

BAB II TINJAUAN PUSTAKA

Dalam penelitian referensi menjadi hal penting untuk dapat dijadikan acuan maupun informasi lebih untuk dapat memperkuat penelitian.

2.1. Pencapaian Terdahulu

Dapat membantu penulis dalam penelitian ini, pencapaian terdahulu yang dijadikan referensi adalah jurnal yang merancang sebuah sistem kerja pengelolaan anggaran berbasis *website* dengan berbagai metode yang digunakan. Berikut beberapa referensi yang dijadikan acuan dalam penelitian ini. Referensi pertama ditulis oleh Nirzal Rusmala dengan judul “Perancangan Aplikasi Pembuatan Rencana Kegiatan dan Anggaran Tahunan (RKAT)” (Rusmala 2019). Jurnal tersebut menjelaskan bagaimana perancangan sebuah aplikasi pembuatan Rencana Kegiatan dan Anggaran Tahunan (RKAT) dengan metodologi interface menggunakan UML. Jurnal tersebut berisikan tahapan alur akses setiap pengguna belum kepada tahap implementasi kode program.

Referensi berikutnya yang dijadikan acuan adalah jurnal yang ditulis oleh Alam Rahmatulloh dan Husen dengan judul “Sistem Informasi Manajemen Anggaran (Simangga) Perguruan Tinggi Berbasis *Web* (Studi Kasus : Universitas Siliwangi)” (Alam Rahmatulloh dan Husen 2017). Pada jurnal tersebut menjelaskan bagaimana merancang sebuah sistem informasi dalam pengelolaan sebuah anggaran dalam bentuk *website*. Jurnal tersebut telah cukup menjelaskan bagaimana tahapan-tahapan serta metode yang digunakan dalam pembuatan sebuah *website* untuk pengelolaan anggaran. Jurnal tersebut menjelaskan belum dapat menampilkan data secara *real time* sehingga memerlukan waktu jeda untuk penampilan data.

Referensi selanjutnya yaitu jurnal yang ditulis oleh Elisa Dwi Jayanti dan Nur Ani dengan judul jurnal “Pembangunan *Dashboard* Untuk Visualisasi Analisa Keuangan” (Jayanti and Ani 2017). Jurnal tersebut menjelaskan bagaimana tahapan dalam membuat sebuah sistem yang digunakan untuk pencatatan keuangan. Fokus utama yaitu membangun sebuah teknologi informasi yang menyajikan sebuah visualisasi dari analisis keuangan disuatu perusahaan.

Jurnal berikutnya yang dijadikan acuan bagi peneliti jurnal berjudul “*A Systematic Mapping of Software Engineering Approaches to Develop Big Data Systems*” (Laigner et al. 2018). Jurnal internasional yang berisikan bagaimana membuat sebuah pengembangan perangkat lunak untuk sistem *big data*. Jurnal tersebut ditulis oleh Rodrigo Nunes Laigner, Marcos Kalinowski, Sérgio Lifschitz, Rodrigo Salvador Monteiro, Daniel de Oliveira

Referensi yang dijadikan acuan terakhir pada penelitian ini adalah jurnal internasional dengan masih dalam cakupan yang sama dipenelitian ini adalah pengelolaan anggaran, jurnal tersebut berjudul “*Web-based System for State Universities and Colleges Budget Office Management*” dibuat oleh Joni Neil Capucan, Thelma Palaoag dan Sean Sierra (Joni Neil Capucan, Thelma Palaoag 2020). Jurnal tersebut menjelaskan bagaimana pembuatan sebuah system pengelolaan anggaran dalam sebuah perguruan tinggi dengan berbasis *web-based*. Di dalam jurnal dijelaskan dengan rinci tahapan perancangan sebuah website pengelolaan anggaran sehingga dapat dijadikan sebuah referensi bagi penulis untuk dapat menjadi catatan dalam merancang penelitian ini.

2.2. Tinjauan Teoritis

Tinjauan teoritis adalah pembahasan mengenai teori-teori yang dibahas dalam penelitian ini. Tujuan dari tinjauan teoritis yaitu untuk memberikan informasi serta pemahaman terhadap topik penelitian.

