

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Ada beberapa penelitian terkait dengan sistem pakar diagnosis penyakit pada anak, diantaranya adalah :

(Artika, 2013), dalam penelitiannya dengan judul “Penerapan Analytical Hierarchy Process (AHP) Dalam Pendukung Keputusan Penilaian Kinerja Guru Pada Sd Negeri 095224” mengatakan bahwa : pemberian kriteria kriteria dalam penentuan penilaian kinerja guru dapat membantu dalam mengambil keputusan untuk menentukan kinerja guru yang berprestasi. Dengan menerapkan metode AHP proses pemilihan guru berprestasi menjadi lebih efisien. Parameter kriteria yang digunakan berjumlah 7 kriteria yaitu kesetiaan, prestasi kerja, tanggung jawab, ketaatan, kejujuran, kerjasama dan prakarsa[1].

(Safitri, Waruwu, & Mesran, 2017), dalam penelitiannya dengan judul “Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi Dengan Menggunakan Metode Analytical Hierarchy Process, Studi Kasus : PT.Capella Dinamik Nusantara Takengon” mengatakan bahwa : penilaian karyawan pada PT.Capella Dinamik Nusantara Takengon belum dilakukan secara optimal karena hanya berdasarkan keputusan pimpinan yang memungkinkan bersifat tidak objektif. Metode yang digunakan adalah AHP (Analytical Hierarchy Process) sebagai metode untuk melakukan perbandingan bobot kriteria untuk menentukan perbandingan. Hasil dari penelitian ini dapat membantu pihak perusahaan dalam menentukan karyawan berprestasi dengan lebih baik. Parameter kriteria yang digunakan ada 4 kriteria yaitu kejujuran, kedisiplinan, rajin dan tanggung jawab[2].

(Saefudin & Wahyuningsih, 2014), dalam penelitiannya dengan judul “Sistem Pendukung Keputusan Untuk Penilaian Kinerja Pegawai Menggunakan Metode Analytical Hierarchy Process (AHP) Pada RSUD Serang” mengatakan bahwa sebelumnya penilaian dilakukan secara manual sehingga proses penilaian menjadi lambat dan selain itu penilaian masih bersifat subjektif. Sistem pendukung keputusan

yang dirancang dapat membantu pihak rumah sakit dalam melakukan penilaian sehingga proses penilaian bisa dilakukan lebih efisien dan juga bersifat objektif[3].

(Hijriani & dkk, 2013), dalam penelitiannya dengan judul “Analisis Dan Perancangan Perekrutan Karyawan Dengan Metode AHP Pada Sistem Berorientasi Service Studi Kasus Usaha Jasa Service Kendaraan” mengatakan bahwa : perekrutan karyawan sebelumnya masih dilakukan tanpa menggunakan kriteria yang pasti sehingga penilaian terhadap calon karyawan belum bisa dilakukan secara maksimal. Dengan penggunaan AHP kedalam sistem proses pengambilan keputusan dalam perekrutan karyawan menjadi lebih mudah dan proposional karena sesuai dengan kriteria kriteria yang sudah ditetapkan. Parameter kriteria yang digunakan berjumlah 3 kriteria yaitu kemampuan, pengalaman dan pendidikan[4].

(Sari & Saleh, 2014) dalam penelitiannya dengan judul “Penilaian Kinerja Dosen Dengan Menggunakan Metode AHP (Studi Kasus : di STMIK Potensi Utama Medan)” mengatakan bahwa : metode AHP ternyata dapat digunakan dalam proses penilaian kinerja dosen, karena metode tersebut mampu menyelesaikan masalah multikriteria yang belum terstruktur menjadi lebih terstruktur dan lebih mudah dipahami dengan hasil yang akurat. Dengan penggunaan metode AHP proses penilaian menjadi lebih obyektif dan akurat. Parameter kriteria yang digunakan berjumlah 4 kriteria yaitu kehadiran dosen, pengumpulan nilai, keterlambatan masuk pembelajaran dan kecepatan selesai pembelajaran[5].

