

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil yaitu dapat dilihat pada tabel 2.1

Tabel 2. 1 Tinjauan Pustaka

No.	Penulis dan Tahun	Judul
1.	(Naurah Gardenifa Sa1 sabila, Suryatiningsih, Dedy Rahman Wijaya ; 2021)	Aplikasi Warga Berseri Modul Pengelolaan Dompot Iuran Warga Studi Kasus: Perumahan Permata Buah Batu
2.	(Ardhi Abdurrahman Ghozi, Muhammad Taufik Hidayat, dan Khoiru Rohman ; 2020)	Sistem Informasi Antar Warga “Si-Anwar” Sebagai Solusi Bermasyarakat Di Perumahan Tigaraksa Berbasis Web
3.	(Wandani Siregar, Irvan, Eka Rahayu ; 2020)	Sistem Informasi Pembayaran Iuran Keamanan Dan Kebersihan Pada Perumahan Berbasis <i>Website</i> Menggunakan Metode <i>Design Thinking</i>
4.	(Eko Saputra ; 2020)	Sistem Pembayaran Dan Manajemen Tagihan Iuran Perumahan Berbasis Web
5.	(Kristina ; 2019)	Pemodelan Sistem Informasi Pendataan Warga Dan Biaya Operasional Lingkungan Pada Komplek Perumahan Harmoni Park Berbasis <i>Zachman Framework</i>

a. Literatur 1

Perumahan Permata Buah Batu membebankan iuran bulanan yang merupakan kesepakatan dengan semua warga untuk berkontribusi terhadap pengelolaan sampah, pembayaran keamanan, dan keperluan lainnya. Namun, saat ini rumah Permata Buah Batu masih menggunakan sistem *Door to Door*. *Door-to-Door* sendiri merupakan cara petugas mengumpulkan sumbangan dengan mengunjungi rumah masing-masing warga. Biasanya petugas akan memberikan kartu iuran kepada warga sebagai bukti pembayaran, dan jika iuran sudah dibayar, kartu tersebut akan dicap berdasarkan bulan pembayaran. Namun, karena kartu hanya terbuat dari kertas, kartu tersebut rentan hilang atau rusak. Setelah iuran dilunasi, biasanya petugas akan menyerahkan iuran tersebut kepada pengelola perumahan untuk data lebih lanjut. Setiap rumah akan menerima selebaran setiap bulan dengan informasi tentang penggunaan dan pendapatan untuk biaya bulanan.

Oleh karena itu, pada penelitian ini dibangun sebuah aplikasi pembayaran dengan menggunakan bahasa pemrograman *PHP*, *Framework Codeigniter* dan database *MySQL*. Pembangunan aplikasi ini menggunakan metode *Waterfall*. Dan sesuai dengan aplikasi yang telah dibuat, yaitu memudahkan pihak pengelola untuk mengatur pemasukan dan pengeluaran iuran warga, dan juga memudahkan warga untuk memahami tujuan dari dana iuran warga (Gardenifa Sa and Rahman Wijaya 2021).

b. Literatur 2

Perumahan Tigaraksa Tangerang Banten memiliki permasalahan yaitu kurangnya transparansi dalam laporan keuangan dan warga sering mengeluh karena kurangnya mendapatkan informasi juga pelaporan yang lebih mudah tersampaikan kepada Ketua RT, mengingat warga dan ketua RT yang memiliki kesibukan masing-masing membuat mereka kesulitan untuk bertemu. Dengan permasalahan diatas, maka dibutuhkan sebuah aplikasi yang dapat mempermudah warga untuk mendapatkan informasi secara digital dan juga secara *real time*. Pada penelitian ini, peneliti menggunakan metode *waterfall* sebagai metode analisa dan perancangan, serta metode UML sebagai analisis kebutuhan sistem, desain, dan perancangan. Untuk pengkodean, peneliti menggunakan bahasa pemrograman *PHP* dan database *MySQL*.

