



BAB II
TINJAUAN PUSTAKA
DAN LANDASAN TEORI

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Ada beberapa penelitian yang dahulu dilakukan yaitu penelitian oleh Pratama,D.M (2021) yang mengusulkan perancangan sistem informasi rapor berbasis web menggunakan Metode V-Model Pada SMK Negeri 7 Pekanbaru yaitu pada SMK Negeri 7 Pekanbaru dalam pengolahan nilai rapor, masih menggunakan *Microsoft excel* untuk input nilai siswa, dengan cara tersebut dinilai masih kurang efektif dan efisien dalam pengolahannya, sehingga dibuat sistem e-rapor untuk mempermudah dalam pengelolaan rapor siswa di sekolah tersebut. Sistem ini menggunakan metode analisa *Object Oriented Analysis and Design* (OOAD) dan pengembangan sistem *v-model*. Output yang dihasilkan dalam sistem yaitu rapor siswa dan laporan setiap kelas [7]. Penelitian lain oleh Susanto, A.H, dkk (2020), mengusulkan *designing student's registration book and student's report card processing application for Pangudi Luhur Don Bosko Elementary School Semarang* yaitu pada SDN Pangudi Luhur Don Bosko Semarang dalam hal penyimpanan data masih memiliki keterbatasan sehingga dibuat sistem agar dapat mempermudah dalam olah data dan mengetahui factor-faktor pengguna untuk menggunakan sistem ini. Sistem ini dibuat berbasis web dengan menggunakan bootstrap dan menggunakan bahasa pemrograman PHP dengan tempat penyimpanannya yaitu MySQL [8].

Penelitian terdahulu oleh Muthaharoh, Z (2020) yang mengusulkan perancangan aplikasi rapor di SD Tiara Harapan Jaya menggunakan *java neatbeans* dari penelitian tersebut yaitu memiliki tujuan untuk membuat sistem dengan menggunakan Java Netbeans, permasalahan dalam studi kasus tersebut yaitu dalam pembuatan rapor masih dilakukan secara manual yaitu wali menuliskan nilai ke dalam rapor, kemudian dalam metode dan desain yang digunakan dalam pembuatan sistemnya yaitu metode *Graunded Research*. Pengimplementasiannya untuk sistem ini menggunakan Diagram Alir Data, Entity Relationship Diagram, Rancangan File juga menggunakan XAMPP dan Netbeans [9]. Penelitian lain oleh Rahayu,T.K, dkk (2020),

yang mengusulkan *application report process of islamic school based on Pesantren Boarding Using Waterfall Model* bahwa pengembangan aplikasi perangkat lunak ini dibuat untuk pembuatan laporan sekolah factor permasalahannya yaitu sekolah memiliki banyak siswa sehingga kemampuan mengolah data dibutuhkan yang lebih cepat dan mudah, dalam pengolahannya masih menggunakan kalkulator sehingga memungkinkan terjadinya kesalahan, sehingga perlu dibuat sistem untuk membantu dalam mengolah agar lebih efektif dan efisien. Dalam pembuatannya menggunakan metode *Software Development Life Cycle* (SDLC) dan metode *waterfall* [10].

Penelitian lainnya yaitu oleh Fitriani,F dan Juanita,F (2019) mengusulkan perancangan sistem informasi akademik berbasis dekstop dengan studi kasus: SMA Muhammadiyah 4 Jakarta, SMK Muhammadiyah 4 Jakarta dalam proses pengolahan akademik masih belum terkomputerisasi sehingga dapat terjadi masalah yaitu nilai sering tertukar dan pengolahan nilai yang membutuhkan waktu lama, juga dalam informasi data rapor siswa masih kurang lengkap karena data orang tua dan nilai sikapnya masih sering kosong. Sehingga dibuat sistem yang dapat membantu dalam pengolahan data akademik dengan menggunakan bahasa pemrograman *Microsoft Visual Basic.net* dengan data basenya menggunakan MySQL [11].

