

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Rancang Bangun**

Menurut Zulfiandri (2016:474) “Rancang Bangun yaitu suatu menerjemahkan hasil analisa kedalam bentuk paket perangkat lunak yang kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada.”

Sedangkan menurut (Hasyim dkk, 2016:2) “Rancang Bangun adalah tahap awal dari membuat sebuah gambaran dan bentuk sketsa yang belum pernah dibuat sama sekali lalu dikelola menjadi gambaran atau sketsa yang memiliki fungsi yang diinginkan.”

#### **2.2 Sistem**

Menurut Arif (2017:11) “Sistem merupakan kumpulan dari komponen apapun yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu.”

Menurut Marimin (2017:1) “Sistem adalah suatu kesatuan usaha yang terdiri atas bagian - bagian yang saling berkaitan secara teratur satu sama lain dan berusaha mencapai suatu tujuan dalam suatu lingkungan kompleks.”

Sedangkan menurut Mulyadi (2016:2) “Sistem adalah kumpulan dari unsur - unsur yang saling berhubungan erat satu sama lain, yang berfungsi bersama - sama untuk mencapai suatu tujuan tertentu.”

### 2.3 Sistem Pendukung Keputusan

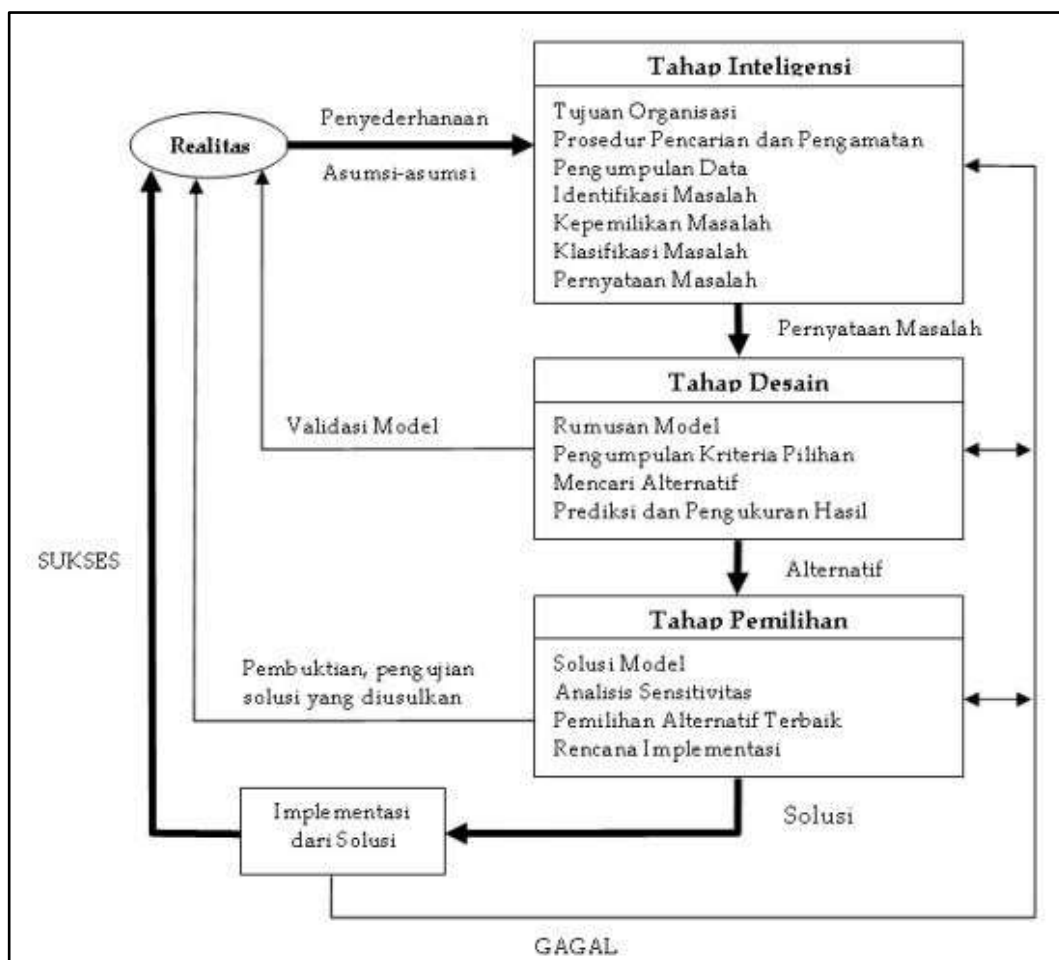
Menurut Nofriansyah & Defit (2017:2) mendiskusikan bahwa “Sistem Pendukung Keputusan adalah suatu sistem informasi spesifik yang ditujukan untuk membantu manajemen dalam mengambil suatu keputusan yang berkaitan dengan persoalan yang bersifat semi terstruktur. Sistem ini mempunyai fasilitas untuk menghasilkan berbagai alternatif yang secara interaktif digunakan oleh pemakai.”

