

**LAPORAN KERJA PRAKTEK**  
**DI PT INTI BANDUNG**  
**ANALISIS DAN IMPLEMENTASI JAVASCRIPT,CSS DAN TEMPLATE**  
**UNTUK RESPONSIVITAS DAN TAMPILAN YANG MENARIK**  
**BERBASIS WEB DENGAN FRAMEWORK CODEIGNITER**

*Diajukan untuk memenuhi Syarat Mata Kuliah Kerja Praktek*

*Program Sarjana Jurusan Teknik Informatika*

*Sekolah Tinggi Teknologi Bandung*

**Disusun Oleh:**

**Nama: Encep Sania**

**NIM: 11111 049**



**SEKOLAH TINGGI TEKNOLOGI BANDUNG PROGRAM STUDY**  
**TEKNIK INFORMATIKA**

**2015**

**LEMBAR PENGESAHAN KAMPUS**  
**LAPORAN KERJA PRAKTEK**  
**PENGUNAAN JAVASCRIPT,CSS DAN TEMPLATE UNTUK**  
**RESPONSIVITAS DAN TAMPILAN YANG MENARIK BERBASIS *WEB***  
**DENGAN FRAMEWORK CODEIGNITER**

Disusun Oleh :

**ENCEP SANIA**

**11111049**

Telah Disahkan Oleh Dosen Pembimbing Kerja Praktek

Pada Tanggal.....

Susunan Dosen Pembimbing

Pembimbing 1

Pembimbing 2

Dosen Pembimbing Kerja Praktek

Staff Sistikfo PT INTI

Jurusan Teknik Informatika

Ir.Iwan Abadi, S.T.,MM.

Anissa Dian P

Laporan kerja praktek ini telah diterima sebagai salah satu persyaratan

Untuk memperoleh gelar sarjana pada jurusan Teknik Informatika

Bandung, Januari 2015

Mengetahui

Ketua Jurusan Teknik Informatika Sekolah Tinggi Teknologi Bandung

Indra Julias Pradana, S.Kom.

## ABSTRAK

Perkembangan dunia digital semakin hari semakin cepat dan dinamis. Tidak terbatas ruang dan waktu,kapanpun dan dimanapun.Salah satu factor pendukung semua itu adalah karena adanya interface berbasis *web*.

Dengan adanya *web* ini,semua pengguna internet dapat melakukan berbagai macam aktivitas dan tugas nya dengan nyaman tak masalah dengan subjek/client ataupun destinasi nya jauh dari tempat ia berada.

Suatu *web* yang baik adalah *web* yang memiliki kecepatan (*speed*), kekuatan (*strength*) dan *user interface* (*ui*) yang menarik.untuk membuat pengguna nyaman dan tenang. Untuk menjawab semua itu maka kini banyak bermunculan javascript ataupun cms dan css dll.

Penggunaan javascript,css dan lain sebagainya ditujukan untuk membuat *web* menjadi lebih hidup. Selain itu banyak pula plugin yg digunakan dalam bahasa pemrograman *web* yang sangat berguna untuk proses development suatu *web*.

Untuk memahami serta mendalami tentang pemrograman *web* dan aplikasi pendukungnya,maka penulis membuat judul **“PENGUNAAN JAVASCRIPT,CSS DAN TEMPLATE UNTUK RESPONSIVITAS DAN TAMPILAN YANG MENARIK BERBASIS WEB DENGAN FRAMEWORK CODEIGNITER”**.

Kata kunci: *web*,javascript,codeigniter.

## **ABSTRACT**

*The development of the digital world is getting fast and dynamic. Boundless space and time, everywhere and anywhere. It's because of supported by web-based interface.*

*With this web, all internet users can do various kind of activities and task with comfortably and cozy and there's no problem how far the subject/client destinations was.*

*A good web is having high speed, strenght and awesome interface. To answer all of it, today there's many open source vendors created javascript, css or cms and etc.*

*The used of javascript, css and many more aimed to make a web be more alive and attractive. In addition also many web plugins used in a very useful to process and development of a web.*

*To understand and explore web programming and support plugins, writer makeing the title “**THE USED OF JAVASCRIPT, CSS AND TEMPALTE TO BUILD RESPONSIVE WEB AND WONDERFULL INTERFACE USING FRAMEWORK CODEIGNITER**”.*

*Keywords: web, javascript, codeigniter.*

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT atas berkah dan karunia-Nya lah penulis dapat menyelesaikan laporan kerja praktek study kasus di PT INTI dengan judul **“PENGUNAAN JAVASCRIPT,CSS DAN TEMPLATE UNTUK RESPONSIVITAS DAN TAMPILAN YANG MENARIK”**.

Dalam pembahasan laporan kerja praktek ini,banyak contoh penggunaan *javascript,templatedan* lain-lain beserta implementasinya dalam pembangunan suatu *web* sehingga menghasilkan tampilan *web* yang responsive,menarik bagi user dan nyaman digunakan.

Laporan kerja praktek ini berdasarkan study kasus dan tugas yang diberikan oleh pembimbing di PT INTI divisi SISTEKFO sub divisi *WEB DEVELOPMENT* yang banyak memberikan imlu yang bermanfaat dalam pemrograman *web*. Disini akan dibahas kegunaan,cara penggunaan dan optimasi javascript,css dan template dalam suatu pembangunan *web*.

Atas selesainya penulisan laporan kerja praktek ini, penulis mengucapkan banyak terima kasih kepada :

1. Bapak Ragil Pardiyono, S.T.,M.T, selaku ketua Sekolah Tinggi Teknologi Bandung (STT-Bandung).
2. Bapak Agus Rahmat Hermawanto, S.H, selaku wali kelas teknik informatika angkatan 2011 di Sekolah Tinggi Teknologi Bandung (STT-BANDUNG)

3. Bapak Ir Iwan Abadi,S.T.,M.M, sebagai dosen pembimbing dalam penulisan laporan kerja praktek ini.
4. Ibu Anissa Dian beserta seluruh pimpinan dan staff divisi Sistekfo PT INTI atas bantuan dan kerjasamanya dalam pembuatan laporan kerja praktek.
5. Seluruh jajaran dosen dan staff karyawan Sekolah Tinggi Teknologi Bandung (STT-Bandung), yang telah membantu kelancaran dalam penyusunan laporan kerja praktek ini.
6. Seluruh teman-teman jurusan Teknik Informatika Sekolah Tinggi Teknologi Bandung (STT-Bandung), khususnya angkatan 2011 yang setia menemani dalam suka dan duka.

Laporan ini masih banyak terdapat kekurangan karena keterbatasan waktu dan ilmu yang dimiliki penulis.maka kritik dan saran yang membangun sangat penulis harapkan untuk perbaikan dimasa mendatang.

Akhir kata, penulis mengucapkan terimakasih banyak atas dukungan dari semua pihak yang terlibat aktif dalam penyelesaian laporan ini.

Bandung,18 Januari 2015

Encep Sania

## DAFTAR ISI

### LEMBAR PENGESAHAN KAMPUS

<b>ABSTRAK .....</b>	<b>i</b>
<b>ABSTRACT .....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>I-1</b>
1.1 Latar Belakang .....	1
1.2 Batasan Masalah .....	2
1.3 Tujuan .....	3
1.4 Metode Penelitian .....	3
1.4.1 Observasi.....	3
1.4.1 Wawancara.....	3
1.5 Sistematika Penulisan .....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>6</b>
2.1 Pengertian Internet .....	6
2.2 Pengertian Web .....	6

2.3 Pengertian Basis Data .....	6
2.4 Pengertian Web Database .....	7
2.5 Pengertian HTML .....	8
2.6Pengertian PHP .....	9
2.7Pengertian CSS .....	9
2.8Pengertian URL.....	9
2.9Pengertian Javascript.....	10
2.10 Pengertian MySQL .....	10
2.11Pengertian CodeIgniter.....	11
2.12Pengertian JSON .....	11
2.13PengertianAjax .....	12
2.14PengertianMVC .....	12
2.15PengertianJquery .....	12
2.16PengertianTemplate.....	13
2.17Pengertian Responsive .....	13
2.18Pengertian Highchart.....	13
2.19Pengertian Getorgchart.....	14
2.20Metodologi .....	14



2.20.1 Konsep Dasar Objek .....	14
2.20.2 Unified Modeling Language (UML).....	15
2.20.2Activity Diagram .....	17
2.20.3Sequence Diagram .....	18
2.20.4Class Diagram .....	19
2.20.5Collaboration Diagram.....	19
<b>BAB III ANALISIS .....</b>	<b>20</b>
3.1 Analisis.....	20
3.2 Gambaran Aplikasi.....	20
3.3 Identifikasi Pengguna.....	21
3.4 Analisa Kebutuhan Fungsional .....	21
3.5 UseCase dan Skenario.....	23
3.5.1Use Case.....	23
3.5.2Skenario .....	24
3.5.2.1Skenario Otorisasi .....	24
3.5.2.2Skenario Pengolahan Data File .....	24
3.5.2.3Skenario Data Presensi .....	25
3.5.2.4Skenario Data Pejabat Organisasi .....	25
3.5.2.5Skenario Pengolahan Data User.....	26
3.5.2.6Skenario Laporan Presensi.....	26
3.5.2.7Skenario Laporan Struktur Organisasi.....	27
3.5.2.8Skenario Laporan File .....	27

<b>BAB IV PERANCANGAN.....</b>	<b>28</b>
4.1 Activity Diagram.....	28
4.1.1Activity Diagram Otorisasi .....	28
4.1.2Activity Diagram Pengolahan Data File .....	29
4.1.3Activity Diagram Pengolahan Data Presensi .....	30
4.1.4Activity Diagram Data User .....	30
4.1.4.1Activity Diagram Tambah User .....	31
4.1.4.2Activity Diagram Ubah User .....	32
4.1.4.3Activity Diagram Hapus User.....	33
4.1.5Activity Diagram Laporan Presensi .....	34
4.1.6Activity Diagram Laporan Struktur Organisasi .....	35
4.1.7Activity Diagram Laporan Data User .....	36
4.2 Sequence Diagram .....	37
4.2.1Sequence Diagram Otorisasi .....	37
4.2.2Sequence Diagram Pengolahan Data Presensi.....	38
4.2.3Sequence Diagram Pengolahan Data User.....	39
4.2.3.1Sequence Diagram Tambah User.....	39
4.2.3.2Sequence Diagram Ubah User .....	40
4.2.3.3Sequence Diagram Hapus User.....	41
4.2.4Sequence Diagram Pengolahan Data File Upload .....	42

4.2.5Sequence Diagram Laporan Presensi.....	43
4.2.6 Sequence Diagram Laporan Upload File .....	44
4.2.7Sequence Diagram Laporan Struktur Organisasi.....	45
4.2.8Class Diagram .....	46
4.3Antar Muka .....	47
4.3.1Form Login .....	47
4.3.2Hak Akses .....	48
4.3.3Form Menu Utama .....	49
4.3.4Menu User.....	50
4.3.5Data Role .....	51
4.3.6Data Menu.....	52
4.3.7User Role .....	53
4.3.8Menu Upload .....	54
4.3.9Menu Show File .....	55
4.3.10Menu Absensi .....	56
4.3.11Menu Struktur Organisasi .....	57
<b>BABVPENUTUP .....</b>	<b>58</b>
5.1Kesimpulan .....	58
5.2Saran .....	59
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

Gambar2.1 Aktor.....	16
Gambar2.2 Use Case.....	16
Gambar2.3 Relationship.....	16
Gambar2.4 Symbol Activity Diagram .....	17
Gambar2.5 Sequence Diagram.....	18
Gambar2.6 Class Diagram .....	19
Gambar4.1 Activity Diagram Otorisasi .....	28
Gambar4.2 Activity Diagram Pengolahan Data File .....	29
Gambar4.3 Activity Diagram Pengolahan Data Presensi .....	30
Gambar4.4 Activity Data Tambah User.....	31
Gambar4.5 Activity Ubah User.....	32
Gambar4.6 Activity Hapus User .....	33
Gambar4.7 Activity Laporan Presensi .....	34
Gambar4.8 ActivityLaporan Struktur Organisasi .....	35
Gambar4.9 Activity Diagram Data User.....	36
Gambar4.10 Sequence DiagramOtorisasi .....	37
Gambar4.11 Sequence DiagramPengolahan Data Presensi .....	38

Gambar4.12 Sequence Diagram Tambah User .....	39
Gambar4.13 Sequence Diagram Ubah Data .....	40
Gambar4.14 Sequence Diagram Hapus User .....	41
Gambar4.15 Sequence Diagram Upload File .....	42
Gambar4.16 Sequence Diagram Laporan Presensi .....	43
Gambar4.17Sequence DiagramLaporan Upload File .....	44
Gambar4.18 Sequence DiagramLaporan Struktur Organisasi .....	45
Gambar4.19 ClassDiagram .....	46
Gambar 4.20 Form Login.....	47
Gambar4.21 Form MenuUtama .....	49
Gambar4.22 FormUserPage .....	50
Gambar4.23 FormRole Page .....	51
Gambar4.24 Form DataMenu .....	52
Gambar4.25 FormUser Role .....	53
Gambar4.26 Form Upload Data .....	54
Gambar4.27 Form Show File .....	55
Gambar4.28 Form Absensi .....	56
Gambar4.29 Form OrgChart .....	57

## DAFTAR TABEL

Tabel 3.2 Skenario Otorisasi .....	24
Tabel 3.3 Skenario Pengolahan Data File .....	24
Tabel 3.4 Skenario Pengolahan Data Presensi .....	25
Tabel 3.5 Skenario Pengolahan Data Pejabat Organisasi .....	25
Tabel 3.6 Skenario Pengolahan Data User .....	26
Tabel 3.7 Skenario Laporan Presensi .....	26
Tabel 3.8 Skenario Laporan File .....	27
Tabel 3.9 TabelSimpanan .....	54
Tabel 3.10 TabelPinjaman .....	55
Tabel 3.11 TabelAnggsuran .....	55
Tabel 3.12 Tabel User .....	56

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengertian Internet**

Internet “induk dari semua jaringan”: internet adalah jantung era informasi. Disebut “induk dari semua jaringan” karena internet ( “net” atau “jaringan” ) adalah jaringan computer diseluruh dunia yang menghubungkan ratusan bahkan ribuan jaringan yang lebih kecil, misalnya jaringan pendidikan, komersial, nirlaba, dan militer, bahkan jaringan individual. Williams dan Sawyer (2007:17)

#### **2.2 Pengertian Web**

*Website* ( situs *web* ) adalah merupakan alamat ( *URL* ) yang berfungsi sebagai tempat penyimpanan data dan informasi dengan berdasarkan topik tertentu. Sutarman ( 2003:6 )

*Web Page* ( halaman *web* ) merupakan halaman khusus dari situs *web* tertentu yang tersimpan dalam bentuk file. Dalam *web page* tersimpan berbagai informasi dan *link* yang menghubungkan suatu informasi ke informasi lain baik itu dalam page yang sama ataupun *web page* lain pada *website* yang berbeda.