2.2.1. Teori Analisis dan Perancangan Data

Analisis merupakan aktivitas yang terdiri dari beberapa rangkaian kegiatan seperti mengurai, membedah dan memilih. Kemudian dikelompokkan kembali menurut kriteria tertentu dengan dicari antar kaitan lalu dibuat tafsiran maknanya.

2.2.1.1. Statistik Deskriptif

Pada penelitian ini untuk dapat lebih memperjelas penulis menggunakan salah satu metode yang ada dalam statistik dimana metode ini berfungsi bagi penulis dalam melakukan pengelolaan data. Metode statistik yang digunakan yaitu metode deskriptif dimana metode tersebut adalah penyajian dari sebuah data untuk

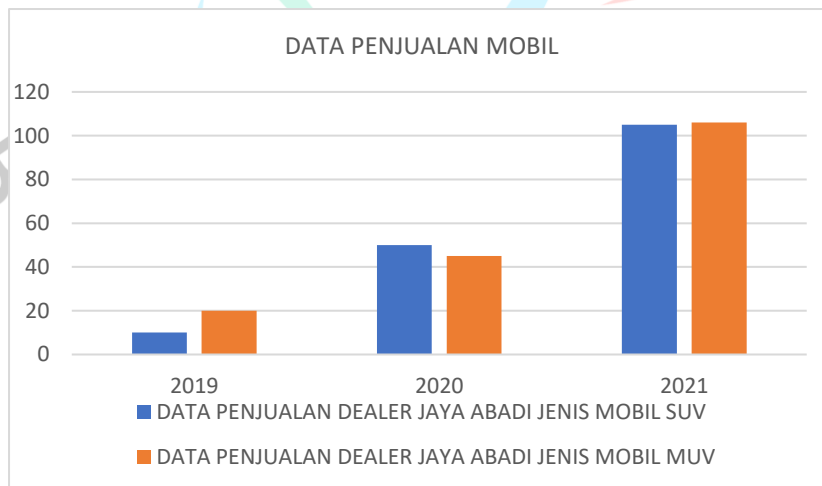
dapat mendeskripsikan atau menggambarkan dari kumpulan data yang dapat berupa representasi dari data atau sample dari suatu data (HAYES 2021). Metode statistik yang akan digunakan sesuai dengan yang akan dilakukan dalam penelitian ini yaitu menampilkan sebuah tren capaian dari data Rencana Kegiatan dan Anggaran Tahunan setiap program studi maupun unit bagian dengan penyajian data menggunakan grafik.

Statistika deskriptif dapat diimplementasi pada perusahaan otomotif yang menjual mobil sebut saja dealer jaya abadi. Jaya abadi memiliki data penjualan mobil tahunan selama tiga tahun terakhir dengan dua jenis mobil berbeda yaitu MUV dan SUV. Data tersebut memiliki rincian berikut.

Tabel 2. 1. Menunjukkan contoh data penjualan mobil

Data penjualan dealer jaya abadi		
Tahun	Jenis mobil	
	SUV	MUV
2019	10	20
2020	50	45
2021	105	106

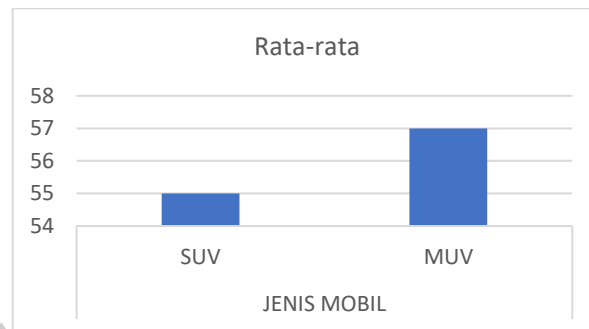
Tabel 2.1. merupakan contoh dari capaian penjualan pada dealer jaya abadi selama tiga tahun berturut. Menampilkan statistik deskriptif dengan penyajian data dalam bentuk grafik maka akan menjadi seperti berikut.



