

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Sebagai acuan dalam pengembangan Sistem Informasi Pengelolaan Tabungan Siswa Berbasis *Website* Dengan Notifikasi *WhatsApp*, maka penulis mencari referensi beberapa penelitian terdahulu yang berkaitan secara langsung maupun tidak langsung dengan penelitian yang penulis kembangkan.

Penelitian pertama oleh Dewi Triyanti dan Tendi Dermawan “Membangun Sistem Informasi Pengelolaan Data Tabungan Siswa Berbasis *Web*” [4]. Permasalahan pada penelitian ini adalah pencatatan transaksi yang masih manual sehingga menyebabkan kesalahan dalam pencatatan data penyetoran dengan jumlah banyak, sering hilangnya buku tabungan siswa dan kurangnya transparansi data bagi orang tua siswa. Penelitian ini menggunakan metode *Web Development Life Cycle* (WDLC). Hasil dari penelitian ini ditujukan untuk memberikan solusi kepada pihak sekolah SMK Islam Nurul Iman untuk mengganti sistem yang masih manual ke sistem yang berbasis *web*, sehingga mempermudah dalam penanganan transaksi tabungan, pencarian data tabungan, input data transaksi penyetoran, memperkecil kemungkinan hilangnya buku tabungan siswa, dan kemudahan pada sisi wali murid dalam menabung dan melihat jumlah/riwayat saldo tabungan.

Penelitian kedua oleh Hayatullah khumaini, Ridarmin, Siti Khamariah “Aplikasi Pengelolaan Data Tabungan Siswa Pada Yayasan Pendidikan Al Ikhlas Dumai” [5]. Permasalahan pada penelitian ini adalah semua transaksi tabungan hanya dilakukan secara manual. Penelitian ini dikembangkan menggunakan metode *waterfall*. Hasil penelitian ini bertujuan supaya memudahkan proses entri data tabungan siswa yang dilakukan oleh petugas karena proses yang berjalan saat ini masih secara manual sehingga proses pendataan tabungan tidak efisien dan butuh waktu lama dalam perhitungan saldo tabungan siswa.

Penelitian ketiga oleh Nia Silfiyanti, Khabibatul Anif, Faridatun Nadziroh “Sistem Informasi Pencatatan Buku Tabungan Siswa Berbasis *Web* di Madrasah Ibtidaiyah Daroyissalam Desa Kertosono Kecamatan Sidayu Kabupaten Gresik” [6]. Permasalahan pada penelitian ini adalah

masih terdapat kekurangan pada penyimpanan yang masih belum terorganisir dan belum terintegrasi dengan sistem. Oleh karena itu, jika terjadi masalah pencatatan data maka harus melakukan pengecekan ulang untuk menghindari kesalahan yang fatal. Penelitian ini bertujuan untuk meningkatkan keefektifan dan keefisienan saat pencarian, pembuatan, dan penyimpanan data tabungan karena sebelumnya penyimpanan data tabungan siswa di MI Daroyissalam. Dalam penelitian ini menggunakan metode *waterfall* dan pemodelan *flowchart* dan metode pengujian *blackbox*. Penelitian ini menghasilkan sistem informasi pencatatan buku tabungan berbasis web yang dapat meningkatkan 90% keefektifan dan keefisienan dalam pencarian, pembuatan, serta penyimpanan data tabungan siswa.

Penelitian keempat oleh Bambang Suprpto, Henry Simanjuntak, Ardi Mahmudi “Aplikasi Tabungan Siswa SMP Negeri 3 Gadingrejo Berbasis *Web*” [7]. Sistem tabungan yang saat ini masih dilakukan secara manual menimbulkan permasalahan seperti proses pencatatan data siswa, penggunaan buku besar yang kurang efisien. Pada penelitian ini metode yang digunakan adalah metode *Extreme Programming*. Hasil penelitian ini di tujukan untuk mengurangi permasalahan yang ditimbulkan dari proses yang digunakan masih secara manual dimana proses pencatatan dan penggunaan buku besar masih kurang efisien.

