

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem informasi adalah sistem komputer yang menyediakan informasi kepada pengguna sesuai kebutuhan yang diinginkan. sistem informasi juga merupakan sebuah sistem yang terintegrasi dari beberapa komponen untuk dapat mengolah sebuah data sehingga menjadi sebuah informasi yang bermakna dan berguna untuk membantu mencapai tujuan sebuah organisasi (Halimah *et al.*, 2023).

2.2 Pengolahan Data

Pengolahan data adalah suatu proses untuk menerima data sebagai masukan (*input*) memproses (*processing*) dengan menggunakan proses tertentu, dan mengeluarkan hasil proses data tersebut dalam bentuk informasi (*output*) (Mahyuni, Martono, 2016).

2.3 Rumah Sakit

Menurut Undang-Undang Republik Indonesia No. 44 Tahun 2009 tentang Rumah Sakit: Rumah sakit adalah fasilitas kesehatan yang menyelenggarakan pelayanan kesehatan perorangan yang bersifat komprehensif, terpadu, dan berkelanjutan, mencakup pelayanan promotif, preventif, kuratif, dan rehabilitatif.

2.4 Aplikasi

Aplikasi merupakan intruksi (*instruction*) atau pernyataan (*statement*) dalam suatu komputer yang tersusun berstruktur sehingga komputer dapat mengendalikan aktivitas *user* dalam memproses sebuah inputan menjadi *output* (Setiada & Setiawan, 2016). Aplikasi berasal dari kata bahasa Inggris “*aplication*” yang artinya pengolah yang di ambil dari kata kerja *to aply*. Aplikasi bersumber dari *aljabar booelan*, yang menggunakan kode *binary* digit (bit) yang terdiri dari 2 angka, *on* artinya benar dan *off* berarti salah. Kode binary ini juga

tersusun dalam beberapa kelompok bit yaitu yang terdiri dari 4 bit (*nible*), 8 bit(*byte*), 2 byte (*word*) dan 32 bit (*double bit*) pengelompokan idni berfungsi untuk mempermudah *software* komputer dalam melakukan kegiatan.

Habibi & karnovi, (2020) menyatakan bahwa aplikasi merupakan *software* siap pakai dalam menjalankan sejumlah fungsi perintah untuk mempermudah berbagai bentuk pekerjaan maupun tugas - tugas tertentu misalnya seperti penerapan, pemakaian dan juga penambahan data. Sedangkan menurut Solichin (2016) Aplikasi terbagi sesuai dari lingkungan pengembangannya yang diuraikan sebagai berikut :

2.4.1 Aplikasi Besbasis Desktop

Aplikasi berbasis *desktop* adalah sebuah aplikasi yang membutuhkan proses *instalasi* di setiap komputer *desktop* yang akan digunakan. Contoh dari aplikasi berbasis *desktop* yaitu *Microsoft Office*, *Mozilla Firefox*, *Adobe Photoshop* dan lainnya.

2.4.2 Aplikasi Besbasis Mobile

Aplikasi berbasis *Mobile* adalah aplikasi yang hanya dapat digunkan atau di jalankan pada sebuah perangkat bergerak (*mobile*) seperti *handphone*, *smartphone*, dan PDA. Contoh dari aplikasi berbasis *desktop* yaitu *Opra Mini*, *Blackbarry Messenger (BBM)* dan *Polaris Office*.

2.4.3 Aplikasi Berbais Web

Aplikasi berbasis *web* merupakan aplikasi yang berada di dalam *server* sehingga tidak di butuhkan *instalasi* setiap komputernya. Situs *web* merupakan sebuah salah satu contoh yang merupakan jenis aplikasi berbasis *web*. Selain itu, aplikasi berbasis *web* berkembang sangat pesat dan memiliki beberapa kelebihan yaitu :

1. *Client* tidak perlu melakukan proses *instalasi* ataupun proses *update* karena hal ini cukup dilakukan pada sisi *sever* saja.

