

BAB II

LANDASAN TEORI

Berikut adalah beberapa penelitian yang diperlukan tinjauan pustaka yang diambil dari beberapa jurnal penelitian yang berkaitan dengan judul penelitian ini dan pokok bahasan, dapat dilihat pada tabel 2.1.

Tabel 2. 1Tinjauan Pustaka

No	Nama Penulis	Judul Penelitian	Metode	Hasil
Literature 01	(Hariadi, Rokhmawati and Kharisma, 2018)	Pengembangan Sistem Informasi Purchasing Order Berbasis Web pada PT. Karya Teknik Mandiri	Metode Penelitian <i>Prototype</i> , Metode Pengujian <i>Validation Testing</i> dan <i>User Acceptance Testing</i>	Hasil penelitian telah berhasil menjadi solusi permasalahan dan dapat diterima oleh pengguna sistem
Literature 02	(Alfiah dan Damayanti, 2020)	Aplikasi E-Marketplace Penjualan Hasil Panen Ikan Lele (Studi Kasus: Kabupaten Pringsewu Kecamatan Pagelaran)	Metode Penelitian <i>Prototype</i> , Metode Pengujian ISO 9126	Hasil penelitian yang dicapai adalah Aplikasi E-Marketplace penjualan hasil panen ikan lele ini dapat memberikan kemudahan bagi penjual dan pembeli dalam memasarkan ikan lele serta mendapatkan informasi mengenai harga dan stok ikan lele.

Literature 03	(Gumelar, Astuti and Sunarni, 2017)	Sistem Penjualan Online Dengan Metode <i>Extreme Programming</i>	Metode <i>Extreme Programming</i> dan pengujian sistem <i>black box</i>	Hasil peneltian yaitu aplikasi penjualan bisa menangani penjualan dan penyediaan barang dan membantu pelanggan untuk bisa melakukan reservasi secara online dengan cepat tanpa harus menunggu lama.
Literature 04	(Fitriyana dan Sucipto, 2020)	Sistem Informasi Penjualan Oleh Sales Marketing Pada PT Erlangga Mahameru	Metode pengembangan sistem <i>prototype</i> dan pengujian sistem <i>black box</i>	Hasil penelitian yaitu sistem informasi yang dapat mempermudah bagian marketing untuk mempromosikan buku.
Literature 05	(Rusliyawati, Oktavia, dan Sucipto, 2021)	Rancang Bangun Aplikasi E-Marketplace Untuk Produk Titik Media Reklame Perusahaan Periklanan (Studi Kasus : P3I Lampung)	Metode pengembangan sistem <i>extreme programming</i> dan metode pengujian ISO 25010	hasil penelitian yaitu sistem aplikasi e-marketplace titik media reklame pada P3I Lampung untuk mempermudah P3I Lampung dalam mendapatkan keuntungan finansial dan tepat mengenai titik reklame yang akan dipesan.

1. Literature 01

Oleh Hariadi, Rokhmawati and Kharisma meneliti tentang Pengembangan Sistem Informasi Purchasing Order Berbasis Web pada PT. Karya Teknik Mandiri. PT. Karya Teknik Mandiri memiliki permasalahan pada proses bisnis purchasing order. Permasalahan tersebut diantaranya, seringkali terjadi ketidaksesuaian antara surat jalan dengan purchasing order customer, rusak atau hilangnya purchasing order customer, serta sulitnya komunikasi pelanggan dengan perusahaan di dalam membahas pemesanan. Oleh karena itu, diberikan suatu solusi dengan mengganti kondisi proses bisnis saat ini ke dalam proses bisnis usulan dengan penerapan sistem informasi yang dilengkapi dengan fitur otomatisasi pelayanan untuk proses bisnis purchasing order PT. Karya Teknik Mandiri. Metode pengembangan sistem menggunakan metode Prototype serta pendekatan Operational CRM modul Service Automation. Pengujian sistem dilakukan dengan Validation Testing dan User Acceptance Testing, dengan hasil pengujian memiliki nilai valid 100% dan nilai penerimaan “Sangat Baik” dari pengguna sistem.

