

BAB 2

KAJIAN PUSTAKA

2.1 Rumah Sakit

2.1.1 Definisi Rumah Sakit

Menurut (UU No.44 tahun 2009) tentang Rumah Sakit adalah bagian *integral* dari suatu organisasi sosial dan kesehatan dengan fungsi menyediakan pelayanan paripurna (*Komprehensif*), penyembuhan penyakit (*Kuratif*) dan pencegahan penyakit (*Preventif*) kepada masyarakat, Rumah Sakit juga merupakan pusat pelatihan bagi tenaga kesehatan dan pusat penelitian medik.

Berdasarkan (Permenkes RI Nomor 56 Tahun 2014 Tentang Klasifikasi Dan Perizinan Rumah Sakit) menyatakan Rumah Sakit Umum adalah rumah sakit yang memberikan pelayanan kesehatan pada semua bidang dan jenis penyakit.

2.1.2 Fungsi dan Tugas Rumah Sakit

Berdasarkan UU RI nomor 44 tahun 2009 tentang Rumah Sakit mempunyai fungsi yaitu :

1. Penyelenggaraan pelayanan pengobatan dan pemulihan kesehatan sesuai dengan standar pelayanan rumah sakit.
2. Pemeliharaan dan peningkatan kesehatan perorangan melalui pelayanan kesehatan yang paripurna.
3. Penyelenggaraan pendidikan dan pelatihan sumber daya manusia dalam rangka peningkatan kemampuan dalam pemberian pelayanan kesehatan.
4. Penyelenggaraan penelitian dan pengembangan serta pemimpin teknologi bidang kesehatan dalam rangka peningkatan pelayanan kesehatan dengan memperhatikan etika ilmu pengetahuan bidang kesehatan.

Berdasarkan (Permenkes RI No 4 Tahun 2018 tentang Kewajiban Rumah Sakit dan Kewajiban Pasien) :

“Pada hakikatnya Rumah Sakit merupakan Institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan secara paripurna yang menyediakan pelayanan Rawat Inap, Rawat Jalan, dan Instalasi Gawat Darurat”

2.2 Sistem Antrian

2.2.1 Pengertian Sistem

Menurut (Indrianto, 2010) menyatakan bahwa :

“Sistem adalah sekumpulan unsur/elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Misalnya: sistem Komputer yang terdiri dari *Software, Hardware, Brainware*. Secara umum model sistem sederhana digambarkan sebagai berikut” :



Gambar 2. 1 Metode Sistem Antrian

1. Masukan (*input*) adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk di *process*.
2. Pengolahan (*process*) adalah bagian dari sistem yang berfungsi sebagai pengolah yang mengubah masukan menjadi keluaran yang berguna.
3. Keluaran (*output*) adalah hasil dari pemrosesan pada sistem informasi.

Menurut Ludwig Von Bertalanfy, Sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi *Input Process Output* diantara unsur-unsur tersebut dengan lingkungan. Menurut Anatol Rapoport, sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.

2.2.2 Karakteristik Sistem

Menurut (Ladjamudin, 2013 : 3) mengenai karakteristik sistem yang menyatakan bahwa:

”Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran atau tujuan.”

Adapun penjelasan dari masing-masing karakteristik sistem menurut (Ladjamudin, 2013 : 3) adalah sebagai berikut:

1. **Komponen Sistem**
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.
2. **Batasan Sistem**
Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.
3. **Lingkungan Luar**
Sistem Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan.
4. **Penghubung Sistem**
Penghubung merupakan media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.
5. **Masukan Sistem**
Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal *maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan. Sinyal input adalah energi yang diproses untuk mendapatkan keluaran dari sistem.
6. **Keluaran Sistem**
Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.
7. **Pengolahan Sistem**
Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

2.2.3 Pengertian Antrian

Antrian dapat terjadi apabila orang, komponen mesin atau *unit* barang harus menunggu untuk mendapatkan pelayanan dari fasilitas pelayanan yang sedang beroperasi pada kapasitas tertentu sehingga tidak melayani mereka untuk sementara waktu. Ketika pelanggan menunggu untuk mendapatkan jasa pelayanan, maka keberadaan sistem antrian sangat diperlukan. Beberapa contoh berikut menunjukkan bahwa penggunaan sistem antrian sangat membantu untuk melancarkan pelayanan kepada pelanggan atau konsumen, seperti : Pelanggan menunggu pelayanan di depan loket pendaftaran untuk mendapatkan Antrian adalah suatu kejadian yang biasa dalam kehidupan sehari-hari. Menunggu di depan loket untuk mendapatkan nomor antrian sesuai dengan urutan.