Penelitian kelima oleh Andi Irfan dan Yuliana “Sistem Informasi Siswa Berbasis *Web* Pada SDN Enrekeng Kabupaten Soppeng” [8]. Permasalahan pada penelitian ini ialah sistem tabungan yang masih menggunakan buku dalam pencatatan data siswa, data setoran, data penarikan dan laporan transaksi siswa. Penelitian ini ditujukan untuk pengelolaan tabungan yang masih belum terkomputasi dan membutuhkan waktu yang cukup lama. Penelitian ini menggunakan pemodelan UML (*Unified Modeling Language*) dengan metode pengembangan sistem yang digunakan adalah metode *Waterfall*. Hasil yang diharapkan dari sistem ini adalah menggantikan sistem yang sebelumnya belum terkomputasi menjadi lebih terkomputasi.

Setelah melihat beberapa penelitian diatas maka penulis akan mengembangkan sistem dengan judul “Sistem Informasi Pengelolaan Tabungan Siswa Berbasis *Website* Dengan Notifikasi *WhatsApp*”. Hasil sistem ini bertujuan untuk mempermudah bendahara atau guru dalam pencatatan tabungan, perhitungan saldo tabungan dan laporan tabungan siswa. Perbedaan yang ada pada sistem yang akan dibangun dengan penelitian sebelumnya yaitu penulis menggunakan *Framework*

*Codeigniter* dan basis data MySQL dengan metode pengembangan sistem *Waterfall*. Selain itu pada sistem ini juga terdapat fitur notifikasi *WhatsApp* dimana fitur tersebut memberikan informasi kepada orang tua wali untuk mengetahui data tabungan anaknya. Sistem ini diharapkan dapat membantu dalam proses menabung di MI Ya BAKII Kesugihan 02.

**Tabel 2. 1** Tabel Penelitian Terdahulu

No	Penelitian Terdahulu	Tujuan	Metode	Proses	Hasil
1.	Dewi Triyanti dan Tendi Dermawan (2019), “Membangun Sistem Informasi Pengelolaan Data Tabungan Siswa Berbasis <i>Web</i> ”	Untuk mempermudah dalam penanganan transaksi tabungan, pencarian, dan input data transaksi	Metode <i>Web Development Life Cycle</i> (WDLC)	<ul style="list-style-type: none"> <li>- Penanganan transaksi tabungan</li> <li>- Pencarian data tabungan</li> <li>- Input data transaksi penyetoran tabungan</li> </ul>	Memberikan solusi kepada SMK Islam Nurul Imam dengan mengganti sistem menjadi berbasis web
2.	Hayatullah khumaini, Ridarmin, Siti Khamariah (2020), “Aplikasi Pengelolaan Data Tabungan Siswa Pada Yayasan	Memudahkan petugas dalam melakukan entri data tabungan siswa	Metode <i>Waterfall</i>	<ul style="list-style-type: none"> <li>- Pendataan tabungan</li> <li>- Perhitungan saldo</li> </ul>	Aplikasi pengelolaan data tabungan siswa untuk Yayasan tersebut.

	Pendidikan Al Ikhlas Dumai”				
3.	Nia Silfiyanti, Khabibatul Anif, Faridatun Nadziroh (2020), “Sistem Informasi Pencatatan Buku Tabungan Siswa Berbasis <i>Web</i> di Madrasah Ibtidaiyah Daroyissalam Desa Kertosono Kecamatan Sidayu Kabupaten Gresik”	Untuk meningkatkan keefektifan dan keefiseinan saat proses pencarian, pembuatan, dan penyimpanan data tabungan	Metode <i>Waterfall</i>	<ul style="list-style-type: none"> <li>- Pencarian dan pembuatan tabungan siswa</li> <li>- Penyimpanan data tabungan siswa</li> </ul>	Sistem informasi berbasis web untuk MI Daroyisasalam dengan keefektifan dan keefiseinan 90%

