



BAB II

LANDASAN TEORI

BAB II

LANDASAN TEORI

2.1 Landasan Teori

2.1.2 Sistem Informasi

Menurut Azhar Susanto, Sistem informasi merupakan kumpulan dari sub sistem baik yang bersifat fisik maupun non fisik yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna. Penggunaan dari sistem informasi ditujukan untuk mengolah berbagai informasi yang diperlukan oleh setiap perusahaan atau organisasi, sehingga sumber daya atau resources yang dibutuhkan tidak terlalu besar dan dapat mempersingkat waktu penanganan proses [3].

Komponen sistem informasi terdiri dari berbagai hal yang memiliki fungsi masing-masing namun tetap memiliki tujuan untuk membentuk satu informasi yang valid. Ada 6 komponen utama dalam sistem informasi yang harus ada sehingga bisa terbentuk sebuah sistem yang bisa bekerja dengan baik. Komponen tersebut antara lain [4]:

1. *Hardware* (perangkat keras), meliputi berbagai piranti fisik dibutuhkan guna mendapatkan dan mengolah data informasi.
2. *Software* (perangkat lunak), meliputi sekumpulan instruksi yang memungkinkan *hardware* dapat memproses data.
3. Proses, meliputi aturan yang digunakan untuk mengolah atau memproses data yang didapatkan guna menghasilkan *output* yang digunakan.
4. Manusia, Semua pihak yang bertanggung jawab dalam proses pengembangan dan penggunaan output dari sistem informasi.
5. *Database*, meliputi sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.
6. Jaringan, memungkinkan sumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

Dengan demikian dapat disimpulkan bahwa sistem informasi adalah suatu sistem terintegrasi yang tersusun dari beberapa komponen yang saling terkait. Komponen tersebut meliputi *hardware*, *software*, proses, manusia, *database*, dan jaringan. Dengan adanya integrasi komponen-komponen tersebut, sistem informasi dapat berfungsi efektif dalam memenuhi kebutuhan informasi organisasi atau perusahaan.

2.1.3 Reservasi

Reservasi atau pemesanan dalam bahasa Inggris adalah *reservation* yang berasal dari kata *to reserve* yaitu menyediakan atau mempersiapkan tempat sebelumnya. *Reservation* yaitu pemesanan suatu tempat fasilitas. Sedangkan menurut kamus besar bahasa Indonesia reservasi diartikan sebagai pengaturan atau pemesanan tiket, meja, kamar hotel, dan sebagainya untuk seseorang pada waktu tertentu. Secara umum reservasi yaitu pemesanan fasilitas yang diantaranya hotel, akomodasi, *meal*, *seat* pada pertunjukan, pesawat terbang, kereta api, bus, hiburan, *night club*, *discoteque* dan sebagainya [5].

Reservasi memiliki manfaat yang signifikan baik bagi pelanggan maupun penyedia layanan. Bagi pelanggan reservasi dapat memberikan kepastian tempat dan waktu, serta dapat datang pada saat yang ditentukan dan langsung memperoleh tempat atau fasilitas. Selain itu, bagi penyedia layanan reservasi dapat memudahkan untuk dalam mengelola kapasitas pelanggan dan mempersiapkan kebutuhan atau permintaan pelanggan sehingga meningkatkan efisiensi operasional dan pengalaman pelanggan secara keseluruhan [6].

Reservasi seringkali memiliki ketentuan yang harus dipenuhi oleh pelanggan. Salah satu ketentuannya adalah pembayaran uang muka atau deposit untuk mengkonfirmasi reservasi. Ketentuan ini bertujuan untuk memastikan keseriusan pelanggan dalam melakukan reservasi.

2.1.4 Billiard

Permainan billiard berasal dari olahraga kriket dengan cara bermainnya adalah memasukan bola ke dalam lubang menggunakan stik yang mirip seperti tuas. Seiring berjalannya waktu, konsep permainan kriket diadaptasi untuk membentuk permainan billiard. Inti dari permainan billiard adalah memasukan keseluruhan bola ke dalam lubang secara berurutan berdasarkan angka yang terdapat pada bola [1].