Gambar 2. 1. Menampilkan Contoh implementasi data deskriptif capaian

Gambar 2.1. menampilkan hasil grafik dari tabel capaian penjualan mobil perusahaan dealer jaya abadi. Grafik tersebut menampilkan hasil data dari tabel menjadi grafik sehingga mudah dalam membaca data. Selain menampilkan grafik

capaian statistik deskriptif juga dapat menampilkan data rata-rata capaian yang dapat dilihat di gambar 2.2.



Gambar 2. 2. Menampilkan contoh implementasi data deskriptif rata-rata

2.2.1.2. Analisis Data Penelitian Kuantitatif

Analisis data merupakan cara dalam mengelola data menjadi sebuah informasi agar dapat setiap karakteristik data tersebut dapat dipahami dan bermanfaat sehingga menjadi solusi dalam permasalahan yang ada (Setiawan 2021). Sedangkan penelitian kuantitatif yaitu suatu cara yang digunakan dalam menjawab sebuah masalah pada penelitian yang berkaitan dengan data angka dan program statistik (Dr. Wahidmurni 2017).

(1) *Editing*

Editing adalah kegiatan mengubah data yang bertujuan untuk dapat mengevaluasi konsistensi, kelengkapan dan kesesuaian antar kriteria setiap data yang dibutuhkan.

(2) *Coding*

Coding merupakan pemberian sebuah kode pada data yang dilakukan dengan bertujuan merubah sebuah data kualitatif menjadi data kuantitatif. Pemberian sebuah kode pada data sangat diperlukan untuk dapat dilakukan pengelolaan data baik secara manual maupun dengan komputer.

(3) Tabulasi data

Tabulasi data adalah memasukkan data ke dalam tabel-tabel yang disediakan, baik tabel untuk data mentah maupun tabel kerja fungsinya untuk dapat menghitung data tertentu secara statistik.

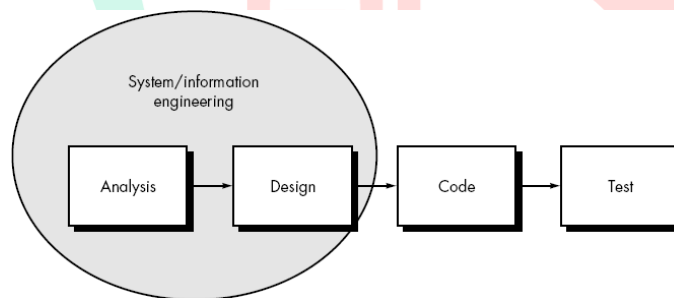
(4) Pembahasan atau diskusi hasil penelitian

Pada tahap terakhir ini ditujukan untuk peneliti dapat mengabstraksikan dari hasil penelitian, tahap ini membahas hasil penelitian serta

mengkonsultasikannya dengan hasil penelitian sebelumnya (jika dimungkinkan).

2.2.1.3. *Sequence Linear*

Sequence linear merupakan metode yang cukup sama dari metode *waterfall*. Dimana yang berbeda adalah tahapan akhir yaitu, *waterfall* setelah melakukan tahapan *testing* selanjutnya memiliki tahapan yang harus dilakukan yaitu *maintenance*. Sedangkan *sequence linear*, *maintenance* adalah tahapan yang tidak wajib dilakukan. Metode *sequence linear* dapat dikatakan metode *waterfall* yang lebih disederhanakan. Metode *waterfall* memiliki tahapan yaitu *requirement analysis*, *system design*, *implementation*, *testing*, *deployment*, dan *maintenance*. Sedangkan *sequence linear* memiliki hanya empat tahapan yaitu *analisis*, *desain*, *code*, dan *testing*. Metode sendiri dalam membangun sebuah sistem perangkat lunak difungsikan untuk perancangan dan tahapan. Berikut merupakan tahapan-tahapan dalam metode *sequence linear*.



Gambar 2. 3. Tahapan *sequence linear*.

(1) *Analysis*

Pada tahap ini dilakukan persiapan dan melakukan penganalisaan kebutuhan dari aplikasi yang akan dikembangkan.

(2) *Design*

Ditahapan ini melakukan pembuatan desain dari aplikasi yang akan di kembangkan sebelum aplikasi masuk dalam tahapan pemberian baris kode.

