

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Kajian Pustaka**

Dalam pembuatan aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta ini menggunakan beberapa penelitian yang terkait sebagai dasar dari penelitian ini sebagai berikut.

Penelitian Yusriannur (2012), dalam penelitiannya tentang aplikasi *e-voting* berbasis web menyatakan dengan *e-voting* diharapkan minat mahasiswa untuk berpartisipasi memberikan suara semakin banyak dan proses pengolahan suara semakin cepat. Aplikasi *e-voting* ini dibuat dengan menggunakan bahasa pemrograman framework PHP, menggunakan database MySQL dan metode pengembangan *system web engineering*. Sistem ini mempermudah mahasiswa dalam memberikan hak suaranya dimanapun berada tanpa harus mendatangi kampus dan membantu panitia dalam pemrosesan suara yang masuk secara cepat, tepat dan efisien.

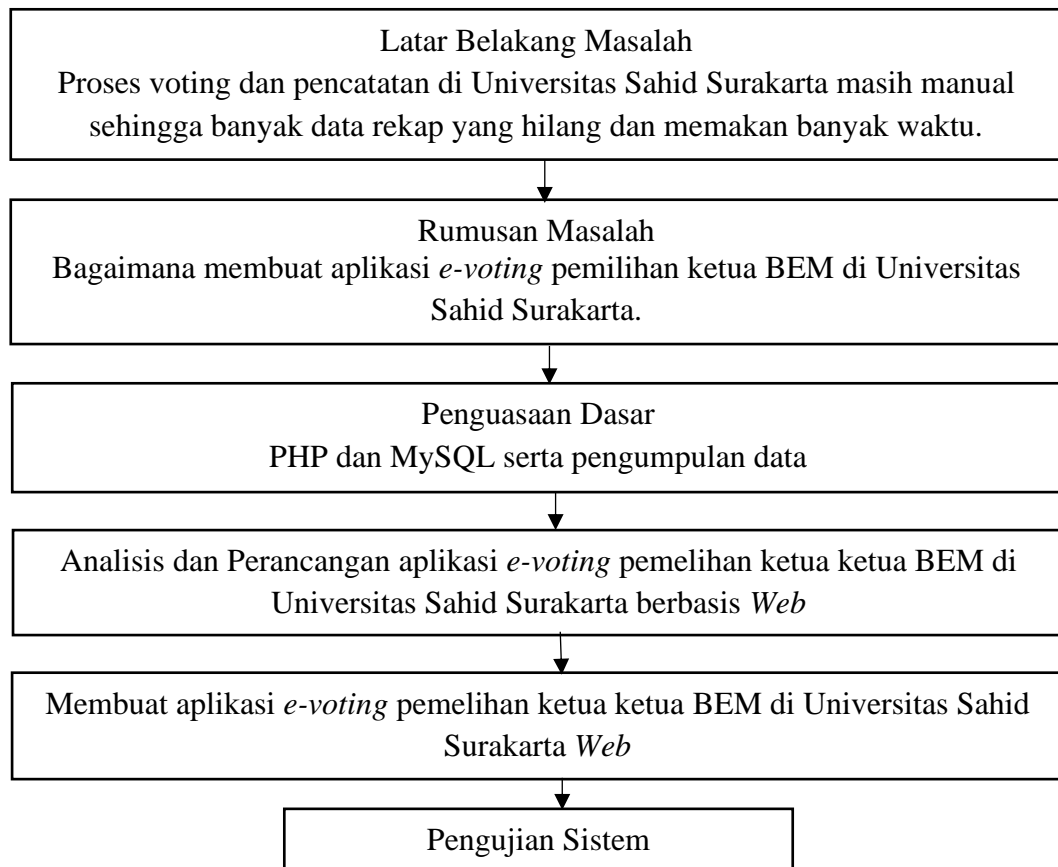
Penelitian Aditiya Wari Nugroho (2011), Perancangan *e-voting* Berbasis WEB (Studi Kasus Pemilihan Kepala Daerah Sukoharjo) Penelitian tersebut mengenai aplikasi untuk melakukan voting kepala daerah Sukoharjo. Aplikasi yang dibangun menggunakan PHP dan MySQL sebagai database server. Dalam aplikasi voting tersebut, pemilih melakukan login dengan menggunakan nomor KTP yang dimilikinya.

