

BAB II

LANDASAN TEORI

2.1 Pengertian Perancangan

Wahyu Hidayat dkk dalam jurnal CERITA (2016:49), “Perancangan adalah proses merencanakan segala sesuatu terlebih dahulu. Perancangan merupakan wujud visual yang dihasilkan dari bentuk-bentuk kreatif yang telah direncanakan. Langkah awal dalam perancangan desain bermula dari hal-hal yang tidak teratur berupa gagasan atau ide-ide kemudian melalui proses penggarapan dan pengelolaan akan menghasilkan hal-hal yang teratur, sehingga hal-hal yang sudah teratur bisa memenuhi fungsi dan kegunaan secara baik. Perancangan merupakan penggambaran, perencanaan, pembuatan sketsa dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi”.

Kristanto (2013:61)“Perancangan sistem adalah suatu fase dimana diperlukan suatu keahlian perancangan untuk elemen-elemen komputer yang akan menggunakan sistem, yaitu pemilihan peralatan dan program komputer untuk sistem yang baru.

jadi perancangan adalah proses mendesain spesifikasi baru yang dapat menyelesaikan masalah-masalah berdasarkan rekomendasi analisis untuk mendapatkan sistem baru di dalam sistem

2.2 Pengertian Sistem

Menurut L Enger dalam Tata Sutabri (2012: 9) mengatakan bahwa “System dapat terdiri dari atas kegiatan-kegiatan yang berhubungan guna mencapai tujuan-

tujuan perusahaan seperti pengendalian inventaris atau penjadwalan produksi “. Selanjutnya Menurut Gordon B. Davis dalam Tata Sutabri (2012: 9) menyatakan “ System bisa seperti abstract atau fisik, sistem yang abstract adalah susunan gagasan-gagasan atau konsep yang teratur yang saling bergantung, sedangkan sistem yang bersifat fisik adalah serangkaian unsur yang bekerja sama untuk mencapai tujuan” Sedangkan menurut Suatu sistem sangatlah dibutuhkan dalam suatu perusahaan atau instansi karena sistem sangat menunjang kinerja perusahaan atau instansi, baik yang berskala kecil maupun besar. buku Bambang Hartono (2013:10).

2.3 Hidroponik

2.3.1 Pengertian Hidroponik

Menurut Setiawan, Hendra(2017,h.1) Hidroponik merupakan salah satu seni menanam tumbuhan tanpa menggunakan media tanah dan memanfaatkan air yang diperkaya dengan unsur hara atau nutrisi yang dibutuhkan tanaman.

2.3.2 Pengertian Fogponik

Menurut Setiawan, Hendra(2017,h.35) Aeroponik merupakan suatu sistem penanaman yang paling baik dengan menggunakan udara dan ekosistem air tanpa menggunakan tanah.

Prinsip kerja tanaman aeroponik adalah dengan memanfaatkan air dan nutrisi yang di berikan ke tanaman dalam bentuk butiran kecil ataupun kabut. Dengan teknik ini, tanaman dapat mendapatkan oksigen dan nutrisi pada saat yang bersamaan dan hasil dari tanama yang dikembangkan dengan teknik ini terbukti memiliki kualitas yang

baik, higienis, segar, renyah, beraroma dan disertai juga dengan citra rasa yang tinggi

2.4 Pengertian Prototype

Menurut Darmawan (2013, h.229),”Prototipe adalah satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam bentuk yang telah selesai”.

Sebaliknya disisi pengembang kurang memperhatikan efesiensi algoritma, kemampuan sistem operasi dan interface yang menghubungkan manusia dan komputer. Untuk mengatasi ketidakserasian antara pelanggan dan pengembang harus dibutuhkan kerjasama yang baik diantara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis dan pelanggan akan mengetahui proses-proses dalam menyelesaikan sistem yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

Model prototipe dapat berupa tiga bentuk :

1. Prototipe kertas atau model berbasis komputer yang menjelaskan bagaimana interaksi antara pemakai dan komputer.
2. Prototipe yang mengimplementasikan beberapa bagian fungsi dari perangkat lunak yang sesungguhnya. Dengan cara ini pemakai akan

lebih mendapatkan gambaran tentang program yang akan dihasilkan, sehingga dapat menjabarkan lebih rinci kebutuhannya.

