



**BAB II**  
**LANDASAN TEORI**

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 *Feedback***

*Feedback* adalah informasi atau tanggapan yang diberikan oleh seseorang mengenai aspek-aspek tertentu tentang kelebihan dan kelemahan dengan tujuan untuk memperbaiki atau meningkatkan kualitas pekerjaan, produk, atau kinerja individu maupun kelompok. *Feedback* yang konstruktif dapat membantu dalam pengembangan diri, penyempurnaan proses kerja, dan peningkatan hasil akhir [3].

##### **2.1.1 Jenis-jenis *Feedback*.**

1. *Feedback* positif: memberikan pujian atau penghargaan terhadap kinerja yang baik. Tujuannya untuk memotivasi dan memperkuat perilaku yang diinginkan.
2. *Feedback* negatif: mengidentifikasi kesalahan atau area yang perlu diperbaiki. Tujuannya untuk memberikan kesempatan belajar dan perbaikan.
3. *Feedback* konstruktif: gabungan antara positif dan negatif, memberikan pujian sekaligus memberi saran perbaikan dengan cara yang membangun.
4. *Feedback* langsung: diberikan segera setelah tindakan terjadi.
5. *Feedback* tidak langsung: diberikan setelah beberapa waktu berlalu.

##### **2.1.2 Manfaat *Feedback***

1. *Feedback* membantu individu atau organisasi memahami kekuatan dan kelemahan mereka, yang dapat digunakan untuk memperbaiki kinerja.
2. Memberikan informasi penting yang dapat digunakan untuk memperbaiki kinerja.
3. *Feedback* yang diberikan dengan cara yang positif dan mendukung dapat meningkatkan motivasi dan keterlibatan.
4. Penerima *feedback* yang bermanfaat dapat memberikan rasa puas dengan pekerjaan mereka.

## **2.2 Web**

*Web* atau situs merupakan kumpulan halaman yang digunakan untuk menampilkan informasi, teks, gambar diam atau bergerak, animasi, suara, dan atau gabungan dari semuanya itu, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian fitur yang saling berkait dimana masing masing dihubungkan dengan jaringan jaringan halaman (*hyperlink*). *Website* juga merupakan media informasi berbasis jaringan komputer yang dapat diakses dimana saja dengan biaya relatif murah dan *website* dapat mengimplementasikan bahasa pemrograman (*Web Programming*). *Web page* adalah dokumen yang ditulis dalam format *HTML (Hyper Text Markup Language)*, yang hampir selalu bisa diakses melalui *HTTP*, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar. Halaman-halaman dari *website* akan bisa diakses melalui sebuah *URL* yang biasa disebut *Homepage*. *URL* ini mengatur halaman-halaman situs untuk menjadi sebuah hirarki, meskipun, *hyperlink-hyperlink* yang ada di halaman tersebut mengatur para pembaca dan memberitahu mereka susunan keseluruhan dan bagaimana arus informasi ini berjalan[6].

## **2.3 Pendidikan**

Pendidikan pada dasarnya merupakan suatu proses transfer ilmu, transformasi nilai, dan pembentukan kepribadian pada semua aspek. Pendidikan juga merupakan proses untuk memperoleh keseimbangan dan kesempurnaan dalam perkembangan seseorang. Maksud dan tujuan tertentu dari pendidikan untuk mengembangkan potensi yang dimiliki oleh individu dan juga Masyarakat [7]. Pendidikan adalah usaha manusia untuk mengembangkan dan memuliakan diri, yaitu dengan membudayakan manusia atau untuk memuliakan manusia. Agar pendidikan bisa berjalan dengan baik dan sesuai tujuan, diperlukan ilmu yang secara mendalam mempelajari cara pelaksanaan pendidikan yang tepat. Ilmu dasar ini harus sudah terbukti benar dan efektif [19].

Ki Hajar Dewantara menjelaskan bahwa pendidikan sebagai upaya untuk meningkatkan budi pekerti, pikiran dan jasmani anak agar dapat meningkatkan kesempurnaan hidup selaras dengan alam dan masyarakatnya, dengan kata lain pendidikan merupakan upaya menuntun anak sejak lahir untuk mencapai kedewasaan jasmani dan rohani, dalam interaksi alam beserta lingkungannya [7].

