

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Beberapa penelitian terdahulu yang relevan dengan sistem penunjang keputusan pemilihan tenant inkubator bisnis berdasarkan jurnal penelitian terlihat pada tabel 2.1 :

Tabel 2.1 Tinjauan Pustaka

No	Judul Jurnal	Metode	Hasil Penelitian
1	(Santoso <i>et al.</i> , 2022) “Sistem Pendukung Keputusan Penerimaan Calon Binaan Inkubator Wirausaha Menggunakan Metode Perbandingan Eksponensial”	Metode Perbandingan Eksponensial (MPE)	Penelitian ini merupakan penelitian terkait pengembangan sistem pendukung keputusan untuk penerimaan calon binaan inkubator wirausaha berbasis WEB menggunakan metode perbandingan eksponensial (MPE) serta bahasa pemrograman PHP untuk membangun sistem. Tahapan penelitian yang dilakukan yaitu proses identifikasi masalah, pendekatan menggunakan metode perbandingan eksponensial (MPE), pengembangan sistem menggunakan PHP, implementasi pada Inkubator Wirausaha Nusa Tenggara Barat. Pada penelitian ini terdapat 6 alternatif dan untuk proses perhitungan alternatif terbaik dengan cara menghitung nilai hasil seleksi calon binaan, kemudian dipangkatkan dengan

Tabel 2.1 Tinjauan Pustaka (lanjutan)

No	Judul Jurnal	Metode	Hasil Penelitian
			masing-masing bobot kriteria. Setelah hasil telah dipangkatkan lalu dijumlahkan, hasil alternatif yang terbesar akan terpilih sebagai calon binaan pada Inkubator Wirausaha Nusa Tenggara Barat (NTB). Dari 6 alternatif yang ada, alternatif terbaik adalah Mirase dengan nama perusahaan UD.Nice Mataram dengan skor 11280.
2	(Kurniawati and Ahmad, 2021) “Sistem Pendukung Keputusan Penentuan Kelayakan Usaha Mikro Kecil Menengah Dengan Menggunakan Metode Profile Matching Pada Uptd Plut Kumkm Provinsi Lampung”	<i>Profile Matching</i>	Penelitian ini merupakan penelitian terkait pembuatan sebuah sistem pendukung keputusan untuk menentukann kelayakan UMKM dengan menggunakan metode profile matching. Proses perhitungan dalam menentukan alternatif terbaik adalah dengan penentuan aspek dan sub kriteria kemudian dikelompokkan menjadi 2 aspek penilaian, yaitu aspek finansial dan non finansial. Kemudian dilakukan proses implementasi dan pengujian menggunakan metode black box testing.

Tabel 2.1 Tinjauan Pustaka (lanjutan)

No	Judul Jurnal	Metode	Hasil Penelitian
3	(Apriliani, Jayanti and Renaningtias, 2020) “Implementasi Metode Ahp-Topsis Dalam Sistem Pendukung Keputusan Penentuan Prioritas Bantuan Usaha Kecil Dan Menengah Di Kota Tegal”	<i>Analytical Hierarchy Process</i> (AHP) dan <i>Technique for Order Preference by Similarity to Ideal Solution</i> (TOPSIS)	<p>Penelitian ini berisi tentang penentuan prioritas penerima bantuan Usaha Kecil dan Menengah yang ada di Kota Tegal. Metode AHP digunakan untuk menentukan bobot kriteria, setelah melakukan perhitungan terhadap bobot kriteria selanjutnya dilakukan proses perankingan menggunakan metode TOPSIS. Alur penelitian terdiri dari beberapa langkah yaitu :</p> <ol style="list-style-type: none"> 1. Identifikasi masalah 2. Analisa masalah 3. Pengumpulan data, proses ini dilakukan dengan beberapa metode, yaitu observasi, study literatur, dan wawancara 4. Analisa data 5. Perancangan sistem menggunakan <i>Unified Modeling Language</i> (UML). <p>Pembuatan sistem menggunakan bahasa pemrograman PHP dan XAMPP sebagai</p>

Tabel 2.1 Tinjauan Pustaka (lanjutan)

No	Judul Jurnal	Metode	Hasil Penelitian
			<p>database server</p> <p>6. Pengujian sistem menggunakan metode <i>black box testing</i> dan <i>usability testing</i></p> <p>7. Hasil yang didapat dari penelitian ini adalah sebuah sistem pendukung keputusan yang dapat memberikan rekomendasi penerima bantuan UKM dengan akurat.</p>
4	(Dona, Yasdomi and Utami, 2018) “Sistem Pendukung Keputusan Karyawan Terbaik Menggunakan Metode Weight Product (WP) (Studi Kasus : Universitas Pasir Pengaraian)”	<i>Weighted Product (WP)</i>	<p>Penelitian ini berisi tentang penentuan karyawan terbaik menggunakan metode <i>Weighted Product</i>. Proses perhitungan adalah dengan menentukan kriteria dan bobot kriteria terlebih dahulu, kemudian dilakukan analisis perankingan setiap kriteria dengan rating Sangat Baik (SB) hingga Kurang Baik (KB), langkah selanjutnya adalah menentukan rating kecocokan antara kriteria dan alternatif yang ada, dan terakhir dilakukan perhitungan perankingan setiap alternatif</p>

Tabel 2.1 Tinjauan Pustaka (lanjutan)

No	Judul Jurnal	Metode	Hasil Penelitian
			dengan menghitung bobot kriteria, menghitung nilai vektor S, menghitung nilai vektor V, melakukan perankingan nilai vektor V hingga akhirnya didapatkan hasil alternatif terbaik.
5	(Priandika and Wantoro, 2017) Sistem Pendukung Keputusan Penerimaan Calon Siswa Baru pada SMK SMTI Bandar Lampung dengan Menggunakan Metode Simple Additive Weighting (SAW)	<i>Simple Additive Weighting</i> (SAW)	Penelitian ini berisi tentang perancangan sebuah sistem penunjang keputusan yang digunakan untuk penerimaan calon siswa baru pada SMK SMTI Bandar Lampung menggunakan metode SAW. Metode yang digunakan dalam penelitian ini memiliki 5 tahapan, yaitu tahap pengumpulan, tahap analisis, tahap perancangan, tahap implementasi, dan penutup. Proses perhitungan tiap alternatif dilakukan dengan terlebih dahulu menentukan kriteria dan bobot kriteria yang dibutuhkan. Kemudian dilakukan pembuatan rating kecocokan antara alternatif yang tersedia dengan kriteria yang telah ditentukan. Kemudian dilakukan proses

