

**APLIKASI PENGOBATAN TRADISIONAL MENGGUNAKAN *BUSINESS*
MODEL CANVAS (BMC) PADA *STARTUP* MEDIS REBORN
BERBASIS WEB**

SKRIPSI

**DINARY DWIHATAMI
NIM. 0702171045**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN**

2021

**APLIKASI PENGOBATAN TRADISIONAL MENGGUNAKAN *BUSINESS*
MODEL CANVAS (BMC) PADA *STARTUP* MEDIS REBORN
BERBASIS WEB**

SKRIPSI

Diajukan untuk Memenuhi Syarat Mencapai Gelar Sarjana Komputer

DINARY DWIHATAMI

NIM. 0702171045



PROGRAM STUDI SISTEM INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SUMATERA UTARA

MEDAN

2021

PERSETUJUAN SKRIPSI

Hal : Surat Persetujuan Skripsi
Lamp : -

Kepada Yth.
Dekan Fakultas Sains dan Teknologi
Universitas Islam Negeri Medan Sumatera Utara Medan

Assalamu 'alaikum Wr, Wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan, maka kami selaku pembimbing berpendapat bahwa skripsi saudara:

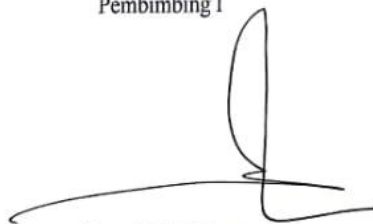
Nama	: Dinary Dwihatami
Nomor Induk Mahasiswa	: 0702171045
Program Studi	: Sistem Informasi
Judul	: Aplikasi Pengobatan Tradisional Menggunakan <i>Business Model Canvas (BMC) Pada Startup Medis Reborn Berbasis</i>

Dapat disetujui untuk segera *dimunaqasyahkan*. Atas perhatiannya kami ucapkan terimakasih.

Medan, 07 September 2021 M
12 Shafar 1443 H

Komisi Pembimbing

Pembimbing I



Suendri, M. Kom
NIP.198712082015031003

Pembimbing II



Raissa Amanda Putri, S. Kom, M.TI
NIP.198611292019032009

SURAT PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Dinary Dwihatami

Nomor Induk Mahasiswa : 0702173197

Program Studi : Sistem Informasi

Judul : Aplikasi Pengobatan Tradisional Menggunakan
Business Model Canvas (BMC) Pada *Startup* Medis
Reborn Berbasis

Menyatakan bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya. Apabila dikemudian hari ditemukan plagiat dalam skripsi ini saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dari sanksi lainnya sesuai dengan peraturan yang berlaku.

Medan, 07 September 2021



Dinary Dwihatami
NIM.0702171045



KEMENTERIAN AGAMA REPUBLIK INDONESIA
UNIVERSITAS ISLAM NEGERI SUMATERA UTARA
MEDAN FAKULTAS SAINS DAN TEKNOLOGI
Jl. Lap. Golf, Desa Durian Jangak, Kec. Pancur Batu 20353
Telp. (+6261) 4536090, Fax. (+6261) 6615683
Url: <http://saintek.uinsu.ac.id>, E-mail: saintek@uinsu.ac.id

PENGESAHAN SKRIPSI

Nomor : B.178/ST/ST.V.2/PP.01.1/11/2021

Judul : Aplikasi Pengobatan Tradisional Menggunakan
Business Model Canvas (BMC) Pada *Startup* Medis
Reborn Berbasis Web

Nama : Dinary Dwihatami

Nomor Induk Mahasiswa : 0702171045

Program Studi : Sistem Informasi

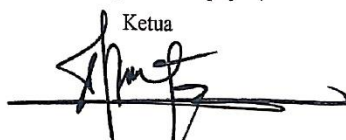
Fakultas : Sains dan Teknologi

Telah dipertahankan di hadapan Dewan Penguji Skripsi Program Studi Sistem
Informasi Fakultas Sains dan Teknologi UIN Sumatera Utara Medan Dan
Dinyatakan LULUS.

Pada Hari/Tanggal : 14 September 2021

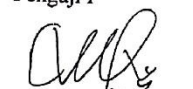
Tim Ujian Munaqasyah,

Ketua



Samsudin S.T., M.Kom
NIP. 197612272011011002

Dewan Penguji,


Penguji I


Ali Ikhwani, M.Kom
NIB. 1100000109


Penguji II


Dr. M. Fakhriza, S.T., M.Kom
NIB. 1100000115

Penguji III


Suendri, M.Kom
NIP. 198712082015031003

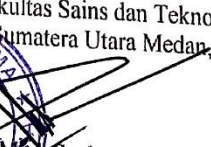
Penguji IV


Raissa Amanda P., S.Kom., M.TI
NIP. 198907102018012002

Mengesahkan,

Dekan Fakultas Sains dan Teknologi
UIN Sumatera Utara Medan,




M. A. Syahnan, M. A
NIP. 196609051991031002

ABSTRAK

Pelayanan kesehatan pada klinik pengobatan tradisional memiliki kendala yaitu tidak adanya sistem pemesanan jadwal pengobatan yang membuat pelanggan harus antri untuk melakukan pengobatan serta minimnya informasi mengenai lokasi klinik. Aplikasi pengobatan tradisional adalah sebuah *flatform* online yang menghubungkan klinik pengobatan tradisional dengan pelanggan dalam satu tempat yaitu menggunakan *website* sehingga pelanggan dapat melakukan pemesanan jadwal pengobatan tradisional secara online. Sistem pemesanan online ini memberikan penjadwalan pengobatan serta pilihan informasi lokasi klinik terdekat. Tujuan dari penelitian ini adalah untuk membangun aplikasi pengobatan tradisional pada *startup* Medis Reborn berbasis *web* yang dilengkapi pemodelan konsep bisnis yang jelas yaitu menggunakan *Business Model Canvas* (BMC) sehingga bisnis lebih terarah dan mampu bersaing dengan *startup* lainnya.

Kata Kunci: Pengobatan tradisional, *flatform* Medis Reborn, *Business Model Canvas* (BMC), *website*

ABSTRACT

Health services in traditional medicine clinics have obstacles, namely the absence of a treatment schedule ordering system that makes customers have to queue for treatment and lack of information about the location of the clinic. Traditional medicine application is an online platform that connects traditional medicine clinics with customers in one place that uses a website so that customers can order traditional medicine schedules online. This online booking system provides treatment scheduling as well as a choice of information on the location of the nearest clinic. The goal of the study is to build a traditional medicine application at web-based Reborn Medical startup that features clear business concept modeling that uses Business Model Canvas (BMC) so that businesses are more targeted and able to compete with other startups.

Key words: Tradisional Medicine, Medis Reborn platform, Business Model Canvas (BMC), website

KATA PENGANTAR



Assalamu'alaikum Warahmatullahi Wabarakatuh.

Penulis mengucapkan Alhamdulillah Rabbil 'Alamin sebagai rasa syukur kepada Allah SWT atas karunia dan kasih sayang nya sehingga penulis mampu menyelesaikan skripsi yang berjudul "Aplikasi Pengobatan Tradisional Menggunakan *Business Model Canvas* (BMC) Pada *Startup* Medis Reborn Berbasis *Web* ". Tak lupa penulis ucapkan Allahumma Sholli'ala Sayyidina Muhammad Wa'ala Ali Sayyidina Muhammad kepada Nabi besar Muhammad SAW semoga kita diakui sebagai ummat Nya di akhirat kelak.

Pada kesempatan ini penulis mengucapkan terima kasih kepada semua pihak yang telah terlibat dalam skripsi ini. Terima kasih tersebut penulis ucapkan kepada:

1. Bapak Prof. Dr. Syahrin, M.A. selaku Rektor Universitas Islam Negeri Sumatera Utara.
2. Bapak Dr. M. Syahnan, M.A. selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
3. Bapak Samsudin, S.T., M.Kom. selaku Ketua Prodi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
4. Bapak Suendri, M.Kom. selaku Dosen Pembimbing I yang telah memberikan bimbingan selama menyelesaikan proposal skripsi.
5. Ibu Raissa Amanda Putri S. Kom., M. TI selaku Dosen Pembimbing II yang telah memberikan bimbingan selama menyelesaikan proosal skripsi.
6. Seluruh Dosen Prodi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara.
7. Ayah saya Drs. Basra Pasaribu, Ibunda saya Lila Masnun. S. Pd, Abang saya Fajry Juangga S.T, dan Adik Saya Raja Akbar yang selalu mencurahkan do'a, nasihat, motivasi, kasih sayang dan harapan serta dukungan.

8. Seluruh teman-teman seperjuangan Sistem Informasi 5 yang tak lelah selalu memberikan dukungan nya.

Demikianlah semoga skripsi ini bermanfaat bagi penulis sendiri, mahasiswa/i Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara serta seluruh masyarakat lain yang membacanya.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Medan, 01 Oktober 2021

Penulis

Dinary Dwihatami

DAFTAR ISI

	HALAMAN
ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	v
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	3
1.5.1. Manfaat Bagi Penulis.....	3
1.5.2. Manfaat Bagi Universitas Islam Negeri Sumatera Utara.....	3
1.5.3. Manfaat Bagi Pusat Bekam-Ruqyah Cabang Medan	3
BAB II LANDASAN TEORI	4
2.1. Konsep Sistem.....	4
2.1.1. Karakteristik Sistem.....	5
2.1.2. Pengertian Subsistem	7
2.2. Pengertian Aplikasi	8
2.2.1. Aplikasi Dekstop.....	8
2.2.2. Aplikasi Web.....	9
2.2.3. Aplikasi <i>Mobile</i>	9
2.3. Pengobatan Tradisional.....	10
2.3.1. Bekam	11
2.3.2. Pijat Refleksi	13

2.3.3. Akupunktur	14
2.4. <i>Business Model Canvas</i> (BMC).....	15
2.4.1. <i>Customer Segments</i>	16
2.4.2. <i>Value Propositions</i>	18
2.4.3. <i>Channels</i>	19
2.4.4. <i>Customer Relationships</i>	20
2.4.5. <i>Revenue Streams</i>	21
2.4.6. <i>Key Resources</i>	22
2.4.7. <i>Key Activities</i>	23
2.4.8. <i>Key Partnerships</i>	23
2.4.9. <i>Cost Structure</i>	24
2.5. Contoh Implementasi <i>Business Model Canvas</i> (BMC)	24
2.6. <i>Unified Modelling Language</i> (UML).....	26
2.6.1. <i>Use Case Diagram</i>	26
2.6.2. <i>Activity Diagram</i>	28
2.6.3. <i>Class Diagram</i>	29
2.6.4. <i>Sequences Diagram</i>	30
2.7. Metode <i>Waterfall</i>	33
2.8. <i>Floyd Warshall</i>	34
2.9. <i>World Wide Web</i> Atau WEB.....	34
2.9.1. HTML	34
2.9.2. PHP	35
2.10. <i>MySQL</i>	36
2.11. <i>XAMPP</i>	36
2.12. <i>Framework Laravel</i>	37
2.12.1 <i>Laravel 8</i>	41
2.13. <i>Penelitian Terdahulu</i>	42
BAB III METODE PENELITIAN	43
3.1. Tempat Penelitian	43
3.2. Waktu Penelitian.....	43

3.3. Kebutuhan Sistem	45
3.3.1. Perangkat Lunak	45
3.3.2. Perangkat Keras	45
3.4. Cara Kerja	46
3.4.1. Metode Penelitian Kualitatif	46
3.4.2. Sumber Data	46
3.4.3. Metode Pengembangan Sistem	47
3.4.4. Analisis BMC	48
3.4.5. Kerangka Berpikir	49
BAB IV HASIL DAN PEMBAHASAN	50
4.1. <i>Requirement</i>	50
4.1.1.1. Profil Pusat Bekam Ruqyah Cabang Medan	50
4.1.1.2. Sejarah Pusat Bekam Ruqyah	50
4.1.1.3. Analisa Sistem Berjalan	51
4.1.1.4. Analisa Sistem Usulan	53
4.1.1.5. <i>Business Model Canvas</i> (BMC)	54
4.2. <i>Design</i>	58
4.2.1. Desain Proses	58
4.2.1.1. <i>Use Case Diagram</i>	58
4.2.1.2. <i>Activity Diagram</i> Registrasi Pasien	59
4.2.1.3. <i>Activity Diagram</i> Login	60
4.2.1.4. <i>Activity Diagram</i> Pilih Pengobatan	60
4.2.1.5. <i>Activity Diagram</i> Pilih Klinik	61
4.2.1.6. <i>Activity Diagram</i> Form Jadwal Pengobatan	62
4.2.1.7. <i>Activity Diagram</i> Pembayaran	62
4.2.1.8. <i>Activity Diagram</i> Tambah Pengobatan	63
4.2.1.9. <i>Activity Diagram</i> Tambah Klinik	64
4.2.1.10. <i>Activity Diagram</i> Data Pasien	64
4.2.1.11. <i>Activity Diagram</i> Konfirmasi Pesanan	65
4.2.1.12. <i>Activity Diagram</i> Laporan Keuangan Medis Reborn	65

4.2.1.13. <i>Activity Diagram</i> Pesanan Bagi Klinik	66
4.2.1.14. <i>Activity Diagram</i> Laporan Keuangan Klinik	67
4.2.1.15. <i>Sequence Diagram</i> Registrasi Pasien.....	67
4.2.1.16. <i>Sequence Diagram</i> Login.....	68
4.2.1.17. <i>Sequence Diagram</i> Pilih Pengobatan.....	68
4.2.1.18. <i>Sequence Diagram</i> Pilih Klinik	69
4.2.1.19. <i>Sequence Diagram</i> Form Jadwal Pengobatan.....	69
4.2.1.20. <i>Sequence Diagram</i> Pembayaran	70
4.2.1.21. <i>Sequence Diagram</i> Tambah Pengobatan	70
4.2.1.22. <i>Sequence Diagram</i> Tambah Klinik.....	71
4.2.1.23. <i>Sequence Diagram</i> Data Pasien	71
4.2.1.24. <i>Sequence Diagram</i> Konfirmasi Pesanan	72
4.2.1.25. <i>Sequence Diagram</i> Laporan Keuangan Medis Reborn...	72
4.2.1.26. <i>Sequence Diagram</i> Pesanan Bagi Klinik	73
4.2.1.27. <i>Sequence Diagram</i> Laporan Keuangan Klinik	73
4.2.2. Desain <i>Database</i>	74
4.2.2.1. <i>Class Diagram</i>	74
4.2.2.2. Tabel Perancangan <i>Database</i>	75
4.2.3. Desain <i>Interface</i> (Antarmuka).....	78
4.3. <i>Implementation</i>	83
4.4. <i>Verification</i>	93
4.5. <i>Maintenance</i> (Pemeliharaan)	99
 BAB V KESIMPULAN DAN SARAN	100
5.1. Kesimpulan	100
5.2. Saran	100
DAFTAR PUSTAKA	101

LAMPIRAN

DAFTAR GAMBAR

GAMBAR	JUDUL GAMBAR	HALAMAN
2.1.	Model Sistem.....	4
2.2.	Karakteristik Sistem	6
2.3.	Gambaran Subsistem dalam Sistem	7
2.4	Kalkulator.....	9
2.5	Bekam.....	12
2.6	Pijat Refleksi	14
2.7	Akupunktur.....	15
2.8.	<i>Business Model Canvas</i> (BMC)	18
2.9.	BMC Depot Selaris	24
2.10.	BMC Pinisi Backpacker	25
2.11.	BMC UD. Cahaya	25
2.12.	Logo HTML	35
2.13.	Logo PHP	36
2.14.	Logo <i>MySQL</i>	36
2.15.	Logo <i>Xampp</i>	37
2.16.	PHP <i>Artisan</i> Laravel.....	39
2.17.	<i>Controller</i> Laravel.....	39
2.18.	<i>Blade Templating</i>	40
2.19.	<i>Middleware</i>	40
2.20	Laravel	41
3.1	Metode <i>Waterfall</i>	52
3.2	Kerangka Berpikir	53
4.1	Gedung Sewa Pusat Bekam Ruqyah Cabang Medan.....	51
4.2	Analisa Sistem Berjalan	52
4.3	Analisa Sistem Usulan	53

4.4	<i>Business Model Canvas (BMC)</i>	54
4.5	<i>Use Case Diagram</i> Medis Reborn	58
4.6	<i>Activity Diagram</i> Registrasi Pasien.....	59
4.7	<i>Activity Diagram</i> Login.....	60
4.8	<i>Activity Diagram</i> Pilih Pengobatan	61
4.9	<i>Activity Diagram</i> Pilih Klinik	61
4.10	<i>Activity Diagram</i> Form Jadwal Pengobatan.....	62
4.11	<i>Activity Diagram</i> Pembayaran	63
4.12	<i>Activity Diagram</i> Tambah Pengobatan	63
4.13	<i>Activity Diagram</i> Tambah Klinik.....	64
4.14	<i>Activity Diagram</i> Data Pasien	64
4.15	<i>Activity Diagram</i> Konfirmasi Pesanan	65
4.16	<i>Activity Diagram</i> Laporan Keuangan Medis Reborn.....	66
4.17	<i>Activity Diagram</i> Pesanan Bagi Klinik	66
4.18	<i>Activity Diagram</i> Laporan Keuangan Klinik	67
4.19	<i>Sequence Diagram</i> Registrasi Pasien.....	67
4.20	<i>Sequence Diagram</i> Login.....	68
4.21	<i>Sequence Diagram</i> Pilih Pengobatan	68
4.22	<i>Sequence Diagram</i> Pilih Klinik	69
4.23	<i>Sequence Diagram</i> Form Jadwal Pengobatan.....	69
4.24	<i>Sequence Diagram</i> Pembayaran.....	70
4.25	<i>Sequence Diagram</i> Tambah Pengobatan.....	70
4.26	<i>Sequence Diagram</i> Tambah Klinik	71
4.27	<i>Sequence Diagram</i> Data Pasien	71
4.28	<i>Sequence Diagram</i> Konfirmasi Pesanan	72
4.39	<i>Sequence Diagram</i> Laporan Keuangan Medis Reborn	72
4.30	<i>Sequence Diagram</i> Pesanan Bagi Klinik	73
4.31	<i>Sequence Diagram</i> Laporan Keuangan Klinik.....	73

4.32	<i>Class Diagram</i>	74
4.33	<i>User Interface</i> Halaman Utama.....	78
4.34	<i>User Interface</i> Register	79
4.35	<i>User Interface</i> Login	79
4.36	<i>User Interface</i> Pilih Klinik.....	80
4.37	<i>User Interface</i> Form Jadwal Pengobatan	80
4.38	<i>User Interface</i> Invoice Bukti Pembayaran	81
4.39	<i>User Interface</i> Bukti Pembayaran Berhasil.....	81
4.40	<i>User Interface</i> Dashboard Klinik	82
4.41	<i>User Interface</i> Dashboard Admin Medis Reborn.....	82
4.42	<i>Implementasi</i> Halaman Utama Medis Reborn	83
4.43	<i>Implementasi</i> Halaman Register.....	84
4.44	<i>Implementasi</i> Login	84
4.45	<i>Implementasi</i> Pilih Pengobatan	85
4.46	<i>Implementasi</i> Pilih Klinik.....	85
4.47	<i>Implementasi</i> Form Jadwal Pengobatan.....	86
4.48	<i>Implementasi</i> Invoice Pembayaran.....	86
4.49	<i>Implementasi</i> Pembayaran Berhasil	87
4.50	<i>Implementasi</i> Profile User	87
4.51	<i>Implementasi</i> Halaman Dashboard Admin Medis Reborn	88
4.52	<i>Implementasi</i> List Pengobatan.....	88
4.53	<i>Implementasi</i> Tambah Pengobatan.....	89
4.54	<i>Implementasi</i> List Klinik	89
4.55	<i>Implementasi</i> Tambah Klinik	90
4.56	<i>Implementasi</i> Data Pasien	90
4.57	<i>Implementasi</i> List Pesanan	91
4.58	<i>Implementasi</i> Halaman Data Konfirmasi Pesanan	91
4.59	<i>Implementasi</i> Halaman Laporan Keuangan Medis Reborn	92

4.60	<i>Implementasi Dashboard Klinik</i>	92
4.61	<i>Implementasi Halaman Laporan Keuangan Klinik</i>	93

DAFTAR TABEL

TABEL	JUDUL TABEL	HALAMAN
2.1.	Simbol-Simbol <i>Use Case Diagram</i>	28
2.2.	Simbol-Simbol <i>Activity Diagram</i>	29
2.3.	Simbol-Simbol <i>Class Diagram</i>	31
2.4.	Simbol-Simbol <i>Sequences Diagram</i>	32
2.5.	Penelitian Terdahulu	44
3.1	Jadwal Penelitian.....	47
4.1	<i>Users</i>	75
4.2	<i>Clinics</i>	75
4.3	<i>Therapies</i>	76
4.4	<i>Orders</i>	76
4.5	<i>Therapy List</i>	77
4.6	Pengujian Registrasi	94
4.7	Pengujian <i>Login</i>	94
4.8	Pengujian Pemesanan	95
4.9	Pengujian <i>Profile User</i>	96
4.10	Pengujian Klinik.....	96
4.10	Pengujian Admin Medis Reborn	97
4.11	Pengujian Kelola Terapi Oleh Admin Medis Reborn	97
4.12	Pengujian Kelola Klinik Oleh Admin Medis Reborn	98

DAFTAR LAMPIRAN

No	Judul Lampiran
1	Surat Balasan Penelitian
2	Lembar Validasi Wawancara
3	Formulir Pengujian Sistem
4	<i>Source Code</i>

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi informasi berkembang pesat seiring dengan kebutuhan akan informasi yang cepat dan akurat. Salah satu kemajuan teknologi informasi adalah *WEB* atau *World Wide Web* yang dapat diakses menggunakan internet. Teknologi internet sudah menjadi bagian dari kehidupan manusia tak terkecuali bidang kesehatan khususnya pengobatan tradisional. Selain itu, adanya teknologi internet juga dirasakan dalam dunia bisnis yaitu *startup*. *Startup* di Indonesia berkembang pesat ditandai dengan munculnya banyak *founder* (pemilik) *startup* yang baru setiap tahun bahkan setiap bulan. *Startup* hadir untuk mengatasi berbagai macam kendala yang dihadapi masyarakat sehingga membuat mekanisme kerja menjadi efektif dan efisien.

Pusat Bekam-Ruqyah (PBR) adalah salah satu klinik pengobatan tradisional di kota Medan yang berlokasi di jalan Prof. H. M. Yamin nomor 305. Klinik tersebut terdiri tiga pelayanan kesehatan, yaitu bekam, ruqyah dan refleksi. Berdasarkan observasi penulis, pelayanan klinik masih dilakukan secara manual. Masyarakat mendapatkan informasi klinik dari lembaran brosur. Minimnya informasi membuat masyarakat kesulitan mendapatkan informasi lokasi klinik terdekat, jadwal buka dan tutup nya klinik, serta detail harga pengobatan. Selain itu, tidak adanya sistem pemesanan jadwal untuk melakukan pengobatan membuat pasien atau *customer* harus menunggu atau antri mengingat pengobatan tradisional memerlukan waktu yang cukup lama.

Permasalahan yang telah dijelaskan diatas melatar belakangi terbentuknya *startup* medis reborn. *Startup* merupakan sebuah bisnis yang memiliki model bisnis berbeda dengan bisnis lainnya. Model bisnis memiliki berbagai macam kanvas, salah satunya adalah *Business Model Canvas* (BMC). *Business Model Canvas* (BMC) adalah salah satu metode yang digunakan dalam strategi bisnis *startup*. Pada *startup* medis reborn, penulis menawarkan solusi dengan cara menghubungkan klinik pengobatan tradisional dengan pelanggan dalam satu

tempat yaitu menggunakan *website* medis reborn sehingga pasien atau *customer* dapat melakukan pemesanan pengobatan tradisional secara online. Inovasi teknologi dan peluang berkembangnya *startup* medis reborn ini dapat terlaksana dengan adanya pemodelan konsep bisnis yang jelas yaitu menggunakan *Business Model Canvas* (BMC) sehingga bisnis lebih terarah dan mampu bersaing dengan *startup* lainnya.

Pada penelitian terdahulu yang dibuat oleh Andi Suprianto dan Asri Amaliza Fathia Matsea dalam jurnalnya yang berjudul “Rancang Bangun Aplikasi Pendaftaran Pasien Online Dan Pemeriksaan Dokter Di Klinik Pengobatan Berbasis Web”. Penelitian tersebut membahas pendaftaran pasien untuk berobat ke klinik secara online. Dengan adanya sistem tersebut, pasien tidak harus mengantri dan menunggu lama. Selain itu, memudahkan pegawai klinik dalam melayani pendaftaran pasien serta meningkatkan efisiensi waktu dalam hal pelayanan terhadap pasien.

Berdasarkan uraian diatas, penulis akan membangun sebuah sistem yang dipaparkan dalam bentuk skripsi dengan judul **“Aplikasi Pengobatan Tradisional Menggunakan *Business Model Canvas* (BMC) Pada *Startup* Medis Reborn Berbasis Web”**. Penulis berharap dengan adanya sistem ini, proses pemesanan klinik pengobatan tradisional pada *startup* medis reborn dapat terlaksana sesuai dengan model bisnis *Business Model Canvas* (BMC) yang dibuat.

1.2 Rumusan Masalah

Berdasarkan latar belakang pada uraian sebelumnya, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimanakah merancang aplikasi pemesanan pengobatan di klinik pengobatan tradisional berbasis *web*?
2. Bagaimanakah rancangan *Business Model Canvas* (BMC) pada *startup* medis reborn ?

1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah :

1. Aplikasi hanya membahas tentang pemesanan klinik pengobatan tradisional.
2. Aplikasi menggunakan bahasa pemrograman *HTML* dan *PHP* versi 8.0 dengan *Framework Laravel* serta *MySQL* sebagai databasenya. Perangkat lunak pendukung yang digunakan yakni, *Visual Studio Code* dan *Xampp* 7.3.27.
3. Sistem terdiri dari data pelanggan, data klinik, data layanan, laporan pemesanan pelanggan dan laporan arus pendapatan medis reborn.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah diatas, terdapat tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Untuk merancang aplikasi pengobatan di klinik pengobatan tradisional berbasis *web*.
2. Untuk merancang *Business Model Canvas* (BMC) pada *startup* medis reborn.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari pembuatan sistem ini adalah sebagai berikut:

1. Bagi Penulis
 - a. Untuk memenuhi salah satu syarat kelulusan strata satu (1) Program Studi Sistem Informasi Fakultas Sains dan Teknologi.
 - b. Dapat mengimplementasikan solusi yang ditawarkan penulis dengan baik.
2. Bagi Universitas Islam Negeri Sumatera Utara
 - a. Sebagai bahan rujukan bagi peneliti selanjutnya.

- b. Sebagai bahan evaluasi untuk mengetahui kemampuan mahasiswa dalam menerapkan ilmu yang diperoleh selama perkuliahan.

3. Bagi Pusat Bekam-Ruqyah Cabang Medan

- a. Memberikan informasi mengenai klinik pengobatan tradisional.
- b. Mempermudah pelanggan dalam melakukan pemesanan pengobatan tradisional.
- c. Memperoleh sistem informasi yang dapat dimanfaatkan dalam proses pengobatan tradisional.

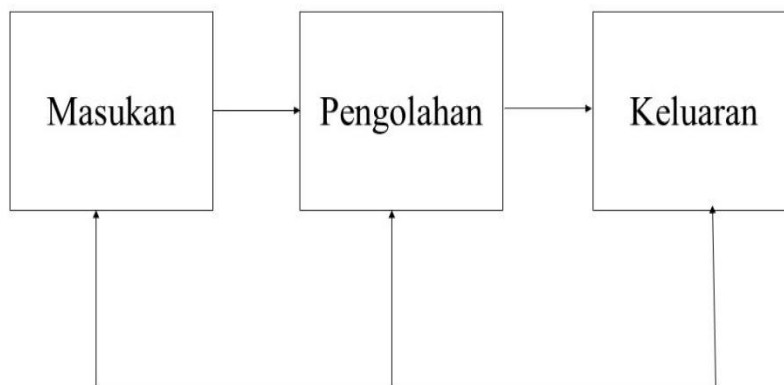
BAB II

LANDASAN TEORI

2.1 Konsep Sistem

Istilah sistem mengacu pada prosedur yang saling berhubungan dan berkumpul bersama-sama bertujuan untuk melakukan kegiatan (Hutahaen, 2014). Sistem pada dasarnya adalah kumpulan komponen dari prosedur yang berkaitan membentuk suatu jaringan kerja untuk mencapai suatu tujuan (Oktafianto, 2016). Menurut (Putri, 2018), sistem lebih menekankan kepada prosedur yang digunakan pada suatu sistem.

Dari beberapa definisi sistem diatas, dapat disimpulkan bahwa sistem adalah sekumpulan elemen dari prosedur yang saling berinteraksi untuk mencapai suatu tujuan tertentu. Setiap sistem merupakan bagian dari sistem lain yang lebih kecil yang disebut sebagai subsistem. Sebagai contoh sistem indera pada manusia yang terdiri dari kulit, hidung, mata, lidah, dan telinga yang masing-masing memiliki fungsi tertentu dan bekerja sesuai fungsinya untuk mencapai tujuan.



Gambar 2.1 Model Sistem

Dari gambar diatas, model sistem terdiri dari 4 unsur subsistem, yaitu :

1. Masukan
2. Pengolahan
3. Keluaran

4. Umpan balik

2.1.1 Karakteristik Sistem

Sebuah sistem dapat dikatakan menjadi sebuah sistem yang baik karena memiliki karakteristik atau mencirikan sifat-sifat tertentu. Adapun karakteristik yang dimaksud adalah sebagai berikut:

1. Komponen (*Component*)

Suatu sistem terdiri dari berbagai komponen yang bekerja sama dalam satu kesatuan. Komponen sistem terdiri dari subsistem memiliki fungsi tertentu.

2. Batasan Sistem (*Boundary*)

Ruang lingkup merupakan wilayah yang membatasi suatu sistem dengan sistem lainnya. Batasan sistem ini adalah suatu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan Luar Sistem (*Environtment*)

Lingkungan Luar Sistem (*Environtment*) merupakan luar batas sistem yang mempengaruhi operasi suatu sistem. Kelangsungan hidup dari sistem harus dapat dikendalikan agar tidak merugikan dan keuntungan dari lingkungan harus dijaga.

4. Penghubung Sistem (*Interface*)

Penghubung sistem adalah media penghubung sistem. Dengan adanya media penghubung ini, sumber daya dapat mengalir dari satu subsistem ke subsistem lainnya. Keluaran dari subsistem akan menjadi masukan untuk subsistem lain melalui penghubung.

5. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan kedalam sistem, dapat berupa perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Perawatan atau *maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi sedangkan masukan sinyal atau *signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh pada sistem komputer, program disebut sebagai *maintenance input* untuk mengoperasikan komputer. Sedangkan untuk pengolahan informasi digunakan data sebagai *signal input*.

6. Keluaran Sistem (*Output*)

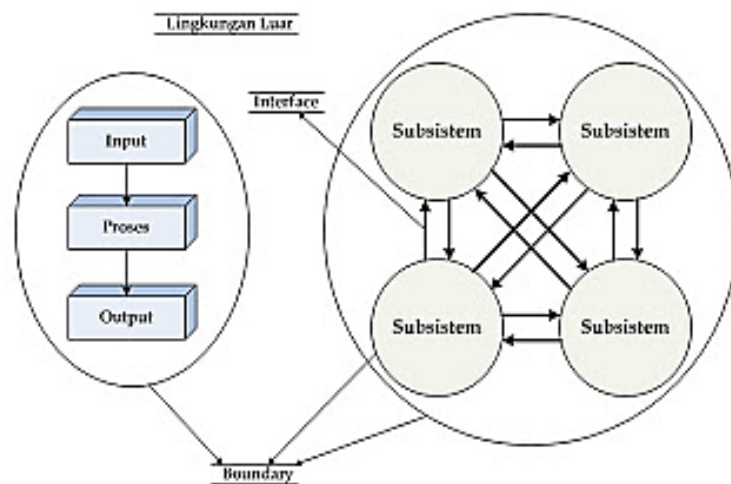
Keluaran sistem adalah hasil dari masukan yang diklasifikasikan menjadi *output* yang berguna. Keluaran ini merupakan masukan bagi subsistem lain. Seperti contoh dalam sistem informasi, keluaran yang dihasilkan adalah informasi, dimana informasi ini digunakan sebagai masukan dalam mengambil keputusan.

7. Pengolah Sistem (*Process*)

Suatu sistem pasti memiliki proses yang mengubah masukan (*input*) menjadi keluaran (*output*). Contohnya adalah sistem akuntansi yang terdiri dari proses pengolahan data transaksi menjadi berbagai macam laporan yang dibutuhkan.

8. Sasaran Sistem (*Objective*)

Suatu sistem mempunyai tujuan yang bersifat deterministik dan pasti. Operasi sistem tidak berguna apabila tidak memiliki tujuan atau sasaran. Dikatakan suatu sistem berhasil adalah jika mengenai sasaran yang direncanakan (Sutabri, 2012).



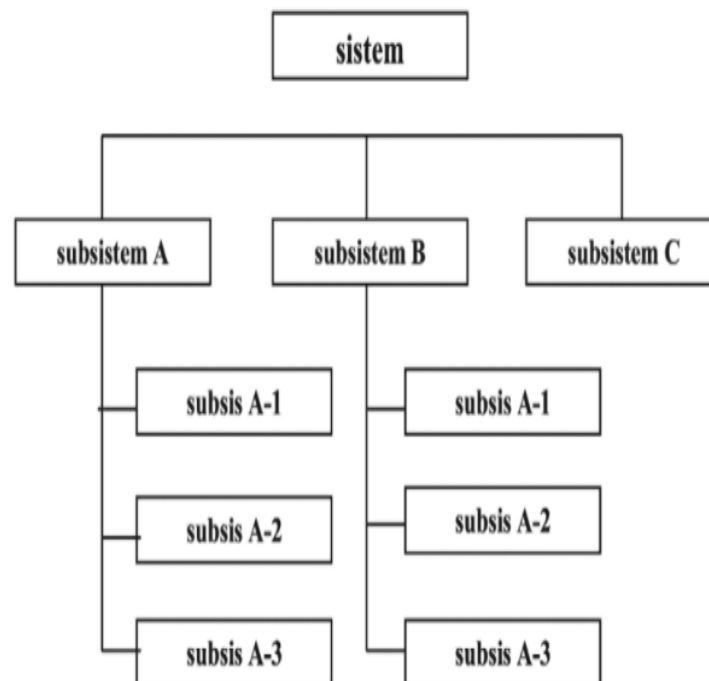
Gambar 2.2 Karakteristik Sistem (Sumber: (Sutabri, 2012))

2.1.2 Pengertian Subsistem

Subsistem adalah bagian kecil dari suatu sistem. Suatu sistem merupakan kesatuan utuh dari beberapa subsistem yang saling terkait untuk mencapai tujuan.

Sebagai contoh, apabila sebuah mobil dibeli tanpa roda maka mobil tidak akan dapat berfungsi.

Konsep subsistem banyak digunakan dalam manajemen proyek dan perancangan sistem. Apabila sistem mempunyai batas dan penghubung yang jelas maka perubahan dapat dilakukan dengan mudah serta subsistem tersebut disimpan dalam proses yang lebih besar (Oktafianto, 2016). Lihat bagan berikut:



Gambar 2.3 Gambaran Subsistem dalam Sistem (Sumber: (Oktafianto, 2016))

2.2 Pengertian Aplikasi

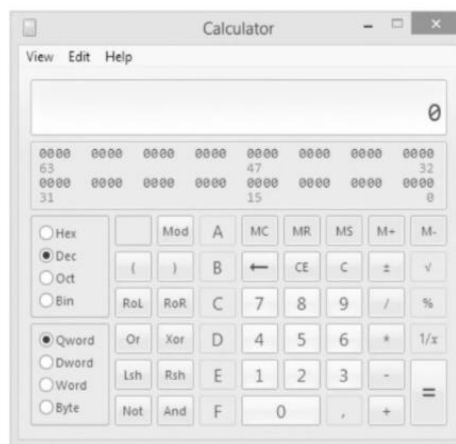
Aplikasi adalah suatu program untuk menjalankan perintah-perintah dari pengguna sesuai dengan tujuan pembuatan aplikasi tersebut. Aplikasi diartikan sebagai pemecahan masalah dengan salah satu teknik pemrosesan data aplikasi berpacu pada sebuah komputansi yang diinginkan maupun pemrosesan data yang diharapkan (Abdurahman & Riswaya, 2014).

Application atau dalam bahasa Indonesia “aplikasi” adalah penerapan penerapan dari program komputer yang bermanfaat untuk pengolahan data sehingga dapat melakukan perintah tertentu sesuai tujuan pembuatan aplikasi tersebut. Aplikasi dapat memecahkan permasalahan secara lebih akurat sehingga dapat membantu pekerjaan manusia menjadi lebih cepat dan tepat. Salah satu contoh dari aplikasi adalah aplikasi dekstop seperti *Microsoft Office*.

2.2.1 Aplikasi Dekstop

Aplikasi perangkat lunak dimulai dengan aplikasi desktop yang hanya bisa digunakan pada mesin *standalone* sejak tahun 1981. Pemrograman desktop merupakan pemrograman berbasis desktop yang dikerjakan dengan memanipulasi elemen-elemen visual pada komputer yang beroperasi pada personal komputer atau melalui jaringan. Pemrograman berbasis dekstop terbagi 3, yaitu *Java*, *Delphi*, dan *Net* (Saputra, 2017).

Menurut (Mardainis & Khusaeri A, 2017), aplikasi desktop adalah suatu program yang dapat beroperasi tanpa menggunakan jaringan internet yang biasanya dibuat dalam satu lokasi area kerja. Secara sederhana, aplikasi dekstop merupakan program yang terinstal di dalam *operating system*, dalam hal ini *Ms Windows*. Contoh dari aplikasi berbasis desktop yang paling sederhana adalah kalkulator (Supriyadi, 2020).



Gambar 2.4 Kalkulator

2.2.2 Aplikasi Web

Aplikasi *web* adalah aplikasi yang tidak memerlukan instalasi di setiap komputer karena aplikasi tersebut berada pada suatu *server* yang terhubung melalui jaringan. Contoh dari jenis aplikasi ini adalah situs *web* (Solichin, 2016).

Pemrograman *web* mengacu pada menulis kode dalam pengembangan *web*, meliputi konten *web*, *clien web*, *script server*, dan keamanan jaringan. Pengembangan aplikasi *web* dibuat dengan arsitektur klien-server dan digunakan pada sebuah *web browser* pada komputer sebagai klien antarmuka (Saputra, 2017).

Kelebihan aplikasi *Web* atau *WebBase*:

1. Akses terhadap informasi mudah.
2. *Setup server* menjadi lebih mudah.
3. Informasi mudah didistribusikan.
4. Informasi disajikan oleh *browser* dan bersifat bebas *platform* (Rerung, 2018) .

2.2.3 Aplikasi Mobile

Aplikasi berbasis *mobile* adalah aplikasi yang dijalankan hanya pada *mobile* seperti *handphone*. Aplikasi *mobile* dapat digunakan kapanpun dan dimanapun. Contoh aplikasi *mobile* yang sering digunakan adalah musik, *browser* dan lain-lain. Pengembangan secara *mobile* yaitu J2ME (*Java 2 Micro Edition*) merupakan program yang sering digunakan pada *PDA*, *pager*, dan pada *smartphone* sampai saat ini (Komputer, 2014).

Aplikasi *mobile* pada saat ini terus mengalami perkembangan yang sangat pesat. Terdapat 2 *platform* yang masih stabil hingga saat ini dalam mengembangkan aplikasi *mobile*, yaitu *IOS* dan *Android*.

2.3 Pengobatan Tradisional

Pada era modern ini, pengobatan tradisional mulai populer kembali di masyarakat. Hal ini disebabkan oleh banyak faktor seperti faktor sugesti,

pelayanan yang cepat, efektif dan murah. Pengobatan tradisional saat ini telah berkembang pesat di seluruh dunia. Menurut undang-undang ‘pasal 1 ayat 7 undang-undang No. 23 tahun 1992 tentang kesehatan, disebutkan bahwa: “Pengobatan tradisional adalah pengobatan dan atau perawatan dengan cara, obat, dan pengobatannya, yang mengacu pada pengalaman dan keterampilan turun temurun dan diterapkan sesuai dengan norma yang berlaku dalam masyarakat”’.

Pengobatan tradisional merupakan salah satu dari tujuh belas macam penyelenggaraan upaya kesehatan yang diatur dalam Undang Undang ‘Nomor 36 Tahun 2009 tentang Kesehatan. Pelayanan kesehatan tradisional merupakan pengobatan dan/atau perawatan dengan cara dan obat yang mengacu pada pengalaman dan keterampilan turun-temurun secara empiris yang dapat dipertanggungjawabkan dan diterapkan sesuai dengan norma yang berlaku di masyarakat. Dalam Pasal 59 ayat (2) disebutkan bahwa pelayanan kesehatan tradisional tentunya dibina dan diawasi oleh pemerintah agar dapat dipertanggungjawabkan manfaat dan keamanannya serta tidak bertentangan dengan norma agama dan kebudayaan masyarakat. Hal senada diatur dalam Keputusan Menteri Kesehatan No. 1076/Menkes/SK/VII/2003 tentang Penyelenggaraan Pengobatan Tradisional untuk memastikan kelayakan obat tradisional di masyarakat’.

Menurut Ulung (2010), pengobatan tradisional atau alternatif adalah bentuk pengobatan dengan menggunakan alat, bahan, dan cara yang tidak termasuk dalam standar pengobatan kedokteran modern. Pengobatan tradisional biasanya dipergunakan sebagai alternatif atau pelengkap pengobatan kedokteran tersebut.

Berikut karakteristik pengobatan tradisional, yaitu :

1. Menggunakan cara penyembuhan tradisional seperti memijat, ramuan, dan jamu.
2. Alat-alat medik yang dipakai dalam dunia kesehatan tidak digunakan.

3. Secara umum pelaku usaha pengobatan tradisional tidak memiliki ilmu pengetahuan dalam bidang kesehatan namun ada yang melalui jalur pendidikan ataupun ajaran dari nenek moyang

2.3.1 Bekam

Bekam adalah salah satu jenis terapi alternatif untuk pengobatan nyeri di seluruh dunia. Bekam dikenal dengan berbagai istilah seperti: canduk, canthuk, kop, dan lainnya. Di dunia barat bekam disebut sebagai *cupping method* atau *cupping therapy*. Bekam merupakan salah satu intervensi medis tertua yang dipercaya ada sejak 3000 sebelum masehi. Ditemukan bahwa catatan paling awal tentang bekam adalah dalam *Ebers Papyrus* yaitu salah satu buku teks kedokteran tertua di dunia. Buku tersebut menjelaskan bahwa bekam dengan mekanisme kerja membuang zat asing dari dalam tubuh telah ada di Mesir sejak tahun 1.550 tahun sebelum masehi (SM)

Alat bekam antara lain adalah pisau bedah (*scalpel* steril) ukuran 15-22 *gauge* yang digunakan untuk menoreh kulit pada prosedur bekam basah, 6-12 kop plastik, pompa penyedot (manual atau elektrik), kertas tisu, lap, alat antiseptik berupa bola kapas, alkohol serta untuk mempermudah gerakan kop yaitu minyak zaitun (Badriyah Hidayati et al., 2019).



