

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

Konsep dasar sistem informasi merupakan kumpulan komponen berbasis komputer yang dibuat oleh manusia untuk mengelola data, menyimpan informasi, membentuk kerangka kerja serta mengkoordinasikan sumber daya manusia dan komputer. Hal ini bertujuan untuk mengubah sistem masukan menjadi sistem keluaran guna mencapai tujuan dan sasaran yang telah ditetapkan.

2.1.1 Pengertian Sistem

Sistem merupakan gabungan dari berbagai elemen yang bekerja sama untuk mencapai suatu target atau tujuan. Dapat disimpulkan bahwa sistem adalah gabungan dari banyak elemen yang terkait lalu bekerja sama agar mencapai suatu target atau tujuan yang diinginkan.[1]

a. Karakteristik Sistem

Sistem akan berjalan dengan baik apabila suatu sistem memiliki karakteristik atau sifat-sifat tertentu. Ada beberapa karakteristik sistem yaitu: [3]

1. Komponen Sistem (*System Component*)

Sebuah sistem terbentuk dari sejumlah komponen yang saling berinteraksi dan bekerja sama untuk membentuk suatu kesatuan. Komponen-komponen ini dapat berupa sub-sistem atau bagian-bagian yang menyusun keseluruhan sistem.

2. Batasan sistem (*System Boundary*)

Pembatasan sistem merujuk pada daerah yang memisahkan suatu sistem dari sistem lainnya atau dari lingkungan eksternal. Batasan sistem berperan penting dalam sebuah sistem untuk menentukan cakupan atau ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*System Environment*)

Mencakup semua elemen di luar batas sistem yang memiliki pengaruh terhadap sistem tersebut. Lingkungan luar yang bersifat menguntungkan perlu dijaga, sementara yang bersifat merugikan harus dikendalikan untuk mencegah gangguan terhadap kelangsungan sistem.

4. Penghubung sistem (*System Interface*)

Penghubung sistem, atau media penghubung, menjadi suatu kebutuhan untuk mengalirkan sumber daya dari satu sub-sistem ke sub-sistem lainnya dalam rangka mencapai koordinasi yang efisien.

5. Masukkan sistem (*System Input*)

Energi yang dimasukkan ke dalam sistem disebut sebagai masukan sistem (*input*), yang berupa perawatan dan masukan sinyal. Perawatan diperlukan untuk menjaga sistem beroperasi dengan baik, sementara masukan sinyal merupakan energi yang diproses untuk menghasilkan keluaran (*output*).

6. Keluaran sistem (*System Output*)

Hasil dari energi yang telah diolah dan dikelompokkan menjadi keluaran sistem (*output*) disebut sebagai hasil yang bermanfaat. Contohnya informasi.

7. Pengolah sistem (*System Process*)

Agar masukan dapat diolah menjadi keluaran, diperlukan suatu komponen pengolah yang sering disebut sebagai pengolah sistem.

8. Sasaran sistem (*System Objective*)

Setiap sistem memiliki tujuan atau sasaran yang menjadi penentu input yang diperlukan oleh sistem dan keluaran yang dihasilkan.

b. Klasifikasi Sistem

Sistem terbentuk melalui integrasi antara satu komponen dengan komponen lainnya karena setiap sistem memiliki tujuan yang unik untuk setiap situasi yang dihadapinya. Oleh karena itu, sistem dapat dikelompokkan dari beberapa perspektif, antara lain:[1]

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak merujuk pada konsep-konsep atau ide-ide yang tidak memiliki bentuk fisik yang terlihat, seperti dalam sistem teologi yang membahas pemikiran tentang hubungan antara manusia dan Tuhan. Di sisi lain, sistem fisik adalah sistem yang nyata secara fisik, seperti sistem komputer.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang muncul melalui proses alam dan bukan merupakan hasil buatan manusia, seperti contohnya sistem perputaran bumi. Sistem buatan manusia melibatkan interaksi antara manusia dengan mesin, dikenal sebagai human machine system. Sistem informasi berbasis komputer

termasuk dalam kategori human machine system karena melibatkan penggunaan komputer yang berinteraksi dengan manusia..

