary Key* : idpoli

**Tabel 5.7 Tabel Data Poli**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idpasien	<i>INT</i>	11	<i>Primary Key</i>
2	namapoli	<i>Varchar</i>	25	Nama setiap poli

#### 8. Tabel Jadwal Dokter

Tabel jadwal dokter digunakan untuk menyimpan jadwal dokter yang bekerja di klinik. Struktur tabel dari tabel jadwal dokter dapat dilihat pada tabel 5.8.

Nama Tabel : jadwaldokter

*Primary Key* : idjadwal

*Foreign Key* : iddokter

**Tabel 5.8 Tabel Jadwal Dokter**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idjadwal	<i>INT</i>	11	<i>Primary Key</i>
2	iddokter	<i>INT</i>	11	<i>Foreign Key</i>
3	statusdokter	<i>Varchar</i>	15	Status setiap dokter
4	senin	<i>Time</i>	-	Waktu masuk senin pagi
5	seninsore	<i>Time</i>	-	Waktu masuk senin sore
6	selasa	<i>Time</i>	-	Waktu masuk selasa pagi
7	selasasore	<i>Time</i>	-	Waktu masuk selasa sore
8	rabu	<i>Time</i>	-	Waktu masuk rabu pagi
9	rabusore	<i>Time</i>	-	Waktu masuk rabu sore
10	kamis	<i>Time</i>	-	Waktu masuk kamis pagi
11	kamissore	<i>Time</i>	-	Waktu masuk kamis sore
12	jumat	<i>Time</i>	-	Waktu masuk jumat pagi

No.	Nama Field	Tipe Data	Ukuran	Keterangan
13	jumatsore	Time	-	Waktu masuk jumat sore
14	sabtu	Time	-	Waktu masuk sabtu pagi
15	sabtusore	Time	-	Waktu masuk sabtu sore
16	ssenin	Time	-	Waktu selesai senin pagi
17	sseninsore	Time	-	Waktu selesai senin sore
18	sselasa	Time	-	Waktu selesai selasa pagi
19	sselasasore	Time	-	Waktu selesai selasa sore
20	srabu	Time	-	Waktu selesai rabu pagi
21	srabusore	Time	-	Waktu selesai rabu sore
22	skamis	Time	-	Waktu selesai kamis pagi
23	skamissore	Time	-	Waktu selesai kamis sore
24	sjumat	Time	-	Waktu selesai jumat pagi
25	sjumatsore	Time	-	Waktu selesai jumat sore
26	ssabtu	Time	-	Waktu selesai sabtu pagi
27	ssabtusore	Time	-	Waktu selesai sabtu sore

#### 9. Tabel Obat Keluar

Tabel obat keluar digunakan untuk menyimpan data dari obat yang keluar atau obat yang telah diambil oleh pasien. Struktur tabel dari tabel obat keluar dapat dilihat pada tabel 5.9.

Nama Tabel : obatkeluar

Primary Key : idkeluar

Index Key : idstokobat, idobat

**Tabel 5.9 Tabel Obat Keluar**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idkeluar	INT	11	Primary Key
2	idstokobat	INT	11	Index Key
3	idobat	INT	15	Index Key
4	tanggalkeluar	Timestamp	-	Tanggal setiap obat keluar
5	jumlahkeluar	INT	11	Jumlah setiap obat keluar

## 10. Tabel Obat Masuk

Tabel obat masuk digunakan untuk menyimpan data obat yang masuk ke klinik atau obat yang baru dibeli oleh farmasi. Struktur tabel dari tabel obat masuk dapat dilihat pada tabel 5.10.

Nama Tabel : obatmasuk

*Primary Key* : idmasuk

**Tabel 5.10 Tabel Obat Masuk**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idmasuk	<i>INT</i>	11	<i>Primary Key</i>
2	idobat	<i>INT</i>	11	<i>Index Key</i>
3	tanggalmasuk	<i>Date</i>	-	Tanggal setiap obat masuk
4	diskon	<i>INT</i>	5	Diskon setiap obat masuk
5	jumlahmasuk	<i>INT</i>	11	Jumlah setiap obat masuk
6	hargasatuan	<i>INT</i>	25	Harga setiap obat masuk
7	subtotal	<i>INT</i>	25	Harga total setiap obat masuk

## 11. Tabel Obat Pasien

Tabel obat pasien digunakan untuk menyimpan data obat yang diberikan dokter untuk pasien, tabel ini memanggil idstokobat yang berada di dalam tabel stokobat. Struktur tabel dari tabel obat pasien dapat dilihat pada tabel 5.11.

Nama Tabel : obatpasien

*Primary Key* : idobatpasien

*Index Key* : idstokobat, idobat, idrekammedik

**Tabel 5.11 Tabel Obat Pasien**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idobatpasien	INT	11	Primary Key
2	idstokobat	INT	11	Index Key
3	idobatpasien	INT	11	Index Key
4	idrekammedik	INT	11	Index Key
5	jumlahobat	INT	11	Jumlah obat yang diberikan ke pasien
6	aturan	Varchar	50	Aturan pemakaian obat
7	statusobat	Varchar	15	Status setiap obat
8	dosisobat	INT	5	Dosis obat yang diberikan ke pasien
9	takaranobat	Varchar	50	Takaran obat yang diberikan ke pasien

## 12. Tabel Pembayaran

Tabel data pembayaran digunakan untuk menyimpan data dari tipe pengobatan yang ada di klinik. Struktur tabel dari tabel pembayaran dapat dilihat pada tabel 5.12.

Nama Tabel : pembayaran

Primary Key : idpembayaran

**Tabel 5.12 Tabel Pembayaran**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idpembayaran	INT	11	Primary Key
2	namapembayaran	Varchar	100	Jenis nama pembayaran
3	hargapembayaran	INT	50	Harga pembayaran

## 13. Tabel Pembayaran Pasien

Tabel pembayaran pasien digunakan untuk menyimpan data pembayaran pasien berupa total pembayaran, uang yang diberikan pasien, uang

kembalian yang diterima pasien. Struktur tabel dari tabel pembayaran pasien dapat dilihat pada tabel 5.13.

Nama Tabel : pembayaranpasien

*Primary Key* : idpp

*Foreign Key* : idrekammedik

**Tabel 5.13 Tabel Pembayaran Pasien**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idpp	INT	11	Primary Key
2	idrekammedik	INT	11	Foreign Key
3	totalbayar	INT	50	Total yang harus dibayarkan pasien
4	dibayarkan	INT	50	Uang yang diberikan pasien
5	kembalian	INT	50	Kembalian uang pasien
6	statuspembayaran	Varchar	5	Status pembayaran pasien

#### 14. Tabel Rekam Medis Pasien

Tabel rekam medis pasien digunakan untuk menyimpan data rekam medis. Struktur tabel dari tabel rekam medis pasien dapat dilihat pada tabel 5.14.

Nama Tabel : rekammedispasien

*Primary Key* : idrekammedik

*Index Key* : idpasien, iddokter

**Tabel 5.14 Tabel Rekam Medis Pasien**

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idrekammedik	INT	11	Primary Key
2	idpasien	INT	11	Index Key
3	iddokter	INT	11	Index Key
4	tanggalberobat	Timestamp	-	Tanggal berobat setiap pasien

No.	Nama Field	Tipe Data	Ukuran	Keterangan
5	keluhanpasien	<i>Varchar</i>	1000	Keluhan setiap pasien
6	catatan	<i>Varchar</i>	1000	Catatan yang diberikan dokter
7	statuspasien	<i>Varchar</i>	50	Status berobat pasien
8	nomorantrian	<i>Varchar</i>	5	Nomor antrian yang di dapatkan pasien setelah melakukan pendaftaran

15. Tabel Stok Obat

Tabel stok obat digunakan untuk menyimpan data obat yang masih memiliki stok di klinik. Struktur tabel dari tabel stok obat dapat dilihat pada tabel 5.15.

Nama Tabel : stokobat

*Primary Key* : idstokobat

*Foreign Key* : idobat

Tabel 5.15 Tabel Stok Obat

No.	Nama Field	Tipe Data	Ukuran	Keterangan
1	idstokobat	<i>INT</i>	11	<i>Primary Key</i>
2	idobat	<i>INT</i>	11	<i>Foreign Key</i>
3	stok	<i>INT</i>	11	Stok obat yang masih ada di klinik

#### 5.1.2.6. Desain *Interface*

Desain *interface* merupakan perancangan tampilan *website* sebelum dibuatkan dalam bentuk sistem pada tahap pengkodean sistem. Pada tahap ini peneliti memperlihatkan beberapa desain *interface* saja yang dapat dilihat sebagai berikut :

## 1. Tampilan Daftar *Online*

Desain *interface* dari halaman tampilan daftar *online* yaitu ketika pasien masuk ke halaman *website* pendaftaran untuk melakukan pendaftaran secara *online*.

Desain *interface* dari halaman tampilan daftar *online* dapat dilihat pada gambar 5.16.

### Selamat Datang

Jadwal Dokter								
No	Nama	Status	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
99	XXXXXXXXXXXXXXXXXX	XXXX	* 99:99- 99:99					
			* 99:99- 99:99					

**Gambar 5.16 Tampilan Daftar *Online***

## 2. Pasien Baru

Desain *interface* dari halaman pasien baru yaitu ketika pasien memilih tombol atau fitur pasien baru pada tampilan daftar *online*. Menu ini berfungsi untuk mendaftarkan diri pasien karena pasien belum terdaftar sebagai pasien di Klinik Moulya. Pada tampilan ini, pasien di haruskan mengisi *form* sesuai dengan data diri pasien. Desain *interface* dari halaman pasien baru dapat dilihat pada gambar 5.17.



