

BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

Dalam penyusunan ini dicantumkan beberapa teori umum yang dapat dijadikan sebagai landasan teori, berikut penjelasan dari beberapa teori, diantaranya adalah sebagai berikut:

2.1.1. Pengertian Sistem

Menurut (Yulianti, 2012: 91) sistem adalah suatu gugus dari elemen-elemen yang saling berhubungan dan terorganisasi untuk mencapai suatu tujuan atau suatu gugus dari tujuan itu sendiri. Pendapat lain mengenai sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran tertentu (Hutahaean, 2016: 2).

Ada juga pendapat menurut (Husda, 2016: 91) kata “sistem” banyak sekali digunakan dalam percakapan sehari-hari, dalam forum diskusi maupun dokumen ilmiah. Kata ini digunakan untuk banyak hal, dan pada banyak bidang pula, sehingga maknanya menjadi beragam. Dalam pengertian paling umum, sebuah sistem adalah sekumpulan benda yang memiliki hubungan diantara mereka.

Dari pengertian ketiga pakar diatas mengenai sistem, maka penulis dapat menyimpulkan bahwa sistem adalah gabungan berbagai elemen sistem yang saling berkaitan dan memiliki tujuan untuk mencapai suatu sasaran yang sesuai dengan kebutuhan dari sebuah sistem tersebut.

2.1.2. Pengertian Informasi

Menurut (Hutahaean, 2016: 10) informasi merupakan data yang telah diolah menjadi suatu bentuk yang penting bagi sipenerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan yang akan datang. Sedangkan menurut (Yulianti, 2012: 91) pengertian informasi sebagai berikut:

1. Informasi adalah data yang sudah diproses dalam artian sudah diorganisasikan dan dihubungkan dengan persoalan yang spesifik
2. Informasi adalah hasil pembentukan, pengorganisasian atau pengubahan data dimana dengan cara itu dapat meningkatkan pengetahuan penerima.
3. Informasi adalah data yang telah diproses untuk menemukan hubungan diantara data tersebut.

Adapun pendapat menurut (Rahmad & Setiady, 2014: 1332) informasi adalah hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang relevan yang dibutuhkan oleh orang untuk menambah pemahamannya terhadap fakta yang ada.

Berdasarkan referensi dari pakar diatas mengenai informasi, maka penulis menyimpulkan bahwa informasi merupakan hasil pengolahan data yang telah di proses sehingga menghasilkan *output* data dimana data tersebut dapat meningkatkan pengetahuan penerimanya.

Secara umum informasi dapat didefenisikan sebagai hasil dari pengolahan data dalam satu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk mengambil keputusan. Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah suatu yang terjadi pada saat tertentu. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, dan gambar.

2.1.3. Pengertian Sistem Informasi

Menurut (Agusvianto, 2017: 41) sistem informasi adalah susatu sistem yang di dalam terdapat suatu organisasi yaitu mempertemukan kebutuhannya pada pengolahan transaksi harian yang mendukung fungsi-fungsi operasi organisasi yang bersifat manajerial dengan kegiatan srategi dari suatu organisasi dan untuk dapat menyediakan pada pihak luar tertentu dengan laporan-laporan yang di perlukan.

Pendapat lain mengenai sistem informasi adalah cara yang terorganisir untuk mengumpulkan, memasukan, dan memproses data dan menyimpannya, mengelola, mengontrol dan melaporkannya sehingga dapat

mendukung perusahaan atau organisasi untuk mencapai tujuan (Tantra, 2012: 19). Adapun pendapat lain dari sistem informasi ialah suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambil keputusan (Husda, 2016: 96).

Sehingga penulis menyimpulkan bahwasan sistem informas adalah sebuah metode untuk menyajikan informasi dengan cara mengumpulkan data lalu memprosesnya sehingga menghasilkan informasi yang di butuhkan.

2.1.4. SDLC (*Software Development Life Cycle*)

SDLC atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya(berdasarkan *best practice* atau cara-cara yang sudah teruji baik). Seperti halnya proses metamorfosis pada kupu-kupu, untuk menjadi kupu-kupu yang indah maka dibutuhkan beberapa tahap untuk dilalui (Shalahuddin, 2013: 26).

Tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*Initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*system concept development*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis, manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

3. Perencanaan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

4. Analisis kebutuhan (*requirements analysis*)

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.

5. Desain (*design*)

Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur kasus pengujian; mempersiapkan berkas atau *file* pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program; peninjauan pengujian.

7. Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user*. Menghasilkan laporan analisis pengujian.

8. Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operation and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

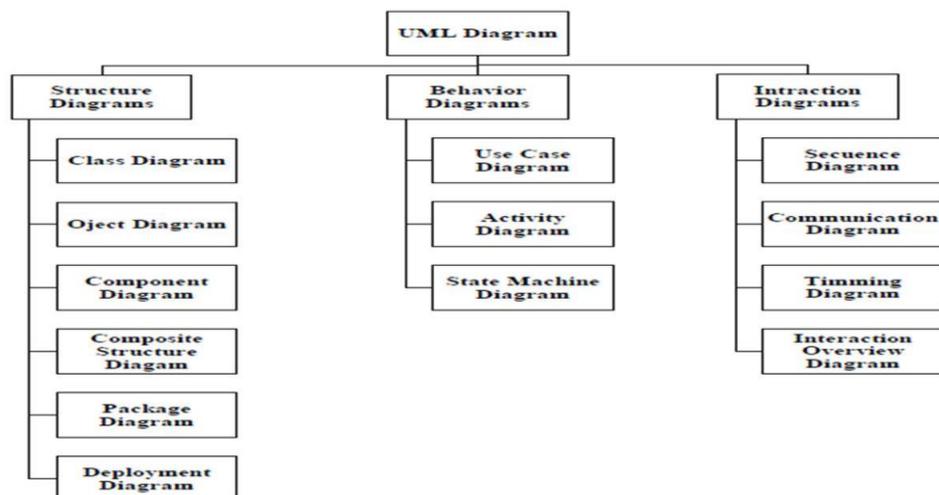
10. Disposisi (*disposition*)

Mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

2.1.5. Diagram UML (*Unified Modeling Language*)

Pada perkembangan teknologi perangkat lunak, diperlukannya adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Seperti yang kita ketahui bahwa menyatukan banyak kepala untuk menceritakan sebuah ide dengan tujuan untuk memahami hal yang sama tidaklah mudah, oleh karena itu diperlukan sebuah bahasa pemodelan perangkat lunak yang dapat dimengerti oleh banyak orang.

UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataan UML paling banyak digunakan pada metodologi berorientasi objek (Shalahuddin, 2013: 140).



Gambar 2.1 Diagram UML (Shalahuddin, 2013: 140)

Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. Structure diagrams yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. Behavior diagrams yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. Interaction diagrams yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.1.5.1. Class Diagram

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Shalahuddin, 2013: 141). Kelas memiliki apa yang disebut atribut dan metode atau operasi.

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

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2. Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3. Kelas yang diambil dari pendefinisian *use case* (*controller*)

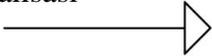
Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut simbol-simbol diagram kelas:

Tabel 2.1 simbol yang ada pada diagram kelas (Shalahuddin, 2013: 146).

Simbol	Deskripsi
Kelas nama_kelas +atribut +operasi	Kelas pada struktur sistem
Antarmuka / <i>interface</i>  Nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan

Tabel 2.1 lanjutan

	multiplicity
Kebergantungan / <i>dependency</i> 	Kebergantungan antarkelas
Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

2.1.5.2. Object Diagram

Menurut (Shalahuddin, 2013: 147) diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Pada diagram objek harus dipastikan semua kelas yang sudah didefinisikan pada diagram kelas harus dipakai objeknya, karena jika tidak, pendefinisian kelas itu tidak dapat dipertanggungjawabkan. Diagram objek juga berfungsi untuk mendefinisikan contoh nilai atau isi dari atribut tiap kelas.

Berikut adalah simbol-simbol yang ada pada diagram objek:

Tabel 2.2 simbol-simbol diagram objek (Shalahuddin, 2013: 147).