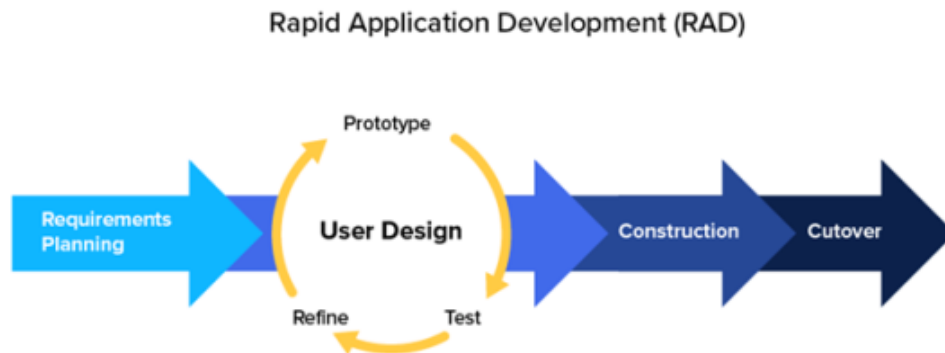
## **2.4 Sekolah**

Sekolah adalah lembaga pendidikan yang sifatnya formal, non formal dan informal, dimana pendiriannya dilakukan oleh negara maupun swasta dengan tujuan untuk memberikan pengajaran, mengelola dan mendidik para siswa melalui bimbingan yang diberikan oleh para pendidik atau guru. Sekolah adalah suatu lembaga yang memang dirancang khusus untuk pengajaran para siswa di bawah pengawasan guru-guru. Kebanyakan dalam sebuah negara memiliki model sistem pendidikan formal yang mana hal ini sifatnya wajib. Sekolah diharapkan dapat memenuhi kebutuhan masyarakat dalam bidang pendidikan yang bermutu. Pendidikan yang bermutu adalah pendidikan yang mampu meningkatkan kualitas siswa dengan membebaskan mereka dari ketidaktahuan, ketidakmampuan, ketidakberdayaan, serta akhlak dan keimanan yang buruk. Pendidikan yang bermutu lahir dari sistem perencanaan yang baik, materi dan tata kelola yang terstruktur, dan disampaikan oleh guru yang kompeten. Guru yang berkualitas adalah kunci utama dalam mewujudkan pendidikan yang bermutu [8].

## **2.5 RAD (*Rapid Application Development*)**

*RAD (Rapid Application Development)* merupakan pendekatan yang berbasis objek terhadap pengembangan sebuah sistem yang berisikan metode pengembangan, serta perangkat lunak yang digunakan selama proses pengembangan. Metode *RAD* memiliki tiga tahapan, yaitu *planning on requirements*, *RAD design workshop*, dan *Implementation*.ni. Dengan penerapan metode *RAD* dalam sebuah pengembangan, maka pengembangan dapat dilakukan dengan cepat dan dapat dilaksanakan bersamaan dengan pengujian terhadap pengguna. Metode *RAD* juga bertujuan untuk mempersingkat waktu pengembangan dan memotong pengeluaran dengan cara melibatkan pengguna

secara langsung dalam proses pengembangan. Penerapan metode RAD memungkinkan pengembang untuk mengetahui kebutuhan dari masing-masing pengguna dengan tepat.



**Gambar 2. 1 Tahapan Pengembangan RAD[1]**

Tahapan pengembangan aplikasi tiap fase *RAD* ada 4 tahap, meliputi:

a. *Requirements Planning.*

Pada tahap *requirement planning phase* pengembang bertemu dengan pengguna untuk mendiskusikan tujuan dari sistem yang ingin dibangun dan mengidentifikasi kebutuhan informasi untuk dapat mencapai tujuan tersebut. Tahap ini memiliki orientasi untuk memecahkan masalah bisnis yang dihadapi dan fokus dari tahap ini adalah untuk mencapai tujuan bisnis.