Logo Klinik Moulya

NIK Pasien \*

Nama Pasien \*

Nomor Rekam Medis

Nama Dokter \*

Waktu Berobat \*

Submit Reset Batal

**Gambar 5.18 Pasien Lama**

#### 4. *Login*

Desain *interface* dari halaman *login* ketika *user* tata usaha, penanggung jawab, farmasi, dokter, dan rekam medis ingin masuk ke dalam akun yang sudah terdaftar ataupun yang sudah didaftarkan oleh *user* tata usaha sebelumnya. Desain *interface* dari halaman *login* dapat dilihat pada gambar 5.19.

Logo Klinik Moulya

Login untuk masuk ke akun anda

Masukkan nama pengguna dan kata sandi yang terdaftar

Nama Pengguna \*

Kata Sandi \*

Login

**Gambar 5.19 Login**

## 5. Dashboard

Desain *interface* dari halaman *dashboard* yaitu ketika *user* telah melakukan *login* akun sebagai tata usaha, penanggung jawab, farmasi, rekam medis, ataupun dokter, tampilan *website* akan berubah ke halaman *dashboard*. Desain *interface* dari halaman *dashboard* dapat dilihat pada gambar 5.20.



**Gambar 5.20 Dashboard**

## 6. Tampilan Data Dokter

Desain *interface* dari halaman tampilan data dokter yaitu kumpulan nama beserta poli dokter yang ada di Klinik Moulya. Pada tampilan ini juga *user* tata usaha dapat melakukan tambah data dokter ataupun tambah poli yang dapat dilihat pada gambar 5.21.



## 8. Tambah Data Dokter

Desain *interface* dari halaman tambah data dokter yaitu ketika *user* tata usaha memilih menu tambah dokter pada tampilan data dokter sebelumnya. Tampilan ini berupa *form simple* yang perlu diisi atau dipilih datanya oleh tata usaha untuk menambahkan data dokter di klinik. Pada tampilan ini tata usaha hanya dapat melakukan pemilihan data saja pada bagian-bagian yang ada. Karena jika ingin melakukan penambahan data dokter, dokter yang bersangkutan haruslah memiliki akun terlebih dahulu yang dapat di daftarkan oleh tata usaha/admin pada menu pengguna. Desain *interface* dari halaman tambah data dokter yang telah peneliti buat dapat dilihat pada gambar 5.23.

Logo Klinik Moulya	
Menu	Tambah Dokter Baru
	Nama Dokter * -- Pilih Disini --
	Poli * -- Pilih Disini --
	<input type="button" value="Submit"/> <input type="button" value="Reset"/> <input type="button" value="Batal"/>

**Gambar 5.23 Tambah Data Dokter**

## 9. Tampilan Data Rekam Medis (Rekam Medis)

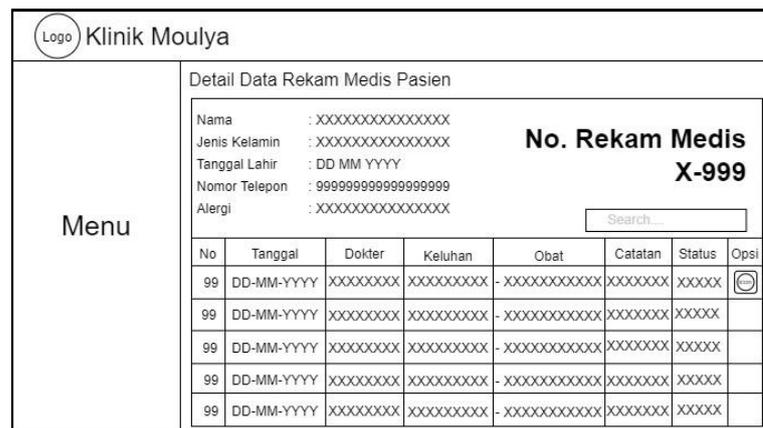
Desain *interface* dari halaman tampilan data rekam medis untuk *user* rekam medis merupakan tampilan yang menampilkan data-data awal atau data sederhana dari rekam medis pasien. Pada tampilan ini data yang ditampilkan hanya berupa nama pasien, nomor rekam medis pasien, tanggal lahir pasien, alamat pasien, dan beberapa tombol atau fitur saja. Salah satu tombol atau fitur yang ada pada tampilan data rekam medis untuk *user* rekam medis adalah tombol detail. Tombol ini berguna untuk melihat detail atau rincian data-data rekam medis sesuai dengan nama pasien yang dipilih. Desain *interface* dari halaman tampilan data rekam medis untuk *user* rekam medis telah peneliti gambarkan yang dapat dilihat pada gambar 5.24.

Logo Klinik Moulya						
Menu	Data Rekam Medis Pasien					
	<input type="text" value="Search..."/>					
	No	Nama	No. RM	Tanggal Lahir	Alamat	Opsi
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
	99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>
99	XXXXXXXXXXXXXXXX	X-999	DD-MM-YYYY	XXXXXXXXXXXXXX	<input type="button" value="Detail"/> <input type="button" value="Print"/>	

**Gambar 5.24 Tampilan Data Rekam Medis (Rekam Medis)**

10. Detail Rekam Medis (Rekam Medis dan Dokter)

Desain *interface* dari halaman Detail Rekam Medis untuk *user* rekam medis dan dokter merupakan tampilan dari detail rekam medis pasien yang dimana saat *user* memilih tombol detail untuk menampilkan data-data rekam medis rekam medis sesuai dengan pasien yang dipilih. Pada tampilan ini setiap data memiliki tombol *edit* untuk melakukan *edit* data jika terjadi kesalahan penulisan pada data, namun tombol *edit* ini hanya akan ada pada tanggal pasien berobat saat itu saja. Desain *interface* dari halaman Detail Rekam Medis untuk *user* rekam medis dan dokter dapat dilihat pada gambar 5.25.



Gambar 5.25 Detail Rekam Medis (Rekam Medis dan Dokter)

## 11. Tambah Isi Rekam Medis (Dokter)

Desain *interface* dari halaman tambah isi rekam medis untuk *user* dokter merupakan tampilan untuk dokter mengisi data-data rekam medis pasien yang telah di pilih sebelumnya. Pada tampilan ini dokter dapat memilih tombol obat pasien untuk menambahkan obat pasien yang telah diperiksa sebelumnya, data yang sudah ditambahkan pada tombol obat pasien dapat langsung ditampilkan di dalam halaman tambah isi rekam medis itu sendiri sehingga dokter dapat mengecek dan memastikan obat pasien terlebih dahulu sebelum menyelesaikan mengisi rekam medis pasien. Desain *interface* dari halaman tambah isi rekam medis untuk *user* dokter yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.26.

The screenshot shows a web interface for 'Klinik Moulya' with a 'Menu' sidebar. The main form is titled 'Tambah Isi Rekam Medis Pasien' and contains the following elements:

- Header:** Logo Klinik Moulya and the title 'Tambah Isi Rekam Medis Pasien'.
- Form Fields:**
  - Nama: :XXXXXXXXXXXXXXXXXXXX
  - Jenis Kelamin: :XXXXXXXXXXXXXXXXXXXX
  - Tanggal Lahir: :DD MM YYYY
  - Nomor Telepon: :999999999999999999
  - Alergi: :XXXXXXXXXXXXXXXXXXXX
  - No. Rekam Medis X-999**
  - Obat Untuk Pasien (button)
  - Keluhan Pasien \* (text area with placeholder text)
  - Catatan Pasien \* (text area with placeholder text)
  - Status Berobat \* (dropdown menu with '- Pilih Disini -')
- Table:** A table with 5 columns: No, Nama Obat, Jumlah, Aturan Pakai, and Opsi. It contains 3 rows of data, each with a circular icon in the 'Opsi' column.
 

No	Nama Obat	Jumlah	Aturan Pakai	Opsi
99	XXXXXXXXXXXXXXXXXXXX	999999999	XXXXXXXXXXXXXXXXXXXX	⊗
99	XXXXXXXXXXXXXXXXXXXX	999999999	XXXXXXXXXXXXXXXXXXXX	⊗
99	XXXXXXXXXXXXXXXXXXXX	999999999	XXXXXXXXXXXXXXXXXXXX	⊗
- Buttons:** Submit, Reset, and Batal.

**Gambar 5.26 Tambah Isi Rekam Medis (Dokter)**

## 12. Tambah Obat Pasien

Desain *interface* dari halaman tambah obat pasien untuk *user* dokter merupakan tampilan ketika *user* dokter memilih tombol obat pasien pada halaman sebelumnya yaitu tambah isi rekam medis. Pada halaman ini dokter menambahkan obat-obat pasien baik nama obat, dosis obat, takaran, dll, yang kemudian data-data obat tersebut akan tampil di halaman tambah isi rekam medis. Desain *interface* dari halaman tambah obat pasien untuk *user* dokter dapat dilihat pada gambar 5.27.

Logo Klinik Moulya	
Menu	Tambah Obat Pasien
	Nama Obat * -- Pilih Disini --
	Jumlah Obat * [Input Field]
	Dosis Obat per Hari * [Input Field]
	Takaran Obat * -- Pilih Disini --
	Aturan Pakai * -- Pilih Disini --
<input type="button" value="Submit"/> <input type="button" value="Reset"/> <input type="button" value="Batal"/>	

**Gambar 5.27 Tambah Obat Pasien**

## 13. Perbarui Obat Pasien

Desain *interface* dari halaman perbarui obat pasien merupakan halaman dimana *user* farmasi melakukan pengeditan terhadap status berobat dan dapat melihat

data-data atau resep obat pasien yang telah diberikan oleh dokter saat mengisi rekam medis pasien sebelumnya. Pada halaman ini terdapat tombol atau fitur *edit* yang dimana tombol tersebut berfungsi untuk mengedit status obat pasien dan hanya dapat melakukan pengeditan satu kali saja, jika sudah melakukan pengeditan data maka tombol tersebut akan berubah menjadi *disable* atau tidak dapat di pilih. Pada tampilan ini juga farmasi dapat mencetak resep obat dari obat yang Tidak Diambil oleh pasien, sehingga pasien dapat membeli obat tersebut di luar klinik. Desain *interface* dari halaman perbarui obat pasien dapat dilihat pada gambar 5.28.