4.	Bambang Suprpto, Henry Simanjuntak, Ardi Mahmudi (2022), “Aplikasi Tabungan Siswa SMP Negeri 3 Gadingrejo Berbasis <i>Web</i> ”	Untuk mengurangi permasalahan yang ditimbulkan dari proses yang digunakan masih secara manual dimana proses pencatatan dan penggunaan buku besar masih kurang efisien.	Metode <i>Extreme Programming</i>	<ul style="list-style-type: none"> <li>- Pencatatan data siswa</li> <li>- Penulisan buku besar</li> <li>- Input data</li> </ul>	Menghasilkan Aplikasi Tabungan siswa berbasis web untuk SMP Negeri 3 Gadingrejo
5.	Andi Irfan dan Yuliana (2022), “Sistem Informasi Tabungan Siswa Berbasis <i>Web</i> Pada	Untuk mengatasi permasalahan dalam pengelolaan tabungan karena sistem berjalan masih belum	Metode <i>Waterfall</i>	<ul style="list-style-type: none"> <li>- Pengelolaan tabungan.</li> </ul>	Menghasilkan Sistem Informasi Tabungan Siswa sehingga sistem yang diterapkan bisa tergantikan menjadi

	SDN 79 Enrekeng Kabupaten Soppeng”	terkomputasi.			sistem yang terkomputasi.
6.	Linatus Sariroh (2023), “Sistem Informasi Pengelolaan Tabungan Siswa Berbasis <i>Website</i> dengan Notifikasi <i>WhatsApp</i> ”	Untuk memudahkan bendahara dalam proses pengelolaan data, pengolahan saldo, dan memberikan informasi kepada wali murid.	Metode <i>Waterfall</i>	<ul style="list-style-type: none"> <li>- Pencatatan data tabungan</li> <li>- Pencatatan datasiswa dan orang tua</li> <li>- Penulisan laporan tabungan.</li> <li>- Pemberitahuan informasi kepada orang tua wali.</li> <li>- Permintaan penarikan tabungan.</li> </ul>	Membangun sistem informasi tabungan bagi MI YA BAKII Kesugihan 02 yang memberikan kemudahan Kelola data dan fitur notifikasi <i>WhatsApp</i> yang akan memberikan informasi tabungan kepada wali murid.

## 2.2 Landasan Teori

### 2.2.1 Sistem Informasi

Sistem adalah satu kesatuan yang utuh yang terdiri dari beberapa bagian yang berbentuk fisik atau non-fisik saling berhubungan dan berinteraksi untuk mencapai suatu tujuan dan sasaran tertentu. Sedangkan informasi merupakan hasil dari pengolahan data yang berguna bagi yang menerimanya dan dapat digunakan untuk alat bantu pengambilan keputusan [9].

Sistem informasi adalah suatu sistem yang dibuat yang terdiri dari komponen-komponen di dalam organisasi untuk menyajikan informasi. Sistem informasi juga didefinisikan sebagai sekumpulan prosedur dalam organisasi yang dilaksanakan akan memberikan informasi untuk pengambilan keputusan. Sistem informasi menerima *input* data dan perintah, mengelola data sesuai dengan perintah, dan *output* yang dihasilkan. Komponen dari sistem informasi yaitu [10]:

- A. *Hardware* (Perangkat Keras), terdiri dari komputer, printer dan komponen perangkat keras lainnya.
- B. *Software* (Perangkat Lunak), merupakan kumpulan perintah-perintah yang ditulis sesuai dengan aturan tertentu untuk memberi perintah kepada komputer untuk melaksanakan tugas tertentu.
- C. *Data*, merupakan alat yang menjadi penghubung antara manusia dengan perangkat terjadinya proses pengolahan data.
- D. *Prosedur*, panduan dan kebijakan untuk mengoperasikan suatu sistem.

### 2.2.2 Tabungan

Tabungan menurut Undang-Undang perbankan No. 10 Tahun 1988 adalah simpanan yang penarikannya hanya dapat dilakukan menurut syarat tertentu yang disepakati, tetapi tidak dapat ditarik dengan cek, bilyet giro dan atau alat lainnya yang dipersamakan dengan itu [1].