Hasil dari penelitian ini yaitu menghasilkan aplikasi yang menyediakan informasi yang berkaitan dengan aktifitas warga dilingkungan secara online agar memudahkan warga perumahan mendapatkan informasi(Ghozi, Hidayat, and Rohman 2020).

c. Literatur 3

Perumahan Menteng Indah di Kota Medan memiliki lebih dari 500 unit rumah atau sampai dengan 500 kepala keluarga (KK), dan setiap penghuni harus membayar iuran keamanan dan kebersihan. Namun Proses pembayaran iuran keamanan dan kebersihan Perumahan Menteng Indah masih belum tersistem, melihat proses yang berjalan saat ini yaitu dalam proses pembayaran iuran keamanan dan kebersihan, satpam akan mendatangi rumah warga atau warga yang datang ke pos satpam, satpam akan melakukan pencatatan sebanyak dua kali yaitu bukti pembayaran iuran keamanan dan pembayaran kebersihan, kemudian diserahkan kepada bendahara untuk dilakukan rekap dan disimpan di lemari arsip. Akibatnya, proses pembayaran iuran menjadi tidak efisien atau memakan waktu lama.

Dalam penelitian ini metode pengembangan dalam membangun Sistem Informasi Pembayaran Iuran Keamanan dan Kebersihan di Perumahan Menteng Indah Berbasis Website menggunakan teori metode Design Thinking. Design thinking adalah suatu metode yang digunakan untuk memecahkan permasalahan yang dihadapi dengan cara mengumpulkan banyak ide – ide dalam sesi brainstorming menggunakan pendekatan user – centered. Dari sistem yang dibangun menghasilkan Sistem Pembayaran Iuran di Perumahan Menteng Indah Berbasis Website, maka akan mempermudah warga dan security dalam melakukan proses pembayaran iuran(Siregar, Irvan, and Rahayu 2020).

d. Literatur 4

Di Desa Bumiharjo, Kecamatan Moilong, Kabupaten Banggai terdapat pengelolaan sampah dan keamanan yg dimana proses pembayaran dan pengelolaan tagihan iuran di masih menggunakan cara manual. Yaitu petugas lapangan melakukan penagihan iuran dari rumah ke rumah warga, pengelolaan tagihan iuran masih menggunakan tulisan tangan dan software Microsoft Word yang tidak terlalu efektif dalam praktiknya.

Karena sulit mendapatkan data real time terkait tagihan karena harus mencari satu per satu data manualnya.

Berdasarkan pernyataan di atas, maka diperlukan suatu sistem aplikasi manajemen pembayaran dan penagihan berbasis web yang dirancang untuk mempermudah dan memperlancar pekerjaan tim yang terlibat yaitu petugas lapangan, pengelola wilayah, pembina dan pemilik rumah itu sendiri. Sistem ini dibuat dengan menggunakan metode waterfall.

Tahapan pembangunan website dimulai dengan analisis, desain sistem, desain database, desain antarmuka dan pengujian. Pengujian dilakukan bersamaan dengan pengujian unit dan pengujian pengguna untuk melihat apakah situs web berfungsi dengan baik dan memenuhi kebutuhan pengguna.

Hasil penelitian ini dikumpulkan melalui kuesioner dan wawancara langsung dengan pengguna. Setelah dilakukan pengumpulan data untuk mendapatkan hasil tugas akhir, maka dapat disimpulkan bahwa sistem telah dapat membantu petugas lapangan, pengelola wilayah, pembina dan pemilik rumah dalam operasional yang terkait. Setiap pengujian dapat dijalankan dan dievaluasi sesuai kebutuhan, termasuk pengujian unit dan pengujian pengguna(Saputra 2020).

e. Literatur 5

Komplek Harmoni Park merupakan komplek yang saat ini pendataan warga komplek didata oleh masing- masing RT, baik penghuni yang merupakan pemilik rumah maupun penyewa. Setiap orang baru yang membeli atau menyewa rumah di komplek tersebut harus melapor kepada ketua RT. Jika dalam waktu 24 jam tidak ada laporan, maka ketua RT akan datang untuk membuat pendataan. Selama ini warga melakukan pendataan secara manual melalui RT, dimana warga harus mengisi formulir, dan bagi yang menyewa diharuskan melampirkan fotokopi KTP dan KK atau tanda pengenal lainnya. Formulir yang sudah diisi hanya diarsipkan oleh Ketua RT.

