

**SKRIPSI**

**SISTEM INFORMASI MANAJEMEN PUKESTREN  
TEBUIRENG JOMBANG BERBASIS WEB  
MENGUNAKAN *FRAMEWORK* LARAVEL**



**Oleh:**

**HAFRIL MAULANA MAHMUD  
NIM: 4115065**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PESANTREN TINGGI DARUL ULUM  
JOMBANG  
2019**



**SISTEM INFORMASI MANAJEMEN PUKESTREN  
TEBUIRENG JOMBANG BERBASIS WEB  
MENGUNAKAN FRAMEWORK LARAVEL**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk menyelesaikan  
Program Sarjana Komputer

**Oleh:**

**HAFRIL MAULANA MAHMUD  
NIM: 4115065**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS PESANTREN TINGGI DARUL ULUM  
JOMBANG  
2019**



## HALAMAN MOTTO

*“Belajar dan bekerja dengan giat, serta tidak lupa bersyukur, tentu akan memberikan hasil yang terbaik. Tak perlu malu karena berbuat kesalahan, sebab kesalahan akan membuatmu lebih bijak dari sebelumnya. Terus menggali ilmu dan pengetahuan baru, maka engkau akan bisa mengenali dan mengembangkan kemampuanmu sendiri. Terus jalani, hadapi apa yang engkau alami.”*

*“Jadilah karang seperti di lautan yang kuat dihantam ombak, kerjakanlah hal yang bermanfaat untuk diri sendiri dan orang lain, karena hidup hanyalah sekali.”*

*“Maka sesungguhnya bersama kesulitan ada kemudahan, sesungguhnya bersama kemudahan ada kesulitan. Maka apabila engkau telah selesai (dari sesuatu urusan), tetaplah bekerja keras (untuk urusan yang lain). Dan hanya kepada Tuhanmulah engkau berharap, (QS. Al-Insyirah : 6)*



## HALAMAN PERSEMBAHAN

Segala puji syukur selalu terucapkan kepada ALLAH SWT. Yang telah memberikan nikmat dan hidayah-Nya. Serta memberikan kekuatan, membekali ilmu yang bermanfaat sehingga peneliti dapat menyelesaikan skripsi ini dengan baik. Ucapan terimakasih yang tak terhingga, yang utama dari segalanya. Peneliti mempersembahkan kepada orang tercinta yang selalu memberikan dukungan dan bimbingan setulus hati, yaitu:

1. Orangtua tercinta, yang telah membesarkan dan mendidiku dengan penuh kasih sayang tanpa lelah, penuh kesabaran serta keikhlasan. Memberikan dukungan material, moral dan spiritual yang tak terhitung. Untaian doa yang selalu mengiringi langkah perjuanganku ini hingga ke puncak kesuksesan.
2. Seluruh Saudaraku tercinta, yang selalu memberikan dukungan penuh serta untaian doa yang dipanjatkan untuk mengiringi langkahku dalam pengerjaan skripsi.
3. Dosen Pembimbing I dan II, Ibu Diema Hernyka Satyareni, S.Kom., M.Kom dan Bapak Chandra Sukma Anugerah, S.Kom., M.Kom Yang telah memberikan bimbingan, saran dan perhatian dalam penyusunan skripsi ini.
4. Segenap Dosen Fakultas Teknik dan *Civitas Akademik* Unipdu Jombang, yang tidak bisa disebutkan satu persatu. Telah memberikan ilmu pengetahuan serta membimbing dengan pengarahan dan kesabaran selama peneliti duduk di bangku perkuliahan demi kesuksesan peneliti.
5. Terman-teman tercinta, terimakasih atas segala bantuannya, dukungan dan doa yang mengiringi dalam menyelesaikan skripsi ini.





## PERNYATAAN KEASLIAN

Saya yang bertanda-tangan di bawah ini menyatakan bahwa skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka. Pernyataan ini saya buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Dinyatakan di Jombang

Tanggal ... ..

{Ttd & materai 6000}

HAFRIL MAULANA MAHMUD

NIM: 4115065



## KATA PENGANTAR

Dengan mengucapkan puji dan syukur kehadirat Allah SWT, karena atas ridho dan hidayah-Nya, sehingga penulis dapat menyelesaikan Skripsi ini. Maksud dan tujuan dari penulisan Skripsi ini adalah untuk memenuhi persyaratan kelulusan Program Studi Strata I pada Jurusan Sistem Informasi di Universitas Pesantren Tinggi Darul 'Ulum Jombang.

Penulis merasa bahwa dalam menyusun laporan ini masih menemui beberapa kesulitan dan hambatan, disamping itu juga menyadari bahwa penulisan laporan ini masih jauh dari sempurna dan masih banyak kekurangan-kekurangan lainnya, maka dari itu penulis mengharapkan saran dan kritik yang membangun dari semua pihak.

Menyadari penyusunan laporan ini tidak lepas dari bantuan berbagai pihak, maka pada kesempatan ini penulis menyampaikan ucapan terima kasih yang setulus-tulusnya kepada: 1. .... 2. .... 3. ....

Akhir kata, semoga Allah SWT senantiasa melimpahkan karunia-Nya dan membalas segala amal budi serta kebaikan pihak-pihak yang telah membantu penulis dalam penyusunan laporan ini dan semoga tulisan ini dapat memberikan manfaat bagi pihak-pihak yang membutuhkan.

Jombang, 25 September 2019

Penulis



## DAFTAR ISI

HALAMAN PERSEMBAHAN .....	vii
PERNYATAAN KEASLIAN .....	ix
KATA PENGANTAR .....	xi
DAFTAR ISI .....	xiii
DAFTAR TABEL .....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR RUMUS.....	xix
DAFTAR LAMPIRAN .....	xxi
ABSTRAK .....	xxiii
BAB 1 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
1.6 Metode Penelitian.....	4
1.7 Sistematika Penulisan .....	6
BAB 2 LANDASAN TEORI.....	9
2.1 Penelitian Terdahulu.....	9
2.2 Kajian Pustaka.....	18
BAB 3 ANALISIS DAN PERANCANGAN.....	33
3.1 Analisis Sistem yang Berjalan .....	33
3.1.1 Analisis Proses Bisnis	<b>Error! Bookmark not defined.</b>
3.1.2 Analisis Sistem Informasi .....	<b>Error! Bookmark not defined.</b>

3.2 Analisis Sistem yang Diusulkan .....	35
3.2.1 Kebutuhan Fungsional .....	36
3.2.2 Kebutuhan Non Fungsional.....	38
3.2.3 Perancangan Algoritma/ <i>Knowledge Base/Rule Base</i> .....	<b>Error! Bookmark not defined.</b>
3.2.4 Desain Sistem yang Diusulkan.....	39
BAB 4 IMPLEMENTASI DAN PENGUJIAN .....	80
4.1 Lingkungan Pengembangan .....	80
4.1.1 Perangkat Keras .....	80
4.1.2 Perangkat Lunak .....	80
4.2 Lingkungan Implementasi .....	80
4.2.1 Perangkat Keras .....	81
4.2.2 Perangkat Lunak .....	81
4.3 Implementasi dan Pengujian Kode Program .....	81
4.3.1 Pengujian <i>Use Case Login</i> .....	81
4.3.2 Pengujian <i>Use Case</i> .....	81
4.4 Implementasi Basis Data .....	82
BAB 5 PENUTUP .....	83
5.1 Kesimpulan .....	83
5.2 Saran .....	83
DAFTAR PUSTAKA .....	85

## DAFTAR TABEL

Tabel 2.1 .....	<b>Error! Bookmark not defined.</b>
-----------------	-------------------------------------





## DAFTAR GAMBAR

Gambar 2.1 ..... **Error! Bookmark not defined.**



## DAFTAR RUMUS

{ Bila ada }



## DAFTAR LAMPIRAN

{ Bila ada }



## **ABSTRAK**

Abstrak berisi tentang latar belakang permasalahan yang di angkat, metodologi pengerjaan skripsi, dan hasil akhir dari skripsi ini, secara ringkas dan padat serta tidak bertele-tele.

Abstrak ditulis dalam 1 spasi dan memiliki kata kunci. Jumlah kata kunci antara 4 sampai 8.

**Kata Kunci:** kata1, kata2, kata3, kata4.





## **BAB 1**

### **PENDAHULUAN**

Bab pendahuluan ini menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, metode penelitian, dan sistematika penulisan dalam penelitian di Puskestren Tebuireng.

#### **1.1 Latar Belakang**

Puskestren (Pusat Kesehatan Pesantren) merupakan suatu organisasi kesehatan fungsional yang merupakan pusat pengembangan kesehatan di lingkup Pesantren Tebuireng Jombang, beralamatkan di Dusun Tebuireng Desa Cukir Kecamatan Diwek Kabupaten Jombang. Pada penelitian ini akan dibangun sebuah sistem informasi yang dapat membantu proses bisnis pelayanan kesehatan dan manajemen Puskestren.

Puskestren di lingkup pesantren Tebuireng dan masyarakat di sekitar Tebuireng mempunyai peran yang sangat vital sebagai institusi pelaksana teknis, dituntut memiliki kemampuan manajerial dan wawasan jauh ke depan untuk meningkatkan kualitas pelayanan kesehatan. Peran tersebut ditunjukkan dalam bentuk keikutsertaan dalam menentukan kebijakan daerah melalui sistem perencanaan yang matang dan realistis, tata laksana kegiatan yang tersusun rapi, serta sistem evaluasi dan pemantauan yang akurat.

Sistem yang sedang berjalan di Pukestren Tebuireng masih dilakukan dengan cara konvensional, artinya untuk melakukan pelayanannya selama ini belum dilakukan secara terkomputerisasi, sering terjadinya salah penulisan identitas pasien, penggandaan nomer rekam medis, data pasien tidak *up-to-date*, sehingga proses yang berjalan selama ini menjadi kurang efektif dan efisien dalam mencakup semua aspek yang

ada di Puskesmas Tebuireng meliputi, pengolahan data pasien, pengolahan data pegawai, pengolahan data inventaris yang ada di Puskesmas, dan lambatnya proses pelayanan pasien.

Penggunaan komputer untuk pengolahan data pasien sangat diperlukan, karena dapat memberikan beberapa keuntungan dan kemudahan dalam pelayanan pasien antara lain: mempercepat pelayanan, informasi yang lebih akurat, manajemen pegawai lebih sistematis, pencarian data lebih tepat, dan laporan yang lebih akurat.

Berdasarkan uraian diatas, maka peneliti tertarik untuk mengatasi permasalahan diatas dan peneliti mengangkat judul “Sistem Informasi Manajemen Puskesmas Tebuireng Jombang Berbasis Web dengan Menggunakan *Framework Laravel*”. Skripsi ini akan membuat sistem informasi berbasis *web* yang diharapkan dapat membantu memecahkan masalah di Puskesmas Tebuireng dan juga sebagai media promosi secara *online*.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang permasalahan diatas, maka rumusan masalah dalam penelitian ini sebagai berikut:.

- 1) Bagaimana merancang sebuah Sistem Informasi yang dapat meningkatkan kinerja pelayanan kesehatan di Puskesmas Tebuireng Jombang?
- 2) Bagaimana membangun sebuah Sistem Informasi yang dapat meningkatkan kinerja pelayanan kesehatan di Puskesmas Tebuireng Jombang?

## **1.3 Batasan Masalah**

Berdasarkan latar belakang dan rumusan masalah maka peneliti hanya membatasi masalah pada:

- 1) Sistem informasi manajemen Puskesmas Tebuireng Jombang berbasis *web* berisi tentang mengelola

pegawai, mengelola dokter (jam pelayanan praktek), mengelola obat, mengelola biaya berobat pasien, mengelola asset Puskesmas Tebuireng Jombang (medis maupun non medis), rekam medis (dilakukan oleh aktor: petugas rekam medis), mengelola laporan (dilakukan oleh aktor: petugas rekam medis), diagnosa pasien (dilakukan oleh aktor: dokter).

- 2) Perancangan sistem dilakukan dengan *Unified Modeling Language* (UML).
- 3) Pembangunan sistem dilakukan dengan bahasa pemrograman PHP dengan basis data MySQL.
- 4) Pembangunan sistem menggunakan *framework* Laravel.
- 5) Tidak membahas sistem jaringan *web*.
- 6) Ruang lingkup masalah hanya berhubungan dengan manajemen pelayanan di Puskesmas Tebuireng Jombang.
- 7) Aktor dalam sistem terdiri dari : admin, pegawai, dan dokter.

#### **1.4 Tujuan Penelitian**

Tujuan dari penelitian merupakan hal yang menjadi tolak ukur berakhirnya suatu penelitian. Adapun tujuan yang ingin dicapai dalam penelitian ini, antara lain.

- 1) Merancang sebuah Sistem Informasi yang dapat meningkatkan kinerja pelayanan kesehatan di Puskesmas Tebuireng Jombang.
- 2) Membangun sebuah Sistem Informasi yang dapat meningkatkan kinerja pelayanan kesehatan di Puskesmas Tebuireng Jombang.

#### **1.5 Manfaat Penelitian**

Dari hasil luaran yang dihasilkan, pada bagian ini disebutkan pihak-pihak mana saja yang akan mendapat manfaat serta disebutkan manfaat seperti apa yang diterima

oleh masing-masing pihak. Misalkan untuk penelitian rancang bangun perangkat lunak Sistem Informasi Akademik untuk Unipdu maka pihak-pihak yang mendapatkan manfaat antara lain:

1) Peneliti

Hasil skripsi ini akan memberikan wawasan pengetahuan tentang masalah yang diteliti, sehingga dapat diperoleh gambaran yang jelas mengenai “Sistem Informasi Manajemen Puskestren Tebuireng Jombang Berbasis *Web* dengan menggunakan *Fremwork* *Laravel*”, baik secara analisis maupun perancangan aplikasi sistem informasi. Penelitian ini dilakukan untuk memenuhi persyaratan kelulusan Program Studi Strata I pada Jurusan Sistem Informasi di Universitas Pesantren Tinggi Darul ‘Ulum Jombang.

2) Puskestren Tebuireng

Sistem Informasi yang dapat meningkatkan kinerja pelayanan kesehatan di Puskestren Tebuireng Jombang.

3) Unipdu

Penelitian ini menjadi salah satu bahan yang akan memperkaya pustaka penelitian Universitas yang dapat digunakan sebagai bahan rujukan untuk penelitian sejenis di masa mendatang. Membantu perkembangan ilmu pengetahuan dalam kajian ilmu dan perkembangan teknologi.

## 1.6 Metode Penelitian

Penelitian ini dilakukan melalui beberapa tahapan. Adapun metode yang digunakan dalam tiap-tiap tahapan antara lain:

1) Metode Pengumpulan Data

a. Wawancara

Dengan mengajukan pertanyaan secara langsung kepada pihak Puskestren Tebuireng dengan

berlandaskan pada tujuan penelitian dan objek yang diteliti.

b. Studi Pustaka

Pada tahap ini dilakukan dengan mempelajari sumber-sumber yang berkaitan dengan penelitian ini, baik dari buku, jurnal, artikel dan lain-lain yang berkaitan tentang penelitian yang sedang dilakukan.

c. Observasi

Pengumpulan data dilakukan dengan pengamatan secara langsung terhadap objek penelitian untuk memperoleh suatu informasi sebagai bahan penelitian.

d. Deskriptif

Metode yang bertujuan untuk menghasilkan gambaran akurat tentang sebuah kelompok, menggambarkan mekanisme sebuah proses atau hubungan secara sistematis, actual, factual serta akurat mengenai fakta-fakta yang diteliti.

2) Metode Rekayasa Perangkat Lunak

Perangkat lunak dalam penelitian ini dirancang, dibangun, serta diuji coba menggunakan metode *waterfall model*. Tahap-tahap proses *waterfall model*, antara lain:

a. Metode Perancangan

Perancangan dilakukan dengan menggunakan notasi UML dan pemodelan berorientasi obyek sehingga diagram yang digunakan adalah *Use Case*, *Activity*, *Sequence* dan *Class Diagram*. Desain basis data menggunakan ORM dari Hibernate.

b. Metode Pembangunan

Sistem Informasi Manajemen Pukestren dirancang menggunakan *Unified Modeling Language* (UML). Perancangan sistem dibuat berorientasi obyek menggunakan *use case diagram*, *activity diagram*, *sequence diagram* dan *class diagram*. Tools yang digunakan dalam proses ini adalah

aplikasi StarUML. Selanjutnya, perancangan antar muka sistem menggunakan aplikasi *Evolus Pencil*.

c. Metode Pembangunan

Sistem Informasi Manajemen Puskestren dibangun menggunakan bahasa pemrograman PHP (PHP *Hypertext Preprocessor*) dengan basis data MySQL.

d. Metode Uji Coba

Uji coba dilakukan menggunakan metode *blackbox testing*. Tahap ini bertujuan untuk memastikan seluruh fungsionalitas sistem sudah berfungsi sebagaimana mestinya.

## 1.7 Sistematika Penulisan

Sistem penulisan dalam tugas akhir ini disusun dalam bentuk karya ilmiah dengan struktur penulisan sebagai berikut:

### BAB 1 PENDAHULUAN

Bab ini berisi uraian tentang latar belakang masalah yang mendasari pentingnya diadakan penelitian, identifikasi, pembatasan dan perumusan masalah penelitian, maksud dan tujuan penelitian, kegunaan penelitian yang diharapkan, dan sistematika penulisan.

### BAB 2 LANDASAN TEORI

Bab ini berisi tentang penelitian terdahulu, pengertian pemesanan, pengertian undangan pernikahan, pengertian sistem informasi, pengertian metode *waterfall*, perancangan UML (*Unified Modeling Language*), pengertian basis data, pengertian PHP (PHP *Hypertext Preprocessor*), pengertian MySQL, pengertian *blackbox testing*.

### BAB 3 ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi tentang analisis kebutuhan dari sistem yang akan dibangun, perbandingan sistem yang sudah ada dengan sistem yang diusulkan, dan perancangan sistem yang diusulkan.