Pada penelitian ini berbeda dengan penelitian sebelumnya, yaitu sebuah sistem informasi pengolahan nilai rapor siswa di SMP Purnama Kesugihan yang dikembangkan dengan menggunakan bahasa pemrograman *Framework PHP* yaitu *CodeIgniter* dan database MySQL. Sistem informasi ini menggunakan metode pengembangan *waterfall* tetapi tanpa adanya proses pemeliharaan sistem. Sistem ini dilengkapi *reminder* sebagai fitur pelengkap untuk siswa apabila terdapat nilai yang masih kosong pada saat pengolahan nilai menjadi rapor, dengan adanya fitur *reminder* bertujuan untuk dapat mengingatkan siswa agar segera mengisi kekosongan nilai. Sistem ini diharapkan dapat menjaga kualitas nilai, menghemat tenaga guru dan juga mempermudah dalam pengolahan nilai rapor di SMP Purnama Kesugihan serta kemudahan untuk siswa dalam mengakses nilai rapor kapanpun dan dimanapun.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem adalah kumpulan dari beberapa elemen yang berinteraksi untuk mencapai suatu tujuan tertentu kesatuan [12]. Agar sebuah sistem

dapat dikatakan sebagai sistem yang baik maka sistem memiliki beberapa karakteristik yaitu:

1. **Komponen**
Sistem berupa komponen yang saling berinteraksi, dan bekerja sama untuk membentuk satu kesatuan. Komponen dari sistem dapat terdiri dari banyaknya subsistem atau bagian dari sistem yang dibentuk.
2. **Batasan Sistem (*boundary*)**
Batasan sistem mampu membatasi sistem satu dengan sistem yang lain, sehingga batasan memungkinkan suatu sistem dipandang menjadi satu kesatuan, batasan sistem inilah yang membuktikan suatu ruang lingkup dari sistem yang dibentuk.
3. **Lingkungan Luar Sistem (*environment*)**
Segala sesuatu yang merupakan batas dari luar yang mempengaruhi operasi sistem disebut lingkungan luar sistem. Lingkungan yang bersifat menguntungkan harus dijaga, sedangkan yang bersifat merugikan harus dikendalikan agar tidak mengganggu sistem yang berlangsung.
4. **Penghubung sistem (*interface*)**
Sebagai penghubung yang menghubungkan subsistem satu dengan yang lainnya, sehingga sumber daya dapat mengalir secara berhubungan.
5. **Masukkan Sistem (*Input*)**
Segala sesuatu yang dimasukkan ke dalam sistem yang kemudian akan diproses melalui sistem.
6. **Keluaran Sistem (*Output*)**
Hasil dari pemrosesan yang diolah untuk menjadi hasil keluaran yang berguna.
7. **Pengolah Sistem (*Process*)**
Sistem akan bekerja mengolah apa yang menjadi masukkan menjadi keluaran yang memiliki daya guna.
8. **Sasaran Sistem (*Objective*)**
Tujuan dari dibuatnya sistem, sasaran dari sistem sangat menentukan input yang dibutuhkan untuk sistem dan menghasilkan keluaran yang sesuai pada sistem.

Informasi adalah hal yang paling mendasar dan diperlukan oleh suatu kegiatan dalam pengambilan keputusan agar dapat menghindari kesalahan dan sebagai olah data sehingga memiliki daya guna bagi

penerima informasi. Informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga memiliki arti bagi para penerimanya [13]. Informasi memiliki karakteristik, sebagai berikut:

1. Relevan, mempunyai makna dan dapat digunakan secara tepat untuk mengambil keputusan.
2. Andal, memiliki tingkat keandalan yang tinggi.
3. Lengkap, memiliki penjelasan yang jelas dan rinci.
4. Tepat Waktu, harus dalam kondisi yang terbaru dan tidak dalam bentuk usang.
5. Dapat dipahami, informasi yang disajikan jelas sehingga mudah dimengerti.

Sistem informasi adalah suatu sistem yang terdiri dari banyak komponen sistem, yaitu *software*, *hardware* dan *brainware* yang kemudian memproses data berupa informasi menjadi sebuah keluaran yang berguna untuk mencapai sasaran tertentu dalam suatu perkumpulan [13].

2.2.2 Framework

Framework adalah suatu struktur yang didalamnya terdapat *class-class*, pustaka-pustaka dan infrastruktur *run-time* untuk membuat *web*, dengan menggunakan *framework* memungkinkan untuk membuat *web* dengan mudah dan cepat. *Framework* saat ini tersedia berbagai macam untuk mengembangkan *web*, salah satunya yaitu *Codeigniter*. *Framework codeigniter* ini *open source* dan mudah untuk dipelajari. *Codeigniter* merupakan *framework* dengan bahasa pemrograman *PHP* dan menggunakan konsep *MVC (Models-View-Controller)* untuk memisahkan data dan presentasi sehingga pengembangan *web* akan mudah dan cepat dalam proses pengelolaannya [14].