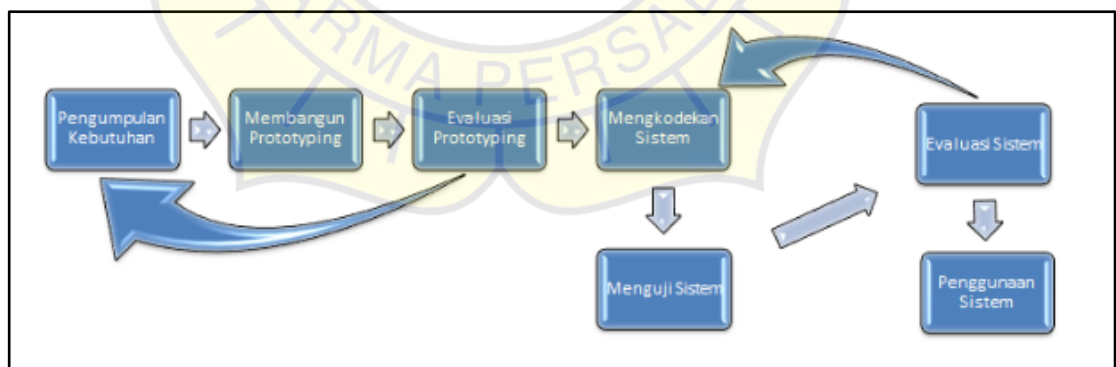
3. Menggunakan perangkat lunak yang sudah ada. Seringkali pembuat software memiliki beberapa program yang sebagian dari program tersebut mirip dengan program yang akan dibuat.

Di dalam proses pengembangan, sering kali pemakai / pelanggan hanya dapat mendefinisikan tujuan dan penggunaan software yang dibutuhkan, tetapi tidak dapat mendefinisikan secara rinci kebutuhan masukan, pengolahan, dan keluarannya. Di sisi lain, pembuat software tidak memiliki kepastian akan hal tersebut. Hal ini menyebabkan pengembang kurang memperhatikan efisiensi algoritma, kemampuan sistem operasi dan interface yang menghubungkan manusia dan komputer. Untuk menyelaraskan antara pelanggan dan pengembang, maka harus dibutuhkan kerjasama yang baik di antara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis. Dan pelanggan akan mengetahui proses-proses dalam menyelesaikan sistem yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

Kunci agar model prototype ini berhasil dengan baik adalah dengan mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang harus setuju bahwa prototype dibangun untuk mendefinisikan kebutuhan.

Prototyping merupakan Javascript Framework yang dibuat untuk lebih memudahkan proses dalam membangun aplikasi berbasis web. Metode prototyping sebagai suatu paradigma baru dalam pengembangan sistem informasi, tidak hanya sekedar suatu evolusi dari metode pengembangan sistem informasi yang sudah ada, tetapi sekaligus merupakan revolusi dalam pengembangan sistem informasi manajemen. Proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Pengumpulan kebutuhan: developer dan klien bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya;
2. Perancangan: perancangan dilakukan cepat dan rancangan mewakili semua aspek software yang diketahui, dan rancangan ini menjadi dasar pembuatan prototype;
3. Evaluasi Prototype: klien mengevaluasi prototype yang dibuat dan digunakan untuk memperjelas kebutuhan software.