#### **2.3 Pengertian Basis Data**

Basis Data terdiri atas dua kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, sedangkan data representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan,

peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bentuk angka, huruf, teks, simbol, gambar, bunyi dan kombinasinya.

Menurut Fathansyah (2004:2), basis data sendiri dapat didefinisikan dalam beberapa sudut pandang, seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang saling disimpan secara bersama sedemikian rupa tanpa pengulangan (redudansi) yang tidak perlu untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

## **2.4 Pengertian *Web Database***

*Web database* merupakan sistem penyimpanan data yang dapat diakses oleh bahasa pemrograman tertentu. *Web database* dapat diakses oleh aplikasi-aplikasi *web* yang dikembangkan dengan HTML tag, kontrol *active X* dan pemrograman yang bersifat *server side* melalui CGI, *Microsoft IIS* ( *Internet Information Server* ).

*Web* atau disebut sebagai *World Wide Web* ( *WWW* ) adalah fasilitas yang menyediakan *interface* yang saling keterkaitan dan berinteraksi antar jaringan



yang sederhana ke sumber daya internet yang sangat besar. *World Wide Web* ( WWW ) bukan sekedar jaringan tetapi didalamnya terdapat suatu sel aplikasi komunikasi dan sistem perangkat lunak yang memiliki karakteristik tertentu diantaranya yaitu :

1. Umumnya tertetak pada *internet host* dan *client*
2. Umumnya menggunakan *protocol* TCP/IP
3. Mengerti HTML
4. Mengikuti model *client/server* untuk komunikasi data dua arah
5. Memungkinkan client untuk mengakses *server* dengan *protocol* seperti HTTP, FTP, Telne, dan Gopher
6. Memungkinkan *client* untuk mengakses informasi dalam berbagai media seperti teks, audio, video
7. Menggunakan model alamat *Unifor Resource Locators* ( URL )

## **2.5 Pengertian HTML**

*Hypertext Markup Language* (HTML) adalah bahasa yang digunakan untuk menulis halaman *web*. *HTML* merupakan pengembangan dari standart pemformatan dokumen text yaitu *Standart Generalized Markup Language* (SGML). HTML sebenarnya adalah dokumen atau text bisaa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu.

Perintah-perintah *HTML* diletakkan dalam file berekstensi \*.html dan ditandai dengan mempergunakan tag ( tanda ) berupa karakter “ < “ dan “ > “. Tidak seperti bahasa pemrograman berstruktur seperti *pascal* atau *C*, *HTML* tidak mengenal *jumping* atau *looping*. Kode-kode *HTML* dibaca oleh *browser* dari atas kebawah tanpa adanya lompatan-lompatan.

## **2.6 Pengertian PHP**

*Hypertext Preprocessor (PHP)* 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*.

## **2.7 Pengertian CSS**

*Cascading Style Sheet (CSS)* merupakan salah satu bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam. Sama halnya *styles* dalam aplikasi pengolahan kata seperti *Microsoft Word* yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images* dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa file. Pada umumnya *CSS* dipakai untuk memformat tampilan halaman *web* yang dibuat dengan bahasa *HTML* dan *XHTML*.

## **2.8 Pengertian URL**

*URL* merupakan system pengalamatan yang digunakan pada *World Wide Web*. Di internet *URL* menggabungkan informasi mengenai jenis protokol yang digunakan alamat situs dimana *resource* ditempatkan, lokasi *sub directory* dan nama file yang digunakan

*Syntax* lengkap suatu *URL*

*Access-method* ://*server\_name* *I*:[*port*]/*directory*/***file***

Contoh:

*https://datatables.net/examples/index*

## **2.9 Javascript**

*Javascript* adalah bahasa pemrograman yang sederhana karena bahasa ini tidak dapat digunakan untuk membuat aplikasi ataupun *applet*. Dengan *javascript*, kita dapat dengan mudah membuat sebuah halaman *web* yang interaktif (Hardjono, 2006, p4)

Sedangkan menurut Ellsworth dan Matthew (1997, p179), *Javascript* adalah pendekatan lain untuk membuat hal *web* menjadi lebih interaktif, baik dalam deteksi maupun tanggapan ke interaksi pengguna dengan halaman *web*. *Javascript* dapat langsung digabungkan dengan *HTML* tanpa harus *decompile* terlebih dahulu.

## **2.10 Pengertian MySQL**

Menurut Hardjono (2006, p181) *DatabaseMySQL* merupakan sistem manajemen berbasis data *SQL* yang sangat terkenal dan bersifat *open source*. *MySQL* dibangun, didistribusikan dan didukung oleh *MySQL AB*. *MySQL AB*

merupakan perusahaan komersial yang dibiayai oleh pengembang (*developer*) *MySQL*. *MySQL* dapat didefinisikan sebagai :

*MySQL* merupakan sistem manajemen *database*. *Database* merupakan struktur penyimpanan data. Untuk menambah, mengakses, dan memproses data yang disimpan dalam sebuah *database* komputer. Diperlukan sistem manajemen *database* seperti *MySQL Server*.

## **2.11 Pengertian CodeIgniter**

Menurut Blanco & Upton (2009:7) *CodeIgniter* adalah *powerful open source PHP framework* yang mudah dikuasai, dibangun untuk *PHP programmers* yang membutuhkan *toolkit* sederhana dan baik untuk membuat *full-featured web applications*. *CodeIgniter* adalah *MVC framework* yang di design untuk mempermudah penggunaanya.

## **2.12 Pengertian Javascript Object Notation(JSON)**

*JSON* adalah sebuah format data berbasis teks yang ringan dan dirancang agar mudah dibaca dan ditulis oleh manusia, ataupun oleh mesin. *JSON* merupakan format teks yang sepenuhnya *independent language* tetapi menggunakan konvensi yang akrab bagi *programmer* dari bahasa *C*, *C++*, *C#*, *Java*, *JavaScript*, *Pearl*, *Phyton*, dan lainnya yang membuat *JSON* menjadi bahasa pertukaran data yang ideal.

(Anonymous.n.d.Introducing  
<http://www.json.org> )

JSON.Retrieved:11-10-2012from

### 2.13 Pengertian Ajax

Menurut Inc (2008:7) Ajax, Asynchronous JavaScript and XM adalah tehnik baru yang mendeskripsikan bagaimana teknologi, JavaScript, DOM (Document Object Model) dan XML bisa digunakan bersamaan untuk membuat *web* aplikasi yang interaktif. Ajax menggunakan asynchronus data transfer antara browsers dan *web* servers, mengizinkan halaman *web* untuk meminta sedikit informasi dari *web* server dari seluruh halaman.

### 2.14 Pengertian MVC

Menurut Mackey (2010:295) *MVC* atau *Model View Controller* memiliki arti sebagai berikut:

- *Model* merupakan *database*.
- *View* merupakan *pages* dan *controls*.
- *Controller* akan mengatur interaksi antara *pages*/ mengontrol *view* dan *model*.

### 2.15 Pengertian JQuery

*jQuery* adalah sebuah *library Javascript* yang sangat ringkas dan sederhana untuk memanipulasi komponen di dokumen *HTML*, menangani *event*, animasi, efek dan memproses interaksi *ajax*. *jQuery* dirancang sedemikian rupa supaya membuat *program* menggunakan *Javascript* menjadi relatif sangat mudah. Sesuai slogan nya, *write less, do more*. Menulis kode lebih sedikit, tetapi melakukan pekerjaan lebih banyak.

*jQuery* ukuran nya cukup kecil, sehingga tidak memperlambat proses loading halaman *web* yang kita buat. *jQuery* juga kompatibel dengan *CSS3* dan yang tak kalah penting adalah *jQuery* bisa berjalan di semua *browser*.

## **2.16 Pengertian Template**

Menurut Tom E Rolnicki, pengantar dasar jurnalisme 2008 prenatal media grup rawangmangu jakarta dialihbahasakan oleh Tri Wibowo, *template* adalah struktur kerangka halaman yang disimpan secara elektronik di program *layout*; dapat memuat *style sheet*.

*Template* merupakan struktur bentuk awal yang dapat digunakan untuk membuat beberapa tipe *Web* dan halaman-halaman *Web*. Pada saat *template* berbentuk struktur, biasanya merupakan suatu bentuk struktur “kosong”, yang berarti bahwa *template* hanya merupakan suatu kerangka atau *framework* yang digunakan untuk membentuk dan menyempurnakan suatu produk.

## **2.17 Pengertian Responsive**

adalah jenis *website* yang dapat mengakomodasi berbagai ukuran lebar layar pengaksesnya. Sederhananya *website responsive* adalah jenis *website* yang enak dibuka di berbagai jenis perangkat.

## **2.18 Pengertian Highchart**

*Highchart* adalah *charting library* yang ditulis dengan bahasa *javascript* murni. *Highchart* menawarkan intuitif, grafik interaktif ke situs *web* atau aplikasi *web*. Saat ini *highchart* mendukung grafik tipe *line*, *spline*, *area*, *areaspline*, *column*, *bar*, *pie*, dan *scatter*. *Highchart* sudah dapat berjalan di

semua *browser* modern termasuk pada *iPhone/iPad*. *Browser* standar menggunakan *SVG* untuk proses render grafik, sedangkan *internet explorer* menggunakan *VML* untuk penggambaran grafik.

## 2.19 Pengertian Getorgchart

*GetOrgChart* adalah library javascript yang menyajikan bagan atau struktur organisasi secara *intuitif* dan interaktif. Selain itu, *GetOrgChart* juga menyediakan cara terbaik untuk memvisualisasikan dan memahami tenaga kerja serta alur jabatan secara simple dan mudah untuk menggambarkan hubungan hirarkis.

## 2.20 Metodologi

Metodologi digunakan untuk mendeskripsikan sistem yang telah maupun yang akan dirancang dengan metode pengembangan tertentu. Dalam analisis dan perancangan ini digunakan metodologi yang berorientasi objek ( *Object – Oriented Analysis and Design* ).

### 2.20.1 Konsep dasar objek

Konsep *object oriented* dibangun atas beberapa prinsip dasar. Objek adalah contoh atau *instance* dari sebuah *class*. Beberapa objek mempunyai *attribute* dan *operation* yang sama akan membentuk *class*.

*Inheritance*, *polymorphisme*, dan *encapsulation* adalah prinsip – prinsip dasar pada metode *object oriented*. *Inheritance* lebih berorientasi ke penurunan

sifat, *polymorphisme* lebih menekankan ke penggunaan terminologi operasi yang sejenis. Sedangkan *encapsulation* lebih menekankan ke penyembunyian informasi untuk menyederhanakan operasi kepada objek lainnya.

## **2.20.2 Unified Modeling Language ( UML )**

Unified Modeling Language adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Munawar ( 2005 : 17 ).

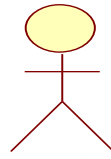
UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML antara lain :

### **1. Diagram Use case ( Use case diagram )**

Diagram use case adalah teknik untuk merekam persyaratan fungsional sebuah sistem. Use case mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sebuah sistem tersebut digunakan. Dalam bahasan use case, para pengguna disebut dengan aktor. Aktor adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem. Use case adalah alat bantu terbaik guna menstimulasi pengguna potensial untuk mengatakan tentang suatu sistem dari sudut pandangnya. Diagram use case mempunyai 3 notasi yang menunjukkan aspek dari sistem ( Munawar, 2005 ) :

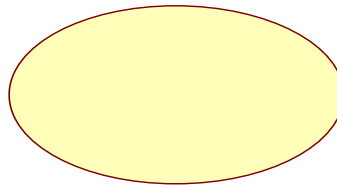
- a. Actor ( pengguna ) yaitu abstraksi dari orang dan sistem lain yang mengaktifkan fungsi dari target sistem.





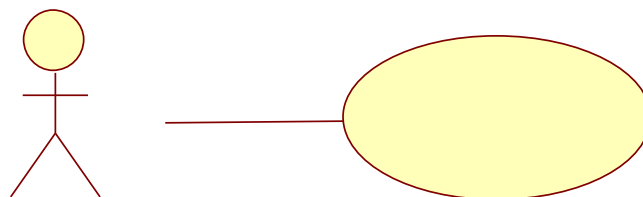
**Gambar 2.1 Aktor**

- b. Use case adalah abstraksi dari interaksi antara sistem dan actor. Use case dibuat berdasarkan keperluan actor. Use case harus merupakan “ apa “ yang dikerjakan software aplikasi, bukan “ bagaimana “ software aplikasi mengerjakannya. Setiap Use case harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksinya dengan actor.



**Gambar 2.2 Use Case**




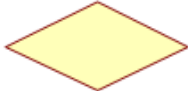

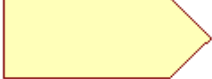
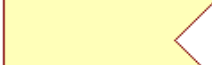

- c. Relationship ( hubungan ) yaitu hubungan antara actor dengan use case dimana terjadi interaksi diantara mereka.



**Gambar 2.3 Relationship**

## 2. Diagram Aktivitas ( Activity diagram )

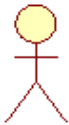

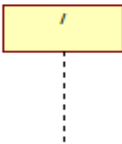

Diagram activity menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing – masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih.

Simbol	Keterangan
	Titik awal
	Titik akhir
	Activity
	Pilihan untuk pengambilan keputusan
	Fork, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu
	Tanda pengiriman
	Tanda penerimaan
	Aliran akhir ( Flow Final )

**Gambar 2.4 Simbol – simbol yang sering digunakan pada activity diagram**

### 3. Diagram Sekuensial ( Sequence diagram )


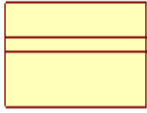



Diagram sekuen menggambarkan interaksi antar objek di dalam dan disekitar sistem ( termasuk pengguna, display dsb ) berupa message yang digambarkan terhadap waktu. Diagram sekuen terdiri dari dimensi vertikal ( waktu ) dan dimensi horizontal ( objek – objek yang terkait ).