Sedangkan menurut Pratiwi (2016:4) “Sistem Pendukung Keputusan (*Decision Support System*) adalah sistem berbasis komputer yang interaktif untuk membantu pengambil keputusan dengan memanfaatkan data serta model dalam menyelesaikan masalah - masalah yang tak terstruktur.”

Sistem Pendukung Keputusan mempunyai tiga tujuan yang akan dicapai yaitu :

1. Membantu manajer membuat keputusan untuk memecahkan masalah semiterstruktur,
2. Mendukung penilaian manajer bukan mencoba menggantikannya,
3. Meningkatkan efektifitas pengambilan keputusan manajer dari pada efisiensinya.

Sistem Pendukung Keputusan secara garis besar memiliki beberapa alur atau proses untuk mendapatkan keputusan yang terbaik seperti ditunjukkan pada gambar 2.1 dibawah ini.



Gambar 2.1. Alur / Proses Sistem Pendukung Keputusan (Pratiwi, 2016)

## 2.4 Zakat

Menurut Jalil (2019:13) “Zakat merupakan rukun ketiga dari Rukun Islam, Secara harfiah zakat merupakan tumbuh, berkembang, menyucikan, atau membersihkan. Sedangkan secara pengertian terminologi syari’ah, zakat merujuk pada suatu aktivitas memberikan sebagian kekayaan dalam jumlah dan perhitungan tertentu untuk orang - orang tertentu sebagaimana ditentukan.”

Sedangkan menurut Erwandi (2018:10) “Zakat merupakan salah satu rukun islam, dan menjadi salah satu unsur pokok bagi tegaknya syariat islam. Oleh sebab itu

hukum zakat adalah wajib (fardu) atas setiap muslim yang telah memenuhi syarat - syarat tertentu.” Zakat terbagi menjadi dua tipe, yakni :

1. Zakat Fitrah

Zakat Fitrah yaitu zakat yang wajib dikeluarkan muslim menjelang Idul Fitri pada bulan Ramadhan. Besar zakat ini setara dengan 2,5 kilogram makanan pokok yang ada di daerah bersangkutan.

2. Zakat Maal (Zakat Harta)

Zakat Maal yaitu zakat yang mencakup hasil perniagaan, pertanian, pertambangan, hasil laut, hasil ternak, harta temuan, emas dan perak serta hasil kerja (profesi). Masing - masing memiliki perhitungannya sendiri - sendiri.

## 2.5 Simple Multi Attribute Rating Technique (SMART)

Menurut Pratiwi (2016:141) “*Simple Multi Attribute Rating Technique* (SMART) merupakan suatu teknik atau metode yang *multiattribut* dalam sistem pengambilan keputusan.”

Metode SMART dikembangkan pada tahun 1977 oleh Edward. Teknik pembuatan keputusan multiatribut ini dapat digunakan untuk mendukung pembuat keputusan dalam memilih antara beberapa alternatif, setiap pembuat keputusan harus memilih sebuah alternatif yang sesuai dengan tujuan yang telah dirumuskan. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai - nilai. Nilai ini di rata - rata dengan skala tertentu.

Setiap atribut mempunyai bobot yang menggambarkan seberapa penting dibandingkan dengan atribut lain. Pembobotan dan pemberian peringkat digunakan untuk menilai setiap alternatif agar diperoleh alternatif terbaik.

Adapun algoritma penyelesaian dalam Metode SMART (Simple Multi Attribute Rating Technique) yaitu sebagai berikut :

**Langkah 1** : Menentukan jumlah kriteria dari keputusan yang akan di ambil.

**Langkah 2** : Menentukan bobot kriteria menggunakan interval 0 - 100 berdasarkan prioritas terpenting dari masing - masing kriteria.

**Langkah 3** : Melakukan normalisasi bobot kriteria menggunakan rumus  $(W_i / \sum W_j)$ .

Keterangan :

$W_i$  = bobot kriteria ke-i

$\sum W_j$  = jumlah bobot kriteria

**Langkah 4** : Memberikan nilai parameter untuk setiap kriteria. Nilai Parameter inilah yang nantinya akan digunakan untuk memberikan nilai kepada alternatif. Contohnya : Kriteria Fasilitas mempunyai list sub kriteria (sangat lengkap, lengkap, dan kurang lengkap) dengan masing - masing mempunyai nilai sub kriteria yaitu sangat lengkap artinya 3, lengkap artinya 2, dan tidak lengkap artinya 1.

**Langkah 5** : Menghitung nilai utility untuk setiap masing - masing kriteria menggunakan rumus  $U_i(a_i) = \frac{(C_{out\ i} - C_{min})}{(C_{max} - C_{min})}$ .