Gambar 2.1 Alur *Prototype* menurut Hanif

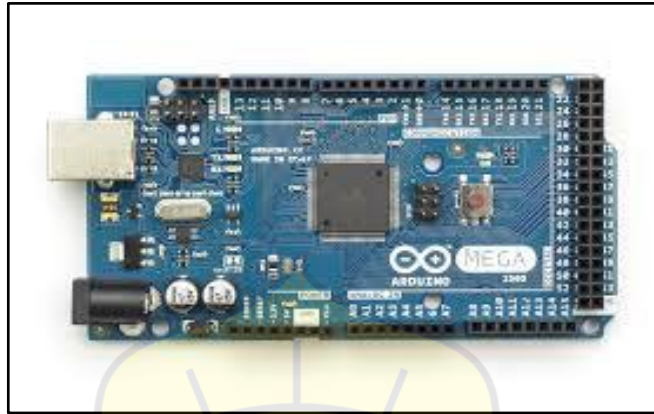
2.5 *Arduino*

Arduino dikenal sebagai papan elektronis yang didalamnya mengandung satu *microcontoller* buatan perusahaan Atmel dan berbagai peranti pendukung yang memungkinkan siapa saja dengan mudah dapat membuat berbagai proyek elektronika. Dalam sebuah sistem ini dibutuhkan sebuah ‘otak’ yang mengatur proses-proses di dalam sistem tersebut. Sebab sebelum ini telah menjelaskan bahwa salah satu contoh dari ‘otak’ tersebut adalah *mikrokontroler* yang umum digunakan adalah *Arduino Mega*. *Arduino Mega* merupakan *board mikrokontroler* yang berbasis *ATmega2560P*. *Mikrokontroler* ini memiliki 54 pin input/output digital (dimana 15 pin tersebut dapat digunakan sebagai output PWM), 16 input analog, sebuah *prosesor* 16 MHz Kristal kuarsa, sebuah *port* koneksi *USB*, sebuah *port power*, sebuah ICSP (*In Circuit Serial Programming*) *header* dan *button reset*. *Board Arduino* kompatibel dengan semua *shield* yang didesain untuk pendahulunya seperti *arduino* .

2.6 *Arduino Atmega 2560*

Arduino Mega 2560 adalah papan pengembangan *mikrokontroller* yang berbasis *Arduino* dengan menggunakan chip *ATmega2560*. Board ini memiliki pin I/O yang cukup banyak, sejumlah 54 buah digital I/O pin (15 pin diantaranya adalah PWM), 16 pin analog input, 4 pin *UART* (*serial port hardware*). *Arduino Mega 2560* dilengkapi dengan sebuah *oscillator* 16 Mhz, sebuah *port USB*, *power jack DC*, *ICSP header*, dan tombol *reset*. *Board* ini sudah sangat lengkap, sudah memiliki segala sesuatu yang dibuthkan untuk sebuah *mikrokontroller*. Dengan

penggunaan yang cukup sederhana, anda tinggal menghubungkan *power* dari *USB* ke *PC* anda atau melalui adaptor *AC/DC* ke *jack DC*.



Gambar 2.2 *Arduino Atmega 2560*

2.7 *Sensor Ping Ultrasonik Hc-sr04*

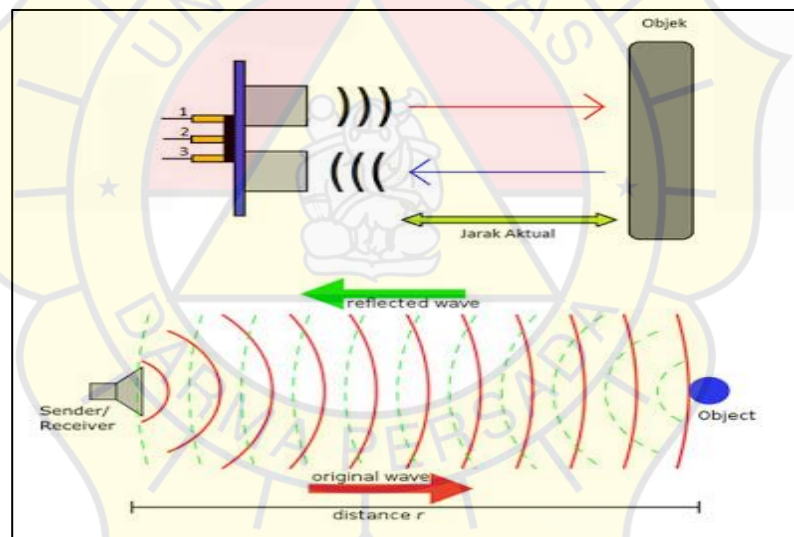
Sensor *ultrasonic* merupakan sensor yang bekerja dengan cara memancarkan suatu gelombang dan kemudian menghitung waktu pantulan gelombang tersebut, sensor yang di gunakan adalah sensor HC-SR04 yang dapat mengukur jarak benda dari 2cm - 4m dengan akurasi 3mm. Alat ini memiliki 4 pin, pin Vcc, Gnd, Trigger, dan Echo. Pin Vcc untuk listrik positif dan Gnd untuk groundnya. Pin Trigger untuk trigger keluarnya sinyal dari sensor dan pin Echo untuk penerima sinyal pantul dari benda. berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya.