Antrian timbul disebabkan oleh kebutuhan akan pelayanan melebihi kemampuan pelayanan, sehingga pengguna fasilitas yang tiba tidak bisa segera mendapat layanan disebabkan kesibukan layanan. Pada banyak hal, tambahan fasilitas pelayanan dapat diberikan untuk mengurangi antrian mencegah timbulnya antrian. Akan tetapi biaya Sering timbulnya antrian yang panjang akan mengakibatkan hilangnya pasien.

2.2.4 Metode Antrian

Komponen Dasar Antrian memiliki beberapa komponen dasar yang harus diperhatikan (Gross, 2008:3). Komponen dasar *process* antrian adalah :

1. Kedatangan

Setiap masalah antrian melibatkan kedatangan, misalnya orang, mobil, panggilan telepon untuk dilayani, dan lain-lain. Unsur ini sering dinamakan *process input*. *Process input* meliputi sumber kedatangan atau biasa dinamakan *calling population*, dan cara terjadinya kedatangan yang umumnya merupakan variabel acak.

2. Pelayanan

Pelayanan atau *mekanisme* pelayanan dapat terdiri dari satu atau lebih pelayanan, atau satu atau lebih fasilitas pelayanan. Tiap-tiap fasilitas pelayanan kadang – kadang disebut sebagai saluran. Contohnya, rumah sakit yang sudah terakreditasi C atau terakreditasi D masih terdapat beberapa pintu loket pendaftaran yang memungkinkan pelayanan disetiap harinya membutuhkan pelayanan yang kondusif agar antrian sesuai urutannya. Maka dari itu dibutuhkan aplikasi antrian pada loket pendaftaran agar pelayanan berjalan dengan lancar.

3. Antrian

Inti dari analisa antrian adalah antri itu sendiri. Timbulnya antrian terutama tergantung dari sifat kedatangan dan proses pelayanan. Jika tak ada antrian berarti terdapat pelayanan yang menganggur atau kelebihan fasilitas pelayanan.

Penentu antrian lain yang penting adalah disiplin antri. Disiplin antri adalah aturan keputusan yang menjelaskan cara melayani pengantri. Menurut (Sinalungga, 2008) ada 4 bentuk disiplin pelayanan yang biasa digunakan, yaitu :

1. *First Come First Served (FCFS)* atau *First In First Out (FIFO)* artinya, lebih dulu datang (sampai), lebih dulu dilayani (keluar). Misalnya, antrian loket pendaftaran nomor antrian.
2. *Last Come First Served (LCFS)* atau *Last In First Out (LIFO)* artinya, yang tiba terakhir yang lebih dulu keluar. Misalnya, sistem antrian dalam elevator untuk lantai yang sama.
3. *Service In Random Order (SIRO)* artinya, panggilan didasarkan pada peluang secara random, tidak soal siapa yang lebih dulu tiba.
4. *Priority Service (PS)* artinya, prioritas pelayanan diberikan kepada pelanggan yang mempunyai prioritas lebih tinggi dibandingkan dengan pelanggan yang mempunyai prioritas lebih rendah, meskipun yang terakhir ini kemungkinan sudah lebih dahulu tiba dalam garis tunggu. Kejadian seperti ini kemungkinan disebabkan oleh beberapa hal, misalnya seseorang yang dalam keadaan penyakit lebih berat dibanding dengan orang lain dalam suatu tempat praktek dokter.

2.3 Perangkat Keras dan Perangkat Lunak

Perangkat keras adalah salah satu komponen dari sebuah komputer yang sifat alatnya bisa dilihat dan diraba secara langsung atau yang berbentuk nyata, yang berfungsi untuk mendukung proses komputerisasi. *Hardware* adalah

perangkat komputer yang terdiri atas susunan komponen-komponen elektronik berbentuk fisik (berupa benda). *Hardware* atau perangkat keras adalah sebuah alat atau benda yang bisa dilihat, sentuh, pegang dan memiliki fungsi tertentu

2.3.1 Perangkat Keras

Spesifikasi *PC* yang digunakan dalam penelitian ini adalah sebagai (Haqi and Sinaga, 2017):

