

## BAB II LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Pada bagian ini terdapat beberapa penelitian yang memiliki keterkaitan dengan penelitian yang penulis lakukan, untuk dapat melihat lebih jelas literasi-literasi untuk tinjauan pustaka dapat dilihat pada tabel 2.1.

**Tabel 2. 1 Daftar Literatur**

No	Nama Peneliti	Tahun	Judul Penelitian
1	(Haryani & Ulum, 2021)	2021	Rancang Bangun Sistem Informasi Pembayaran SPP Dengan <i>Virtual Account</i> Menggunakan <i>Framework CodeIgniter</i> .
2	(Herlita, Sari, & Zuraidah, 2021)	2021	Perancangan Sistem Informasi Pembayaran SPP Berbasis <i>Website</i> Pada SMA Fajrul Islam Jakarta.
3	(Melfiani, Andriani, & Hidayatullah, 2020)	2021	Rancang Bangun Sistem Informasi Pembayaran Sumbangan Pembinaan Pendidikan (SPP) Pada SMAK ST. Gregorius Sumbawa Menggunakan Metode <i>Waterfall</i> .
4	(Sujono, Maxrizal, & Sari, 2021)	2021	Analisis dan Perancangan Sistem Informasi Pengelolaan Data Administrasi Sekolah Dasar Islam Terpadu.
5	(Rizaldi, Andrianto, & Ardhiansyah, 2021)	2021	Perancangan Sistem Informasi Administrasi Pembayaran Sekolah Berbasis <i>Web</i> .

#### a. *Literatur* (Haryani & Ulum, 2021)

##### 1. Masalah

Dalam pengolahan data pembayaran pada saat pandemi covid 19 berlangsung masih dilakukan secara *offline*, maka dari perkembangan teknologi yang berkembang peneliti ingin melakukan pengembangan sistem.

2. Metode

Menggunakan metode pengembangan sistem yaitu *Waterfall*, dengan model *UML* yaitu *sequence diagram*, pendekatan/ paradigma menggunakan konsep *OOP*.

3. Hasil atau Kesimpulan

Sistem pembayaran *SPP online* berbasis *web* dapat mempermudah siswa dalam melakukan pembayaran *SPP*, di mana pun, kapan pun, dan *real time* serta dapat mempermudah pihak sekolah dalam pendataan administrasi keuangan sekolah.

**b. *Literatur* (Herlita, Sari, & Zuraidah, 2021)**

1. Masalah

SMA Fajrul Islam masih menggunakan pengolahan pembayaran yang manual, dimana aktifitas yang manual membuat semua proses tidak efektif dan efisien yang membuat laporan yang di hasilkan kurang memadai.

2. Metode

Metode pengembangan yang digunakan menggunakan metode *waterfall*, Menggunakan metode paradigm yang dipakai menggunakan paradigm *OOP*, dengan model perancangan menggunakan *UML* yaitu *usecase diagram* dan *activity diagram*.

3. Hasil atau Kesimpulan

Sistem yang dikembangkan adalah sistem informasi pembayaran *SPP* yang dapat membantu mempermudah siswa dalam melakukan pembayaran secara *online*, mempermudah pencatatan dan pencarian data pembayaran, dan informasi yang dihasilkanpun akan lebih akurat.

**c. *Literatur* (Melfiani, Andriani, & Hidayatullah, 2020)**

1. Masalah

SMAK ST. Gregorius dalam pengolahan pembayaran SPP masih menggunakan semi komputer atau masih belum menggunakan sistem terakreditasi atau masih menggunakan *Microsoft Excel* sehingga proses seperti ini dirasa kurang maksimal dan kurang efektif.

2. Metode

Menggunakan metode pengembangan sistem dengan metode *waterfall*, metode paradigma pemrograman yaitu OOP, serta perancangan model dengan *usecase diagram* dan *class diagram*. Metode pengujian menggunakan metode *black box testing*.

3. Hasil atau Kesimpulan

Sistem yang telah dikembangkan merupakan sistem pengolahan pembayaran yang dapat mengelola pembayaran lebih baik dari sistem sebelumnya.

**d. *Literatur* (Sujono, Maxrizal, & Sari, 2021)**

1. Masalah

Sekolah dasar islam terpadu memiliki masalah yaitu dalam pengolahan administrasi pembayaran masih menggunakan cara manual sehingga menyebabkan kesulitan mencari berkas pendaftaran, dan data pembayaran siswa sehingga pembuatan laporan menjadi terhambat.