Logo Klinik Moulya

Menu

**Perbarui Obat Pasien**

Nama : XXXXXXXXXXXXXXXX

Jenis Kelamin : XXXXXXXXXXXXXXXX

Tanggal Lahir : DD MM YYYY

Nomor Telepon : 9999999999999999

Alergi : XXXXXXXXXXXXXXXX

**No. Rekam Medis**

**X-999**

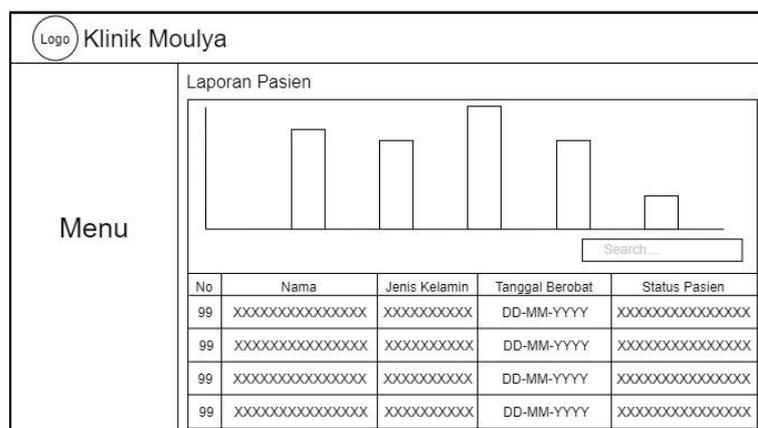
Status Berobat \*  
 -- Pilih Disini --

No	Nama Obat	Jumlah	Dosis	Takaran	Aturan Pakai	Status Obat	Ops
99	XXXXXXXXXX	999999999	999999999	XXXXXXXXXX	XXXXXXXXXX	XXXXX	<input type="radio"/>
99	XXXXXXXXXX	999999999	999999999	XXXXXXXXXX	XXXXXXXXXX	XXXXX	<input type="radio"/>
99	XXXXXXXXXX	999999999	999999999	XXXXXXXXXX	XXXXXXXXXX	XXXXX	<input type="radio"/>

**Gambar 5.28 Perbarui Obat Pasien**

## 14. Laporan Pasien

Desain *interface* dari halaman laporan pasien merupakan halaman yang menampilkan laporan dari pasien-pasien yang berobat ke klinik setiap harinya. Pada halaman laporan pasien ini terdapat grafik yang menunjukkan jumlah pasien yang berobat ke klinik setiap bulannya, grafik ini dapat di gunakan penanggung jawab klinik sebagai bahan acuan kegiatan apa yang akan dilakukan klinik kedepannya untuk menarik minat orang-orang agar melakukan pengobatan dasar di Klinik Moulya. Desain *interface* dari halaman laporan pasien dapat dilihat pada gambar 5.29.



**Gambar 5.29 Laporan Pasien**

### 5.1.3. Evaluasi *Prototyping*

Pada tahap evaluasi ini peneliti melakukan evaluasi dengan cara memberikan *desain interface* ke penanggung jawab klinik

secara langsung yang kemudian peneliti jelaskan fitur-fitur apa saja yang ada beserta kegunaan dari fitur tersebut. Pemberian *desain interface* ini dilakukan untuk mengetahui apakah fungsi-fungsi yang ada pada *prototype* sudah sesuai dan sudah lengkap?. Desain *interface* yang ada saat ini telah dilakukan beberapa kali perombakan. Hal ini disebabkan karena fitur-fitur yang ada pada desain tidak sesuai ataupun belum lengkap dengan kebutuhan di klinik, sehingga terjadilah perombakan desain hingga mendapatkan persetujuan dari penanggung jawab klinik.

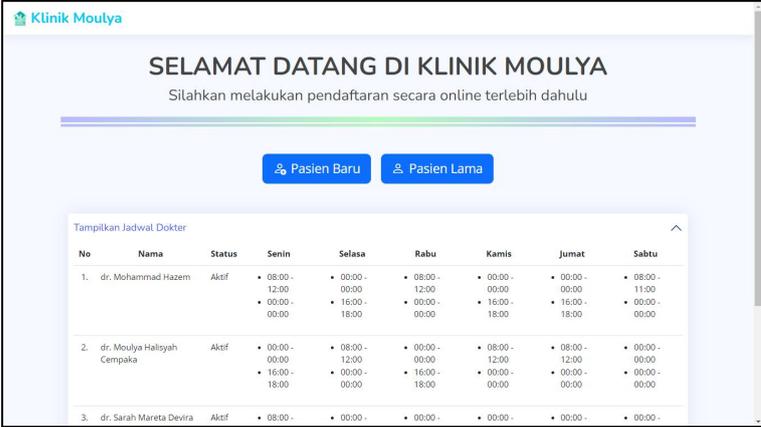
#### **5.1.4. Mengkodekan Sistem**

Pada tahap ini peneliti telah melakukan pengkodean sistem atau pembuatan sistem sesuai dengan hasil dari kesepakatan antara peneliti dan penanggung jawab klinik yang telah dilakukan pada evaluasi *prototyping* sebelumnya. Pada tahap ini peneliti memberikan beberapa hasil dari tampilan sistem saja, yang dapat dilihat sebagai berikut :

##### **1. Hasil Daftar *Online***

Tampilan halaman daftar *online* yaitu tampilan dimana saat pasien memasukkan alamat *website* Klinik Moulya, maka sistem pertama kali akan menampilkan halaman *website* klinik. Halaman ini digunakan untuk melakukan pendaftaran secara *online* baik pasien baru maupun pasien lama. Hasil dari

tampilan halaman daftar *online* yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.30.

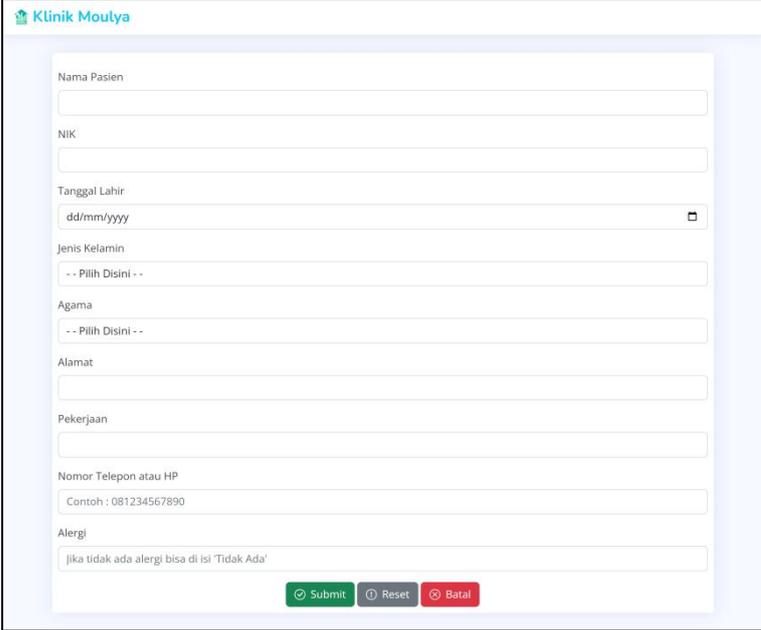


No	Nama	Status	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
1.	dr. Mohammad Hazem	Aktif	• 08:00 - 12:00 • 00:00 - 00:00 • 16:00 - 18:00	• 08:00 - 08:00 • 08:00 - 12:00 • 16:00 - 18:00	• 08:00 - 12:00 • 00:00 - 00:00	• 08:00 - 08:00 • 08:00 - 12:00 • 16:00 - 18:00	• 08:00 - 08:00 • 08:00 - 11:00 • 16:00 - 18:00	• 08:00 - 08:00 • 08:00 - 11:00 • 00:00 - 00:00
2.	dr. Moulya Halisyah Cempaka	Aktif	• 08:00 - 08:00 • 16:00 - 18:00	• 08:00 - 12:00 • 00:00 - 00:00	• 08:00 - 08:00 • 16:00 - 18:00	• 08:00 - 12:00 • 08:00 - 08:00	• 08:00 - 12:00 • 08:00 - 08:00	• 08:00 - 08:00 • 08:00 - 08:00
3.	dr. Sarah Mareta Devira	Aktif	• 08:00 - 08:00	• 08:00 - 08:00	• 08:00 - 08:00	• 08:00 - 08:00	• 08:00 - 08:00	• 08:00 - 08:00

**Gambar 5.30 Hasil Tampilan Daftar *Online***

## 2. Hasil Pasien Baru

Tampilan halaman pasien baru merupakan tampilan yang berisi *form* data diri dimana saat pasien belum terdata sebagai pasien di klinik dan memilih tombol pasien baru di tampilan sebelumnya yaitu tampilan daftar *online* untuk mendaftarkan dirinya di Klinik Moulya. Pada tampilan ini pasien diharuskan mengisi *form* yang telah disediakan atau ditampilkan oleh sistem sesuai dengan data diri pasien. Hasil dari tampilan halaman pasien baru yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.31.