## BAB 4 TESTING DAN IMPLEMENTASI SISTEM

Bab ini berisi tentang hasil-hasil dari tahapan penelitian mulai dari tahap penelitian, desain sistem, implementasi sistem dan pengujian *blackbox*. Selain itu sebaiknya hasil penelitian juga dibandingkan dengan hasil penelitian terdahulu yang sejenis.

## BAB 5 PENUTUP

Bab ini berisi pokok-pokok kesimpulan dan saran-saran yang perlu disampaikan kepada pihak-pihak yang berkepentingan dengan hasil penelitian.

## DAFTAR PUSTAKA

Daftar pustaka berisi daftar referensi yang digunakan dalam penelitian.





## **BAB 2**

### **LANDASAN TEORI**

Bab ini memaparkan tentang penelitian-penelitian terdahulu yang berhubungan dengan topik skripsi. Bab 2 ini juga menjelaskan teori-teori yang digunakan oleh penulis dalam pembuatan tugas skripsi seperti : penelitian terdahulu mengenai sistem informasi yang pernah dikembangkan, pengertian sistem informasi, metodologi Penelitian yang digunakan, pengertian pukestren, pengertian sistem informasi, metodologi penelitian terdiri dari *waterfall* sebagai metode pengembangan sistem, *Unified Modeling Language* (UML) sebagai metode perancangan sistem, *PHP Hypertext Preprocessor* (PHP) sebagai metode pembangunan sistem, *framework* laravel, serta *Blackbox Testing* sebagai metode pengujian sistem.

#### **2.1 Penelitian Terdahulu**

Penelitian Terdahulu merupakan sarana untuk menunjukkan keaslian penelitian. Bagian ini memuat uraian sistematis tentang informasi hasil penelitian sebelumnya.

##### **2.1.1 Rachmat Agusli, Muhammad Iqbal Hanafri, Hernita Sari (2015) Rancang Bangun Sistem Informasi Klinik Menggunakan VB.Net (Studi Kasus : PT. Surya Toto Indonesia)**

Penelitian ini mengangkat masalah pengelolaan data pendaftaran pasien di sebuah perusahaan Untuk memenuhi pelayanan kesehatan kepada karyawan yang semakin hari semakin meningkat serta demi kelancaran proses aktivitas kerja di perusahaan, maka perancangan sistem berbasiskomputer sangat diperlukan untuk mencapai pelayanan kesehatan yang optimal.

Rachmat Agusli, Muhammad Iqbal Hanafri, Hernita Sari membuat aplikasi Pendaftaran Pasien berbasis *desktop* yang dibangun menggunakan perangkat lunak *visual basic* dan pembuatan *database* menggunakan perangkat lunak *Microsoft Access*. Untuk perancangan sistem ini menggunakan perancangan UML (*Unified Modeling Language*) dan perancangan ERD (*Entity Relationship Diagram*) dalam menyajikan informasi yang sesuai dengan kebutuhan dan pembuatan laporan yang akan memudahkan dalam pengambilan keputusan lebih baik.

Tujuan penelitian ini dimaksudkan Untuk mengatasi masalah tersebut maka dirancang sistem informasi pendaftaran pasien yang terkomputerisasi, sehingga karyawan dapat langsung melakukan pendaftaran pada admin plant di departement masing-masing. Sehingga memiliki nomor antrian dan jam pelayanan yang sudah ditentukan penjadwalannya.

### **2.1.2 Andi Suprianto, Asri Amaliza Fathia Matsea, (2018) Rancang Bangun Aplikasi Pendaftaran di Klinik Pengobatan Medisina berbasis Web**

Penelitian ini mengangkat masalah yang sering terjadi pada Klinik Medisina yaitu pada pasien akan melakukan pemeriksaan dan belum mempunyai kartu berobat harus mendaftarkan diri kepada bagian pendaftaran dan memberikan kartu identitas diri yang dicatat oleh bagian pendaftaran selanjutnya pasien diberi kartu berobat dan nomor antrian untuk menunggu panggilan oleh perawat untuk masuk ke ruangan dokter yang akan memeriksanya. Jika pasien sudah mempunyai kartu berobat pasien harus tetap mendaftar dan hanya menyebutkan nama beserta alamat. Bagian pendaftaran mencari rekam medis pasien tersebut.

Andi Suprianto, Asri Amaliza Fathia Matsea membuat sebuah sistem informasi berbasis *web* untuk menangani masalah pendaftaran, poliklinik, dan pelaporan. Sistem

informasi ini dibangun menggunakan bahasa pemrograman PHP dan MySQL untuk mengelola basis datanya. Tujuan dibuat sistem ini untuk menghasilkan sistem informasi yang dapat diakses melalui *web browser*.

Tujuan dari penelitian ini adalah merancang sebuah aplikasi berbasis web untuk memudahkan pasien dalam mendaftarkan dirinya atau anggota keluarganya tanpa harus pasien mengantri dan menunggu lama. Memudahkan pegawai klinik dalam melayani pendaftaran pasien. Meningkatkan efisiensi waktu dalam hal pelayanan terhadap pasien. Memudahkan pasien dalam mengetahui hasil rekam mediknya, Sehingga pasien dapat mengetahui penyakit yang sedang menyerang sistem kekebalan tubuhnya. Memudahkan staff administrasi klinik medisina dalam pembuatan laporan tanpa harus mengeluarkan biaya yang besar.

### **2.1.3 Johni S Pasaribu, Johnson Sihombing (2018) Perancangan Sistem Informasi Rekam Medis Pasien Jalan Berbasis Web di Klinik Sehat Margasari Bandung**

Penelitian ini mengangkat Penelitian ini mengangkat proses yang berjalan, Klinik Sehat Margasari melakukan proses rawat jalan dengan kartu riwayat kesehatan pasien yang disimpan oleh pihak klini yang berbentuk kertas sehingga kartu kesehatan pasien sulit untuk ditemukan ketika data riwayat pasien dibutuhkan untuk pemeriksaan rawat jalan. Oleh karena itu dibuatlah rancang bangun sistem informasi rekam medis pasien rawat jalan berbasis web di Klinik Sehat Margasari. Adapun ruang lingkup dari sistem pelayanan pasien rawat jalan ini meliputi pengolahan.

Johni S Pasaribu, Johnson Sihombing membuat sebuah sistem informasi berbasis web untuk menangani masalah data pasien, data kunjungan pasien, data riwayat kesehatan pasien, biaya pengobatan pasien dan data resep untuk pasien. Sistem informasi ini dibangun menggunakan

bahasa pemrograman PHP dan MySQL untuk mengelola basis datanya. Tujuan dibuat sistem ini untuk menghasilkan sistem informasi yang dapat diakses melalui *web browser*

Tujuan dari adanya sistem informasi pelayanan pasien rawat jalan ini adalah merancang dan membangun sebuah sistem informasi berbasis web pada Klinik Sehat Margasari. Manfaat dari sistem informasi pelayanan pasien rawat jalan adalah memudahkan resepsionis dalam pengolahan data pasien, biaya pengobatan dan mempermudah pemuatan rekapitulasi pasien rawat jalan yang dapat membantu dalam pengambilan keputusan oleh manajemen rumah sakit.

#### **2.1.4 I Putu Agus Yoga Permana, (2015) Rancang Bangun Sistem Informasi Klinik Praktik Dokter Berbasis Web dengan Menggunakan PHP dan MySQL (Studi Kasus: Klinik Utama Dharma Sidhi Denpasar, Bali)**

Penelitian ini mencoba menerapkan konsep sistem informasi klinik dokter praktik berbasis web untuk mengelolah data pasien, data dokter, data obat , data tindakan dan data rekam medis. Untuk itu diperlukansuatu sistem yang sekiranya dapat meningkatkan efektifitas dan efisiensi kerja. Selain itu, masih banyak klinik menggunakan cara manual untuk mencatat seluruh data kesehatan pasien, sehingga data kesehatan pasien sulit dikontrol mengakibatkan *human error* dan tidak dapat memberikan informasi yang akurat.

Tujuan dari peneliti ini adalah ingin membuat Sistem Informasi berbasis Web yang dirancang dengan menggunakan pemrograman PHP dan MySQL sebagai databasenya dapat menjadi solusi atas permasalahan yang dihadapi.

I Putu Agus Yoga Permana membuat sebuah Aplikasi sistem informasi klinik berbasis web sebagai solusi untuk mengelola data-data klinik secara cepat dan mudah dibandingkan dengan manual sehingga lebih efisien dan

menghemat tempat penyimpanan maupun pengelolaan seperti menambah data, mengubah (Update) data, dan menghapus data.

### **2.1.5 Yehezkiel Yanu Putranto, Toni Wijanarko Adi Putra, Fitro Nur Hakim (2017) Rancang Bangun Sistem Informasi Rekam Medis Klinik Berbasis Web (Studi Kasus: Klinik Utama Meditama Semarang)**

Penelitian ini mengangkat masalah pencatatan medis yang dilakukan oleh 1 orang petugas rekam medis menggunakan *software Microsoft Excel* selama proses rekam medis ini berjalan adalah salah ketik atau penulisan identitas pasien, penggandaan nomer rekam medis, data pasien tidak up-to-date, salah penulisan diagnosa dengan dokter periksa, sehingga proses rekam medis menjadi kurang efektif dan efisien.

Yehezkiel Yanu Putranto, Toni Wijanarko Adi Putra, Fitro Nur Hakim membuat sebuah sistem informasi berbasis *web* untuk menangani masalah pendaftaran, poliklinik, dan pelaporan. Sistem informasi ini dibangun menggunakan bahasa pemrograman PHP dan MySQL untuk mengelola basis datanya. Tujuan dibuat sistem ini untuk menghasilkan sistem informasi yang dapat diakses melalui *web browser*.

Tujuan penelitian ini dimaksudkan untuk membentuk bertujuan supaya pengelolaan data medis dapat berjalan dengan cepat dan tepat. Sistem yang dibuat oleh Yehezkiel Yanu Putranto, Toni Wijanarko Adi Putra, Fitro Nur Hakim menghasilkan beberapa menu, diantaranya: menu pendaftaran baru riwayat pasien dan jam praktek dokter.

Untuk menyimpulkan perbandingan antara peneliti dan penelitian terdahulu. Maka peneliti menyajikan data perbandingan dalam bentuk table matriks penelitian terdahulu. Tabel matriks penelitian dapat dilihat pada Tabel 2.1 , Tabel 2.2 dan tabel 2.3.



Tabel 2. 1 Tabel Penelitian Terdahulu

No	Judul Penelitian	Nama Peneliti	Persamaan	Perbedaan	Hasil
1)	Rancang Bangun Sistem Informasi Klinik Menggunakan VB.Net(Studi Kasus : PT. Surya Toto Indonesia)	Rachmat Agusli, Muhammad Iqbal Hanafri, Hernita Sari (2015))	1) Data pasien 2) Pendaftaran pasien baru	1) Aplikasi berbasis dekstop (tidak bisa diakses secara online)	Memudahkan Petugas kesehatan Surya Toto Indonesia dalam mengelola data pasien
2)	Rancang Bangun Aplikasi Pendaftaran Pasien di Klinik Pengobatan Medisina berbasis Web	Andi Suprianto, Asri Amaliza Fathia Matsea, (2018)	1) Mengeloa data pasien	1) Studi kasus yang di ambil	Mempermudah staff administrasi dalam membuat laporan
<b>B A T A S</b>					

Tabel 2. 2.LanjutanTabel Penelitian Terdahulu

No	Judul Penelitian	Nama Peneliti	Persamaan	Perbedaan	Hasil
3)	Perancangan Sistem Informasi Rekam Medis Pasien Jalan Berbasis Web di Klinik Sehat Margasari Bandung	Johni S Pasaribu, Johnson Sihombing (2018)	1) Mengelola data Pasien 2) Mengelola data Obat	1) Belum bisa mencetak nomer antrian pasien	Mempermudah petugas rekam medis dalam mencetak laporan data pasien
4)	Rancang Bangun Sistem Informasi Klinik Praktik Dokter Berbasis Web dengan Menggunakan PHP dan MySQL (Studi Kasus: Klinik Utama Dharma Sidhi Denpasar, Bali)	I Putu Agus Yoga Permana, (2015)	1) Mengelola jam praktik dokter 2) Mengelola data pasien	1) Belum bisa mencetak riwayat diagnosa penyakit pasien	Mempermudah pasien dalam melihat jam praktik dokter
<b>B A T A S</b>					



Tabel 2.3. LanjutanTabel Penelitian Terdahulu

No	Judul Penelitian	Nama Peneliti	Persamaan	Perbedaan	Hasil
5)	Rancang Bangun Sistem Informasi Rekam Medis Klinik Berbasis Web (Studi Kasus: Klinik Utama Meditama Semarang)	Yehezkiel Yanu Putranto, Toni Wijanarko Adi Putra, Fitro Nur Hakim (2017)	1) Mengelola data pasien 2) mengelola data obat 3)mengelola data riwayat pasien	1) studi kasus yang di ambil	Mempermudah petugas rekam medis dalam mengelola laporan riwayat pasien

## **2.2 Kajian Pustaka**

Bagian ini memuat rangkuman teori-teori yang terdapat dalam Sistem Informasi manajemen berbasis *Web* di Puskestren Tebuireng Jombang, yaitu pengertian Puskestren, pengertian sistem informasi manajemen, metodologi yang digunakan dalam penelitian ini terdiri dari metode pengembangan sistem (*Waterfall Mode*), metode perancangan (*Unified Modeling Language*), metode pembangunan sistem (PHP dan MySQL), serta metode pengujian sistem (*Blackbox Testing*).

### **2.2.1 Puskestren**

Puskestren merupakan salah satu bentuk Upaya Kesehatan Bersumberdaya Masyarakat (UKBM) di lingkungan pondok pesantren, dengan prinsip dari, oleh, dan untuk warga pondok pesantren, yang mengutamakan pelayanan promotif, preventif, tanpa mengabaikan aspek kuratif dan rehabilitatif, dengan binaan puskesmas setempat. Pedoman penyelenggaraan dan pembinaan puskestren dilakukan diatur dalam Permenkes (No. 1 tahun 2013).

### **2.2.2 Sistem Informasi Manajemen**

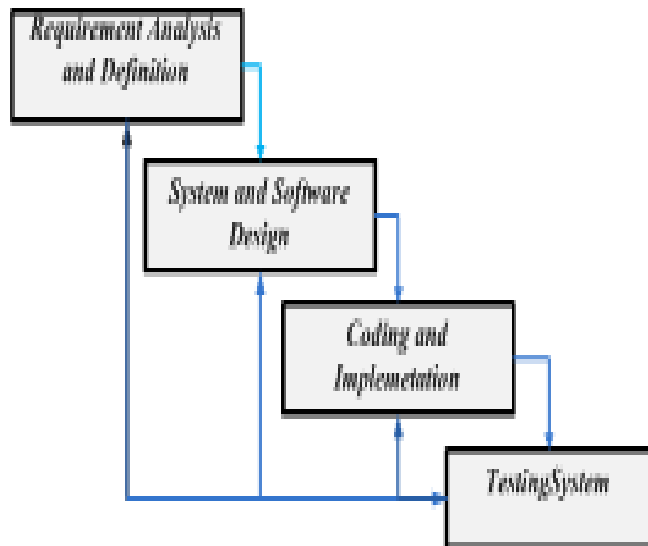
Menurut Hanif Al Fatta (2010:12) Sistem Informasi Manajemen (SIM) adalah sebuah sistem informasi pada level manajemen yang berfungsi untuk membantu perencanaan, pengendalian, dan pengambilan keputusan dengan menyediakan resume rutin dan laporan-laporan tertentu. SIM mengambil data mentah dari TPS (Transaction Processing System) dan mengubahnya menjadi kumpulan data yang lebih berarti yang dibutuhkan manager untuk menjalankan tanggung jawabnya. Untuk mengembangkan suatu SIM, diperlukan pemahaman yang baik tentang

informasi apa saja yang dibutuhkan manajer dan bagaimana mereka menggunakan informasi tersebut.

### 2.2.3 Metode *Waterfall*

Menurut Pressman (2010), model *waterfall* adalah model klasik yang bersifat sistematis berurutan dalam membangun *software*, Nama model ini sebenarnya adalah “*Linear Sequenced Model*”. Model ini disebut dengan “*Classic Life Cycle*” atau model *waterfall*.

Alur model *waterfall* dapat dilihat pada Gambar 2.1



Gambar 2.1 Model *Waterfall*

Berikut ini penjelasan mengenai metode penelitian *waterfall* beserta alur prosesnya

*a) Requirements Analysis (Analisis Kebutuhan)*

Proses pengumpulan kebutuhan diintegrasikan difokuskan, khususnya pada perangkat lunak. Untuk memahami sifat program yang dibangun, rekayasa perangkat lunak (analisis) harus memahami domain informasi tingkah laku, unjuk kerja dan antar muka (*interface*) yang diperlukan. Kebutuhan baik untuk sistem maupun perangkat lunak didokumentasikan dan dilihat dengan pelanggan.

Mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh *software* yang akan dibangun. Hal ini sangat penting mengingat *software* harus dapat berinteraksi dengan elemen-elemen yang lain seperti *database*.

*b) System and Software Design (Perancangan Sistem)*

Proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda, struktur dan *detail* (algoritma) procedural. Proses desain menerjemahkan syarat atau kebutuhan kedalam sebuah *representasi* perangkat lunak yang didapat diperkirakan demi kualitas sebelum dimulai permunculan kode. Sebagaimana persyaratan desain didokumentasikan dan menjadi bagian dari konfigurasi perangkat lunak. Proses pencarian kebutuhan diintegrasikan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, *user interface* dan sebagainya. Dari dua aktifitas tersebut (pencarian kebutuhan sistem dan *software* harus didokumentasikan dan ditunjukkan kepada *user* . Proses *software design* untuk mengubah kebutuhan-kebutuhan diatas menjadi *representasi* kedalam bentuk "*blueprint*" *software* sebelum *coding* dimulai.

Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya, seperti dua aktifitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

*c) Coding dan Implementation (Pengkodean, Implementasi)*

Desain harus diterjemahkan dalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuk yang dapat dimengerti oleh mesin, yaitu kedalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap *design* yang secara teknis nantinya dikerjakan oleh seorang *programmer*.

*d. Testing System (Pengujian Sistem)*

Proses pengujian dilakukan pada logika internal untuk memastikan semua pernyataan sudah diuji. Pengujian eksternal fungsional untuk menemukan kesalahan-kesalahan dan memastikan bahwa *input* akan memberikan hasil yang actual sesuai yang dibutuhkan.


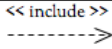
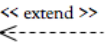



## **2.2.4 Unified Modeling Language (UML)**

Menurut Nugroho (2010:6), mengatakan bahwa *Unified Modeling Language* adalah bahasa pemodelan untuk sistem maupun perangkat lunak yang berparadigma berorientasi objek. Pemodelan sesungguhnya digunakan untuk penyederhanaan permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipahami. *Use case diagram* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari tiap interaksi.

Berikut ini merupakan bagian-bagian dari diagram *Unified Modeling Language* (UML):

### 1) Use Case Diagram

Diagram ini memperlihatkan himpunan *Use Case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna (Widodo, 2011). Simbol *use case diagram* dapat dilihat pada Gambar 2.2.

NO	GAMBAR	NAMA	KETERANGAN
1		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		Include	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
3		Extend	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
4		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.
5		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
6		Use Case	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

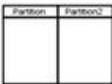





Gambar 2.2 Simbol *Use Case Diagram*

### 2) Activity Diagram

Diagram aktifitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini sangat penting, terutama dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek, dapat didefinisikan urutan atau pengelompokan tampilan dari

sistem atau *user interface* di mana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan serta rancang menu yang ditampilkan pada perangkat lunak (Widodo, 2011).

Simbol *activity diagram* dapat dilihat pada Gambar 2.3.

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Swimlane</i>	Menunjukkan siapa yang bertanggung jawab dalam melakukan aktivitas dalam suatu diagram.
2		<i>Action</i>	Langkah-langkah dalam sebuah activity. Action bisa terjadi saat memasuki activity, meninggalkan activity, atau pada event yang spesifik.
3		<i>Initial State</i>	Menunjukkan dimana aliran kerja dimulai.
4		<i>Activity Final Node</i>	Menunjukkan dimana aliran kerja diakhiri.
5		<i>Decision Node</i>	Menunjukkan suatu keputusan yang mempunyai satu atau lebih transisi dan dua atau lebih transisi sesuai dengan suatu kondisi.
6		<i>Control Flow</i>	Menunjukkan bagaimana kendali suatu aktivitas terjadi pada aliran kerja dalam tindakan tertentu.







Gambar 2.3 Simbol Activity Diagram

### 3) Sequence Diagram

Menurut (Widodo, 2011) diagram urutan adalah interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu. Dalam *sequence diagram*, setiap obyek hanya memiliki garis yang digambarkan garis putus-putus ke bawah. Pesan antar obyek digambarkan dengan anak panah

dari obyek yang mengirimkan pesan ke obyek yang menerima pesan. Diagram yang menggambarkan interaksi obyek dan mengindikasikan petunjuk atau tanda komunikasi diantara obyek-obyek tertentu.

Untuk simbol *Sequence Diagram* dapat dilihat pada Gambar 2.4.

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Swimlane</i>	Menunjukkan siapa yang bertanggung jawab dalam melakukan aktivitas dalam suatu diagram.
2		<i>Action</i>	Langkah-langkah dalam sebuah activity. Action bisa terjadi saat memasuki activity, meninggalkan activity, atau pada event yang spesifik.
3		<i>Initial State</i>	Menunjukkan dimana aliran kerja dimulai.
4		<i>Activity Final Node</i>	Menunjukkan dimana aliran kerja diakhiri.
5		<i>Decision Node</i>	Menunjukkan suatu keputusan yang mempunyai satu atau lebih transisi dan dua atau lebih transisi sesuai dengan suatu kondisi.
6		<i>Control Flow</i>	Menunjukkan bagaimana kendali suatu aktivitas terjadi pada aliran kerja dalam tindakan tertentu.

Gambar 2.4 Simbol *Sequence Diagram*




#### 4) *Class Diagram*

Menurut (Prados, 2010) dalam perancangan *database* pada *Unified Modeling Language* (UML), biasanya



menggunakan *Class Diagram*. Sedangkan pada *Entity Relationship Diagram* (ERD) menggunakan *Conceptual Data Model* (CDM). *Class Diagram* merupakan sebuah jenis dari diagram yang menjelaskan struktur statis dari sebuah sistem yang menunjukkan bahwa *class diagram* sama dengan *Conceptual Data Model* (CDM).

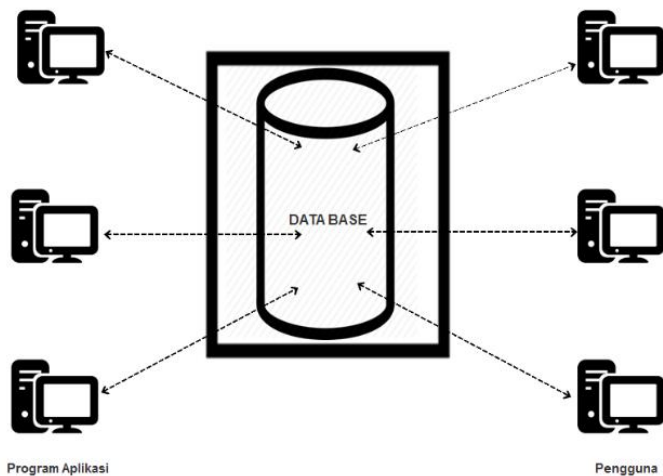
Tabel pada *class diagram* disebut *class*, sedangkan pada *conceptual data model* (CDM) disebut *entity*. Pada *class diagram* terdapat *operation*, sedangkan pada *conceptual data model* (CDM) tidak ada. Hubungan antar tabel pada *class diagram* disebut *association*, sedangkan pada *conceptual data model* (CDM) disebut *relation*. Tabel yang terdapat pada *class diagram* dan *conceptual data model* (CDM) jumlahnya sama (Prados, 2010). Simbol-simbol *class diagram* dapat dilihat pada Gambar 2.5.

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
3		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
4		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
5		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Gambar 2.5 Simbol *Class Diagram*

### 2.2.5 Basis Data

Menurut Eri Mardiani (2016), basis data adalah sekumpulan data operasional lengkap dari organisasi maupun perusahaan yang dikelola dan disimpan secara terintegrasi menggunakan metode tertentu, dengan menggunakan komputer sehingga mampu menyediakan informasi secara optimal yang diperlukan oleh pemakainya. Contohnya, *database* penjualan, *database* akademik, *database* pembelian. Sistem *database* adalah suatu sistem penyusunan dan pengelolaan *record-record* menggunakan komputer yang mempunyai tujuan untuk menyimpan atau merekam dan memelihara data operasional lengkap sebuah organisasi maupun perusahaan sehingga dapat menyediakan informasi secara optimal yang diperlukan oleh pemakai untuk kepentingan proses pengambilan keputusan. Berikut ini merupakan gambaran dari basis data, dapat dilihat pada Gambar 2.6.



Gambar 2.6. Gambaran Basis Data

### 2.2.6 PHP (PHP *Hypertext Preprocessor*)

Menurut Kustiyaningsih (2011:114), “PHP (atau resminya PHP: *Hypertext Preprocessor*) adalah skrip bersifat *server-slide* yang ditambahkan ke dalam HTML”. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode PHP untuk mengirimkan permintaan ke *server*. Sistem kerja dari PHP diawali dengan permintaan yang berasal dari halaman *website* oleh *browser*. Berdasarkan URL atau alamat *website* dalam jaringan *internet*, *browser* menemukan alamat dari *webserver*, memeriksa halaman yang disetujui dan menyampaikan informasi yang diperlukan oleh *webserver*. Penggunaan PHP memungkinkan *web* dapat dibuat secara dinamis sehingga *maintenance* situs *web* tersebut menjadi lebih mudah dan efisien. PHP merupakan *software open source* yang disebarkan dan dilisensikan secara gratis serta dapat *download* secara bebas dari situs resmi.

Sebagai *engine* dari aplikasi *web*, PHP sangat cepat, *portable* dan sangat terintegrasi dengan berbagai aplikasi *server* dan *database* yang ada. Seperti MySQL, MsSQL, SQL, Oracle dan Sybase. Lagi pula PHP mudah dipelajari dan mudah pula dikembangkan fungsi-fungsinya atau *extensible* banyak *library* baik dalam bentuk *class*, *script* dan lain-lain bertebaran di *internet* menunggu kita unduh. Alasan mengapa *hypertext preprocessor* (PHP) digunakan dalam pembuatan program adalah sebagai berikut:

- 1) Bahasa pemrograman PHP terbukti sangat handal dalam membangun sebuah program berbasis *web*.
- 2) Waktu yang digunakan untuk memproses data dan menjalankan perintah-perintah *query* sangat cepat.
- 3) Dengan berjalan dalam sebuah *web server*, maka secara otomatis program ini bersifat *multiuser*.
- 4) *Database* MySQL dalam menyimpan data ditempatkan kedalam direktori khusus dan terpisah dengan *file* program

PHP yang lain sehingga keamanan data dari orang yang tidak bertanggung jawab lebih terjamin.

- 5) *Web server* dan *database server* terpisah, sehingga menyulitkan pihak luar yang tidak mempunyai akses untuk mengakses data didalam *database*.

Menurut (Setiawan, 2017), PHP merupakan kependekan dari *Hypertext Preprocessor* yang merupakan sebuah bahasa *script* tingkat tinggi yang dipasangkan pada dokumen HTML. Sebagian besar sintaks PHP mirip dengan bahasa C, java dan perl. Namun, PHP terdapat beberapa fungsi yang lebih spesifik PHP digunakan untuk merancang yang sifatnya dinamis dan dapat bekerja secara otomatis.

### 2.2.7 MySQL (*My Structured Query Language*)

Menurut Kustiyaningsih (2011:145), "MySQL adalah sebuah basis data yang mengandung jumlah tabel yang terdiri dari sejumlah baris dan setiap baris mengandung sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung sejumlah tabel". Tipe data MySQL menurut (Kustiyaningsih, 2011:147), "Tipe data MySQL merupakan data yang terdapat dalam tabel yang berupa *field-field* berisi nilai dari data tersebut. Nilai data didalam *field* memiliki tipe sendiri-sendiri".

Menurut AS. Rossa dan Shalahuddin. M (2011), MySQL adalah sebuah perangkat lunak *database* relasi (*relation database management system* atau RDMS) seperti halnya *oracle*, *postgresql*, *microsoft SQL*. MySQL tidak disamakan dengan SQL (*structure query language*) yang didefinisikan sebagai sintaks perintah-perintah tertentu dalam bahasa pemrograman yang digunakan untuk mengelola suatu *database*. Sebagai *database server* yang mampu untuk mengatur *database* dengan baik. MySQL terhitung lebih baik daripada *database server* yang lain. Berikut ini merupakan

kelebihan-kelebihan yang dimiliki oleh MySQL. Kelebihan MySQL antara lain:

- 1) *Database server* yang menyimpan data berkapasitas besar.
- 2) MySQL didukung oleh *server ODBC*, yang artinya *database* MySQL dapat diakses menggunakan aplikasi seperti *delphi* maupun *visual basic*.
- 3) MySQL merupakan *database server* yang *multi-user* dapat digunakan banyak orang.
- 4) MySQL dapat menciptakan lebih dari 16 kunci per tabel dan satu kunci memungkinkan belasan *field*.
- 5) MySQL adalah *database* yang menggunakan enkripsi *password*.

### 2.2.8 Framework Laravel

Pengertian *framework* menurut Naista (2017) adalah suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah yang kompleks. Singkatnya, *framework* adalah wadah atau kerangka kerja dari sebuah *website* yang akan dibangun. Dengan menggunakan kerangka tersebut waktu yang digunakan dalam membuat *website* lebih singkat dan memudahkan dalam melakukan perbaikan.

Salah satu *framework* yang banyak digunakan oleh programmer adalah *framework* laravel. Laravel adalah *framework* berbasis PHP yang sifatnya *open source*, dan menggunakan konsep *model*, *view*, dan *controller*. Laravel berada di bawah lisensi MIT *License* dengan menggunakan *github* sebagai tempat berbagi *code* menjalankannya.

Dalam penggunaannya, laravel memiliki beberapa kekurangan salah satunya yaitu ukuran file yang cukup besar. Berikut adalah dasar-dasar laravel:

- 1) Artisan, adalah *command line* atau perintah yang dijalankan melalui terminal dan disediakan beberapa perintah-perintah yang dapat digunakan untuk

pengembangan dan pembuatan aplikasi. Berfungsi membuka *website* yang telah dibuat tanpa menggunakan *web server* lokal.

- 2) *Routing*, suatu proses yang bertujuan agar suatu *item* yang diinginkan dapat sampai ke tujuan. Dengan menggunakan *routing* dapat ditentukan halaman-halaman yang akan muncul ketika dibuka oleh *user*.
- 3) *Controller*, suatu proses yang bertujuan untuk mengambil permintaan, menginisialisasi, memanggil model untuk dikirimkan ke *view*.
- 4) *View (blade templating)* adalah *template engine* bawaan dari laravel memiliki kode-kode yang lebih mudah untuk menghasilkan laravel. Cara membuat *file .blade* dilakukan secara manual dengan membuat nama\_file.php.blade di dalam *folder views*.
- 5) *Middleware* adalah penengah antara *request* yang masuk dengan *controller* yang dituju. Cara membuat *middleware* menggunakan artisan dengan mengetikkan "php artisan make:middleware nama\_file". *File middleware* berada di dalam *folder middleware*.
- 6) *Session* adalah sebuah cara yang digunakan untuk penyimpanan pada *server* dan penyimpanan tersebut digunakan pada beberapa halaman termasuk halaman itu sendiri. Dalam menggunakan *session* ada dua cara. Cara yang pertama *session* dapat dibuat menggunakan *request*. Cara yang kedua dapat digunakan fungsi *global helper session*.

### 2.2.9 Blackbox Testing

Menurut Nidhra dan Dondetti (2012), mengatakan bahwa *black box testing* berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* mendefinisikan kumpulan kondisi *input* dan melakukan percobaan pada spesifikasi fungsional aplikasi. *Black box testing* bukanlah solusi alternatif dari *white box testing* tapi lebih sebagai pelengkap untuk menguji hal-hal

yang tidak bisa dicakup oleh *white box testing*. Metode *black box testing* salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan.

Menurut Pratama (2014), pengujian ini menekankan pada sejauh mana fungsionalitas sistem informasi dapat berjalan dengan baik sesuai keinginan *user* dan *developer* yang meliputi empat pengujian dari sisi pengembang:

- 1) Pengujian *interface* (tata muka) aplikasi.
  - 2) Pengujian fungsi dasar sistem.
  - 3) Pengujian *form handle* sistem.
- Pengujian keamanan sistem.





## **BAB 3**

### **ANALISIS DAN PERANCANGAN SISTEM**

Bab Analisis dan Perancangan ini berisi tentang pengaplikasian sistem. Selain itu, pada bab ini juga dipaparkan perbandingan sistem yang sudah ada dengan sistem yang diusulkan, sehingga dapat ditemukan keunggulan yang diberikan oleh sistem baru yang diusulkan. Terakhir, bab ini memaparkan pula perancangan sistem yang diusulkan. Perancangan dilakukan menggunakan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*. Selain perancangan *diagram*, disajikan pula rancangan desain basis data dan rancangan *user interface* (antarmuka).

#### **3.1 Analisis Sistem yang Berjalan**

Pada umumnya pelayanan kesehatan mempunyai peran yang sangat vital sebagai institusi pelaksana teknis, dituntut memiliki kemampuan manajerial dan wawasan jauh ke depan untuk meningkatkan kualitas pelayanan kesehatan. Peran tersebut ditunjukkan dalam bentuk keikutsertaan dalam menentukan kebijakan daerah melalui sistem perencanaan yang matang dan realistis, tata laksana kegiatan yang tersusun rapi, serta sistem evaluasi dan pemantauan yang akurat.

Puskestren (Pusat Kesehatan Pesantren) yang beralamatkan di Dusun Tebuireng Desa Cukir Kecamatan Diwek Kabupaten Jombang. Sistem yang sedang berjalan di Puskestren belum terkomputerisasi, artinya sistem pelayanan masih dilakukan dengan cara manual sebagai contoh untuk proses pendaftaran pasien, pencatatan hasil pemeriksaan pasien, pencatatan data transaksi pembayaran pasien masih menggunakan pencatatan dalam buku besar selain itu melihat langsung proses pasien berobat ke Puskestren dimana untuk mendaftar saja membutuhkan waktu yang cukup lama.

Masyarakat dan santri yang datang dengan kondisi sakit pastinya ingin segera mendapatkan penanganan, namun dengan proses pendaftaran saja yang membutuhkan waktu cukup lama.