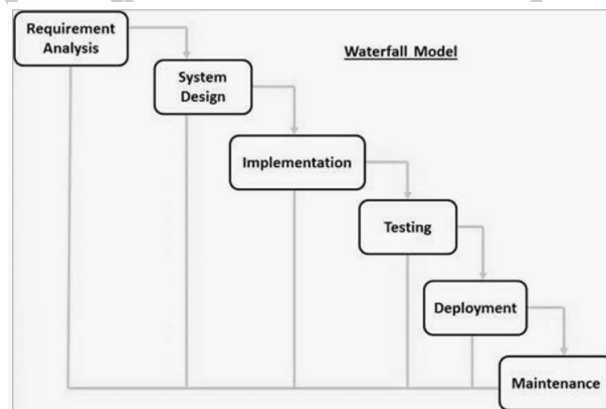
(3) *Code*

Tahapan ini melaksanakan pemberian kode program dengan menggunakan berbagai *tools* dan bahasa pemrograman untuk dapat membangun aplikasi yang telah di desain dan dipersiapkan.

(4) Test

Ini adalah tahapan yang cukup penting dimana tahap ini melakukan pengecekan kembali pengujian sistem dari sistem yang telah dibuat dengan kode program yang ada.

Berdasarkan penjelasan di atas mengenai model *sequence linear* penulis menjabarkan pula mengenai *waterfall* untuk dapat melihat perbedaan keduanya. *waterfall* adalah peneliti menampilkan alur dari metode *waterfall* sebagai berikut (Tutorialspoint 2022).



Gambar 2. 4. Tahapan model waterfall

(1) Requirement analysis

Tahapan ini adalah tahapan dimana melakukan analisis spesifikasi kebutuhan yang akan digunakan didalam sistem.

(2) System design

Setelah mendapatkan spesifikasi sistem pada tahap pertama. Tahap berikutnya adalah melakukan desain sistem yaitu menentukan perangkat keras yang digunakan, menjabarkan persyaratan sistem dan mendefinisikan arsitektur sistem secara keseluruhan.

(3) Implementation

Pada tahap ini sistem yang telah dirancang berikutnya diimplementasikan dengan menggunakan kode-kode program untuk menciptakan sistem yang telah dirancang.

(4) Testing

Setelah implementasi kode program maka dibutuhkan tahapan testing untuk memastikan setiap kode program telah berjalan dan telah sesuai dengan sistem yang direncanakan.

(5) *Deployment*

Tahapan ini adalah tahapan produksi atau sistem yang dibangun telah layak digunakan oleh seluruh pengguna yang telah ditargetkan.

(6) *Maintenance*

Tahapan ini dilakukan secara berkala yaitu dengan selalu memastikan bahwa sistem tetap berjalan dengan baik.

2.2.2. Teori Perancangan Proses

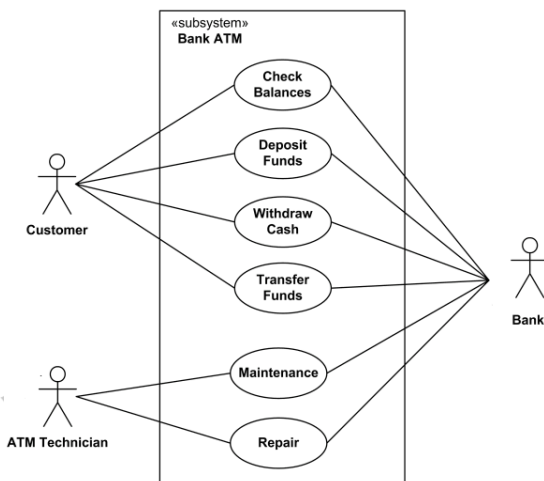
Merancang sebuah aplikasi dibutuhkan perancangan yang berproses untuk dapat membantu dalam perancangan. Dengan dilakukan perancangan secara bertahap maka dapat diketahui apa yang akan dituju dari pembuatan aplikasi.