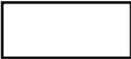
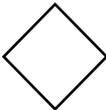
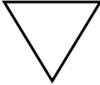
### 2.2.3 Siswa

Siswa merupakan salah satu komponen manusiawi yang berada pada posisi sentral proses belajar mengajar sebagai pihak yang memiliki tujuan dan ingin mencapainya secara optimal [4]. Siswa merupakan faktor penentu yang mempengaruhi kegiatan dan tujuan dari pembelajaran.

### 2.2.4 Flowchart

*Flowchart* atau diagram alir adalah diagram yang mempresentasikan langkah atau algoritma yang urut dalam sistem. *Flowchart* digambarkan dengan menggunakan symbol-simmol dimana setiap simbol memiliki arti proses tertentu[11]. Simbol-simbol *flowchart* dapat dilihat pada tabel 2.2.

**Tabel 2. 2** Simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1.		Proses	Digunakan untuk menunjukkan pengolahan yang dilakukan komputer
2.		<i>Manual Operation</i>	Digunakan untuk menunjukkan pengolahan yang tidak dilakukan komputer
3.		<i>Decision</i>	Digunakan untuk memilih proses berdasarkan kondisi
5.		Terminal	Digunakan untuk memulai atau mengakhiri proses.
6.		<i>Offline Storage</i>	Digunakan untuk memberitahukan bahwa data akan disimpan ke suatu media tertentu
7.		Manual Input Simbol	Digunakan untuk input data secara manual dengan keyboard
8.		Input/Output	Digunakan untuk Input dan Output.
9.		Dokumen	Digunakan untuk data masukan dan keluaran dari dokumen

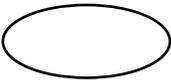
### 2.2.5 UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) merupakan bahasa yang digunakan untuk spesifikasi, visualisasi, membangun dan dokumentasi dari sistem. UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, OMT (*Object Modelling Technique*) dan OOSE (*Object Oriented Engineering*). UML digunakan untuk pemodelan sistem dengan berorientasi objek dan menciptakan suatu bahasa pemodelan yang bisa digunakan baik itu manusia ataupun komputer[12]. Adapun jenis-jenis dari UML:

#### A. *Use Case Diagram*

*Use case diagram* merupakan suatu pemodelan untuk menentukan kelakuan dari sistem yang akan dibangun. *Use Case* dapat mendeskripsikan interaksi antara sistem informasi yang dibuat dengan satu atau lebih aktor. *Use Case* digunakan untuk mengetahui fungsi yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi tersebut[13]. Simbol *Use Case* dapat dilihat pada tabel 2.3.

**Tabel 2. 3** Simbol *Use Case*

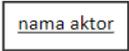
No.	Simbol	Nama	Fungsi
1.		<i>Use Case</i>	Digunakan sebagai unit yang saling bertukar pesan antar unit dan aktor.
2.		Aktor	Orang yang berinteraksi dengan sistem yang dibuat.
3.		Asosiasi	Penghubung komunikasi antar aktor dan <i>Use Case</i>
4.		<i>Extend</i>	Relasi <i>Use Case</i> yang menjadi penghubung komunikasi actor dan <i>Use Case</i> .

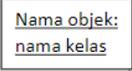
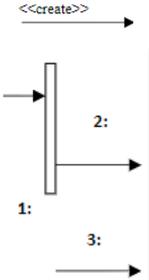
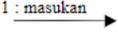
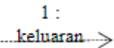
5.		<i>Include</i>	Menunjukkan bahwa suatu <i>Use case</i> seluruhnya merupakan fungsionalitas dari <i>Use case</i> lainnya.
6.		Generalisasi	Menunjukkan spesialisasi actor untuk dapat berpartisipasi dengan <i>Use Case</i> .

