BAB II
LANDASAN TEORI

2.1 Teori – Teori Dasar

2.1.1 Kepegawaian

Menurut penjelasan umum dalam Undang-Undang (UU) Republik Indonesia Nomor 5 Tahun 2014 yang dimaksud dengan Kepegawaian adalah segala hal mengenai kedudukan, kewajiban, hak dan pembinaan pegawai negeri. Secara sederhana Pegawai merupakan tenaga kerja manusia, jasmaniah maupun rohaniah (mental dan pikiran), yang senantiasa dibutuhkan dan menjadi modal pokok setiap badan usaha untuk mencapai tujuan tertentu.

2.1.2 Fungsi Bagian Kepegawaian

Menurut (Hasibuan, 2013) menyebutkan bahwa fungsi manajemen sumber daya manusia terdiri dari :

1. Fungsi Manajemen

a. Perencanaan *(planning)*

Perencanaan mempunyai arti penentuan mengenai program tenaga kerja yang akan mendukung pencapaian tujuan yang telah ditetapkan oleh perusahaan.

b. Pengorganisasian *(organizing)*

Organisasi dibentuk dengan merancang struktur hubungan yang mengaitkan antara pekerjaan, karyawan, dan faktor-faktor fisik sehingga dapat terjalin kerja sama satu dengan yang lainnya.

c. Pengarahan *(directing)*

Pengarahan terdiri dari fungsi *staffing* dan *leading*. Fungsi *staffing* adalah menempatkan orang-orang dalam struktur organisasi, sedangkan fungsi *leading* dilakukan pengarahan SDM agar karyawan bekerja sesuai dengan tujuan yang ditetapkan.

d. Pengawasan *(controlling)*

Adanya fungsi manajerial yang mengatur aktifitas-aktifitas agar sesuai dengan rencana yang telah ditetapkan organisasi sesuai dengan tujuan yang ingin

dicapai, bila terjadi penyimpangan dapat diketahui dan segera dilakukan perbaikan.

2. Fungsi Operasional

a. Pengadaan *(procurement)*

Usaha untuk memperoleh sejumlah tenaga kerja yang dibutuhkan perusahaan, terutama yang berhubungan dengan penentuan kebutuhan tenaga kerja, penarikan, seleksi, orientasi dan penempatan.

b. Pengembangan *(development)*

Usaha untuk meningkatkan kualitas karyawan melalui program pendidikan dan latihan yang tepat agar karyawan dapat melakukan tugasnya dengan baik. Aktifitas ini penting dan akan terus berkembang karena adanya perubahan teknologi, penyesuaian dan meningkatnya kesulitan tugas manajer.

c. Kompensasi *(compensation)*

Fungsi kompensasi diartikan sebagai usaha untuk memberikan balas jasa atau imbalan yang memadai kepada pegawai sesuai dengan kontribusi yang telah disumbangkan kepada perusahaan atau organisasi.

d. Integrasi *(integration)*

Merupakan usaha untuk menyelaraskan kepentingan individu, organisasi, perusahaan maupun masyarakat. Oleh sebab itu harus dipahami sikap prinsip-prinsip pegawai.

e. Pemeliharaan *(maintenance)*

Setelah keempat fungsi dijalankan dengan baik, perusahaan akan mendapat pegawai yang baik. Maka fungsi pemeliharaan adalah dengan memelihara sikap-sikap pegawai yang menguntungkan perusahaan.

f. Pemutusan Hubungan Kerja *(separation)*

Usaha terakhir dari fungsi operasional ini adalah tanggung jawab perusahaan untuk mengembalikan pegawainya ke lingkungan masyarakat dalam keadaan sebaik mungkin, bila organisasi atau perusahaan mengadakan pemutusan hubungan kerja.

2.1.3 Sistem

Menurut (I Putu Agus Eka Pratama 2016) Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama.

2.1.4 Informasi

Menurut (I Putu Agus Eka Pratama 2016) Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti, dan manfaat.