2.2.2.1. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) merupakan sebuah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasi dari sebuah sistem perangkat lunak (Pratama 2019). UML juga merupakan bahasa pemodelan yang menggunakan konsep orientasi objek. UML berisikan beberapa diagram yang dapat digunakan. Adapun diagram tersebut adalah sebagai berikut.

(1) *Use Case Diagram*

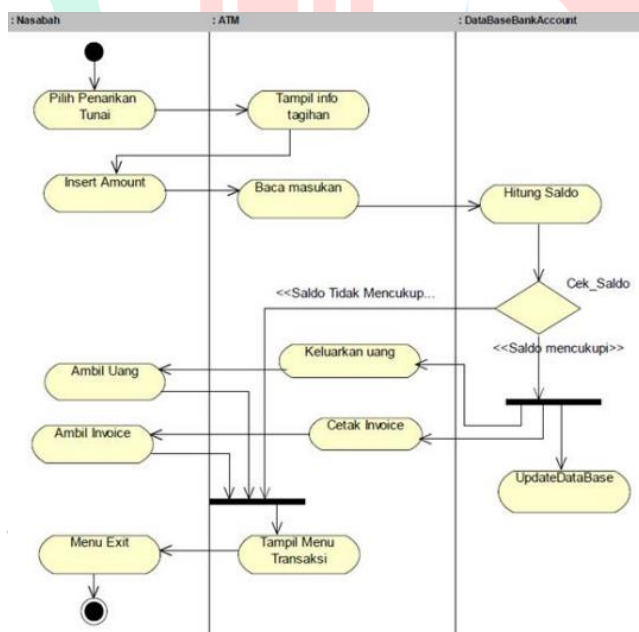
Use Case Diagram adalah diagram yang menggambarkan hubungan interaksi antara sistem dengan aktor (Dicoding 2021). Diagram ini menjelaskan bagaimana aksi aktor dengan aksi yang ada di dalam sistem. Diagram ini membantu untuk mengembangkan sebuah aplikasi agar dapat mengetahui setiap akses yang ada bagi setiap aktor yang akan menggunakannya. *Use case* juga menggambarkan tahapan apa yang ada dan tahapan mana saja yang dapat digunakan oleh setiap *user*. Berikut merupakan contoh dari bentuk *use case diagram* yang dapat dilihat pada gambar 2.5.



Gambar 2. 5. Menampilkan contoh *use case* diagram.

(2) *Activity* Diagram

Activity diagram merupakan sebuah diagram yang dapat memodelkan berbagai aktivitas proses yang ada pada sistem (Dicoding 2021). Singkatnya adalah sebuah gambaran perjalanan sistem yang bergambar secara vertikal atau menyamping. Berikut merupakan sebuah gambaran dari *activity* diagram.

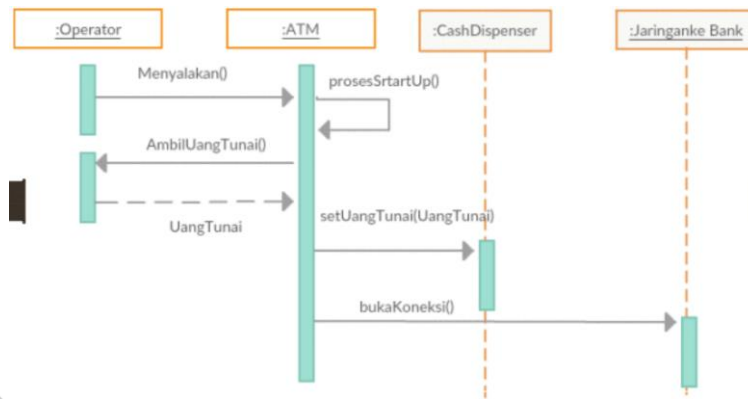


Gambar 2. 6. Menggambarkan contoh *activity* Diagram.

