



**TUGAS AKHIR – KI141502**

**RANCANG BANGUN SISTEM  
PENGUSULAN E-INTERNSHIP SECARA  
KOLABORATIF**

**Dhanang Cindra Riayawan**  
NRP. 5107 100 035

Dosen Pembimbing 1  
**Dr.Eng. RADITYO ANGGORO, S.Kom., M.Sc.**  
NIP. 198410162008121002

Dosen Pembimbing 2  
**Ir. F.X. ARUNANTO, M.Sc.**  
NIP. 195701011983031004

JURUSAN TEKNIK INFORMATIKA  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2015



**FINAL PROJECT – KI141502**

# **DEVELOPING COLLABORATIVE E- INTERNSHIP INFORMATION SYSTEM**

**Dhanang Cindra Riayawan**  
NRP. 5107 100 035

Advisor

**Dr.Eng. RADITYO ANGGORO, S.Kom., M.Sc.**  
**NIP. 198410162008121002**

Second Advisor

**Ir. F.X. ARUNANTO, M.Sc.**  
**NIP. 195701011983031004**

DEPARTMENT OF INFORMATICS  
Faculty of Information Technology  
Sepuluh Nopember Institute Of Technology  
Surabaya 2015

# LEMBAR PENGESAHAN

## RANCANG BANGUN SISTEM PENGUSULAN E- INTERNSHIP SECARA KOLABORATIF

### TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Arsitektur Jaringan Komputer  
Program Studi S-1 Jurusan Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh:

**DHANANG CINDRA RIAYAWAN**  
NRP. 51071000035

Disetujui oleh Dosen Pembimbing Tugas Akhir

1. Radityo Anggoro, S.Kom. M.Sc.  
(NIP. 198410162008121002)
2. Ir. F.X. Arunanto, M.Sc.  
(NIP. 195701011983031004)



*[Signature]*  
.....  
(Pembimbing 1)

*[Signature]*  
.....  
(Pembimbing 2)

**SURABAYA**

## **“RANCANG BANGUN SISTEM PENGUSULAN E- INTERNSHIP SECARA KOLABORATIF”**

**Nama Mahasiswa** : Dhanang Cindra Riayawan  
**NRP** : 5107 100 035  
**Jurusan** : Teknik Informatika FTIF-ITS  
**Dosen Pembimbing 1** : Radityo Anggoro, S.Kom., M.Sc.  
**Dosen Pembimbing 2** : Ir. F.X. Arunanto, M.Sc.

### **ABSTRAK**

*Kebutuhan akan sistem online meningkat seiring dengan banyaknya manfaat dan kemudahan yang disediakan untuk membantu berbagai macam pekerjaan manusia. Perkembangan teknologi dan tingginya kebutuhan terhadap teknologi yang dapat mempermudah pekerjaan manusia menuntut sistem monitoring dan penilaian mahasiswa kerja praktik yang pada awalnya diproses secara manual untuk di "online"kan.*

*Kerja praktik merupakan hal yang penting dan selalu ada di sebuah universitas. Dan prosedur kerja praktik ini pun lumayan rumit sehingga seringkali membutuhkan waktu yang cukup lama bagi mahasiswa maupun dosen untuk mengurusnya.*

*Saat ini proses manajemen kerja praktik di Teknik Informatika ITS masih dilakukan secara manual. Oleh karena itu diperlukan sebuah sistem yang mampu mengelola proses manajemen kerja praktik secara komputerisasi. Sehingga diharapkan prosesnya dapat berjalan lebih cepat, akurat, dan efisien.*

*Di dalam sistem online ini kita memanfaatkan bahasa php dan menggunakan database MySQL. Sehingga proses manajemen kerja praktik mulai proses pengajuan, disetujui, pemilihan pembimbing sampai penilaian dapat dilakukan secara online.*

*Selain itu pada sistem ini, terdapat menu monitoring kerja praktik, sehingga mahasiswa bimbingan dapat dipantau keaktifannya selama menjalani masa kerja praktik.*

***Kata kunci: E-INTERNSHIP, KERJA PRAKTIK, PENGAJUAN, PERSETUJUAN.***

## **“DEVELOPING COLLABORATIVE E-INTERNSHIP INFORMATION SYSTEM”**

**Student’s name** : Dhanang Cindra Riayawan  
**NRP** : 5107 100 035  
**Major of Department** : Informatics, FTIF-ITS  
**First Advisor** : Radityo Anggoro, S.Kom., M.Sc.  
**Second Advisor** : Ir. F.X. Arunanto, M.Sc.

### **ABSTRACT**

*The rapid improvement in the world of IT, supported with invention and innovation have brought a lot of change in human life. More and more aspect in our live using IT to operate.*

*Internship is an essential curriculum in every university. These internship grading and monitoring procedure is pretty complicated, which often take considerable amount of time for student and lecturer to take care of it.*

*As for now, internship management in Information Department of Institut Teknologi Sepuluh Nopember still using manual way. Therefore we need an online system that is able to manage the process of internship monitoring and grading. So we can hope that the process will be faster, more accurate and efficient.*

*In these online system, we use php language and MySQL as database. So that the process of internship monitoring and grading management, starting from submission, approval, advisor appointment, and grading process can be done online. In addition, this system also provide monitoring process so that the advisor can check student activeness.*

***Keywords: E-INTERNSHIP, SUBMISSION, APPROVAL.***

## KATA PENGANTAR

Puji syukur penulis panjatkan bagi Allah SWT, karena atas limpahan rahmat dan hidayahNya sehingga penulis bisa menyelesaikan Tugas Akhir yang berjudul "***Rancang Bangun Sistem Pengusulan E-Internship Secara Kolaboratif***". Penulis berharap semoga Tugas Akhir ini bermanfaat bagi ilmu pengetahuan, khususnya bidang Teknologi Informasi.

Terselesainya Tugas Akhir ini juga tidak lepas dari dukungan dan doa banyak pihak, oleh karena itu penulis dengan segala hormat mengucapkan terima kasih kepada:

1. Kedua orang tua penulis, Ibu Ismi Hartini dan Bapak Sutardjo, atas doa yang selama ini mengiringi setiap nafas penulis.
2. Bapak Dr.Eng. Radityo Anggoro, S.Kom., M.Sc dan Bapak Ir. F.X. Arunanto M.Sc, selaku dosen pembimbing Tugas Akhir ini, yang telah memberikan kepercayaan, motivasi, bimbingan, perhatian, serta semua bantuan yang telah diberikan kepada penulis.
3. Ibu Dr.Eng. Nanik Suciati, S.Kom, M.Kom selaku Ketua Jurusan Teknik Informatika, Ibu Isye Arieshanti, S.Kom, M.Phil, selaku Dosen Wali penulis, dan seluruh dosen Teknik Informatika ITS atas segala ilmu yang telah diamalkan pada penulis.
4. Pak Yudi serta seluruh staf TU Teknik Informatika ITS yang telah memberikan segala bantuan dan kemudahan kepada penulis selama menjalani kuliah di Teknik Informatika ITS.
5. Teman-teman penulis, yang selalu memberikan bantuan baik pikiran maupun tenaga sehingga memberikan kelancaran dalam mengerjakan.
6. Teman-teman sekontrakkan dan seperjuangan Handito Putro Utomo, Afwan Burhan Prahasto, Dimas Angga

Prayoga, dan A. Enang Nugroho, terima kasih atas dukungan dan motivasinya.

7. Untuk Teja dan Fikar, terimakasih sudah banyak membantu penulis dalam penyelesaian tugas akhir ini.
8. Untuk Savitri Kurniasari yang tiada lelah selalu memberikan semangat untuk menyelesaikan tugas akhir ini.
9. Semua pihak yang secara langsung ataupun tidak mendukung penulis menyelesaikan Tugas Akhir ini sehingga Tugas Akhir ini dapat diselesaikan.

Besar harapan penulis untuk kritik dan saran dari pembaca demi kesempurnaan dan perbaikan ke depan.

Surabaya, Juni 2015

## DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK .....	ix
ABSTRACT .....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR TABEL .....	xxi
BAB 1.....	1
PENDAHULUAN.....	1
1.1    Latar Belakang .....	1
1.2    Rumusan masalah.....	1
1.3    Batasan masalah .....	2
1.4    Tujuan dan Manfaat.....	2
1.5    Metodologi .....	2
BAB 2.....	5
LANDASAN TEORI.....	5
2.1    Apa itu PHP ?.....	5
2.1.1    Sejarah PHP.....	5
2.1.2    Cara kerja PHP .....	6
2.1.3    Sintaks Dasar Pada PHP.....	7
2.1.4    Kelebihan PHP dari bahasa pemrograman lain. ....	9
2.2    XAMPP .....	10
2.2.1    Asal kata dari XAMPP .....	10

2.2.2	Bagian penting XAMPP .....	11
2.3	SQL (Structured Query Language) .....	12
2.3.1	Tabel pada SQL.....	13
2.3.2	Query pada SQL.....	13
2.3.3	Mengetahui DML pada SQL .....	14
2.3.4	Mengetahui DDL pada SQL.....	14
2.3.5	MySQL.....	15
2.4	Code Igniter.....	19
2.4.1	Framework.....	19
2.4.2	Design Patern: MVC (Model, View, Controller) .....	20
2.5	Arsitektur Aplikasi Web.....	21
BAB 3.....		23
PERANCANGAN PERANGKAT LUNAK.....		23
3.1	Deskripsi Umum.....	23
3.2	Flowchart.....	24
3.3	Rancang Bangun Sistem Desain.....	26
3.3.1	Usecase Diagram.....	26
3.3.2	DFD Diagram .....	27
3.4	Struktur Database .....	27
BAB 4.....		33
IMPLEMENTASI PERANGKAT LUNAK .....		33
4.1	Lingkungan Pembangun Perangkat Lunak.....	33
4.2	Implementasi Perangkat Lunak .....	34
4.2.1	Operasi PHP pada CodeIgniter.....	34

4.2.2	Implementasi Fungsi <i>Sign-up</i> .....	37
4.2.3	Implementasi Fungsi <i>Login</i> .....	38
4.2.4	Administrator.....	41
4.2.5	Modul Mahasiswa .....	42
BAB 5.....		47
UJI COBA DAN EVALUASI .....		47
5.1	Lingkungan uji coba.....	47
5.2	Data Uji Coba.....	48
5.3	Uji Coba Fungsionalitas .....	48
5.3.1	Fungsi <i>Sign-up</i> .....	48
5.3.2	Fungsi <i>Login</i> .....	50
BAB 6.....		55
KESIMPULAN DAN SARAN .....		55
6.1	Kesimpulan.....	55
6.2	Saran.....	55
DAFTAR PUSTAKA.....		57
BIODATA PENULIS.....		59