Keterangan :

$U_i(a_i)$  = nilai *utility* kriteria ke-1 untuk alternatif ke-i

$C_{out\ i}$  = nilai kriteria alternatif ke-i

$C_{min}$  = nilai kriteria minimal

$C_{max}$  = nilai kriteria maksimal

**Langkah 6** : Menghitung nilai akhir, Yaitu menentukan nilai akhir dari masing-masing alternatif dengan mengalikan nilai yang didapat dari normalisasi nilai kriteria data baku dengan nilai normalisasi pada bobot kriteria. Kemudian jumlahkan nilai dari perkalian tersebut. Rumus menghitung nilai akhir yaitu  $u_i(a_i) = w_j * u_j(a_i)$ .

Keterangan :

$u_i(a_i)$  = nilai total untuk alternatif ke-i

$w_j$  = nilai bobot kriteria ke-j yang sudah ternormalisasi

$u_j(a_i)$  = nilai *utility* kriteria ke-j untuk alternatif ke-i

**Langkah 7** : Perangkingan. Hasil perhitungan nilai akhir kemudian diurutkan dari nilai terbesar hingga terkecil, alternatif dengan nilai akhir yang terbesar menunjukkan alternatif yang terbaik.

## 2.6 Service Quality (SERVQUAL)

Para peneliti mendefinisikan kualitas dengan berbagai cara. Feigenbaum (1951, dalam Juwaheer, 2004) mengartikan kualitas sebagai nilai. Crosby (1979, dalam Juwaheer, 2004) mengartikannya sebagai konformansi terhadap persyaratan. Sedangkan Juran (1974, dalam Juwaheer, 2004) menyatakan bahwa kualitas adalah kesesuaian dengan penggunaan. Sementara itu Parasuraman *et al.* (1985, dalam

Juwaheer, 2004) maupun Koch (1991, dalam Lim *et al.*, 1999) lebih mengartikan kualitas sebagai pemenuhan harapan pelanggan.

Di antara berbagai metode pengukuran kualitas pelayanan, *SERVQUAL* merupakan metode yang paling banyak digunakan (Wisniewski, 2001; Arasli *et al.*, 2005). Karena frekuensi penggunaannya yang tinggi, *SERVQUAL* dipandang memenuhi syarat validitas secara statistik (Bryceland dan Curry, 2001). Pada bentuk awalnya, *SERVQUAL* terdiri atas sepuluh komponen kualitas pelayanan (Buttle, 1996). Dalam perkembangannya kemudian, kesepuluh komponen tersebut dimampatkan menjadi lima komponen, sebagai berikut (Juwaheer, 2004):

- a. *Reliability*, merujuk kepada kemampuan untuk memberikan pelayanan yang dijanjikan secara akurat dan handal.
- b. *Assurance*, merupakan tenaga kerja yang sopan dan berpengetahuan luas yang memberikan rasa percaya serta keyakinan.
- c. *Tangibles*, menggambarkan fasilitas fisik, perlengkapan, dan tampilan dari personalia serta kehadiran para pengguna.
- d. *Empathy*, mencakup kepedulian serta perhatian individual kepada para pengguna.
- e. *Responsiveness*, yaitu kesediaan untuk membantu partisipan dan memberikan perhatian yang tepat. Secara lengkap, *SERVQUAL* mengukur lima *gap* (kesenjangan), yaitu (Curry dan Sinclair, 2002; Antony *et al.*, 2004):
  - a. *Gap 1*, antara harapan pelanggan dan persepsi manajemen tentang harapan tersebut
  - b. *Gap 2*, antara persepsi manajemen tentang harapan pelanggan dan spesifikasi dari kualitas pelayanan
  - c. *Gap 3*, antara spesifikasi kualitas pelayanan dan pemberian pelayanan
  - d. *Gap 4*, antara pemberian pelayanan dan komunikasi eksternal
  - e. *Gap 5*, antara persepsi dan harapan pelanggan

Terkait dengan titik tekan dan perhatian kepada pelanggan, seringkali *Gap* yang diperlukan adalah *Gap* kelima, yaitu *Gap* antara persepsi dan harapan pelanggan. Dengan diterapkannya *SERVQUAL*, terdapat beberapa manfaat yang dapat diperoleh (Wisniewski, 2001):

- a. Mendapatkan pemahaman tentang kualitas pelayanan saat ini
- b. Mendapatkan pemahaman menyeluruh tentang bobot kepentingan tiap-tiap dimensi berdasarkan sudut pandang pelanggan
- c. Dapat melakukan perbandingan terhadap berbagai kelompok pelanggan yang terdapat pada suatu organisasi
- d. Dapat melakukan perbandingan kualitas pelayanan yang diberikan oleh berbagai bagian yang ada di dalam suatu organisasi
- e. Dapat digunakan untuk memahami pelanggan internal secara lebih baik.

Tujuan yang diharapkan tercapai dengan dilaksanakannya pengabdian masyarakat ini adalah sebagai berikut:

1. Khalayak sasaran memahami *SERVQUAL*, diukur dengan hasil *post test* yang melebihi hasil *pre test* secara signifikan.
2. Khalayak sasaran mampu menggunakan *SERVQUAL* di dalam mengukur kualitas pelayanan, dibuktikan dengan berhasil dioperasionalisasinya metode tersebut di dalam program pengabdian.

