



# Aplikasi E-Tilang Pada Tablet Sistem Operasi Berbasis Android

TUGAS AKHIR

Disusun dan Diajukan untuk Melengkapi Syarat-syarat yang Diperlukan guna  
Memperoleh Diploma Tiga Politeknik

Oleh :

Ibnu Peristiwaan Azis

NIM. 130926010Y

Konsentrasi Teknik Informatika  
Program Studi Teknik Telekomunikasi  
Jurusan Teknik Elektro

**POLITEKNIK NEGERI JAKARTA**  
**DEPOK**  
**2012**

## **LEMBAR PERSETUJUAN PEMBIMBING**

Tugas akhir dengan judul “**Aplikasi E-Tilang pada Tablet Sistem Operasi Berbasis Android**” pada Program Diploma III Politeknik Negeri Jakarta, Jurusan Teknik Elektro, Konsentrasi Teknik Informatika telah diuji dalam Sidang Tugas Akhir hari Selasa, 14 Agustus 2012 dan dinyatakan LULUS.

Depok, 3 Agustus 2012

Disetujui,  
Dosen Pembimbing

**Agus Wagyana,ST.MT**  
NIP. 19680824 199903 1 002

## **LEMBAR PENGESAHAN**

Tugas akhir dengan judul “**Aplikasi E-Tilang pada Tablet Sistem Operasi Berbasis Android**” pada Program Diploma Tiga Politeknik Negeri Jakarta, Jurusan Teknik Elektro, Program Konsentrasi Informasi Teknologi telah diuji dalam sidang Tugas Akhir hari Selasa, 14 Agustus 2012 dan dinyatakan LULUS

Depok, 3 Agustus 2012

Disahkan oleh,  
Jurusan Teknik Elektro Politeknik Negeri Jakarta  
Ketua,

**Iwa Sudradjat, ST., MT.**

NIP. 196106071986011002

## ABSTRAK

Aplikasi E-Tilang adalah salah satu prototype atau rancang bangun sistem yang bertujuan untuk memudahkan kepolisian dalam menjalankan tugasnya. E-Tilang berjalan pada dua sisi, yaitu kepolisian dan kejaksaan. pada sisi kepolisian sistem akan berjalan pada tablet bersistem operasi Android yang akan mendigitalisasi form bukti pelanggaran yang menyimpan ke dalam suatu database yang akan dieksekusi atau disubmit oleh sisi kejaksaan yang dalam hal ini medianya adalah website dengan PHP yang menggunakan Codeigniter sebagai frameworknya. E-Tilang dibangun dengan menggunakan Java dan MySQL sebagai database, pengguna kendaraan yang melanggar lalu lintas akan diberikan bukti tilang berupa ID Tilang yang dikirim melalui SMS dengan teknologi SMS Gateway dengan Gammu. Setelah menerapkan E-Tilang pihak kepolisian akan lebih mudah dalam melaksanakan tugas dalam tindak pelanggaran lalu lintas, dan tertib dalam administrasi juga mengurangi penyimpangan dalam operasionalnya.

**Kata kunci:** *E-Tilang, Aplikasi Tablet, Sms Gateway, Android, CodeIgniter. PHP, Website,*

## KATA PENGANTAR

Alhamdulillah Puji dan syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, Sholawat serta salam tercurah kepada Nabi Muhammad SAW, sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “**Aplikasi E-Tilang pada Sistem Operasi Tablet Berbasis Android**” ini tepat pada waktunya. Laporan Tugas Akhir ini disusun sebagai persyaratan untuk mengikuti sidang Tugas Akhir program Diploma 3 jurusan Teknik Elektro konsentrasi Informasi Teknologi di Politeknik Negeri Jakarta.

Dalam penyusunan laporan ini penulis banyak menemui kendala, namun berkat adanya bimbingan dari pembimbing maka akhirnya penyusunan laporan ini dapat terlaksana dengan baik dan tepat pada waktu yang telah ditentukan.

Penulis mengucapkan terima kasih kepada :

1. Bapak Agus Wagyana, ST. MT. selaku pembimbing Tugas Akhir dari Politeknik Negeri Jakarta.
2. Keluarga penulis atas dukungan dan doanya selama penulis mengerjakan tugas akhir.
3. Teman-teman penulis terutama IT-6 yang telah mendukung penulis selama pembuatan laporan ini.

Dengan segala keterbatasan yang ada, penulis menyadari bahwa laporan ini masih banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu saran ataupun kritik yang bersifat membangun sangat diharapkan oleh penulis. Dan pada akhirnya harapan penulis laporan ini dapat bermanfaat bagi penulis sendiri dan pembaca sehingga dapat menambah pengetahuan dan wawasan.

Depok, 3 Agustus 2012

**Penulis**

## DAFTAR ISI

HALAMAN JUDUL .....	i
LEMBAR PERSETUJUAN .....	ii
LEMBAR PENGESAHAN .....	iii
ABSTRAK.....	iv
KATA PENGANTAR .....	v
DAFTAR ISI .....	vi
DAFTAR GAMBAR .....	ix
DAFTAR TABEL .....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Tujuan Penulisan.....	1
1.3 Perumusan Makalah.....	2
1.4 Batasan Masalah .....	2
1.5 Metodologi Penelitian .....	3
1.6 Sistematika Penulisan.....	4
BAB II DASAR TEORI .....	6
2.1 Bukti Pelanggaran .....	6
2.2 Mekanisme Tilang.....	6
2.2 Bahasa Pemrograman.....	7
2.2.1 PHP.....	8
2.2.2 MySQL.....	9
2.2.3 HTML .....	10
2.2 4 Javascript.....	11
2.2 5 CSS.....	12
2.3 Jaringan.....	14
2.4 Software.....	15

BAB III Perancangan dan Realisasi Sistem.....	22
3.1 Deskripsi Sistem.....	22
3.1.1 Perancangan Server.....	23
3.1.2 Client.....	23
3.2 Perancangan Database.....	24
3.3 Perancangan Server .....	28
3.3.1 Usecase diagram .....	28
3.3.1 Activity diagram Kejaksaa.....	29
3.3.1 Class diagram Kepolisian .....	28
3.4 Perancangan Client.....	35
3.4.1 Use Case Diagram.....	35
3.4.2 Activty Diagram.....	36
3.4.2 Sequence Diagram.....	36
3.4.3 Class Diagram.....	37
3.5 Realisasi Sistem .....	38
3.5.1 Realisasi pada Server.....	38
3.4.2 Realisai pada Client.....	39
BAB IV PENGUJIAN DAN ANALISIS DATA.....	41
4.1 Pengujian.....	41
4.2 Deskripsi Perangkat Pengujian.....	42
4.2.peangkat pengujian server.....	42
4.3 Prosedur Pengujian.....	43
4.3 Data Hasil Pengujian.....	44
4.4.1 Pengujian server.....	45
4.4.2 Hasil penguji pada Client.....	49
4.5 Analisis Pengujian.....	51
BAB V PENUTUP .....	52
5.1 Kesimpulan .....	52
5.2 Saran .....	52

DAFTAR PUSTAKA .....	54
LAMPIRAN .....	55



## DAFTAR GAMBAR

Gambar 2.1 PHP Logo .....	9
Gambar 2.2 MySQL.....	10
Gambar 2.3 JQuery .....	12
Gambar 2.4 Jaringan .....	15
Gambar 2.5 Logo Komodo .....	16
Gambar 2.6 <i>Tampilan komodo</i> .....	17
Gambar 2.7 Eclipse .....	17
Gambar 2.8 Photosop .....	18
Gambar 3.1 Usecase Administrator .....	29
Gambar 3.2 Activity Diagram Administrator .....	30
Gambar 3.3 Activity Diagram Eksekusi Tilang .....	31
Gambar 3.4 Activity Diagram Mmanagement Pasal .....	32
Gambar 3.5 Class Diagram Kejaksaan .....	33
Gambar 3.6 Use Case Kepolisian atau Penyidik.....	35
Gambar 3.7 Activity Diagram Client .....	36
Gambar 3.8 Class Diagram client .....	37
Gambar 3.9 Home Page Website Kejaksaan atau Pengadilan .....	38
Gambar 3.10 Form Login E-Tilang .....	39
Gambar 3.11 Form Utama E-Tilang pada sisi kepolisian.....	40
Gambar 3.12 Form Tilang E-Tilang .....	41
Gambar 4.1 Halaman Beranda pada Website .....	44
Gambar 4.2 Halaman About Me pada Website .....	45
Gambar 4.3 Halaman FAQ pada Website.....	45
Gambar 4.4 Tampilan Halaman Utama Website .....	46
Gambar 4.5 Tampilan Halaman Pasal.....	47
Gambar 4.6 Halaman Tambah Pasal pada Website .....	47
Gambar 4.7 Halaman Laporan Admin pada Website .....	48
Gambar 4.8 Halaman Tilang pada Website .....	49
Gambar 4.9 Implementasi Tilang pada Android.....	50

## DAFTAR TABEL

Tabel 3.1 Spesifikasi Server.....	23
Tabel 3.2 Spesifikasi Client .....	23
Tabel 3.3 database.....	24
Tabel 3.4 admin .....	24
Tabel 3.5 hukuman.....	25
Tabel 3.6 Polisi .....	25
Tabel 3.7 Tilang .....	25
Tabel 3.8 Sentintems.....	26
Tabel 3.1 twit .....	27
Tabel 3.1 Laporan .....	27

# BAB I

## PENDAHULUAN

### Latar Belakang

Kemajuan teknologi semakin berkembang begitu pesat terutama teknologi komputer dan telepon seluler, tingkat efektifitas dan juga efisiensi teknologi menjadi alasan dipilih sebagai alat yang digunakan dalam memenuhi kebutuhan hidup manusia. Manfaat teknologi sudah merambah ke berbagai bidang, dan juga mengubah bentuk interaksi sosial terhadap manusia. Salah satu produk yang dihasilkan dari teknologi adalah Internet dan juga komputer tablet, kedua alat tersebut merupakan hasil dari perkembangan teknologi. Internet merubah cara berkomunikasi manusia, mulai dari adanya sosial media, jual beli secara elektronik dan juga membantu manusia dalam *management* administrasi. Berbagai bidang sudah mulai memanfaatkan Internet sebagai tools sebagai contoh bidang pendidikan, bidang kesehatan, bidang pertahanan dan juga keamanan.

Aplikasi E-Tilang ini adalah digitalisasi proses tilang, dengan memanfaatkan teknologi diharapkan seluruh proses tilang akan lebih efisien dan juga efektif juga membantu pihak kepolisian dalam *management* administrasi. Aplikasi dikategorikan kedalam dua *user*, yang pertama yaitu pihak kepolisian dan yang kedua adalah pihak kejaksaan. Pada sisi kepolisian, system akan berjalan pada komputer tablet dengan system operasi Android sedangkan pada pihak kejaksaan system akan berjalan dalam bentuk website, sebagai eksekutor seperti proses sidang manual.

Aplikasi E-Tilang pada sisi kepolisian dibangun dengan menggunakan Java sebagai bahasa pemrograman untuk tablet bersystem operasi android dan akan terhubung oleh website yang berbasis PHP dengan *Codeigniter* sebagai *framework*. Mekanisasi E-Tilang mengadaptasi proses manualnya, nantinya para polisi Lalu lintas tidak akan menggunakan kertas sebagai media atau bukti pelanggaran, dengan menggunakan tablet polisi menginput data atau atribut pelanggar Lalu lintas layaknya proses manual

dan menginput nomer handphone atau email pelanggar Lalu lintas yang akan dikirim ID Tilang sebagai pengganti kertas dan reminder. Data yang sudah diinput akan masuk kedalam database, yang juga akan diakses oleh kejaksaan secara realtime. sehingga pelanggar Lalu lintas cukup datang dengan menunjukan ID Tilang yang sudah dikirim dengan Sms atau Email. Aplikasi ini dibuat untuk membantu pihak KorpLantas RI.

## **Perumusan Masalah**

Di dalam menyusun dan merancang tugas akhir ini, penyusun memiliki beberapa permasalahan, diantaranya:

1. Bagaimana merancang system yang dapat memudahkan pihak kepolisian dalam *management* administrasi berlalu lintas?
2. Bagaimana merancang sebuah website dan aplikasi yang berhubungan dengan Pihak Kepolisian dan juga Kejaksaan yaitu dalam tertib berlalu lintas?
3. Bagaimana membuat aplikasi dalam tablet berbasis android yang dapat menginput data ke dalam database website dan mengirim sms/email reminder kepada pelanggar Lalu lintas?
4. Bagaimana membuat website yang bertindak sebagai eksekutor data tilang, yang mengadaptasi fungsi kejaksaan atau pengadilan.
5. Bagaimana mengolah dan mengelola database dalam kedua aplikasi tersebut agar terhubung dan saling berinteraksi.

## **Batasan Masalah**

Dalam penulisan Tugas Akhir ini, pembatasan masalah bertujuan agar pokok pembahasan lebih terarah. Adapun pembatasan masalah dalam laporan ini yaitu:

1. Pembuatan Aplikasi E-Tilang dan website kejaksaan pengadilan sebagai prototipe.
2. Aplikasi E-Tilang digunakan untuk Input data pelanggar Lalu lintas sebagaimana form bukti pelanggaran, jenis tilang yang digitalisasi

adalah tilang yang hadir pada persidangan, bukan yang ditanggihkan melalui Bank atau Transfer.

3. Pembuatan website kejaksaan bertindak sebagai eksekutor, mensubmit data pelanggaran Lalu lintas yang tersimpan dalam database hasil input dari E-Tilang.

## **Tujuan**

Adapun tujuan dari tugas akhir ini adalah :

1. Membuat prototype system tindak pelanggaran lalu lintas (E-Tilang)
2. Membantu pihak kepolisian dalam upaya menangani tindak pelanggaran Lalu lintas dan tertib administrasi Lalu lintas

## **Metode Penyelesaian Masalah**

Adapun metode penyelesaian masalah yang telah disebutkan di atas adalah sebagai berikut :

1. Studi pustaka  
Mencari beberapa informasi dari berbagai sumber seperti internet dan mempelajari buku-buku yang berhubungan dengan tugas akhir sebagai bahan referensi untuk mengatasi dan menyelesaikan permasalahan tersebut.
2. Konsultasi dan bimbingan  
Meminta pendapat, saran dan kritik kepada pembimbing atau orang yang berkompeten dalam pelaksanaan tugas akhir ini sehingga dapat memudahkan penulis untuk menyelesaikan masalah yang ada.
3. Perancangan  
Merancang pembuatan aplikasi E-Tilang pada tablet android yang akan terhubung dengan website yang berfungsi sebagai kejaksaan yang mengeksekusi data bukti pelanggaran yang masuk seperti proses manual secara realtime.