## **DAFTAR TABEL**

Tabel 3.1 Tabel tbl_mahasiswa .....	28
Tabel 3.2 Tabel tbl_dosen .....	28
Tabel 3.3 Tabel tbl_admin.....	29
Tabel 3.4 Tabel tbl_dosen_bimb .....	29
Tabel 3.5 Tabel tbl_dosen_uji .....	30
Tabel 3.6 Tabel tbl_nilaiuji .....	30
Tabel 3.7 Tabel tbl_tempatkp.....	31
Tabel 3.8 Tabel tbl_login .....	31
Tabel 3.9 Tabel tbl_info .....	32
Tabel 4.1 : Lingkungan implementasi .....	34
Tabel 5.1 : Lingkungan uji coba.....	47

## DAFTAR GAMBAR

Gambar 2.2 : cara kerja PHP .....	7
Gambar 2.3 : contoh fungsi pada PHP. ....	9
Gambar 2.4 : contoh tabel pada SQL .....	13
Gambar 2.5 : Contoh Hasil Query .....	14
Gambar 2.6 : Create Table .....	16
Gambar 2.7 : Constraint pada MySQL.....	17
Gambar 2.8 : Select Table .....	17
Gambar 2.9 : Insert Into Table .....	18
Gambar 2.10 : Update Table .....	18
Gambar 2.11 : Delete field Table .....	19
Gambar 2.12 : Arsitektur Aplikasi Web.....	22
Gambar 3.1 : flowchart diagram.....	24
Gambar 3.2 : flowchart koordinator KP.....	24
Gambar 3.3 : flowchart diagram mahasiswa. ....	25
Gambar 3.4 : Usecase diagram.....	26
Gambar 3.5 : DFD diagram.....	27
Gambar 4.1 .....	35
Gambar 4.2.....	35
Gambar 4.3.....	36
Gambar 4.4.....	36
Gambar 4.5.....	36
Gambar 4.6.....	37
Gambar 4.7 : fungsi <i>Sign-up</i> .....	38

Gambar 4.8 : Fungsi <i>login</i> .....	39
Gambar 4.9 : <i>login</i> sebagai mahasiswa .....	40
Gambar 4.10 : <i>login</i> sebagai dosen .....	40
Gambar 4.11 : <i>login</i> sebagai admin.....	41
Gambar 4.12 : validasi user mahasiswa .....	42
Gambar 4.13 : implementasi pengajuan izin kerja praktek .....	43
Gambar 4.14 : fungsi detail KP mahasiswa.....	45
Gambar 5.1 : Halaman awal SimKP.....	48
Gambar 5.2 : Halaman <i>Sign-up</i> .....	49
Gambar 5.3 : Contoh <i>sign-up</i> berhasil.....	49
Gambar 5.4 : Contoh <i>sign-up</i> gagal.....	50
Gambar 5.5 : Halaman <i>login</i> sebagai admin .....	51
Gambar 5.6 : Validasi mahasiswa .....	51
Gambar 5.7 : Halaman validasi mahasiswa.....	52
Gambar 5.8 : Validasi mahasiswa berhasil.....	53
Gambar 5.9 : halaman <i>login</i> sebagai mahasiswa.....	53
Gambar 5.10 : halaman pengajuan kerja praktek .....	54
Gambar 5.11 : Pengajuan kerja praktek berhasil.....	54

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kerja praktek adalah suatu bentuk pendidikan dengan cara memberikan pengalaman belajar bagi mahasiswa untuk berpartisipasi dengan tugas langsung di Lembaga BUMN, BUMD, Perusahaan Swasta, dan Instansi Pemerintahan setempat.

Kerja Praktek akan menambah kemampuan untuk mengamati, mengkaji serta menilai antara teori dengan kenyataan yang terjadi dilapangan yang pada akhirnya dapat meningkatkan kualitas managerial mahasiswa dalam mengamati permasalahan dan persoalan, baik dalam bentuk aplikasi teori maupun kenyataan yang sebenarnya.

Kerja Praktek memberi kesempatan kepada mahasiswa untuk mengenal dan mengetahui secara langsung tentang instansi sebagai salah satu penerapan disiplin dan pengembangan karier. Ketika di lapangan melaksanakan praktek kerja, mahasiswa dapat menilai tentang pengembangan dari ilmu yang mereka miliki.

Pada tugas akhir ini penulis akan membangun sebuah sistem informasi berbasis web yang dapat memudahkan mahasiswa teknik informatika dalam melakukan pengajuan izin kerja praktek.

### **1.2 Rumusan masalah**

Rumusan masalah yang diangkat pada tugas akhir ini dapat dipaparkan sebaagi berikut:

1. Bagaimana merancang sistem informasi yang memudahkan mahasiswa untuk melakukan pengajuan izin kerja praktek.
2. Sulitnya pihak jurusan dalam mendata mahasiswa yang sedang melakukan kerja praktek.

### 1.3 Batasan masalah

Batasan dalam tugas akhir ini adalah:

1. Sistem informasi berbasis *web* ini hanya digunakan untuk melakukan pengajuan izin kerja praktek.
2. Bahasa pemrograman yang digunakan dalam tugas akhir ini adalah PHP dan HTML serta *XAMPPserver* sebagai *database*.
3. Fokus dari tugas akhir ini adalah untuk mempermudah mahasiswa dalam melakukan pendaftaran kerja praktek.

### 1.4 Tujuan dan Manfaat

Tujuan dan manfaat dari pembuatan tugas akhir ini adalah:

1. Merancang sistem informasi berbasis *web* yang memudahkan mahasiswa untuk melakukan pengajuan izin kerja praktek.
2. Memudahkan pihak jurusan dalam mendata mahasiswa yang sedang melakukan kerja praktek.

### 1.5 Metodologi

Berikut merupakan langkah-langkah yang dilakukan dalam pembuatan tugas akhir:

- Studi Literatur  
Pada tahapan ini akan dilakukan studi literatur mengenai metode yang digunakan, yaitu bagaimana membangun sebuah sistem informasi dengan menggunakan bahasa pemrograman PHP dan HTML.
- Perancangan Perangkat Lunak dan Desain Sistem  
Pada tahapan ini dilakukan proses perancangan perangkat lunak, berdasarkan literatur yang telah di kaji kemudian di buat desain model data, diagram alir proses-proses yang ada, dan desain antar muka aplikasi. Yang selanjutnya akan diimplementasikan.
- Implementasi Sistem

Pada tahapan implementasi ini akan dilakukan proses pembuatan sistem informasi menggunakan PHP dan HTML yang biasa digunakan untuk membangun sebuah situs *web*.

- Pengujian dan Evaluasi  
Pengujian aplikasi *web* ini dapat dilakukan antara lain sebagai berikut:
  1. *User* dapat melakukan *sign-up* untuk mendapatkan *account* yang digunakan untuk melakukan pendaftaran kerja praktek.
  2. *User* dapat melakukan *sign-in* dan kemudian mengisi *form* pendaftaran kerja praktek.
- Penyusunan Buku Tugas Akhir  
Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pengerjaan serta uji coba yang dilakukan. Secara garis besar, Buku Tugas Akhir yang nantinya akan dibuat terdiri dari beberapa bagian antara lain:

#### 1. BAB I. PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan permasalahan, tujuan dan manfaat pembuatan tugas akhir, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

#### 2. BAB II. LANDASAN TEORI

Bab ini membahas beberapa pengertian dan teori penunjang yang berhubungan dengan pokok pembahasan pembuatan tugas akhir ini.

#### 3. BAB III. PERANCANGAN PERANGKAT LUNAK

Bab ini membahas desain dari sistem yang akan dibuat meliputi arsitektur dan proses kerja perangkat lunak.

4. BAB IV. IMPLEMENTASI PERANGKAT LUNAK

Bab ini membahas implementasi dari desain sistem yang dilakukan pada tahap desain, melalui potongan *pseudocode* yang terdapat dalam perangkat lunak.

5. BAB V. UJI COBA DAN EVALUASI

Bab ini membahas uji coba dari perangkat lunak yang dibuat dengan melihat keluaran yang dihasilkan oleh perangkat lunak, analisis dan evaluasi untuk mengetahui kemampuan perangkat lunak.

6. BAB VI. PENUTUP

Bab ini berisi kesimpulan dari hasil uji coba serta saran untuk pengembangan lebih lanjut.

## BAB 2

### LANDASAN TEORI

Pada bab ini akan dibahas mengenai teori yang menjadi dasar dari pembuatan tugas akhir ini. Teori yang dibahas mencakup tentang elemen-elemen yang terkait dalam topik tugas akhir mulai dari sumber dari permasalahan, pendekatan yang digunakan, serta metode dan teknologi yang digunakan untuk pengerjaan tugas akhir ini.

#### 2.1 Apa itu PHP ?

**PHP: Hypertext Preprocessor** adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS.

##### 2.1.1 Sejarah PHP

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut

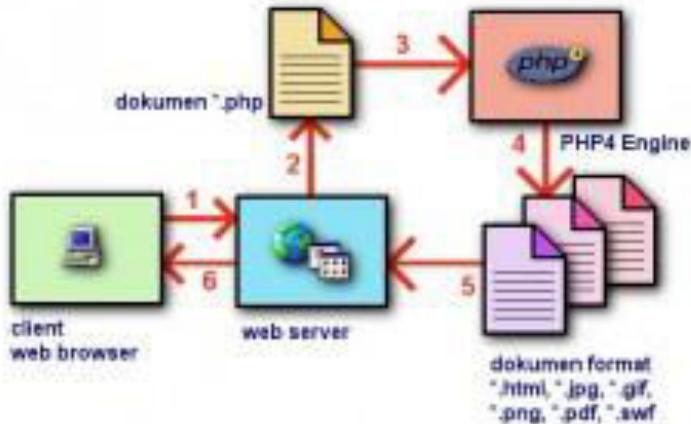
merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

### **2.1.2 Cara kerja PHP**

Dalam cara kerjanya PHP, yaitu pertama komputer pengguna menjalankan file PHP, yang kemudian file PHP tersebut di kirim ke web server dan setelah itu oleh web server, dikirim lagi ke engine PHP. Di dalam engine PHP inilah file PHP di eksekusi dan menghasilkan file HTML. File HTML ini kemudian dikirim lagi ke web server dan nantinya akan ditampilkan ke browser pengguna dalam bentuk plain HTML.



Gambar 2.2 : cara kerja PHP

### 2.1.3 Sintaks Dasar Pada PHP

Kode PHP disimpan sebagai plain text dalam format ASCII, sehingga kode PHP dapat ditulis hampir di semua editor text seperti windows notepad, windows wordpad, dll. Kode PHP adalah kode yang disertakan di sebuah halaman HTML dan kode tersebut dijalankan oleh server sebelum dikirim ke browser.