2.2.3 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah aplikasi yang berasal dari adanya pendekatan kuantifiabel, disiplin dan secara sistematis kepada proses, ekspansi dan pemeliharaan perangkat lunak yaitu aplikasi dari Rekayasa perangkat lunak itu sendiri. Definisi lain Rekayasa Perangkat Lunak adalah pembuatan dan penggunaan banyaknya prinsip suatu keahlian untuk membuat perangkat lunak yang bernilai ekonomis, handal dan berdaya guna pada mesin nyata [15].

A. Metode Pengembangan Sistem

Metode pengembangan yang digunakan dalam pembuatan Sistem Informasi Pengolahan Nilai Rapor Siswa di SMP Purnama Kesugihan yaitu metode *waterfall*. Metode *waterfall* termasuk dalam *Classic Life Cycle* atau biasa disebutkan siklus hidup klasik, yang mana pada metode ini menggambarkan pendekatan yang sistematis dengan menekankan siklus yang berurutan. Disebut *waterfall* karena kemajuan dipandang sebagai terus mengalir dan melewati fase-fase perencanaan, pemodelan, implementasi dan pengujian [5]. Berikut adalah tahapan metode *waterfall*:

- a. Analisis Kebutuhan
Pada tahap pertama ini pengembang perangkat lunak harus mengetahui kebutuhan dan informasi serta kegunaannya dalam perangkat lunak yang dikembangkan. Metode Analisis Kebutuhan ini dapat dilakukan dengan berbagai cara wawancara, observasi dan sebagainya. Kemudian dari data yang sudah diperoleh tersebut dikaji agar kebutuhan sistem untuk pengembangan perangkat lunak dapat dilakukan dengan baik.
- b. Desain Sistem
Pada tahap perancangan Desain Sistem ini berfokus pada pembuatan desain *interface* yang disesuaikan dengan kebutuhan sistem yang sudah dianalisis pada tahapan sebelumnya. Pembuatan Desain Sistem ini bertujuan membantu memberikan gambaran lengkap mengenai apa saja yang harus dikerjakan dalam sistem.
- c. Penulisan Kode Program
Mengimplementasikan hasil desain ke dalam kode atau bahasa yang dapat dimengerti oleh komputer, dalam hal ini menggunakan bahasa pemrograman tertentu. Dalam tahap ini menggunakan bahasa pemrograman *framework* PHP *CodeIgniter*, Database MySQL.
- d. Pengujian Program
Setelah proses penulisan kode selesai, kemudian dilakukan pengujian pada program. Pengujian pada eksternal fungsi ini dilakukan untuk memastikan sistem sudah berjalan sempurna tanpa adanya kesalahan (*error*) pada program juga memastikan bahwa masukan sudah sesuai. Pengujian ini berpusat pada bagian internal perangkat untuk memastikan bahwa semua pernyataan sudah teruji. Setelah semuanya diuji, kemudian dilakukan proses

perbaikan agar menjadi lebih sempurna dari sebelumnya.

- e. Penerapan Program
Perangkat lunak yang sudah jadi kemudian dijalankan dan dioperasikan oleh penggunanya.

B. Metode Pengujian Sistem

Metode *black-box* atau *Behavioral Testing* adalah metode yang hanya berfokus untuk mengamati pada *input* dan *output* aplikasi dan menunjukkan fungsi aplikasi tentang cara beroperasinya tanpa mengetahui struktur kode dari aplikasi [6]. Keuntungan dari metode ini yaitu pengujian tidak harus memiliki pengetahuan tentang program dan pengujian juga tidak harus memeriksa keseluruhan kode, dan hanya berfokus pada *output* yang dihasilkan. Sehingga bertujuan untuk menunjukkan fungsi dari aplikasi tentang cara bekerja, apakah *input* dan *output* sudah sesuai dengan yang diharapkan.