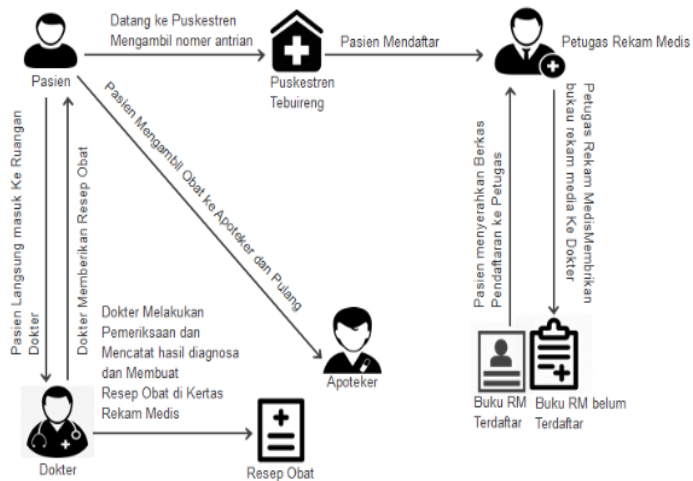
Salah satu faktor penyebab kegiatan pelayanan penerimaan pasien di puskestren hal ini mengakibatkan proses kegiatan pelayanan membutuhkan waktu yang lama. Terutama pada saat proses pencarian nomor rekam medis pasien yang tidak membawa kartu berobat. sering terjadi kesalahan pencatatan data dan pelayanan penerimaan pasien. Sistem ini dinilai kurang efektif jika dikaitkan dengan perkembangan teknologi yang sudah berkembang pesat.

Penjelasan proses yang sedang berjalan di Puskestren Tebuireng Jombang, antara lain:

- a) Pasien mendaftarkan diri pada bagian administrasi.
- b) Bagian administrasi menanyakan nama, umur, dan alamat pasien, untuk kemudian:
  1. Menuliskannya semua data di atas pada Kartu Berobat sebagai pegangan pasien.
  2. Menuliskan huruf pertama pada nama pasien untuk didaftarkan pada buku indeks.
  3. Menuliskannya semua data di atas pada kartu berwarna hijau yang berfungsi sebagai rekam medis.
- c) Kartu rekam medis diberikan pada Bagian Keperawatan di ruang tunggu pasien. Sementara itu pasien duduk menunggu panggilan.
- d) Bagian Keperawatan memanggil pasien, menanyakan kendala awal pasien, kemudian melakukan pengecekan tensi darah dan gula darah bila diperlukan.
- e) Hasil pengecekan tensi dan/atau gula darah, juga kartu rekam medis diberikan Bagian Keperawatan kepada dokter di ruang konsultasi, diikuti pasien.
- f) Konsultasi antara pasien dengan dokter.
- g) Pengecekan fisik (bila diperlukan).
- h) Dokter mengisi buku rekam medis dan menuliskan resep untuk pasien.

- i) Pasien membawa buku rekam medis dan resep ke ruang obat.
- j) Petugas Apoteker mengecek resep, kemudian memberikan obat.

Untuk gambaran sistem yang berjalan pada Puskesmas Tebuireng Jombang dapat dilihat pada Gambar 3.1.



Gambar 3.1 Sistem yang berjalan di Puskesmas Tebuireng Jombang

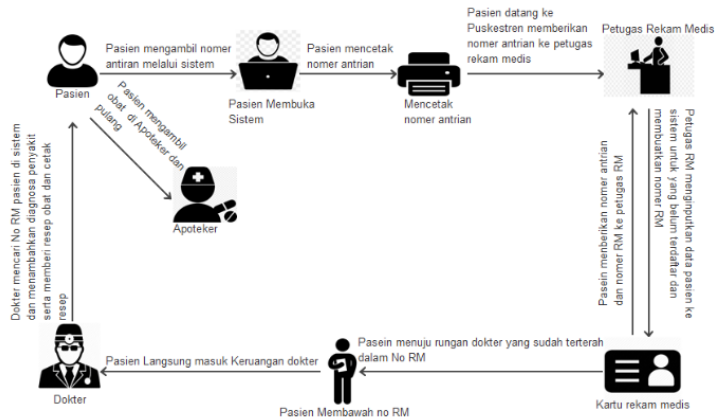
### 3.2 Analisis Sistem yang Diusulkan

Untuk mengatasi permasalahan yang terjadi di Puskesmas Tebuireng Jombang, perlu dibangun sebuah sistem informasi. Sistem informasi yang diusulkan harus mampu digunakan untuk mengelola proses meningkatkan pelayanan kesehatan bagi masyarakat dan santri. Melalui sistem informasi dalam penelitian ini, Proses bisnis yang

diusulkan dapat dilihat pada Gambar 3.2. Penjelasan proses bisnis yang diusulkan antara lain:

- a) Pasien mengambil nomer antrian melalui *website* Puskestren Tebuireng Jombang.
- b) Pasien melakukan pendaftaran di loket administrasi (bagi pasien yang belum mempunyai kartu medis). Bagi yang belum mempunyai kartu medis dapat langsung mengurus pendaftaran sesuai ketentuan yang berlaku.
- c) Pasien yang sudah terdaftar di Puskestren Tebuireng Jombang, pasien hanya menyerahkan kartu medis selanjutnya petugas administrasi mencari nomer rekam medis pasien.
- d) Setelah petugas rekam medis memanggil pasien untuk menanyakan keluhan pasien. Petugas rekam medis mengisikan data pada sistem dan dicetak untuk diserahkan kepada pasien.
- e) Petugas rekam medis *menginputkan* data pasien ke sistem dan menambahkan poli serta menentukan dokter yang sesuai dengan keluhan pasien tersebut.
- f) Petugas rekam medis memberikan *form* rekam medis pasien ke dokter untuk dilakukan diagnosa menuju ke poli yang sudah ditentukan oleh petugas rekam medis.
- g) Pasien menunggu panggilan dari dokter. Setelah mendapatkan panggilan dari dokter, pasien masuk ke ruangan untuk dilakukan diagnosa oleh dokter.
- h) Dokter mendiagnosa pasien melalui sistem dengan cara pasien menunjukkan nomor rekam medis yang ada pada selebaran cetak dari petugas rekam medis untuk dilakukan diagnosa melalui sistem.
- i) Dokter mencetak selebaran resep obat (termasuk hasil diagnose pasien) untuk diberikan ke pasien.
- j) Pasien menerima selebaran resep obat dan menuju ke loket pengambilan obat (apoteker).
- k) Apoteker menerima resep obat dari pasien dan mengecek obat yang akan diberikan ke pasien.

Untuk gambaran sistem yang diusulkan pada Puskesmas Tebuireng Jombang dapat dilihat pada Gambar 3.2.



Gambar 3.2 sistem yang diusulkan pada Puskesmas Tebuireng Jombang

### 3.2.1 Kebutuhan Fungsional

Setelah usulan sistem baru dijelaskan dengan baik maka akan dapat diidentifikasi daftar kebutuhan fungsional dari sistem yang diusulkan. Kebutuhan fungsional dapat dijabarkan dalam bentuk poin-poin, tabel ataupun bentuk yang susai. Kebutuhan fungsional Sistem Informasi Manajemen berbasis Web di Puskesmas Tebuireng berisi kebutuhan yang mencakup proses apa yang bisa dilakukan oleh sistem. Kebutuhan fungsional diuraikan pada Tabel 3.1.

Tabel 3.1 Tabel Kebutuhan Fungsional

No	Kebutuhan Fungsional	Proses
1)	Sistem harus mampu melakukan pendaftaran pasien.	1) Dapat memasukkan data diri sebagai pasien.

Tabel 3.2 Tabel Lanjutan Kebutuhan Fungsional

No	Kebutuhan Fungsional	Proses
	Sistem harus mampu melakukan pendaftaran pasien	2) Pasien mendapatkan kartu medis jika sudah terdaftar
2)	Sistem harus mampu menampilkan nomer antrian pasien	1) Pasien dapat mengambil nomer antrian melalui sistem.
3)	Sistem harus mampu menampilkan riwayat penyakit pasien	1) Pasien dapat mengetahui riwayat penyakit di <i>website</i>
4)	Sistem harus mampu menampilkan diagnose penyakit dan resep obat dari dokter	1) Dokter mendiagnosa penyakit dan memberi resep pada pasien di sistem

### 3.2.2 Kebutuhan Non Fungsional

Kebutuhan non-fungsional Sistem Informasi Manajemen berbasis *Web* di Puskestren Tebuireng Jombang berisi kebutuhan yang mencakup properti yang dimiliki oleh sistem. Kebutuhan non-fungsional diuraikan pada Tabel 3.3. dan Tabel 3.4.

Tabel 3.3 Tabel Kebutuhan Non Fungsional

No	Kebutuhan Non Fungsional	Properti
1	Pelaksanaan	1) Menggunakan sistem informasi.

Tabel 3.4 Tabel Lanjutan Kebutuhan Non Fungsional

No	Kebutuhan Non Fungsional	Properti
	Pelaksanaan	2) Spesifikasi minimum komputer Processor Intel(R) Core(TM) i3-2348M CPU @2.30 GHz 2.30 GHz. 3) Kebutuhan RAM 2.00 GB; 4) Kebutuhan harddisk 249 GB. 5) Printer, internet.
2	Kemanan	1) Aplikasi dan basis data dilengkapi dengan <i>username,password</i> menggunakan MD5.
3	Informasi	1) Menampilkan informasi nomor antrian; 2) Menampilkan informasi riwayat pasien; 3) Menampilkan informasi jam praktek dokter.

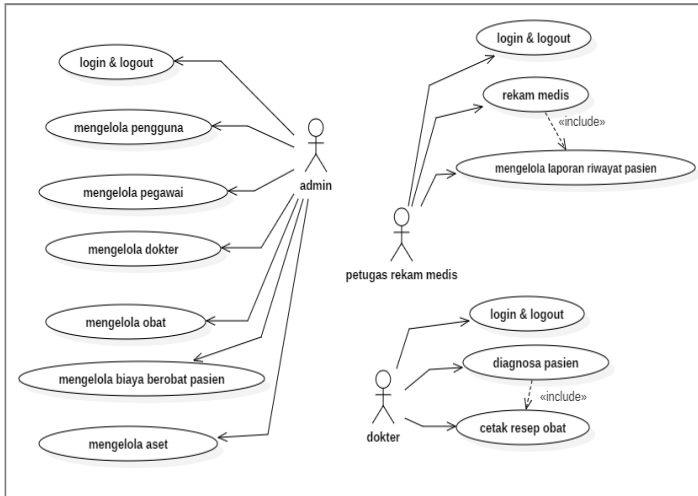
### 3.2.3 Perancangan Sistem yang Diusulkan

Desain sistem yang diusulkan ini dibuat dengan asumsi pemodelan berorientas obyek maka dari itu komponen yang ada adalah *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, *Class Diagram*, dan Desain Basis Data.

#### A. Pemodelan Use Case

Pada bagian ini dimunculkan *Use Case Diagram* serta deskripsi tiap aktor di dalam sistem. Untuk masing-

masing *use case* dalam *use case diagram* akan dijabarkan dengan detail pada sub bab berikutnya. *Use Case Diagram* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Pemodelan *Use Case Diagram*

Dalam perancangan untuk membangun sistem informasi manajemen puskestren berbasis *web*. Peneliti memberikan 10 *Use Case Diagram*. Penjelasan *Use Case Diagram* dapat dilihat pada Tabel 3.5

Tabel 3.5 Tabel Deskripsi *Use Case Diagram*

<i>Use Case Diagram</i>	Deskripsi
<i>Login dan Logout</i>	Proses yang mengawali masuk dan keluar sistem. Pengguna dapat memasukkan username dan password ke dalam halaman login.



Tabel 3.6 Tabel Lanjutan Deskripsi *Use Case Diagram*

<i>Use Case Diagram</i>	Deskripsi
Lanjutan <i>Login</i> dan <i>Logout</i>	Admin dapat menambahkan data pegawai jika ada pegawai baru, admin juga dapat merubah data pegawai sesuai dengan kebutuhan.
Mengelola Pengguna	Pada <i>dashboard</i> admin terdapat menu “Pengguna” berisi data-data pengguna yang sudah dibuatkan oleh admin. Admin dapat mengelola pengguna seperti: tambah data, <i>edit</i> data.
Mengelola Data Pegawai	Pada <i>dashboard</i> admin terdapat menu “Pegawai” yang berfungsi menampilkan data-data pegawai yang ada di Puskesmas Tebuireng.
Mengelola Data Dokter	Pada <i>dashboard</i> admin terdapat menu “Dokter” yang berfungsi untuk menambahkan data dokter yang nantinya akan muncul pada halaman utama sistem. Seperti memunculkan jam praktik dokter sesuai harinya.
Mengelola Data Obat	Pada <i>dashboard</i> admin terdapat menu “Obat” yang berfungsi menambahkan data obat, hasil <i>input</i> data akan muncul pada sistem yang nantinya diakses oleh dokter ketika dokter memberikan resep ke pasien.

Tabel 3.7 Tabel Lanjutan Deskripsi *Use Case Diagram*

<b><i>Use Case Diagram</i></b>	<b>Deskripsi</b>
Lanjutan Mengelola Data Obat	Admin dapat menambahkan data obat, merubah data obat maupun menghapus obat tersebut sesuai dengan kebutuhan.
Mengelola Biaya Berobat	Admin dapat mengelola biaya yang ditentukan oleh Puskestren sesuai dengan jenis pengobatan yang berlaku. Hasil <i>input</i> biaya obat akan tampil pada sistem yang nantinya diakses oleh petugas rekam medis.
Mengelola Aset Barang	Petugas pengelola inventori dapat mengelola data aset barang medis maupun non medis yang ada di Puskestren Tebuireng pada halaman aset barang yang diakses oleh petugas inventori. Petugas inventori dapat menambahkan data, merubah data, dan menghapus data sesuai kebutuhan.
Rekam Medis	Pada halaman petugas rekam medis terdapat menu “Rekam Medis” yang berfungsi untuk petugas rekam medis melayani pendaftaran pasien secara sistematis.

Tabel 3.8 Tabel Lanjutan Deskripsi *Use Case Diagram*

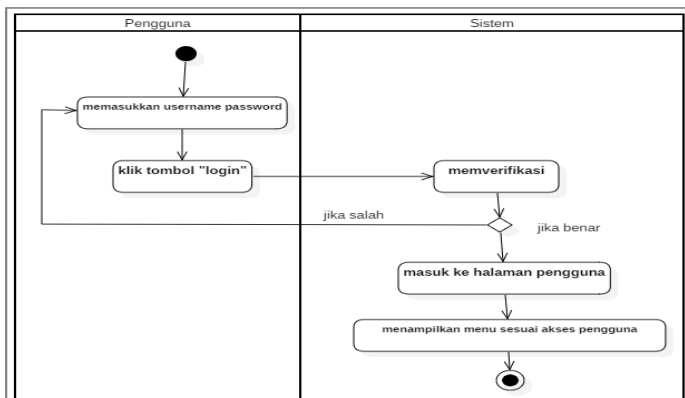
<i>Use Case Diagram</i>	Deskripsi
	yang didasarkan dari berkas-berkas persyaratan pendaftaran pasien berobat yang nantinya dicatat ke dalam buku rekam medis melalui sistem. Petugas rekam medis dapat mencetak hasil rekaman pasien dan diberikan kepada pasien yang berobat, pasien tersebut dapat menemui dokter yang sudah ditetapkan oleh petugas sesuai dengan jenis keluhan pasien. Petugas dapat mencetak riwayat berobat pasien pada menu “Rekam Medis” ini.
Diagnosa Pasien	Pada halaman dokter yang diakses oleh dokter yang ada di Puskestren sesuai dengan bidangnya terdapat menu “Diagnosa Penyakit” yang mencakup hasil diagnosa penyakit beserta resep yang ditentukan dokter tersebut. Setelah dokter memeriksa pasien, dokter dapat menginputkan data hasil diagnosa ke dalam sistem, menentukan resep obat untuk pasien. Dokter dapat mencetak selemba hasil diagnosa beserta resep obat.

## B. Activity Diagram

*Activity Diagram* berfungsi memodelkan alur kerja sebuah proses bisnis dan urutan aktifitas pada suatu proses. Dalam perancangan untuk membangun sebuah Sistem Informasi Manajemen Puskestren Tebuireng, peneliti memberikan 10 aktifitas yang terdiri dari *Activity Diagram Login* dan *Logout* untuk admin, petugas rekam medis, dokter dan petugas aset, *Activity Diagram Mengelola Pengguna*, *Activity Diagram Mengelola Data Pegawai*, *Activity Diagram Mengelola Data Dokter*, *Activity Diagram Mengelola Data Obat*, *Activity Diagram Mengelola Biaya Berobat*, *Activity Diagram Mengelola Inventori*, *Activity Diagram Rekam Medis*, *Activity Diagram Mengelola Laporan*, *Activity Diagram Diagnosa Penyakit*.

### 1) Activity Diagram Login

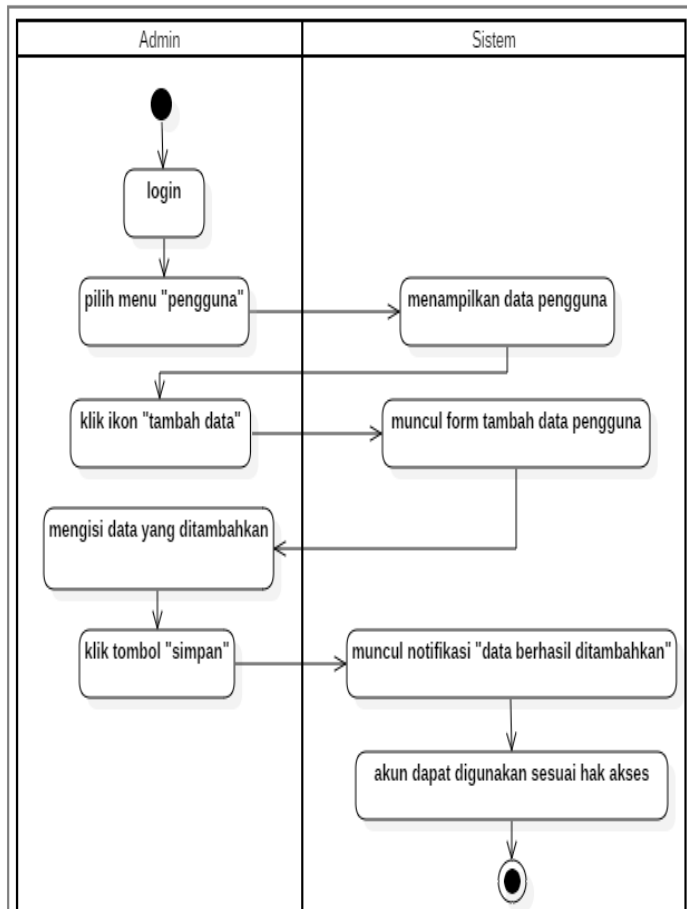
Activity diagram login pengguna menggambarkan pengguna memasukkan username dan password pada form login dan sistem memverifikasi kebenaran data jika data yang dimasukkan tidak valid, maka pengguna kembali ke halaman login. Apabila data valid, maka pengguna masuk ke halaman pengguna sesuai bidang masing-masing. *Activity diagram* login dapat dilihat pada Gambar 3.4.