- Pembatas

PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Apapun di luar pembatas tidak diproses oleh PHP (meskipun teks PHP ini masih mengendalikan struktur yang dijelaskan dalam kode PHP. Pembatas yang paling umum adalah "`<?php`" untuk membuka dan "`?>`" untuk menutup kode PHP. Tujuan dari pembatas ini adalah untuk memisahkan kode PHP dari kode diluar PHP, seperti HTML, Javascript.

- Variabel

Variabel diawali dengan simbol dolar '\$'. Pada versi *php* PHP 5 diperkenalkan jenis isyarat yang memungkinkan fungsi untuk memaksa mereka menjadi parameter objek dari class tertentu, array, atau fungsi. Namun, jenis petunjuk tidak dapat digunakan dengan jenis skalar seperti angka atau string. Contoh variabel dapat ditulis sebagai '\$nama\_variabel'.

Penulisan fungsi, penamaan kelas, nama variabel sangat peka akan huruf besar (Kapital) dan huruf kecil. Kedua kutip ganda "" dari *string* memberikan kemampuan untuk interpolasi nilai variabel ke dalam *string* PHP. PHP menerjemahkan baris sebagai spasi, dan pernyataan harus diakhiri dengan titik koma ;.

- *Comment*

PHP memiliki 3 jenis sintaks sebagai komentar pada kode yaitu tanda blok /\* \*/ , komentar 2 baris // Serta tanda pagar # digunakan untuk komentar satu baris. Komentar bertujuan untuk meninggalkan catatan pada kode PHP dan tidak akan diterjemahkan ke program.

- Fungsi (*function*)

Ratusan fungsi yang disediakan oleh PHP serta ribuan lainnya yang tersedia melalui berbagai ekstensi tambahan. fungsi-fungsi ini didokumentasikan dalam dokumentasi PHP. Namun, dalam berbagai tingkat pengembangan, kini memiliki berbagai konvensi penamaan. Sintaks fungsi adalah seperti dibawah ini:

```
function tampilkan($data="") // Mendefinisikan fungsi, "tampilkan" adalah nama sebuah fungsi
{ // Diapit oleh tanda kurung kurawal
  if($data) return $data; else return 'Tidak ada data'; // Melakukan proses pengolahan data, contohnya melalui kondisi
}
echo tampilkan("sni halaman") // Menjalankan fungsi
```

Gambar 2.3 : contoh fungsi pada PHP.

#### 2.1.4 Kelebihan PHP dari bahasa pemrograman lain.

Beberapa kelebihan PHP dari bahasa pemrograman web, antara lain :

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

## 2.2 XAMPP

**XAMPP** adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya.

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (Core Team), Tim Pengembang (Development Team) & Tim Dukungan (Support Team).

### 2.2.1 Asal kata dari XAMPP

**XAMPP** adalah singkatan yang masing-masing hurufnya adalah :

- **X** : Program ini dapat dijalankan pada bermacam sistem operasi, seperti *Windows*, *Linux*, *Mac OS*, dan *Solaris*.
- **A** : Merupakan singkatan dari *Apache*; aplikasi *web server*. Tugas utama *Apache* adalah menghasilkan halaman *web* yang benar kepada user berdasarkan kode PHP yang dituliskan oleh pembuat halaman *web*. jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam *MySQL*) untuk mendukung halaman web yang dihasilkan.

- **M** : *MySQL*, merupakan aplikasi database *server*. Perkembangannya disebut SQL yang merupakan kepanjangan dari Structured Query Language. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah database. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam database.
- **P** : PHP, bahasa pemrograman web. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat web yang bersifat server-side scripting. PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP adalah MySQL. namun PHP juga mendukung sistem manajemen database *Oracle*, *Microsoft Access*, *Interbase*, *d-base*, *PostgreSQL*, dan sebagainya.
- **P** : Perl, bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin *Unix*. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (*SunOS*, *Linux*, *BSD*, *HP-UX*), juga tersedia untuk sistem operasi seperti *DOS*, *Windows*, *PowerPC*, *BeOS*, *VMS*, *EBCDIC*, dan *PocketPC*.

### 2.2.2 Bagian penting XAMPP

Mengenal bagian XAMPP yang biasa digunakan pada umumnya.

- **htdocs** adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
- **phpMyAdmin** merupakan bagian untuk mengelola *database* MySQL yang ada di komputer. Untuk

membukanya, buka browser lalu ketikkan alamat <http://localhost/phpMyAdmin>, maka akan muncul halaman phpMyAdmin.

- **Kontrol Panel** yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

### 2.3 SQL (Structured Query Language)

SQL adalah suatu bahasa komputer yang mengikuti standard ANSI (American National Standard Institute), yaitu sebuah bahasa standard yang digunakan untuk mengakses dan melakukan manipulasi suatu sistem *database*. Statemen dalam SQL dapat digunakan untuk mengakses data atau meng-*update* data pada suatu *database*. SQL utamanya berfungsi dalam suatu relational database seperti misalnya Oracle, SQL Server, DB2, Informix, Sybase, MS Acces, MySQL, Firebird dan masih banyak lagi yang lainnya.

Dengan adanya berbagai perusahaan/vendor yang membuat berbagai produk SQL maka efeknya adalah timbul berbagai macam perbedaan dalam bahasa SQL yang dikembangkan oleh tiap-tiap perusahaan tersebut. Contohnya saja, bahasa SQL yang dimiliki oleh Microsoft (yang bernama T-SQL) akan berbeda dengan bahasa SQL yang dikembangkan oleh Oracle (yang bernama PL/SQL). Namun demikian semua vendor diwajibkan untuk mendukung bahasa standard yang ditentukan oleh ANSI, misalnya semua pengembang bahasa SQL wajib dalam mengimplementasikan kata kunci atau statemen standard SQL seperti SELECT, UPDATE, DELETE, INSERT, WHERE dan lain sebagainya. Perbedaan bahasa SQL yang dikembangkan oleh setiap vendor itu dinamakan extension atau juga disebut dengan dialek.

### 2.3.1 Tabel pada SQL

Suatu database berisi dari satu atau lebih tabel. Setiap tabel diberi nama tertentu yang berhubungan dengan isinya, contohnya tabel yang isinya mengenai data pelanggan diberi nama tabel Customers. Tabel itu sendiri berisi dari satu atau sekumpulan record. Record adalah satu baris data yang dibentuk oleh satu atau beberapa field. Field-field inilah yang berisi data.

Berikut ini adalah contoh dari suatu tabel yang berisi data. Berikut ini adalah contoh dari tabel yang bernama 'Karyawan'.

<b>LastName</b>	<b>FirstName</b>	<b>Address</b>	<b>City</b>
Hansen	Ola	Timoteivn 10	Sandnes
Svendson	Tove	Borgvn 23	Sandnes
Pettersen	Kari	Storgt 20	Stavanger

Gambar 2.4 : contoh tabel pada SQL

Tabel di atas terdiri dari tiga buah record. Masing -masing record terdiri dari empat buah field (LastName, FirstName, Address, City).

### 2.3.2 Query pada SQL

Dengan SQL kita bisa membuat sebuah query ke database yang hasilnya bisa dilihat. Query adalah istilah untuk mengambil data dari database dengan suatu kriteria tertentu. Contoh dari sebuah query adalah sebagai berikut.

"Select Last from Karyawan"

Dari query di atas akan menghasilkan data sebagai berikut :

LastName
Hansen
Svendson
Pettersen

Gambar 2.5 : Contoh Hasil Query

### 2.3.3 Mengenal DML pada SQL

SQL adalah sekumpulan sintaks-sintaks atau statemen untuk mengakses data dalam database, tetapi SQL sendiri juga bisa digunakan untuk melakukan proses insert, update atau delete ke dalam suatu database. Sintaks -sintaks ini yang disebut dengan Data Manipulation Language (DML) yang merupakan bagian dari SQL.

Berikut ini adalah penjelasan singkat dari sintaks -sintaks tersebut.

- **SELECT**, bertugas untuk mengakses data dari suatu tabel dalam database.
- **UPDATE**, bertugas untuk meng-update data dalam suatu tabel pada database.
- **DELETE**, bertugas untuk menghapus data dari suatu tabel dalam database.
- **INSERT**, bertugas untuk menambahkan data ke dalam suatu tabel dalam database.

### 2.3.4 Mengenal DDL pada SQL

DDL merupakan singkatan dari Data Definition Language yang juga bagian dari SQL. DDL ini berfungsi lebih ke dalam memanipulasi struktur dari database. Contohnya DDL ini bisa digunakan untuk membuat tabel atau menghapus tabel. Kita juga bisa membuat key atau index dengan menggunakan DDL ini, membuat relasi antar tabel juga bisa dilakukan dengan DDL ini. Beberapa statemen atau sintaks yang sering dijumpai dalam DDL adalah sebagai berikut :

- **CREATE TABLE**, bertugas untuk membuat tabel.
- **ALTER TABLE**, bertugas untuk merubah struktur suatu tabel.
- **DROP TABLE**, bertugas untuk menghapus suatu tabel.
- **CREATE INDEX**, bertugas untuk membuat suatu index dalam tabel.
- **DROP INDEX**, bertugas untuk menghapus suatu index dalam tabel.

### 2.3.5 MySQL

MySQL (*My Structure Query Language*) adalah salah satu database dari sekian banyak database lain seperti Oracle, MS SQL, PostgreSQL dan banyak lagi. Kesemuanya itu mempunyai fungsi dan manfaat yang hampir sama namun dalam pengerjaannya sedikit berbeda tetapi MySQL adalah penggunaan yang paling mudah. DBMS (Database Management System) merupakan salah satu system dalam mengakses database yang menggunakan bahasa SQL, MySQL menggunakan bahasa SQL dan dapat dikatakan sebagai DBMS. MySQL software open source. Open Source artinya memungkinkan untuk semua orang untuk menggunakan dan memodifikasi software. Setiap orang dapat men-download MySQL dari internet dan menggunakannya tanpa membayar apapun.

*Database MySQL* sangat cepat, *reliable*, dan mudah untuk digunakan. MySQL memiliki fungsi-fungsi standar SQL dan beberapa kemampuan tambahan. Berikut ini adalah perintah-perintah standar yang dapat digunakan untuk melakukan hampir semua hal yang perlu dilakukan terhadap sebuah *database*[5].