(3) *Sequence* Diagram

Sequence diagram adalah diagram yang menjelaskan interaksi setiap object yang ada dengan berdasarkan urutan waktu (Dicoding 2021). *Sequence* menggambarkan setiap urutan yang ada ataupun tahapan yang memang harus

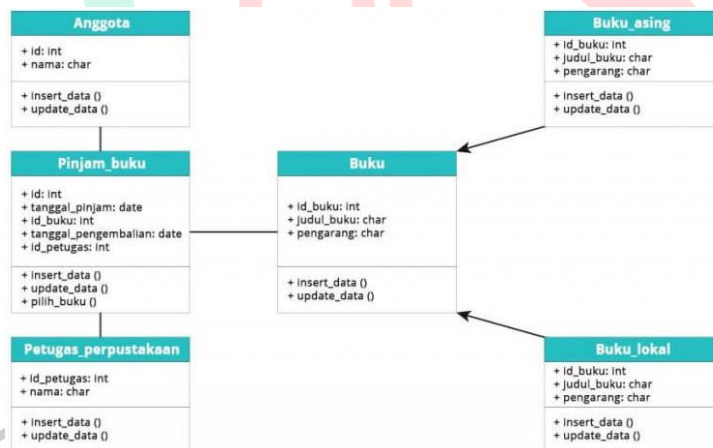
dilakukan untuk dapat menghasilkan tujuan. Berikut merupakan gambaran dari *sequence diagram*.



Gambar 2. 7. Menggambarkan contoh *sequence diagram*.

(4) *Class Diagram*

Class Diagram adalah diagram yang menggambarkan kelas-kelas yang digunakan dengan terhubung pada *database* yang akan dipakai. Berikut contoh gambaran *class diagram*.



Gambar 2. 8. Menggambarkan contoh *class diagram*.

2.2.3. Rencana Kegiatan dan Anggaran Tahunan (RKAT)

Rencana Kegiatan dan Anggaran Tahunan (RKAT) adalah sebuah kegiatan pencatatan keuangan yang dilakukan pertahun untuk dapat menggambarkan rencana kegiatan serta anggaran setiap tahun maupun periode. RKAT menggambarkan perencanaan maupun strategi keuangan di sebuah perusahaan maupun instansi. Dengan adanya Rencana Kegiatan dan Anggaran Tahunan (RKAT) maka setiap perencanaan di dalam sebuah perusahaan/instansi dapat di

dokumentasi dengan baik serta dapat dijadikan sebuah evaluasi setiap anggaran yang keluar. Jika sebuah perencanaan dapat disusun dengan baik maka perusahaan dapat mengetahui efisiensi setiap anggaran yang dikeluarkan.

2.2.3.1. Metode Pencatatan RKAT di UPJ

Pada penelitian ini tempat studi kasus adalah Universitas Pembangunan Jaya. UPJ dalam melakukan pencatatan perencanaan Rencana Kegiatan dan Anggaran Tahunan masih menggunakan metode yang rumit. Metode yang digunakan adalah menggunakan metode Microsoft Excel. Dengan Menggunakan metode excel memiliki banyak kekurangan yang terjadi sehingga membutuhkan metode baru yang lebih mudah digunakan, serta dalam melihat setiap data dapat dilakukan dengan mudah untuk dijadikan evaluasi dan *monitoring*. Kekurangan dari penggunaan metode Excel yaitu dalam pengumpulan data, penampilan data dan pengisian data cukup rentan memiliki kesalahan karena penumpukan data sering terjadi. Dengan metode Excel membutuhkan banyak kolom serta banyak *cell* yang perlu di isi oleh pengguna saat melakukan pencatatan yang membuat data terlihat menumpuk. Lalu dalam penampilan data yang telah terkumpul menjadi sulit untuk dibaca dan dipahami. Berikut gambaran dari metode excel yang digunakan selama ini.

Tabel 2. 2. Formulir isian metode Excel

Nama Kolom	Isian Kolom
Nama Unit Kerja	Nama Unit/Program Studi
Kategori	Program Kerja (PK), Oprasional (OPS) atau Inventaris (INV)
Nomor	Nomor Urut Kegiatan
Nama Kegiatan	Nama dari Kegiatan yang dilaksanakan
Komp	Butir yang ada di dalam KPI
Keterangan Komp	Keterangan Butir
No	Nomor yang ada di dalam butir KPI
Komp & No	Butir dan nomor di dalam butir
Indikator Standar	Jenis-jenis KPI yang ada
Gasal	Anggaran yang dikeluarkan pada semester Gasal
Genap	Anggaran yang dikeluarkan pada semester Genap
Total	Total anggaran yang dikeluarkan setiap kegiatan