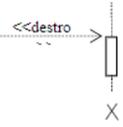
### B. *Sequence Diagram*

*Sequence* diagram merupakan gambaran kelakuan objek pada *Use Case* mendefinisikan waktu objek dan pesan yang dikirimkan dan diterima antar objek. Untuk menggambar sebuah *sequence diagram* harus diketahui objek yang terlibat dalam sebuah *use case* serta metode yang dimiliki pada kelas yang diinstansiasi menjadi objek[13]. Simbol *sequence diagram* dapat dilihat pada tabel 2.4.

**Tabel 2. 4** Simbol *Sequence Diagram*

No.	Simbol	Nama	Fungsi
1.	  	Aktor  Tanpa Waktu aktif	Orang, pproses, atau sistem lain yang berinteraksi dengan sistem informas yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi actor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor

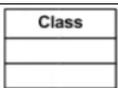
2.		Garis hidup / <i>Lifeline</i>	Menyatakan kehidupan suatu objek
3.		Objek	Menyatakan objek yang berinteraksi pesan.
4.		Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.		Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi maka operasi yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
6.		Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data ke objek lain, dan arah panah mengarah pada objek yang dikirim.
7.		Pesan tipe return	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah

			mengarah ke objek yang mengirim pesan.
8.		Pesan tipe <i>destroy</i>	Menyatakan objek mengakhiri hidup objek lain, arah panah mengarah ke objek yang diakhiri, jika ada create maka akan ada destroy.
9.		<i>Boundary</i>	Digunakan untuk menggambarkan sebuah <i>form</i> .
10.		<i>Control</i>	Digunakan untuk menghubungkan <i>Boundary</i> dengan <i>table</i> .
11.		<i>Entity Class</i>	Menggambarkan awal sistem yang menjadi sebuah landasan Menyusun basis data.

### C. *Class Diagram*

*Class diagram* adalah sebuah diagram yang menunjukkan hubungan antara kelas yang didalamnya terdapat fungsi dan atribut dari suatu objek [14]. *Class Diagram* sendiri menggambarkan struktur sistem dari segi pendefinisi kelas-kelas yang dibuat untuk membangun sebuah sistem. Kelas dalam *Class Diagram* didalamnya terdapat atribut dan metode atau proses [13]. Simbol *Class Diagram* dapat dilihat pada tabel 2.5.

**Tabel 2. 5** Simbol *Class Diagram*

No.	Simbol	Nama	Fungsi
1.		Class	Kelas pada struktur sistem

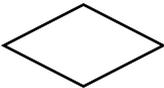
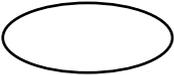
2.		Antarmuka / <i>Interface</i>	Sama dengan konsep interface dalam pemrograman berbasis objek.
3.		Generalisasi	Relasi antar kelas dengan makna generalisasi (umum khusus)
4.		Asosiasi Berarah	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain
5.		Asosiasi	Hubungan antar kelas dengan makna umum
6.		kebergantungan	Hubungan antar kelas dengan makna kebergantungan antar kelas.
7.		Agregasi	Hubungan antar kelas dengan makna semua bagian.

### 2.2.6 ERD Diagram

*Entity Relationship Diagram* adalah suatu diagram yang digunakan untuk perancangan basis data serta dipergunakan untuk menunjukkan hubungan atau relasi antar entitas atau objek yang terlibat dalam basis data beserta atribut-atributnya [15]. Simbol-simbol ERD dapat dilihat pada Tabel 2.6.

**Tabel 2. 6** Simbol Class Diagram [16]

No.	Simbol	Nama	Fungsi
1.		Entitas	Simbol disamping menggambarkan sebuah objek yang memiliki identitas yang unik