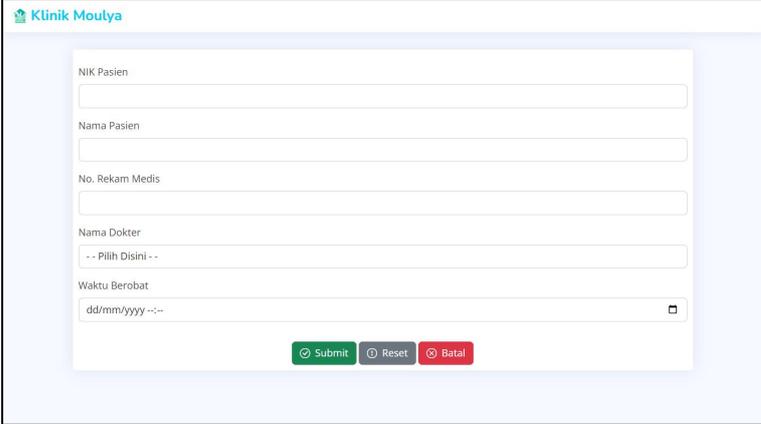


The image shows a web form for patient registration at 'Klinik Mulya'. The form is titled 'Klinik Mulya' in the top left corner. It contains several input fields: 'Nama Pasien' (Patient Name), 'NIK' (National Identity Card Number), 'Tanggal Lahir' (Date of Birth) with a date picker showing 'dd/mm/yyyy', 'Jenis Kelamin' (Gender) with a dropdown menu showing '-- Pilih Disini --', 'Agama' (Religion) with a dropdown menu showing '-- Pilih Disini --', 'Alamat' (Address), 'Pekerjaan' (Occupation), 'Nomor Telepon atau HP' (Phone or Mobile Number) with a placeholder 'Contoh : 081234567890', and 'Alergi' (Allergy) with a placeholder 'Jika tidak ada alergi bisa di isi 'Tidak Ada''. At the bottom of the form, there are three buttons: 'Submit' (green), 'Reset' (grey), and 'Batal' (red).

**Gambar 5.31 Hasil Pasien Baru**

### 3. Hasil Pasien Lama

Tampilan halaman pasien lama merupakan tampilan dimana ketika pasien telah mengirim data pada tampilan sebelumnya yaitu tampilan pasien baru, tampilan *website* akan langsung berpindah ke halaman pasien lama. Pada halaman ini juga jika pada halaman daftar *online* pasien memilih tombol pasien lama, halaman akan menampilkan tampilan halaman pasien lama. Halaman ini berfungsi untuk mendaftarkan dirinya berobat di klinik dengan mengisi NIK, memilih nama dokter, dan memilih waktu berobat. Hasil dari tampilan halaman pasien lama dapat dilihat pada gambar 5.32.



Klinik Moutya

NIK Pasien

Nama Pasien

No. Rekam Medis

Nama Dokter  
-- Pilih Disini --

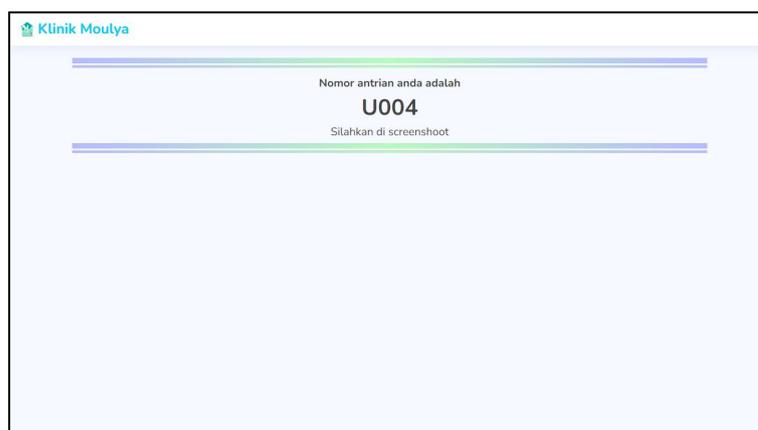
Waktu Berobat  
dd/mm/yyyy --:--

**Gambar 5.32 Hasil Pasien Lama**

#### 4. Hasil Nomor Antrian

Tampilan halaman nomor antrian merupakan tampilan yang akan muncul ketika pasien telah melakukan pengiriman data dari *form* yang ada di halaman pasien lama, maka pasien tersebut akan mendapatkan nomor antrian berobat sesuai dengan poli dokter yang dipilih pada tampilan sebelumnya pada tampilan pasien lama. Jika pasien memilih dokter yang berada dipoli umum maka nomor antrian akan berawalan dengan huruf U, dan jika pasien memilih dokter yang berada di poli gigi maka nomor antrian akan berawalan dengan huruf G. Sebagai contoh pasien ke empat pada hari kamis mendaftarkan diri pada dokter yang berada di poli umum, maka nomor antrian yang didapatkan oleh pasien tersebut adalah U001. Pada saat pasien telah mendapatkan nomor antrian, pasien diharapkan untuk mengingat nomor tersebut ataupun

melakukan *screenshot* agar tidak lupa. Hasil dari tampilan halaman nomor antrian yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.33.

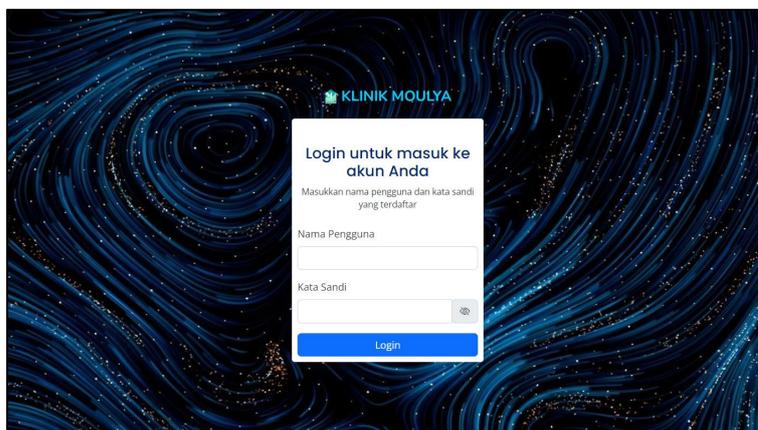


**Gambar 5.33 Hasil Nomor Antrian**

#### 5. Hasil *Login User*

Tampilan halaman *login* merupakan tampilan dimana *user* tata usaha, penanggung jawab, rekam medis, farmasi, dan dokter melakukan *login* atau masuk ke akun masing-masing yang sudah terdaftar di dalam *database*. Pada halaman *login* ini, tampilan yang ditampilkan sistem berupa *form* berisi nama pengguna dan kata sandi pengguna yang harus diisi ketika ingin masuk ke akun. Jika nama pengguna atau kata sandi terdapat kesalahan di salah satu bagiannya, maka sistem akan menampilkan notifikasi salah, sehingga pengguna haruslah memasukkan nama pengguna dan kata sandi pengguna dengan

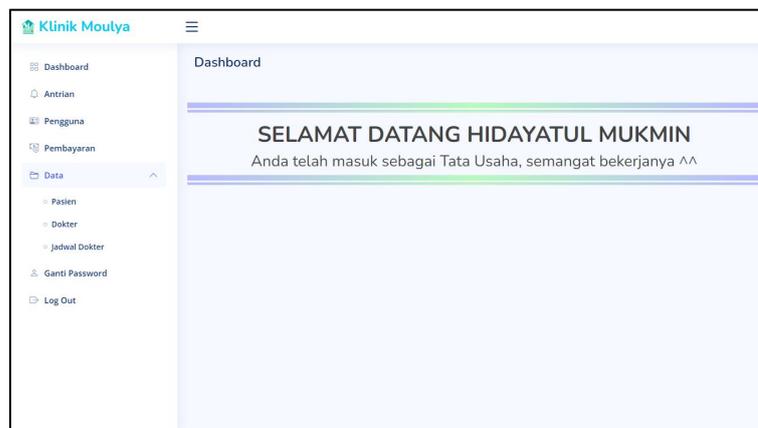
benar. Hasil dari tampilan halaman *login user* yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.34.



**Gambar 5.34 Hasil *Login User***

#### 6. Hasil *Dashboard*

Tampilan halaman *dashboard* merupakan tampilan awal dimana ketika *user* telah berhasil melakukan *login* akun, maka *website* akan berpindah ke tampilan *dashboard*. Pada tampilan ini semua *user* yang berhasil melakukan *login* memiliki tampilan yang sama, namun ada beberapa bagian yang berbeda yaitu nama *user* pada bagian selamat datang, level *user*, dan menu yang berada kiri halaman. Hasil dari tampilan halaman *dashboard* yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.35.



**Gambar 5.35 Hasil *Dashboard***

## 7. Hasil Pemanggilan Pasien

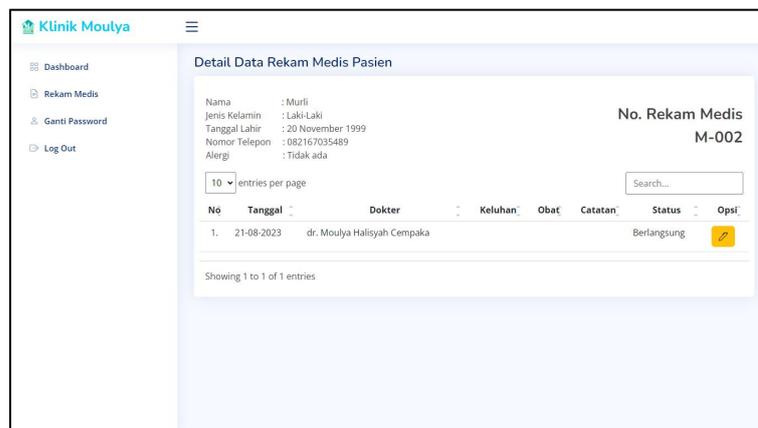
Tampilan halaman hasil pemanggilan pasien merupakan salah satu menu pada *user* tata usaha yang berisikan tampilan data-data pasien yang sedang melakukan antri. Data-data yang ada tampil secara berurut sesuai dengan waktu berobat pasien. Pada tampilan ini terdapat *checkbox* yang berfungsi sebagai pengubah status pasien dari Antri menjadi Berlangsung atau berfungsi sebagai pemanggilan pasien untuk masuk ke dalam ruang pemeriksaan dokter yang dipilih sebelumnya pada halaman pasien lama. Jika tata usaha telah melakukan pemanggilan dengan menekan *checkbox* yang ada pada nama pasien yang akan dipanggil, maka nama pasien tersebut akan hilang dari halaman. Hasil dari tampilan halaman pemanggilan pasien dapat dilihat pada gambar 5.36.