Di lingkungan komplek ini, warga harus membayar iuran yang dikelola oleh RT masing-masing. Penghuni rumah itu sendiri atau bagi mereka yang menyewa. Untuk sistem pembayaran, warga bisa mengajukan langsung ke Bendahara RT. Bagi warga

yang menunggak atau terlambat membayar, RT akan mendatangi di rumah warga melalui security untuk menagihnya.

Dalam penelitian ini digunakan metode observasi dan wawancara untuk pengumpulan data guna mendukung perancangan aplikasi pendataan penduduk dan biaya operasional kompleks perumahan Harmoni Park. Alat yang digunakan dalam pemodelan sistem informasi adalah framework Zachman. Berdasarkan sistem yang telah dibuat maka menghasilkan sebuah sistem informasi pengelolaan data warga dan biaya operasional yang lebih cepat, efektif, efisien dan transparan pada Komplek Perumahan Harmoni Park(Kristina 2019).

Berdasarkan tinjauan literatur yang dilakukan maka peneliti mendapatkan perbedaan antara penelitian-penelitian sebelumnya dengan penelitian yang dilakukan peneliti saat ini, yaitu:

- 1) Fomulir tagihan iuran yang ditujukan untuk warga perumahan akan dibuat secara online yaitu melalui pesan *whatsapp*
- 2) Disediaknya periode perhari,bulan, dan tahun pada laporan terkait
- 3) Pada penelitian ini menggunakan metode pengembangan sistem *Extreme Programming*, pengujian sistem menggunakan *ISO 25010*, serta *PIECES* sebagai analisis perbandingan antara sistem lama dengan sistem baru yang dibuat
- 4) Perbedaan pada tempat penelitian

2.2 Pengertian Sistem Pembayaran

Dalam Undang-undang No. 23 tahun 1999 BI pasal 1 poin ke 6 dijelaskan yaitu “Sistem pembayaran adalah sistem yang mencakup sekumpulan aturan, institusi dan mekanisme pelaksanaan mentransfer dana untuk memenuhi kewajiban yang timbul dari suatu kegiatan ekonomi. Sistem pembayaran harus dapat menjamin transfer dana publik yang efisien dan aman .Dengan cara ini menjamin kenyamanan dalam melakukan setiap transaksi dalam kegiatan ekonomi. Jadi Bank Indonesia yang sebagai Bank sentral pada dasarnya berkewajiban untuk mengawasi. Mengawasi sistem pembayaran yang berlangsung dalam kegiatan mencapai ekonomi masyarakat dengan menerapkan sistem yang diinginkan peserta kegiatan ekonomi”.

2.3 Pengertian Iuran

“Iuran adalah pungutan yang dilakukan sehubungan dengan sesuatu layanan atau fasilitas yang diberikan pemerintah secara langsung dan nyata kepada kelompok atau golongan pembayar. Contoh: biaya televisi, biaya keamanan, biaya sampah, biaya air, dan lain sebagainya. Iuran tidak termasuk pajak”(Devano, Sony 2006).

2.4 Website

“Website atau Situs merupakan kumpulan halamayang menampilkan informasi atau data dalam bentuk gambar, teks, animasi , suara, video yang bersifat statis atau dinamis yang membentuk suatu rangkaian yang saling terikat dan dihubungkan dengan *hyperlink*”(Wulandari 2018).

2.5 Framework CodeIgniter

“*Framework* adalah kumpulan perintah yang membentuk aturan-aturan dan saling berinteraksi satu sama lain sehingga dalam pembuatan *website* kita mengikuti aturan-aturan dari *framework*”(Wardana n.d.).

2.6 MySQL

“MySQL Merupakan sebuah database server yang free, artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. MySQL pertama kali dirintis oleh seorang programmer database bernama Michael Widenius . Selain database server, MySQL juga merupakan program yang dapat mengakses suatu database MySQL yang berposisi sebagai Server, yang berarti program kita berposisi sebagai Client. Jadi MySQL adalah sebuah database yang dapat digunakan sebagai Client maupun server”(Saputro 2020).