Dengan dilaksanakannya pengabdian masyarakat ini, diharapkan adanya dua manfaat berikut ini:

1. Khalayak sasaran (ta'mir masjid) memahami *SERVQUAL* dan mampu menerapkan metode tersebut.



2. Kemampuan khalayak sasaran untuk membuat program kegiatan yang didasarkan pada kebutuhan pengguna masjid yang masih belum terpenuhi akan semakin meningkat.

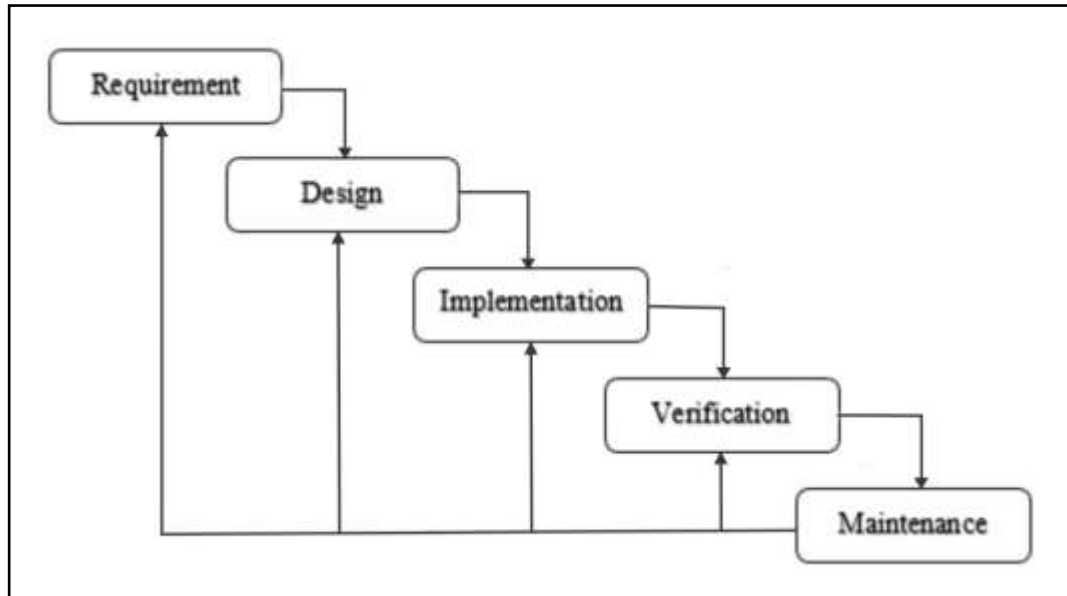
## **2.7 Metode Pengembangan**

Dalam perancangan sistem ini, metode pengembangan yang digunakan adalah metode *system development life cycle* dengan Model Waterfall.

### **2.7.1 Model Waterfall**

Menurut Pressman (2015:42) “waterfall model merupakan pola model tertua pada pengembangan software yang bertujuan untuk memberikan saran yang sistematis kepada pengembang software yang diawali dari kebutuhan pelanggan yang kemudian diteruskan hingga berpuncak pada dukungan untuk penyelesaian software. Proses pengembangan yang sangat terstruktur ini dapat membuat potensi kerugian akibat kesalahan pada proses sebelumnya sangat besar dan acap kali mahal karena membengkaknya biaya pengembangan ulang.”

Model waterfall melakukan pendekatan secara sistematis dan terurut. Sesuai dengan namanya yaitu Waterfall, maka tahapan dalam model ini disusun bertingkat. setiap tahap dalam model ini dilakukan berurutan, satu sebelum yang lainnya. Model ini biasanya digunakan untuk membuat sebuah software dalam skala besar dan yang akan dipakai dalam waktu yang lama. Berikut fase dari metode waterfall :



Gambar 2.2. Model Waterfall (Pressman, 2015)

Adapun penjelasan urutan tahapan - tahapan yang dimiliki oleh waterfall adalah sebagai berikut :

### 2.7.1.1 Requirement Analysis

Menurut Sugiyono (2015:335) “Analisis adalah sebuah kegiatan untuk mencari suatu pola. selain itu analisis juga merupakan cara berpikir yang berkaitan dengan pengujian secara sistematis terhadap sesuatu untuk menentukan bagian hubungan antar bagian dan hubungannya dengan keseluruhan.”

Seluruh kebutuhan software didapatkan pada fase ini. Pada tahap ini informasi dapat didapatkan melalui wawancara, survei atau diskusi yang kemudian informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahapan selanjutnya.

### **2.7.1.2 Design**

Tahap ini adalah tahapan sebelum memulai pemrograman. Tahap ini bertujuan untuk memberikan gambaran tentang apa yang harus dikerjakan dan bagaimana tampilannya. Tahap ini membantu dalam menspesifikasikan kebutuhan hardware dan sistem serta mendefinisikan arsitektur sistem secara keseluruhan.