3. Sistem Deterministik dan Sistem Probabilistik

Sistem deterministik adalah jenis sistem yang beroperasi dengan tingkah laku yang dapat diprediksi. Contohnya, sistem komputer termasuk dalam kategori ini karena tingkah lakunya dapat diprediksi berdasarkan program-program komputer yang dijalankan. Sebaliknya, sistem probabilistik adalah tipe sistem yang melibatkan unsur probabilitas, sehingga kondisi masa depannya tidak dapat diprediksi secara pasti karena adanya faktor probabilitas yang terlibat.

4. Sistem Terbuka dan Sistem Tertutup

Sistem terbuka adalah tipe sistem yang memiliki interaksi dan dipengaruhi oleh lingkungan eksternalnya. Dalam sistem ini, masukan diterima dari lingkungan dan keluaran yang dihasilkan dapat memengaruhi sub-sistem lainnya. Sementara itu, sistem tertutup adalah jenis sistem yang beroperasi secara mandiri tanpa keterlibatan atau pengaruh dari lingkungan luar. Sistem ini berfungsi secara otomatis tanpa memerlukan intervensi dari pihak eksternal.

2.1.2 Pengertian Informasi

Umumnya informasi adalah data yang sudah diolah menjadi suatu bentuk yang mempunyai arti. Informasi adalah data yang telah dikelola dan diproses untuk memberikan arti dalam memperbaiki proses pengambilan keputusan. Jadi informasi dapat disimpulkan adalah sebuah data atau fakta yang sudah diolah secara terorganisir menjadi suatu bentuk yang mempunyai arti dan bermanfaat bagi orang yang menerimanya.[4]

2.1.3 Pengertian Sistem Informasi

Sistem Informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi. Sistem informasi adalah sistem yang dapat di definisikan dengan mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu.[5] Jadi Sistem Informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan, memproses, dan menyimpan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi tersebut.

2.1.4 Pengertian WEB

World Wide Web (WWW), atau yang lebih populer dikenal sebagai web, merupakan layanan yang dapat diakses oleh pengguna komputer yang terhubung ke internet. Melalui fasilitas teks yang terkait dengan dokumen lain atau teks yang terhubung dengan informasi lain (hypertext), web memungkinkan pengguna untuk menampilkan berbagai jenis data, termasuk teks, gambar, suara, animasi, dan multimedia lainnya. Pada awalnya, web adalah ruang informasi dalam internet yang menggunakan teknologi teks yang terkait dengan dokumen atau teks yang mengandung link ke informasi lain (hypertext). Dengan bantuan web browser, pengguna dipandu untuk menemukan informasi dengan mengikuti link yang disediakan dalam dokumen web.[6]

2.1.5 Pengertian Alumni

Alumni adalah individu yang pernah menjalani proses pembelajaran di sebuah sekolah atau perguruan tinggi. Alumni diidentifikasi sebagai seseorang yang telah mengikuti dan berhasil menyelesaikan pendidikan formal di suatu lembaga pendidikan, seperti sekolah atau perguruan tinggi. Dengan demikian, dapat disimpulkan bahwa status alumni diberikan kepada individu yang telah mengikuti dan sukses menyelesaikan program pendidikan di institusi pendidikan tertentu.[6]

2.2 Metode Pengembangan Sistem

Metode Pengembangan Sistem dirancang untuk menetapkan pendekatan umum yang digunakan dalam membangun sistem baru. Pengembangan Sistem dapat diartikan sebagai proses penggambarannya, perencanaan, serta penyusunan sketsa atau pengaturan berbagai elemen terpisah menjadi suatu kesatuan yang berfungsi secara menyeluruh.