Simbol	Deskripsi
Objek Nama_objek : nama_kelas Atribut : nilai	objek dari kelas yang berjalan saat sistem dijalankan
Link 	relasi antar objek

2.1.5.3. Use Case Diagram

Use case atau diagram use case merupakan permodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case*

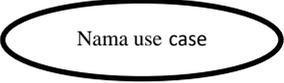
mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Shalahuddin, 2013: 155).

Syarat penamaan pada *use case* adalah nam didefenisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisikan apa yang disebut aktor dan *use case*.

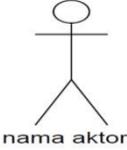
1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram use case:

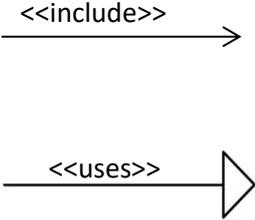
Tabel 2.3 simbol diagram use case (Shalahuddin, 2013: 156)

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case</p>

Tabel 2.3 lanjutan

<p>Aktor / actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar 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</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki di awal frase nama aktor</p>
<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu; mirip dengan prinsip inheritance pada pemograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya use yang menjadi extend-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum</p>

Tabel 2.3 lanjutan

<p>Menggunakan <i>/include/uses</i></p> <p style="text-align: center;">  </p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsi atau sebagai syarat dijalankan <i>use case</i> ini. Ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i>:</p> <ol style="list-style-type: none"> 1. Include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan, 2. Include berarti <i>use case</i> yang ditambahkan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan.
--	---

2.1.5.4. Activity Diagrams

Menurut (Shalahuddin, 2013: 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 sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

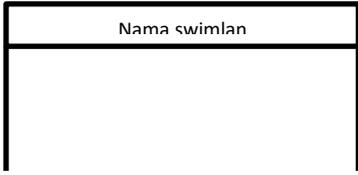
Diagram aktifitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis yang didefinisikan.

2. Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak

Berikut adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.4 simbol-simbol diagram aktivitas

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 / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas gabungan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.2. Teori Khusus

Dalam penyusunan ini dicantumkan beberapa teori khusus yang menjelaskan proses pembuatan persediaan barang berbasis web pada Bank Mandiri Batu Aji, berikut penjelasan dari beberapa teori, diantaranya sebagai berikut :

2.2.1. Persediaan (Inventory)

Menurut (Rohayati, 2014: 2) inventory meliputi semua barang yang dimiliki perusahaan, dengan tujuan dikomsumsikan dalam siklus operasi normal perusahaan sebagai barang yang dimiliki, semua barang yang berwujud dapat disebut sebagai *inventory*, tergantung dari sifat dan jenis usaha perusahaan. Pada Bank Mandiri Cabang Batu aji ini inventory yang di maksud adalah perlengkapan yang akan digunakan dalam pelayanan. Adapun pendapat lain bahwa pengendalian dan pemeliharaan persediaan sangat diperlukan oleh setiap bisnis, karena pemeliharaan persediaan dapat berdampak dalam keberlanjutan dan pertumbuhan suatu organisasi, mengabaikan pemeliharaan persediaan dapat berpengaruh buruk pada suatu organisasi (Madamidola & daramola, 2017:164).

2.2.2. Fungsi Pengendalian Persediaan (Inventory)

Fungsi utama pengendalian persediaan adalah “menyimpan” untuk melayani kebutuhan perusahaan akan bahan mentah atau barang jadi dari waktu ke waktu (Anisya & Wandyra, 2016: 13).

2.2.3. PHP (Hypertext Preprocessor)

Menurut (Saputra, 2012: 91) untuk menggunakan PHP, kita harus memulainya dengan tanda khusus seperti

```
<?php
//sintaks PHP
?>
```

Ada 4 macam format yang bisa digunakan untuk memulai pemograman PHP didalam kode yaitu:

1. <?php ?>
2. <? ?>
3. <script language="php"> </script>
4. <% %>

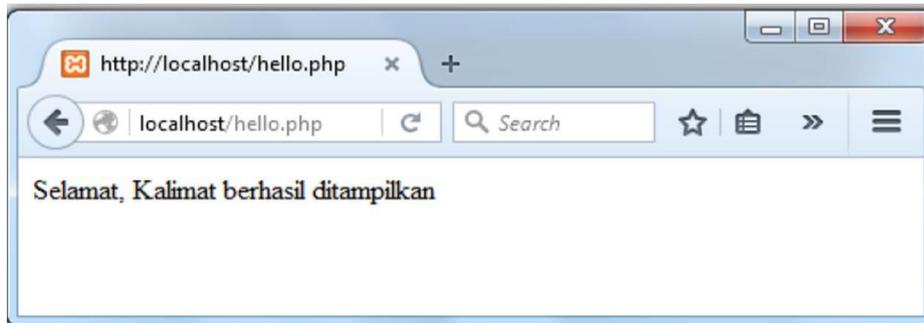
Dari ke-4 macam format tersebut, sintaks <?php ... ?> dan <? ... ?> merupakan format yang paling banyak digunakan oleh programmer. Selain itu ada 2 macam perintah untuk menampilkan suatu kalimat atau string, yaitu menggunakan perintah echo dan print, terserah anda ingin menggunakan yang mana. Berikut contoh skripnya:

```
<?php
Echo "selamat, kalimat berhasil ditampilkan"; ?>

Atau

<?php
Print "selamat, kalimat berhasil ditampilkan"; ?>
```

Simpan dengan nama `hello.php` ke dalam direktori `htdocs` (`C:/xampp/htdocs`). Setelahnya, buka melalui web browser dengan url <http://localhost/hello.php>, maka akan terlihat hasil seperti gambar berikut:



Gambar 2.2. Tampilan Hello.php (Saputra, 2012: 92)

2.2.4. HTML (Hyper Text Markup Language)

HTML merupakan singkatan dari *Hyper Text Markup Language*. HTML bisa disebut bahasa paling dasar dan yang paling penting digunakan untuk menampilkan dan mengelola tampilan pada halaman website. HTML digunakan untuk menampilkan berbagai informasi didalam sebuah penjelajah *web* internet dan *formatting hypertext* sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi (Saputra, 2012: 1).

Dengan kata lain, berrkas yang dibuat dalam perangkat lunak pengolah kata, disimpan kedalam format *ASCII* normal sehingga menjadi homepage denga perintah-perintah *HTML*. *HTML* menggunakan 2 macam ekstensi file yaitu **.htm** dan **.html**. Format ekstensi berformat **.htm** awalnya hanyalah untuk mengakomodasi penggunaan *html* dalam operasi *DOS*.

Menurut (Saputra, 2012: 4) dokumen *html* memiliki sebuah struktur yang harus kita ikuti aturan pembuatannya. Pada sub bab ini, kita akan mengenal beberapa elemen-elemen wajib yang ada pada *file html* apabila kita ingin membangun suatu pondasi kerangka *website*. Elemen tersebut diantaranya:

1. Elemen *HTML*

Tag ini merupakan perintah wajib bagi pemrograman *web* untuk menulis tag pertama dalam dokumen *html*.

Contoh tagnya adalah `<html>` dan diakhiri dengan `</html>`

2. Elemen *Head*

Head merupakan tag berikutnya setelah elemen *html* (`<html>`), yang berfungsi untuk menuliskan keterangan tentang dokumen *web* yang akan ditampilkan. Elemen ini nantinya akan diakhiri dengan tanda penutu `</head>`. Jadi jika secara runtun dapat dituliskan format berikut:

```
<html>
```

```
<head>
```

```
</head>
```

```
<html>
```

3. Elemen *Title*

Element *title* merupakan suatu elemen yang harus dituliskan dalam elemen *head* yang digunakan untuk memberi judul/ informasi dokumen *web* yang ditampilkan pada *browser*. Berikut struktur penggunaanya:

```
<html>
```

```

<head>

<title> Tuliskan judul disini </title>

</head>

</html>

```

4. Elemen *body*

Elemen *body* merupakan bagian utama dalam dokumen *web*. Jika dengan sebutan konten, maka kita harus meletakkan *teks* tersebut pada elemen *body*. Struktur elemennya sebagai berikut:

```

<html>

<head>

<title> Tuliskan judul disini </title>

</head>

<body> tuliskan konten disini

</body>

</html>

```

2.2.5. CSS (Casading Style Sheet)

Menurut (Saputra, 2012: 27) merupakan singkatan dari *Cascading Style Sheet* merupakan bahasa pemograman web yang didesain khusus untuk mengendalikan dan membangun komponen dalam web sehingga tampilan web lebih rapih, terstruktur, dan seragam. CSS merupakan salah satu pemrograman wajib disamping html yang harus dikuasai oleh para setiap pemrograman web, terlebih lagi itu adalah *Web Designer*.