2. Metode

Menggunakan metode pengembangan sistem dengan menggunakan metode *prototype*, metode paradigma pemrograman yaitu OOP, dengan menggunakan model perancangan UML yaitu dengan *usecase diagram* dan *activity diagram*.

3. Hasil atau Kesimpulan

Sistem yang dikembangkan adalah sistem informasi pengolahan administrasi pembayaran sekolah yang dapat membantu mencari data dengan cepat, proses pengolahan data lebih



<b>tan</b>						
<b>Model Perancangan</b>	<i>Sequence Diagram</i>	<i>Usecase Diagram, dan Activity Diagram</i>	<i>Usecase Diagram, dan Activity Diagram</i>	<i>Usecase Diagram, dan Activity Diagram</i>	<i>Usecase diagram</i>	<i>Usecase, activity diagram</i>
<b>Pengujian Sistem</b>	-	-	<i>Blackbox</i>	<i>Blackbox</i>	-	<i>ISO 25010</i>
<b>Fitur Sistem</b>	Data siswa, data pembayaran	Data siswa, data pembayaran	Data pengguna, data siswa, data guru, data pembayaran.	Data biaya, data kelas, data siswa, data SPP	Data pengguna, data siswa, data pembayaran	Data user atau pengguna, kategori pembayaran, data transaksi pembayaran, validasi pembayaran, data siswa, data SMK, dan dapat mencetak laporan transaksi pembayaran.
<b>Kelemahan sistem</b>	Tidak <i>management user/</i> pengguna, tidak ada kategori pembayaran, tidak ada verifikasi pembayaran	Tidak <i>management user/</i> pengguna, tidak ada kategori pembayaran, tidak ada verifikasi pembayaran.	Tidak ada kategori pembayaran, dan tidak ada data mengenai SMAK ST. Gregorius.	Tidak ada pengolahan data <i>user</i> , tidak adanya pengolahan data kategori pembayaran SPP	Hanya sebagai pencatatan intern saja tidak dapat diakses oleh siswa, tidak ada pengolahan data kategori pembayaran	Terdapat <i>management user</i> , adanya data user, pengolahan data kategori pembayaran, pengolahan data multi user karena siswa dapat mengakses sistem, data sekolah, terdapat verifikasi pembayaran dari pimpinan atau kepala sekolah

## 2.2 Sistem Informasi

Sistem informasi merupakan suatu kombinasi yang terstruktur antara manusia dan jaringan komunikasi yang saling bekerja sama dan berhubungan untuk mengumpulkan dan menyebarkan informasi kedalam sebuah organisasi ataupun perusahaan (Herlita, Sari, & Zuraidah, 2021).

Sistem informasi adalah suatu gabungan dari hardware dan software komputer dan manusia yang melakukan pengolahan data menggunakan kedua perangkat tersebut. Secara garis besarsistem informasi adalah gabungan dari manusia, *hardware*, *software*, *communications networks*, serta asal data yang dikumpulkan, dan melakukan proses pengaliran dalam organisasi (Rizaldi, Andrianto, & Ardhiansyah, 2021).

Sistem informasi adalah sistem buatan manusia yang terdiri dari komponen baik manual maupun yang berbasis komputer dan berintegrasi untuk mengumpulkan, menyimpan dan mengelola data serta menyediakan informasi untuk pihak-pihak yang bersangkutan sebagai pemakai sistem tersebut (Alotia, Parasala, Sampetoding, Parauba, & Sipota, 2021).

Dari ketiga literasi yang telah dikumpulkan maka dapat disimpulkan sistem informasi adalah suatu gabungan, manusia, *hardware* dan *software* yang saling berintegritas melakukan pengolahan data untuk menghasilkan informasi yang berguna untuk penerimanya atau suatu organisasi.

### **2.3 Sumbangan Pembinaan Pendidikan (SPP)**

Sumbangan pembinaan pendidikan ialah sumbangan pembinaan pendidikan yang dibayarkan oleh siswa di sekolah-sekolah, tujuan sumbangan pembinaan pendidikan agar sekolah dapat membiayai keperluan penyelenggaraan pendidikan sehingga kegiatan belajar mengajar dapat berjalan dengan baik. Sumbangan Pembinaan Pendidikan pada umumnya dibayarkan setiap bulan oleh siswa (Herlita, Sari, & Zuraidah, 2021).