Kelima penelitian tersebut dianggap relevan dalam penelitian ini karena dapat digunakan sebagai acuan perancangan sistem. Dari penelitian tersebut terdapat kesamaan yaitu dalam penggunaan metode Analytical Hierarchy Process untuk menyelesaikan masalah yang dihadapi. Dari penelitian tersebut yang membedakan dengan penelitian yang sedang di lakukan adalah jumlah parameter kriteria pembandingan yang digunakan, dalam penelitian yang sedang dilakukan menggunakan parameter kriteria pembandingan berjumlah 8 kriteria yaitu kesetiaan, kerjasama, prestasi kerja, prakarsa, ketaatan, kejujuran, tanggung jawab dan kedisiplinan.

2.2 Landasan Teori

2.2.1 Sistem

Sistem merupakan kumpulan elemen yang saling berkaitan yang bertanggung jawab memproses masukan (inputan) sehingga menghasilkan keluaran (output) (Kusrini, 2007). Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan sistem tersebut[6].

2.2.2 Sistem Penunjang Keputusan

Sistem Pendukung Keputusan (SPK) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat (Turban, 2005)[7].

Sistem pendukung keputusan merupakan sebuah sistem yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah semi-terstruktur yang spesifik. Jadi sistem pendukung keputusan adalah sistem yang dipakai untuk mendukung pengambilan keputusan dalam menyelesaikan suatu masalah agar masalah yang ada dapat diselesaikan dengan baik.

Karakteristik SPK menurut (B. S. D. Oetomo, 2002)[8], adalah sebagai berikut:

a. Interaktif

SPK memiliki user interface yang komunikatif sehingga pemakai dapat melakukan akses ke data dan memperoleh informasi yang dibutuhkan.

b. Fleksibel

SPK memiliki sebanyak mungkin variabel masukan, kemampuan untuk mengolah, dan memberikan keluaran yang menyajikan alternatif-alternatif keputusan kepada pemakai.

c. Data kualitas

SPK memiliki kemampuan menerima data kualitas yang dikuantitaskan yang sifatnya subyektif dari pemakainya, sebagai data masukan untuk pengolahan data. Misalnya penilaian terhadap kecantikan yang bersifat kualitas, dapat dikuantitaskan pemberian bobot nilai seperti 75 atau 90.

d. Prosedur pakar

SPK mengandung suatu prosedur yang dirancang berdasarkan rumusan formal atau juga beberapa prosedur kepakaran seseorang atau kelompok dalam menyelesaikan suatu bidang masalah dengan fenomena tertentu.

Komponen-komponen dari SPK adalah :

1. Data Management Termasuk database, yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh software yang disebut Database Management System (DBMS).
2. Model Management Melibatkan model finansial, statistikal, management science, atau berbagai model kualitatif lainnya, sehingga dapat memberikan ke sistem suatu kemampuan analitis, dan manajemen software yang dibutuhkan.
3. Communication User dapat berkomunikasi dan memberikan perintah pada DSS melalui subsistem ini. Ini berarti menyediakan antarmuka.
4. Knowledge Management Subsistem optional ini dapat mendukung subsistem lain atau bertindak atau bertindak sebagai komponen yang berdiri sendiri.
5. User, Pemakai yang mengaplikasikan pengetahuan ataupun sebagai pengguna dari sistem.

2.2.3 Sistem Penilaian Kinerja

Penilaian kinerja pegawai merupakan kegiatan pimpinan untuk mengevaluasi perilaku kinerja pegawai serta untuk menentukan kebijakan yang berhubungan dengan perusahaan (Malayu S.P Hasibuan, 2012)[9]. Penilaian kinerja pegawai di Dinas tanaman pangan, hortikultura, dan perkebunan Kabupaten Enrekang belum

menggunakan kriteria yang pasti sehingga menyulitkan dalam melakukan penilaian. Penilaian hanya didasarkan pada perilaku pegawai saja. Diperlukan pengembangan dalam proses penilaian. Penilaian kinerja meliputi penilaian kesetiaan, prestasi kerja, tanggung jawab, ketaatan, kejujuran, kerjasama, prakarsa dan kedisiplinan. Penilaian kinerja adalah menilai hasil kerja nyata dengan standar kualitas maupun kuantitas yang dihasilkan pegawai. Menetapkan kebijakan berarti apakah pegawai layak untuk di perpanjang kontraknya dan atau layak tidak untuk dilakukan promosi jabatan.

Penilaian kinerja pegawai biasanya menggunakan bobot dalam setiap indikator atau kriteria yang terkait dengan derajat kepentingan dari item tersebut. Pembobotan akan disesuaikan dengan besar kepentingannya. Manfaat dari penilaian ini adalah untuk perbaikan prestasi kerja, keputusan penempatan, pengembangan karir dan kesempatan kerja yang adil.