Gambar 2.5 Bekam (Sumber:(Badriyah Hidayati et al., 2019))

Dalam pandangan islam, bekam tak hanya sebagai terapi alternatif kesehatan namun juga sebagai anjuran dari Rasulullah SAW. Hal ini dibuktikan dengan banyaknya hadist tentang bekam. Berikut adalah hadis tentang bekam:

حَدَّثَنَا يَحْيَى بْنُ أَبِي أُيُوبَ وَفُتَيْبَةُ بْنُ سَعِيدٍ وَعَلِيُّ بْنُ حُجْرٍ قَالُوا حَدَّثَنَا إِسْمَاعِيلُ بْنُ يَحْيَى عَنْ ابْنِ جَعْفَرٍ عَنْ حُمَيْدٍ قَالَ ،
سُئِلَ أَنَسُ بْنُ مَالِكٍ عَنْ كَسْبِ الْحَجَّامِ فَقَالَ احْتَجَمَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ حَجَمَهُ أَبُو طَيْبَةَ فَأَمَرَ لَهُ
بِصَاعَيْنِ مِنْ طَعَامٍ وَكَلَّمَ أَهْلَهُ فَوَضَعُوا عَنْهُ مِنْ خَرَايجِهِ وَقَالَ إِنَّ أَفْضَلَ مَا تَدَاوَيْتُمْ بِهِ الْحَجَامَةُ أَوْ هُوَ مِنْ أَمَثَلِ
دَوَائِكُمْ حَدَّثَنَا ابْنُ أَبِي عُمَرَ حَدَّثَنَا مَرْوَانُ يَعْنِي الْفَزَارِيَّ عَنْ حُمَيْدٍ قَالَ سُئِلَ أَنَسٌ عَنْ كَسْبِ الْحَجَّامِ فَذَكَرَ بِمِثْلِهِ
'غَيْرَ أَنَّهُ قَالَ إِنَّ أَفْضَلَ مَا تَدَاوَيْتُمْ بِهِ الْحَجَامَةُ وَالْفُسْطُ الْبَحْرِيُّ وَلَا تُعَذِّبُوا صِبْيَانَكُمْ بِالْعَمَزِ

“Telah menceritakan kepada kami [Yahya bin Ayyub] dan [Qutaibah bin Sa'id] dan [Ali bin Hujr] mereka berkata; telah menceritakan kepada kami [Isma'il] yang mereka maksudkan adalah Ibnu Ja'far, dari [Humaid] dia berkata, " [Anas bin Malik] ditanya mengenai tukang bekam, dia lalu menjawab, "Rasulullah shallallahu 'alaihi wasallam pernah berbekam dan yang membekam beliau adalah Abu Thaibah, lantas beliau memerintahkan (keluarganya) supaya memberikan kepada Abu Thaibah dua gantang makanan, dan beliau menganjurkan kepada tuannya supaya dia (tuannya) meringankan tugas yang dibebankan kepada Abu Thaibah. Beliau bersabda: "Sesungguhnya berbekam adalah pengobatan yang paling utama atau termasuk terapi yang paling baik." Telah menceritakan kepada kami [Ibnu Abu Umar] telah menceritakan kepada kami [Marwan] -yaitu Al Fazari- dari [Humaid] dia berkata; [Anas] ditanya mengenai tukang bekam, lalu dia menyebutkan hadis seperti di atas. Namun ia menambahkan, "Sesungguhnya bekam dan Qusth Al Bahri (sejenis tumbuhan) adalah pengobatan yang paling utama buat kalian, dan janganlah kalian menyakiti anak-anak kalian dengan ghamz (yaitu memasukkan jari jemari ke kerongkongan seorang anak untuk menghilangkan rasa sakit". (Hadis Shahih Muslim Nomor 4087)

حَدَّثَنَا أَبُو نُعَيْمٍ حَدَّثَنَا عَبْدُ الرَّحْمَنِ بْنُ الْعَسِيلِ عَنْ عَاصِمِ بْنِ عُمَرَ بْنِ قَتَادَةَ قَالَ سَمِعْتُ جَابَ

حَدَّثَنِي مُحَمَّدُ بْنُ عَبْدِ الرَّحِيمِ أَخْبَرَنَا سُرَيْجُ بْنُ يُونُسَ أَبُو الْحَارِثِ حَدَّثَنَا مَرْوَانُ بْنُ شُجَاعٍ عَنْ سَالِمِ الْأَفْطَسِ عَنْ سَعِيدِ بْنِ جُبَيْرٍ عَنْ ابْنِ عَبَّاسٍ عَنِ النَّبِيِّ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ الشِّفَاءُ فِي ثَلَاثَةٍ فِي شَرْطَةٍ مَحْجَمٍ أَوْ شَرْبَةِ عَسَلٍ أَوْ كَيْةٍ بِنَارٍ وَأَنَا أَنْهَى أُمَّتِي عَنِ الْكَيِّ

“Telah menceritakan kepada kami [Muhammad bin Abdurrahim] telah mengabarkan kepada kami [Suraij bin Yunus Abu Al Harits] telah menceritakan kepada kami [Marwan bin Suja'] dari [Salim Al Afthas] dari [Sa'id bin Jubair] dari [Ibnu Abbas] dari Nabi shallallahu 'alaihi wasallam beliau bersabda: "Terapi pengobatan itu ada tiga cara, yaitu; berbekam, minum madu dan kay (menempelkan besi panas pada daerah yang terluka), sedangkan aku melarang ummatku berobat dengan kay". (Hadis Bukhori Nomor 5263)

2.3.2 Pijat Refleksi

Pijat refleksi merupakan cara memijat anggota tubuh seperti tangan dan kaki dengan mengarah pada titik pusat urat-urat saraf. Pemijatan di bagian-bagian tertentu mewakili semua organ internal, anggota tubuh, sistem tubuh, dan kelenjar. Dengan dilakukannya stimulasi titik-titik tersebut dengan teknik pemijatan, organ-organ yang berhubungan akan mendapatkan efek langsung (Widyaningrum, 2013).

Pijat adalah terapi yang memanfaatkan jari untuk pengobatan yang dilakukan dengan cara menstimulasi titik-titik akupunktur manusia. Pijat refleksi menjadi salah satu cara untuk mencegah penyakit ataupun memperbaiki kesehatan secara alami dan tradisional pada segala usia. Untuk mendapatkan keberhasilan terhadap keluhan penyakit maka diperlukan pemahaman yang dalam tentang simpul-simpul saraf pada telapak kaki. Pijat refleksi dapat mengolah aliran energi dalam tubuh sehingga kondisi dapat diperbaiki (Wong, 2011).



Gambar 2.6 Pijat Refleksi (Sumber: (Wong, 2011))

2.3.3 Akupunktur

Akupunktur adalah sistem pengobatan tradisional yang berasal dari cina. Akupunktur sebenarnya berasal dari dua kata “*acus*” yaitu jarum dan “*puncture*” yaitu tusuk. Jadi, akupunktur merupakan pengobatan dengan jarum melalui cara menusuk (Saputra, 2017).

Pengobatan akupunktur telah ada semenjak dua ribu tahun lalu dan berkembang pesat di Amerika Serikat sekitar dua puluh tahun yang lalu. Teknik tusuk jarum ini di gunakan oleh para ahli anastesi, kedokteran fisik, penyakit syaraf, dan perawat. Akupunktur adalah penusukan jarum ke tubuh melalui titik khusus secara anatomis dan disesuaikan dengan keluhan. Terapi ini memiliki tujuan sebagai rasa nyaman dan mengurangi rasa sakit. Akupunktur adalah pemikiran bahwa pola aliran suatu energi (*Qi*) yang melalui sistem meridian tubuh. Jika redapat gangguan pada aliran ini maka mengakibatkan penyakit pada manusia. Akupunktur bekerja untuk memulihkan kembali pola aliran energi tersebut sehingga suatu penyakit dapat disembuhkan. Penusukan jarum akupunktur didasarkan pada prinsip pemijatan dengan titik utama 2 di leher, 3 di perut, dan 2 di tungkai bawah. Disarankan selama dua bulan untuk melakukan terapi ini (Stroke et al., 2011).

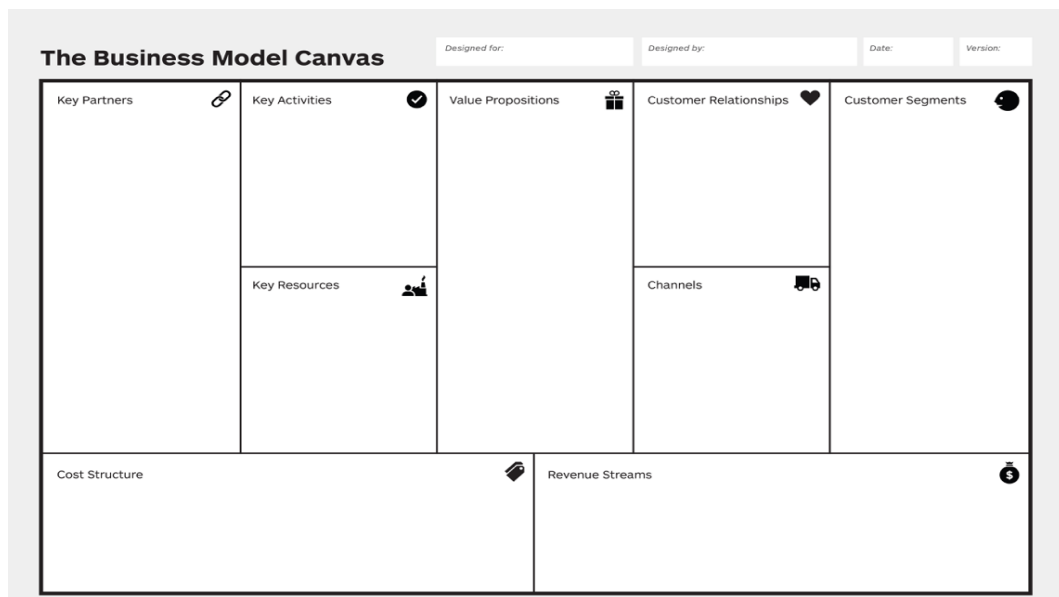


Gambar 2.7 Akupunktur (Sumber: (Stroke et al., 2011))

2.4 Business Model Canvas (BMC)

Dunia bisnis era ini sangat kompetitif dan cepat berubah-ubah sehingga membuat perusahaan kesulitan dalam mengambil keputusan bisnis. Sekarang ini perusahaan harus sedemikian rupa dapat mengelola penerapan teknologi informasi, rantai pasok, saluran distribusi, serta perusahaan harus bisa bereaksi fleksibel dengan perubahan pasar. Tidak adanya perangkat lunak dalam hal mensimulasikan model bisnis membuat manajer kesulitan memfasilitasi keputusan strategis bisnis. Bahkan dalam banyak kasus, banyak manajer dalam perusahaan tidak memiliki pengetahuan cukup tentang model bisnis perusahaannya.

(Pigneur, 2012) membuat 9 elemen pada kerangka model bisnis dalam bentuk kanvas. Sembilan elemen ini saling berkaitan yang dipaparkan dengan satu definisi untuk memvisualisasikan, menilai, mengubah, dan menggambar model bisnis. Cara yang paling efektif dalam menilai integritas keseluruhan model bisnis adalah melalui *Business Model Canvas*. Model bisnis terbaik dapat digambarkan melalui 9 blok dasar yang menjelaskan bagaimana uang atau keuntungan dapat diperoleh perusahaan. Elemen *Business Model Canvas* terdiri dari *customer segments*, *value propositions*, *channels*, *customer relationships*, *revenue streams*, *key resources*, *key activities*, *keypartnerships*, dan *cost structure*.



Gambar 2.8 *Business Model Canvas* (Pigneur (2012))

2.4.1 Customer Segments (Segmen Pelanggan)

Segmen pelanggan adalah sekelompok orang yang dilayani perusahaan. Singkatnya, segmen pelanggan menjelaskan siapa yang akan menggunakan produk perusahaan, karakteristik masalah dan kebiasaan pelanggan. Pelanggan (*customer*) adalah orang yang memberikan keuntungan (*profit*) bagi perusahaan. Jika perusahaan tidak memiliki pelanggan, maka perusahaan akan bangkrut. Perusahaan dapat mengklasifikasikan pelanggan ke dalam segmen pasar sesuai dengan keinginan, sumber daya, lokasi, dan kebiasaan membeli. Ada tipe-tipe dalam segmentasi pasar antara lain:

1. Pasar Terbuka (*Mass Market*)

Pasar terbuka menargetkan semua orang sebagai pelanggan dan berfokus pada permasalahan yang hampir sama pada masyarakat umum. Contohnya adalah pelanggan elektronik.

2. Ceruk Pasar (*Niche Market*)

Pasar ceruk menargetkan segmen pasar spesifik yang jumlahnya kecil dan belum terlayani dengan baik. Contohnya adalah bahan bahan dari pabrik

automobil utama dibutuhkan pabrik perakitan mobil. Biasanya terdapat pada hubungan bisnis antara *supplier-buyer*.

3. Pasar Tersegmentasi (*Segmented*)

Pasar tersegmentasi menargetkan pelanggan dalam banyak segmen dengan pendekatan demografi berupa usia, jenis kelamin, pendidikan, pekerjaan dan pendekatan geografi (wilayah tempat tinggal) serta pendekatan psikografi (sikap, minat pelanggan, sikap, dan gaya hidup).

4. Diversifikasi Pasar (*Diversified*)

Diversified mengklasifikasikan pelanggan yang memiliki ciri berbeda namun kebutuhan sama. Misalnya, *Amazon.com* menjual jasa layanan “*cloud computing*” yang menyebabkan *amazon.com* harus memulai bisnisnya dengan segmentasi pasar yang berbeda.

5. Multipasar (*Multi-sided Platform*)

Multipasar berfokus pada pelanggan yang saling berhubungan. Sebagai contoh pada perusahaan koran membutuhkan pembaca yang banyak agar iklan tertarik kepada perusahaan.

2.4.2 *Value Propositions* (Proposisi Nilai)

Value Propositions menawarkan kelebihan produk atau jasa dari sebuah perusahaan dibanding perusahaan pesaing (*competitor*). Nilai yang ditawarkan harus dapat memecahkan permasalahan dan mempunyai kelebihan untuk memenuhi kebutuhan pelanggan. *Value Propositions* (Proposisi nilai) merupakan nilai yang tercipta dari penjualan produk dan jasa.

Value Propositions harus bersifat inovatif sehingga membuat pelanggan tanpa enggan membayar lebih mahal dan ini akan meningkatkan keuntungan atau pendapatan. *Value Propositions* atau bisa disebut nilai tambah berupa kuantitatif, seperti harga ataupun kecepatan pelayanan dan kualitatif seperti desain serta pengalaman pelanggan.

Elemen-elemen yang berkontribusi pada *value proposition* adalah:

1. Kebaruan (*Newness*), suatu hal yang baru berarti belum pernah diciptakan atau ditawarkan oleh perusahaan sebelumnya. Proposisi nilai sering terjadi pada produk teknologi, misalnya *smartphone*.
2. Kinerja (*Performance*), pelayanan ataupun kualitas produk dapat meningkat.
3. Kustomisasi (*Customization*), konsep produk atau jasa yang disesuaikan dengan kebutuhan pelanggan.
4. Penyelesaian Pekerjaan (*Getting the Job Done*), membantu pelanggan dalam menyelesaikan pekerjaan. Seperti, pelanggan dibantu oleh perusahaan periklanan untuk membuat iklan. Dengan begitu, pelanggan dapat berfokus pada strategi promosi saja.
5. Desain (*Design*), desain yang bagus merupakan nilai tambah yang penting khususnya dalam industri fashion.
6. Merek/Status (*Brand/ Status*), pelanggan menganggap merek dapat meningkatkan status sosial ekonomi jika memakai produknya.
7. Harga (*Price*), menawarkan produk yang lebih murah dibanding produk yang berada di pasar.
8. Pengurangan Biaya (*Cost Reduction*), biaya pelanggan menjadi berkurang dari aktivitas yang dilakukan.
9. Pengurangan Risiko (*Risk Reduction*), garansi produk atau jasa mampu mengurangi risiko yang dihadapi pelanggan
10. Akses (*Accessibility*), proposisi ini membuat pelanggan tidak dapat menjangkau produk atau jasa sebelumnya.
11. Kenyamanan/Kemudahan Penggunaan (*Convenience/ Usability*), proposisi ini sangat penting karena nyaman dan mudah adalah nilai tambah yang harus dimiliki perusahaan.

2.4.3 Channels (Saluran)

Channels adalah saluran penghubung dengan para pelanggan. Secara sederhana, *channels* menjelaskan bagaimana *value propositions* yang ditawarkan bisa sampai ke pelanggan. *Channels* adalah media perusahaan untuk

berkomunikasi dengan pelanggannya dalam hal penyampain nilai perusahaan. Salah satu usaha perusahaan untuk berkomunikasi dengan pelanggan adalah dengan distribusi dan *sales* atau jaringan penjual.

Channels atau saluran terbagi ke dalam 5 (lima) fase yaitu:

1. Fase *awareness*, berfungsi memperkenalkan perusahaan kepada pelanggan.
2. Fase *evaluation*, berfungsi untuk saling menilai antara perusahaan dan pelanggan.
3. Fase *purchase*, berfungsi untuk proses transaksi jual beli produk perusahaan dan pelanggan.
4. Fase *delivery*, berfungsi sebagai pembuktian value propositions.
5. Fase *after sales*, berfungsi sebagai penentu bagi perusahaan apakah pelanggannya akan melakukan transaksi ulang.

2.4.4 Customer Relationships (Hubungan Pelanggan)

Customer relationships menggambarkan penjagaan hubungan antara perusahaan dan konsumen. Perusahaan dituntut untuk menjelaskan jenis hubungan yang ingin dibangun oleh masing-masing segmen pelanggan. *Customer relationships* fokus dalam hal menjaga hubungan baik dengan pelanggan agar mau menggunakan produk atau jasa yang ditawarkan. Misalnya promosi *buy one get one*. Kategori-kategori dari *customer relationships* yang dapat dipadukan dengan *customer segments*, antara lain:

1. *Personal Assistance, call center* biasanya dijadikan perusahaan agar dapat berkomunikasi dengan pelanggan.
2. *Dedicated Personal Assistance*, pelanggan diberi perlakuan istimewa oleh perusahaan secara pribadi.
3. *Self Service*, membuat pelanggan tidak berinteraksi secara langsung dengan perusahaan.
4. *Automated Service*, pelanggan tidak berinteraksi dengan perusahaan secara langsung namun perusahaan menyediakan hal penting yang diproses

otomatis. Penarapan *Automated service* adalah mampu mengenali individu serta karakteristik pelanggan sehingga dapat menawarkan produk yang sesuai dengan pelanggan.

5. *Communities*, perusahaan menggunakan komunitas sebagai pendekatan dan fasilitator pelanggan.
6. *Co-creation*, pelanggan dilibatkan dalam hal menciptakan nilai bagi pelanggan itu sendiri.

2.4.5 Revenue Streams (Arus Pendapatan)

Revenue streams sering diartikan sebagai laba, *income*, dan keuntungan. *Revenue streams* adalah pendapatan yang diterima perusahaan atas *value proposition* yang diberikan. Terdapat cara untuk mendapatkan *Revenue Streams*, yaitu:

1. Melakukan penjualan Aset (*Asset Sale*).
2. Penggunaan jasa pelayanan (*Usage Fee*).
3. Biaya Langganan (*Subscription Fees*), perusahaan memberikan kartu anggota.
4. Biaya Sewa (*Lending/ Renting/ Leasing*).
5. Imbalan Biaya Lisensi (*Licensing*).
6. Biaya Jasa Perantara (*Brokerage Fees*). Sebagai contoh, komisi yang didapat oleh agen real estate setiap kali berhasil mencocokkan pembeli dan penjual.
7. Biaya dari iklan (*Advertising*)
8. Donasi (*Donation*) (Hermawan & Pravitasari, 2013).

2.4.6 Key Resources (Sumber Daya Utama)

Key resources adalah aset atau sumber daya utama untuk sebuah model bisnis. *Key resources* menciptakan dan menawarkan *value proposition* (proposisi nilai), memperoleh pendapatan, menjangkau pasar, dan menjaga hubungan dengan pelanggan. *Key resource* menjelaskan aset yang diperlukan perusahaan untuk

membuat model bisnis *startup* dapat berfungsi. *Key resources* dapat dikelompokkan sebagai berikut:

1. Manusia (*Human*), yaitu sumber daya manusia.
2. Intelektual (*Intellectual*) meliputi hak cipta, pengetahuan, paten dan brands.
3. Fasilitas (*Physical*) meliputi fasilitas bangunan, pabrik, sistem penjualan, mesin dan peralatan serta jaringan distribusi.
4. Finansial (*Financial*), sumber daya finansial seperti kredit dan uang tunai.
5. Teknologi (*Technology*), teknologi menjadi sumber daya utama khususnya perusahaan telekomunikasi.
6. Saluran Distribusi (*Channel*).

2.4.7 Key Activities (Aktivitas Kunci)

Aktivitas kunci merupakan aktivitas utama yang dilakukan oleh perusahaan untuk mendapatkan nilai tambah. *Key activities* berfokus pada pembuatan model bisnis agar dapat bekerja. *Key activities* dapat dikelompokkan sebagai berikut:

1. Operasi Produksi (*Production*)

Pada bagian produksi, aktivitas yang dilakukan adalah pengadaan bahan, pengolahan, serta penyaluran produk jadi atau jasa kepada pelanggan.

2. Operasi Jasa (*Problem Solving*)

Pada bagian operasi jasa, aktivitas yang dilakukan adalah memberikan solusi terhadap permasalahan pelanggan secara individu. Aktivitas pemecahan masalah adalah jenis kegiatan operasi bagi rumah sakit dan konsultan.

3. Platform dan Jaringan (*Platform/ Network*)

Kegiatan utama bisnis yang berbasis *platform* dan jaringan adalah merancang, membangun, mengembangkan *hardware* dan *software*, termasuk jaringan internet dan *website*.

2.4.8 Key Partnerships (Kemitraan Utama)

Key partnership biasa disebut sebagai mitra kerja utama perusahaan. Contoh dari mitra kerja adalah *supplier*. *Key partnership* menjelaskan jaringan pemasok dan mitra yang membuat model bisnis berjalan dengan baik. Perusahaan harus mempunyai mitra kerja agar mengurangi resiko dan mengoptimalkan bisnis.

2.4.9 Cost Structure (Struktur Biaya)

Untuk mengoperasikan suatu model bisnis diperlukan adanya struktur biaya pengeluaran. Tentu saja dalam hal memberikan nilai, menjaga hubungan dengan pelanggan serta untuk menghasilkan pendapatan dikenakan biaya-biaya. (Pigneur, 2012). Cost Structure memiliki beberapa karakteristik, antara lain:

1. Biaya tetap yaitu biaya yang tetap sama walaupun kapasitas barang atau jasa yang dihasilkan naik atau turun.
2. Biaya variabel biaya yang bervariasi sesuai dengan kapasitas barang atau jasa yang dihasilkan.
3. Skala Ekonomi yaitu keuntungan biaya karena outputnya bertambah.
4. *Economies of Scope* yaitu pemanfaatan kapasitas aktivitas sehingga biaya dapat turun.

2.5 Contoh Implementasi *Business Model Canvas* (BMC)

1. *Business Model Canvas* (BMC) Pada Depot Selaris

Key Partners	Key Activities	Value Proposition	Customer Relationships	Customer Segments
<ul style="list-style-type: none">- Kolaborasi dengan mitra lain seperti bank dan sponsor- Bekerja sama dengan <i>merchant-merchant</i> lain untuk memaksimalkan fungsi <i>member card</i>	<ul style="list-style-type: none">- Aktivitas pemasaran melalui sosial media yang mencakup mengunggah foto-foto Depot Selaris, menginfokan promo-promo yang sedang berlangsung	<ul style="list-style-type: none">- Elemen <i>brand</i>: nama produk yang unik dan menarik- Memaksimalkan pelayanan cepat walau dalam kondisi ramai- Memudahkan pembayaran <i>customer</i> menggunakan sistem debit dan kredit- Membuat paketan untuk anak-anak kecil- Membuat desain makanan lebih menarik- Membuat paketan keluarga	<ul style="list-style-type: none">- Memperbanyak media komunikasi dengan pelanggan- Menerapkan sistem <i>member card</i> agar pelanggan loyal- Melalui media sosial (<i>Facebook, Instagram</i>) dan lain-lain	<ul style="list-style-type: none">- Fokus dengan pekerja kantor dan keluarga dan yang memiliki anak kecil
	<p>Key Resource</p> <ul style="list-style-type: none">- Pelatihan karyawan- Membuat hak paten- Penambahan <i>supplier</i> baru untuk setiap jenis bahan baku- Penambahan SDM untuk pembelian dan <i>marketing</i>		<p>Channels</p> <ul style="list-style-type: none">- Bekerja sama dengan banyak kantor untuk <i>catering</i> makan siang- Membuka cabang di tempat yang prospeknya tinggi	
Cost Structure			Revenue Streams	
<ul style="list-style-type: none">- Mengganti produk yang boros energi menjadi hemat energi (<i>low watt</i>)- Mencari bahan baku yang lebih murah			<ul style="list-style-type: none">- Penjualan melalui <i>asset sale</i>- Melakukan konsinyasi	

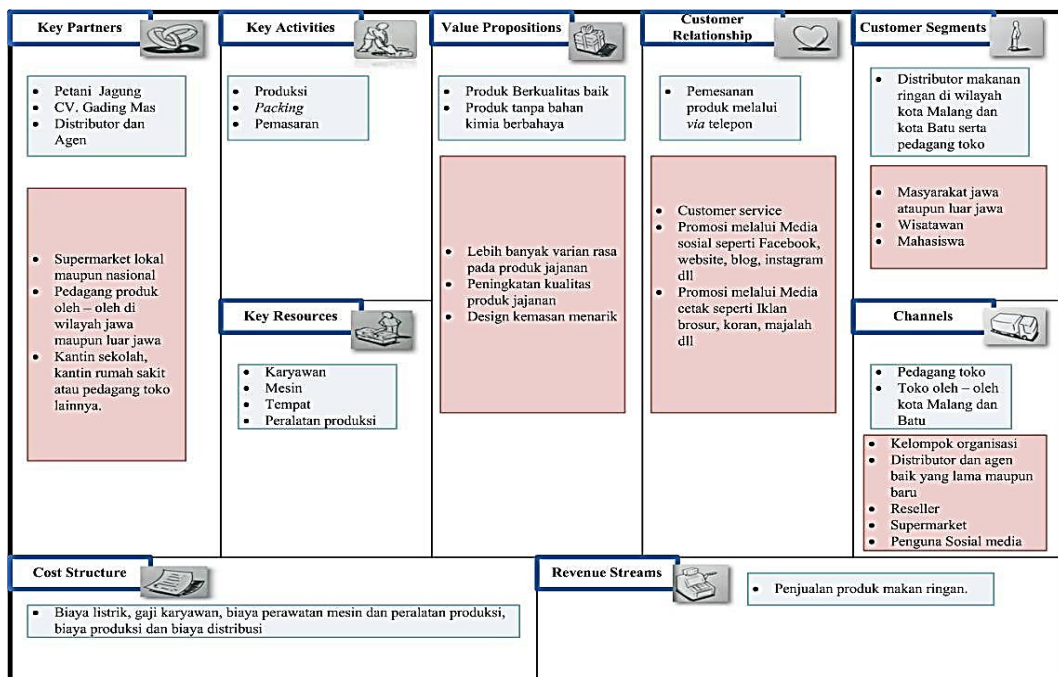
Gambar 2.9 BMC Depot Selaris (Dhyah, 2015)

2. Business Model Canvas (BMC) Pada Bisnis Hostel Bandung Pinisi Backpacker

Key Partners	Key Activities	Value Propositions	Customer Relationships	Customer Segments
1. Supplier makanan pagi 2. Supplier kasur 3. Pengrajin kayu 4. Laundry 5. Agen website traveling (agoda, pegipegi, traveloka, booking.com) 6. Agen travel offline 7. Supplier perlengkapan mandi 8. Supplier makanan ringan 9. Ojek/rental kendaraan wisata	1. Operasional <i>check in & checkout</i> 2. <i>Maintenance</i> kamar 3. <i>Maintenance</i> website 4. Penyediaan sarapan 5. Promosi online	Hostel murah dengan lokasi strategis dan dapat dijangkau melalui <i>online media</i>	1. <i>directphone</i> kepada <i>frontliner</i> 2. website pinisibackpacker.com	1. wisatawan nusantara 2. wisatawan mancanegara 3. 17–60 tahun
Key Resources			Channels	
1. Lokasi hostel 2. Pegawai 3. website			1. website pinisibackpacker.com 2. agen travel online (agoda, pegipegi, traveloka, booking.com)	
Cost Structure		Revenue Streams		
1. Gaji pegawai 2. Biaya <i>hosting</i> dan <i>domain</i> 3. Biaya sarapan 4. Biaya laundry 5. Biaya <i>maintenance</i> peralatan dan bangunan 6. Biaya langganan internet 7. Biaya listrik 8. Biaya TV Kabel 9. Biaya persediaan makanan ringan 10. Biaya persediaan perlengkapan mandi		1. Penjualan kamar 2. Agen wisata dalam kota 3. Agen transportasi 4. Makanan ringan 5. Perlengkapan mandi 6. Layanan laundry		

Gambar 2.10 BMC Pinisi Backpacker (Sovia Pramudita, 2018)

3. Business Model Canvas (BMC) Pada UD. Cahaya



Gambar 2.11 BMC UD. Cahaya (Hartatik & Baroto, 2017)

2.6 Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah bahasa untuk memodelkan secara rinci kebutuhan sistem. Hal ini bertujuan agar *programmer* dapat membangun sistem lebih terarah (Suendri, 2018).

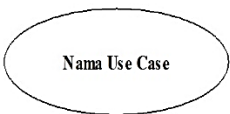

Pada tahun 1994 pertama kali UML dipopulerkan oleh *Grady Booch* dan *James Rumbaugh*. UML adalah penggunaan teknik pengembangan sistem dengan bahasa grafis sebagai alat untuk melakukan spesifikasi pada sistem. Beberapa pemodelan yang termasuk kedalam pemodelan UML seperti *use case diagram*, *class diagram*, *activity diagram*, dan *sequence diagram* (Mulyani, 2016)


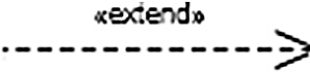

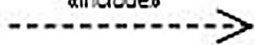
2.6.1 Use Case Diagram

Menurut (Salahuddin M, 2018) *Use case* merupakan sebuah model yang akan dibuat untuk suatu kelakuan (*behavior*) sistem informasi.” *Use case* menggambarkan adanya interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Adanya *use case* berguna untuk mengetahui fungsi yang terdapat pada sistem informasi dan mengetahui orang yang memakai sistem tersebut. Secara sederhana, *use case diagram* merupakan diagram yang berisi hubungan aktor dan sistem secara global (Mulyani, 2016).

Berikut simbol-simbol pada *use case diagram*:

Tabel 2.1 Simbol-simbol *Use Case Diagram* (Salahuddin M, 2018)

No	Simbol	Fungsi
1.	<p><i>USE CASE</i></p> 	Penjelasan tentang Fungsionalitas aksi aksi pada sistem yang dinyatakan dengan menggunakan kata kerja diawal frase.
2.	<p><i>ACTOR</i></p> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi. Aktor adalah peran pengguna dalam sistem informasi.

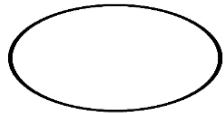
3.	<p><i>ASSOCIATION</i></p> 	Penghubung atau sebagai komunikasi antara aktor dan <i>use case</i> .
4.	<p><i>EXTEND</i></p> 	Relasi tambahan dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri
5.	<p><i>GENERALIZATION</i></p> 	Hubungan umum dan khusus antara dua buah <i>use case</i> sementara fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	<p><i>INCLUDE</i></p> 	Relasi tambahan pada sebuah <i>use case</i> dimana <i>use case</i> memerlukan <i>use case</i> lain untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini


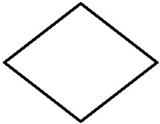

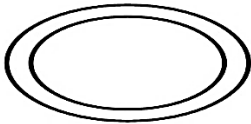
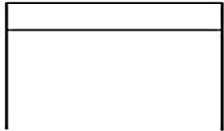
2.6.2 Activity Diagram

Activity Diagram adalah penggambaran alur kerja atau aktivitas pada *use case* (proses), logika, proses bisnis dan hubungan antara aktor dengan alur-alur kerja *use case*. *Flowchart* diidentikkan dengan *activity diagram*, ada beberapa notasi tambahan yang digunakan untuk kasus-kasus tertentu (Mulyani, 2016). *Activity Diagram* memberikan keterangan aktivitas dan proses yang dilalui yang pengguna lakukan terhadap sistem (Suendri, 2018).

Berikut adalah simbol-simbol pada *activity diagram*:

Tabel 2.2 Simbol-simbol *Activity Diagram* (Salahuddin M, 2018)

No	Simbol	Fungsi
1.	<p>Status awal</p> 	Setiap sistem pasti memiliki status awal aktivitas sistem.






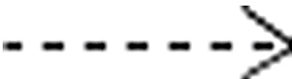
2.	Aktivitas 	Kegiatan yang dilakukan sistem yang diawali dengan kata kerja.
3.	Percabangan 	Apabila ada pilihan aktivitas lebih dari satu maka digunakan percabangan.
4.	Penggabungan 	Aktivitas yang lebih dari satu digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan sistem.
6.	Swimlane 	Memisahkan antara organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.6.3 Class Diagram

Diagram kelas berfungsi sebagai penghubung antara perangkat lunak dan dokumentasi perancangan sehingga pengguna (*user*) dapat membuat kelas-kelas yang sesuai rancangan didalam diagram kelas. Pada banyak kasus, perancangan tidak berguna karena kelas yang dibuat tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak (Salahuddin M, 2018).

Berikut adalah simbol-simbol pada *class diagram*:

Tabel 2.3 Simbol-simbol *Class Diagram* (Salahuddin M, 2018)

No	Simbol	Fungsi
1.	<p><i>Generalization</i></p> 	Hubungan berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>) dengan objek anak (<i>descendent</i>).
2.	<p><i>Nary Association</i></p> 	Penghindaran asosiasi dengan lebih dari dua objek.
3.	<p><i>Class</i></p> 	Kumpulan dari berbagai objek yang terbagi dalam satu atribut serta operasi yang sama.
4.	<p><i>Collaboration</i></p> 	Keterangan dari urutan aksi yang ditampilkan sistem.
5.	<p><i>Realization</i></p> 	suatu objek melakukan operasi yang sesuai atau benar.
6.	<p><i>Dependency</i></p> 	Relasi perubahan pada elemen <i>independent</i> yang akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.


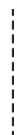
	<i>Association</i> <hr/>	Sebagai penghubung objek satu dengan objek lainnya.
--	-----------------------------	---


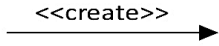
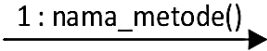
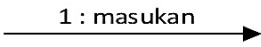
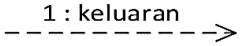
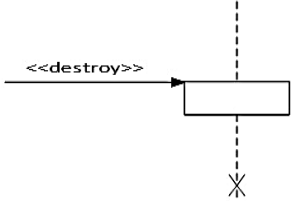
2.6.4 Sequence Diagram

Menurut (Saputra, M & Lusia, 2020), *sequence diagram* membuat suatu pemodelan tentang bagaimana interaksi objek dengan orang lain dalam *use case* dengan melakukan kolaborasi objek berdasarkan urutan. Sedangkan menurut (Suendri, 2018), *sequence diagram* adalah deskripsi aktivitas pengguna yang didasarkan pada rangkaian urutan waktu penggunaan sistem.

Berikut adalah simbol-simbol pada diagram *sequence*:

Tabel 2.4 Simbol-simbol *Sequences Diagram* (Salahuddin M, 2018)

No	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>nama aktor</p>	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi. Aktor adalah peran pengguna dalam sistem informasi.
2.	<p><i>Life Line</i></p> 	Pemberian gambaran kehidupan pada suatu objek.
3.	<p>Objek</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">nama objek : nama_kelas</div>	Interaksi pesan pada objek.

4.	<p>Waktu Aktif</p> 	Objek dalam keadaan aktif dan berinteraksi.
5.	<p>Pesan Tipe <i>Create</i></p> 	Suatu objek membuat objek yang lain dengan arah yang sesuai.
6.	<p>Pesan Tipe <i>Call</i></p> 	Suatu objek memanggil metode ataupun operasi pada objek lain atau dirinya sendiri.
7.	<p>Pesan Tipe <i>Send</i></p> 	Suatu objek mengirim data ke objek lainnya sesuai dengan arah panah pada objek yang akan mengirim data.
8.	<p>Pesan Tipe <i>Return</i></p> 	Suatu objek akan menghasilkan suatu kembalian ke objek tertentu apabila telah menjalankan suatu operasi .
9.	<p>Pesan Tipe <i>Destroy</i></p> 	Suatu objek yang mengakhiri hidup objek yang lain.

2.7 Metode *Waterfall*

Metode *waterfall* atau air terjun adalah suatu model dalam hal model pengembangan sistem informasi yang bersifat sekuensial dan sistematis. Terdapat lima tahapan pada metode *waterfall*, yaitu:

1. *Requirements Analysis*

Pada tahap analisis kebutuhan sistem hal yang dilakukan adalah mendefinisikan spesifikasi sistem secara rinci. Adapun mengenai kendala, layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna sistem.

2. *System Design*

Tahapan perancangan sistem adalah mengelompokkan kebutuhan sistem perangkat lunak dan perangkat keras. Pembentukan arsitektur sistem secara keseluruhan harus dilakukan.

3. *Implementation*

Pada tahap ini, realisasi dari perancangan perangkat lunak sebagai serangkaian program. Selanjutnya dilakukan pengujian sistem yang memerlukan adanya *verifikasi* bahwa setiap unit memenuhi spesifikasinya.

4. *Testing*

Pada tahap pengujian sistem, jika suatu program telah utuh maka perlu diuji secara keseluruhan untuk memastikan kesesuaian dengan perangkat lunak. Setelah pengujian selesai dan berhasil, perangkat lunak dapat dikirimkan ke *customer*.

5. *Maintenance*

Tahapan akhir ini biasanya tahapan yang paling lama karena sistem digunakan secara nyata. *Maintenance* bekerja dalam hal memperbaiki kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya dan meningkatkan layanan sistem sebagai kebutuhan baru (Sasmito, 2017).

2.8 Floyd Warshall

Floyd warshall adalah salah satu metode perhitungan rute terpendek. Warshall menemukan sebuah algoritma dalam hal pencarian *path* terpendek dengan menggunakan *graf* berarah berbobot. Pencarian rute terpendek pada algoritma *Warshall* dimulai dengan iterasi dari titik awal lalu memperpanjang *path*. *Path* diperpanjang dengan mengevaluasi titik demi titik hingga mendapatkan rute yang paling pendek. Metode ini mengambil jarak minimal dari suatu titik ke titik lainnya. Algoritma dinamis yang diterapkan pada algoritma ini yang menyebabkan terambilnya jarak lintasan terpendek secara benar. Algoritma *Floyd Warshall* merupakan matriks hubung *graf* berarah berlabel, dan *output* yang dihasilkan adalah *path* terpendek dari semua titik ke semua titik (Hasibuan, 2016).

2.9 World Wide Web Atau WEB

World Wide Web lebih dikenal dengan singkatan WWW adalah ruang informasi oleh pengenalan global yang dinamakan *URL (Uniform Resource Locator)* untuk mengenal pasti sumber daya berguna (Saputra, 2020).

Web adalah jaringan sajian dan layanan informasi yang mendunia, yang sangat mudah diakses dan digunakan oleh pengguna internet, karena pada halaman informasi yang ditampilkan setiap kata bisa memiliki tautan (*link*) atau sambungan informasi yang ada pada halaman lain, yang berasal dari suatu komputer lain yang terhubung dengan internet. *Web* hanyalah salah satu aplikasi dari keberadaan internet berupa jaringan layanan informasi (Betha, 2019).

2.9.1 HTML

HTML adalah singkatan dari *HyperText Markup Language*. HTML diartikan sebagai bahasa penanda berbasis *text* untuk membuat sebuah halaman web yang dinamis. Bahasa penanda atau markah yang digunakan peramban adalah untuk menerjemahkan dan menulis teks, gambar, dan bahan lainnya ke dalam halaman *web* baik secara visual maupun suara. Ciri dasar setiap item dari markah

HTML didefinisikan di dalam peramban, dan ciri ini dapat diubah dengan cara menambahkan halaman *web desainer CSS* (Jannah, 2020).

Sedangkan menurut (Betha, 2019), *HTML* adalah sekumpulan kode, yang dapat digunakan untuk menspesifikasikan teks dalam dokumen untuk menjadi elemen dokumen dan bisa untuk saling bertaut.



Gambar 2.12 logo *HTML*

(Sumber: ([Www.W3.Org/Html/Logo](http://www.W3.Org/Html/Logo)))

2.9.2 PHP

PHP berasal dari kata “*Hypertext Preprocessor*” bahasa pemrograman yang banyak digunakan pengembang *web*. *PHP* dapat disematkan ke dalam *IHTML* dan bersifat umum atau *open source*. Pada saat mengakses halaman *web*, kode *PHP* dibaca oleh *server* tempat halaman itu berada. *Output* (keluaran) dari fungsi *PHP* pada halaman biasanya dikembalikan sebagai kode *HTML* yang dibaca oleh *browser*. Bahasa ini memiliki tujuan agar pengembang *web* dapat menulis halaman yang dihasilkan secara dinamis dengan cepat dan mudah (Saputra, 2017).

Menurut (Suposono, 2018), *PHP* diartikan sebagai bahasa pemrograman dalam hal penerjemahan baris-baris kode program menjadi kode mesin yang dimengerti. *PHP* bersifat *server side* yang dapat ditambahkan ke dalam *HTML*.



Gambar 2.13 Logo *PHP* (Sumber: *(Www.Php.Net)*)

2.10 MySQL

MySQL menjadi yang paling populer dalam hal u jenis *database server*. *Software* ini memiliki lisensi *GNU General Public License* yang berarti dapat diunduh secara gratis. Sistem basis data *MySQL* menggunakan arsitektur *client-server*. *Client-server* memiliki kendali pusat di *server* dan *server* tersebut bertindak sebagai sebuah program yang dapat memanipulasi *database*. Program *client* menggambarkan tujuan *user* kepada *server* dengan cara menuliskan *query* dengan bahasa *SQL (Structured Query Language)* (Sianipar, 2015).



Gambar 2.14 Logo *MySQL* (Sumber: *(Www.Mysql.Com)*)

2.11 XAMPP

XAMPP merupakan paket dari perangkat lunak atau *software* yang berisi *Apache HTTP server* dan *MySQL*. Dengan meng-*install XAMPP* dapat menjalankan sebuah *web server* pada komputer karena *XAMPP* mendukung program *PHP* dan *Perl* dan berjalan di beberapa sistem operasi (Dantes, 2019)

XAMPP ditulis dengan bahasa pemrograman *PHP*. Fungsi dari *XAMPP* adalah sebagai *localhost* yang terdiri dari beberapa program antara *Apache*, *MySQL database* dan penerjemah bahasa yang ditulis. Tanpa memakai *XAMPP*, file aplikasi yang memiliki format *PHP* tidak bisa terbaca. Format *PHP* mengidentifikasi bahasa pemrograman yang dipakai (Habibi, Roni, 2020).



Gambar 2.15 Logo *XAMPP* (Sumber : (Www.Apachefriends.Org))

2.12 Framework Laravel

Framework adalah kerangka kerja yang dibuat untuk mempermudah *develover/programmer* dalam menangani berbagai macam masalah dalam pemrograman seperti pemanggilan variabel dan koneksi ke *database*. Dengan adanya *framework*, *developer* dapat membuat dua *script* dalam tugas yang sama. Penggunaan *framework* membuat aplikasi tersusun terstruktur dan rapi karena sebuah *framework* diciptakan telah menggunakan MVC yaitu *Model-View-Controller* yang telah dikenal bagi kalangan *programmer PHP*. Jika dalam pembuatan aplikasi tidak menggunakan kerangka kerja atau *framework*, maka akan terjadi kesulitan seperti ketika terjadi *handling error* harus membaca ribuan baris kode yang telah dibuat (Prasetyo, 2018).

Framework adalah kumpulan *libraries* yang membuat seorang *developer/programmer* tidak harus membuat fungsi dari awal. Hal yang dilakukan hanya memanggil kumpulan fungsi yang sudah ada dalam *framework*. Salah satu contoh dari *framework PHP* adalah *Laravel*.

Laravel diciptakan oleh Taylor Otwell pada april 2011 dengan versi *php* 5.3 ke atas. Menurut (Sulaeman, 2019), laravel adalah sebuah *framework PHP* dengan konsep *MVC (Model View Controller)*. Laravel dirancang untuk meningkatkan kualitas perangkat lunak dengan pengurangan biaya pengembangan awal dan biaya pemeliharaan. Laravel menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu.

MVC berfungsi memisahkan aplikasi berdasarkan komponen aplikasi. Sebagai contoh *controller* dan *user interface* serta manipulasi data. *MVC* memisahkan aplikasi antara presentasi dan logika.

