

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **1.1 Sejarah Desa Paku Alam**

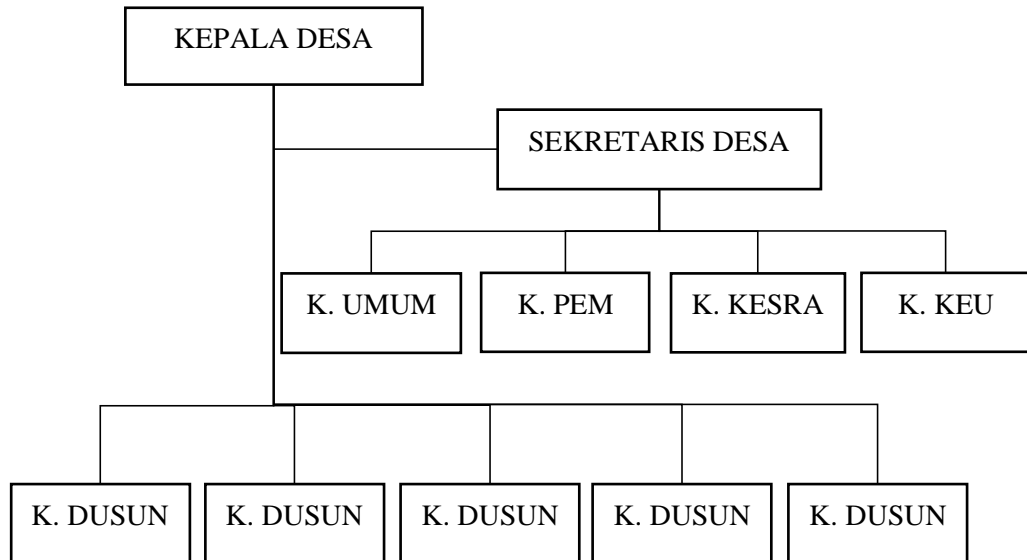
Desa Paku Alam merupakan sebuah desa yang berada di wilayah Kecamatan Darmaraja. Letaknya berada di bagian paling utara wilayah Kecamatan dan berbatasan langsung dengan Kecamatan Cisitu di bagian utara dan Kecamatan Jatigede di bagian timurnya. Dilihat dari posisinya, wilayah Desa Paku Alam berada di sebelah barat daya bendungan Jatigede dan bagian timurnya berbatasan langsung dengan Sungai Cimanuk yang berada di genangan bendungan Jatigede.

Berdasarkan data Potensi Desa 2013 yang dikutip oleh BPS Kabupaten Sumedang, Desa Paku Alam memiliki status sebagai pedesaan dengan klasifikasi sebagai desa swakarsa. Secara topografis, wilayah Desa Paku Alam memiliki bentuk bentang permukaan tanah berupa pedataran dengan ketinggian wilayah tempat kantor desa berada sekitar 258 meter di atas permukaan laut. Secara geografis, wilayah Desa Paku Alam dikelilingi oleh wilayah-wilayah sebagai berikut: Desa Cigintung dan Desa Pajagan (kedua berada di Kecamatan Cisitu) di sebelah utaranya, Desa Jemah Kecamatan Jatigede dan Desa Cipaku di sebelah timurnya, Desa Cipaku dan Desa Karang Pakuan di sebelah selatan dan Desa Cisitu Kecamatan Cisitu di sebelah baratnya. Secara administratif, wilayah Desa Paku Alam terdiri atas dua Dusun, empat Kampung dan delapan Rukun Warga serta 24 Rukun Tetangga. Dusun yang pertama yaitu Dusun Cilembu yang terbagi menjadi RW 01 – RW 05 serta RT 01 – RT 17. Dusun keduanya adalah Dusun Cisema yang terbagi menjadi RW 06 – RW 08, terdiri dari RT 18 – RT 24.

Berdasarkan data yang disajikan blog KKNM Unpad di Paku Alam tahun 2013, luas wilayah Desa Paku Alam sebesar 477 hektar. Luas wilayah tersebut terbagi ke dalam beberapa peruntukan yaitu digunakan sebagai lahan pesawahan seluas 150 hektar. Yang lainnya digunakan sebagai lahan perkebunan atau ladang seluas 99 hektar, tanah negara seluas 160 hektar. Sisanya merupakan lahan pekarangan dan pemukiman seluas 39 hektar dan lahan pangangan seluas 27 hektar. Luas

pesawahannya sendiri terbagi ke dalam dua jenis yaitu yang diairi dengan menggunakan irigasi seluas 120 hektar.[9]

### 1.2 Struktur Organisasi



Berikut adalah tugas dan wewenang dari aparatur desa :

#### 1. Kepala Desa

Kepala desa atau yang biasa disebut dengan kades memiliki tugas pokok dan fungsi sebagai berikut :

- a. Menjalankan roda pemerintahan desa dengan dasar kebijakan yang telah ditetapkan bersama dengan BPD (Badan Perwakilan Desa).
- b. Mengajukan suatu rancangan tentang peraturan yang akan diterapkan di suatu desa.
- c. Menetapkan peraturan desa yang sudah disetujui bersama dengan BPD (Badan Perwakilan Desa).
- d. Menyusun serta membuat peraturan tentang anggaran pendapatan desa yang selanjutnya akan dibahas dan ditetapkan bersama-sama dengan BPD (Badan Perwakilan Desa).
- e. Melakukan pembinaan terhadap masyarakat desa dan juga ekonomi desa.
- f. Pembangunan yang hendak dilakukan di desa, lebih dahulu dikoordinasikan dan dilaksanakan dengan partisipasi semua warga

- g. Mewakili desa baik di luar pengadilan atau di dalam pengadilan serta memiliki hak menunjuk kuasa hukum sebagai wakil dirinya, tentunya sesuai dengan perundang-undangan yang berlaku.
- h. Melakukan kewajiban dan wewenang kepala desa sesuai dengan peraturan yang berlaku.

## 2. Sekretaris Desa

Tugas pokok dari sekretaris desa antara lain Membantu persiapan kepala desa dan melakukan kegiatan administrasi desa, Menyiapkan bahan untuk menyusun laporan penyelenggaraan pemerintahan desa.

Fungsi dari sekretaris desa antara lain :

- a. Melaksanakan beraneka macam tugas administrasi dan menyiapkan keperluan kepala desa supaya tugasnya berjalan dengan lancar
- b. Mempersiapkan bantuan dalam melaksanakan penyusunan peraturan desa
- c. Mempersiapkan bahan laporan penyelenggaraan pemerintahan desa
- d. Melakukan tugas lain yang diperintahkan oleh kepala desa.

## 3. Kepala Urusan Umum

Kepala urusan umum atau yang disebut dengan kaur umum ini tugas pokoknya yaitu membantu sekretaris desa dalam melaksanakan tugas administrasi umum, kearsipan, tata usaha, inventaris desa, dan menyiapkan segala bahan untuk rapat dan pembuatan laporan.

Fungsi kepala urusan umum antara lain :

- a. Pengendalian kearsipan dan surat masuk serta surat keluar
- b. Mencatat inventaris atau kekayaan desa
- c. Melakukan tugas administrasi umum
- d. Menyimpan, menyediakan, dan menyalurkan alat-alat tulis kantor serta bertanggungjawab dalam memelihara dan perbaikan terhadap perabot atau perlengkapan kantor.
- e. Melaksanakan pengelolaan administrasi perangkat desa
- f. Menyiapkan bahan untuk membuat laporan

- g. Melakukan tugas lain yang diperintahkan oleh sekretaris desa.

#### 4. Kepala Urusan Keuangan

Kepala urusan keuangan atau yang dikenal dengan kaur keuangan ini tugas pokoknya antara lain membantu tugas sekretaris desa dalam mengelola sumber penghasilan desa, administrasi keuangan desa, persiapan bahan untuk membuat APB (Anggaran Pendapatan dan Belanja) Desa.

Fungsi kepala urusan keuangan antara lain :

- a. Mengelola administrasi keuangan desa
- b. Mempersiapkan bahan untuk menyusun APB (Anggaran Pendapatan dan Belanja) Desa.
- c. Melakukan tugas lain yang diberikan atau diperintahkan oleh sekretaris desa.

#### 5. Kepala Urusan Pemerintah

Tugas pokok dari kaur pemerintahan antara lain membantu Kepala desa untuk melakukan pengelolaan administrasi kependudukan, administrasi pertanahan, pembinaan, ketertiban masyarakat suatu desa, persiapan bahan dalam pelaksanaan kebijakan seperti kebijakan penataan dan kebijakan hukum desa.