Sistem penilaian kinerja pegawai akan menghasilkan keluaran berupa penilaian kinerja pegawai dalam bentuk penilaian yang terdiri dari angka-angka yang menunjukkan kualitas kerja pegawai atau peringkat. Peringkat disini bisa menjadi alternatif, dimana pegawai dengan peringkat tertinggi dianggap sebagai pegawai dengan kinerja yang baik sebaliknya pegawai dengan peringkat terendah dianggap mempunyai kinerja yang masih kurang.

2.2.4 Metode *Analytical Hierarchy Proses* (AHP)

A. Definisi Analytical Hierarchy Process

Metode Analytical Hierarchy Process (AHP) dikembangkan oleh Thomas L. Saaty, seorang matematikawan di Universitas Pittsburgh Amerika Serikat sekitar tahun 1970. Tujuan utama AHP adalah untuk membuat rangking alternatif keputusan dan memilih salah satu yang terbaik bagi kasus multi kriteria yang menggabungkan faktor kualitatif dan kuantitatif di dalam keseluruhan evaluasi alternatif-alternatif yang ada (Shega dkk., 2012)[10].

AHP digunakan untuk mengkaji permasalahan yang dimulai dengan mendefinisikan permasalahan secara seksama kemudian disusun ke dalam suatu hirarki. AHP memasukan pertimbangan dan nilai-nilai pribadi secara logis.

B. Prinsip prinsip dasar AHP

Dalam (Shega dkk., 2012), ada beberapa prinsip yang harus dipahami dalam menyelesaikan permasalahan menggunakan AHP, yaitu :

1) Penyusunan hirarki

Merupakan langkah penyederhanaan masalah ke dalam bagian yang menjadi elemen pokoknya, kemudian ke dalam bagian-bagiannya lagi, dan seterusnya secara hirarki agar lebih jelas, sehingga mempermudah pengambilan keputusan untuk menganalisis dan menarik kesimpulan terhadap permasalahan tersebut.

2) Matriks perbandingan berpasangan

Dalam (Shega dkk., 2012), langkah awal dalam menentukan prioritas dari masing-masing elemen yang digunakan adalah dengan menyusun matriks perbandingan berpasangan.

Tabel 2.1 Skala Perbandingan Tingkat Kepentingan

Nilai	Interpretasi
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada elemen yang lainnya
5	Elemen yang satu lebih penting daripada yang lainnya
7	Satu elemen jelas lebih mutlak penting daripada elemen lainnya
9	Satu elemen mutlak penting daripada elemen lainnya
2,4,6,8	Nilai-nilai antara dua nilai pertimbangan-pertimbangan yang berdekatan
Kebalikan	Jika aktivitas i mendapat satu angka dibandingkan dengan aktivitas j, maka j memiliki nilai kebalikannya dibandingkan dengan i

Misalkan kriteria C memiliki beberapa elemen di bawahnya, yaitu A_1, A_2, \dots, A_n .

Tabel matriks perbandingan berpasangan berdasarkan kriteria C sebagai berikut :

Tabel 2.2 Matriks Perbandingan Berpasangan

C	A ₁	A ₂	...	A _n
A ₁	1	α_{12}	α_{1n}	A ₁
A ₂	A ₂₁	1	A _{2n}	A ₂
...
A _n	A _{n1}	A _{n2}	...	1

C adalah kriteria yang digunakan sebagai dasar perbandingan. A_1, A_2, \dots, A_n adalah elemen-elemen pada satu tingkat di bawah C. elemen kolom sebelah kiri selalu dibandingkan dengan elemen baris puncak. Nilai kebalikan diberikan kepada elemen baris ketika tampil sebagai elemen kolom dan elemen kolom tampil sebagai elemen baris. Dalam matriks ini terdapat perbandingan dengan selemen itu sendiri pada diagonal utama dan bernilai 1.

