

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Ada beberapa definisi mengenai sistem menurut para ahli, diantaranya: Sistem merupakan kumpulan elemen yang saling berkaitan dan bekerja sama dalam melakukan kegiatan untuk mencapai suatu tujuan. Sistem juga dapat dilihat dari elemen-elemennya yang bisa dalam bentuk keluaran dan masukan yang dapat menerima input, mengolah input, dan menghasilkan hasil yang diharapkan (Sujarweni, 2015).

Menurut Mulyadi (2014), sistem merupakan suatu rangkaian prosedur yang dibuat menurut pola tertentu dan dibuat untuk melaksanakan kegiatan pokok suatu usaha. Seperti pernyataan di atas, sistem mempermudah kerja karena prosedur merupakan langkah yang dilakukan oleh beberapa orang untuk menyelesaikan pekerjaan secara seragam serta berulang-ulang.

Dari pengertian sistem di atas, dapat disimpulkan bahwa adanya sistem dapat mempermudah pekerjaan. Sistem yang dibuat sedemikian rupa akan menjadi alat penting untuk mempermudah kerja perusahaan. Sistem juga dapat menjadi bentuk proteksi internal dalam perusahaan untuk mencapai hasil yang akan dicapai. Mudah-mudahan karyawan dalam melakukan pekerjaan akan membuat kinerja makin baik dan efektif sehingga apa yang menjadi tujuan perusahaan akan mudah tercapai.

Kemudian informasi merupakan bagian dari sistem yang tidak dapat dipisahkan untuk menyempurnakan suatu system. Menurut Romney dan Steinbart (2015), informasi adalah data yang sudah diproses dan diorganisasikan arti bagi penggunaanya.

Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan dan mengolah serta menyimpan data, dengan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan, (Krismiaji, 2015:16).

Definisi sistem informasi menurut Laudon, yang diterjemahkan oleh Ardana dan Lukman (2016:5) adalah secara teknis sebagai suatu rangkaian yang komponen-komponennya saling berkaitan dalam mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan mengendalikan perusahaan.

Berdasarkan definisi diatas penulis dapat menyimpulkan bahwa sistem informasi adalah rangkaian atau komponen yang saling berkaitan dalam mengumpulkan, memproses, menyimpan dan melaporkan informasi untuk mendukung pengambilan keputusan demi mencapai tujuan perusahaan.

2.2 Rekam Medis

Rekam medis adalah berkas yang berisikan catatan dan dokumen tentang identitas, anamnesis, diagnosis pengobatan, pemeriksaan, pengobatan, tindakan, dan pelayanan lain yang diberikan kepada pasien pada sarana pelayanan kesehatan yang meliputi pendaftaran pasien yang dimulai dari tempat penerimaan pasien, kemudian bertanggung jawab untuk mengumpulkan, menganalisa, mengolah, dan menjamin kelengkapan berkas rekam medis dari unit rawat jalan, unit rawat inap, unit gawat darurat, dan unit penunjang lainnya (Hatta, dkk. dalam Lubis, 2009).

Menurut Peraturan Menteri Kesehatan Republik Indonesia Nomor 269/MENKES/PER/2008 pasal 1, Tentang Rekam Medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain yang telah diberikan kepada pasien.

Rekam Medis adalah siapa, apa, dimana, dan bagaimana perawatan pasien selama di rumah sakit, untuk melengkapi rekam medis harus memiliki data yang cukup tertulis dalam rangkaian kegiatan guna menghasilkan diagnosis, jaminan, pengobatan, dan hasil akhir. Rekam medis adalah keterangan baik yang tertulis maupun yang terekam tentang identitas pasien, anamnese penentuan fisik laboratorium, diagnose segala pelayanan dan tindakan medik yang diberikan kepada pasien dan pengobatan baik yang dirawat inap, rawat jalan maupun yang mendapatkan pelayanan gawat darurat (Rustiyanto, 2009).

2.3 Dashboard

Dashboard merupakan sebuah model aplikasi sistem informasi yang disediakan bagi para manager untuk menyajikan informasi kualitas kinerja, dari sebuah perusahaan atau lembaga organisasi. Konsep kinerja dashboard sudah ada selama bertahun-tahun, dashboard telah banyak diadopsi oleh perusahaan atau kalangan bisnis (Ilhamsyah & Syahru, 2017).