Penelitian dari Yuniarti (2015), dalam penelitiannya tentang aplikasi pendaftaran dan pemilihan pengurus osis menyatakan aplikasi yang dibangun dapat mempermudah user dalam penggunaan, membantu panitia (admin) dalam mengelola data dan mempersiapkan waktu kegiatan pemilihan yang akan diselenggarakan disekolah, yang lebih efisien dan efektif, dan mampu memberikan alternatif serta membantu pihak sekolah dalam pemilihan pengurus OSIS. Aplikasi pendaftaran dan pemilihan pengurus OSIS dibuat dengan menggunakan bahasa pemrograman framework PHP *open source* dengan

menggunakan konsep MVC (*Model View Controller*) dan menggunakan database MySQL.

## 2.2 Kerangka Pemikiran

Berikut ini adalah tahapan kerangka pemikiran dalam membuat aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta. pada Gambar 2.1 berikut.



Gambar 2.1 Kerangka Pemikiran

Penjelasan dari kerangka pemikiran tersebut adalah :

### 1. Latar Belakang Masalah

Melakukan *voting* untuk pemilihan ketua BEM di Universitas Sahid Surakarta masih menggunakan cara manual. Proses ini membutuhkan waktu yang lama karena admin harus mengumpulkan dan menyimpan data secara manual.

### 2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan maka rumusan masalahnya adalah bagaimana membuat aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta.

### 3. Penguasaan dasar PHP dan MySQL

Penguasaan dasar PHP dan MySQL merupakan suatu kunci terciptanya aplikasi *e-voting* pemilihan ketua BEM tersebut, karena pembuatan aplikasi ini menggunakan bahasa pemrograman PHP dan menggunakan database MySQL sebagai media penyimpan datanya, sehingga semakin mengerti tentang dasar PHP dan MySQL pembuatan sistem menjadi lebih cepat.

### 4. Pengumpulan data

Tahap pengumpulan data pada penelitian ini melalui observasi, wawancara dan studi literatur. Observasi dilakukan di sekitar kampus Universitas Sahid Surakarta, wawancara akan dilakukan kepada staf informasi dan mahasiswa. Pengumpulan data bertujuan untuk mengetahui permasalahan dan kebutuhan informasi mengenai *e-voting* di Universitas Sahid Surakarta

### 5. Analisis dan perancangan sistem

Analisis dan perancangan aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta berbasis *web* bertujuan untuk mengetahui kesiapan data dan rancangan dari aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta yang diharapkan dapat memberikan suatu informasi dengan efektif. Perancangan sistem pada penelitian ini menggunakan UML (*Unified Modeling Language*).

### 6. Rancang Bangun Sistem

Aplikasi *e-voting* pemilihan ketua BEM di Universitas Sahid Surakarta menggunakan bahasa pemrograman PHP dan database MySQL sebagai media penyimpanan datanya.

### 7. Pengujian Sistem

Pengujian sistem merupakan tahap setelah sistem berhasil dibuat. Pengujian sistem bertujuan untuk mengetahui kelebihan, kelemahan serta mengetahui kelayakan suatu sistem untuk digunakan. Pengujian sistem pada penelitian ini

menggunakan metode *BlackBox* karena pengujian pada sistem ini lebih mengutamakan fungsionalitas aplikasi.

## **2.3 Landasan Teori**

### **2.3.1 Sistem**

Pengertian sistem menurut Romney dan Steinbart (2015), sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari sub sistem yang lebih kecil yang mendukung sistem yang lebih besar.

Definisi sistem menurut Mulyadi (2016), sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan.

Menurut Susanto (2013), sistem adalah kumpulan atau grup dari sub sistem/bagian/komponen apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. Berdasarkan pengertian di atas dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

### **2.3.2 Informasi**

Pengertian menurut Krismaji (2015), informasi adalah data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat. Sementara menurut Romney dan Steinbart (2015), informasi adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan. Sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi.

Berdasarkan pengertian di atas dapat disimpulkan bahwa pengertian informasi adalah data yang diolah agar bermanfaat dalam pengambilan keputusan bagi penggunanya.

### 2.3.3 Sistem Informasi

Pengertian menurut Kadir (2014), sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai.

Pengertian menurut Krismaji (2015), sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan, dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah ditetapkan.