2.7 PHP

Menurut Supono & Putratama (2018: 1) mengemukakan bahwa “*PHP (PHP: hypertext preprocessor)* adalah suatu bahasa pemrograman yang digunakan untuk

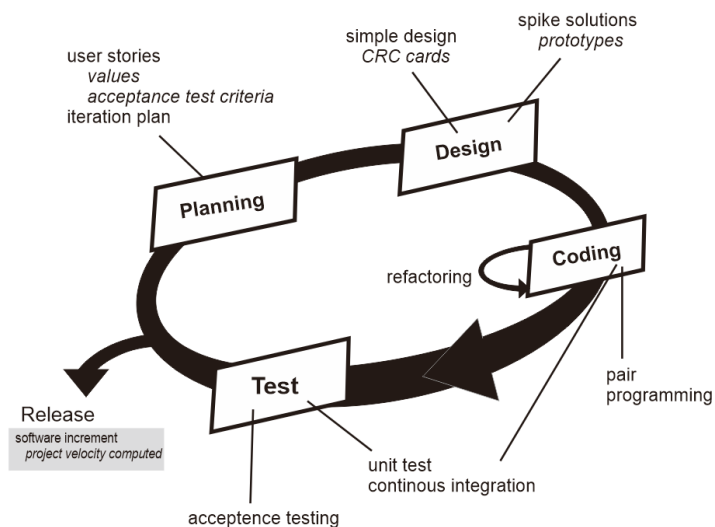
menterjemahkan basis kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server-side yang ditambahkan ke *HTML*”.

“*PHP* merupakan bahasa pemrograman untuk pembuatan website dinamis, yang mampu berinteraksi dengan pengunjung atau penggunanya”(Wardana, 2016:1).

Berdasarkan pengertian diatas, maka dapat disimpulkan bahwa hypertext preprocessor (*PHP*) merupakan bahasa pemrograman yang mengolah database, content website sehingga website yang dibuat merupakan web dinamis, dan *PHP* merupakan bahasa pemrograman yang dikombinasikan dengan *HTML*.

2.8 Metode Pengembangan Sistem *Extreme Programming*

Pada penelitian ini, peneliti menggunakan metode *Extreme Programming* sebagai metode pengembangan sistem. Menurut Prabowo dalam (Supriyatna 2018) *Extreme Programming* (XP) merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan requirement yang tidak jelas maupun terjadi perubahan-perubahan requirement yang sangat cepat. Tahapan metode *Extreme Programming* dapat dilihat pada gambar 2.1



Gambar 2.1 *Extreme Programming*

Sumber: (Supriyatna 2018)

Empat tahapan dalam *Extreme Programming* yaitu sebagai berikut:

1. *Planning*

Pada tahap *planning* ini berfokus pada gambaran umum mengenai fitur dan fungsi perangkat lunak yang akan dibangun.

2. *Design*

Pada tahap *design* ini berfokus pada perancangan yang dimulai dengan membangun model dan *user interface* yang akan digunakan saat pembangunan sistem

3. *Coding*

Pada tahapan *coding* pada *XP* dimulai dengan merangkai satu set unit test. Lalu setelah itu pengembang kemudian fokus pada implementasi.

4. *Testing*

Pada tahapan *testing* akan dilakukan langsung oleh *klien* dan pengguna untuk menguji fungsi-fungsi dan fitur sistem secara keseluruhan

2.9 Unified Modelling Language (UML)

“*UML* adalah pengembangan teknologi pemrograman berorientasi objek, menciptakan bahasa pemodelan standar untuk pengembangan perangkat lunak untuk analisis dan desain, dan menggambarkan arsitektur pemrograman berorientasi objek”(Rosa 2018).

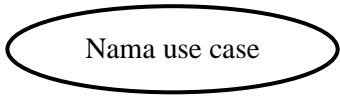
2.9.1 Usecase Diagram

“*Usecase* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”(Rosa 2018).