2. Dapat diakses darimana pun melalui sebuah jaringan. Jika *server* aplikasi berada di jaringan *intranet* (LAN), aplikasi dapat diakses dari seluruh komputer di dalam jaringan *intranet* tersebut. Lain halnya jika *server* berada di jaringan *internet* (memiliki *IP Public* atau diletakkan di *web hosting*), maka aplikasi dapat diakses dari *internet*.
3. Data disimpan dalam sisi *server*, sehingga akses terhadap data (pengguna) dapat diatur sesuai kebutuhan.
4. *Cross-platform*, artinya aplikasi dapat diakses melalui komputer dengan berbagai sistem operasi seperti *Windows*, *Linux*, *Mac* dengan syarat memiliki *browser*.
5. Pada sisi *client* hampir seluruh proses aplikasi dilakukan disisi *server* sehingga tidak memerlukan spesifikasi komputer yang besar.
6. *Client* (pengguna) lebih aman dari virus ataupun gangguan keamanan lainnya karena aplikasi berjalan diatas *browser*.

Kelemahan aplikasi berbasis *web* ini memiliki kelemahan yaitu aplikasi sangat bergantung pada sebuah kondisi jaringan.

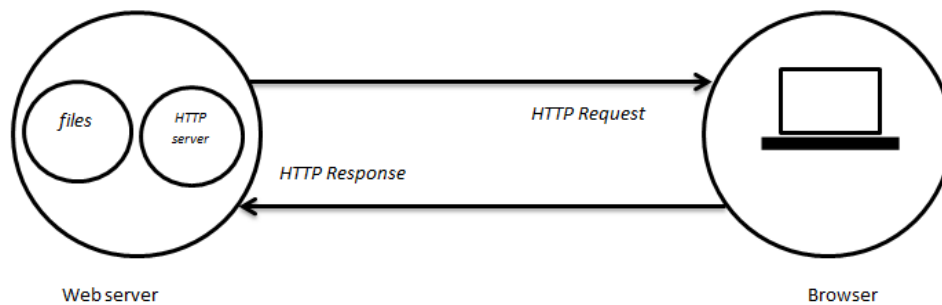
2.5 Web Server

Menurut Yudhanto & Prasetyo (2018) dalam buku yang berjudul “Panduan Mudah Belajar *Framework Laravel*” *webserver* merupakan sebuah perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui *browser* yang memberikan tanggapan permintaan dalam bentuk halaman situs web dalam bentuk dokumen HTML.

Dalam *webserver* memiliki dua perangkat yaitu perangkat keras (*hardware*) yang digunakan untuk menyimpan semua data seperti HTML dokumen, gambar, file *CSS stylesheets*, dan file *JavaScript*. Sedangkan pada perangkat lunak (*software*) memiliki fungsi sebagai pusat kontrol dalam memproses permintaan yang diterima dari *browser*.

Sehingga dapat disimpulkan *webserver* merujuk kepada *software* dan *hardware* yang memiliki tugas untuk mengatur semua komunikasi yang

terjadi antara *browser* dengan *server* untuk memproses sebuah website. Cara kerja *webserver* dapat di lihat pada gambar 2.1.



Gambar 2. 1 Cara kerja webserver.

Cara kerja *webserver* yaitu saat mengambil halaman website, *browser* akan mengirimkan permintaan ke *server* yang kemudian akan diproses oleh *webserver*. Selanjutnya *HTTP request* akan dikirim ke *webserver* dan *web server* akan mengecek keamanan terlebih dahulu. *HTTP request* di bantu dengan *HTTP server* untuk mengakses alamat situs web (menerjemahkan URL). Kemudian, *webserver* mengirimkan *HTTP response* ke *browser* dan memproses menjadi sebuah halaman situs web.

2.6 Website

Website menurut Ibrahim dan Ambarita (2018), adalah kumpulan dari halaman *web* yang telah di *publish* pada jaringan *internet* dan memiliki *domain* atau URL (*Uniform Resource Locator*) yang dapat diakses melalui alamatnya oleh semua pengguna *internet*. Hal ini di mungkin dengan adanya teknologi *World Wide Web* (WWW).

Halaman website biasanya berisikan dokumen dalam bentuk format penulisan *Hyper Text Markup Language* (HTML) yang dapat diakses menggunakan HTTP ataupun HTTPS untuk berkomunikasi di antara komputer dalam satu jaringan, sehingga dapat menyampaikan berbagai informasi dari *server website* untuk ditampilkan kepada para *user*.