Simbol	Nama	Keterangan
	Actor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi
	Time active	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
	Life line	Menyatakan kehidupan suatu objek
	Message	Spesifikasi dari komunikasi antar objek yang memuat informasi - informasi tentang aktifitas yang terjadi

**Gambar 2.5 Simbol – simbol sequence diagram yang sering digunakan**

#### 4. Diagram Kelas ( Class diagram )

Diagram ini memperlihatkan interaksi antar kelas. Diagram ini umum dijumpai pada permodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas – kelas aktif.

Gambar	Nama	Keterangan
	Generalization	Hubungan dimana objek anak ( descendent ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( ancestor )
	Class	Himpunan dari objek - objek yang berbagi atribut serta operasi yang sama
	Realization	Operasi yang benar - benar dilakukan oleh suatu objek
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( independent ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	Association	Apa yang menghubungkan antara objek satu dengan objek yang lainnya

**Gambar 2.6 Simbol – simbol class diagram yang sering digunakan**

#### 5. Collaboration Diagram

Collaboration diagram adalah perluasan dari objek diagram. Objek diagram menunjukkan objek – objek yang huungannya satu dengan yang lain.

Collaboration diagram menunjukkan message – message objek yang dikirim satu sama lain.

## **BAB III**

### **ANALISIS**

#### **3.1 Analisis**

Setelah melakukan penelitian dan observasi di divisi sistekfo PT INTI, dibutuhkan metode otorisasi yang mengatur karyawan yang berhak dan tidaknya mengakses suatu informasi. Selain itu, juga dibutuhkan sistem dinamis yang dapat menyimpan data banyak dan menampilkannya dengan cepat. Dan hal yang tidak kalah pentingnya adalah tampilan yang menarik dan dapat dengan mudah dipahami hanya dengan melihatnya saja sebagai laporan. Permasalahan yang timbul diantaranya:

1. Tidak adanya otorisasi dalam manajemen hak akses suatu informasi.
2. Manajemen upload file yang kurang baik, sehingga sulit melakukan pencarian data yang dibutuhkan.
3. Pengolahan data yang kompleks.
4. Tampilan yang kurang menarik.

#### **3.2 Gambarab Aplikasi**

Web yang dirancang ini berbasiskan framework CodeIgniter supaya mudah dalam development dan pengembangannya. Selain itu juga diterapkannya template dan javascript membuat tampilan lebih menarik dan data laporan yang diterima dapat dengan mudah dan cepat dipahami, serta penggunaan database yang dapat menampung banyak data sekaligus untuk ketepatan dan kecepatan akses data.

### 3.3 Identifikasi Pengguna

Pengguna sistem dikategorikan menjadi 4, yaitu admin, adm.IT\_helpdesk dan manajer. Setiap kategori pengguna memiliki hak akses yang berbeda pada setiap fitur sistem. Setiap pengguna sistem memiliki identitas pengenalan dan *password* yang unik dan disimpan dalam basis data. Identitas pengenalan dan *password* digunakan pada saat akan memasuki sistem.

### 3.4 Analisis Kebutuhan Fungsional

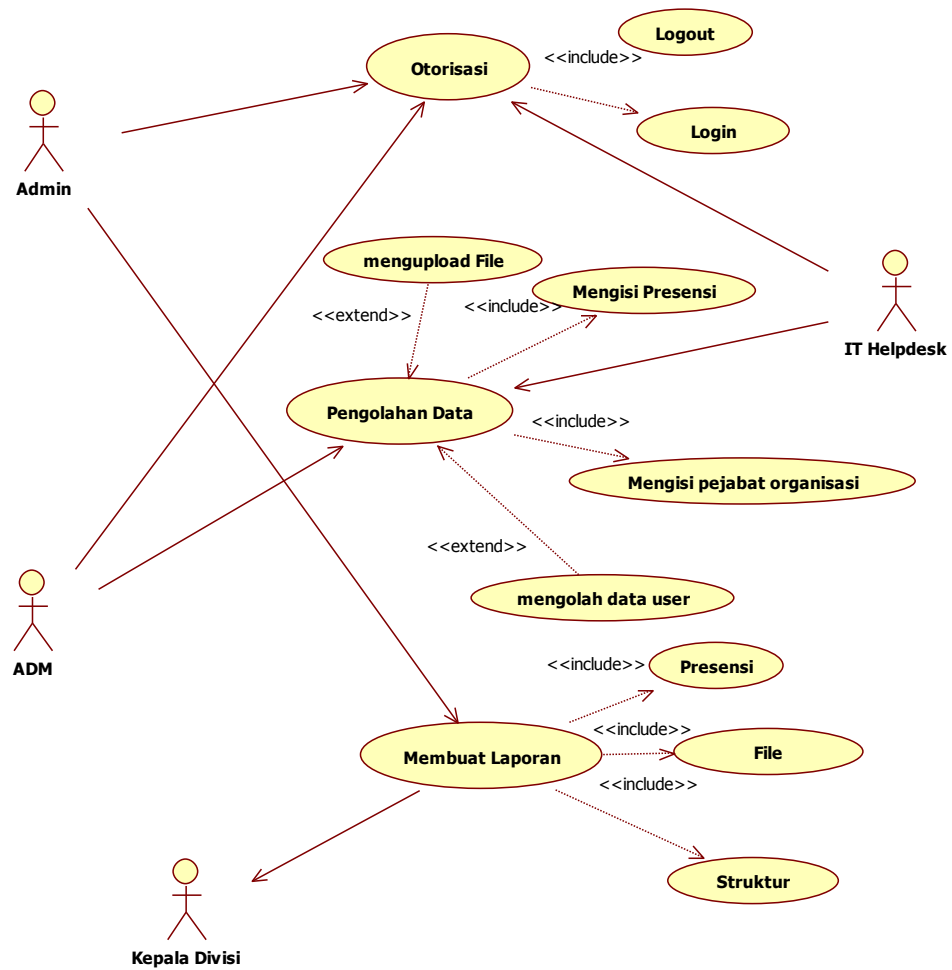
Tabel 3.1 Analisis Kebutuhan fungsional

No REQ	Kebutuhan	Keterangan
REQ 1.0	Mengelola data user	Sistem informasi ini dapat mengelola data user berupa pemegang hak akses.
REQ 1.1	Mengelola data presensi	Sistem informasi ini dapat mengelola data presensi.
REQ 1.2	Menyimpan data file upload	Sistem informasi ini dapat menyimpan data yang diupload.
REQ 2.0	Mengelola data organisasi	Sistem informasi ini dapat mengelola data organisasi berupa pemegang nama, jabatan, atasan dan foto pejabat.
REQ 2.1	Tambah user	Sistem informasi ini dapat menambah data user baru seperti nama, email dan statusnya.
REQ 3.0	Tambah pejabat	Sistem informasi ini dapat menambah data pejabat berupa nama, jabatan, atasan dan foto pejabat.

REQ 3.1	Tambah data presensi	Sistem informasi ini dapat menambah data presensi berupa tanggal masuk,jam masuk,jam datang dan telat.
REQ 3.2	Tampil data user	Sistem informasi ini dapat menampilkan user data yang tersimpan.
REQ 3.3	Tampil data presensi	Sistem informasi ini dapat menampilkan data presensi yang tersimpan dalam bentuk grafik.
REQ 3.4	Tampil data organisasi	Sistem informasi ini dapat menampilkan data organisasi yang tersimpan dalam bentuk grafik struktur organisasi.
REQ 3.5	Tampil upload data	Sistem informasi ini dapat menampilkan data file yang sudah diupload.
REQ 4.0	Ubah data user	Sistem informasi ini dapat mengubah data user yang sudah ada.
REQ 4.1	Hapus data user	Sistem informasi ini dapat menghapus data yang sudah ada.
REQ 4.2	Ubah data presensi	Sistem informasi ini dapat mengubah data presensi yang sudah ada.
REQ 4.3	Hapus data presensi	Sistem informasi ini dapat menghapus data presensi yang sudah ada.
REQ 4.4	Ubah data pejabat	Sistem informasi ini dapat mengubah data pejabat yang sudah ada.
REQ 5.0	Menghapus data pejabat	Sistem informasi ini dapat menghapus data yang sudah ada.
REQ 5.1	Cari nama user	Sistem informasi ini dapat mencari nama user yang akan dilihat detailnya.
REQ 5.2	Tambah struktur pejabat	Sistem informasi ini dapat menambahkan struktur pejabat.

### 3.5 Use Case dan Skenario

#### 3.5.1 Use Case



Gambar 3.1 diagram *use case*



### 3.5.2 Skenario

#### 3.5.2.1 Skenario Otorisasi

Tabel 3.2 Skenario Otorisasi

Identifikasi	
Nomor	REQ 1.1
Nama	Otorisasi
Tujuan	Masuk kedalam sistem
Deskripsi	
Aktor	Admin,Adm.IT Helpdesk
Skenario Utama	
Kondisi awal	System belum aktif
Aksi Aktor	Tanggapan system
1.Masukan Username	1.Membaca username dari input
2.Masukan Password	2.Membaca password dari input
	3.Mengecek input di database
Kondisi akhir	System siap digunakan

#### 3.5.2.2 Skenario Pengolahan Data File

Tabel 3.3 Skenario Pengolahan Data File

Identifikasi	
Nomor	REQ 1.2
Nama	Data File
Tujuan	Simpan data file
Deskripsi	
Aktor	Admin,Adm.IT Helpdesk
Skenario Utama	
Kondisi awal	System belum aktif
Aksi Aktor	Tanggapan system
1.Masukan Keterangan file yang diupload	1.Menyimpan keterangan file
2.Upload File	2.Menyimpan data upload file
Kondisi akhir	File dan keterangan berhasil disimpan

### 3.5.2.3 Skenario Pengolahan Data Presensi

Tabel 3.4 Skenario Data Presensi

Identifikasi	
Nomor	REQ 1.3
Nama	Data presensi
Tujuan	Simpan data presensi
Deskripsi	
Aktor	Adm,IT Helpdesk
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masukan tanggal presensi	1.Menyimpan tanggal presensi
2.Masukan jam datang	2.Menyimpan jam datang
	3.Tanggal dan jam masuk disimpan di database
Kondisi akhir	Data tersimpan dalam database

### 3.5.2.4 Skenario Pengolahan Data Pejabat Organisasi

Tabel 3.5 Skenario Data Pejabat Organisasi

Identifikasi	
Nomor	REQ 1.4
Nama	Data pejabat organisasi
Tujuan	Simpan data pejabat organisasi
Deskripsi	
Aktor	Adm,IT Helpdesk
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masukan nama pejabat	1.Menyimpan nama pejabat
2.Masukan jabatan	2.Menyimpan jabatan
3.Masukan Id atasan	3.Menyimpan Id atasan
4.Upload foto pejabat	4.Menyimpan foto pejabat
Kondisi akhir	Data dan struktur organisasi tersimpan dalam database

### 3.5.2.5 Skenario Pengolahan Data User

Tabel 3.6 Skenario Pengolahan Data User

Identifikasi	
Nomor	REQ 1.5
Nama	Data User
Tujuan	Mengolah data user
Deskripsi	
Aktor	Admin
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masuk menu user	1.Menampilkan halaman user
2.Tambah User	2.Menyimpan user baru
3.Edit user	3.Menyimpan perubahan data user
4.Delete User	4.Menghapus data user
Kondisi akhir	Data user tersimpan dalam database

### 3.5.2.6 Skenario Laporan Presensi

Tabel 3.7 Skenario Laporan Presensi

Identifikasi	
Nomor	REQ 1.6
Nama	Laporan Presensi
Tujuan	Menampilkan laporan presensi
Deskripsi	
Aktor	Admin,Kepala divisi
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masuk menu laporan presensi	1.Menampilkan halaman laporan presensi
2.Pilih bulan presensi	2.Menampilkan presensi bulan yang dipilih
	3.Data presensi ditampilkan
Kondisi akhir	Data presensi ditampilkan

### 3.5.2.7 Skenario Laporan Struktur Organisasi

Tabel 3.8 Skenario Laporan Struktur Organisasi

Identifikasi	
Nomor	REQ 1.1
Nama	Laporan struktur organisasi
Tujuan	Menampilkan laporan pejabat dan struktur organisasi
Deskripsi	
Aktor	Admin,Kepala divisi
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masuk menu laporan jabatan organisasi	1.Menampilkan halaman laporan jabatan organisasi
2.Pilih pejabat	2.Menampilkan info pejabat yang dipilih
	3.Data pejabat dan struktur organisasi ditampilkan
Kondisi akhir	Data pejabat dan struktur organisasi ditampilkan