1. *Processor* : Intel Core 2Duo 2.10 GH
2. *Memory* : DDR 2 GB
3. *VGA card* : Intel
4. *Hard Disk* : 250 GB HDD
5. *Motherboard* : Asus
6. *Speaker PC*
7. *Monitor touch screen all for one*



Gambar 2. 2 *Monitor Touch Scren All For One*

8. Printer *Thermal*



Gambar 2. 3 Printer *Thermal*

2.3.2 Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan system ini adalah sebagai berikut :

1. *Microsoft Visual Basic 6.0*
2. *SQL Server 2.0*
3. *Database*

2.4 *Microsoft Visual Basic 6.0*

2.4.1 Definisi *Visual Basic 6.0*

Microsoft Visual Basic 6.0 adalah bahasa program yang bekerja dalam lingkup *MS- Windows* (M. Agus J. Alam, 2001:1). *Microsoft Visual Basic* berasal dari bahasa pemrograman yang populer yang disebut *Basic (Beginner's All Purpose Symbol Instruction Code)* (Wahana Komputer, 2004:2). Bahasa *Basic* diperkenalkan pertama kali oleh Dartmouth College pada tahun 1960.

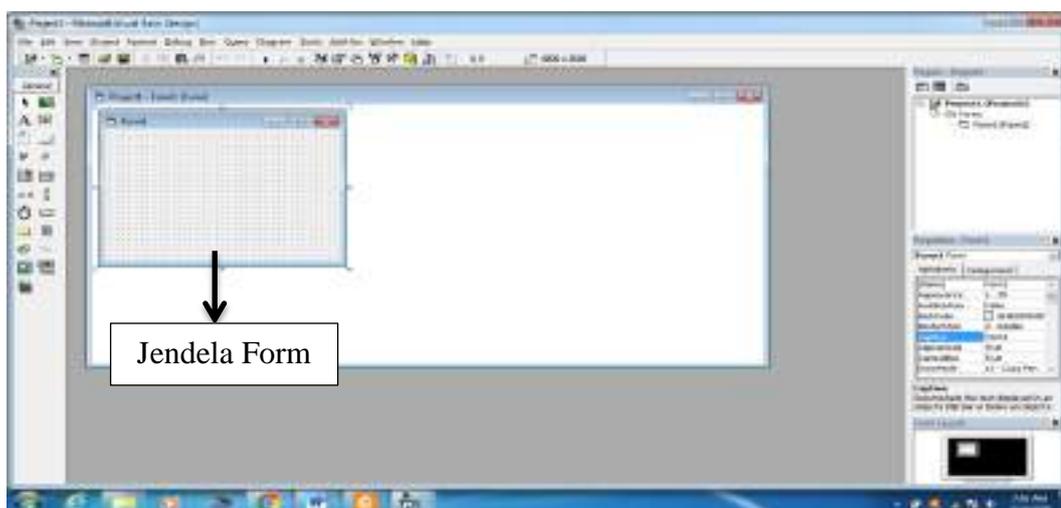
Menurut (Sarifudin, 2014) Menyatakan bahwa :

“pengembang *software* mempopulerkan bahasa *basic* dengan membuat berbagai macam aplikasi seperti *GW- Basic*, *Qbasic* dan *Quick Basic*. *Visual Basic 6.0* berbasis DOS untuk *Windows* diperkenalkan pada tahun 1991. *Visual Basic 6.0* memiliki kelebihan-

kelebihan yang tidak dimiliki oleh versi sebelumnya. Kelebihannya antara lain kompilasi (*process compile*) dapat dilakukan dengan cepat, mendukung kontrol data objek yang baru, mendukung berbagai macam database, pembuatan laporan yang lebih mudah dan mendukung pengaksesan terhadap internet”

Visual Basic 6.0 dapat menghasilkan tipe data yang dapat disesuaikan sendiri. data tersebut dapat berupa argumen dan properti. Dengan adanya fasilitas *Native Code* untuk mengkompilasikan *source code*, maka akan menghasilkan suatu aplikasi dengan waktu eksekusi lebih cepat. *Visual Basic 6.0* menyediakan tiga macam *interface* yang bisa digunakan untuk merancang aplikasi sesuai dengan kebutuhan. *Interface* tersebut berupa MDI (*Multi Document Interface*), SDI (*Single Document Interface*) dan EDI (*Exploler Document Interface*). Modifikasi pada masing-masing bagian akan terasa lebih mudah. Fasilitas yang disediakan juga lebih lengkap sehingga bisa memenuhi selera programmer yang pada akhirnya akan meningkatkan produktivitas kerja.