No	Nama	Poli	Nomor Antrian	Status
1.	Indi Febrina Latifah Sidik	Gigi	G001	<input type="checkbox"/>
2.	Criffin Zefana Tari	Umum	U002	<input type="checkbox"/>

**Gambar 5.36 Hasil Pemanggilan Pasien**

## 8. Hasil Rekam Medis (Dokter)

Tampilan halaman hasil rekam medis bagian dokter merupakan tampilan data rekam medis pasien yang akan di periksa oleh dokter sesuai dengan pasien yang dipilih sebelumnya. Pada tampilan ini, terdapat data diri pasien, rekam medis pemeriksaan pasien, dan tombol atau fitur *edit*. Tombol *edit* ini hanya akan muncul dalam jangka waktu hari itu saja. Jika lebih dari hari tanggal pengobatan berlangsung, maka tombol *edit* ini tidak akan muncul. Hal ini dimaksudkan untuk meminimalisir hal-hal yang tidak diinginkan terjadi, seperti manipulasi data rekam medis pasien. Hasil dari tampilan halaman rekam medis pasien bagian dokter dapat dilihat pada gambar 5.37.



**Gambar 5.37 Hasil Rekam Medis (Dokter)**

#### 9. Hasil Isi Rekam Medis (Dokter)

Tampilan halaman hasil isi rekam medis bagian dokter merupakan tampilan dimana dokter mengisi data-data rekam medis pasien sesuai dengan yang dirasakan oleh pasien dan penyakit apa yang di diagnosis oleh dokter. Pada tampilan ini dokter dapat memilih tombol obat pasien untuk menambahkan obat pasien yang datanya akan langsung tampil di dalam halaman tambah isi rekam medis tersebut. Pada bagian data-data obat pasien, terdapat fitur untuk hapus obat pasien. Fitur tersebut dapat melakukan penghapusan data, data obat yang telah dihapus akan kembali kedalam stok obat. Sehingga jumlah stok pada stok obat akan kembali bertambah sesuai dengan jumlah obat yang diberikan dokter pada tampilan obat pasien. Hasil dari tampilan halaman isi rekam medis pasien

bagian dokter yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.38.

The screenshot shows a web interface for 'Klinik Moulya' with a sidebar menu containing 'Dashboard', 'Rekam Medis', 'Ganti Password', and 'Log Out'. The main content area is titled 'Tambah Isi Rekam Medis Pasien' and contains the following information:

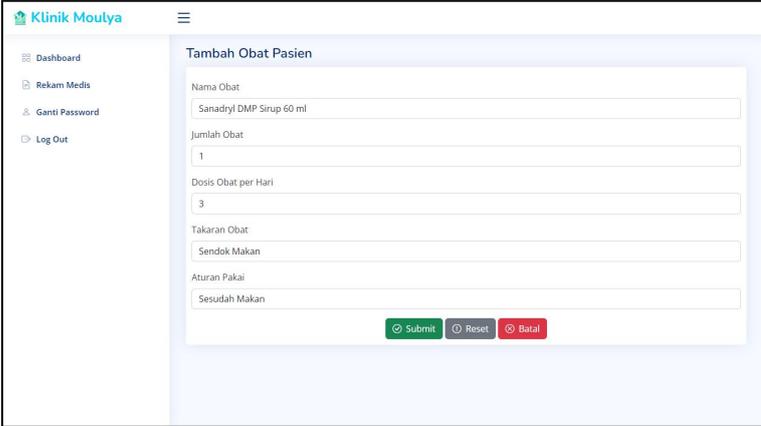
- Patient Details:**
  - Nama: Murli
  - Jenis Kelamin: Laki-Laki
  - Tanggal Lahir: 20 November 1999
  - Nomor Telepon: 082167035489
  - Alergi: Tidak ada
- No. Rekam Medis:** M-002
- Obat Untuk Pasien:** A green button labeled 'Obat Untuk Pasien'.
- Keluhan Pasien:** Tekanan darah tinggi dan batuk berdarah
- Obat Pasien Table:**

No	Nama Obat	Jumlah	Aturan Pakai	Opsi
1.	Amlodipine 10 mg	12	Sesudah Makan	[X]
2.	Sanadryl DMP Sirup 60 ml	1	Sesudah Makan	[X]
- Catatan Pasien:** tidak ada
- Status Berobat:** Rawat Jalan
- Buttons:** Submit (green), Reset (grey), and Batal (red).

**Gambar 5.38 Hasil Isi Rekam Medis (Dokter)**

#### 10. Hasil Tambah Obat Pasien

Tampilan halaman hasil tambah obat pasien merupakan tampilan dimana ketika dokter memilih tombol obat pasien maka tampilan akan menampilkan tampilan tambah obat pasien agar dokter dapat mengisi data-data obat untuk pasien. Ketika dokter mengirimkan data obat, data-data obat yang telah terkirim dapat langsung ditampilkan di halaman sebelumnya yaitu tampilan isi rekam medis bagian dokter. Hasil dari tampilan halaman tambah obat pasien dapat dilihat pada gambar 5.39.

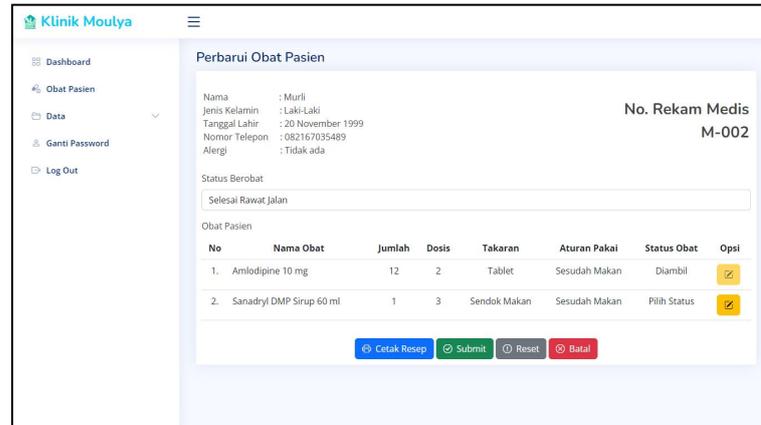


The screenshot shows a web application interface for 'Klinik Moutya'. On the left is a sidebar menu with options: Dashboard, Rekam Medis, Ganti Password, and Log Out. The main content area is titled 'Tambah Obat Pasien' and contains a form with the following fields: 'Nama Obat' (filled with 'Sanadryl DMP Sirup 60 ml'), 'Jumlah Obat' (filled with '1'), 'Dosis Obat per Hari' (filled with '3'), 'Takaran Obat' (filled with 'Sendok Makan'), and 'Aturan Pakai' (filled with 'Setelah Makan'). At the bottom of the form are three buttons: 'Submit' (green), 'Reset' (grey), and 'Batal' (red).

**Gambar 5.39 Hasil Tambah Obat Pasien**

#### 11. Hasil Perbarui Obat Pasien

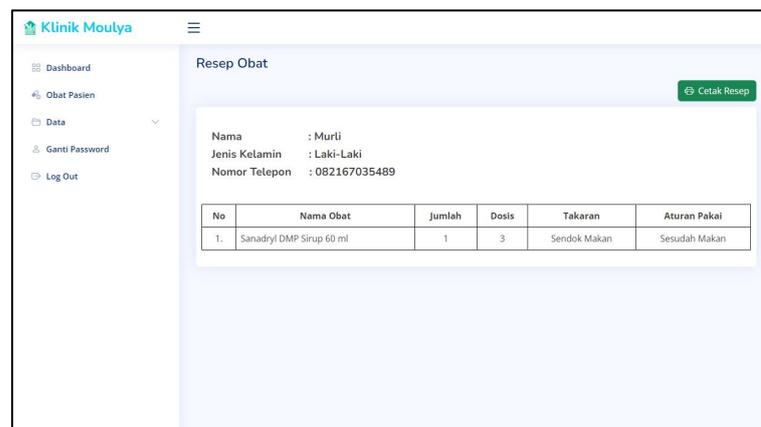
Tampilan hasil perbarui obat pasien merupakan halaman dimana *user* farmasi melakukan pengeditan terhadap status berobat dan dapat melihat data-data obat pasien. Pada halaman ini terdapat tombol *edit* yang dimana tombol tersebut hanya dapat melakukan pengeditan satu kali saja, jika sudah melakukan pengeditan data maka tombol tersebut akan berubah menjadi *disable* atau tidak dapat di pilih kembali. Pada tampilan ini juga farmasi dapat melakukan cetak resep dari obat yang status obatnya telah diubah menjadi Tidak Diambil, sehingga obat yang telah diberikan status Tidak Diambil tersebut dapat dibeli di luar klinik. Hasil dari tampilan halaman perbarui obat pasien yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.40.



**Gambar 5.40 Hasil Perbarui Obat Pasien**

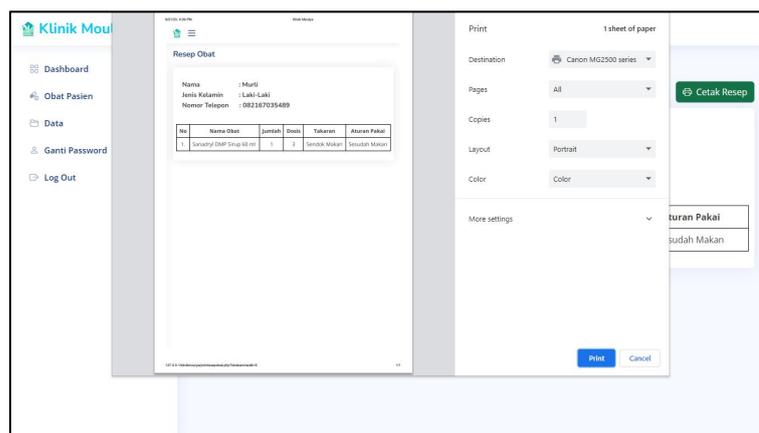
## 12. Hasil Cetak Resep

Tampilan hasil cetak resep merupakan tampilan dimana ketika farmasi memilih tombol cetak resep pada tampilan sebelumnya yaitu tampilan perbarui obat pasien. Tampilan ini hanya menampilkan data-data obat yang memiliki status obat Tidak Diambil saja. Hasil dari tampilan halaman cetak resep dapat dilihat pada gambar 5.41.