Jadi berdasarkan pengertian di atas dapat disimpulkan bahwa sistem informasi adalah kumpulan data yang terintegritasi dan saling melengkapi dengan menghasilkan output yang baik guna untuk memecahkan masalah dan pengambilan keputusan.

### 2.3.4 Website

#### 1. Konsep dasar *Web*

Menurut Ardhana (2012), *web* adalah salah satu layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui internet).

Menurut Risnandar, dkk. (2013), *web* adalah salah satu fitur dari internet di samping fitur-fitur lain seperti *e-mail*, *remote* (SSH), dan FTP, berdasarkan uraian di atas dapat disimpulkan *web* sebagai salah satu fitur dari internet dan layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui internet).

#### 2. *Website*

Menurut Yuhefizar (2013), *website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan

antara satu halaman *web* dengan halaman *web* yang lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

Menurut Hikmah, dkk (2015), *website* atau situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Berdasarkan pengertian para ahli di atas dapat disimpulkan bahwa *website* adalah kumpulan dari halaman-halaman *web* yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

### 3. *Web Server*

Menurut Supono dan Putratama (2016), *web-server* adalah perangkat lunak *server* yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien itu, yang dikenal sebagai *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML.

Menurut Sibero (2013), *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak server yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML.

### 4. *Web Browser*

Menurut Sibero (2012), *web browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *web*.

Menurut Winarno (2007), *web browser* merupakan suatu program yang dirancang untuk mengambil informasi-informasi dari suatu *server* komputer pada

jaringan internet. Jadi untuk mengakses *web* diperlukan suatu program yaitu *web browser* atau disebut *browser* saja.

Berdasarkan pengertian di atas dapat disimpulkan bahwa *browser* adalah sebuah perangkat lunak yang dirancang untuk menampilkan dan mengakses informasi atau halaman-halaman yang tersedia di *web server*.

### 2.3.5 PHP (*Hypertext Preprocessor*)

Menurut Hidayatullah dan Kawistara (2017), PHP atau disingkat *Hypertext Preprocessor* ini adalah sebuah Bahasa *scripting* khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan PHP harus menggunakan *web server*.

PHP juga dapat diintegrasikan dengan HTML, JavaScript, JQuery, Ajax. Namun, pada umumnya PHP lebih banyak digunakan bersamaan dengan file bertipe HTML. Dengan menggunakan PHP bisa membuat *website powerful* yang dinamis dengan disertai manajemen *database*-nya. Selain itu juga penggunaan PHP yang sebagian besar dapat jalan di banyak *platform*.

### 2.3.6 HTML (*Hyperlink Text Markup Language*)

Menurut Sidik dan Husni (2017), HTML kependekan dari *Hyperlink Text Markup Language*. Dokumen HTML adalah file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet.

Menurut Sibero (2013), *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*.

Berdasarkan uraian di atas maka kesimpulannya *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan untuk menulis halaman *web* sebagai pertukaran dokumen *web* dan dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet.

### 2.3.7 Internet

Menurut Ahmadi dan Hermawan (2013), internet adalah komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin.

Sedangkan Menurut Sibero (2013), internet (*interconnected network*) adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat juga disebut jaringan dalam suatu jaringan yang luas.

Berdasarkan kesimpulan di atas internet adalah sistem jaringan komunikasi secara global yang menghubungkan seluruh komputer di dunia dengan waktu dan wilayah tak terbatas sehingga semua orang dapat selalu terhubung di waktu siang maupun malam dengan platform apapun.

### 2.3.8 Definisi Basis Data

Definisi menurut Hidayatullah dan Kawistara (2017), basis data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah. Prinsip utamanya adalah pengaturan data. Tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data.

Secara lebih lengkap pemanfaatan data dilakukan untuk memenuhi berikut ini :

1. Kecepatan dan Kemudahan (*Speed*)
2. Efisiensi ruang penyimpanan (*Space*)
3. Keakuratan (*Accuracy*)
4. Ketersediaan (*Availability*)
5. Kelengkapan (*Completeness*)
6. Keamanan (*Security*)
7. Pemakaian Bersama (*Sharability*)



### 2.3.9 MySQL

Menurut Sibero (2013), MySQL atau dibaca “My Sekuel” dengan suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data.