#### **2.7.1.2.1 UML (Unified Modelling Language)**

UML merupakan singkatan dari Unified Modeling Language yang bisa berarti bahasa pemodelan standar. Menurut Mulyani (2016:48) UML (Unified Modeling Language) merupakan “Sebuah Teknik pada pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk mendokumentasikan dan melakukan spesifikasi pada sistem.”

Menurut A.s & Shalahuddin (2016:133) mendefinisikan bahwa “UML (Unified Modeling Language) adalah suatu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”. Berikut design sistem yang penulis butuhkan :



##### **2.7.1.2.1.1 Use Case Diagram**


Menurut Rachmaniah (2018:75) “Use Case Diagram berisi kebutuhan bisnis dari sistem dan juga dapat mengilustrasikan interaksi antara sistem dengan lingkungannya. Use Case Diagram juga merupakan ikhtisari dalam bentuk grafis dari aktor - aktor yang terlibat dalam sistem.”

Menurut A.s & Shalahuddin (2016:155) mendefinisikan bahwa “Use case atau diagram use case adalah pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”.

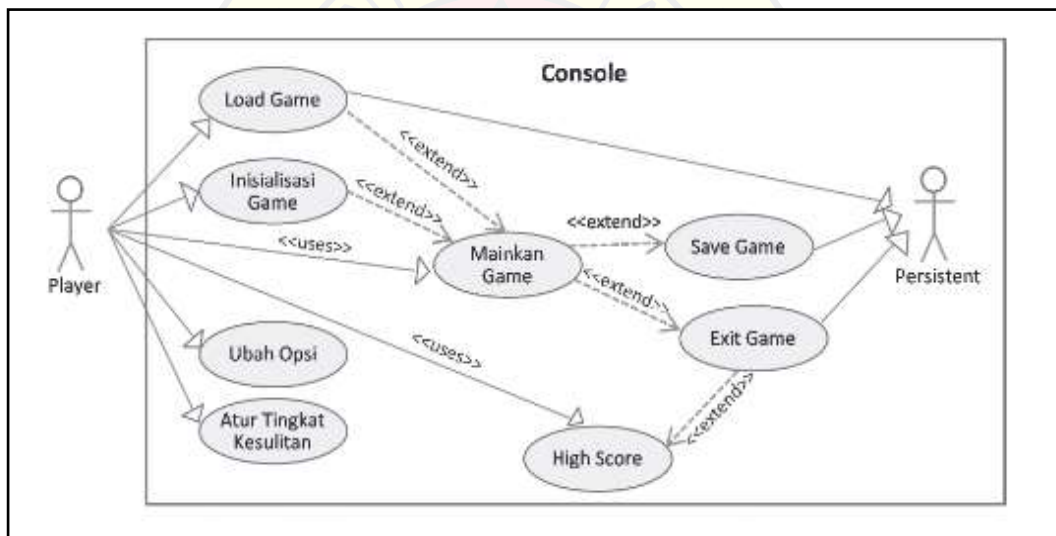
Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut ini merupakan komponen - komponen use case diagram.

Tabel 2.1. Sintaks Elemen - Elemen Use Case Diagram (Rachmaniah, 2018)

Terminologi atau Definisi	Simbol
<p>Aktor</p> <ul style="list-style-type: none"> <li>• Orang atau sistem yang memperoleh manfaat dari sistem dan bersifat eksternal dari sistem.</li> <li>• Diberi label sesuai perannya.</li> <li>• Bisa berasosiasi dengan aktor lain dengan cara <i>specialization / association</i>, dicirikan dengan tanda panah ujung yang kosong (<i>hollow arrow head</i>).</li> <li>• Ditempatkan di luar system boundary.</li> </ul>	 <p>Actor1</p>
<p>Use Case</p> <ul style="list-style-type: none"> <li>• Dapat mempresentasikan suatu fungsional dari sistem.</li> <li>• Bisa diperluas ke use case lain.</li> </ul>	 <p>UseCase1</p>

<ul style="list-style-type: none"> <li>Bisa menggunakan use case lain.</li> </ul>	
<p>Association Relationship</p> <ul style="list-style-type: none"> <li>Dapat menghubungkan aktor dengan use case di mana aktor tersebut berinteraksi dengan use case tersebut.</li> </ul>	

Contoh Use Case Diagram seperti pada gambar di bawah ini :



Gambar 2.3. Contoh Use Case Diagram (Rachmaniah, 2018)

### 2.7.1.2.1.2 Activity Diagram

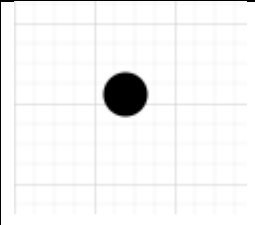
Menurut Rachmaniah (2018:81) “Activity Diagram yaitu workflow secara grafis untuk mengilustrasikan sebuah alur bisnis atau workflow operasional dari komponen yang terdapat pada sistem.”