Berikut adalah beberapa teknik yang dapat digunakan, sebagai berikut:

- a. *All Pair Testing*
Dikenal sebagai *pairwise testing* dan digunakan untuk menguji adanya kemungkinan kombinasi berdasarkan *input* parameter.
- b. *Boundary Value Analysis*
Berfokus dalam pencarian *error* dari dalam maupun luar perangkat lunak.
- c. *Cause-effect graph*
Teknik ini menggunakan grafik sebagai dasar dan menggambarkan hubungan antara efek dan penyebab dari *error*.
- d. *Equivalence Partitioning*
Teknik ini membagi data *input* dari perangkat lunak menjadi beberapa pemisah data.
- e. *Fuzzing*
Teknik ini mencari bug dalam perangkat lunak dengan cara *input* data yang tidak sempurna.
- f. *Orthogonal Array Testing*
Teknik digunakan jika *input* bernilai kecil, tetapi berat jika digunakan dalam ukuran besar.
- g. *State transition*
Teknik yang dalam bentuk grafik berguna untuk menguji mesin dan navigasi dari UI.

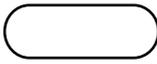
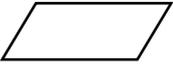
2.2.4 UML (Unified Modelling Language)

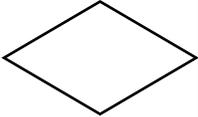
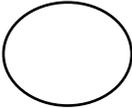
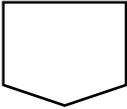
UML (*Unified Modelling Language*) adalah suatu bahasa spesifikasi menjadi standar pada visualisasi, perancangan, dan dokumentasi sistem aplikasi [16]. Saat ini UML menjadi bahasa standar yang digunakan dalam penulisan arsitektur. Tujuan penggunaan UML agar dapat memberikan bahasa permodelan secara visual kepada pengguna, hingga memberikan model yang siap digunakan. Berikut merupakan jenis jenis UML:

A. *Flowchart*

Flowchart adalah bagan yang menggunakan simbol dan menggambarkan adanya proses dengan detail. Simbol *flowchart* dipakai sebagai alat untuk membantu dalam mendefinisikan sebuah proses. Proses dalam *flowchart* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan. Aktifitas yang digambarkan kemudian didefinisikan dengan jelas sehingga dimengerti oleh pembacanya [17]. Simbol *flowchart* dapat dilihat pada tabel 2.1.

Tabel 2. 1 Simbol *Flowchart*

No.	Nama	Simbol	Keterangan
1.	Terminal		Menyatakan permulaan atau akhir suatu program
2.	<i>Input dan Output</i>		Menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya
3.	Proses		Menyatakan proses yang dilakukan oleh komputer

No.	Nama	Simbol	Keterangan
4.	Simbol arus/ <i>flow</i>		Menyatakan jalannya arus suatu proses
5.	<i>Decision / keputusan</i>		Menunjukkan suatu kondisi tertentu yang akan menghasilkan data kemungkinan dengan jawaban ya atau tidak
6.	<i>Connector (on-page)</i>		Menghubungkan proses ke proses lain dalam halaman yang sama
7.	<i>Connector (off-page)</i>		Menghubungkan proses ke proses lain dalam halaman yang berbeda
8.	<i>Manual Operation</i>		Menunjukkan kegiatan yang tidak dilakukan oleh komputer

No.	Nama	Simbol	Keterangan
9.	Dokumen		Mencetak <i>output</i> dalam dokumen melalui printer
10.	Manual <i>Input</i>		Memasukkan data secara manual dengan menggunakan online keyboard

B. *Use Case Diagram*

Use Case Diagram adalah suatu pemodelan untuk menentukan kelakuan pada sistem informasi yang akan dibangun. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada pada sistem dan siapa saja yang dapat menggunakan fungsi tersebut [17]. Simbol *use case* diagram dapat dilihat pada tabel 2.2.

Tabel 2. 2 Simbol *Use Case Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Use Case</i>		Fungsi yang disediakan sistem untuk unit yang saling bertukar pesan antar unit dan aktor
2.	<i>Actor</i>		Orang yang berinteraksi dengan sistem informasi yang dibuat

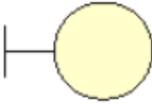
No.	Nama	Simbol	Keterangan
3.	<i>Asosiation</i>		Penghubung komunikasi antar aktor dan <i>use case</i>
4.	<i>Ekstend</i>		Penghubung komunikasi antar aktor dan <i>use case</i>
5.	<i>Include</i>		Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
6.	Generalisasi		Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .

C. *Sequence Diagram*

Sequence Diagram adalah gambaran kelakuan objek terhadap use case dengan mendefinisikan pesan yang dikirim dan diterima antar objek [17]. Simbol *sequence diagram* dapat dilihat pada tabel 2.3.