### 3.5.2.8 Skenario Laporan File

Tabel 3.9 Skenario Laporan File

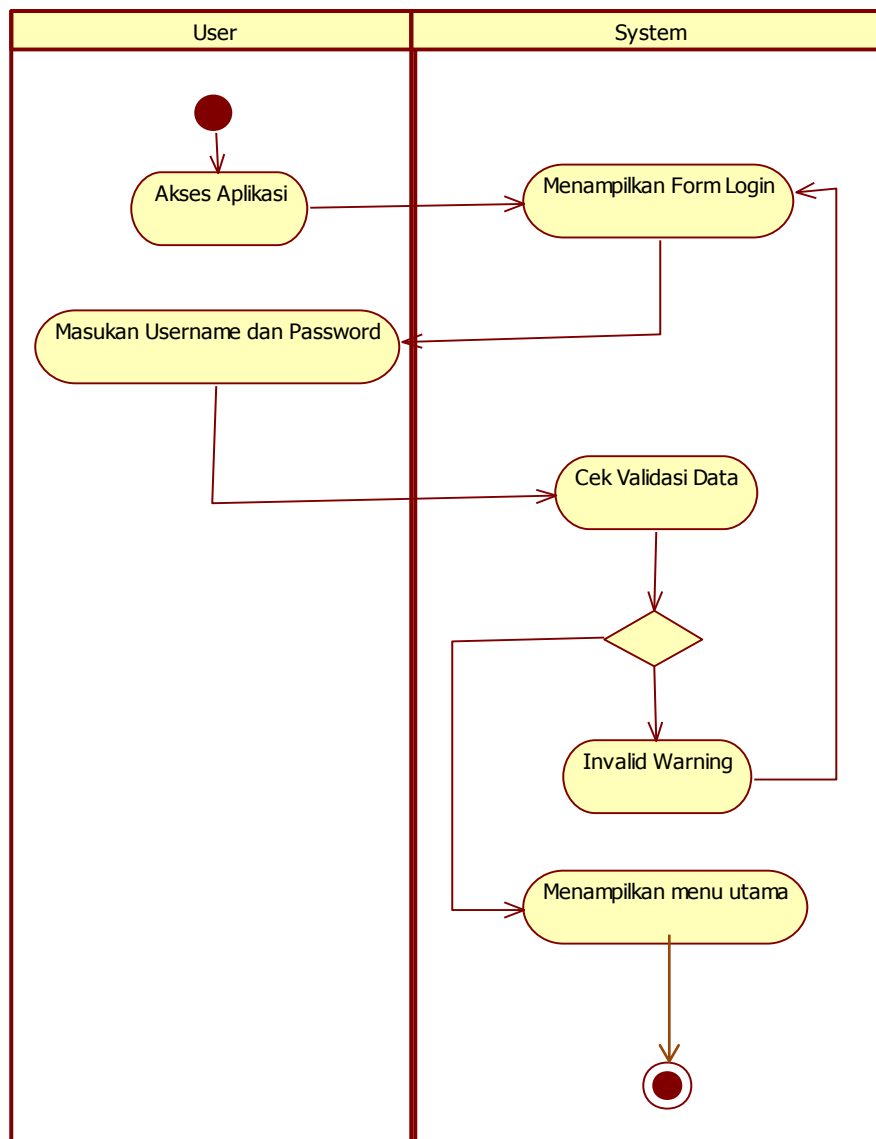
Identifikasi	
Nomor	REQ 1.1
Nama	Laporan file
Tujuan	Menampilkan laporan file
Deskripsi	
Aktor	Admin,Kepala divisi
Skenario Utama	
Kondisi awal	System <i>standby</i> data belum diolah
Aksi Aktor	Tanggapan system
1.Masuk menu laporan File	1.Menampilkan halaman laporan file
2.Pilih File	2.Menampilkan info file yang dipilih
	3.Data file ditampilkan
Kondisi akhir	Data file ditampilkan

**BAB IV**

**PERANCANGAN**

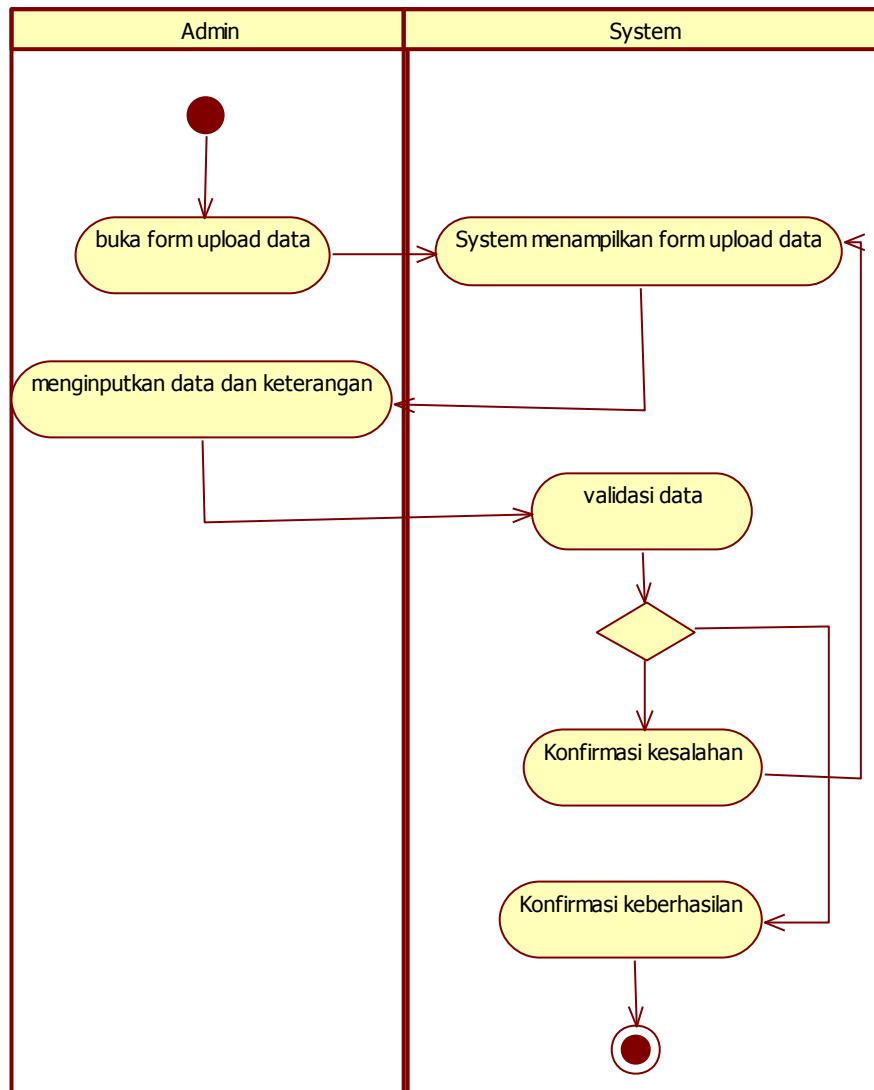
**4.1 Activity Diagram**

**4.1.1 Activity Diagram Otorisasi**



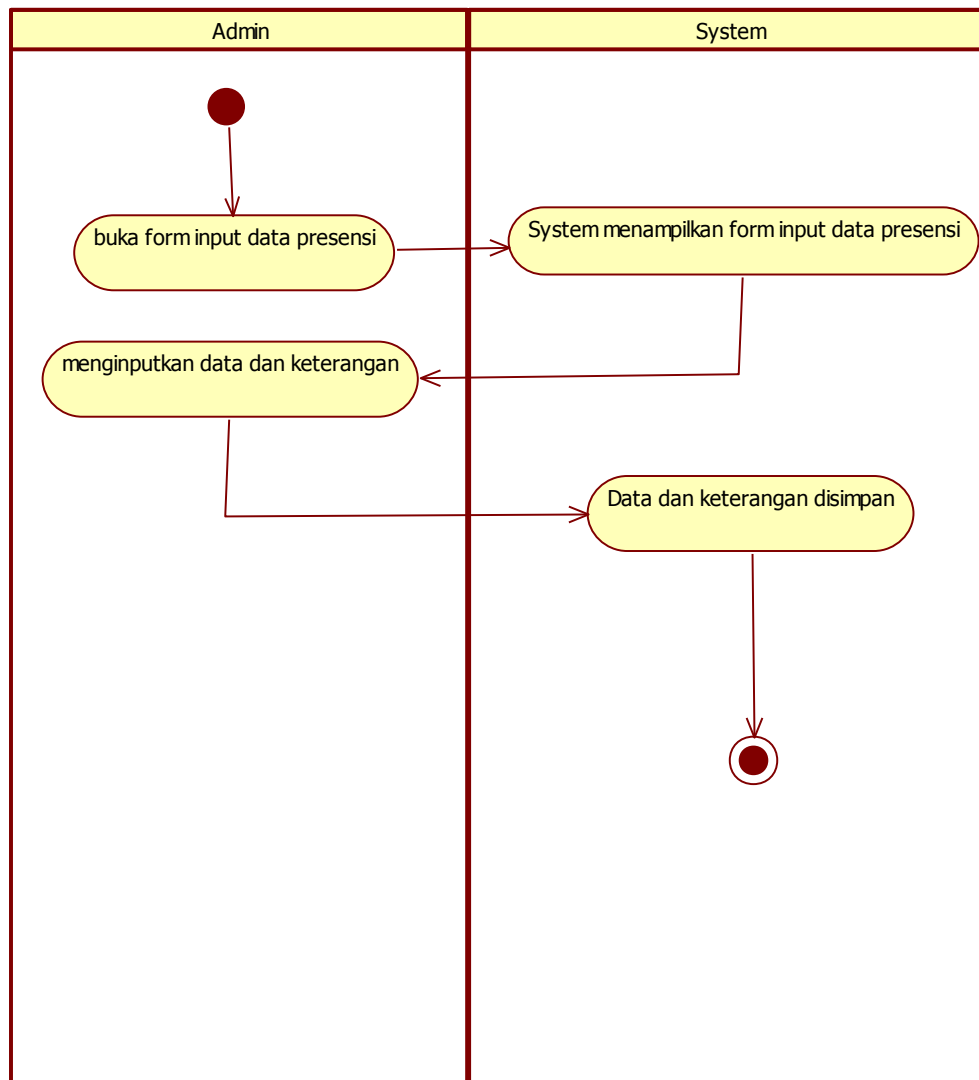
**Gambar 4.1 Activity Diagram Otorisasi**

#### 4.1.2 Activity Diagram Pengolahan Data File



Gambar 4.2 activity diagram pengolahan data file

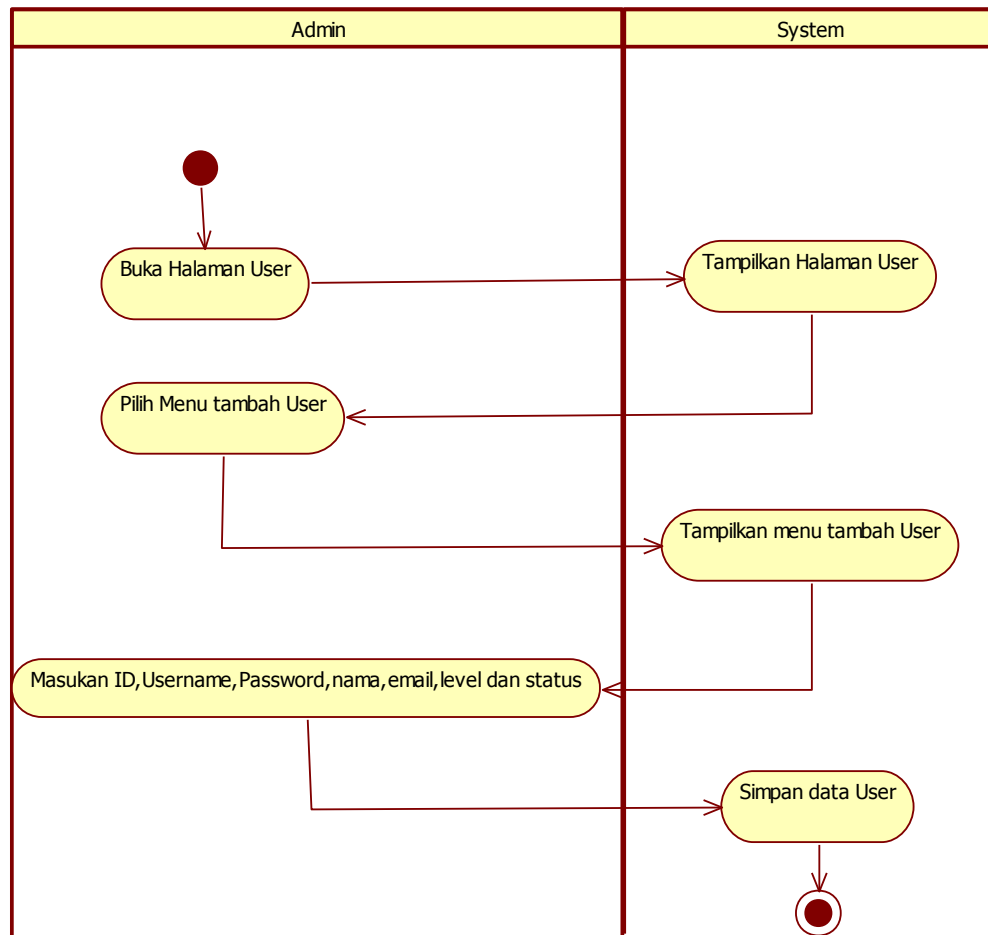
#### 4.1.3 Activity Diagram Pengolahan Data Presensi



**Gambar 4.3 activity diagram pengolahan data presensi**

#### 4.1.4 Activity Pengolahan Data User

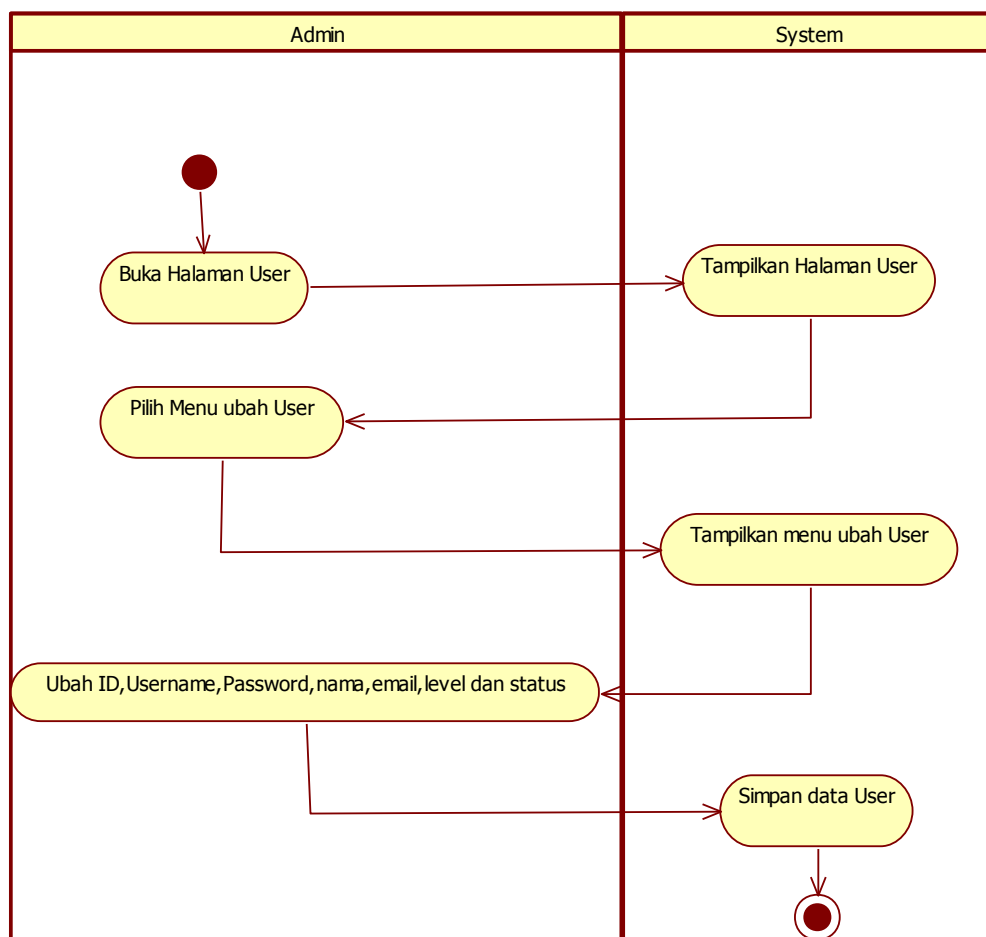
##### 4.1.4.1 Activity Pengolahan Data User Tambah User