Model memiliki fungsi-fungsi yang dapat mempermudah seseorang dalam mengelola database. Sedangkan *controller* berfungsi sebagai jembatan antara *model* dan *view* (Prasetyo, 2018).

Dasar-Dasar Laravel menurut (Andi, 2018) dijelaskan sebagai berikut:

1. *Artisan*

Artisan berisi fungsi untuk membuka *website* yang telah dibuat tanpa menggunakan *web server* lokal. Fungsi ini adalah *php artisan* yaitu “*php artisan serve*”. Contoh penggunaan *artisan* dalam laravel:

```
C:\Users\StaNoIa\ite\Desktop\blog>php artisan serve
Laravel development server started: <http://127.0.0.1:8000>
[Wed Apr 26 07:40:29 2017] 127.0.0.1:61888 [200]: /favicon.ico
[Wed Apr 26 07:44:48 2017] 127.0.0.1:61905 [200]: /favicon.ico
[Wed Apr 26 07:45:29 2017] 127.0.0.1:61906 Invalid request (Unexpected EOF)
[Wed Apr 26 07:45:29 2017] 127.0.0.1:61913 [200]: /favicon.ico
[Wed Apr 26 07:45:45 2017] 127.0.0.1:61914 Invalid request (Unexpected EOF)
[Wed Apr 26 07:45:46 2017] 127.0.0.1:61922 [200]: /favicon.ico
[Wed Apr 26 07:46:00 2017] 127.0.0.1:61923 Invalid request (Unexpected EOF)
[Wed Apr 26 07:46:02 2017] 127.0.0.1:61938 [200]: /favicon.ico
[Wed Apr 26 07:46:12 2017] 127.0.0.1:61939 Invalid request (Unexpected EOF)
[Wed Apr 26 07:46:19 2017] 127.0.0.1:61951 [200]: /favicon.ico
[Wed Apr 26 07:47:07 2017] 127.0.0.1:61953 Invalid request (Unexpected EOF)
```

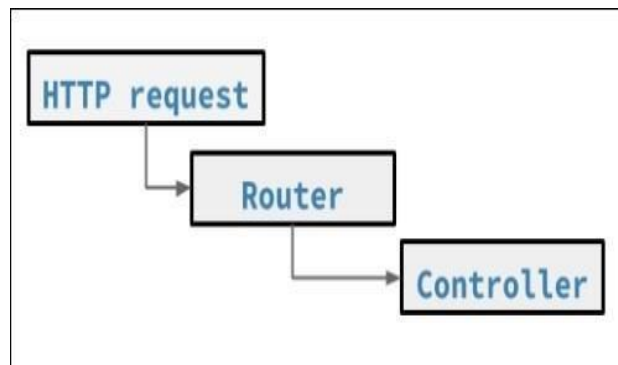
Gambar 2.16 *Php Artisan Laravel* (Sumber: Andi, 2018)

2. Routing

Routing merupakan proses agar *item* yang diinginkan dapat sampai ke tujuan. Penggunaan *routing* menentukan halaman-halaman yang akan muncul ketika layar dibuka oleh pengguna. Letak file *web.php* dalam hal pengaturan *routing* di *laravel* adalah di dalam folder *routes*.

3. Controller

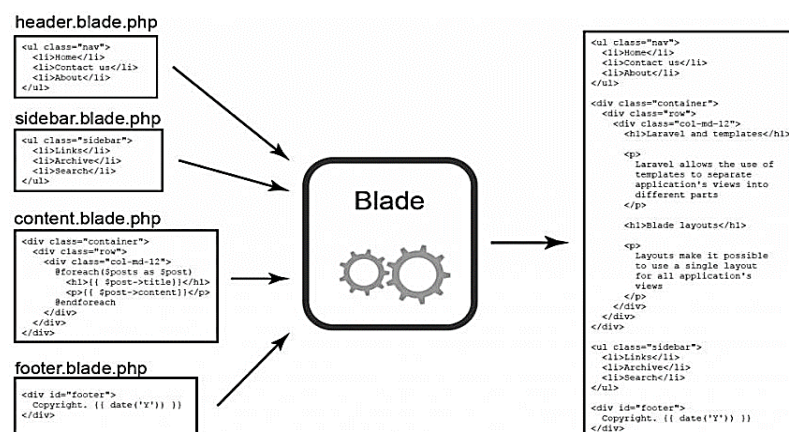
Controller merupakan proses dalam menginisialisasi, mengambil permintaan, dan memanggil model untuk di kirim ke *view*. Permintaan yang dibuat harus berada di dalam *controller*, kemudian dilempar melalui *routing*.



Gambar 2.17 *Controller Laravel* (sumber: Andi, 2018)

4. View (Blade Templating)

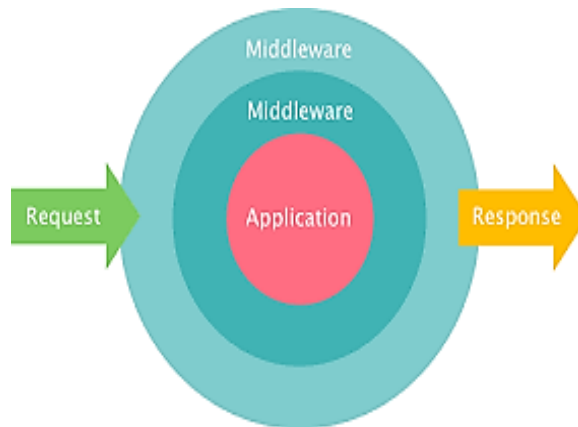
Blade adalah *template engine* bawaan dari *laravel*. Cara membuat *file.blade* adalah dengan membuat *nama_file.php.blade* di dalam folder *views* secara manual.



Gambar 2.18 *Blade Templating* (Sumber: Andi, 2018)

5. *Middleware*

Middleware adalah penengah antara *controller* yang akan dituju dan *request* yang masuk.



Gambar 2.19 *Middleware* (Sumber: Andi, 2018)

6. *Session*

Session adalah suatu cara dalam menggunakan penyimpanan pada *server* dan penyimpanan. Penyimpanan tersebut digunakan pada beberapa halaman termasuk halaman itu sendiri.

2.12.1 Laravel Versi 8

Pada laravel versi 8 ada beberapa fitur tambahan seperti, *Zttp For Http Client*, *Laravel Airlock* dan lain-lain.



Gambar 2.20 Laravel 8

Kelebihan-kelebihan *Framework* Laravel adalah sebagai berikut:

1. *Expressif*

Laravel bersifat *expressif* berarti seseorang akan secara langsung paham kegunaan dari sintaks laravel.

2. *Simple*

Laravel dikatakan *simple* karena laravel sederhana dalam masalah *routing*. Sebenarnya membangun *website* hanya berkaitan dengan *request-response*. (Aminudin, 2015).

3. *Accessible*

Laravel selalu melengkapi berbagai *fitur* terutama bila versi terbaru rilis sehingga mudah untuk digunakan. Hal ini membuat laravel menjadi PHP *framework* paling populer seperti *CodeIgniter*.

2.13 Penelitian Terdahulu

Berikut ini penelitian terdahulu yang digunakan peneliti sebagai bahan referensi dalam melakukan penelitian ini:

Tabel 2.5 Penelitian Terdahulu

No	Nama	Judul	Kelebihan	Kekurangan
1	(Suprianto & Matsea, 2018)	‘Rancang Bangun Aplikasi Pendaftaran Pasien Online Dan Pemeriksaan Dokter Di Klinik Pengobatan Berbasis Web’	Jurnal membahas bagaimana merancang aplikasi pendaftaran pasien dan melihat hasil rekam medis secara online menggunakan	Jurnal ini tidak membahas terkait sistem pembayaran online

			bahasa pemrograman PHP dan <i>database MySQL</i> .	
2	(Ahmad Amzul dan Imam, 2018)	‘Strategi Pengembangan Bisnis Cargo PT. Garuda Indonesia, Tbk Dengan Pendekatan <i>Business Model Canvas</i> ’	Penelitian jurnal ini membahas strategi yang digunakan ‘Bisnis Cargo PT. Garuda Indonesia untuk berkembang dengan pendekatan <i>Business Model Canvas</i> ’ dan disertai Pendekatan Internal maupun eksternal.	Pada penelitian jurnal ini hanya berfokus pada 3 elemen pada Business Model Canvas.
3	(Amin et al., 2018)	‘Strategi Perencanaan Model Bisnis Perusahaan Jasa Konsultan Arsitektur dan Jasa Kontraktor PT Architectaria Media Cipta’	Penelitian jurnal ini membahas ‘strategi perencanaan model bisnis PT Architectaria Media Cipta menggunakan pendekatan Business Model	Pada penelitian ini kurang detail dalam menjelaskan fokus ancaman dan cara mengatasinya dari ancaman yang dianggap cukup kuat pada

			Canvas (BMC), analisis SWOT, dan <i>Blue Ocean Strategy</i> (BOS).	berdasarkan hasil analisis SWOT.
5	(Widayati & Sahfitri, 2020)	‘Sistem Informasi Penjadwalan Dokter Dan Pendaftaran Pasien Menggunakan Metode <i>Round Robin</i> ’	Penelitian jurnal ini membahas bagaimana membangun ‘Sistem Informasi Penjadwalan Dokter Dan Pendaftaran Pasien Menggunakan Metode <i>Round Robin</i> ’ untuk sistem <i>time sharing</i>	Penelitian ini tidak menjelaskan bahasa pemrograman dan <i>database</i> yang digunakan
6	(Varianto, 2017)	‘Model Bisnis Colleges Need Menggunakan Pendekatan <i>Business Model Canvas</i> ’	Jurnal ini menjelaskan tahapan penyusunan BMC pada bisnis Colleges Need	Tidak menyertakan gambar 9 blok BMC. Jurnal hanya berisi penjelasan saja.

BAB III

METODE PENELITIAN

3.1. Tempat Penelitian

Peneliti mengambil tempat penelitian pada Pusat Bekam-Ruqyah (PBR) Cabang Medan yang berlokasi di Jalan. Prof. H. M. Yamin nomor 305 Medan Perjuangan, Medan.

3.2. Waktu Penelitian

Waktu penelitian dimulai diberi batasan waktu agar lebih terkonsep dan tepat pada target waktu yang dicapai dalam penelitian. Waktu penelitian dimulai pada bulan maret hingga september 2021. Adapun tahap-tahap pelaksanaan pada penelitian ini yang sudah ditargetkan pada penelitian ini adalah sebagai berikut:

1. Pengajuan judul, penulis menemukan permasalahan lalu mengajukan beberapa judul yang akan diajukan kepada prodi sistem informasi untuk mendapatkan persetujuan judul.
2. Riset kebutuhan, hal ini secara langsung dilaksanakan melalui observasi wawancara terhadap objek yang akan diteliti pada Pusat Bekam-Ruqyah (PBR) Cabang Medan.
3. Bimbingan proposal, bimbingan proposal dilakukan agar dalam pembuatan proposal lebih terarah yang dibimbing langsung oleh pembimbing 1 yaitu Bapak Suendri, M.Kom dan pembimbing 2 Ibu Raissa Amanda Putri, S. Kom., M.TI. Proposal penelitian dibimbing Bab per Bab oleh dosen pembimbing.
4. Seminar proposal, seminar proposal dilaksanakan ketika penulis sudah mendaftar untuk melaukan seminar proposal. Seminar proposal merupakan bentuk pengujian dari penguji terhadap objek yang diteliti.
5. Pengumpulan dan analisis data, setelah terlaksananya seminar proposal, penulis dapat melakukan lebih banyak lagi mengumpulkan informasi-informasi dan analisis data.

6. Perancangan sistem, perancangan sistem dibuat agar sistem usulan yang akan dibuat dapat lebih terarah sehingga mengetahui bagaimana alur dari sistem yang akan dibuat. Perancangan sistem dapat dengan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan perancangan antarmuka (*interface*).
7. Pembuatan sistem, langkah selanjutnya adalah membuat skrip atau *coding* program untuk membangun sistem yang berfungsi berbasis *web* menggunakan *framework* Laravel.
8. Pengujian program, setelah pembuatan sistem selesai, sistem akan diuji agar mendapatkan seberapa persen tingkat keberhasilan yang diharapkan dan berfungsi dengan baik.
9. Analisis BMC, melakukan analisis terhadap model bisnis yang akan dikembangkan pada *startup* Medis Reborn.
10. Sidang Skripsi, penulis menjelaskan serta mempertanggung jawabkan hasil penelitian yang telah dilakukan. Setelah itu, dosen penguji menguji hasil penelitian.

3.3. Kebutuhan Sistem

Adapun pada tahap ini penulis menganalisa dua jenis kebutuhan sistem yang diperlukan dalam pembuatan sistem.

3.3.1. Perangkat Lunak

Adapun spesifikasi kebutuhan dari perangkat lunak (*software*) yang digunakan dalam menyelesaikan pembuatan sistem adalah sebagai berikut:

1. *Operating System Windows 10 / 64 bit*
2. *XAMPP 8.0.8 64 bit*
3. *Visual Studio Code*
4. *Visual Paradigm*
5. *Browser (Google Chrome)*

3.3.2. Perangkat Keras

Adapun spesifikasi dari perangkat keras (*hardware*) yang minimal digunakan dalam menyelesaikan pembuatan sistem adalah sebagai berikut:

1. *Laptop atau PC*
2. *RAM DDR3 4GB, HDD 500GB*
3. *Processor i3*

3.4. Cara Kerja

3.4.1. Metode Penelitian Kualitatif

Metode yang penulis gunakan pada penelitian adalah metode kualitatif. Penelitian kualitatif merupakan penelitian dengan kecenderungan menggunakan analisis dengan pendekatan induktif. Riset dari kualitatif bersifat deskriptif. Landasan teori sangat dibutuhkan agar agar fokus penelitian sesuai dengan fakta di lapangan (Rukin, 2019). Pada pengumpulan data dilakukan dengan observasi atau pengamatan, wawancara, dan studi pustaka seperti jurnal atau buku yang terkait dengan penelitian. Berikut ini metode yang digunakan untuk pengumpulan data:

1. Observasi

Observasi dilakukan dengan mengamati langsung objek penelitian yaitu Pusat Bekam-Ruqyah (PBR) Cabang Medan agar mengetahui pelayanan alur sistem pemesanan pengobatan tradisional dan strategi bisnis yang sedang berjalan.

2. Wawancara

Wawancara dilakukan untuk mendapatkan informasi dari narasumber untuk mengetahui permasalahan yang ada. Dengan adanya wawancara, diharapkan penulis dapat mengetahui alur sistem pemesanan pengobatan tradisional dan strategi bisnis yang diterapkan Pusat Bekam-Ruqyah (PBR) Cabang Medan. Pada penelitian ini, penulis mewawancarai pemilik Pusat-Bekam-Ruqyah Cabang Medan.

3. Studi Pustaka

Studi pustaka dilakukan dengan memanfaatkan buku-buku atau jurnal-jurnal untuk dipelajari dan dibaca untuk mencari data berupa informasi yang terkait dalam masalah dalam penelitian.

3.4.2. Sumber Data

Adapun sumber data pada penelitian ini adalah sebagai berikut:

1. Data Primer

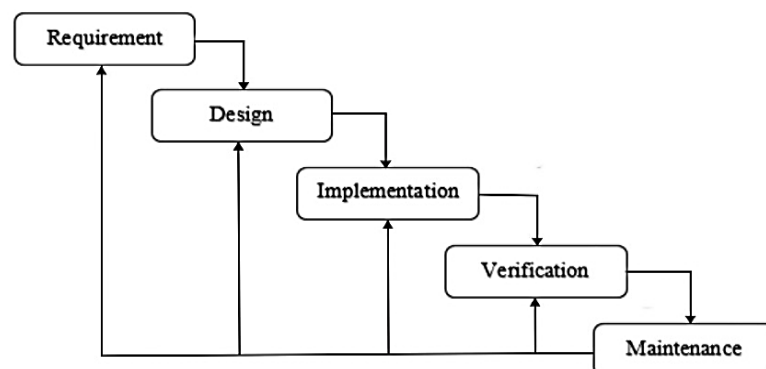
Data primer adalah data yang didapatkan oleh suatu individu ke dalam suatu instansi atau perusahaan. Dalam hal ini, penulis melakukan pengamatan dan pengumpulan data untuk pembuatan aplikasi pengobatan tradisional berbasis web serta rancangan strategi bisnis yang akan diterapkan dalam bentuk *Business Model Canvas* (BMC) pada Pusat Bekam-Ruqyah (PBR) Cabang Medan.

2. Data Sekunder

Data sekunder merupakan data-data yang didapatkan melalui studi pustaka seperti buku, e-book, jurnal, dan lain-lain. Data yang didapatkan tentunya berupa teori yang berhubungan atau dijadikan referensi dalam pembuatan aplikasi pengobatan tradisional berbasis web serta rancangan strategi bisnis yang akan diterapkan dalam bentuk *Business Model Canvas* (BMC) pada Pusat Bekam-Ruqyah (PBR) Cabang Medan.

3.4.3. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *Waterfall* (air terjun). Adapun langkah-langkah pada metode *Waterfall* adalah sebagai berikut:



Gambar 3.2. Metode *Waterfall*

1. *Requirement*

Pada tahap ini, pengumpulan data dapat diperoleh dengan observasi, wawancara, studi pustaka. Hal ini untuk mendapatkan data-data yang diperlukan.

2. *System Design*

Pada tahap ini, dilakukan hubungan antar tabel, desain sistem serta persyaratan dan spesifikasi perangkat keras yang digunakan.

3. *Implementation*

Pada tahap ini, sistem akan dikembangkan sesuai desain yang telah ditentukan dan menggunakan kode program *PHP*, *HTML*, *MySQL* sebagai *database* dan *Laravel* sebagai *framework*.

4. *Verification*

Pada tahap ini, dilakukan revisi jika pada sistem adanya malfungsi atau kesalahan pada sistem.

5. *Maintenance*

Pada tahapan terakhir adalah *maintenance*. Sistem sudah sesuai dengan kebutuhan yang diperlukan dan dapat menyelesaikan masalah. Perangkat lunak dapat dijalankan dan dipelihara. Meningkatkan implementasi sistem dan meningkatkan layanan adalah kebutuhan baru.

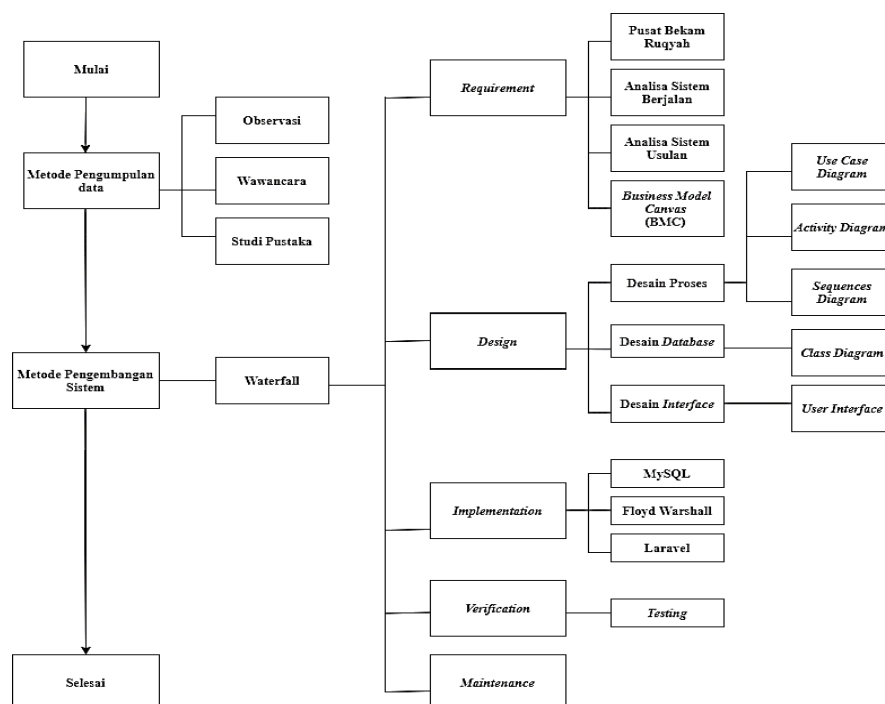
3.4.4 Analisis *Business Model Canvas* (BMC)

Dalam penelitian ini, penulis mengidentifikasi strategi bisnis yang diterapkan sekarang atau yang sedang berjalan. Lalu menyajikan dalam bentuk *Business Model Canvas* (BMC) yang sekarang (*current BMC*) . Setelah itu, penulis mengevaluasi BMC dengan teknik analisa deskriptif sehingga menghasilkan sebuah model bisnis baru yaitu *future BMC*. Berikut merupakan 9 elemen pada *Business Model Canvas* (BMC):

1. *Customer Segments* menjelaskan siapa yang akan menggunakan jasa pada Medis Reborn.
2. *Value Propositions* menjelaskan keunggulan yang ditawarkan Medis Reborn dibanding *competitor* lain

3. *Channels* menjelaskan bagaimana masyarakat atau pelanggan mengetahui jasa yang ditawarkan Medis Reborn.
4. *Customer Relationships* menjelaskan bagaimana Medis Reborn *connect* (terhubung) dengan pelanggan serta memastikan kepuasan pelanggan terhadap jasa yang ditawarkan.
5. *Revenue Streams* menjelaskan cara Medis Reborn menghasilkan uang.
6. *Key Resources* menjelaskan sumber daya utama seperti aset yang digunakan untuk menjalankan Medis Reborn.
7. *Key Activities* menjelaskan aktivitas kunci yang dilakukan pada Medis Reborn.
8. *Key Partnerships* menjelaskan mitra kerja Medis Reborn seperti *supplier* atau *vendor*.
9. *Cost Structure* menjelaskan komponen biaya yang dikeluarkan untuk menjalankan Medis Reborn.

3.4.5 Kerangka Berpikir



Gambar 3.3 Kerangka Berpikir

BAB IV

HASIL DAN PEMBAHASAN

4.1. Requirement

Tahapan *Requirement* atau analisa kebutuhan perangkat lunak adalah fase perencanaan yang terdiri dari beberapa tahapan yaitu profil pusat bekam ruqyah cabang Medan, analisa sistem berjalan dan analisa sistem usulan.

4.1.1. Profil Pusat Bekam Ruqyah Cabang Medan

Pusat Bekam Ruqyah terletak di jalan di Jalan Prof. H. M. Yamin nomor 305 Medan Perjuangan, Medan. Pusat Bekam Ruqyah telah berdiri selama kurang lebih 10 tahun. Visi dari Pusat Bekam Ruqyah Medan adalah mencari ridho Allah. Adapun misi dari Pusat Bekam Ruqyah Medan adalah sebagai berikut:

1. ‘Menegakkan tauhid melalui pengobatan.
2. Menjadikan ruqyah dan bekam yang paling utama untuk mendapatkan kesembuhan kesehatan.
3. Menjadi pusat pengobatan islami yang modern, profesional, terbaik dan terbesar di Indonesia.
4. Melahirkan *entrepreneur* beriman dan bertakwa.
5. Menjadi garda terdepan dalam melaksanakan dan mendukung aktifitas dakwah.’

Pada Pusat Bekam Ruqyah cabang medan terdiri dari 3 layanan pengobatan tradisional yaitu, bekam, ruqyah, dan refleksi atau pijat. Harga dari masing-masing pengobatan adalah Rp. 65000,-. Adapun jumlah karyawan yang bekerja adalah 15 orang yang terdiri dari laki-laki dan perempuan. Tersedia 4 bed untuk laki-laki dan 3 bed untuk perempuan. Jika pelanggan ingin berkunjung bisa datang setiap hari pada pukul 07.00 hingga 22.00 WIB.

4.1.2. Sejarah Pusat Bekam Ruqyah

Pusat Bekam Ruqyah berawal pada tahun 2006 oleh Ustadz Yusman Dawolo M.Kom.I yang mulai mendalami pengobatan bekam. Tahun 2007 Ustadz

Yusman Dawolo M.Kom.I magang di salah satu tempat ruqyah dan bekam terkenal di Bekasi kemudian diangkat menjadi karyawan.

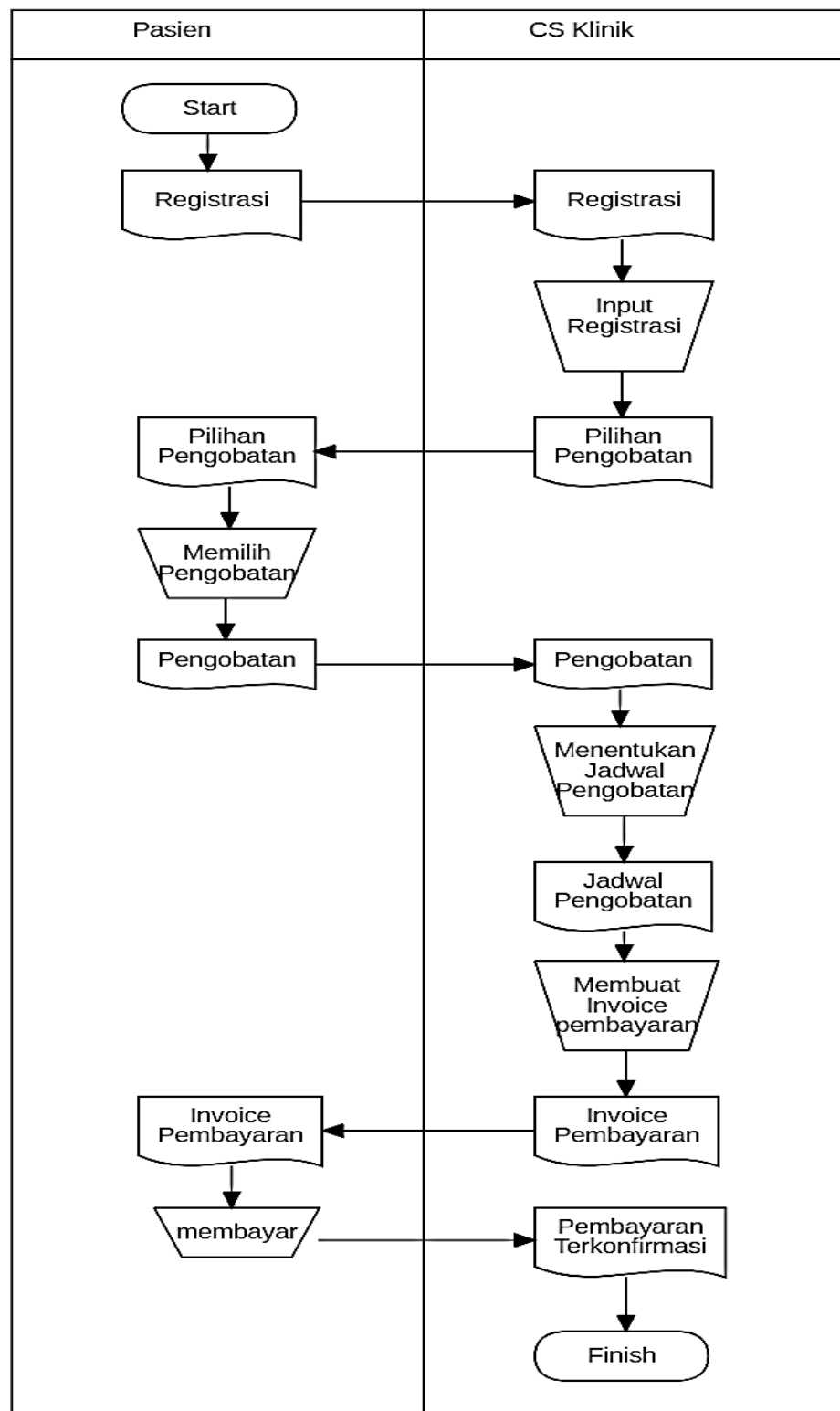
Pada tanggal 15 juli 2009 Ustadz Yusman Dawolo M.Kom.I beserta Yulianingsih, istrinya membuka Pusat Bekam Ruqyah di jalan kebon jeruk raya nomor 12, Jakarta Barat. Saat ini Pusat Bekam Ruqyah (PBR) sudah tersebar sebanyak 12 cabang di JABODETABEK, dan 2 cabang di Sumatra Utara. Komitmen Pusat Bekam Ruqyah adalah untuk menjadi yang terbesar dan terbaik di Indonesia.



Gambar 4.1 Gedung Sewa Pusat Bekam Ruqyah Cabang Medan

4.1.3. Analisa Sistem Berjalan

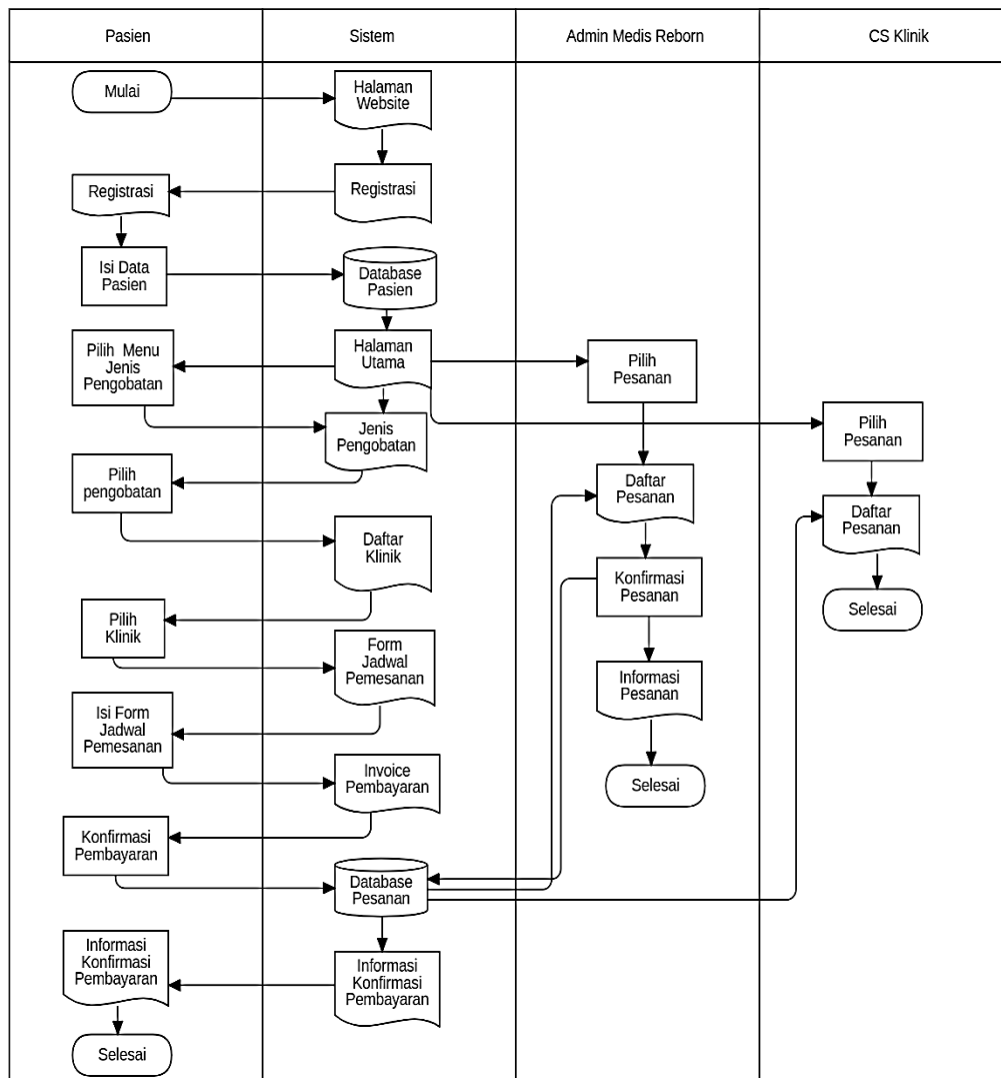
Untuk mengetahui secara detail kelemahan dari suatu sistem secara keseluruhan maka perlu dilakukan analisa sistem berjalan. Berdasarkan observasi yang dilakukan penulis, kelemahan pada sistem yang sedang berjalan pada Pusat Bekam Ruqyah cabang Medan adalah tidak adanya sistem pemesanan jadwal untuk melakukan pengobatan sehingga *customer* harus mengantri dan sulit dalam pembuatan laporan keuangan. CS klinik masih menggunakan buku untuk mencatat data *customer*. Pembayaran pengobatan juga dilakukan secara manual. Berikut *Flowchart* sistem berjalan pada Pusat Bekam Ruqyah (PBR) cabang Medan.



Gambar 4.2 *Flowchart* Sistem Berjalan

4.1.4. Analisa Sistem Usulan

Penulis merancang sebuah sistem usulan yaitu sebuah sistem yang dapat menghubungkan antara *customer* dan klinik agar *customer* dapat memilih lokasi klinik terdekat dari lokasi mereka serta memesan jadwal untuk melakukan pengobatan secara online sehingga *customer* tidak harus mengantri. Untuk klinik, klinik tidak harus menginputkan data pemesanan pengobatan dan dapat melihat laporan keuangan atau total pendapatan klinik. Berikut *flowchart* sistem usulan Pusat Bekam Ruqyah (PBR) cabang Medan:



Gambar 4.3 Flowchart Sistem Usulan

4.1.5. *Business Model Canvas (BMC) Pada Startup Medis Reborn*

Business Model Canvas merupakan kerangka kerja untuk mendefinisikan model bisnis *startup*. *Business Model Canvas* adalah bagian penting dalam perencanaan bisnis agar bisnis dapat berkembang. Oleh karena itu, *startup* medis reborn membutuhkan suatu rancangan bisnis model agar *startup* Medis Reborn dapat diketahui dan digunakan oleh masyarakat sehingga menghasilkan keuntungan. Rancangan model bisnis yang digunakan penulis adalah *Business Model Canvas (BMC)*. BMC terdiri dari sembilan elemen yaitu, *Customer Segments*, *Value Propositions*, *Channels*, *Customer Relationships*, *Revenue Streams*, *Key Resources*, *Key Activities*, *Key Partnerships* dan *Cost Structure*. Berikut rancangan *Business Model Canvas (BMC)* pada *startup* Medis Reborn:

Key Partnerships 1. Klinik pengobatan tradisional 2. Bank	Key Activities 1. Menghubungkan klinik pengobatan tradisional dengan pelanggan 2. Pemesanan jadwal pengobatan Key Resources 1. Hak paten medis reborn 2. Kantor perwakilan 3. Komputer 4. Software engineer	Value Propositions 1. Beragam pilihan jenis pengobatan tradisional 2. Penentuan lokasi klinik terdekat 3. Pemesanan jadwal pengobatan secara online (anti antri) 4. Therapist handal dan berpengalaman 5. Pengobatan Tradisional di Era Modern	Customer Relationships 1. Promo paket pengobatan 2. Fitur “customer complain” 3. Broadcasting via Whatsapp Channels 1. Website Medis Reborn 2. Media Sosial : Facebook dan Instagram	Customer Segments 1. Pelanggan pengobatan tradisional usia 18-60 tahun
Cost Structure 1. Biaya pembuatan sistem 2. Biaya pemeliharaan dan perbaikan sistem 3. Gaji pegawai 4. Sewa gedung 5. Air dan Listrik 6. Internet			Revenue Streams 1. Penjualan jasa pengobatan tradisional 2. Keuntungan 7% dari harga jual	

Gambar 4.4 *Business Model Canvas (BMC) Startup Medis Reborn*

1. *Customer Segments*

Customer segments menggambarkan siapa target *customer* yang akan menggunakan produk atau jasa dari suatu perusahaan. *Customer* atau pelanggan

merupakan inti dari suatu model bisnis karena pelangganlah yang memberikan keuntungan pada perusahaan. Tidak ada perusahaan yang akan bertahan lama tanpa pelanggan. *Customer segments* membantu perusahaan fokus pada pelanggan potensial yang akan mereka layani daripada pelanggan lain yang belum tentu menggunakan produk atau jasanya. Pada Pusat Bekam Ruqyah (PBR) cabang Medan, target pelanggan usaha adalah bapak-bapak, ibu-ibu, dan anak muda hingga lansia. Berdasarkan hasil wawancara, maka *customer segments* yang sesuai bagi *startup* medis reborn adalah masyarakat usia 18-60 Tahun. *Customer segments* ini merupakan jenis *Mass Market* karena beranggapan bahwa semua orang adalah pelanggan.

2. *Value Propositions*

Value propositions adalah nilai yang diberikan oleh perusahaan kepada pelanggan. Pada Pusat Bekam Ruqyah (PBR) cabang Medan, nilai yang diberikan adalah tenaga kerja yang *professional*, bekam rasa cinta yaitu bekam tanpa meninggalkan rasa sakit, ruqyah anti galau yaitu tak kan lagi galau atau sedih setelah pengobatan, dan sudah berpengalaman sejak tahun 2010. Berdasarkan nilai tersebut, maka *value propositions* yang sesuai dengan *startup* medis reborn adalah *newness*, *performance*, *customization*, dan *merk/Brand*.

- Pada elemen *newness*, medis reborn merupakan satu-satunya aplikasi pelayanan pengobatan tradisional dengan berbagai pilihan pengobatan dan pelanggan dapat menentukan lokasi terdekat klinik serta pemesanan jadwal pengobatan secara online sehingga pelanggan tak harus mengantri untuk melakukan pengobatan.
- Pada elemen *performance*, medis reborn memastikan *therapist* handal dan berpengalaman.
- Pada elemen *customization*, medis reborn memungkinkan pelanggan memilih jenis pengobatan yang mereka inginkan dan memilih jadwal pengobatan yang diinginkan sehingga pelanggan tak perlu mengantri.
- Pada elemen *merk/Brand*, “Medis Reborn” secara bahasa berarti medis yang lahir kembali. Makna nya adalah pengobatan tradisional yang mulai populer kembali ditandai dengan banyaknya muncul penelitian tentang

obat-obat herbal. Banyak masyarakat juga mulai sadar tentang efek samping dari mengonsumsi obat-obatan kimia sehingga mereka memilih pengobatan tradisional yang tanpa efek samping. Semboyan dari medis reborn adalah “Pengobatan Tradisional Pada Era Modern”. Medis Reborn bermimpi menggabungkan pengobatan tradisional dengan ilmu pengetahuan/*Science*.

3. *Channels*

Channels merupakan cara sebuah bisnis berkomunikasi dengan pelanggan. Berdasarkan hasil wawancara, pada Pusat Bekam Ruqyah (PBR) cabang Medan mempromosikan usaha dengan menyebar selebaran brosur biasanya pada masjid-masjid. Selain itu, mereka juga memiliki sosial media yaitu *instagram* dan *facebook*. Dari penjelasan sebelumnya maka *channels* yang sesuai dengan *startup* Medis reborn adalah menyediakan *website* untuk memesan pengobatan tradisional yang diinginkan. Dengan adanya *website* medis reborn, klinik-klinik dapat mempromosikan jasa mereka. Media sosial juga solusi yang tepat untuk mempromosikan jasa yang ditawarkan. Brosur yang digunakan merupakan gambar digital yang diunggah di *facebook* dan *instagram* sehingga masyarakat dapat mengenal Medis Reborn.

4. *Customer Relationships*

Customer relationships merupakan cara perusahaan membangun serta menjaga hubungan dengan pelanggan. Pada Pusat Bekam Ruqyah (PBR) cabang Medan, hal yang dilakukan untuk menjaga hubungan baik adalah dengan pelayanan yang baik, *broadcasting* menggunakan pesan *whatssapp*, mengingatkan pelanggan secara langsung untuk melakukan pengobatan selanjutnya, dan melakukan survey tentang kepuasan pelanggan terhadap pelayanan. Dari penjelasan sebelumnya, maka *customer relationships* yang sesuai dengan *startup* medis reborn adalah sebagai berikut:

- Memberikan promo paket pengobatan

- Menyediakan fitur “*customer complain*”. Dengan adanya fitur tersebut, pelanggan dapat memberi tahu medis reborn tentang hal yang mereka keluhkan.
- *Broadcasting* melalui pesan *whatssapp*. Hal ini membuat medis reborn mudah berkomunikasi dengan pelanggan.

5. Revenue Streams

Revenue streams adalah pendapatan atau keuntungan yang diperoleh oleh perusahaan dari pelanggan. Sumber utama pendapatan pada medis reborn adalah *usage fee* yaitu penggunaan jasa pelayanan. Medis Reborn menghitung keuntungan per satu kali transaksi pemesanan yaitu sebanyak 7 % dari harga jual.

6. Key Resources

Key resources menggambarkan aset-aset terpenting yang diperlukan perusahaan agar model bisnis nya bekerja. Berdasarkan hasil wawancara, aset yang digunakan pada Pusat Bekam Ruqyah (PBR) cabang Medan adalah gedung, komputer, tempat tidur, kursi, meja, dan lemari. Pada *startup* medis reborn, *key resource* yang diperlukan adalah hak paten medis reborn, kantor perwakilan, komputer, dan *software engineer*.

7. Key Activities

Key activities menjelaskan kegiatan utama yang dilakukan perusahaan. Pada *startup* medis reborn, aktivitas utama yang dilakukan adalah pemesanan jadwal pengobatan tradisional. Medis reborn menghubungkan klinik pengobatan tradisional dengan pelanggan. *Key activities* medis reborn termasuk ke dalam kategori operasi jasa.

8. Key Partnerships

Key partnerships adalah mitra kerja utama perusahaan. Namun pada *startup* medis reborn, *key partnership* yang sesuai adalah klinik pengobatan tradisional dan Bank.

9. Cost Structure

Cost structure menjelaskan biaya yang dikeluarkan agar dapat mengoperasikan model bisnis. Biaya yang dikeluarkan pada *startup* medis reborn

yaitu biaya pembuatan sistem, biaya pemeliharaan dan perbaikan sistem, gaji pegawai, sewa gedung, air dan listrik, dan biaya internet.

4.2. Design

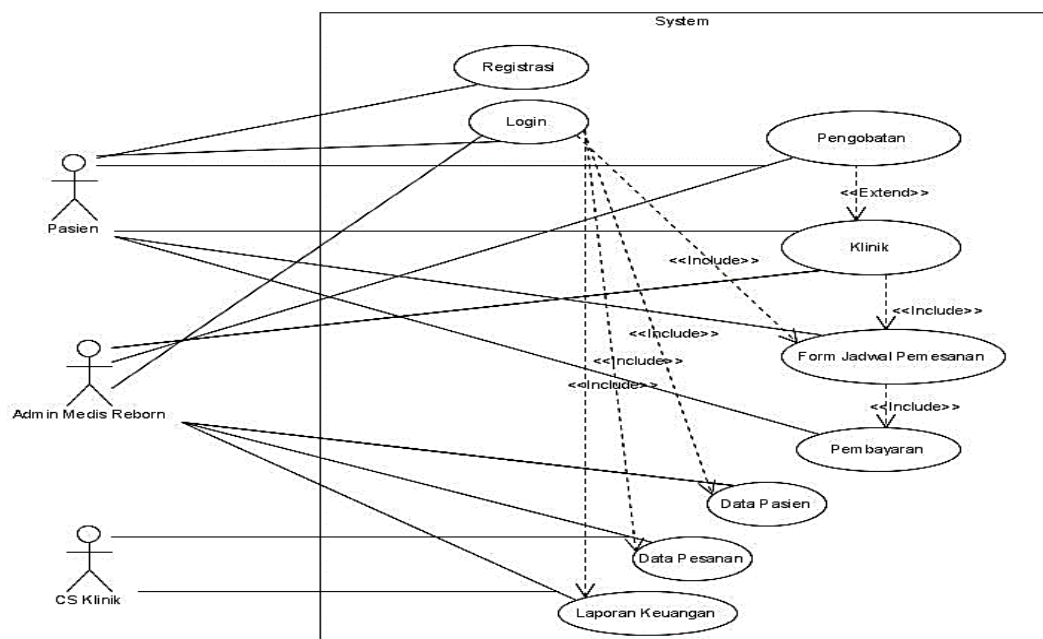
Setelah menganalisa kebutuhan sistem, hal yang dilakukan selanjutnya adalah desain sistem. Dalam tahap desain sistem terdiri dari desain proses, desain *database*, dan perancangan *user interface*.