b. *User Design*

Tahap ini merupakan fase untuk merancang sebuah sistem yang akan dibangun. Penganalisis dan pemrogram saling bekerjasama dalam membangun sistem, serta menunjukkan pola kerja dan *visual* desain kepada pengguna, pengguna merespon *prototype* yang telah dirancang. Pemrogram dan penganalisis memperbaiki dan menganalisis dari modul-modul yang telah dirancang berdasarkan respon dari pengguna.

c. *Construction*

*Construction* diawali dari tahap desain hingga model kerja. *Developer* memastikan bahwa sistem atau aplikasi yang dibangun berfungsi dengan baik dan sesuai dengan

yang diharapkan *user*. Tahapan pertama yang dilakukan yaitu membangun *front-end* dari sistem/aplikasi. Setelah *front-end* dibangun selanjutnya dilakukan pembangunan *back-end* sistem/aplikasi dimana proses back-end dilakukan dengan menerapkan *framework* tertentu.

#### d. *Cutover*

Tahap ini merupakan tahap terakhir dari metode RAD. Pengujian pada penelitian ini dilakukan setelah *construction* baik dari segi *front-end* dan *back-end* telah sesuai. Pengujian dilakukan dengan meninjau kembali sistem yang sudah jadi. Setelah dilakukan peninjauan dan sistem sudah layak digunakan, maka diberikan pelatihan kepada *user* [1].

## **2.6 Database**

*Database* adalah kumpulan data yang disimpan dengan sistem tertentu, dan saling berhubungan, sehingga dapat dikelola dengan mudah dan merupakan sekumpulan data yang merepresentasikan dan digunakan untuk tujuan tertentu oleh satu atau lebih pengguna. Data merupakan fakta-fakta yang diketahui, direkam dan memiliki makna tersirat.

*Database* adalah kumpulan dari data yang mempunyai hubungan antara satu dengan yang lainnya, yang tersimpan dalam perangkat keras komputer dan dapat digunakan pada perangkat lunak untuk memanipulasinya (Sauerwein & Dalton, 1985). Dari definisi ini, ada tiga hal yang berhubungan dengan basis data, yaitu sebagai berikut:

1. Data yang berada pada komputer itu sendiri yang diorganisasikan dalam bentuk basis data.
2. Simpanan permanen (*storage*) digunakan untuk menyimpan basis data. Simpanan tersebut salah satu bagian dari teknologi perangkat keras yang digunakan pada sistem informasi. Simpanan permanen pada umumnya berupa sebuah hard disk
3. Perangkat lunak untuk memanipulasi data. Perangkat lunak ini dapat dibuat sendiri dengan menggunakan bahasa pemrograman komputer dapat juga dibeli

dalam bentuk paket. Banyak paket perangkat lunak yang disediakan untuk memanipulasi basis data. Paket perangkat lunak ini disebut dengan database manajemen sistem [9].

## **2.7 PHP**

*PHP* merupakan kepanjangan dari (*Hypertext Preprocessor*). *PHP* merupakan bahasa pemrograman web yang berjalan di sisi server. *PHP* merupakan bahasa yang bersifat *open source*, atau sumber terbuka dimana para pengguna bahasa pemrograman ini bebas bebas untuk menggunakan atau mengeksplornya. *PHP* merupakan bahasa pemrograman yang menggunakan sintaks keturunan *C/C++*. Ekstensi dari file *PHP* diakhiri dengan *.php*. sehingga dari ekstensi tersebut akan tahu bahwa file tersebut merupakan *file* dari *PHP*. Versi *PHP* terus berkembang membuat *PHP* menjadi bahasa pemrograman yang dinamis dan mengikuti perkembangan teknologi [10].

## **2.8 Framework**

*Framework* yaitu sebuah kerangka kerja yang digunakan untuk mengembangkan situs *website* dan diciptakan untuk membantu *web developer* dalam menulis baris kode. Dengan menggunakan *framework* penulisan kode akan jauh lebih mudah, cepat, dan terstruktur rapi. Fungsi dari *Framework* ini adalah *Framework* beroperasi dengan menyediakan struktur atau kerangka kerja yang telah dirancang sebelumnya untuk konstruksi aplikasi atau situs *web*. Dalam pengembangan perangkat lunak atau aplikasi, *framework* membantu pengembang menghemat waktu dan upaya yang biasanya diperlukan untuk membangun fitur dan fungsi dasar. Pengembang fokus terarah pada pengembangan fitur khusus dan fungsionalitas yang unik [11].