Tabel 2.1 Tinjauan Pustaka (lanjutan)

No	Judul Jurnal	Metode	Hasil Penelitian
			<p>normalisasi matriks keputusan yang nantinya akan menghasilkan matriks ternormalisasi. Dan terakhir adalah menentukan nilai preferensi yang ditentukan dari setiap alternatif yang dijumlahkan dengan hasil kali dari matriks ternormalisasi dengan nilai bobot. Penelitian ini menghasilkan sebuah sistem pendukung keputusan yang memberikan kontribusi yang positif dalam mendukung proses pemilihan siswa baru di SMK SMTI Bandar Lampung.</p>

2.1.1. Perbedaan Dengan Penelitian Terdahulu

- a. Penelitian yang dilakukan oleh (Santoso *et al.*, 2022) dengan judul Sistem Pendukung Keputusan Penerimaan Calon Binaan Inkubator Wirausaha Menggunakan Metode Perbandingan Eksponensial, terdapat beberapa persamaan dan perbedaan dengan penelitian ini. Persamaan antara penelitian yang penulis lakukan dengan penelitian yang dilakukan oleh (Santoso *et al.*, 2022) adalah dari segi topik yang diangkat, yaitu tentang penerimaan calon binaan inkubator bisnis. Sedangkan perbedaannya terletak pada metode yang digunakan, serta pada studinya. Penulis

menggunakan metode *Simple Additive Weighting*, dimana metode ini belum pernah digunakan oleh penelitian terdahulu pada topik yang terkait.

- b. Penelitian yang dilakukan oleh (Kurniawati and Ahmad, 2021) dengan judul Sistem Pendukung Keputusan Penentuan Kelayakan Usaha Mikro Kecil Menengah Dengan Menggunakan Metode Profile Matching Pada Uptd Plut Kumkm Provinsi Lampung, juga memiliki perbedaan dengan penelitian yang penulis lakukan saat ini dari sisi metode dimana pada penelitian tersebut menggunakan metode profile matching untuk menentukan kelayakan UMKM.
- c. Penelitian yang dilakukan oleh (Apriliani, Jayanti and Renaningtias, 2020) dengan judul Implementasi Metode Ahp-Topsis Dalam Sistem Pendukung Keputusan Penentuan Prioritas Bantuan Usaha Kecil Dan Menengah Di Kota Tegal. Pada penelitian ini terdapat beberapa perbedaan dari penelitian yang penulis lakukan. Perbedaan penelitian yang penulis lakukan dengan penelitian ini terletak pada metode yang digunakan.
- d. Penelitian yang dilakukan oleh (Dona, Yasdomi and Utami, 2018) dengan judul Sistem Pendukung Keputusan Karyawan Terbaik Menggunakan Metode Weight Product (WP) (Studi Kasus : Universitas Pasir Pengaraian), memiliki perbedaan pada topik penelitian dan metode yang digunakan. Pada penelitian ini topik penelitian lebih berfokus pada pembuatan sebuah sistem penunjang keputusan untuk menyeleksi karyawan terbaik pada universitas pasir pengaraian menggunakan metode Weight Product (WP).
- e. Penelitian yang dilakukan oleh (Priandika and Wantoro, 2017) dengan judul Sistem Pendukung Keputusan Penerimaan Calon Siswa Baru pada SMK

SMTI Bandar Lampung dengan Menggunakan Metode Simple Additive Weighting (SAW). Perbedaan antara penelitian yang penulis lakukan dengan penelitian ini terletak pada topik penelitian nya serta lokasi penelitian. Penelitian ini berfokus pada pembuatan sistem penunjang keputusan untuk penerimaan calon siswa baru pada SMK SMTI Bandar Lampung.

2.2. Sistem Penunjang Keputusan

2.2.1. Konsep Sistem

Menurut (Romney and Steinbart, Paul John, Summers, Scott L, Wood, 2021) sistem merupakan sekumpulan metode, prosedur, dan rutinitas yang terperinci yang melakukan aktivitas yang spesifik, melakukan suatu tugas, mencapai tujuan, ataupun memecahkan satu atau lebih masalah. Sebagian besar sistem terdiri dari subsistem yang lebih kecil yang mendukung sistem yang lebih besar.

Kemudian menurut (Rahmawati and Sumarno, 2020) sistem adalah sekumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan dalam usahanya mencapai suatu tujuan.

Dari pemaparan pendapat ahli di atas, maka penulis dapat memberikan kesimpulan bahwa sistem merupakan sekumpulan komponen atau sub sistem yang saling berinteraksi dalam satu kesatuan untuk mencapai tujuan tertentu.

2.2.2. Karakteristik Sistem

Menurut (Anna, Nurmalasari and Yusnita, 2018) sistem memiliki beberapa karakteristik atau ciri-ciri agar dapat dikategorikan sebagai suatu sistem yang baik.

Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen Sistem (*Components*)

Suatu sistem dapat tercipta karena adanya sejumlah komponen yang melakukan interaksi. Sekecil apapun sistem tersebut pasti terdiri dari beberapa komponen.

2. Batasan Sistem (*Boundary*)

Batasan sistem atau *boundary* merupakan daerah yang membatasi satu sistem dengan sistem yang lainnya ataupun antara sistem dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem atau *environment* merupakan daerah yang berada diluar batas dari sebuah sistem yang mempengaruhi operasi dari sistem tersebut.