2.6.1 Element Website

Website dibangun dari berbagai elemen yang saling berkaitan dan mendukung satu sama lain, elemen – elemen itu dapat dijelaskan sebagai berikut :

2.6.1.1 *Hypertext Markup Language (HTML)*

Hypertext Markup Language (HTML) adalah sebuah bahasa standar yang digunakan dalam membuat sebuah *web* yang berfungsi untuk menandai bagian – bagian dari sebuah halaman (Rerung, 2018) .

Menurut Enterprise (2018) *Hypertext Markup Language* (HTML) adalah struktur yang digunakan untuk membuat desain sebuah *website*.

2.6.1.2 *Hypertext Preprocessor (PHP)*

Hypertext Preprocessor (PHP) menurut wahyuno (2013) merupakan *script* untuk membangun aplikasi yang dapat terintegrasi ke dalam halaman HTML, sehingga dapat membuat *web* bersifat menjadi dinamis.

PHP di kelompokkan dalam katagori bahasa pemograman sisi *server* (*server side programming*) yang artinya *script* yang dibuat dengan php akan disimpan pada *server* untuk memenuhi permintaan *client*.

2.6.1.3 *Cascading Style Sheet (CSS)*

Cascading Style Sheet (CSS) adalah bahasa yang digunakan untuk mengatur dan mempercantik tampilan pada suatu *website* yang ditulis menggunakan format HTML / HTMTML (Koesheryatin, 2014).

2.6.1.4 *JavaScript*

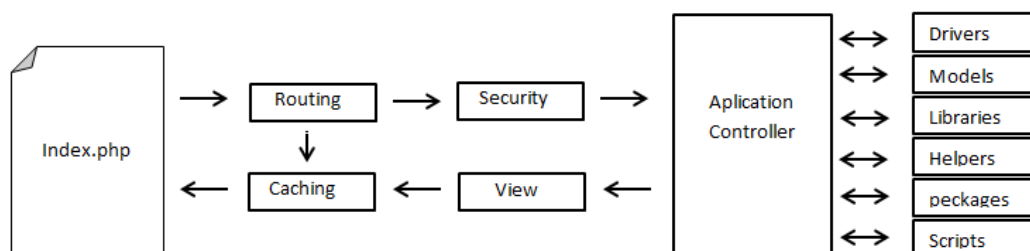
JavaScript menurut saihaan (2018) adalah bahasa sisi *client* yang digunakan dalam membuat halaman *web* yang dapat berintraksi. *javascript* didesain pada *browser* komputer, bukan pada *server* sehingga dapat dijalankan pada semua *flatfrom*.

2.7 Framework Codeigniter

Menurut Supardi & Hermawan 2018, *Framework* adalah kumpulan komponen yang berisikan (fungsi, prosedur, dan *class*) yang dapat langsung digunakan, sehingga membantu meringankan pekerjaan *developer / programmer* dalam membuat *software*. Sedangkan *CodeIgniter* merupakan aplikasi *open source* berupa *framework* yang menggunakan MVC dalam membangun *website* dinamis dengan bahasa pemograman PHP.

Framework codeigniter adalah sebuah kerangka kerja terstruktur yang menggunakan model MVC (*Model, View, Control*) untuk membangun sebuah *website* yang dinamis menggunakan PHP (Airputih & Non-Commertia 2014).

Menurut widodo (2015) dalam bukunya yang berjudul “Membangun Web Super Cepat dengan Codeigniter GroceryCURD dan TnakAuth” menjelaskan alur kerja Framework yang dapat di lihat pada Gambar 2.1.