Cara kerja sensor ini didasarkan pada prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi (jarak) suatu benda dengan frekuensi tertentu. Disebut sebagai sensor ultrasonik karena sensor ini menggunakan gelombang ultrasonik (bunyi ultrasonik). Sinyal dipancarkan oleh pemancar ultrasonik dengan frekuensi tertentu dan dengan durasi waktu tertentu.

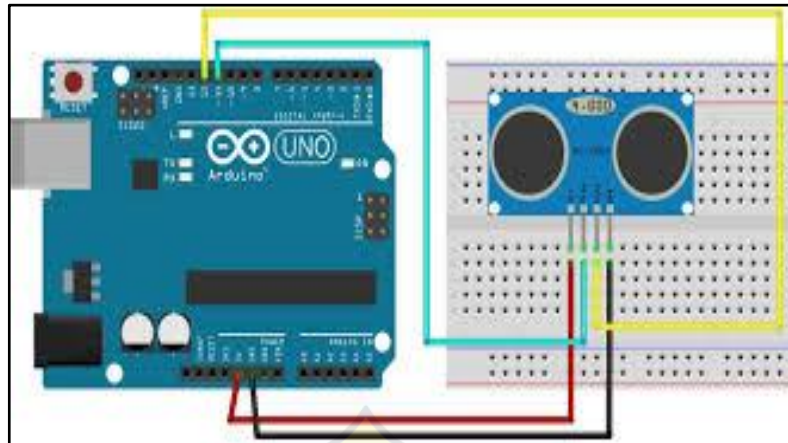
Sinyal tersebut berfrekuensi diatas 20kHz. Untuk mengukur jarak benda (sensor jarak), frekuensi yang umum digunakan adalah 40kHz.

Sinyal yang dipancarkan akan merambat sebagai gelombang bunyi dengan kecepatan sekitar 340 m/s. Ketika menumbuk suatu benda, maka sinyal tersebut akan dipantulkan oleh benda tersebut.

Setelah gelombang pantulan sampai di alat penerima, maka sinyal tersebut akan diproses untuk menghitung jarak benda tersebut. Jarak benda dihitung berdasarkan rumus $S = 340.t/2$, (S sama dengan jarak dan t sama dengan selisih waktu waktu dan V sama dengan cepat rambat gelombang ultrasonic (340m/s).)



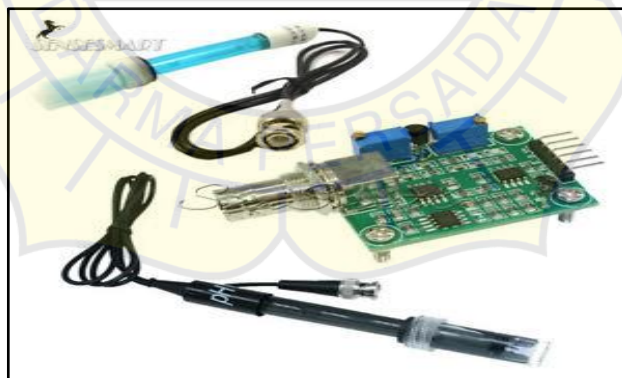
Gambar 2.3 Alur kerja *Sensor Ping Ultrasonik Hc-sr04*