Tujuan utama dari CSS adalah :

1. Memisahkan konten utama dengan tampilan dokumen lainnya (html dan sejenisnya).
2. *Web* yang menggunakan CSS akan lebih ringan dan mudah dibuka dibandingkan dengan web yang tidak menggunakan CSS
3. Untuk mempercepat pembuatan halaman *web*, hanya perlu membuat satu properti dan properti tersebut dapat digunakan pada elemen lainnya, artinya, anda tidak perlu menulis ulang kode program yang digunakan berulang kali.

Cara menggunakan CSS adalah sebagai berikut:

1. *Embedded Style Sheet*

Merupakan cara penulisan kode dimana penulis CSS dilakukan pada tag html, yaitu pada tag `<style> ... </style>` dan sebelum tag `<body>`. Pada tag tersebut disisipkan kode CSS yang akan digunakan oleh tag html.

2. *Inline Style Sheet*

Merupakan salah satu cara penggunaan CSS langsung pada tag html yang dibutuhkan saja. Cara ini dilakukan umumnya dikarenakan hanya sedikit properti yang dibutuhkan, misalnya kita hanya perlu merubah warna pada teks tertentu yang dimana properti warna tersebut tidak digunakan pada elemen lain (sekali pakai). Format dasarnya sebagai berikut:

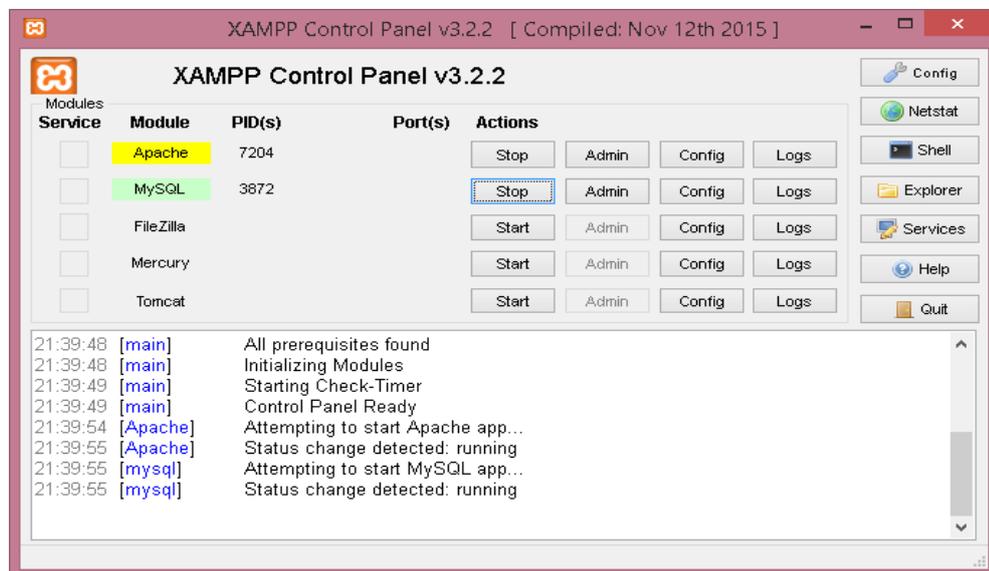
```
<tag HTML style="properti: value/nilai"> ... </tag HTML>
```

3. *Linked Style Sheet*

Metode ini merupakan cara pengerjaan dimana antara kode css dan html telah dipisahkan. Untuk menggunakan kode css yang telah dipisahkan ini, maka dalam kode html dibuat skrip yang isinya adalah memanggil file css tersebut untuk dalam kode html.