Permainan billiard terdiri dari beberapa jenis yang diakui dalam turnamen nasional hingga internasional. Jenis-jenis permainan tersebut meliputi [7]:

1. Permainan 8 bola, dua pemain dengan masing-masing harus memasukkan 7 bola ke dalam lubang (bola solid nomor 1-7 untuk pemain pertama, bola stripes nomor 9-15 untuk tim kedua).
2. Permainan 9 bola, pemain harus memasukkan bola secara berurutan dari nomor 1 sampai 9. Pukulan pertama harus mengenai bola nomor 1.
3. One Pocket, menggunakan 15 bola, pemain memilih satu lubang untuk memasukkan bola sebelum permainan dimulai.
4. English Billiard, mirip dengan permainan billiard lainnya, namun menggunakan meja yang lebih besar.

5. Bola Pembunuh, pemain hanya diperbolehkan memukul bola satu kali dan harus memasukkan bola dalam satu pukulan untuk menang. Gagal tiga kali berarti kalah.
6. Permainan Pola Cuthroat, dimainkan oleh tiga pemain dengan 15 bola yang dibagi menjadi tiga kelompok. Pemain yang pertama kali memasukkan semua bolanya menang.

Billiard membutuhkan akurasi, kekuatan, dan ketajaman mata. Setiap pemain harus mempertimbangkan dengan cermat sudut dan kekuatan pukulan untuk mengarahkan bola ke sasaran dengan tepat. Kemampuan untuk mengukur jarak dan sudut dengan akurat serta mengontrol kekuatan pukulan sangatlah penting untuk mencapai tujuan dalam permainan billiard. Selain itu, ketajaman mata sangat dibutuhkan untuk melihat dan merencanakan gerakan yang tepat dalam setiap situasi permainan. Kemampuan memukul bola yang baik merupakan kunci utama yang mendukung pemain dalam memenangkan atau memainkan permainan billiard dengan baik.

2.1.5 Website

Website merupakan sekumpulan halaman yang digunakan untuk menampilkan berbagai informasi informasi seperti teks, gambar diam atau bergerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman [8].

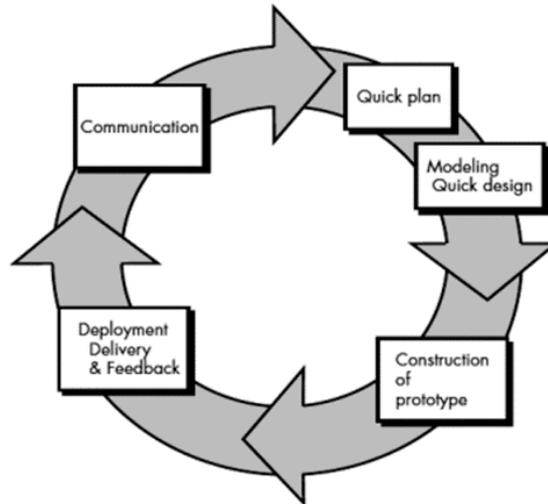
Website yang sudah dipublikasikan di internet dan memiliki domain/URL disebut situs *website*. Website terbagi menjadi dua jenis, yaitu [9]:

- a. *Website* Statis, merupakan *website* yang memiliki sifat tidak mudah diubah oleh pengguna. Jika pengguna ingin mengubah kontennya, pengguna perlu melakukan *coding* pada halaman website atau merubahnya melalui database.
- b. *Website* Dinamis, merupakan *website* yang mudah untuk beradaptasi dengan mudah dan secara otomatis menyesuaikan perubahan konten tanpa perlu mengubah strukturnya secara manual.

Website dapat diakses dengan menggunakan berbagai jenis browser seperti Google Chrome, Mozilla Firefox, Safari, Opera, dan Internet Explorer. Keberagaman ini memastikan bahwa pengguna dapat mengaksesnya dari berbagai *platform* dan perangkat sesuai dengan preferensi mereka. Selain itu, *website* memiliki banyak manfaat salah satunya sebagai wadah untuk membagikan sumber informasi. Dengan kemampuan untuk menjangkau pengguna yang luas, *website* menjadi sarana yang efektif untuk menyebarkan pengetahuan, berita, panduan, dan konten informatif lainnya kepada pengguna di seluruh dunia.