4.2.1. Desain Proses

Pada tahap desain proses, penulis menggunakan teknik pemodelan *UML* (*Unified Modelling Language*) untuk memudahkan dalam menganalisa pembuatan suatu sistem. Desain proses sistem akan digambarkan dalam 3 diagram, yaitu *Usecase Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

4.2.1.1. Use Case Diagram

Use case diagram adalah diagram untuk menggambarkan hubungan aktor yang terlibat dengan sistem yang akan dibuat. Berikut rancangan *use case diagram startup Medis Reborn*:

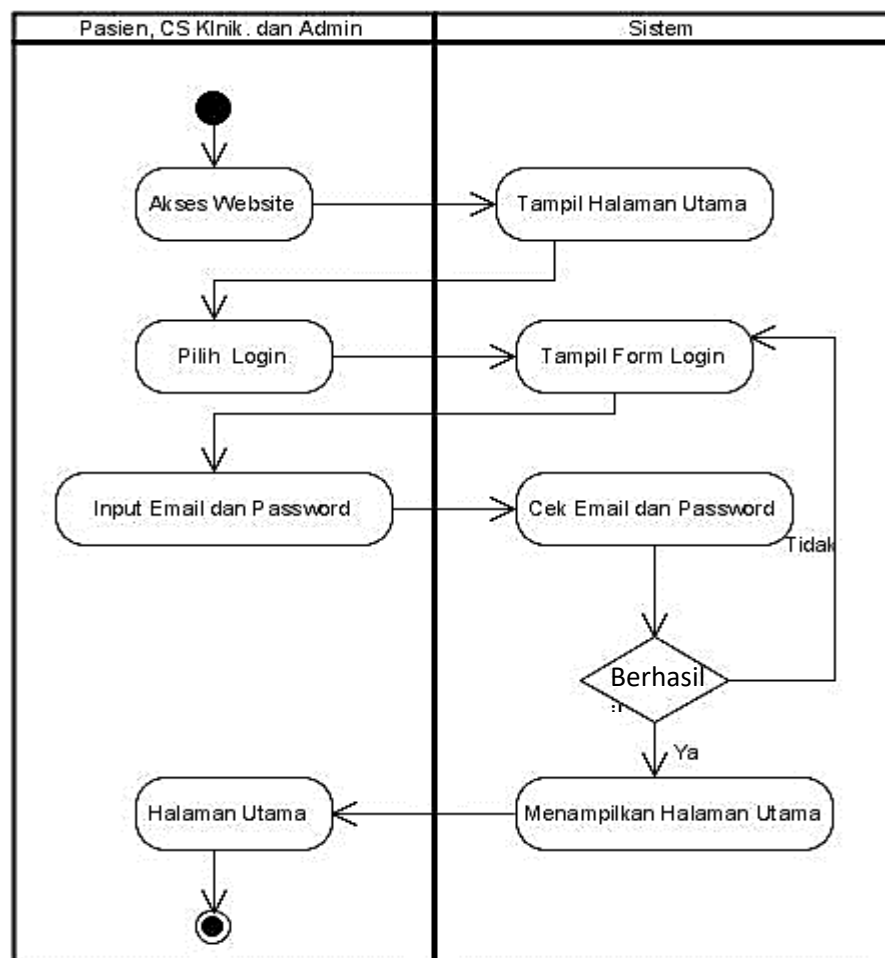


Gambar 4.5 *Use Case Diagram* Medis Reborn

Use case diagram diatas menggambarkan bahwa ada 3 aktor yang terlibat dalam sistem yaitu pasien, CS klinik, dan admin Medis Reborn. Aktor pasien dapat melakukan registrasi, *login*, memilih jenis pengobatan, memilih lokasi klinik terdekat, melakukan pemesanan, dan melakukan pembayaran. Untuk aktor CS klinik, hanya dapat melihat list jadwal pemesanan serta dapat melihat laporan keuangan. Aktor ketiga yaitu admin medis reborn. Admin dapat mengelola data pesanan, mengelola data klinik, mengelola data keuangan, melihat data pasien, serta mengelola jenis pengobatan.

4.2.1.2. Activity Diagram Registrasi Pasien

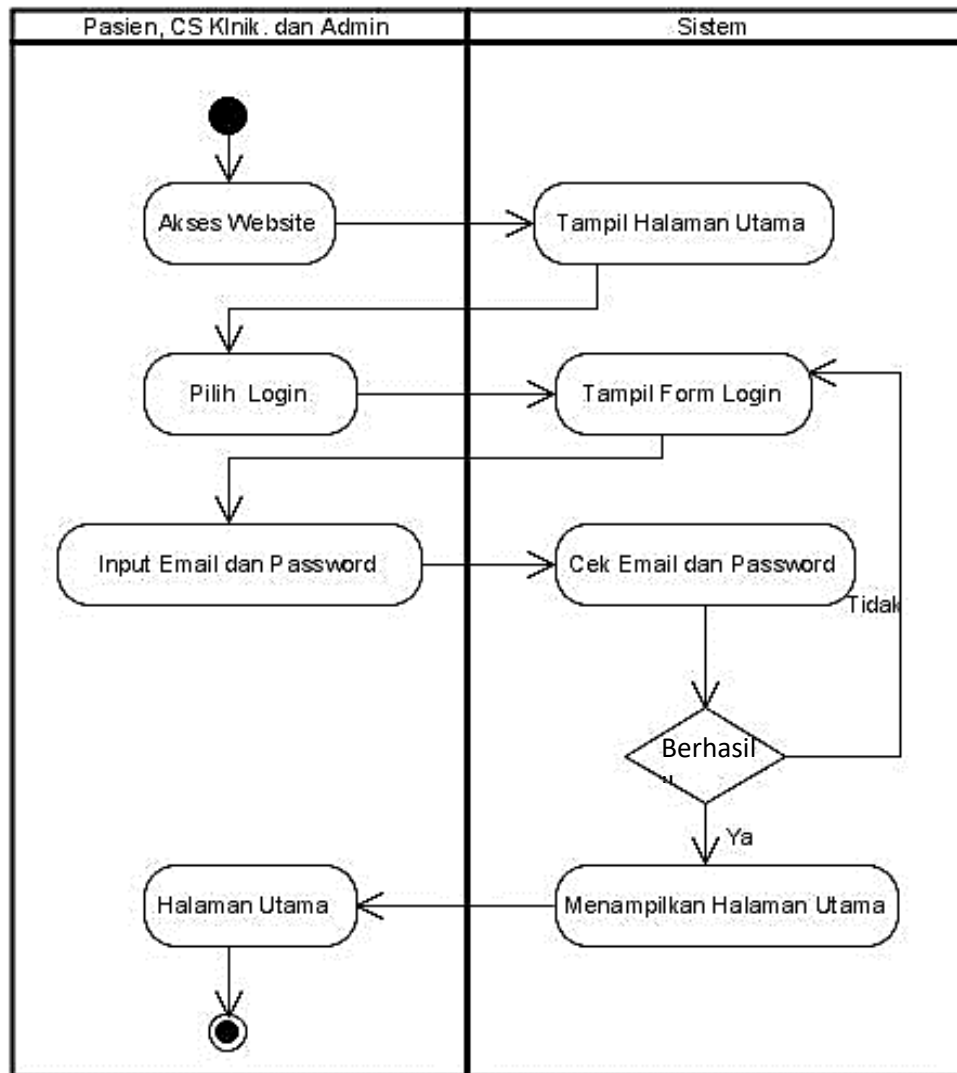
Aktifitas pasien untuk melakukan registrasi pada *website* Medis Reborn dapat dilihat pada gambar *activity diagram* registrasi pasien berikut:



Gambar 4.6 Activity Diagram Registrasi Pasien

4.2.1.3. Activity Diagram Login

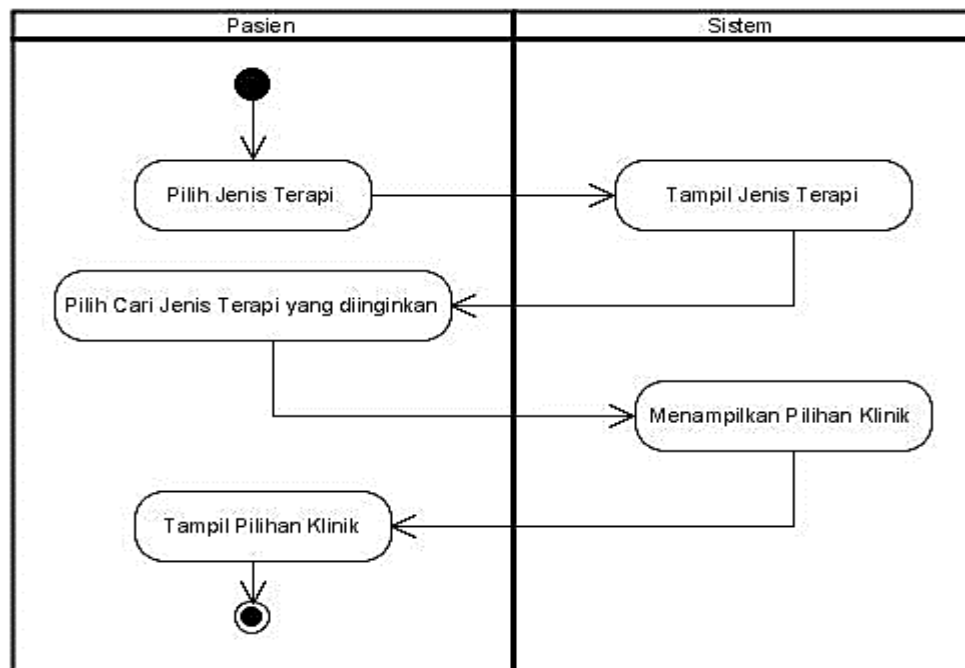
Activity diagram login menggambarkan proses login aktor yaitu pasien, CS Klinik, serta Admin Medis Reborn dengan menginput *email* dan *password*. Berikut gambar rancangan Activity diagram login:



Gambar 4.7 Activity Diagram Login

4.2.1.4. Activity Diagram Pilih Pengobatan

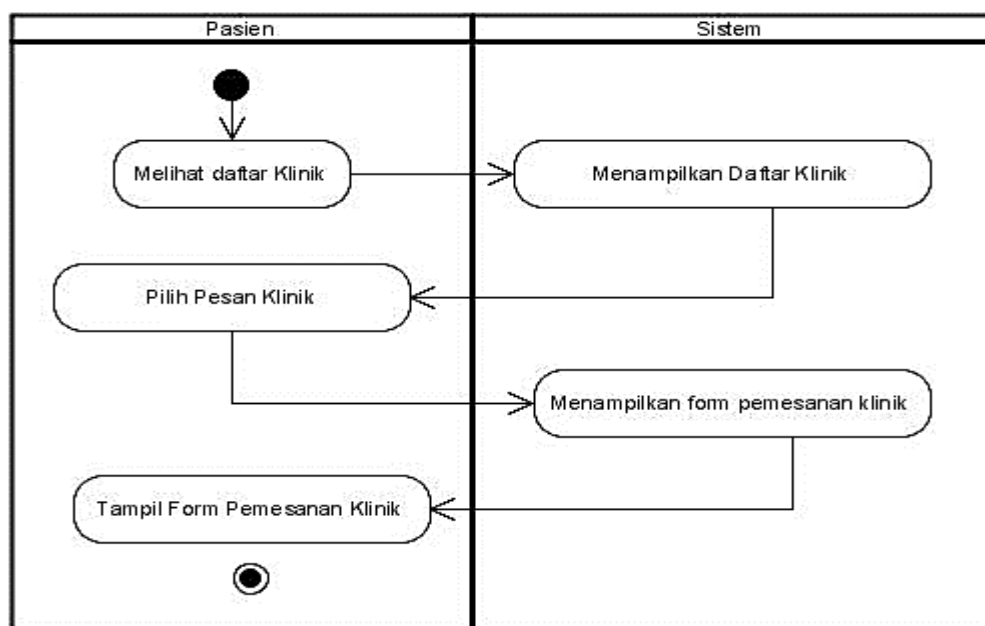
Aktifitas pasien sebelum melakukan pemesanan adalah memilih jenis pengobatan yang ingin dilakukan. Berikut gambar *activity diagram* pilih pengobatan:



Gambar 4.8 *Activity Diagram* Pilih Pengobatan

4.2.1.5. *Activity Diagram* Pilih Klinik

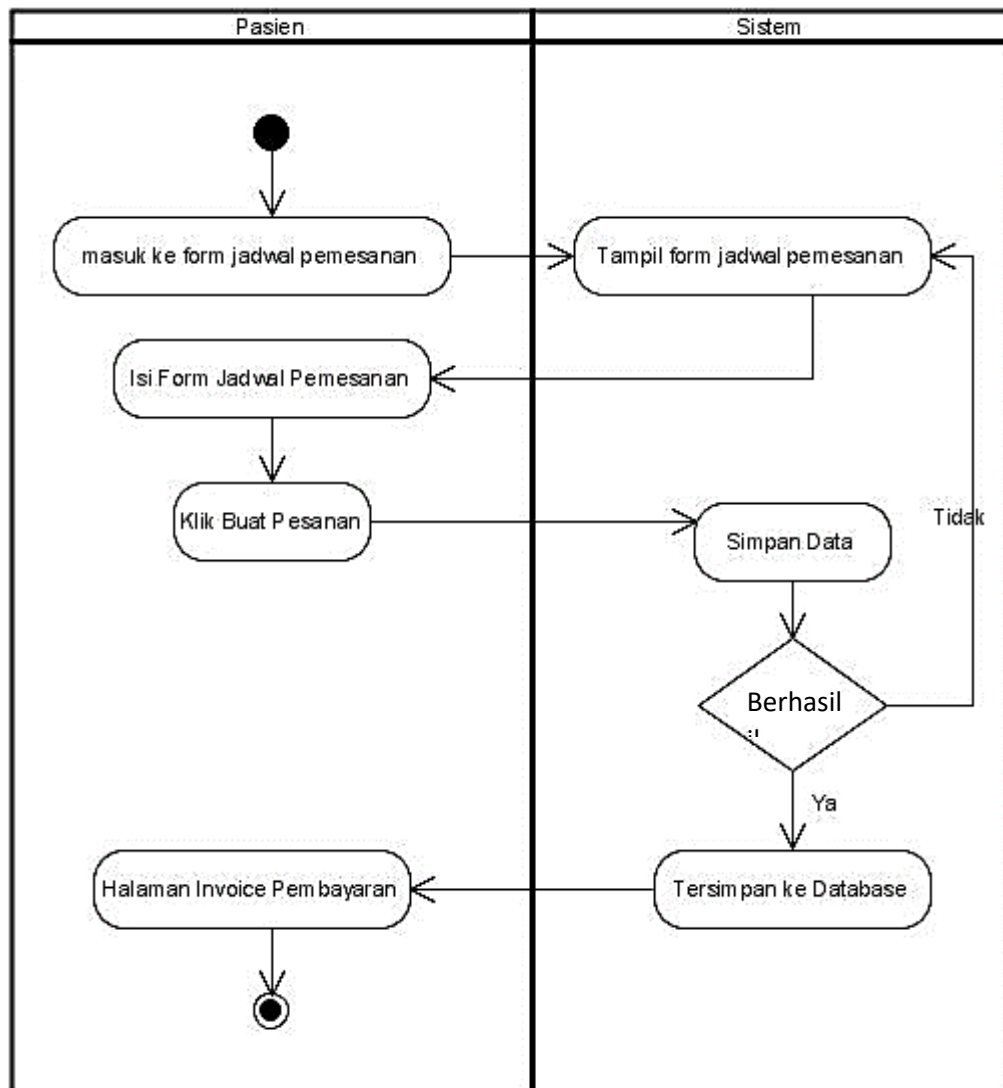
Aktifitas pasien sebelum melakukan pemesanan adalah memilih klinik yang ingin dikunjungi. Berikut gambar *activity diagram* pilih klinik:



Gambar 4.9 *Activity Diagram* Pilih Klinik

4.2.1.6. Activity Diagram Form Jadwal Pengobatan

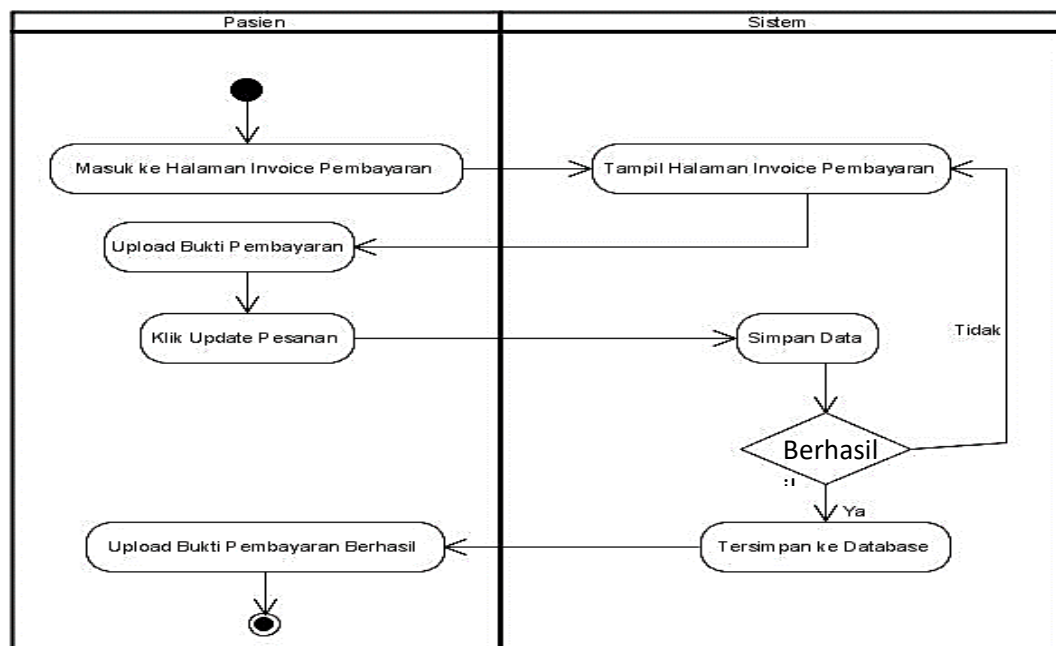
Aktifitas pasien sebelum membuat pesanan adalah mengisi formulir jadwal pengobatan. Berikut gambar *activity diagram form* jadwal pengobatan:



Gambar 4.10 Activity Diagram Form Jadwal Pengobatan

4.2.1.7. Activity Diagram Pembayaran

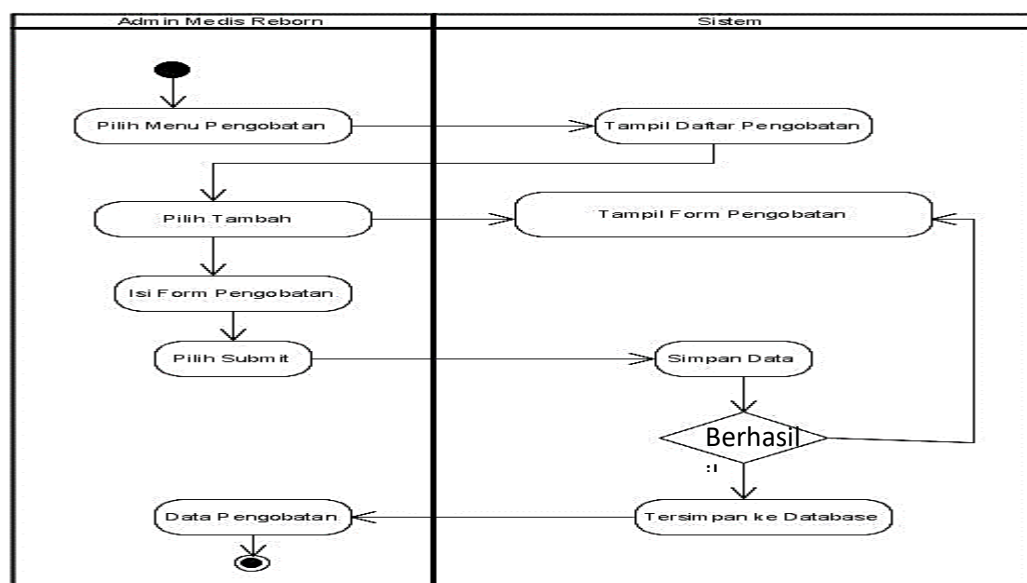
Aktifitas pasien pada menu pembayaran adalah melakukan upload bukti pembayaran sesuai dengan *invoice* yang diterima. Pasien menerima *invoice* pembayaran setelah melakukan. Berikut gambar *activity diagram* pembayaran:



Gambar 4.11 Activity Diagram Pembayaran

4.2.1.8. Activity Diagram Tambah Pengobatan

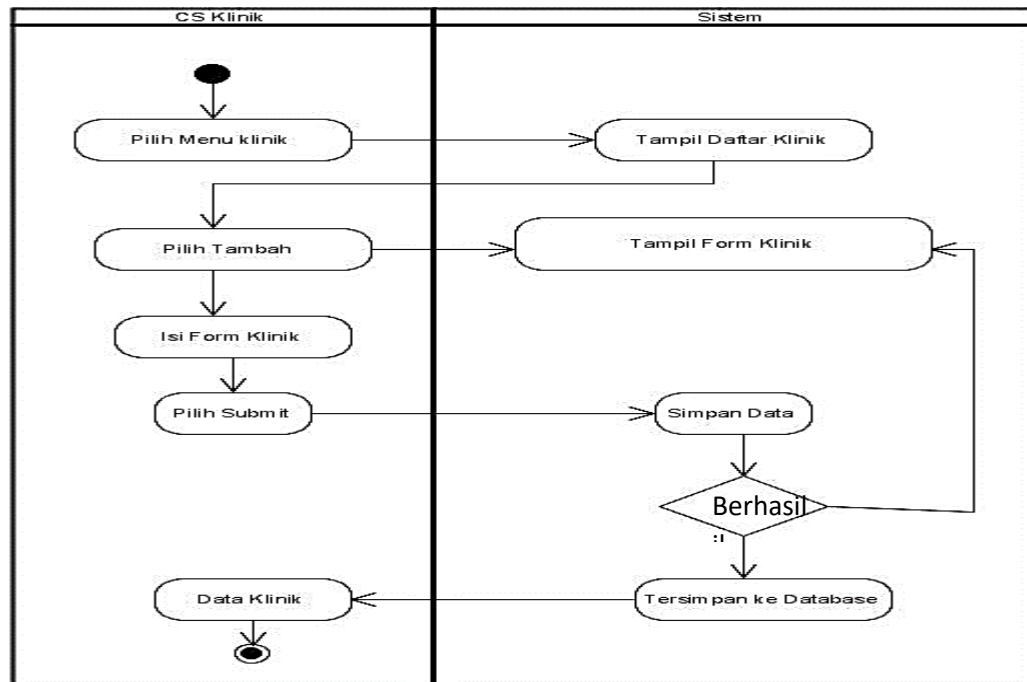
Berikut Activity Diagram tambah pengobatan yang dapat diinputkan oleh admin medis reborn:



Gambar 4.12 Activity Diagram Tambah Pengobatan

4.2.1.9. Activity Diagram Tambah Klinik

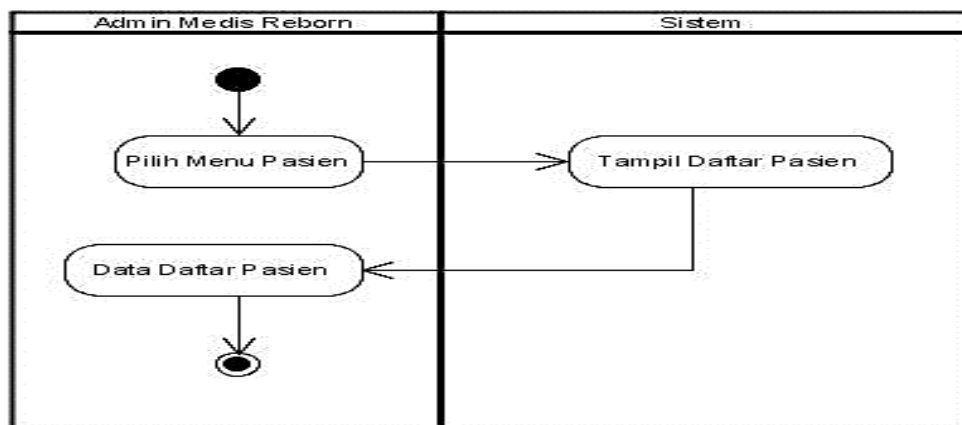
Berikut *Activity Diagram* tambah pengobatan yang dapat diinputkan oleh admin medis reborn:



Gambar 4.13 Activity Diagram Tambah Klinik

4.2.1.10 Activity Diagram Data Pasien

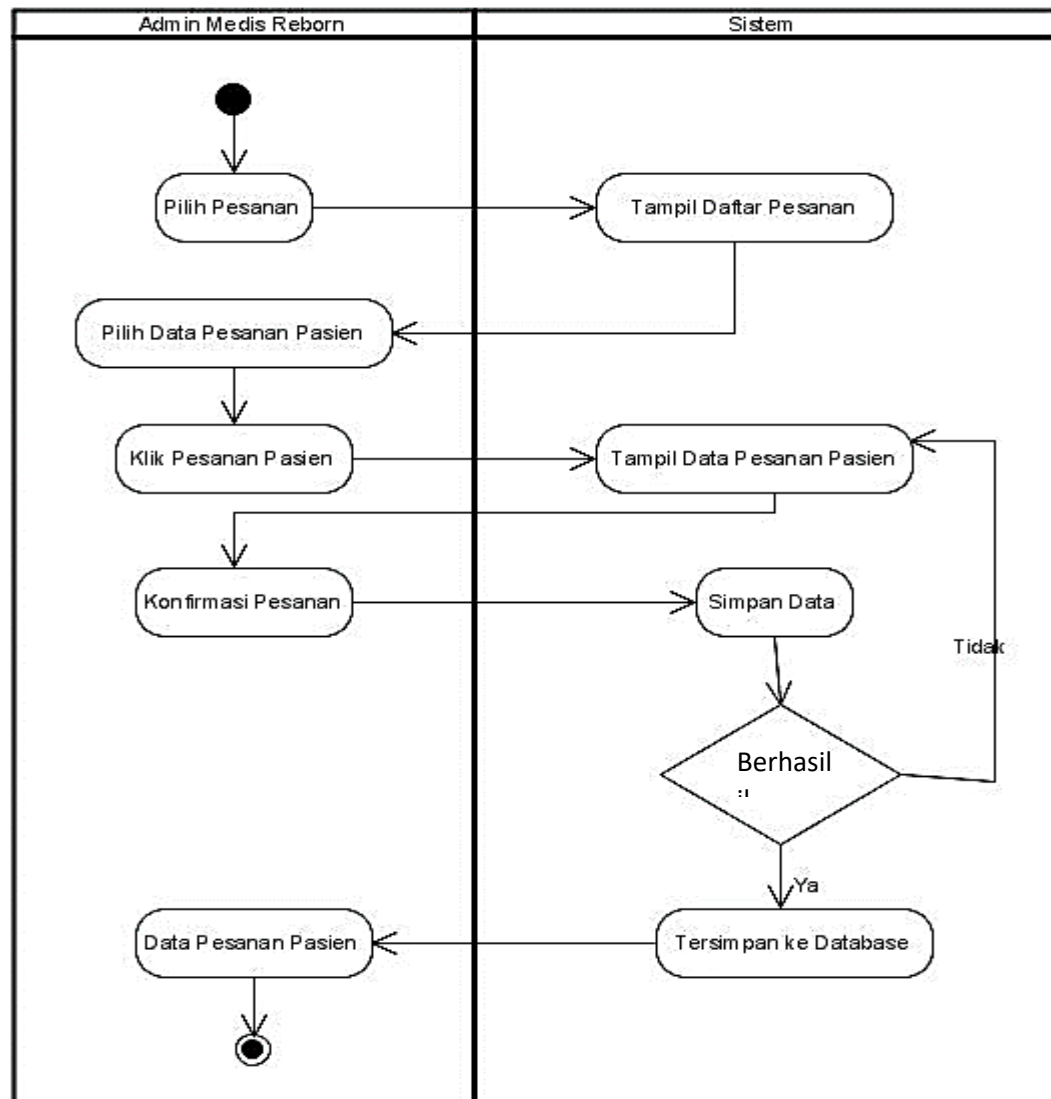
Berikut *Activity Diagram* data pasien yang dapat dilihat oleh admin medis reborn:



Gambar 4.14 Activity Diagram Data Pasien

4.2.1.11 Activity Diagram Konfirmasi Pesanan

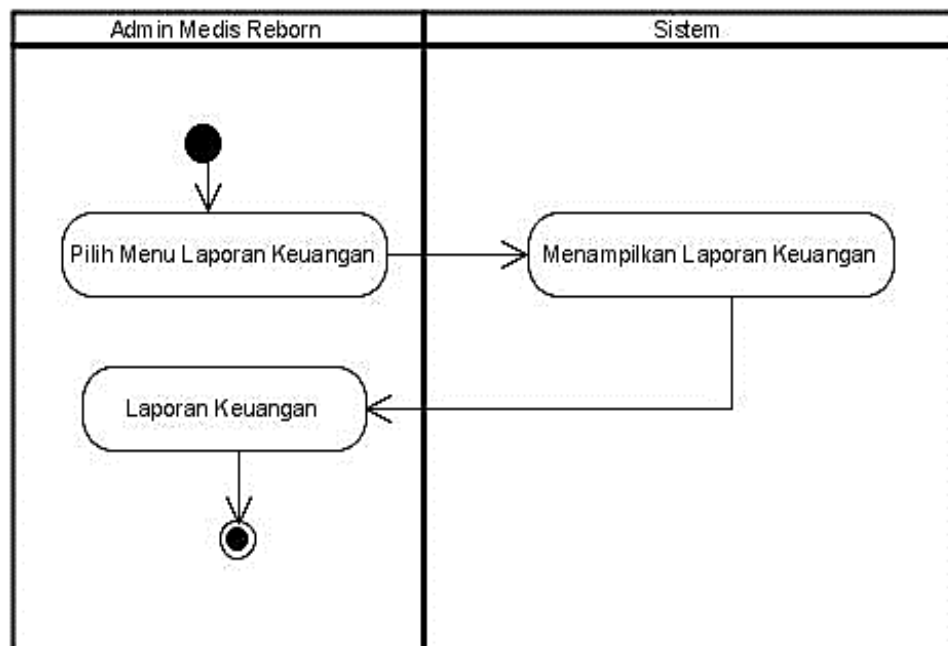
Berikut *Activity Diagram* konfirmasi pesanan yang dapat diinputkan oleh admin medis reborn:



Gambar 4.15 *Activity Diagram* Konfirmasi Pesanan

4.2.1.12 Activity Diagram Laporan Keuangan Medis Reborn

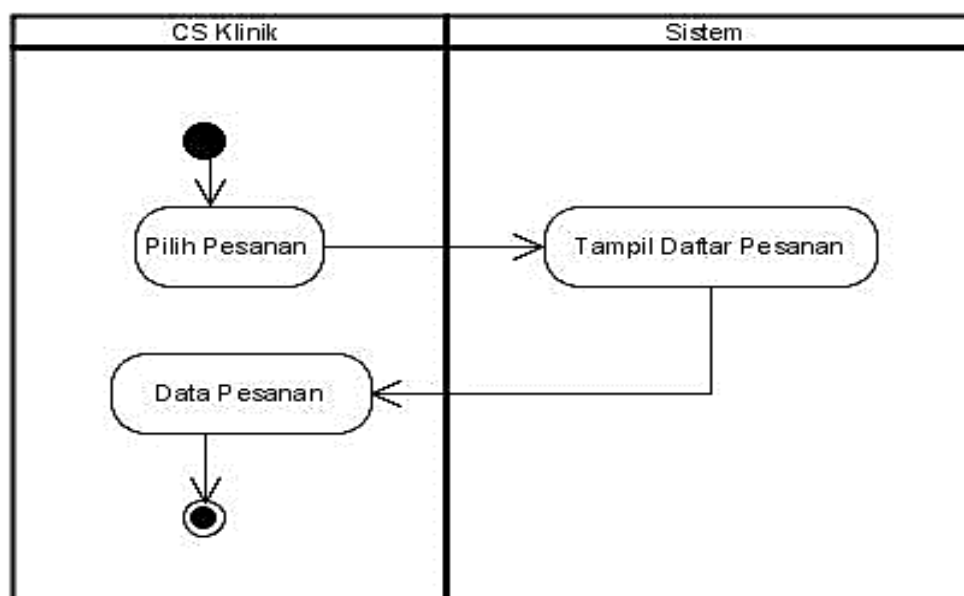
Berikut *Activity Diagram* melihat laporan keuangan medis reborn yang dapat dilihat oleh admin medis reborn:



Gambar 4.16 *Activity Diagram* Laporan Keuangan Medis Reborn

4.2.1.13 *Activity Diagram* Pesanan Bagi Klinik

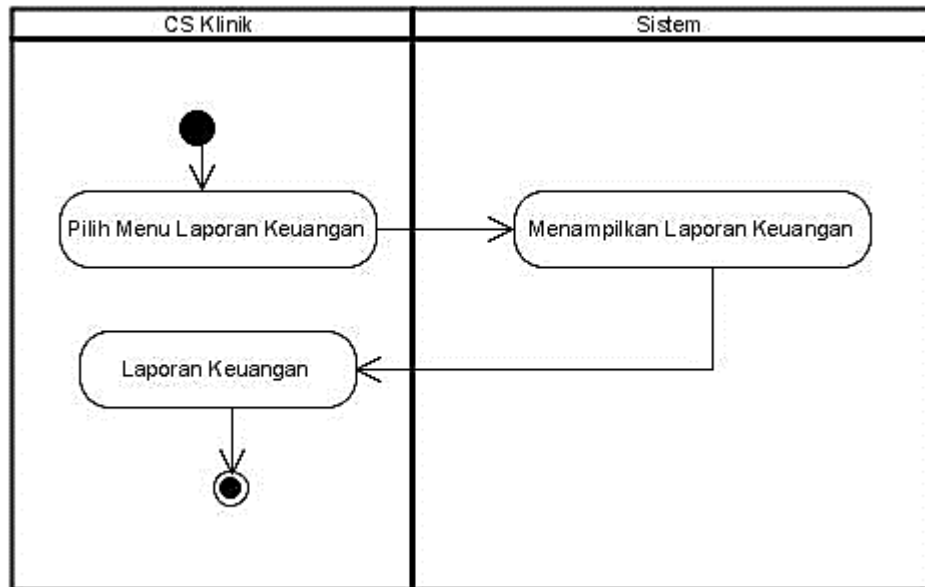
Berikut *Activity Diagram* untuk melihat data pesanan bagi klinik oleh CS Klinik:



Gambar 4.17 *Activity Diagram* Pesanan Bagi Klinik

4.2.1.14 Activity Diagram Laporan Keuangan Klinik

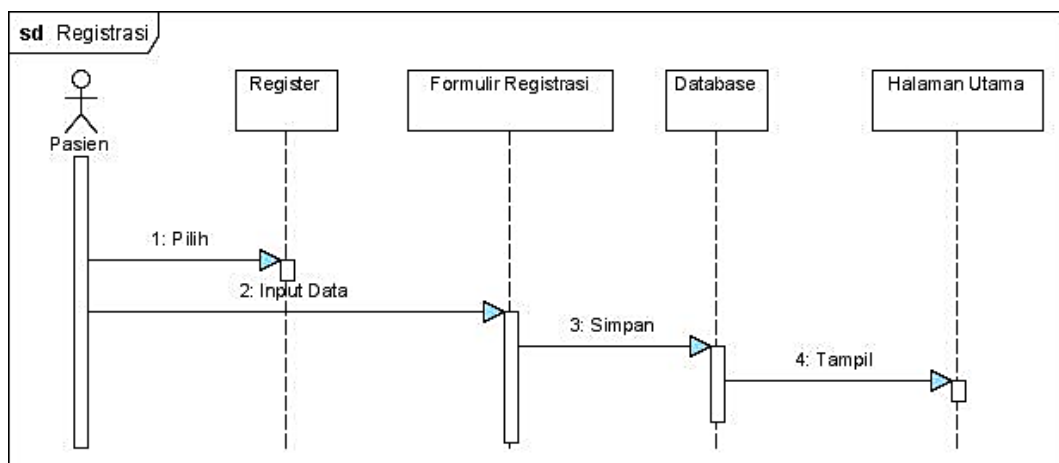
Berikut *Activity Diagram* melihat laporan keuangan medis reborn yang dapat dilihat oleh admin medis reborn:



Gambar 4.18 Activity Diagram Laporan Keuangan Klinik

4.2.1.15 Sequence Diagram Registrasi Pasien

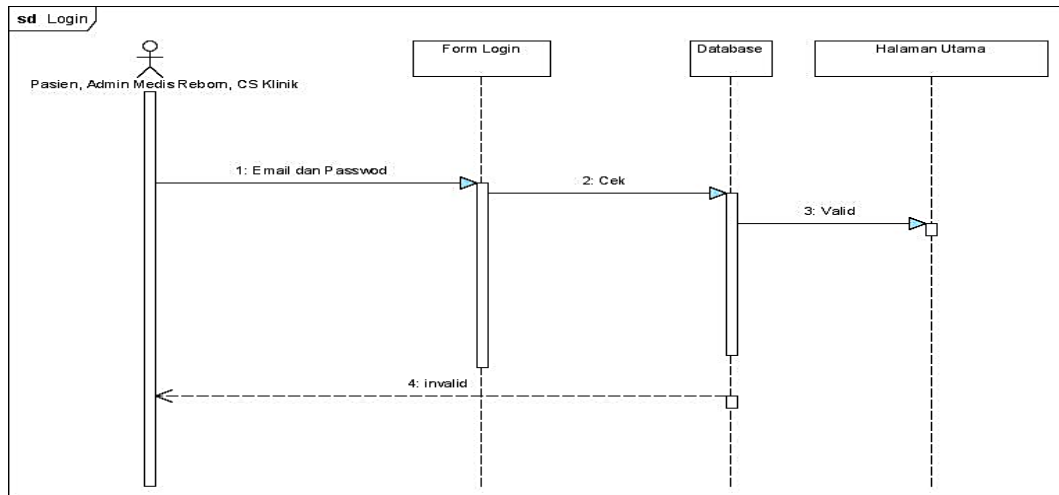
Sequence diagram untuk registrasi pasien pada *startup* Medis Reborn adalah sebagai berikut:



Gambar 4.19 Sequence Diagram Registrasi Pasien

4.2.1.16 Sequence Diagram Login

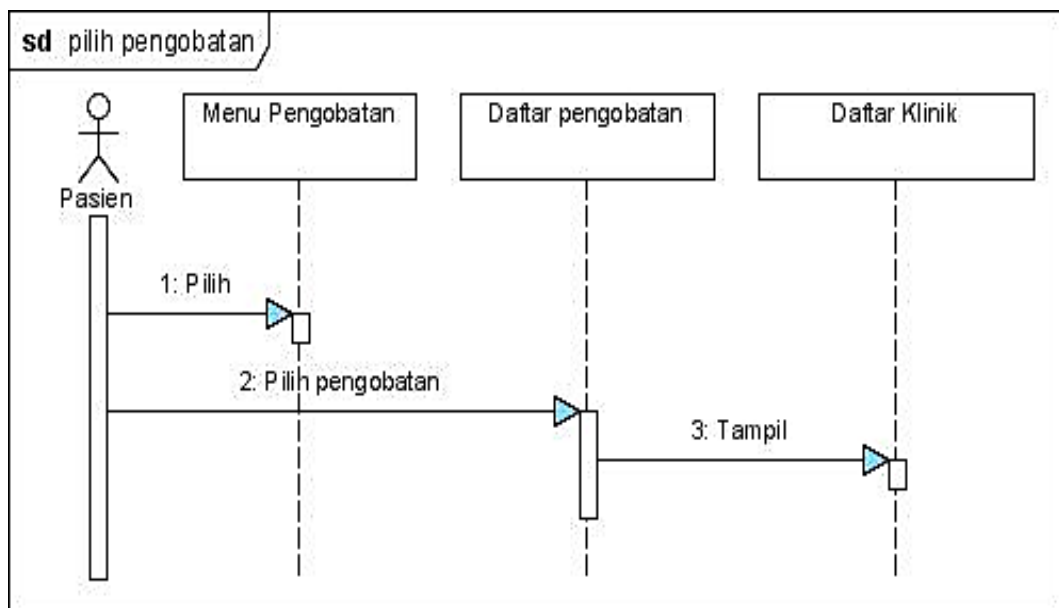
Sequence diagram untuk login sebagai pasien, admin medis reborn, dan CS Klinik pada startup Medis Reborn adalah sebagai berikut:



Gambar 4.20 Sequence Diagram Login

4.2.1.17 Sequence Diagram Pilih Pengobatan

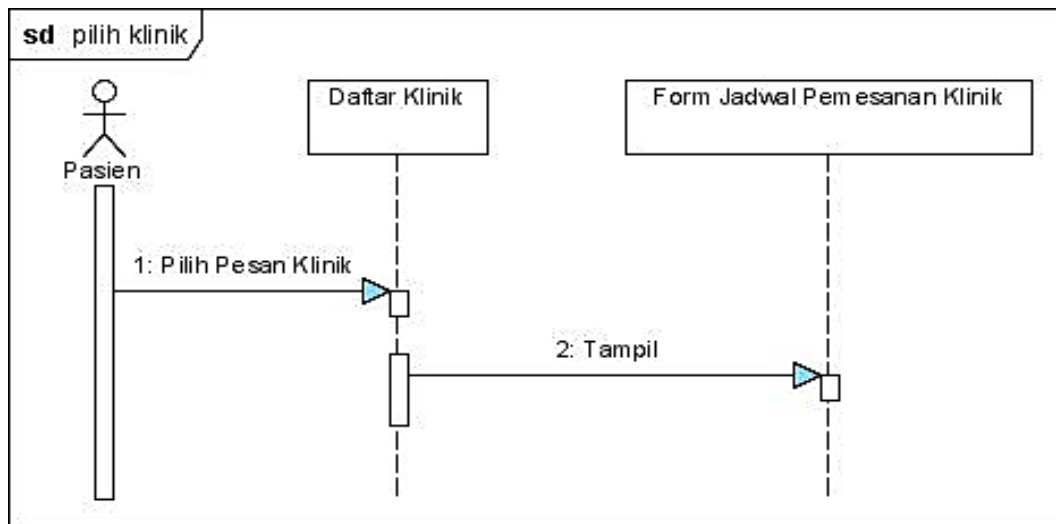
Sequence diagram untuk pilih pengobatan yang diinginkan oleh pasien adalah sebagai berikut:



Gambar 4.21 Sequence Diagram Pilih Pengobatan

4.2.1.18 Sequence Diagram Pilih Klinik

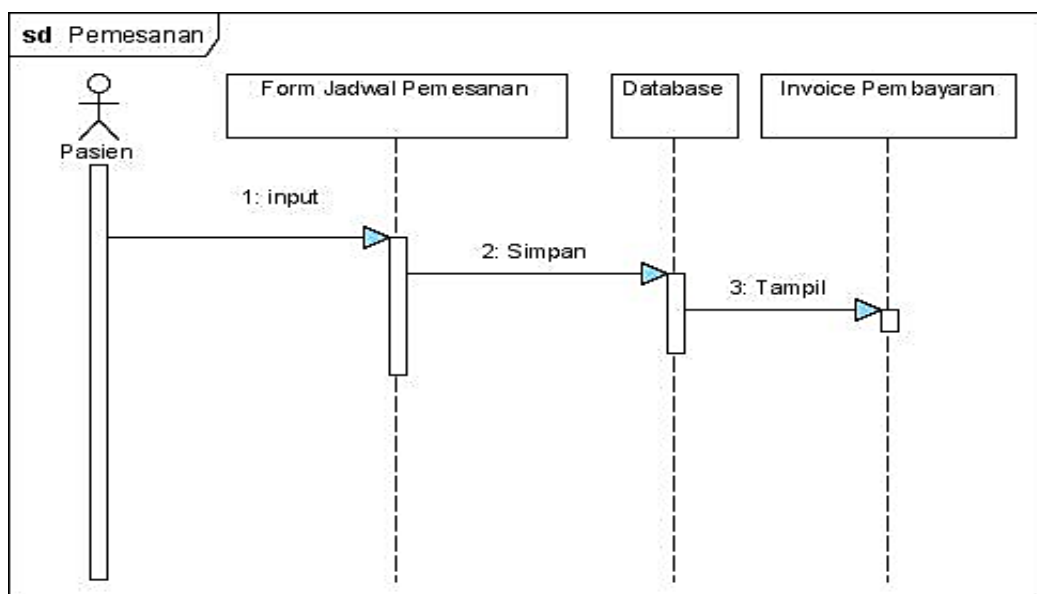
Sequence diagram untuk pilih klinik yang ingin dikunjungi oleh pasien adalah sebagai berikut:



Gambar 4.22 *Sequence Diagram Pilih Klinik*

4.2.1.19 Sequence Diagram Form Jadwal Pengobatan

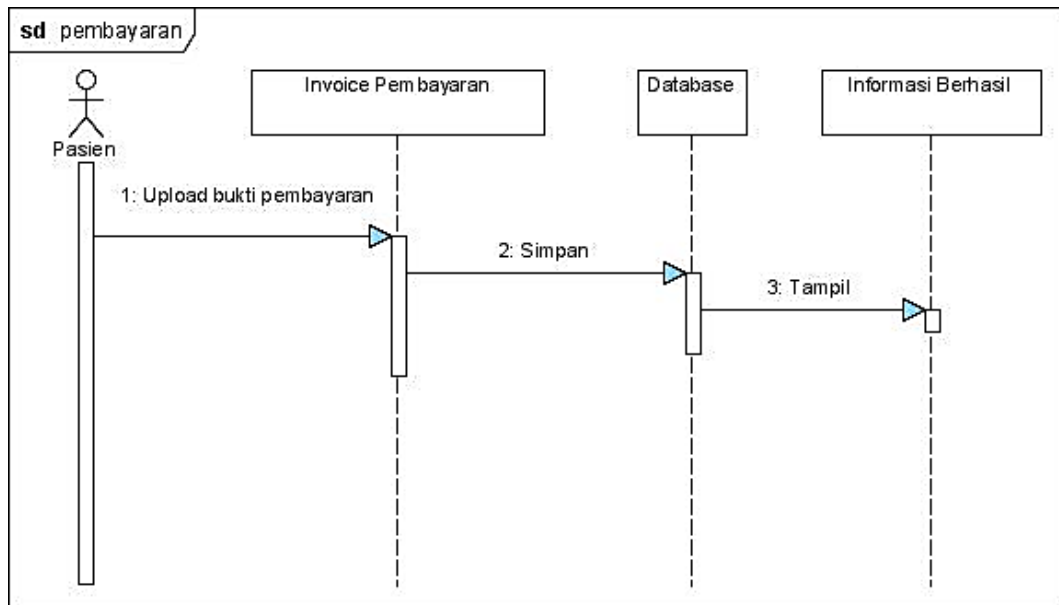
Sequence diagram untuk pasien mengisi *form* jadwal pengobatan adalah sebagai berikut:



Gambar 4.23 *Sequence Diagram Form Jadwal Pengobatan*

4.2.1.20 Sequence Diagram Pembayaran

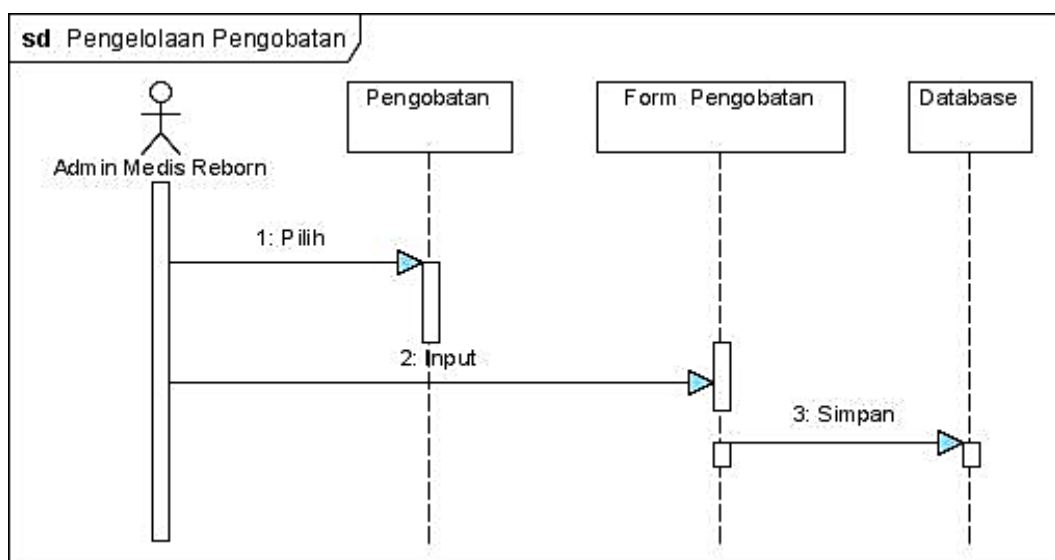
Sequence diagram untuk pembayaran transaksi adalah sebagai berikut:



Gambar 4.24 Sequence Diagram Pembayaran

4.2.1.21 Sequence Diagram Tambah Pengobatan

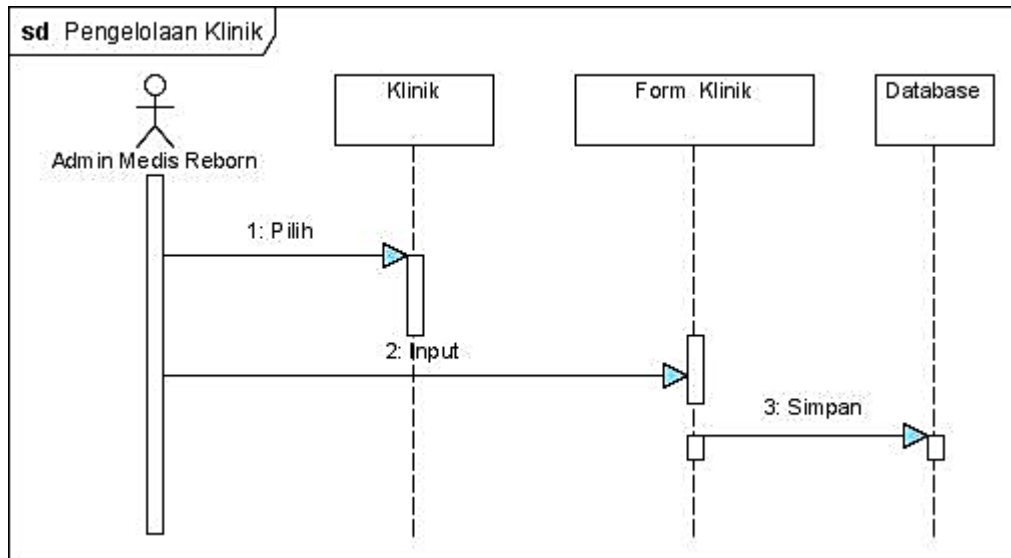
Sequence diagram untuk melakukan penambahan pengobatan tradisional oleh admin medis reborn adalah sebagai berikut:



Gambar 4.25 Sequence Diagram Tambah Pengobatan

4.2.1.22 Sequence Diagram Tambah Klinik

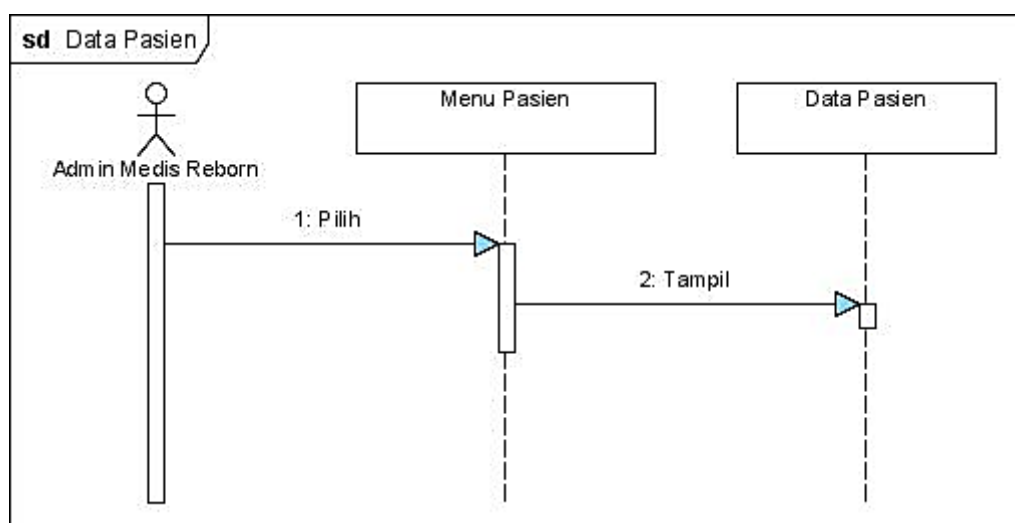
Sequence diagram untuk melakukan penambahan pengobatan tradisional pada admin medis reborn adalah sebagai berikut:



Gambar 4.26 *Sequence Diagram* Tambah Klinik

4.2.1.23 Sequence Diagram Data Pasien

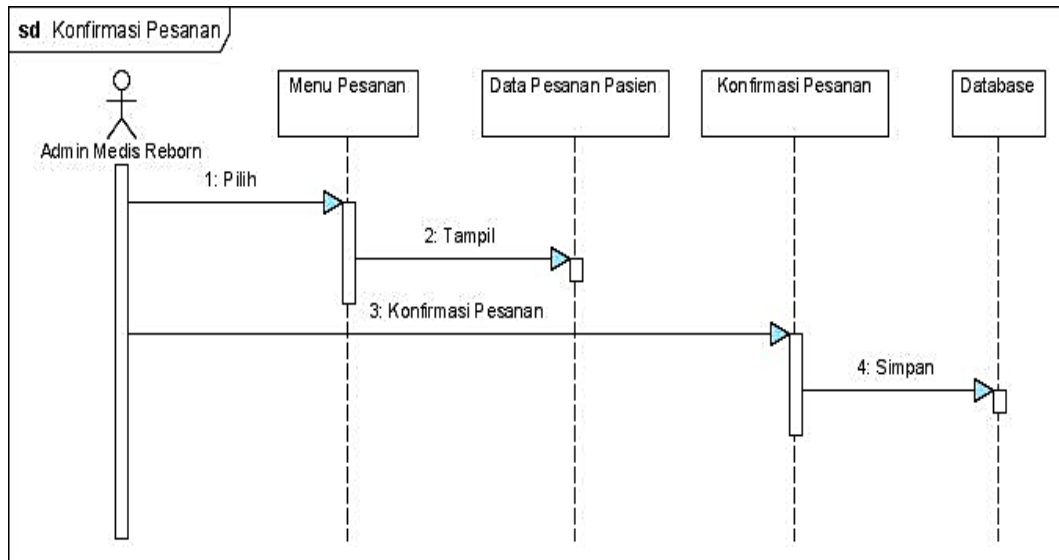
Sequence diagram untuk melihat data pasien oleh admin medis reborn adalah sebagai berikut:



Gambar 4.27 *Sequence Diagram* Data Pasien

4.2.1.24 Sequence Diagram Konfirmasi Pesanan

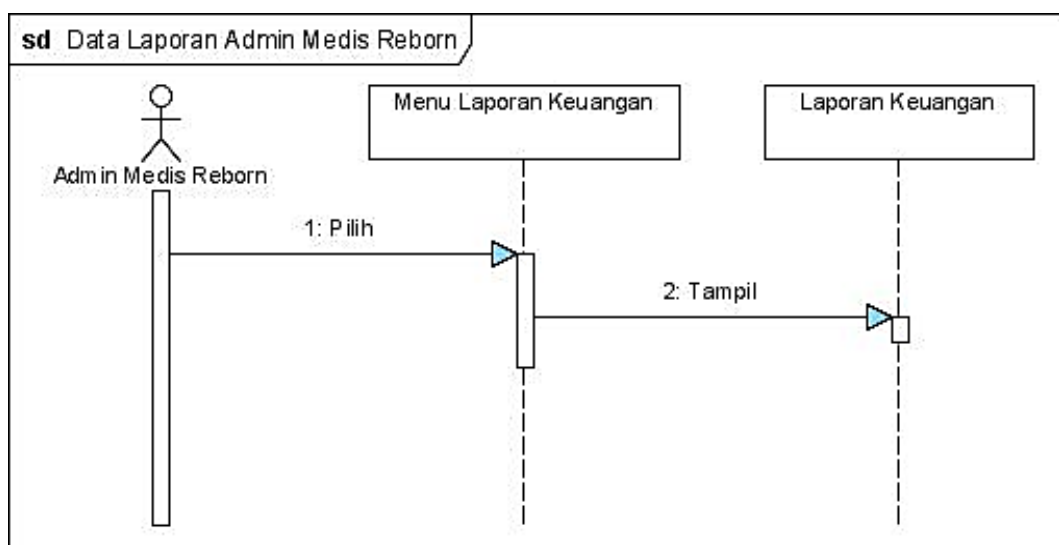
Sequence diagram untuk melakukan konfirmasi pesanan oleh admin medis reborn adalah sebagai berikut:



Gambar 4.28 *Sequence Diagram* Konfirmasi Pesanan

4.2.1.25 Sequence Diagram Laporan Keuangan Medis Reborn

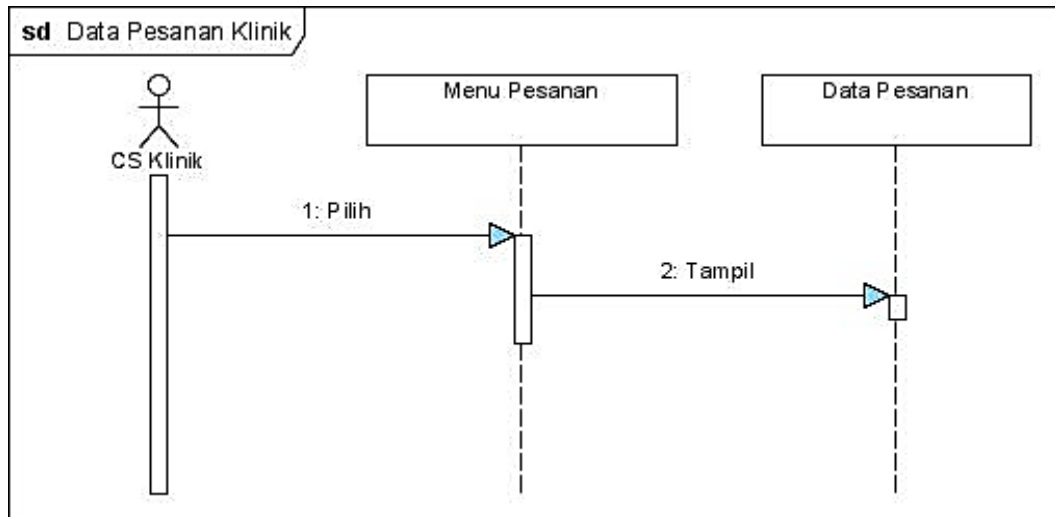
Sequence diagram untuk melihat laporan keuangan oleh admin medis reborn adalah sebagai berikut:



Gambar 4.29 *Sequence Diagram* Laporan Keuangan Medis Reborn

4.2.1.26 Sequence Diagram Pesanan Bagi Klinik

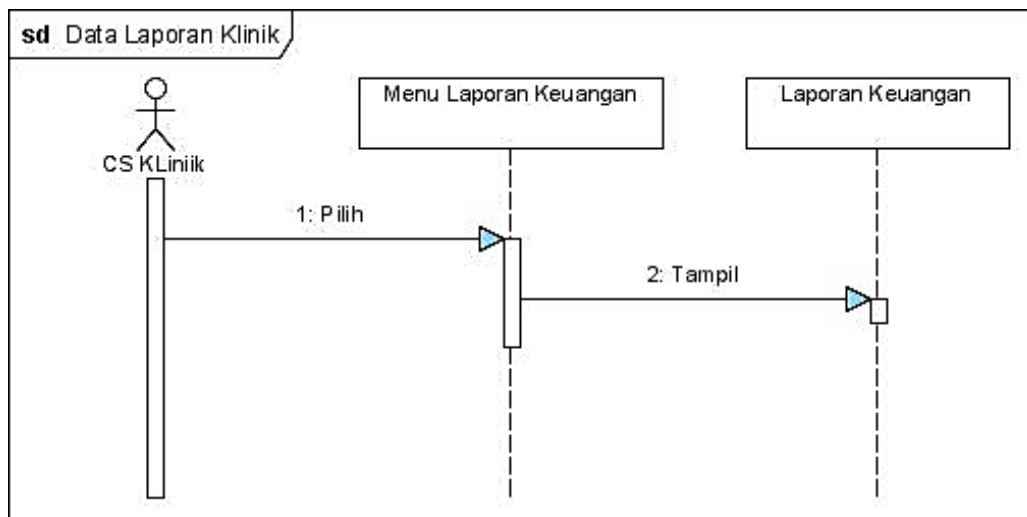
Sequence diagram untuk melihat pesanan bagi klinik oleh CS adalah sebagai berikut:



Gambar 4.30 *Sequence Diagram* Pesanan Bagi Klinik

4.2.1.27 Sequence Diagram Laporan Keuangan Klinik

Sequence diagram untuk melihat laporan keuangan oleh CS Klinik adalah sebagai berikut:



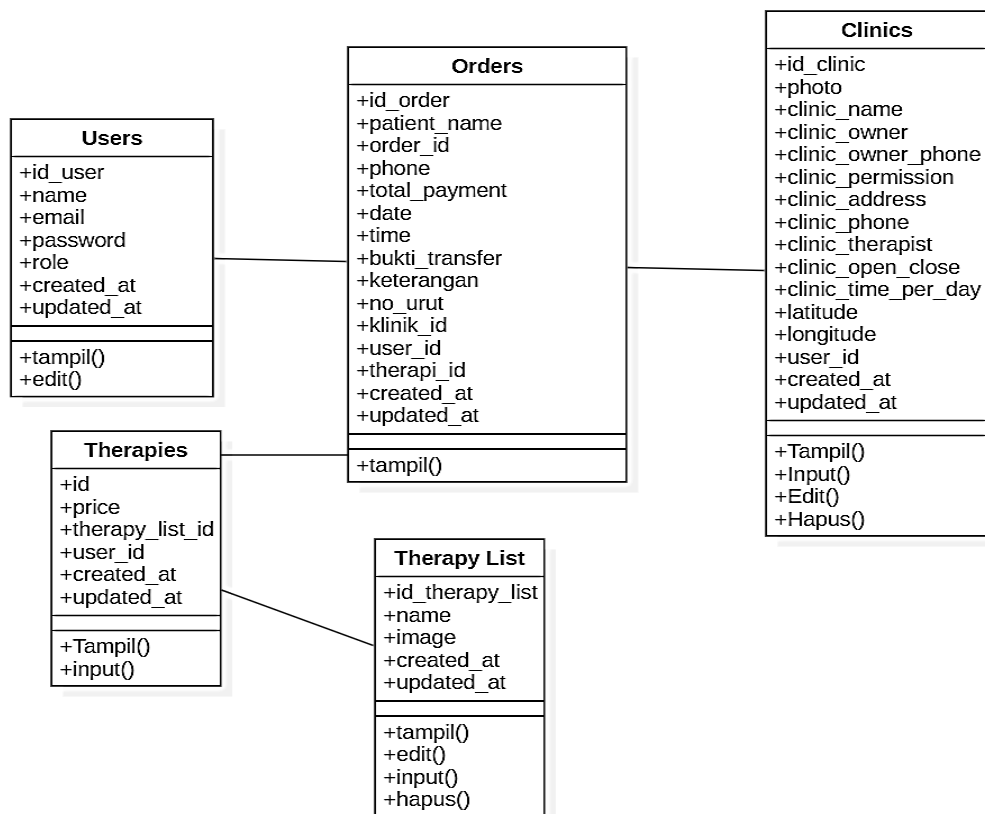
Gambar 4.31 *Sequence Diagram* Keuangan Medis Reborn

4.2.2. Desain Database

Desain *database* berfungsi sebagai gambaran relasi data antar tabel. Tahap ini terdiri dari rancangan *class diagram* dan tabel perancangan *database*.

4.2.2.1. Class Diagram

Class diagram merupakan penggambaran dari hubungan antar kelas, atribut dan objek dari suatu sistem yang akan dibangun. Pada sistem Medis Reborn terdapat 5 kelas, yaitu *users*, *clinics*, *therapies*, *orders*, dan *therapy_list*. Kelas *users* berisi atribut data pengguna sistem. Kelas *clinics* berisi atribut data klinik. Kelas *therapies* berisi jenis terapi yang disediakan oleh setiap klinik. Kelas *orders* berisi data transaksi pemesanan pengobatan. Adapun yang terakhir yaitu kelas *therapy_list* yaitu berisi jenis pengobatan yang tersedia. Berikut *class diagram* pada *startup* Medis Reborn:



Gambar 4.32 Class Diagram

4.2.2.2. Tabel Perancangan Database

Tabel perancangan *database* pada sistem yang akan dibuat terdiri dari 5 tabel yaitu, tabel *users*, tabel *clinics*, tabel *therapies*, tabel *orders*, dan tabel *therapy list*.

1. Tabel *Users*

Tabel *users* digunakan untuk menyimpan data-data pengguna atau *user* yang menggunakan sistem.

Tabel 4.1 *Users*

No	Nama	Jenis	Ukuran
1	id_user	Bigint	20
2	Name	Varchar	50
3	Email	Varchar	30
4	Password	Varchar	20
5	Role	Varchar	10
	created_at	timestamp	
	updated_at	timestamp	

2. Tabel *Clinics*

Tabel *clinics* digunakan untuk menyimpan data klinik.

Tabel 4.2 *Clinics*

	Nama	Jenis	Ukuran
	clinic	int	
	photo	char	

	ic_name	char	
	ic_owner	char	
	ic_owner_phone	char	
	ic_permission	char	
	ic_address	char	
	ic_phone	char	
	ic_therapist	eger	
	ic_open_close	char	
	ic_time_per_day	char	
	itude	char	
	ngitude	char	
	r_id	int	
	ated_at	estamp	
	lated_at	estamp	

3. Tabel *Therapies*

Tabel *therapies* digunakan untuk menyimpan data jenis terapi/pengobatan yang terdapat pada setiap klinik.

Tabel 4.3 *Therapies*

No	Nama	Jenis	Ukuran
1	id_therapies	Bigint	20
2	therapi_name	Varchar	50

3	Price	Ineteger	11
4	user_id	Bigint	20
5	created_at	Timestamp	-
6	updated_at	Timestamp	-

4. Tabel *Orders*

Tabel *orders* digunakan untuk menyimpan data transaksi pemesanan pengobatan.

Tabel 4.4 *Orders*

No	Nama	Jenis	Ukuran
1	id_order	Bigint	20
2	patient_name	Varchar	50
3	order_id	Varchar	20
4	Phone	Varchar	20
5	total_payment	Varchar	20
6	Date	Date	-
7	Time	Varchar	20
8	bukti_transfer	Varchar	100
9	Keterangan	Varchar	255
10	no_urut	Integer	11
11	clinic_id	Bigint	20
12	user_id	Bigint	20

13	therapies_id	Bigint	20
14	created_at	Timestamp	-
15	updated_at	Timestamp	-

5. Tabel *Therapy List*

Tabel *therapy list* digunakan untuk menyimpan data jenis-jenis terapi/pengobatan yang tersedia pada sistem Medis Reborn.

Tabel 4.5 *Therapy List*

	Id	Therapy Name	Duration
	therapy_id	int	
	name	varchar	
	duration	varchar	
	created_at	timestamp	
	updated_at	timestamp	

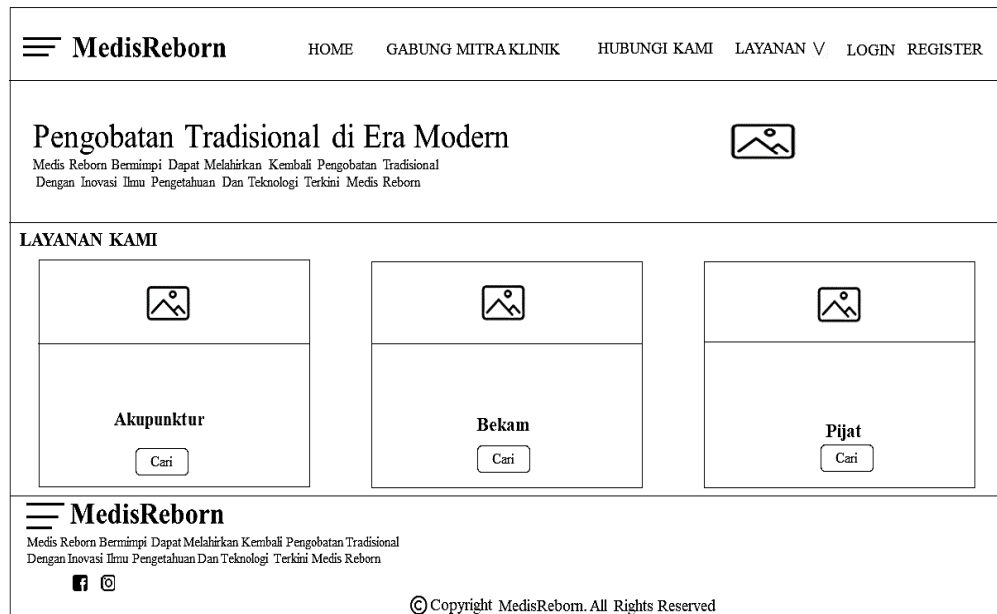
4.2.3. Desain *User Interface*

Desain *user interface* adalah rancangan tampilan visual dari sistem yang akan dibuat. *User Interface* dalam aplikasi pengobatan tradisional pada *startup* Medis Reborn terbagi dalam beberapa *layout* sebagai berikut:

1. *User Interface* Halaman Utama

Pada halaman utama terdapat 6 menu yaitu *home*, gabung mitra klinik, hubungi kami, layanan, *login*, dan register. Menu *home* adalah tampilan awal web. Menu gabung mitra klinik berisi penjelasan mengenai pendaftaran klinik sebagai mitra Medis Reborn. Menu Hubungi Kami berisi cara berkomunikasi dengan Medis Reborn. Menu layanan berisi jenis pengobatan pada Medis Reborn. Dan

yang terakhir menu login pasien/*customer* dan menu registrasi untuk mendaftar.



Gambar 4.33 *User Interface* Halaman Utama

2. *User Interface* Register


Pada halaman register, *customer* mendaftarkan diri dengan mengisi nama, *email*, dan *password*. Berikut rancangan tampilan halaman register *customer*.

The screenshot shows a registration form titled "Register". The form is contained within a white box with a thin border. It has four input fields for "Name", "E-Mail Address", "Password", and "Confirm Password". Each input field is a simple rectangular box with a thin border. Below the input fields is a "Register" button, which is a rounded rectangular button with a thin border. The form is set against a light gray background.

Gambar 4.34 *User Interface* Register

3. User Interface Login

Pada halaman *login*, *user/customer* menginputkan *email* dan *password*. Berikut rancangan tampilan halaman *login*.



The login form is titled "Login" and is contained within a light gray box. It features two input fields: "E-Mail Address" and "Password". Below the password field is a checkbox labeled "Remember me". At the bottom of the form, there is a "Log In" button and a link labeled "Forgot Your Password?".

Gambar 4.35 User Interface Login

4. User Interface Pilih Klinik

Pada halaman pilih klinik, *customer* atau pasien dapat melihat daftar klinik pengobatan tradisional. *Customer* dapat memilih lokasi klinik terdekat dari lokasi mereka.



The "Pilih Klinik" page features a header with the "MedisReborn" logo and navigation links: HOME, GABUNG MITRA KLINIK, HUBUNGI KAMI, LAYANAN, LOGIN, and REGISTER. The main content area is titled "Pengobatan Tradisional di Era Modern" and includes a sub-header: "Medis Reborn Bermimpi Dapat Melahirkan Kembali Pengobatan Tradisional Dengan Inovasi Ilmu Pengetahuan Dan Teknologi Terkini Medis Reborn". Below this, the "LAYANAN KAMI" section displays three service cards: "Akupunktur", "Bekam", and "Pijat". Each card has a search button labeled "Cari". The footer contains the "MedisReborn" logo, a brief description, social media icons, and a copyright notice: "© Copyright MedisReborn. All Rights Reserved".

Gambar 4.36 User Interface Pilih Klinik

5. User Interface Form Jadwal Pengobatan

Pada halaman *form* jadwal pengobatan, *customer* atau pasien menginput data tentang jenis dan jadwal pengobatan.

The screenshot shows the MedisReborn website's appointment form. The header includes the MedisReborn logo, navigation links (HOME, GABUNG MITRA KLINIK, HUBUNGI KAMI, LAYANAN, USER), and a user profile dropdown (Profile, Logout). The main heading is "Selamat Datang di Nama Klinik". Below this, the user's address is displayed as "Alamat Buka". The form fields include: Nama (text input), No Hp/ WA (text input), Jenis Pengobatan (text input), Tentukan Jadwal (text input), Tanggal Pengobatan (date picker with dd/mm/yyyy format), and Waktu Pengobatan (text input). A "Buat Pesanan" button is at the bottom of the form. The footer contains the MedisReborn logo, a mission statement, social media icons, and a copyright notice: "© Copyright MedisReborn. All Rights Reserved".

Gambar 4.37 User Interface Form Jadwal Pengobatan

6. User Interface Invoice Bukti Pembayaran

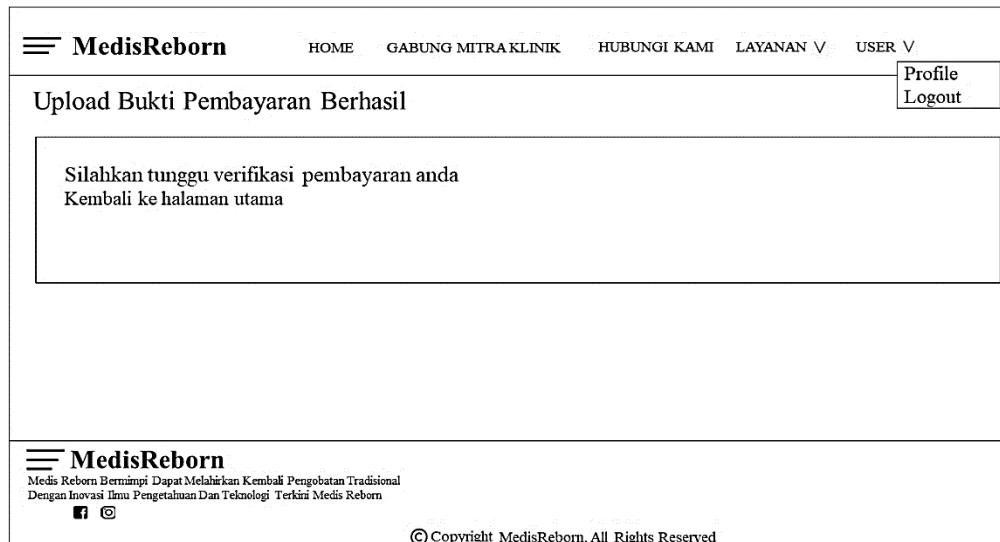
Setelah mengisi *form* jadwal pemesanan, *customer* mendapatkan *invoice* pembayaran. Setelah melakukan pembayaran, *customer* mengupload bukti pembayaran.

The screenshot shows the MedisReborn website's payment proof upload form. The header is identical to the previous form. The main heading is "Id Pesanan". Below this, the order details are listed: Nama, No Hp/Wa, Klinik, Pengobatan, Total Pembayaran, Tanggal Pengobatan, No Urut, Waktu Pengobatan, and a bank transfer instruction: "Silahkan Transfer Ke Rekening Medis Reborn Berikut : 206201007896505 (BRI) atas nama Dinary Dwihatami". The form includes an "Upload Bukti Pembayaran" section with a cloud icon and the text "Drag and drop a file here or click". An "Update Pesanan" button is at the bottom of the form. The footer is identical to the previous form.

Gambar 4.38 User Interface Invoice Bukti Pembayaran

7. User Interface Bukti Pembayaran Berhasil

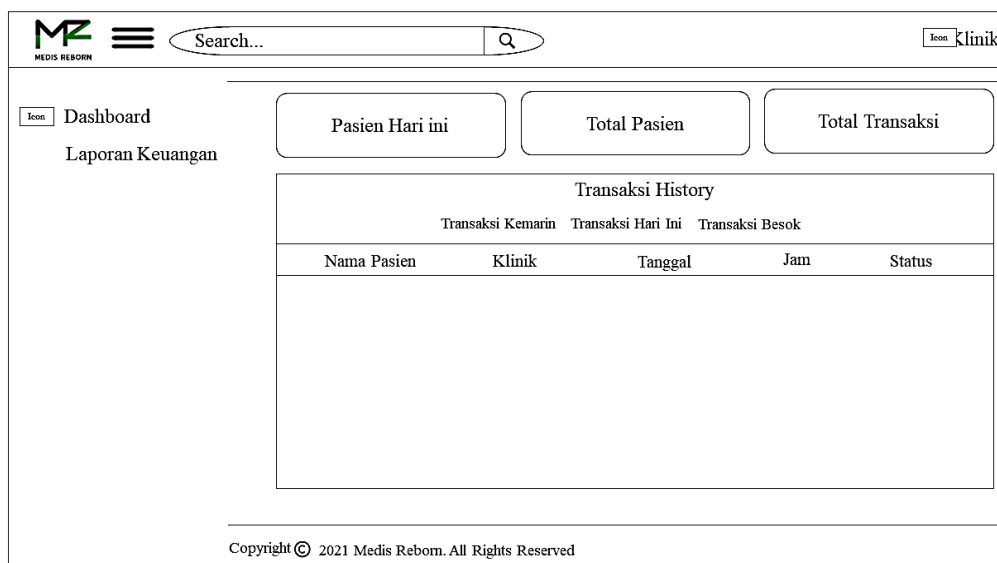
Setelah mengupload bukti pembayaran, *customer* masuk ke halaman informasi bahwa upload bukti pembayaran berhasil.



Gambar 4.39 User Interface Bukti Pembayaran Berhasil

8. User Interface Dashboard Klinik

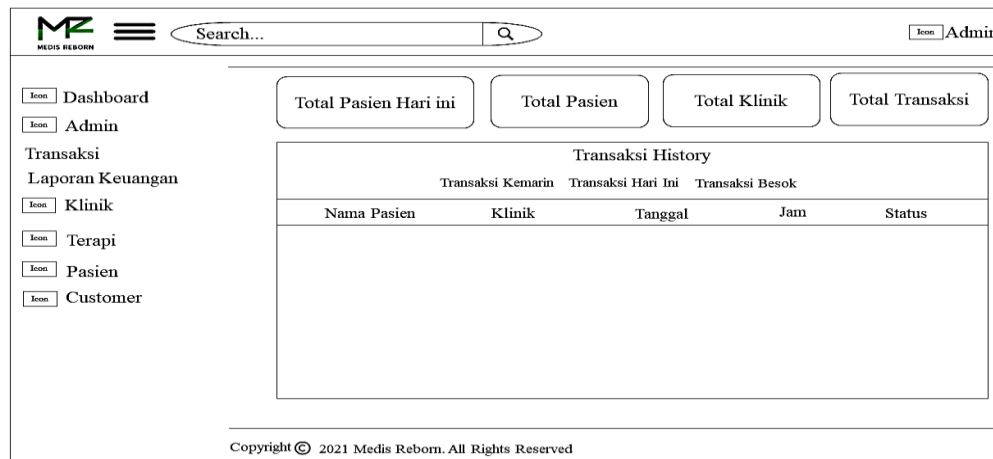
Pada halaman *Dashboard* klinik, admin dapat melihat daftar pesanan yang masuk beserta laporan keuangan.



Gambar 4.40 User Interface Dashboard Klinik

9. User Interface Dashboard Admin Medis Reborn

Pada *Dashboard* admin, terdapat beberapa lima menu yaitu menu transaksi untuk konfirmasi dan melihat pesanan pengobatan, menu laporan keuangan, menu klinik untuk data klinik, menu terapi untuk data terapi, dan menu pasien.



Gambar 4.41 User Interface Dashboard Admin Medis Reborn

4.3. Implementation

Untuk mengimplementasikan aplikasi dilakukan pengkodean program dengan hasil, sebagai berikut:

1. Halaman Utama

Pada halaman utama terdapat menu home, menu gabung mitra klinik, menu hubungi kami, menu layanan kami, menu *login*, dan menu registrasi. Pada halaman ini *user* tidak harus login untuk dapat melihat daftar pengobatan atau layanan maupun daftar klinik. Menu halaman utama dilengkapi dengan semboyan Medis Reborn untuk melengkapi keindahan *website* Medis Reborn. Pada menu gabung mitra klinik, klinik dapat mendaftarkan diri sebagai mitra Medis Reborn dengan mengisi google form yang tersedia. Sedangkan menu hubungi kami adalah layanan komunikasi bagi *user* terhadap Medis Reborn dengan menggunakan *email*.



Gambar 4.42 Implementasi Halaman Utama Medis Reborn

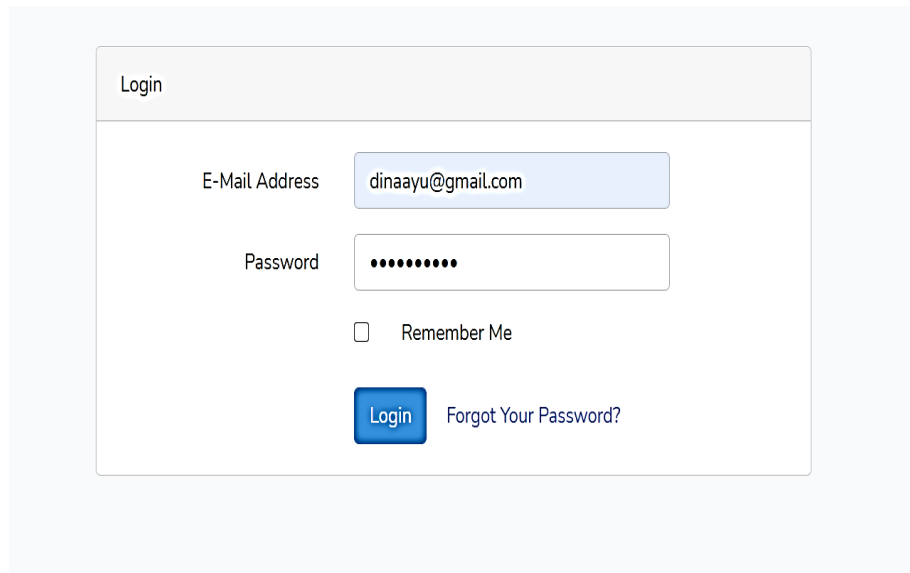
2. Halaman Register

Pada halaman register, pasien atau *user* wajib mengisi data *name*, *email*, dan *password*. Jika register berhasil, *user* akan masuk ke halaman utama.

Gambar 4.43 Implementasi Halaman Register

3. Halaman *Login*

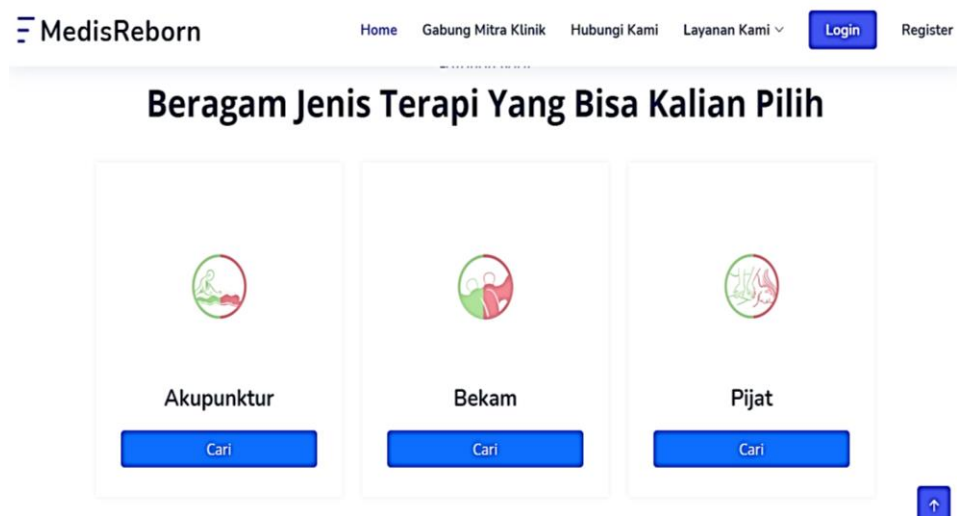
Untuk masuk ke dalam sistem, pasien, CS Klinik, dan Admin harus *login* terlebih dahulu dengan memasukkan *email* dan *password*.



Gambar 4.44 Implementasi Halaman *Login*

4. Halaman Pilih Pengobatan

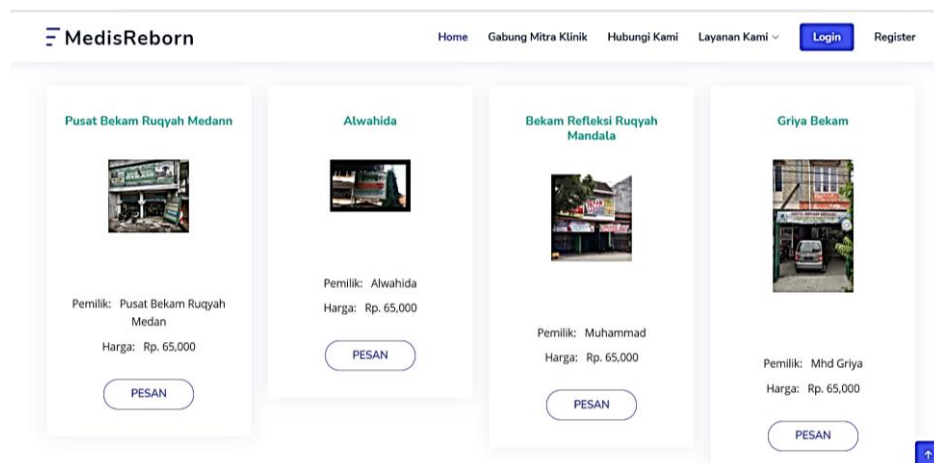
Sebelum memilih klinik yang ingin dikunjungi, pasien terlebih dahulu memilih jenis pengobatan yang akan dilakukan.



Gambar 4.45 Implementasi Pilih Pengobatan

5. Halaman Pilih Klinik

Setelah memilih jenis pengobatan yang diinginkan, maka akan muncul daftar klinik yang menyediakan pengobatan yang dipilih tersebut. Lalu pasien dapat memilih klinik terdekat dari lokasi mereka atau memilih klinik yang mereka inginkan.



Gambar 4.46 Implementasi Pilih Klinik

6. Halaman Form Jadwal Pengobatan

Pada halaman *form* jadwal pengobatan, pasien mengisi data pasien yang ingin berobat serta menentukan tanggal dan waktu pengobatan.

Selamat Datang di Klinik Alwahida
 Alamat: Jalan Mongonsidi Gang D No. 15/16, Polonia, Medan Polonia
 buka: 09.00-21.00

Nama	dinaayu
No Hp/WA	082243564532
Jenis Pengobatan	bekam Rp. 65.000,00
Tentukan Jadwal	
Tanggal Pengobatan	31/08/2021
Waktu Pengobatan	12:00-13:00

Buat Pesanan

Gambar 4.47 Implementasi Form Jadwal Pengobatan

7. Halaman *Invoice* Pembayaran

Halaman *invoice* pembayaran akan muncul setelah pasien melakukan pemesanan. Setelah membayar pemesanan, pasien harus mengunggah foto bukti pembayaran agar pesanan dapat terkonfirmasi.

MedisReborn Home Gabung Mitra Klinik Hubungi Kami Layanan Kami dinaayu

ID Pesanan	TRF202108311630290076
Nama	dinaayu
No Hp/Wa.	082243564532
Klinik	Alwahida
Pengobatan	bekam
Total Pembayaran	Rp. 65,000
Tanggal Pengobatan	2021-08-31
No Urut	2
Waktu Pengobatan	12:00-13:00
Silahkan Transfer Ke Rekening Medis Reborn Berikut : BRI: 382381703 an fslsf	

Upload Foto Bukti Pembayaran

Your so fresh input file

[Update Pesanan](#)

Gambar 4.48 *Implementasi Invoice* Pembayaran

8. Halaman Pembayaran Berhasil

Halaman pembayaran berhasil berisi informasi bahwa unggah foto bukti pembayaran telah berhasil.