Gambar 2.4 *Sensor Ping Ultrasonik Hc-sr04*

2.8 Sensor pH SEN0161-V2.

Jenis alat ukur untuk mengukur derajat keasaman atau kebasaan suatu cairan, pada pH meter digital terdapat elektroda khusus yang berfungsi untuk mengukur pH bahan-bahan semi padat, elektroda (probe pengukur) terhubung sebuah alat elektronik yang mengukur dan menampilkan nilai pH.



Gambar 2.5 alat sensor pH

Modul sensor ini difungsikan kedalam berbagai aplikasi seperti aquaponik, pengujian air lingkungan, hidroponik dan lain-lain.

Tipe SEN0161-V2 merupakan module versi kedua dimana telah diimprove dari segi tingkat kepresisian pembacaan datanya. Selain itu telah diimprove juga untuk

bagian hardware yang memiliki jitter yang rendah. Jitter itu merupakan perbedaan waktu antara pembacaan dan pengiriman sinyal. Secara pengertian, pH itu merupakan nilai yang digunakan untuk mengukur tingkat keasaman atau alkalinitas dalam suatu larutan. Range nilai PH yaitu antara angka antara 0 hingga 14 dengan ketentuan seperti pada tabel dibawah ini

Tabel 2.1 Range nilai PH

pH larutan	Sifat
$pH < 7$	Asam
$pH = 7$	Netral
$pH > 7$	Basa

2.9 Esp8266

Menurut Limantara et al. (2017, h.3), ESP8266 merupakan modul wifi yang berfungsi sebagai perangkat agar dapat terhubung langsung dengan wifi dan membuat koneksi TCP/IP.

Menurut (Arafat 2016, h.2), ESP 8266 adalah sebuah chip yang sudah lengkap dimana didalamnya sudah termasuk processor, memori dan juga akses ke GPIO. Hal ini menyebabkan ESP8266 dapat secara langsung menggantikan Arduino dan ditambah lagi dengan kemampuannya untuk mensupport koneksi wifi secara langsung.

Berdasarkan 2 (dua) definisi diatas dapat disimpulkan bahwa ESP8266 adalah modul wifi yang dapat menghubungkan perangkat melalui koneksi IP.

2.10 Perangkat lunak yang digunakan

2.10.1 Website

Sardizar, Edward, dan Johan (2017, h.8) mendefinisikan website sebagai kumpulan dari beberapa halaman web dalam sebuah domain yang dapat digunakan sebagai suatu alat untuk menyebarkan informasi atau dapat digunakan sebagai salah satu alat untuk berbisnis. Berdasarkan pengertian diatas, Arief (2011, h.7) mengklaim bahwa web adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, animasi, video) didalamnya yang menggunakan protokol HTTP (Hypertext Transfer Protocol) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut browser. Menurut para ahli lain. Gregorius (2000, h.30) memberikan pendapat yakni *Website* adalah kumpulan halaman web yang saling terhubung dan file-filenya saling terkait. Web terdiri dari page atau halaman, dan kumpulan halaman yang dinamakan *homepage*. Homepage berada pada posisi teratas, dengan halaman-halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut *child page*, yang berisi hyperlink ke halaman lain dalam web dapat disimpulkan website merupakan sekumpulan halaman-halaman terkait berisi sekumpulan informasi (dapat berupa

informasi teks, gambar, animasi, suara, dan lainnya) yang disediakan oleh perseorangan, kelompok, atau organisasi untuk disebarluaskan melalui internet.