SPP (Sumbangan Pembinaan Pendidikan) merupakan iuran wajib bagi siswa atau siswi yang dipergunakan oleh pihak sekolah untuk memfasilitasi segala kegiatan pembelajaran yang dilakukan siswa/siswi, dengan waktu pembayaran ditentukan sebelumnya dan SPP menjadi salah satu bentuk kewajiban setiap siswa yang masih aktif disekolah tersebut (Nurrahmi, 2020).

SPP (Sumbangan Pembinaan Pendidikan) adalah biaya yang dibebankan kepada mahasiswa yang harus dibayar per semester selama menjadi mahasiswa. Besaran SPP juga beragam baik antara mahasiswa program diploma/sarjana dan program pascasarjana, maupun antara mahasiswa local dan mahasiswa asing (Astriyani, Sari, & Herman, 2020).

Dari ketiga literasi diatas dapat disimpulkan bahwa SPP adalah biaya yang dibebankan kepada siswa/I ataupun mahasiswa yang nantinya akan dipergunakan oleh pihak sekolah untuk memfasilitasi kegiatan sekolah dalam proses pembelajaran agar dapat belajar dengan baik.

## **2.4 Sistem Informasi Pengendalian *Intern***

Sistem informasi pengendalian *intern* merupakan sistem terkomputerisasi yang mengelola data dalam lingkup didalam organisasi itu saja, dengan kata lain sistem ini tidak mengelola data dalam lingkup luar organisasi atau organisasi lainnya. Sistem informasi pengendalian *intern* tersebut dapat dikelompokkan menjadi dua macam yaitu, pengendalian *intern* akuntansi (*intern accounting control*) dan pengendalian *intern* administratif (*intern administrative control*). Pengendalian *intern* akuntansi merupakan bagian dari sistem pengendalian *intern*, meliputi organisasi yang dikoordinasikan terutama untuk menjaga kekayaan organisasi dan mengecek ketelitian dan keandalan dalam segi data keuangan. Pengendalian *intern* administratif meliputi organisasi yang dikoordinasikan terutama untuk mendorong efisiensi serta kebijakan-kebijakan manajemen atau pengolahan data yang mengikuti proses bisnis yang berlaku (Hidayat, 2018).

## **2.5 Alat Pengembang Sistem**

### **2.5.1 *Xampp***

*Xampp* merupakan aplikasi atau perangkat lunak bebas atau yang dapat berfungsi sebagai *server* yang berdiri sendiri (*localhost*) dan dapat mendukung banyak sistem operasi, *xampp* terdiri dari beberapa program yaitu: *Apache HTTP Server*, *MySQL database*, dan bahasa yang ditulis dapat diterjemahkan dengan bahasa pemograman PHP dan *Perl*. *General Public License* (GNU) tersedia di dalam program ini, GNU yaitu *web server* yang dapat mempresentasikan halaman web secara dinamis sehingga mudah untuk digunakan (Ningsih, Sari, & Zuraidah,

2021). *Xampp* adalah perangkat lunak yang bebas, untuk mendukung banyak sistem operasi yaitu, komplikasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *Xampp* merupakan singkatan dari *X* (empat sistem operasi adapun *Apache*, *MySQL*, *PHP* dan *Perl*). Program ini tersedia dalam *GNU (General Public License)* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis (Waidah & Hursali, 2020).

### **2.5.2 *Sublime Text***

*Sublime text* merupakan salah satu *text editor* yang sangat *powerfull* yang dapat meningkatkan produktivitas dan mengembangkan kualitas kode yang tinggi. *Sublime text* adalah aplikasi *editor* untuk kode dan *teks tang* dapat berjalan di berbagai *platform operating system* dengan menggunakan teknologi *Phyton API* (Mubarok, Purnomo, & Noor, 2019). Terciptanya aplikasi ini terinspirasi dari aplikasi *Vim*, aplikasi ini sangat fleksibel dan *powerfull*. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan *sublime-packages* (Waidah & Hursali, 2020).

### **2.5.3 *CodeIgniter***

*Code Igniter* adalah sebuah aplikasi perangkat lunak *open source* yang berupa *framework* *PHP* dengan konsep *MVC (Model View Controller)*. *Code Igniter* memudahkan pengembang perangkat lunak untuk membuat aplikasi *web* dengan cepat, mudah dan terstruktur dibandingkan dengan membuatnya dari awal, kelebihan penggunaan *Code Igniter* dibandingkan *framework* *PHP* lainnya adalah karena *Code Igniter* memiliki konfigurasi yang sangat minim, performa yang cepat, banyaknya komunitas pengembang perangkat lunak yang menggunakan serta



dokumentasi penggunaan program yang sangat lengkap dibanding *framework* PHP lainnya (Setyawan & Asroni, 2018).