1. Create, Merupakan fungsi pada MySQL yang digunakan untuk membuat tabel pada *database*. Sintaksnya adalah :

```

CREATE TABLE <namatabel> (
    <kolom1><tipedata1>
    [<aturan1>],
    <kolom2><tipedata2>
    [<aturan2>],

```

Gambar 2.6 : Create Table

Berdasarkan gambar 2.3, [aturan<sub>n</sub>] berisi aturan untuk *field* yang bersangkutan, bersifat opsional, boleh ada dan boleh tidak. Jika lebih dari satu dibatasi dengan koma. Biasanya berupa :

- *NOT NULL*, yang berarti *field* tersebut harus ada isinya.
- *DEFAULT<nilai>*, yang berarti *field* tersebut jika tidak diisi nilainya, maka nilai defaultnya <nilai>.

[aturanTabel] berisi aturan-aturan yang berlaku untuk tabel tersebut. Jika aturan lebih dari satu, maka dibatasi oleh tanda koma. Aturan tabel biasanya :

- *primary key* (<DaftarKolomKey>)
- *foreignkey*(<daftarKolomForeignKey>  
*references*<namaTabel> (<daftarKolom>)  
 [ON DELETE <aturanDelete>] [ON UPDATE <aturanUpdate>])

[AturanTabel] bisa diberi nama. Jika diberi nama, maka sintaknya seperti yang terlihat pada gambar 2.5 berikut :

```
CONSTRAINT <namaAturan><AturanTabel>
```

Gambar 2.7 : Constraint pada MySQL

2. Select, merupakan fungsi pada MySQL yang digunakan untuk menampilkan *field – field* pada database diinginkan. Contoh untuk menampilkan keseluruhan data pada tabel yang ada pada *database* dapat dilihat pada gambar 2.5.

```
SELECT [DISTINCT] select_list
FROM table_source
[WHERE search_condition]
[GROUP BY group_by_expression]
[ORDER BY order_expression [ASC | DESC] ]
```

Gambar 2.8 : Select Table

Pada gambar 2.5 bisa dilihat bahwa

- *SELECT [distinct] select\_list* merupakan *field-field* pada tabel yang ingin ditampilkan.
- *FROM Table\_source* merupakan tabel sumber, dimana didalamnya terdapat *field-field* yang ingin ditampilkan.
- *WHERE Search\_condition* merupakan sebuah kondisi dimana akan mempengaruhi *field-field* yang ingin ditampilkan.
- *GROUP BY Group\_by\_expression* mengelompokkan berdasarkan *field* yang ditentukan.

- *ORDER BY Order\_expression [ASC | DESC]*, mengurutkan data berdasarkan field yang telah ditentukan baik *ascending* maupun *descending*.
3. Insert, merupakan fungsi pada MySQL yang digunakan untuk memasukkan data pada *database*. Sintaks insert pada tabel bisa dilihat pada gambar 2.6

```
INSERT INTO <namaTabel> [(field1,
field2, ... fieldn)] VALUES <ListValues>
```

Gambar 2.9 : Insert Into Table

- <namaTabel>, adalah nama dari tabel yang datanya ingin di masukkan.
  - [(*field1*, *field2*, ...*fieldn*)] merupakan *field-field* pada tabel yang dipilih. Secara *default* apabila *field1*, *field2*, ...*fieldn* tidak disertakan, maka nilai pada <ListValues> harus lengkap sesuai dengan urutan *field* yang ada pada tabel. Namun apabila *field1*, *field2*, ..*fieldn* disertakan, maka nilai <ListValues> harus sesuai dengan *field-field* yang dipilih.
4. Update, merupakan fungsi pada MySQL yang digunakan untuk mengubah isi data pada tabel yang ada di *database*. Sintaks untuk *update Table* bisa dilihat pada gambar 2.7

```
UPDATE <NamaTabel> SET
<field1>=<nilai1>[, <field2>=<nilai2>, ...
<fieldn>=<nilain>]
```

Gambar 2.10 : Update Table

Pada gambar 2.7 dapat dilihat bahwa :

- *<NamaTabel>*, merupakan nama tabel pada database yang ingin dirubah *field-field* nya.
  - *<field> = <nilai>*, nilai merupakan nilai baru yang ingin diubah pada *field*.
5. Delete, merupakan fungsi pada MySQL yang digunakan untuk menghapus data pada sebuah tabel yang ada di *database*. Sintaks *Delete* bisa dilihat padagambar 2.8

```
DELETE FROM <NamaTabel> [WHERE]
```

Gambar 2.11 : Delete field Table

## 2.4 Code Igniter

CodeIgniter merupakan aplikasi sumber terbuka yang berupa framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP. CodeIgniter memudahkan developer untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006.

### 2.4.1 Framework

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

Ada beberapa alasan mengapa menggunakan Framework:

- Mempercepat dan mempermudah pembangunan sebuah aplikasi web.

- Relatif memudahkan dalam proses maintenance karena sudah ada pola tertentu dalam sebuah framework (dengan syarat programmer mengikuti pola standar yang ada).
- Umumnya framework menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, pagination, multiple database, scaffolding, pengaturan session, error handling, dll).
- Lebih bebas dalam pengembangan jika dibandingkan CMS.

#### **2.4.2 Design Patern: MVC (Model, View, Controller)**

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu aplikasi yaitu :

1. View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.
2. Model, biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.
3. Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

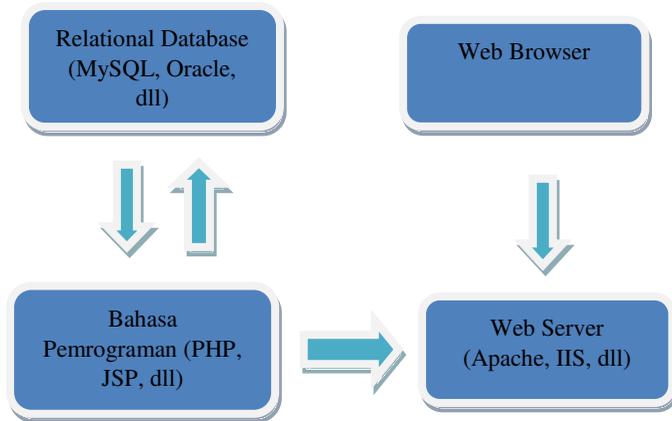
Dengan menggunakan prinsip MVC suatu aplikasi dapat dikembangkan sesuai dengan kemampuan developernya, yaitu programmer yang menangani bagian model dan controller, sedangkan designer yang menangani bagian view, sehingga penggunaan arsitektur MVC dapat meningkatkan maintainability dan organisasi kode. Walaupun demikian dibutuhkan komunikasi yang baik antara programmer dan designer dalam menangani variabel-variabel yang akan ditampilkan.

Ada beberapa kelebihan CodeIgniter (CI) dibandingkan dengan Framework PHP lain,

- Performa sangat cepat : salah satu alasan tidak menggunakan framework adalah karena eksekusinya yang lebih lambat daripada PHP from the scratch, tapi Codeigniter sangat cepat bahkan mungkin bisa dibilang codeigniter merupakan framework yang paling cepat dibanding framework yang lain.
- Konfigurasi yang sangat minim (nearly zero configuration) : tentu saja untuk menyesuaikan dengan database dan keleluasaan routing tetap diizinkan melakukan konfigurasi dengan mengubah beberapa file konfigurasi seperti database.php atau autoload.php, namun untuk menggunakan codeigniter dengan setting standard, anda hanya perlu mengubah sedikit saja file pada folder config.
- Banyak komunitas: dengan banyaknya komunitas CI ini, memudahkan kita untuk berinteraksi dengan yang lain, baik itu bertanya atau teknologi terbaru.
- Dokumentasi yang sangat lengkap : Setiap paket instalasi codeigniter sudah disertai user guide yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

## **2.5 Arsitektur Aplikasi Web**

Hubungan antara web server, database dan bahasa pemrograman bisa dilihat pada gambar dibawah ini



Gambar 2.12 : Arsitektur Aplikasi Web

Secara teknis database-lah yang menyimpan hampir semua data dan data tersebut diolah oleh bahasa pemrograman yang kemudian ditampilkan ke internet dengan menggunakan web server.

## **BAB 3**

### **PERANCANGAN PERANGKAT LUNAK**

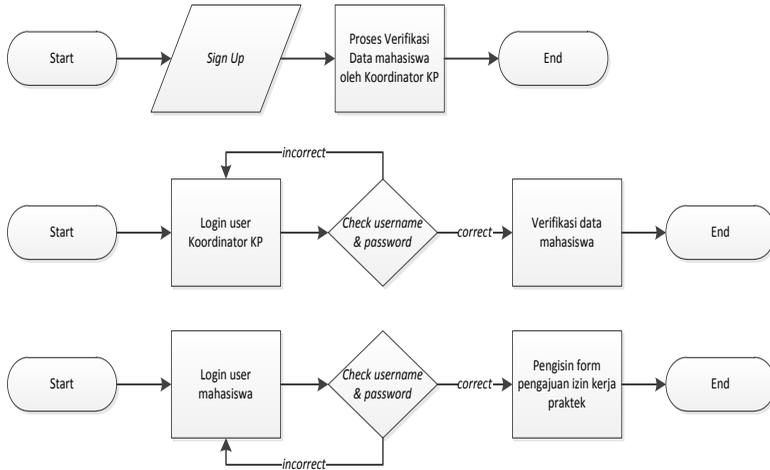
Pada bab ini akan dijelaskan hal-hal yang berkaitan dengan perancangan sistem yang akan dibuat pada karya tugas akhir ini, dimulai dengan deskripsi umum mengenai aplikasi yang akan dibuat, perancangan proses-proses yang ada dan alur prosesnya

#### **3.1 Deskripsi Umum**

Pada tugas akhir ini penulis akan membangun sebuah sistem informasi berbasis *web* yang dapat memudahkan mahasiswa teknik informatika dalam melakukan pengajuan izin kerja praktek.

Aplikasi yang akan dibangun ini memanfaatkan CI ( Code Igniter ) sebagai framework. Bahasa pemrograman yang digunakan adalah bahasa pemrograman PHP.

### 3.2 Flowchart

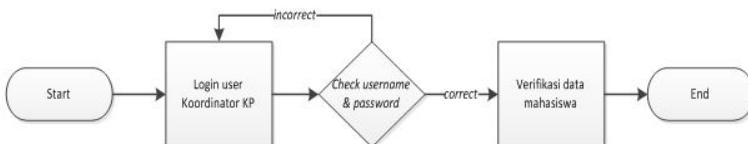


Gambar 3.1 : flowchart diagram

Flowchart digaram diatas menunjukkan bahwa aplikasi berbasis web ini memiliki 2 (dua) macam user :

#### 1. Koordinator KP (Kerja Praktek)

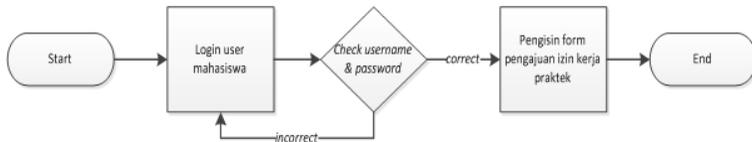
Koordinator KP merupakan admin dari aplikasi berbasis web ini, yang bertugas melakukan validasi pendaftaran untuk *login* pada web.