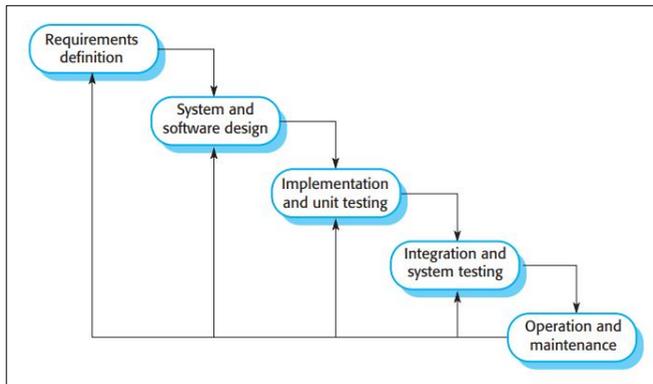
2.		Relasi	Simbol disamping adalah hubungan yang mengikat dua identitas atau lebih. Relasi memiliki beberapa jenis yaitu, <i>one to one, one to many, many to many</i>
3.		Atribut	Karakteristik atau ciri khas yang dimiliki entitas ataupun relasi yang merupakan detail dari masing-masing objek.
4.		Asosiasi	Hubungan antar entitas dengan atributnya maupun dengan entitas dan relasi ataupun atribut.

### 2.2.7 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak adalah disiplin teknologi yang terdiri atas gabungan konsep ilmu komputer, ekonomi, kemampuan komunikasi dan ilmu manajemen dengan pendekatan pemecahan masalah. Tujuan utama dari rekayasa perangkat lunak adalah membantu pengembang dalam mendapatkan sistem perangkat lunak yang berkualitas tinggi. Sedangkan tujuan dasar rekayasa perangkat lunak adalah untuk mengembangkan metode dan prosedur pengembangan perangkat lunak untuk menghasilkan perangkat lunak berkualitas tinggi dengan biaya dan waktu yang sedikit[17]. Aspek rekayasa perangkat lunak yaitu:

#### A. Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem informasi pengelolaan tabungan siswa berbasis *website* dengan notifikasi *whatsapp* ini yaitu metode *waterfall*. Metode *waterfall* menggambarkan metode pengembangan perangkat lunak secara berurutan. Rangkaian aktivitas dalam metode ini disajikan dalam proses yang terpisah dengan 5 tahapan. Tahapan-tahapan pengembangan sistem yaitu[2]:



**Gambar 2. 1** Metode pengembangan sistem *Waterfall* [18]

- a. *Requirements Definition* (Analisis Kebutuhan)  
Fase Analisa Kebutuhan adalah langkah awal yang dilakukan dengan tujuan untuk mengetahui kebutuhan yang berkaitan dengan sistem. Pada tahapan ini juga merupakan tahapan untuk merincikan baik kebutuhan data maupun kebutuhan pengguna yang diterapkan di dalam sistem.
- b. *System and Software Design* (Perancangan)  
Tahapan perancangan desain sistem yang didalamnya termasuk desain arsitektur desain secara keseluruhan. Perancangan desain bertujuan untuk membantu memberikan gambaran secara lengkap mengenai apa yang harus dikerjakan nantinya.
- c. *Implementation and Unit Testing* (Coding)  
Tahap implementasi perangkat lunak yang sesuai dengan rancangan yang telah dibuat fase sebelumnya. Pada tahap ini rancangan sistem yang sebelumnya dibuat akan diubah menjadi bahasa yang mudah dipahami komputer.
- d. *Integration and System Testing* (Pengujian)  
Tahap ini merupakan tahap pengujian pada semua fungsi sistem perangkat lunak yang dihasilkan telah sesuai dengan keinginan pengguna dan perancangan sistem. Pengujian dilakukan juga dilakukan untuk memeriksa fungsi sistem berjalan dengan baik tanpa ditemukan kesalahan.
- e. *Operation and Maintenance* (Perawatan)

Tahap ini merupakan tahap terakhir dalam proses pembuatan sistem, setelah tahap sebelumnya selesai dan kemudian sistem akan digunakan oleh pengguna. Perbaikan dan pengembangan sistem dilakukan untuk pemantauan kinerja kerja dari sistem yang dibuat.

