

BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka adalah sebuah penelitian sebelumnya yang dapat digunakan sebagai pendukung penelitian yang sedang dilakukan sekarang. Peneliti telah mengumpulkan beberapa tinjauan pustaka yang dapat dilihat pada tabel 2.1 yang ada dibawah.

Tabel 2.1 Studi Literatur

Nomor	Detail Jurnal	
1	Judul	Sistem Pendukung Keputusan Penerimaan Bantuan Program Keluarga Harapan Menggunakan Metode AHP
	Tahun Terbit	2024
	Penulis	Alfian Kurniawan, Naely Farkhatin, Muslihatul Hidayah
	Metode	<i>Analytical Hierarchy Process</i> (AHP)
	Hasil	Hasil dari penelitian ini yaitu pengolahan data penerima bantuan tidak lagi manual dan menjadi terkomputerisasi serta proses pengumpulan data yang berhak sebagai penerima bantuan lebih mudah dan cepat.
	Perbedaan: Pada penelitian (Alfian Kurniawan, Naely Farkhatin & Muslihatul Hidayah, 2024) melakukan pengambilan keputusan bantuan PKH menggunakan kriteria pekerjaan, penghasilan, nilai survey tempat tinggal dan domisili, sedangkan penulis melakukan penelitian untuk pengambilan keputusan bantuan dana desa menggunakan kriteria kehilangan mata pencaharian, mempunyai anggota keluarga yang	

	rentan sakit menahun/kronis dan/atau difabel, rumah tangga dengan anggota rumah tangga tunggal lanjut usia, perempuan kepala keluarga dari keluarga miskin dan tidak menerima bantuan sosial program keluarga harapan serta menggunakan metode pengembangan sistem <i>Extreme Programming</i> .	
2	Judul	Sistem Pendukung Keputusan Pemberian Dana Bantuan Menggunakan Metode AHP
	Tahun Terbit	2020
	Penulis	Heni Ayu Septilia, Styawati
	Metode	<i>Analytical Hierarchy Process (AHP)</i>
	Hasil	Hasil dari penelitian ini adalah Sistem Pendukung Keputusan untuk pemberian dana bagi masyarakat kurang mampu menggunakan metode Analytical Hierarchy Process (AHP) yang dapat mendukung dalam penentuan bantuan untuk masyarakat.
	Perbedaan: Pada penelitian (Heni Ayu Septilia & Styawati, 2020) melakukan pengambilan keputusan pemberian dana bantuan PKH dengan kriteria pendidikan, pekerjaan, penghasilan, status, umur, tempat tinggal, kesehatan, dan jumlah anak.	
3	Judul	Sistem Pendukung Keputusan Penerimaan BLT Di Kecamatan Sampang Menggunakan Metode Saw Dan Metode AHP Berbasis Web
	Tahun Terbit	2021
	Penulis	Rehadian Teja Sukmana Putra, Suryo Adi Wibowo, Yosep Agus Pranoto
	Metode	<i>Simple Additive Weighting(SAW)</i> dan <i>Analytical Hierarchy Process (AHP)</i>

	Hasil	Hasil dari penelitian tersebut adalah Sistem Pendukung Keputusan penerimaan bantuan langsung tunai dengan menggunakan metode <i>Simple Additive Weighting</i> (SAW) dan <i>Analythic Hierarchy Process</i> (AHP) digunakan untuk mendukung pengambilan keputusan penerimaan bantuan langsung tunai terutama dalam proses perangkingan berdasarkan kriteria – kriteria telah ditentukan sehingga dapat memberikan rekomendasi evaluasi pemilihan penerimaan bantuan langsung tunai yang lebih objektif karena dapat dilakukan pembobotan terhadap kriteria yang telah di tentukan.
	Perbedaan: Pada penelitian (Rehadian Teja Sukmana Putra, Suryo Adi Wibowo, Yosep Agus Pranoto, 2021) melakukan pengambilan keputusan BLT menggunakan kriteria pekerjaan, penghasilan per-bulan, tanggungan keluarga, kepemilikan aset dan tempat tinggal.	
4	Judul	Sistem Pendukung Keputusan Penerima Bantuan Covid-19 menggunakan Metode Simple Additive Weighting (SAW)
	Tahun Terbit	2020
	Penulis	Falentino Sembiring, Mohamad Tegar Fauzi, Siti Khalifah, Ana Khusnul Khotimah, Yayatillah Rubiati
	Metode	<i>Simple Additive Weighting</i> (SAW)
	Hasil	Hasil dari penelitian tersebut adalah sebuah sistem menggunakan metode Simple Additive Weighting (SAW)