4. Pembuatan Aplikasi dan Web  
Membuat Aplikasi E-Tilang dalam Tablet berbasis Android dan membuat website yang berfungsi sebagai pengadilan kejaksaan.
5. Pengujian dan Penerapan  
Menguji kemampuan dan penerapan dari aplikasi tersebut.

## **Sistematika Penulisan**

Sistematika penulisan ini merupakan pembahasan singkat dari setiap bab yang menjelaskan hubungan antara bab yang satu dengan bab yang lainnya, yaitu sebagai berikut :

### **BAB I PENDAHULUAN**

Pada bab ini dijelaskan mengenai latar belakang penulisan laporan dan ruang lingkup tujuan Tugas Akhir, metode pengumpulan data yang digunakan serta sistematika penulisannya.

### **BAB II TEORI DASAR**

Pada bab ini menjelaskan tentang teori umum mengenai hal-hal yang terkait dengan pembangunan Sistem Informasi.

### **BAB III PERANCANGAN DAN REALISASI**

Pada bab ini akan menjelaskan tentang perancangan Sistem Informasi pada usecase diagram, sequence diagram dan desain halaman yang akan di buat.

#### **BAB IV    PENGUJIAN DAN ANALISIS DATA**

Pada bab ini hasil pengujian dan pengoperasian aplikasi akan dilakukan apakah sesuai dengan tujuan dan akan terus dilakukan revisi bila terdapat kekurangan.

#### **BAB V    PENUTUP**

Bab ini berisikan tentang kesimpulan dan saran dari penulisan laporan ini.

## **BAB II**

### **TEORI DASAR SISTEM**

#### **2.1. Bukti Pelanggaran ( Tilang)**

Bukti Pelanggaran atau disingkat Tilang adalah denda yang dikenakan oleh Polisi kepada pengguna jalan yang melanggar peraturan. Para pengguna jalan atau mereka para pengendara kendaraan bermotor sering kali melanggar peraturan yang telah ditetapkan oleh undang-undang lalu lintas. Tilang diharapkan mampu menangani permasalahan berlalu lintas. tilang merupakan alat utama yang dipergunakan dalam penindakan bagi pelanggar peraturan-peraturan lalu lintas jalan tertentu, sebagaimana tercantum dalam Bab VI Pasal 211 sampai dengan Pasal 216 KUHAP dan penjelasannya.

Ada tiga utama fungsi tilang yaitu:

1. Sebagai surat panggilan ke Pengadilan Negeri
2. Sebagai Pengantar untuk membayar denda ke Bank / Panitera.
3. Sebagai tanda penyitaan atas barang bukti yang disita baik berupa SIM, STNK atau Kendaraan Bermotor.

Dari ketiga fungsi utama di atas, aplikasi E-Tilang tidak menerapkan fungsi sebagai pengantar untuk membayar denda ke Bank / Panitera karena mekanisme melibatkan *form* atau kertas tilang, pada E-Tilang *form* atau kertas bukti pelanggaran tidak digunakan, aplikasi ini hanya mengirim reminder berupa ID Tilang yang menyimpan seluruh data atau catatan Polisi mengenai kronologis tilang yang akan diberikan kepada pengadilan atau Kejaksaan yang memiliki website dengan integrasi database yang sama, sehingga Aplikasi ini hanya mendigitalisasi tilang pada fungsi nomor dua.



### 2.1.1 Mekanisme Tilang

Pada Saat pengguna lalu lintas terbukti melakukan kesalahan atau pelanggaran maka petugas kepolisian akan melakukan beberapa tindakan, mekanisme Tilang untuk formulir berwarna merah adalah sebagai berikut:

1. Polri menindak menggunakan formulir berwarna merah
2. Penetapan hari sidang harus memperhatikan ketetapan dari pengadilan.
3. Jelaskan kapan dan dimana pelanggar harus menghadiri sidang
4. Bila pelanggar tidak hadir, Polri wajib 2 kali memanggil dan ke 3 kalinya melakukan penangkapan.
5. Pengembalian barang bukti menunggu selesainya sidang dan setelah pelanggar membayar denda ke Panitera.

Dari mekanisme diatas, maka aplikasi E-Tilang akan mengadaptasi proses diatas, dan yang perlu diperhatikan adalah *form* tilang akan diformat dalam bentuk digital sehingga yang didapat oleh pelanggar hanya berupa ID Form Tilang yang mewakili kesemua data yang tersimpan.

## 2.2 Bahasa Pemrograman

Bahasa pemrograman adalah bahasa komputer atau teknik instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang digunakan untuk mendefinisikan sebuah program komputer. Bahasa ini memungkinkan seorang *programmer* dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan atau diteruskan dan jenis langkah apa secara persis yang akan diambil dalam berbagai situasi.

Menurut tingkat kedekatannya dengan mesin komputer, bahasa pemrograman terdiri dari:

1. **Bahasa Mesin**, yaitu memberikan perintah kepada komputer dengan menggunakan kode bahasa biner, contohnya 01100101100110.
2. **Bahasa Tingkat Rendah**, atau dikenal dengan istilah bahasa rakitan (bahasa Inggris: *Assembly*), yaitu memberikan perintah kepada

komputer dengan menggunakan kode-kode singkat (kode *Mnemonic*), contohnya MOV, SUB, CMP, JMP, JGE, JL, LOOP dan sebagainya..

3. **Bahasa Tingkat Menengah**, yaitu bahasa komputer yang memakai campuran instruksi dalam kata-kata bahasa manusia seperti pada Bahasa Tingkat Tinggi dan instruksi yang bersifat simbolik, contoh {, }, ?, <<, >>, &&, || dan sebagainya.
4. **Bahasa Tingkat Tinggi**, yaitu bahasa komputer yang menggunakan instruksi berasal dari unsur kata-kata bahasa manusia, contohnya adalah begin, end, if, for, while, and, or dan sebagainya

Sebagian besar bahasa pemrograman digolongkan sebagai Bahasa Tingkat Tinggi, hanya bahasa C yang digolongkan sebagai Bahasa Tingkat Menengah dan Assembly yang merupakan Bahasa Tingkat Rendah.

Adapun sistem informasi dan aplikasi yang dibuat pada Tugas Akhir ini adalah menggunakan bahasa pemrograman PHP untuk membangun sisi *server* dan menggunakan bahasa pemrograman Java untuk membangun *client*. Beberapa bahasa pemrograman yang digunakan pada pembuatan sistem informasi ini akan dijelaskan pada sub bab di bawah ini.

#### 2.2.1. PHP

<sup>1</sup>PHP adalah bahasa pemrograman web yang sudah terkenal dan banyak digunakan oleh para programmer, ada beberapa alasan PHP menjadi populer yaitu karena sifatnya yang *open source* dan *cross-platform*. <sup>4</sup>Pada awalnya PHP merupakan singkatan dari *Personal Home Page* yang diciptakan oleh Rasmus Lerdorf pada tahun 1994 dan kemudian berubah menjadi *PHP: Hypertext Preprocessor* oleh karena penggunaannya yang semakin merambah ke situasi yang lebih profesional, logo PHP dapat dilihat pada gambar di bawah ini

---

<sup>1</sup> Lurig, Mario. 2008. PHP Reference . USA: Peachpit Press



**Gambar 2.1 Logo PHP**

PHP mempunyai *tag – tag* khusus sama seperti bahasa pemrograman lainnya, namun dapat dikombinasikan dengan HTML, *tag* PHP dapat di tempatkan pada setiap komponen HTML, contoh sederhana penggunaanya adalah seperti gambar di bawah ini

```
<html>
<head>
<?php echo '<title>My First PHP Script</title>'; ?>
</head>
<body>
<?php echo '<p>This content was generated by PHP</p>'; ?>
<p>And this content is static HTML</p>
</body>
</html>
```

### 2.2.2 MySQL

<sup>1</sup>MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. <sup>1</sup>MySQL juga merupakan DBMS *open source* yang dikembangkan, didistribusikan dan didukung penuh oleh MySQL AB, sebuah perusahaan komersil yang didirikan oleh pengembang MySQL. Logo MySQL dapat dilihat pada gambar di bawah ini

---

<sup>1</sup> MySQL, diakses dari <http://id.wikipedia.org/wiki/Mysql>, 17 Juli 2012, 12:33

<sup>2</sup> Ullman, Larry. 2008. PHP 6 and MySQL 5. USA: Peachpit Press



Gambar 2.2 Logo MySQL

Dalam penggunaannya, MySQL sering kali dipadukan dengan PHP untuk penyimpanan data karena biasanya keduanya telah di *bundle* dalam satu paket baik dalam satu *software* (XAMPP) maupun pada *hosting*. Cara mengoperasionalkannya setelah terinstal dan membuat database, maka dalam *script* PHP kita tinggal memanggil database dengan membuat *script* seperti di bawah ini

```
<?php
$hostmysql = "localhost";
$username = "mysql_username";
$password = "mysql_password";
$database = "nama_database";
$conn = mysql_connect("$hostmysql","$username","$password");
if (!$conn) die ("Gagal Melakukan Koneksi");
mysql_select_db($database,$conn) or die
("Database Tidak Ditemukan di Server"); ? >
```

### 2.2.3 HTML

<sup>1</sup>HTML atau HyperText Markup Language (HTML) adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman web, menampilkan berbagai informasi di dalam sebuah Penjelajah web Internet dan formatting hypertext sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi.

Dengan HTML, setiap halaman *web* ditampilkan dengan struktur sintaks yang dikenali oleh *web browser* sebagai *tag* html dan menampilkannya sesuai dengan konfigurasi struktur tersebut. Cara

<sup>1</sup> HTML, diakses dari <http://id.wikipedia.org/wiki/MySQL>, 17 Juli 2012, 13:00

sintaks penggunaan HTML adalah seperti di bawah ini

```
<!DOCTYPE tml>
<html>
<body>

<h1>My First HTML</h1>

<p>My first paragraph.</p>

</body>
</html>
```

#### 2.2.4 Javascript

<sup>1</sup>JavaScript adalah bahasa skrip yang populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti Internet Explorer (IE), Mozilla Firefox, Netscape dan Opera. Kode JavaScript dapat disisipkan dalam halaman web menggunakan tag SCRIPT. JavaScript pertama kali dikembangkan oleh Brendan Eich dari Netscape dengan nama Mocha yang kemudian berganti menjadi LiveScript dan hingga kini menjadi JavaScript. JavaScript bukanlah Java, selain berbeda dari sisi pembuatnya, teknologinya pun berbeda.

Seiring dengan berkembangnya dan kebutuhan yang semakin meningkat, akhirnya John Resig, *application developer* asal Boston, Amerika Serikat, melakukan *release* sebuah *JavaScript library* dengan sebutan <sup>2</sup>**jQuery**.

---

<sup>1</sup> Javascript, diakses dari <http://id.wikipedia.org/wiki/JavaScript>, 17 Juli 2012, 13:30.

<sup>2</sup> jQuery, diakses dari <http://en.wikipedia.org/wiki/jQuery>, 17 Julis 2012, 14:03

Untuk menggunakan JQuery, user harus mengunduh terlebih dahulu, setelah selesai *file* tersebut ditaruh satu *folder* dengan *file* HTML dan jangan lupa untuk memanggil javascript yang telah diunduh seperti sintaks dibawah ini

```
<head>
<script type="text/javascript" src="jquery.js"></script>
</head>
```

Setelah memanggil, langkah selanjutnya adalah menyisipkan fungsi yang akan kita jalankan pada HTML seperti contoh di bawah ini

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $("button").click(function(){
        $("p").hide();
    });
});
</script>
</head>

<body>
<h2>This is a heading</h2>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<button>Click me</button>
</body>
</html>
```

### 2.2.5 CSS

CSS merupakan singkatan dari Cascading Style Sheet, yakni sebuah style sheet language yang digunakan untuk mengubah tampilan sebuah dokumen yang ditulis dalam markup language (HTML). CSS telah distandarisasi oleh World Wide Web Consortium (W3C) sebagai teknologi untuk melakukan format terhadap dokumen HTML. Berikut ini adalah contoh penggunaan CSS yang telah tergabung dengan HTML

```
<!DOCTYPE html>
<html>
```

```
<head>
  <style type="text/css">
    body {
      background-color:#d0e4fe;
    }

    h1 {
      color:orange;
      text-align:center;
    }

    p {
      font-family:"Times New Roman";
      font-size:20px;
    }
  </style>
</head>
<body>
  <h1>CSS example!</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

## 2.2.6 Java

<sup>1</sup>Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal.

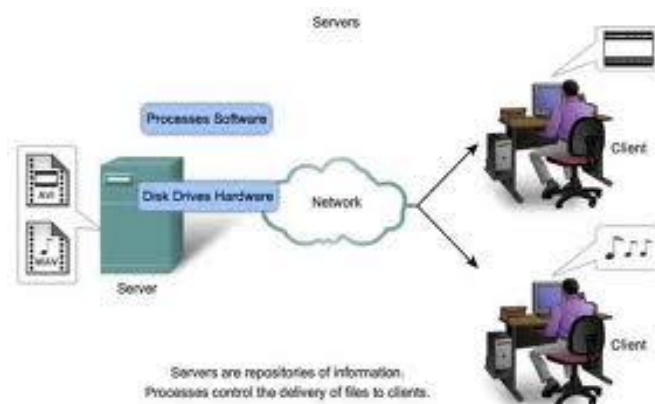
---

<sup>1</sup> Java, diakses dari <http://en.wikipedia.org/wiki/Java>, 17 Julis 2012, 15:03

## 2.3 Jaringan

<sup>1</sup>Jaringan komputer adalah sebuah sistem yang terdiri atas komputer-komputer yang dirancang untuk dapat saling berbagi sumber daya seperti *printer* dan CPU, berkomunikasi melalui surat elektronik ataupun pesan instan, dan dapat mengakses informasi melalui peramban *web* dan sebagainya.