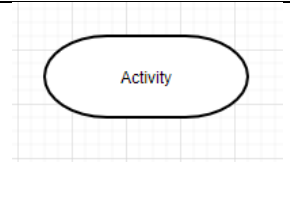
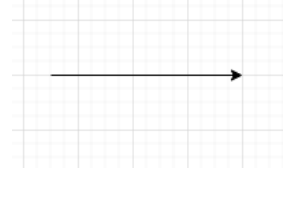
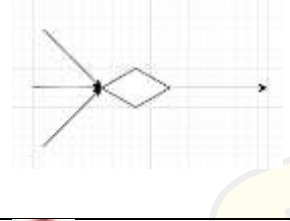

Sedangkan menurut A.s & Shalahuddin (2016:161) mendefinisikan bahwa “Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. perlu diperhatikan, diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas apa saja yang dapat dilakukan oleh sistem.”

Sedangkan menurut Rahayu, dkk (2015 : 8) Activity Diagram merupakan “bentuk visual dari alur kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, atau pengulangan.”

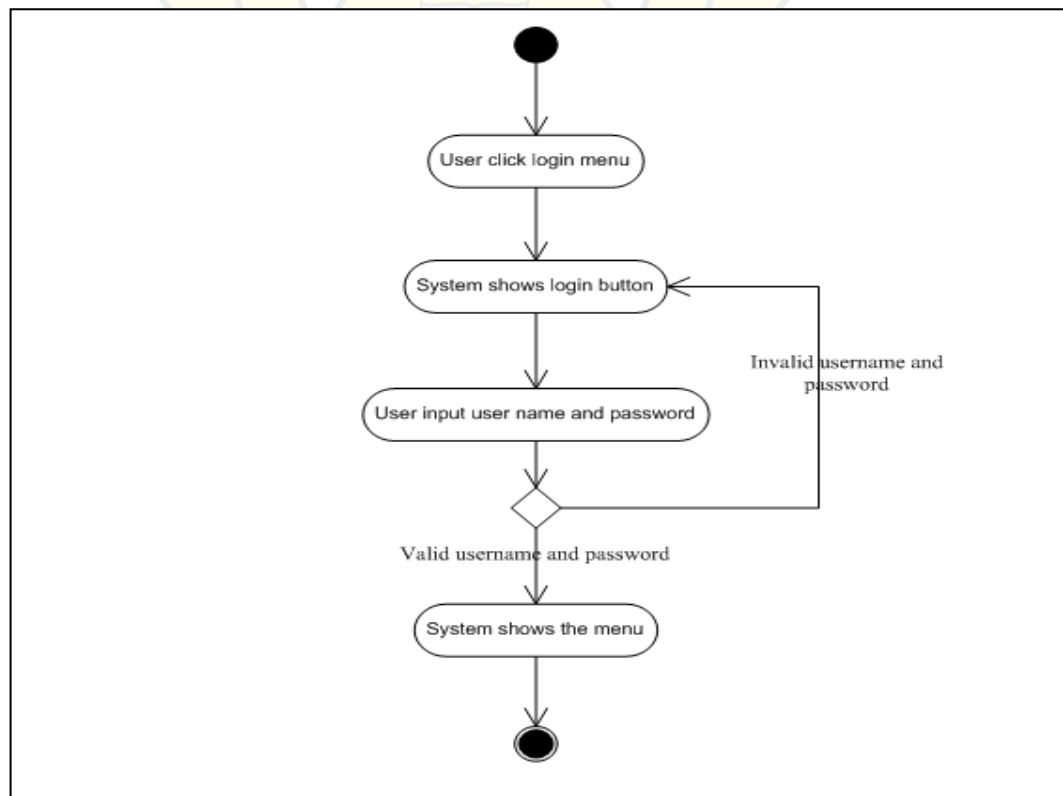
Diagram aktivitas memiliki komponen - komponen dengan bentuk yang tertentu, dihubungkan dengan tanda panah. Panah tersebut juga mengarahkan urutan aktivitas yang terjadi, dari awal sampai akhir. Yang perlu diperhatikan adalah diagram aktivitas bukan menggambarkan aktivitas - aktivitas sistem yang dilakukan oleh aktor, tetapi menggambarkan aktivitas apa saja yang dapat dilakukan oleh sistem. Berikut ini merupakan komponen - komponen pada Activity Diagram.

Tabel 2.2. Komponen - Komponen Activity Diagram (A.s dan Shalahuddin, 2016)

No	Gambar	Nama	Keterangan
1		Start Point	Start Poin / Initial State adalah sebuah lingkaran hitam kecil, yang dapat menandakan bahwa titik awal aktivitas untuk setiap diagram aktivitas.

2		Activity	Activity menunjukkan aktivitas yang sedang dilakukan / sedang terjadi dalam activity diagram.
3		Action / Flow	Action Flow digunakan untuk melakukan transisi dari suatu tindakan ke tindakan yang lain.
4		Merge Event	Merge Event digunakan untuk menggabungkan flow / action yang dipecah oleh decision.
5		Final State	Final State menunjukkan bagian akhir dari aktivitas.