4. Penghubung Sistem (*interface*)

Interface merupakan Media yang menghubungkan antara subsistem dengan subsistem yang lain. Penghubung ini memungkinkan sumber-sumber daya beredar dari satu subsistem ke subsistem yang lain. Bentuk *output* dari satu subsistem akan berubah menjadi *input* untuk subsistem lain melalui penghubung tersebut.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut dengan masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi *output* yang berguna. *Output* ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. *Output* yang dihasilkan yaitu informasi. Informasi ini dapat digunakan sebagai *input* untuk sistem pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi subsistem lain.

7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah (*process*) atau sistem tersebut yang menjadi pengolahnya sendiri. Hasil dari pengolahan ini akan merubah *input* menjadi *output*.

8. Sasaran Sistem (*Objective*)

Sebuah sistem memiliki tujuan (*goal*) dan sasaran (*objective*) yang pasti. Jika sebuah sistem tidak mempunyai sasaran, maka sistem tersebut tidaklah berguna. Suatu sistem dikatakan berhasil apabila mencapai sasaran atau tujuan yang telah direncanakan.

2.2.3. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, menurut (Anna, Nurmalasari and Yusnita, 2018) klasifikasi sistem dilihat dari sudut pandang adalah sebagai berikut :

1. Sistem Alamiah (*Natural System*) Dan Sistem Buatan Manusia (*Artificial System*)

Sistem alamiah merupakan sistem yang terjadi dari proses alam dan tidak dibuat oleh manusia, contohnya adalah sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi antara manusia dengan mesin yang disebut human machine sistem. Sistem informasi berbasis komputer merupakan salah satu contoh human machine sistem karena didalamnya menyangkut penggunaan komputer yang berinteraksi dengan manusia.

2. Sistem Deterministik (*Deterministic System*) Dan Sistem Probabilistik (*Probabilistic System*)

Sistem deterministik adalah sistem yang beroperasi dengan tingkah laku (*behavior*) yang dapat diprediksi. Interaksi antar bagian-bagiannya dapat dideteksi secara pasti sehingga output dari sistem dapat diramalkan. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sementara sistem probabilistik merupakan suatu sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas. Contoh dari sistem probabilistik adalah sistem tubuh manusia.

3. Sistem Terbuka (*Opened System*) Dan Sistem Tertutup (*Closed System*)

Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Lebih spesifiknya dikenal juga dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia serta berinteraksi dengan kontrol dari satu komputer atau lebih. Sistem ini

menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sedangkan sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar. Secara teoritis sistem ini tidak ada, tetapi nyatanya tidak ada sistem yang benar-benar tertutup, namun yang ada hanyalah *relatively closed system*.

2.2.4. Penunjang / Pendukung

Menurut Kamus Besar Bahasa Indonesia (KBBI, 2016) kata pendukung mempunyai arti sesuatu atau orang yang mendukung; pembantu; penyokong; penunjang

2.2.5. Keputusan

Menurut (Widodo and Misdrum, 2019) keputusan merupakan sebuah kesimpulan yang dicapai setelah dilakukannya proses pertimbangan, yang terjadi setelah sebuah kemungkinan dipilih, dan yang lainnya dikesampingkan.

Sedangkan menurut (Simbolon and Sihombing, 2022) keputusan merupakan tindakan yang diambil dari beberapa alternatif yang dibuat untuk mengatasi suatu permasalahan.

2.2.6. Konsep dan Definisi Sistem Penunjang Keputusan

Menurut (Sharda, Delen and Turban, 2020) sistem penunjang keputusan (DSS) merupakan sistem yang mempunyai tujuan untuk membantu pengambilan keputusan di tingkat manajerial dalam situasi semi terstruktur tidak terstruktur. Sistem penunjang keputusan dimaksudkan untuk membantu pembuat keputusan

dan memperluas kemampuan pengambil keputusan, namun tidak menggantikan penilaian mereka.

Kemudian (Mayana, Tarigan and Yunita, 2021) berpendapat bahwa sistem penunjang keputusan merupakan bagian dari sistem informasi berbasis komputer, termasuk sistem berbasis pengetahuan atau manajemen pengetahuan yang digunakan suatu organisasi atau perusahaan untuk mendukung proses pengambilan keputusan. Dapat dikatakan juga bahwa SPK merupakan sistem komputer yang mengolah data menjadi informasi yang digunakan untuk mendukung pengambilan keputusan dari masalah semi terstruktur yang spesifik.

Dari beberapa sumber diatas, dapat disimpulkan bahwa Sistem Pendukung Keputusan adalah suatu sistem yang bertujuan untuk memberikan sebuah dukungan bagi pengambil keputusan dalam menentukan suatu keputusan yang bersifat terstruktur dan semi terstruktur.

2.2.7. Karakteristik Sistem Penunjang Keputusan

Menurut (Mayana, Tarigan and Yunita, 2021) Berikut ini merupakan beberapa karakteristik sistem penunjang keputusan :

1. SPK dapat mendukung pengambilan keputusan untuk membahas masalah-masalah yang bersifat terstruktur, semi terstruktur, maupun tidak terstruktur.
2. *Output* ditujukan untuk personel organisasi pada semua tingkatan.
3. SPK Mendukung semua fase dalam proses pengambilan keputusan.
4. Adanya *interface* manusia atau mesin dimana manusia tetap memegang kendali penuh terhadap pengambilan keputusan.
5. Menggunakan model-model matematis dan statistik sesuai yang sesuai dengan pembahasan.

6. Mempunyai kemampuan dialog untuk mendapatkan informasi yang sesuai dengan kebutuhan.
7. Mempunyai subsistem-subsistem yang terintegrasi sehingga bisa berfungsi sebagai satu kesatuan sistem. Membutuhkan struktur data yang komprehensif yang dapat melayani kebutuhan informasi seluruh tingkatan manajerial.
8. Pendekatan *easy to use*. Ciri-ciri dari suatu sistem pendukung keputusan yang efektif adalah kemudahannya untuk dipakai serta memungkinkan keleluasaan *user* dalam memilih ataupun mengembangkan pendekatan-pendekatan baru dalam membahas masalah yang dihadapi suatu kelompok dalam menyelesaikan suatu bidang masalah dengan fenomena tertentu.