Dua buah komputer yang masing-masing memiliki sebuah *Network Interface Card* (NIC) kemudian dihubungkan melalui kabel maupun nirkabel sebagai medium transmisi data, dan dengan dukungan perangkat lunak sistem operasi jaringan akan membentuk sebuah jaringan komputer yang sederhana. Apabila ingin membuat jaringan komputer yang lebih luas lagi jangkauannya, maka diperlukan peralatan tambahan seperti Hub, Bridge, Switch, Router dan Gateway sebagai peralatan interkoneksinya. Ilustrasi jaringan bisa digambarkan seperti di bawah ini:



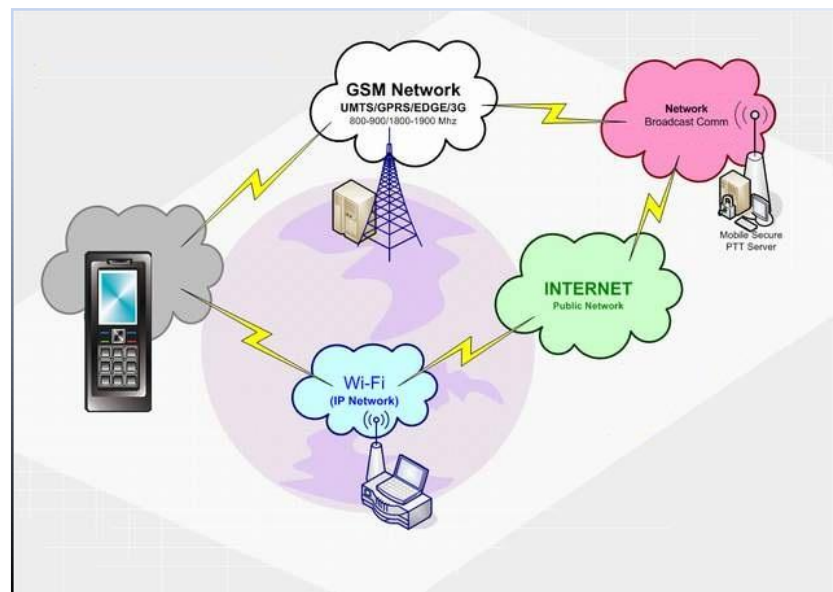
**Gambar 2.3 Ilustrasi jaringan**

Semua permintaan atau lalu lintas jaringan dari komputer client harus dilewatkan ke komputer server karena server yang mengatur pelayanannya. Juga diperlukan adanya pembagian tugas

<sup>1</sup>Jaringan, diakses dari [http://id.wikipedia.org/wiki/Jaringan\\_komputer](http://id.wikipedia.org/wiki/Jaringan_komputer), 17 Juli 2012, 15:30.



Tidak jauh berbeda dengan jaringan yang diterapkan pada komputer, jaringan pada *handphone* juga menggunakan metode yang sama, yakni terhubung melalui jaringan GSM lalu terhubung ke sebuah *Broadcast Communication Network* lalu terhubung ke *internet*. Tidak jauh berbeda apabila *handphone* terhubung dengan jaringan *Wi-Fi*, hanya saja komunikasi langsung diarahkan ke *internet*, tidak melalui *Broadcast Communication Network* terlebih dahulu.



**Gambar 2.4 Ilustrasi Jaringan pada Handphone**

## 2.4 Software

Dalam membuat atau merancang suatu Program atau suatu Sistem Informasi, para pengembang membutuhkan alat atau *tools* yang dapat mewujudkan apa yang mereka inginkan, hal ini tidak terlepas dengan adanya *software*. Software tersebut meliputi level-level pada setiap perancangan atau pembuatan Sistem Informasi, diantaranya berupa *text editor*, *image editor*, *UML* dan sebagainya. Adapun *software* yang digunakan untuk membangun sistem informasi ini akan dijabarkan pada sub bab di bawah.

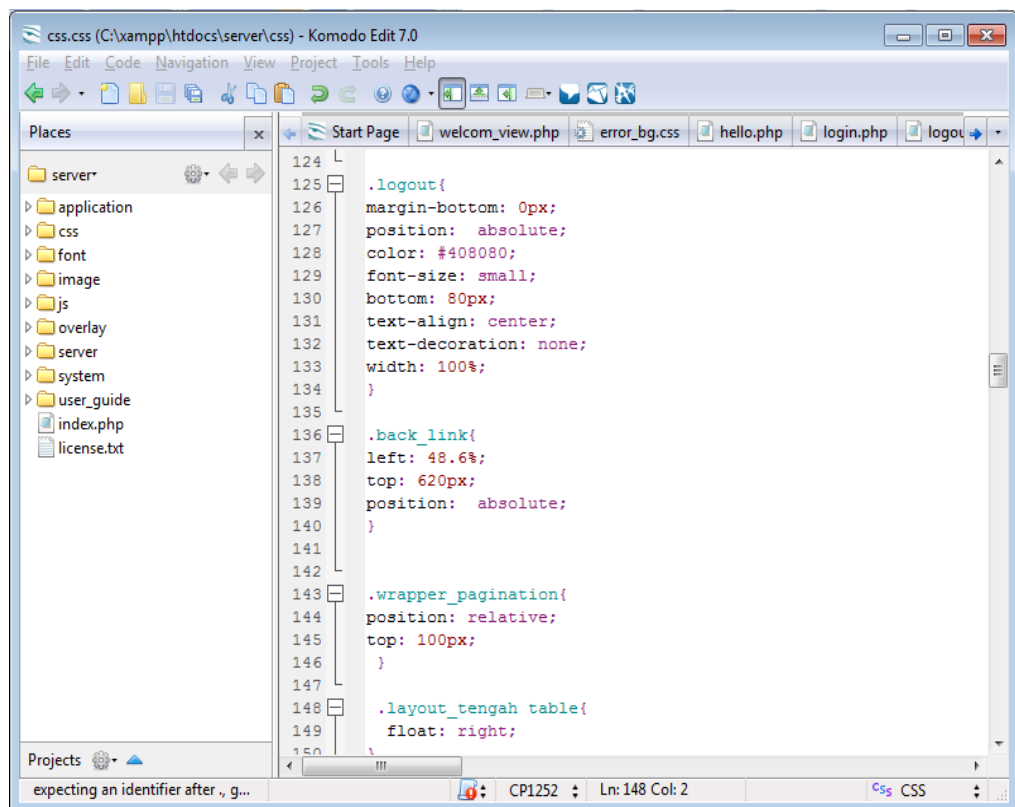
### 2.4.1 Komodo Editor

Komodo Edit merupakan salah satu pilihan terbaik bagi seorang programmer. Khususnya programmer web, baik itu PHP, perl, python, dan lain sebagainya.



**Gambar 2.5 Logo Komodo Editor**

Komodo memiliki beberapa kelebihan dari text *editor* lainnya, yang sangat spesial dari program ini adalah fitur autocomplete dan calltips yang sangat memudahkan kita yang mungkin tidak ingat tentang variabel apa yang harus dimasukkan dalam fungsi tertentu.



**Gambar 2.6 Tampilan Komodo Editor**

### 2.4.2 IDE Eclipse Helios

<sup>1</sup>Eclipse adalah sebuah *Integrated Development Environment* yang biasa digunakan sebagai *text editor* untuk bahasa pemrograman Java. Eclipse awalnya merupakan proyek milik IBM dan hingga November 2001, Eclipse diresmikan sebagai *Free and Open-Source Software* guna pengembangan lebih lanjut oleh Borland, IBM, Merant, QNX Software Systems, Rational Software, Red Hat, SuSE, TogetherSoft dan WebGain sebagai komunitas pengembang utamanya.



**Gambar 2.7 Logo Eclipse**

Eclipse juga menyediakan fitur layaknya *plug-in* fitur ini dapat menambahkan beberapa aplikasi atau menginstalasi emulator yang dapat digunakan untuk mengembangkan aplikasi handphone.

### 2.4.3 Adobe Photoshop

<sup>2</sup>Adobe Photoshop adalah perangkat lunak editor citra buatan Adobe Systems yang dikhususkan untuk pengeditan foto/gambar dan pembuatan efek.

---

<sup>1</sup> *Eclipse (Software)*, diakses dari [http://en.wikipedia.org/wiki/Eclipse\\_\(software\)](http://en.wikipedia.org/wiki/Eclipse_(software)), 18 Juli 2012, 04:41

<sup>2</sup> *Adobe Photoshop (Software)*, diakses dari [http://id.wikipedia.org/wiki/Adobe\\_Photoshop](http://id.wikipedia.org/wiki/Adobe_Photoshop), 18 Juli 2012, 04:41

pada tahun 1987 seorang mahasiswa PhD di Universitas Michigan, Thomas Knoll membuat sebuah program di platform Mac Plus untuk menampilkan gambar, yang pada waktu itu adalah layar monokrom. John Knoll (saudaranya) tertarik dan bergabung dengannya untuk mengembangkan software yang mereka namakan "Display". Program ini kemudian di demonstrasikan ke pihak Apple Computer pada tahun 1988 yang waktu itu nama program sudah berubah menjadi Photoshop.



**Gambar 2.8 Logo Photoshop**

Photoshop dikenal karena fiturnya yang hampir lengkap dibanding dengan image editor lainnya, Perangkat lunak ini banyak digunakan oleh Fotografer Digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (market leader) untuk perangkat lunak pengolah gambar.

#### **2.4.4. XAMPP 1.7.7 VC9**

<sup>1</sup>XAMPP adalah sebuah *Free Open Source Software* yang dapat berjalan pada jenis sistem operasi apapun (*cross-platform*). XAMPP merupakan susunan kata dari X (baca: *cross* - Inggris) yang artinya *cross-platform*, A untuk *Apache HTTP Server*, M untuk *MySQL*, P untuk *PHP* dan P untuk *Perl*. Dari akronim tersebut dapat disimpulkan bahwa XAMPP merupakan sebuah paket *web server* yang telah di-*bundle* menjadi satu paket dan dapat berjalan untuk *platform* apapun

---

<sup>1</sup> XAMPP, diakses dari <http://en.wikipedia.org/wiki/xampp>, 18 Juli 2012, 00:21



**Gambar 2.9 Logo XAMPP**

Dengan *web server* ini, seorang *programmer* dapat melakukan *development* terhadap aplikasi *web* yang ia inginkan, tentunya dengan keunggulan utamanya yakni sifatnya yang *open-source* dan gratis.

Di bawah ini akan dijelaskan tentang fitur-fitur yang digunakan untuk membangun dan mengembangkan Tugas Akhir ini.

### **1. Apache**

<sup>1</sup>Apache adalah sebuah *web server software* yang memainkan peranan penting pada perkembangan *World Wide Web*. Apache dikembangkan dan dipelihara oleh komunitas pengembang terbuka di bawah pengawasan dan bantuan Apache Software Foundation. Apache dibuat berbasiskan kode dari <sup>17</sup>NCSA HTTPd, yaitu sebuah *web server* paling awal yang dikembangkan di NCSA oleh Robert McCool namun dengan pengembangan lebih lanjut.



**Gambar 2.10 Logo Apache**

---

<sup>1</sup> Apache HTTP Server, diakses dari [http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server), 18 Juli 2012, 00:31

### **2. MySQL**

Seperti yang telah dijelaskan sebelumnya pada sub bab 2.1.2, MySQL adalah DBMS yang dikembangkan oleh MySQL AB dan MySQL adalah sebuah basis data yang berada dalam paket *web server* XAMPP.

#### 2.4.5 Start UML

<sup>1</sup>StarUML adalah sebuah alat bantu untuk membuat bagan UML (*Unified Modeling Language*) untuk dapat menjelaskan pengertian, maksud dan cara kerja sebuah sistem. StarUML merupakan *software open source* yang dikembangkan oleh Plastic Software dengan kerjasamanya dengan komunitas *open source*,



Gambar 2.11 Logo Start UML

#### 2.4.6 Codeigniter

CodeIgniter merupakan salah satu dari sekian banyak framework PHP yang ada. Salah satu kelebihan Codeigniter adalah menggunakan MVC ( model – view – controller ) yaitu sebuah metode yang sangat memudahkan para pengembang, karena memisahkan antara data (model) tampilan (view) dan proses (controller).

---

<sup>1</sup> *StarUML*, diakses dari <http://en.wikipedia.org/wiki/StarUML>, 18 Juli 2012, 01:28

#### 2.4.7 GAMMU

GAMMU merupakan *software SMS gateway* yang cukup bagus dan terkenal selain mudah penggunaannya, perangkat modem gsm yang support cukup banyak mulai dari Nokia, Siemen dan Sony Ericsson. Selain itu perangkat lain yang lebih cocok untuk dijadikan SMS gateway dengan software gammu seperti modem *gsm itegno*. GAMMU bahkan sudah menyediakan service *online* untuk proses *update* data SMS ke database. Database yang disupport GAMMU adalah MySQL.

Ada dua mekanisme kerja dari Gammu yaitu sebagai Aplikasi dan Sebagai Deamon. Gammu sebagai Aplikasi akan bekerja ketika perintah Gammu di jalankan pada lingkungan shell beserta perintahnya di sertakan sesuai fungsi yang di inginkan. Sedangkan sebagai deamon gammu di tandai dengan di jalankannya perintah *smsd* pada shell. *Smsd* bukan lah perintah yang langsung terinstal melainkan perintah yang di jalankan pada shell atau MS-Dos Prompt.

Pada dasarnya gammu mampu didukung oleh hampir sebagian besar handphone yang ada di pasaran, namun ada beberapa spesifikasi handphone yang tidak dapat didukung sepenuhnya oleh gammu, hal ini karena pada gammu tidak mendukung semua jenis koneksi yang terdapat pada handphone tersebut.

## **BAB III**

### **PERANCANGAN DAN REALISASI**

#### **3.1 Deskripsi Sistem**

Sistem ini merupakan Digitalisasi daripada proses Tilang ( Bukti Pelanggaran ),system tilang yang diadaptasi adalah jenis Tilang yang konvensional atau system tilang yang datang kepersidangan, semua proses input atau *form* tilang akan didigitalisasikan dan tersubmit dalam database, dan pelanggar lalu lintas akan diberikan reminder atau bukti pelanggaran berupa ID tilang yang dikirim dengan teknologi SMS Gateway. untuk sisi Kejaksaan yang diibaratkan sebagai *server* menggunakan website sebagai eksekutor, sehingga pelanggar lalu lintas hanya tinggal menunjukan ID Tilang kepada kejaksaan yang akan meload seluruh data. Nantinya akan terlihat semua kronologi dan juga denda yang harus dibayar.

Nama Sistem : Aplikasi E-Tilang pada Tablet System Operasi Berbasis Android.