Contoh Activity Diagram seperti pada gambar di bawah ini :



Gambar 2.4. Contoh Activity Diagram (A.s dan Shalahuddin, 2016)

### **2.7.1.2.1.3 Sequence Diagram**

Menurut Rahmaniah (2018:82) “Sequence Diagram memperlihatkan bagaimana sebuah objek berinteraksi satu dengan lainnya disertai urutan terjadinya interaksi tersebut serta difokuskan pada sebuah *message interchange* antar lifeline (objek).”

Sedangkan Menurut A.s & Shalahuddin (2016:165) yaitu “Diagram sequence menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup sebuah objek dan message yang dikirimkan dan diterima antar objek”. Sequence Diagram dapat menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Diagram sequence merupakan salah satu yang menjelaskan bagaimana suatu operasi itu dapat dilakukan, message (pesan) apa saja yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Objek-objek yang saling berkaitan dengan proses berjalannya operasi dapat diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.

### **2.7.1.3 Implementation & Testing**

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*, yang terintegrasi dalam tahap selanjutnya. Setiap *unit* dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*. Selain itu pada tahap ini juga dilakukan pemeriksaan terhadap modul yang dibuat apakah sudah memenuhi fungsi yang diinginkan atau belum.



### **2.7.1.3.1 Website**

Menurut Abdulloh (2018:1) “Website dapat diartikan sebagai sebuah kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara, video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses serta dilihat oleh semua orang di seluruh dunia.”

Secara umum, website dapat dibagi menjadi 3 jenis, yaitu website statis, dinamis, dan interaktif. Website statis adalah suatu jenis website yang isinya tidak diperbaharui secara berkala, sehingga isinya dari waktu ke waktu akan selalu tetap. Website dinamis yaitu jenis website yang isinya terus diperbaharui secara berkala oleh pengelola web atau pemilik website. Website interaktif yaitu isi informasinya tidak hanya diubah oleh pengelola website tetapi juga lebih banyak dilakukan oleh para pengguna website itu sendiri.

### **2.7.1.3.2 HTML**

Menurut Abdulloh (2018:7) “HTML singkatan dari Hypertext Markup Language yaitu suatu bahasa standar web yang dikelola oleh W3C (World Wide Web Consortium) berupa tag - tag yang menyusun setiap elemen dari website.”

HTML terdiri atas unsur - unsur yang membentuk struktur skrip HTML, yaitu tag, atribut, dan elemen. Tag merupakan symbol khusus (markup) berupa dua karakter “<” dan “>” yang mengapit suatu teks sebagai nama tag. Atribut adalah property yang mengatur bagaimana sebuah elemen dari suatu tag akan ditampilkan. Elemen adalah bagian dari skrip HTML yang terdiri dari tag pembuka, isi elemen, dan tag penutup.

### **2.7.1.3.3 CSS**

Menurut Pamungkas (2017:32) “Cascading Style Sheet (CSS) adalah suatu file yang berisi rangkaian intruksi untuk mengatur komponen dalam sebuah halaman web sehingga akan lebih terstruktur dan rapi.”

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran border, warna border, warna hyperlink, warna mouse over, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya.

Cara penulisan CSS dibedakan menjadi 3 macam yaitu inline, internal, dan external. Ketiganya dapat digunakan sesuai dengan kebutuhan. Inline CSS yaitu menuliskan CSS dengan menggunakan atribut style yang langsung dituliskan di dalam tag HTML. Internal CSS yaitu menuliskan CSS menggunakan tag `<style> ... </style>`. External CSS yaitu menuliskan CSS dimana skrip CSS disimpan dalam file tersendiri dengan ekstensi `.css` dan terpisah dengan file HTML.

### **2.7.1.3.4 JavaScript**

Menurut Agusriandi (2018:128) “JavaScript adalah bahasa script yang disisipkan pada kode HTML dan diproses pada sisi klien yaitu yang sering disebut client side. Bahasa ini menjadikan dokumen HTML menjadi semakin luas. Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk mem-validasi data - data yang dimasukkan pada formulis sebelum data tersebut dikirimkan ke server.”

#### **2.7.1.3.5 PHP**

Menurut Abdulloh (2018:127) “PHP merupakan singkatan Hypertext Preprocessor yaitu sebuah bahasa pemrograman web yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server. Tujuan dari bahasa ini adalah membantu para pengembangan web untuk membuat web dinamis dengan cepat.”

Skrip PHP dituliskan di antara tanda `<?php` dan `?>` yang memisahkan skrip PHP dengan skrip lainnya. Satu file PHP dapat berisi full skrip PHP atau dapat disisipkan diantara skrip lain seperti HTML, CSS maupun JavaScript. File yang berisi skrip PHP wajib disimpan dengan ekstensi `*.php` dan disimpan pada server (folder `htdocs` atau `www`). Jika disimpan dengan ekstensi HTML atau disimpan sembarang tempat maka skrip php tidak diproses sebagaimana mestinya.