## **2.9 Rekayasa Perangkat Lunak**



Rekayasa Perangkat Lunak (*Software Engineering*) adalah suatu disiplin ilmu yang berkaitan dengan proses pembuatan, pengembangan, pemeliharaan, dan dokumentasi perangkat lunak secara sistematis dan terstruktur. Tujuannya untuk menghasilkan perangkat lunak yang berkualitas, efisien, dan dapat dipercaya. Proses pengembangan perangkat lunak terdiri dari beberapa tahapan mulai dari

analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan. Proses ini dilakukan secara iteratif dan incremental untuk memastikan bahwa perangkat lunak yang dihasilkan sesuai dengan kebutuhan pengguna dan dapat dikembangkan dengan mudah di masa depan. Pengujian perangkat lunak dilakukan untuk memastikan bahwa perangkat lunak yang dihasilkan berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Pengujian dilakukan dalam beberapa tahap, mulai dari pengujian unit, integrasi, sistem, hingga pengujian penerimaan [12].

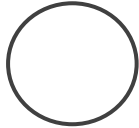
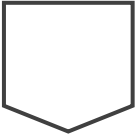


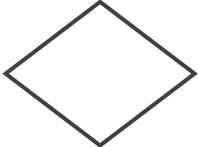


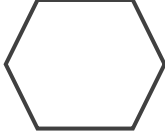
### 2.10 Flowchart







*Flowchart* merupakan diagram alir untuk menggambarkan langkah, urutan dan keputusan sebuah proses dalam suatu sistem. *Flowchart* menggunakan simbol-simbol untuk menunjukkan aktivitas, kondisi maupun alur logika. Fungsi *flowchart* yang utama adalah untuk mendokumentasikan, merencanakan maupun mengkomunikasikan proses yang rumit menjadi diagram. Tujuannya agar lebih mudah dipahami orang awam, memberikan gambaran proses secara visual ketika proyek akan dimulai, mengelola alur kerja menjadi lebih mudah dipahami dan terkontrol, mendokumentasikan alur kerja sehingga tidak perlu menjelaskan ulang ke semua pihak serta menjadi jembatan untuk semua pihak dalam memahami alur kerja[13]. Simbol flowchart ditunjukkan pada table 2.1

**Tabel 2.1 Simbol Flowchart [13]**

No.	Simbol	Nama	Keterangan
1.		<i>Flow</i>	Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan yang lain.
2.		<i>Terminator</i>	Simbol yang menyatakan awal atau akhir suatu kegiatan.



3.		<i>On-page-reference</i>	Simbol yang berfungsi untuk keluar – masuk atau penyambungan proses dalam lembar kerja yang sama.
4.		<i>off-page-reference</i>	Simbol untuk keluar -masuk atau penyambungan proses pada lembar kerja yang berbeda.
5.		<i>Process</i>	Simbol yang menyatakan suatu proses yang dilakukan komputer.
6.		<i>Manual Operation</i>	Simbol yang menyatakan proses yang tidak dilakukan oleh komputer.
7.		<i>Decision</i>	Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban yang berbeda, ya atau tidak.
8.		<i>Input – Output</i>	Simbol yang menyatakan proses input dan output tidak tergantung dengan peralatan.
9.		<i>Manual operation</i>	Simbol yang menyatakan pengolahan data tidak dilakukan oleh computer.
10.		<i>Preparation</i>	Simbol untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan yang memberikan nilai awal.