2.1.5 Sistem Informasi

Menurut (I Putu Agus Eka Pratama 2016) berdasarkan definisi mengenai sistem dan informasi yang telah dijelaskan di atas, maka dapat dinyatakan bahwa sistem informasi merupakan gabungan dari empat bagian utama. Keempat bagian utama tersebut mencakup perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur, dan Sumber Daya Manusia (SDM) yang terlatih. Keempat bagian utama ini saling berkaitan untuk menciptakan sebuah sistem yang dapat mengolah data menjadi informasi yang bermanfaat. Di dalamnya juga termasuk proses perencanaan, kontrol, koordinasi, dan pengambil keputusan. Sehingga, sebagai sebuah sistem yang mengolah data menjadi informasi yang akan disajikan dan digunakan oleh pengguna, maka sistem informasi merupakan sebuah sistem yang kompleks. Beberapa manfaat dari sistem informasi sebagai berikut:

- Data yang terpusat
- Kemudahan dalam mengakses informasi
- Efesiensi waktu

2.1.6 Karakteristik Sistem

Menurut (Sutabri, 2012) Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yakni :

a. Komponen Sistem

   Pengertian sistem adalah suatu sistem yang terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling berkerja sama membentuk satu

kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa satu subsistem atau bagian-bagian sistem.

b. Batasan Sistem

Definisi atau batasan sistem adalah daerah yang membatasi antara suatu sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan.

c. Lingkungan Luar Sistem

Lingkungan luar sistem dari suatu sistem adalah apapun diluar batasan dari sistem yang mempengaruhi operasional sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

d. Penghubung Sistem

Pengertian penghubung sistem (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem lainnya.

e. Masukan Sistem

Pengertian masukan (*input*) adalah energi yang dimasukkan kedalam sistem. Masukan dapat berupa perawatan(*maintenance input*) dan sinyal (*signal input*). Masukan sistem (*maintenance input)* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Dan yang dimaksud dengan masukan sinyal *(signal input)* adalah energi yang diproses untuk didapatkan keluaran.

f. Keluaran Sistem

Pengertian keluaran sistem (*output*) adalah suatu sistem yang merupakan hasil dari input yang telah diproses oleh bagian pengolah dan merupakan tujuan akhir sistem. *Output* ini berupa laporan grafik, diagram batang dan lain-lainnya.

g. Pengolah Sistem

Pengertian pengolah sistem adalah suatu sistem yang terdiri dari bagian pengolah yang akan merubah masukan *(input)* menjadi keluaran *(output).*

h. Sasaran Sistem

Sasaran sistem adalah suatu sistem yang jelas mempunyai tujuan (*goal*) atau sasaran (*objective*). Jika suatu sistem tidak mempunyai sasaran yang jelas, maka operasional sistem tidak ada gunanya.

2.1.7 Kualitas Informasi

Menurut (Sutabri, 2012) Kualitas dari suatu informasi yang bermutu harus mengutamakan tiga hal, yaitu :

1. Akurat *(accurate)*

   Informasi harus bebas dari kesalahan-kesalahan yang biasanya terjadi selain itu harus jelas maksud dan tujuannya, sehingga *output* bisa dipertanggungjawabkan.
2. Tepat Waktu *(timeliness)*

   Informasi yang datang pada penerima tidak boleh terlambat karena informasi yang terlambat tidak akan mempunyai nilai untuk mendukung pengambilan keputusan.
3. Relevan *(relevance)*

   Informasi tersebut mempunyai manfaat dan informasi yang diterima pemakai. Hal ini disebabkan relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

2.1.8 *E-Document*

E-Document atau Dokumen elektronik adalah informasi yang disimpan dengan memerlukan perangkat komputer atau perangkat elektronik lain untuk menampilkan, menafsirkan dan memprosesnya. Contoh bentuk dokumen berupa: *email, voice mail, instans messages (IM), e-calender, audio-file, video,* data-data hasil transfer *handphone*, animasi, grafik, foto, presentasi, *web pages*, dan data *digital* lainnya yang dihasilkan oleh perangkat lunak yang disimpan di komputer, jaringan *(network), back-up* ke dalam *Compact Disk (CD)* ataupun *DVD.*