Setiap baris skrip PHP harus diakhiri dengan tanda semicolon yaitu `(;)`. Jika tidak, maka akan menampilkan pesan error.

#### **2.7.1.3.6 Bootstrap**

Menurut Abdulloh (2018:261) “Bootstrap yaitu salah satu framework CSS paling populer dari sekian banyak framework css yang ada. Bootstrap juga memungkinkan desain sebuah web menjadi responsive sehingga dapat dilihat dari berbagai macam ukuran devices dengan tampilan tetap menarik. Bootstrap juga membuat proses pada pengaturan desain menjadi lebih cepat karena tidak perlu lagi banyak menulis skrip CSS, bahkan hampir tidak perlu kecuali jika memerlukan pengaturan desain yang berbeda dengan style Bootstrap. Bootstrap telah didukung oleh hampir semua browser baik pada desktop maupun mobile.”

#### **2.7.1.3.7 JQuery**

Menurut Setiawan (2017:210) “Jquery merupakan sebuah library pada javascript. Dalam dunia pemrograman, library adalah kumpulan dari berbagai jenis fungsi yang ‘siap pakai’ untuk memudahkan pembuatan sebuah aplikasi. Dengan demikian, jquery adalah kumpulan fungsi - fungsi javascript yang memudahkan penulisan kode javascript.”

#### **2.7.1.3.8 Ajax**

Asynchronous JavaScript And XML, atau disingkat Ajax, adalah suatu teknik pada pemrograman berbasis web untuk menciptakan aplikasi web interaktif. Tujuannya adalah untuk memindahkan sebagian besar interaksi pada komputer web server, melakukan pertukaran data dengan server di belakang layar, sehingga pada halaman web tidak harus dibaca berulang - ulang secara keseluruhan setiap kali seorang pengguna melakukan perubahan. Hal ini akan meningkatkan interaktivitas, kecepatan, serta usability.

#### **2.7.1.3.9 Database**

Menurut Mandar (2017:25) “Database atau basis data merupakan suatu kumpulan data yang terhubung dan disimpan secara bersama - sama pada suatu media tanpa adanya suatu kerangkapan data, sehingga mudah saat digunakan kembali serta tidak mengalami suatu ketergantungan pada program yang akan menggunakannya dan dapat diakses oleh satu atau lebih program aplikasi secara optimal.”

#### **2.7.1.3.10 MySQL**

Menurut Penerbit Andi (2016:2) “MySQL adalah sistem manajemen database SQL yang bersifat Open Source serta paling populer saat ini. Sistem Database MySQL mendukung beberapa fitur seperti multithreaded, multi-user dan SQL database management system (DBMS). Database ini dibuat untuk keperluan sistem database yang cepat, handal, dan mudah digunakan.”

#### **2.7.1.3.11 PhpMyAdmin**

Menurut Mandar (2017:133) “PhpMyAdmin merupakan sebuah software manajemen database berbasis antar muka sehingga tidak perlu lagi mengetikkan perintah - perintah sql dalam mengolah database.”

PhpMyAdmin menjadi salah satu pilihan utama bagi para pengembang web sebab sudah tersedia dalam paket WampServer, XAMPP, atau aplikasi server local lainnya sehingga tidak perlu lagi menginstallnya secara terpisah.

#### **2.7.1.3.12 Xampp**

Menurut Aryanto (2016:5) “XAMPP adalah sebuah aplikasi perangkat lunak pemrograman dan database yang di dalamnya terdapat berbagai macam aplikasi pemrograman seperti; Apache HTTP Server, MySql Database, bahasa pemrograman PHP dan Perl.”

#### **2.7.1.4 Verification (Verifikasi)**

Menurut Pressman (2015:67) “Tahapan Verification yaitu meliputi pengintegrasian sistem dan juga melakukan testing terhadap aplikasi yang telah dibuat. Sistem akan diverifikasi untuk diuji sejauh mana kelayakannya.”

Dalam tahapan ini semua modul yang dikerjakan oleh programmer yang berbeda akan digabungkan kemudian diuji apakah telah sesuai dengan spesifikasi yang ditetapkan atau terdapat kesalahan / error dalam sistem sebelum kemudian diperbaiki ulang.

#### **2.7.1.5 Maintance (Pemeliharaan)**

Menurut Pressman (2015:67) “Tahap ini merupakan tahap terakhir pada waterfall model. Software yang telah dibuat akan dilakukan maintenance. Maintenance termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.”

Sedangkan menurut Sugiyono (2015:338) “Pemeliharaan merupakan sebuah operasi atau aktivitas yang harus dilakukan secara berkala dengan tujuan untuk mempercepat pergantian kerusakan peralatan dengan resources yang ada. Perawatan juga ditujukan untuk mengembalikan suatu sistem pada kondisinya agar dapat berfungsi, memperpanjang usia kegunaan mesin dan menekan failure sekecil mungkin.”