Fungsi dari kepala urusan pemerintahan antara lain :

- a. Melaksanakan administrasi kependudukan
- b. Mempersiapkan bahan untuk menyusun rancangan peraturan serta keputusan kepala desa
- c. Melaksanakan administrasi pertanahan
- d. Mencatat monografi desa
- e. Mempersiapkan bantuan yang bertujuan untuk memperlancar penyelenggaraan dalam pemeritahan di desa dan melakukan penataan kelembagaan masyarakat
- f. Melakukan tugas lain yang diberikan atau diperintahkan oleh kepala desa.

#### 6. Kepala Urusan Kesejahteraan Rakyat

Disingkat dengan kaur kesra, memiliki tugas Membantu kepala desa dalam mempersiapkan bahan perumusan kebijakan teknis dalam hal memberdayakan masyarakat dan sosial kemasyarakatan.

Fungsi kaur kesra antara lain :

- a. Mempersiapkan bahan yang akan dipakai dalam program keagamaan.
- b. Mempersipakan bahan yang akan dipakai dalam program pemberdayaan masyarakat dan sosial kemasyarakatan.
- c. Mempersiapkan bahan yang akan dipakai dalam perkembangan kehidupan beragama.
- d. Melakukan tugas lain yang diberikan atau diperintahkan oleh kepala desa.

#### 7. Kepala Dusun

Kepala dusun atau dukuh memiliki tugas sebagai berikut :

- a. Membantu melaksanakan tugas dari kepala desa yang masih berada dalam wilayah kerjanya.
- b. Melakukan sosialisasi program-program pemerintah kepada masyarakat.
- c. Membantu tugas kepala desa dalam membina serta koordinasi kegiatan RT (Rukun Tetangga) maupun RW (Rukun Warga) yang masih berada dalam wilayah kerjanya.
- d. Melakukan tugas lain yang diperintahkan oleh kepala desa.

Fungsi Kepala Dusun antara lain :

- a. Melaksanakan koordinasi pembangunan desa, jalannya pemerintahan desa, serta membina masyarakat yang ada di dusun.
- b. Melaksanakan pembinaan-pembinaan terhadap masyarakat dan tugas yang berhubungan dengan pembangunan atau melaksanakan koordinasi perihal masalah pembangunan yang terjadi di desa maupun di dusun.
- c. Berusaha untuk terus meningkatkan rasa kebersamaan dan gotong royong sesama warga dengan kata lain meningkatkan partisipasi masyarakat.
- d. Melakukan usaha dalam rangka memelihara ketentraman dan ketertiban masyarakat.

- e. Melaksanakan beragam fungsi yang lain yang telah dilimpahkan oleh desa.

### 1.3 Lambang Kabupaten Sumedang

Lambang Kabupaten Sumedang diciptakan oleh R. Maharmartanagara, yang merupakan seorang putra Bupati Bandung Raden Adipati Aria Martanegara sebagai keturunan Sumedang. Lambang ini diresmikan menjadi Lambang Kabupaten Sumedang pada tanggal 13 Mei 1959.



**Gambar 2.3 Lambang Kabupaten Sumedang**

Komponen yang terdapat dalam logo atau lambang tersebut:

1. Perisai  
Melambangkan jiwa kesatria utama, percaya pada diri sendiri.
2. Sisi Merah  
Melambangkan semangat keberanian
3. Dasar Hijau  
Melambangkan kesuburan pertanian
4. Bentuk Setengah Bola dan Setengah Kubus pada Lingga  
Melambangkan bahwa manusia tidak ada yang sempurna.
5. Sinar Matahari  
Melambangkan semangat rakyat dalam mencapai kemajuan
6. Warna Kuning Emas  
Melambangkan keluhuran budi dan kebesaran jiwa.
7. Sinar yang berjumlah 17  
Melambangkan Angka Sakti tanggal Proklamasi Kemerdekaan RI

8. Delapan Bentuk pada Lingga

Lambang Bulan Proklamasi Kemerdekaan Republik Indonesia

9. Batu pada Tangga: 4 buah Kaki Tembik dan 5 buah Anak Tangga

Lambang Tahun Proklamasi Kemerdekaan RI Tahun 1945

10. Tulisan Insun Medal

Tulisan Insun Medal erat kaitannya dengan kata Sumedang.[2]

#### 1.4 Landasan Teori

Landasan teori merupakan penjelasan berbagai konsep dasar dan teori – teori yang berkaitan dengan pembangunan sistem di Desa Paku Alam. Penjelasan secara umum dari perangkat lunak, perangkat keras dan bahasa pemrograman yang diperlukan.

#### 1.5 Promosi

Menurut Menurut Djaslin Saladin S.E. dan Yevis Marty Oesman SE (1997:194), "bauran promosi adalah kombinasi dari penjualan tatap muka, periklanan, promosi penjualan, publisitas dan hubungan yang membantu pencapaian tujuan perusahaan. Alat promosi yang sering digunakan dalam mengkomunikasikan produk (Saladin, Djaslim. 1996) adalah :

- a. Periklanan (*Advertising*) Semua bentuk penyajian dan promosi nonpersonal atas ide, barang atau jasa yang dilakukan oleh perusahaan sponsor tertentu. Karena banyaknya bentuk dan penggunaan periklanan, sangat sulit untuk membuat generalisasi yang merangkum semuanya.
- b. Penjualan Personal (*Personal Selling*) Interaksi langsung dengan satu calon pembeli atau lebih, guna melakukan presentasi, menjawab pertanyaan, dan menerima pesanan.
- c. Promosi Penjualan (*Sales Promotion*) Berbagai insentif jangka pendek untuk mendorong keinginan mencoba atau membeli suatu produk atau jasa.
- d. Hubungan Masyarakat dan Publisitas (*Public Relation and Publicity*) Berbagai program untuk mempromosikan dan/atau melindungi citra perusahaan atau masing-masing produknya.[10]

Menurut kemendag.go.id promosi adalah kegiatan, mempertunjukan, memperagakan, memperkenalkan, dan / atau menyebar luaskan informasi mengenai Produk Dalam Negeri untuk menarik minat masyarakat dan pelaku usaha agar meningkatkan penggunaan produk dalam negeri dan pemasaran adalah kegiatan memasarkan perdagangan Produk Dalam Negeri di dalam negeri. Promosi sebagaimana dimaksud dalam pasal 2 ayat (1) huruf a dilaksanakan paling sedikit dalam bentuk:

- a. Penyelenggaraan pameran Produk Dalam Negeri antara lain melalui kegiatan pameran dagang, pameran sandang dan pangan, pameran budaya, dan pameran pembangunan.
- b. Penyediaan fasilitas ruang Promosi secara permanen sebagai sarana untuk mempromosikan Produk Dalam Negeri yang potensial; dan / atau. Penggunaan Produk Dalam Negeri pada acara pemerintahan dan / atau acara tertentu secara lokal, nasional maupun internasional. Pemasaran sebagaimana di maksud dalam pasal 2 ayat (1) huruf c dilaksanakan paling sedikit dalam bentuk: Pengembangan kemitraan usaha antara pelaku usaha menengah atau besar dengan pelaku usaha mikro dan kecil melalui temu usaha, forum dagang, dan misi dagang lokal.
- c. Peningkatan pemasaran prosuk usaha mikro dan kecil melalui private label atau house brand; dan / atau.
- d. Fasilitas mengikuti pameran baik pameran mandiri dan pameran partisipasi.[11]

## **2.6 Aplikasi berbasis Website**

Dengan berkembangnya internet, situs yang ada di internet tidak hanya berfungsi untuk mempresentasikan content tetapi cenderung berupa aplikasi yang kebanyakan terhubung ke suatu basis data. Pada tahapan ini situs akan bersifat dinamis, karena *content* yang dipresentasikan akan bervariasi dan berubah-ubah sesuai dengan data yang diminta dan action dari user. Untuk mengembangkan situs yang dinamis diperlukan teknologi server side seperti *PHP*, *Webgl*, *ASP*, *Perl* dan *CGI* yang lain. Dengan teknologi server side kita dapat mengembangkan suatu



aplikasi berbasis internet yang dapat menghasilkan dan menampilkan content secara dinamis.

## **2.7 Virtual Tour**

Virtual Tour Adalah simulasi dari sebuah lokasi yang sesungguhnya, umumnya terdiri dari sequence video atau kumpulan foto. Virtual Tour juga dapat menggunakan beberapa elemen multimedia lain, contohnya seperti sound effect, musik, narasi, dan teks. Ungkapan “*virtual tour*” sering digunakan untuk berbagai video dan media fotografi. Panorama menunjukkan pandangan tak terputus, karena panorama dapat berupa serangkaian foto-foto atau rekaman video panning. Namun, “tour panorama” dan “*virtual tour*” sebaigan besar telah dikaitkan dengan wisata yang telah dibuat dengan kamera statis ataupun yang terbaru sekarang kamera 360°.