Menurut Hidayatullah dan Jauhari (2015), MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi *web*. Contoh DBMS lainnya adalah PostgreSQL (freeware), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dsb.

Berdasarkan penjelasan di atas dapat disimpulkan bahwa MySQL adalah aplikasi DBMS yang menjalankan fungsi pengelolaan data untuk membangun sebuah aplikasi berbasis *web* maupun desktop.

### 2.3.10 Metode Angket atau Kuesioner

Menurut Sugiyono (2014), kuesioner merupakan teknik pengumpulan data yang efisien apabila peneliti tahu dengan siapa variabel akan diukur dan tahu apa yang bisa diharapkan dari responden. Kuesioner dapat berupa pertanyaan-pertanyaan tertutup atau terbuka, dapat diberikan kepada responden secara langsung atau dikirim melalui pos atau internet. Angket keterampilan menjelaskan dan bertanya, untuk mengetahui persepsi siswa 42 mengenai keterampilan mengajar guru di kelas. Dalam angket tersebut terdapat komponen-komponen yang dapat dinilai oleh siswa terhadap kemampuan guru dalam keterampilan menjelaskan dan bertanya selama pembelajaran berlangsung. Untuk angket data yang diperoleh angket berupa nilai skor. Untuk menentukan skor pilihan jawaban angket menggunakan skala *Likert*. Dikemukakan Sugiyono (2014) bahwa skala *Likert* digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Kriteria pemberian skor untuk alternatif jawaban untuk setiap item sebagai berikut : (1) Skor 4 untuk jawaban sangat setuju, (2) Skor 3 untuk jawaban setuju, (3) Skor 2 untuk jawaban tidak setuju, (4) skor 1 untuk jawaban sangat tidak setuju.

### 2.3.11 *Black Box Testing*

Pengujian menggunakan sekumpulan aktifitas validasi, dengan pendekatan *black box testing*. Menurut Shalahuddin dan Sukamto (2011), *black box testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai

### 2.3.12 *Framework*

*Framework* menurut Hidayatullah dan Kawistara (2017), *framework* dapat diartikan sekumpulan perintah/program dasar di mana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa harus membuat program dari awal.

### 2.3.13 **PHPMyAdmin dan XAMPP**

#### 1. PHPMyAdmin

Nugroho (2004), PhpMyAdmin adalah suatu program *open source* yang berbasis *web* yang dibuat menggunakan aplikasi PHP. Program ini digunakan untuk mengakses *database* MySQL dan untuk mempermudah dan mempersingkat kerja kita. Dengan kelebihanannya, para pengguna awam tidak harus paham sintax-sintax SQL dalam pembuatan *database* dan tabel.

Hidayatullah dan Kawistara (2014) menyatakan, PHPMyAdmin adalah *tool open source* yang ditulis dalam bahasa PHP untuk menangani administrasi MySQL berbasis *world wide web*.

#### 2. XAMPP

Menurut Purbadian (2016), berpendapat bahwa, Xampp merupakan suatu *software* yang bersifat *open source* yang merupakan pengembangan dari LAMP (Linux, Apache, MySQL, PHP dan Perl).

Menurut Kartini (2013), Xampp merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket.

Sedangkan menurut Riyanto (2010), Xampp merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP.

#### 2.3.14 Sublime Text

Menurut Supono (2016), sublime text merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi. Sublime text mempunyai fitur *plugin* tambahan yang memudahkan programmer. Selain itu, sublime text terkesan elegan untuk sebuah *syntax editor*. Selain ringan, IDE ini memiliki kecepatan proses simpan dan buka file. Tidak heran kalau IDE ini paling banyak digunakan terutama dikalangan programmer berbasis *web*.

Berikut keunggulan-keunggulan fitur yang dimiliki Sublime Text 3, yaitu:

a. *Multiple Selection*

*Multiple Selection* mempunyai fungsi untuk membuat perubahan pada sebuah kode pada waktu yang sama dan dalam baris yang berbeda. Multiple selection ini juga merupakan salah satu fitur unggulan dari Sublime Text 3. Kita dapat meletakkan kursor pada kode yang akan di ubah/edit, lalu tekan Ctrl+klik atau blok kode yang akan diubah kemudian Ctrl+D setelah itu kita dapat merubah kode secara bersamaan.