MedisReborn Home Gabung Mitra Klinik Hubungi Kami Layanan Kami dinaayu

Upload Bukti Pembayaran Anda Berhasil

Silahkan tunggu verifikasi pemesanan

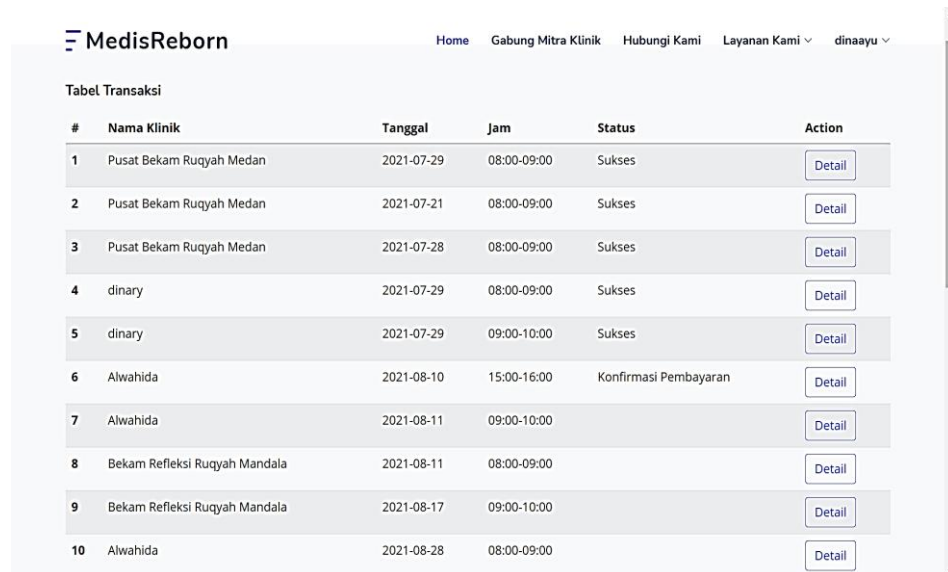
[kembali ke halaman utama](#)

MedisReborn
 Medis Reborn Bermimpi Dapat Melahirkan Kembali Pengobatan
 Tradisional Dengan Inovasi Ilmu Pengetahuan Dan Teknologi Terkini
 Medis Reborn

Gambar 4.49 *Implementasi Pembayaran Berhasil*

9. Halaman *Profile User*

Halaman *Profile User* berisi daftar riwayat pesanan *user*. *User* dapat mengubah data diri pada halaman ini.



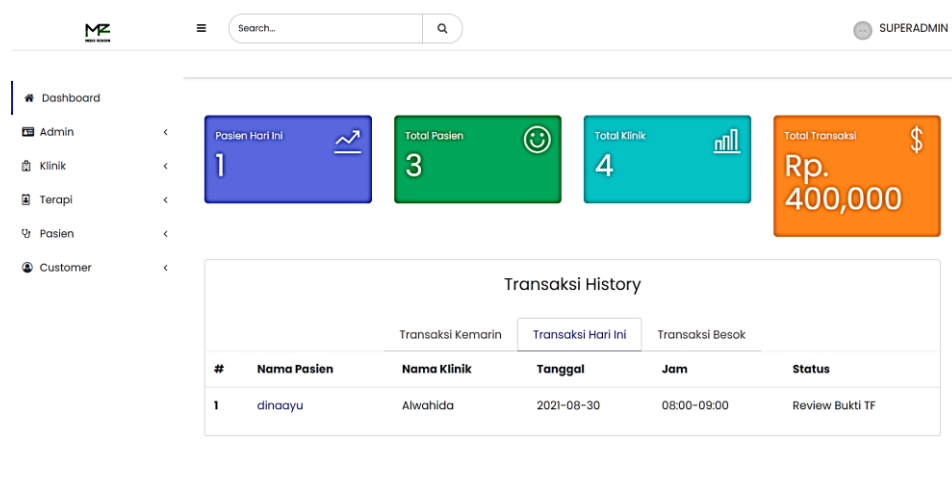
The screenshot shows the MedisReborn user profile page. At the top, there is a navigation bar with the MedisReborn logo and links for Home, Gabung Mitra Klinik, Hubungi Kami, Layanan Kami, and a dropdown menu for 'dinaayu'. Below the navigation bar, there is a section titled 'Tabel Transaksi' (Transaction Table). This section contains a table with 10 rows of transaction data. Each row includes a transaction number, the clinic name, the date, the time, the status, and an action button labeled 'Detail'.

#	Nama Klinik	Tanggal	Jam	Status	Action
1	Pusat Bekam Ruqyah Medan	2021-07-29	08:00-09:00	Sukses	Detail
2	Pusat Bekam Ruqyah Medan	2021-07-21	08:00-09:00	Sukses	Detail
3	Pusat Bekam Ruqyah Medan	2021-07-28	08:00-09:00	Sukses	Detail
4	dinary	2021-07-29	08:00-09:00	Sukses	Detail
5	dinary	2021-07-29	09:00-10:00	Sukses	Detail
6	Alwahida	2021-08-10	15:00-16:00	Konfirmasi Pembayaran	Detail
7	Alwahida	2021-08-11	09:00-10:00		Detail
8	Bekam Refleksi Ruqyah Mandala	2021-08-11	08:00-09:00		Detail
9	Bekam Refleksi Ruqyah Mandala	2021-08-17	09:00-10:00		Detail
10	Alwahida	2021-08-28	08:00-09:00		Detail

Gambar 4.50 Implementasi *Profile User*

10. Halaman *Dashboard Admin Medis Reborn*

Dashboard admin medis berisi tampilan total pasien, total transaksi, total klinik, dan total pasien hari ini.



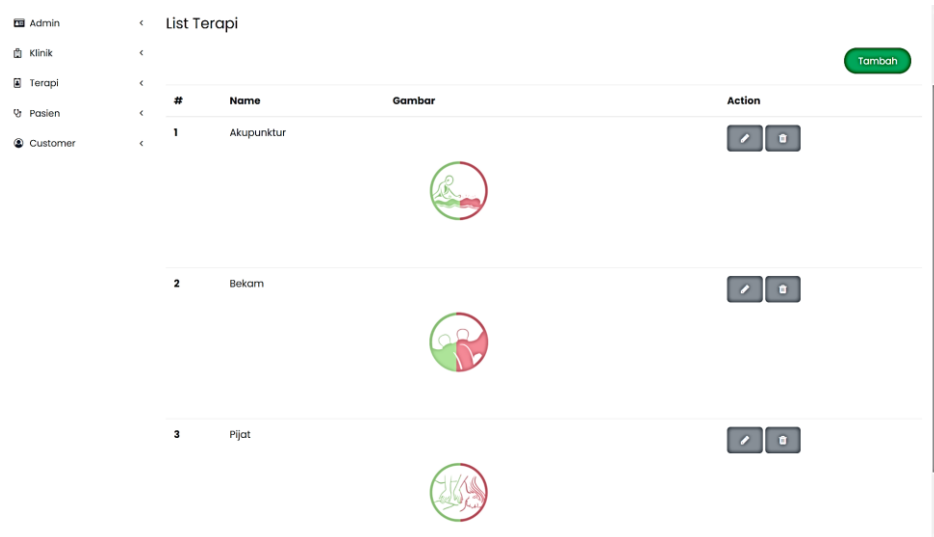
The screenshot shows the MedisReborn admin dashboard. At the top, there is a navigation bar with the MedisReborn logo, a search bar, and a user profile icon labeled 'SUPERADMIN'. Below the navigation bar, there is a sidebar with a menu containing 'Dashboard', 'Admin', 'Klinik', 'Terapi', 'Pasien', and 'Customer'. The main content area displays four summary cards: 'Pasien Hari Ini' (1), 'Total Pasien' (3), 'Total Klinik' (4), and 'Total Transaksi' (Rp. 400,000). Below these cards, there is a section titled 'Transaksi History' with a table showing the latest transaction.










Transaksi History					
Transaksi Kemarin					
Transaksi Hari Ini					
Transaksi Besok					
#	Nama Pasien	Nama Klinik	Tanggal	Jam	Status
1	dinaayu	Alwahida	2021-08-30	08:00-09:00	Review Bukti TF

Gambar 4.51 Implementasi *Dashboard Admin Medis Reborn*

11. Halaman *List Pengobatan*

Halaman *list* pengobatan berisi daftar pengobatan. Pada halaman ini, admin dapat mengedit dan menghapus data serta *button* tambah untuk menambah data pengobatan.

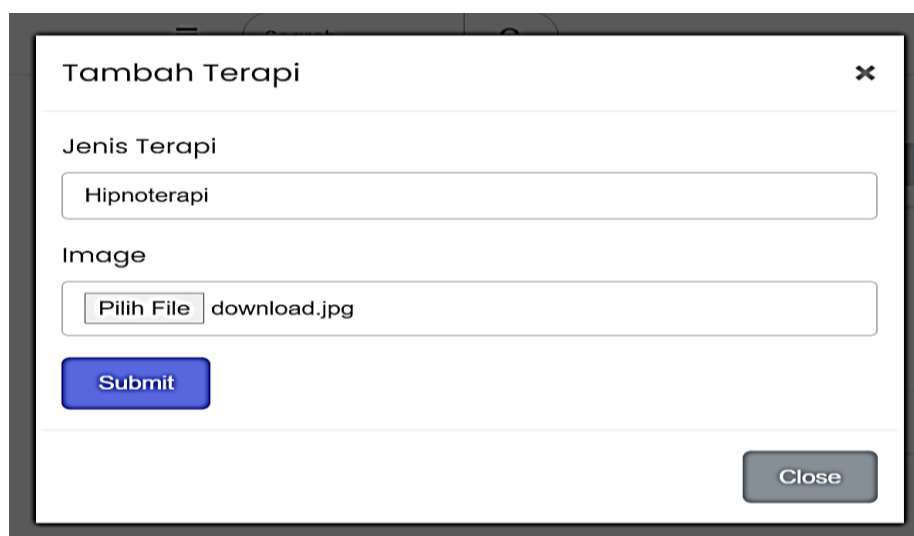


Admin	<	List Terapi		Tambah
Klinik	<			
Terapi	<			
Pasien	<			
Customer	<			
#	Name	Gambar	Action	
1	Akupunktur		 	
2	Bekam		 	
3	Pijat		 	

Gambar 4.52 Implementasi *List Pengobatan*

12. Halaman Tambah Pengobatan

Pada halaman tambah pengobatan, admin dapat menambah pengobatan dengan mengisi nama pengobatan dan gambar.



Tambah Terapi

Jenis Terapi

Hipnoterapi

Image

Pilih File download.jpg

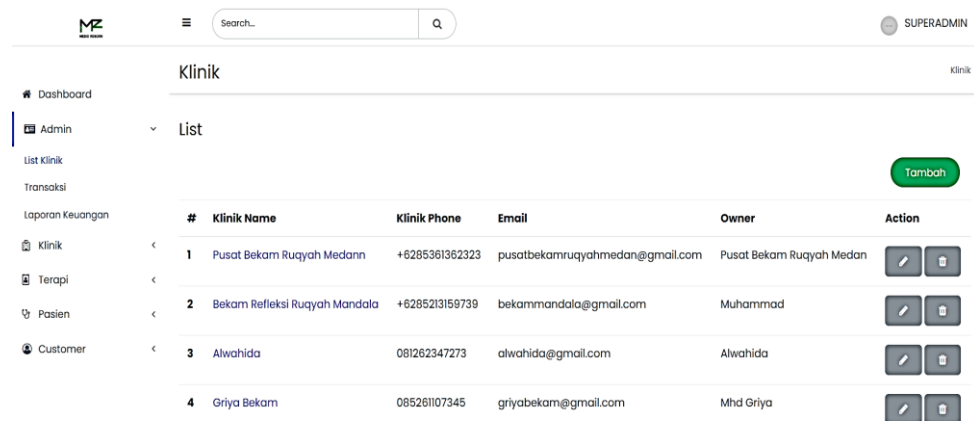
Submit









Close

Gambar 4.53 Implementasi Tambah Pengobatan

13. Halaman *List Klinik*

Halaman *list* klinik berisi daftar klinik. Pada halaman ini, admin dapat mengedit dan menghapus data serta *button* tambah untuk menambah data klinik.

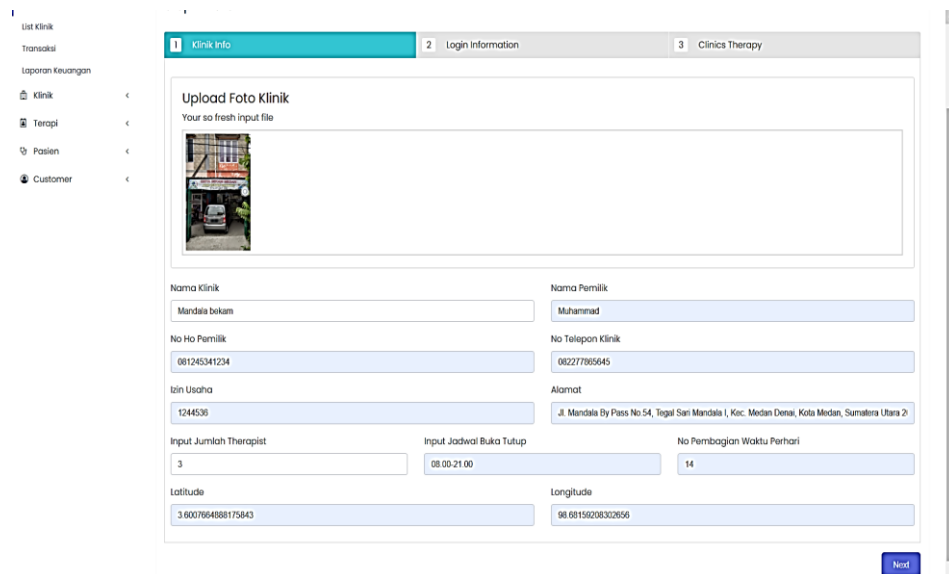


#	Klinik Name	Klinik Phone	Email	Owner	Action
1	Pusat Bekam Ruqyah Medan	+6285361362323	pusatbekamruqyahmedan@gmail.com	Pusat Bekam Ruqyah Medan	 
2	Bekam Refleksi Ruqyah Mandala	+6285213159739	bekammandala@gmail.com	Muhammad	 
3	Alwahida	081262347273	alwahida@gmail.com	Alwahida	 
4	Griya Bekam	085261107345	griyabekam@gmail.com	Mhd Griya	 

Gambar 4.54 *Implementasi List Klinik*


14. Halaman Tambah Klinik

Pada halaman tambah klinik, admin dapat menambahkan data klinik melalui pendaftaran yang diajukan klinik.



1 Klinik Info | 2 Login Information | 3 Clinics Therapy

Upload Foto Klinik
Your so fresh input file



Nama Klinik: Nama Pemilik:

No Hp Pemilik: No Telepon Klinik:

Isin Usaha: Alamat:

Input Jumlah Therapist: Input Jadwal Buka Tutup: No Pembagian Waktu Perhari:

Latitude: Longitude:

[Next](#)

Gambar 4.55 *Implementasi Tambah Klinik*

15. Halaman Data Pasien

Halaman Data Pasien berisi Daftar Pasien pada Medis Reborn.

#	Name	Email
1	dinaayu	dinaayu@gmail.com
2	dwinenda	user@gmail.com
3	Bunga	bunga@gmail.com

Gambar 4.56 Implementasi Data Pasien

16. Halaman List Pesanan

Halaman *list* pesanan berisi daftar pesanan pasien. Pada halaman ini, admin dapat memilih nama pasien untuk selanjutnya melakukan konfirmasi pemesanan.

#	Nama Pasien	Nama Klinik	Tanggal	Jam	Status
1	dinaayu	Alwahida	2021-08-31	12:00-13:00	
2	dinaayu	Alwahida	2021-08-31	12:00-13:00	
3	dinaayu	Alwahida	2021-08-30	08:00-09:00	Sukses
4	dinaayu	Alwahida	2021-08-28	08:00-09:00	
5	dinaayu	Bekam Refleksi Ruqyah Mandala	2021-08-17	09:00-10:00	
6	dinaayu	Bekam Refleksi Ruqyah Mandala	2021-08-11	08:00-09:00	
7	dinaayu	Alwahida	2021-08-11	09:00-10:00	
8	dinaayu	Alwahida	2021-08-10	15:00-16:00	Konfirmasi Pembayaran
9	Bunga	Alwahida	2021-08-09	16:00-17:00	Konfirmasi Pembayaran
10	Bayu Agung	Klinik Sejahtera	2021-08-02	08:00-09:00	Pembayaran Gagal
11	jdkkfnvlasva	Pusat Bekam Ruqyah Medan	2021-07-21	08:00-09:00	Sukses
12	putri	dinary	2021-07-30	09:00-10:00	
13	putri	dinary	2021-07-30	08:00-09:00	
14	dinaayubikaterapi	dinary	2021-07-30	08:00-09:00	Sukses

Gambar 4.57 Implementasi Halaman List Pesanan

17. Halaman Data Konfirmasi Pesanan

Pada halaman data konfirmasi pesanan admin melihat bukti pembayaran kemudian melakukan konfirmasi terhadap pesanan.

The screenshot shows the 'Transaksi' page in the SUPERADMIN interface. It features a sidebar with navigation options: Dashboard, Admin, Klinik, Terapi, Pasien, and Customer. The main content area displays a transaction confirmation form. At the top, there is a QR code with the text '002 - BRI 174501000614533 WAHYU LARASWATI Rp 1,000,000.00'. Below the QR code, there is a table with transaction details:

Nama Pasien	:	dinaayu
Nama Klinik	:	Alwahida
Jenis Therapy	:	Bekam
Tanggal	:	2021-08-01
Jam	:	15:00-16:00
No Urut	:	1

At the bottom of the form, there is a 'Konfirmasi Pembayaran' dropdown menu, a 'Catatan Transaksi' text area, and a 'Konfirmasi' button.

Gambar 4.58 Implementasi Halaman Data Konfirmasi Pesanan

18. Halaman Laporan Keuangan Medis Reborn

Pada halaman laporan admin dapat melihat data total transaksi pesanan dan total keuntungan yang didapat oleh Medis Reborn.

The screenshot shows the 'Semua Transaksi' page in the SUPERADMIN interface. It features a sidebar with navigation options: Admin, Klinik, Terapi, Pasien, and Customer. The main content area displays a table of transactions with the following columns: #, Nama Pasien, Nama Klinik, Tanggal, Jam, and Harga.

#	Nama Pasien	Nama Klinik	Tanggal	Jam	Harga
1	dinarydewhatami	Pusat Bekam Ruyah Medan	2021-07-29	08:00-09:00	Rp. 65,000
2	jdkthvlasva	Pusat Bekam Ruyah Medan	2021-07-21	08:00-09:00	Rp. 65,000
3	jdkthvlasvab	Pusat Bekam Ruyah Medan	2021-07-28	08:00-09:00	Rp. 65,000
4	jdkthvlasva	dinary	2021-07-29	08:00-09:00	Rp. 70,000
5	dinarydewhatami	dinary	2021-07-29	09:00-10:00	Rp. 70,000
6	putri	dinary	2021-07-30	08:00-09:00	Rp. 70,000
7	putri	dinary	2021-07-30	09:00-10:00	Rp. 70,000
8	Bayu Agung	Klinik Sejahtera	2021-08-02	08:00-09:00	Rp. 200,000
9	Bunga	Alwahida	2021-08-09	16:00-17:00	Rp. 65,000
10	dinaayu	Alwahida	2021-08-10	15:00-16:00	Rp. 65,000
11	dinaayu	Alwahida	2021-08-11	09:00-10:00	Rp. 65,000
12	dinaayu	Bekam Refleksi Ruyah Mandala	2021-08-11	08:00-09:00	Rp. 65,000
13	dinaayu	Bekam Refleksi Ruyah Mandala	2021-08-17	09:00-10:00	Rp. 65,000
14	dinaayu	Alwahida	2021-08-28	08:00-09:00	Rp. 65,000
15	dinaayu	Alwahida	2021-08-30	08:00-09:00	Rp. 65,000
16	dinaayu	Alwahida	2021-08-31	12:00-13:00	Rp. 65,000
17	dinaayu	Alwahida	2021-08-31	12:00-13:00	Rp. 65,000

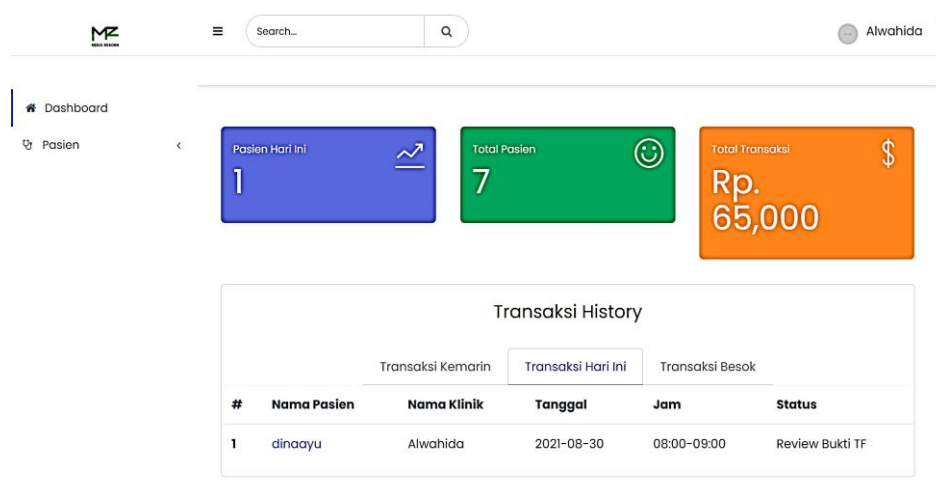
At the bottom of the table, there is a summary table with the following columns: #, Total Transaksi, Potongan, and Pendapatan.

#	Total Transaksi	Potongan	Pendapatan
Total Transaksi	:	Rp. 1,260,000	-7%
			Rp. 88,200

Gambar 4.59 Implementasi Halaman Laporan Keuangan Medis Reborn

19. Halaman *Dashboard* Klinik

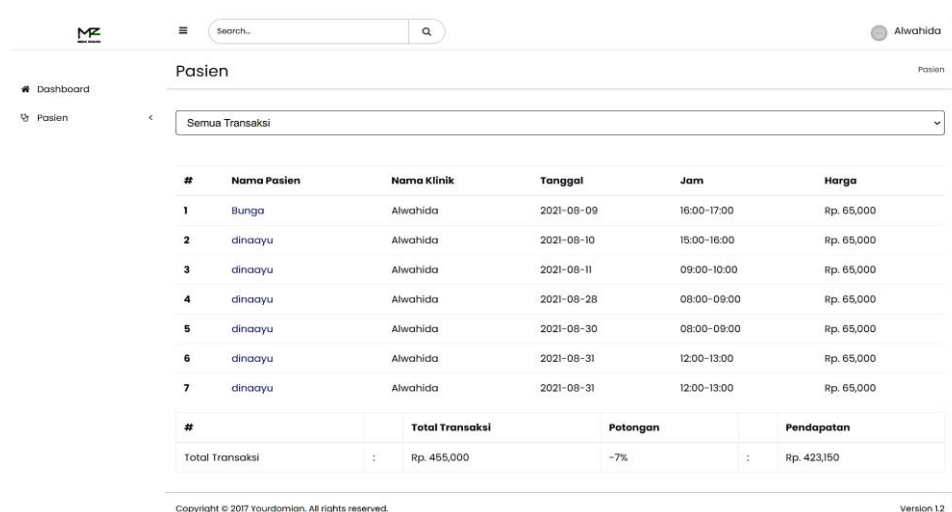
Dashboard klinik berisi daftar pesanan untuk klinik. Pada halaman ini, klinik dapat melihat total pasien, total pasien hari ini dan total transaksi.



Gambar 4.60 Implementasi Halaman *Dashboard* Klinik

20. Halaman Laporan Keuangan Klinik

Pada halaman laporan admin dapat melihat data total transaksi pesanan dan total keuntungan yang didapat oleh klinik.



Gambar 4.61 Implementasi Halaman Laporan Keuangan Klinik

4.4. Verification

Pengujian sistem dilakukan untuk mengetahui apakah perangkat lunak sudah sesuai dengan desain, dan fungsionalitas dari aplikasi apakah berjalan dengan baik atau tidak. Berikut pelaksanaan dan hasil dari pengujian sistem yang telah divalidasi oleh validator, yaitu:

Nama Sistem : Aplikasi Pengobatan Tradisional pada *Startup* Medis Reborn

Tanggal Pengujian : 6 September 2021

Nama Validator : Susanto, S. Kom

Pekerjaan : Kepala Bidang IT Dinas Pariwisata Kota Medan

Adapun hasil pengujian disajikan dalam tabel berikut ini:

Tabel 4.6 Pengujian Registrasi

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	<i>Name</i> atau <i>email</i> atau <i>password</i> dan <i>confirm password</i> tidak diisi	Sistem akan menampilkan pesan isi bidang ini	Berhasil
2.	<i>password</i> dan <i>confirm password</i> berbeda dan kurang dari 8 karakter	Sistem akan menampilkan pesan “ The password must be at least 8 characters ”	Berhasil
3.	Mengisi <i>Name</i> , <i>email</i> , <i>password</i> dan <i>confirm password</i> sesuai kemudian klik register	Sistem menampilkan halaman utama <i>user</i>	Berhasil

Tabel 4.7 Tabel Pengujian *Login*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	<i>Email dan password</i> tidak diisi	Sistem akan menampilkan pesan isi bidang ini	Berhasil
2.	<i>Password</i> yang diinputkan tidak sesuai	Sistem akan menampilkan pesan “ <i>These credentials do not match our records.</i> ”	Berhasil
3.	Mengisi <i>email</i> dan <i>password</i> yang sesuai kemudian klik <i>login</i>	Sistem menampilkan halaman utama <i>user</i>	Berhasil
4.	Akses <i>form</i> pemesanan harus <i>login</i> terlebih dahulu	Sistem menampilkan halaman <i>form login</i>	Berhasil

Tabel 4.8 Pengujian Pemesanan

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Tekan tombol cari untuk memilih jenis pengobatan	Sistem menampilkan daftar klinik terdekat	Berhasil
2.	lokasi klinik terdekat	Sistem akan menampilkan urutan lokasi terdekat dari lokasi <i>user</i>	Berhasil
3.	Tekan tombol pesan untuk memilih klinik	Sistem akan menampilkan <i>form</i> jadwal pemesanan	Berhasil
4.	Tidak dapat memesan waktu dan tanggal yang sama jika jumlah therapist sudah mencukupi kuota pemesanan	Sistem akan menampilkan “Pilih Jam Lain atau Jam yang Sama Pada Tanggal Yang Lain”	Berhasil

5.	Nomor urut pasien pada klinik	Sistem menampilkan nomor urut pasien saat pengobatan	Berhasil
6.	Mengisi form jadwal pengobatan	Sistem menampilkan halaman <i>invoice</i> pembayaran	Berhasil
7.	Form Jadwal pengobatan tidak diisi	Sistem akan menampilkan isi bidang ini	Berhasil
8.	<i>Upload</i> bukti pembayaran	Sistem akan menampilkan informasi <i>upload</i> bukti pembayaran anda berhasil	Berhasil

Tabel 4.9 Pengujian *Profile User*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Klik Menu <i>Profile User</i>	Sistem akan menampilkan tabel transaksi yang dilakukan <i>user</i>	Berhasil
2.	Klik <i>button</i> detail	Sistem akan menampilkan <i>invoice</i> pembayaran	Berhasil
3.	Klik nama <i>user</i>	Sistem menampilkan <i>form</i> edit data <i>user</i>	Berhasil
4.	Edit nama <i>user</i>	Nama <i>user</i> berhasil diubah	Berhasil

Tabel 4.10 Tabel Pengujian Klinik

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Klik Dashboard	Sistem akan menampilkan total pasien hari ini, total pasien,	Berhasil

		dan total transaksi	
2.	Klik Menu Laporan Keuangan	Sistem akan menampilkan daftar transaksi klinik, total transaksi, total potongan, dan pendapatan.	Berhasil

Tabel 4.11 Pengujian Admin Medis Reborn

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Klik Dashboard	Sistem akan menampilkan total pasien hari ini, total pasien, total klinik, dan total transaksi	Berhasil
2.	Klik Menu Pasien	Sistem akan menampilkan data pasien	Berhasil
3.	Klik Menu Transaksi	Sistem akan menampilkan daftar transaksi	Berhasil
4.	Klik Nama Pasien	Sistem akan menampilkan data pesanan	Berhasil
5.	Konfirmasi Pesanan	Sistem akan menampilkan data pesanan sukses	Berhasil
6.	Klik Menu Laporan Keuangan	Sistem akan menampilkan daftar transaksi klinik, total transaksi, total potongan, dan pendapatan.	Berhasil

Tabel 4.12 Tabel Pengujian Kelola Terapi Oleh Admin Medis Reborn

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Klik menu terapi	Sistem akan menampilkan daftar pengobatan	Berhasil
2.	Klik button tambah pada menu terapi	Sistem akan menampilkan <i>form</i> terapi	Berhasil
3.	Mengisi <i>form</i> terapi	Sistem akan menampilkan daftar terapi terbaru	Berhasil
4.	Pilih <i>icon</i> edit untuk Mengubah data terapi	Sistem akan menampilkan <i>form</i> terapi data yang ingin diedit	Berhasil
5.	Mengisi <i>form</i> edit terapi	Sistem akan menampilkan terapi yang sudah diedit	Berhasil
6.	Pilih <i>icon</i> delete untuk menghapus terapi	Sistem akan menampilkan apakah anda ingin menghapus data ini	Berhasil
7.	Pilih button <i>delete</i> pada <i>event</i> hapus data	Sistem akan menampilkan daftar terapi terbaru setelah data dihapus	Berhasil
8.	Pilih button cancel pada <i>event</i> hapus data	Sistem akan menampilkan kembali daftar terapi	Berhasil

Tabel 4.13 Tabel Pengujian Kelola Klinik Oleh Admin Medis Reborn

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1.	Klik menu klinik	Sistem akan menampilkan daftar klinik	Berhasil

2.	Klik button tambah pada menu klinik	Sistem akan menampilkan <i>form</i> klinik	Berhasil
3.	Mengisi <i>form</i> klinik	Sistem akan menampilkan daftar klinik terbaru	Berhasil
4.	Pilih <i>icon</i> edit untuk Mengubah data klinik	Sistem akan menampilkan <i>form</i> klinik data yang ingin diedit	Berhasil
5.	Mengisi <i>form</i> edit klinik	Sistem akan menampilkan klinik yang sudah diedit	Berhasil
6.	Pilih <i>icon</i> delete untuk menghapus klinik	Sistem akan menampilkan apakah anda ingin menghapus data ini	Berhasil
7.	Pilih button <i>delete</i> pada <i>event</i> hapus data klinik	Sistem akan menampilkan daftar klinik terbaru setelah data dihapus	Berhasil
8.	Pilih button cancel pada <i>event</i> hapus data	Sistem akan menampilkan kembali daftar klinik	Berhasil

4.5. *Maintenance*

Pada tahap *maintenance* aplikasi yang sudah dijalankan akan dilakukan pemeliharaan oleh penulis seperti melakukan perbaikan maupun perubahan sistem sesuai dengan perkembangannya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan penulis, maka dapat diperoleh kesimpulan, yaitu:

1. Aplikasi pengobatan tradisional berbasis *web* membantu *customer* atau pasien dalam mencari pengobatan tradisional terdekat dengan sistem pemesanan jadwal pengobatan secara online.
2. Dengan adanya aplikasi pengobatan tradisional membantu klinik dalam memperkenalkan klinik pengobatan tradisional mereka kepada masyarakat serta memudahkan sistem informasi klinik dalam pencatatan data pasien dan melihat laporan keuangan.
3. *Business Model Canvas* (BMC) dapat membantu *startup* Medis Reborn serta Pusat Bekam Ruqyah Cabang Medan dalam mendefinisikan model bisnis agar bisnis dapat berkembang.

5.2 Saran

Saran-saran dalam penelitian ini adalah sebagai berikut:

1. Laporan Keuangan hanya dapat dilihat dalam jangka waktu satu bulan, sehingga perlu bagi pengguna untuk melakukan *backup* data laporan setiap bulannya.
2. Untuk menambahkan titik *latitude longitude* lokasi klinik agar menggunakan *google maps*.
3. Aplikasi dijalankan menggunakan *browser* (*Google*, *Mozilla*, dan *Microsoft Edge*).

DAFTAR PUSTAKA

- Abdurahman, H., & Riswaya, A. R. (2014) "*Aplikasi Pinjaman Pembayaran Secara Kredit Pada Bank Yudha Bhakti*". Jurnal Computech dan Bisnis 3(2), 61–69.
- Ahmad Amzul dan Imam. (2018). "Strategi Pengembangan Bisnis Cargo PT. Garuda Indonesia, Tbk Dengan Pendekatan Business Model Canvas. *Jurnal Aplikasi Bisnis Dan Manajemen (JABM)*, 4(3), 474–485.
- Amin, A. M., Baga, L. M., & Tinaprilla, N. (2018). Strategi Perencanaan Model Bisnis Perusahaan Jasa Konsultan Arsitektur dan Jasa Kontraktor PT Architectaria Media Cipta. *MANAJEMEN IKM: Jurnal Manajemen Pengembangan Industri Kecil Menengah*, 13(1), 55-65
- Aminudin. (2015). *Cara Efektif Belajar Framework Laravel*. Yogyakarta: Lokomedia.
- Andi, D. &. (2018) "*Rancang Bangun Aplikasi Helpdesk (A-Desk) Berbasis Web Menggunakan Framework Laravel (Studi Kasus Di Pdam Surya Sembada Kota Surabaya)*". Jurnal Manajemen Informatika 8(2), 75-81
- Badriyah Hidayati, H., Hasan Machfoed, M., Santoso, B., & Utomo, B. (2019) "*Bekam Sebagai Terapi Alternatif Untuk Nyeri*". Tinjauan Pustaka Neurona, 36(2), 148.
- Betha, S. &. (2019). *HTML 5 Dasar-Dasar Untuk Pengembangan Aplikasi Berbasis Web*. Bandung: Informatika Bandung.
- Dantes, dkk. (2019). *Pengantar Basis Data*. Depok: PT. Raja Grafindo Persada.
- Dhyah, B. &. (2015) "*Strategi Pengembangan Bisnis Pada Depot Selaris Dengan Pendekatan Business Model Canvas*". AGORA 3(2), 292–301.
- Habibi, Roni, D. (2020). *Aplikasi Kehadiran Dosen Menggunakan PHP OOP*. Bandung: Kreatif Industri Nusantara.
- Hartatik, , & Baroto, T. (2017) "*Strategi Pengembangan Bisnis Dengan Metode*

- Business Model Canvas*". Jurnal Teknik Industri 18(2), 113-120.
- Hasibuan, A. R. (2016) "*Penerapan Algoritma Floyd Warshall Untuk Menentukan Jalur Terpendek Dalam Pengiriman Barang*". JURIKOM, 3(6), 20–24.
- Hermawan, A., & Pravitasari, J. (2013). Business Model Canvas (Kanvas Model Bisnis). *Akselerasi.Id*, 1–23.
- Hutahaen, J. (2014). *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Jannah, C. S. & M. (2020). *Desain Web Bagi Pemula*. Jakarta: Elex Media Komputindo.
- Komputer, W. (2014). *Membangun Aplikasi Mobile Cross Platform dengan Phonegap*. Jakarta: PT. Elex Media Komputindo.
- Mardainis & Khusaeri A. (2017) "*Integrasi Pemrograman Web Pada Pemrograman Desktop Sebagai Alternatif Fasilitas Laporan Dalam Pengembangan Program Aplikasi*". Jurnal Teknologi Informasi & Komunikasi Digital Zone 8(1), 104–113.
- Mulyani, S. (2016). *Analisis dan Perancangan Sistem Informasi Manajemen Keuangan Daerah: Notasi Pemodelan Unified Modelling Language (UML)*. Bandung: Abdi Sistematika.
- Oktafianto, M. M. &. (2016). *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: Andi Offset.
- Pigneur, A. O. & Y. (2012). *Business Model Generation*. Jakarta: Elex Media Komputindo.
- Prasetyo, Y. Y. & H. A. (2018). *Panduan Mudah Belajar Framework Laravel*. Jakarta: Elex Media Komputindo.
- Putri, R. A. & L. (2018) "*Pengembangan Sistem Informasi Akuntansi Berbasis Desktop Dengan Metode Stradis*".J urnal Ilmu Komputer dan Informatika 3(1), 21–30.

- Hasibuan, Ahyar Rivai (2016) "*Penerapan Algoritma Floyd Warshall Untuk Menentukan Jalur Terpendek Dalam Pengiriman Barang*". JURIKOM. Vol. 3, no.6, hlm. 20-24.
- Rerung, R. R. (2018). *Pemrograman Web Dasar*. Yogyakarta: Deepublish.
- Rukin. (2019). *Metodologi Penelitian Kualitatif . Sulawesi Selatan: Ahmar Cendikia Indonesia*. Ahmar Cendikia Indonesia.
- Salahuddin M, A. . R. &. (2018). *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Badnang: Modula.
- Saputra, M & Lusia, V. A. (2020). *Belajar Cepat Metode SAW*. Bandung: Kreatif Industri Nusantara.
- Saputra. (2020). *Perkembangan Teknologi Informasi di Indonesia Menghadapi Industri 5.0*. Surabaya: Jakad Media Publishing.
- Saputra, K. (2017). *Akupunktur Indonesia*. Surabaya: Airlangga University Press.
- Sasmito, G. W. (2017) "*Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal*". Jurnal Informatika:Jurnal Pengembangan IT 2(1), 6–12.
- Sianipar, R. H. (2015). *Pemrograman Database Menggunakan MySQL*. Yogyakarta: Andi Publisher.
- Solichin, A. (2016). *Pemrograman Web dengan PHP dan MySQL*. Jakarta : Budi Luhur. Budi Luhur.
- Sovia Pramudita, A. (2018) "*Formulasi Model Bisnis Hostel di Bandung dengan Pendekatan Value Chain dan Business Model Canvas (Studi Kasus: Pinisi Backpacker)*". ISEI Business and Management Review 2(1), 32–38.
- Stroke, S., Haryono, D. C., Soemardji, A. A., Fanty, F., Sains, S., Farmasi, S., & Bandung, I. T. (2011). "*Peranan Terapi Akupunktur "GI" pada Penderita Stroke*". JKM 10(2), 142–150.

- Suendri(2018) "*Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)*". Jurnal Ilmu Komputer dan Informatika 3(1), 1–9.
- Sulaeman, Y. S. &. (2019). *Semua Bisa Menjadi Programmer Laravel*. Jakarta: Elex Media Komputindo.
- Suposono. (2018). *Pemrograman Web dengan Menggunakan PHP dan Framework Codeigniter*. Yogyakarta: CV. Budi Utama.
- Supriyadi, E. (2020). *Sistem Informasi Bisnis Dunia Versi 4.0*. Yogyakarta: Andi.
- Sutabri, T. (2012). *Konsep sistem informasi*. Yogyakarta: Andi.
- Varianto, V. (2017). Model Bisnis Colleges Need Menggunakan Pendekatan Business Model Canvas. *Jurnal Manajemen Dan Start-Up Bisnis*, 2(3), 351–358.
- Widayati, Q., & Sahfitri, V. (2020). Sistem Informasi Penjadwalan Dokter dan Pendaftaran Pasien Menggunakan Metode Round Robin. *Seri Prosiding Seminar Nasional ...*, 104–111.
<http://prosiding.senadi.upy.ac.id/index.php/senadi/article/view/136>
- Widyaningrum, Herlina (2013) *Pijat Refleksi dan 6 Terapi Alternatif Lainnya*. Yogyakarta: Medpress.
- Wong, M. F. (2011). *Panduang Lengkap Pijat*. Depok: Penebar Plus.
- www.apachefriends.org. diakses pada tanggal 26 April 2020.
- www.mysql.com. diakses pada tanggal 26 April 2020.
- www.php.net. diakses pada tanggal 26 April 2020.
- www.w3.org/html/logo. diakses pada tanggal 26 April 2020.
- Yuningsih, Rahmi "*Pengobatan Tradisional di Unit Pelayanan Kesehatan*".
[berkas.dpr.go.id InfoSingkat-IV-5-I-P3DI-Maret-2012-82.pdf](https://berkas.dpr.go.id/InfoSingkat-IV-5-I-P3DI-Maret-2012-82.pdf) (dpr.go.id)

(diakses April 27, 2020).