**Gambar 5.41 Hasil Cetak Resep**

Setelah dirasa obat yang Tidak Diambil sudah benar, maka farmasi dapat memilih tombol cetak resep yang kedua kalinya. Dimana untuk melakukan pencetakan resep melalui mesin *print*. Memiliki tampilan pada gambar 5.42.



**Gambar 5.42 Lanjutan Hasil Cetak Resep**

### 13. Hasil Tambah Pembayaran Pasien

Tampilan hasil tambah pembayaran pasien merupakan tampilan dimana ketika tata usaha memilih tipe pengobatan yang telah dilakukan di klinik, nominal uang yang harus dibayarkan oleh pasien, memasukkan uang yang diberikan oleh pasien, memberikan kembalian, dan melakukan cetak nota. Cetak nota dapat dilakukan setelah pasien melakukan pembayaran di tata usaha terlebih dahulu. Hasil dari tampilan halaman tambah pembayaran pasien yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.43.

**Klinik Moulya**

**Tambah Pembayaran**

Tipe Pengobatan:

Harga Pengobatan:

No	Tipe Pengobatan	Harga Pengobatan	Opsi
1.	Konsultasi	Rp60.000	<input type="button" value="X"/>
2.	Obat	Rp40.000	<input type="button" value="X"/>
<b>Total Bayar :</b>		Rp100.000	

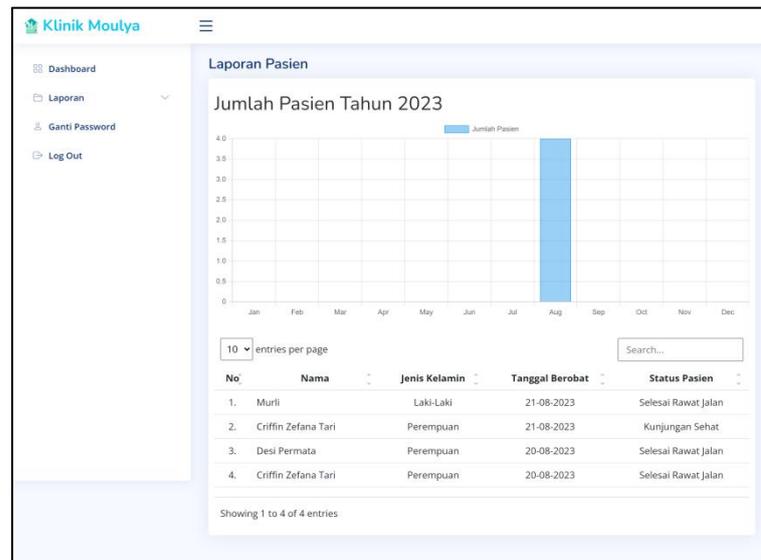
Total Bayar:

Dibayar:  Kembalian:

**Gambar 5.43 Hasil Tambah Pembayaran Pasien**

#### 14. Hasil Laporan Pasien

Tampilan hasil laporan pasien merupakan tampilan dimana penanggung jawab dapat melihat laporan pasien. Pada tampilan ini telah terdapat grafik pasien yang telah melakukan pengobatan di klinik selama satu tahun. Grafik ini dibuat agar penanggung jawab dapat melihat total jumlah pasien yang datang di Klinik Moulya, sehingga grafik tersebut dapat digunakan sebagai bahan acuan kegiatan apa yang akan dilakukan atau dilaksanakan klinik kedepannya untuk menarik minat orang-orang agar melakukan pengobatan dasar di Klinik Moulya. Hasil dari tampilan halaman laporan pasien yang telah dibuat oleh peneliti dapat dilihat pada gambar 5.44.



**Gambar 5.44 Hasil Laporan Pasien**

### 5.1.5. Menguji Sistem

#### 5.1.5.1. *Black-Box Testing*

Pada tahap pengujian sistem ini, peneliti menggunakan *black-box testing* dengan teknik *equivalence partitioning*. Pengujian akan dilakukan pada *user* pasien dan *user* tata usaha yang dapat dilihat pada tabel 5.16.

**Tabel 5.16 Rencana *Test Case* dan Butir Uji**

No	<i>Test Case</i>	Butir Uji
1	Pasien	Halaman Pasien Baru, Halaman Pasien Lama
2	Tata Usaha	Halaman <i>Login</i> , Halaman Tambah Data Pengguna, Halaman <i>Edit</i> Pengguna, Halaman Tambah Bayar, Halaman Pembayaran Pasien, Halaman Tambah Pengobatan.

### 5.1.5.1.1. Skenario Pengujian

Skenario pengujian digunakan untuk mengetahui macam-macam pengisian data pada *form* yang kemudian akan dilakukan pengujian sesuai dengan skenario yang telah ditentukan.

#### 1. Halaman Pasien Baru

Halaman ini diuji karena berupa *form* yang perlu diisi pasien, ketika pasien belum terdaftar di klinik, maka pasien harus melakukan pendaftaran terlebih dahulu. Pendaftaran tersebut haruslah mengisi *form* dengan data diri pasien. Skenario pengujian dari halaman pasien baru dapat dilihat pada tabel 5.17.

**Tabel 5.17 Skenario Pasien Baru**

No	ID	Test Case	Harapan
1	A01	Mengosongkan seluruh data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak akan tersimpan
2	A02	Mengisi data nama pasien, tanggal lahir, dengan NIK yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf!Nama pasien dalam NIK tersebut sudah terdata"
3	A03	Mengisi data nama pasien, tanggal lahir yang sama dengan yang sudah terdata di dalam <i>database</i> namun dengan NIK yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama

No	ID	Test Case	Harapan
4	A04	Mengisi data nama pasien, tanggal lahir yang belum terdata namun dengan NIK yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama
5	A05	Mengisi data nama pasien, tanggal lahir yang belum terdata dengan NIK yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama

## 2. Halaman Pasien Lama

Halaman ini diuji karena berupa *form* yang perlu diisi pasien, ketika pasien telah melakukan pendaftaran atau namanya telah terdata di klinik. Pendaftaran tersebut haruslah mengisi *form* dengan data diri pasien yang telah terdata. Skenario pengujian dari halaman pasien lama dapat dilihat pada tabel 5.18.

**Tabel 5.18 Skenario Pasien Lama**

No	ID	Test Case	Harapan
1	B01	Mengosongkan seluruh data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak akan tersimpan
2	B02	Mengisi data NIK yang sudah terdata	Sistem akan menampilkan pilihan nama pasien yang terdapat pada NIK tersebut
3	B03	Mengisi data NIK yang belum terdata	Sistem tidak akan menampilkan pilihan nama pasien dengan NIK yang belum terdata
4	B04	Memilih nama pasien yang sama di dalam <i>database</i> namun dengan NIK yang berbeda	Sistem akan menampilkan data yang sesuai dengan denga NIK

### 3. Halaman *Login*

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha, penanggung jawab, rekam medis, farmasi, dan dokter. Saat melakukan *login* akun, *user* haruslah mengisi *form* dengan nama lengkap dan kata sandi yang telah terdata. Skenario pengujian dari halaman *login* dapat dilihat pada tabel 5.19.

**Tabel 5.19 Skenario Halaman *Login***

No	ID	Test Case	Harapan
1	C01	Mengosongkan semua isian data	Sistem akan menolak akses <i>login</i> dengan menampilkan <i>alert</i> pada data yang kosong
2	C02	Mengisi nama pengguna dan mengosongkan kata sandi	Sistem akan menolak akses <i>login</i> dengan menampilkan <i>alert</i> pada data yang kosong
3	C03	Mengisi data nama pengguna dan kata sandi dengan salah satu data yang salah	Sistem akan menampilkan notifikasi "Nama atau Kata Sandi yang anda masukkan salah"
4	C04	Mengisi data nama pengguna dan kata sandi dengan isian yang benar	Sistem akan menerima akses <i>login</i> dan sistem akan masuk ke halaman <i>dashboard</i>

### 4. Halaman Tambah Data Pengguna

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha, agar *user* yang lainnya dapat melakukan *login* akun. Halaman ini berupa pengisian *form* dengan nama pengguna,

*password*, dan memilih level *user*. Skenario pengujian dari halaman tambah data pengguna dapat dilihat pada tabel 5.20.

**Tabel 5.20 Skenario Tambah Data Pengguna**

No	ID	Test Case	Harapan
1	D01	Mengosongkan semua isian data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak tersimpan
2	D02	Mengisi nama pengguna dan mengosongkan <i>password</i>	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan <i>alert</i> pada data yang kosong
3	D03	Mengisi nama pengguna dengan nama yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"
4	D04	Mengisi nama pengguna dengan nama yang belum terdata namun dengan <i>password</i> yang sudah terdata	Sistem akan menyimpan data ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna
5	D05	Mengisi nama pengguna yang sudah terdata dengan memilih <i>level user</i> yang berbeda	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"
6	D06	Mengisi nama pengguna dan <i>password</i> yang belum terdata	Sistem akan menyimpan data ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna

#### 5. Halaman *Edit* Pengguna

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha. Halaman ini merupakan halaman melakukan pengeditan data

dengan mengisi *form* yang berisi nama pengguna, dan *password*. Skenario pengujian dari halaman *edit* pengguna dapat dilihat pada tabel 5.21.