Berbeda dengan dokumen kertas, dokumen elektronik dapat berisi informasi data *non-linear* seperti *hypertext* yang bisa terkoneksi melalui *hyperlinks.*

Dokumen eletronik dibuat dan diperoleh dengan cara sebagai berikut :

1. Membuat secara langsung dokumen elektronik melalui berbagai aplikasi komputer yang menghasilkan keluaran *(output)* dalam bentuk *digital*, misalnya aplikasi pengolahan huruf, angka dan gambar grafik.
2. Memindahkan dokumen melalui Pemindaian *(Scanning),* Konversi File *(conversion)* dan *Importing* data dari berbagai format.

Berikut ini adalah Pengelolaan *E-Document (Document management)* :

1. Memasukkan Dokumen *(Input)*, prosesnya yaitu:

a. Menentukan Kategori dokumen

b) Penentuan kata tangkap *(caption)*

c) Penentuan sifat dokumen

d) Penentuan masa simpan

e) Unggah File

2. Pencarian Dokumen

Merupakan fitur yang digunakan untuk melakuan pencarian dokumen elektronik sesuai keinginan pengguna, dengan melakukan pegetikan satu atau beberapa kata tangkap. Setelah dokumen yang dicari ditemukan, kemudian dokumen tersebut didistribusikan sesuai kebutuhan, yaitu: membaca, simpan dan cetak.

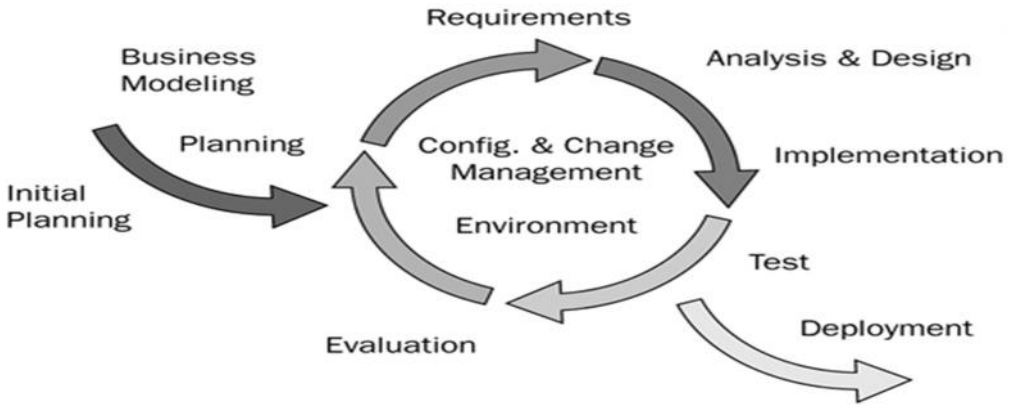
2.2 Teori Pengembangan Sistem

Menurut Rosa A.S M. Shalahudin (2016, p.26) proses pengembangan mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodelogi yang digunakan dalam mengembangkan sistem-sistem perangkat lunak sebelumnya.

2.2.1 RUP *(Rational Unified Process)*

Menurut Rosa A.S M. Shalahudin (2016, p.124) *Unified Process* atau dikenal juga dengan proses *iterative* dan *incremental* merupakan sebuah proses pengembangan perangkat lunak yang dilakukan secara iteraktif (berulang) dan *incremental* (bertahap dengan progress menarik). Iteraktif bisa dilakukan di dalam setiap tahap, atau iteraktif tahap pada proses pengembangan perangkat lunak untuk menghasilkan perbaikan fungsi yang *incremental* (bertambah menarik) di mana setiap iterasi akan memperbaiki iterasi berikutnya. Salah satu *Unified Process* yang terkenal adalah RUP *(Rational Unified Process).*