Gambar 3.4 Activity Diagram Login Pengguna

## 2) Activity Diagram Mengelola Pengguna

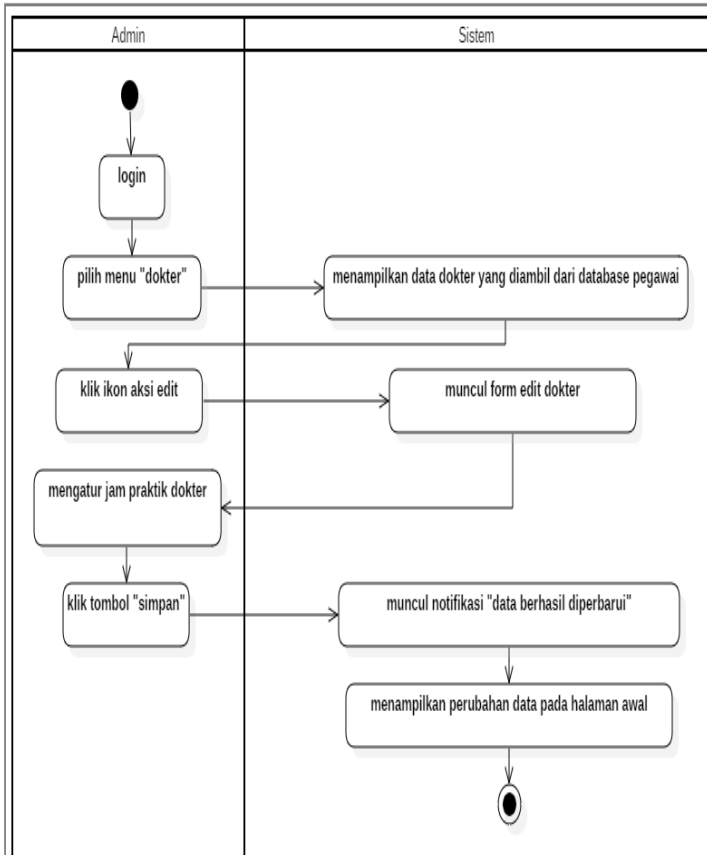
Mengelola pengguna menggambarkan bahwa admin dapat menambah (create), merubah (edit), menghapus (delete) akun pengguna yang sudah terdaftar akun di database. *Activity diagram* mengelola pengguna dapat dilihat pada Gambar 3.5.



Gambar 3.5 Activity Diagram Mengelola Pengguna

### 3) Activity Diagram Mengelola Pegawai

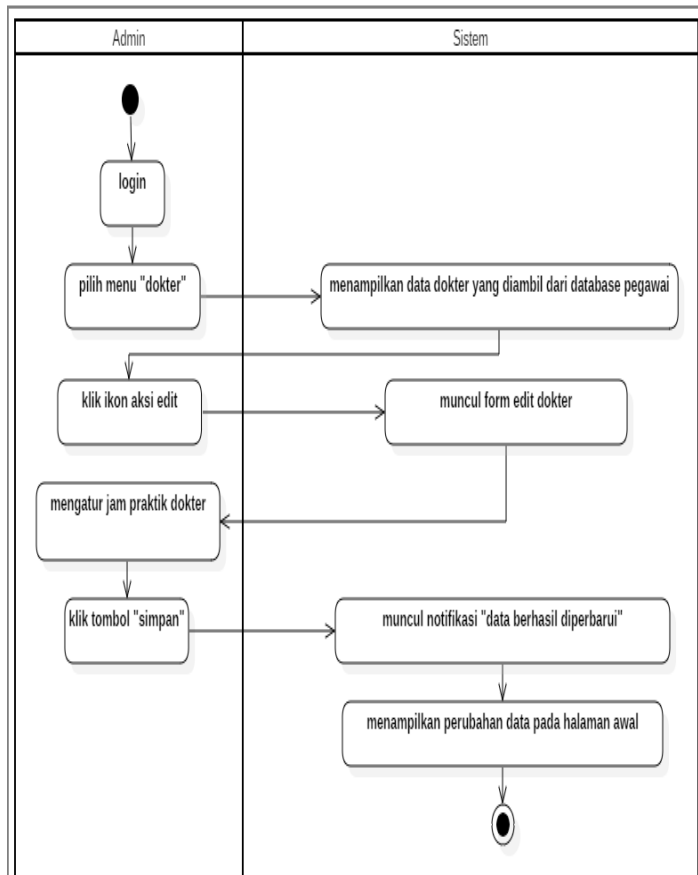
Mengelola data pegawai menggambarkan bahwa admin dapat menambah data pegawai di puskestren, merubah data pegawai seperti biodata pegawai dan menghapus data pegawai yang sudah tidak bekerja di puskestren tersebut. *Activity diagram* mengelola pegawai dapat dilihat pada Gambar 3.6.



Gambar 3.6 Activity Diagram Mengelola Pegawai

#### 4) Activity Diagram Mengelola Dokter

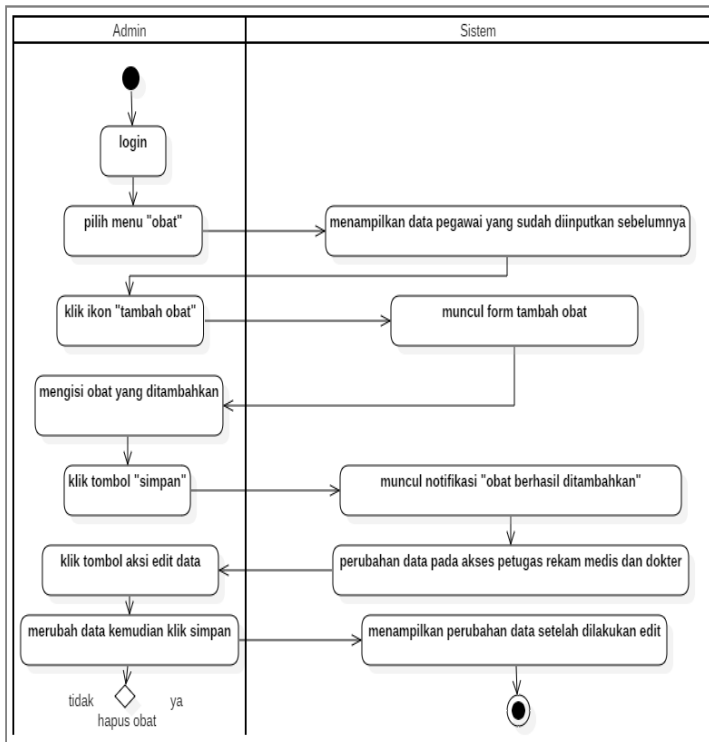
Menggambarkan bahwa admin dapat merubah jam praktek dokter tersebut sesuai dengan ketentuan yang berlaku di puskestren tersebut. Hasil perubahan data akan tampil (include) pada halaman rekam medis yang diakses oleh petugas rekam medis. Activity diagram mengelola data dokter dapat dilihat pada Gambar 3.7.



Gambar 3.7 Activity Diagram Mengelola Dokter

### 5) Activity Diagram Mengelola Obat

Menggambarkan bahwa admin dapat menambahkan data obat sesuai dengan kebutuhan, merubah data obat dan menghapus data obat. Hasil tambah data maupun ubah data akan tampil (include) pada halaman diagnose dokter yang diakses oleh dokter. *Activity diagram* mengelola data obat dapat dilihat pada Gambar 3.8.



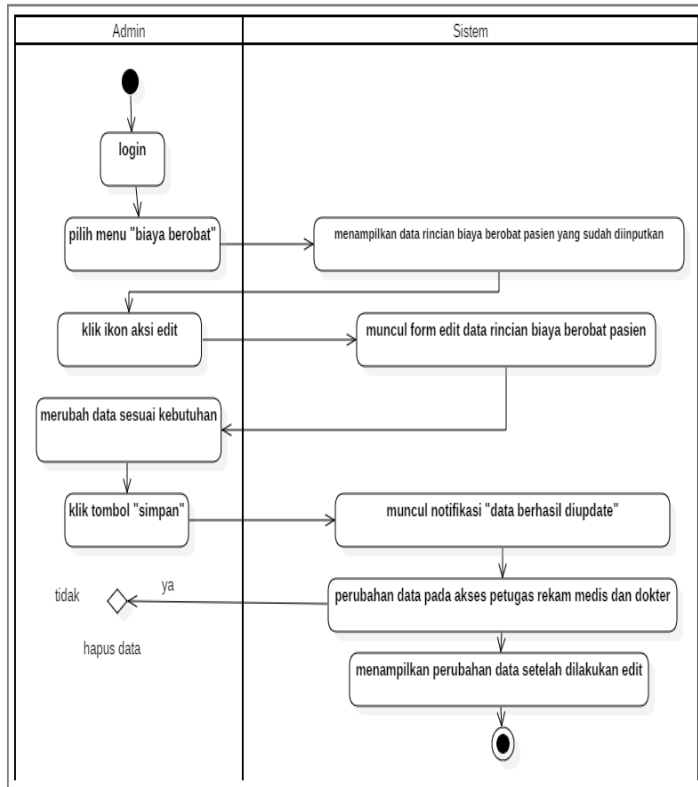
Gambar 3.8 Activity Diagram Mengelola Obat

### 6) Activity Diagram Mengelola Biaya Berobat Pasien

Pada halaman admin terdapat menu "Biaya Berobat" yang berfungsi untuk menambahkan biaya berobat pasien



yang nantinya keluar pada menu rekam medis yang diakses oleh petugas rekam medis. Admin dapat merubah (edit) biaya berobat pasien apabila terjadi kesalahan dalam menambahkan data. *Activity diagram* mengelola biaya berobat dapat dilihat pada Gambar 3.9.

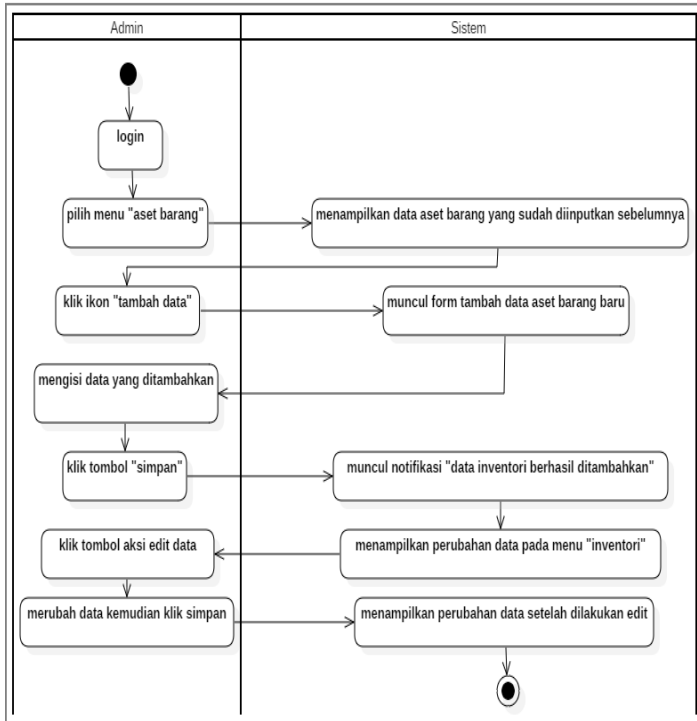


Gambar 3.9 Activity Diagram Mengelola Biaya Berobat Pasien

## 7) Activity Diagram Mengelola Aset

Pada halaman admin terdapat menu “Aset/Barang” yang berfungsi bagi admin untuk menambahkan data aset medis dan non medis yang ada di Puskesmas Tebuireng.

Admin dapat merubah (edit) data aset apabila terjadi kesalahan dalam menambahkan data. *Activity diagram* mengelola aset barang dapat dilihat pada Gambar 3.10.

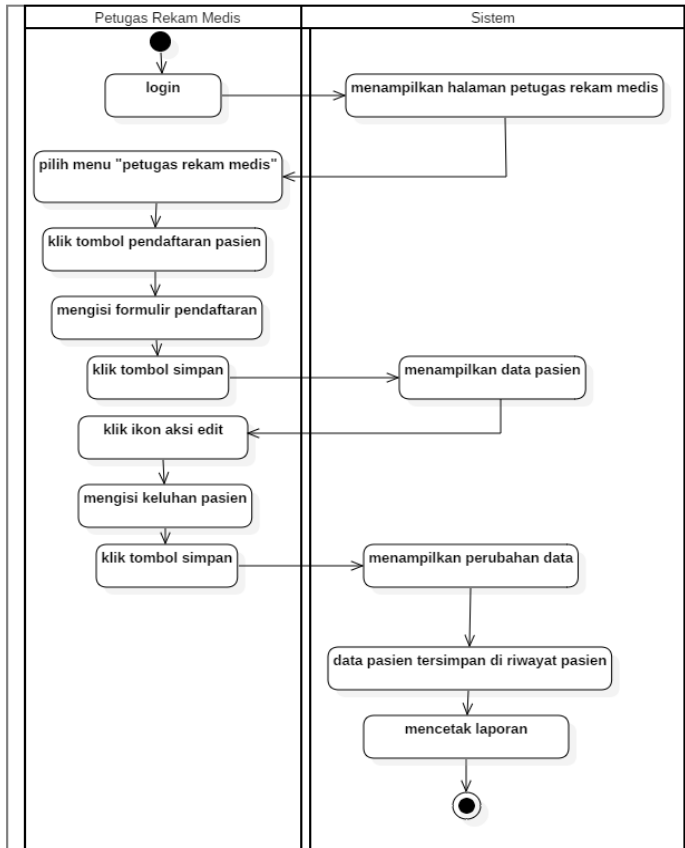


Gambar 3.10 *Activity Diagram* Mengelola Aset

## 8) *Activity Diagram* Rekam Medis

Aktifitas yang ada pada rekam medis dilakukan oleh petugas rekam medis dengan cara pilih menu "Rekam Medis" untuk melakukan pendaftaran pasien yang belum pernah mendaftar, petugas rekam medisi menambahkan identitas pasien di sistem beserta keluhan yang dialami oleh pasien. Petugas rekam medis dapat mencetak selebar buku rekam medis pasien untuk diserahkan ke dokter untuk dilakukan

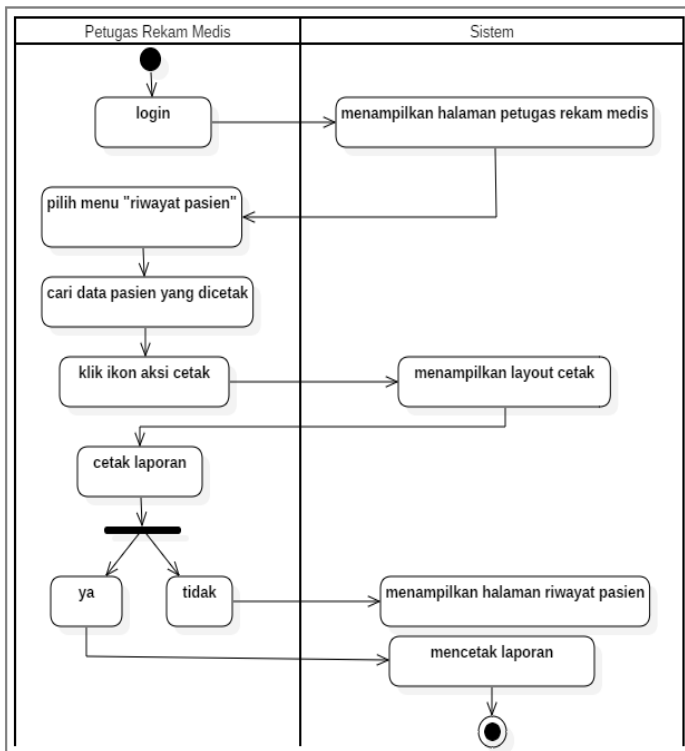
tindakan berupa diagnose penyakit yang dialami pasien sesuai dengan keluhan yang disampaikan pasien ke petugas rekam medis. Petugas rekam medis dapat merubah data pasien jika terjadi kesalahan dalam menambahkan data tersebut. *Activity diagram* rekam medis dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Activity Diagram* Rekam Medis

### 9) Activity Diagram Mengelola Laporan Riwayat Pasien

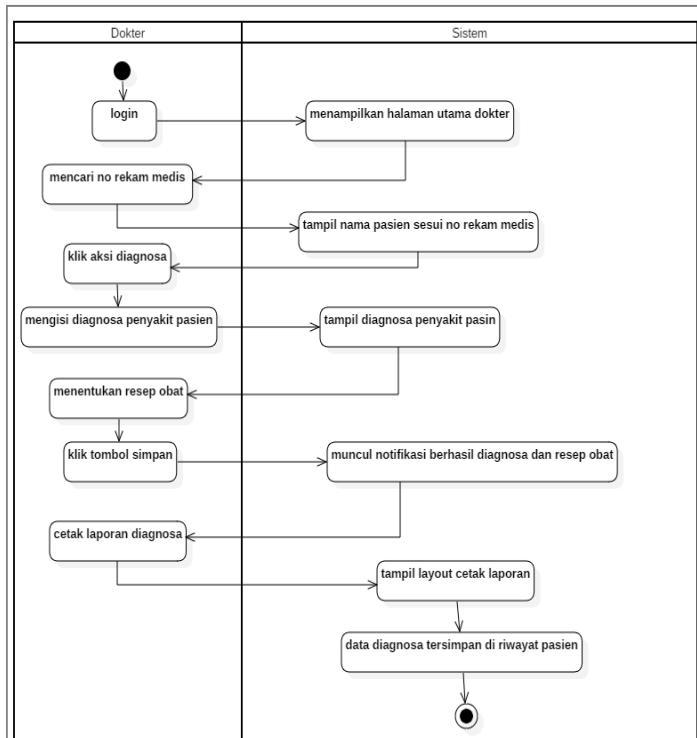
Menu riwayat pasien menampilkan data pasien yang terdaftar di sistem dan telah didiagnosa oleh dokter. Apabila pasien ingin melihat riwayatnya, petugas rekam medis dapat mencetak sesuai dengan permintaan pasien tersebut. Petugas rekam medis hanya dapat mencetak laporan, tidak untuk merubah data maupun menghapus data riwayat pasien. Jika pasien tersebut complain tentang identitasnya yang keliru, maka petugas rekam medis dapat merubahnya pada menu “Rekam Medis”. Activity diagram laporan dapat dilihat pada Gambar 3.12.