b. *Command Pallete*

*Command Pallete* mempunyai fungsi yang berguna untuk mengakses file shortcut dengan mudah. Untuk mencari file tersebut kita dapat tekan Ctrl+Shift+P, kemudian cari perintah yang kita inginkan.

c. *Distraction Free Mode*

Fitur ini mempunyai fungsi untuk merubah tampilan layar menjadi penuh dengan menekan Shift+F11. Fitur ini sangat dibutuhkan ketika pengguna ingin fokus pada pekerjaan yang sedang dikerjakannya.

d. *Find in project*

Fitur ini kita dapat mencari dan membuka file di dalam sebuah project dengan cepat dan mudah. Hanya dengan menekan Ctrl+P anda dapat mencari file yang diinginkan.

e. *Plugin API Switch*

Sublime Text mempunyai keunggulan dengan plugin yang berbasis Python Plugin API. Teks editor ini juga mempunyai plugin yang sangat beragam, dan ini dapat memudahkan pengguna dalam mengembangkan softwarena.

f. *Drag and Drop*

Dalam teks editor ini pengguna dapat menyeret dan melepas file teks ke dalam editor yang akan membuka tab baru secara otomatis.

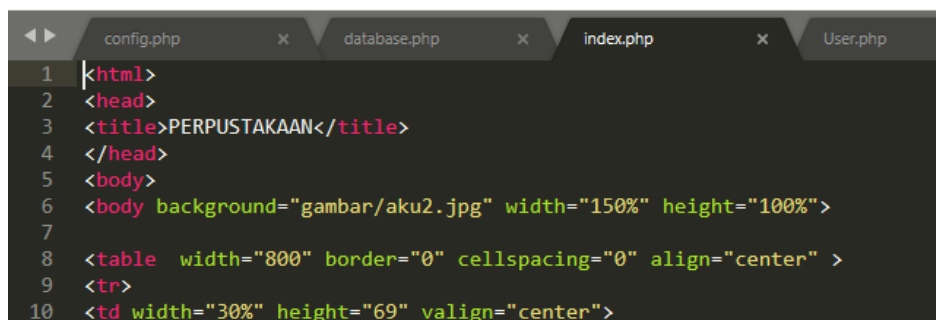
g. *Split Editing*

Di dalam fitur ini pengguna dapat mengedit file secara berdampingan dengan klik File->New menu into file.

h. *Multi Platform*

Sublime Text juga mempunyai keunggulan dalam berbagai platform. Sublime text sendiri sudah tersedia dalam berbagai platform sistem operasi, yaitu Windows, Linux dan MacOS.

Berikut ini adalah contoh *script* menggunakan *software* sublime seperti pada Gambar 2.2:



```
1 <html>
2 <head>
3 <title>PERPUSTAKAAN</title>
4 </head>
5 <body>
6 <body background="gambar/aku2.jpg" width="150%" height="100%">
7
8 <table width="800" border="0" cellspacing="0" align="center" >
9 <tr>
10 <td width="30%" height="69" valign="center">
```

Gambar 2.2 *Script* menggunakan *software* sublime

### 2.3.15 Codeigniter

Pengertian codeigniter menurut Hidayatullah dan Kawistara (2017), codeigniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling *powerful* saat ini karena di dalamnya terdapat fitur lengkap aplikasi *web* di mana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu codeigniter juga saat ini banyak digunakan khususnya bagi *developer web* untuk mengembangkan aplikasi berbasis *webnya* tersebut.

### 2.3.16 Adobe Dreamweaver

Menurut Sibero (2013), adobe dreamweaver adalah suatu produk *web developer* yang dikembangkan oleh Adobe System Inc. Sebelumnya produk dreamweaver dikembangkan oleh Macromedia Inc. Adobe dreamweaver merupakan suatu program yang digunakan dalam membangun sebuah *website* secara grafis.

### 2.3.17 Bootstrap

Menurut Husein (2013), bootstrap merupakan *framework* ataupun *tools* untuk membuat aplikasi *web* ataupun *web responsive* secara tepat, mudah, dan gratis. Bootstrap terdiri dari CSS (*Cascading Style Sheet*) dan HTML (*Hypertext Markup Language*) untuk menghasilkan *grid*, *layout*, *typography*, *table*, *form*, *navigation*, dan lain-lain. Di dalam bootstrap juga sudah terdapat *JQuery plugins* untuk menghasilkan komponen UI yang menarik.