2.2.1 Waterfall

Waterfall adalah salah satu metode pengembangan perangkat lunak atau juga dikenal dengan istilah *Software Development Life Cycle (SDLC)*. Dinamakan waterfall karena model pengembangannya dianalogikan seperti air terjun, dimana tiap tahapannya dikerjakan secara berurutan dari atas ke bawah. Metode pengembangan perangkat sistem menggunakan model waterfall mengikuti langkah-langkah yang berurutan, dengan setiap fase memiliki tanggung jawabnya sendiri. Tahapan-tahapan yang umumnya ada dalam model waterfall meliputi:[7]

Tahapan-tahapan waterfall

1. Analisis

Pada tahap ini pengembang harus melakukan analisis untuk mengetahui seluruh informasi mengenai sistem. Seperti kebutuhan perangkat lunak dan metode pengumpulan data. Kebutuhan perangkat lunak adalah hardware, software dan tools apa saja yang digunakan. Sementara metode pengumpulan data dilakukan melalui wawancara, pengamatan dokumen, dan diskusi.

2. Desain Sistem

Langkah selanjutnya adalah tahap desain, yang dilakukan setelah proses analisis sistem selesai. Tujuannya adalah memberikan gambaran menyeluruh tentang langkah-langkah yang perlu diambil dan bagaimana tampilan sistem yang diinginkan akan terbentuk.

3. Implementasi

Pada tahap ini, dilakukan proses penulisan kode. Setelah itu melakukan implementasi dari sistem yang sudah direncanakan dan dirancang sebelumnya.

4. Uji Coba Program

Pengujian dilakukan untuk memastikan bahwa setiap bagian dari sistem berfungsi dengan baik. Pengujian keseluruhan sistem biasanya dilakukan disini.

5. Pemeliharaan

Tahap dimana sistem siap untuk diimplementasikan secara penuh. Instalasi, konfigurasi dan peluncuran sistem ke lingkungan produksi dilakukan disini. Setelah itu tetap melakukan pemeliharaan agar sistem tetap terjaga dengan baik.

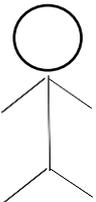
2.3 Alat Bantu Perancangan

Unified Modelling Language (UML) merupakan salah satu alat bantu yang sangat andal dalam pengembangan sistem berorientasi objek. UML dibentuk oleh empat model dan satu pandangan. Model-model ini didasarkan pada fakta-fakta struktur sistem yang dijelaskan melalui lima pandangan. Setiap pandangan saling terkait dengan perspektif tertentu di mana sistem akan dikembangkan. Kelima pandangan tersebut mencakup *Scenario*, *Development View*, *Logical View*, *Physical View* dan *Process View*. [8]

2.3.1 Use Case Diagram

Use Case diagram adalah salah satu bentuk diagram UML (*Unified Modeling Language*) yang mengilustrasikan interaksi antara sistem dan aktor-aktor yang terlibat dalam suatu sistem. Diagram ini berfungsi untuk memperlihatkan fungsionalitas sistem dari perspektif pengguna. Aktor dalam diagram use case dapat mencakup entitas-entitas eksternal, seperti pengguna, sistem lain, atau perangkat lain yang terlibat dalam interaksi dengan sistem yang sedang dianalisis. [9]

Tabel 2.1 Simbol Use Case Diagram

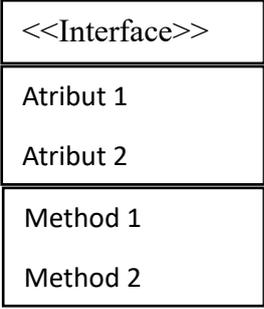
Gambar	Keterangan
	<p><i>Actor</i> menentukan sejumlah peran yang dimainkan oleh pengguna ketika terlibat dalam interaksi dengan use case.</p>