### 1. Virtual Tour 360°

Virtual Tour 360° adalah hasil dari pengolahan foto digital yang berbentuk foto panorama. Foto panorama tersebut kemudian didevelop untuk dijadikan software Virtual.

### 2. Manfaat Virtual Tour 360°

- a. Sebagai media promosi Online (media interaktif virtual tour meyakinkan calon pengunjung lokasi mengekspose fasilitas lokasi secara detail sebagai dokumentasi lokasi secara 360°).
- b. Menjadi inspirator dalam proses membangun sebuah usaha / bisnis.
- c. Terciptanya sebuah konsep produk dan jasa yang dapat menjembatani kebutuhan informasi serta promosi ditengah kemajuan teknologi yang semakin pesat. [6]

## **2.8 HTML (*Hyper Text Markup Language*)**

Pengertian HTML (*Hyper Text Markup Language*) menurut para ahli antara lain :

### 1. Lia Kuswayatno

HTML merupakan halaman yang berada pada suatu situs internet atau web. HTML merupakan metode yang menautkan (*link*) satu dokumen ke dokumen lain melalui teks.

## 2. Deris Stiawan

HTML merupakan framework internet, hampir semua situs web yang ada menggunakan HTML untuk menampilkan teks, grafik, suara, dan animasinya.

## 3. Diar Puji Oktavian

HTML (*Hyper Text Markup Language*) adalah suatu bahasa yang dikenali oleh web browser untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa (*plain text*). Berdasarkan pengertian HTML menurut para ahli di atas dan penjelasan dapat disimpulkan bahwa HTML (*Hyper Text Markup Language*) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah penjelajah web Internet dan pemformatan *hiperteks* sederhana yang ditulis dalam berkas format *ASCII* (*American Standard Code for Information Interchange*) agar dapat menghasilkan tampilan wujud yang terintegrasikan. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format *ASCII* (*American Standard Code for Information Interchange*) normal sehingga menjadi halaman web dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan *SGML* (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium* (*W3C*). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa).[3]

## 2.9 Javascript

Menurut Yeni Kustiyahningsih dan Devie Rosa Anamisa, 2011 *JavaScript* adalah bahasa yang berbentuk kumpulan skrip berjalan pada suatu dokumen

HTML. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap HTML dengan mengizinkan pengeksekusian perintah-perintah disisi user artinya disisi *browser* bukan disisi server web. *JavaScript* adalah bahasa yang “*case sensitive*” artinya memnedakan penamaan variabel dan fungsi yang menggunakan huruf besar dan huruf kecil, contoh variabel atau fungsi dengan nama TEST berbeda dengan variabel dengan nama test dan setiap instruksi diakhiri dengan karakter titik koma.

### 2.10 Pannelum

Pannelum adalah Sebuah plug-in open source *viewer* panorama yang ringan untuk platform website yang didalamnya terdapat hotspot yang mampu menampilkan informasi, menghubungkan beberapa panorama dalam virtual tour, dan dukungan video. Plug-in ini mendukung *developer* untuk melakukan *customize* terhadap fitur yang diperlukan untuk mengembangkan aplikasi *virtual tour* yang lebih interaktif. [12]

### 2.11 Multimedia

Menurut Steinmetz (1995) Multimedia adalah gabungan dari seminimalnya sebuah media diskrit dan sebuah media kontinu. Media diskrit adalah sebuah media dimana validitas datanya tidak tergantung dari kondisi waktu, termasuk didalamnya teks dan grafik. Sedangkan yang dimaksud dengan media kontinu adalah sebuah media dimana validitas datanya tergantung dari kondisi waktu, termasuk di dalamnya suara dan video.[4].

Multimedia dari kata ‘multi’ dan ‘media’. Multi berarti banyak, dan media berarti tempat, sarana atau alat yang digunakan untuk menyampaikan informasi. Jadi multimedia dapat diartikan sebagai penggunaan beberapa media yang berbeda untuk menggabungkan dan menyampaikan informasi dalam bentuk text, audio, grafik, animasi, dan video. Sebagai salah satu komponen sistem pembelajaran, pemilihan dan penggunaan multimedia pembelajaran harus memperhatikan karakteristik komponen lain, seperti tujuan, materi, strategi, dan juga evaluasi

pembelajaran. Selain memenuhi ketiga karakteristik tersebut, multimedia pembelajaran sebaiknya memenuhi fungsi sebagai berikut:

- a. Mampu memperkuat respon pengguna secepatnya dan sesering mungkin,
- b. Mampu memberikan kesempatan kepada siswa untuk mengontrol laju kecepatan belajarnya sendiri,
- c. memperhatikan bahwa siswa mengikuti suatu urutan yang koheren dan terkendalikan,
- d. Mampu memberikan kesempatan adanya partisipasi dari pengguna dalam bentuk respon, baik berupa jawaban, pemilihan, keputusan, percobaan dan lain-lain.[6]

## **2.12 Database Relational**

Basis data adalah sebuah cara mendokumentasikan berbagai macam data yang kemudian dimanajemen dengan sebuah sistem untuk kemudian disimpan dalam sebuah media penyimpanan. Dalam basis data, data yang ada tidak hanya disimpan begitu saja dalam suatu media penyimpanan, akan tetapi dikelola dengan sistem pengaturan basis data yang sering disebut Database Management System (DBMS).

### **1. Kegunaan Basis Data**

Penyusunan satu basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu:

- a. Redundansi dan inkonsistensi data
- b. Kesulitan pengaksesan data
- c. Isolasi data untuk standarisasi
- d. *Multiple user* (banyak pemakai)
- e. Masalah keamanan (*security*)
- f. Masalah integrasi (kesatuan)
- g. Masalah data *independence* (kebebasan data)

### **2. MySQL**

MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis di bawah lisensi *GPL*

(*General Public License*). Berikut ini bahasa yang digunakan MySQL untuk pengoperasiannya.

- a. SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Table "T"	Query	Result												
<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> <tr> <td>2</td> <td>b</td> </tr> </tbody> </table>	C1	C2	1	a	2	b	<code>SELECT * FROM T;</code>	<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> <tr> <td>2</td> <td>b</td> </tr> </tbody> </table>	C1	C2	1	a	2	b
C1	C2													
1	a													
2	b													
C1	C2													
1	a													
2	b													
<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> <tr> <td>2</td> <td>b</td> </tr> </tbody> </table>	C1	C2	1	a	2	b	<code>SELECT C1 FROM T;</code>	<table border="1"> <thead> <tr> <th>C1</th> </tr> </thead> <tbody> <tr> <td>1</td> </tr> <tr> <td>2</td> </tr> </tbody> </table>	C1	1	2			
C1	C2													
1	a													
2	b													
C1														
1														
2														
<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> <tr> <td>2</td> <td>b</td> </tr> </tbody> </table>	C1	C2	1	a	2	b	<code>SELECT * FROM T WHERE C1 = 1;</code>	<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> </tbody> </table>	C1	C2	1	a		
C1	C2													
1	a													
2	b													
C1	C2													
1	a													
<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>a</td> </tr> <tr> <td>2</td> <td>b</td> </tr> </tbody> </table>	C1	C2	1	a	2	b	<code>SELECT * FROM T ORDER BY C1 DESC;</code>	<table border="1"> <thead> <tr> <th>C1</th> <th>C2</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>b</td> </tr> <tr> <td>1</td> <td>a</td> </tr> </tbody> </table>	C1	C2	2	b	1	a
C1	C2													
1	a													
2	b													
C1	C2													
2	b													
1	a													

**Gambar 2.4 syntax umum SQL.**

Dalam pembangunan sistem ini, digunakan beberapa fungsi MySQL untuk mendukung data yang saling berelasi satu sama lain seperti beberapa penggunaan fungsi *Join*, *Right Outer Join*, *Left Outer Join*, *Inner Join*, *Foreign Key*. Berikut adalah contoh syntax dari fungsi yang dipakai dalam sistem:

- a. Join

Penggunaan fungsi *Join* untuk menggabungkan 2 tabel atau lebih berdasarkan atribut (*field*) yang sama, tanpa memprioritaskan table manapun.

Contoh *syntax* :

```
SELECT * FROM TABLE_1 JOIN TABLE_2 ON TABLE_1.FIELD_1 =
TABLE_2.FIELD_2;
```

- b. Right Outer Join

Penggunaan fungsi *Right Outer Join* untuk menggabungkan 2 tabel atau lebih berdasarkan atribut (*field*) yang sama dengan memprioritaskan table yang berada di sebelah kanan.