		dengan kriteria-kriteria yang sudah ditetapkan, hasil dari nilai pranking yang terbesar yang berhak mendapat bantuan sosial tersebut.
	<p>Perbedaan:</p> <p>Pada penelitian (Falentino Sembiring, Mohamad Tegar Fauzi, Siti Khalifah, Ana Khusnul Khotimah dan Yayatillah Rubiati, 2020) melakukan pengambilan keputusan bantuan covid-19 dengan metode SAW, sedangkan penulis melakukan sistem pendukung keputusan BLT-DD dengan metode AHP menggunakan pengembangan sistem <i>Extreme Programming</i>.</p>	
5	Judul	Sistem Pendukung Keputusan Penerima Bantuan Bedah Menggunakan Metode AHP di Desa Sukoharjo
	Tahun Terbit	2022
	Penulis	Reyno Yusuf Zakaria, Ardi Sanjaya, Danang Wahyu Widodo
	Metode	Analytical Hierarchy Process (AHP)
	Hasil	Hasil dari penelitian tersebut adalah sebuah sistem dengan menggunakan metode AHP untuk penentuan calon penerima bantuan bedah rumah di desa Sukoharjo.
	<p>Perbedaan:</p> <p>Pada penelitian (Reyno Yusuf Zakaria, Ardi Sanjaya, dan Danang Wahyu Widodo, 2022) melakukan pengambilan keputusan bantuan bedah rumah dengan kriteria pekerjaan, penghasilan, tanggungan, kondisi rumah dan aset, sedangkan penulis melakukan sistem pendukung keputusan BLT-DD dengan menggunakan pengembangan sistem <i>Extreme Programming</i>.</p>	

2.2 Sistem Pendukung Keputusan

Menurut Prayetno dkk. (2018) Konsep Sistem Pendukung Keputusan pertama kali diperkenalkan pada awal tahun 1970-an oleh Michael S. Scott Morton dengan istilah *Management Decision System*. Konsep pendukung keputusan ditandai dengan sistem interaktif berbasis komputer yang membantu pengambil keputusan memanfaatkan data dan model untuk menyelesaikan masalah-masalah yang tidak terstruktur (Prayetno *et al.*, 2018).

Menurut Najirah Umar (2023) Sistem Pendukung Keputusan adalah alat terkomputerisasi yang membantu pengambil keputusan dalam memecahkan masalah tanpa aturan atau batasan yang sudah ada sebelumnya. Sistem ini menggunakan algoritma dan data untuk menemukan solusi atas masalah yang kompleks.

Berdasarkan definisi tersebut, dapat ditarik kesimpulan bahwa sistem pendukung keputusan merupakan proses aktivitas yang dilaksanakan oleh individu atau kelompok untuk mengambil keputusan.

2.3 BLT-Dana Desa (Bantuan Langsung Tunai Dana Desa)

Bantuan Langsung Tunai atau disingkat BLT adalah program bantuan pemerintah yang berupa pemberian uang tunai atau berbagai bantuan lainnya kepada masyarakat miskin, baik yang bersyarat (*conditional cash transfers*) maupun yang tidak bersyarat (*unconditional cash transfers*) (Paramita *et al.*, 2022).

Dalam Peraturan Menteri Keuangan disebutkan bahwa Bantuan Langsung Tunai Dana Desa (BLT-DD) Nomor 146/PMK/2023 merupakan bentuk dukungan kepada masyarakat berpenghasilan rendah yang dibiayai dari alokasi dana desa.

Bantuan ini termasuk dalam kategori perlindungan dan jaminan sosial yang diberikan oleh pemerintah.

BLT-Dana Desa merupakan salah satu bentuk bantuan sosial yang diberikan oleh pemerintah desa kepada keluarga miskin dan tidak mampu yang tinggal di desa yang belum menerima bantuan sosial lainnya seperti bantuan tunai, program keluarga harapan, atau bantuan pangan non tunai. BLT-DD memiliki beberapa kriteria, antara lain kehilangan mata pencaharian, mempunyai anggota keluarga yang rentan sakit menahun/kronis dan/atau difabel, rumah tangga dengan anggota rumah tangga tunggal lanjut usia, perempuan kepala keluarga dari keluarga miskin dan tidak menerima bantuan sosial program keluarga harapan (PMK No 146/PMK/2023)

2.4 AHP (*Analytical Hierarchy Process*)

Menurut Fadli dan Imtihan (2019) yang mengutip buku Saaty (2008) *Analytical Hierarchy Process* (AHP) merupakan suatu metode pendekatan yang sesuai untuk menangani sistem yang kompleks yang berhubungan dengan penentuan keputusan dari beberapa alternatif dan memberikan pilihan yang dapat dipertimbangkan. Metode ini dikembangkan pertama kali oleh Saaty. Model hierarki yang dinyatakan oleh Saaty adalah model hierarki fungsional dengan input utamanya adalah persepsi manusia.