Dashboard adalah komponen yang umumnya memiliki *Performance Management Systems*, *Performance Measurement Systems*, *BPM Suites*, dan *BI Platforms*. Dashboard menyediakan tampilan visual dari informasi penting yang disatukan dan diatur dalam sebuah layar tunggal sehingga informasi dapat dipahami cukup dengan sekali lihat, serta mudah untuk dieksplorasi (Turban, et al., 2011 : 137).

Adapun pengertian Dashboard menurut Johan, et al. (2004, p93), Dashboard adalah sebuah aplikasi bagi knowledge workers yang berguna untuk menampilkan informasi yang sesuai dengan kebutuhan dalam satu layar. Informasi yang ditampilkan telah dipilah dan digabungkan dari berbagai sumber informasi yang ada. Dashboard merupakan bagian dari Business Intelligence yang secara langsung akan menampilkan berbagai informasi yang dibutuhkan oleh suatu organisasi atau perusahaan dengan beragam bentuk seperti grafik dan indikator warna yang dapat memudahkan pengguna dalam mengambil keputusan secara cepat dan tepat.

2.4 Hypertext Preprocessor (PHP)

Supono dan Putratama (2016:3) mengemukakan bahwa PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang berbasis *server-side* yang dapat ditambahkan ke dalam HTML. Sedangkan, menurut Solichin (2016:11) mengemukakan bahwa PHP merupakan

salah satu bahasa pemrograman berbasis *web* yang ditulis oleh dan untuk pengembang *web*.

Menurut Maimunah (2017:2) PHP singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada server (*server-side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru *uptodate*. Semua *script* dieksekusi pada server dimana *script* tersebut dijalankan.

Berdasarkan kalimat diatas menerangkan bahwa *hypertext preprocessor* (PHP) merupakan bahasa pemrograman untuk membuat/mengembangkan aplikasi berbasis *web* dan bersifat *open source* dan ditanamkan ke dalam **script** HTML.

2.5 *Hypertext markup language* (HTML)

Menurut Solichin (2016:10) “HTML merupakan bahasa pemrograman *web* yang memberitahukan peramban *web* (*web browser*) bagaimana menyusun dan menyajikan konten di halaman *web*”.

Hypertext markup language (HTML) adalah bahasa *markup* standar untuk membuat dan menggambarkan struktur halaman *web* yang terdiri dari elemen HTML sebagai blok bangunan sebuah halaman dan direpresentasikan dengan tag (Atmoko, 2018:1).

Berdasarkan definisi dari para ahli yang telah dikemukakan di atas, dapat disimpulkan bahwa *hypertext markup language* (HTML) merupakan bahasa standar yang digunakan untuk menyusun dan menyebarkan informasi serta menampilkan halaman *web*.

2.6 *Cascading style sheet* (CSS)

Cascading style sheet (CSS) merupakan bahasa pemrograman yang berfungsi untuk mempercantik tampilan *web* (Solichin, 2016:10).

Menurut Prasetio (2014:252) menyatakan bahwa “CSS adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman *website* (situs)”.

Berdasarkan pengertian diatas, maka dapat disimpulkan bahwa *cascading style sheets* (CSS) merupakan bahasa yang digunakan dan membantu *programmers* dalam merancang sebuah tampilan *website* dan menimbulkan efek animasi yang bagus.

2.7 Javascript

Javassript dikembangkan oleh *Netscape* dengan nama awal *LiveScript* yang berfokus pada proses pengolahan data di sisi *client* dan menyajikan komponen *web* yang lebih interaktif serta berfungsi untuk menambah fungsionalitas dan kenyamanan halaman *web* (Solichin, 2016:11).

Menurut Suryana & Koesheriyatin (2014:181) “*Javascript* adalah bahasa *script* berdasar pada objek yang memperbolehkan pemakai untuk mengendalikan banyak aspek interaksi pemakai pada suatu dokumen HTML dimana. Dimana objek tersebut dapat berupa suatu *window*, *frame*, URL, dokumen, *form*, *button*, atau *item* yang lain”.

Dapat disimpulkan bahwa *javascript* merupakan bahasa pemrograman yang berbasis *client* dan *script* untuk tampilan pendukung pada *website* sehingga membuat halaman menjadi lebih menarik dan interaktif.

2.8 Bootstrap

Menurut Subagia (2018:45) “*bootstrap* adalah *template* desain *web* dengan fitur *plus* (*framework* CSS). *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman”.