Contoh *syntax* :

```
SELECT * FROM TABLE_1 RIGHT OUTER JOIN TABLE_2 ON
TABLE_1.FIELD_1 = TABLE_2.FIELD_2
```

c. Left Outer Join

Penggunaan fungsi *Left Outer Join* untuk menggabungkan 2 tabel atau lebih berdasarkan atribut (*field*) yang sama dengan memprioritaskan table yang berada di sebelah kiri.

Contoh *syntax* :

```
SELECT * FROM TABLE_1 LEFT OUTER JOIN TABLE_2 ON
TABLE_1.FIELD_1 = TABLE_2.FIELD_2;
```

d. Inner Join

Penggunaan fungsi *Inner Join* untuk menggabungkan 2 tabel atau lebih berdasarkan atribut (*field*) yang sama, tanpa memprioritaskan tabel manapun dan akan menampilkan data yang berada di *table 1* dan *table 2*.

Contoh *syntax* :

```
SELECT * FROM TABLE_1 INNER JOIN TABLE_2 ON TABLE_1.FIELD_1 =
TABLE_2.FIELD_2;
```

e. Foreign Key

1. Penggunaan *Foreign Key* untuk menghubungkan *table 1* dan 2 dengan fungsi *Update Cascade* untuk membuat *update* otomatis pada saat terjadi perubahan disalah satu *table*. Dan *Delete Cascade* akan berfungsi untuk melakukan menghapus data otomatis pada saat data pada *table parent* (Induk) telah didelete.

Contoh *syntax* :

```
ALTER TABLE TABLE_1 ADD FOREIGN KEY (FIELD_1) REFERENCES
TABLE_2 (FIELD_2) ON UPDATE CASCADE ON DELETE CASCADE
```

2. Penggunaan *Foreign Key* untuk menghubungkan *table 1* dan 2 dengan fungsi *Update Cascade* untuk membuat *update* otomatis pada saat terjadi perubahan disalah satu *table*. Dan *Delete Restrict* akan berfungsi untuk melakukan

menghapus data diperbolehkan jika data pada *table parent* (induk) telah dihapus.

Contoh *syntax* :

```
ALTER TABLE TABLE_1 ADD FOREIGN KEY (FIELD_1) REFERENCES
TABLE_2 (FIELD_2) ON UPDATE CASCADE ON DELETE RESTRICT
```

3. Penggunaan *Foreign Key* untuk menghubungkan *table* 1 dan 2 dengan fungsi *Update Cascade* untuk membuat *update* otomatis pada saat terjadi perubahan disalah satu *table*. Dan *Delete Set Null* akan berfungsi untuk melakukan menghapus data dan pada *table* yang telah berelasi akan diberikan nilai NULL.

Contoh *syntax* :

```
ALTER TABLE TABLE_1 ADD FOREIGN KEY (FIELD_1) REFERENCES
TABLE_2 (FIELD_2) ON UPDATE CASCADE ON DELETE SET NULL.
```

### 2.13 Object Oriented Programming (OOP)

Object Oriented Programming (OOP) adalah paradigma pemrograman berdasarkan konsep objek, yang dapat berisi data, dalam bentuk *field* atau dikenal juga sebagai atribut, serta kode, dalam bentuk fungsi/prosedur dan dikenal sebagai *method*. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya. Dalam pembangunan struktur *Object Oriented Programming* (OOP) menggunakan bahasa-bahasa seperti dibawah ini :

#### 1. *Unfied Modeling Language (UML)*

*Unfied Modeling Language (UML)* adalah sistem arsitektur yang mengganti dari metode *Analysis and Design (OOAD)*. *Unified Modeling Language (UML)* merupakan gabungan dari metode Booch, OMT, dan Jacobson. Pada pengembangan *Unified Modeling Language (UML)* dilakukan standarisasi proses dengan *OMG (Object Management Group)* dengan harapan *Unifie Modeling Language (UML)* akan menjadi bahasa pemodelan pada masa yang akan datang.

*Unified Modeling Language (UML)* disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat.

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi, jika anda ingin berdiskusi tentang desain dengan seseorang, maka anda hanya membutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendesain secara cepat. UML merupakan bahasa standar untuk penulisan *blueprint* software yang digunakan visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak.[8]

Bagian-bagian utama dari UML adalah view, diagram, model element dan *general Mechani SM*.

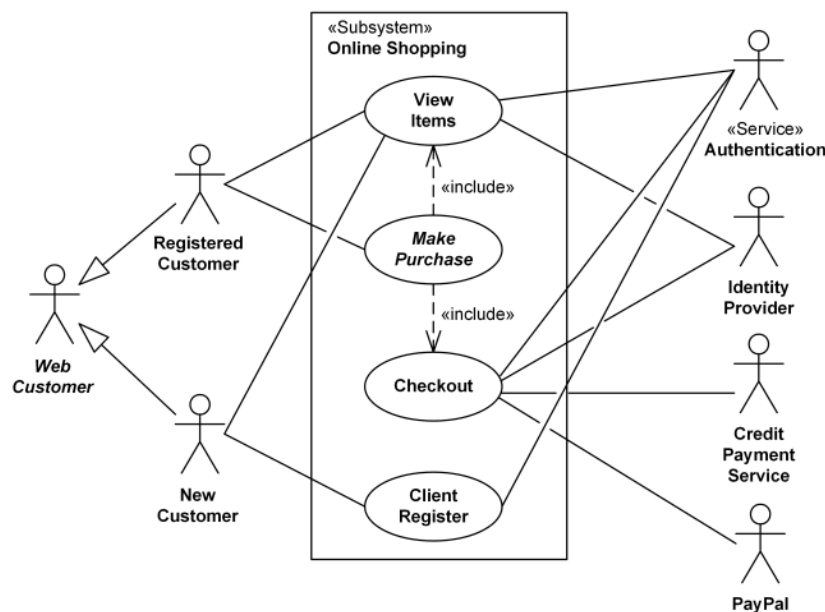
1. View

*View* digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. *View* bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram. Beberapa jenis *view* dalam UML antara lain: *use case view*, *logical view*, *component view*, *concurrency view*, dan *deployment view*.

- a. *Use Case View*

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai dengan yang diinginkan *external actors. actor* yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya. *View* ini digambarkan dalam *use case diagrams* dan kadang – kadang *activity diagrams*. *View* ini digunakan terutama untuk pelanggan, perancang (designer), pengembang (developer), penguji sistem (tester). (Seperti pada gambar dibawah ini).





**Gambar 2.5 Use Case View**

## 2. Diagram

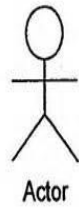
Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis diagram antara lain :

### a. *Use Case Diagram*

*Use case diagram* adalah gambaran sistem dari sudut pandang pengguna (user) sistem tersebut, sehingga pembuatan use case lebih dititik beratkan pada fungsional yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian, sebuah *use case diagram* mengandung :

#### a. Aktor

Menggambarkan seseorang yang berinteraksi dengan system, dimana hanya bisa menginputkan informasi dan menerima informasi dari system dan tidak memegang kendali pada use case. Dan actor digambarkan dengan stickman. (Seperti pada gambar dibawah ini).



**Gambar 2.6 Ikon Aktor Pada UML**

b. *Use Case*

Gambaran fungsional system yang akan dibuat, agar pengguna lebih mengerti penggunaan system. (Seperti pada gambar dibawah ini).