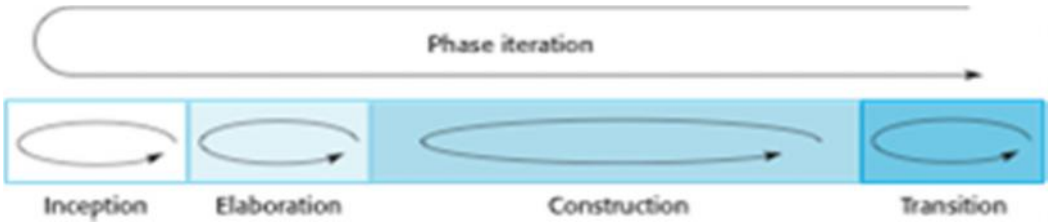
Proses pengulangan/iteratif pada RUP secara umum dapat dilihat pada gambar berikut:



Gambar 2.1 Proses Iteratif RUP

2.2.2 Fase RUP

RUP memiliki empat buah tahapan atau fase yang dapat dilakukan pula secara iterative. Berikut ini gambar alur hidup RUP.



Gambar 2.2 Alur Hidup RUP

Berikut ini penjelasan untuk setiap fase pada RUP:

1. *Inception* (permulaan)

   Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan *(business modeling)* dan mendefinisikan kebutuhan akan sistem yang akan dibuat *(requirements).*

2. *Elaboration* (perluasan/perencanaan)

   Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dari arsitektur yang dibuat.

Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem *(prototype).*

3. *Construction* (kontruksi)

   Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal.

4. *Transition* (transisi)

   Tahap ini lebih pada *deployment* atau instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal. Aktifitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan *user*.

2.3 Teori Alat Pengembangan Sistem

2.3.1 Definisi *Unified Modeling Language* (UML)

Menurut Rosa A.S M. Shalahudin (2016, p.137) UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. 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. Perkembangan penggunaan UML bergantung pada level abstraksi penggunaannya, dalam penggunaan UML adalah suatu yang salah, tapi perlu ditelaah dimanakah UML digunakan dan hal apa yang ingin divisualkan.

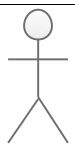
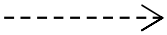
Menurut Rosa A.S M. Shalahudin (2016, p.140) pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori.

a. *Use Case Diagram*

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 detail, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesederhana mungkin dan dapat dipahami.

Tabel 2.1 Simbol *Use Case* Diagram

Simbol	Nama	Keterangan
	*Aktor*	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
	Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
	Straight Connector	Relasi antara aktor dan *use case*.
	Generalization	Untuk memperlihatkan struktur pewaris yang terjadi.
	*Include*	Relasi jika terdapat perilaku yang mirip dengan beberapa *use case.*
	Exclude	Relasi yang digunakan jika use case yang satu mirip dengan use case yang lain.

b. *Activity Diagram*

Diagram aktivitas atau *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem, proses bisnis, menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktifitas

menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

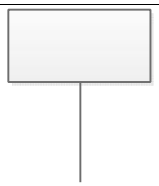
Tabel 2.2 Simbol *Activity* Diagram

Simbol	Nama	Keterangan
	Start state	Sebagai titik Awal
	End state	Sebagai titik Akhir
	State	Aktivitas yang ada
	Decision	Mengambil keputusan
	Transition	Penghubung aktivitas

c. *Sequence Diagram*

Sequence diagram adalah grafik dua dimensi dimana objek ditunjukkan dalam dimensi horizontal, sedangkan *lifeline* ditunjukkan dalam dimensi vertikal.

Table 2.3 Simbol *Sequence* Diagram

Simbol	Nama	Keterangan
	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
	*Object*	*Object entity* antarmuka saling berinteraksi.

	Activate	Mengindikasikan keberadaan objek dalam basis waktu.
	Stimulus	Menghubungkan objek.

d. *Class Diagram*

Class diagram merupakan himpunan dari objek-objek yang sejenis. Sebuah objek memiliki keadaan sesaat *(state)* dan perilaku *(behavior)*. State sebuah objek adalah kondisi objek tersebut yang dinyatakan dalam *attribute* atau *properties*. Sedangkan perilaku suatu objek mendefinisikan bagaimana sebuah objek bertindak dan memberikan reaksi.