Gambar 3.12 Activity Diagram Mengelola Laporan Riwayat Pasien

### 10) Activity Diagram Diagnosa Pasien

Aktifitas ini menjelaskan bahwa dokter dapat melakukan diagnose penyakit pasien melalui sistem. Dokter dapat login terlebih dahulu kemudian memasukkan nomor rekam medis pasien, jika nomor tersebut valid maka dokter diarahkan pada halaman diagnose penyakit. Dokter dapat menentukan resep obat untuk pasien. Dokter dapat mencetak selebaran resep obat. Data hasil *input* dokter tersebut tersimpan dalam database dan tampil pada menu riwayat pasien yang diakses oleh petugas rekam medis. Dokter hanya dapat merubah isi diagnosa dan tidak dapat menghapus diagnose tersebut. *Activity diagram* diagnose penyakit dapat dilihat pada Gambar 3.13.



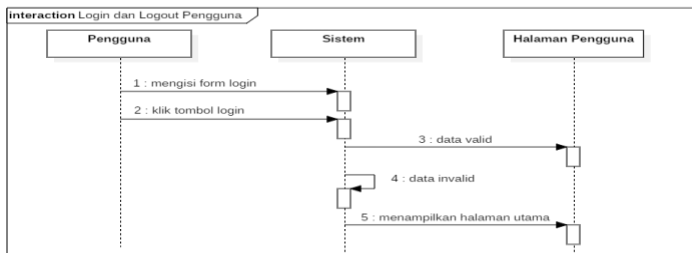
Gambar 3.13 Activity Diagram Diagnosa Pasien

### C. Sequence Diagram

Diagram ketiga adalah *sequence diagram* yang menunjukkan interaksi dengan menampilkan setiap partisipan sistem melalui garis vertikal dan pengurutan pesan dari atas ke bawah. Dalam perancangan untuk membangun sebuah Sistem Informasi Manajemen Puskestren Berbasis *Web* untuk masing-masing *use case* akan dibuat satu *sequence diagram*. Sehingga apabila dalam perangkat lunak terdapat 10 *use case* harus dapat pula ditemukan 10 *sequence diagram* diantaranya: *Sequence Diagram Login dan Logout*, *Sequence Diagram Mengelola Pengguna*, *Sequence Diagram Mengelola Data Pegawai*, *Sequence Diagram Mengelola Data Dokter*, *Sequence Diagram Mengelola Obat*, *Sequence Diagram Mengelola Biaya Berobat*, *Sequence Diagram Mengelola Inventori*, *Sequence Diagram Mengelola Rekam Medis*, *Sequence Diagram Mengelola Laporan*, *Sequence Diagram Diagnosa Penyakit*.

#### 1) Sequence Diagram Login

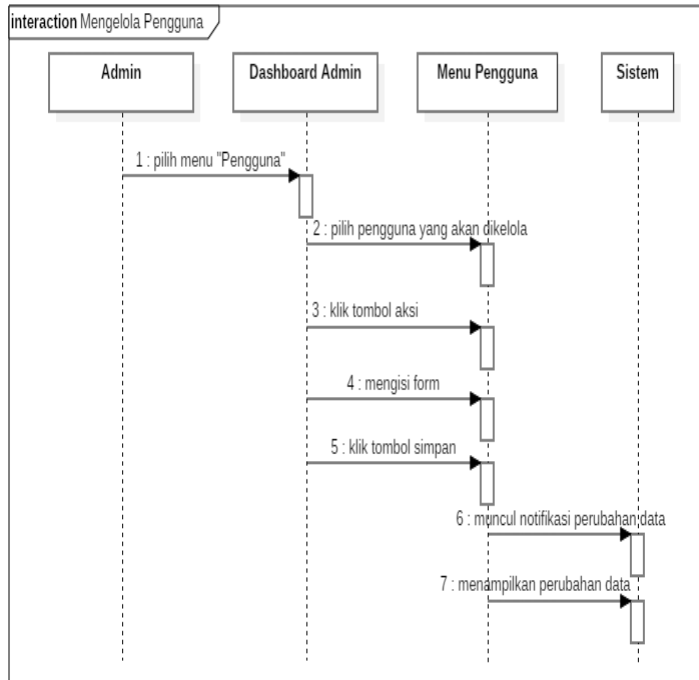
Sequence diagram login pengguna menggambarkan alur bahwa pengguna bisa mengakses sistem dengan terlebih dahulu login memasukkan username dan password. Apabila data yang dimasukkan di form login valid, maka pengguna dapat masuk ke halaman pengguna, jika data yang dimasukkan invalid maka pengguna kembali ke halaman login dan memasukkan kembali username dan password. Sequence diagram login dapat dilihat pada Gambar 3.14



Gambar 3.14 Sequence Diagram Login

## 2) *Sequence Diagram* Mengelola Pengguna

Sequence diagram mengelola pengguna menggambarkan alur bahwa admin bisa mengatur data *user*. Fungsi pada system ini yaitu tambah (create), ubah (edit), dan hapus (delete) akun pengguna yang tersimpan di dalam database. Sequence diagram mengelola pengguna dapat dilihat pada Gambar 3.15

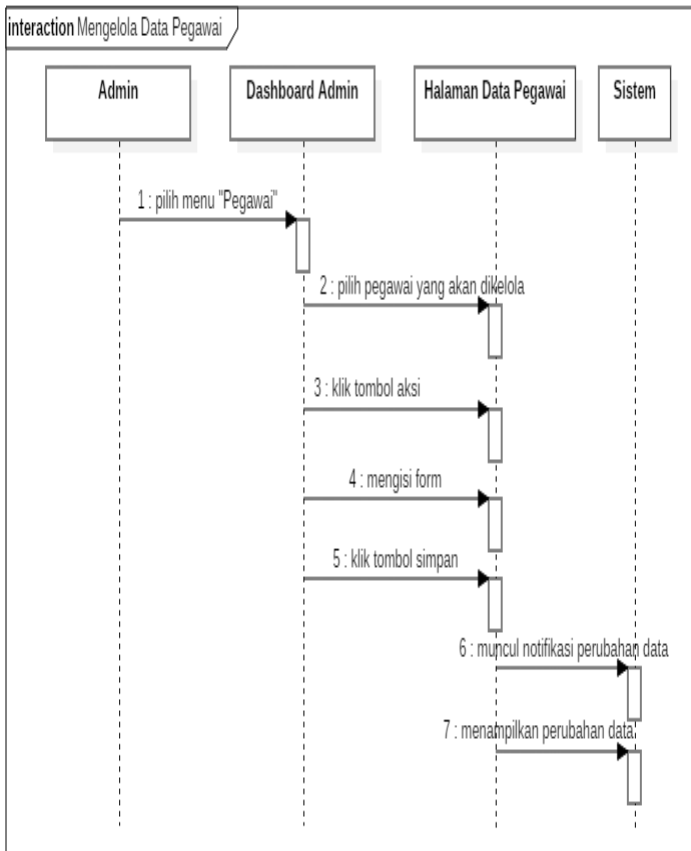


Gambar 3.15 *Sequence Diagram* Pengguna

## 3) *Sequence Diagram* Mengelola Pegawai

Sequence diagram mengelola pegawai menggambarkan bahwa admin dapat menambah data pegawai pada menu "Pegawai", merubah data pegawai seperti biodata pegawai dan menghapus data pegawai yang

sudah tidak bekerja di puskestren tersebut. Data pegawai termasuk dokter nantinya include pada menu “Dokter” yang dapat dikelola oleh admin menambahkan jam dokter sesuai dengan kebutuhan, hasil update jam dokter akan muncul pada menu “Rekam Medis” pada saat petugas rekam medis mengarahkan pasien pada saat pendaftar pasien. *Sequence diagram* mengelola data pegawai dapat dilihat pada Gambar 3.16.

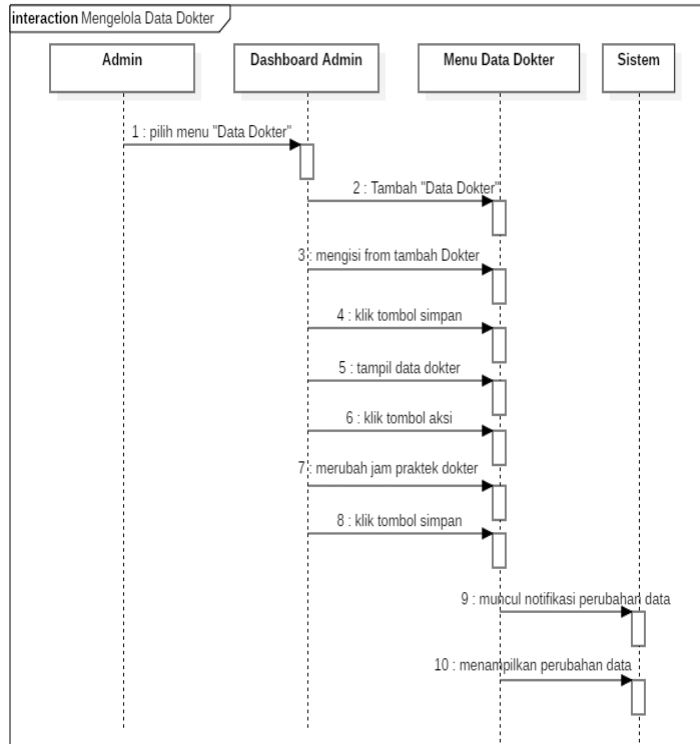


Gambar 3.16 Sequence Diagram Mengelola Pegawai



#### 4) **Sequence Diagram Mengelola Dokter**

*Sequence diagram* mengelola dokter menggambarkan alur menambahkan waktu praktek dokter sesuai dengan bidangnya, admin dapat merubah waktu praktek dokter sesuai ketentuan yang berlaku. *Sequence diagram* mengelola dokter dapat dilihat pada Gambar 3.17.

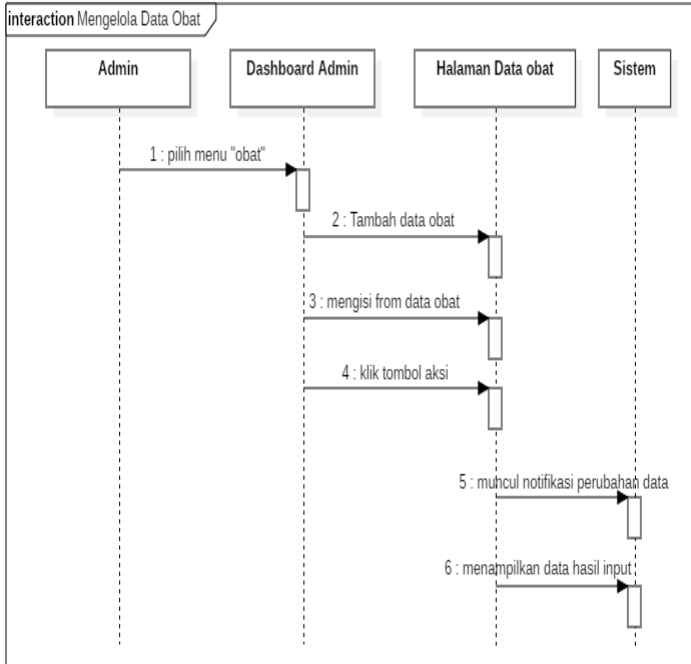


Gambar 3.17 *Sequence Diagram* Mengelola Dokter

#### 5) **Sequence Diagram Mengelola Obat**

*Sequence diagram* mengelola data obat menggambarkan bahwa pada halaman admin terdapat menu “Obat” yang berfungsi menambahkan data obat sesuai

dengan kebutuhan, merubah data obat dan menghapus data obat. Hasil tambah data maupun ubah data akan tampil (*include*) pada halaman diagnose dokter yang diakses oleh dokter. *Sequence diagram* mengelola data obat dapat dilihat pada Gambar 3.18.

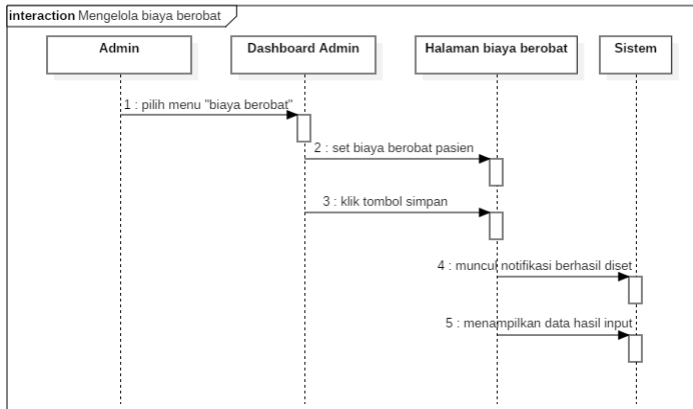


Gambar 3.18 *Sequence Diagram* Mengelola Obat

## 6) *Sequence Diagram* Mengelola Biaya Berobat Pasien

*Sequence diagram* mengelola biaya berobat pada halaman admin terdapat menu “Biaya Berobat” yang berfungsi untuk menambahkan biaya berobat pasien yang nantinya keluar pada menu rekam medis yang diakses oleh petugas rekam medis. Admin dapat merubah (edit) biaya berobat pasien apabila terjadi kesalahan dalam menambahkan data.

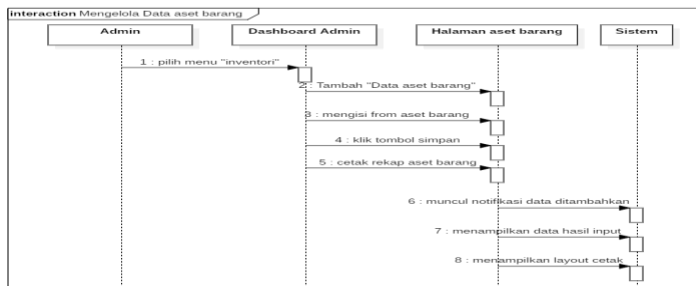
*Sequence diagram* mengelola biaya berobat dapat dilihat pada Gambar 3.19.



Gambar 3.19 Sequence Diagram Mengelola Biaya Berobat

## 7) Sequence Diagram Mengelola Aset

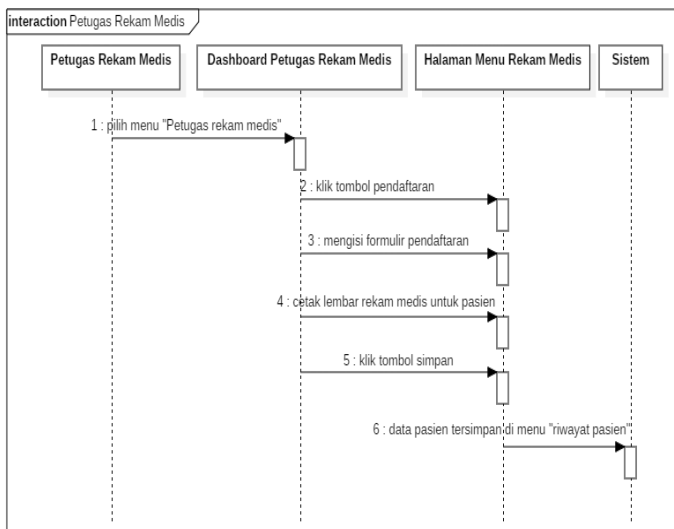
Menggambarkan alur pada halaman admin terdapat menu “Aset/Barang” yang berfungsi bagi admin untuk menambahkan data aset medis dan non medis yang ada di puskestren tebuireng. Admin dapat merubah (edit) data aset apabila terjadi kesalahan dalam menambahkan data. *Sequence diagram* mengelola aset barang dapat dilihat pada Gambar 3.20



Gambar 3.20 Sequence Diagram Mengelola Aset

### 8) *Sequence Diagram* Rekam Medis

Menggambarkan alur yang ada pada rekam medis dilakukan oleh petugas rekam medis dengan cara pilih menu “Rekam Medis” untuk melakukan pendaftaran pasien yang belum pernah mendaftar, petugas rekam medis menambahkan identitas pasien di sistem beserta keluhan yang dialami oleh pasien. Petugas rekam medis dapat mencetak selebar buku rekam medis pasien untuk diserahkan ke dokter untuk dilakukan tindakan berupa diagnose penyakit yang dialami pasien sesuai dengan keluhan yang disampaikan pasien ke petugas rekam medis. Petugas rekam medis dapat merubah data pasien jika terjadi kesalahan dalam menambahkan data tersebut. *Sequence diagram* rekam medis dapat dilihat pada Gambar 3.21.