#### **2.5.4 Bootstrap**

*Bootstrap* merupakan *framework* untuk membangun desain *web* secara responsif. Artinya, tampilan *web* yang dibuat oleh *bootstrap* akan menyesuaikan ukuran layar dan *browser* yang kita gunakan baik di *desktop*, *tablet* ataupun *mobile device*. Dengan *bootstrap* kita juga bisa membangun *web* dinamis ataupun statis (Waidah & Hursali, 2020). *Bootstrap* merupakan *framework* CSS yang digunakan untuk membantu membuat *website responsive* dengan mudah. *website* yang dibuat dengan *bootstrap* desainnya akan menyesuaikan dengan ukuran layar *device* yang digunakan sehingga tampilan pada *smartphone* berbeda dengan tampilan pada laptop, namun tetap terlihat rapi (Mubarok, Purnomo, & Noor, 2019).

#### **2.5.5 MySQL**

MySQL adalah salah satu jenis *database* yang banyak digunakan untuk membuat aplikasi berbasis *web* yang dinamis. MySQL termasuk jenis RDBMS (*Relational Database Management Sistem*). MySQL ini mendukung Bahasa pemrograman PHP. MySQL juga mempunyai *query* atau bahasa SQL (*Structured Query Language*) yang *simple* dan menggunakan *escape character* yang sama dengan PHP (Welerubun & Bahri, 2019). *MySQL* adalah Aplikasi DBMS yang sudah sangat banyak digunakan oleh para pemrograman aplikasi *web*. Sedangkan menurut Sugiarti Yuni dan Oman Sulaeman dalam jurnal Teknik Informatika Vol.8 No.2 (2015) MySQL adalah RDBMS (*Relational Data Base Management System*) yaitu aplikasi sistem yang berfungsi menjalankan pengelolaan data (Mubarok, Purnomo, & Noor, 2019). MySQL didefinisikan nama *database server*. *Database server* adalah *server* yang berfungsi untuk menangani database. Database adalah suatu pengorganisasian data. Dengan menggunakan *MySQL*, kita bisa

menyimpan data dan kemudian data bisa diakses cara yang mudah dan cepat (Rochman, Sidik, & Nazabah, 2018). MySQL adalah *database server relational* yang gratis dibawah lisensi GNU *General Public Licence*. Dengan sifatnya yang open source, memungkinkan juga user untuk melakukan modifikasi pada souce code-nya untuk memenuhi kebutuhan spesifik mereka sendiri. MySQL merupakan *database server multi-user* dan *multi-threaded* yang tangguh (*rebut*). Dengan memiliki banyak *feature* MySQL bisa bersaing dengan *database* komersial sekalipun. MySQL menjadi *database* pilihan untuk banyak pengguna PHP (Anggraini, Sofiyani, & Khumaini, 2018).

### **2.5.6 PHP**

PHP adalah bahasa pemrograman untuk dijalankan melalui halaman *web*, umumnya digunakan untuk mengolah informasi di *internet*. Sedangkan dalam pengertian lain PHP adalah singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server side* yang bersifat *open source* atau gratis. PHP merupakan *script* yang menyatu dengan HTML dan berada pada server (Welerubun & Bahri, 2019). PHP juga diartikan suatu bahasa pemrograman sisi *server* yang dapat digunakan untuk membuat halaman *web* dinamis. Dalam suatu halaman HTML dapat ditanamkan kode PHP yang akan dieksekusi setiap kali halaman tersebut dikunjungi (Rochman, Sidik, & Nazabah, 2018). Karena kekayaannya akan *fitur* yang mempermudah perancangan dan pemrograman *Web*, PHP memiliki popularitas yang tinggi. PHP adalah kependekan dari *HyperText Preprocessor* (suatu akronim rekursif) yang dibangun oleh Rasmus Lerdorf pada tahun 1994. Dahulu, pada awal pengembangannya PHP disebut sebagai kependekan dari *Personal Home Page*. PHP merupakan produk *Open Source* sehingga dapat mengakses *source code*, menggunakan, dan mengubahnya tanpa harus membayar (Juniko, Hernawati, & Sanjaya, 2019).