2.10.2 HTML

Menurut Edy Winarmo ST, M Eng, Ali Zaki, Dan SmithDev Community (2014, h.1) Mendiskusikan bahwa HTML adalah singkatan dari *hypertext markup language*. Sebuah bahasa untuk menampilkan konten di web. HTML sendiri bahasa pemrograman yang bebas, Artinya tidak dimiliki oleh siapa pun, pengembangannya dilakukan oleh banyak orang di banyak negara dan bisa dikatakan sebagai sebuah bahasa yang dikembangkan bersama-sama secara global.

Menurut Priyanto Hidayatullah dan Jauhari Khairul Kawistara (2017, h.15) bahwa HTML adalah bahasa standart yang digunakan untuk menampilkan halaman *web*, yang bisa dilakukan dengan HTML yaitu,

- a) Mengatur tampilan dari halaman *web* dan isinya.
- b) Membuat tabel dalam halaman *web*.
- c) Mempublikasikan halaman *web* secara *online*.
- d) Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi via *web*.
- e) Menampilkan area gambar (*canvas*) di *browser*.

Jadi HTML adalah bahasa pemrograman untuk menampilkan tampilan web yang dilakukan oleh banyak orang karena bahasa yang cukup mudah dan dapat dikembangkan secara global

2.10.3 CSS

Menurut Jubilee Enterprise (2016, h.93) Mengeklaim CSS (*Cascading Style Sheet*) adalah kumpulan kode untuk mendefinisikan desain dan bahasa markup. Karena ada kata markup pada definisi CSS, maka relasi antara CSS dan HTML sangatlah dekat. Dengan CSS lah, sebuah website yang dibangun menggunakan HTML akan menjadi lebih menarik dan variatif. Jika didefinisikan secara bebas, CSS merupakan kumpulan kode untuk mendesain atau mempercantik tampilan halaman *web*. Dengan artian lain, dengan memanfaatkan CSS kita bisa mengubah desain standar yang dihasilkan oleh HTML menjadi Variasi-variasi yang lebih kompleks.

2.10.4 PHP

Menurut Edy Winarmo ST, M Eng, Ali Zaki, Dan SmithDev Community (2014, h.49) Mendiskusikan topik ini bahwa Php adalah bahasa script yang cocok untuk pengembangan dalam web dan dapat dimasukkan kedalam HTML. PHP awalnya dikembangkan oleh seorang programmer bernama Ramus Ledrof pada tahun 1995, namun semenjak itu selalu dikembangkan oleh kelompok independen yang disebut Group PHP dan mendefinisikan standar de facto untuk PHP karena tidak ada spesifikasi formal. Saat itu sedang dipimpin oleh Andi gutmans dan Zeev suraski. Php saat ini sering digunakan karena perangkat lunak bebas (open source) yang dirilis bawah lisensi PHP, dapat dikatakan bahasa php bebas dan terbuka.

Contoh bahasa PHP. `<? Php`

```
    echo "Hello World";
```

```
    ?>
```

Perintah *echo* di dalam PHP berguna untuk mencetak nilai, baik teks maupun numerik ke layar *web browser*.

Jadi Php adalah bahasa pemrograman web yang berbasis web sering digunakan karena perangkat lunak bebas (open source) sehingga menghasilkan tampilan website yang dinamis di sisi client (*Browser*).

2.10.5 JavaScript

Menurut Edy Winarmo ST, M Eng, Ali Zaki, Dan SmithDev Community (2014, h.129) Mengeklaim bahwa *Javascript* adalah bahasa *scripting client side yang populer* di internet dan dapat bekerja di sebagian besar browser populer seperti *Internet Explorer, Mozilla Firefox, Netscape* dan *Opera*. Kode *javascript* dapat disisipkan dalam halaman *web* menggunakan tag *script*. Java script juga menjadi dasar yang bisa digunakan untuk teknologi lainnya seperti AJAX, JQUERY dan JQUERY MOBILE. Javascript sangat paling dipakai oleh programmer karena yang dapat dipakai secara global

Berikut ini beberapa sifat dari *javascript*:

1. Menambahkan interaktivitas ke halaman HTML.
2. Merupakan bahasa pemrograman *scripting*.
3. Bahasa *Scripting* merupakan bahasa yang ringan.
4. *Javascript* merupakan bahasa terinterpretasi.