**Gambar 4.4 activity diagram tambah user**

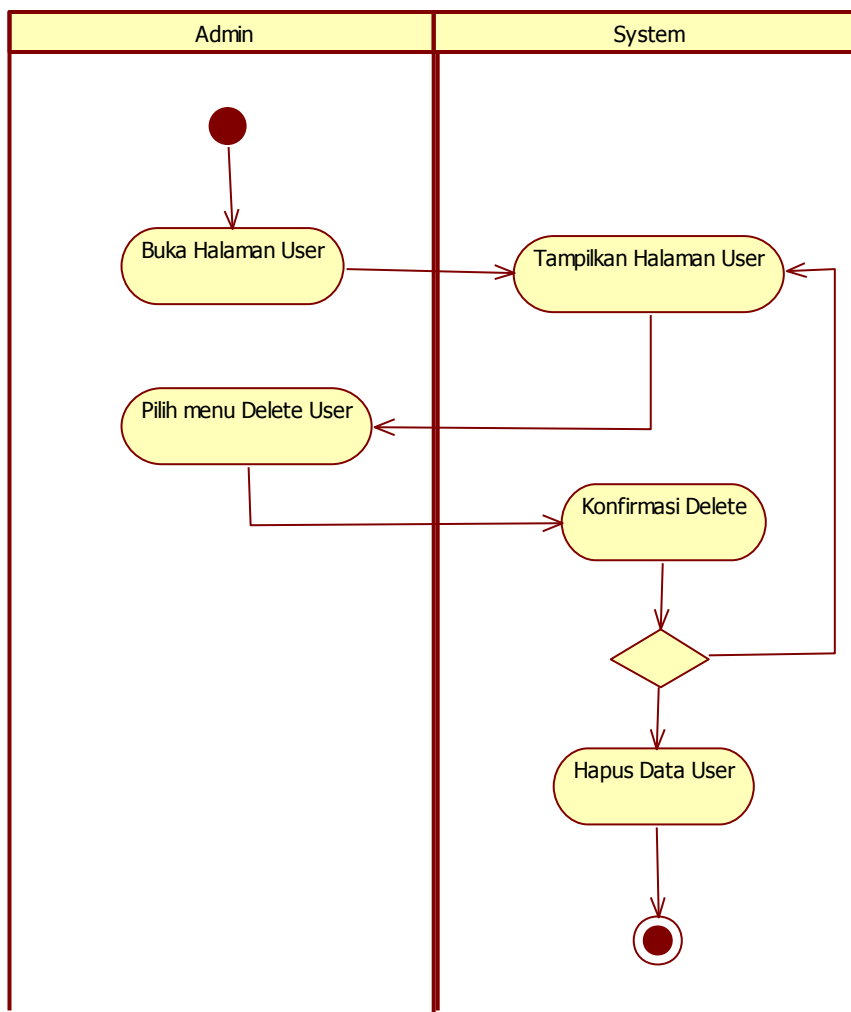


#### 4.1.4.2 Activity Diagram Ubah User



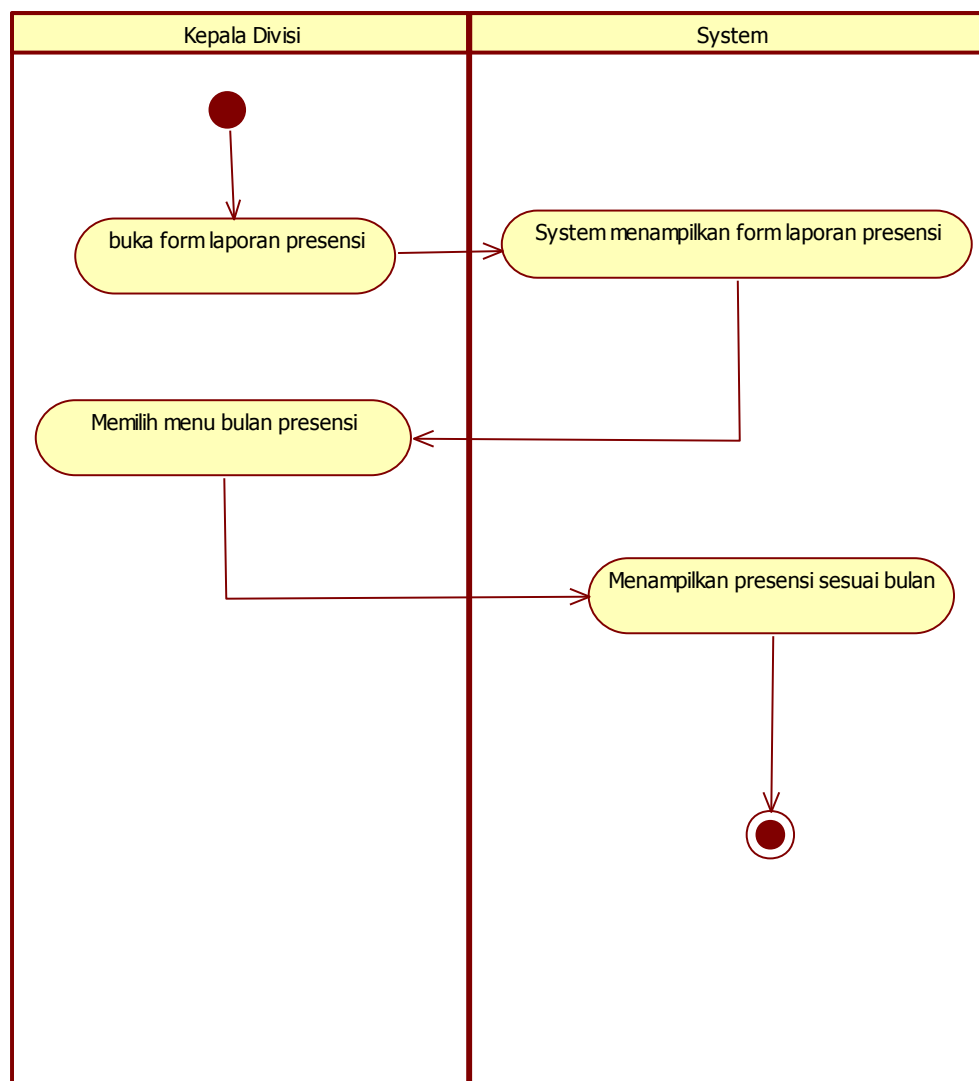
**Gambar 4.5 activity diagram ubah user**

#### **4.1.4.3 Activity Diagram Hapus User**



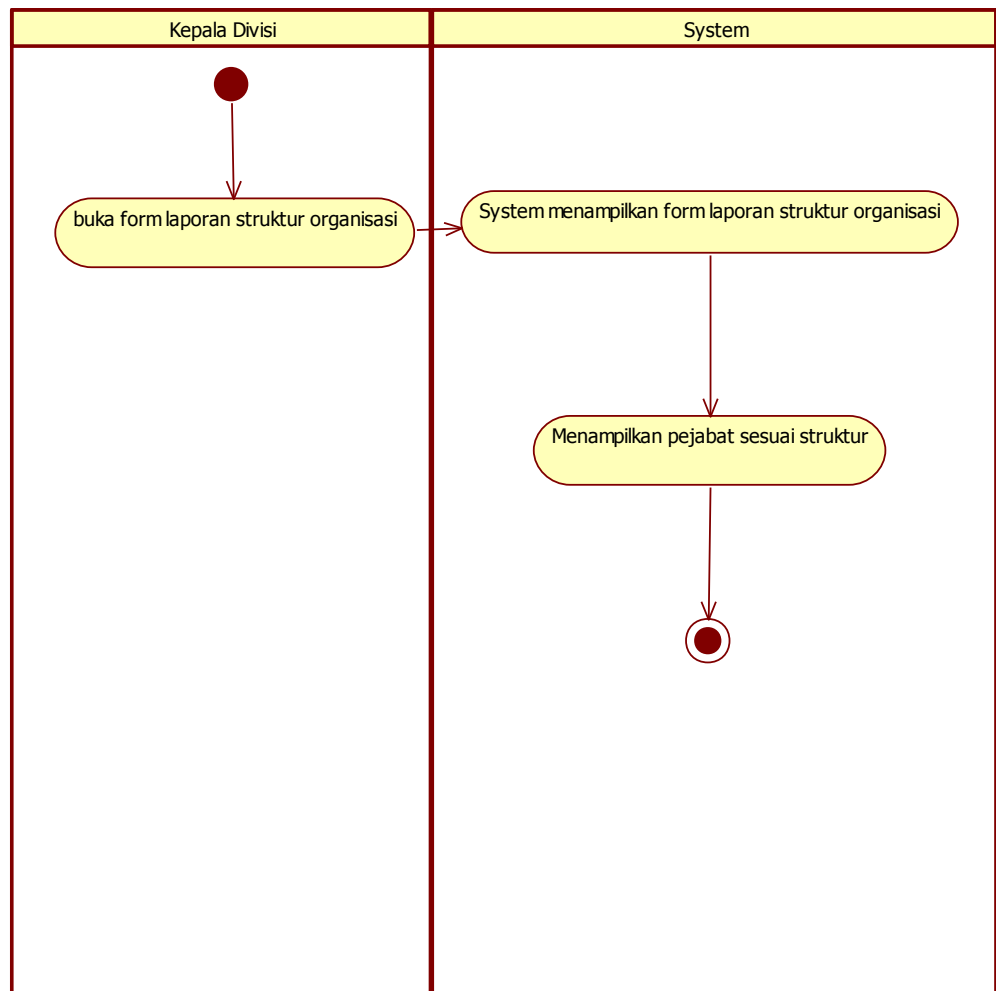
**Gambar 4.6 Activity Diagram Hapus User**

#### 4.1.5 Activity Diagram Laporan Presensi



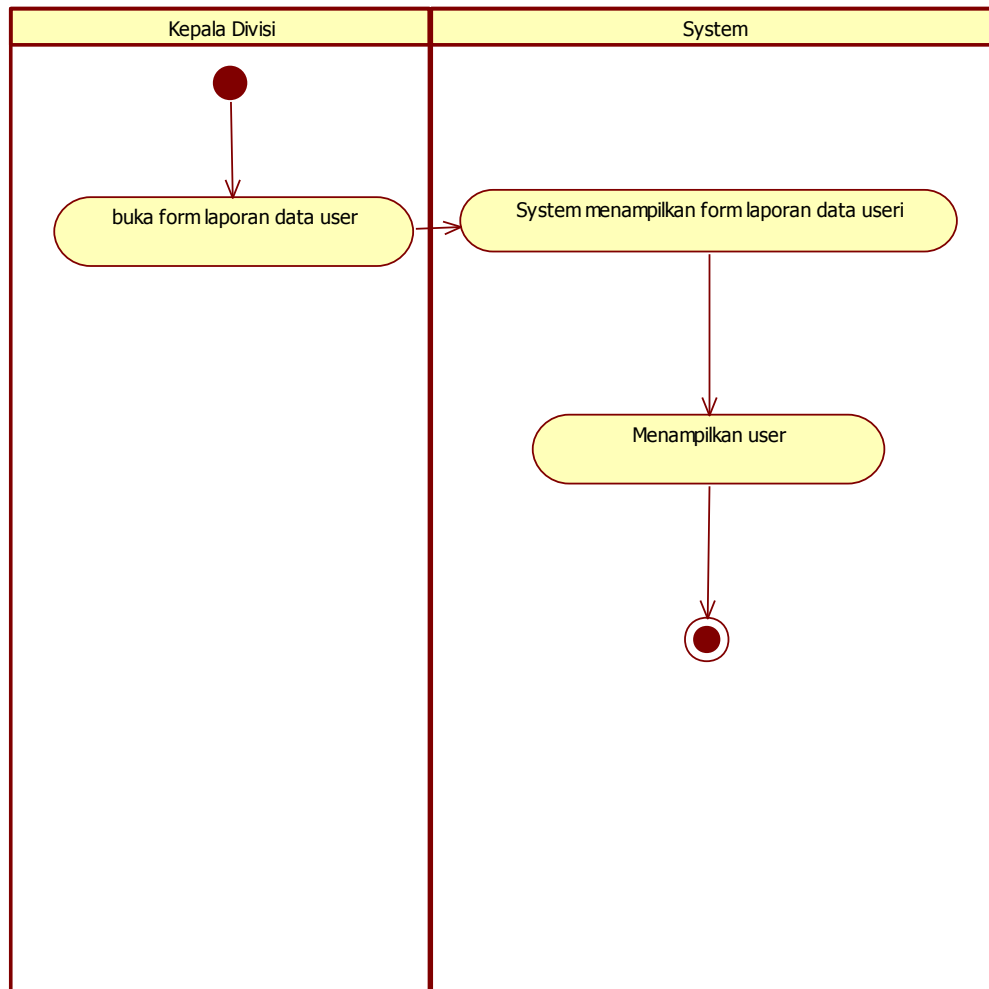
Gambar 4.7 Activity Diagram Laporan Presensi