**Tabel 5.21 Skenario *Edit* Pengguna**

No	ID	Test Case	Harapan
1	E01	Menampilkan isian data sesuai dengan <i>iduser</i> yang dipilih	Sistem akan menampilkan isian form data yang sesuai dengan <i>iduser</i> yang dipilih
2	E02	Mengganti nama pengguna dengan nama pengguna yang telah terdata dengan <i>password</i> yang sama	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"
3	E03	Mengganti nama pengguna dengan nama pengguna yang telah terdata dengan <i>password</i> yang berbeda	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"
4	E04	Mengganti <i>password</i> namun tidak melakukan pergantian pada data nama pengguna	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"
5	E05	Mengganti nama pengguna dengan yang belum terdata dengan <i>password</i> yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna
6	E06	Mengganti nama pengguna dengan yang belum terdata dengan <i>password</i> yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna

## 6. Halaman Tambah Bayar

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha. Halaman ini berupa *form* untuk memasukkan tipe pengobatan atau tindakan yang dilakukan oleh dokter. Skenario pengujian dari halaman tambah bayar dapat dilihat pada tabel 5.22.

**Tabel 5.22 Skenario Tambah Bayar**

No	ID	Test Case	Harapan
1	F01	Memilih salah satu nama pengobatan	Sistem akan menampilkan otomatis harga dari pengobatan tersebut
2	F02	Memilih nama pengobatan yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"
3	F03	Memilih nama pengobatan yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan tetap berada di tampilan tersebut

## 7. Halaman Pembayaran Pasien

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha. Halaman ini berupa *form* yang diisi saat pasien menyerahkan uang yang akan dibayarkan. Skenario pengujian dari halaman pembayaran pasien dapat dilihat pada tabel 5.23.

**Tabel 5.23 Skenario Pembayaran Pasien**

No	ID	Test Case	Harapan
1	G01	Mengisi data dibayar dengan jumlah kurang dari total bayar	Sistem akan menampilkan nominal yang minus pada data kembalian
2	G02	Mengisi data dibayar dengan jumlah lebih dari total bayar	Sistem akan menampilkan nominal yang harus di kembalikan
3	G03	Mengisi data di bayar dengan jumlah yang sama dari total bayar	sistem akan menampilkan angka "0"
4	G04	Mengubah data ketika telah melakukan pembayaran	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Anda telah melakukan pembayaran"

## 8. Halaman Tambah Pengobatan

Halaman ini diuji karena berupa *form* yang perlu diisi oleh *user* tata usaha. Halaman ini berupa pengisian *form* dengan nama pengobatan, dan harga pengobatan. Skenario pengujian dari halaman tambah pengobatan dapat dilihat pada tabel 5.24.

**Tabel 5.24 Skenario Tambah Pengobatan**

No	ID	Test Case	Harapan
1	H01	Mengosongkan semua isian data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak akan tersimpan
2	H02	Mengisi nama pengobatan dengan harga pengobatan yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"

No	ID	Test Case	Harapan
3	H03	Mengisi nama pengobatan yang belum terdata dengan harga pengobatan yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan tambah pembayaran sesuai dengan idrekammedik yang dipilih sebelumnya
4	H04	Mengisi nama pengobatan yang sudah terdata dengan harga pengobatan yang belum ada	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"

#### 5.1.5.1.2. Hasil Pengujian

Hasil pengujian adalah hasil dari pengujian yang telah dilakukan peneliti sesuai dengan skenario yang telah dibuat sebelumnya.

##### 1. Hasil Pengujian Pasien Baru

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman pasien baru dapat dilihat pada tabel 5.25.

**Tabel 5.25 Hasil Pengujian Pasien Baru**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	A01	Mengosongkan seluruh data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak akan tersimpan	Sistem menolak pengiriman data	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
2	A02	Mengisi data nama pasien, tanggal lahir, dengan NIK yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama pasien dalam NIK tersebut sudah terdata"	Sistem menolak pengiriman data dan memunculkan notifikasi	<i>Valid</i>
3	A03	Mengisi data nama pasien, tanggal lahir yang sama dengan yang sudah terdata di dalam <i>database</i> namun dengan NIK yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan daftar pasien lama	<i>Valid</i>
4	A04	Mengisi data nama pasien, tanggal lahir yang belum terdata namun dengan NIK yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan daftar pasien lama	<i>Valid</i>
5	A05	Mengisi data nama pasien, tanggal lahir yang belum terdata dengan NIK yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman daftar pasien lama	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan daftar pasien lama	<i>Valid</i>

Pada tabel 5.25 dapat dijelaskan bahwa terdapat

5 butir pengujian dengan hasil yang didapatkan

dalam pengujian halaman pasien baru adalah 5 butir pengujian memiliki hasil yang *valid*.

## 2. Hasil Pengujian Pasien Lama

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman pasien lama dapat dilihat pada tabel 5.26.

**Tabel 5.26 Hasil Pengujian Pasien Lama**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	B01	Mengosongkan seluruh data	Sistem akan menolak pengiriman data ke database dan data tidak akan tersimpan	Sistem menolak pengiriman data	<i>Valid</i>
2	B02	Mengisi data NIK yang sudah terdata	Sistem akan menampilkan pilihan nama pasien yang terdapat pada NIK tersebut	Sistem menampilkan nama-nama pasien	<i>Valid</i>
3	B03	Mengisi data NIK yang belum terdata	Sistem tidak akan menampilkan pilihan nama pasien dengan NIK yang belum terdata	Sistem tidak menampilkan nama-nama pasien	<i>Valid</i>
4	B04	Memilih nama pasien yang sama di dalam <i>database</i> namun dengan NIK yang berbeda	Sistem akan menampilkan data yang sesuai dengan denga NIK	Sistem menampilkan data yang sesuai dengan denga NIK	<i>Valid</i>

Pada tabel 5.26 dapat dijelaskan bahwa terdapat 4 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman pasien lama adalah 4 butir pengujian memiliki hasil yang *valid*.

### 3. Hasil Pengujian *Login*

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman *login* dapat dilihat pada tabel 5.27.

**Tabel 5.27 Hasil Pengujian *Login***

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	C01	Mengosongkan semua isian data	Sistem akan menolak akses <i>login</i> dengan menampilkan <i>alert</i> pada data yang kosong	Sistem menolak akses <i>login</i> dan menampilkan <i>alert</i> dibagian yang kosong	<i>Valid</i>
2	C02	Mengisi nama pengguna dan mengosongkan kata sandi	Sistem akan menolak akses <i>login</i> dengan menampilkan <i>alert</i> pada data yang kosong	Sistem menolak akses <i>login</i> dan menampilkan <i>alert</i> dibagian yang kosong	<i>Valid</i>
3	C03	Mengisi data nama pengguna dan kata sandi dengan salah satu data yang salah	Sistem akan menampilkan notifikasi "Nama atau Kata Sandi yang anda masukkan salah"	Sistem menolak akses <i>login</i> dan menampilkan notifikasi	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
4	C04	Mengisi data nama pengguna dan kata sandi dengan isian yang benar	Sistem akan menerima akses <i>login</i> dan sistem akan masuk ke halaman <i>dashboard</i>	Sistem menerima akses <i>login</i> dan tampilan sistem berubah ke tampilan <i>dashboard</i>	<i>Valid</i>

Pada tabel 5.27 dapat dijelaskan bahwa terdapat

4 butir pengujian dengan hasil yang didapatkan

dalam pengujian halaman *login* adalah 4 butir

pengujian memiliki hasil yang *valid*.

#### 4. Hasil Pengujian Tambah Data Pengguna

Pengujian ini dilakukan sesuai dengan skenario

yang telah dirancang sebelumnya. Hasil yang

didapat dari pengujian pada halaman tambah

data pengguna dapat dilihat pada tabel 5.28.

**Tabel 5.28 Hasil Pengujian Tambah Data Pengguna**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	D01	Mengosongkan semua isian data	Sistem akan menolak pengiriman data ke <i>database</i> dan data tidak tersimpan	Sistem menolak pengiriman data	<i>Valid</i>
2	D02	Mengisi nama pengguna dan mengosongkan <i>password</i>	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan <i>alert</i> pada data	Sistem menolak pengiriman data dan menampilkan <i>alert</i> pada bagian yang	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
			yang kosong	kosong	
3	D03	Mengisi nama pengguna dengan nama yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>
4	D04	Mengisi nama pengguna dengan nama yang belum terdata namun dengan <i>password</i> yang sudah terdata	Sistem akan menyimpan data ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan data pengguna	<i>Valid</i>
5	D05	Mengisi nama pengguna yang sudah terdata dengan memilih <i>level user</i> yang berbeda	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>
6	D06	Mengisi nama pengguna dan <i>password</i> yang belum terdata	Sistem akan menyimpan data ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan data pengguna	<i>Valid</i>

Pada tabel 5.28 dapat dijelaskan bahwa terdapat 6 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman tambah data pengguna adalah 6 butir pengujian memiliki hasil yang *valid*.

#### 5. Hasil Pengujian *Edit* Pengguna

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman *edit* pengguna dapat dilihat pada tabel 5.29.

**Tabel 5.29 Hasil Pengujian *Edit* Pengguna**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	E01	Menampilkan isian data sesuai dengan <i>iduser</i> yang dipilih	Sistem akan menampilkan isian form data yang sesuai dengan <i>iduser</i> yang dipilih	Sistem menampilkan data sesuai dengan <i>id</i> yang dipilih	<i>Valid</i>
2	E02	Mengganti nama pengguna dengan nama pengguna yang telah terdata dengan <i>password</i> yang sama	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
3	E03	Mengganti nama pengguna dengan nama pengguna yang telah terdata dengan <i>password</i> yang berbeda	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Nama Pengguna yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>
4	E04	Mengganti <i>password</i> namun tidak melakukan pergantian pada data nama pengguna	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Invalid</i>
5	E05	Mengganti nama pengguna dengan yang belum terdata dengan <i>password</i> yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan data pengguna	<i>Valid</i>
6	E06	Mengganti nama pengguna dengan yang belum terdata dengan <i>password</i> yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan data pengguna	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan data pengguna	<i>Valid</i>

Pada tabel 5.29 dapat dijelaskan bahwa terdapat 6 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman *edit* pengguna adalah 5 butir pengujian memiliki hasil yang *valid* dan 1 butir pengujian memiliki hasil yang *invalid*.