### 2.3.18 Pengertian UML

*Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi

penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Sukamto dan Shalahudin, 2014:133).


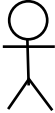

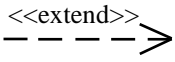
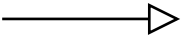
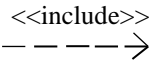
Sukamto dan Shalahudin (2014), pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. *Structure diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Structure diagram terdiri dari *class diagram*, *object diagram*, *component diagram*, *composite structure diagram*, *package diagram* dan *deployment diagram*.
2. *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. *Behavior diagram* terdiri dari *use case diagram*, *activity diagram*, *state machine system*.
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar sub sistem pada suatu sistem. *Interaction diagram* terdiri dari *sequence diagram*, *communication diagram*, *timing diagram*, *interaction overview diagram*

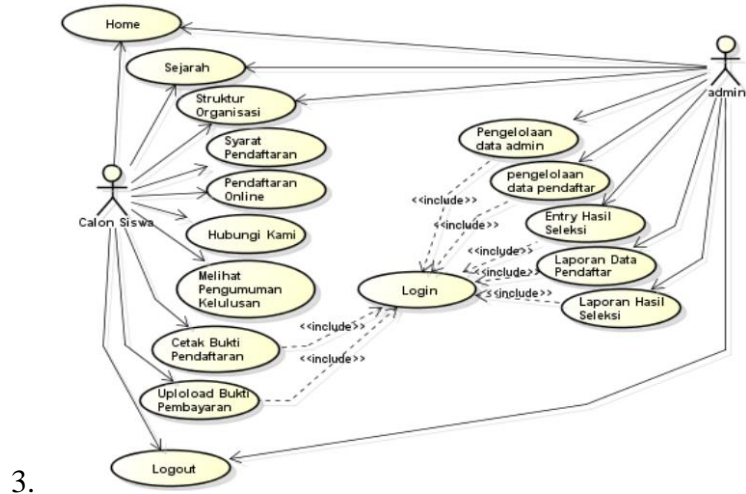
### **2.3.19 Use Case Diagram**

Sukamto dan Shalahudin (2014), *use case* atau *diagram use case* merupakan 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 di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Simbol *use case diagram* disajikan pada Tabel 2.1.

Tabel 2.1 Simbol *Use Case Diagram* (Sukamto dan Shalahuddin, 2013).

NO	SIMBOL	NAMA	KETERANGAN
1		<i>Use case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan kata kerja di awal di awal frase nama <i>use case</i> .
2		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5		<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

Gambar 2.3 adalah contoh *use case diagram* dari Eviani dkk. (2016) :



Gambar 2.3 Contoh *use case diagram*

### 2.3.20 Pengertian *Class Diagram*

Tabel 2.2 Simbol *Class Diagram* (Sukamto dan Shalahuddin, 2013).

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang
3		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sukamto dan Shalahudin (2014), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan

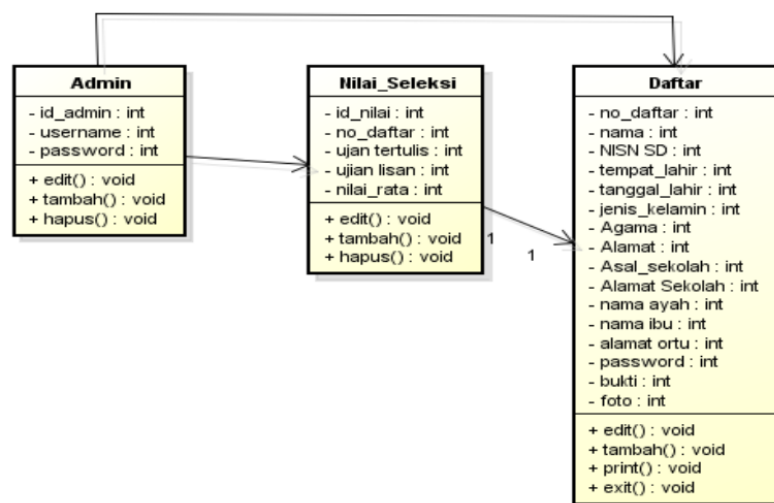


dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan method atau operasi. Berikut penjelasan atribut dan method:

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau method adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Simbol *class diagram* disajikan pada Tabel 2.2.