Fungsi Sistem : Aplikasi ini digunakan untuk membantu para Kepolisian Jakarta Timur menyelesaikan masalah tertib administrasi berlalu lintas, dengan membuat prototype system informasi tilang pada sisi kepolisian dan juga pada sisi kejaksaan, Aplikasi E-Tilang nantinya akan berinteraksi dengan website kejaksaan dengan berbasis PHP yang menggunakan framework Codeigniter. Aplikasi ini diharapkan dapat memudahkan kepolisian dalam mengatasi masalah berlalulintas dan juga mengurangi adanya penyimpangan dalam tindak pelanggaran lalu lintas..



### 3.1.1 Perancangan *Server*

*Server* atau Kejaksaan yang digunakan pada Tugas Akhir kali ini berjalan pada Localhost. Spesifikasi pada server dapat dilihat pada tabel 3.1 di bawah ini:

**Tabel 3.1 Spesifikasi *Server***

Spesifikasi	Keterangan
Port	Localhost:8080
PHP	Tersedia
MySQL	Tersedia
Web Mail	Tersedia

### 3.1.2 *Client*

*Client* atau pada sisi Kepolisian adalah adalah sebuah *handphone* dengan sistem operasi Android berjalan pada platformnya. Tidak dibutuhkan banyak spesifikasi khusus pada sisi *client*. Spesifikasi pada *client* dapat dilihat pada tabel 3.2 di bawah ini:

**Tabel 3.2. Spesifikasi *Client***

Spesifikasi	Keterangan
Handphone dengan OS Android 2.3	Tersedia
Support GPS	Tersedia

### 3.2 Perancangan Database

Database pada sisi *Server* atau Kejaksaan menggunakan database MySQL. Berikut adalah detail tabel yang tersedia pada *server*.

**Tabel 3.3. Daftar Tabel**

Nama Tabel	Keterangan
Admin	Berisi mengenai data Admin, sudah ditetapkan
Hukuman	Berisi mengenai Pasal,keterangan dan besaran denda yang harus dibayar pelanggar
Polisi	Berisi mengenai data Polisi atau Petugas
Tilang	Berisi mengenai data pelanggar yang melanggar lalu lintas, yang telah
Sentitems	Berisi mengenai pesan yang telah terkirim kepada si pelanggar
Twit	Berisi mengenai laporan singkat tentang keadaan lalu lintas dimana polisi bertugas.
Laporan	Berisi mengenai laporan, atau report detail dari data yang masuk.

**Tabel 3.4 Tabel Admin**

Nama Kolom	Tipe Data	Length	Keterangan
Userid	Int	1	ID admin
Username	Varchar	5	Nama Admin
userpass	Varcha	5	Paswword

**Tabel 3.5 Hukuman**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Length</b>	<b>Keterangan</b>
Pasal	varchar	50	Berisi Pasal yang berkaitan dengan Pelanggaran
Keterangan	text		Keterangan atau deskripsi pasal
Kategori	Char	200	Kategori atau jenis kendaraan apa yang terkena sanksi
Nominal	Int	11	Nominal denda yang harus dibayar

**Tabel 3.6 Polisi**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Length</b>	<b>Keterangan</b>
ID_Polisi	Varchar	10	ID Polisi
Nama_Polisi	Char	50	Nama Polisi
Gender	Char	10	Jenis Kelamin
Daerah_Tugas	Varchar	50	Daerah tugas Polisi
Jabatan	Varchar	50	Jabatan

**Tabel 3.7 Tilang**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Length</b>	<b>Keterangan</b>
ID_Tilang	Varchar	100	ID Tilang
ID_Polisi	Varchar	10	Id Polisi yang menyidik
No_SIM	Varchar	20	Nomer SIM pelanggar lalulintas
Nomer_STNK	Varchar	20	Nomer STNK kendaraan pelanggar
Pasal	Varchar	10	Pasal yang dilanggar
Keterangan			Deskripsi Pasal yang

	Text		dilanggar
Lokasi	Varchar	1000	Tempat penilangan
Tanggal	timestamp		Waktu penilangan

**Tabel 3.8 Sentitems**

Nama Kolom	Tipe Data	Length	Keterangan
UpdatedInDB	timestamp		Waktu update setelah pergantian antara terkirim dengan laporan
InsertIntoDB	timestamp		Waktu masuknya pesan kedalam tabel Sentitems
SendingDateTime	timestamp		Waktu pengiriman pesan
DeliveryDateTime	timestamp		Laporan terkirim nya pesan sampai kepada pelanggan
Text	text		Berisi isi pesan yang di dekrip
DestinationNumber	Varchar	20	Nomer tujuan
Coding	enum		
UDH	-	-	-
SMSCNumber	Varchar	20	Nomer <i>server</i>
Class	int	11	Status
TextDecoded	varchar	300	Isi pesan
ID	int	10	ID
SenderID	varchar	255	ID profil
SequencePosition	Int	11	Posisi Pesan
Status	enum		Status Pesan
StatusError	Int	11	Status Error
TPMR	Int	11	

RelativeValidity	int	11	-
CreatorID	text	-	ID Profil Sms

**Tabel 3.9 Twit**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Length</b>	<b>Keterangan</b>
ID_Twit	Varchar	11	ID Twit
Twit	Varchar	1000	Isi twit
ID_Polisi	Varchar	10	ID Polisi yang memposting

**Tabel. 3.10 Laporan**

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Length</b>	<b>Keterangan</b>
ID_Laporan	Varchar	10000	ID Laporan
ID_Tilang	Varchar	100	ID Tilang
No_SIM	Varchar	20	Nomer SIM pelanggar
Nomer_STNK	Varchar	20	Nomer Kendaraan Pelanggar
Pasal	varchar	50	Pasal yang dilanggar
Keterangan	Text		Deskripsi pasal yang dilanggar
Lokasi	Varchar	1000	Lokasi tempat pelanggaran lalu lintas
Tanggal	Timestamp		Tanggal terjadinya Pelanggaran
ID_Polisi	Varchar	10	ID Polisi yang bertugas
Nama_Polisi	Char	50	Nama Polisi atau penyidik
Jumlah_Tilang	Int	255	Jumlah Tilang pelanggar lalu lintas apabila lebih dari satu kali.

### 3.3 Perancangan Server

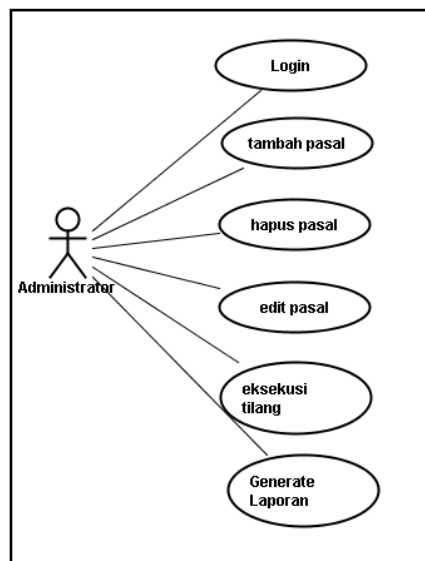
Pada sub bab ini akan dijabarkan tentang rancang bangun *server* dengan menggunakan *Use Case Diagram*, *Class Diagram* dan *Activity Diagram* dan tabel yang berisi tentang *file* yang digunakan .

#### 3.3.1 Use Case Diagram

*Use Case Diagram* adalah diagram yang digunakan untuk menjelaskan keterkaitan *user* sebagai pengguna sistem dan sistem sebagai penyedia fungsi layanan. Pada diagram ini dijelaskan fungsi-fungsi apa saja yang bisa dijangkau oleh seorang *user* sebagai pengguna sistem.

##### 3.3.1.1 Administrator

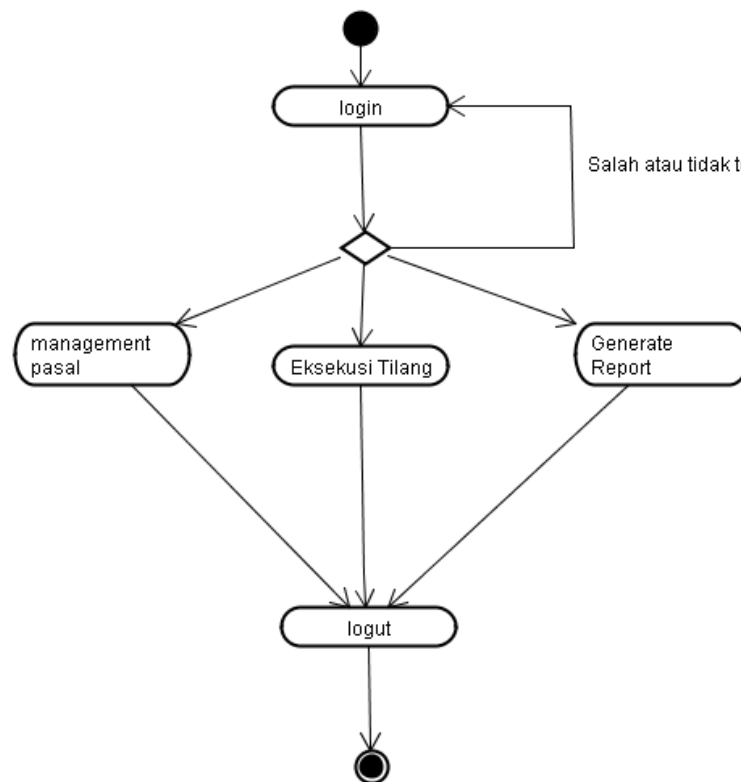
Pada sisi *server* atau Kejaksaan hanya ada User tunggal yang menjadi *Super Administrator*, Admin ini menjalankan fungsi – fungsi utama dalam sistem. Gambar 3.1 menunjukan fungsi – Fungsi apa saja yang dapat dilakukan oleh seorang Administrator.



Gambar 3.1 Usecase Administrator

#### 3.3.2 Activity Diagram Kejaksaan atau Pengadilan

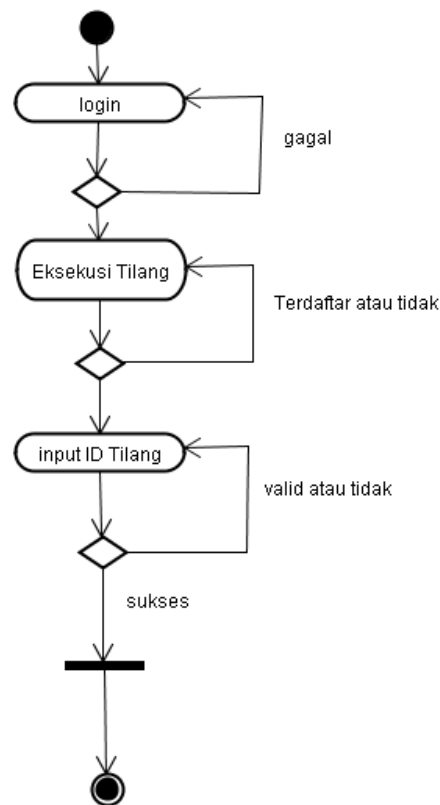
*Activity Diagram* adalah diagram yang digunakan untuk menggambarkan alur kerja suatu aktivitas. Gambar 3.5 menjelaskan *Activity Diagram* dari sisi Kejaksaan atau yang diibaratkan sebagai *Server*.



**Gambar 3.2 Activity Diagram Administrator**

Gambar di atas menunjukkan aktivitas ketika awal mengakses system pada sisi kejaksaan atau bertindak sebagai *server*, diawal admistrator harus login terlebih dahulu, adminsitrator diwajibkan mengisi username dan *password* terlebih dahulu. Apabila salah satu diantaranya salah maka akan kembali pada form yang sama validasi ini berlaku hingga berhasil. Apabila login berhasil maka Administrator akan masuk kedalam *page* utama.

Setelah masuk pada *page* utama, Administrator akan menjalankan fungsi utama, secara besar ada tiga fungsi utama pada sisi kejaksaan atau server, gambar 3.3 di bawah ini akan menunjukkan *activity* diagram untuk menjalankan fungsi utama yaitu eksekusi tilang

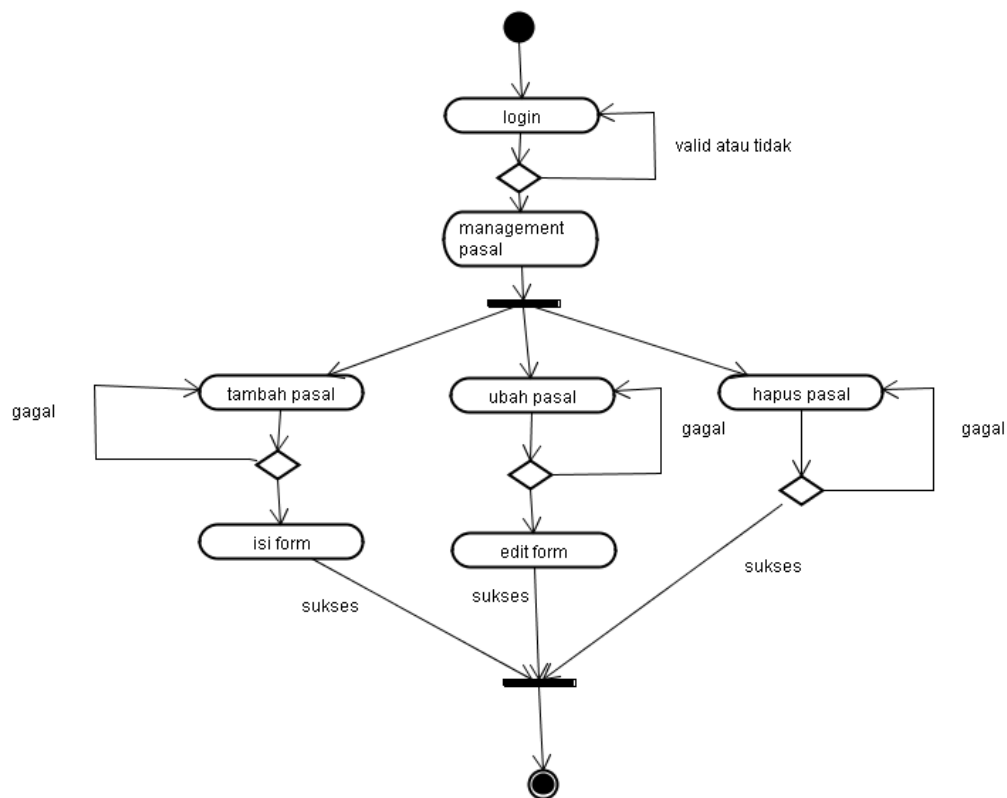


**Gambar 3.3. Activity Diagram Eksekusi Tilang**

Pada gambar di atas administrator harus login terlebih dahulu, setelah login, dan memilih page eksekusi tilang selanjutnya, administrator menginput ID Tilang yang diberikan dari si pelanggar lalu lintas, apabila ID tersebut ada atau tercantum maka eksekusi tilang bisa dilakukan, jika tidak tercantum atau terdaftar maka system akan memberi tau bahwa ID tidak ada.