Interface antar muka *Visual Basic 6.0*, berisi *menu*, *toolbar*, *toolbox*, *form*, *project explorer* dan *property* seperti berikut :



Gambar 2. 4 Tampilan Menu di dalam *aplikasi Visual Basic 6.0*

Pembuatan program aplikasi menggunakan *Visual Basic 6.0* dilakukan dengan membuat tampilan aplikasi pada *form*, kemudian diberi *script* program di dalam komponen-komponen yang diperlukan. Form disusun oleh komponen-komponen yang berada di [*Toolbox*], dan setiap komponen yang dipakai harus diatur propertinya lewat jendela [*Property*]

2.5 SQL Server

Pada aplikasi *database*, terdapat bahasa yang mengatur agar pengguna dapat berkomunikasi dengan *database*. SQL Server (*Structured query language*) dikategorikan sebagai bahasa untuk memanipulasi dan digunakan untuk mengakses/manajemen data di dalam sebuah *database* relasional.

SQL (*T-SQL*) adalah bahasa basis data yang digunakan pada *MS. SQL Server*. *T-SQL* memiliki tiga kategori yaitu : (Puspitasari, Watequlis and Asmara, 2017)

2.5.1 *Data Definition Language (DDL)*

Data Definition Language (DDL) adalah bahasa basis data yang digunakan untuk menggambarkan desain basis data secara keseluruhan. Dengan statemen DDL, kita dapat membuat, mengubah dan menghapus struktur dan definisi mendata dari objek-objek *database*. Objek objek pada *database* yang dimaksud adalah sebagai berikut :

1. *Database*
2. *Table*
3. *View*
4. *Index*

5. *Procedure (Store Procedur)*
6. *Function*
7. *Trigger*

2.5.2 Data Manipulation Language (DML)

Data Manipulation Language (DML) berguna untuk melakukan manipulasi dan pengambilan data pada suatu basis data. Manipulasi data dapat berupa :

1. Mendapatkan/mencari data
2. Penyimpanan/Penambahan data
3. Mengubah data
4. Menghapus data

SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada DBMS”. Sedangkan menurut (Fauzi & Miftakul, 2012) mengemukakan bahwa “SQL merupakan sebuah alat untuk melakukan proses organisasi, manajemen, dan pengambilan data yang tersimpan dalam sebuah *database*”. SQL merupakan singkatan dari *structured query language* yang digunakan untuk berkomunikasi dengan *database* dan melakukan pengolahan data.

Berdasarkan pernyataan di atas, maka SQL merupakan suatu bahasa standar yang digunakan untuk mengakses data dalam sebuah basis data dan melakukan pengolahan data. Berikut ini adalah contoh pengaksesan data pada DBMS dengan SQL yang secara umum terdiri dari empat (4) hal yaitu:

1. Memasukkan data (*insert*)

Perintah yang digunakan untuk menambah atau memasukkan data pada basis data.

2. Mengubah data (*update*)

Perintah yang digunakan untuk mengubah atau memperbarui data pada basis data.

3. Menghapus data (*delete*)

Perintah yang digunakan untuk menghapus data pada basis data.

4. Menampilkan data (*select*)

Perintah yang digunakan untuk menampilkan data pada basis data.

2.6 Database

2.6.1 Pengertian Database

Basis data merupakan salah satu teknologi yang harus dimiliki sebuah perusahaan, institusi, ataupun organisasi adalah teknologi yang dapat memproses data. Secara konsep basis data atau *database* adalah kumpulan dari data-data yang membentuk suatu berkas yang saling berhubungan (*relation*) dengan tata cara tertentu untuk membentuk data baru atau informasi. Kumpulan dari data yang saling berhubungan antara satu dengan lainnya yang diorganisasikan berdasarkan skema atau struktur tertentu (Supriyanto, 2005).

2.6.2 Tujuan Perancangan Database

1. Untuk memenuhi informasi yang berisikan kebutuhan-kebutuhan *user* secara khusus dan aplikasi-aplikasinya.
2. Memudahkan pengertian struktur informasi

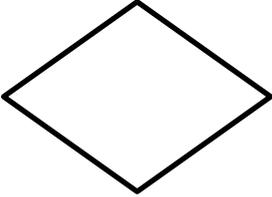
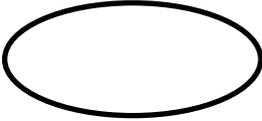
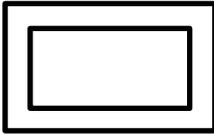
3. Mendukung kebutuhan-kebutuhan pemrosesan dan beberapa obyek penampilan (*respon time, processing time, dan storage space*).