Gambar 2.4 adalah contoh *class diagram* dari Eviani dkk. (2016) :



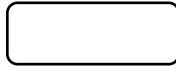





Gambar 2.4 Contoh *class diagram*

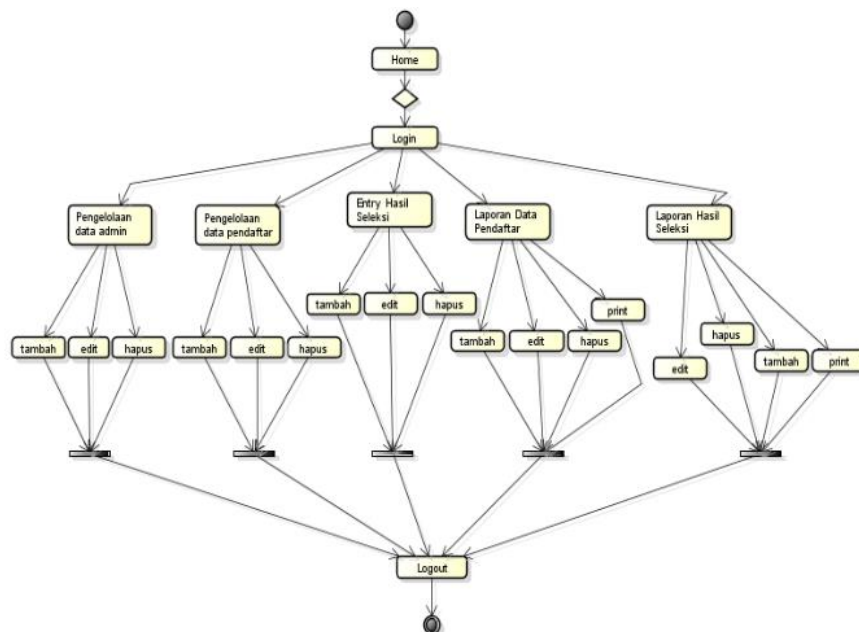
### 2.3.21 Pengertian *Activity Diagram*

Sukamto dan Shalahudin (2014), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol *activity diagram* disajikan pada Tabel 2.3.

Tabel 2.3 Simbol *Activity Diagram* (Sukamto dan Shalahuddin, 2013).

No	Gambar	Nama	Keterangan
1		<i>Start Point</i>	Diletakkan pada objek kiri atas dan merupakan awal aktivitas
2		<i>End Point</i>	Akhir aktivitas
3		<i>Activities</i>	Menggambarkan suatu proses atau kegiatan bisnis
4		<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau digunakan untuk menggabungkan dua kegiatan menjadi satu
5		<i>Join</i>	Digunakan untuk menunjukkan dekomposisi
6		<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan true atau false


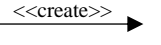

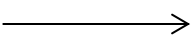
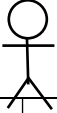

Gambar 2.5 adalah contoh *activity diagram* dari Eviani dkk. (2016) :

Gambar 2.5 Contoh *activity diagram*

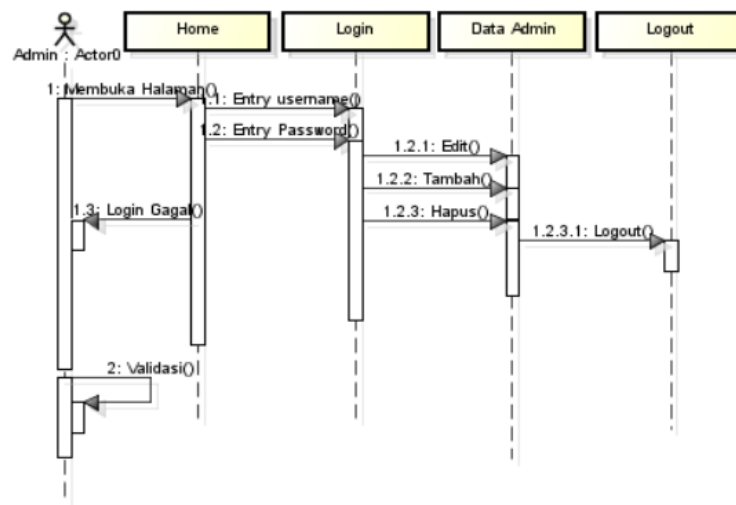
### 2.3.22 Pengertian *Sequence Diagram*

Sukamto dan Shalahudin (2014), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksinya jalannya pesan sudah dicakup dalam diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak. *Sequence diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.4.