2.2.4. Tools Pengembangan Kode Program

Dalam mengembangkan sebuah aplikasi maka dibutuhkan beberapa *tools* yang dapat membantu dalam membangun sebuah aplikasi yang dirancang. *Tools*

tersebut bisa bahasa program, pengolahan data ataupun *code editor*. Berikut beberapa *tools* yang digunakan.

a) *HTML (Hypertext Markup Language)*

HTML merupakan sebuah bahasa markup utama yang dapat digunakan dalam menyusun sebuah isi konten dalam *web* (Fisher 2021). *HTML* menjadi sebuah *canvas* untuk dapat melihat dengan baik setiap warna, tabel, teks dan lain-lain untuk dapat mudah dibaca (Christy 2020). Disaat menggunakan bahasa ini struktur kode sederhana yang digunakan adalah *tag* dan *attribute* contoh sederhana saat ingin membuat sebuah tulisan paragraf maka yang dilakukan yaitu tag pembuka `<p>` dan tag penutup `</p>` ditengah antara keduanya adalah *attribute*, yang dapat dilihat melalui gambar berikut untuk dapat memperjelas.

```
186 |
187 | <p> halo apakabar hari ini semoga selalu semangat </p>
188 |
```

Gambar 2. 9. Pembuatan paragraf menggunakan HTML.

Selain paragraf ada banyak elemen lain yang biasa digunakan disaat menggunakan bahasa HTML ini yang tersusun sebagai berikut.

```
<div>
  <h1>The Main Heading</h1>
  <h2>A catchy subheading</h2>
  <p>Paragraph one</p>
  
  <p>Paragraph two with a <a href="https://example.com">hyperlink</a></p>
</div>
```

Gambar 2. 10. Susunan struktur HTML.

- Elemen teratas serta terbawah merupakan elemen division dengan bentuk `<div></div>` dapat digunakan ketika ingin membuat sebuah *markup* yang besar.
- Kemudian susunan di atas terdiri dari beberapa susunan yaitu heading yang tertulis `<h1></h1>`, subheading yang tertulis `<h2></h2>`, paragraf `<p></p>` dan gambar yang tertulis ``.
- Kemudian untuk dapat menggunakan link maka dapat menggunakan tulisan `<a>`.

Kemudian dalam menggunakan *HTML* ada yang dinamakan *Tag Block Level* dimana setiap dokumen *HTML* memiliki *Level* `<html>`, `<head>` dan `<body>`. Yang tersusun dengan rapi seperti gambar 2.11. berikut ini.

```

<html>
  <head>
    <!-- META INFORMATION -->
  </head>
  <body>
    <!-- PAGE CONTENT -->
  </body>
</html>

```

Gambar 2. 11. *Block Level HTML*.

b) *CSS (Cascading Style Sheet)*

CSS adalah bahasa *style sheet* untuk dapat mengatur sebuah tampilan pada website agar dapat lebih indah dilihat. *HTML* dan *CSS* memiliki keterikatan yang erat. Karena *HTML* adalah fondasi situs dan *CSS* adalah untuk semua aspek yang terkait dengan tampilan *website*, maka kedua bahasa pemrograman ini harus berjalan beriringan (Christy 2019). Untuk cara kerjanya adalah sebagai berikut.

```

<style>
p {
  color: blue;
  text-weight: bold;
}
</style>

```

Gambar 2. 12. Cara kerja *CSS*.