LAMPIRAN

Source code

Controller.php

```
<?php
namespace App\Http\Controllers;
use
Illuminate\Foundation\Auth\Access\AuthorizesRequests;
use Illuminate\Foundation\Bus\DispatchesJobs;
use
Illuminate\Foundation\Validation\ValidatesRequests;
use Illuminate\Routing\Controller as BaseController;
```

```
class Controller extends BaseController
{
    use AuthorizesRequests, DispatchesJobs,
    ValidatesRequests;
}
```

LoginController.php

```
<?php

namespace App\Http\Controllers\Auth;

use App\Http\Controllers\Controller;
use App\Providers\RouteServiceProvider;
use
Illuminate\Foundation\Auth\AuthenticatesUsers;

class LoginController extends Controller
{
    /**
     |-----
     | Login Controller
     |-----
     |
     | This controller handles authenticating users
     for the application and
     | redirecting them to your home screen. The
     controller uses a trait
     | to conveniently provide its functionality to
     your applications.
     |
     */
    use AuthenticatesUsers;

    /**
     * Where to redirect users after login.
     *
     * @var string
     */
    protected $redirectTo =
    RouteServiceProvider::HOME;
```

```
/**
 * Create a new controller instance.
 *
 * @return void
 */
public function __construct()
{
    $this->middleware('guest')-
>except('logout');
}
```

RegisterController.php

```
<?php

namespace App\Http\Controllers\Auth;

use App\Http\Controllers\Controller;
use App\Providers\RouteServiceProvider;
use App\User;
use Illuminate\Foundation\Auth\RegistersUsers;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;

class RegisterController extends Controller
{
    /**
     |-----
     | Register Controller
     |-----
     |
     | This controller handles the registration of new
     users as well as their
     | validation and creation. By default this
     controller uses a trait to
     | provide this functionality without requiring
     any additional code.
     |
     */

    use RegistersUsers;

    /**
     * Where to redirect users after registration.
     *
     * @var string
     */
    protected $redirectTo =
    RouteServiceProvider::HOME;

    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
```

```

        $this->middleware('guest');
    }

    /**
     * Get a validator for an incoming registration
     request.
     *
     * @param array $data
     * @return
     */
    \Illuminate\Contracts\Validation\Validator
    */
    protected function validator(array $data)
    {
        return Validator::make($data, [
            'name' => ['required', 'string', 'max:255'],
            'email' => ['required', 'string', 'email',
            'max:255', 'unique:users'],
            'password' => ['required', 'string', 'min:8',
            'confirmed'],
        ]);
    }

    /**
     * Create a new user instance after a valid
     registration.
     *
     * @param array $data
     * @return \App\User
     */
    protected function create(array $data)
    {
        return User::create([
            'name' => $data['name'],
            'email' => $data['email'],
            'role' => "PASIEN",
            'password' =>
            Hash::make($data['password']),
        ]);
    }
}

```

AdminController.php

```

<?php

namespace App\Http\Controllers;

use App\Models\Order;
use App\Terapi;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\DB;

class AdminController extends Controller
{
    public function index()
    {
        if(Auth::user()->role=='PASIEN'){
            return redirect()->route('landing');
        }
        if(Auth::user()->role=='ADMIN'){

```

```

            $data['today']=DB::table('orders')-
            >whereDate('tanggal_pengobatan',date('Y-m-d'))-
            >where('total_payment','Sukses')->count();
            $data['pasien']=DB::table('users')-
            >where('role','PASIEN')->count();
            $data['klinik']=DB::table('users')-
            >where('role','KLINIK')->count();
            $data['transaksi']=DB::table('orders')-
            >where('total_payment','Sukses')-
            >sum('therapy_price');

            $data['olds']=DB::table('orders')-
            >whereDate('tanggal_pengobatan','<',date('Y-m-
            d'))->where('total_payment','Sukses')->get();
            $data['days']=DB::table('orders')-
            >whereDate('tanggal_pengobatan',date('Y-m-d'))-
            >where('total_payment','Sukses')->get();
            $data['soons']=DB::table('orders')-
            >whereDate('tanggal_pengobatan','>',date('Y-m-
            d'))->where('total_payment','Sukses')->get();
            return view('backend.admin.index',$data);
        }else{
            $data['today']=DB::table('orders')-
            >whereDate('tanggal_pengobatan',date('Y-m-d'))-
            >where('total_payment','Sukses')-
            >where('klinik_id',Auth::id())->count();
            $data['pasien']=DB::table('orders')-
            >where('klinik_id',Auth::id())->count();
            $data['transaksi']=DB::table('orders')-
            >where('total_payment','Sukses')-
            >where('klinik_id',Auth::id())-
            >sum('therapy_price');

            $data['olds']=DB::table('orders')-
            >whereDate('tanggal_pengobatan','<',date('Y-m-
            d'))->where('total_payment','Sukses')-
            >where('klinik_id',Auth::id())->get();
            $data['days']=DB::table('orders')-
            >whereDate('tanggal_pengobatan',date('Y-m-d'))-
            >where('total_payment','Sukses')-
            >where('klinik_id',Auth::id())->get();
            $data['soons']=DB::table('orders')-
            >whereDate('tanggal_pengobatan','>',date('Y-m-
            d'))->where('total_payment','Sukses')-
            >where('klinik_id',Auth::id())->get();
            // dd($data);
            return view('backend.klinik.admin',$data);
        }
    }

    public function transaksi()
    {
        $data['orders']=Order::orderBy('updated_at','
        DESC')->get();
        // dd($data);
        return view('backend.transaksi.index',$data);
    }

    public function detail($id)
    {
        $data['order']=Order::find($id);
        // dd($data);

```

```

        return view('backend.transaksi.detail',$data);
    }

    public function konfirmasi($id,Request
$request)
    {
        $order=new Order();
        $order=Order::find($id);
        $order->total_payment=$request->review;
        $order->pesan=$request->pesan;
        $order->total_payment='Sukses';
        $order->save();
        return redirect()->route('admin.transaksi');
    }

    public function pasien()
    {
        $data['pasiens']=DB::table('users')-
>where('role','PASIEN')->get();
        return view('backend.pasien.index',$data);
    }
    public function listpasien()
    {
        if(Auth::user()->role=='KLINIK'){
            $data['pasiens']=DB::table('orders')
            -
            >select('orders.*','users.name','users.email')
            ->leftJoin('users','users.id','orders.user_id')
            ->where('total_payment','Sukses')
            ->where('tanggal_pengobatan','>=',
date('Y-m-d'))
            ->where('klinik_id',Auth::id())->get();
        }else{
            $data['pasiens']=DB::table('orders')
            -
            >select('orders.*','users.name','users.email')
            ->leftJoin('users','users.id','orders.user_id')
            ->where('total_payment','Sukses')
            ->where('tanggal_pengobatan','>=',
date('Y-m-d'))->get();
        }
        // dd($data);
        return
view('backend.pasien.listpasien',$data);
    }

    public function terapi()
    {
        $data['terapi']=Terapi::all();
        return view('backend.terapi.index',$data);
    }

    public function terapiStore(Request $request)
    {
        // dd($request);
        $file = $request->file('image');
        $fileName=time().'.'.$file-
>getClientOriginalExtension();
        if ($request->hasFile('image')) {
            $path = public_path().'uploads/terapi/';
            // dd($path);
            $file->move($path,$fileName);

```

```

        }
        $terapi=new Terapi();
        $terapi->name=$request->terapi;
        $terapi->image=$fileName;
        $terapi->save();
        return redirect()->route('terapi.list');
    }

    public function terapiUpdate(Request
$request,$id)
    {
        if ($request->hasFile('image')) {
            $file = $request->file('image');
            $fileName=time().'.'.$file-
>getClientOriginalExtension();
            $path = public_path().'uploads/terapi/';
            // dd($path);
            $file->move($path,$fileName);
        }
        $terapi=new Terapi();
        $terapi=Terapi::find($id);
        $terapi->name=$request->terapi;
        if ($request->hasFile('image')) {
            $terapi->image=$fileName;
        }
        $terapi->save();
        return redirect()->route('terapi.list');
    }

    public function delete($id)
    {
        DB::table('terapi')->where('id',$id)-
>delete();
    }

}

```

FrontEndController.php

```

<?php

namespace App\Http\Controllers;

use App\Models\Order;
use App\Models\User;
use App\Terapi;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\DB;

class FrontEndController extends Controller
{
    //
    public function index(){
        // $data=DB::select(DB::raw('SELECT
        // id,
        // (
        // 3959 *
        // acos(cos(radians(37)) *
        // cos(radians(latitude)) *
        // cos(radians(longitude) -
        // radians(-122)) +

```

```

// sin(radians(37)) *
// sin(radians(latitude )))
// ) AS distance
// FROM clinics
// HAVING distance > 2
// ORDER BY distance LIMIT 0, 20;));
// dd($data);
$data['terapis']=Terapi::all();

return view('frontend.index',$data);
}

public function auth()
{
    if(Auth::user()->role=='PASIEN'){
        return redirect()->route('landing');
    }else{
        return redirect()->route('klinik');
    }
}

public function profile()
{
    $data['user']=Auth::user();
    $data['orders']=Order::where('user_id',Auth::
id()->get());
    // dd($data);
    return view('frontend.user',$data);
}

public function updateprofile(Request
$request)
{
    $user=new User();
    $user=User::find(Auth::id());
    $user->name=$request->name;
    $user->save();
    return redirect()->route('profile');
}

public function detail($id)
{
    $data['order']=Order::find($id);
    return view('frontend.detail',$data);
}

public function searchKlinik() {
    return view('frontend.search_klinik');
}

public function search($slug,Request $request)
{
    // dd($request);
    $latlong=explode(',',$request->latlong);

    $data['kliniks']=DB::select(DB::raw('SELEC
T
    users.id,users.name,
    users.email,
    clinics.klinik_name,
    clinics.klinik_owner,

```

```

    clinics.klinik_owner_phone,
    clinics.klinik_permission,
    clinics.klinik_address,
    clinics.photo,
    clinics.klinik_phone,
    clinics.klinik_therapist,
    clinics.klinik_open_close,
    clinics.klinik_time_per_day,
    therapies.therapy_name,
    therapies.price,
    (3956 * 2 * ASIN(SQRT( POWER(SIN((
'. $latlong[0].' - latitude) * pi()/180 / 2), 2) +COS(
'. $latlong[0].' * pi()/180 * COS(latitude *
pi()/180 * POWER(SIN(( '. $latlong[1].' -
longitude) * pi()/180 / 2), 2) ))) as distance
FROM medisreborn.clinics
LEFT JOIN users ON users.id =
clinics.user_id
LEFT JOIN therapies ON users.id =
therapies.user_id
WHERE therapy_name = '".$slug.'"
having distance <= 10000
order by distance LIMIT 0, 20;));
// dd($data);
return view('frontend.search_akupuntur',
$data);
}

public function searchBekam() {
    $data['kliniks'] = DB::table('users')
->select(
        'users.id',
        'users.klinik_name',
        'users.klinik_owner',
        'users.klinik_owner_phone',
        'users.klinik_address',
        'users.photo',
        'users.klinik_phone',
        'users.klinik_therapist',
        'users.klinik_open_close',
        'users.klinik_time_per_day',
        't.therapy_name',
        't.price',
    )->join('therapies as t', 't.user_id', '=',
'users.id')
->where('t.therapy_name', '=', 'bekam')
->get();
    // dd($data);
    return view('frontend.search_bekam', $data);
}

public function searchPijat() {
    $data['kliniks'] = DB::table('users')
->select(
        'users.id',
        'users.klinik_name',
        'users.klinik_owner',
        'users.klinik_owner_phone',
        'users.klinik_address',
        'users.photo',
        'users.klinik_phone',
        'users.klinik_therapist',

```

```

        'users.klinik_open_close',
        'users.klinik_time_per_day',
        't.therapy_name',
        't.price',
    )->join('therapies as t', 't.user_id', '=',
'users.id')
    ->where('t.therapy_name', '=', 'pijat')
    ->get();
    // dd($data);
    return view('frontend.search_pijat', $data);
}
}

```

HomeController.php

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

```

class HomeController extends Controller
{
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('auth');
    }

    /**
     * Show the application dashboard.
     *
     * @return
     */
    \Illuminate\Contracts\Support\Renderable
    */
    public function index()
    {
        return view('home');
    }
}

```

KlinikController.php

<?php

namespace App\Http\Controllers;

```

use App\Clinic;
use App\Models\User;
use App\Models\Therapy;
use App\Terapi;
use Carbon\Carbon;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Facades\Hash;

```

class KlinikController extends Controller

```

{
    //
    public function index(){
        if(Auth::user()->role=='ADMIN'){
            $data['kliniks'] = DB::table('users')
                ->select('users.id',
                    'clinics.klinik_name',
                    'users.email', 'clinics.klinik_owner',
                    'clinics.klinik_owner_phone',
                    'clinics.klinik_permission',
                    'clinics.klinik_address',
                    'clinics.photo',
                    'clinics.klinik_phone',
                    'clinics.klinik_therapist',
                    'clinics.klinik_open_close',
                    'clinics.klinik_time_per_day',
                )
                -
            >leftJoin('clinics','users.id','clinics.user_id')
                ->where('users.role', '=', 'KLINIK')
                ->get();
            return view('backend.klinik.index', $data);
        }elseif(Auth::user()->role=='KLINIK'){
            // echo "ME"; die;
            return redirect()-
            >route('laporan.keuangan');
        }else{
            return redirect()->route('landing');
        }
    }

    public function detail($id)
    {
        $data['klinik']=DB::table('users')
            ->select('users.id','users.name',
                'clinics.klinik_name',
                'users.email', 'clinics.klinik_owner',
                'clinics.klinik_owner_phone',
                'clinics.klinik_permission',
                'clinics.klinik_address',
                'clinics.photo',
                'clinics.klinik_phone',
                'clinics.klinik_therapist',
                'clinics.klinik_open_close',
                'clinics.klinik_time_per_day',
            )
            ->leftJoin('clinics','users.id','clinics.user_id')
            ->where('users.id',$id)->first();
            return view('backend.klinik.detail',$data);
        }

    public function add(){
        $data['terapis']=Terapi::all();
        // dd($data);
        return view('backend.klinik.create',$data);
    }

    public function save(Request $request){
        // dd($request);
        $folderPath = public_path('uploads/klinik/');

        $file = $request->file('foto_klinik');
    }
}

```

```

        $fileName=time().'.'.$file-
>getClientOriginalExtension();
        if ($request->hasFile('foto_klinik')) {
            $path = public_path().'/uploads/klinik/';
            // dd($path);
            $file->move($path,$fileName);
        }
        $klinik = new User();
        $klinik->name = $request->klinik_name;
        // for login data
        $klinik->role = "KLINIK";
        $klinik->password =
Hash::make($request->password);
        $klinik->email = $request->email;
        $klinik->save();

        $clinic=new Clinic();
        $clinic->user_id = $klinik->id;
        $clinic->photo = 'klinik/' . $fileName;
        $clinic->klinik_name = $request-
>klinik_name;
        $clinic->klinik_owner = $request-
>owner_name;
        $clinic->klinik_owner_phone = $request-
>owner_phone;
        $clinic->klinik_permission = $request-
>permission;
        $clinic->klinik_address = $request-
>address;
        $clinic->klinik_phone = $request-
>klinik_phone;
        $clinic->klinik_therapist = $request-
>therapist;
        $clinic->klinik_open_close = $request-
>open_close;
        $clinic->klinik_time_per_day = $request-
>time_per_day;
        $clinic->latitude = $request->latitude;
        $clinic->longitude = $request->longitude;
        $clinic->save();

        $lengTp = count($request->therapy);
        for ($i=0; $i < $lengTp ; $i++) {
            // dd($request->therapy[$i] );
            if($request->price[$i] != null &&
$request->therapy[$i] != null){
                $therapy = new Therapy();
                $therapy->price = $request-
>price[$i];
                $therapy->therapy_name =
strtolower($request->therapy[$i]);
                $therapy->user_id = $klinik->id;
                $therapy->save();
            }
        }
        return redirect()->route('klinik');
    }

    public function edit($id)
    {
        $data['klinik']=DB::table('users')
->select('users.*',

```

```

        'clinics.klinik_name',
        'users.email', 'clinics.klinik_owner',
        'clinics.klinik_owner_phone',
        'clinics.klinik_permission',
        'clinics.klinik_address',
        'clinics.photo',
        'clinics.user_id',
        'clinics.klinik_phone',
        'clinics.klinik_therapist',
        'clinics.klinik_open_close',
        'clinics.klinik_time_per_day',
        'clinics.latitude',
        'clinics.longitude',
    )
    ->leftJoin('clinics','users.id','clinics.user_id')
    ->where('users.id',$id)->first();
    $data['terapis']=Therapy::where('user_id',$d
ata['klinik']->user_id)->get();
    $data['terapis']=Terapi::all();
    // dd($data);
    return view('backend.klinik.edit',$data);
}

    public function update(Request $request,$id)
    {
        // dd($request);
        $folderPath = public_path('uploads/klinik/');

        if ($request->hasFile('foto_klinik')) {
            $file = $request->file('foto_klinik');
            $fileName=time().'.'.$file-
>getClientOriginalExtension();
            $path = public_path().'/uploads/klinik/';
            // dd($path);
            $file->move($path,$fileName);
        }
        $clinic=new Clinic();
        $clinic=Clinic::where('user_id',$request-
>user_id)->first();

        if ($request->hasFile('foto_klinik')) {
            $clinic->photo = 'klinik/' . $fileName;
        }
        $clinic->klinik_name = $request-
>klinik_name;
        $clinic->klinik_owner = $request-
>owner_name;
        $clinic->klinik_owner_phone = $request-
>owner_phone;
        $clinic->klinik_permission = $request-
>permission;
        $clinic->klinik_address = $request-
>address;
        $clinic->klinik_phone = $request-
>klinik_phone;
        $clinic->klinik_therapist = $request-
>therapist;
        $clinic->klinik_open_close = $request-
>open_close;
        $clinic->klinik_time_per_day = $request-
>time_per_day;
        $clinic->latitude = $request->latitude;

```



```

        $clinic->longitude = $request->longitude;
        $clinic->save();

        $lengTp = count($request->therapy);
        for ($i=0; $i < $lengTp ; $i++) {
            // dd($request->therapy[$i] );
            if($request->price[$i] != null &&
            $request->therapy[$i] != null){
                $therapy = new Therapy();
                $therapy=Therapy::where('user_id',$
request->user_id)->first();
                $therapy->price = $request-
>price[$i];
                $therapy->therapy_name =
strtolower($request->therapy[$i]);
                $therapy->user_id = $request-
>user_id;
                $therapy->save();
            }
        }
        return redirect()->route('klinik');
    }

    public function laporan(Request $request)
    {
        // dd(date('Y-m-d', strtotime("-7 days")));
        $data['orders'] = DB::table('orders')
            ->where(function ($query) use ($request)
            {
                if(Auth::user()->role=='KLINIK'){
                    $query-
>where('klinik_id',Auth::id());
                }

                if ($request->date == "Hari Ini") {
                    $query-
>whereDate('updated_at',date('Y-m-d'));
                }
                elseif($request->date == "Minggu"){
                    $query-
>whereDate('updated_at','>',date('Y-m-d',
strtotime("-7 days")));
                }
                elseif($request->date == "Bulan"){
                    $query-
>whereDate('updated_at','>',date('Y-m-d',
strtotime("-30 days")));
                }else{

                }
            })
        // ->sum('therapy_price');
        ->get();
        $data['sum'] = DB::table('orders')
            ->where(function ($query) use ($request)
            {
                if(Auth::user()->role=='KLINIK'){
                    $query-
>where('klinik_id',Auth::id());
                }

                if ($request->date == "Hari Ini") {

```

```

                    $query-
>whereDate('updated_at',date('Y-m-d'));
                }
                elseif($request->date == "Minggu"){
                    $query-
>whereDate('updated_at','>',date('Y-m-d',
strtotime("-7 days")));
                }
                elseif($request->date == "Bulan"){
                    $query-
>whereDate('updated_at','>',date('Y-m-d',
strtotime("-30 days")));
                }else{

                }
            })
        ->sum('therapy_price');

        // dd($data);

        return view('backend.admin.laporan',$data);
    }

    public function delete($id)
    {
        DB::table('users')->where('id',$id)->delete();
        DB::table('clinics')->where('user_id',$id)-
>delete();
        DB::table('therapis')->where('user_id',$id)-
>delete();
    }
}

```

Order.Controller.php

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
use App\Models\Therapy;
use App\Models\User;
use App\Models\Order;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Redirect;

class OrderController extends Controller
{
    public function __construct(){
        $this->middleware('auth');
    }
    //
    public function
orderPage($klinik_id,$jenis_klinik){
        $klinik_id = (int)$klinik_id;
        $data["klinik"] = DB::table('users')
            ->select(
                'users.id',
                'clinics.klinik_name',
                'clinics.klinik_address',
                'clinics.photo',

```

```

        'clinics.klinik_open_close',
        'clinics.klinik_time_per_day',
    )
    -
>leftJoin('clinics','clinics.user_id','users.id')
    ->where('users.id', '=', $klinik_id)
    ->first();

    $data["therapy"] = Therapy::where('user_id',
    $klinik_id)
        ->get();
    $data["jenis_terapi"] = $jenis_klinik;
    $data["jam"] = ['08:00-09:00','09:00-
    10:00','10:00-11:00','11:00-12:00','12:00-
    13:00','13:00-14:00','14:00-15:00','15:00-
    16:00','16:00-17:00','17:00-18:00','18:00-
    19:00','19:00-20:00','20:00-21:00'];
    // dd($data);
    return view('frontend.order.index', $data);
}

public function generate(Request $request)
{
    // dd($request);
    $bed = DB::table('users')-
>select('klinik_therapist')->where('id',$request-
>klinik_id)->first();

    $total = DB::table('orders')
    ->where('klinik_id',$request->klinik_id)
    ->where('total_payment','Sukses')
    ->where('jam_pengobatan',$request->time)
    ->whereDate('tanggal_pengobatan',date('Y-
    m-d',strtotime($request->tanggal)))
    ->get();
    // dd($bed->klinik_therapist);
    if(count($total)>$bed->klinik_therapist){
        return Redirect::back()->withErrors('Pilih
        Jam Lain atau Jam yang Sama Pada Tanggal
        Yang Lain');
    }
    $no_urut = DB::table('orders')
    ->where('klinik_id',$request->klinik_id)
    ->where('jam_pengobatan',$request->time)
    ->whereDate('tanggal_pengobatan',date('Y-
    m-d',strtotime($request->tanggal)))
    ->get();

    $orderId = "TRF" . str_replace("-", "",
    $request->tanggal) . strtotime(now());
    $therapy = Therapy::where('id',
    (int)$request->jenis_pengobatan)->first();
    $order = new Order();
    $order->order_id = $orderId;
    $order->name = $request->name;
    $order->phone = $request->phone;
    $order->therapy_name = $therapy-
>therapy_name;
    $order->klinik_name = $request-
>klinik_name;
    $order->therapy_price = $therapy->price;

```

```

    $order->tanggal_pengobatan = $request-
>tanggal;
    $order->jam_pengobatan = $request->time;
    $order->no_urut = count($no_urut)+1;
    $order->klinik_id = $request->klinik_id;
    $order->user_id = Auth::user()->id;
    $order->save();

    return redirect("/page/order/payment/" .
    $orderId);
}

public function paymentPage($order_id){
    $data['order'] = Order::where('order_id',
    $order_id)->first();
    // dd($data);
    return view('frontend.order.payment', $data);
}

public function paymentSave(Request
$request){
    $folderPath =
    public_path('uploads/bukti_transfer/');

    $image_parts = explode(";base64,",
    $request->iconMerk);
    $image_type_aux = explode("image/",
    $image_parts[0]);
    $image_type = $image_type_aux[1];
    $image_base64 =
    base64_decode($image_parts[1]);
    $filename = uniqid() . '.png';
    $file = $folderPath . $filename;
    $uploadMerkIcon = file_put_contents($file,
    $image_base64);
    $order = Order::find((int)$request-
>order_id);
    $order->bukti_transfer = 'bukti_transfer/' .
    $filename;
    $order->save();
    return redirect()->route('after-transfer');
}

public function afterTransfer(){
    return view('frontend.order.after_payment');
}
}

```

User.php

```

<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Foundation\Auth\User as
Authenticatable;
use Illuminate\Notifications\Notifiable;

class User extends Authenticatable
{
    use Notifiable;

```

```

/**
 * The attributes that are mass assignable.
 *
 * @var array
 */
protected $fillable = [
    'name',
    'email',
    'password',
];

/**
 * The attributes that should be hidden for
arrays.
 *
 * @var array
 */
protected $hidden = [
    'password',
    'remember_token',
];

/**
 * The attributes that should be cast to native
types.
 *
 * @var array
 */
protected $casts = [
    'email_verified_at' => 'datetime',
];
}

```

Create_user_table.php

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateUsersTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('users', function (Blueprint
$table) {
            $table->id();
            $table->string('name');
            $table->string('email')->unique();
            $table->timestamp('email_verified_at')-
>nullable();
            $table->string('password');
            $table->string('role')->nullable();
            $table->string('gender')->nullable();
            $table->string('birth')->nullable();

```

```

            $table->string('address')->nullable();
            $table->string('phone')->nullable();
            $table->string('photo')->nullable();
            $table->rememberToken();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('users');
    }
}

```

Create_reset_password_table.php

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreatePasswordResetsTable extends
Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('password_resets', function
(Blueprint $table) {
            $table->string('email')->index();
            $table->string('token');
            $table->timestamp('created_at')-
>nullable();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('password_resets');
    }
}

```

Create_table_therapy.php

```

<?php

use Illuminate\Database\Migrations\Migration;

```

```
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
```

```
class CreateTableTherapy extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('therapies', function
(Blueprint $table) {
            $table->id();
            $table->string('therapy_name');
            $table->integer('price');
            $table->foreignId('user_id');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('therapy');
    }
}
```

Create_table_order.php

```
<?php
```

```
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
```

```
class CreateTableOrder extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('orders', function (Blueprint
$table) {
            $table->id();
            $table->string('name');
            $table->string('phone');
            $table->string('order_id');
            $table->string('klinik_name');
            $table->string('therapy_name');
            $table->string('therapy_price');
            $table->string('total_payment')-
>nullable();
            $table->date('tanggal_pengobatan');
```

```
            $table->string('jam_pengobatan')-
>nullable();
            $table->string('bukti_transfer')-
>nullable();
            $table->string('pesan')->nullable();
            $table->integer('no_urut')->nullable();
            $table->index('order_id');

            $table->foreignId('klinik_id');
            $table->foreignId('user_id');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('table_order');
    }
}
```

Create_clinics_tabel.php

```
<?php
```

```
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
```

```
class CreateClinicsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('clinics', function (Blueprint
$table) {
            $table->id();
            $table->foreignId('user_id');
            $table->string('photo');
            $table->string('klinik_name');
            $table->string('klinik_owner');
            $table->string('klinik_owner_phone');
            $table->string('klinik_permission');
            $table->string('klinik_address');
            $table->string('klinik_phone');
            $table->integer('klinik_therapist');
            $table->string('klinik_open_close');
            $table->string('klinik_time_per_day');
            $table->string('latitude');
            $table->string('longitude');
            $table->timestamps();
        });
    }
}
```

```

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('clinics');
}

```

Create_terapis_table.php

```
<?php
```

```

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

```

```

class CreateTerapisTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('terapis', function (Blueprint
$table) {
            $table->id();
            $table->string('name');
            $table->string('image');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('terapis');
    }
}

```

Admin (index.blade.php)

```

@extends('layouts.backend')
@section('css')
<link href="/css/tabs.css" rel="stylesheet">
@endsection
@section('content')
<div class="content">
    <!-- Small boxes (Stat box) -->
    <div class="row">
        <div class="col-lg-3 col-sm-6 col-xs-12">

```

```

            <div class="info-box bg-darkblue"> <span
class="info-box-icon bg-transparent"><i
class="ti-stats-up text-white"></i></span>
            <div class="info-box-content">
                <h6 class="info-box-text text-
white">Pasien Hari Ini</h6>
                <h1 class="text-white">{{ $today }}</h1>
            <!-- /.info-box-content -->
            </div>
        <!-- /.info-box -->
    </div>
    <div class="col-lg-3 col-sm-6 col-xs-12">
        <div class="info-box bg-green text-white">
            <span class="info-box-icon bg-transparent"><i
class="ti-face-smile"></i></span>
            <div class="info-box-content">
                <h6 class="info-box-text text-white">Total
Pasien</h6>
                <h1 class="text-white">{{ $pasien }}</h1>
            <!-- /.info-box-content -->
            </div>
        <!-- /.info-box -->
    </div>
    <!-- /.col -->
    <div class="col-lg-3 col-sm-6 col-xs-12">
        <div class="info-box bg-aqua"> <span
class="info-box-icon bg-transparent"><i
class="ti-bar-chart"></i></span>
        <div class="info-box-content">
            <h6 class="info-box-text text-white">Total
Klinik</h6>
            <h1 class="text-white">{{ $klinik }}</h1>
        <!-- /.info-box-content -->
        </div>
    <!-- /.info-box -->
    </div>
    <!-- /.col -->
    <div class="col-lg-3 col-sm-6 col-xs-12">
        <div class="info-box bg-orange"> <span
class="info-box-icon bg-transparent"><i
class="ti-money"></i></span>
        <div class="info-box-content">
            <h6 class="info-box-text text-white">Total
Transaksi</h6>
            <h1 class="text-white">Rp.
{{ number_format($transaksi) }}</h1>
        <!-- /.info-box-content -->
        </div>
    <!-- /.info-box -->
    </div>
    <!-- /.col -->
    </div>
    <div class="row">
        <div class="col-md-12">
            <div class="card">
                <div class="card-body p-b-0">

```

```

        <h4 class="card-title text-black text-center">Transaksi History</h4>
    </div>
    <!-- Nav tabs -->
    <ul class="nav nav-tabs customtab"
    role="tablist" style="display: flex; margin:
    auto;">
        <li class="nav-item"> <a class="nav-link"
        data-toggle="tab" href="#yesterday"
        role="tab"><span class="hidden-sm-up"><i
        class="ti-home"></i></span> <span
        class="hidden-xs-down">Transaksi
        Kemarin</span></a> </li>
        <li class="nav-item"> <a class="nav-link"
        active" data-toggle="tab" href="#today"
        role="tab"><span class="hidden-sm-up"><i
        class="ti-user"></i></span> <span
        class="hidden-xs-down">Transaksi Hari
        Ini</span></a> </li>
        <li class="nav-item"> <a class="nav-link"
        data-toggle="tab" href="#tomorrow"
        role="tab"><span class="hidden-sm-up"><i
        class="ti-email"></i></span> <span
        class="hidden-xs-down">Transaksi
        Besok</span></a> </li>
    </ul>
    <!-- Tab panes -->
    <div class="tab-content">
        <div class="tab-pane p-20" id="yesterday"
        role="tabpanel">
            <div class="table-responsive">
                <table class="table table-hover">
                    <thead>
                        <tr>
                            <th scope="col">#</th>
                            <th scope="col">Nama
                            Pasien</th>
                            <th scope="col">Nama
                            Klinik</th>
                            <th scope="col">Tanggal</th>
                            <th scope="col">Jam</th>
                            <th scope="col">Status</th>
                        </tr>
                    </thead>
                    <tbody>
                        @foreach ($solds as $sold)
                            <tr>
                                <th scope="row">{{ $loop-
                                >iteration }}</th>
                                <td><a href="{{ {
                                route('admin.detail', $sold->id) }}">{{ $sold-
                                >name }}</a></td>
                                <td>{{ $sold-
                                >klinik_name }}</td>
                                <td>{{ $sold-
                                >tanggal_pengobatan }}</td>
                                <td>{{ $sold-
                                >jam_pengobatan }}</td>
                                <td>
                                    @if ($sold->bukti_transfer)
                                        Review Bukti TF
                                    @else
                                        Belum Ada Bukti TF
                                    @endif
                                </td>
                            </tr>
                        </tbody>
                    </table>
                </div>
            </div>
        </div>
    </div>

```

```

        Belum Ada Bukti TF
    @endif
</td>
</tr>
</tbody>
</table>
</div>
<div class="tab-pane active" id="today"
role="tabpanel">
    <div class="table-responsive">
        <table class="table table-hover">
            <thead>
                <tr>
                    <th scope="col">#</th>
                    <th scope="col">Nama
                    Pasien</th>
                    <th scope="col">Nama
                    Klinik</th>
                    <th scope="col">Tanggal</th>
                    <th scope="col">Jam</th>
                    <th scope="col">Status</th>
                </tr>
            </thead>
            <tbody>
                @foreach ($days as $sold)
                    <tr>
                        <th scope="row">{{ $loop-
                        >iteration }}</th>
                        <td><a href="{{ {
                        route('admin.detail', $sold->id) }}">{{ $sold-
                        >name }}</a></td>
                        <td>{{ $sold-
                        >klinik_name }}</td>
                        <td>{{ $sold-
                        >tanggal_pengobatan }}</td>
                        <td>{{ $sold-
                        >jam_pengobatan }}</td>
                        <td>
                            @if ($sold->bukti_transfer)
                                Review Bukti TF
                            @else
                                Belum Ada Bukti TF
                            @endif
                        </td>
                    </tr>
                </tbody>
            </table>
        </div>
    </div>
    <div class="tab-pane p-20" id="tomorrow"
    role="tabpanel">
        <div class="table-responsive">
            <table class="table table-hover">
                <thead>
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Nama
                        Pasien</th>

```

```

        <th scope="col">Nama
Klinik</th>
        <th scope="col">Tanggal</th>
        <th scope="col">Jam</th>
        <th scope="col">Status</th>
    </tr>
</thead>
<tbody>
    @foreach ($soons as $old)
    <tr>
        <th scope="row">{{ $loop-
>iteration }}</th>
        <td><a href="{{ {
route('admin.detail', $old->id) }}">{{ $old-
>name }}</a></td>
        <td>{{ { $old-
>klinik_name }}</td>
        <td>{{ { $old-
>tanggal_pengobatan }}</td>
        <td>{{ { $old-
>jam_pengobatan }}</td>
        <td>
            @if ($old->bukti_transfer)
                Review Bukti TF
            @else
                Belum Ada Bukti TF
            @endif
        </td>
    </tr>
    @endforeach
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
@endsection
Admin (laporan.blade.php)

@extends('layouts.backend')
@section('title', 'Pasien')
@section('content')
<div class="row">
    <div class="col-md-12 mb-5">
        <form action="{{ { route('laporan.keuangan')
}} " method="POST" id="form">@csrf
        <select id="mySelect" name="date"
onchange="myFunction()" class="form-control">
            <option value="Semua">Semua
Transaksi</option>
            <option value="Hari Ini">Hari
Ini</option>
            <option value="Seminggu">1 Minggu
Yang Lalu</option>
            <option value="Sebulan">1 Bulan
Yang Lalu</option>
        </select>
        </form>
    </div>

```

```

<div class="col-12">
    <div class="row">
        <div class="col-12">
            <div class="table-responsive">
                <table class="table table-hover">
                    <thead>
                        <tr>
                            <th scope="col">#</th>
                            <th scope="col">Nama
Pasien</th>
                            <th scope="col">Nama
Klinik</th>
                            <th
scope="col">Tanggal</th>
                            <th scope="col">Jam</th>
                            <th scope="col">Harga</th>
                        </tr>
                    </thead>
                    <tbody>
                        @foreach ($orders as $order)
                        <tr>
                            <th scope="row">{{ $loop-
>iteration }}</th>
                            <td><a href="{{ {
route('admin.detail', $order->id) }}">{{ $order-
>name }}</a></td>
                            <td>{{ { $order-
>klinik_name }}</td>
                            <td>{{ { $order-
>tanggal_pengobatan }}</td>
                            <td>{{ { $order-
>jam_pengobatan }}</td>
                            <td>
                                Rp.
                                {{ { number_format($order->therapy_price) }}
                                </td>
                        </tr>
                        @endforeach
                    </tbody>
                </table>
            </div>
            <div class="table-responsive">
                <table class="table table-bordered">
                    <thead>
                        <tr>
                            <th scope="col">#</th>
                            <th scope="col"></th>
                            <th scope="col">Total
Transaksi</th>
                            <th
scope="col">Potongan</th>
                            <th scope="col"></th>
                            <th
scope="col">Pendapatan</th>
                        </tr>
                    </thead>
                    <tbody>
                        <tr>
                            <td>Total Transaksi</td>
                            <td>:</td>
                            <td>Rp.
                                {{ { number_format($sum) }}</td>

```

```

                <td>-7%</td>
                <td>:</td>
                @if (Auth::user()-
>role=='KLINIK')
                <td>Rp.
            {{ number_format($sum-($sum*0.07)) }}</td>
            @else
            <td>Rp.
            {{ number_format(($sum*0.07)) }}</td>
            @endif
        </tr>
    </tbody>
</table>
</div>
</div>
</div>
</div>
@endsection
@section('js')
<script>
    function myFunction() {
        document.getElementById("form").submi
t();
    }
</script>
@endsection

```

Dashboard Backend (index.blade.php)

```

@extends('layouts.backend')
@section('content')
<div class="table-responsive">
    <table class="table table-hover">
        <thead>
            <tr>
                <th scope="col">#</th>
                <th scope="col">First Name</th>
                <th scope="col">Last Name</th>
                <th scope="col">Username</th>
                <th scope="col">Role</th>
                <th scope="col">Country</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <th scope="row">1</th>
                <td>Alexander</td>
                <td>Orton</td>
                <td>@mdorton</td>
                <td>Admin</td>
                <td>USA</td>
            </tr>
            <tr>
                <th scope="row">2</th>
                <td>John Deo</td>
                <td>Deo</td>
                <td>@johndeo</td>
                <td>User</td>
                <td>USA</td>
            </tr>
        </tbody>
    </table>

```

```

                <th scope="row">3</th>
                <td>Randy Orton</td>
                <td>the Bird</td>
                <td>@twitter</td>
                <td>admin</td>
                <td>UK</td>
            </tr>
            <tr>
                <th scope="row">4</th>
                <td>Randy Mark</td>
                <td>Ottandy</td>
                <td>@mdothe</td>
                <td>user</td>
                <td>AUS</td>
            </tr>
            <tr>
                <th scope="row">5</th>
                <td>Ram Jacob</td>
                <td>Thornton</td>
                <td>@twitter</td>
                <td>admin</td>
                <td>IND</td>
            </tr>
        </tbody>
    </table>
</div>
@endsection

```

Klinik (add.blade.php)

```

@extends('layouts.backend')
@section('title', 'Klinik add')
@section('css')
    <link rel="stylesheet"
href="{{ {{ URL::asset('backend') }} }}/plugins/dropify
/dropify.min.css">
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/cropperjs/1.5.6/cropper.css" integrity="sha256-
jKV9n9bkk/CTP8zbtEtnKaKf+ehRovOYeKoyft
hwbC8=" crossorigin="anonymous" />
<style>
    img {
        display: block;
        max-width: 100%;
    }

    .preview {
        overflow: hidden;
        width: 160px;
        height: 160px;
        margin: 10px;
        border: 1px solid red;
    }

    .modal-lg {
        max-width: 1000px !important;
    }
</style>
@endsection
@section('content')
<div class="row">

```



```

<div class="col-12">
  <h4 class="text-black">Form Input Mitra
  Klinik</h4>
  <span class="m-3 pull-right">
    </span>
  </div>
</div>
<form action="{ { route('klinik-save') } }"
method="POST" enctype="multipart/form-data">
  @csrf
  <!-- {{ csrf_field() }} -->

  <div class="row">
    <div class="col-lg-5">
      <fieldset class="form-group">
        <label>Nama Klinik</label>
        <input class="form-control" id="klinik-
name" type="text" name="klinik_name"
placeholder="" required>
      </fieldset>
      <fieldset class="form-group">
        <label>Nama Pemilik</label>
        <input class="form-control" id="klinik-
owner" type="text" name="owner_name"
placeholder="" required>
      </fieldset>
      <fieldset class="form-group">
        <label>No Ho Pemilik</label>
        <input class="form-control" id="klinik-
owner-phone" type="text" name="owner_phone"
placeholder="" required>
      </fieldset>
      <fieldset class="form-group">
        <label>Izin Usaha</label>
        <input class="form-control" id="klinik-
permission" type="text" name="permission"
placeholder="" required>
      </fieldset>
      <fieldset class="form-group">
        <label>Alamat</label>
        <input class="form-control" id="klinik-
address" type="text" name="address"
placeholder="" required>
      </fieldset>
      <fieldset class="form-group">
        <label>No Telepon Klinik</label>
        <input class="form-control" id="klinik-
address" type="text" name="klinik_phone"
placeholder="" required>
      </fieldset>
    </div>
    <div class="col-lg-5">
      <fieldset class="form-group">
        <label class=" align-items-center">
          <center> Input Jenis Pengobatan
          Yang tersedia</center>
        </label>
      </fieldset>
    </div>
  </div>
  <div class="row">
    <div class="col-md-6">
      Pengobatan

```

```

        <input class="form-control"
id="therapy-1" type="text" name="therapy[0]"
placeholder="" value="akupuntur"><br>
        <input class="form-control"
id="therapy-2" type="text" name="therapy[1]"
placeholder="" value="pijat"><br>
        <input class="form-control"
id="therapy-3" type="text" name="therapy[2]"
placeholder="" value="bekam">
      </div>
    <div class="col-md-6">
      Harga
      <input class="form-control"
id="price-1" type="text" name="price[0]"
placeholder=""><br>
      <input class="form-control"
id="price-2" type="text" name="price[1]"
placeholder=""><br>
      <input class="form-control"
id="price-3" type="text" name="price[2]"
placeholder=""><br>
    </div>
  </div>
  </fieldset>
  <fieldset class="form-group">
    <label>Input Jumlah Therapist</label>
    <input class="form-control" id="klinik-
therapiest" type="number" name="therapiest"
placeholder="" required>
  </fieldset>
  <fieldset class="form-group">
    <label>Input Jadwal Buka Tutup ( ex:
Senin-Jumat )</label>
    <input class="form-control" id="klinik-
open-close" type="text" name="open_close"
placeholder="" required>
  </fieldset>
  <fieldset class="form-group">
    <label>No Pembagian Waktu Perhari (
ex: 08:00 am to 09:00 pm ) </label>
    <input class="form-control" id="klinik-
time-per-day" type="text" name="time_per_day"
placeholder="" required>
  </fieldset>
</div>

</div>
<div class="row">
  <div class="col-5">
    <hr>
    <p>Field Untuk Login Klinik</p>
    <fieldset class="form-group">
      <label>Email</label>
      <input class="form-control" id="klinik-
address" type="text" name="email"
placeholder="" required>
    </fieldset>
    <fieldset class="form-group">
      <label>Password</label>

```

```

        <input class="form-control" id="klinik-
address" type="password" name="password"
placeholder="" required>
    </fieldset>
    <span class="pull-right">
        <button type="submit" class="btn btn-
primary">Simpan</button>
    </span>
</div>
<div class="col-lg-5 col-md-5">
    <div class="card">
        <div class="card-body">
            <h4 class="text-black">Upload Foto
Klinik</h4>
            <label for="input-file-max-fs">Your so
fresh input file</label>
            <input type="file" class="image dropify"
id="fileInput" accept="image/*"
name="foto_klinik" required data-max-file-
size="2M"/>
            <input type="hidden"
name="iconMerk" id="iconMerk" />
        </div>
    </div>
</div>
</div>
</form>

```

```

<div class="modal fade" id="modal" tabindex="-
1" role="dialog" aria-labelledby="modalLabel"
aria-hidden="true">
    <div class="modal-dialog modal-lg"
role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title"
id="modalLabel">Crop Your Merk</h5>
                <button type="button" class="close"
data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">×</span>
                </button>
            </div>
            <div class="modal-body">
                <div class="img-container">
                    <div class="row">
                        <div class="col-md-8">
                            
                        </div>
                        <div class="col-md-4">
                            <div class="preview"></div>
                        </div>
                    </div>
                </div>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-
secondary" data-
dismiss="modal">Cancel</button>
            </div>
        </div>
    </div>
</div>

```

```

        <button type="button" class="btn btn-
primary" id="crop">Crop</button>
    </div>
</div>
</div>

```

@endsection

@section('js')

```

<script
src="{ {URL::asset('backend')}}/plugins/dropify/
dropify.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/croppe
rjs/1.5.6/cropper.js" integrity="sha256-
CgvH7sz3tHhkiVKh05kSUG97YtzYNnWt6OX
cmYzqHY="
crossorigin="anonymous"></script>

```

```

<script>
$(document).ready(function () {
    $('dropify').dropify();
    var $modal = $('#modal');
    var iconMerk = $('#iconMerk');
    var image =
document.getElementById('image');
    var cropper;

    $("body").on("change", ".image",
function (e) {
        var files = e.target.files;
        var done = function (url) {
            image.src = url;
            $modal.modal('show');
        };
        var reader;
        var file;
        var url;

        if (files && files.length > 0) {
            file = files[0];

            if (URL) {
                done(URL.createObjectURL(file))
            }
            ;

            } else if (FileReader) {
                reader = new FileReader();
                reader.onload = function (e) {
                    done(reader.result);
                };
                reader.readAsDataURL(file);
            }
        }
    });

    $modal.on('shown.bs.modal', function () {
        cropper = new Cropper(image, {
            aspectRatio: 16 / 16,
            viewMode: 3,
            preview: '.preview'

```

```

    });
    }).on('hidden.bs.modal', function () {
        cropper.destroy();
        cropper = null;
    });

    $("#crop").click(function () {
        canvas = cropper.getCroppedCanvas({
            width: 200,
            height: 200,
        });

        canvas.toBlob(function (blob) {
            url = URL.createObjectURL(blob);
            var reader = new FileReader();
            reader.readAsDataURL(blob);
            reader.onloadend = function () {
                var base64data = reader.result;
                iconMerk.val(base64data);
                $modal.modal('hide');
            }
        });
    });
});
</script>
@endsection

```

Klinik (admin.blade.php)

```

@extends('layouts.backend')
@section('css')
<link href="/css/tabs.css" rel="stylesheet">
@endsection
@section('content')
<div class="content">
    <!-- Small boxes (Stat box) -->
    <div class="row">
        <div class="col-lg-4 col-sm-6 col-xs-12">
            <div class="info-box bg-darkblue"> <span
class="info-box-icon bg-transparent"><i
class="ti-stats-up text-white"></i></span>
            <div class="info-box-content">
                <h6 class="info-box-text text-
white">Pasien Hari Ini</h6>
                <h1 class="text-white">{{ $today }}</h1>
            <!-- /.info-box-content -->
            </div>
            <!-- /.info-box -->
        </div>
        <!-- /.col -->
        <div class="col-lg-4 col-sm-6 col-xs-12">
            <div class="info-box bg-green text-white">
<span class="info-box-icon bg-transparent"><i
class="ti-face-smile"></i></span>
            <div class="info-box-content">
                <h6 class="info-box-text text-white">Total
Pasien</h6>
                <h1 class="text-white">{{ $pasien }}</h1>
            <!-- /.info-box-content -->
            </div>
            <!-- /.info-box -->

```

```

    </div>
    </div>
    <!-- /.col -->
    <div class="col-lg-4 col-sm-6 col-xs-12">
        <div class="info-box bg-orange"> <span
class="info-box-icon bg-transparent"><i
class="ti-money"></i></span>
        <div class="info-box-content">
            <h6 class="info-box-text text-white">Total
Transaksi</h6>
            <h1 class="text-white">Rp.
{{ number_format($transaksi) }}</h1>
        <!-- /.info-box-content -->
        </div>
        <!-- /.info-box -->
    </div>
    <!-- /.col -->
    </div>
    <div class="row">
        <div class="col-md-12">
            <div class="card">
                <div class="card-body p-b-0">
                    <h4 class="card-title text-black text-
center">Transaksi History</h4>
                </div>
                <!-- Nav tabs -->
                <ul class="nav nav-tabs customtab"
role="tablist" style="display: flex; margin:
auto;">
                    <li class="nav-item"> <a class="nav-link"
data-toggle="tab" href="#yesterday"
role="tab"><span class="hidden-sm-up"><i
class="ti-home"></i></span> <span
class="hidden-xs-down">Transaksi
Kemarin</span></a> </li>
                    <li class="nav-item"> <a class="nav-link
active" data-toggle="tab" href="#today"
role="tab"><span class="hidden-sm-up"><i
class="ti-user"></i></span> <span
class="hidden-xs-down">Transaksi Hari
Ini</span></a> </li>
                    <li class="nav-item"> <a class="nav-link"
data-toggle="tab" href="#tomorrow"
role="tab"><span class="hidden-sm-up"><i
class="ti-email"></i></span> <span
class="hidden-xs-down">Transaksi
Besok</span></a> </li>
                </ul>
                <!-- Tab panes -->
                <div class="tab-content">
                    <div class="tab-pane p-20" id="yesterday"
role="tabpanel">
                        <div class="table-responsive">
                            <table class="table table-hover">
                                <thead>
                                    <tr>
                                        <th scope="col">#</th>
                                        <th scope="col">Nama
Pasien</th>
                                        <th scope="col">Nama
Klinik</th>

```