#### 4.1.6 Activity Diagram Laporan Struktur Organisasi



**Gambar 4.8 Activity Diagram Laporan Struktur Organisasi**

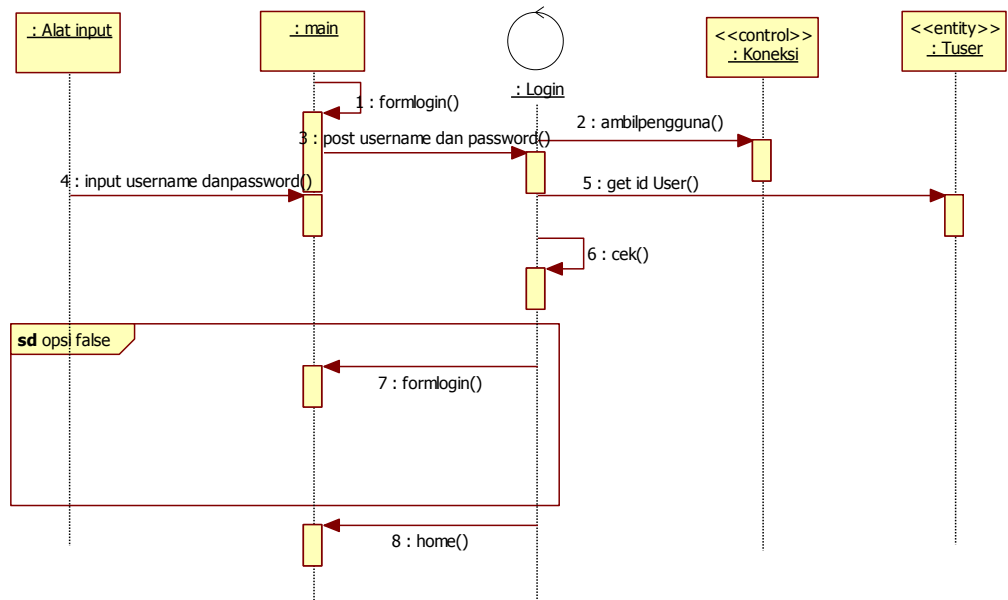
#### 4.1.7 Activity Diagram Laporan Data User



**Gambar 4.9 Activity Diagram Data User**

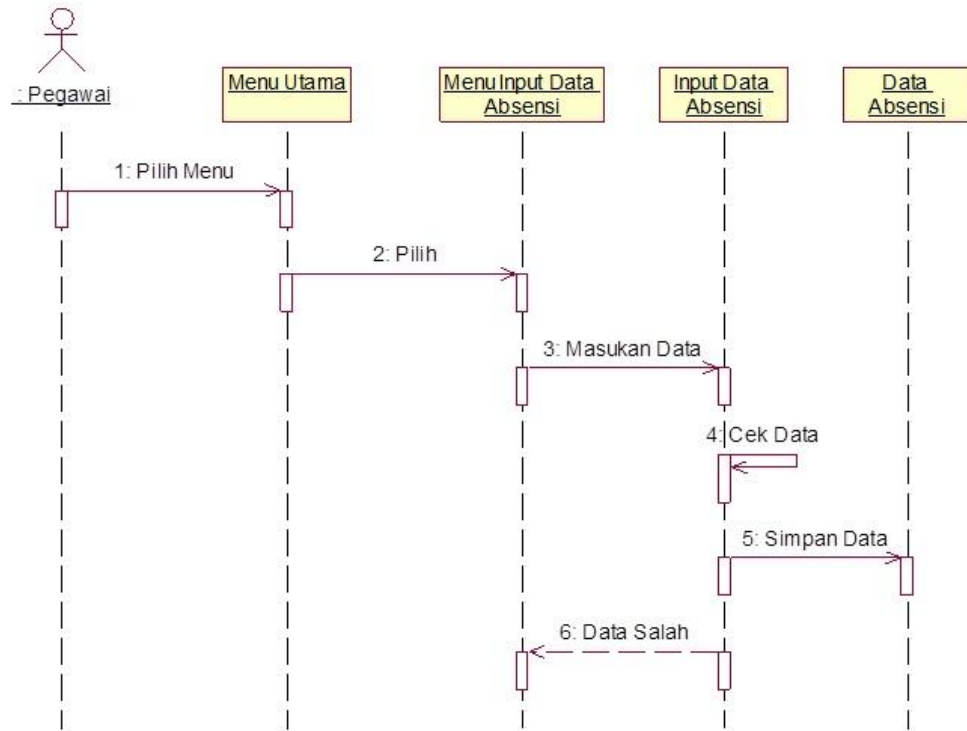
## 4.2 Sequence Diagram

### 4.2.1 Sequence Diagram Otorisasi



Gambar 4.10 Sequence Diagram Otomatisasi

#### 4.2.2 Sequence Diagram Pengolahan Data Presensi

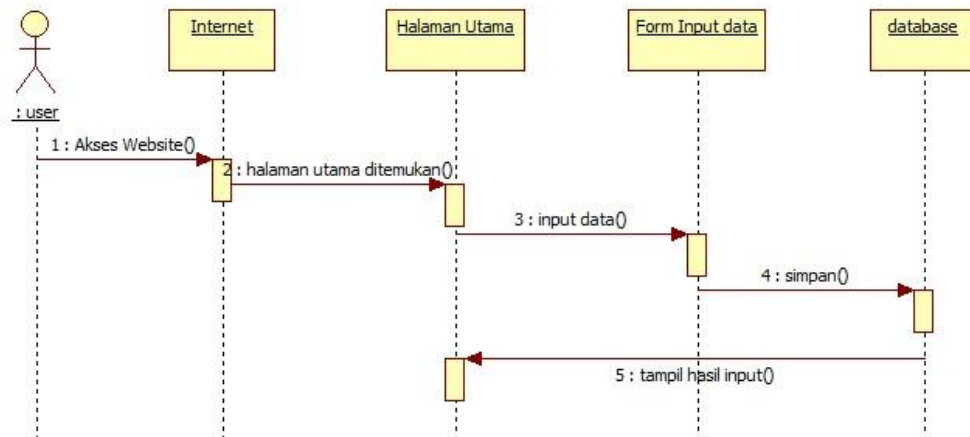


Gambar 4.11 Sequence Diagram Pengolahan Data Presensi



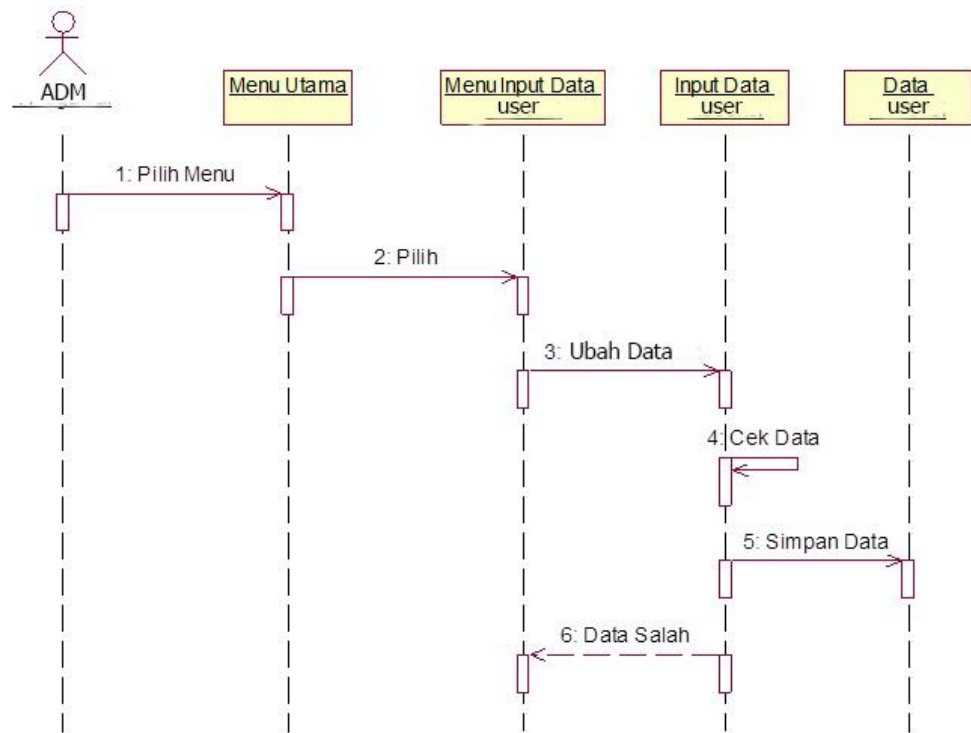
### 4.2.3 Sequence Diagram Pengolahan Data User

#### 4.2.3.1 Sequence Diagram Pengolahan Data Tambah User



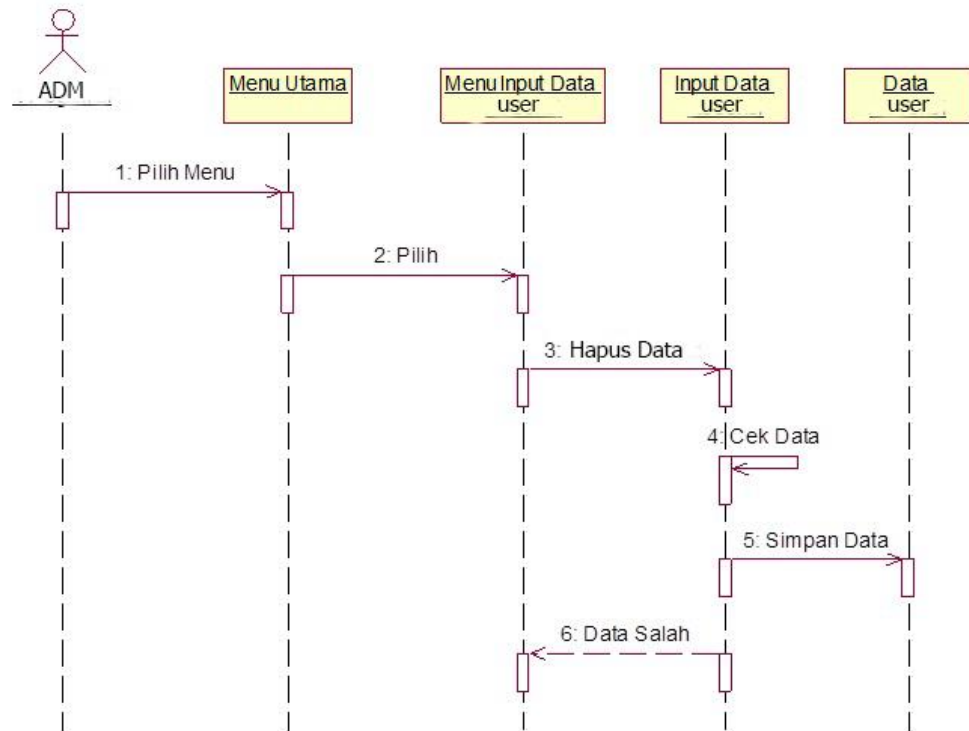
Gambar 4.12 Sequence Diagram Tambah User

#### 4.2.3.2 Sequence Diagram Pengolahan Data Ubah User



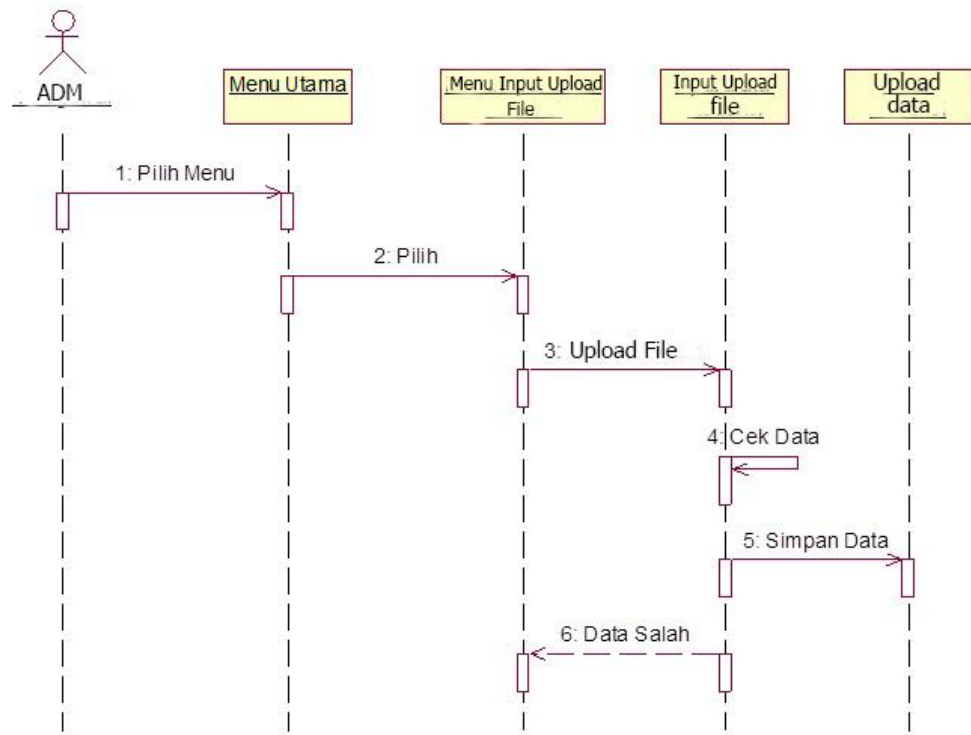
Gambar 4.13 Sequence Diagram Ubah User

#### 4.2.3.3 Sequence Diagram Pengolahan Data Hapus User



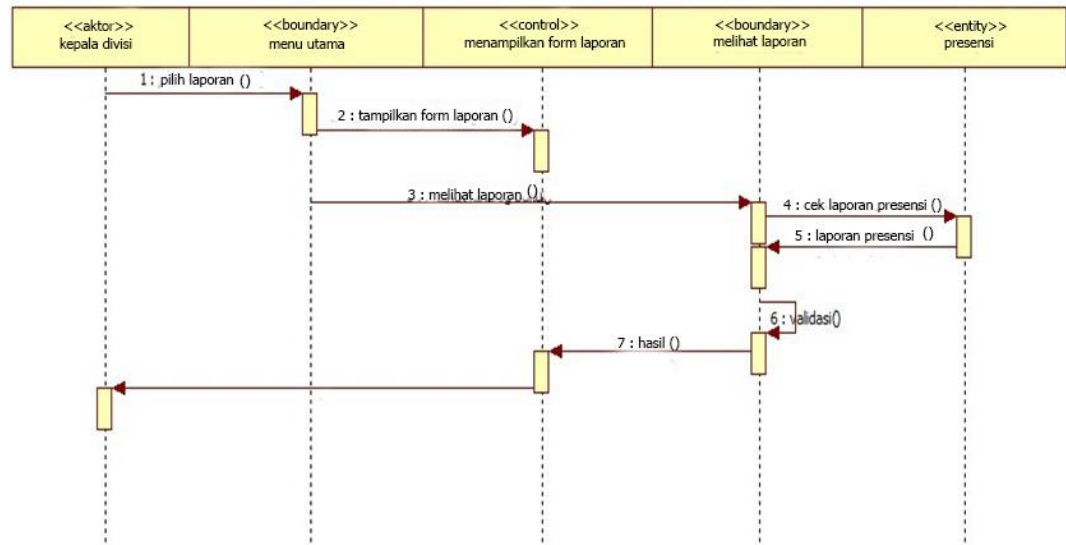
#### 4.14 Sequence Diagram Hapus User

#### 4.2.4 Sequence Diagram Pengolahan Data File Upload



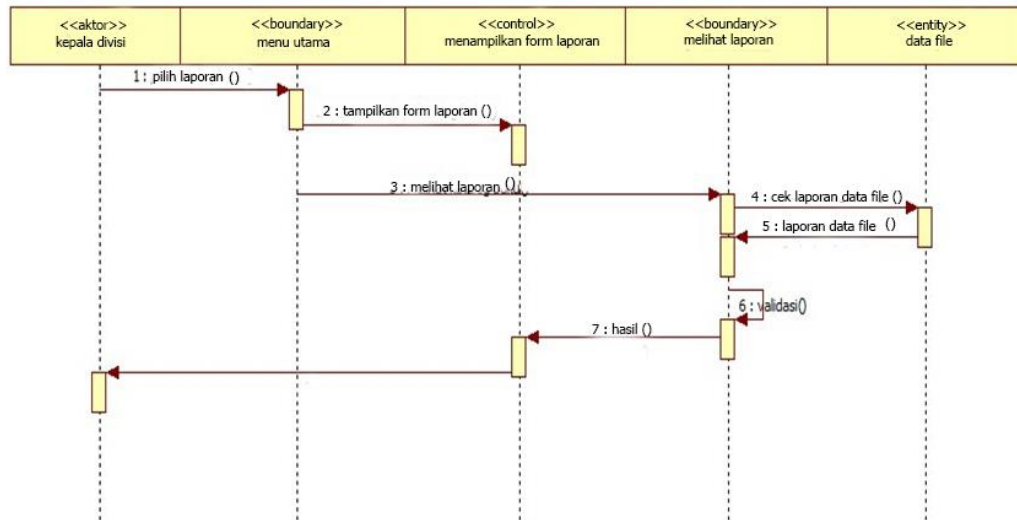
**Gambar 4.15 Sequence Diagram Upload File**