2.1.6 Metode *Prototype*

Metode *Prototype* merupakan metode pengembangan perangkat lunak yang memungkinkan adanya interaksi antara pengembang sistem dengan pengguna sistem, sehingga dapat mengatasi ketidakserasian antara pengembang dan pengguna [2]. Berikut tahapan *prototype* menurut Pressman dilihat pada **Gambar 2.1** berikut [10]:



Gambar 2. 1 Metode *Prototype*

a. Komunikasi (*Communication*)

Tahapan pertama merupakan komunikasi dilakukan dengan mengidentifikasi kebutuhan aplikasi dengan melibatkan pengguna agar memberikan hasil sesuai keinginan pengguna. Proses ini mencakup observasi, wawancara, dan kajian pustaka untuk memahami masalah dan kebutuhan pengguna.

b. Perencanaan Cepat (*Quick Plan*)

Tahap kedua adalah menetapkan bagaimana perangkat lunak akan dioperasikan.

c. Pemodelan Perancangan Cepat (*Modeling Quick Design*)

Tahap ketiga adalah melakukan pemodelan perancangan cepat yang direpresentasikan dalam *Unified Modelling Language (UML)* untuk mendeskripsikan kebutuhan pengguna.

d. Pembuatan Prototype (*Contruction of Prototype*)

Tahap keempat adalah menerapkan hasil desain sistem ke dalam bahasa pemrograman.

e. Penyerahan dan Umpan Balik (*Deployment Delivery and Feedback*)

Tahap terakhir adalah menyerahkan kepada pengguna untuk mendapatkan feedback yang akan digunakan sebagai landasan untuk memperbaiki agar sesuai dengan spesifikasi kebutuhan pengguna.

2.1.7 PHP

PHP singkatan dari *Hypertext Preprocessor* yang digunakan sebagai Bahasa *script server-side* dalam pengembangan web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan web dapat dibuat dinamis sehingga *maintenance* situs web tersebut menjadi lebih mudah dan efisiensi. PHP merupakan *software Open-Source* yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya. PHP ditulis menggunakan Bahasa C [11].

Menurut Alexander F. K. Sibero, PHP adalah pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode di jalankan. PHP sebenarnya bisa dipakai secara *command line*, artinya skrip PHP dapat dijalankan tanpa melibatkan web server maupun browser [12].

Menurut Tim EMS, Salah satu kelebihan PHP adalah kemudahannya untuk berintegrasi dengan *database*. PHP dapat mendukung beberapa database secara langsung tanpa menginstal konektor, seperti yang diperlukan dalam bahasa pemrograman Java. Dengan demikian, PHP sangat fleksibel berhubungan dengan berbagai *database*. Dari beberapa database yang ada, MySQL adalah yang paling sering disandingkan dengan PHP [13].

2.1.8 Framework

Framework adalah kumpulan pernyataan atau instruksi yang dikumpulkan dalam class dan function dengan fungsi-fungsi tertentu untuk mempermudah dalam pemanggilan *syntax* serta mempermudah *developer* atau *programmer* dalam mengembangkan suatu sistem tanpa harus membuat membuat fungsi atau class dari awal, dan menghemat waktu pengerjaan [14].

a. Model-View-Controller (MVC)

Model-View-Controller (MVC) adalah sebuah konsep untuk meng-enkapsulasi data bersama dengan pemrosesan (model), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada sebuah *user interface*. MVC merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web [15].

b. Framework Laravel

Framework Laravel merupakan sebuah *framework* PHP yang dibangun dengan konsep *MVC (Model View Controller)*. Laravel merupakan kerangka yang digunakan dalam pengembangan suatu website yang berdasarkan pada *MVP (Model View Presenter)* yang dituliskan dengan menggunakan bahasa pemrograman PHP dimana kerangka ini dirancang untuk meningkatkan tingkat kualitas suatu perangkat lunak yang akan membantu untuk mengurangi biaya pengembangan serta biaya dalam pemeliharaan dan juga untuk menambah pengalaman dalam

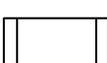
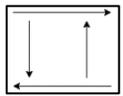
bekerja dengan suatu aplikasi yang memberikan sintaks yang ekspresif, jelas serta menghemat waktu [16].