### 2.5.7 HTML

HTML (*Hyper Text Mark Up Language*) merupakan bahasa yang digunakan untuk mendeskripsikan struktur sebuah halaman *web*. HTML berfungsi untuk mempublikasi dokumen *online*. *Statement* dasar dari HTML disebut *tags*. Sebuah tag dinyatakan dalam sebuah kurung siku (<>). Tags yang ditujukan untuk sebuah dokumen atau bagian dari suatu dokumen haruslah dibuat berupa pasangan. Terdiri dari *tag* pembuka dan *tag* penutup. Dimana *tag* penutup menggunakan tambahan tanda garis miring (/) di awal nama tag (Welerubun & Bahri, 2019). HTML (*Hyper Text Markup Language*) adalah sekumpulan simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah *file* yang digunakan untuk menampilkan halaman pada *web browser*. *Tag-tag* HTML selalu diawali dengan < dan diakhiri dengan > dimana *x tag HTML* itu seperti *b, i, u*, dll (Mubarok, Purnomo, & Noor, 2019).

### 2.5.8 CSS

*Style sheet* merupakan tempat untuk mengontrol dan mengatur *style* yang ada. *Style sheet* mendeskripsikan bagaimana tampilan dokumen HTML dilayar. Selain itu juga bisa membuat efek-efek spesial dari *web* dengan menggunakan *style sheet*. Akan tetapi pada prakteknya hanya *Cascading Style Sheet (CSS) Technology* yang digunakan pada hampir semua *web browser*. Karena CSS telah distandarkan oleh *World Wide Web Consortium (W3C)* untuk digunakan di *web browser* (Waidah & Hursali, 2020). CSS adalah singkatan dari *Cascading Style Sheet*. CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi dimana HTML ditujukan untuk membuat struktur, atau konten dari halaman *web*. sedangkan CSS digunakan untuk tampilan dari halaman *web* tersebut. Istilahnya *HTML for Content, CSS for Presentation* (Mubarok, Purnomo, & Noor, 2019).

## **2.6 Analisis PIECES**

PIECES adalah metode analisis sebagai dasar untuk memperoleh pokok-pokok permasalahan yang lebih spesifik. Dalam menganalisis sebuah sistem, biasanya akan dilakukan terhadap beberapa aspek antara lain adalah kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi dan pelayanan pelanggan. Analisis ini disebut dengan *PIECES Analysis (Performance, Information, Economy, Control, Eficiency dan Service*. Analisis PIECES ini sangat penting untuk dilakukan sebelum mengembangkan sebuah sistem informasi karena dalam analisis ini biasanya akan ditemukan beberapa masalah utama maupun masalah yang bersifat gejala dari masalah utama. Metode ini menggunakan enam variabel evaluasi sebagai berikut (Pamungkas, Rachmatullah, & Gumelar, 2018).

### **1. Analisis Kinerja Sistem (*Performance*)**

Kinerja merupakan variabel pertama dalam metode analisis piecess. Dimana memiliki peran penting untuk menilai apakah proses atau prosedur yang ada masih mungkin ditingkatkan kinerjanya, dan melihat sejauh mana dan seberapa handalkah suatu sistem informasi dalam berproses untuk menghasilkan tujuan yang diinginkan. Dalam hal kinerja diukur dari:

- a.** Produksi yaitu jumlah pekerjaan/output yang dapat dilakukan atau dihasilkan pada saat tertentu.
- b.** Waktu respons, yaitu waktu yang dibutuhkan untuk menyelesaikan serangkaian kegiatan untuk menghasilkan output atau deliverables tertentu.

### **2. Analisis Informasi (*Information*)**

Menilai merupakan apakah prosedur yang ada saat ini masih dapat diperbaiki sehingga kualitas informasi yang dihasilkan menjadi semakin baik. Informasi yang disajikan

haruslah benar-benar mempunyai nilai yang berguna hal ini dapat diukur dengan: 1) Keluaran (output) suatu sistem dalam memproduksi keluaran. 2) Masukan (Input) dalam memasukkan suatu data sehingga kemudian diolah untuk menjadi informasi yang berguna..

### **3. Analisis Ekonomi (*Economy*)**

Menilai apakah prosedur yang ada saat ini masih dapat ditingkatkan manfaatnya (nilai gunanya) atau diturunkan biaya penyelenggaraannya.

### **4. Analisis Pengendalian (*Control*)**

Menilai apakah prosedur yang ada saat ini masih dapat ditingkatkan sehingga kualitas pengendalian menjadi semakin baik, dan kemampuannya untuk mendeteksi kesalahan atau kecurangan menjadi semakin baik pula.