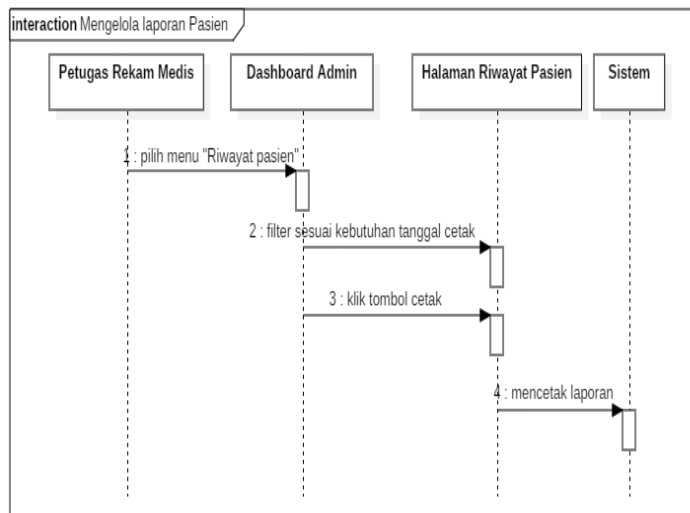


Gambar 3.21 *Sequence Diagram* Mengelola Rekam Medis

### 9) *Sequence Diagram* Mengelola Laporan Riwayat Pasien

*Sequence diagram* mengelola laporan menggambarkan alur untuk mencetak laporan riwayat pasien.

Pada menu riwayat pasien menampilkan data pasien yang terdaftar di sistem dan telah didiagnosa oleh dokter. Apabila pasien ingin melihat riwayatnya, petugas rekam medis dapat mencetak sesuai dengan permintaan pasien tersebut. Petugas rekam medis hanya dapat mencetak laporan, tidak untuk merubah data maupun menghapus data riwayat pasien. Jika pasien tersebut complain tentang identitasnya yang keliru, maka petugas rekam medis dapat merubahnya pada menu “Rekam Medis”. *Sequence diagram* laporan dapat dilihat pada Gambar 3.22.

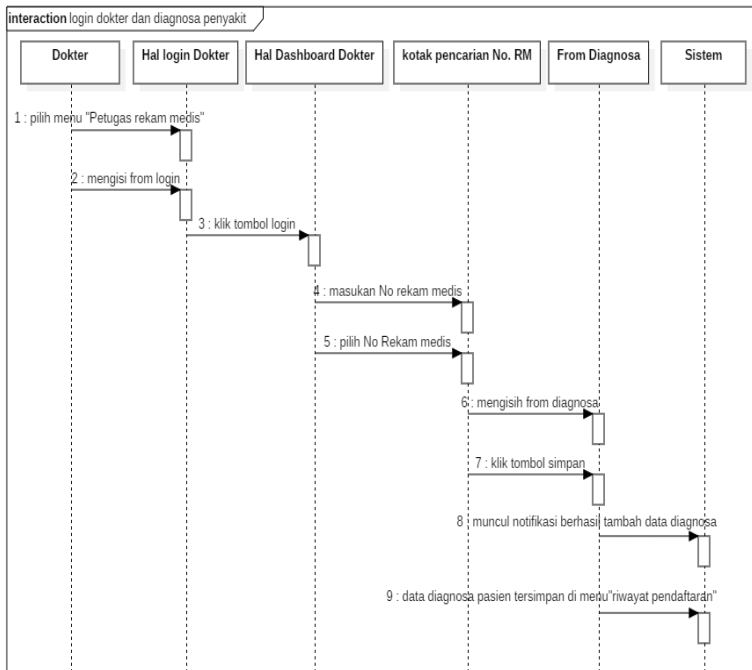


Gambar 3.22 *Sequence Diagram* Mengelola Laporan Riwayat Pasien

#### 10) *Sequence Diagram* Diagnosa Pasien

*Sequence diagram* diagnose penyakit menggambarkan alur bahwa dokter dapat melakukan diagnose penyakit pasien melalui sistem. Dokter dapat login terlebih dahulu kemudian memasukkan nomor rekam medis pasien, jika nomor tersebut valid maka dokter diarahkan pada halaman diagnose penyakit. Dokter dapat menentukan resep

obat untuk pasien. Dokter dapat mencetak selebaran resep obat. Data hasil *input* dokter tersebut tersimpan dalam database dan tampil pada menu riwayat pasien yang diakses oleh petugas rekam medis. Dokter hanya dapat merubah isi diagnosa dan tidak dapat menghapus diagnose tersebut. *Sequence diagram* diagnose penyakit dapat dilihat pada Gambar 3.23.

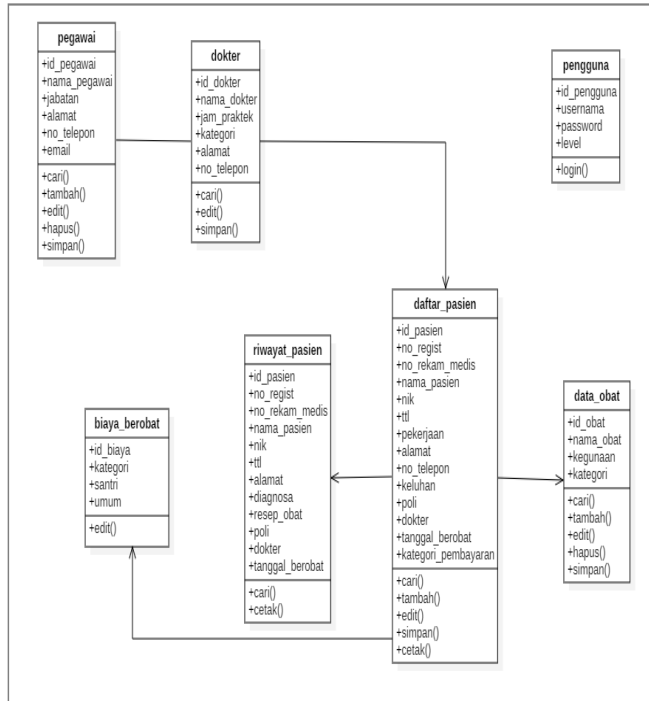


Gambar 3.23 *Sequence Diagram* Mengelola Diagnosa Penyakit

#### D. *Class Diagram*

Diagram keempat adalah *class diagram* model yang menggambarkan struktur dan deskripsi *class* serta hubungan antar *class*. *Class diagram* mirip dengan *entity relationship diagram* pada perancangan. Perbedaannya *entity relationship diagram* tidak terdapat operasi atau metode, melainkan hanya

terdapat atribut. *Class* terdiri dari nama kelas, atribut. *Class* yang terdiri pada perancangan sistem informasi manajemen puskestren ini meliputi: pegawai, dokter, pengguna, biaya berobat, riwayat pasien, daftar pasien, dan data obat. *Class diagram* Sistem Informasi Manajemen Puskestren Tebuireng dapat dilihat pada Gambar 3.24.



Gambar 3.24 *Class Diagram* Sistem Informasi Manajemen Puskestren

## E. Desain Perancangan Basis Data

Desain sistem yang akan dibangun ini dapat disimpulkan bahwa data-data yang akan digunakan dalam membangun Sistem Informasi Manajemen Puskestren Tebuireng yaitu desain perancangan basis data berbentuk tabel-tabel yang dikelompokkan berdasarkan fungsi masing-

masing basis data. Kebutuhan tabel basis data dapat dilihat pada Tabel 3.9.

*Tabel 3.9 Desain Perancangan Basis Data*

No	Tabel	Keterangan
1)	Pengguna	Tabel ini berisi data pengguna aplikasi
2)	Pegawai	Tabel ini berisi data pegawai
3)	Dokter	Tabel ini berisi data dokter beserta jam praktek
4)	Obat	Tabel ini berisi data obat
5)	Biaya Berobat	Tabel ini berisi rincian biaya berobat untuk pasien
6)	Daftar Pasien	Tabel ini berisi data pendaftaran pasien dan catatan rekam medis pasien
7)	Riwayat Pasien	Tabel ini berisi riwayat pasien yang sudah mendaftar dan berobat

Penjelasan mengenai Tabel 3.9. dapat dilihat pada rincian sebagai berikut:

### **1) Tabel Pengguna**

Tabel pengguna berisi data-data pengguna yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel pengguna berfungsi untuk menyimpan akun pengguna yang sudah dibuatkan oleh admin. Sebelum masuk ke dalam sistem, pengguna diharuskan terlebih dahulu untuk login supaya dapat melanjutkan akses lebih luas. Dalam tabel pengguna ini terdapat atribut yang terdiri dari *id\_pengguna*, *username*, *password*, *level*.



Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.10.

Tabel 3.10 Desain Perancangan Basis Data Tabel Pengguna

#	Atribut	Type	Keterangan
1	id_pengguna	int(10)	*auto increment *primary key
2	username	varchar(50)	-
3	password	varchar(35)	md5
4	level	varchar(25)	-

## 2) Tabel Pegawai

Tabel pegawai berisi data-data pegawai yang ada yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel pegawai berfungsi untuk menyimpan data pegawai yang ada di Puskestren Tebuireng berupa identitas atau biodata pegawai termasuk dokter. Dalam tabel pegawai terdapat atribut diantaranya: id\_pegawai, nama\_pegawai, jabatan, alamat, no\_telepon, email. Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.11.

Tabel 3.11 Desain Perancangan Basis Data Tabel Pegawai

#	Atribut	Type	Keterangan
1	id_pegawai	int(10)	*auto increment *primary key
2	nama_pegawai	varchar(50)	-
3	jabatan	varchar(30)	-
4	alamat	varchar(200)	-
5	no_telepon	varchar(15)	-
6	email	varchar(30)	-

### 3) Tabel Dokter

Tabel dokter berisi data-data dokter yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel dokter berfungsi untuk menyimpan data dokter yang di ambil dari data pegawai dalam tabel dokter terdapat atribut diantaranya: id\_dokter, nama\_dokter, jam\_peraktek, kategori, no\_telepon, alamat. Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.12.

Tabel 3.12 Desain Perancangan basis Data Tabel Dokter

#	Atribut	Type	Keterangan
1	id_dokter	Int(10)	*auto increment
2	nama_dokter	varchar(50)	-
3	jam_praktek	Date	-
4	kategori	varchar(20)	-
5	no_telepon	varchar(20)	-
6	alamat	varchar(40)	-

### 4) Tabel Obat

Tabel obat berisi data-data obat yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel obat berfungsi untuk menyimpan data obat sesuai dengan kategori penyakitnya, table ini mempunyai relasi dengan table riwayat pasien. Admin menambahkan data obat yang nantinya muncul pada halaman dokter yang dapat diakses oleh dokter untuk mendiagnosa pasien dan menentukan resep obat. Jika terdapat perubahan data obat dapat di rubah oleh admin. Dalam tabel obat ini terdapat atribut yang terdiri dari: id\_obat, nama\_obat, kegunaan, kategori.

Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.13.

Tabel 3.13 Desain Perancangan Basis Data Tabel Obat

#	Atribut	Type	Keterangan
1	id_obat	int(10)	*auto increment *primary key
2	nama_obat	varchar(20)	-
3	kegunaan	varchar(30)	-
4	kategori	varchar(15)	-

### 5) Tabel Biaya Berobat

Tabel biaya berobat berisi data-data biaya berobat yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel biaya berobat berfungsi untuk menyimpan tagian biaya berobat pasien. Dalam tabel biaya berobat ini terdapat atribut yang terdiri dari: id\_biaya, kategori, santri, umum. Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.14.

Tabel 3.14 Desain Perancangan Basis Data Tabel Biaya Berobat

#	Atribut	Type	Keterangan
1	id_biaya	int(10)	*auto increment *primary key
2	kategori	varchar(15)	-
3	santri	varchar(20)	-
4	umum	varchar(20)	-

## 6) Tabel Daftar Pasien

Tabel daftar pasien berisi data-data daftar pasien yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel daftar pasien berfungsi untuk menyimpan data pasien, table ini mempunyai relasi dengan riwayat pasien. Dalam tabel daftar pasien ini terdapat atribut yang terdiri dari: id\_pasien, no\_regist, no\_rekam\_medis, nama\_pasien, nik, ttl, pekerjaan, alamat, no\_telepon, keluhan, poli, dokter, tanggal\_berobat, kategori\_pembayaran. Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.15.

Tabel 3.15 Desain Perancangan Basis Data Tabel Daftar

#	Atribut	Type	Keterangan
1	id_pasien	int(10)	*auto increment *primary key
2	no_regist	int(70)	-
3	no_rekam_medis	int(70)	-
4	nama_pasien	varchar(20)	-
5	nik	int(25)	-
6	ttl	Date	-
7	pekerjaan	varchar(20)	-
8	Alamat	varchar(100)	-
9	no_telepon	varchar(15)	-
10	Keluhan	varchar(150)	-
11	Poli	varchar(50)	-
12	Dokter	varchar(40)	-
13	tanggal_berobat	Vate	-
14	kategori_pembayaran	Varchar(50)	-

## 7) Tabel Riwayat pasien

Tabel riwayat pasien berisi data-data riwayat pasien yang dapat melakukan pengelolaan data seperti *create*, *read*, *update* dan *delete*. Tabel riwayat pasien berfungsi untuk menyimpan data pasien. Hasil *input* petugas rekam medis dan hasil diagnose pasien akan tersimpan dalam table ini. Dalam tabel riwayat pasien ini terdapat atribut yang terdiri dari: *id\_pasien*, *no\_regist*, *no\_rekam\_medis*, *nama\_pasien*, *nik*, *tgl*, *alamat*, *diagnosa*, *resep\_obat*, *poli*, *dokter*, *tanggal\_berobat*. Adapun penjabaran tabelnya, dapat dilihat pada Tabel 3.16.

Tabel 3.16 Desain Perancangan Basis Data Tabel Riwayat Pasien

#	Atribut	Type	Keterangan
1	<i>id_pasien</i>	int(10)	*auto increment *primary key
2	<i>no_regist</i>	int(70)	-
3	<i>no_rekam_medis</i>	int(70)	-
4	<i>nama_pasien</i>	varchar(20)	-
5	<i>nik</i>	int(25)	-
6	<i>tgl</i>	date	-
7	<i>alamat</i>	varchar(100)	-
8	<i>diagnosa</i>	varchar(100)	-
9	<i>resep_obat</i>	varchar(80)	-
10	<i>poli</i>	varchar(50)	-
11	<i>dokter</i>	varchar(40)	-
12	<i>tanggal_berobat</i>	date	-

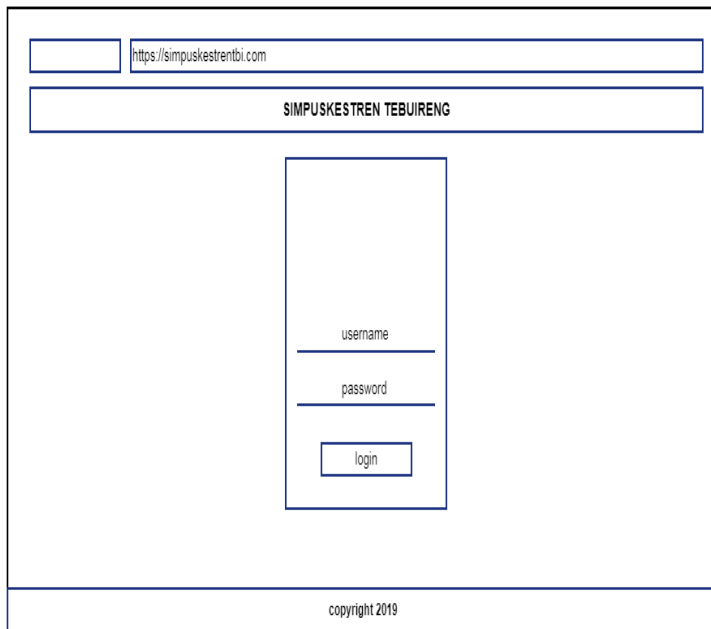
### 3.2.4 Perancangan User Interface (Antarmuka Pengguna)

Perancangan *user interface* (antarmuka pengguna) digunakan untuk kendali operasi dan memasukkan data dalam

aplikasi. Terdapat 10 rancangan *user interface* (antarmuka pengguna) Sistem Informasi Manajemen Puskestren Berbasis *Web* diuraikan sebagai berikut:

### 1) *User Interface Login*

Menggambarkan bahwa perancangan *login* ini digunakan oleh pengguna keluar masuk pada sistem. Pengguna memasukkan username dan password pada form login dan sistem memverifikasi kebenaran data jika data yang dimasukkan tidak valid, maka pengguna kembali ke halaman login. Apabila data valid, maka pengguna masuk ke halaman pengguna sesuai level masing-masing pengguna. *User interface login* dapat dilihat pada Gambar 3.25.

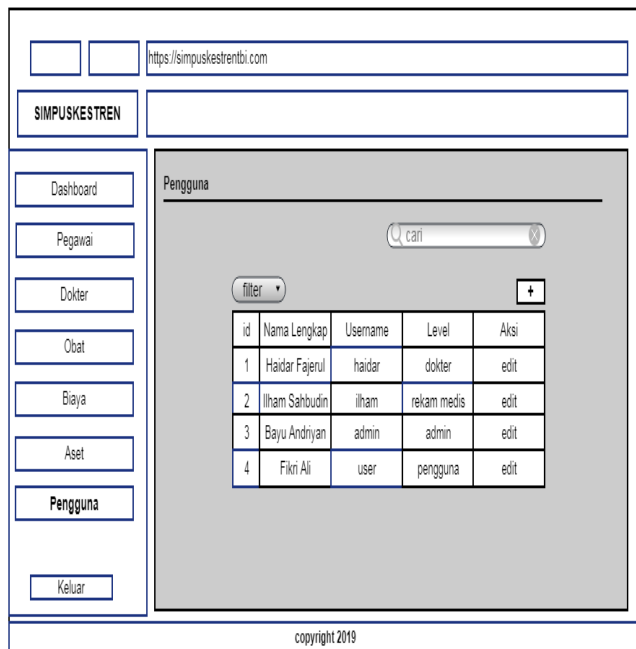


The image shows a web-based login interface for 'SIMPUSKESTREN TEBUIRENG'. At the top left, there is a small square icon next to the URL 'https://simpuskestrentbi.com'. Below the URL bar is a header bar with the text 'SIMPUSKESTREN TEBUIRENG'. The main content area contains a login form with two input fields labeled 'username' and 'password', and a 'login' button below them. At the bottom of the page, there is a footer bar with the text 'copyright 2019'.