3) Menentukan prioritas

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

4) Konsistensi logis

Konsistensi memiliki dua makna, yang pertama Objek-objek yang serupa bisa dikelompokkan sesuai dengan keseragaman dan relevansi. Kedua menyangkut

tingkat hubungan antar objek yang didasarkan pada kriteria tertentu. Perhitungan konsistensi logis dilakukan dengan mengikuti langkah-langkah sebagai berikut :

- a) Mengalikan matriks dengan prioritas prioritas bersesuaian

$$(A)(W^T) \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \dots \\ W_n \end{bmatrix} \dots \dots \dots (2.1)$$

- b) Menjumlahkan hasil perkalian perbaris.

c) Hasil penjumlahan tiap baris dibagi prioritas bersangkutan dan hasilnya dibagi dengan jumlah elemen yang didapatkan

$$t = \frac{1}{n} \sum_{i=1}^n \left(\frac{\text{elemen ke-}i \text{ pada } (A)W^T}{\text{elemen ke-}i \text{ pada } W^T} \right) \dots \dots \dots (2.2)$$

- d) Indeks Konsistensi (CI)

$$CI = \frac{t-n}{n-1} \dots \dots \dots (2.3)$$

e) Rasio Konsistensi = CI/RI, di mana RI adalah indeks random konsistensi. Jika rasio konsistensi ≤ 0.1 , hasil perhitungan data dapat dibenarkan atau cukup konsistensi.

Tabel 2.3 Nilai Random Indeks Saaty

Ukuran Matriks	Nilai Random Index (RI)
1,2	0,00
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

2.2.5 Pengembangan Sistem

A. Metode *Waterfall*

Metode *waterfall* adalah metode pengembangan perangkat lunak yang dilakukan secara sistematis dan terurut. Analoginya yakni seperti air terjun, jadi setiap proses perlu dikerjakan secara berurutan (dari atas sampai kebawah). Tahapan dalam melakukan pengembangan perangkat lunak dengan menggunakan metode *waterfall* ada 5, diantaranya[11] :

1. *Requirements*

Pada tahap ini, yang pertama dilakukan yakni mempersiapkan dan menganalisa kebutuhan dari perangkat lunak (*software*) yang akan dikerjakan. Informasinya dapat diperoleh melalui survei, wawancara, studi literatur, observasi hingga diskusi.

2. *Design*

Selanjutnya adalah tahap pembuatan desain dari aplikasi yang akan dikerjakan sebelum proses *coding*. Tahapan ini bertujuan agar memberikan gambaran yang jelas mengenai struktur data ataupun arsitektur *software*, fungsi internal dan eksternal dari algoritma yang diperlukan hingga tampilan dari *software*.

3. *Implementation*

Pada tahap ini desain dari *software* yang diinginkan diimplementasikan kedalam kode program dengan menggunakan berbagai *tools* dan bahasa pemrograman yang diinginkan.

4. *Integration & Testing*

Kemudian tahapan selanjutnya ialah *integration & testing*. Pada tahap ini dilakukan proses integrasi dan pengujian dari sistem yang telah dibuat. Yang bertanggung jawab untuk melakukan proses *testing* biasanya adalah QA (*Quality Assurance*) dan QC (*Quality Control*). Mereka mengecek apakah *software* sudah sesuai dengan desain tadi serta apakah terdapat *error* atau *bug*.

5. *Maintenance*

Tahapan yang terakhir adalah *maintenance*. Pada tahapan ini dilakukan pengoperasian dan perbaikan dari *software* atau aplikasi. Setelah pengujian sistem telah selesai, maka akan masuk ke tahap *software* tersebut dicoba oleh *user* (pengguna). Untuk proses pemeliharaan dari *software*, pengembang biasanya meminta *feedback* atau laporan dari *user* apabila *user* mendapat *error* atau *bug* dari aplikasi yang telah dibuat.

Adapun kelebihan dan kekurangan dari metode *waterfall*, yakni :

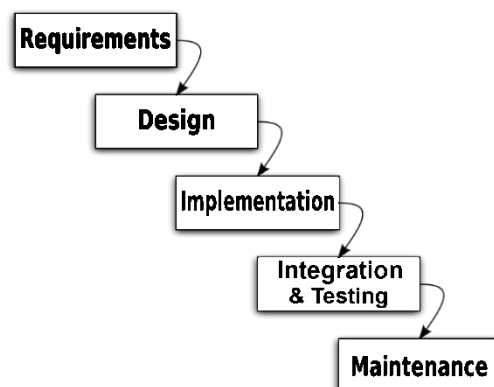
a. Kelebihan metode *waterfall*

- 1) *Workflow* (aliran kerja) yang jelas
- 2) Hasil dokumentasi yang baik
- 3) Dapat menghemat biaya
- 4) Digunakan untuk pengembangan *software* berskala besar.

b. Kekurangan metode *waterfall*

- 1) Membutuhkan tim yang *solid*
- 2) Masih kurangnya fleksibilitas
- 3) Tidak dapat melihat gambaran sistem yang jelas
- 4) Membutuhkan waktu yang lama.