2. Literature 02

Oleh Alfiah and Damayanti meneliti tentang Aplikasi E-Marketplace Penjualan Hasil Panen Ikan Lele (Studi Kasus: Kabupaten Pringsewu Kecamatan Pagelaran). Dalam hal pemasaran, petani merasa sangat kurang karena tidak ada media yang mendukung. Proses bisnis yang saat ini berjalan yakni pembeli yang akan membeli harus datang langsung ke rumah dan apabila sudah berlangganan dapat memesan melalui telepon. Kurangnya informasi mengenai harga ikan lele membuat pelanggan atau calon pembeli merasa rugi jika ternyata ada beberapa petani yang memberikan harga yang murah. Oleh sebab

itu penjualan ikan lele dapat di tunjang dengan memanfaatkan teknologi informasi sebagai pendukung dalam proses penjualan dan pemasaran ikan lele. Hasil yang dicapai adalah Aplikasi E-Marketplace penjualan hasil panen ikan lele ini dapat memberikan kemudahan bagi penjual dan pembeli dalam memasarkan ikan lele serta mendapatkan informasi mengenai harga dan stok ikan lele. Hasil dari pengujian ISO 9126 yang telah dilakukan dengan melibatkan 14 Responden bahwa kesimpulan kualitas kelayakan perangkat lunak yang dihasilkan secara keseluruhan mempunyai nilai 95,58%.

3. Literature 03

Oleh Gumelar, Astuti and Sunarni meneliti tentang Sistem Penjualan Online Dengan Metode *Extreme Programming*. Terbatasnya jangkauan produk pemasaran dan kurangnya sistem promosi produk yang efisien kepada masyarakat

menyebabkan kegagalan untuk mendapatkan keuntungan maksimal dari perusahaan. Pengolahan data yang tidak terintegrasi dengan baik, mengakibatkan pelanggan menunggu terlalu lama untuk membeli produk yang ready stock. Masalah yang terjadi, perusahaan membutuhkan sistem informasi yang saling terintegrasi sehingga menghasilkan produk dan pemesanan barang yang bisa diolah dengan cepat dan mudah. Tujuan penelitian ini adalah membuat aplikasi yang bisa menangani penjualan dan penyediaan barang dan membantu pelanggan untuk bisa melakukan reservasi secara online dengan cepat tanpa harus menunggu lama. Penulis mengembangkan sistem e-commerce menggunakan pendekatan metodologi *Extreme Programming*, yang dianggap tepat saat ini karena semuanya membutuhkan waktu yang cepat.

4. Literature 04

Oleh Fitriyana and Sucipto meneliti tentang Sistem Informasi Penjualan Oleh Sales Marketing Pada PT Erlangga Mahameru. Proses penjualan buku yang telah diterbitkan ditujukan bagi sekolah-sekolah, masih memiliki beberapa kendala seperti sekolah harus datang ke perusahaan untuk melakukan pemesanan buku atau melalui jasa, dan kurangnya informasi yang didapatkan oleh pihak-pihak sekolah terhadap buku – buku yang diterbitkan berdasarkan kurikulum yang berlaku di Indonesia. Penelitian ini dilakukan untuk membuat sistem informasi yang dilakukan oleh sales marketing. Perancangan sistem ini menggunakan UML, Usecase Diagram, serta Activity Diagram. Sedangkan bahasa pemrograman yang digunakan adalah PHP, untuk databasenya menggunakan MySQL. Hasil dari penelitian ini adalah sistem informasi yang dapat mempermudah bagian marketing untuk mempromosikan buku, selain itu konsumen juga dapat melakukan transaksi secara online. Sistem informasi ini diuji menggunakan black box serta dilakukan pengukuran pengujian menggunakan quisioner. Hasil pengujian adalah sebesar 89,58%. Dengan