Setelah itu, pada sisi kejaksaan atau server juga ada proses *CRUD Create Read Update Delete*, proses ini berada pada *management* pasal, dimana seorang administrator dapat menambah, mengedit atau menghapus pasal yang nantinya akan digunakan sebagai pasal tindak pidana melanggar lalu lintas, gambar 3.4 di bawah ini menjelaskan *activity* diagram administrator pada level *management* pasal.



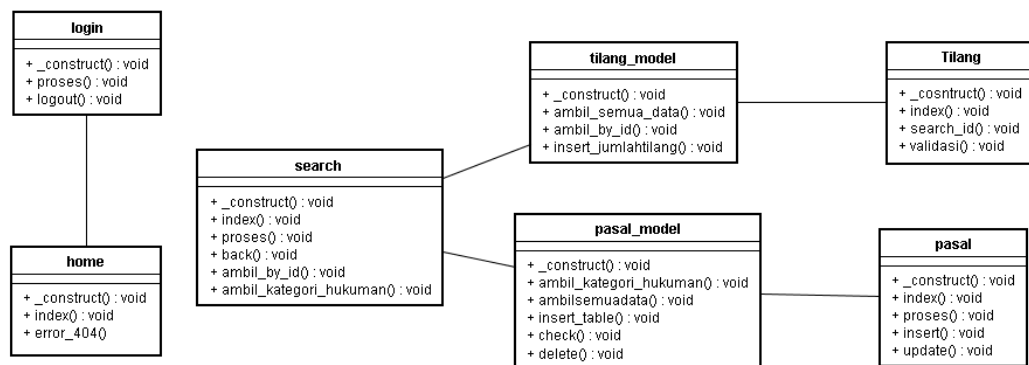


**Gambar 3.4 Activity Diagram Mmanagement Pasal**

Pada gambar di atas, adalah proses management pasal untuk sisi kejaksaan atau server dimana nantinya pasal ini akan digunakan sebagai acuan pada sisi kepolisian atau *client*, setelah sukses login maka administrator akan menuju halaman pasal, administrator bisa menambah pasa dengan mengisi form samapi valid, jika berhasil maka pasal akan bertambah. Sama halnya dengan ubah atau *update* dan juga hapus.

### 3.3.3. Class Digaram Pihak Kepolisian atau Penyidik

*Class Diagram* adalah diagram yang menggambarkan bagaimana sebuah *class* terhubung dengan *class* yang lain. Dalam pembuatan Tugas Akhir ini, dibuat 7 *class* untuk membangun sisi *server*, masing-masing 5 *Controller* dan 2 *Model*. Gambar 3.5 menjelaskan tentang *Class Diagram* yang terdapat pada sisi *server*.



**Gambar 3.5 Class Diagram Kejaksaaan**

### 3.3.4 File

Berikut adalah daftar file yang digunakan untuk membangun *server* Tugas Akhir ini

#### 3.3.4.1 Controller

*Controller* adalah bagian yang mengatur pemrosesan pada model MVC. Di bawah ini adalah *Controller* yang digunakan.

**Tabel 3.11 Daftar File pada Controller**

Nama File	Keterangan
login.php	<i>Controller</i> untuk menangani pemrosesan bagian “Login”
Home.php	<i>Controller</i> untuk menangani pemrosesan pada tampilan dan validasi page utama
Search.php	<i>Controller</i> untuk menangani pemrosesan fungsi pencarian
Tilang.php	<i>Controller</i> untuk menangani pemrosesan yang terkait dengan <i>tilang</i>
Pasal.php	<i>Controller</i> untuk menangani pemrosesan yang terkait dengan <i>Pasal</i>

#### 3.3.4.2 Model

*Model* adalah bagian yang mengatur pemrosesan pengambilan dan pengiriman data ke database pada model MVC. Tabel di bawah adalah file model.

**Tabel 3.12. Daftar File pada Model**

Nama File	Keterangan
Tilang_model.php	<i>Model</i> untuk memproses data ketika melihat menjalankan fungsi Tilang
Pasal_model.php	<i>Model</i> untuk memproses data ketika ingin menambah,edit dan hapus pasal

#### 3.3.4.3 View

*View* adalah bagian yang berisi tampilan / *user interface* sebuah aplikasi. Tabel 3.13 di bawah adalah file view

**Tabel 3.13. Daftar File pada View**

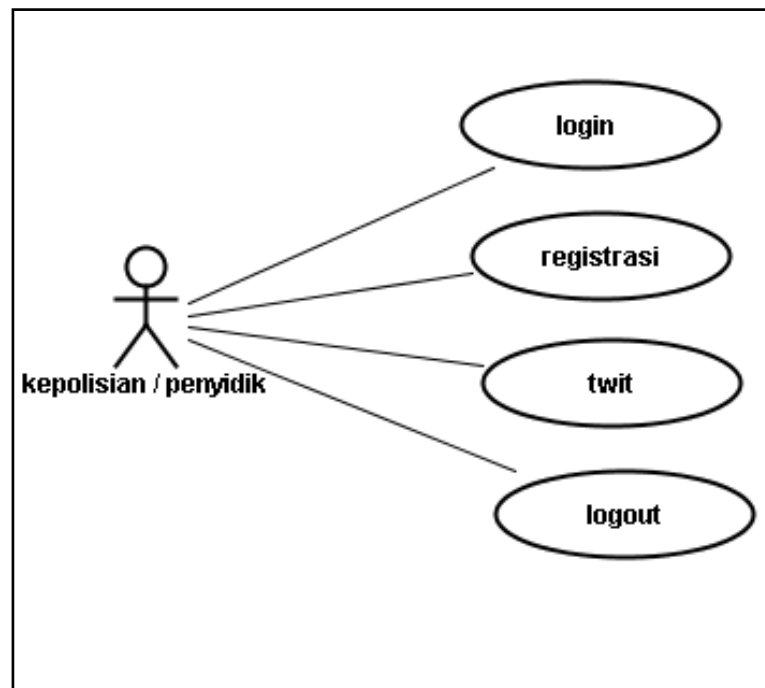
Nama File	Keterangan
Hello.php	Tampilan login
Welcome_view.php	Tampilan utama setelah berhasil login
Form_tilang_v.php	Tampilan page tilang
Search_v.	Tampilan pada halaman search
Tambah_pasal.php	Page untuk menampilkan form tambah pasal
Pesanter kirim.php	Menampilkan laporan pesan sms
laporan	Menampilkan laporan data
Footer.php	Menampilkan footer
Aboutme.php	Menampilkan deskripsi website
Faq.php	Menampilkan pertanyaan seputar website
Isi_insert_pasal.php	Menampilkan form tambah pasal

### 3.4 Perancangan Client

Pada sub bab ini akan dijabarkan tentang rancang bangun *client* dengan menggunakan *Use Case Diagram*, *Activity Diagram* dan *Class Diagram* dan tabel yang berisi tentang *file* yang digunakan.

#### 3.4.1 Use Case Diagram

Pada sisi *client* atau kepolisian kali ini, user atau pengguna tunggal, yaitu para petugas lalu lintas. Gambar 3.6 di bawah ini adalah menggambarkan use case pada client atau kepolisian.

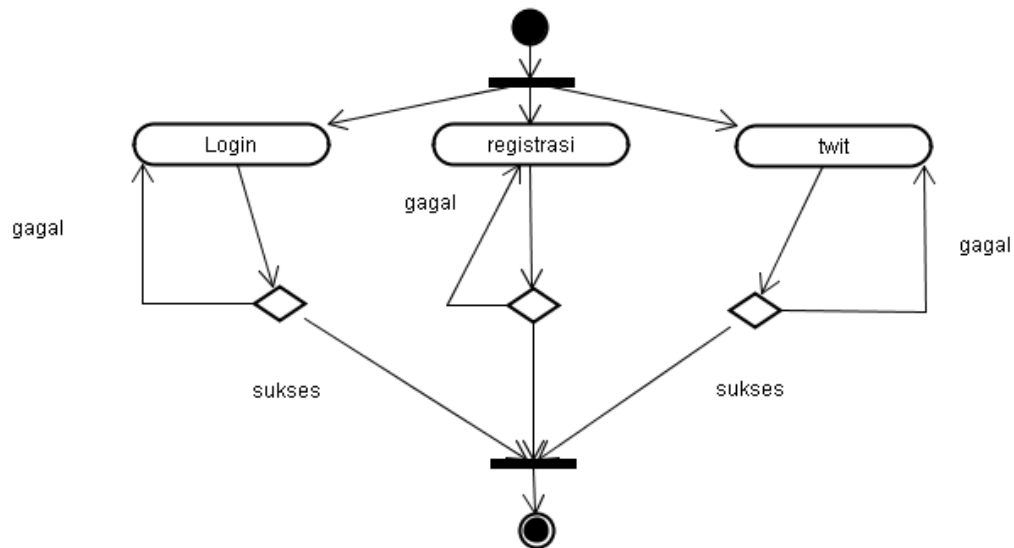


**Gambar 3.6 Use Case Kepolisian atau Penyidik**

User dapat melakukan empat hal utama, mereka tidak dapat melakukan pengaturan atau pengubahan *username* dan *password* karena parameter yang digunakan adalah nomor induk karyawan mereka dan nama mereka sehingga tidak mungkin melakukan perubahan.

### 3.4.2 Activity Diagram Kepolisian atau Penyidik

*Activity Diagram* pada *Client* juga sama seperti *Use Case* memiliki perbedaan dengan yang ada pada *server*. Berikut akan dijelaskan beberapa aktivitas utama pada *client*. *Activity Diagram* pada sisi kepolisian atau penyidik bis dilihat seperti gambar 3.7 di bawah ini

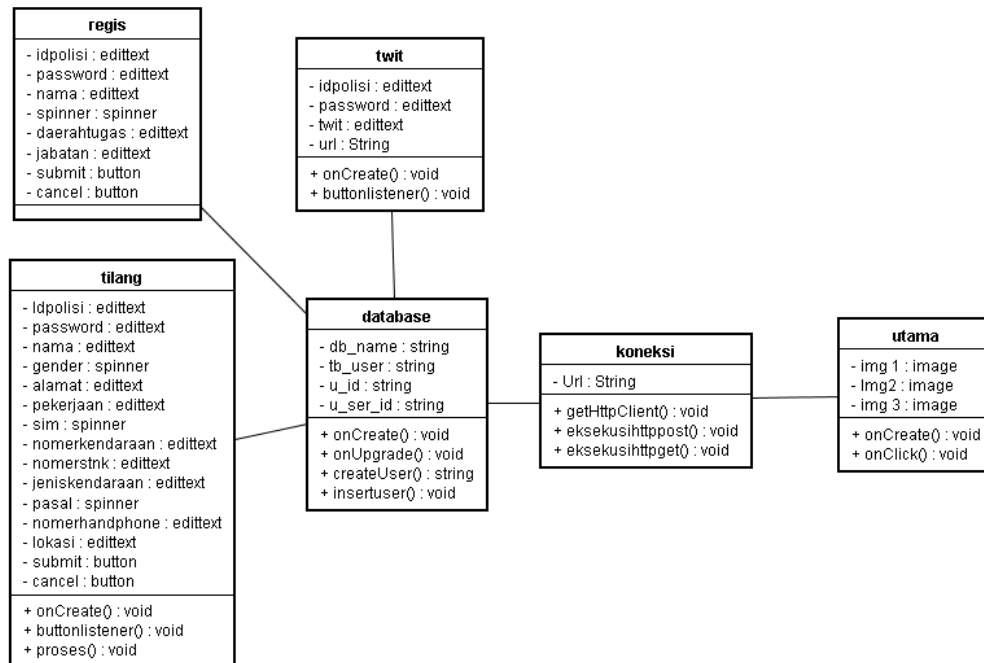


**Gambar 3.7 Activity Diagram Client**

Pada gambar di atas, ada tiga fungsi utama pada kepolisian atau penyidik. Registrasi dilakukan jika penyidik belum memiliki *account*, parameter yang penting adalah nomer induk karyawan dan nama penyidik yang akan diniai sebagai username dan *password*. Jika sudah memiliki *account* maka dapat melakukan login. Penyidik juga dapat melakukan *Twit* atau semacam laporan singkat mengenai keadaan lalu lintas yang akan diterima oleh server.

### 3.4.3 Class Diagram

Terdapat cukup banyak *class* yang dipergunakan untuk membuat aplikasi *client* ini karena untuk mempermudah *management* fungsi, setiap *method*-nya dipisahkan kedalam *class* yang berbeda-beda. Di bawah ini akan diberikan *Class Diagram* tentang aplikasi *client* ini



Gambar 3.8 Class Diagram client

### 3.5 Realisasi System

Pada tahap realisasi ini akan dibagi menjadi dua, yakni realisasi pada *server* dan realisasi pada *client*.

#### 3.5.1 Realisasi pada Server

Telah dijelaskan sebelumnya bahwa sisi *server* dibangun dengan menggunakan PHP dan bantuan *framework* CodeIgniter yang berjalan pada *web server* Apache menggunakan database MySQL. Gambar 3.9 di bawah ini adalah *home page* dari *website* kejaksaan



**Gambar 3.9 Home Page Website Kejaksaan / Pengadilan**

Form login berada pada logo utama, dengan menggunakan javascript sebagai validasinya, jika berhasil login maka akan masuk kebagian page utama, form login dapat dilihat pada gambar 3.10 di bawah ini



**Gambar 3.10 Form Login**

Setelah berhasil login maka akan langsung masuk kedalam *page* utama yang . *Interface* dibuat dengan mengadopsi tema *simple* agar tidak *user friendly* dan juga tidak menggunakan sumber daya yang besar

### **3.5.2 Realisasi pada Client**

Pada sisi *client*, aplikasi dibangun menggunakan bahasa pemrograman Java dikolaborasikan dengan file berekstensi xml untuk membangun *user interface* atau *layout* aplikasi *client* ini. Di bawah ini merupakan beberapa tampilan yang disuguhkan pada *client*.