### **5. Analisis Efisiensi (*Efficiency*)**

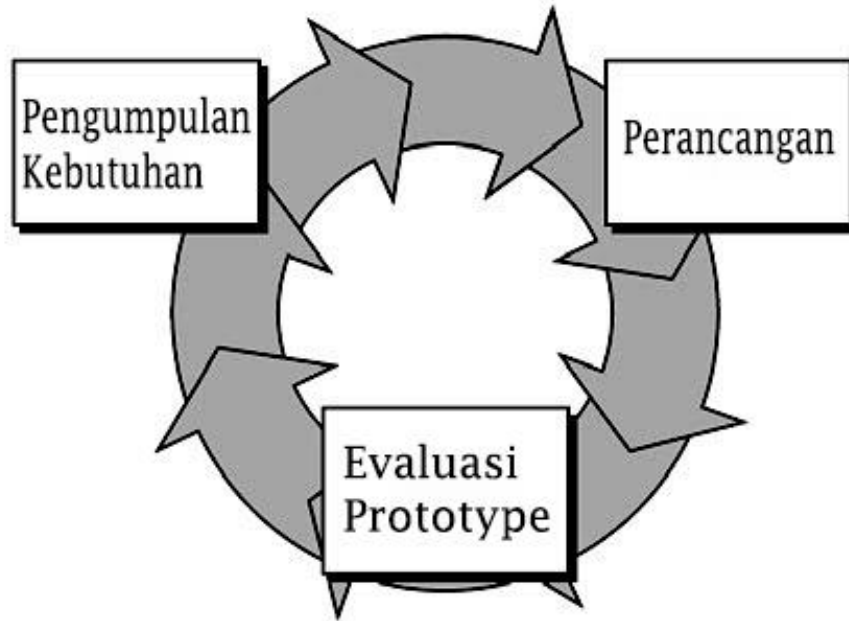
Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki, sehingga tercapai peningkatan efisiensi operasi, dan harus lebih unggul dari pada sistem manual.

### **6. Analisis Pelayanan (*Service*)**

Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki kemampuannya untuk mencapai peningkatan kualitas layanan. Kualitas layanan yang sangat user friendly untuk end-user (pengguna) sehingga pengguna mendapatkan kualitas layanan yang baik.

## **2.7 Metode Pengembangan Sistem Model *Prototype***

(Rosa A. S & M. Shalahuddin, 2016) menyatakan bahwa model *prototype* dapat digunakan untuk menyambungkan ketidak pahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak. Berikut ini adalah gambar dari model *prototype* pada gambar berikut.



**Gambar 2. 1 Ilustrasi Model *Prototype***  
*Sumber : (Rosa A. S & M. Shalahuddin, 2016)*

*Prototype* juga dapat didefinisikan sebagai proses pengembangan suatu *prototype* secara cepat untuk digunakan terlebih dahulu dan ditingkatkan terus menerus sampai didapatkan sistem yang utuh. Proses yang digunakan untuk membantu pengembang perangkat lunak dalam membentuk *prototype* dari perangkat lunak yang harus dibuat. Proses pada model prototyping dapat dijelaskan sebagai berikut:

1. Pengumpulan kebutuhan, *developer* dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya.
2. Perancangan, dilakukan cepat dan rancangan mewakili semua aspek perangkat lunak yang diketahui, dan rancangan ini menjadi dasar pembuatan *prototype*.
3. Evaluasi *prototype*, klien mengevaluasi *prototype* yang dibuat dan digunakan untuk memperjelas kebutuhan perangkat lunak.

Perulangan ketiga proses ini terus berlangsung hingga semua kebutuhan terpenuhi. *Prototype-prototype* dibuat untuk memuaskan kebutuhan klien dan untuk membangun perangkat lunak lebih cepat, namun tidak semua *prototype* bisa dimanfaatkan. Demi kebutuhan klien lebih baik *prototype* yang dibuat diusahakan dapat dimanfaatkan.



## 2.2 Unified Modeling Language (UML)






*Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi *object* (Rosa dan M shalahuddin, 2018).

### 2.2.1 Use case Diagram

*Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *usecase* digunakan untuk mengetahui fungsi apa saja dan siapa saja yang berhak menggunakan fungsi-fungsi sistem (Rosa dan M shalahuddin, 2018). Perilaku sistem memperlakukan bagaimana sistem beraksi dan bereaksi. Perilaku ini merupakan aktifitas sistem yang bisa dilihat dari luar dan bisa diuji. Perilaku sistem ini di *capture* di dalam *usecase*.