2.2.6. XAMPP (Xapache MySQL PHP Perl)

XAMPP (Xapache MySQL PHP Perl) merupakan *software* yang berfungsi menjalankan *website* berbasis PHP dan menggunakan pengolahan data MySQL dikomputer lokal. XAMPP berperan sebagai *server web* komputer (Khasani & Surjawan, 2016: 320).



Gambar 2.3. XAMPP Control Panel (Khasani & Surjawan, 2016: 320)

2.2.7. MySQL

Menurut (Saputra, 2012: 77) MySQL merupakan satu *database* kelas dunia yang sangat cocok bila dipadukan dengan bahasa pemrograman PHP. MySQL bekerja menggunakan bahasa SQL (*Structure Query Language*) yang merupakan bahasa standar yang digunakan untuk memanipulasi database. Pada umumnya perintah yang paling sering digunakan dalam MySQL adalah *SELECT* (mengambil), *INSERT* (menambah), *UPDATE* (mengubah) dan *DELETE* (menghapus).

2.2.8. JavaScript

JavaScript merupakan salah satu bahasa *script website* yang paling banyak digunakan untuk menambahkan manipulasi script *HTML* dan *CSS* pada sisi *client browser*. *JavaScript* mampu memberikan fungsionalitas dengan server, serta membuat website lebih interaktif dan animatif (Khasani & Surjawan, 2016: 320).

2.2.9. Microsoft Visio 2010

Microsoft Visio 2010 adalah sebuah perangkat lunak atau aplikasi yang digunakan untuk mendesain, menggambarkan, dan membuat sebuah diagram dari sebuah data untuk menggambarkan sistem dan informasi (Fauzi, 2011: 03)

Visio dapat menghasilkan suatu diagram mulai dari yang sederhana hingga diagram yang lebih kompleks. Anda hanya perlu menambahkan

shape dengan menarik *shape* kehalaman kerja. Oleh karena *Visio* merupakan salah satu produk dari *Microsoft Office* lainnya, seperti *Microsoft Word*, *Excel*, *PowerPoint*, dan *Project*. Bentuk kolaborasi dapat dilakukan dengan fasilitas ekspor dan impor data.

2.3. Penelitian Terdahulu

Demi menunjang keberhasilan penelitian ini, maka penulis tidak lupa mencari sumber-sumber pendukung berupa jurnal penelitian terdahulu yang bersumber dari jurnal yang sudah terdaftar oleh lembaga yang terpercaya dibidangnya. Dalam hal ini penulis telah merangkum hasil dari setiap penelitian terdahulu didalam sebuah tabel dibawah ini:

Tabel 2.5 Kumpulan Jurnal Penelitian Terdahulu

No	Penelitian terdahulu	Judul jurnal penelitian	Hasil dan Kesimpulan Jurnal
1	Anisya ¹ , Yunita Wandrya ² , Vol 4 No. 2 Oktober 2016. ISSN 2338-2724.	REKAYASA PERANGKAT LUNAK PENGENDALIAN INVENTORI MENGGUNAKAN METODE SMA (SINGLE MOVING AVERAGE) BERBASIS AJAX (ASYNCHRONOUS JAVASCRIPT AND XML) (STUDI KASUS: PTP NUSANTARA VI (Persero) UNIT USAHA KAYU ARO)	Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa aplikasi pengendalian persediaan barang berbasis AJAX mampu memberikan informasi persediaan barang yang tersedia dan aplikasi memprediksi persediaan barang untuk bulan berikutnya menggunakan metode single moving average.
2	Liza Yulianti ¹ , Yupianti ² , Vol 8 No.1 Februari 2012, ISSN 1858-2680	SISTEM INFORMASI PERSEDIAAN BARANG PADA PT. SURYA NUSA BHAKTINDO BENGKULU	Pada PT. Surya Nusa Bhaktindo Bengkulu banyak terdapat permasalahan pada sistem persediaan barang Material yaitu masih menggunakan program aplikasi excel, sehingga waktu yang dibutuhkan untuk melakukan Permintaan Material hingga Penginputan material masuk membutuhkan waktu yang cukup lama