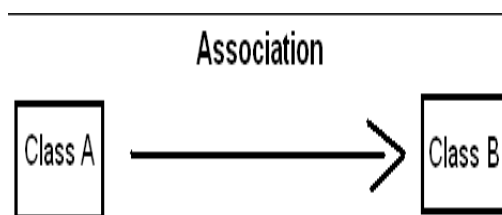


**Gambar 2.7 Ikon Use Case UML**

c. Relasi Dalam Use Case

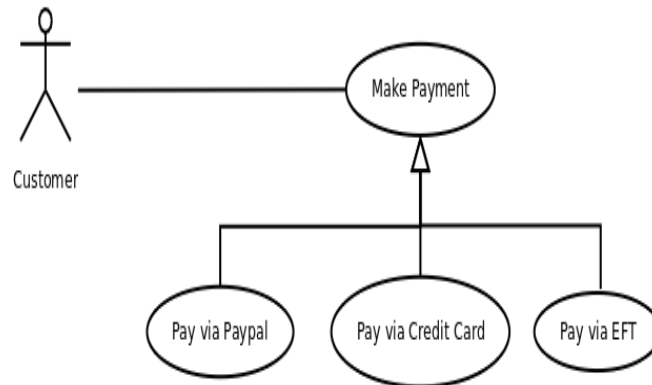
Ada beberapa relasi yang terdapat pada use case diagram :

1. Association, menghubungkan link antar elemen.



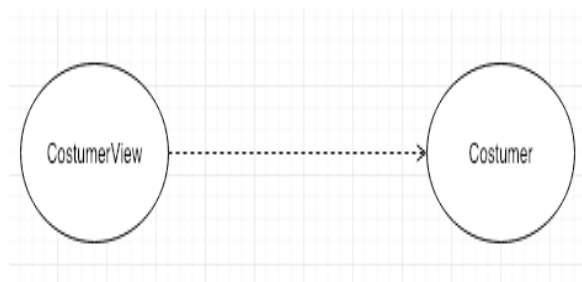
**Gambar 2.8 Relasi Association**

2. Generalization, sebuah elemen yang menjadi spesialisasi dari elemen yang lain.



**Gambar 2.9 Relasi Generalization**

3. Dependency, sebuah elemen yang bergantung beberapa cara kepada elemen lainnya.



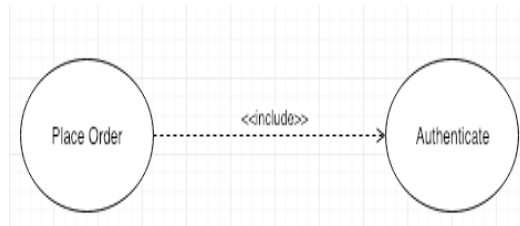
**Gambar 2.10 Relasi Dependency**

4. Aggregation, bentuk association yang dimana sebuah elemen berisi elemen lainnya.



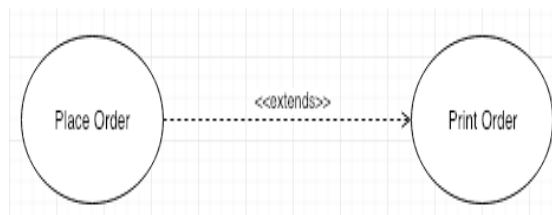
**Gambar 2.11 Relasi Aggregation**

5. <<include>>, kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, diimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.



**Gambar 2.12 Relasi <<include>>**

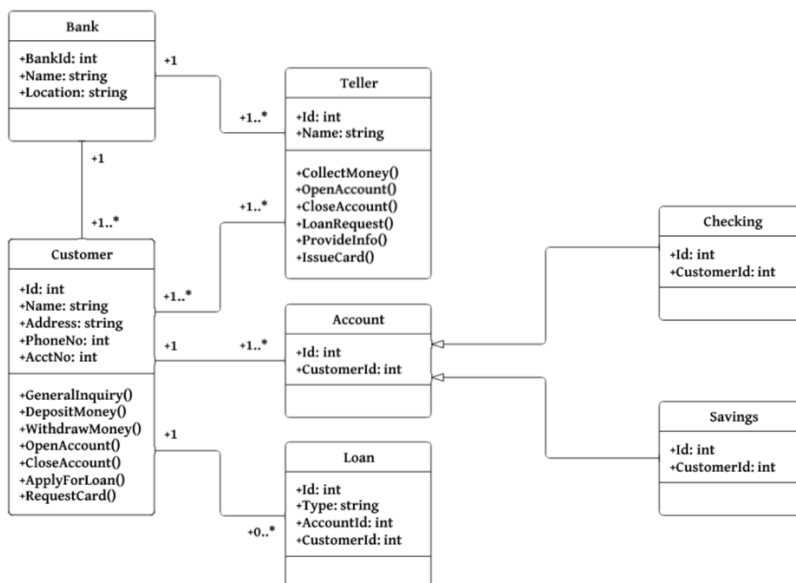
6. <<extends>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.



**Gambar 2.13 Relasi <<extends>>**

b. *Class Diagram*

*Class diagram* mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara mereka. *Class diagram* juga menunjukkan property dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tertentu. (Seperti pada gambar dibawah ini).



**Gambar 2.14 Class Diagram**

Atribut dan operasi (metoda) dapat memiliki salah satu sifat berikut :

1. Private, hanya bisa dipanggil dari dalam kelas itu sendiri, atribut diawali dengan tanda “-“.
2. Protected, hanya dapat dipanggil oleh kelas yang bersangkutan dan kelas turunannya, atribut diawali dengan tanda “#”.
3. Public, dapat dipanggil dari semua objek, atribut diawali dengan tanda “+”.

Relationship antar kelas yang digunakan pada diagram kelas :

asosiasi / <i>association</i> —————	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
asosiasi berarah / <i>directed association</i> —————>	relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
generalisasi —————▷	relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
kebergantungan / <i>dependency</i> —————→	relasi antar kelas dengan makna kebergantungan antar kelas
agregasi / <i>aggregation</i> —————◊	relasi antar kelas dengan makna semua-bagian ( <i>whole-part</i> )

**Gambar 2.14 Atribut Relationship Class Diagram**

Relationship Multiplicity menunjukkan jumlah suatu objek yang bisa berhubungan dengan objek lain. (Seperti pada gambar dibawah ini).

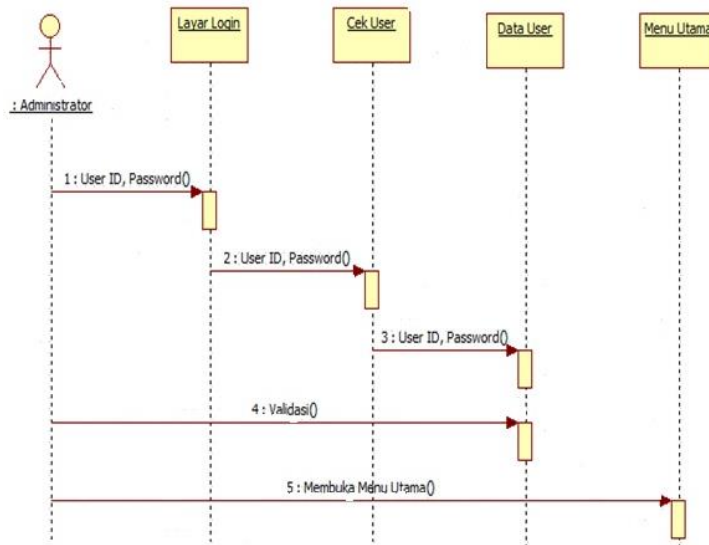
Nilai Kardinalitas	Arti	Contoh
0..1	Nol atau satu	karyawan — 0..1 istri
1	Hanya satu	negara — 1 presiden
0..*	Nol atau lebih	karyawan — 0..* anak
1..*	Satu atau lebih	bos — 1..* bawahan
n	Hanya n (dengan n > 1)	karyawan — n cek up
0..n	Nol sampai n (dengan n > 1)	karyawan — 0..n sim
1..n	Satu sampai n (dengan n > 1)	kereta api — 1..n gerbong

**Gambar 2.15 Atribut Relationship Multiplicity**

c. *Sequence Diagram*

*Sequence Diagram* digunakan untuk menggambarkan perilaku pada sebuah scenario. Kegunaannya untuk menunjukkan rangkaian pesan yang

dikirim antara *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.



**Gambar 2.16 Sequence Diagram**

Berikut komponen – komponen yang ada pada sequence diagram :

1. Object, adalah komponen berbentuk kotak yang mewakili sebuah kelas atau objek. Digunakan untuk mendemonstrasikan bagaimana sebuah objek berperilaku pada sebuah system.



**Gambar 2.17 Ikon Objek**

2. Activation boxes, adalah komponen yang berbentuk persegi Panjang yang menggambarkan waktu yang diperlukan sebuah objek untuk menyelesaikan tugas. Lebih lama waktu yang diperlukan, maka activation boxes akan lebih panjang.



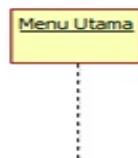
**Gambar 2.18 Ikon Activation boxes**

3. Actors, komponen yang berbentuk stick figure. Komponen yang mewakili seorang pengguna yang berinteraksi dengan system.