2.10.6 Bootstrap

Menurut Zaenal A.Rozi dan SmitDev Community (2015, h.1) *Bootstrap* adalah paket aplikasi siap pakai untuk membuat *Front-end* sebuah *website*. Bisa dikatakan, *Bootstrap* adalah *template* desain *web* dengan fitur plus. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna mulai dari *level* pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai *HTML* dan *CSS* anda pun siap menggunakan *Bootstrap*. Berisi kumpulan file *CSS*, *Font*, dan *Javascript*. Yang siap diintegrasikan ke sebuah dokumen *HTML* yang dihasilkan pun secara dinamis akan tampil dalam *layout* yang disesuaikan dengan ukuran layar piranti pengunjung.

2.10.7 Database

Menurut Kendall dan Kendall dalam Septimas dkk pada Jurnal Unda (2017:2) ^[23], "Database atau sering juga disebut basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan merupakan sumber informasi yang dapat di periksa menggunakan suatu program komputer".

Menurut Fathansyah dalam Khuzainah pada *Journal Speed* (2013:2) ^[25], "Pengertian Database adalah kumpulan data yang disimpan secara sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan".

Menurut Santoso dalam Seminar Nasional Inovasi dan Aplikasi Teknologi Di Industri (2016:B.332) ^[26], "Database adalah susunan record

data operasional lengkap dari suatu organisasi atau perusahaan, yang diorganisir dan disimpan secara terintegrasi dengan menggunakan metode tertentu sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna".

Dari ketiga definisi diatas database merupakan sekumpulan informasi ataupun data yang diorganisir dan disimpan secara terintegrasi pada komputer.

2.10.8 Web server

Menurut Priyanto Hidayatullah,,(2014, h.123) *web server* adalah Tempat di mana anda menyimpan aplikasi *web* anda kemudian mengaksesnya melalui internet. Setiap perubahan, kecil maupun besar, dapat di *Upload* ke *web server* baru setelah itu anda periksa apakah perubahan itu sudah sesuai dengan yang anda inginkan atau belum. Web server ini dapat di akses menggunakan aplikasi *XAMPP*.

2.11 Pemodelan Objek

2.11.1 UML (Unified Modelling Language)







Menurut Widodo (2011:6), “ UML singkatan dari *Unified Modeling Languages* yang berarti bahasa pemodelan standar yang memiliki sintak dan semantik.” Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti standar yang ada. UML bukan hanya sekedar diagram tetapi juga menceritakan konteksnya. UML dapat diaplikasikan untuk maksud tertentu , antara lain untuk:

1. Merancang perangkat lunak
2. Sarana komunikasi antara perangkat lunak dalam proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Berdasarkan pendapat yang dikemukakan di atas maka dapat disimpulkan bahwa UML adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis objek (*Object Oriented Programming*).

2.11.2 Use Case Diagram







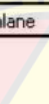

Menurut Widodo (2011) bahwa *Use case* atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem.

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

Gambar 2.6 Simbol pada *Use case Diagram*

2.11.3 Activity Diagram



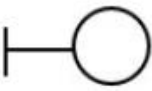



Menurut Rosa A.S dan M Shalahuddin, (2015, h.161) Diagram aktivitas atau Activity Diagram menggambarkan Workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Simbol	Deskripsi
status awal 	status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
aktivitas 	aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
percabangan / decision 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
penggabungan / join 	asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
status akhir 	status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
swimlane 	memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
fork, 	digunakan utk menunjukkan kegiatan yg dilakukan secara paralel
join, 	digunakan utk menunjukkan kegiatan yg digabungkan

Gambar 2.7 Simbol pada *Activity Diagram*

2.11.4 *Sequence Diagram*

Menurut Widodo (2011) Diagram sequence menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *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 *skanario* yang ada *use case*.

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari form
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan Pengiriman Pesan

Gambar 2.8 Simbol pada *SequenceDiagram*