Gambar 2. 2 Alur Kerja Framework Codeigniter

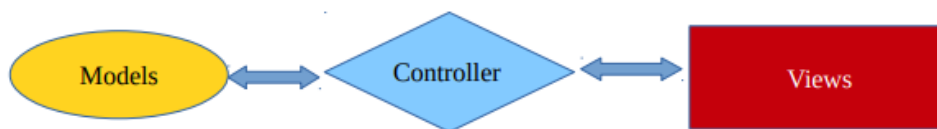
Dari gambar diatas dapat dijelaskan sebagai berikut :

1. Index.php merupakan controller awal yang menginisialisasikan kebutuhan untuk menjalankan Codeigniter.
2. Router/Routing merupakan bagian yang menentukan kegiatan yang harus dilakukan ketika ada request/permintaan dari client/browser.
3. Caching merupakan bagian yang mengecek apakah data sudah pernah diminta atau belum, jika cache dalam keadaan aktif, maka akan langsung dikirimkan kepada client/browser dengan mengabaikan alur kerja normal.
4. Security, sebelum aplikasi dikirimkan, maka akan lebih dahulu data difilter sebagai keamanan. Controller merupakan pengendali dari

jalannya aplikasi, dan akan segera memproses sesuai request/permintaan yang diminta, yaitu models, libraries, helpers, plugins dan script.

2.7.1 Konsep MVC (Model, View, Controller)

Menurut Airputih & Non-Commercialia (2014) dalam “Modul Panduan *Framework Codeigniter (Ci)*” dijelaskan bahwa MVC (*Model, View, Controller*) secara sederhana merupakan patent / teknik dalam *framework* yang memisahkan antara desain, data, dan proses. Berikut ini adalah penjelasan mengenai konsep MVC dapat dilihat pada gambar 2.3.



Gambar 2. 3 Konsep MVC

1. Model adalah suatu hal yang langsung berhubungan dengan database untuk memanipulasi data (*insert, update, delete, search*) serta menangani validasi dari bagian *controller*, tetapi tidak dapat berhubungan langsung dengan bagian *view*.
2. View adalah bagian yang menangani *presentation logic* yang menerima dan mempresentasikan data (*file template HTML*, yang diatur oleh *controller*) kepada *user*. View juga tidak memiliki akses langsung terhadap bagian model.
3. *Controller* adalah jembatan yang mengatur antara bagian model yang terkoneksi pada database dengan dibuatkan class dan functions di controllers dan dihubungkan ke bagian *views* untuk di tampilkan aplikasi. *controller* juga berfungsi untuk menerima request dan data dari *user*, kemudian menentukan apa yang akan diproses oleh aplikasi.

Ada beberapa keuntungan dalam menggunakan *framework codeigniter* dalam membangun sebuah aplikasi berbasis *web*, yaitu :

1. Mudah dipelajari dan dimengerti karena *framework codeigniter* merupakan turunan dari php.

2. Menghemat waktu pembuatan karena memiliki struktur MVC dan *library* yang telah disediakan sehingga *programmer* tidak perlu membangun dari awal.
3. *URL friendly* cukup dengan menggunakan `routes.php` kita dapat membuat url dengan meminimalisasi penggunaan `$_GET`.

2.8 MySQL

MySQL adalah salah satu perangkat lunak berbasis data yang bersifat *open source* yang menyediakan *fitur – fitur* berbasis *GUI* yang memudahkan pengguna dalam menggunakan dan mengolahnya (solichin, 2016).

Menurut komputer (2010) dalam buku yang berjudul “Panduan Belajar *MySQL Database Server*” *MySQL* adalah *database server* yang dapat mengirim dan menerima data dengan sangat cepat dan *multi user*. *MySQL* memiliki kelebihan sebagai berikut :

1. *MySQL* adalah *server* tercepat dan merupakan *system manajemen databse* yang *OpenSource* dan memiliki perfoma yang tinggi yang simpel.
2. *Database MySQL* mengerti bahasa *SQL (Structured Query Language)*.
3. *MySQL* dapat diakses oleh semua *server*, selain itu semua klien juga dapat mengakses *server* dalam satu waktu.
4. *MySQL* dapat berjalan di berbagai *operating system* seperti *linux*, *windows*, *solaris*, dan lain-lain.