Dapat dilihat pada gambar 2.1



Gambar 2.1 Alur Metode Waterfall

2.2.6 Desain Database

A. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek.” *Entity Relationship Diagram* (ERD) digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Entity Relationship Diagram* (ERD) didasarkan pada suatu persepsi bahwa *real world* terdiri atas obyek-obyek dasar tersebut. Penggunaan *Entity Relationship Diagram* (ERD) relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, *Entity Relationship Diagram* (ERD) berguna untuk memodelkan sistem yang nantinya, basis data akan di kembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antardata didalamnya[12].

Terdapat 3 komponen dalam membentuk suatu ERD, diantaranya :

1. Entitas, adalah sebuah objek untuk membedakan dari yang lain yang akan diwujudkan dalam basis data nantinya.
2. Hubungan / Relasi, adalah hubungan antara 2 jenis entitas yang digambarkan melalui garis lurus.
3. Atribut, memberikan detail informasi dari entitas. Jenis atribut ada 5 tergantung dari tipe data entitas, diantaranya :
 - a. Atribut *Primary Key*, ialah atribut yang unik karena tidak memiliki nilai yang sama pada baris data yang lain.
 - b. Atribut *Simple*, ialah atribut yang bernilai *atomic* atau tidak dapat dipecah maupun dipilah lagi.
 - c. Atribut *Multivalued*, ialah atribut yang memiliki lebih dari 1 nilai dari atribut yang bersangkutan.
 - d. Atribut *Composite*, ialah atribut yang terdiri dari beberapa atribut yang lebih kecil yang berarti masih dapat dipecah lagi atau memiliki sub atribut.
 - e. Atribut Derivatif, ialah atribut yang tidak harus disimpan dalam basis data.

Dalam perancangan ERD seringkali dijumpai derajat relasi, yang bertujuan untuk menjelaskan jumlah maksimum antara 1 entitas dengan entitas yang lain.

1. *One to One* (1 : 1)

Setiap anggota entitas A hanya dapat berhubungan dengan 1 anggota dalam entitas B, begitu pula sebaliknya.

2. *One to Many* (1 : M)

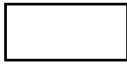
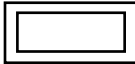







Setiap anggota entitas A dapat berhubungan dengan banyak anggota dalam entitas B, namun tidak sebaliknya.

3. *Many to Many* (M : M)

Setiap anggota entitas A dapat berhubungan dengan banyak anggota dalam entitas B, begitu pula sebaliknya.

Berikut adalah simbol-simbol yang sering digunakan dalam pembuatan ERD, diantaranya :

Tabel 2.4 Simbol ERD

No	Notasi	Arti
1		<i>Entity</i>
2		<i>Weak Entity</i>
3		<i>Relationship</i>
4		<i>Identifying Relationship</i>
5		Atribut
6		Atribut <i>Primary Key</i>
7		Atribut <i>Multivalued</i>
8		Atribut <i>Composite</i>
9		Atribut Derivatif

2.2.7 Desain Sistem

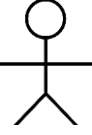





A. *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah sebuah metode untuk memodelkan suatu rancangan sistem berorientasi objek secara visual. Dalam UML terdapat 14 macam diagram seperti *Class Diagram*, *Object Diagram*, *Component Diagram*, *Composite Diagram*, *Composite Structure Diagram*, *Package Diagram*, *Deployment Diagram*, *Use Case Diagram*, *Activity Diagram*, *State Machine Diagram*, *Sequence Diagram*, *Communication Diagram*, *Timing Diagram*, dan *Interaction Overview*. Disini penulis hanya akan menggunakan 2 macam diagram[13], yaitu :