2.3.2 Basis Data

Menurut (Bunafit Nugroho, 2013) Basis data (*database*) adalah kumpulan dari berbagai data yang saling berhubungan satu dengan yang lain. Basis data tersimpan di perangkat keras, serta dimanipulasi dengan menggunakan perangkat lunak. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau informasi yang akan disimpan. *Database* merupakan salah satu kemponen yang paling penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi pada para pengguna. Penyusunan basis data meliputi proses memasukkan data ke dalam media penyimpanan data, dan diatur dengan menggunakan perangkat sitem manajemen basis data *(Database Management System- DBMS).* Manipulasi basis data meliputi pembuatan pernyataan (*query*) untuk mendapatkan informasi tertentu, melakukan pembaharuan (*update)* serta pembuatan *report* dari data. Tujuan utama dari DBMS adalah bagaimana merancang struktur data bagi user. Jadi, sistem menyembunyikan informasi mengenai bagaimana data disimpan dan dirawat. Tetapi data tetap dapat diambil dengan efisien. Pertimbangan efisien yang digunakan adalah bagaimana merancang struktur data yang kompleks, tetapi tetap dapat digunakan oleh pengguna yang masih awam, tanpa mengetahui kompleksitas stuktur.

2.4 Teori Pembuatan Aplikasi

2.4.1 *Web* Server

Menurut (Bunafit Nugroho, 2013) *Web server* sebuah aplikasi tempat menyimpan file-file maupun data-data untuk membuat website berbasis *client-server* dan dapat diterapkan pada sebuah sistem lokal atau pada jaringan komputer. Yang digunakan untuk menangani permintaan-permintaan dari seorang pengguna akan sebuah halaman *web* melalui sebuah *web server*. *Web server* memberikan halaman *web* yang diminta oleh pengguna kedalam *web browser* yang digunakannya. Dalam melakukan tugasnya *web server* dijalankan oleh sebuah *daemon* dengan membuka sebuah *koneksi* TCP (*Transmission Control Protocol*) yang umumnya menggunakan *port* 80 (HTTP). Sebuah *web server* tidak hanya dapat memberikan *respon* berupa data yang terdapat pada *file* sistem lokal dari *web server* tersebut, namun *web server* dapat memberikan respon berupa *hypertext* yang berada pada *web server* lain. Contoh dari apikasi *web server* adalah *xampp*, *apache*, *xitami*, IIS, *zope*, dan PWS. Dalam Skripsi ini digunakan *web server Xampp* yang mampu mengembangkan aplikasi web dengan PHP (Hypertext Preprocessor).

2.4.1.1 XAMPP

Menurut Riyanto (2014:1) XAMPP merupakan paket PHP dan MySQL berbasis *open source,* yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP. Salah satu paket instalasi Apache, PHP dan MySQL *instant* yang dapat kita gunakan untuk membantu proses instalasi ketiga produk tersebut. Untuk berpindah versi PHP yang ingin digunakan juga sangat mudah dilakukan dengan mengunakan bantuan PHP-Switch yang telah disertakan oleh XAMPP, dan yang terpenting XAMPP bersifat *free* atau gratis untuk digunakan. Fungsi XAMPP sendiri adalah sebagai server yang berdiri sendiri (localhost) yang terdiri beberapa program antara lain : Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

2.4.2 MySQL

Menurut (Bunafit Nugroho, 2013) MySQL merupakan *database* yang paling digemari di kalangan *programmer* Web, dengan alasan bahwa program ini merupakan *database* yang sangat kuat dan cukup stabil untuk digunakan sebagai

media penyimpan data. Sebagai sebuah *database server* yang mampu untuk memanajemen *database* dengan baik.