```

                <th scope="col">Tanggal</th>
                <th scope="col">Jam</th>
                <th scope="col">Status</th>
            </tr>
        </thead>
        <tbody>
            @foreach ($solds as $sold)
            <tr>
                <th scope="row">{{ $loop-
>iteration }}</th>
                <td><a href="{{ {
route('admin.detail', $sold->id) }}">{{ $sold-
>name }}</a></td>
                <td>{{ $sold-
>klinik_name }}</td>
                <td>{{ $sold-
>tanggal_pengobatan }}</td>
                <td>{{ $sold-
>jam_pengobatan }}</td>
                <td>
                    @if ($sold->bukti_transfer)
                        Review Bukti TF
                    @else
                        Belum Ada Bukti TF
                    @endif
                </td>
            </tr>
            @endforeach
        </tbody>
    </table>
</div>
</div>
<div class="tab-pane active" id="today"
role="tabpanel">
    <div class="table-responsive">
        <table class="table table-hover">
            <thead>
                <tr>
                    <th scope="col">#</th>
                    <th scope="col">Nama
Pasien</th>
                    <th scope="col">Nama
Klinik</th>
                    <th scope="col">Tanggal</th>
                    <th scope="col">Jam</th>
                    <th scope="col">Status</th>
                </tr>
            </thead>
            <tbody>
                @foreach ($days as $sold)
                <tr>
                    <th scope="row">{{ $loop-
>iteration }}</th>
                    <td><a href="{{ {
route('admin.detail', $sold->id) }}">{{ $sold-
>name }}</a></td>
                    <td>{{ $sold-
>klinik_name }}</td>
                    <td>{{ $sold-
>tanggal_pengobatan }}</td>
                    <td>{{ $sold-
>jam_pengobatan }}</td>

```

```

                <td>
                    @if ($sold->bukti_transfer)
                        Review Bukti TF
                    @else
                        Belum Ada Bukti TF
                    @endif
                </td>
            </tr>
            @endforeach
        </tbody>
    </table>
</div>
</div>
<div class="tab-pane p-20" id="tomorrow"
role="tabpanel">
    <div class="table-responsive">
        <table class="table table-hover">
            <thead>
                <tr>
                    <th scope="col">#</th>
                    <th scope="col">Nama
Pasien</th>
                    <th scope="col">Nama
Klinik</th>
                    <th scope="col">Tanggal</th>
                    <th scope="col">Jam</th>
                    <th scope="col">Status</th>
                </tr>
            </thead>
            <tbody>
                @foreach ($soons as $sold)
                <tr>
                    <th scope="row">{{ $loop-
>iteration }}</th>
                    <td><a href="{{ {
route('admin.detail', $sold->id) }}">{{ $sold-
>name }}</a></td>
                    <td>{{ $sold-
>klinik_name }}</td>
                    <td>{{ $sold-
>tanggal_pengobatan }}</td>
                    <td>{{ $sold-
>jam_pengobatan }}</td>
                    <td>
                        @if ($sold->bukti_transfer)
                            Review Bukti TF
                        @else
                            Belum Ada Bukti TF
                        @endif
                    </td>
                </tr>
                @endforeach
            </tbody>
        </table>
    </div>
</div>
</div>
</div>
@endsection

```

Klinik (create.blade.php)

```
@extends('layouts.backend')
@section('css')
<link rel="stylesheet"
href="/backend/plugins/formwizard/jquery-
steps.css">
<link rel="stylesheet"
href="{{ URL::asset('backend') }}/plugins/dropify-
/dropify.min.css">
<style>
    img {
        display: block;
        max-width: 100%;
    }

    .preview {
        overflow: hidden;
        width: 160px;
        height: 160px;
        margin: 10px;
        border: 1px solid red;
    }

    .modal-lg {
        max-width: 1000px !important;
    }
</style>
@endsection
@section('content')
<div class="content">
    <div class="info-box">
        <h4 class="text-black m-b-3">Step
        wizard</h4>
        <div id="demo">
            <div class="step-app">
                <ul class="step-steps">
                    <li><a href="#step1"><span
                    class="number">1</span> Klinik Info</a></li>
                    <li><a href="#step2"><span
                    class="number">2</span> Login
                    Information</a></li>
                    <li><a href="#step3"><span
                    class="number">3</span> Clinics
                    Therapy</a></li>
                </ul>
                <div class="step-content">
                    <div class="step-tab-panel" id="step1">
                        <form method="POST"
                        action="{ {route('klinik-save')}}" id="my_form"
                        enctype="multipart/form-data">@csrf
                            <div class="row m-t-2">
                                <div class="col-md-12">
                                    <div class="card">
                                        <div class="card-body">
                                            <h4 class="text-black">Upload
                                            Foto Klinik</h4>
                                            <label for="input-file-max-
                                            fs">Your so fresh input file</label>
                                            <input type="file" class="image
                                            dropify" id="fileInput" accept="image/*"
```

```
name="foto_klinik" required data-max-file-
size="2M"/>
                                <input type="hidden"
                                name="iconMerk" id="iconMerk" />
                            </div>
                        </div>
                    </div>
                    <div class="row m-t-2">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Nama Klinik</label>
                                    <input class="form-control"
                                    id="klinik-name" type="text"
                                    name="klinik_name" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Nama Pemilik</label>
                                    <input class="form-control"
                                    id="klinik-owner" type="text"
                                    name="owner_name" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                    <div class="row">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>No Ho Pemilik</label>
                                    <input class="form-control"
                                    id="klinik-owner-phone" type="text"
                                    name="owner_phone" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>No Telepon
                                    Klinik</label>
                                    <input class="form-control"
                                    id="klinik-address" type="text"
                                    name="klinik_phone" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                    <div class="row">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Izin Usaha</label>
                                    <input class="form-control"
                                    id="klinik-permission" type="text"
                                    name="permission" placeholder="" required>
```

```

        </fieldset>
    </div>
</div>
<div class="col-md-6">
    <div class="form-group">
        <fieldset class="form-group">
            <label>Alamat</label>
            <input class="form-control"
id="klinik-address" type="text" name="address"
placeholder="" required>
        </fieldset>
    </div>
</div>
</div>
<div class="row">
    <div class="col-md-4">
        <div class="form-group">
            <fieldset class="form-group">
                <label>Input Jumlah
Therapist</label>
                <input class="form-control"
id="klinik-therapiest" type="number"
name="therapiest" placeholder="" required>
            </fieldset>
        </div>
</div>
<div class="col-md-4">
        <div class="form-group">
            <fieldset class="form-group">
                <label>Input Jadwal Buka
Tutup</label>
                <input class="form-control"
id="klinik-open-close" type="text"
name="open_close" placeholder="" required>
            </fieldset>
        </div>
</div>
<div class="col-md-4">
        <div class="form-group">
            <fieldset class="form-group">
                <label>No Pembagian Waktu
Perhari </label>
                <input class="form-control"
id="klinik-time-per-day" max="13" type="text"
name="time_per_day" placeholder="" required>
            </fieldset>
        </div>
</div>
</div>
<div class="row">
    <div class="col-md-6">
        <div class="form-group">
            <fieldset class="form-group">
                <label>Latitude</label>
                <input class="form-control"
id="klinik-permission" type="text"
name="latitude" placeholder="" required>
            </fieldset>
        </div>
</div>
</div>
<div class="col-md-6">
    <div class="form-group">

```

```

        <fieldset class="form-group">
            <label>Longitude</label>
            <input class="form-control"
id="klinik-address" type="text"
name="longitude" placeholder="" required>
        </fieldset>
    </div>
</div>
</div>
<div class="step-tab-panel" id="step2">
    <div class="row m-t-2">
        <div class="col-md-6">
            <div class="form-group">
                <fieldset class="form-group">
                    <label>Email</label>
                    <input class="form-control"
id="klinik-address" type="text" name="email"
placeholder="" required>
                </fieldset>
            </div>
</div>
<div class="col-md-6">
            <div class="form-group">
                <fieldset class="form-group">
                    <label>Password</label>
                    <input class="form-control"
id="klinik-address" type="password"
name="password" placeholder="" required>
                </fieldset>
            </div>
</div>
</div>
<div class="step-tab-panel" id="step3">
    <div class="row m-t-2">
        <div class="col-md-6">
            <div class="form-group">
                <label for="int1">Pengobatan
:</label>
                <select name="therapy[]"
id="therapy" class="form-control">
                    @foreach ($terapi as $terapi)
                        <option
value="{{ strtolower($terapi-
>name)}}">{{ $terapi->name }}</option>
                    @endforeach
                </select><br>
                <select name="therapy[]"
id="therapy" class="form-control">
                    @foreach ($terapi as $terapi)
                        <option
value="{{ strtolower($terapi-
>name)}}">{{ $terapi->name }}</option>
                    @endforeach
                </select><br>
                <select name="therapy[]"
id="therapy" class="form-control">
                    @foreach ($terapi as $terapi)
                        <option
value="{{ strtolower($terapi-
>name)}}">{{ $terapi->name }}</option>

```



```

<link rel="stylesheet"
href="/backend/plugins/formwizard/jquery-
steps.css">
<link rel="stylesheet"
href="{ {URL::asset('backend')}}/plugins/dropify
/dropify.min.css">
<style>
    img {
        display: block;
        max-width: 100%;
    }

    .preview {
        overflow: hidden;
        width: 160px;
        height: 160px;
        margin: 10px;
        border: 1px solid red;
    }

    .modal-lg {
        max-width: 1000px !important;
    }
</style>
@endsection
@section('content')
<div class="content">
    <div class="info-box">
        <h4 class="text-black m-b-3">Step
wizard</h4>
        <div id="demo">
            <div class="step-app">
                <ul class="step-steps">
                    <li><a href="#step1"><span
class="number">1</span> Klinik Info</a></li>
                    <li><a href="#step3"><span
class="number">3</span> Clinics
Therapy</a></li>
                </ul>
                <div class="step-content">
                    <div class="step-tab-panel" id="step1">
                        <form method="POST"
action="{ {route('klinik.update',$klinik->id)} }"
id="my_form" enctype="multipart/form-
data">@csrf
                            <input type="hidden"
value="{ { $klinik->user_id } }" name="user_id"
id="">
                                <div class="row m-t-2">
                                    <div class="col-md-12">
                                        <div class="card">
                                            <div class="card-body">
                                                <h4 class="text-black">Upload
Foto Klinik</h4>
                                                <label for="input-file-max-
fs">Your so fresh input file</label>
                                                <input type="file" class="image
dropify" id="fileInput" accept="image/*"
name="foto_klinik" required data-max-file-
size="2M"/>
                                                <input type="hidden"
name="iconMerk" id="iconMerk" />

```

```

                                </div>
                            </div>
                        </div>
                    </div>
                    <div class="row m-t-2">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Nama Klinik</label>
                                    <input class="form-control"
id="klinik-name" type="text" value="{ { $klinik-
>klinik_name } }" name="klinik_name"
placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Nama Pemilik</label>
                                    <input class="form-control"
id="klinik-owner" type="text" value="{ { $klinik-
>name } }" name="owner_name" placeholder=""
required>
                                </fieldset>
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>No HP Pemilik</label>
                                    <input class="form-control"
id="klinik-owner-phone" value="{ { $klinik-
>klinik_owner_phone } }" type="text"
name="owner_phone" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>No Telepon
Klinik</label>
                                    <input class="form-control"
id="klinik-address" type="text"
value="{ { $klinik->klinik_phone } }"
name="klinik_phone" placeholder="" required>
                                </fieldset>
                            </div>
                        </div>
                    </div>
                    <div class="row">
                        <div class="col-md-6">
                            <div class="form-group">
                                <fieldset class="form-group">
                                    <label>Izin Usaha</label>
                                    <input class="form-control"
id="klinik-permission" type="text"

```



```

value="{{ $klinik->klinik_permission }}"
name="permission" placeholder="" required>
</fieldset>
</div>
</div>
<div class="col-md-6">
<div class="form-group">
<fieldset class="form-group">
<label>Alamat</label>
<input class="form-control"
id="klinik-address" type="text" name="address"
value="{{ $klinik->klinik_address }}"
placeholder="" required>
</fieldset>
</div>
</div>
</div>
<div class="row">
<div class="col-md-4">
<div class="form-group">
<fieldset class="form-group">
<label>Input Jumlah
Therapist</label>
<input class="form-control"
id="klinik-therapiest" type="number"
name="therapiest" value="{{ $klinik-
>klinik_therapist }}" placeholder="" required>
</fieldset>
</div>
</div>
<div class="col-md-4">
<div class="form-group">
<fieldset class="form-group">
<label>Input Jadwal Buka
Tutup</label>
<input class="form-control"
id="klinik-open-close" type="text"
name="open_close" value="{{ $klinik-
>klinik_open_close }}" placeholder="" required>
</fieldset>
</div>
</div>
<div class="col-md-4">
<div class="form-group">
<fieldset class="form-group">
<label>No Pembagian Waktu
Perhari </label>
<input class="form-control"
id="klinik-time-per-day" max="13" type="text"
value="{{ $klinik->klinik_time_per_day }}"
name="time_per_day" placeholder="" required>
</fieldset>
</div>
</div>
</div>
<div class="row">
<div class="col-md-6">
<div class="form-group">
<fieldset class="form-group">
<label>Latitude</label>
<input class="form-control"
id="klinik-permission" type="text"

```

```

value="{{ $klinik->latitude }}" name="latitude"
placeholder="" required>
</fieldset>
</div>
</div>
<div class="col-md-6">
<div class="form-group">
<fieldset class="form-group">
<label>Longitude</label>
<input class="form-control"
id="klinik-address" type="text"
value="{{ $klinik->longitude }}"
name="longitude" placeholder="" required>
</fieldset>
</div>
</div>
</div>
<div class="step-tab-panel" id="step3">
<div class="row m-t-2">
<div class="col-md-6">
<div class="form-group">
<label for="int1">Pengobatan
:</label>
<select name="therapy[]"
id="therapy" class="form-control">
@foreach ($terapis as $terapi)
<option
value="{{ strtolower($terapi-
>name) }}">{{ $terapi->name }}</option>
@endforeach
</select><br>
<select name="therapy[]"
id="therapy" class="form-control">
@foreach ($terapis as $terapi)
<option
value="{{ strtolower($terapi-
>name) }}">{{ $terapi->name }}</option>
@endforeach
</select><br>
<select name="therapy[]"
id="therapy" class="form-control">
@foreach ($terapis as $terapi)
<option
value="{{ strtolower($terapi-
>name) }}">{{ $terapi->name }}</option>
@endforeach
</select><br>
</div>
</div>
<div class="col-md-6">
<div class="form-group">
<label for="intType1">Harga
Pengobatan :</label>
<input class="form-control"
id="price-1" type="text" name="price[]"
placeholder=""><br>
<input class="form-control"
id="price-2" type="text" name="price[]"
placeholder=""><br>

```

```

        <input class="form-control"
id="price-3" type="text" name="price[]"
placeholder=""><br>
        </div>
    </div>
</div>
</div>
</div>
</form>
<div class="step-footer">
    <button data-direction="prev" class="btn
btn-light">Previous</button>
    <button data-direction="next" class="btn
btn-primary">Next</button>
    <button data-direction="finish"
class="btn btn-primary">Submit</button>
</div>
</div>
</div>
</div>
<!-- Main row -->
</div>
@endsection
@section('js')
<script
src="/backend/plugins/formwizard/jquery-
steps.js"></script>
<script
src="{ { URL::asset('backend') } }/plugins/dropify/
dropify.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/cropper
rjs/1.5.6/cropper.js" integrity="sha256-
CgvH7sz3tHhkiVKh05kSUG97YtzYNnWt6OX
cmYzqHY="
crossorigin="anonymous"></script>
<script>
$(document).ready(function () {
    $('#demo').steps({
        onFinish: function () {
            $('#my_form').submit()
        }
    });
    $('.dropify').dropify();
});
</script>
@endsection

```

```

        <h4 class="modal-title"
id="vcenter">Hapus Event</h4>
        <button type="button" class="close"
data-dismiss="modal" aria-
hidden="true">×</button>
    </div>
    <div class="modal-body">
        <p>Anda yakin ingin menghapus data
ini?</p>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-
default" data-dismiss="modal">Cancel</button>
        <button type="button" class="btn btn-
danger btn-ok">Delete</button>
    </div>
</div>
</div>
</div>
@endsection
@section('js')
<script>
    $('#confirm-delete').on('click', '.btn-ok',
function(e) {
        var $modalDiv = $(e.delegateTarget);
        var id = $(this).data('recordId');
        $.ajaxSetup({
            headers: {
                'X-CSRF-TOKEN':
$('meta[name="csrf-token"]').attr('content')
            }
        });

        $.post('/admin/delete/klinik/' + id).then()
        $modalDiv.addClass('loading');
        setTimeout(function() {
            location.reload();
        })
    });
    $('#confirm-delete').on('show.bs.modal',
function(e) {
        var data = $(e.relatedTarget).data();
        $('.btn-ok', this).data('recordId',
data.recordId);
    });
</script>
@endsection

```

Pasien (index.blade.php)

```

@extends('layouts.backend')
@section('title', 'Pasien')
@section('content')
<div class="row">
    <div class="col-12">
        <h4 class="text-black">List</h4>
    </div>
</div>
<div class="row">
    <div class="col-12">
        <div class="table-responsive">
            <table class="table table-hover">

```

```

                <thead>
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Name</th>
                        <th scope="col">Email</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach ($pasien as $pasien)
                        <tr>
                            <th scope="row">{ { $loop-
>iteration } }</th>
                            <td>{ { $pasien->name } }</td>
                            <td>{ { $pasien->email } }</td>
                        </tr>
                    @endforeach
                </tbody>
            </table>
        </div>
    </div>
</div>
@endsection

```

Terapi (index.blade.php)

```

@extends('layouts.backend')
@section('title', 'Terapi')
@section('content')
<div class="row">
    <div class="col-12">
        <h4 class="text-black">List Terapi</h4>
        <span class="m-3 pull-right">
            <a data-toggle="modal" href="#" data-
target="#exampleModal" class="btn btn-rounded
btn-success">Tambah</a>
        </span>
    </div>
</div>
<div class="row">
    <div class="col-12">
        <div class="table-responsive">
            <table class="table table-hover">
                <thead>
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Name</th>
                        <th scope="col">Gambar</th>
                        <th scope="col">Action</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach ($terapi as
$key=>$terapi)
                        <tr>
                            <th scope="row">{ { $loop-
>iteration } }</th>
                            <td>{ { $terapi->name } }</td>
                            <td></td>

```

```

                <td>
                    <a href="#"><button
type="button" class="btn btn-secondary" data-
toggle="modal" data-target="#update-
terapi{{ $key }}"><i class="fa fa-
pencil"></i></button></a>
                    <a href="#"><button
type="button" data-record-id="{{ $terapi->id }}"
data-toggle="modal" data-target="#confirm-
delete" class="btn btn-secondary"><i class="fa
fa-trash"></i></button></a>
                </td>
            </tr>
        @endforeach
    </tbody>
</table>
</div>
</div>
</div>

<div class="modal fade" id="exampleModal"
tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-
hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title"
id="exampleModalLabel">Tambah Terapi</h5>
                <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
                    <span aria-
hidden="true">&times;</span>
            </button>
            </div>
            <div class="modal-body">
                <form action="{{ route('terapi.store') }}"
method="POST" enctype="multipart/form-
data">@csrf
                    <div class="form-group">
                        <label
for="exampleInputEmail1">Jenis Terapi</label>
                        <input type="text" name="terapi"
class="form-control" id="exampleInputEmail1"
placeholder="Jenis Terapi">
                    </div>
                    <div class="form-group">
                        <label
for="exampleInputPassword1">Image</label>
                        <input type="file" name="image"
class="form-control"
id="exampleInputPassword1">
                    </div>
                    <button type="submit" class="btn btn-
primary">Submit</button>
                </form>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-
secondary" data-
dismiss="modal">Close</button>
            </div>
        </div>
    </div>
</div>

```

```

        </div>
    </div>
</div>
@foreach ($terapis as $key=>$terapi)
<div class="modal fade" id="update-
terapi{{ $key }}" tabindex="-1" role="dialog"
aria-hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title"
id="exampleModalLabel">Edit Terapi</h5>
                <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
            </button>
            </div>
            <div class="modal-body">
                <form action="{{
route('terapi.update',$terapi->id) }}"
method="POST" enctype="multipart/form-
data">@csrf
                    <div class="form-group">
                        <label
for="exampleInputEmail1">Jenis Terapi</label>
                        <input type="text" name="terapi"
value="{{ $terapi->name }}" class="form-control"
id="exampleInputEmail1" placeholder="Jenis
Terapi">
                    </div>
                    <div class="form-group">
                        <label
for="exampleInputPassword1">Image</label>
                        <input type="file" name="image"
class="form-control"
id="exampleInputPassword1">
                    </div>
                    <button type="submit" class="btn btn-
primary">Submit</button>
                </form>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-
secondary" data-
dismiss="modal">Close</button>
            </div>
        </div>
    </div>
</div>
@endforeach

<div class="modal fade" id="confirm-delete"
tabindex="-1" role="dialog" aria-
labelledby="myModalLabel" aria-hidden="true">
    <meta name="csrf-token" content="{{
csrf_token() }}" />
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <h4 class="modal-title"
id="vcenter">Hapus Event</h4>

```

```

        <button type="button" class="close"
data-dismiss="modal" aria-
hidden="true"></button>
    </div>
    <div class="modal-body">
        <p>Anda yakin ingin menghapus data
ini?</p>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-
default" data-dismiss="modal">Cancel</button>
        <button type="button" class="btn btn-
danger btn-ok">Delete</button>
    </div>
</div>
</div>
@endsection
@section('js')
<script>
    $('#confirm-delete').on('click', '.btn-ok',
function(e) {
        var $modalDiv = $(e.delegateTarget);
        var id = $(this).data('recordId');
        $.ajaxSetup({
            headers: {
                'X-CSRF-TOKEN':
$(meta[name="csrf-token"]).attr('content')
            }
        });

        $.post('/admin/terapi-delete/' + id).then()
        $modalDiv.addClass('loading');
        setTimeout(function() {
            location.reload();
        })
    });
    $('#confirm-delete').on('show.bs.modal',
function(e) {
        var data = $(e.relatedTarget).data();
        $('#btn-ok', this).data('recordId',
data.recordId);
    });
</script>
</div>

```

Transaksi (detail.blade.php)

```

@extends('layouts.backend')
@section('content')
<div class="content-header sty-one">
    <h1>Transaksi</h1>
    <ol class="breadcrumb">
        <li><a href="{ { route('admin.transaksi')
}}">Transaksi</a></li>
        <li><i class="fa fa-angle-right"></i>
Transaksi Detail</li>
    </ol>
</div>
<div class="content">
    <div class="row">
        <div class="col-md-12">

```

```

<div class="card">
    @if ($order->bukti_transfer)
        
    @else
        <h1 style="text-align: center">Belum ada
Bukti Transaksi</h1>
    @endif
    <table class="table">
        <tr>
            <td>Nama Pasien</td>
            <td>:</td>
            <td>{ { $order->name } }</td>
        </tr>
        <tr>
            <td>Nama Klinik</td>
            <td>:</td>
            <td>{ { $order->klinik_name } }</td>
        </tr>
        <tr>
            <td>Jenis Therapy</td>
            <td>:</td>
            <td>{ { $order-
>therapy_name } }</td>
        </tr>
        <tr>
            <td>Tanggal</td>
            <td>:</td>
            <td>{ { $order-
>tanggal_pengobatan } }</td>
        </tr>
        <tr>
            <td>Jam</td>
            <td>:</td>
            <td>{ { $order-
>jam_pengobatan } }</td>
        </tr>
        <tr>
            <td>No Urut</td>
            <td>:</td>
            <td>{ { $order->no_urut } }</td>
        </tr>
        @if (Auth::user()->role=='ADMIN')
            <form action="{ {
route('admin.transaksi.confirm',$order->id) } }"
method="POST">@csrf
            <tr>
                <td>
                    <select name="review"
id="review" class="form-control">
                        <option value="Konfirmasi
Pembayaran">Konfirmasi Pembayaran</option>
                        <option value="Pembayaran
Gagal">Pembayaran Gagal</option>
                    </select>
                </td>
                <td>
                    <textarea name="pesan"
id="pesan" placeholder="Catatan Transaksi"
class="form-control"></textarea>

```

```

        </td>
        <td>
            <button class="btn btn-primary"
type="submit">Konfirmasi</button>
        </td>
    </tr>
</form>
@endif
</table>
</div>
</div>
</div>
</div>
@endsection

```

Transaksi (index.blade.php)

```

@extends('layouts.backend')
@section('title', 'order')
@section('content')
<div class="row">
    <div class="col-12">
        <h4 class="text-black">List Transaksi</h4>
    </div>
</div>
<div class="row">
    <div class="col-12">
        <div class="table-responsive">
            <table class="table table-hover">
                <thead>
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Nama
Pasien</th>
                        <th scope="col">Nama
Klinik</th>
                        <th scope="col">Tanggal</th>
                        <th scope="col">Jam</th>
                        <th scope="col">Status</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach ($orders as $order)
                    <tr>
                        <th scope="row">{{ $loop-
>iteration }}</th>
                        <td><a href="{{ {
route('admin.detail', $order->id) }}">{{ $order-
>name }}</a></td>
                        <td>{{ $order-
>klinik_name }}</td>
                        <td>{{ $order-
>tanggal_pengobatan }}</td>
                        <td>{{ $order-
>jam_pengobatan }}</td>
                        <td>{{ $order-
>total_payment }}</td>
                    </tr>
                    @endforeach
                </tbody>
            </table>

```

```

        </div>
    </div>
</div>
@endsection

```

App.blade.php

```

<!doctype html>
<html lang="{{ { str_replace('_', '-', app()-
>getLocale()) }}">

<head>
    <meta charset="utf-8">
    <meta name="viewport"
content="width=device-width, initial-scale=1,
shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="csrf-token" content="{{ {
csrf_token() }}">

    <meta name="author" content="Mark Otto,
Jacob Thornton, and Bootstrap contributors">
    <meta name="generator" content="Hugo
0.80.0">
    <title>{{ { config('app.name', 'Medis Reborn')
}}</title>

    <link rel="stylesheet"
href="{{ { URL::asset('backend') }} /bootstrap/css/b
ootstrap.min.css">
    <link rel="stylesheet"
href="{{ { URL::asset('backend') }} /css/font-
awesome/css/font-awesome.min.css">
    <link rel="preconnect"
href="https://fonts.googleapis.com">
    <link rel="preconnect"
href="https://fonts.gstatic.com" crossorigin>
    <link
href="https://fonts.googleapis.com/css2?family=
Roboto&display=swap" rel="stylesheet">
    <style>
        body{
            font-family: 'Roboto',
        }
    </style>
    @yield('css')

<!-- Bootstrap core CSS -->

<style>
    .bd-placeholder-img {
        font-size: 1.125rem;
        text-align: middle;
        -webkit-user-select: none;
        -moz-user-select: none;
        -ms-user-select: none;
        user-select: none;
    }

```

```

        @media (min-width: 768px) {
            .bd-placeholder-img-lg {
                font-size: 3.5rem;
            }
        }
        .container {
            max-width: 960px;
        }
        html {
            font-size: 14px;
        }
        @media (min-width: 768px) {
            html {
                font-size: 16px;
            }
        }

        .shadow-sm{box-shadow:0 .125rem .25rem
        rgba(0,0,0,.075)!important}

</style>

<!-- Custom styles for this template -->
<link href="{{ asset('css/app.css') }}"
rel="stylesheet">

</head>

<body>
    @php
        $terapis=\App\Terapi::all();
        // dd($terapis);
    @endphp

    <main role="main" class="mt-5">
        @yield('content')

    </main>
    <!-- jQuery 3 -->
    <script
    src="{{ URL::asset('backend') }}/js/jquery.min.js"
    ></script>

    <!-- v4.0.0-alpha.6 -->
    <script
    src="{{ URL::asset('backend') }}/bootstrap/js/boot
    strap.min.js"></script>

    @yield('js')

</body>

</html>
Backend.blade.php

!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">

```

```

    <meta http-equiv="X-UA-Compatible"
    content="IE=edge">
    <title>Medic Reborn - @yield('title')</title>
    <!-- Tell the browser to be responsive to screen
    width -->
    <meta name="viewport"
    content="width=device-width, minimum-scale=1,
    maximum-scale=1" />

    <!-- v4.0.0-alpha.6 -->
    <link rel="stylesheet"
    href="{{ URL::asset('backend') }}/bootstrap/css/b
    ootstrap.min.css">

    <!-- Google Font -->
    <link
    href="https://fonts.googleapis.com/css?family=P
    oppins:300,400,500,600,700" rel="stylesheet">

    <!-- Theme style -->
    <link rel="stylesheet"
    href="{{ URL::asset('backend') }}/css/style.css">
    <link rel="stylesheet"
    href="{{ URL::asset('backend') }}/css/font-
    awesome/css/font-awesome.min.css">
    <link rel="stylesheet"
    href="{{ URL::asset('backend') }}/css/et-line-
    font/et-line-font.css">
    <link rel="stylesheet"
    href="{{ URL::asset('backend') }}/css/themify-
    icons/themify-icons.css">
    @yield('css')
    <!-- HTML5 Shim and Respond.js IE8 support
    of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if
    you view the page via file:// -->
    <!--[if lt IE 9]>
    <script
    src="https://oss.maxcdn.com/html5shiv/3.7.3/htm
    l5shiv.min.js"></script>
    src="https://oss.maxcdn.com/respond/1.4.2/respo
    nd.min.js"></script>
    <![endif]>

</head>

<body class="skin-blue sidebar-mini">
    <div class="wrapper boxed-wrapper">
        <header class="main-header">
            <!-- Logo -->
            <a href="{{ route('admin') }}"
            class="logo blue-bg">
                <!-- mini logo for sidebar mini 50x50
                pixels -->
                <span class="logo-mini"></span>
                <!-- logo for regular state and mobile
                devices -->
                <span class="logo-lg"></span>

```

```

</a>
<!-- Header Navbar: style can be found in
header.less -->
<nav class="navbar blue-bg navbar-static-
top">
  <!-- Sidebar toggle button-->
  <ul class="nav navbar-nav pull-left">
    <li><a class="sidebar-toggle" data-
toggle="push-menu" href=""></a> </li>
  </ul>
  <div class="pull-left search-box">
    <form action="#" method="get"
class="search-form">
      <div class="input-group">
        <input name="search"
class="form-control" placeholder="Search..."
type="text">
        <span class="input-group-btn">
          <button type="submit"
name="search" id="search-btn" class="btn btn-
flat"><i class="fa fa-search"></i> </button>
        </span>
      </div>
    </form>
    <!-- search form -->
  </div>
  <div class="navbar-custom-menu">
    <ul class="nav navbar-nav">
      <!-- User Account: style can be
found in dropdown.less -->
      <li class="dropdown user user-
menu p-ph-res"><a href="#" class="dropdown-
toggle" data-toggle="dropdown">  <span
class="hidden-xs">{{ Auth::user()-
>name }} </span> </a>
      <ul class="dropdown-menu">
        <li class="user-header">
          <div class="pull-left user-
img"></div>
          <p class="text-
left">{{ Auth::user()-
>name }} <small>{{ Auth::user()-
>email }} </small> </p>
        </li>
        <li><a class="dropdown-item"
href="{{ route('logout') }}"
onclick="event.preventDefault();
document.getElementById('logout-
form').submit();"><i class="fa fa-power-
off"></i>
          {{ __('Logout') }}
        </a>
      </li>
    </ul>
    <form id="logout-form"
action="{{ route('logout') }}" method="POST"
class="d-none">
      @csrf

```

```

</form>
</ul>
</li>
</ul>
</div>
</nav>
</header>
<!-- Left side column. contains the logo and
sidebar -->
<aside class="main-sidebar">
  <!-- sidebar: style can be found in
sidebar.less -->
  <div class="sidebar">
    <!-- Sidebar user panel -->
    <div class="user-panel">
      </div>

    <!-- sidebar menu: : style can be found
in sidebar.less -->
    <ul class="sidebar-menu" data-
widget="tree">
      <li class="{{
Route::currentRouteNamed('admin') ? 'active' : ''
}}"><a href="{{ route('admin') }}"> <i class="fa
fa-home"></i> <span>Dashboard</span></a>
      </li>
      @if (Auth::user()->role=='ADMIN')
      <li class="treeview {{
Route::currentRouteNamed('klinik*') ? 'active' : ''
}}"><a href="#"> <i class="fa fa-drivers-
license"></i> <span>Admin</span> <span
class="pull-right-container"> <i class="fa fa-
angle-left pull-right"></i> </span> </a>
      <ul class="treeview-menu">
        <li class="{{
Route::currentRouteNamed('klinik') ? 'active' : ''
}}"><a href="{{ route('klinik') }}">List
Klinik</a></li>
        <li class="{{
Route::currentRouteNamed('admin.transaksi') ?
'active' : '' }}"><a href="{{
route('admin.transaksi') }}">Transaksi</a></li>
        <li class="{{
Route::currentRouteNamed('laporan.keuangan') ?
'active' : '' }}"><a href="{{
route('laporan.keuangan') }}">Laporan
Keuangan</a></li>
      </ul>
      </li>
      <li class="treeview"> <a href="#">
<i class="fa fa-hospital-o"></i>
      <span>Klinik</span> <span class="pull-right-
container"> <i class="fa fa-angle-left pull-
right"></i> </span> </a>
      <ul class="treeview-menu">
        <li><a href="{{ route('klinik')
}}">List Klinik</a></li>
        <li><a href="{{ route('pasien')
}}">Pasien</a></li>
      </ul>
    </li>

```



```

        <li class="treeview"> <a href="#">
<i class="fa fa-id-badge"></i>
<span>Terapi</span> <span class="pull-right-
container"> <i class="fa fa-angle-left pull-
right"></i> </span> </a>
        <ul class="treeview-menu">
            <li><a href="#" { {
route('terapi.list') } }>List Terapi</a></li>
        </ul>
        </li>
    @endif
    @if (Auth::user()->role!='ADMIN')
        <li class="treeview"> <a href="#">
<i class="fa fa-stethoscope"></i>
<span>Pasien</span> <span class="pull-right-
container"> <i class="fa fa-angle-left pull-
right"></i> </span> </a>
        <ul class="treeview-menu">
            <li><a href="#" { {
route('pasien.list') } }>List Pasien</a></li>
            @if (Auth::user()-
>role=='KLINIK')
                <li><a href="#" { {
route('laporan.keuangan') } }>Laporan
Keuangan</a></li>
            @endif
        </ul>
        </li>
    @endif
    @if (Auth::user()->role=='ADMIN')
        <li class="treeview"> <a href="#">
<i class="fa fa-user-circle-o"></i>
<span>Customer</span> <span class="pull-right-
container"> <i class="fa fa-angle-left pull-
right"></i> </span> </a>
        <ul class="treeview-menu">
            <li><a href="#" { { route('pasien')
}}>List Customer</a></li>
        </ul>
        </li>
    @endif
</ul>
</div>
<!-- /.sidebar -->
</aside>

<!-- Content Wrapper. Contains page
content -->
<div class="content-wrapper">
    <!-- Content Header (Page header) -->
    <div class="content-header sty-one">
        <h1>@yield('title')</h1>
        <ol class="breadcrumb">
            <li>@yield('title')</li>
        </ol>
    </div>

    <!-- Main content -->
    <div class="content">
        <!-- Small boxes (Stat box) -->
        @yield('content')

```

```

    </div>
    <!-- /.content -->
</div>
<!-- /.content-wrapper -->
<footer class="main-footer">
    <div class="pull-right hidden-xs"></div>
    Copyright © 2021 Medis Reborn. All
rights reserved.
</footer>
</div>
<!-- /wrapper -->

<!-- jQuery 3 -->
<script
src="{ {URL::asset('backend') } }/js/jquery.min.js"
></script>

<!-- v4.0.0-alpha.6 -->
<script
src="{ {URL::asset('backend') } }/bootstrap/js/boot
strap.min.js"></script>

<!-- template -->
<script
src="{ {URL::asset('backend') } }/js/niche.js"></sc
ript>

<!-- Morris JavaScript -->
<script
src="{ {URL::asset('backend') } }/plugins/raphael/r
aphael-min.js"></script>
<script
src="{ {URL::asset('backend') } }/plugins/morris/
morris.js"></script>
<script
src="{ {URL::asset('backend') } }/plugins/function
s/morris-init.js"></script>
    @yield('js')
</body>

</html>

```

Frontend.blade.php

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta content="width=device-width, initial-
scale=1.0" name="viewport">

    <title>Medis Reborn Website</title>
    <meta content="" name="description">

    <meta content="" name="keywords">

    <!-- Favicons -->
    <link href="/assets/img/favicon.png"
rel="icon">

```

```
<link href="/assets/img/apple-touch-icon.png"
rel="apple-touch-icon">
```

```
<!-- Google Fonts -->
<link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i|Nunito:300,300i,400,400i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,600i,700,700i"
rel="stylesheet">
```

```
<!-- Vendor CSS Files -->
<link
href="/assets/vendor/bootstrap/css/bootstrap.min.css"
rel="stylesheet">
<link href="/assets/vendor/bootstrap-icons/bootstrap-icons.css"
rel="stylesheet">
<link href="/assets/vendor/aos/aos.css"
rel="stylesheet">
<link
href="/assets/vendor/remixicon/remixicon.css"
rel="stylesheet">
<link href="/assets/vendor/swiper/swiper-bundle.min.css"
rel="stylesheet">
<link
href="/assets/vendor/glightbox/css/glightbox.min.css"
rel="stylesheet">
```

```
<!-- Template Main CSS File -->
<link href="/assets/css/style.css"
rel="stylesheet">
```

```
<!--
=====
=====
* Template Name: FlexStart - v1.4.0
* Template URL:
https://bootstrapmade.com/flexstart-bootstrap-startup-template/
* Author: BootstrapMade.com
* License: https://bootstrapmade.com/license/
===== -->
```

```
</head>
<style>
.btn-dropdown{
padding: 10px 20px;
font-size: 15px;
text-transform: none;
font-weight: 600;
border: 0px;
background-color: white;
color: #5969F3;
}
</style>
@yield('css')
```

```
<body>
@php
$terapis=\App\Terapi::all();
// dd($terapis);
@endphp
```

```
<!-- ===== Header ===== -->
<header id="header" class="header fixed-top">
<div class="container-fluid container-xl d-flex
align-items-center justify-content-between">
```

```
<a href="{{ route('landing') }}" class="logo
d-flex align-items-center">

<span>MedisReborn</span>
</a>
```

```
<nav id="navbar" class="navbar">
<ul>
<li><a class="nav-link scrollto active"
href="{{ route('landing') }}">Home</a></li>
<li><a class="nav-link scrollto"
href="/gabung-mitra">Gabung Mitra
Klinik</a></li>
<li><a class="nav-link scrollto"
href="/tentang-kami">Hubungi Kami</a></li>
<li class="dropdown"><a
href="#"><span>Layanan Kami</span> <i
class="bi bi-chevron-down"></i></a>
<ul>
@foreach ($terapis as $terapi)
<form action="{{ route('search-terapi',strtolower($terapi->name)) }}"
method="POST">@csrf
<input type="hidden" name="latlong"
id="latlong">
<button type="submit" class="btn-
dropdown" style="width: 100%">{{ $terapi-
>name }}</a>
</form>
{{ -- <li><a href="{{ route('search-terapi',strtolower($terapi->name)) }}">{{ $terapi->name }}</a></li> -- }}
@endforeach
</ul>
</li>
@guest
<li><a class="getstarted scrollto"
href="{{ route('login') }}">{{ __('Login')
}}</a></li>
@if (Route::has('register'))
<li><a class="nav-link scrollto"
href="{{ route('register') }}">{{ __('Register')
}}</a></li>
@endif
@else
<li class="dropdown"><a
href="#"><span>{{ Auth::user()->name
}}</span> <i class="bi bi-chevron-
down"></i></a>
<ul>
<li><a href="{{ route('profile')
}}>Profile</a></li>
<li>
<a class="dropdown-item"
href="{{ route('logout') }}"
```

```

        onclick="event.preventDefault()
;
        document.getElementById(
ById('logout-form').submit();">
        {{ __('Logout') }}
</a>

<form id="logout-form"
action="{{ route('logout') }}" method="POST"
class="d-none">
        @csrf
</form>
</li>
</ul>
</li>
@endguest
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav><!-- .navbar -->

</div>
</header><!-- End Header -->

<main id="main">
    @yield('content')
</main><!-- End #main -->

<!-- ===== Footer ===== -->
<footer id="footer" class="footer">

<div class="footer-top">
<div class="container">
<div class="row gy-4">
<div class="col-lg-5 col-md-12 footer-
info">
<a href="{{ route('landing') }}"
class="logo d-flex align-items-center">

<span>MedisReborn</span>
</a>
<p>Medis Reborn Bermimpi Dapat
Melahirkan Kembali Pengobatan Tradisional
Dengan Inovasi Ilmu Pengetahuan Dan
Teknologi Terkini Medis Reborn</p>
<div class="social-links mt-3">
<a
href="https://web.facebook.com/profile.php?id=1
00021882166893" class="facebook"><i class="bi
bi-facebook"></i></a>
<a
href="https://www.instagram.com/medis_reborn/"
class="instagram"><i class="bi bi-instagram bx
bxl-instagram"></i></a>

</div>
</div>

</div>
</div>
</div>

```

```

<div class="container">
<div class="copyright">
    &copy; Copyright <strong><span>Medis
Reborn</span></strong>. All Rights Reserved
</div>
<div class="credits">
    <!-- All the links in the footer should remain
intact. -->
    <!-- You can delete the links only if you
purchased the pro version. -->
    <!-- Licensing information:
https://bootstrapmade.com/license/ -->
    <!-- Purchase the pro version with working
PHP/AJAX contact form:
https://bootstrapmade.com/flexstart-bootstrap-
startup-template/ -->

</div>
</div>
</footer><!-- End Footer -->

<a href="#" class="back-to-top d-flex align-
items-center justify-content-center"><i class="bi
bi-arrow-up-short"></i></a>

<!-- Vendor JS Files -->
<script
src="/assets/vendor/bootstrap/js/bootstrap.bundle.
js"></script>
<script
src="/assets/vendor/aos/aos.js"></script>
<script src="/assets/vendor/php-email-
form/validate.js"></script>
<script src="/assets/vendor/swiper/swiper-
bundle.min.js"></script>
<script
src="/assets/vendor/purecounter/purecounter.js">
</script>
<script src="/assets/vendor/isotope-
layout/isotope.pkgd.min.js"></script>
<script
src="/assets/vendor/glightbox/js/glightbox.min.js
"></script>

<!-- Template Main JS File -->
<script src="/assets/js/main.js"></script>
<!-- jQuery 3 -->
<script
src="{{ URL::asset('backend') }}/js/jquery.min.js"
></script>

<!-- v4.0.0-alpha.6 -->
<script
src="{{ URL::asset('backend') }}/bootstrap/js/boot
strap.min.js"></script>
<script>
    if (navigator.geolocation) {
        navigator.geolocation.getCurrentPosition(show
Position);
    } else {

```

```

        x.innerHTML = "Geolocation is not supported
by this browser.";
    }

function showPosition(position) {
    $( "input#latlong"
).val(position.coords.latitude+", "+position.coords
.longitude)
    // $("#latlong").each(function() {
    //     //
$(this).val(position.coords.latitude+", "+position.c
oords.longitude)
    //     //
$("#latlong").val(position.coords.latitude+", "+pos
ition.coords.longitude)
    //     console.log("1")

    // });

}
</script>
@yield('js')
</body>

</html>

```

home.blade.php

```

@extends('layouts.app')

@section('content')
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-8">
            <div class="card">
                <div class="card-header">{{
__('Dashboard') }}</div>

                <div class="card-body">
                    @if (session('status'))
                        <div class="alert alert-success"
role="alert">
                            {{ session('status') }}
                        </div>
                    @endif

                    {{ __('You are logged in!') }}
                </div>
            </div>
        </div>
    </div>
</div>
@endsection

```