No	Penelitian Terdahulu	Judul Jurnal Penelitian	Hasil dan Kesimpulan Jurnal
3	Hendra Agusvianto, Vol 1 Nomor 012017, ISSN 2549-869X	Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus : PT.Alaisys Sidoarjo	Berdasarkan uraian yang telah di jelaskan pada bab bab sebelumnya maka dapat disimpulkan bahwa : 1. Sistem ini merupakan suatu aplikasi system inventori pada gudang yang berfungsi untuk pencatatan keluar masuk barang oleh staf PT.Alaisys dapat dilakukan dengan struktur sehingga dapat memberikan bantuan dalam hal efisiensi waktu kegiatan pencatatan dan penyusunan data pada barang. 2. Dengan adanya aplikasi berbasis web ini dapat memberikan laporan pada pengolahan data barang dari gudang ke kantor pusat secara tepat, akurat dan tepat sasaran. 3. Sistem informasi manajemen yang diusulkan untuk perusahaan yaitu dengan melakukan pemisahan bagian pekerjaan yang pada sistem awal semua dilakukan secara sentralistik.
4	Gilang Permadi Khasani ¹ , Daniel Jahja Surjawan ² . Vol 2 No. 3 Desember 2016, ISSN 2443-2229	Aplikasi Inventory Berbasis Web Pada PT Telkomsel NS Tasikmalaya	Berdasarkan hasil analisis dan perancangan, dapat disimpulkan bahwa : 1. Aplikasi ini membantu proses pengelolaan data Serial Number modul yang telah di RMA 2. Aplikasi ini membantu pengecekan data Serial Number modul jika Serial Number modul tersebut sedang ada di site 3. Aplikasi ini memudahkan dalam membuat laporan modul per periode 4. Aplikasi ini membuat setiap modul memiliki kode barcode untuk percepatan pencarian.
5	Mita Rohayati, Vol 1 No 1 Februari 2014, ISSN 2089-9033	MEMBANGUN SISTEM INFORMASI MONITORING DATA INVENTORY DI VIO HOTEL INDONESIA	Kesimpulan terhadap pembangunan Sistem Informasi Monitoring Data Inventory di Vio Hotel Indonesia sebagai berikut: 1. Sistem monitoring data inventory ini dapat berjalan sesuai dengan tujuannya, yaitu membantu memudahkan proses monitoring yang tepat dan cepat. 2. Dengan adanya sistem monitoring data inventory ini, Koordinator Engineering lebih mudah mengolah data inventory. 3. Sistem monitoring data inventory yang telah dibangun mempermudah Koordinator Engineering untuk melakukan peminjaman barang antar cabang Vio sehingga dapat

No	Penelitian Terdahulu	Judul Jurnal Penelitian	Hasil dan Kesimpulan Jurnal
6	Madamidola O.A, Daramola O.A, Akintola K.G, Vol1 No. 4 ISSN 2456- 6470	Web – Based Intelligent Inventory Management System	The control and maintenance of inventory is a problem experienced virtually by all business sectors. Inventories management is crucial for the sustainability and growth of any organization. Disregarding the necessity of inventory in any organization can lead to its shutting down, especially if the variables of productivity are poorly managed in progression to meet customers' need or desires.
7	Priya P, Dr.K.Iyakutti, Vol No. 25 International Journal 0975-8887	Web based Multi Product Inventory Optimization Using Genetic Algorithm	Inventory encompasses all raw materials, work in process, and finished goods within the supply chain. Changing Inventory policies can dramatically alter the supply chain's efficiency and responsiveness. Inventory is an important cross functional driver of supply chain performance. An important role that can be satisfied by having the product ready and available when the customer wants it. Inventory is held throughout the supply chain in the form of raw materials, work in progress, and finished goods.
8	Mhd Bustanur Rahmad(07018151), Tedy Setiady(0407016801), Volume 2 Nomor 2, Juni 2014. e-ISSN: 2338-5197	PERANCANGAN SISTEM INFORMASI INVENTORY SPARE PART ELEKTRONIK BERBASIS WEB PHP (Studi CV.Human Global Service Yogyakarta)	Informasi adalah hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang relevan yang dibutuhkan oleh orang untuk menambah pemahamannya terhadap fakta yang ada.