5. Literature 05

Oleh Rusliyawati, Oktavia and Sucipto meneliti tentang Rancang Bangun Aplikasi E-Marketplace Untuk Produk Titik Media Reklame Perusahaan Periklanan (Studi Kasus : P3I Lampung). Pada proses pemesanan reklame pelanggan masih dilakukan secara manual yaitu dicatat didalam buku selanjutnya akan direkap kedalam excel sehingga lama dalam proses perekapan dikarenakan harus melihat buku satu persatu, dan tidak adanya laporan secara periode. Pada penelitian ini dibahas bagaimana membuat dan merancang aplikasi marketplace dimulai dari metode pengumpulan data (wawancara,

pengamatan dan dokumentasi) menggunakan metode pengembangan Extreme Programming, pembuatan rancangan sistem menggunakan UML dengan model perancangan Usecase Diagram, Activity Diagram, Class Diagram. Hasil pengujian ISO 25010 yang telah dilakukan dengan melibatkan 50 Responden bahwa kesimpulan kualitas kelayakan perangkat lunak dengan skor 91.62% yang dihasilkan secara keseluruhan mempunyai skala “Sangat Baik”.

Berdasarkan penelitian terdahulu kaitan dan perbedaan dengan peneliti yaitu:

1. Memberikan informasi mengenai barang yang dijual.
2. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *prototype*.
3. Pengujian sistem menggunakan *ISO 25010* dengan aspek *functionality*, *Reliability* dan *efficiency*
4. Dilakukan menggunakan *web*.

2.2. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dalam kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Kristanto, 2018).

Menurut Maniah and Haminidin agar sistem dapat dikatakan sistem yang baik, maka sistem harus memiliki:

1. Komponen Sistem (*Componentsystem*)

Memiliki beberapa komponen yang saling berinteraksi, yang memiliki arti saling bekerja sama untuk membentuk satu kesatuan. Komponen sisten sendiri terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.

2. Batasan Sistem (*Boundarysystem*)

Batasan sistem adalah batasan antara sistem satu dengan sistem lain. Batasan sistem ini dapat memungkinkan sistem ini ini dipandang sebagai satu kesatuan. Batasan dari suatu sistem juga menunjukkan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environmentsystem*)

Lingkungan luar sistem adalah kondisi yang ada di luar sistem, namun dapat memengaruhi jalan atau beroperasinya sistem. Lingkungan sendiri dapat bersifat menguntungkan dan merugikan, sifat menguntungkan ini harus tetap dijaga, sedangkan sifat merugikan harus dijaga dan dikendalikan agar tidak merusak kelangsungan hidup sistem.

4. Penghubung Sistem (*Interfacedsystem*)

Penghubung sistem adalah media bantu untuk menghubungkan antara suatu subsistem dengan subsistem lain. Keluaran dari subsistem akan menjadi masukan untuk subsistem lain melalui penghubung.

5. Masukan Sistem (*inputsystem*)

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem, yang berupa perawatan dan masukan sinyal. Perawatan input adalah energi yang dimasukkan agar sistem dapat beroperasi, sedangkan sinyal input adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem (*outputsystem*)

Keluaran sistem adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang memiliki nilai guna dan sisa pembuangan.

7. Pengolahan Sistem (*processsystem*)

Pengolahan sistem adalah bagian yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*objectivesystem*)

Sasaran sistem adalah tujuan (*goal*) atau sasaran (*objective*) dari sebuah sistem. Sasaran dari sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan oleh sistem.

2.3. *E-Purchasing*

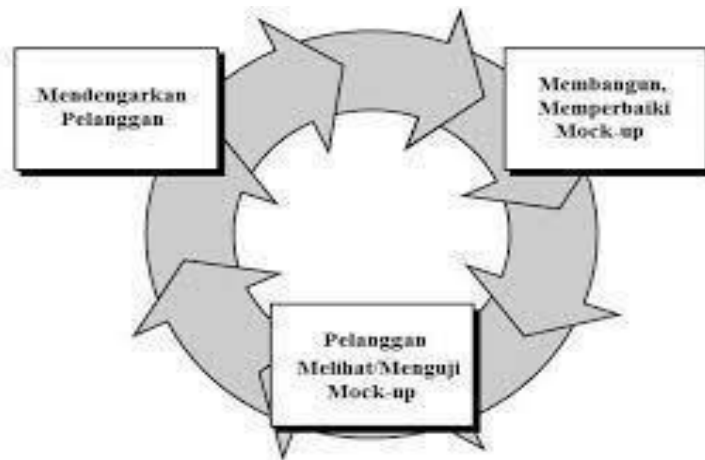
E-Purchasing adalah sistem yang menerapkan suatu aktifitas yang dilakukan oleh konsumen sebelum membeli. *E-Purchasing* juga dapat diartikan sekumpulan alat elektronik pemasaran yang dapat digunakan oleh perusahaan untuk mencapai tujuan pemasarannya dalam pasar sasaran (Swasta and Irwan, 2015).