1. *Use Case Diagram*

Use case diagram adalah jenis diagram pemodelan yang menggambarkan hubungan interaksi antara 1 atau lebih aktor dengan sistem atau aplikasi yang akan dibuat. Aktor disimbolkan menyerupai seseorang sedangkan sistem disimbolkan menyerupai elips. Aktor merupakan orang, proses atau sistem lain yang melakukan interaksi dengan sistem informasi yang akan dibuat, yang berada diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor menyerupai orang, belum tentu aktor itu sendiri merupakan orang. *Use case* adalah fungsi yang disediakan sistem sebagai unit-unit yang saling berinteraksi atau bertukar pesan antar unit atau aktor. Berikut adalah simbol-simbol yang sering digunakan dalam membuat *use case diagram* :

Tabel 2.5 Simbol Use Case Diagram

No	Nama	Simbol	Keterangan
1.	<i>Actor</i>		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> belum tentu merupakan orang: biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama <i>actor</i> .
2.	<i>Use Case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau <i>actor</i> ; biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>Use Case</i> .
3.	<i>Asosiation</i>		Komunikasi antara <i>actor</i> dan <i>Use Case</i> yang berpartisipasi pada <i>Use Case</i> atau <i>Use Case</i> memiliki interaksi dengan <i>actor</i> .
4.	<i>Generalization</i>		Menspesifikasikan bahwa <i>Use Case</i> target memperluas perilaku dari <i>Use Case</i> sumber pada suatu titik yang diberikan.
5.	<i>Include</i>		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.
6.	<i>Extend</i>		Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>Use Case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari hasilnya.

2. Activity Diagram



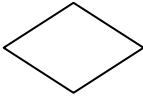


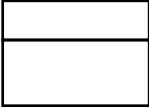
Activity diagram adalah jenis diagram pemodelan yang menggambarkan *workflow* atau aliran kerja dari sebuah sistem atau aplikasi yang dibuat. Pada diagram ini hanya aktivitas dari sistem saja yang digambarkan, jadi aktivitas dari aktor tidak dapat dilihat. *Activity* diagram sering digunakan untuk hal-hal berikut, yaitu :

- a. Rancang menu yang ditampilkan pada *software*.
- b. Urutan atau pengelompokkan tampilan dari sistem atau *user interface*, dimana setiap aktivitas dianggap memiliki suatu antarmuka tampilan.

- c. Rancangan proses bisnis, dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis dari sistem yang didefinisikan.
- d. Rancangan pengujian, dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Berikut adalah simbol-simbol yang sering digunakan dalam pemodelan *activity* diagram :

Tabel 2.6 Simbol Activity Diagram

No	Nama	Simbol	Keterangan
1.	Status Awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah awal.
2.	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan atau <i>Decision</i>		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan atau <i>join</i>		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status Akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.2.8 Pengujian

A. *Black Box Testing*

Black Box Testing berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Black Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

1. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan valid?
2. Input seperti apa yang dapat menjadi bahan kasus uji yang baik?
3. Apakah sistem sensitif pada input-input tertentu?
4. Bagaimana sekumpulan data dapat diisolasi?
5. Berapa banyak rata-rata data dan jumlah data yang dapat ditangani sistem?

Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem[14]?

B. *White Box Testing*

White box testing atau yang dapat diartikan menjadi “pengujian kotak putih” adalah pengujian yang dilakukan untuk menguji perangkat lunak dengan cara menganalisa dan meneliti struktur internal dan kode dari perangkat lunak. Lain halnya dengan *black box testing* yang hanya melihat hasil input dan output dari perangkat lunak, pengujian *white box testing* berfokus pada aliran input dan output dari perangkat lunak[15].

C. *User Acceptance Testing (UAT)*

User Acceptance Testing (UAT) merupakan salah satu metodologi yang sangat inovatif untuk mencegah kegagalan proyek IT. Dalam pengembangan perangkat lunak, terdapat tiga hal yang dilakukan dalam proses UAT yaitu :

- a. UAT mengukur bagaimana sistem sudah sesuai dengan kebutuhan pengguna.
- b. UAT mengekspos fungsionalitas/*logic* bisnis yang belum ditemukan, karena *unit testing* dan *system testing* tidak berfokus pada fungsionalitas/*logic* bisnis.
- c. UAT membatasi bagaimana sistem telah “selesai” dibuat.