2.2.8. Tujuan Sistem Penunjang Keputusan

(Mayana, Tarigan and Yunita, 2021) berpendapat bahwa sistem penunjang keputusan tidak ditujukan untuk pengambilan keputusan secara otomatis, namun memberikan perangkat interaktif yang memungkinkan pengambil keputusan agar dapat melakukan berbagai analisis menggunakan model-model yang tersedia. Berikut ini adalah tujuan dari sistem penunjang keputusan :

1. Membantu manajer dalam proses pengambilan keputusan terhadap masalah semi terstruktur
2. Memberikan dukungan pada pertimbangan keputusan manajer dan bukan digunakan untuk menggantikan peran dari manajer.
3. Meningkatkan efektifitas keputusan yang dibuat manajer lebih daripada perbaikan efisiensinya.

4. Memungkinkan proses komputasi menggunakan komputer berlangsung dengan cepat. Hal ini menguntungkan pengambil keputusan untuk melakukan banyak proses komputasi secara cepat dan dengan biaya yang rendah.
5. Meningkatkan produktivitas. Membangun sebuah kelompok pengambil keputusan yang berisikan para pakar dapat memakan biaya yang tinggi.
6. Dukungan terhadap kualitas. Proses pengambilan keputusan dengan komputer dapat meningkatkan kualitas keputusan yang dibuat.
7. Meningkatkan daya saing manajemen serta pemberdayaan sumber daya perusahaan.
8. Mengatasi keterbatasan kognitif dalam pemrosesan serta penyimpanan.

2.2.9. Komponen Sistem Penunjang Keputusan

Terdapat beberapa komponen yang membentuk suatu Sistem Penunjang Keputusan. Berikut ini merupakan 4 komponen Sistem Penunjang Keputusan menurut (Mayana, Tarigan and Yunita, 2021) :

1. **Subsistem Manajemen Data**

Subsistem manajemen data adalah komponen sistem penunjang keputusan yang menyediakan data bagi sistem. Data yang dimaksud disimpan dalam *database* yang dikelola oleh suatu sistem dengan sistem manajemen basis data (*Database Management System / DBMS*). Melalui basis data inilah data dapat diambil dan diekstraksi secara cepat.

2. **Subsistem Manajemen Model**

Keunikan dari sistem penunjang keputusan adalah kemampuannya dalam mengintegrasikan data terhadap model-model keputusan. Model merupakan

peniruan variabel dari alam nyata. Hal lain yang harus diperhatikan yaitu pada setiap model yang disimpan seharusnya ditambahkan rincian keterangan serta penjelasan yang komprehensif terkait model yang dibuat.

3. Subsistem Dialog (*User System Interface*)

Keunikan lain dari sistem penunjang keputusan yaitu adanya fasilitas yang dapat mengintegrasikan sistem terpasang dengan *user* secara interaktif. Melalui subsistem inilah sistem diartikulasikan serta diimplementasikan sehingga *user* dapat berinteraksi dengan sistem yang dibuat.

2.2.10. Keuntungan Sistem Penunjang Keputusan

Berikut ini merupakan keuntungan yang didapat dari penggunaan Sistem Penunjang Keputusan (SPK) menurut (Utama, 2017) :

1. SPK dapat mendukung pengambil keputusan dalam pencarian solusi dari berbagai masalah yang kompleks.
2. SPK mampu merespon situasi yang tidak diharapkan dengan cepat dalam kondisi yang berubah-ubah.
3. SPK mampu menerapkan berbagai macam strategi yang berbeda kepada konfigurasi yang berbeda pula secara tepat dan cepat.
4. Sebagai pandangan dan pembelajaran baru.
5. Sebagai penyedia fasilitas dalam komunikasi.
6. SPK dapat meningkatkan kontrol terhadap manajemen dan kinerja.
7. SPK dapat menghemat biaya serta sumber daya manusia.
8. SPK dapat menghemat waktu karena keputusan bisa diambil dalam waktu yang singkat.

9. SPK dapat meningkatkan efektivitas manajemen sehingga manajer dapat bekerja lebih singkat dan lebih sedikit usaha.
10. SPK dapat meningkatkan produktivitas analisis.

2.3. Metode Simple Additive Weighting (SAW)

Metode *Simple Additive Weighting* (SAW) sering diartikan dengan metode penjumlahan terbobot. Konsep dasar dari metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada tiap alternatif pada semua atribut. Metode *Simple Additive Weighting* (SAW) memerlukan proses normalisasi matriks keputusan (X) ke suatu skala yang bisa dibandingkan dengan semua rating alternatif yang tersedia (Mayana, Tarigan and Yunita, 2021). Formula untuk melakukan normalisasi tersebut adalah sebagai berikut :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max } x_{ij}} & \text{Jika } j \text{ merupakan atribut keuntungan (benefit)} \\ \frac{\text{Min } x_{ij}}{x_{ij}} & \text{Jika } j \text{ merupakan atribut biaya (cost)} \end{cases}$$

Keterangan :

r_{ij} = rating kinerja ternormalisasi

$\text{Max } x_{ij}$ = nilai terbesar dari tiap kriteria i

$\text{Min } x_{ij}$ = nilai terkecil dari tiap kriteria i

X_{ij} = nilai atribut yang dimiliki dari tiap kriteria

Benefit = jika nilai terbesar adalah nilai terbaik

Cost = jika nilai terkecil adalah nilai terbaik

Nilai preferensi bagi setiap alternatif (V_i) ditentukan sebagai berikut :

$$V_i = \sum_{j=1}^n w_j r_{ij}$$

Keterangan :

V_i = ranking dari setiap alternatif

w_j = Bobot yang ditentukan dari tiap kriteria

r_{ij} = Nilai rating kinerja ternormalisasi

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih

2.3.1. Langkah Penyelesaian SAW

Menurut (Mayana, Tarigan and Yunita, 2021) langkah penyelesaian masalah dalam metode SAW adalah sebagai berikut :

1. Menentukan kriteria apa saja yang akan dijadikan acuan pada pengambilan keputusan, yaitu C_i .
2. Menentukan nilai bobot terhadap masing-masing kriteria sebagai w .
3. Memberikan nilai rating kecocokan setiap alternatif kepada tiap kriteria.
4. Membuat matriks keputusan berdasarkan pada kriteria (C_i), kemudian melakukan proses normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis tiap atribut (*benefit* atau *cost*) hingga diperoleh matriks ternormalisasi R

5. Hasil akhir didapatkan dari proses perankingan, yaitu penjumlahan serta perkalian matriks ternormalisasi R dengan vektor bobot hingga diperoleh hasil berupa nilai terbesar yang akan dipilih sebagai alternatif terbaik A_i sebagai solusi.