**Gambar 3.11 Form Utama E-Tilang pada sisi kepolisian**

*Design* yang sederhana membuat pengguna mudah untuk mengoperasionalkannya, pada saat pertama kali mengakses Aplikasi E-Tilang petugas atau penyidik dihadapkan dengan tiga pilihan yaitu, Tilang untuk menjalankan fungsi utama, Registrasi bagi yang tidak memiliki *account* dan Info untuk melaporkan keadaan lalu lintas sekitar yang akan diterima di sisi kejaksaan, berupa laporan singkat.

Jika penyidiki sudah registrasi sebelumnya maka langsung mengakses menu utama dan juga bisa posting laporan singkat mengenai keadaan sekitar. Gambar 3.12 di bawah ini menunjukkan form utama

The image shows a mobile application interface for 'E-TILANG'. It features a series of text input fields for user registration or login. The fields are labeled: 'ID Polisi', 'Password', 'Nama', 'Pria' (with a dropdown arrow), 'Alamat', 'pekerjaan', 'SIM A' (with a dropdown arrow), 'Nomer Kendaraan' (which is highlighted with an orange border), 'Nomer STNK', and 'Senedah Motor' (partially visible at the bottom). The status bar at the top of the screen displays the time '4:51 AM' and various system icons.

**Gambar 3.12 Form Tilang E-Tilang.**

Pada form di atas, terdapat *field* yang wajib diisi oleh penyidik yaitu ID Polisi dan *Passwordnya*, jika tidak diisi maka tidak akan berjalan. Setelah terisi dengan benar maka kesemua data akan masuk kedalam database dan pelanggar lalu lintas akan mendapat sms berupa ID Tilang.

## **BAB IV**

### **PENGUJIAN DAN ANALISIS DATA**

#### **4.1 Pengujian**

Sebelum memastikan bahwa Tugas Akhir ini tidak terdapat *error* atau *bug*, Penulis melakukan pengujian dalam segala sisi, baik dalam otorisasi *user*, *input*, *output*, maupun pemrosesan data. Pengujian meliputi *web* pada *server* dan aplikasi pada *client*.

Tujuan dari pengujian ini adalah untuk mengetahui bahwa menu yang tersedia dapat diakses dengan baik dan fungsi utama dapat berjalan dengan benar.

Adapun pengujian dilakukan pada perangkat yang Penulis miliki, sehingga belum ada pembiayaan lebih lanjut mengenai perangkat pengujian

#### **4.2 Deskripsi Perangkat Pengujian**

Perangkat pengujian akan meliputi 2 sisi, yakni halaman *web* pada *server* dan aplikasi pada *client*.

##### **4.2.1 Perangkat Pengujian untuk Server**

Berikut ini akan dijabarkan mengenai spesifikasi perangkat pengujian pada sisi *server* yang dilakukan oleh Penulis. Spesifikasi tersebut meliputi:

1. Menggunakan Notebook HP Pavilion DV3 berprosesor Intel Core i3, RAM 1 GB
2. OS Windows 7 Home Premium
3. *Web Browser* (direkomendasikan untuk menggunakan Google Chrome untuk tampilan terbaik).

##### **4.2.2 Perangkat Pengujian untuk Client**

Berikut ini akan dijabarkan mengenai spesifikasi perangkat pengujian pada sisi *client* dilakukan oleh Penulis. Spesifikasi tersebut meliputi:

1. Menggunakan Emulator Android 2.2. (API 8)
2. IDE Eclipse Helios 3.6
3. OS Gingerbread (2.3.7)

### 4.3 Prosedur Pengujian

Prosedur pengujian akan dibagi menjadi dua bagian yakni pengujian pada sisi server dan pada sisi *client*. Adapun prosedur yang dilakukan akan dijelaskan pada sub bab 4.3.1 dan 4.3.2

#### 4.3.1 Prosedur Pengujian pada Server

Berikut ini akan dijabarkan prosedur pengujian yang dilakukan Penulis pada sisi *server*.

1. Membuka *web browser* Google Chrome
2. Mengakses alamat  
`ttp://localhost:8080/server/index.php/login`
3. Mengakses setiap menu yang tersedia
4. Menjalankan fungsi-fungsi utama

#### 4.3.2 Prosedur Pengujian pada Client

Berikut ini akan dijabarkan prosedur pengujian yang dilakukan Penulis pada sisi *client*.

1. Menjalankan Eclipse dengan SDK *Managernya*
2. Jalankan Aplikasi
3. Mengakses menu yang tersedia
4. Menjalankan fungsi-fungsi utama

## 4.4 Data Hasil Pengujian

Setelah dilakukan pengujian baik dari sisi *server* maupun *client*, Penulis mendapatkan hasil yang akan dijabarkan pada sub bab dibawah ini. Sub bab akan dibagi menjadi 2 sisi yakni sisi *server* dan sisi *client*.

### 4.4.1. Hasil Pengujian pada Server

Berikut ini akan dijabarkan hasil pengujian pada sisi *server*. Pengujian dilakukan dengan fokus kepada fungsi terkait

#### 4.4.1.1 Beranda

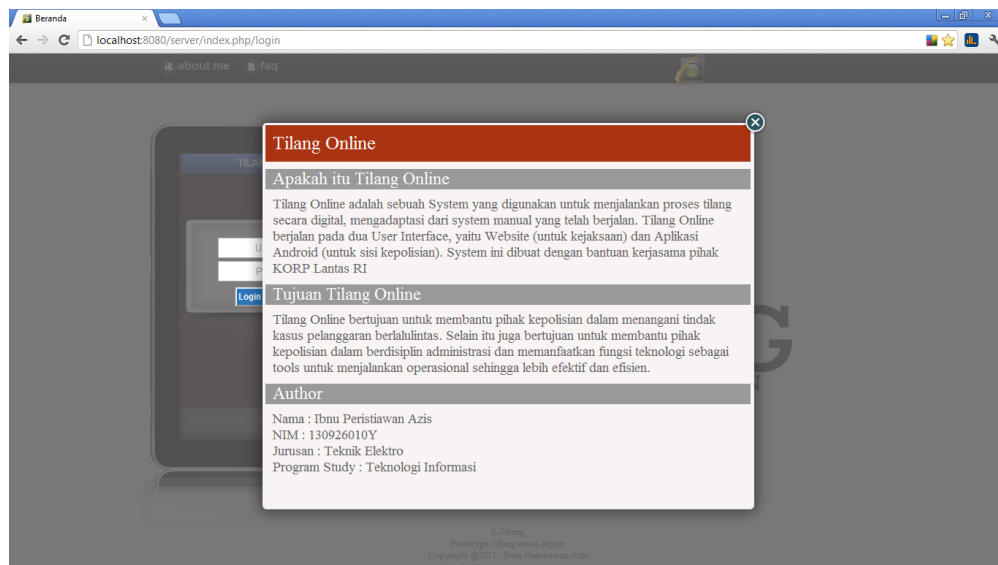
Ini adalah halaman yang pertama kali tampil ketika *user* mengakses halaman *web*. Tidak banyak yang terdapat pada halaman beranda ini hanya form login dan beberapa logo juga tulisan yang deskriptif, tampilan beranda seperti gambar 4.1 di bawah ini



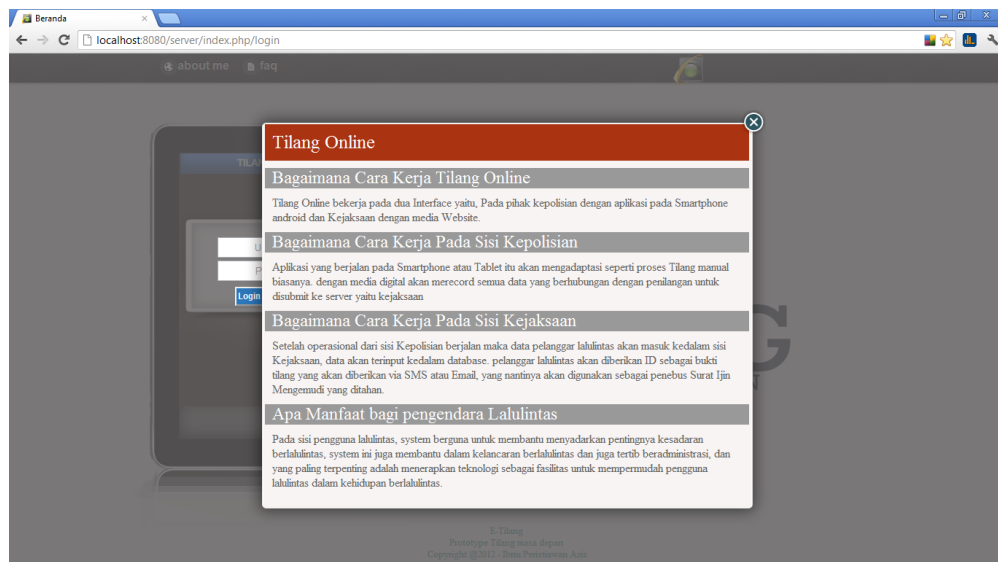
**Gambar 4.1 Tampilan beranda**

#### 4.4.1.2 About Me dan FAQ

About Me adalah deskripsi singkat mengenai system, pengertian kegunaan dan juga *author* dari system tersebut, guna meneguhkan dan membuat profil website. FAQ berisi mengenai pertanyaan yang sering ditujukan kepada website atau mekanisme penggunaannya. Gambar 4.2 dan 4.3 masing – masing menjelaskan tentang About Me dan FAQ.



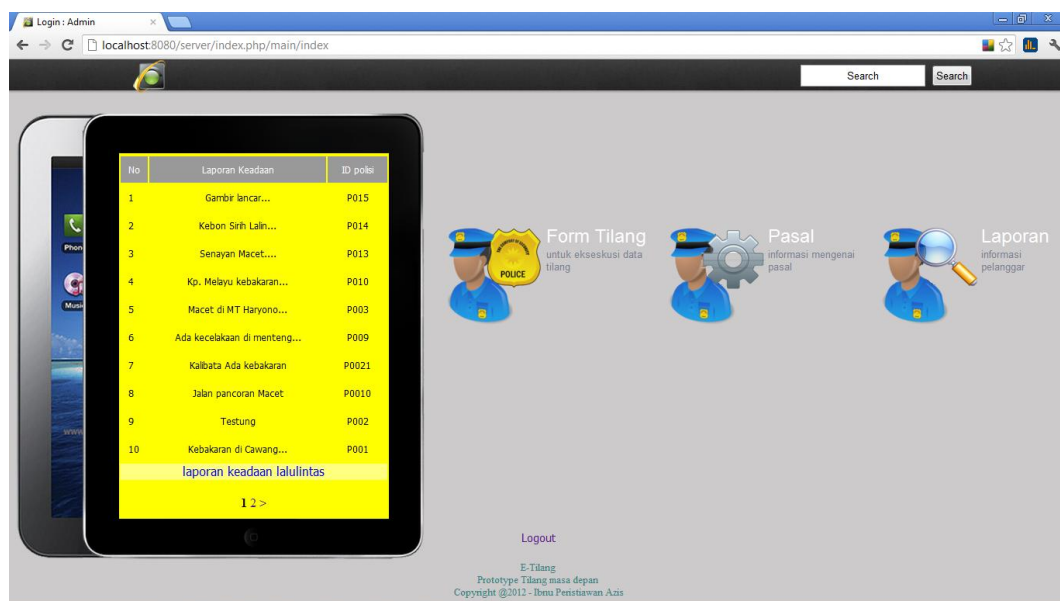
Gambar 4.2 About Me



Gambar 4.3 FAQ

#### 4.4.1.3 Halaman Utama

Setelah berhasil login maka user akan langsung menuju halaman utama yaitu halaman yang berisi gambar dengan penjelasan dan link kepada setiap page, pada halaman utama juga pengguna bisa melihat Laporan singkat keadaan lau lintas yang dipost oleh petugas kepolisian. Tampilan halaman utama bisa dilihat pada gambar 4.4 di bawah ini



**Gambar 4.4 Tampilan halaman utama**

Pada gambar tablet adalah laporan singkat mengenai keadaa lalu lintas yang dipost oleh petugas atau penyidik, sehingga pihak kejaksaan tau atau mempunya informasi daerah mana saja yang sedang bermasalah.

#### 4.4.1.4 Halaman Pasal

Pada halaman ini, sisi kejaksaan atau server dapat mengubah atau mengedit pasal yang nantinya akan disanksi kan oleh pelanggar lalu lintas. Pasal tresebut sesuai dengan UU LLAJ yang telah ditetapkan, dalam page ini juga disediakan untuk mendownload sebagai pdf File.