#### 6. Hasil Pengujian Tambah Bayar

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman tambah bayar dapat dilihat pada tabel 5.30.

**Tabel 5.30 Hasil Pengujian Tambah Bayar**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	F01	Memilih salah satu nama pengobatan	Sistem akan menampilkan otomatis harga dari pengobatan tersebut	Sistem menampilkan harga pengobatan sesuai nama pengobatan yang dipilih	<i>Valid</i>
2	F02	Memilih nama pengobatan yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
3	F03	Memilih nama pengobatan yang belum terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan tetap berada di tampilan tersebut	Sistem menerima data untuk disimpan dan tampilan tidak berubah	<i>Valid</i>

Pada tabel 5.30 dapat dijelaskan bahwa terdapat 3 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman tambah bayar adalah 3 butir pengujian memiliki hasil yang *valid*.

#### 7. Hasil Pengujian Pembayaran Pasien

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman pembayaran pasien dapat dilihat pada tabel 5.31.

**Tabel 5.31 Hasil Pengujian Pembayaran Pasien**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	G01	Mengisi data dibayar dengan jumlah kurang dari total bayar	Sistem akan menampilkan nominal yang minus pada data kembalian	Pada bagian kembalian, sistem menampilkan nominal yang minus	<i>Valid</i>
2	G02	Mengisi data dibayar dengan jumlah lebih dari total bayar	Sistem akan menampilkan nominal yang harus di kembalikan	Sistem menampilkan nominal kembalian	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
3	G03	Mengisi data di bayar dengan jumlah yang sama dari total bayar	sistem akan menampilkan angka "0"	Sistem menampilkan angka "0"	<i>Valid</i>
4	G04	Mengubah data ketika telah melakukan pembayaran	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! Anda telah melakukan pembayaran"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>

Pada tabel 5.31 dapat dijelaskan bahwa terdapat 4 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman pembayaran pasien adalah 4 butir pengujian memiliki hasil yang *valid*.

#### 8. Hasil Pengujian Tambah Pengobatan

Pengujian ini dilakukan sesuai dengan skenario yang telah dirancang sebelumnya. Hasil yang didapat dari pengujian pada halaman tambah pengobatan dapat dilihat pada tabel 5.32.

**Tabel 5.32 Hasil Pengujian Tambah Pengobatan**

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
1	H01	Mengosongkan semua isian data	Sistem akan menolak pengiriman	Sistem menolak pengiriman	<i>Valid</i>

No	ID	Test Case	Harapan	Hasil Pengujian	Kesimpulan
			data ke <i>database</i> dan data tidak akan tersimpan	data dan menampilkan <i>alert</i> pada bagian yang kosong	
2	H02	Mengisi nama pengobatan dengan harga pengobatan yang sudah terdata	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>
3	H03	Mengisi nama pengobatan yang belum terdata dengan harga pengobatan yang sudah terdata	Sistem akan menyimpan data tersebut ke dalam <i>database</i> dan sistem akan masuk ke halaman tampilan tambah pembayaran sesuai dengan idrekammedik yang dipilih sebelumnya	Sistem menerima data untuk disimpan dan tampilan sistem berubah ke tampilan tambah pembayaran sesuai idrekammedik sebelumnya	<i>Valid</i>
4	H04	Mengisi nama pengobatan yang sudah terdata dengan harga pengobatan yang belum ada	Sistem akan menolak pengiriman data ke <i>database</i> dan menampilkan notifikasi "Maaf! data yang anda masukkan sudah terdata"	Sistem menolak pengiriman data dan menampilkan notifikasi	<i>Valid</i>

Pada tabel 5.32 dapat dijelaskan bahwa terdapat 4 butir pengujian dengan hasil yang didapatkan dalam pengujian halaman Tambah Pengobatan adalah 4 butir pengujian memiliki hasil yang *valid*.

Setelah pengujian sistem dilakukan, dapat diketahui total keseluruhan butir pengujian adalah 36 butir. Hasil dari pengujian tersebut terdapat 1 butir pengujian yang memiliki hasil *invalid*, sehingga presentase tingkat keberhasilan sistem yang didapatkan adalah 97,22%.

#### **5.1.6. Penggunaan Sistem**

Pada tahap ini *website* telah di *publish* dan dapat dilihat secara online. Jika masuk sebagai pasien dapat langsung ke halaman untuk tampilan daftar *online* pada *link* <http://klinikmoulya.000webhostapp.com/tampilandaftaronline.php> dan jika ingin masuk ke halaman sebagai tata usaha, penanggung jawab, farmasi, dokter, dan rekam medis dapat langsung ke halaman *login* terlebih dahulu pada *link* <http://klinikmoulya.000webhostapp.com/login.php>.

## BAB VI

### PENUTUP

#### 6.1. Kesimpulan

Berdasarkan hasil dari penelitian, disimpulkan bahwa sistem informasi *management* ini diharapkan dapat mempermudah pegawai dalam proses mengelola dan menampung data, serta dapat menghemat tempat dan biaya.

Hasil pengujian sistem menggunakan *black-box testing* dengan teknik *equivalence partitioning* telah menghasilkan 35 butir telah sesuai dan 1 butir tidak sesuai dengan harapan. Sehingga, dari hasil pengujian tersebut dapat diketahui tingkat presentase keberhasilan dari sistem informasi *management* ini adalah 97,22%.

#### 6.2. Saran

Sistem informasi *management* ini masih belumlah sempurna dan masih memerlukan pengembangan lebih lanjut. Saran yang ditunjukkan ke dalam penelitian ini, antara lain :

1. Sistem dapat diperbaiki pada bagian-bagian yang masih terdapat *error*.
2. Melakukan pemeliharaan maupun pengembangan sistem, agar dapat terus digunakan dalam jangka panjang dengan fitur-fitur yang lebih baik lagi.

3. Menyesuaikan ukuran kertas pada saat melakukan pencetakan baik pencetakan nota maupun resep, karena pada sistem saat ini penggunaan kertas yang digunakan untuk pencetakan masih menggunakan kertas yang berukuran A4.
4. Pada proses pemanggilan nomor antrian, dapat dilakukan pengembangan dengan menambahkan fitur pemanggilan nomor antrian yang menggunakan audio.

## DAFTAR PUSTAKA

- Agustin, H. 2018. *Sistem informasi manajemen menurut prespektif islam. Jurnal Tabarru': Islamic Banking and Finance*, 1(1), 63-70.
- Andaru, A. 2018. *Pengertian Database Secara Umum*. OSF Prepr, 2.
- Fauzi, J. R. 2020. *Algoritma Dan Flowchart Dalam Menyelesaikan Suatu Masalah*. Jurnal Teknik Informatika.
- Harsuda, M. B. 2018. *Pembuatan Sistem Informasi Pembelajaran E-Learning Menggunakan Moodle Pada Smpn 2 Kecamatan Suliki (Doctoral dissertation, Universitas Widyatama)*.
- Ichsanudin, M. N., Yusuf, M., & Suraya, S. 2022. *Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula*. Storage: Jurnal Ilmiah Teknik dan Ilmu Komputer, 1(2), 1-8.
- Indrasari, Y. 2020. *Efisiensi Saluran Distribusi Pemasaran Kopi Rakyat Di Desa Gending Waluh Kecamatansempol (Ijen) Bondowoso*. Jurnal Manajemen Pemasaran, 14(1), 44-50.
- Jaya, T. S. 2018. *Pengujian Aplikasi Dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)*. Jurnal Informatika: Jurnal Pengembangan IT, 3(1), 45-48.

- Kurniawan, T. B. 2020. *Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman pada Cafeteria No Caffè di Tanjung Balai Karimun Menggunakan Bahasa Pemograman PHP Dan MySQL*. Jurnal Tikar, 1(2), 192-206.
- Latukolan, M. L. A., Arwan, A., & Ananta, M. T. 2019. *Pengembangan Sistem Pemetaan Otomatis Entity Relationship Diagram Ke Dalam Database*. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 3(4), 4058–4065.
- Mahardini, M. M. A. 2020. *Analisis Situasi Penggunaan Google Classroom Pada Pembelajaran Daring Fisika*. Jurnal Pendidikan Fisika, 8(2), 215-224.
- Meilani, Y. I. 2019. *Implementasi Metode Proyotyping Dan Twitter Bootstrap Responsive Pada Sistem Pembelajaran Bagi Siswa/I Sma Di Palembang*. Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer, 10(2), 753-762.
- Nisaunnajah, V. I. 2021. *Analisis Metode Pembelajaran Eksperimen Terhadap Hasil Belajar Siswa Sekolah Dasar (Analisis Deskriptif Kualitatif dengan Teknik Studi Pustaka)* (Doctoral dissertation, FKIP UNPAS).
- Oktaviani, I., Mustofa, K., & Maragawati, S. 2020. *Sistem Informasi Manajemen Pelayanan pada Klinik Pratama Dengan Metode CRM*. Jurnal Elektronika Listrik dan Teknologi Informasi Terapan, 2(1), 14-20.

- Rochman, A., Hanafri, M. I., & Wandira, A. 2020. *Implementasi Website Profil SMK Kartini Sebagai Media Promosi dan Informasi Berbasis Open Source*. Academic Journal of Computer Science Research, 2(1).
- Ropianto, Indra. Mulyana. Muhammad. Indra. 2020. *Entity Relationship Diagram & Praktik DBMS*.
- Tanjung, R., Supriani, Y., Mayasari, A., & Arifudin, O. 2022. *Manajemen Mutu Dalam Penyelenggaraan Pendidikan*. Jurnal Pendidikan Glasser, 6(1), 29-36.
- Wijaya, A., Damayanti, D., & An'ars, M. G. 2022. *Rancang Bangun Sistem Informasi Manajemen Kepegawaian (Simpeg) Berbasis Web (Studi Kasus: Pt Sembilan Hakim Nusantara)*. Jurnal Teknologi dan Sistem Informasi, 3(1), 77-82.