#### 4.2.5 Sequence Diagram Laporan Presensi



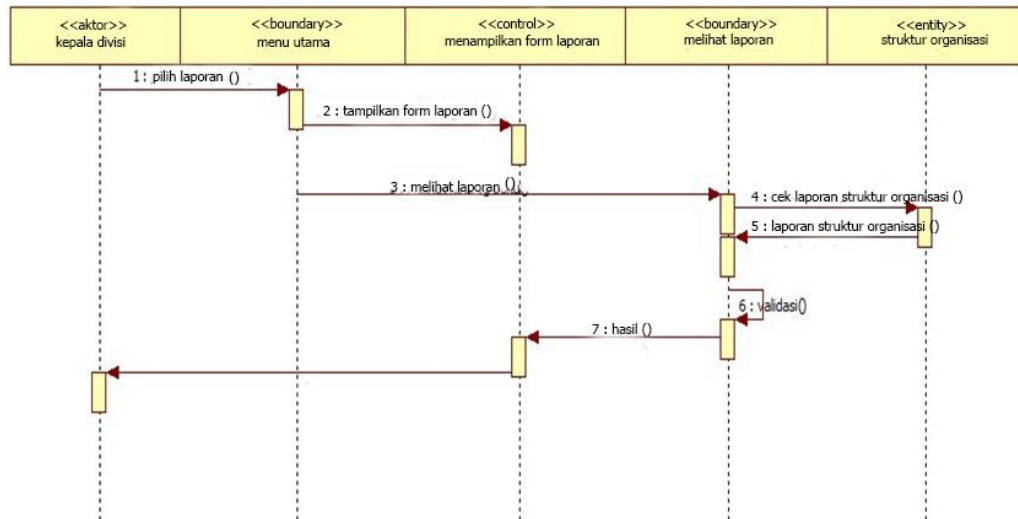
Gambar 4.16 Sequence Diagram Laporan Presensi

#### 4.2.6 Sequence Diagram Laporan Upload File



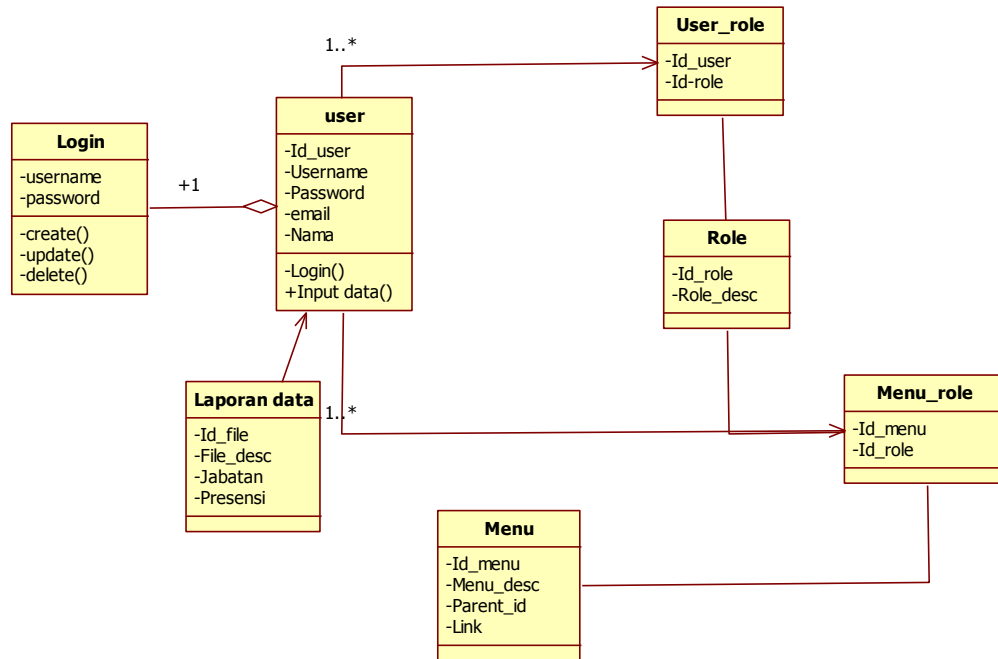
Gambar 4.17 Sequence Diagram Laporan Upload Data

#### 4.2.7 Sequence Diagram Laporan Struktur Organisasi



Gambar 4.18 Sequence Diagram Laporan Struktur Organisasi

#### 4.2.8 Class Diagram



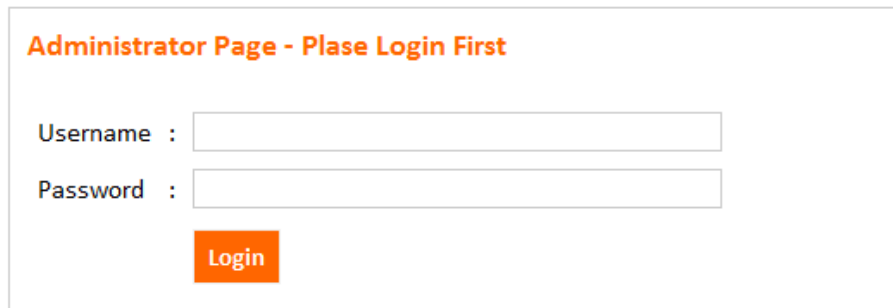
Gambar 4.19 Class Diagram



### 4.3 Antar Muka

Antar muka *input-output* merupakan rancangan antar muka yang akan digunakan sebagai perantara *user* dengan *system web*. Layout antar muka dari perancangan *web* ini adalah sebagai berikut:

#### 4.3.1 Form Login



The image shows a web form titled "Administrator Page - Plase Login First" in orange text. Below the title, there are two input fields: "Username :" and "Password :". Each label is followed by a white rectangular input box with a thin grey border. Below the password field is an orange rectangular button with the word "Login" in white text.

**Gambar 4.20 Form Login**

Form *login* adalah form yang ditampilkan pertama kali ketika aplika *web* ini dijalankan. Form ini digunakan untuk masuk/*login* kedalam *system*, dan hanya *user* yang terdaftar yang bisa login kedalam *system*.

#### 4.3.2 Hak Akses

Karena *system* ini bersifat multi *user* maka menu menu yang ada dibatasi sesuai *user* role dan menu role yang telah didefinisikan permasing-masing *user*. Batasan hak aksesnya yaitu sebagai berikut:

1. IT *User/Administrator*

Seorang administrator dapat mengakses seluruh menu yang ada dalam *web*.

2. Admin AKT

Seorang admin AKT ini hanya dapat mengakses menu *Dashboard,history* dan absensi.

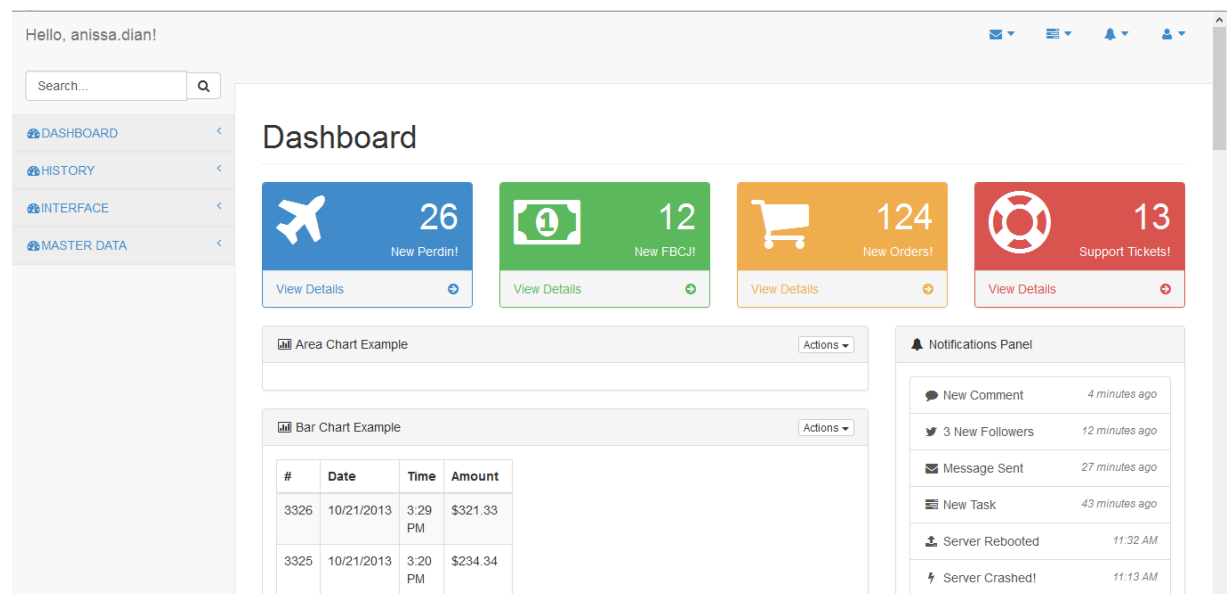
3. IT Helpdesk

Seorang IT Helpdesk hanya bisa mengakses sebagian sub menu *Interface*.

4. Kepala Divisi

Seseorang yang mengakses laporan

### 4.3.3 Form Menu Utama



**Gambar 4.21 Menu Utama**

Gambar diatas merupakan tampilan menu utama apabila *user* telah melakukan login. Menu utama/*main page* ini berisi menu-menu inti yang dapat diakses sesuai kebutuhan.

#### 4.3.4 Menu *User*

The screenshot displays the 'DATA USER' management interface. On the left, there is a form to add a new user with fields for ID USER, USERNAME, PASSWORD, EMAIL, NAMA, LEVEL, and STATUS, followed by a 'Simpan' button. The main area shows a table of existing users with columns for ID USER, USERNAME, PASSWORD, NAMA, EMAIL, and ACTION. The table contains three entries. The interface includes a sidebar with navigation links (DASHBOARD, HISTORY, INTERFACE, MASTER DATA) and a top navigation bar with a search bar and user profile information.

**Form Fields:**

- ID USER :
- USERNAME :
- PASSWORD :
- EMAIL :
- NAMA :
- LEVEL :
- STATUS :
- [Simpan](#)

**DATA USER**

DataTables Advanced USER

Sun, 18 Jan 2015

10 records per page

Search:

ID USER	USERNAME	PASSWORD	NAMA	EMAIL	ACTION
1	anissa.dian	1234	Annisa Dian	annisadian@inti.co.id	<a href="#">Ubah</a>    <a href="#">Hapus</a>
2	nelly	11111	Nelly	nelly@inti.co.id	<a href="#">Ubah</a>    <a href="#">Hapus</a>
3	wulan	guest	Wulan	wulan@inti.co.id	<a href="#">Ubah</a>    <a href="#">Hapus</a>

Showing 1 to 3 of 3 entries

Previous **1** Next


**Gambar 4.22** *User Page*

Gambar diatas adalah halaman *user* dimana administrator dapat menambahkan *user* baru atau mengedit data *user* yang ada atau menghapus *user*.

### 4.3.5 Data Role

ID ROLE :

ROLE DESCRIPTION :

 Simpan

## DATA ROLE

DataTables Advanced ROLE

10 records per page

Sun, 18 Jan 2015

Search:

ID ROLE	ROLE DESC	ACTION
1	ADM_AKT	Ubah    Hapus
2	ADM_KSR	Ubah    Hapus
3	IT_USER	Ubah    Hapus
4	ADMINISTRATOR	Ubah    Hapus
5	IT_HELPDESK	Ubah    Hapus

Hello, anissa.dian!

Search...

DASHBOARD

HISTORY

INTERFACE

MASTER DATA

## DATA ROLE

DataTables Advanced ROLE

10 records per page

Search:

ID ROLE	ROLE DESC	ACTION
1	ADM_AKT	Ubah    Hapus
2	ADM_KSR	Ubah    Hapus
3	IT_USER	Ubah    Hapus
4	ADMINISTRATOR	Ubah    Hapus
5	IT_HELPDESK	Ubah    Hapus

Showing 1 to 5 of 5 entries

Previous 1 Next

**Gambar 4.23 Role Page**

Gambar diatas adalah halaman aturan/*role* yang diberikan kepada *user*. Admin dapat menambah,mengedit dan menghapus *role*.

### 4.3.6 Data Menu

Hello, anissa.diani

Search...