E-Purchasing adalah proses kegiatan secara elektronik yang berkaitan dengan pengelolaan inventory atau persediaan tempat pendistribusian produk dan catatan keseluruhan transaksi pemesanan tempat baik untuk penumpang maupun barang (cargo) (Jusup, 2016).

2.4. Metode Pengembangan Pada *Prototype*

Model prototype dapat digunakan untuk menyambung ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa and Shalahudin, 2018).

Adapun tahapan – tahapan dalam metode *Prototype* dapat dilihat pada gambar 2.1:



Gambar 2. 1 Tahapan – Tahapan Metode *Prototype*

(Rosa and Shalahudin, 2018).

Pada metode *prototype* terdapat tiga tahap yaitu (Rosa and Shalahudin, 2018):

1. Mendengarkan Pelanggan

Pada tahap ini, dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengar kebutuhan pelanggan sebagai pengguna sistem perangkat lunak untuk menganalisis serta mengembangkan kebutuhan pengguna. Pada tahapan ini pelanggan dan pengembang bersama sama mendefinisikan format seluruh perangkat lunak, semua kebutuhan sistem yang akan dibangun, dan garis besar sistem yang dibuat. Dalam tahapan ini dilakukan dengan cara pengumpulan data (wawancara, observasi, dan dokumentasi) selanjutnya dibuatkan analisis kebutuhan sistem yang akan dibangun seperti analisis fungsional dan analisis non fungsional.

2. Merancang dan Membuat Prototipe

Pada tahap ini, dilakukan perancangan dan pembuatan prototipe sistem yang disesuaikan dengan kebutuhan pengguna. Pengguna membuat rancangan sementara yang berfokus dalam menggambarkan sistem untuk disajikan kepada pelanggan dengan cara membuat gambaran alur sistem menggunakan UML, penggambaran rancangan aplikasi yang akan dibuat, setelah itu tahapan membuat sistem menggunakan bahasa pemrograman PHP dan MySQL sebagai *database*.

3. Uji Coba

Pada tahap ini, dilakukan pengujian *prototipe* sistem oleh pengguna kemudian dilakukan evaluasi sesuai dengan kekurangan-kekurangan dari kebutuhan pelanggan. Jika sistem sudah sesuai dengan prototipe, maka sistem akan diselesaikan sepenuhnya. Namun, jika masih tidak sesuai kembali ke tahap pertama. Dalam tahapan ini pengujian sistem dapat dilakukan menggunakan teknik pengujian sistem yaitu metode pengujian *black box* atau ISO 25010 yang bertujuan untuk melakukan evaluasi apakah sistem layak digunakan atau tidak sesuai dengan kebutuhan pengguna.

2.4.1. Kelebihan *Prototype*

1. Adanya komunikasi yang baik antara pengembang dan pelanggan
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
3. Pelanggan berperan aktif dalam pengembangan system
4. Lebih menghemat waktu dalam pengembangan system
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.

2.4.2. Kekurangan *Prototype*

1. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu lama.
2. Pengembang biasanya ingin cepat menyelesaikan proyek, Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat *prototyping* lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.

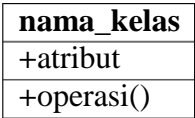






2.5. Perancangan Sistem UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa and Shalahudin, 2018).

1. Class Diagram

Diagram kelas atau *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, Simbol-simbol yang ada pada diagram kelas pada tabel 2.2 di bawah ini:

Tabel 2. 2Simbol Class Diagram






Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>Interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>asociation</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
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>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependecy</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>agregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)


2. Use Case Diagram

Use case diagram atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat (Rosa and Shalahudin, 2018).