**Tabel 2. 3 Usecase Diagram**

No.	Simbol	Keterangan
1.	 Aktor/Actor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem Informasi yang akan dibuat itu sendiri.
2.	<i>Use case</i>  Nama	Fungsionalitas yang disediakan sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .

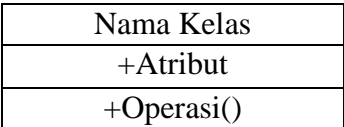

3.	<i>Association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<i>Extend/Ekstensi</i> <i>&lt;&lt;extend&gt;&gt;</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5.	<i>Generalization/</i> <i>Generalisasi</i> 	Hubungan generalisasi dan spesialisasi (umum khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih dari lainnya.
6.	<i>Menggunakan/</i> <i>Include/Uses</i> <i>&lt;&lt;include&gt;&gt;</i>  <i>&lt;&lt;uses&gt;&gt;</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat.

(Sumber : Rosa A.S-M. Shalahuddin, 2018)

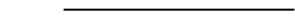

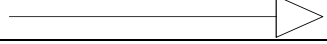


### 2.2.2 Class Diagram

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron (Rosa dan M shalahuddin, 2018).

**Tabel 2. 4 Simbol Class Diagram**

No	Simbol	Keterangan
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka/ <i>Interfaces</i>  Nama_ <i>interfaces</i>	Sama dengan konsep <i>interfaces</i> dalam pemrogram berorientasi objek





3	Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antar kelas dengan makna generalisasi/spesialisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Kebergantungan antar kelas.
7	Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna-bagian ( <i>wholepart</i> ).

(Sumber : Rosa -M. Shalahuddin, 2018)



### 2.2.3 Activity Diagram


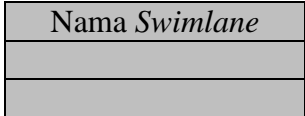
Diagram aktifitas atau *activity Diagram* menggambarkan workflow (aliran kerja) atau aktivitas sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan M shalahuddin, 2018).

**Tabel 2. 5 Activity Diagram**

No	Simbol	Keterangan
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

**Tabel 2.5 Activity Diagram (Lanjutan)**

No	Simbol	Keterangan
3	Percabangan/ decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

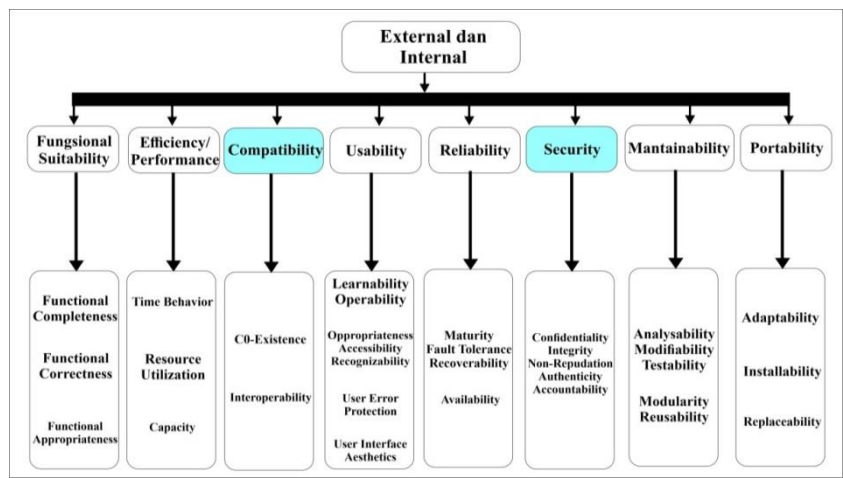
5	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6	<i>Swimlane</i> 	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

(Sumber : Rosa -M. Shalahuddin, 2018)

## 2.8 Pengujian ISO 25010

Model ISO 25010 didefinisikan salah satu model pengujian dan evaluasi kualitas perangkat lunak dan model ISO 25010 ini terdapat delapan domain (kriteria) dimana terdiri dari sejumlah *sub* domain tambahan dan beberapa *sub* domain yang dipindahkan ke domain lainnya. Berikut struktur ISO 25010 dapat dilihat pada gambar 2.3 dibawah ini.