2.4. Tenant

Menurut (Yunita, 2016) tenant adalah orang yang mengambil atau sudah mengambil barang-barang dari pemilik dalam sebuah perjanjian sewa beli untuk pemakaian sementara suatu benda, baik berupa benda bergerak maupun tidak bergerak dengan pembayaran dengan harga tertentu yang menyewa barang ataupun benda dari pihak yang menyewakan.

Jadi, tenant merupakan orang atau badan yang menyewa, memiliki, ataupun pengguna benda atau barang yang didapatkan dari pihak yang menyewakan.

2.5. Inkubator Bisnis

2.5.1. Inkubator

(Darmawan, 2019) berpendapat bahwa inkubator merupakan alat untuk mengumpulkan serta mengatur kekuatan yang ada untuk memberikan fasilitas pada penciptaan perusahaan.

Sedangkan menurut (Andi Nur, 2018) inkubator merupakan sebuah bangunan fisik yang dipergunakan untuk mendukung bisnis yang berkualifikasi melalui program mentoring, pelatihan, jejaring profesi, dan bantuan pencarian pendanaan hingga mereka lulus dan bisa bertahan dalam lingkungan bisnis.

2.5.2. Bisnis

Menurut (Purwanto, 2020) bisnis adalah usaha perdagangan dengan menjual barang atau jasa kepada konsumen dalam rangka mencapai tujuan untuk mendapatkan keuntungan.

Sedangkan menurut (Safitri and Fajrin, 2019) bisnis adalah organisasi yang menyediakan barang ataupun jasa untuk dijual dengan tujuan mendapatkan keuntungan.

Dari beberapa pengertian tersebut, dapat disimpulkan bahwa bisnis merupakan upaya individu ataupun kelompok untuk menyediakan barang ataupun jasa untuk mendapatkan sebuah keuntungan.

2.5.3. Inkubator Bisnis

Menurut (Hakim, 2020) inkubator bisnis merupakan sebuah lembaga yang menyediakan inkubasi kepada calon wirausaha atau wirausaha baru agar bisa menjalankan usahanya dengan sukses.

Menurut (Hausberg and Korreck, 2020) inkubator bisnis merupakan organisasi yang menyelenggarakan inkubasi bisnis dengan sumber daya berwujud dan tak berwujud. Sumber daya berwujud yang dimaksud contohnya ruangan, peralatan bersama, dan layanan administrasi. Sedangkan yang dimaksud sumber daya tak berwujud contohnya adalah pengetahuan, akses, dan jejaring. Proses inkubasi ini dijalankan selama periode tertentu dan dibiayai oleh sponsor maupun dana mandiri yang didapat dari biaya sewa peserta inkubasi.

Menurut (Lutfiani, Rahardja and Manik, 2020) inkubator bisnis merupakan lembaga intermediasi yang bertujuan untuk membantu tumbuh kembang wirausaha pemula berbasis teknologi dalam kurun waktu tertentu.

Dari pendapat di atas, dapat disimpulkan bahwa inkubator bisnis merupakan suatu lembaga atau organisasi yang menyediakan inkubasi dalam suatu tempat atau ruangan kepada calon wirausaha dan wirausaha baru dalam kurun waktu tertentu untuk mencapai kesuksesan.

2.5.4. Jenis Inkubator Bisnis

Menurut (Atmaja, 2022) inkubator bisnis dibagi menjadi beberapa jenis, yaitu :

a. *Industrial Incubator*

Merupakan inkubator yang didukung oleh pemerintah serta lembaga *non-profit*. Tujuannya adalah untuk menciptakan lapangan pekerjaan untuk mengurangi tingkat pengangguran.

b. *University-related incubator*

Inkubator ini bertujuan untuk komersialisasi sains, teknologi, dan HAKI dari hasil penelitian. Inkubator perguruan tinggi memberikan perusahaan pemula atau *start-up* untuk mendapatkan fasilitas layanan laboratorium, komputer, perpustakaan, dan jasa kepakaran dari perguruan tinggi. Inkubator ini didukung secara langsung oleh perguruan tinggi dan juga bekerjasama dengan pihak lain yang memiliki perhatian.

c. *For-profit property development incubators*

Merupakan inkubator yang menyediakan perkantoran, tempat produksi, dan fasilitas jasa secara bersama. Beberapa fasilitas perkantoran yang

mendukung citra perusahaan digunakan secara bersama-sama dan inkubator menerapkan biaya sewa atas penggunaan fasilitas tersebut.

d. *For-Profit investment incubator*

Adalah inkubator yang menyerupai perusahaan modal ventura dan *business angel*, yang menggunakan kantor yang sama dengan yang digunakan oleh tenant (perusahaan) yang dibiayainya. Inkubator ini memiliki perhatian yang sangat ketat terhadap portofolio tenant.

e. *Corporate Venture incubator*

Merupakan model inkubator yang paling sukses dan perkembangannya yang tercepat. Perusahaan yang telah mapan mendirikan inkubator yang ditujukan untuk mengambil alih perusahaan kecil dan memberikan bantuan keuangan, keahlian, dan bahkan pasar.

2.6. **Database (Basis Data)**

Menurut (dalleh, Akrim and Baharuddin, 2020) basis data merupakan kumpulan data yang tersimpan di dalam komputer serta dapat diperiksa, diolah, maupun dimanipulasi secara sistematis menggunakan suatu program komputer.