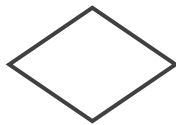
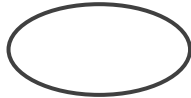


11.		<i>Predefine Proses</i>	Simbol yang menggambarkan pelaksanaan bagian (sub-program atau prosedur).
12.		<i>Display</i>	Simbol yang menyatakan suatu peralatan output yang digunakan.
13.		<i>Disk and On-line Storage</i>	Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
14.		<i>Magnetik tape Unit</i>	Simbol yang menggambarkan input berasal dari pita magnetik atau output disimpan ke dalam pita magnetik.
15.		<i>Punch Card</i>	Simbol yang menunjukkan bahwa input berasal dari kartu atau output ditulis ke kartu.
16.		<i>Document</i>	Simbol yang menggambarkan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

### 2.11 Entity Relationship Diagram (ERD)

*ERD* merupakan gambaran grafis suatu model data yang menyertakan deskripsi detail dari seluruh *entity*, *relationship*, dan *constraint*. *Entity Relationship Diagram (ERD)* digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh sistem analis digunakan dalam tahap analisis persyaratan proyek pengembangan. *Entity Relationship Diagram (ERD)* berupa metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual [14].

Berikut simbol-simbol yang terdapat pada *entity relationship diagram* dapat dilihat pada table 2.2

**Tabel 2. 2 Simbol ERD [14]**

No.	Simbol	Nama	Keterangan
1.		Entitas	Entitas merupakan suatu yang nyata atau abstrak dimana kita akan menyimpan data
2.		Relasi	Relasi merupakan hubungan alamiah yang terjadi antar satu atau lebih.
3.		Atribut	Atribut merupakan ciri umum atau sebagian besar intisari pada entitas tertentu.
4.		Garis	Garis merupakan penghubung antara relasi dengan entitas relasi dan entitas dengan atribut.
5.		<i>Input-Output</i>	Input-Output merupakan proses <i>input</i> atau <i>output</i> data, parameter, informasi.

### 2.12 UML (*Unified Modeling Language*)

*UML* adalah salah satu *tool/model* untuk merancang pengembangan software yang berbasis *object-oriented*. *UML* sendiri juga memberikan standar penulisan sebuah sistem blueprint, yang meliputi konsep proses bisnis, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen yang diperlukan dalam sistem software. *UML* digunakan sebagai alat komunikasi yang efektif antara pengembang perangkat lunak, pemangku kepentingan, dan tim pengembangan untuk memahami, merencanakan, dan merancang sistem perangkat lunak yang kompleks.[15]

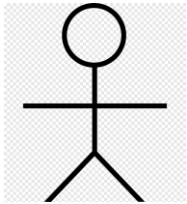
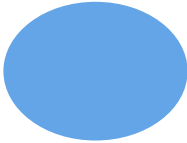

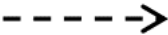
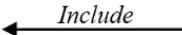
Beberapa jenis diagram *UML* yang umum digunakan, antara lain:

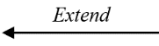
### 1. *Use Case Diagram*

Menurut Rosa dan Shalahuddin, (2014) mengemukakan *Use Case Diagram* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu [16].

Berikut adalah simbol-simbol yang ada pada diagram *use case* dapat dilihat pada tabel 2.3

**Tabel 2. 3 Simbol *Use Case* [16]**

No.	Notasi	Keterangan	Simbol
1.	<i>Actor</i>	Pengguna sistem atau yang berinteraksi langsung dengan sistem	
2.	<i>Use Case</i>	Lingkaran elips dengan nama <i>use case</i> nya tertulis ditengah lingkaran	
3.	<i>Association</i>	Garis yang berfungsi menghubungkan <i>actor</i> dengan <i>usecase</i> .	
4.	<i>Relasi</i>	Sebagai penghubung antara <i>actor usecase, use case-usecase</i> dll	
5.	<i>Include Relationship</i>	Memungkinkan suatu <i>usecase</i> untuk	

		menggunakan fungsionalitas yang disediakan oleh <i>usecase</i> yang lainnya.	
6.	<i>ExtendRelationship</i>	Memungkinkan <i>usecase</i> memiliki kemungkinan untuk memperluas fungsionalitas yang disediakan oleh <i>usecase</i> yang lainnya.	

**-HALAMAN INI SENGAJA DIKOSONGKAN-**