2.9 Metode dan Teknik Pengembangan Sistem

Metode merupakan tahap – tahap ataupun aturan untuk melakukan sesuatu. Metode dan teknik pengembangan sistem yang digunakan adalah :

2.9.1 System Development Life Cycle (SDLC)

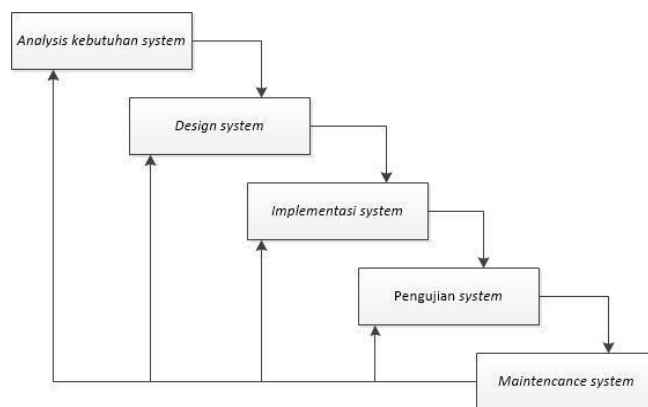
Menurut Mulyani (2017) *System development life cycle (SDLC)* adalah sebuah metode atau proses logika yang digunakan seorang *system analist* untuk mengembangkan sebuah sistem informasi dengan melibatkan

requirements, validation, training, dan pemilik sistem yang identik dengan teknik pengembangan sistem waterfall.

2.9.2 Waterfall

Waterfall adalah proses pengembangan perangkat lunak berurutan mengalir kebawah (seperti air terjun). Fase atau tahapan dalam pengembangan metode *waterfall* dimulai dari tahapan analisis kebutuhan, desain sistem, implementasi pengujian dan pemeliharaan (Tristiano, 2018).

Selain itu Rosa A.S & M. Shalahuddin (2018) juga menyatakan bahwa *Waterfall* adalah model sekuesial linier atau alur hidup klasik yang menyediakan pendekatan alur hidup perangkat secara terurut. Sekuensial adalah metode dalam *SDLC* yang pengerjaannya harus dilakukan secara urut, sesuai dengan fase yang harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Fase dalam metode *waterfall* dimulai dari analisis kebutuhan perangkat lunak, desain perangkat lunak, *implementasi* perangkat lunak, pengujian perangkat lunak, dan pemeliharaan perangkat lunak. Tahapan dalam model *waterfall* dapat dilihat pada Gambar 2.4.



Gambar 2. 4 Tahap kerja dengan menggunakan model waterfall.

Kelebihan menggunakan metode *waterfall* sebagai berikut:

1. Memiliki proses yang urut, karena pengerjaan metode ini dimulai dari menyelesaikan satu tahap ke tahap yang selanjutnya.
2. Pengguna lebih mudah dalam merencanakan dan menyiapkan kebutuhan data yang diperlukan.

3. Memiliki jadwal yang jelas, yang dilihat dari target penyelesaian pengembangan program berdasarkan urutan tahapan metode, sehingga menghasilkan hasil yang sesuai dengan perencanaan.

2.10 Mapping Chart

Mapping chart atau diagram alir dokumen adalah diagram yang menyajikan aliran dokumen dalam suatu sistem informasi dengan menggunakan diagram. Fungsi *mapping chart* ini adalah mempermudah melakukan analisa terhadap sistem, dan sarana komunikasi antar *user* dan analisis. Simbol-simbol yang terdapat didalam Bagan Alir Dokumen (*mapping chart*), disajikan pada Tabel 2.1.

Tabel 2. 1 Simbol-simbol bagan alir dokumen

Sistem	Keterangan
	Terminal yang menunjukkan sumber atau tujuan dokumen dan laporan
	Dokumen dumber atau laporan
	Operasi manual
	File untuk menyimpan dokumen sumber atau laporan
	Catatan akuntansi (jurnal, <i>register</i> , log, buku besar)
	Konektor intrahalaman
	Konektor antarhalaman
	Deskripsi proses atau komentar
	Garis alir dokumen

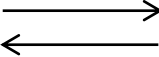
Sumber : Komputer. 2010.

2.11 Data Flow Diagram

Data Flow Diagram (DFD) adalah diagram notasi yang menggambarkan sebuah arus data pada suatu sistem agar mudah dipahami oleh sistem secara logika dan terstruktur (Rusmawan, 2019).