Gambar 3.2 : flowchart koordinator KP

## 2. Mahasiswa

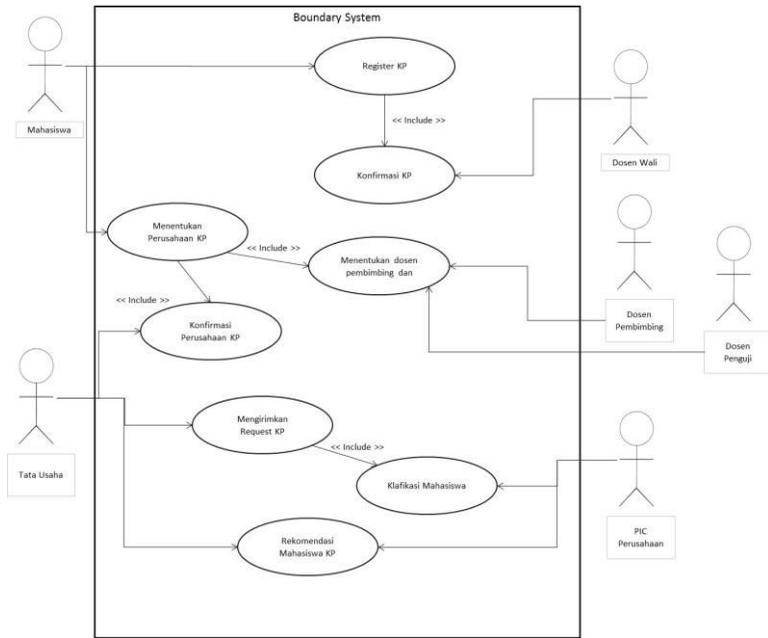
Sebelum bisa melakukan akses lebih ke dalam web ini, mahasiswa harus melakukan *sign-up* terlebih dahulu untuk mendapat sebuah *login account*. *Login account* didapat setelah user 'koordinator KP' melakukan validasi *account* tersebut.



Gambar 3.3 : flowchart diagram mahasiswa.

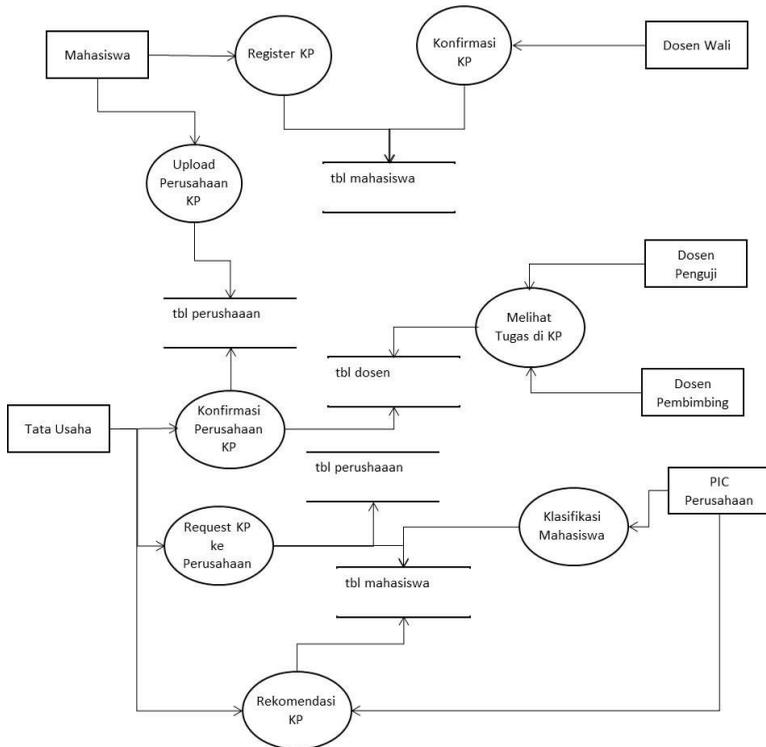
### 3.3 Rancang Bangun Sistem Desain

#### 3.3.1 Usecase Diagram



Gambar 3.4 : Usecase diagram

### 3.3.2 DFD Diagram



Gambar 3.5 : DFD diagram

### 3.4 Struktur Database

Struktur *Database* merupakan penjabaran dan penjelasan *database* tersebut, dari fungsi masing-masing *table* sampai masing-masing *field* yang ada dalam *table*. Adapun struktur *database* yang telah dibuat berdasarkan *Entity Relational Diagram*, yaitu:

1. Nama : **tbl\_mahasiswa**  
 Primary key : nim  
 Foreign key : -  
 Fungsi : Menyimpan data Mahasiswa

Tabel 3.1 Tabel tbl\_mahasiswa

No	Column	Data Type	Length
1.	nim	varchar	20
2.	nama_mahasiswa	varchar	100
3.	angkatan	integer	5
4.	jurusan	varchar	30
5.	valid	varchar	12

2. Nama : **tbl\_dosen**  
 Primary key : kd\_dosen  
 Foreign key : -  
 Fungsi : Menyimpan data Dosen

Tabel 3.2 Tabel tbl\_dosen

No	Column	Data Type	Length
1.	kd_dosen	varchar	5
2.	nidn	varchar	10
3.	nama_dosen	varchar	30

3. Nama : **tbl\_admin**  
 Primary key : username  
 Foreign key : -  
 Fungsi :Menyimpan data Admin/Koordinator  
 Kerja Praktek.

Tabel 3.3 Tabel tbl\_admin

No	Column	Data Type	Length
1.	Username	varchar	50
2.	nama_lengkap	varchar	100

4. Nama : **tbl\_dosen\_bimb**  
 Primary key : -  
 Foreign key : nim, kd\_dosen  
 Fungsi :Menyimpan data Dosen Pembimbing  
 Kerja Praktek.

Tabel 3.4 Tabel tbl\_dosen\_bimb

No	Column	Data Type	Length
1.	Nim	varchar	20
2.	kd_dosen	varchar	20

5. Nama : **tbl\_dosen\_uji**  
 Primary key : -  
 Foreign key : nim, kd\_dosen  
 Fungsi :Menyimpan data Dosen Penguji Kerja  
 Praktek.

Tabel 3.5 Tabel tbl\_dosen\_uji

No	Column	Data Type	Length
1.	nim	varchar	20
2.	kd_dosen	varchar	20

6. Nama : **tbl\_nilaiuji**  
 Primary key : -  
 Foreign key : nim  
 Fungsi :Menyimpan data nilai Mahasiswa Kerja  
 Praktek.

Tabel 3.6 Tabel tbl\_nilaiuji

No	Column	Data Type	Length
1.	nim	varchar	20
2.	kd_dosen	varchar	20
3.	nilai	integer	20

7. Nama : **tbl\_tempatkp**  
 Primary key : -  
 Foreign key : nim  
 Fungsi :Menyimpan data tempat Kerja Praktek.

Tabel 3.7 Tabel tbl\_tempatkp

No	Column	Data Type	Length
1.	Nim	varchar	20
2.	nama_kp	varchar	100
3.	alamat_kp	varchar	100
4.	notelp_kp	varchar	20

8. Nama : **tbl\_login**  
 Primary key : username  
 Foreign key : -  
 Fungsi : Menyimpan data login.

Tabel 3.8 Tabel tbl\_login

No	Column	Data Type	Length
1.	Username	varchar	50
2.	Password	varchar	75
3.	Stts	varchar	10

9. Nama : **tbl\_info**  
Primary key : kd\_info  
Foreign key : -  
Fungsi : Menyimpan data Info Kerja Praktek.

Tabel 3.9 Tabel tbl\_info

No	Column	Data Type	Length
1.	kd_info	integer	10
2.	judul	varchar	200
3.	waktu_post	varchar	30
4.	isi	text	

## **BAB 4**

### **IMPLEMENTASI PERANGKAT LUNAK**

Pada bab ini akan dibahas tentang implementasi dari perancangan perangkat lunak yang telah dijelaskan pada bab 3.

#### **4.1 Lingkungan Pembangun Perangkat Lunak**

Didalam membangun sistem perangkat lunak ini dibutuhkan baik *software* maupun *hardware*. Untuk server sendiri, menggunakan Sistem Operasi Windows 7 yang didalamnya sudah terinstall MySQL, XAMPP, dan juga PHPMyadmin. Sedangkan untuk client, membutuhkan *browser application* seperti mozilla, Internet Explorer, dll.

Spesifikasi *software* dan *hardware* yang digunakan dalam implementasi ini ditampilkan pada tabel dibawah.

Tabel 4.1 : Lingkungan implementasi

Perangkat	Spesifikasi
Perangkat Keras	Prosesor : Intel(R) Core(TM)2 Duo CPU @ 2.10 GHz 2.10 GHz  Memori : 2 GB
Perangkat Lunak	Sistem Operasi : Windows 7 Ultimate 32-bit  Perangkat Pengembang : Ms. Visio 2007 PHP Framework CodeIgniter XAMPP Notepad++

## 4.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak pada sistem ini, meliputi implementasi kode baik pada sisi user maupun administrator.

### 4.2.1 Operasi PHP pada CodeIgniter

Dalam membangun sistem pengusulan *E-Internship* ini, penulis memanfaatkan *framework* CodeIgniter sebagai penunjang bahasa pemrograman PHP. *Framework* CodeIgniter ini memiliki beberapa macam *library*, dan yang banyak digunakan dalam membangun tugas akhir ini adalah *database library*. Beberapa fungsi yang penulis gunakan pada *database library* ini adalah :

#### 4.2.1.1 Fungsi `$this→db→get()`

Sebuah fungsi yang digunakan untuk mengambil hasil dari query, salah satu contoh penggunaan fungsi ini adalah :

```
Public function getAllData($table)
{
    return $this→db→get($table);
}
```

Gambar 4.1

Pada gambar diatas, fungsi dari `getAllData($table)` digunakan untuk mengambil semua hasil *query* dari tabel.