Sedangkan menurut Rozi (2016:225) “*bootstrap* adalah paket aplikasi siap pakai untuk membuat *template web*”.

Berdasarkan pendapat dari para ahli diatas, dapat disimpulkan bahwa *bootstrap* merupakan paket aplikasi yang menyediakan *template* dengan format *cascading style sheet* (CSS) untuk mendesain *web* agar tampilan *web* menjadi lebih menarik.

2.9 MySQL

MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi *web*. Contoh DBMS lainnya adalah : *PostgreSQL* (*freeware*), *SQL Server*, *MS Access* dari *Microsoft*, *DB2* dari *IBM*, *Oracle* dan *Oracle Corp*, *Dbase*, *FoxPro*, dsb (Hidayatullah dan Jauhari,2015:180).

Sibero (2013:97) MySQL atau dibaca “My Sekuel” dengan adalah suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data. Menurut Raharjo (2011:21), MySQL merupakan RDBMS (atau server *database*) yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak *user*.

Menurut Ahmar (2013:11) mengemukakan bahwa MySQL adalah sistem yang berguna untuk melakukan proses pengaturan koleksi-koleksi struktur data (*database*) baik meliputi proses pembuatan atau proses pengelolaan *database*. MySQL merupakan sebuah software database yang mengelola tipe data relasional, menyimpan data-data tersebut ke dalam bentuk tabel-tabel yang saling berhubungan.

2.10 Rumah Sakit

Pengertian rumah sakit telah diatur dalam undang-undang dan peraturan menteri kesehatan, Menurut Rikomah (2017:2), Rumah Sakit merupakan institusi pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan secara paripurna yang menyediakan pelayanan rawat inap, rawat jalan dan rawat darurat. Rumah sakit sebagai organisasi badan usaha di bidang kesehatan mempunyai peranan penting dalam mewujudkan derajat kesehatan masyarakat secara optimal.

2.11 Model Prototipe

Model prototipe digunakan untuk merancang sistem informasi. Model *prototype* memberikan kesempatan untuk pengembang program dan objek penelitian untuk saling berinteraksi selama proses perancangan sistem (Sukamto & Shalahuddin, 2015:33).

Sedangkan menurut Yurindra (2017:47) model *prototype* adalah “suatu proses yang memungkinkan *developer* membuat sebuah model *software*, metode ini baik digunakan apabila *client* tidak bisa memberikan informasi yang maksimal mengenai kebutuhan yang diinginkannya”.

Maka dari itu, dapat disimpulkan bahwa model prototipe merupakan salah satu model pengembangan perangkat lunak dimana pengembang program dan objek penelitian dapat saling berkomunikasi dan memberikan informasi yang terdiri dari mendengarkan pelanggan atau analisa kebutuhan, membuat rancangan (*mock-up*) dan pengujian rancangan). Model *prototype* ini memiliki beberapa tahapan (Sukamto & Shalahuddin, 2015:32), yaitu:

1. Mendengarkan Pelanggan

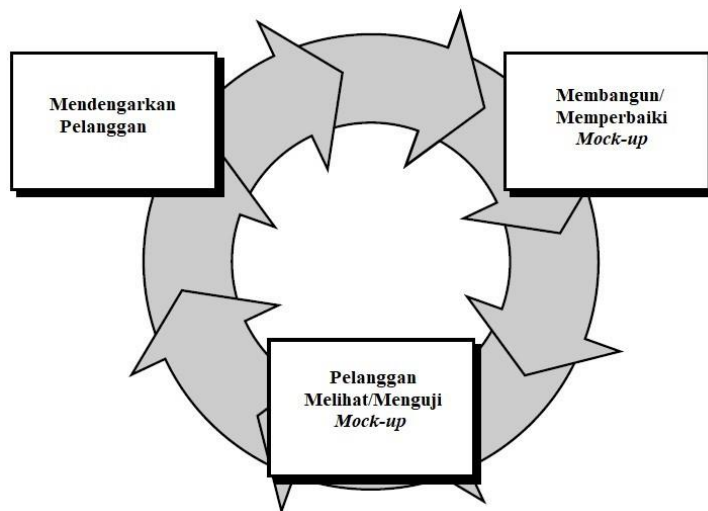
Pengembang program dan objek penelitian bertemu dan menentukan tujuan umum dan kebutuhan dasar. Detail kebutuhan mungkin pada awal pengumpulan kebutuhan.