DASHBOARD

HISTORY

INTERFACE

MASTER DATA

## DATA MENU

DataTables Advanced MENU

10 records per page

Search:

ID MENU	MENU DESC	LINK	LEVEL	ACTION
1	DASHBOARD		0	Ubah    Hapus
2	PROJECT TELCO		1	Ubah    Hapus
3	PROJECT CELCO		1	Ubah    Hapus
4	PROJECT PE		1	Ubah    Hapus
5	HISTORY		0	Ubah    Hapus
6	ABSENSI		1	Ubah    Hapus
7	CUTI		1	Ubah    Hapus
8	PERDIN		1	Ubah    Hapus
9	INTERFACE		0	Ubah    Hapus
10	SAP-ORACLE		1	Ubah    Hapus

**Gambar 4.24 Data Menu Page**


Halaman ini berisi data menu-menu yang tersedia yang dapat diakses oleh *user*.

Administrator dapat menambah, mengedit dan menghapus menu.

#### 4.3.7 User Role

ID USER :

id\_role :

 Simpan

ID USER	ID ROLE	ACTION
1	1	<a href="#">Ubah</a>    <a href="#">Hapus</a>
2	1	<a href="#">Ubah</a>    <a href="#">Hapus</a>
1	2	<a href="#">Ubah</a>    <a href="#">Hapus</a>
3	2	<a href="#">Ubah</a>    <a href="#">Hapus</a>
1	3	<a href="#">Ubah</a>    <a href="#">Hapus</a>
2	3	<a href="#">Ubah</a>    <a href="#">Hapus</a>
3	3	<a href="#">Ubah</a>    <a href="#">Hapus</a>
1	4	<a href="#">Ubah</a>    <a href="#">Hapus</a>
1	5	<a href="#">Ubah</a>    <a href="#">Hapus</a>
3	5	<a href="#">Ubah</a>    <a href="#">Hapus</a>

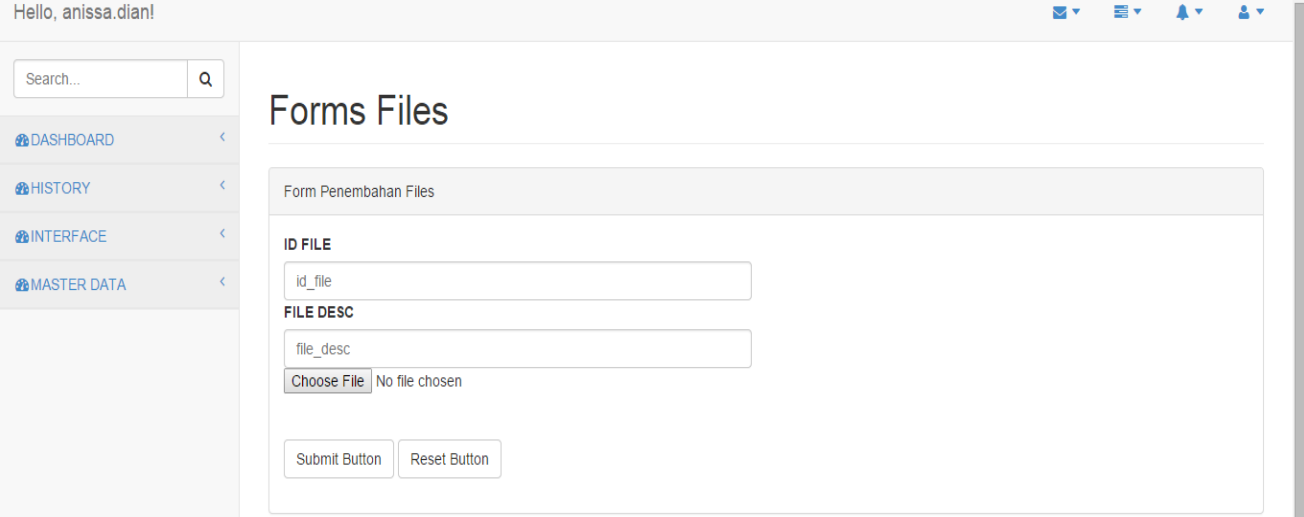
Sun, 18 Jan 2015

**Gambar 4.25 User Role Page**

Halaman ini berisi aturan yang disematkan kepada masing masing *user*.

Administrator dapat menambah,mengedit dan menghapus *user* role.

### 4.3.8 Menu Upload



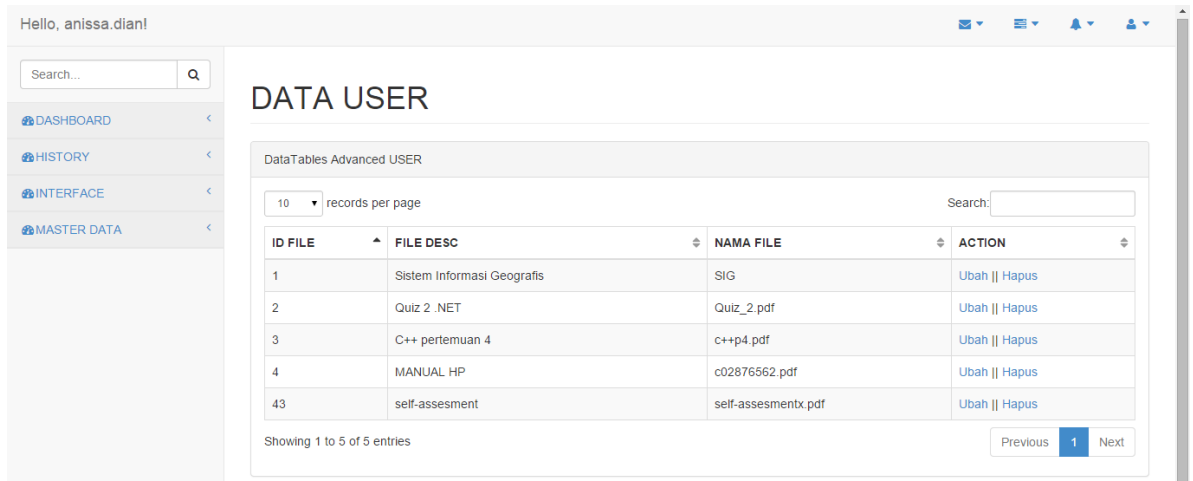
The screenshot shows a web application interface for uploading files. At the top, a header bar displays 'Hello, anissa.dian!' on the left and navigation icons (mail, menu, bell, user) on the right. Below the header is a sidebar with a search bar and four menu items: 'DASHBOARD', 'HISTORY', 'INTERFACE', and 'MASTER DATA', each with a left-pointing arrow. The main content area is titled 'Forms Files' and contains a form titled 'Form Penambahan Files'. The form has two sections: 'ID FILE' with a text input field containing 'id\_file', and 'FILE DESC' with a text input field containing 'file\_desc'. Below these fields is a file selection area with a 'Choose File' button and the text 'No file chosen'. At the bottom of the form are two buttons: 'Submit Button' and 'Reset Button'.

**Gambar 4.26 Upload Page**

Halaman ini merupakan halaman untuk *mengupload file* untuk kemudian disimpan *didatabase* dan *path directory*.



### 4.3.9 Menu Show File



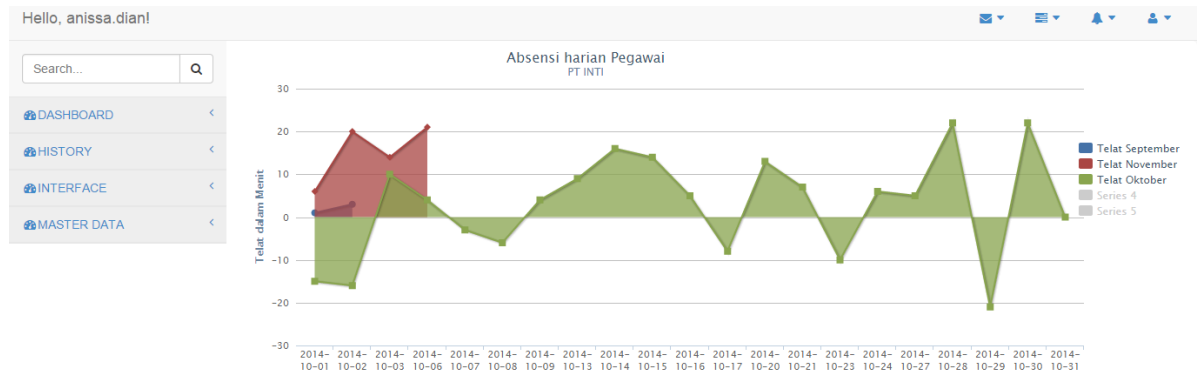
The screenshot displays a web application interface. At the top, a header bar shows a greeting 'Hello, anissa.dian!' and navigation icons. A sidebar on the left contains a search bar and menu items: DASHBOARD, HISTORY, INTERFACE, and MASTER DATA. The main content area is titled 'DATA USER' and features a 'DataTables Advanced USER' table. The table has four columns: 'ID FILE', 'FILE DESC', 'NAMA FILE', and 'ACTION'. It contains five rows of data. Below the table, it indicates 'Showing 1 to 5 of 5 entries' and includes 'Previous', '1', and 'Next' pagination links.

ID FILE	FILE DESC	NAMA FILE	ACTION
1	Sistem Informasi Geografis	SIG	Ubah    Hapus
2	Quiz 2 .NET	Quiz_2 .pdf	Ubah    Hapus
3	C++ pertemuan 4	c++p4 .pdf	Ubah    Hapus
4	MANUAL HP	c02876562 .pdf	Ubah    Hapus
43	self-assesment	self-assesmentx .pdf	Ubah    Hapus

**Gambar 4.27 Show File**

Halaman ini berisi tampilan dari semua data yang sudah *diupload* beserta *link* untuk membuka *attachment* yang disematkan.

#### 4.3.10 Menu Absensi

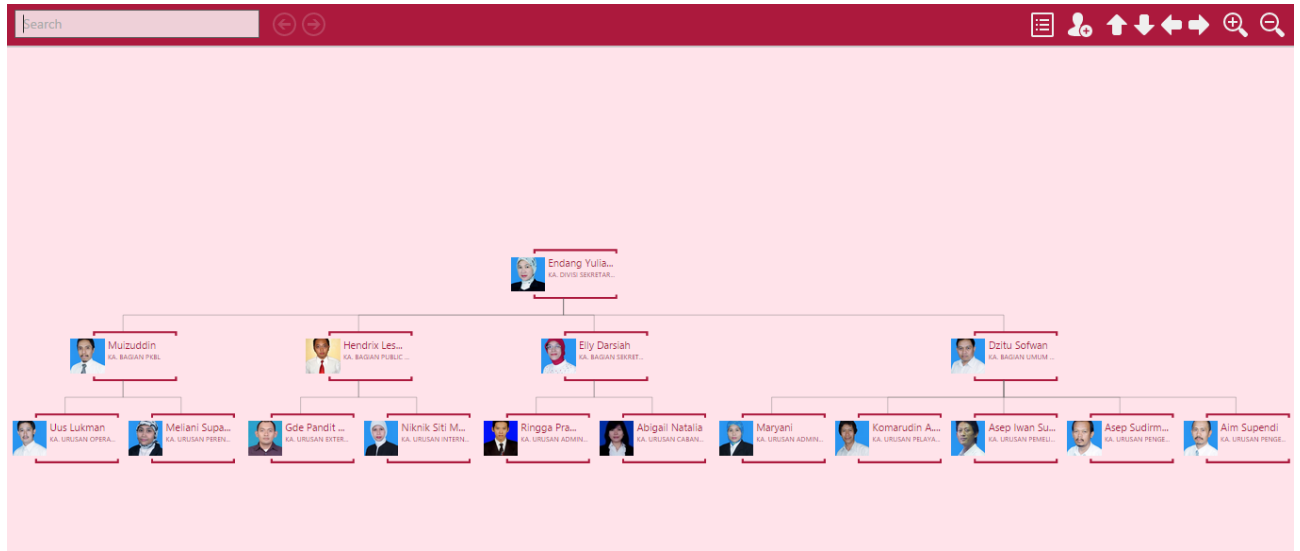


**Gambar 4.28 Absensi Page**

Halaman ini berisi data absensi dan waktu telat masuk harian berupa grafik.

Masing-masing warna mewakili bulan absensi.

### 4.3.11 Menu Struktur Organisasi



**Gambar 4.29 Org Chart**

Halaman ini berisi struktur organisasi perusahaan. Diktuai oleh (*/root*) dan dapat mempunyai *child* (bertindak sebagai *parent*). Dapat ditambahkan anggota baru dan menempatkan posisi sesuai *parent* yang dikehendaki. Administrator juga dapat menghapus anggota.

## BAB V

### PENUTUP

Berdasarkan hasil perancangan *system* yang telah dilakukan, dapat diambil kesimpulan dan banyak saran yang berhubungan dengan pembahasan yang telah dikemukakan pada bab-bab sebelumnya.

#### 5.1 Kesimpulan

Dari berbagai penjelasan yang telah diuraikan dalam laporan tugas kerja praktek ini, maka dapat disimpulkan beberapa hal sebagai berikut:

1. Perancangan *web* ini menyajikan tampilan yang dinamis, responsive dan tampil baik diakses oleh *browser* google chrome, mozilla firefox, opera dan mobile device.
2. Dengan menggunakan *framework Codeigniter* perancangan *web* menjadi lebih mudah dan tertata dengan baik karena dibagi dalam *controller, model* dan *view*
3. Dengan penggunaan *template* SB ADMIN 2, tampilan *web* menjadi lebih enak dipandang dan *user friendly* dan juga *responsive*
4. Penggunaan database membuat *web* menjadi dinamis, cepat dalam prosesnya dan juga mudah dalam proses perubahan data.
5. Penggunaan *javascript* membuat proses menampilkan absensi dan struktur organisasi menjadi lebih hidup dan mudah.

## **5.2 Saran**

Penulis berharap semoga dimasa yang akan datang aplikasi ini dapat berisi menu yang lebih lengkap lagi dan juga lebih interaktif serta mampu menjawab kebutuhan konsumen dimasa mendatang.

## DAFTAR PUSTAKA

1. Sidik,Betha.2012,*Menggunakan framework CodeIgniter 2.x untuk memudahkan pengembangan pemrograman aplikasi web dengan PHP* 5,Informatika,Bandung.
2. Arief,M.R.2011,*pemrograman web dinamis menggunakan PHP & MySQL*,Andi Offset,Yogyakarta.