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Simbol-simbol yang ada pada diagram *use case* dapat dilihat pada gambar 2.3 di bawah ini:

Tabel 2. 3 Simbol diagram *use case*

Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan 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 di awal frase nama <i>actor</i></p>
<p>Asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan <i>actor</i></p>
<p>Ekstensi/<i>extend</i></p> <p><<<i>extend</i>>></p> 	<p>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 mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan</p>
<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
<p>Menggunakan/<i>Include/uses</i></p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use</i></p>

Simbol	Deskripsi
<code><<include>></code> 	<i>case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk <u>menjalankan</u> fungsinya

2.6. Pengertian MySQL

MySQL adalah sebuah relasional *database server* yang menawarkan berbagai macam mekanisme dalam pengolahan data yang dikenal dengan *storage engines*(Gilmore, 2015).

MySQL sama seperti sistem *DBMS lainnya, seperti Oracle, DB2, dan SQL Server*. Memungkinkan untuk mengakses, memanipulasi, melindungi, dan memelihara metadata yang diperlukan untuk mendefinisikan data yang disimpan(Sheldon and Moes, 2015).

2.7. XAMPP

XAMPP adalah paket program *web* lengkap yang dapat Anda pakai untuk belajar pemrograman *web*, khususnya PHP dan MySQL (Nugroho, 2013).

XAMPP adalah perangkat lunak *opensource* yang diunggah secara gratis dan bisa dijalankan di semua semua operasi seperti *windows, linux, solaris, dan mac* (Buana, 2014).

2.6.1. Kelebihan

1. Performanya yang tinggi, stabil, memiliki banyak fitur
2. Mudah dikonfigurasi
3. Menggunakan hanya sedikit sumberdaya pada *server*
4. Tidak bergantung kepada *thread* untuk melayani *klien*

2.6.2. Kekurangan

1. Belum support IPV6
2. *Update/patchnya* versi terbarunya lama keluar

3. *Fast-CGI*nya tidak berfungsi *maximal*

4. Pemakainya tidak sebanyak

2.8. Pengujian ISO 25010

Model ISO-25010 merupakan bagian dari *Software product Quality Requirements and Evaluation (SQuaRE)*, yang merupakan pengembangan dari model kualitas perangkat lunak sebelumnya yaitu ISO-9126. Dalam model ISO-25010 ini digunakan untuk melihat kualitas suatu perangkat lunak yang digunakan oleh perusahaan, instansi ataupun organisasi. Metode ISO 25010 ini dapat digunakan untuk mengevaluasi kualitas sistem perangkat lunak secara spesifik berdasarkan dua dimensi umum, yaitu dimensi *product quality*, dimana prosesnya mengacu pada karakteristik intrinsik dari sebuah produk perangkat lunak, memiliki beberapa elemen antara lain meliputi *functional suitability, reliability, operability, performance efficiency, security, compatibility, maintainability* dan *transferability*. *Quality in use* dan *product quality*. Sedangkan pada *dimensionality in use*, terdapat beberapa karakteristik relatif yang ditinjau dari perspektif user antara lain *Usability in use, Flexibility in use, dan Safety* (Alfian, 2010).

Adapun dimensi yang pertama terdapat beberapa faktor elemen diantaranya

- 1) *Functionality* (Fungsionalitas). Kemampuan perangkat lunak untuk Merupakan tingkatan dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi spesifik tertentu dalam hal ini perangkat lunak dapat memenuhi kelayakan dari sebuah fungsi untuk melakukan pekerjaan yang spesifik bagi

pengguna dan dapat memberikan hasil yang tepat dan ketelitian terhadap tingkat kebutuhan pengguna.