(Patni *et al.*, 2022) berpendapat bahwa *database* merupakan sekumpulan informasi yang terorganisir dan saling berhubungan. Semua data yang tersimpan pada basis data saling terhubung satu sama lain.

Dari definisi di atas, dapat disimpulkan bahwa *database* atau basis data merupakan sekumpulan file yang dapat dikelola oleh pengguna. Data ini saling terhubung sehingga membentuk suatu data yang saling terkait yang tersimpan dalam suatu media penyimpanan.

2.6.1. Database Management System (DBMS)

Menurut (Rahmawati and Sumarno, 2020) sistem manajemen basis data (*database management system*) atau yang biasa disingkat DBMS merupakan suatu sistem atau perangkat lunak yang dibuat untuk mengelola suatu *database* dan menjalankan operasi terhadap data yang diminta oleh *user*.

Sedangkan menurut (Patni *et al.*, 2022) DBMS merupakan sebuah paket perangkat lunak yang mengelola data yang tersimpan didalam sebuah *database*.

Dari pendapat di atas, dapat disimpulkan bahwa DBMS merupakan sebuah sistem perangkat lunak yang digunakan untuk mengelola data-data yang tersimpan dalam suatu basis data.

2.6.2. MySQL

MySQL adalah sistem manajemen basis data yang paling populer digunakan untuk web *server*. *MySQL* disebut juga dengan sistem manajemen basis data relasional (RDBMS) karena tabel nya bukan hanya berisikan data, namun juga menyimpan relasi antar data (Nixon, 2018).

MySQL adalah sistem *database server* yang sangat terkenal. *MySQL* menggunakan bahasa SQL untuk mengakses data didalam database nya (Ramadhan and Mukhaiyar, 2020).

2.7. HTML

HTML adalah singkatan dari *Hypertext Markup Language* yang merupakan bahasa standar dalam web yang dikelola penggunaanya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun tiap elemen-elemen dari sebuah

website. HTML berperan sebagai penyusun struktur kerangka halaman *website* yang menempatkan setiap elemen *website* sesuai dengan layout yang diinginkan (Astutik and Rosid, 2019).

Menurut (Robbins, 2018) HTML bukanlah bahasa pemrograman, tetapi HTML adalah bahasa *markup* yang berarti HTML adalah sebuah sistem yang digunakan untuk mengidentifikasi dan mendefinisikan berbagai komponen dari sebuah dokumen, seperti *heading*, paragraf, dan *list*.

Dari pendapat tersebut, dapat disimpulkan bahwa HTML merupakan suatu bahasa dalam pemrograman web yang digunakan untuk menampilkan dan mengatur suatu halaman pada *web browser*.

2.8. PHP

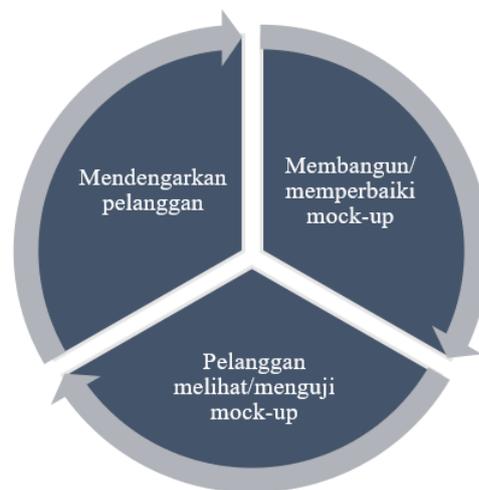
PHP atau *Hypertext Preprocessor* merupakan suatu bahasa *script* berbasis *server* (*server-side*) yang dapat mem-parsing kode php dari kode web dengan ekstensi file .php, sehingga menghasilkan tampilan website yang dinamis di sisi *client* (browser) (Sari, Abdilah and Sunarti, 2019).

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang berjalan di sisi server yang digunakan secara luas untuk penanganan, pembuatan, serta pengembangan sebuah situs web dan dapat digunakan bersamaan dengan HTML (Astutik and Rosid, 2019).

2.9. Model Prototype

Menurut (Findawati, 2018) *prototyping* merupakan proses merancang suatu *prototype* dimana definisi *prototype* merupakan sebuah model dari suatu produk

yang mungkin belum memiliki semua fitur produk yang sesungguhnya tetapi sudah memiliki fitur-fitur utama dari produk yang sesungguhnya serta dapat digunakan untuk keperluan pengujian atau uji coba sebelum berlanjut ke fase pembuatan produk yang sesungguhnya. Dengan menggunakan model *prototype* ini, pengembang dan juga pelanggan dapat saling berinteraksi selama proses pembuatan suatu produk. Berikut ini ilustrasi dari model prototipe ditunjukkan pada gambar 2.1 :



Gambar 2.1 Ilustrasi model prototype
(Findawati, 2018)

Tahapan pengembangan model *prototype* yaitu :

1. Mendengarkan pelanggan, pada tahap ini dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengarkan keluhan-keluhan dari pelanggan. Untuk membuat sebuah sistem yang sesuai dengan kebutuhan, maka hal yang harus diketahui terlebih dahulu adalah bagaimana sistem sedang berjalan untuk kemudian mengetahui masalah yang terjadi.
2. Merancang serta membuat *prototype*, pada tahap ini dilakukan proses perancangan serta pembuatan *prototype* sistem. *Prototype* yang akan dibuat

disesuaikan dengan kebutuhan sistem yang telah ditentukan sebelumnya dari keluhan pelanggan/pengguna.

3. Uji coba, pada tahap ini *prototype* sistem yang telah dibuat di uji coba oleh pelanggan, kemudian dilakukan proses evaluasi kekurangan-kekurangan dari kebutuhan pelanggan. Pengembangan kemudian kembali mendengarkan keluhan dari pelanggan untuk perbaikan *prototype* yang ada.

2.10. Unified Modeling Language (UML)

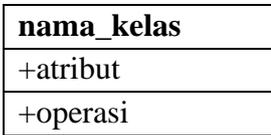
Menurut (Rosa and Shalahuddin, 2018) UML merupakan suatu bahasa visual yang digunakan untuk melakukan pemodelan dan komunikasi mengenai suatu sistem dengan menggabungkan diagram serta teks-teks pendukung. UML muncul sebagai jawaban atas kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari suatu sistem perangkat lunak.