Sistem ini menggunakan metode *Waterfall* untuk metode pengembangan sistem karena metode pengembangan ini dilakukan secara berurutan dan linier. Kualitas sistem yang dihasilkan akan lebih baik karena setiap tahap dilakukan secara bertahap. Karena tahap demi tahap yang dilalui harus menunggu tahap sebelumnya selesai dan saling berkaitan sehingga mengurangi terjadinya kesalahan.

## **B. Metode Black-box Testing**

Pengujian *black-box* merupakan metode pengujian perangkat lunak dimana penguji tidak mengetahui secara internal dari struktur sistem yang akan diuji. Pengujian dilakukan pada setiap bagian untuk mengetahui keberhasilan dari sistem yang dikembangkan. Metode pengujian ini difokuskan pada *input* dan *output* dari sistem apakah sistem tersebut layak untuk digunakan atau belum. Metode ini bertujuan untuk mengetahui apakah Penguji hanya menguji pada bagian *interface* sistem saja dan tidak perlu memiliki pengetahuan Bahasa Pemrograman yang digunakan [19].

### **2.2.8 PHP (*Hypertext Preprocessor*)**

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang digunakan untuk membuat web yang sifatnya *server-side scripting*. PHP memungkinkan untuk membuat sebuah halaman yang bersifat dinamis, dan untuk menghasilkan halaman *web* yang dinamis tersebut PHP dapat memanfaatkan *database* [7]. PHP juga merupakan bahasa pemrograman yang digunakan secara luas untuk pembuatan dan pengembangan dari sebuah situs *web* dan dapat digunakan bersamaan dengan penggunaan CSS dan HTML [20].

### **2.2.9 Framework**

*Framework* adalah sekumpulan intruksi atau perintah yang disatukan dalam kelas dan fungsi-fungsi masing-masing untuk memudahkan pengembang dalam menuliskan *syntax* program yang berulang-ulang[21]. Selain itu, *Framework* merupakan komponen program yang membentuk aturan tertentu dan saling berinteraksi sehingga dalam pembuatan sistem, pengembang harus mengikuti aturan

dari *framework* tersebut. *Framework* digunakan dengan tujuan agar waktu pembuatan sistem menjadi lebih singkat dan pembuatan alur kode program lebih terarah [22].

#### A. *Codeigniter*

*Codeigniter* adalah *framework* PHP yang berupa Kumpulan folder dan file PHP, *Javascript*, CSS, TXT dan file berbasis *web* dengan pengaturan tertentu untuk menggunakannya dan menyediakan *library* serta *helper* yang dapat dimanfaatkan [22]. *Codeigniter* adalah sebuah *framework* bahasa pemrograman PHP yang bersifat *open source* dan menggunakan modeto *move, view, controller* (MVC) yang berguna untuk memudahkan pengembang atau programmer dalam membangun sebuah sistem tanpa harus membuatnya dari awal[21]. *Codeigniter* merupakan *framework* dengan ukuran kecil dan membutuhkan *web server* agar dapat dijalankan.

#### 2.2.10 *Website*

*Website* adalah kumpulan dari beberapa halaman web dimana informasi dalam bentuk teks, suara, gambar, dan lainnya yang dipresentasikan dalam *hypertext* dan dapat diakses oleh *browser*[4]. *Website* adalah halaman-halaman *web* yang digabungkan atau dikumpulkan menjadi satu kesatuan yang menghasilkan sebuah informasi tertentu dan dapat diakses oleh penggunanya.

#### 2.2.11 *Basis Data*

Basis data atau *database* adalah sekumpulan data yang saling berhubungan dan diatur sehingga dapat diakses dan diproses dengan cepat menjadi sebuah informasi. Basis data juga merupakan mekanisme pengolahan data secara terstruktur dan memudahkan program untuk mengambil dan menyimpan data[23].

#### A. *DBMS (Database Management System)*

*DBMS (Database Management System)* atau Sistem manajemen basis data merupakan suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data [24]. *DBMS (Database Management System)* merupakan suatu perangkat lunak pengelola yang digunakan untuk menciptakan, mengakses, mendefinisikan dan merawat basis data. Tujuan *DBMS* adalah menyediakan lingkungan yang mudah dan aman[25].