	<p><i>Use Case</i> merinci urutan aksi yang dilakukan oleh sistem dan menghasilkan hasil yang dapat diukur bagi seorang aktor.</p>
	<p><i>Association</i> menghubungkan antara objek satu dengan objek lain.</p>

2.3.2 Class Diagram

Class diagram adalah komponen dalam UML yang sangat penting selama tahap analisis dan desain pengembangan sistem. Diagram kelas membantu dalam pemodelan struktur awal sistem, hubungan antar kelas, dan perilaku sistem.[10]

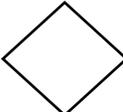
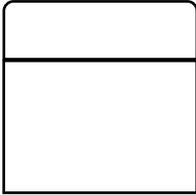
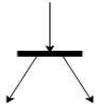
Tabel 2.2 Simbol *Class Diagram*

Gambar	Keterangan
	<p><i>Class</i> menunjukkan bentuk yang dibangun berdasarkan proses-proses sebelumnya.</p>
	<p><i>Unidirectional Association</i> simbol yang menunjukkan hubungan antara komponen satu dengan komponen lain.</p>

2.3.3 Activity Diagram

Activity diagram adalah suatu metode untuk menggambarkan logika prosedural, proses bisnis, dan alur kerja dalam berbagai konteks. Biasanya, *activity diagram* digunakan untuk mengilustrasikan aktivitas bisnis yang kompleks, menunjukkan hubungan antara satu *use case* dengan *use case* lainnya.[10]

Tabel 2.3 Simbol *Activity Diagram*

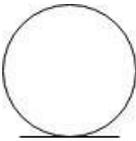
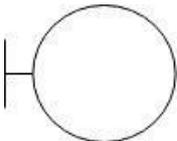
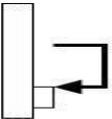
Gambar	Keterangan
	Start Point menunjukkan dimana aliran kerja dimulai.
	End Point menunjukkan dimana aliran kerja diakhiri.
	Activities menggambarkan suatu proses atau kegiatan bisnis.
	Decision Node menunjukkan suatu keputusan yang mempunyai satu atau lebih transisi.
	Control Flow menunjukkan bagaimana kendali suatu aktivitas terjadi dalam tindakan tertentu.
	Swimlane siapa yang bertanggung jawab dalam melakukan aktivitas dalam suatu diagram.
	Fork digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

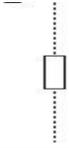
	<p>Join atau rake digunakan untuk menunjukkan adanya dekomposisi.</p>
---	--

2.3.4 Sequence Diagram

Sequence Diagram menggambarkan perilaku objek pada suatu *Use Case* dengan mendeskripsikan waktu hidup objek serta pesan yang dikirim dan diterima oleh objek tersebut. Beberapa simbol yang umum digunakan dalam *Sequence Diagram* meliputi:[10]

Tabel 2.4 Simbol *Sequence Diagram*

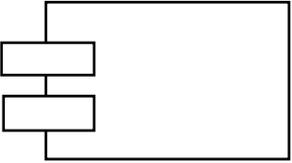
Gambar	Keterangan
	<p>Entity Class digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.</p>
	<p>Boundary Class digunakan untuk menggambarkan sebuah form (bisa data, dll).</p>
	<p>Control Class digunakan untuk menghubungkan boundary dan tabel database.</p>
	<p>Message simbol mengirim pesan antar <i>class</i>.</p>
	<p>Recursive menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>

	<p>Activation mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p>Lifeline garis titik-titik yang berhubungan dengan objek sepanjang lifeline terdapat activation yang saling berinteraksi.</p>

2.3.5 Component Diagram

Component diagram adalah jenis diagram yang secara khusus difokuskan pada perangkat lunak dan komponen perangkat keras. Diagram ini digunakan untuk menggambarkan komponen-komponen dalam sistem. *Component Diagram* berfungsi untuk memodelkan aspek fisik dari suatu sistem, serta membantu memvisualisasikan organisasi dan relasi antara komponen-komponen dalam suatu sistem.[10]