- 2) *Reliability* Merupakan tingkatan dimana perangkat lunak dapat bertahan pada tingkatan tertentu ketika digunakan oleh pengguna pada kondisi yang spesifik dalam hal ini perangkat lunak dapat beroperasi dan siap ketika dibutuhkan untuk digunakan dan juga dapat bertahan pada tingkat kemampuan tertentu terhadap kegagalan, kesalahan serta perangkat lunak kembali pada tingkat tertentu dalam mengembalikan pengembalian data yang disebabkan kegagalan atau kesalahan pada perangkat lunak.
- 3) *Performance efficiency* Merupakan tingkatan dimana perangkat lunak dapat memberikan kinerja terhadap sejumlah sumber daya yang digunakan pada kondisi tertentu dalam hal ini *performance efficiency* dapat memberikan reaksi dan waktu yang dibutuhkan ketika melakukan aksi dari sebuah fungsi dan perangkat lunak dapat menggunakan sejumlah sumber daya ketika melakukan aksi dari sebuah fungsi.
- 4) *Operability* Perangkat lunak dapat dimengerti, dipelajari, digunakan, dan menarik pengguna ketika digunakan dalam hal ini perangkat lunak mudah dipelajari oleh pengguna, perangkat lunak dapat digunakan dan dioperasikan oleh pengguna, perangkat lunak dapat memberikan bantuan ketika pengguna membutuhkan panduan, perangkat lunak dapat menarik perhatian pengguna, perangkat lunak memenuhi kebutuhan pengguna yang memiliki keterbatasan dan perangkat memungkinkan untuk dianalisis oleh pengguna apakah perangkat lunak sudah memenuhi kebutuhan mereka.

- 5) *Security* Merupakan perlindungan terhadap perangkat lunak dari berbagai ancaman atau keganjalan dalam hal ini perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna dan merupakan dari kelengkapan, ketepatan dari sejumlah *asset* yang telah dijaga sehingga aksi atau tindakan yang dilakukan telah terbukti dan hal tersebut tidak dapat ditolak.
- 6) *Compability* Faktor ini merupakan kemampuan dari dua atau lebih komponen perangkat lunak dapat melakukan pertukaran informasi dan melakukan fungsi yang dibutuhkan ketika digunakan pada *hardware* atau lingkungan perangkat lunak yang sama.
- 7) *Maintainability* Merupakan tingkat dimana sebuah perangkat lunak dapat dimodifikasi. Dalam hal ini modifikasi adalah perbaikan, perubahan atau penyesuaian perangkat lunak untuk dapat berubah pada lingkungan , kebutuhan dan fungsionalitas yang spesifik. Selain itu perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak untuk mengidentifikasi bagian yang dapat dimodifikasi.
- 8) *Transferability*. Merupakan kemudahan dimana sistem atau komponen dapat berpindah dari lingkungan satu ke lingkungan yang lain dalam hal ini perangkat lunak dapat beradaptasi dengan cepat pada spesifikasi lingkungan yang berbeda tanpa menerapkan aksi atau cara lain dari pada memberikan tujuan tertentu terhadap perangkat lunak yang telah ada.

2.9. Skala Pengukuran

Skala pengukuran yang digunakan adalah skala Likert, skala yang didasarkan pada penjumlahan sikap responden dalam merespon pernyataan berkaitan indikator-indikator suatu konsep atau variable yang sedang diukur (Sugiyono, 2017). Skala Likert umumnya menggunakan lima titik dengan label kurang setuju pada posisi tengah (ketiga). Skala Likert dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Skala Likert

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

Hasil penilaian responden akan dihitung *persentase* kelayakannya dengan menggunakan perhitungan, dapat dilihat dibawah ini:

$$\text{Persentase} = \frac{\text{Skor Aktual (f)}}{\text{Skor Ideal (n)}} \times 100\%$$

Persentase kelayakan yang diperoleh kemudian dibandingkan dengan Tabel konversi yang berpedoman pada acuan konversi nilai, dapat dilihat pada Tabel 2.5.

Tabel 2. 5 Skala Konversi Nilai

Persentase Pencapaian (%)	Interpretasi
$90 \leq x$	Sangat Baik
$80 \leq x < 90$	Baik
$70 \leq x < 80$	Cukup
$60 \leq x < 70$	Kurang
$X < 60$	Sangat Kurang