#### 4.2.1.2 Fungsi `$this→db→where()`

Fungsi ini digunakan untuk melakukan penyaringan terhadap hasil dari *query* yang dijalankan, salah satu contoh implementasi dari fungsi ini adalah :

```
Public function getSelectedData($table,$key,$value)
{
    $this→db→where($key, $value);
    return $this→db→get($table);
}
```

Gambar 4.2

Pada gambar diatas menunjukkan bahwa fungsi tersebut digunakan untuk memanfaatkan sintaks *WHERE* pada MySQL, sehingga hasil *query* dari tabel dapat disaring sesuai dengan kebutuhan.

#### 4.2.1.3 Fungsi `$this→db→insert()`

Fungsi ini digunakan untuk memasukkan data ke dalam *database*, salah satu contoh implementasinya adalah :

```
function insertData($table,$data)
{
    $this->db->insert($table, $data);
}
```

Gambar 4.3

#### 4.2.1.4 Fungsi `$this->db->update()`

Sebuah fungsi yang digunakan untuk merubah data pada tabel yang sudah terdapat isi di dalamnya, salah satu contoh

```
function updateData($table,$data,$field,$key)
{
    $this->db->where($key, $field);
    $this->db->update($table, $data);
}
```

implementasinya adalah :

Gambar 4.4

Pada gambar diatas menunjukkan bahwa fungsi tersebut untuk mengganti isi dari sebuah tabel sesuai dengan *filter* yang dimasukkan ke dalam fungsi `$this->db->where()`.

#### 4.2.1.5 Fungsi `$this->db->delete()`

Sebuah fungsi yang digunakan untuk menghapus data suatu tabel di dalam *database*. Salah satu contoh implementasi fungsi tersebut adalah sebagai berikut

```
function deleteData($table,$data)
{
    $this->db->delete($table, $data);
}
```

Gambar 4.5

#### 4.2.1.6 Fungsi `$this->db->query()`

Sebuah fungsi yang digunakan untuk mengeksekusi *query* sesuai dengan kebutuhan sistem.

```
function cekNimMax($nim) {
    $q = $this->db->query("select * from tbl_mahasiswa where nim='".$.$nim."'");
    $hasil = 0;
    if($q->num_rows()>0)
    {
        $hasil = 1;
    }
    return $hasil;
}
```

Gambar 4.6

Pada gambar diatas menunjukkan fungsi *query* untuk *check* 'nim mahasiswa' yang sudah tersedia pada 'tbl\_mahasiswa'.

#### 4.2.2 Implementasi Fungsi *Sign-up*

Fungsi *sign-up* diperuntukkan bagi mahasiswa yang akan melakukan kerja praktek. Sebelum mendapatkan akun untuk *user login*, mahasiswa harus melakukan *sign-up* agar bisa *login* dan melakukan pengajuan izin kerja praktek.

```

public function signup_mahasiswa()
{
    $simpan["nim"] = $this->input->post("nim");
    $simpan["nama_mahasiswa"] = $this->input->post("nama_mahasiswa");
    $simpan["username"] = $this->input->post("username");
    $simpan["password"] = $this->input->post("password");
    $simpan["valid"] = "validasi";

    $simpan3["username"] = $this->input->post("nim");
    $simpan3["password"] = md5($this->input->post("nim"));
    $simpan3["ata"] = "mahasiswa";

    if($this->web_app_model->cekNimMax($simpan["nim"])==0)
    {
        $this->web_app_model->insertData('tbl_mahasiswa',$simpan);
        //$this->web_app_model->insertData('tbl_login',$simpan3);
        $this->session->set_flashdata("save_signup", "
        <p style='padding:10px; background-color:#7CFC00; text-align:center; margin:0px;'>
        Signup berhasil, tunggu validasi</p>");
        header('location:'.base_url().'web/tampilan_signup');
    }
    <script>
    window.parent.location.reload(true);
    </script>
<?php
}
else
{
    $this->session->set_flashdata("save_signup", "
    <p style='padding:10px; background-color:#0BE0F6; text-align:center; margin:0px;'>
    NIM Tidak Validasi</p>");
    header('location:'.base_url().'web/tampilan_signup');
}
}

```

Gambar 4.7 : fungsi *Sign-up*

Pada gambar diatas menunjukkan bahwa mahasiswa yang ingin mendapatkan *user login* harus melakukan *sign-up* terlebih dahulu serta menunggu validasi dari admin (koordinador kerja praktek).

### 4.2.3 Implementasi Fungsi *Login*

Fungsi login diperuntukkan bagi semua user (admin, mahasiswa, dan dosen).

```

public function index()
{
    $cek = $this->session->userdata('logged_in');
    if(empty($cek))
    {
        $d['judul'] = "Login - Sistem Monitoring dan Penilaian KP";

        //buat atribut form
        $frm['username'] = array('name' => 'username',
            'id' => 'username',
            'type' => 'text',
            'value' => '',
            'class' => 'input-teks-login',
            'placeholder' => 'Masukkan username.....'
        );
        $frm['password'] = array('name' => 'password',
            'id' => 'password',
            'type' => 'password',
            'value' => '',
            'class' => 'input-teks-login',
            'placeholder' => 'Masukkan password.....'
        );

        $this->load->view('global/bg_top',$d);
        $this->load->view('web/bg_login',$frm);
        $this->load->view('global/bg_footer',$d);
    }
}

```

Gambar 4.8 : Fungsi *login*

```

public function getLoginData($usr,$psw)
{
    $u = mysql_real_escape_string($usr);
    $p = md5(mysql_real_escape_string($psw));
    $q_cek_login = $this->db->get_where('tbl_login', array('username' => $u, 'password' => $p));
    if(count($q_cek_login->result())>0)
    {
        foreach($q_cek_login->result() as $qck)
        {
            if($qck->stts=='mahasiswa')
            {
                $q_ambil_data = $this->db->get_where('tbl_mahasiswa', array('nim' => $u));
                foreach($q_ambil_data ->result() as $qad)
                {
                    $sess_data['logged_in'] = 'yes';
                    $sess_data['nim'] = $qad->nim;
                    $sess_data['nama'] = $qad->nama_mahasiswa;
                    $sess_data['angkatan'] = $qad->angkatan;
                    $sess_data['jurusan'] = $qad->jurusan;
                    $sess_data['stts'] = 'mahasiswa';
                    $sess_data['program'] = $qad->kelas_program;
                    $this->session->set_userdata($sess_data);
                }
                header('location:'.base_url().'mahasiswa');
            }
        }
    }
}

```

Gambar 4.9 : *login* sebagai mahasiswa

```

else if($qck->stts=='dosen')
{
    $q_ambil_data = $this->db->get_where('tbl_dosen', array('kd_dosen' => $u));
    foreach($q_ambil_data ->result() as $qad)
    {
        $sess_data['logged_in'] = 'yes';
        $sess_data['kd_dosen'] = $qad->kd_dosen;
        $sess_data['nidn'] = $qad->nidn;
        $sess_data['nama'] = $qad->nama_dosen;
        $sess_data['stts'] = 'dosen';
        $this->session->set_userdata($sess_data);
    }
    header('location:'.base_url().'dosen');
}

```

Gambar 4.10 : *login* sebagai dosen

```

else if($qck->stts=='admin')
{
    $q_ambil_data = $this->db->get_where('tbl_admin', array('username' => $u));
    foreach($q_ambil_data ->result() as $qad)
    {
        $sess_data['logged_in'] = 'yes';
        $sess_data['username'] = $qad->username;
        $sess_data['nama'] = $qad->nama_lengkap;
        $sess_data['stts'] = 'admin';
        $this->session->set_userdata($sess_data);
    }
    header('location:'.base_url().'admin');
}

```

Gambar 4.11 : *login* sebagai admin

Pada gambar 4.8, 4.9, 4.10, dan 4.11 merupakan fungsi *login* dengan menggunakan *user* yang berbeda-beda.

#### 4.2.4 Administrator

Pada tugas akhir ini terdapat user admin yang memiliki hak akses untuk validasi mahasiswa yang melakukan *sign-up* untuk mendapatkan *user login*.

### 4.2.4.1 Implementasi Validasi User Mahasiswa

```

public function simpan_validasi()
{
    $cek = $this->session->userdata('logged_in');
    $stts = $this->session->userdata('stts');
    if(!empty($cek) && $stts=='admin')
    {
        $st = $this->input->post("stts");

        $simpan["nama_mahasiswa"] = $this->input->post("nama_mahasiswa");
        $simpan["angkatan"] = $this->input->post("angkatan");
        $simpan["jurusan"] = $this->input->post("jurusan");
        $simpan["valid"] = "aktif";

        $simpan3["username"] = $this->input->post("nim");
        $simpan3["password"] = md5($this->input->post("nim"));
        $simpan3["stts"] = "mahasiswa";

        if($st=="edit")
        {
            $nim = $this->input->post('nim');
            $where = array('nim'=>$nim);
            $this->web_app_model->updateDataMultiField("tbl_mahasiswa",$simpan,$where);
            $this->web_app_model->insertData('tbl_login',$simpan3);
            $this->session->set_flashdata("save_validasi","
                <p style='padding:10px; background-color:#7CFC00; text-align:center; margin:0px;'>
                    Akun Telah Diaktifkan</p>");
            header('location:'.$base_url().'admin/validasi_mahasiswa');
            ?>
                <script>
                    window.parent.location.reload(true);
                </script>
            <?php
        }
    }
    else
    {
        header('location:'.$base_url().'web');
    }
}

```

Gambar 4.12 : validasi user mahasiswa

Pada gambar diatas menunjukkan bahwa mahasiswa yang melakukan *sign-up* untuk mendapatkan *user login* memiliki status 'nonaktif'. Kemudian admin akan memvalidasi guna merubah status 'nonaktif' menjadi 'aktif' agar mahasiswa dapat melakukan *login*.

### 4.2.5 Modul Mahasiswa

Mahasiswa yang akan melakukan pengajuan izin kerja praktek diharuskan melakukan *sign-up* terlebih dahulu untuk

mendapatkan akun yang bisa digunakan untuk mengakses sistem monitoring KP. Setelah melakukan *sign-up* dan divalidasi oleh admin, mahasiswa bisa melakukan *login* untuk melakukan pengajuan izin kerja praktek.