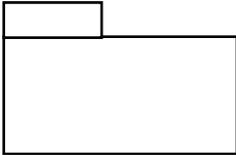
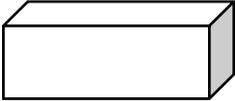
Tabel 2.5 Simbol *Component Diagram*

Gambar	Keterangan
	<p>Component simbol yang menjelaskan perangkat keras atau objek dalam suatu sistem.</p>
	<p>Dependency simbol yang menjelaskan sebuah keterkaitan antara komponen satu dengan komponen lain.</p>

2.3.6 Deployment Diagram

Diagram ini memberikan gambaran visual tentang bagaimana komponen perangkat lunak, seperti aplikasi dan server, berinteraksi dan berkomunikasi di dalam suatu arsitektur. Dengan menggunakan simbol-simbol seperti node untuk merepresentasikan perangkat keras fisik atau virtual, deployment diagram dapat membantu tim pengembangan untuk merencanakan infrastruktur yang diperlukan dan mengoptimalkan kinerja sistem.[10]

Tabel 2.6 Simbol Deployment Diagram

Gambar	Keterangan
	<p>Component simbol yang menjelaskan perangkat keras atau objek dalam suatu sistem.</p>
	<p>Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.</p>
	<p>Dependency simbol yang menjelaskan sebuah keterkaitan antara komponen satu dengan komponen lain.</p>

2.4 Alat Bantu Pemrograman

Alat bantu pemrograman adalah perangkat lunak atau aplikasi yang digunakan oleh pengembang untuk meningkatkan efisiensi, produktivitas, dan kualitas dalam proses pengkodean.

2.4.1 PHP

PHP (*Hypertext Processor*) adalah bahasa pemrograman skrip yang sederhana dan digunakan untuk memproses formulir HTML di halaman web.[11] Pembuatan program menggunakan PHP sangat mudah. Hanya perlu menyediakan editor teks sederhana seperti Notepad (*Windows*) atau vi editor (*Linux*), atau menggunakan editor program yang lebih canggih seperti EditPlus, *Notepad++*, atau *Dreamweaver*.

2.4.2 Laravel

Laravel adalah sebuah *framework* bahasa pemrograman PHP yang menawarkan banyak fitur dan memberikan dukungan yang signifikan bagi para pengembang dalam membangun aplikasi berbasis web. Framework ini dirancang khusus untuk meningkatkan kualitas aplikasi dengan mengurangi biaya pengembangan, mempermudah proses pemeliharaan, serta meningkatkan produktivitas pekerjaan melalui penggunaan kode program yang rapi dan terstruktur. Beberapa kelebihan *Laravel* mencakup penggunaan *Command Line Interface (CLI) Artisan*, kemampuan untuk menggunakan package manager PHP Composer, penulisan kode program yang rapi, singkat, terstruktur, dan mudah dimengerti oleh para pengembang.[12]

2.4.3 Bootstrap

Bootstrap adalah paket aplikasi siap pakai yang digunakan untuk membuat *front-end* sebuah website. Dapat dianggap sebagai template desain web dengan fitur tambahan. *Bootstrap* diciptakan untuk menyederhanakan proses desain web bagi

pengguna dengan berbagai tingkat keahlian, mulai dari pemula hingga yang sudah berpengalaman. Dengan pengetahuan dasar mengenai HTML dan CSS.[13]

2.4.4 MySQL

MySQL sebuah sistem manajemen basis data yang sangat terkenal, dipilih oleh banyak pengembang karena menggunakan SQL sebagai bahasa utamanya untuk berinteraksi dengan basis data. Selain itu, MySQL merupakan perangkat lunak *OpenSource* yang menyediakan kode sumber lengkap, memungkinkan pengguna untuk memahami dan memodifikasi kode yang digunakan untuk menciptakan MySQL.[14]