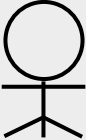


Simbol - simbol use case atau diagram use case yang ditampilkan pada tabel berikut.

Tabel 2. 2 Simbol-simbol *Usecase Diagram*

Simbol	Keterangan
--------	------------

<p>Use Case</p> 	<p>Menggunakan kata kerja di awal use case untuk memberi nama use case dengan menggunakan frasa nama kasus.</p>
---	---

Tabel 2. 2 Simbol-simbol Usecase Diagram (Lanjutan)


Simbol	Keterangan
<p>Generalisasi</p> 	<p>Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case</p>
<p>Aktor</p>  <p>Nama Aktor</p>	<p>Aktor adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat. Biasanya aktor nama diberi nama dengan menggunakan kata benda di awal frasa nama aktor.</p>
<p>Asosiasi</p> 	<p>Asosiasi adalah penghubung antara aktor dengan use case.</p>
<p>Extend / Ekstensi</p> <p><<extend>></p> 	<p>Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi</p>

2.9.2 Activity Diagram






“Activity Diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol- simbol yang digunakan dalam activity diagram”(Rosa 2018).

Berikut ini adalah simbol-simbol yang ada di *Activity Diagram* :

Tabel 2. 3 Simbol-simbol Activity Diagram

Simbol	Keterangan
Status Awal 	Status awal aktivitas sistem, diagram aktivitas memiliki status awal.

Tabel 2. 3 Simbol-simbol Activity Diagram (Lanjutan)

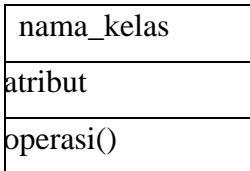
Simbol	Keterangan
Aktivitas 	Aktivitas dilakukan oleh sistem, yang biasanya dimulai dengan kata kerja.
Percabangan 	Percabangan yang dimana ada pilihan aktivitas yang lebih dari satu.
Penggabungan 	Penggabungan yang dimana lebih dari satu aktivitas yang digabungkan menjadi satu.
Status Akhir 	Status aakhir aktivitas sistem, diagram aktivitas memiliki status akhir.
Arah aliran 	Aliran arah dari dan ke masing-masing aktivitas

2.9.3 Class Diagram




“Class Diagram Menggambarkan struktur sistem dalam hal mendefinisikan kelas-kelas yang akan dibuat untuk membuat sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut adalah variabel yang dimiliki oleh sebuah kelas. Meskipun operasi atau metode adalah fungsi yang dimiliki oleh kelas” (Rosa 2018).

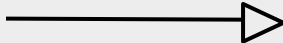
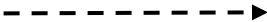

Berikut ini adalah simbol-simbol yang ada di *Class Diagram* :

Tabel 2. 4 Simbol-simbol *Class Diagram*

Simbol	Keterangan
Kelas 	Kelas pada struktur sistem.

Tabel 2. 4 Simbol-simbol *Class Diagram* (Lanjutan)

Simbol	Keterangan
Interface/antarmuka  nama_interface	Konsep yang sama dengan antarmuka interface pemrograman berorientasi objek.
Asosiasi 	Hubungan antar kelas memiliki arti umum, dan asosiasi biasanya disertai dengan multiplisitas.
Asosiasi berarah 	Hubungan antar kelas dengan makna kelas Ini adalah jenis lain atau digunakan oleh kelas lain.


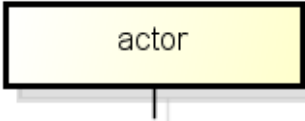
Generalisasi		Hubungan antara kelas dan makna umum-khusus.
Dependency		Hubungan antara kelas dengan makna kebergantungan antarkelas.
Agregasi		Hubungan antara kelas dengan makna semua bagian kelas.





2.9.4 Sequence Diagram

“Sequence Diagram menggambarkan kelakuan objek pada *Usecase* dengan mendeskripsikan waktu hidup objek atau *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sequence maka harus diketahui objek-objek yang terlibat dalam sebuah usecase beserta metode-metode yang dimiliki kelas”(Rosa 2018).