**Gambar 2.19 Ikon Actors / Aktor**

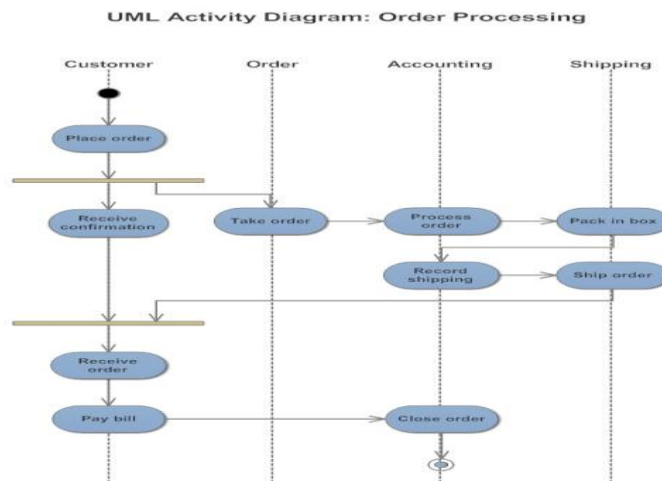
4. Lifeline, komponen yang berbentuk garis putus – putus. Lifeline biasanya memuat kotak yang berisi nama dari sebuah objek. Berfungsi menggambarkan aktifitas dari objek.



**Gambar 2.20 Ikon Lifeline**

#### d. Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.



**Gambar 2.21 Activity Diagram**

Komponen yang ada pada Activity Diagram antara lain :

1. *Start Poin* ( Titik Mulai ), merupakan lingkaran hitam kecil yang menandakan tindakan atau titik awal aktivitas untuk setiap diagram aktivitas.



**Gambar 2.22 Ikon Start Point**

2. *Activity* ( Aktivitas ), menunjukan aktivitas yang dilakukan atau yang sedang terjadi dalam activity diagram.



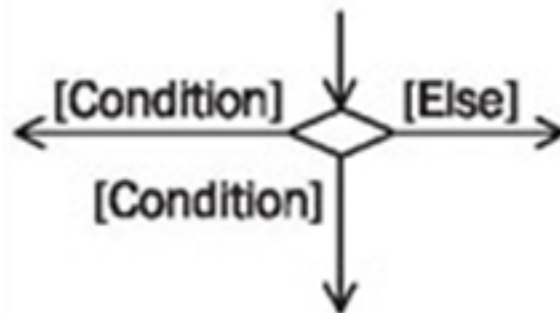
**Gambar 2.23 Ikon Activity**

3. *Action Flow* ( Arah ), digunakan untuk transisi dari suatu tindakan ke tindakan yang lain atau menunjukan aktivitas selanjutnya setelah aktivitas sebelumnya.



**Gambar 2.24 Ikon Action Flow**

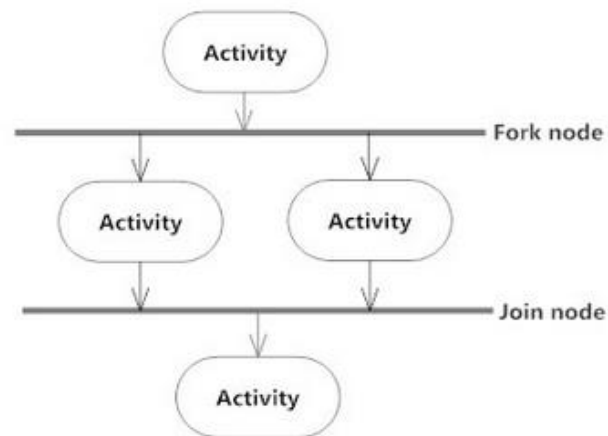
4. *Decision* ( Keputusan ), suatu titik atau point pada activity diagram yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.



**Gambar 2.25 Ikon Decision**

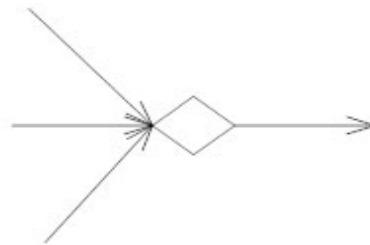
5. *Synchornization*, dibagi menjadi 2 yaitu fork dan join. *Fork* ( Percabangan ) digunakan untuk memecah *behavious* menjadi *activity* atau *action* yang paralel, sedang kan *Join* ( Penggabungan ) untuk menggabungkan kembali *activity* atau *action* yang paralel.





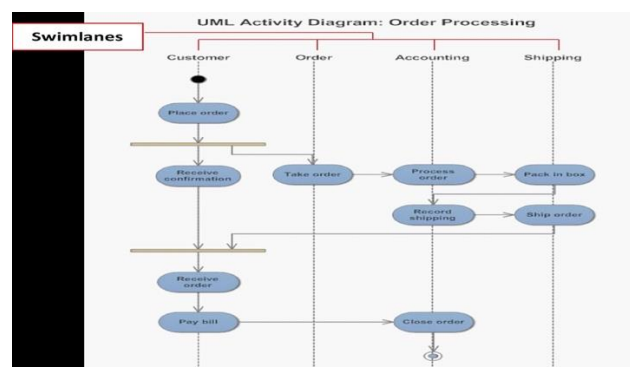
**Gambar 2.26 Ikon Synchronization**

6. Merge Event ( Menggabungkan ), berfungsi untuk menggabungkan flow yang dipecah oleh decision.



**Gambar 2.27 Ikon Merge Event**

7. Swimlanes, berfungsi untuk memecah activity diagram menjadi baris dan kolom untuk membagi tanggung jawab obyek – obyek yang melakukan aktivitas.



**Gambar 2.28 Swimlanes**

8. Final State ( Titik Akhir ), menunjukan bagian aktif dari aktivitas.



**Gambar 2.29 Ikon Final State**

1. Metode MVC (Model, View, Controller)

Model-View-Controller atau MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Dalam implementasinya kebanyakan framework dalam aplikasi website adalah berbasis arsitektur MVC. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi web. Implementasi dari MVC ini dari sistem menggunakan *framework Codeigniter*. Berikut ini adalah penjelasan dari *model, view, controller*.

a. Model

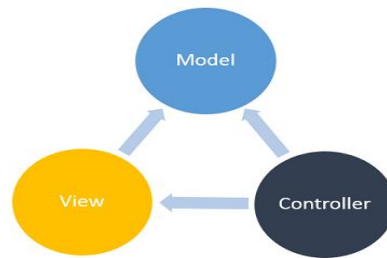
Model, Model mewakili struktur data. Biasanya model berisi fungsi fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.

b. View

View adalah bagian yang mengatur tampilan ke pengguna. Bisa di katakan berupa halaman web.

c. Controller

Controller merupakan bagian yang menjembatani model dan view. Controller berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web.



**Gambar 2.30 Struktur metode MVC (Model, View, Controller).**

### 1. Codeigniter

Codeigniter adalah sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model, View, Controller). codeigniter bersifat free. framework codeigniter di buat dengan tujuan sama seperti framework lainnya yaitu untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal. MVC adalah teknik atau konsep yang memisahkan komponen utama menjadi tiga komponen yaitu model, view dan controller.

- a. Model merupakan bagian yang bertugas untuk berhubungan dengan informasi yang ada didalam database.

Berikut contoh *syntax* dari penggunaan model didalam Codeigniter.

```

1  <?php
2  class M_proses extends CI_Model{
3
4      public function InsertData($table, $data){
5          return $this->db->insert($table,$data);
6      }
7
8  }
9  ?>
  
```

**Gambar 2.31 syntax model codeigniter.**

- b. View merupakan bagian yang bertugas untuk berhubungan dengan informasi yang ada didalam database.

Berikut contoh *syntax* dari penggunaan *view* didalam Codeigniter.

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta name="theme-color" content="#1488CC" />
5 <!-- URL Theme Color untuk Windows Phone -->
6 <meta name="msapplication-navbutton-color" content="#1488CC" />
7 <!-- URL Theme Color untuk iOS Safari -->
8 <meta name="apple-mobile-web-app-capable" content="yes" />
9 <meta name="apple-mobile-web-app-status-bar-style" content="#1488CC" />
10 <meta charset="utf-8">
11 <meta http-equiv="X-UA-Compatible" content="IE=edge">
12 <meta name="viewport" content="width=device-width, shrink-to-fit=no, initial-scale=1">
13 <title><?php echo $title; ?> &rsquo;</title>
14 <link href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.0/css/bootstrap.min.css" rel="stylesheet" id="
bootstrap-css">
15 <link rel="stylesheet" href="<?php echo base_url('asset/css/styleadmin.css'); ?>">
16 </head>
17 <body>
18
19 <!-- konten aplikasi -->
20
21 </body>
22 </html>
23

```

**Gambar 2.32 syntax view codeigniter.**

- c. *Controller* merupakan bagian yang bertugas untuk berhubungan dengan informasi yang ada didalam database. Berikut contoh *syntax* dari penggunaan *controller* login didalam Codeigniter.