Dalam menyelesaikan permasalahan dengan metode AHP ada beberapa prinsip yang harus dipahami, diantaranya adalah (Sofiansyah & Khairul, 2019) :

1. Dekomposisi (*Decomposition*)

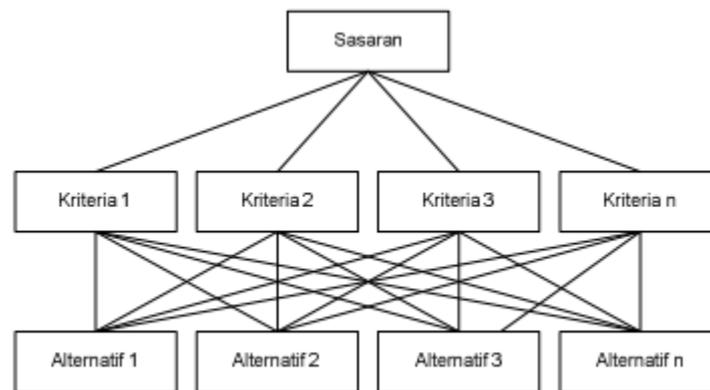
Sistem yang kompleks bisa dihadapi dengan memecahkannya menjadi elemen-elemen pendukung, menyusun elemen menjadi hierarki dan

menggabungkannya. Struktur penyusunan hierarki dapat dilihat pada Gambar 2.1 berikut.

Tingkat pertama : Sasaran Keputusan (Goal)

Tingkat kedua : Kriteria – kriteria

Tingkat Tiga : Alternatif – alternatif



Gambar 2.1 Struktur Hirarki AHP

2. Penilaian Komparatif (*Comparative judgment*)

Kriteria dan alternatif dilakukan dengan perbandingan berpasangan menurut saaty (1988), untuk berbagai persoalan, skala 1 sampai 9 adalah skala terbaik untuk mengekspresikan pendapat, nilai dan definisi pendapat kualitatif dari skala perbandingan bisa diukur dengan menggunakan tabel analisis seperti ditunjukkan pada tabel 2.2:

Tabel 2.2 Skala penilaian dalam metode AHP

Intensitas Kepentingan	Definisi	Keterangan
1	Sama penting	Kedua elemen sama pentingnya
3	Sedikit lebih penting	Elemen yang satu sedikit lebih penting dari pada elemen yang lainnya

5	Lebih penting	Elemen yang satu lebih penting daripada elemen yang lainnya
7	Sangat penting	Satu elemen jelas lebih mutlak penting daripada elemen lainnya
9	Mutlak lebih penting	Satu elemen mutlak penting daripada elemen lainnya
2,4,6,8	Nilai tengah	Nilai-nilai antara dua nilai pertimbangan yang berdekatan

3. Menentukan prioritas (*Synthesis of priority*)

Untuk setiap kriteria dan alternatif, perlu dilakukan perbandingan berpasangan. Nilai-nilai perbandingan relatif dari seluruh alternatif kriteria bisa disesuaikan dengan judgment yang telah ditentukan untuk menghasilkan bobot prioritas. Bobot prioritas dihitung dengan manipulasi matriks atau melalui penyelesaian persamaan matematika.

4. Konsistensi logis (*Logical consistensi*)

Konsistensi memiliki dua makna. Pertama, objek-objek yang serupa bisa dikelompokkan sesuai dengan keseragaman dan relevansi. Kedua, menyangkut hubungan antar objek yang didasarkan pada kriteria tertentu.

2.4.1 Proses AHP (*Analytical Hierarchy Process*)

Secara umum langkah-langkah dalam menggunakan metode AHP untuk pemecahan suatu masalah adalah sebagai berikut (Sugianto et al., 2016).

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki dari permasalahan yang dihadapi.
2. Menentukan prioritas elemen

Langkah pertama dalam menentukan prioritas elemen adalah membuat perbandingan pasangan, yaitu membandingkan elemen secara berpasangan sesuai kriteria yang diberikan.