Untuk mengelola *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Data Base Management System*). DBMS merupakan suatu sistem perangkat lunak yang memungkinkan pengguna untuk membuat, memelihara, mengontrol, dan mengakses *database*. Dengan DBMS, pengguna dapat mengontrol dan memanipulasi data yang ada.

2.7 *Entity Relationship Diagram (ERD)*

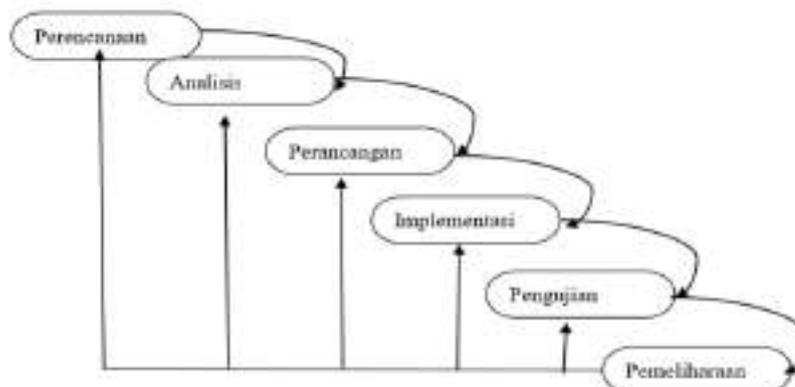
Menurut *Brady dan Loonam* (2010) Untuk mendokumentasikan data perusahaan dengan mengidentifikasi jenis entitas (*entity*) dan hubungannya. *ERD* merupakan suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara abstrak. Juga menggambarkan hubungan antara satu entitas yang memiliki sejumlah atribut dengan entitas yang lain dalam suatu sistem yang terintegrasi. *ERD* ini juga merupakan model konseptual yang dapat mendiskripsikan hubungan antara file yang digunakan untuk memodelkan struktur data serta hubungan antar data. *ERD* menggunakan simbol-simbol khusus untuk menggambarkan elemen-elemen.

No	Simbol	Nama	Keterangan
1.		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <i>field-fieldnya</i> dipergunakan dalam aplikasi program

No	Simbol	Nama	Keterangan
2.		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya
3.		<i>Atribut</i>	Atribut adalah karakteristik dari sebuah entitas
4.		Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas
5.		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

2.8 Pengembangan Sistem

Penelitian ini menggunakan metodologi pengembangan sistem Menurut (Simarmata, 2010). “*System Development Life Cycle (SDLC)*” dengan metode *waterfall*. Metode air terjun atau istilah lainnya *waterfall* model adalah model yang memacu tim pengembang untuk mengumpulkan dan menentukan apa yang seharusnya dilakukan sebelum sistem dikembangkan. Model ini cocok untuk sistem yang mengedepankan kualitas dibandingkan biaya pengembangan atau waktu pengembangan. Kelebihan *waterfall* model adalah memberikan kemudahan serta kejelasan interpretasinya. Model ini terstruktur serta cocok diadaptasi untuk management control.



Gambar 2.5 Metode SDLC

2.9 Metodologi pengembangan sistem informasi

2.9.1 Metode SDLC

Menurut Azhar Susanto (2004) menyatakan bahwa : “*System Development Life Cycle (SDLC)* adalah salah satu metode pengembangan sistem informasi yang populer pada saat sistem informasi pertama kali dikembangkan.”

Metode SDLC (*System Development Life Cycle*) memiliki 5 tahapan sebagai berikut : (Susanto, 2004)

1. Perencanaan (*Planning*)
Tahap perencanaan adalah tahap awal pengembangan sistem yang mendefinisikan perkiraan kebutuhan-kebutuhan sumberdaya. Dalam tahap ini juga dilakukan langkah-langkah berupa: mendefinisikan masalah, menentukan tujuan sistem, mengidentifikasi kendala-kendala sistem dan membuat studi kelayakan.
2. Analisis (*Analysis*)
Tahap analisis merupakan tahap penelitian atas sistem yang berjalan dengan tujuan untuk merancang sistem yang baru dengan menggunakan *tools* atau alat bantu DFD (*Data Flow Diagram*) dengan *software visual basic*.
3. Desain (*Design*)
Tahap Desain yaitu tahap dalam menentukan proses data yang diperlukan oleh sistem baru dengan tujuan memenuhi kebutuhan *user* dengan alat bantu *Unified Modeling Language (UML)* dengan *software visual* paradigma *Use Case Diagram*, *Class Diagram*, *Sequence Diagram*, *Statechart Diagram* dan *Activity Diagram*.