Tabel 2. 3 Simbol *Sequence Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Actor</i>		Menggambarkan orang yang berinteraksi dengan sistem

No.	Nama	Simbol	Keterangan
2.	<i>Entity Class</i>		Menggambarkan hubungan yang akan dilakukan
3.	<i>Boundary Class</i>		Menggambarkan sebuah gambaran dari <i>form</i>
4.	<i>Control Class</i>		Menggambarkan hubungan antara <i>boundary</i> dengan tabel
5.	<i>Lifeline</i>		Menggambarkan tempat mulai dan berakhirnya <i>message</i>
6.	Line Message		Menggambarkan pengiriman <i>message</i> antar <i>class</i>

D. *Class Diagram*

Class Diagram merupakan sebuah diagram untuk menjelaskan adanya hubungan antar *class* dalam sistem dan menjelaskan bagaimana cara agar saling terhubung dan berkolaborasi [18]. Simbol *Class Diagram* dapat dilihat pada tabel 2.4.

Tabel 2. 4 Simbol *Class Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Generalization</i>		Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
2.	<i>Nary Association</i>		Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3.	<i>Class</i>		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4.	<i>Realization</i>		Operasi yang dilakukan oleh suatu objek
5.	<i>Dependency</i>		Relasi antarkelas dengan makna kebergantungan antarkelas
6.	<i>Association</i>		Diagram yang menghubungkan antara objek satu dengan objek lainnya.

2.2.5 Rekayasa Web

Rekayasa *web* adalah aplikasi yang merupakan subdisiplin dari rekayasa perangkat lunak dan melakukan pendekatan yang sistematis, disiplin, terukur untuk pengembangan, operasi dan pemeliharaan aplikasi berbasis *web*. Rekayasa *web* membantu dalam menyediakan metodologi untuk merancang, mengembangkan, memelihara dan melibatkan aplikasi

web. Tujuan utama dari rekayasa *web* adalah kesuksesan dalam mengatur kompleksitas serta keanekaragaman pengembangan aplikasi *web* [19].

2.2.6 Basis Data

Basis data (*database*) adalah kumpulan data yang saling berkaitan untuk memenuhi kebutuhan informasi. Data tersebut biasanya terdapat pada tabel-tabel yang saling berhubungan satu sama lain. Sehingga *database* merupakan kumpulan dari banyaknya data yang saling berhubungan dan menggambarkan adanya suatu hubungan tabel satu dengan tabel lainnya untuk kebutuhan informasi [17].

A. DBMS (Database Management System)

Database Management System (DBMS) adalah perangkat lunak yang memungkinkan pemakai untuk dapat menjelaskan, mengelola, dan mengontrol akses ke basis data. Komponen DBMS diklasifikasikan menjadi *Hardware* dan *Software*, *People* dan *Procedure*, *Data* [20].

B. MySQL

MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL *Database Management System* atau DBMS dari banyaknya DBMS seperti *Oracle*, *Ms SQL*, *Postagre SQL* dan lainnya [21].

2.2.7 Rapor

Rapor adalah laporan nilai siswa yang didapatkan selama menempuh pendidikan dalam satu semester oleh siswa [2]. Penilaian berfungsi untuk mengukur sejauh mana siswa dapat mencapai tujuan yang sudah ditetapkan dalam pembelajaran, penilaian membantu guru untuk mengevaluasi keefektifan kurikulum dan strategi belajar-mengajar siswa. Rapor juga merupakan laporan hasil dari suatu kegiatan yang disusun dengan tepat dan benar [22].

Nilai dapat dibedakan menjadi 2 macam yaitu [23]:

- 1) Nilai Intrinsik, adalah nilai dari sesuatu yang sudah bernilai dari sejak semula.
- 2) Nilai instrumental adalah nilai yang digunakan untuk mencapai suatu tujuan dari sesuatu yang sudah bernilai.

Reminder adalah sebuah pesan yang membantu seseorang untuk mengingat sesuatu. *Reminder* dapat berfungsi saat informasi kontekstual digunakan untuk menyajikan informasi pada waktu yang tepat dan tempat yang sesuai. *Reminder* digunakan dalam proses manajemen waktu untuk memberi alarm peringatan berupa pemberitahuan [24].

2.2.8 Usability Testing

Usability adalah atribut yang memiliki kualitas dan menjelaskan atau mengukur kemudahan dalam menggunakan suatu antar muka (*interface*). Kata “*usability*” berarti suatu metode untuk meningkatkan kemudahan pemakaian selama proses desain [25].