```

1 <?php
2 class Login extends CI_Controller{
3
4     public $localadmin = array('title' => 'Administrator');
5     public function __construct(){
6         parent::__construct();
7         $this->load->model('M_proses', 'proses', TRUE);
8         $this->load->model('M_login', 'Login', TRUE);
9         $this->load->model('M_basic', 'basic', TRUE);
10        $this->localadmin['getfeeling'] = getfeeling();
11    }
12
13    public function index(){
14        $this->localadmin['page'] = 'administrator/Login';
15        if($this->session->userdata('Login') == TRUE){
16            redirect('administrator/home');
17        }else{
18            //validasi
19            if($this->Login->validasi()){
20                //cek di database
21                if($this->Login->cek_user()){
22                    redirect('administrator/home');
23                }else{
24                    $this->localadmin['pesan'] = 'Invalid Email or password';
25                    $this->load->view('administrator/Login', $this->localadmin);
26                }
27            }else{
28                $this->load->view('administrator/Login', $this->localadmin);
29            }
30        }
31    }
32 }

```

**Gambar 2.33 syntax controller codeigniter.**

```

1  <?php
2  /**
3   *
4   */
5  class ClassName extends AnotherClass
6  {
7
8  function __construct(argument)
9  {
10     # code...
11 }
12
13 public function index(){
14     # code...
15 }
16
17 }
18 ?>

```

**Gambar 2.34 syntax dasar PHP (OOP)**

## 2.14 Review Literatur

Pada *review* literatur ini dibahas penelitian sebelumnya yang berkaitan dengan permasalahan yang sedang diteliti, sehingga dapat mengetahui tentang uraian teori, temuan dan bahan penelitian lain yang diperoleh dari bahan acuan untuk dijadikan landasan penelitian ini. Berikut adalah *review* literatur dari beberapa penelitian sebelumnya, antara lain :

**Tabel 2.1 Review Literatur kesatu**

Judul Literatur	Virtual Tour Sebagai Media Promosi Interaktif Penginapan Di Kepulauan Bunaken
Penulis	Dianto G. Thomas, Sherwin R. U. A. Sompie, Brave A. Sugiarto
Rangkuman	Cottage-cottage memegang peran penting dalam bidang pariwisata karena menyediakan fasilitas dan pelayanan penginapan, makanan, dan minuman serta jasa-jasa lainnya. Sejalan dengan perkembangan jaman, cottage sudah berkembang ke pemenuhan kebutuhan lainnya seperti terapi hiburan dan kelengkapan alat – alat diving. Masih banyak

wisatawan mancanegara yang belum mengetahui di mana lokasilokasi penginapan tersebut, dan lebih buruknya lagi kita wisatawan lokal atau masyarakat Sulawesi utara yang tidak mengetahui ataupun belum pernah berkunjung ke Taman Nasional Bunaken, di sebabkan karena kurang media informasi dan promosi, media informasi dan promosi yang di berikan berupa website, brosur dan media cetak lainnya hanya menggunakan gambar diam. Meskipun sudah banyak regulasi dan kebijakan pemerintah yang memayungi kepariwisataan, namun ini masih belum cukup untuk meningkatkan kunjungan wisatawan. Untuk itu perlu adanya implementasi virtual tour media promosi tentang Cottage-cottage di kepulauan bunaken. Pengembangan sistem penelitian ini menggunakan metode pengembangan Luther-Sutopo, yang melalui beberapa tahapan yaitu concept, design, material collecting, assembly, testing, dan distribution. Melalui penulisan ini juga kita dapat melihat, proses penggabungan beberapa gambar menjadi satu gambar panorama 360 derajat. Dalam aplikasi ini pengguna seolah-olah berada di lokasi penginapan dan dapat melihat seluruh titik lokasi penginapan yang ada di kepulauan bunaken dengan sudut pandang 360°. Dari hasil aplikasi virtual tour media promosi penginapan di kepulauan bunaken ini menghasilkan berkas HTML5, kapasitas 200 MB, dan berbasis online

**Tabel 2.2 Review Literatur kedua**

Judul Literatur	Aplikasi Virtual Tour Dinamis Pada Universitas Dr. Soetomo Surabaya Berbasis Web.
Penulis	Achmad Choiron, Irfian Lesmana
Rangkuman	Pada aplikasi virtual tour Dinamis pada Universitas Dr. Soetomo Surabaya Berbasis Web, berawal dari penyampaian informasi mengenai letak gedung di Universitas Dr. Soetomo baik itu Jurusan, Fakultas dan TU administrasi yang ditujukan kepada mahasiswa ataupun pengunjung masih dilakukan secara lisan, poster ataupun brosur. Tentunya Penyampaian informasi tersebut agak menjadi sulit bagi orang awam yang akan ingin mengetahui letaknya. Melihat kebutuhan tersebut maka di buatlah sebuah Aplikasi Virtual Tour Pada Universitas Dr. Soetomo Berbasis Web. Virtual Tour sendiri adalah sebuah simulasi dari lokasi yang ada, terdiri dari urutan gambar diam yang dibentuk panorama dengan pandangan yang tidak terputus, didalam Virtual Tour bisa ditambahkan elemen multimedia seperti efek suara, musik, narasi, dan teks Aplikasi Virtual Tour Pada Universitas Dr. Soetomo ini akan memberikan informasi yang lebih atraktif, pengguna akan dapat melihat lingkungan kampus secara visual dengan pandangan panorama 360° serta informasi letak gedung dan personal seseorang dapat diakses didalam aplikasi Virtual Tour ini.

**Tabel 2.3 Review Literatur Ketiga**

Judul Literatur	Pembangunan Aplikasi Virtual Tour Sebagai Media Promosi di Rest Area Griya Dahar Ibu Kadi
Penulis	Ahmad Wijayanto, Richi Dwi Agustia
Rangkuman	Aplikasi yang dibangun yaitu aplikasi virtual tour dengan menggunakan metode stitching dimana proses yang digunakan

	bertahap dari mulai pegumpulan foto, proses stitching, selanjutnya menjadi gambar 360 dan menentukan hotspot. Proses pengujian dilakukan dengan cara melakukan wawancara kepada konsumen. Berdasarkan hasil pengujian, dengan adanya aplikasi virtual tour ini dapat digunakan sebagai media promosi yang interaktif untuk pemilik dan juga untuk memudahkan konsumen dalam memperoleh informasi seperti fasilitas, menu, dan harga yang terdapat di Rest Area Griya Dahar Ibu Kadi.
--	--

### 2.15 Skala Likert

Skala likert merupakan skala yang dapat dipergunakan untuk mengukur sikap, pendapat dan persepsi seseorang tentang suatu gejala atau fenomena tertentu. Instrumen penelitian yang menggunakan skala likert dapat dibuat dalam bentuk checklist ataupun pilihan ganda [15]. Seperti menentukan lokasi kedudukan seseorang dalam suatu *kontinuum* sikap terhadap objek sikap, mulai dari sangat negative sampai dengan positif. Untuk setiap satu pernyataan responden memilih salah satu tanggapan berupa SS = Sangat Setuju, S = Setuju, Kurang Setuju=KS, TS = Tidak Setuju, STS = Sangat Tidak Setuju.

Tanggapan responden dikodekan kedalam bilangan 1 sampai dengan 5. Pemberian bilangan ini bergantung kepada bentuk pernyataan yang dijawab oleh responden. Ada dua macam bentuk pernyataan dalam *skala likert*, pertama adalah pernyataan yang diharapkan untuk disetujui oleh responden, disebut bentuk positif. Yang kedua adalah pernyataan yang diharapkan untuk tidak disetujui oleh responden, bentuk negatif. Biasanya kuesioner berisikan pernyataan positif dan negatif agar responden berfikir dulu sebelum memberi jawaban. Contoh kuesioner sikap masyarakat menggunakan *skala likert* mengenai website 360 view yang dibangun untuk promosi Desa Wisata Kampung Buricak Burinong. Pegunjung menjawab:

A = Sangat Setuju,

B = Setuju,

C = Ragu,



D = Tidak Setuju,

E = Sangat Tidak Setuju

**Tabel 2.4 Contoh Skor Skala Likert**

Pilihan Jawaban	Pernyataan	
	Positif	Negatif
Sangat Setuju	5	1
Setuju	4	2
Ragu-ragu	3	3
Tidak Setuju	2	4
Sangat Tidak Setuju	1	5