### Karakteristik Kualitas Perangkat Lunak Model ISO 25010



**Gambar 2. 2 Karakteristik ISO 25010**

Sumber: (Mubarok, Suaidah, & Priandika, 2020)

Dari kedelapan domain atau kriteria diatas penulis menggunakan 3 domain antara lain *functional suitability*, *performance efficiency*, dan *usability*. Penjelasan domain yang digunakan dalam penelitian ini pada Tabel 2.5.

**Tabel 2. 6 Domain ISO 25010**

No	Domain	Sub-Domain	Indikator Testing
1	<i>Functional Suitability</i>	<i>Functional completeness</i>	sejauh mana fungsi yang disediakan mencakup semua tugas dan tujuan pengguna secara spesifik.
		<i>Functional correctness</i>	sejauh mana produk atau sistem menyediakan hasil yang benar sesuai kebutuhan.
		<i>Functional appropriateness</i>	sejauh mana fungsi yang disediakan mampu memfasilitasi penyelesaian tugas dan tujuan tertentu.
2	<i>Performance Efficiency</i>	<i>Time behavior</i>	sejauh mana respon dan pengolahan waktu produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
		<i>Resource utilization</i>	sejauh mana jumlah dan jenis sumber daya yang digunakan oleh produk atau sistem dapat memenuhi persyaratan ketika menjalankan fungsi.
		<i>Capacity</i>	sejauh mana batas maksimum parameter produk dapat memenuhi persyaratan.
3	<i>Usability</i>	<i>Appropriateness recognizability</i>	sejauh mana pengguna dapat mengetahui apakah sistem atau produk sesuai kebutuhan mereka.
		<i>Learnability</i>	sejauh mana produk dapat digunakan oleh pengguna untuk mencapai tujuan tertentu yang belajar menggunakan produk dengan efisien, efektif.
		<i>Operability</i>	sejauh mana produk atau sistem mudah dioperasikan dan dikontrol
		<i>User error protection</i>	sejauh mana produk atau sistem melindungi pengguna terhadap membuat kesalahan.
		<i>User interface aesthetics</i>	sejauh mana antarmuka pengguna sistem memungkinkan interaksi yang menyenangkan dan memuaskan <i>user</i> .
		<i>Accessibility</i>	sejauh mana produk dapat digunakan oleh semua kalangan untuk mencapai tujuan sesuai konteks penggunaan.

Sumber: (Mubarok, Suaidah, & Priandika, 2020)

### 2.8.1 Skala Likert

Menurut (Sugiyono, 2018) *skala likert testing* adalah Skala yang digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena

sosial. Dalam penelitian fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian.

Dengan *skala likert* maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator variabel tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau pertanyaan. Jawaban setiap item instrumen yang menggunakan skala *Likert* mempunyai gradasi dari sangat positif sampai sangat negatif, yang dapat berupa kata-kata antara lain:

- a. Sangat Setuju
  - b. Setuju
  - c. Ragu-ragu
  - d. Tidak setuju
  - e. Sangat tidak setuju
- a. Selalu
  - b. Sering
  - c. Kadang-kadang
  - d. Tidak pernah

- a. Sangat Positif
  - b. Positif
  - c. Negatif
  - d. Sangat Negatif
- a. Sangat baik
  - b. Baik
  - c. Tidak baik
  - d. Sangat tidak baik

Untuk keperluan analisis kuantitatif, maka jawaban itu dapat diberi skor, misalnya.

**Tabel 2. 7 Score Skala Likert**

<b>Keterangan</b>	<b>Score</b>
Setuju/Selalu/Sangat Positif Diberi Skor	5
Setuju/Sering/Positif Diberi Skor	4
Ragu-Ragu/Kadang-Kadang/Netral Diberi Skor	3

Tidak Setuju/Hampir Tidak Pernah/Negatif Diberi Skor	2
Sangat Tidak Setuju/Tidak Pernah Diberi Skor	1

Instrumen kualifikasi kelayakan sistem yang menggunakan skala likert dapat dibuat dalam bentuk tabel dibawah ini.

**Tabel 2. 8 Skala *Likert***

Tingkat Pencapaian	Kualifikasi	Keterangan
90% - 100%	Sangat Baik	Tidak Perlu Revisi
75% - 89%	Baik	Sedikit Revisi
65% - 74%	Cukup	Di Revisi Secukupnya
55% - 64%	Kurang	Banyak hal yang direvisi
0% - 54%	Sangat Kurang	Diulang Membuat Produk