MySQL mempunyai tiga sub bahasa, yaitu :

1. *Data Definition Language* (DDL)

   DDL merupakan kelompok perintah yang berfungsi untuk mendefinisikan atribut-atribut basis data, tabel atribut (kolom), batasan-batasan terhadap suatu atribut, serta hubungan antara tabel. Yang termasuk dalam kelompok DDL ini adalah *Create, Alter*, dan *Drop.*

2. *Data Manipulation Language* (DML)

   DML adalah kelompok perintah yang berfungsi untuk memanipulasi data dalam basis data. Misalnya untuk pengambilan, penyisipan, pengubahan dan penghapusan data. Perintah yang temasuk dalam kategori DML adalah:

   a. *select* : memilih data;
   b. *insert* : menambah data;
   c. *delete* : menghapus data;
   d. *update* : mengubah data.

3. *Data Control Language* (DCL)

   DCL berisi perintah untuk mengendalikan pengaksesan data. Pengendalian data dapat dilakukan berdasarkan per pengguna, per tabel, per kolom, maupun per operasi yang boleh dilakukan. Perintah-perintah yang termasuk DCL adalah:

   1. *grant* : memberikan kendali pengaksesan data;
   2. *revoke* : mencabut kemampuan pengaksesan data;
   3. *block table* : mengunci tabel.

2.4.3 PHP

Menurut Kadir (2013:120) PHP adalah bahasa pemrograman yang ditujukan untuk membuat aplikasi berbasis web, *server-side scripting* yang menyatu dengan HTML. Maksud dari *server-side scripting* adalah *syntax* dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di server tetapi disertakan pada dokumen html. Pembuatan *web* ini merupakan kombinasi antara PHP sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman *web*.

PHP merupakan *script* untuk pemrograman. *Script web server side* adalah *script* untuk membuat dokumen *Hypertext Markup Language* (HTML). Dokumen yang dihasilkan melalui aplikasi bukan dokumen yang dibuat dengan editor teks. Dengan menggunakan PHP maka *maintenance* suatu situs *web* menjadi mudah, proses *update* data dapat dilakukan menggunakan aplikasi yang disebut *script* PHP. Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (*software* di belakang Wikipedia). PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#VB.NET Microsoft, ColdFusion Macromedia dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Pustnuke, dan Xaraya.

Berikut contoh gambar PHP saat dijalankan:

Gambar 2.3 Tampilan PHP saat dijalankan

2.4.4 *CodeIgniter*

CodeIgniter adalah sebuah *Framework* PHP yang bersifat *open source* dan menggunakan model *MVC (Model, View, Controller.* Kelebihan *CodeIgniter* adalah membuat web dinamis dan memudahkan developer untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal.

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi atau prosedur dan *class-class* untuk tujuan tertentu yang sudah siap digunakan sehingga lebih mempermudah dan mempercepat *programmer*, tanpa harus membuat fungsi atau *class* dari awal.

2.4.5 Atom

Atom adalah sebuah text editor yang memiliki lisensi *open source* yang tersedia untuk platform OS X, Linux dan Windows. Atom ini dibuat oleh GitHub dan di klaim sebagai text editor yang bisa di custom dengan merubah file configurasinya. Keunggulan dari atom text editor adalah tersedianya banyak plugins yang bisa membantu dan melihat plugins yang paling banyak digunakan.

2.4.6 *Web Browser*

Web browser adalah sebuah *software* yang digunakan untuk mengakses, menampilkan halaman web. *Browser* berkomunikasi dengan *web server* melalui protokol HTTP, yang membaca dan menerjemahkan bahasa HTML kemudian menampilkannya secara visual sehingga informasi yang ada dapat dibaca. Beberapa contoh *browser*:

- *Internet Explore (IE),* adalah *browser* yang dibuat oleh *Microsoft* dan dibundel dengan sistem operasi *Windows*.
- *Mozilla* bersifat *open source* dan dikembangkan oleh komunitas *programmer*.
- *Opera*. *Browser* ini mempunyai kelebihan dalam kecepatan *download* halaman *web*, menyatukan banyak *windows* dalam satu *interface* sehingga tidak terlalu merepotkan user dalam berpindah-pindah halaman.
- *Firefox*, menggunakan teknologi *Mozilla* sebagai dasarnya. Kelebihannya adalah ukuran *file* yang lebih kecil dibanding dengan *file browser* lainya.