2. Membangun atau Memperbaiki *Mock-Up*

Perancangan sistem dapat dikerjakan apabila data-data yang berkaitan telah dikumpulkan selama pengumpulan kebutuhan. Rancangan ini menjadi dasar pembuatan *prototype*. Pembuatan *prototype* ini merupakan tahapan perealisasi rancangan *prototype* menggunakan bahasa pemrograman.

3. Pelanggan Melihat dan Menguji *Mock-Up*

Objek penelitian mengevaluasi *prototype* yang dibuat dan dipergunakan untuk memperjelas kebutuhan *software*.



Gambar 2.1 Ilustrasi Model *Prototype* (Sumber: Sukamto & Shalahuddin, 2015:32)

2.12 *Unified Modeling Language* (UML)

UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem (Hendini, 2016). Sedangkan Menurut Sukamto & Shalahuddin (2015:137) mendefinisikan bahwa “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

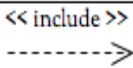
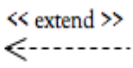
Maka dari itu dapat ditarik kesimpulan bahwa *unified modeling language* (UML) merupakan suatu bahasa standar yang digunakan untuk pemodelan dan komunikasi rancangan perangkat lunak dengan menggunakan diagram atau simbol simbol tertentu.

1. Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat”. *Use case* diagram digunakan untuk mendeskripsikan tipikal interaksi antara pengguna dengan sistem informasi (Maulana, 2014). Sedangkan menurut Sukamto & Shalahuddin (2015:155) mendefinisikan bahwa “Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Berdasarkan beberapa pendapat ahli di atas, dapat disimpulkan bahwa *use case* diagram merupakan diagram UML yang berfungsi sebagai alat bantu pemodelan untuk menggambarkan tingkah laku (*behavior*) dari sudut pandang luar sistem untuk menjelaskan interaksi dan peran antara aktor dengan sistem yang dirancang.

Tabel 2.1 Simbol *Use Case* Diagram

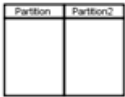
NO	GAMBAR	NAMA	KETERANGAN
1		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		Include	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
3		Extend	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</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

2. Activity Diagram

Activity diagram merupakan diagram yang menerangkan tentang aktifitas-aktifitas yang dapat dilakukan oleh seorang *entity* atau pengguna yang akan diterapkan pada aplikasi (Meilinda, 2016). Sedangkan menurut Sukanto & Shalahuddin (2015:161) mendefinisikan bahwa “Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Dapat disimpulkan bahwa *activity* diagram merupakan diagram yang menggambarkan aktifitas-aktifitas sistem dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.

Tabel 2.2 Simbol *Activity* Diagram

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.

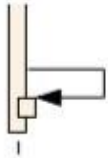
3. Sequence Diagram

Sequence diagram merupakan *UML* yang menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu (Meilinda, 2016).

Sukanto & Shalahuddin (2015:165) mengemukakan bahwa: Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima oleh objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang di instansiasi menjadi objek itu.

Dapat disimpulkan bahwa *sequence* diagram dapat diartikan sebagai alat pemodelan rancangan sistem yang menggambarkan alur atau urutan sistem yang bersinkronisasi dengan *use case* diagram untuk mendeskripsikan waktu hidup objek dan *message* yang dikirim atau diterima oleh objek tersebut.

Tabel 2.3 Simbol *Sequence* Diagram

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan form entry dan form cetak.
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas,
	contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar class.
	Recursive, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.

	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang lifeline terdapat activation.

4. Class Diagram

Class diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab *entitas* yang menentukan perilaku sistem (Hendini, 2016). Sedangkan menurut Sukamto & Shalahuddin (2015:141), “diagram kelas atau *class* diagram menggambarkan struktur sistem dari sini pendefinisian kelas kelas yang akan dibuat untuk membangun sistem”.

Maka dari itu, dapat disimpulkan bahwa *class* diagram merupakan diagram *UML* yang mirip dengan *ERD* atau *LRS* yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas dan memperlihatkan aturan serta tanggung jawab *entitas* serta berisikan metode-metode atau aturan tertentu.

Tabel 2.4 Multiplicity *Class* Diagram

Multiplicity	Keterangan
1	Satu dan hanya satu.
0..*	Boleh tidak ada atau 1 atau lebih.
1..*	1 atau lebih.
0..1	Boleh tidak ada, maksimal 1.
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4.