Gambar 3.25 *User Interface Login*

## 2) User Interface Mengelola Pengguna

Menggambarkan bahwa admin dapat menambah (create), merubah (edit), menghapus (delete) akun pengguna yang sudah terdaftar akun di database. Semua dapat dilakukan pada menu “Pengguna”. *User interface* mengelola pengguna dapat dilihat pada Gambar 3.26.



Gambar 3.26 User Interface Pengguna

## 3) User Interface Mengelola Pegawai

Tambah (*create*), ubah (*edit*), hapus (*delete*) dapat dilakukan pada halaman admin tepatnya menu “Pegawai”. Admin dapat menambah data pegawai di puskestren, merubah data pegawai seperti biodata pegawai dan

menghapus data pegawai yang sudah tidak bekerja di puskestren tersebut. *User interface* mengelola data pegawai dapat dilihat pada Gambar 3.27.

The screenshot shows the SIMPUKSTREN web application. At the top, there is a URL bar with 'https://simpukstrentnbi.com'. Below it, the application name 'SIMPUKSTREN' is displayed. The sidebar on the left contains the following menu items: Dashboard, Pegawai, Dokter, Obat, Biaya, Aset, Pengguna, and Keluar. The main content area is titled 'Pegawai' and features a search bar with the placeholder text 'cari'. Below the search bar is a 'filter' dropdown menu and a '+' button. The main area contains a table with the following data:

id	Nama pegawai	Alamat	No telpon	Jabatan	Aksi
1	Hardar Fajenul	Tebuireng	08570023134	Dokter	edit
2	Ilham Sahbudin	Cukir	08195824167	Pegawai	edit
3	Bayu Andriyan	Tebon	08387298147	Apoteker	edit
4	Fikri Ali	Kayen	08741241248	Perawat	edit

At the bottom of the page, the text 'copyright 2019' is visible.

Gambar 3.27 User Interface Mengelola Pegawai

#### 4) User Interface Mengelola Dokter

Menggambarakan alur menambahkan waktu praktek dokter sesuai dengan bidangnya, admin login terlebih dahulu

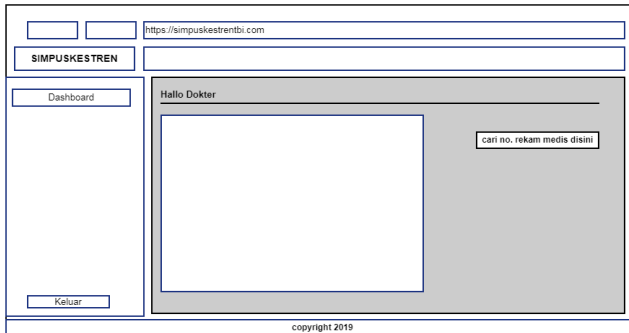


ke dalam sistem, setelah login memilih menu “Dokter”, admin tidak bisa menambahkan dokter secara langsung pada menu ini, admin dapat menambahkan pada menu “Pegawai” dikarenakan isian dokter pada menu ini secara otomatis include pada saat admin menambahkan pegawai. Admin dapat merubah waktu praktek dokter sesuai ketentuan yang berlaku, semua dilakukan oleh admin pada menu “Dokter”. *User interface* mengelola dokter dapat dilihat pada Gambar 3.28 dan 3.29.

The screenshot displays the SIMPUKSTREN web application. On the left is a sidebar menu with buttons for Dashboard, Pegawai, Dokter, Obat, Biaya, Aset, Pengguna, and Keluar. The main area is titled 'Dokter' and contains a search bar with the placeholder 'cari'. Below the search bar is a table with columns: id, Nama Dokter, Alamat, Jam Praktek, Kategori, and Aksi. The table lists four doctors. Above the table is a 'filter' dropdown and a '+' button. At the bottom of the page, it says 'copyright 2019'.

id	Nama Dokter	Alamat	Jam Praktek	Kategori	Aksi
1	Haider Fajenul	Keras	08.00-12.00	dokter Gigi	edit
2	Ilham Sahbudin	Cukir	10.00-16.00	Dokter Beda	edit
3	Bayu Andriyan	Kayangan	11.00-17.00	Dokter Jantung	edit
4	Fikri Ali	Tebuireng	14.00-21.00	Dokter Umum	edit

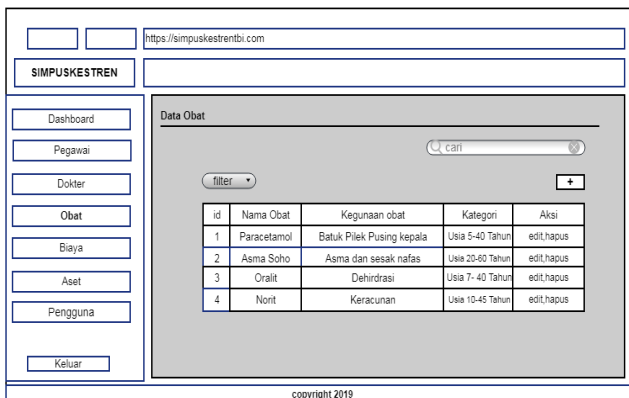
Gambar 3.28 *User Interface* Mengelola Dokter



Gambar 3.29 User Interface Halaman Dokter

### 5) User Interface Mengelola Obat

Menu “Obat” pada halaman admin menggambarkan bahwa admin dapat menambahkan data obat sesuai dengan kebutuhan, merubah data obat dan menghapus data obat. Hasil tambah data maupun ubah data akan tampil (include) pada halaman diagnose dokter yang diakses oleh dokter. *User interface* mengelola obat dapat dilihat pada Gambar 3.30.



Gambar 3.30 User Interface Mengelola Obat

### 6) *User Interface* Mengelola Biaya Berobat Pasien

Pada halaman admin terdapat menu “Biaya Berobat Pasien” yang berfungsi untuk menambahkan biaya berobat pasien yang nantinya keluar pada menu rekam medis yang diakses oleh petugas rekam medis. Admin dapat merubah (edit) biaya berobat pasien apabila terjadi kesalahan dalam menambahkan data. *User Interface* mengelola biaya berobat dapat dilihat pada Gambar 3.31.

The screenshot displays the 'Biaya berobat pasien' (Patient Treatment Cost) management interface. On the left is a sidebar with navigation buttons: Dashboard, Pegawai, Dokter, Obat, **Biaya** (highlighted), Aset, Pengguna, and Keluar. The main area is titled 'Biaya berobat pasien' and features a search bar labeled 'cari' and a 'filter' dropdown. Below these is a table with the following data:

id	Kategori	Santri	Umum	Aksi
1	Poli Umum	15.000	30.000	edit
2	Poli gigi	10.000	25.000	edit

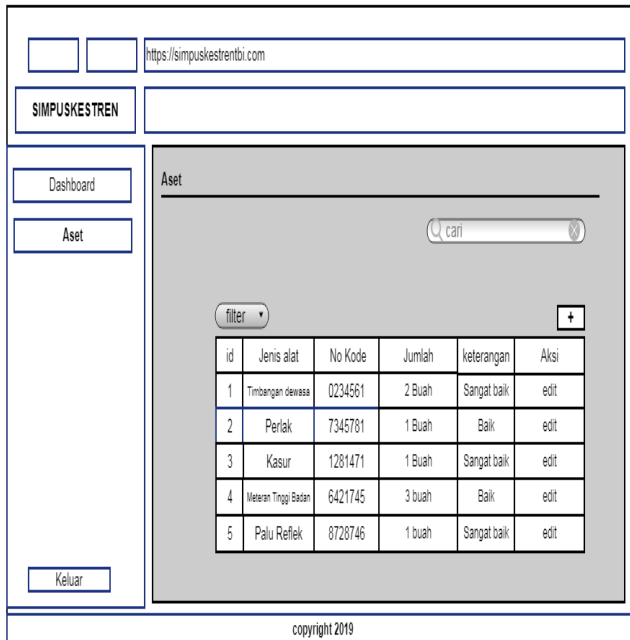
The footer of the interface shows 'copyright 2019'.

Gambar 3.31 *User Interface* Mengelola Biaya Berobat

### 7) *User Interface* Mengelola Aset

Pada halaman admin terdapat menu “Aset/Barang” yang berfungsi bagi admin untuk menambahkan data aset medis dan non medis yang ada di puskestren Tebuireng.

Admin dapat merubah (edit) data aset apabila terjadi kesalahan dalam menambahkan data. *User Interface* mengelola aset barang dapat dilihat pada Gambar 3.32



*Gambar 3.32 User Interface Mengelola Aset Barang*

## 8) *User Interface* Rekam Medis

Aktifitas yang ada pada rekam medis dilakukan oleh petugas rekam medis dengan cara pilih menu “Rekam Medis” untuk melakukan pendaftaran pasien yang belum pernah mendaftar, petugas rekam medisi menambahkan identitas pasien di sistem beserta keluhan yang dialami oleh pasien. Petugas rekam medis dapat mencetak selembaar buku rekam medis pasien untuk diserahkan ke dokter untuk dilakukan tindakan berupa diagnose penyakit yang dialami pasien

sesuai dengan keluhan yang disampaikan pasien ke petugas rekam medis. Petugas rekam medis dapat merubah data pasien jika terjadi kesalahan dalam menambahkan data tersebut. *User interface* rekam medis dapat dilihat pada Gambar 3.33.

The image shows a web-based user interface for a medical record system. On the left is a vertical sidebar with four buttons: 'Dashboard', 'Rekam medis', 'Riwayat pasien', and 'Keluar'. The 'Rekam medis' button is highlighted. The main area of the page is titled 'Rekam medis' and contains a sub-section 'Formulir Pendaftaran Pasien'. This form has six input fields arranged in two columns: 'No. Pendaftaran', 'No. Rekam medis', 'Nama pasien', 'Tempat tanggal lahir', 'Alamat', and 'NIK'. Below these fields is a 'Simpan' button. At the top right of the main area, there is a URL 'https://simpukstrentbi.com'. At the bottom center, it says 'copyright 2019'.

Gambar 3.33 *User Interface* Rekam Medis

## 9) *User Interface* Mengelola Laporan Riwayat Pasien

Menggambarkan bahwa petugas rekam medis dapat mengakses menu riwayat pasien menampilkan data pasien yang terdaftar di sistem dan telah didiagnosa oleh dokter. Apabila pasien ingin melihat riwayatnya, petugas rekam medis dapat mencetak sesuai dengan permintaan pasien tersebut.

Petugas rekam medis hanya dapat mencetak laporan, tidak untuk merubah data maupun menghapus data riwayat pasien. Jika pasien tersebut komplain tentang identitasnya yang keliru, maka petugas rekam medis dapat merubahnya pada menu “Rekam Medis”. Petugas rekam medis dapat mencetak selebar buku hasil pengisian data pasien yang sudah daftar nantinya diserahkan ke dokter untuk dilakukan diagnosa, data hasil isian petugas rekam medis akan tersimpan dalam database. *User interface* laporan dapat dilihat pada Gambar 3.34.

The screenshot shows a web application interface for 'SIMPUKSTREN'. At the top, there is a URL bar showing 'https://simpukstrentbi.com'. Below it, a sidebar contains navigation buttons: 'Dashboard', 'Rekam medis', 'Riwayat pasien', and 'Keluar'. The main content area is titled 'Data Riwayat Pasien' and features a search bar with the placeholder 'cari', a 'filter' dropdown menu, and a 'Cetak' button. A table displays patient history data with the following columns: 'id', 'No. Regis', 'No. Rekam medis', 'Nama Pasien', 'Keterangan', and 'Tanggal Berobat'.

id	No. Regis	No. Rekam medis	Nama Pasien	Keterangan	Tanggal Berobat
1	001	987	Haidar fajrul	<a href="#">lihat diagnosa</a>	23-12-2018
2	002	988	Ilham sabudin	<a href="#">lihat diagnosa</a>	13-09-2019
3	003	718	Fikri ali	<a href="#">lihat diagnosa</a>	17-10-2019
4	004	123	Bayu Ardian	<a href="#">lihat diagnosa</a>	29-04-2017

copyright 2019

Gambar 3.34 *User Interface* Mengelola Laporan Riwayat Pasien

### 10) *User Interface Diagnosa Pasien*

Menggambarkan alur bahwa dokter dapat melakukan diagnose penyakit pasien melalui sistem. Dokter dapat login terlebih dahulu kemudian memasukkan nomor rekam medis pasien, jika nomor tersebut valid maka dokter diarahkan pada halaman diagnose penyakit. Dokter dapat menentukan resep obat untuk pasien. Dokter dapat mencetak selebaran resep obat. Data hasil *input* dokter tersebut tersimpan dalam database dan tampil pada menu riwayat pasien yang diakses oleh petugas rekam medis. Dokter hanya dapat merubah isi diagnosa dan tidak dapat menghapus diagnose tersebut. *User interface* diagnose pasien dapat dilihat pada Gambar 3.35.

The screenshot displays a web-based user interface for patient diagnosis. At the top, there is a header section containing a logo on the left, a text input field with the URL 'https://simpuskestrentbi.com' in the center, and a navigation bar on the right with the text 'SIMPUSKESTREN'. Below the navigation bar, there is a sidebar menu with a button labeled 'Diagnosa Pasien'. The main content area is titled 'Diagnosa Pasien' and features a large gray box labeled 'Diagnosa'. Inside this box, there are three text input fields labeled 'Isi Diagnosa', 'Resep', and 'Tanggal'. Below these fields are two buttons: 'Simpan' (Save) and 'Cetak' (Print). At the bottom left of the sidebar, there is a 'Keluar' (Logout) button. The footer of the page contains the text 'copyright 2019'.

Gambar 3.35 *User Interface Diagnosa Pasien*

## **BAB 4**

### **TESTING DAN IMPLEMENTASI SISTEM**

Hasil penelitian memuat sesuatu yang anda buat berdasarkan analisis permasalahan pada Bab III yang disesuaikan dengan tema yang dipilih.

Sebagai contoh jika mahasiswa mengambil **tema rancang bangun** sistem maka pembahasan dalam bab ini adalah "Testing dan Implementasi Sistem ". Bab IV pada tema ini akan memaparkan hasil-hasil dari tahapan penelitian, dari tahap analisis, perancangan, hasil testing dan implementasinya, berupa penjelasan teoritik, baik secara kualitatif, kuantitatif, atau secara statistik. Selain itu, sebaiknya hasil penelitian juga dibandingkan dengan hasil penelitian terdahulu yang sejenis atau keadaan sebelumnya.

#### **4.1 Lingkungan Pengembangan**

Lingkungan pengembangan adalah spesifikasi perangkat keras maupun lunak yang digunakan selama proses pembangunan perangkat lunak berlangsung.

##### **4.1.1 Perangkat Keras**

Komponen yang dijelaskan antara lain *Harddisk*, RAM, Processor.

##### **4.1.2 Perangkat Lunak**

Komponen yang dijabarkan antara lain OS dan perangkat lunak aplikasi yang digunakan.

#### **4.2 Lingkungan Implementasi**

Lingkungan implementasi adalah spesifikasi perangkat dimana luaran perangkat lunak dari penelitian



diharapkan akan beroperasi. Komponen yang dijelaskan sama dengan lingkungan pengembangan.

#### **4.2.1 Perangkat Keras**

Komponen yang dijelaskan antara lain *Harddisk*, RAM, *Processor*.

#### **4.2.2 Perangkat Lunak**

Komponen yang dijabarkan antara lain OS dan perangkat lunak aplikasi yang digunakan.

### **4.3 Testing/Pengujian Kode Program dan Implementasi Sistem**

Testing/Pengujian ditujukan untuk memastikan bahwa tiap *use case* (fitur) telah dapat berjalan sebagaimana mestinya. Jumlah test case paling tidak satu untuk masing-masing *use case*.

#### **4.3.1 Pengujian Use Case Login**

Pada bagian ini dijabarkan mengenai hasil uji coba *use case* login. Hasil uji coba disertai dengan Screenshot perangkat lunak.

#### **4.3.2 Pengujian Use Case ...**

Pada bagian ini dijabarkan mengenai hasil uji coba *use case* yang lain.

#### **4.4 Implementasi Basis Data**

Pada bagian ini tunjukkan hasil implementasi dalam database yang digunakan.

## **BAB 5**

### **PENUTUP**

Bab ini berisi pokok-pokok kesimpulan dan saran-saran yang perlu disampaikan kepada pihak-pihak yang berkepentingan dengan hasil penelitian.

#### **5.1 Kesimpulan**

Kesimpulan dapat mengemukakan kembali masalah penelitian (mampu menjawab pertanyaan dalam rumusan masalah), menyimpulkan bukti-bukti yang diperoleh dan akhirnya menarik kesimpulan apakah hasil yang didapat (dikerjakan), layak untuk digunakan (diimplementasikan). Penulis tidak diperkenankan menyimpulkan masalah jika pembuktian tidak terdapat dalam hasil penelitian. Hal-hal yang diperkuat :

- 1) Berhubungan dengan apa yang dikerjakan
- 2) Didasarkan pada analisis yang objektif
- 3) Bukti-bukti yang telah ditemukan

#### **5.2 Saran**

Saran merupakan manifestasi dari penulis untuk dilaksanakan (sesuatu yang belum ditempuh dan layak untuk dilaksanakan). Saran dicantumkan karena peneliti melihat adanya jalan keluar untuk mengatasi masalah (kelemahan yang ada), saran yang diberikan tidak terlepas dari ruang lingkup penelitian (untuk objek penelitian maupun pembaca yang akan mengembangkan hasil penelitian).



## DAFTAR PUSTAKA

Penulisan daftar pusaka menggunakan format APA 5  
(*American Psychological Assosiation*).