## B. MySQL

SQL (*Structured Query Language*) merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. MySQL adalah sistem manajemen database yang bersifat *open source* yang dapat digunakan untuk membuat dan mengelola database. MySQL bersifat relational yang artinya data yang dikelola akan diletakan pada tabel yang terpisah[7].

### 2.2.12 WhatsApp

*WhatsApp* adalah aplikasi perpesanan yang memungkinkan pengguna untuk bertukar pesan tanpa harus membayar biaya pengiriman pesan seperti SMS. *WhatsApp* menggunakan paket internet untuk mengirim paket pesan sehingga pesan yang terkirim tidak dikenakan biaya. Paket data diperlukan berguna untuk melakukan obrolan video, kirim foto, dan berbagi file kepada orang lain[26]. Selain itu, *Whatsapp* merupakan aplikasi pesan dengan *basic* mirip dengan *Blackberry Messenger* yang memungkinkan bertukar pesan tanpa biaya SMS, karena *whatsapp* menggunakan paket internet atau *wi-fi* [27].

### 2.2.13 Pengujian Usability

*Usability* adalah atribut yang memiliki kualitas dan menjelaskan kualitas serta menilai bagaimana kemudahan antarmuka pengguna untuk digunakan. *Usability* sendiri memiliki komponen-komponen yang digunakan untuk mengukur pendapat dari responden. Berikut merupakan komponen-komponen dalam pengujian *usability*[28]:

1. *Learnability*  
*Learnability* merupakan pengukuran tingkat kemudahan pengguna dari penggunaan fungsi serta fitur yang ada pada sistem atau aplikasi.
2. *Efficiency*  
*Efficiency* merupakan kecepatan pengguna dalam mengerjakan perintah pada sistem atau aplikasi.
3. *Memorability*  
*Memorability* merupakan kemampuan pengguna dalam menjaga pengetahuannya setelah jangka waktu tertentu.
4. *Errors*  
*Errors* menjelaskan tentang cara memperbaiki *error* serta jumlah *error* yang dilakukan pengguna,
5. *Satisfaction*

*Satisfaction* merupakan tingkat kepuasan pengguna setelah pengguna menggunakan sistem atau aplikasi.

Setiap pertanyaan di atas menggunakan skala likert yang digunakan untuk mengukur pendapat atau sikap dari perorangan atau sekelompok terhadap fenomena sosial tertentu. Setiap pertanyaan memiliki interval jawaban 1-4. Adapaun jawaban dari setiap pertanyaan yang digunakan dapat dinilai dengan skor yang dapat dilihat pada Tabel 2.7 [28].

**Tabel 2. 7** Skala Likert

<b>Jawaban</b>	<b>Skor</b>
Sangat tidak setuju	1
Kurang Setuju	2
Setuju	3
Sangat Setuju	4

Hasil dari kuesioner dari responden, kemudian dilakukan perhitungan dengan menggunakan rumus sebagai berikut [28]:

$$U = \frac{np}{nm} \times 100\%$$

$U$  = Nilai hasil *usability*

$np$  = Nilai perolehan, jumlah seluruh hasil kuesioner/jawaban dari responden

$nm$  = Nilai maksimal, jumlah maksimal yang dapat diperoleh (jumlah responden x nilai maksimal skala likert yang dipakai x jumlah pertanyaan)

Kemudian hasil dari pengujian *usability* dicocokkan dengan kategori interpretasi peringkat sistem. Berikut merupakan kategori interpretasi peringkat sistem yang dapat dilihat pada Tabel 2.8.

**Tabel 2. 8** Kategori Interpretasi Peringkat Sistem

<b>Presentasi hasil uji <i>usability</i></b>	<b>Predikat</b>
81 – 100 %	Sangat Baik
61 – 80 %	Baik
41 – 60 %	Cukup Baik
21 – 40 %	Kurang Baik
0 – 20 %	Tidak Baik