3. Matriks perbandingan berpasangan diisi menggunakan bilangan untuk merepresentasikan kepentingan relatif dari suatu elemen terhadap elemen yang lainnya.

4. Sintesis

Pertimbangan-pertimbangan terhadap perbandingan berpasangan disintesis untuk memperoleh keseluruhan prioritas. Hal-hal yang dilakukan dalam langkah ini adalah:

- a. Menjumlahkan nilai-nilai dari setiap kolom pada matriks
- b. Membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks.
- c. Menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah elemen untuk mendapatkan nilai rata-rata.

5. Mengukur Konsistensi

Dalam pembuatan keputusan, penting untuk mengetahui seberapa baik konsistensi yang ada karena kita tidak menginginkan keputusan berdasarkan pertimbangan dengan konsistensi yang rendah. Hal-hal yang dilakukan dalam langkah ini adalah sebagai berikut:

- a. Setiap nilai pada kolom pertama dikalikan dengan bobot prioritas elemen pertama, kemudian setiap nilai pada kolom kedua dikalikan dengan bobot prioritas elemen kedua dan seterusnya.
- b. Jumlahkan setiap baris (Σ baris).

- c. Hasil dari penjumlahan baris dibagi dengan elemen prioritas yang bersangkutan sehingga didapat lamda.

$$\lambda = \frac{\sum \text{baris}}{\text{prioritas}}$$

- d. Jumlahkan lamda (λ) dan hasilnya dibagi dengan banyaknya elemen yang ada, hasilnya disebut λ maks

$$\lambda \text{ maks} = \frac{\sum \lambda}{n}$$

dengan n = banyaknya elemen yang dibandingkan

6. Hitung *Consistency Index* (CI) dengan rumus:

$$CI = \left(\frac{\lambda \text{ maks} - n}{n - 1} \right)$$

n = banyaknya elemen yang dibandingkan

7. Hitung *Consistency Ratio* (CR) dengan rumus berikut:

$$CR = \frac{CI}{IR}$$

Keterangan:

CR = Rasio Konsistensi (*Consistency Ratio*)

CI = Indeks Konsistensi (*Consistency Index*)

IR = Indeks Random Konsistensi (*Index Random Consistency*)

8. Memeriksa konsistensi hierarki. Jika nilainya lebih dari 10%, maka penilaian data judgment harus diperbaiki. Namun jika Rasio Konsistensi (CI/CR) kurang atau sama dengan 0,1, maka hasil perhitungan bisa dinyatakan benar.

Nilai *Random Index* dapat dilihat pada table dibawah ini.

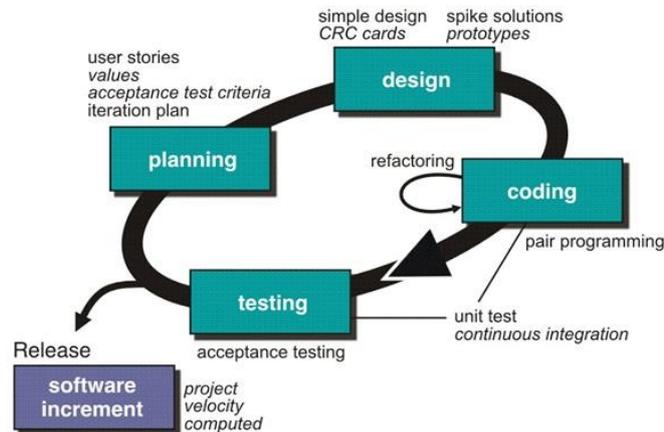
Tabel 2.3 Nilai *Random Index*

Ukuran Matriks	Nilai IR
1	0
2	0
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45
10	1.49
11	1.51
12	1.48
13	1.56
14	1.57
15	1.59

2.5 Metode Pengembangan *Extreme Programming*

Menurut Supriyatna (2018) *Extreme Programming* (XP) adalah proses rekayasa perangkat lunak yang biasanya menggunakan pendekatan berorientasi objek. Metode ini cocok untuk tim 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.

Sedangkan menurut Fatoni and Dwi (2016) *Extreme programming* adalah model pengembangan perangkat lunak yang menyederhanakan berbagai tahapan pengembangan sistem menjadi lebih efisien, adaptif dan fleksibel. Berikut ini empat tahapan XP dalam pengembangan sistem yang harus dilakukan.