Menurut Muslihudin & Oktafianto (2016) *Data Flow Diagram* (DFD) merupakan aliran sebuah alur data secara logika yang mampu menggambarkan sistem lama ataupun sistem baru tanpa melibatkan sebuah lingkungan diluar sistem. DFD juga mampu memberikan pemahaman bagi pengguna sistem yang kurang memahami bidnag komputer untuk mengerti sistem yang sedang berjalan. Simbol *Data Flow Diagram* dapat dilihat pada tabel 2.2

Tabel 2. 2 Simbol – Simbol Data Flow Diagram

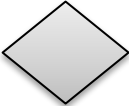
Yourdon/De Marco	Keterangan
(1)	(2)
	Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem, tetapi diluar sistem.
	Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem, tetapi diluar sistem.
	Penyimpanan data atau tempat data di refer oleh proses.
	Aliran arah data dari arah khusus dan dari sumber ke tujuan.

(Sumber : Rusmawan, 2019).

2.12 Flowchart

Flowchart merupakan gambaran prosedur atau langkah – langkah terurut dari suatu program yang mampu menganalisis dalam memecahkan masalah (Ridho, 2017). *Flowchart* adalah *diagram* alir yang berisikan simbol – simbol yang mengekspresikan proses intruksi aliran kegiatan dari awal hingga akhir (Anggarawan, 2018). Simbol *Flowchart* dapat di lihat pada tabel 2.3.

Tabel 2. 3 Simbol-simbol flowchart.

Simbol	Nama Simbol	Keterangan
(1)	(2)	(3)
	Proses	Menunjukkan proses pengolahan ada sistem
	<i>Input/Output</i>	Menunjukkan sebuah keluaran atau masukan pada program
	<i>Decision</i>	Percabangan pada program yang memiliki nilai ya/tidak
	Terminal	Menunjukkan awal atau akhir suatu program atau <i>flowchart</i>
	Penghubung	Digunakan untuk menghubungkan antar symbol
	Modul	Menunjukkan proses yang telah ditentukan baik berupa prosedur atau fungsi
	Panah	Digunakan untuk menunjukkan arah dari suatu proses

(Bayu Rahayudi, 2011)

2.13 Black Box Testing

Black box testing adalah metode untuk melakukan pengujian sebuah perangkat lunak dalam segi spesifikasi dan fungsional tanpa menguji desain dan kode program. Pengujian ini berfokus untuk memvalidasi dan memenuhi kebutuhan dari apa yang diminta dan diuji/*requirement* (Yuliawati, Andriyadi and Nursiyanto, 2022).

2.14 Artikel Terkait

Penelitian terkait merupakan sebuah teori yang didapatkan dari karya ilmiah atau penelitian – penelitian sebelumnya, yang dapat digunakan sebagai acuan dan data dukung dalam studi kasus yang diangkat oleh penulis. Adapun referensi karya ilmiah yang didapatkan oleh penulis adalah sebagai berikut:

1. Rahmayu, M. (2016), dalam karya ilmiah nya yang berjudul “Rancang Bangun Sistem Informasi Pada Rumah Sakit Dengan Layanan Intranet Menggunakan Metode Waterfall” yang memiliki tujuan merancang aplikasi web pada rumah sakit yang menyediakan fasilitas bagi petugas untuk melayani pasien, mengetahui data transaksi yang terjadi pada rumah sakit, serta mempermudah proses pencatatan dan dokumentasi data sistem pelayanan pasien dengan menggunakan web sistem.
2. Nurfityanti, I. (2011), dalam karya ilmiah nya yang berjudul “Rancang bangun sistem informasi penerimaan pegawai baru pada rumah sakit Syarif Hidayatullah” yang memiliki tujuan merancang sistem penerimaan calon pegawai baru dengan menggunakan internet yang mencakup dan penerimaan calon pegawai selain itu untuk mendapatkan mengetahui SOP atau prosedur penerimaan pegawai baru.

3. Rijki, N. M., & Iksari, I. H. (2023), dalam karya ilmiah nya yang berjudul “Implementasi Sistem Infomasi Manajemen Rumah Sakit” yang memiliki tujuan mempercepat proses pengolahan data, meminimalkan risiko kesalahan, dan meningkatkan koordinasi antara berbagai departemen rumah sakit. SIMRS menjadi alat yang penting dalam meningkatkan pengelolaan rumah sakit secara menyeluruh dan meningkatkan pelayanan kesehatan yang diberikan kepada pasien.