No	Klasifikasi dan Pasal yang melanggar	Bukti Pelanggaran	Kendaraan Tidak Bermotor	Sepeda Motor	Mobil Pribadi	Mobil Penumpang Umum / Minibus	Pick Up / Mobil Box	Bus Umum / Non Umum	Truck	Truck Gandeng / Tronton	Kendaraan Kontainer	Action
1	299 UU LLAJ Jo 122 a, b, c, UU LLAJ	Mengendarai kendaraan tidak bermotor yang dengan sengaja berpegang pada kendaraan bermotor untuk ditarik, menarik benda-benda yang dapat membahayakan pengguna jalan lain, dan/ atau menggunakan jalur lalu lintas kendaraan bermotor.		100000	100000	100000						
2	293 (2) UU LLAJ Jo 107 (2) UU LLAJ	Mengendarai Sepeda motor di jalan tanpa menyalakan lampu utama disang hari		100000	100000	100000						
3	292 UU LLAJ Jo 106 (9) UU LLAJ	mengendarai sepeda motor yang membiarkan penumpangnya tidak memakai helm		250000	250000	250000						
4	291 (2) UU LLAJ Jo 106 (8) UU LLAJ	Mengendarai Sepeda Motor tidak menggunakan helm standard Nasional Indonesia		250000	250000	250000						
5	291 (2) UU LLAJ Jo 106 (8) UU LLAJ	Mengendarai sepeda motor tanpa kereta samping yang mengangkut penumpang lebih dari 1 (satu) orang.		250000	250000	250000						
6	289 UULAJ Jo 106 (7) UU LLAJ	Mengemudikan kendaraan bermotor atau penumpang yang duduk disamping pengemudi yang tidak menggunakan sabuk keselamatan.		250000	250000	250000						
7	285 (1) UU LLAJ Jo 106 (3) 48 (2) (3) UU LLAJ	Mengendarai Sepeda Motor di jalan tidak memenuhi persyaratan teknis dan laik jalan yang meliputi kaca spion, kaskon, lampu utama, lampu rem, lampu penunjuk arah, alat pemantul cahaya, alat pengukur kecepatan, knalpot dan kedalaman alur ban.		250000	250000	250000						

**Gambar 4.5 Tampilan Halaman Pasal**

Pada gambar diatas, tabel pasal menjadi pengisi halaman pasal, dilengkapi dengan *action* hapus dan edit, juga ada sub menu tambah pasal dan download sebagai Pdf. Untuk tambah pasal dapat dilihat pada gambar 4.6 di bawah ini

**Gambar 4.6 Form Tambah Pasal**

#### 4.4.1.5 Halaman Laporan

Halaman Laporan berisi mengenai laporan pesan atau SMS yang terkirim kepada pelanggar lalu lintas, sehingga



dapat dijadikan sebagai laporan untuk referensi atau administrasi data. Halaman Laporan dapat dilihat pada gambar 4.7 di bawah ini

Report

localhost:8080/server/index.php/pesanterkirim\_c/index

download

Semua Data Pesan Terkirim (sentitens)  
Jumlah Data : 20

No	Time	No's Tujuan	Isi Pesan	Status	Aksi
1	2012-08-06 16:05:01	085691384671	asdasd	SendingOK	<a href="#">Hapus</a>
2	2012-08-06 16:06:08	085690742812	HALOO INI WEBSITE WHITE	SendingError	<a href="#">Hapus</a>
3	2012-08-06 16:25:14	085691384671	Test	SendingOK	<a href="#">Hapus</a>
4	2012-08-06 16:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
5	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
6	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
7	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
8	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
9	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
10	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
11	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
12	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
13	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
14	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
15	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
16	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
17	2012-08-06 19:28:16	085691384671	a	SendingOK	<a href="#">Hapus</a>
18	2012-08-07 04:31:43	085691384671	a	SendingOK	<a href="#">Hapus</a>

Back 1 2 Next

Back

E-Tilang  
Prototype Tilang masa depan  
Copyright @2012 - Ilmu Pengetahuan Aza

**Gambar 4.7 Halaman Laporan**

#### 4.4.1.6 Halaman Tilang

Ini merupakan halaman utama, karena pada halaman ini pihak kejaksaan menginput ID Tilang yang diberikan oleh pelanggar lalu lintas untuk disubmit, proses ini diibaratkan sebagai sidang di kejaksaan, parameter yang paling penting adalah ID tilang karena menyimpan semua data. Dengan menginput ID akan keluar semua atribut pelanggar, halaman Tilang bisa dilihat pada gambar 4.8 di bawah ini

**Form Tilang**

Data Pelanggr  
Jumlah Data : 1

1234567890

No	ID Tilang	Nama	Jenis Kelamin	Alamat	Profesi	Nomer Sim	Pasal	Nomer STNK	Jenis Kendaraan	Nomer Handphone	Aksi
1	1	Ibnu P	Pria	Jalan cipinang besar utara 10/10 jatinegara jakarta	mahasiswa	1234567890	UU 291 (2) UU (LAJ)	1234567	motor	098765432	<a href="#">SUBMIT</a>

[Back](#)

E-Tilang  
Prototype Tilang masa depan  
Copyright @2012 - Ilmu Pengetahuan Asis

**Gambar 4.8 Halaman Tilang**

#### 4.4.2 Hasil Pengujian pada Client

Berikut ini akan dijabarkan hasil pengujian pada sisi *client*. Pengujian dilakukan dengan fokus kepada fungsi terkait

##### 4.4.2.1 Form Tilang

Form tilang adalah form utama yang akan mempassing parameter ke dalam database, parameter seputar data pelanggar, pasal, nomer SIM dan nomer handphone pelanggar yang sangat penting, implementasi terlihat seperti pada gambar 4.9 di bawah ini

The image shows a mobile application interface for 'E-TILANG'. At the top, the status bar displays '3G', signal strength, battery level, and the time '4:51 AM'. Below the title bar, the form contains several input fields: 'ID Polisi', 'Password', 'Nama', a gender dropdown menu currently set to 'Pria', 'Alamat', 'pekerjaan', a SIM dropdown menu currently set to 'SIM A', 'Nomer Kendaraan' (highlighted with an orange border), 'Nomer STNK', and a 'Senedah Motor' dropdown menu at the bottom.

**Gambar 4.9 Implementasi Tilang**

Setelah diinput maka kesemua atribut akan masuk kedala tabel database dan akan langkah selanjutnya adalah pelanggar yang telah menerima ID Tilang dan pemberitahuan akan pergi ke kejaksaan yang dimaksudkan dalam SMS tersebut dan memberikan ID tilang tersebut.

## 4.5 Analisa Pengujian

Setelah dilakukan pengujian pada sisi *server* dan *client*, akses kepada setiap halaman bekerja dengan baik, begitu pula untuk setiap fungsi utama yang terdapat pada masing-masing tipe *user*. *Bug* dan *error* pun tidak ditemukan karena telah dilakukan pengujian dan perbaikan pada tahap pembangunan sistem.

Seluruh *input* dapat ditangani dengan baik, begitu juga dengan validasi yang dibantu dengan javascript pada sisi *server* sehingga dapat dengan mudah tangani.

Dari pengujian-pengujian diatas, anomali terjadi pada *server* dengan *web browser* yang berbeda yakni *user interface* telah sedikit berubah pada *web browser* Mozilla Firefox, meskipun demikian, masalah ini tidak terlalu signifikan hingga dapat membahayakan *server* dan sebagainya.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

1. Aplikasi E-Tilang ini merupakan Prototype sebuah system Tilang yang mendigitalisasi proses tilang manual. aplikasi ini mampu mengurangi penggunaan kertas atau sumber daya lainnya dan juga meminimalisir penyimpangan tindak pidana melanggar lalu lintas.
2. Pada sisi kejaksaan atau pengadilan seharusnya tidak sekedar sebagai eksekutor, optimasi sharing informasi harus dilakukan.
3. E-Tilang mengadaptasi *form* tilang manual dengan atribut tambahan yaitu nomer *handphone* pelanggar yang akan dikirimkan ID Tilang. ID Tilang tersebut menyimpan semua aktivitas atau keterangan sesuai dengan parameter yang ada pada aplikasi atau pada kertas bukti pelanggaran.

## **5.2. Saran**

Untuk kedepan aplikasi ini tidak hanya mengadaptasi *form* tilang konvensional tapi juga mengadaptasi *form* tilang untuk sistem penitipan atau membayar via bank. Aplikasi ini juga akan terhubung dengan *E-Banking* sehingga para pelanggar lalu lintas tidak perlu lagi beranjak dari tempat tilangnya.

## DAFTAR PUSTAKA

1. *Bahasa Pemrograman*, diakses dari [http://id.wikipedia.org/wiki/Bahasa\\_pemrograman](http://id.wikipedia.org/wiki/Bahasa_pemrograman), 7 Juli 2012, 15:21
2. *PHP*, diakses dari <http://id.wikipedia.org/wiki/PHP>, 7 Juli 2012 15:28
3. Ullman, Larry. 2008. *PHP 6 and MySQL 5*. USA: Peachpit Press
4. *MySQL*, diakses dari <http://id.wikipedia.org/wiki/Mysql>, 7 Juli 2012, 15:33
5. Felker, Donn. 2011. *Android Application Development for Dummies*. Canada: Wiley Publishing, Inc.
6. Meier, Reto. 2009. *Professional Android Application Development*. Canada: Wiley Publishing, Inc.
7. Welling, Luke. 2009. *PHP and MySQL Web Development*. USA: Pearson Education, Inc.

# LAMPIRAN

## 1. Login.php (*Controllers*)

```
<?php
class Login extends CI_Controller {
public function __construct() {
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->model('M_session');
$this->load->library('javascript');
}
function index() {
    session_start();
    $session = $this->M_session->get_session();
    if (!$session['session_userid']) {
        $this->load->view('hello');
    } else {
        redirect('main/index');
    }
}
function proses() {
    $this->load->model('M_master_userid','',true);
    $username = $this->input->post('username'); //read the username
    that filled by the user
    $password = $this->input->post('password'); //read the password
    that filled by the user
    $data['username'] = '';
    $data['password'] = '';
    $result= $this->M_master_userid->get_userid($username);
    foreach ($result->result() as $row ) {
    $data['username'] = $row->username;
    $data['password'] = $row->userpass;
    }
    $this->form_validation->set_rules('username', 'Username',
    'required');
    $this->form_validation->set_rules('password', 'Password',
    'required');
    $this->form_validation->run();
    if ($username and $username==$data['username'] and
    $password==$data['password']) {
    $this->M_session->store_session($username); // simpan session
    userid
    }
    else if($username!=$data['username'] and
    $password!=$data['password']) {
    redirect('login/proses?error=2'); }
    $this->index(); }
```



```

function logout() {
    $session = $this->M_session->get_session();
    if (!$session['session_userid']) {
        redirect(base_url()/'index.php'. 'login');    }
    else {
        redirect('logout?cond=true');    }    }
?>

```

## 2. Main.php (Controllers)

```

<?php
class Main extends CI_Controller {
public function __construct() {
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->library('pagination');
$this->load->model('Twit_model');
$this->load->model('M_session');
$this->getparam();
}
function getparam() {
    @session_start();
    $this->page['iserror']= false;
    $this->page['footer']= "Error";
    $this->page['error_mesg']= "";    }
function index(){
    $idpolisi = $this->input->post('idpolisi');
    $result= $this->Twit_model->getpolisi($idpolisi);
    $session = $this->M_session->get_session();
    if (!$session['session_userid']) {
        redirect('login?error=3');    }
    elseif($session['session_userid']) {
        $this->db->select('*');
        $this->db->from('twit');
        $getData = $this->db->get('');
        $a = $getData->num_rows();
        $config['base_url'] = base_url(). 'index.php/main/index';
        $config['total_rows'] = $a; //total rows
        $config['per_page'] = 9; //the number of per page for pagination
        $config['uri_segment'] = 3; //see from base_url. 3 for this case
        $config['full_tag_open'] = '<br>';
        $config['full_tag_close'] = '<br>';
        $this->pagination->initialize($config); //initialize pagination
        //$data['detail'] = $this->Pasal_model-
        >ambilsemuadata($config['per_page'],$this->uri->segment(3));
        $data['twit'] = $this->Twit_model-
        >ambiltwit($config['per_page'],$this->uri->segment(3));
        $this->load->view('welcom_view', $data);
    }
}

```

```
}
```

### 3. Pasal.php (Controllers)

```
<?php
class Pasal extends CI_Controller {
public function __construct() {
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->library('pagination');
$this->load->model('Pasal_model');
$this->load->library('cezpdf');}
function index(){
$this->proses();
}
function proses()
{
    $session = $this->M_session->get_session();
    if (!$session['session_userid']){
        //user belum login
        redirect('login/proses?error=3');
    }
    else{
        $this->db->select('*');
        $this->db->from('hukuman');
        $getData = $this->db->get('');
        $a = $getData->num_rows();
        $config['base_url'] = base_url().'index.php/pasal/proses'; //set
        the base url for pagination
        $config['total_rows'] = $a; //total row
        $config['per_page'] = 7; //the number of per page for pagination
        $config['uri_segment'] = 3; //see from base_url. 3 for this case
            $config['full_tag_open'] = '<br>';
            $config['full_tag_close'] = '<br>';
        $this->pagination->initialize($config); //initialize pagination
        $data['detail'] = $this->Pasal_model-
        >ambilsemuadata($config['per_page'],$this->uri->segment(3));
        $this->load->view('tambah_pasal', $data); }}
function insert(){
    $pasal = $this->input->post('pasal');
    $deskripsi = $this->input->post('deskripsi');
    $result= $this->Pasal_model->check($pasal);
    foreach ($result->result() as $row ){
        $data['_pasal'] = $row->Pasal;
        $data['_keterangan'] = $row->Keterangan;
        if ($pasal == "" || $deskripsi == ""){
            redirect('pasal/proses?error=1'); //ini diambil dari
            Application/config/routes.php
        }
    }
}
```

```

elseif($pasal == $data['_pasal'] || $deskripsi ==
$data['_keterangan']){
redirect('pasal/proses?error=2');}
else{ $this->Pasal_model->insert_table($pasal, $deskripsi);
redirect('pasal/proses?error=3');}
function delete($pasal){
$this->Pasal_model->delete($pasal);
redirect('pasal/proses?delete=1'); }
function ezpdf(){
$this->cezpdf->selectFont('./fonts/Helvetica.afm');
$this->cezpdf->ezText('Hello World!',30, array('justification' =>
'center'));
$this->cezpdf->ezStream();
$this->cezpdf->ezSetMargins(10,10,50,50);
$this->cezpdf->line(20,40,578,40);} }?>?>

```

#### 4. Form\_tilang.php (Controllers)

```

<?php
class Form_tilang extends CI_Controller{
public function __construct(){
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->model('Tilang_model');
$this->load->model('M_session');
}
function getparam() {
    @session_start();
    $this->page['iserror']= false;
    $this->page['error_mesg']= "";
}
function index(){
    $session = $this->M_session->get_session();
    $data = array();
    if (!$session['session_userid']){
        //user belum login
        redirect('login/proses?error=3');
    }
    else{
        $data['logout'] = 'You have been logged out.';
        $this->load->view('form_tilang_v', $data);}
function back(){
    redirect(base_url()/'index.php'. 'main' );
} }?>

```

#### 5. Hapuspesanter kirim.php (Controllers)

```

<?php
class Isi_insert_pasal_c extends CI_Controller{
public function __construct(){

```

```

parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->model('Pasal_model');
}
function index(){
$this->load->view('isi_insert_pasal');
}
function insert(){
$pasal = $this->input->post('pasal');
$deskripsi = $this->input->post('deskripsi');
$nominal = $this->input->post('nominal');
$menu = $this->input->post('menu');
$foo2 = $this->input->post('foo2');
$result= $this->Pasal_model->check($pasal);
foreach ($result->result() as $row ){
$data['_pasal'] = $row->Pasal;
$data['_keterangan'] = $row->Keterangan;
$data['_nominal'] = $row->Nominal;
$data['_menu'] = $row->Kategori;
}
if ($pasal == "" || $deskripsi == ""){
redirect('pasal/proses?error=1'); //ini diambil dari
Application/config/routes.php }
elseif($pasal == $data['_pasal'] || $deskripsi ==
$data['_keterangan'] || $nominal == $data['_nominal'] ){
redirect('pasal/proses?error=2'); }
else{
$this->Pasal_model->insert_table($pasal, $deskripsi, $nominal,
$foo2);
redirect('pasal/proses?saved=1'); }}}
?>