Simbol – simbol yang ada pada sequence diagram dapat dilihat pada tabel 2.5 berikut ini :

Tabel 2. 5 Sequence Diagram

Simbol	Keterangan
Aktor  atau 	Orang,proses,atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang;biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.

<p>Garis Hidup</p> 	<p>Mengartikan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Objek entitas, antarmuka yang saling berinteraksi</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>
<p>Pesan Tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirim data /masukan/informasi ke objek lainnya , arah panah mengarah pada objek yang dikirim</p>

2.10 Pengujian Sistem *ISO 25010*

Pada penelitian ini, peneliti menggunakan metode pengujian sistem menggunakan *ISO/IEC 2510*. *ISO / IEC 25010* adalah bagian dari Sistem dan Rekayasa Perangkat Lunak-Sistem dan Persyaratan dan Evaluasi Kualitas Perangkat Lunak (SQuaRE) -Model Kualitas Sistem dan Perangkat Lunak. *ISO / IEC 25010* merupakan perpanjangan dari *ISO / IEC 9126*, dan secara resmi membatalkan dan menggantikan *ISO / IEC 9126: 2001*. *ISO / IEC 25010: 2011* edisi pertama telah mengalami revisi teknis menjadi standar internasional terkini dan terkait dengan pengujian sistem informasi yang akan dikembangkan. *ISO / IEC 25010* telah menjadi tolok ukur yang digunakan untuk menganalisis kualitas perangkat lunak yang digunakan oleh suatu perusahaan, institusi atau organisasi. Dengan bantuan perangkat lunak *ISO / IEC 25010*, 8 karakteristik dapat disusun sesuai dengan dimensi kualitas produk (termasuk penerapan fungsional, efisiensi kinerja, kompatibilitas, kegunaan, keandalan,

keamanan, pemeliharaan, dan portabilitas) untuk secara khusus melakukan evaluasi sistem kualitas(International Organization for Standardization and International Electrotechnical Commission, 2011).



Gambar 2. 2 Model kualitas produk ISO/IEC 25010

Berikut penjelasan dari indikator dan sub-bab mengenai model ISO 25010 :

- 1) **Fungsionalitas (*Functional Suitability*)**, seberapa jauh mana sistem dapat menyediakan fungsi yang memenuhi persyaratan dan standar yang ada (*ISO / IEC, 2011*). Ciri-cirinya yaitu :
 - a) Kelengkapan Fungsional (*Functional Completeness*), yaitu seberapa jauh mana rangkaian fungsi ini mencakup semua tujuan dan tugas pengguna.
 - b) Ketepatan Fungsional (*Functional Correctness*), yaitu seberapa jauh mana produk dapat memberikan atau membuktikan tingkat hasil yang wajar sampai batas tertentu sesuai dengan kebutuhan.
 - c) Kesesuaian Fungsional (*Functional Appropriateness*), yaitu seberapa jauh mana sejauh mana fungsi sistem memfasilitasi penyelesaian tugas dan tujuan yang diidentifikasi.

- 2) **Efisiensi Kinerja (*Performance Efficiency*)**, yaitu seberapa jauh mana tingkat kapasitas sistem dan jumlah sumber daya yang digunakan (*ISO / IEC, 2011*).