Gambar 2.1 Logo MySQL

Sumber : <http://www.mysql.com>

2.4.5 XAMPP

XAMPP adalah sebuah aplikasi perangkat lunak pemrograman dan database yang menyediakan berbagai aplikasi pemrograman, termasuk Apache HTTP Server, MySQL sebagai sistem manajemen basis data, serta bahasa pemrograman PHP dan Perl.[15]



Gambar 2.3 Logo XAMPP

Sumber : <http://www.xampp.com>

2.4.6 HTML

HTML adalah bahasa terstandarisasi yang digunakan oleh browser untuk menciptakan antarmuka pengguna. HTML berperan sebagai bahasa pemrograman yang digunakan sebagai alat untuk mengatur format tampilan pada web *browser* seperti *Netscape Navigator* atau *Internet Explorer*. Format dalam konteks ini adalah mekanisme yang mengatur bentuk visual dari objek-objek yang muncul di *browser*, seperti teks, gaya teks (*font*), ukuran, ketebalan, gambar, suara, koneksi (*link*), dan elemen-elemen lainnya.[16]

2.5 State of Nature

No	Judul	Nama Peneliti	Tahun	Hasil
1	Sistem Informasi Tracer Study Alumni Universtas Islam Indragiri	Ahmad Rusli dan Ilyas	2020	Sistem ini memberikan solusi efektif untuk manajemen data alumni bagi Universitas Islam Indragiri.
2	Pengembangan Sistem Informasi Tracer Study: Pemetaan dan Kebekerjaan	Khoirul Munir, Muh. Nurtamto dan Haris Abizar	2021	Sistem informasi tracer study ini sangat baik untuk digunakan alumni guna memberikan informasi kompetensi yang relevan bagi Universitas.
3	Analisis dan Perancangan Aplikasi Pelacakan Alumni (Tracer Study) Untuk Perguruan Tinggi Swasta di Indonesia	Marissa Utami dan Erwin Dwika Putra	2022	Aplikasi yang diusulkan akan memungkinkan pengguna untuk mendaftar serta menghasilkan informasi untuk tujuan melacak keberadaan alumni.
4	Rancang Bangun Sistem Informasi Tracer Study Bagi Alumni Fakultas Teknik Universitas Mataram Berbasis Web	Nadiyasari Agitha, Yunita Lestari dan Sri Endang Anjarwani	2021	Sistem ini memudahkan alumni untuk mendapatkan informasi mengenai berita fakutas, kegiatan Fakultas, informasi lowongan kerja dan informasi beasiswa di Universiyas Mataram.
5	Perancangan Sistem Informasi Aplikasi Tracer Study Alumni Berbasis Website	Rizal Bagus Pambudi, Agung Triayudi dan Andrianingsih	2020	Perancangan aplikasi tracer study ini di desain baru dan memiliki fitur baru yaitu fitur lowongan pekerjaan agar alumni dapat tertarik untuk melihat fitur yang telah disediakan dan dapat digunakan dengan baik.
6	Rancang Sistem Informasi dalam Pengembangan Model Tracer Study di Universitas Dharmawangsa	Jovi Antares, Zelvi Gustiana dan Ibnu Rusyadi	2021	Pengembangan Sistem Informasi tracer study berbasis web ini menjdi solusi yang efektif dalam meningkatkan efisiensi pelaksanaan tracer study dilingkungan Universitas Dharma Wangsa
7	Rancang Bangun Aplikasi Alumni dan Tracer Study Berbasis Web Menggunakan Metode (RAD) Rapid Application Development (Studi Kasus SMK Bhara Trikora 1 Jakarta)	Ari Ardiyansyah dan Budi Apritanto	2022	Berdasarkan penelitian yang telah dilakukan pada SMK Bhara Trikora 1 Jakarta, menghasilkan sebuah aplikasi pengelolaan data alumni berbasis website yang dapat dikelola oleh pengguna sistem dengan baik.