Berikut merupakan fungsi implementasi pengajuan izin kerja praktek.

```

public function simpan_kp()
{
    $cek = $this->session->userdata('logged_in');
    $stts = $this->session->userdata('status');

    if(!empty($cek) && $stts=="mahasiswa")
    {
        $st = $this->input->post("status");

        $snrp = $this->session->userdata('nirp');
        $snrpcek = array('nirp'=>$snrp);
        $simpan = array('nirp'=>$snrp);
        $simpan["nama_kp"] = $this->input->post("nama_kp");
        $simpan["alamat_kp"] = $this->input->post("alamat_kp");
        $simpan["notelp_kp"] = $this->input->post("notelp_kp");

        if($this->web_app_model->cekTempatkp($simpan["nirp"])==0)
        {
            $this->web_app_model->insertData('tbl_tempatkp',$simpan);

            $this->session->set_flashdata("save_kp",
            <p style='padding:10px; background-color:#7FFF00; text-align:center; margin:0px;'>
            Selamat! Data telah tersimpan!</p>");
            header('location:'.base_url().'/submit_kp');
        }
        else
        {
            $this->session->set_flashdata("save_kp",
            <p style='padding:10px; background-color:#0E0F6; text-align:center; margin:0px;'>
            Data telah tersimpan!</p>");
            header('location:'.base_url().'/submit_kp');
        }
    }
    else
    {
        header('location:'.base_url().'/web');
    }
}

```

Gambar 4.13 : implementasi pengajuan izin kerja praktek

Pada gambar diatas menunjukkan bahwa mahasiswa yang memiliki akun, bisa melakukan pengajuan izin kerja praktek.

#### **4.2.5.1 Fungsi Detail KP Mahasiswa**

Fungsi ini merupakan fungsi untuk melihat detail kerja praktek yang akan dilakukan oleh mahasiswa. Didalamnya meliputi detail tempat kerja praktek, nama dosen pembimbing, nama penguji, serta nilai yang akan diperoleh setelah melakukan kerja praktek.

```

public function detail_kp()
{
    $cek = $this->session->userdata('logged_in');
    $stts = $this->session->userdata('stts');
    $nrp = $this->session->userdata('nrp');
    // $nim = array('nim'=>$nrp);

    if(!empty($cek) && $stts=='mahasiswa')
    {
        $d['judul'] = "Detail Mahasiswa KP - Sistem Monitoring dan Renilaian KP";

        $page=$this->uri->segment(3);
        $limit=5;
        if(!$page):
            $offset = 0;
        else:
            $offset = $page;
        endif;

        $bc['nama'] = $this->session->userdata('nama');
        $bc['status'] = $this->session->userdata('stts');
        $bc['nim'] = $this->session->userdata('nim');
        $bc['menu'] = $this->load->view('mahasiswa/menu', '', true);
        $bc['bic'] = $this->load->view('mahasiswa/bic', $bc, true);

        $bc['detailkp'] = $this->web_app_model->getDetailkp($nrp);
        $tot_hal = $this->web_app_model->getDetailkp($nrp);
        $config['base_url'] = base_url() . 'mahasiswa/detail_kp/';
        $config['total_rows'] = $tot_hal->num_rows();
        $config['per_page'] = $limit;
        $config['uri_segment'] = 3;
        $config['first_link'] = 'Awal';
        $config['last_link'] = 'Akhir';
        $config['next_link'] = 'Selanjutnya';
        $config['prev_link'] = 'Sebelumnya';
        $this->pagination->initialize($config);
        $bc["paginator"] = $this->pagination->create_links();

        $this->load->view('global/bg_top', $d);
        $this->load->view('mahasiswa/bg_detailkp', $bc);
        $this->load->view('global/bg_footer', $d);
    }
    else
    {
        header('location:'.base_url().'web');
    }
}

```

Gambar 4.14 : fungsi detail KP mahasiswa

Gambar diatas merupakan fungsi detail kerja praktek mahasiswa untuk melihat detail tempat kerja praktek, nama dosen pembimbing, nama penguji, serta nilai yang akan diperoleh setelah melakukan kerja praktek.

## BAB 5

### UJI COBA DAN EVALUASI

Pada bab ini akan dilakukan uji coba dan evaluasi perangkat lunak yang telah dibuat.

#### 5.1 Lingkungan uji coba

Lingkungan uji coba implementasi perangkat lunak yang digunakan dalam pembuatan tugas akhir ini meliputi *software* dan *hardware* dengan spesifikasi sebagai berikut

Tabel 5.1 : Lingkungan uji coba

Perangkat	Spesifikasi
Perangkat Keras	Prosesor : Intel(R) Core(TM)2 Duo CPU @ 2.40 GHz 2.40 GHz  Memori : 2 GB
Perangkat Lunak	Sistem Operasi : Windows 7 Ultimate 32-bit  Perangkat Pengembang : Mozilla Firefox XAMPP

## 5.2 Data Uji Coba

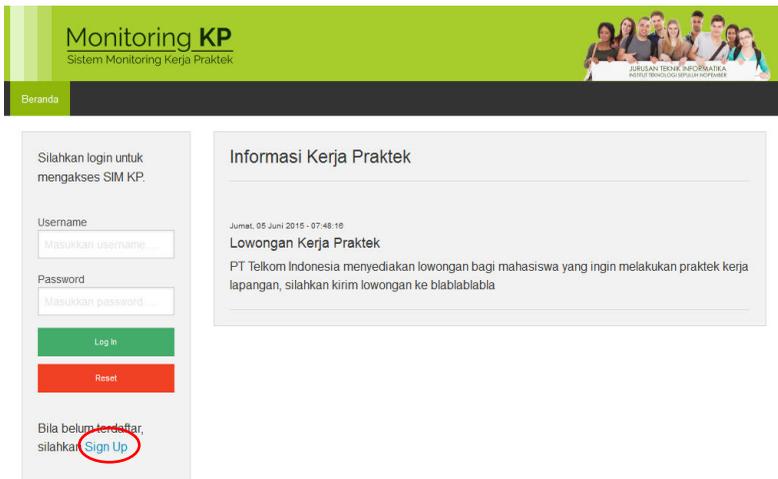
Data yang digunakan pada uji coba ini adalah data mahasiswa jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember yang akan melakukan kerja praktek.

## 5.3 Uji Coba Fungsionalitas

Pada uji coba fungsionalitas ini dilakukan untuk mengetahui apakah fitur – fitur yang ada pada sistem sudah berjalan dengan baik dan sesuai dengan tujuan sistem.

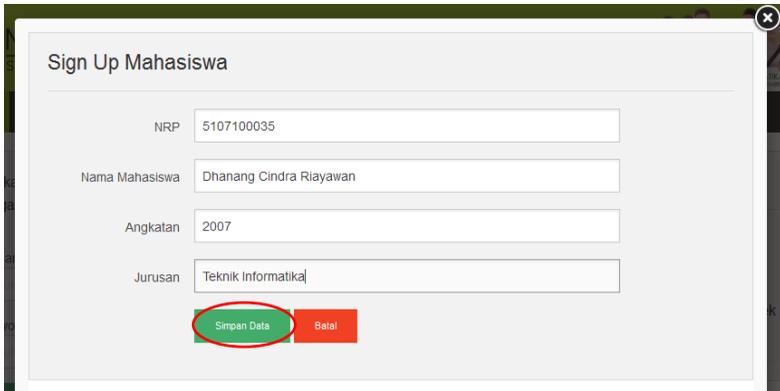
### 5.3.1 Fungsi *Sign-up*

Fungsi *sign-up* diperuntukkan bagi mahasiswa yang akan melakukan kerja praktek. Sebelum mendapatkan akun untuk *user login*, mahasiswa harus melakukan *sign-up* agar bisa *login* dan melakukan pengajuan izin kerja praktek.



Gambar 5.1 : Halaman awal SimKP

Dengan memilih menu *sign-up* seperti pada gambar yang dilingkar merah, maka user dapat melakukan *sign-up* seperti gambar dibawah berikut.



The screenshot shows a web form titled "Sign Up Mahasiswa". It contains four input fields: "NRP" with the value "5107100035", "Nama Mahasiswa" with "Dhanang Cindra Rlayawan", "Angkatan" with "2007", and "Jurusan" with "Teknik Informatika". Below the fields are two buttons: a green "Simpan Data" button and a red "Batal" button. The "Simpan Data" button is circled in red.

Gambar 5.2 : Halaman *Sign-up*

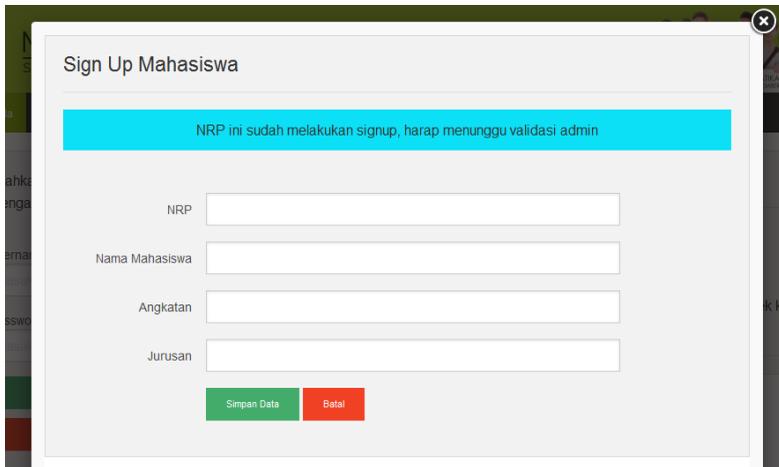
Pilih simpan data untuk melanjutkan proses *sign-up*. Apabila proses berhasil, maka akan muncul halaman seperti pada gambar dibawah ini.



The screenshot shows the same "Sign Up Mahasiswa" form, but with a green success message banner at the top that reads "Signup berhasil, tunggu validasi". The input fields and buttons are still visible below the banner.

Gambar 5.3 : Contoh *sign-up* berhasil

Apabila proses *sign-up* mengalami kegagalan, hal tersebut dikarenakan mahasiswa tersebut sudah melakukan *sign-up* dan harus menunggu validasi dari admin (koordinator kerja praktek).



The image shows a web form titled "Sign Up Mahasiswa". At the top, a blue message box states: "NRP ini sudah melakukan signup, harap menunggu validasi admin". Below this, there are four input fields: "NRP", "Nama Mahasiswa", "Angkatan", and "Jurusan". At the bottom of the form, there are two buttons: a green "Simpan Data" button and a red "Batal" button.

Gambar 5.4 : Contoh *sign-up* gagal

## 5.3.2 Fungsi *Login*

### 5.3.2.1 Admin

Admin memiliki hak akses yang lebih luas dibandingkan dengan hak akses pada user lain. Pada sisi admin bisa dilakukan login sebagai admin dan melakukan validasi mahasiswa yang melakukan proses *sign-up*.

**Monitoring KP**  
Sistem Monitoring Kerja Praktek

BERANDA DOSEN MAHASISWA **VALIDASI** INFO KP

Selamat Datang  
**Dhanang**

admin  
admin

Pengaturan Akun  
Keluar

**Beranda**

Selamat Datang di Sistem Monitoring dan Penilaian Kerja Praktek Teknik Informatika ITS. Setelah melakukan login pertama kali ke Sistem Perwalian Intranet Online ini, harap segera melakukan pergantian password pada menu Pengaturan Akun, guna menanggulangi hal-hal yang tidak diinginkan.