2.1.9 Flowchart

Flowchart atau diagram alir merupakan penggambaran secara grafis dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah tertentu yang perlu dikaji dan dievaluasi lebih lanjut [17].

Flowchart dapat diterapkan untuk menggambarkan kegiatan manual, pemrosesan, atau keduanya. Dengan adanya *flowchart*, setiap urutan proses dapat digambarkan menjadi lebih jelas. Berikut **Tabel 2.1** merupakan simbol-simbol yang terdapat pada *flowchart* [18]:

Tabel 2. 1 Simbol - simbol flowchart

No.	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Memulai dan mengakhiri suatu program
2.		<i>Proses</i>	Proses perhitungan atau proses pengolahan data
3.		<i>Predefined Process</i>	Permulaan sub program atau proses pengolahan data
4.		<i>Preparation</i>	Proses inisialisasi atau pemberian harga awal
5.		<i>Input – Output</i>	Memasukan data maupun menunjukkan hasil dari suatu <i>process</i> tanpa tergantung dengan jenis peralatannya
6.		<i>Decision</i>	Memilih proses berdasarkan kondisi yang ada.
7.		<i>Flow</i>	Menghubungkan antara simbol satu dengan yang lain
8.		<i>Document</i>	Merupakan simbol untuk data yang terbentuk informasi.
9.		<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman

No.	Simbol	Nama	Keterangan
10.		<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda
11.		<i>Manual Operation</i>	Menunjukkan pengolahan yang tidak dilakukan oleh komputer/pc.
12.		<i>Manual Input</i>	Memasukan data secara manual <i>on-line keyboard</i> .

2.1.10 Use Case Diagram

Use case diagram merupakan permodelan yang memiliki kemampuan dalam merepresentasikan sebuah interaksi antara aktor dengan sistem. Seorang atau sebuah aktor adalah sebuah entitas dapat berupa manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Berikut **Tabel 2.2** adalah simbol-simbol yang ada pada *use case diagram* [19]:

Tabel 2. 2 Simbol - simbol *use case diagram*

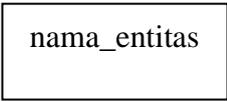
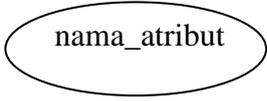
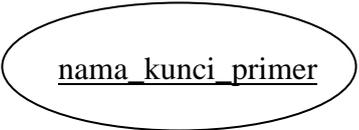
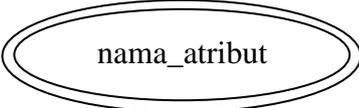
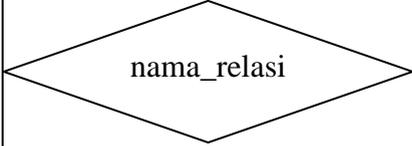
No.	Simbol	Nama	Keterangan
1.		<i>Use case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
2.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.

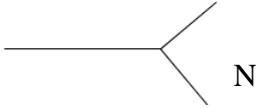
No.	Simbol	Nama	Keterangan
6.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas

2.1.11 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan representasi visual dari model data yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD menekankan pada struktur entitas dan hubungan antara entitas tersebut dalam suatu sistem. Berikut **Tabel 2.3** adalah daftar simbol dalam ERD [19]:

Tabel 2.3 Simbol - simbol ERD

No	Simbol	Keterangan
1.	Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan, biasanya berupa id.
4.	Atribut multi nilai / multivalue 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja

No	Simbol	Keterangan
6.	<p data-bbox="331 248 624 286">Asosiasi / <i>Association</i></p> 	<p data-bbox="751 248 1374 674">Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> yang menghubungkan entitas A dan entitas B.</p>