Dimana gambar tersebut menjelaskan yang akan diubah *style*-nya adalah paragraf dengan berwarna biru dan dengan diberikan ketebalan *bold*.

c) *PHP*

PHP sendiri merupakan bahasa pemrograman *server side scripting* yang bersifat *Open source* (Awwaabiin 2020). Pada umumnya *PHP* berfungsi untuk pengembangan sebuah *website* baik itu *website* sederhana hingga *website* yang memiliki fitur yang berlimpah. *PHP* bukan hanya berfungsi dalam pengembangan *website* saja tetapi dapat membuat sebuah aplikasi komputer. Seperti halnya bahasa pemrograman yang lainnya yang *PHP* memiliki aturan dalam membuat programnya, *PHP* memiliki aturan yang harus menuliskan `<?php ?>` untuk dapat menjalankan bahasa pemrograman *PHP*. Salah satu sintaks sederhana dalam *PHP* adalah sebagai berikut.

```
<?php
echo "Selamat datang";
?>
```

Gambar 2. 13. Program sederhana PHP.

Gambar 2.13. menjelaskan bagaimana penggunaan sebuah bahasa *PHP* dimana *echo* merupakan sebuah perintah untuk menampilkan sebuah *teks* yang bertuliskan selamat datang dengan diakhiri sebuah titik koma.

d) *JavaScript*

JavaScript adalah bahasa pemrograman berorientasi objek yang berbasis model *prototipe* objek (Mikkonen & Taivalaari, 2007). *JavaScript* juga merupakan bahasa pemrograman komputer yang dinamis, ringan dan paling banyak digunakan sebagai bagian dari perancangan halaman *web* maupun *mobile* (Tutorialspoint, 2015). Bahasa komputer sendiri merupakan serangkaian instruksi yang memberitahu komputer untuk melakukan sesuatu hal yang diinstruksi (Wilton & Mcpeak, 2010). Biasanya instruksi atau kode diproses dari baris atas ke bawah. Sederhananya komputer melihat kode yang anda tulis dan mencari tau tindakan apa yang ingin anda lakukan kemudian komputer mengambil tindakan itu. Tindakan yang dimaksud adalah menjalankan atau mengeksekusi pada kode yang diproses. *JavaScript* merupakan bahasa yang ditafsirkan dari bahasa yang komplikasi, maksudnya adalah saat *JavaScript* digunakan maka komputer tidak langsung memahami bahasa *JavaScript* tetapi harus terlebih dahulu diterjemahkan, biasanya saat *browser* melewati bahasa *JavaScript* maka *browser* akan meneruskan ke program khusus yang disebut *interpreter* untuk mengubah bahasa *JavaScript* ke kode mesin komputer anda (Wilton & Mcpeak, 2010).

e) *Framework CodeIgniter 4*

CodeIgniter adalah sebuah kerangka pengembangan aplikasi *website* dengan menggunakan bahasa pemrograman dasar *PHP* dengan telah menyediakan *libraries* yang berlimpah sesuai yang dibutuhkan secara umum dan juga tampilan antarmuka yang sederhana dan struktur serta logis dalam menggunakannya (CodeIgniter 2021). *CodeIgniter 4* sendiri merupakan versi terbaru dari *CodeIgniter* yang baru dirilis pada Februari 2020 lalu setelah

update versi terakhir adalah 4 tahun silam dengan nama CodeIgniter 3 (ilmucoding 2020). *Framework* CodeIgniter 4 menjadi salah satu *Framework* yang digemari karena kelengkapan dari setiap *libraries* dengan telah dikelompokkan setiap apa yang dibutuhkan sehingga mempermudah pekerjaan serta mempersingkat waktu bagi para programmer. Kemudian perkembangan antara CodeIgniter 3 dengan CodeIgniter 4 cukup banyak seperti *Restful API* yang dapat mempermudah serta memperluas pergerakan yang ingin dilakukan oleh *programmer*.

f) *MySQL Database*

MySQL adalah sebuah metode penyimpanan data *Database Management System* yang menggunakan perintah dari SQL (*Structured Query Language*). MySQL termasuk ke dalam *Relational Database Management System* dimana dapat menggunakan tabel, kolom, baris pada struktur database-nya sehingga

- proses pengambilan data menggunakan metode relational database serta dapat menjadi penghubung antara perangkat lunak dan database server (Adani 2020).

Fungsi dari MySQL merupakan pengelolaan sebuah database server yang berisi informasi yang menggunakan sistem SQL. Serta mempermudah bagi para pengguna untuk dapat menggunakannya karena dalam mengakses informasi dapat dilihat dalam bentuk teks.