Pada bentuk negatif, sangat setuju memperoleh skor terendah dan sangat tidak setuju memperoleh skor tinggi. *Skala Likert* digunakan untuk mengukur penilaian memperkenalkan Desa Wisata Kampung Buricak Burinong kepada calon wisatawan. Untuk mencari nilai presentase dari masing-masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

$$P = \frac{S}{Skor\ Ideal} \times 100\%$$

Berikut ini adalah keterangan dari rumus *skala likert* mencari nilai presentase dari jawaban kuesioner :

*P* = Nilai Presentasi yang dicari

*S* = Jumlah frekuensi dilakukan dengan skor yang ditetapkan jawaban

*Skor Ideal* = Nilai tertinggi dikalikan dengan jumlah sampel. Nilai tertingginya yaitu 5.

*Skala Likert* digunakan untuk mengukur promosi dengan website 360 view. Skala Likert ini akan digunakan pada Bab 4 implementasi dan pengujian.

## 2.16 Leaflet

Leaflet adalah perpustakaan JavaScript *Open Source* yang *mobile-friendly* untuk peta interaktif. Dengan file sebesar sekitar 38 KiloByte, leaflet memiliki semua fitur pemetaan yang paling dibutuhkan pengembang. Leaflet dirancang dengan kesederhanaan, kinerja dan kegunaan. Dalam hal ini leaflet bekerja secara efisien di semua *platform* desktop dan seluler, dapat diperluas dengan banyak plugin, memiliki API yang indah, mudah digunakan dan terdokumentasi dengan baik, dan kode sumber sederhana yang mudah dibaca yang dapat berkontribusi.[18]

Dalam pembangunannya Leaflet menggunakan *map Open Source* yaitu *OpenStreetMap*. *OpenStreetMap* adalah sebuah proyek berbasis web untuk membuat peta seluruh dunia yang gratis dan terbuka, dibangun sepenuhnya oleh sukarelawan dengan melakukan survey menggunakan GPS, mendigitasi citra satelit, dan mengumpulkan serta membebaskan data geografis yang tersedia di publik. [17].

## 2.17 Kuesioner

Kuesioner merupakan teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk di jawabnya. Kuesioner merupakan teknik pengumpulan data yang efisien bila peneliti tahu dengan pasti variabel yang akan diukur dan tahu apa yang bisa diharapkan dari responden. Selain itu kuesioner juga cocok digunakan bila jumlah responden cukup dan tersebar di wilayah yang luas. Kuesioner dapat berupa pertanyaan/pernyataan tertutup atau terbuka, dapat diberikan kepada responden secara langsung atau dikirim melalui pos, atau internet.

Bila penelitian dilakukan pada lingkungan yang tidak terlalu luas, sehingga kuesioner dapat diantarkan langsung dalam waktu tidak terlalu lama, maka pengiriman angket kepada responden tidak perlu melalui pos. Dengan adanya kontak langsung antara peneliti dengan responden akan menciptakan suatu kondisi yang cukup baik, sehingga responden dengan sukarela akan memberikan data obyektif dan cepat.[16]

## 2.18 Populasi

Populasi adalah wilayah generalisasi yang terdiri atas: obyek/subyek yang mempunyai kualitas dan karakteristik tertentu yang ditetapkan oleh peneliti untuk dipelajari dan kemudian ditarik kesimpulannya.

Populasi bukan hanya orang, tetapi juga obyek dan benda-benda alam yang lain. Populasi juga bukan sekedar jumlah yang ada pada obyek/subyek yang dipelajari, tetapi meliputi seluruh karakteristik/sifat yang dimiliki oleh subyek atau obyek itu. [16]

## 2.19 Sampel

Sampel adalah bagian dari jumlah dan karakteristik yang dimiliki oleh populasi tersebut. Bila populasi besar, dan peneliti tidak mungkin mempelajari semua yang ada pada populasi, misalnya karena keterbatasan dana, tenaga dan waktu, maka peneliti dapat menggunakan sampel yang diambil dari populasi itu. Apa yang dipelajari dari sampel itu, kesimpulannya akan dapat diberlakukan untuk populasi. Untuk itu sampel yang diambil dari populasi harus betul-betul representatif (mewakili) [16].

### a. Menentukan ukuran sample

Jumlah anggota sample sering dinyatakan dengan ukuran sampel. Jumlah sampel yang diharapkan 100% mewakili populasi adalah sama dengan jumlah anggota populasi itu sendiri. Berikut ini tabel penentuan jumlah sampel dari populasi tertentu yang dikembangkan dari *Isaac* dan *Michael*, untuk tingkat kesalahan, 1%, 5%, dan 10%. Rumus untuk menghitung ukuran sampel dari populasi yang diketahui jumlahnya adalah sebagai berikut :

$$s = \frac{\lambda^2 \cdot N \cdot P \cdot Q}{d^2(N - 1) + \lambda^2 \cdot P \cdot Q}$$

$\lambda^2$  dengan dk = 1, taraf kesalahan dapat 1%, 5%, 10%.

P = Q = 0,5. d = 0,05. s = Jumlah sampel.

Tabel 2.5 Penentuan Jumlah Sampel dari Populasi Tertentu

N	s			N	s			N	s		
	1%	5%	10%		1%	5%	10%		1%	5%	10%
10	10	10	10	280	197	155	138	2800	537	310	247
15	15	14	14	290	202	158	140	3000	543	312	248
20	19	19	19	300	207	161	143	3500	558	317	251
25	24	23	23	320	216	167	147	4000	569	320	254
30	29	28	27	340	225	172	151	4500	578	323	255
35	33	32	31	360	234	177	155	5000	586	326	257
40	38	36	35	380	242	182	158	6000	598	329	259
45	42	40	39	400	250	186	162	7000	606	332	261
50	47	44	42	420	257	191	165	8000	613	334	263
55	51	48	46	440	265	195	168	9000	618	335	263
60	55	51	49	460	272	198	171	10000	622	336	263
65	59	55	53	480	279	202	173	15000	635	340	266
70	63	58	56	500	285	205	176	20000	642	342	267
75	67	62	59	550	301	213	182	30000	649	344	268
80	71	65	62	600	315	221	187	40000	653	345	269
85	75	68	65	650	329	227	191	50000	655	346	269
90	79	72	68	700	341	233	195	75000	658	346	270
95	83	75	71	750	352	238	199	100000	659	347	270
100	87	78	73	800	363	243	202	150000	661	347	270
110	94	84	78	850	373	247	205	200000	661	347	270
120	102	89	83	900	382	251	208	250000	662	348	270
130	109	95	88	950	391	255	211	300000	662	348	270
140	116	100	92	1000	399	258	213	350000	662	348	270
150	122	105	97	1100	414	265	217	400000	662	348	270
160	129	110	101	1200	427	270	221	450000	663	348	270
170	135	114	105	1300	440	275	224	500000	663	348	270
180	142	119	108	1400	450	279	227	550000	663	348	270
190	148	123	112	1500	460	283	229	600000	663	348	270
200	154	127	115	1600	469	286	232	650000	663	348	270
210	160	131	118	1700	477	289	234	700000	663	348	270
220	165	135	122	1800	485	292	235	750000	663	348	270
230	171	139	125	1900	492	294	237	800000	663	348	271
240	176	142	127	2000	498	297	238	850000	663	348	271
250	182	146	130	2200	510	301	241	900000	663	348	271
260	187	149	133	2400	520	304	243	950000	663	348	271
270	192	152	135	2600	529	307	245	1000000	663	348	271
								∞	664	349	272

## 2.20 Metode Pengujian Sistem

Metode pengujian sistem terdiri dari pengujian *black box* dan *beta* yang digunakan untuk mengetahui efektifitas dari perangkat lunak (*software*) yang digunakan.

### 2.20.1 Pengujian Alpha

Pengujian alpha merupakan pengujian yang dilakukan dengan melakukan cek sistem fungsionalitas yang terdapat di aplikasi. Dengan melakukan pengujian ini diharapkan dapat mengurangi *bug* atau kesalahan program sebelum pada saatnya dipakai oleh *user* (pengguna). Pengujian alpha sangat berpengaruh kepada baik atau buruknya suatu aplikasi dalam teknis *programming*.

### 2.20.2 Pengujian Black Box

Pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian ini dianalogikan seperti melihat suatu kotak hitam yang hanya bisa dilihat penampilan luarnya saja, tanpa tahu ada apa dibalik bungkus hitamnya. *Black Box* testing melakukan evaluasi untuk menemukan kesalahan dalam kategori berikut :

- a. Fungsi tidak benar atau hilang.
- b. Kesalahan interface atau antarmukaaa
- c. Kesalahan dalam struktur data atau akses database eksternal
- d. Kesalahan kinerja atau perilaku dan kesalahan inisialiasai dan terminasi.

### 2.21 Pengujian Beta

Pengujian merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan kuesioner mengenai tanggapan pengguna terhadap aplikasi yang telah dibangun. Adapun metode penilaian pengujian yang digunakan yaitu metode kuantitatif berdasarkan data sampel dari pengguna.