Proses UAT diawali dengan menyediakan dokumentasi persyaratan bisnis, kemudian dilanjutkan dengan proses bisnis (alur kerja) atau skenario dan yang terakhir yaitu pengujian menggunakan data. Efektifitas dalam pengujian sangat dibutuhkan dalam pengembangan sebuah aplikasi ataupun sistem informasi sehingga produk tersebut dapat sampai kepada pengguna dengan tepat waktu dan sesuai dengan kebutuhan pengguna. Berdasarkan penelitian yang dilakukan oleh *Zhang*, efektifitas terhadap kriteria pengujian yang sudah ada dan yang baru harus dievaluasi untuk membangun teori pengujian yang lebih berguna[16].

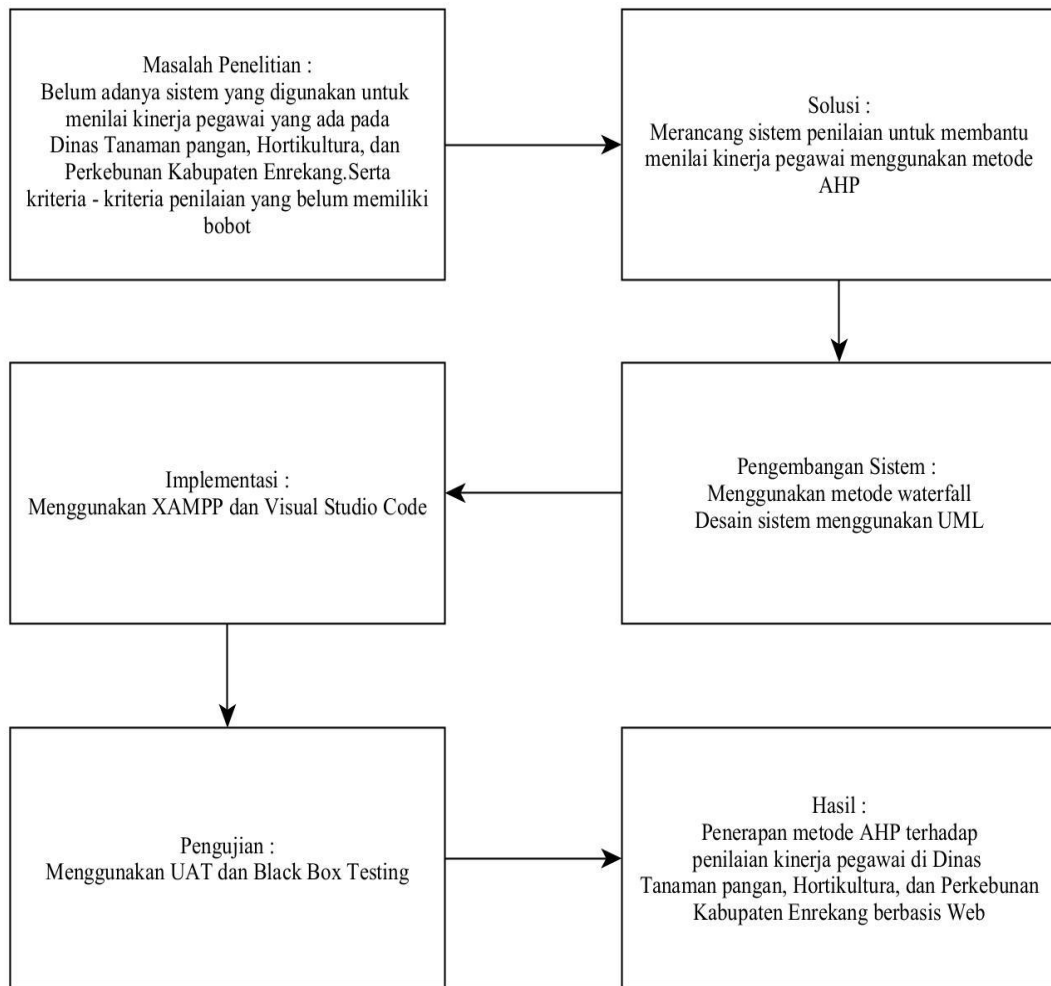
2.2.9 XAMPP

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah *database MySQL* di komputer lokal. XAMPP berperan sebagai *server web* pada komputer[17].

2.3 Kerangka Pikir

Kerangka pikir atau tahapan penelitian yang akan dilakukan dapat dilihat pada

Gambar 2.2 berikut :



Gambar 2.2 Kerangka Pikir