Tabel 2.4 Simbol *Sequence Diagram* (Sukamto dan Shalahuddin, 2013).

No	Gambar	Nama	Keterangan
1		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
2		Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
3		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
4		<i>A line of life</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi
5		<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem
6		<i>A focus of control</i>	Menggambarkan tempat mulai dan berakhirnya sebuah <i>message</i> (pesan).

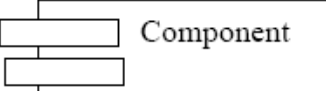
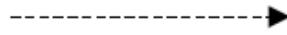
Gambar 2.6 adalah contoh *sequence diagram* dari Eviani dkk. (2016) :



Gambar 2.6 Contoh *sequence diagram*

### 2.3.23 Pengertian *Component Diagram*

Tabel 2.5 Simbol *Component Diagram* (Sukamto dan Shalahudin, 2014).

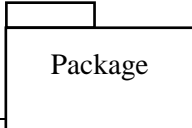
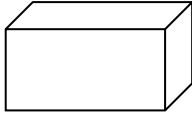
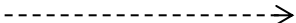

SIMBOL	NAMA	KETERANGAN
 Component	komponen	Sebuah komponen melambangkan sebuah entitas software dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.
	Depedency	Sebuah Dependency digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.

*Component diagram* menurut Sukamto dan Shalahudin (2014), diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem.

Hubungan antara *component* dan *class*, *Component* adalah implementasi *software* dan sebuah *class*. *Class* mewakili abstraksi dari serangkaian *attribute* dan *operation*. Hal terpenting yang perlu diingat tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi dari lebih dari sebuah *class*. *Component diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.5.

### 2.3.24 Pengertian *Deployment Diagram*

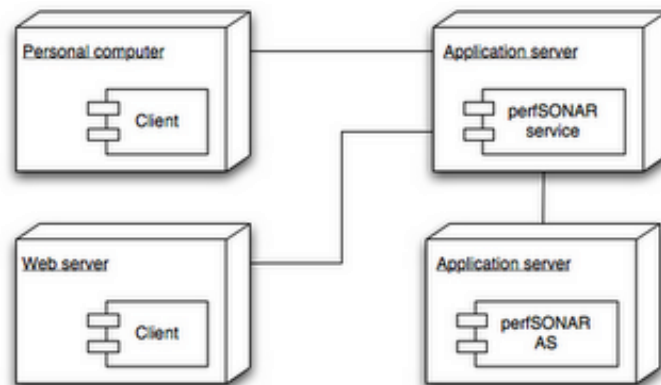
Tabel 2.6 Simbol *Deployment Diagram* (Sukamto dan Shalahudin, 2014).

SIMBOL	NAMA	KETERANGAN
	<i>Package</i>	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih komponen.
	<i>Node</i>	<i>Node</i> biasa mengacu pada perangkat keras ( <i>hardware</i> ), perangkat lunak yang tidak dibuat sendiri ( <i>software</i> ), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang telah didefinisikan sebelumnya pada diagram komponen.
	<i>Dependency</i>	Ketergantungan atau <i>dependency</i> atau kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
	<i>Link</i>	Menggambarkan relasi antar <i>node</i> .

Gambar Menurut Sukamto dan Shalahuddin (2014), mengatakan bahwa diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Secara umum pengertian diagram komponen adalah menggambarkan struktur fisik dari sebuah sistem dan digunakan untuk mengilustrasikan bagaimana kode program dibagi menjadi beberapa komponen, dan mendeksripsikan hubungan antar komponen. Diagram komponen memberikan gambaran tingkat tinggi dan sederhana dari sebuah sistem.

Mengklasifikasikan sekelompok kelas menjadi komponen akan membantu pertukaran dan penggunaan kembali dari sebuah kode. Tujuan dari diagram komponen adalah menunjukkan hubungan struktural antar komponen dalam sebuah sistem. *Deployment diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.6.

2.6 adalah contoh *Deployment Diagram* dari novian adi. (2012) :



Gambar 2.6 Contoh *Deployment Diagram*