Gambar 2.2 Tahapan pada *Extreme Programming* (XP)

Sumber: Supriyatna (2018)

1. Perencanaan (*Planning*)

Tahap *planning* dimulai dengan membuat *user stories* yang menggambarkan output, fitur, dan fungsi-fungsi dari *software* yang akan dibuat. *User stories* tersebut kemudian diberikan *bobot* seperti *prioritas* dan dikelompokkan untuk selanjutnya dilakukan proses *delivery* secara *incremental*. *Extreme Programming* (XP) team kemudian menilai setiap *story* dan menentukan *cost* diukur dalam *development week*. *Customer* dan *Extreme Programming* (XP) team bekerja bersama untuk memutuskan bagaimana *group story* untuk *release* berikutnya (*software increment*) berikutnya untuk dibangun oleh *Extreme Programming* (XP) team. Jika komitmen telah dibuat, *Extreme Programming* (XP) team akan membangun *story-story* dengan cara:

- a. Semua *story* segera diimplementasikan (dalam beberapa minggu).
- b. *Story* dengan *value* tertinggi akan dipindahkan dari jadwal dan diimplementasikan pertama.

c. *Story* dengan resiko paling tinggi akan diimplementasikan lebih dulu.

Setelah *project* pertama di-*release* dan *delivery*, *Extreme Programming* (XP) team memperhitungkan kecepatan *project*. Selama *development*, *customer* dapat menambah *story*, merubah *value*, membagi *story* atau menghapusnya. Sedangkan *Acceptance Test Criteria* dari tiap fungsional sistem yang diinginkan oleh *Customer*.

2. Perancangan (*Design*)

Tahapan design pada *Extreme Programming* mengikuti prinsip *Keep It Simple* (KIS). Untuk desain sederhana menggunakan kartu CRC (*Class Responsibility Collaborator*). CRC ini digunakan untuk pemetaan (membangun) kelas – kelas yang akan digunakan pada *diagram use case*, *diagram class*, dan *diagram objek*. Sedangkan untuk design yang sulit, *Extreme Programming* akan menggunakan *Spike Solution* dimana pembuatan design dibuat langsung ke tujuannya. *Extreme Programming* juga mendukung adanya *Prototype* digunakan untuk mengimplementasikan dan evaluasi bertujuan untuk menurunkan risiko ketika implementasi yang sebenarnya dimulai dan untuk memvalidasi perkiraan awal untuk *stories* yang berisi masalah desain.

3. Pengkodean (*Coding*)

Tahapan proses *coding* pada *Extreme Programming* diawali dengan membangun serangkaian unit test. Setelah itu pengembang akan berfokus untuk mengimplementasikannya. Dalam *Extreme Programming* diperkenalkan istilah *Pair Programming* dimana proses penulisan program dilakukan secara berpasangan. Dua orang programmer saling bekerjasama

di satu komputer untuk menulis program. Dengan melakukan ini akan didapat *real-time problem solving* dan *real-time quality assurance*. Setelah *pair programming* selesai, code diintegrasikan dengan kerja lainnya (*continou sintegration*). Tahap ini dapat dilakukan secara berulang – ulang (*refactoring*) apabila terdapat koreksi dari tahap berikutnya.

4. Pengujian (*Testing*)

Tahapan ini merupakan tahapan pengujian terhadap aplikasi yang sudah dibangun, pada tahapan ini ditentukan oleh pengguna sistem dan berfokus pada fitur dan fungsionalitas dari keseluruhan sistem kemudian ditinjau oleh pengguna sistem. Metode yang digunakan dalam melakukan pengujian terhadap aplikasi sistem pendukung keputusan penerima bantuan dana desa adalah *Black-Box Testing* dengan melakukan pengujian terhadap masukan dan keluaran yang dihasilkan sistem.

2.6 UML (*Unified Modeling Language*)

Menurut Rosa dan Shalahuddin (2018) Bahasa Pemodelan Pengembangan Sistem (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek:

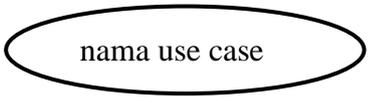
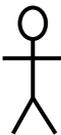
2.6.1 *Use Case Diagram*

Use Case Diagram merupakan gambaran dari fungsionalitas yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat *actor* yang merupakan sebuah gambaran

entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem.