- a) Perilaku Waktu (*Time Behavior*), yaitu seberapa jauh mana
 - b) Pemanfaatan Sumber Daya (*Resource Utilization*), yaitu seberapa jauh mana ketika fungsi dijalankan sebagai kualifikasi, berapa banyak sumber daya yang digunakan pada produk.
 - c) Kapasitas (*Capacity*), yaitu seberapa jauh mana menurut parameter sistem yang memenuhi persyaratan, berapa batas atau kapasitas maksimum produk.
- 3) **Kompabilitas (*Compatibility*)**, yaitu seberapa jauh mana sistem dapat bertukar informasi dalam menjalankan sistem lainnya yang digunakan dengan secara bersamaan di berbagai lingkungan perangkat lunak maupun di lingkungan perangkat keras yang sama (*ISO/IEC, 2011*). Pada bagian ini terbagi menjadi 2, yaitu :
- a) Ko-eksistensi (*Co-existence*), yaitu dengan berbagi lingkungan dan sumber daya dari satu sistem ke sistem lainnya, sejauh mana sistem dapat menjalankan fungsi dengan lebih efisien. Tidak perlu ada efek berbahaya pada sistem.
 - b) Interoperabilitas (*Interoperability*), yaitu seberapa jauh mana satu, dua atau lebih jarak dengan sistem yang dapat bertukar data dan menggunakan informasi yang dipertukarkan sebelumnya.
- 4) **Kegunaan (*Usability*)**, yaitu seberapa jauh mana pengguna tertentu dapat menggunakan produk atau sistem sampai batas tertentu untuk secara efektif mencapai tujuan, efisiensi, dan tingkat kepuasan mereka.
- a) Ketepatan Pengenalan (*Appropriateness Recognizability*), yaitu pengguna dapat memahami sejauh mana sistem atau produk memenuhi kebutuhan mereka.
 - b) *Learnability*, yaitu dengan belajar menggunakan sistem atau produk dengan cara yang efektif, efektif, dan dalam beberapa kasus tanpa risiko dan kepuasan, sejauh mana pengguna dapat menggunakan produk atau sistem tersebut untuk mencapai tujuan tertentu.

- c) Operabilitas (*Operability*), yaitu seberapa jauh mana produk atau sistem mudah dioperasikan dan dikendalikan.
- d) Perlindungan kesalahan pengguna (*User Error Protection*), yaitu seberapa jauh mana produk atau sistem melindungi pengguna dari kesalahan.
- e) Estetika antarmuka pengguna (*User Interface Aesthetics*), yaitu seberapa jauh mana antarmuka pengguna produk atau sistem dapat mencapai interaksi yang memuaskan dan ramah pengguna.
- f) Aksesibilitas (*Accessibility*), yaitu seberapa jauh mana semua kelompok dapat menggunakan produk atau sistem untuk mencapai tujuan tertentu sesuai dengan konteks penggunaannya.

5) **Reliability** yaitu seberapa jauh mana sistem bisa menjalankan fungsi yang ditentukan selama batas waktu yang telah ditentukan (*ISO/IEC, 2011*).

- a) Kematangan (*Maturity*), yaitu seberapa jauh mana produk atau sistem dapat memenuhi persyaratan dengan andal dalam keadaan normal.
- b) Ketersediaan (*Availability*), yaitu seberapa jauh mana produk atau sistem siap untuk dioperasikan dan sejauh mana produk atau sistem dapat diakses saat diperlukan
- c) Toleransi Kesalahan (*Fault Tolerance*), yaitu seberapa jauh mana produk atau sistem terus beroperasi seperti yang diharapkan meskipun ada kesalahan perangkat keras atau perangkat lunak.
- d) Dapat Dipulihkan (*Recoverability*), yaitu seberapa jauh mana produk atau sistem dapat memulihkan data yang terpengaruh secara langsung dan mengatur ulang status sistem sesuai kebutuhan jika terjadi kegagalan

6) **Keamanan (*Security*)**, yaitu seberapa jauh mana sistem bisa melindungi data dan informasi-informasi yang diakses (*ISO/IEC, 2011*).

- a) Kerahasiaan (*Confidentiality*), yaitu seberapa jauh mana produk atau perangkat lunak memastikan bahwa data hanya dapat diakses oleh personel yang berwenang.

- b) Integritas (*Integrity*), yaitu seberapa jauh mana produk atau perangkat lunak dapat mencegah akses tidak sah dan modifikasi data.
- c) Non repudiation (*Non-repudiation*), yaitu seberapa jauh mana peristiwa atau tindakan tersebut dapat dibuktikan terjadi, sehingga peristiwa atau tindakan tersebut tidak akan ditolak.
- d) Keaslian (*Authenticity*), yaitu seberapa jauh mana identitas subjek atau sumber dapat dibuktikan sebagai identitas yang diklaim
- e) Akuntabilitas (*Accountability*), yaitu seberapa jauh mana perilaku suatu entitas dapat ditelusuri kembali secara unik sejauh entitas tersebut.