2.10.1. Class Diagram

Menurut (Rosa and Shalahuddin, 2018) *class diagram* atau diagram kelas menggambarkan struktur dari sebuah sistem dari segi pendefinisian kelas kelas yang nantinya akan dibuat untuk membangun sebuah sistem. Sebuah kelas memiliki apa yang disebut dengan atribut dan metode atau operasi. Atribut adalah variabel variabel yang dimiliki oleh suatu kelas, sedangkan metode atau operasi merupakan fungsi-fungsi yang dimiliki oleh suatu kelas.

Berikut ini adalah simbol-simbol yang digunakan dalam *class diagram* terlihat pada tabel 2.2 :

Tabel 2.2 Simbol Class Diagram (Rosa and Shalahuddin, 2018)

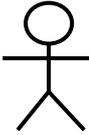
Simbol	Deskripsi
kelas 	Kelas pada struktur sistem
antarmuka / <i>Interface</i> 	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek
asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
generalisasi 	Relasi antarkelas dengan makna generalisasi-spesifikasi (umum khusus)
kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas
agregasi / <i>agregation</i> 	Relasi antarkelas dengan makna semua-bagian (whole-part)

2.10.2. Use Case Diagram

Use case diagram atau diagram usecase adalah pemodelan untuk tingkah laku (*behavior*) dari sistem informasi yang akan dibuat. *Use case* mendeskripsikan suatu interaksi antara satu atau lebih aktor terhadap sistem informasi yang akan dibangun (Rosa and Shalahuddin, 2018).

Simbol-simbol yang digunakan pada *use case diagram* ditunjukkan pada tabel 2.3 berikut :

Tabel 2.3 Simbol Use Case Diagram (Rosa and Shalahuddin, 2018)

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal fase nama <i>use case</i>.</p>
<p>Aktor / actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi meskipun simbol dari aktor adalah orang, namun belum tentu aktor merupakan orang, biasanya dinyatakan menggunakan kata benda di awal fase nama aktor.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan use case yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Ekstensi / <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 walaupun tanpa usecase tambahan itu, mirip dengan</p>

Tabel 2.3 Simbol Use Case Diagram (Lanjutan)

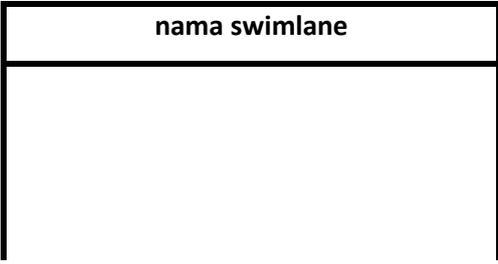
Simbol	Deskripsi
	konsep <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.
Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara 2 buah use case dimana fungsi yang satu merupakan fungsi yang lebih umum daripada yang lainnya.
Menggunakan / <i>include</i> / <i>uses</i> << <i>include</i> >>  << <i>uses</i> >> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan usecase ini.

2.10.3. Activity Diagram

(Rosa and Shalahuddin, 2018) menjelaskan bahwa *activity diagram* atau diagram aktivitas menggambarkan suatu *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau suatu proses bisnis atau menu yang ada pada perangkat lunak. Aktivitas yang dimaksud dalam *activity diagram* bukanlah aktivitas yang dilakukan oleh aktor, melainkan aktivitas yang dilakukan oleh sistem.

Simbol-simbol yang digunakan pada *activity diagram* akan ditampilkan pada tabel 2.4 berikut :

Tabel 2.4 Simbol Activity Diagram (Rosa and Shalahuddin, 2018)

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal.
Aktivitas 	Aktivitas yang dilakukan oleh sistem, aktivitas biasanya selalu diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada aktivitas dengan pilihan lebih dari satu
Penggabungan / <i>join</i> 	Aktivitas penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan oleh sistem, suatu diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.11. Pengujian *Black Box*

Black box testing (pengujian kotak hitam) merupakan salah satu metode pengujian perangkat lunak dimana proses pengujian nya hanya menguji fungsional tanpa melakukan pengujian desain dan kode program. Pengujian ini dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sudah sesuai dengan apa yang dibutuhkan. Pengujian *black box* ini dilakukan dengan membuat sebuah kasus uji yang dapat mencoba semua fungsi dari perangkat

lunak yang dibuat, apakah sesuai dengan spesifikasi yang dibutuhkan ataupun tidak. Kasus yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan 2 jenis kasus, yaitu kasus benar dan kasus salah (Rosa and Shalahuddin, 2018).

Adapun kerangka yang akan digunakan dalam pengujian ditunjukkan pada tabel 2.5 dibawah :

Tabel 2.5 Kerangka Pengujian

Kasus dan Hasil Pengujian			
Data Masukan	Data Diharapkan	Pengamatan	Kesimpulan
			Diterima [] Ditolak []
			Diterima [] Ditolak []

Perhitungan dari hasil kuesioner *black box testing* menggunakan analisa deskriptif ditunjukkan pada persamaan berikut :

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

Keterangan :