Proses *design* akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*.

4. Implementasi (*Implementation*)

Tahap implementasi adalah tahap dimana rancangan sistem yang dibentuk menjadi suatu kode (program) yang siap untuk dioperasikan. Langkah-langkahnya yaitu : menyiapkan fasilitas fisik dan personil, dan melakukan simulasi.

5. Pemeliharaan (*Maintenance*)

Setelah melakukan implementasi terhadap sistem baru, tahap berikutnya yang perlu dilakukan adalah pemakaian atau penggunaan, audit sistem, penjagaan, perbaikan dan pengembangan system.

2.10 *Black Box Testing*

Menurut (Pressman, 2015) *Black-Box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black-Box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau *akses database eksternal*
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

Tes ini dirancang untuk menjawab beberapa pertanyaan-pertanyaan berikut ini:

1. Bagaimana validitas fungsional diuji?
2. Bagaimana perilaku dan kinerja sistem diuji?
3. Apa kelas *input* akan membuat kasus uji yang baik?
4. Apakah sistem *sensitive* terhadap nilai *input* tertentu?

5. Bagaimana batas-batas kelas data yang terisolasi?
6. Kecepatan dan volume data seperti apa yang dapat ditolerir sistem?

2.11 Penelitian Relevan

Penelitian yang dilakukan sebelumnya menurut (Septiani, Wigati, & Fatmasari, 2017) menyatakan Bahwa :

“Dari hasil penelitian dapat disimpulkan bahwa seringkali menimbulkan keluhan pasien mengenai pengambilan nomor antrian akibat letak mesin nomor antrian *Q-matic* ini berada tersendiri di luar gedung IRJ. Kebanyakan pasien rawat jalan langsung menuju gedung IRJ untuk melakukan pendaftaran, meskipun telah tersedia spanduk yang memuat tulisan “tempat pengambilan nomor antrian” di samping gedung BPJS *Center* tersebut; namun informasi tersebut kurang sepenuhnya menjangkau pasien yang ingin mendaftar pelayanan rawat jalan. Dalam hal ini, telah terjadi kesenjangan komunikasi yaitu pesan yang disampaikan kurang dipahami oleh pelanggan. Komunikasi merupakan faktor yang penting, sebab jika terjadi kesenjangan dalam komunikasi maka akan timbul penilaian dan persepsi yang negative terhadap kualitas pelayanan seluruhnya. Permasalahan lain juga terjadi di tempat pengambilan nomor antrian. Berdasarkan pengamatan, tidak tersedianya informasi mengenai cara pengambilan nomor antrian untuk pilihan poliklinik tujuan, dapat menyebabkan pasien bingung untuk menekan tombol mesin *Q-matic*, khususnya pada pasien dengan tujuan Loker Poliklinik Lainnya”

Penelitian yang dilakukan sebelumnya menurut (Haqi & Sinaga, 2017) menyatakan Bahwa :

“Dari hasil penelitian dapat disimpulkan bahwa suatu fasilitas kesehatan publik kecil yang didirikan untuk memberikan perawatan kepada pasien luar. Biasanya klinik hanya mengobati penyakit-penyakit ringan seperti demam dan sebagainya, sedangkan kasus-kasus yang lebih parah diajukan ke rumah sakit. Untuk meningkatkan kualitas pelayanan kepada pasien, sudah saatnya sebuah klinik dilengkapi dengan aplikasi computer untuk menangani pasien yang seringkali harus antri panjang untuk mendapatkan pelayanan medis. Aplikasi antrian berbasis desktop dapat membantu pasien untuk mendaftar secara cepat dan efektif. Pada klinik menerapkan aturan antrian First Come First Served (FCFS) yaitu pasien yang mendaftar terlebih dahulu yang akan mendapatkan pelayanan. Aplikasi antrian juga dapat membantu petugas klinik untuk mengelolah data-data

pasien tanpa harus di reprotkan memanggil pasien satu persatu. Petugas klinik tinggal menekan tombol pada aplikasi antrian untuk memajukan nomor antrian berikutnya jika pasien sudah selesai dilayani.