Simbol-simbol yang ada pada *use case diagram* dapat dilihat pada tabel 2.5

Tabel 2.4 Simbol *use case diagram*

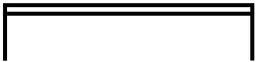
Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses atau sistem lain yang 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 aktor</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 aktor</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><<<i>include</i>>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p>

Sumber: (Rosa & Shalahuddin, 2018)

2.6.2 Activity Diagram

Activity Diagram merupakan gambaran alir dari aktivitas-aktivitas didalam sistem yang berjalan. Simbol-simbol yang ada pada *activity* diagram dapat dilihat pada tabel 2.7

Tabel 2.5 Simbol *Activity Diagram*

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Tabel 	Suatu file komputer dari mana data bisa dibaca atau direkam selama kejadian bisnis
Dokumen 	Menunjukkan dokumen sumber atau laporan
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: (Rosa & Shalahuddin, 2018)

2.7 *Black Box Testing*

Metode *Black Box Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang *output* pakai. Proses *Black Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan (Zuhair et al., 2020).

Pada *Black Box Testing* terdapat jenis teknik design tes yang dapat dipilih berdasarkan pada tipe testing yang akan digunakan, diantaranya sebagai berikut:

1. *Equivalence Class Partitioning*
2. *Boundary Value Analysis*
3. *State Transitions Testing*
4. *Cause-Effect Graphing*

Kategori kesalahan/*error* yang akan diketahui melalui *black box testing*:

1. Fungsi yang hilang atau tidak benar
2. *Error* dari interface/antar-muka
3. *Error* dari struktur data atau akses eksternal *database*
4. *Error* dari kinerja atau tingkah laku/perform
5. *Error* dari inisialisasi dan terminasi

2.8 Website

Menurut Silaban dkk. (2022) *Website* atau disingkat web, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa *text*, gambar, video, audio dan animasi lainnya yang disediakan melalui jalur koneksi internet (Silaban et al., 2022).

Website atau situs dapat juga diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan- jaringan halaman (Suheri, 2022).

2.9 MySQL

Menurut Apriliani dkk. (2019) yang mengutip buku Raharjo (2011) MySQL merupakan salah satu *Relational Database Management System* (RDBMS) yang saat ini sedang banyak digandrungi oleh para pengembang aplikasi *database*, baik untuk aplikasi desktop maupun aplikasi web untuk menyimpan, mengatur, dan mengelola data pada aplikasi tersebut. Beberapa kelebihan MySQL dibandingkan dengan RDBMS lain adalah mudah, simple, gratis, stabil, dan portable (dapat diterapkan pada beberapa sistem operasi yang berbeda) (Apriliani et al., 2019).

Menurut MADCOMS (2016) “MySQL adalah sistem manajemen *Database SQL* yang bersifat *Open Source* dan paling populer saat ini. Sistem Database MySQL mendukung beberapa fitur seperti multithreaded, multi-user dan *SQL Database Management System* (DBMS)”.

2.10 XAMPP

Menurut Sheldon dan Moes (2015) XAMPP merupakan pengembangan dari LAMP (Linux, Apache, MySQL, PHP and PERL), XAMPP ini merupakan project *nonprofit* yang di kembangkan oleh Apache Friends yang didirikan Kai 'Oswalad' Seidler dan Kay Vogelgesang pada tahun 2002, project mereka ini bertujuan mempromosikan penggunaan Apache web serve. XAMPP merupakan paket program web lengkap yang dapat Anda gunakan untuk mempelajari pemrograman web khususnya PHP dan MySQL.

Xampp adalah sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah *server*. Kegunaan Xampp ini untuk membuat jaringan lokal sendiri dalam artian kita dapat membuat *website* secara offline untuk masa coba-coba di komputer sendiri. Jadi fungsi dari Xampp server itu sendiri merupakan *server website* kita untuk cara memakainya. Disebut *server* karena dalam hal ini komputer yang akan kita pakai harus memberikan pelayanan untuk mengakses web, untuk itu komputer kita harus menjadi *server* (Utomo et al., 2020).

Dapat disimpulkan Xampp adalah aplikasi yang menyediakan konfigurasi Apache, PHP, dan MySQL untuk membantu kita dalam proses pembuatan aplikasi web yang terintegrasi, yang membuatnya lebih mudah untuk digunakan.

2.11 PHP

Menurut MADCOM (2016) “PHP (*Hypertext Preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis”.

Menurut Rohi Abdulloh (2015) “PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programing*, yaitu bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun website adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke *database*, diedit, dihapus, dan ditampilkan pada *website* yang diatur oleh PHP”.