Skor Aktual : Jawaban diterima responden

Skor Ideal : Total jumlah soal yang telah diujikan kepada responden

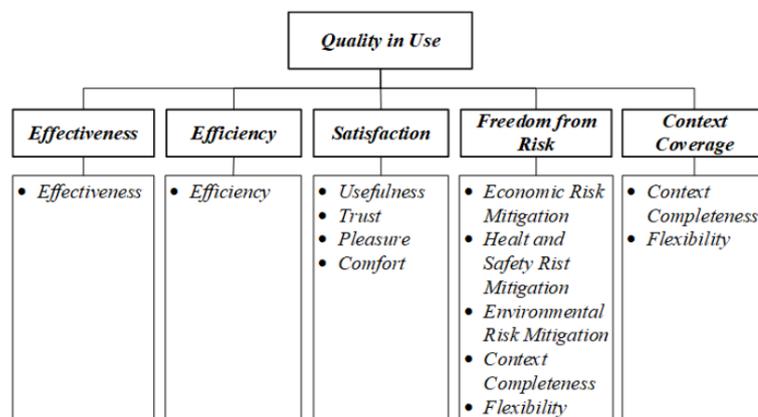
2.12. ISO/IEC 25010

ISO/IEC 25010 adalah panduan yang diterbitkan oleh *Canadian Standards Association* pada tahun 2011 dan digunakan dalam evaluasi perangkat lunak. ISO/IEC 25010 adalah model baru dalam seri ISO/IEC 250n, yang merupakan

pengembangan dari versi ISO/IEC 9126. model 25010 memiliki karakteristik kualitas yang dapat dipertimbangkan saat mengevaluasi perangkat lunak. Dalam ISO/IEC 25010 terdapat dua model yang dapat digunakan untuk mengukur kualitas sistem, yaitu *quality in use model*, dan *software product quality model* (Mulyawan *et al.*, 2021). Adapun yang dimaksud *quality in use model*, dan *software product quality model* :

1. *Quality in use model*

Quality in use model adalah model yang memperkirakan seberapa baik pengguna dapat menggunakan perangkat lunak untuk memenuhi kebutuhan bisnis tertentu dalam konteks penggunaan tertentu. *Quality in use model* terdiri dari lima karakteristik kualitas seperti yang ditunjukkan pada gambar 2.2.



Gambar 2.2 Karakteristik Quality in Use Model
(Mulyawan *et al.*, 2021)

Gambar 2.2 menunjukkan karakteristik *quality in use model* yang terdiri dari: *Effectiveness* adalah karakteristik untuk mengukur seberapa akurat dan lengkap pengguna dalam mencapai tujuan yang ditetapkan.

Efficiency adalah karakteristik yang mengukur sumber daya yang digunakan terkait dengan akurasi serta kelengkapan yang digunakan pengguna dalam mencapai tujuan.

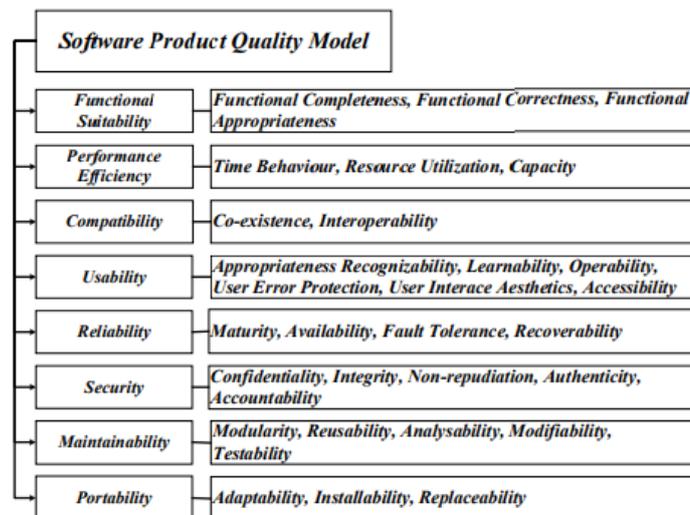
Satisfaction merupakan karakteristik untuk mengukur seberapa jauh kebutuhan pengguna dapat terpenuhi saat suatu produk atau sistem digunakan didalam konteks penggunaan yang ditentukan.

Freedom from Risk merupakan karakteristik yang digunakan untuk mengukur seberapa jauh suatu sistem dapat meminimalisir risiko potensial terhadap status ekonomi, kehidupan manusia, kesehatan, atau lingkungan.

Context Completeness merupakan karakteristik yang digunakan untuk mengukur seberapa jauh suatu sistem digunakan dengan efektif, efisien, kebebasan dari risiko dan kepuasan dalam seluruh konteks penggunaa yang ditentukan.

2. *Software product quality model*

Software product quality model merupakan sebuah model yang hanya dapat diterapkan pada produk perangkat lunak dikarenakan sebagian besar sub karakteristiknya terkait dengan perangkat lunak dan sistem. *Software product quality model* terdiri dari delapan karakteristik kualitas seperti yang ditunjukkan pada gambar 2.3.



Gambar 2.3 Karakteristik Software Product Quality Model (Mulyawan et al., 2021)

Gambar 2.3 menunjukkan karakteristik serta sub karakteristik *software product quality model* yang terdiri dari :

Functional Suitability merupakan karakteristik yang mengukur seberapa jauh produk atau sistem menyediakan fungsi yang dapat memenuhi kebutuhan saat digunakan dalam kondisi tertentu.

Performance Efficiency merupakan karakteristik yang digunakan untuk mengukur kinerja relatif terhadap sumber daya yang dipakai dalam suatu kondisi didalam suatu sistem.

Compability merupakan karakteristik yang mengukur seberapa jauh suatu sistem dapat melakukan pertukaran informasi dengan sistem lain dan melakukan fungsi yang disyaratkan ketika berbagi lingkungan perangkat keras ataupun perangkat lunak yang sama.

Usability merupakan karakteristik yang mengukur seberapa jauh sistem bisa digunakan oleh pengguna dalam mencapai tujuan yang ditentukan dengan efektivitas, efisiensi, serta kepuasan dalam konteks penggunaan tertentu.

Reliability merupakan karakteristik yang mengukur seberapa jauh sistem dapat melakukan fungsi dalam kondisi yang ditentukan selama periode waktu tertentu.

Security merupakan karakteristik yang mengukur suatu sistem dalam melakukan perlindungan terhadap informasi dan data, sehingga sistem memiliki tingkatan akses data yang sesuai dengan jenis dan tingkat otorisasi.

Maintainability merupakan karakteristik untuk mewakili tingkat efektivitas serta efisiensi dalam melakukan modifikasi untuk perbaikan suatu sistem sesuai dengan perubahan pada lingkungan operasional.

Portability merupakan karakteristik untuk mewakili tingkat efektivitas dan efisiensi suatu sistem dalam melakukan proses transfer dari satu perangkat ke perangkat yang lain.