7) **Pemeliharaan (*Maintanability*)**, yaitu sejauh mana efektivitas dan efisiensi produk atau sistem dapat dicapai. (*ISO/IEC, 2011*).

- a) Modularitas (*Modularity*), yaitu seberapa jauh mana sistem terdiri dari komponen individu, sehingga perubahan atau modifikasi pada salah satu komponen ini berdampak kecil pada komponen lainnya.
- b) Dapat digunakan kembali (*Reusability*), yaitu seberapa jauh mana suatu sistem dapat menggunakan aset lebih banyak atau membangun aset lain
- c) Analisabilitas (*Analysability*), yaitu seberapa jauh mana sistem dapat mengevaluasi dampak perubahan sistem pada diagnosis cacat.
- d) Modifikasi (*Modifiability*), yaitu seberapa jauh mana hasil atau metode dapat digunakan secara efektif dan efisien dapat diubah tanpa kegagalan atau penurunan kualitas produk.
- e) Testabilitas (*Testability*), yaitu seberapa jauh mana nilai tingkat efektivitas dan efisiensi tergantung pada kebutuhan pengujian, dan dapat diterapkan oleh sistem, produk atau komponen, dan dapat juga dicek apakah nilai tersebut telah terpenuhi.

8) **Portabilitas (*Portability*)**, yaitu seberapa jauh mana efektivitas dan efisiensi sistem, produk, atau komponen dapat ditransfer dari satu jenis perangkat keras, perangkat lunak, atau digunakan di lingkungan yang berbeda (*ISO/IEC, 2011*).

- a) *Adaptability*, yaitu seberapa jauh mana hasil atau metode tersebut efektif atau efektif, dan disesuaikan dengan penggunaan perangkat keras (hardware), perangkat lunak (software) dan lingkungan operasi / langit yang berbeda.
- b) *Installability*, yaitu seberapa jauh mana perangkat sistem ini telah berhasil diinstal (dipasang) dan / atau dihapus (dihapus) dalam operasi tertentu.
- c) *Replaceability*, yaitu seberapa jauh mana suatu produk atau sistem dapat diganti dengan produk atau sistem lain yang ditujukan untuk tujuan yang sama di lingkungan yang sama

2.11 Skala Pengukuran

Penelitian ini peneliti menggunakan skala likert sebagai skala pengukuran.

“Skala likert adalah skala yang didasarkan dari penjumlahan sikap responden yang menanggapi pernyataan yang terkait indikator suatu variabel atau konsep yang diukur” (Sugiyono:2017).

Pada umumnya, skala likert menggunakan lima titik dengan label netral pada posisi tengah atau ketiga. Berikut skala likert dapat dilihat pada tabel 2.6

Tabel 2. 6Skala Likert

Jawaban	Skor
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

Sumber : Sugiyono (2017)

Dari hasil penilaian responden maka akan dihitung nilai *persentase* kelayakan menggunakan perhitungan dibawah ini :

$$\text{Persentase} = \frac{\text{Nilai Aktual}}{\text{Nilai Ideal}} \times 100\%$$

Dari hasil persentase kelayakan yang telah diperoleh, lalu dibandingkan dengan tabel konversi yang berpedoman dari acuan konversi nilai. Skala Konversi Nilai dapat dilihat pada Tabel 2.7 dibawah ini :

Tabel 2. 7 Skala Konversi Nilai

Persentase Pencapaian (%)	Interpretasi
90-100	Sangat Baik
80-90	Baik
70-80	Cukup
60-70	Kurang
<60	Sangat Kurang

2.12 Metode Slovin

Metode Slovin adalah sebuah rumus atau formula untuk menghitung jumlah sampel minimal apabila nilai dari populasi tidak diketahui secara pasti (Sugiyono, 2011). Ukuran sampel ditentukan dengan menggunakan rumus Slovin:

$$n = \frac{N}{1 + (N.e^2)}$$

Keterangan:

n : Jumlah Sampel

N : Jumlah Populasi

e : Total jumlah butir yang gagal (5%, 10%)