- **Pembagian dosen**  
Untuk Melakukan Pembagian Dosen Pembimbing dan Penguji
- **Penambahan mahasiswa**  
Untuk Menambahkan Mahasiswa KP
- **Info Kerja Praktek**  
Membagikan Info Lowongan Kerja Praktek
- **Log Out**  
Keluar dari sistem monitoring KP

Gambar 5.5 : Halaman *login* sebagai admin

Kemudian untuk melanjutkan ke proses validasi mahasiswa, pilih menu validasi seperti pada gambar diatas yang dilingkari merah.

**Monitoring KP**  
Sistem Monitoring Kerja Praktek

BERANDA DOSEN MAHASISWA **VALIDASI** INFO KP

Selamat Datang  
**Dhanang**

admin  
admin

Pengaturan Akun  
Keluar

**Validasi Mahasiswa**

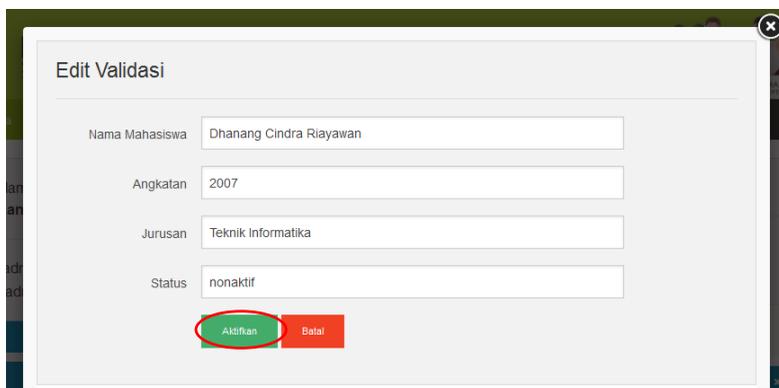
+ Tambah Mahasiswa

**Daftar Mahasiswa**

NRP	Nama Mahasiswa	Angkatan	Jurusan	Status	
5107100035	Dhanang Cindra Riyawan	2007	Teknik Informatika	nonaktif	<input checked="" type="checkbox"/> Aktifkan <input type="checkbox"/> Hapus

Gambar 5.6 : Validasi mahasiswa

Pilih 'aktifkan' dan kemudian akan muncul halaman untuk mengaktifkan status mahasiswa.



Edit Validasi

Nama Mahasiswa	Dhanang Cindra Riayawan
Angkatan	2007
Jurusan	Teknik Informatika
Status	nonaktif

Gambar 5.7 : Halaman validasi mahasiswa

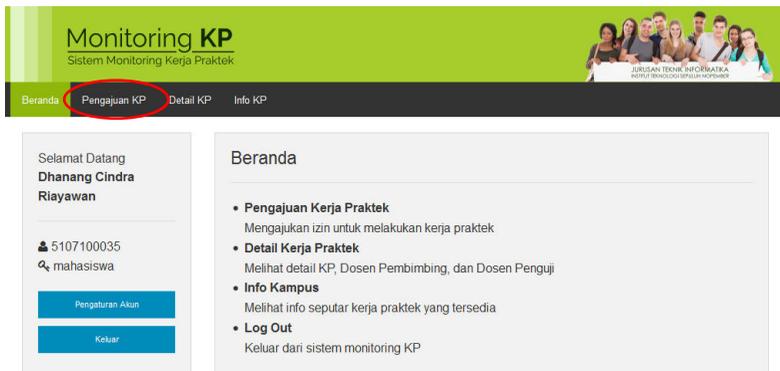
Setelah dipilih ‘aktifkan’ maka akan muncul halaman pemberitahuan seperti pada gambar dibawah ini.



Gambar 5.8 : Validasi mahasiswa berhasil

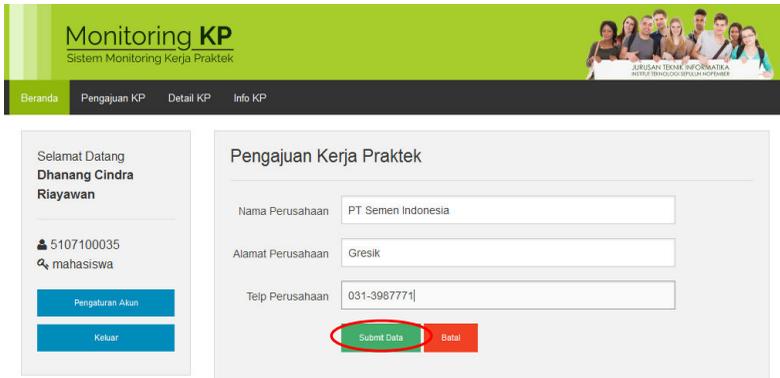
### 5.3.2.2 Mahasiswa

Mahasiswa yang sudah melakukan proses *sign-up* dan sudah divalidasi oleh admin, bisa melakukan proses *login* seperti pada gambar dibawah ini



Gambar 5.9 : halaman *login* sebagai mahasiswa

Setelah melakukan *login* mahasiswa bisa melanjutkan ke proses pengajuan izin kerja praktek dengan memilih menu 'Pengajuan KP'.



**Monitoring KP**  
Sistem Monitoring Kerja Praktek

Beranda Pengajuan KP Detail KP Info KP

Selamat Datang  
**Dhanang Cindra**  
Riayawan

5107100035  
mahasiswa

Pengaturan Akun  
Keluar

**Pengajuan Kerja Praktek**

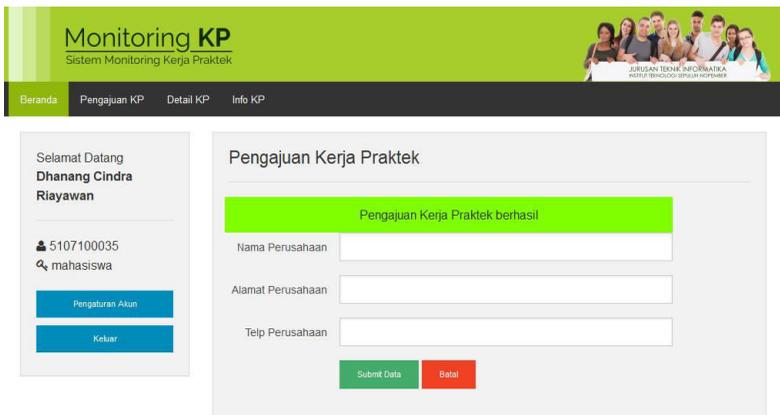
Nama Perusahaan

Alamat Perusahaan

Telp Perusahaan

Gambar 5.10 : halaman pengajuan kerja praktek

Kemudian pilih 'submit data' untuk melakukan pengajuan izin kerja praktek.



**Monitoring KP**  
Sistem Monitoring Kerja Praktek

Beranda Pengajuan KP Detail KP Info KP

Selamat Datang  
**Dhanang Cindra**  
Riayawan

5107100035  
mahasiswa

Pengaturan Akun  
Keluar

**Pengajuan Kerja Praktek**

Pengajuan Kerja Praktek berhasil

Nama Perusahaan

Alamat Perusahaan

Telp Perusahaan

Gambar 5.11 : Pengajuan kerja praktek berhasil

## **BAB 6**

### **KESIMPULAN DAN SARAN**

Pada bab ini dijelaskan mengenai kesimpulan akhir yang didapat setelah melakukan serangkaian uji coba pada bab sebelumnya. Dalam bab ini dijelaskan pula mengenai saran pengembangan tugas akhir lebih lanjut.

#### **6.1 Kesimpulan**

Pada proses pengembangan Sistem Monitoring dan Penilaian E-Internship ini dapat diambil beberapa kesimpulan sebagai berikut:

1. Rancang bangun sistem pengusulan *E-Internship* secara kolaboratif telah berhasil dibuat dan siap untuk diimplementasikan.
2. Diharapkan sistem pengusulan ini mampu membantu mahasiswa dalam mempermudah proses pengajuan izin kerja praktek.

#### **6.2 Saran**

Dalam pengembangan perancangan Sistem Monitoring dan Penilaian E-Internship, dapat diajukan beberapa saran, yaitu:

1. Perlu diadakan *survey* terhadap mahasiswa untuk penilaian sistem, agar sistem bisa berjalan sesuai dengan harapan, yaitu untuk mempermudah mahasiswa dalam melakukan pengajuan izin kerja praktek.

## DAFTAR PUSTAKA

- [1] Consentino, Christopher. 2002. *Advanced PHP for Web Profesional*. Penerbit Prentice Hall PTR.
- [2] Fahrur, Ahmad Haris. 2010. [PHP] Apa itu PHP. <http://komunitaskita.net/apa-itu-php/>. Diakses pada 20 mei 2015.
- [3] A. Philips, Jon; E. Davis, Michelle. 2007. *Learning PHP and MySQL, second edition*. Penerbit O'Reilly Media.
- [4] Wikipedia. 2015. Apa itu PHP, <URL: <http://id.wikipedia.org/wiki/PHP> diakses 20 April 2015.
- [5] SQL. 2015. SQL, <URL: <http://www.mssqlcity.com/> diakses 20 April 2015.

## BIODATA PENULIS



**Dhanang Cindra Riayawan**, atau lebih dikenal dengan Dhanang, Merupakan anak pertama dari dua bersaudara dari bapak Sutardjo dan ibu Ismi Hartini yang lahir di Gresik, 05 Mei 1989. Penulis menempuh pendidikan di TK Semen Gresik, SD II Semen Gresik, SLTPN 3 Gresik, SMA Muhammadiyah 1 Gresik hingga di Teknik Informatika ITS mulai tahun 2007.

Semenjak duduk di bangku kuliah, penulis mendapatkan banyak ilmu tentang bahasa pemrograman, jaringan dan tentang IT. Selama menjalani perkuliahan, penulis aktif menjadi anggota Himpunan Mahasiswa Teknik Computer (HMTTC). Semasa di Himpunan, penulis pernah menjadi staf departemen MPR (*Management Pengembangan Resource*) pada tahun 2008, dan sebagai OC OMB 2008.

Di awal masa kuliah, penulis merasa sulit untuk beradaptasi. Tapi dengan dukungan dari orang-orang terdekat menjadikan motivasi tersendiri bagi penulis untuk menjadi seseorang yang lebih baik. Selain itu, penulis juga gemar untuk belajar dan saling berbagi ilmu terutama di bidang jaringan, website dan aplikasi berbasis *open source*. Penulis dapat dihubungi melalui email [dhanang05@gmail.com](mailto:dhanang05@gmail.com).