```

## 6. isi\_insert\_pasal.php (Controllers)

```
<?php
class Isi_insert_pasal_c extends CI_Controller{
public function __construct(){
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->model('Pasal_model');
}
function index(){
    $this->load->view('isi_insert_pasal');
}
function insert(){
    $pasal = $this->input->post('pasal');
    $deskripsi = $this->input->post('deskripsi');
    $nominal = $this->input->post('nominal');
    $menu = $this->input->post('menu');
    $foo2 = $this->input->post('foo2');
    $result= $this->Pasal_model->check($pasal);
    foreach ($result->result() as $row ){
        $data['_pasal'] = $row->Pasal;
        $data['_keterangan'] = $row->Keterangan;
        $data['_nominal'] = $row->Nominal;
        $data['_menu'] = $row->Kategori;
    }
    if ($pasal == "" || $deskripsi == ""){
        redirect('pasal/proses?error=1'); //ini diambil dari
Application/config/routes.php    }
    elseif($pasal == $data['_pasal'] || $deskripsi ==
$data['_keterangan'] || $nominal == $data['_nominal'] ){
        redirect('pasal/proses?error=2');
    }
    else{
        $this->Pasal_model->insert_table($pasal, $deskripsi, $nominal,
$foo2);
        redirect('pasal/proses?saved=1'); }    }}
?>
```

## 7. Aboutme.php (Controllers)

```
<?php
class Isi_insert_pasal_c extends CI_Controller{
public function __construct(){
parent::__construct();
$this->load->helper('url');
$this->load->helper('form');
$this->load->library('form_validation');
$this->load->library('session');
$this->load->model('Pasal_model');
}    function index(){
```

```

        $this->load->view('isi_insert_pasal');
    }
function insert(){
    $pasal = $this->input->post('pasal');
    $deskripsi = $this->input->post('deskripsi');
    $nominal = $this->input->post('nominal');
    $menu = $this->input->post('menu');
    $foo2 = $this->input->post('foo2');
    $result= $this->Pasal_model->check($pasal);
    foreach ($result->result() as $row ){
        $data['_pasal'] = $row->Pasal;
        $data['_keterangan'] = $row->Keterangan;
        $data['_nominal'] = $row->Nominal;
        $data['_menu'] = $row->Kategori;
    }
    if ($pasal == "" || $deskripsi == ""){
        redirect('pasal/proses?error=1'); //ini diambil dari
        Application/config/routes.php }
    elseif($pasal == $data['_pasal'] || $deskripsi ==
    $data['_keterangan'] || $nominal == $data['_nominal'] ){
        redirect('pasal/proses?error=2');
    }
    else{
        $this->Pasal_model->insert_table($pasal, $deskripsi,
        $nominal, $foo2);
        redirect('pasal/proses?saved=1');    }    }    }
?>

```

## 8. Form\_tilang.php (Views)

```

<html>
<head>
<title>Tilang</title>
<link rel="shortcut icon" href="<?=base_url();?>/image/icon.png">
<link rel="stylesheet" type="text/css" media="screen, projection"
href="<?=base_url();?>css/css.css" />
</head>
<body id="body">
<div class="header_navigasi">
<div id="wrapper_navigasi">
</div>
</div>
<h1 align=center>Form Tilang</h1>
<table align="center">
<td><input type="text" name="id_tilang" placeholder="Username"
autocomplete="off"></td>
<div class="wrapper_button_login">
<td><button name="submit" class="button_login" type="submit" ><a
href="<?=base_url();?>index.php/form_tilang/proses"> Submit
</button></td>
</div>
</table>

```



## 9 Hello.php (Views)

```
<html>
<head>
<title>Beranda</title>
<link rel="shortcut icon" href="=base_url();?&gt;/image/icon.png"&gt;
&lt;script src="<?=base_url();?&gt;js/jquery.tools.min.js"&gt;&lt;/script&gt;
&lt;script src="<?=base_url();?&gt;js/aboutme_js.js"&gt;&lt;/script&gt;
&lt;link rel="stylesheet" type="text/css" media="screen, projection"
href="<?=base_url();?&gt;css/css.css" /&gt;
&lt;link rel="stylesheet" type="text/css" media="screen, projection"
href="<?=base_url();?&gt;css/error_bg.css" /&gt;
    &lt;link rel="stylesheet" type="text/css" media="screen,
projection" href="<?=base_url();?&gt;css/csss.css" /&gt;
    &lt;link rel="stylesheet" type="text/css" media="screen,
projection" href="<?=base_url();?&gt;css/aboutme_css.css" /&gt;
&lt;/head&gt;
&lt;body id="body" padding-top: 90px;&gt;
&lt;div class="header_icon" style="padding-bottom: 900px;"&gt; &lt;img
src="<?=base_url();?&gt;/image/icon_kecil.png" /&gt; &lt;/div&gt;
&lt;div class="header_navigasi"&gt;
&lt;div id="wrapper_navigasi"&gt;
&lt;div id="navigasi"&gt;
&lt;ul&gt;
&lt;li&gt;
&lt;a href="<?=base_url();?&gt;index.php/aboutme_c" rel="#overlay"
&gt;&lt;img src="<?=base_url();?&gt;/image/world.png" /&gt;About me &lt;/a&gt;
&lt;a href="<?=base_url();?&gt;index.php/faq_controller" rel="#overlay"
&gt;&lt;img src="<?=base_url();?&gt;/image/page.png" /&gt;FAQ &lt;/a&gt;
    &lt;/li&gt;
    &lt;/ul&gt;
    &lt;/div&gt;
    &lt;/div&gt;
    &lt;/div&gt;
&lt;div class="frame"&gt;
&lt;div id="ipad"&gt;
&lt;img src="<?=base_url();?&gt;/image/ipad.png" /&gt;
    &lt;/div&gt;
&lt;div class="logoutama"&gt;
&lt;img src="<?=base_url();?&gt;/image/icon.png" /&gt;
    &lt;/div&gt;
&lt;div class="etilang"&gt;
&lt;img src="<?=base_url();?&gt;/image/etilang.png" /&gt;
    &lt;/div&gt;
    &lt;div class="ads"&gt;
&lt;a href="https://twitter.com/peristiwaanazis" class="twit"&gt;&lt;img
src="<?=base_url();?&gt;/image/twitter.png" /&gt;&lt;/a&gt;
&lt;a href="http://www.facebook.com/ibnu.p.azis" &gt;&lt;img
src="<?=base_url();?&gt;/image/facebook.png" /&gt;&lt;/a&gt;
    &lt;/div&gt;</pre
```



```

<form action="<?=base_url();?>index.php/login/proses"
method="post">
<div class="background_login">

</div>
<div class="login_wrapper">
<div class="placeholdering">
<input type="text" name="username" placeholder="Username"
autocomplete="off">
</div>
<input type="password" name="password" placeholder="Password"
autocomplete="off" >
<div class="wrapper_button_login">
<button name="submit" class="button_login" type="submit"
tabindex="0" > Login </button>
<button class="button_login" type="reset" tabindex="0" > Reset
</button>
</div>
</form>
</div>
<div class="error">
<div class="errorbiasa" align="center"><h4><? echo
form_error('username'); ?></h4></div>
<div class="errorbiasa2" align="center"><h4><? echo
form_error('password'); ?></h4></div>
</div>
</div>
<?php if(isset($logout)) {?>
<div class="error" align="center"><h2> <?= @$logout ?>
</div></h2>
<?}&?>
<?
if(!isset($_GET['error'])) {}
else{
$error = $_GET['error'];
}
if(empty($error)){
}
elseif($error == 2){
echo '<div class="error" id="Slideshow" align="center"> <h2>
Paswword dan Username Salah </h2></div>' ;
}
elseif($error == 3){
echo '<div class="error" align="center" > <h2> Harus Login
Terlebih dahulu </h2></div>' ;
}
elseif($error == 4){
echo '<div class="error" align="center" > <h2> error </h2></div>'
;
}
}
?>

```

```

        </div>
    </div>
</div>
</body>
</div>
<div class="referensi" >
    <nav>
        <ul>
<li><a href="http://www.tmcmetro.com/">TMC Polda Metro
Jaya</a></li>
<li><a href="http://lewatmana.com/">Lewat Mana</a></li>
<li><a href="http://infolalulintas.com/">Info Lalu Lintas</a></li>
<li><a href="www.lantas.polri.go.id">Lantas Polri</a></li>
<li></li>
        </ul>
    </nav>
</div>
<? $this->load->view('footer'); ?>
<div class="apple_overlay" id="overlay"><a class="close"></a>
<!-- the external content is loaded inside this tag -->
<div class="contentWrap"></div>
</div>
<!-- Untuk Munculin si Overlay kedalam DIV ContentWrap -->
    </body>
</div>
<div class="palsu">s</div>
</html>

```

## 10. Utama.Java

```
package halo.source;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class Utama extends Activity {
    ImageView img1 ;
    ImageView img2 ;
    ImageView img3 ;
    Button btn ;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.utama);
        img1 = (ImageView)findViewById(R.id.img1);
        img2 = (ImageView)findViewById(R.id.img2);
        img3 = (ImageView)findViewById(R.id.img3);
        onclick();
        public void onclick(){
            img1.setOnClickListener(new View.OnClickListener() {
                public void onClick(View arg0) {
                    Intent i = new Intent(Utama.this, Tilang.class);
                    startActivity(i); });
            img2.setOnClickListener(new View.OnClickListener() {
                public void onClick(View arg0) {
                    Intent i = new Intent(Utama.this, Regis.class);
                    startActivity(i); });
            img3.setOnClickListener(new View.OnClickListener() {
                public void onClick(View arg0) {
                    Intent i = new Intent(Utama.this, Twit.class);
                    startActivity(i); });
        }
    }
}
```

## 11. Twit.Java

```
package halo.source;

import java.util.ArrayList;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.Activity;
import android.content.Intent;
import android.database.SQLException;
import android.os.Bundle;
import android.view.View;
```

```

import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Twit extends Activity {
    EditText idpolisi ;
    EditText password ;
    EditText twit ;
    Button submit;
    Button cancel ;
    String url = koneksi.url;
    @Override
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.twit);
        Database dbUser = new Database(this);
        dbUser.open();
        dbUser.createUser();
        dbUser.close();
        idpolisi = (EditText) findViewById(R.id.idpolisi);
        password = (EditText) findViewById(R.id.password);
        twit = (EditText) findViewById(R.id.twit);
        submit = (Button) findViewById(R.id.submit);
        cancel = (Button) findViewById(R.id.cancel);
        buttonListener();
    }
    private void buttonListener() {
        submit.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if(idpolisi.getText().toString().equals("")) {
                    Toast.makeText(Twit.this, "Masukan ID",
                        Toast.LENGTH_SHORT).show();
                } else if(password.getText().toString().equals("")) {
                    Toast.makeText(Twit.this, "Masukan Password",
                        Toast.LENGTH_SHORT).show();
                } else if(twit.getText().toString().equals("")) {
                    Toast.makeText(Twit.this, "Masukan twit",
                        Toast.LENGTH_SHORT).show();
                } else {
                    process();
                }
            }
        });
    }
    protected void process() {
        ArrayList<NameValuePair> twit = new ArrayList<NameValuePair>();
        twit.add(new BasicNameValuePair("function", "twit"));
        twit.add(new BasicNameValuePair("idpolisi",
            idpolisi.getText().toString()));
        twit.add(new BasicNameValuePair("password",
            password.getText().toString()));
        twit.add(new BasicNameValuePair("twit",
            this.twit.getText().toString()));
        String respon = null;
        try {

```

```
        respon = koneksi.eksekusiHttpPost(url, twit);
        String res = respon.toString().trim();
        if(res.equals("0")){
            Toast.makeText(this, "Gagal", Toast.LENGTH_SHORT).show();
        }
        else{
            Toast.makeText(this, "Sukses", Toast.LENGTH_SHORT).show();
            Intent i = new Intent(this, Utama.class);
            startActivity(i);
            finish(); } } catch(Exception e) {
            e.printStackTrace();
            Toast.makeText(this, e.toString(), Toast.LENGTH_SHORT).show();
        }
    }
}
```

## 12. Tilang.Java

```
package halo.source;
import java.util.ArrayList;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import android.app.Activity;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;
public class Tilang extends Activity {
    EditText idpolisi ;
    EditText password ;
    EditText nama;
    Spinner gender;
    EditText alamat;
    EditText pekerjaan;
    Spinner sim;
    EditText nomerkendaraan ;
    EditText nomerstnk ;
    Spinner jeniskendaraan ;
    Spinner pasal;
    EditText nomerhandphone;
    EditText lokasi ;
    Button submit;
    Button cancel ;
    String url = koneksi.url;
    @Override
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.tilang);
        idpolisi = (EditText) findViewById(R.id.idpolisi);
        password = (EditText) findViewById(R.id.password);
        nama = (EditText) findViewById(R.id.nama);
        gender = (Spinner) findViewById(R.id.gender);
        alamat = (EditText) findViewById(R.id.alamat);
        pekerjaan = (EditText) findViewById(R.id.pekerjaan);
        sim = (Spinner) findViewById(R.id.sim);
        nomerkendaraan = (EditText) findViewById(R.id.nomerkendaraan);
        nomerstnk = (EditText) findViewById(R.id.nomerstnk);
        jeniskendaraan = (Spinner) findViewById(R.id.jeniskendaraan);
        pasal = (Spinner) findViewById(R.id.pasal);
        nomerhandphone = (EditText) findViewById(R.id.nomerhandphone);
        lokasi = (EditText) findViewById(R.id.lokasi);
        submit = (Button) findViewById(R.id.submit);
```

```

        cancel = (Button) findViewById(R.id.cancel);
        buttonListener(); }

private void buttonListener() {
    submit.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if(idpolisi.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan ID",
                    Toast.LENGTH_SHORT).show(); }
            else if(password.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan Password",
                    Toast.LENGTH_SHORT).show(); }
            else if(nama.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan Nama",
                    Toast.LENGTH_SHORT).show(); }
            else if(gender.getSelectedItem().toString().equals("")) {
                Toast.makeText(Tilang.this, "pilih gender",
                    Toast.LENGTH_SHORT).show(); }
            else if(alamat.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan alamat",
                    Toast.LENGTH_SHORT).show(); }
            if(pekerjaan.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan pekerjaan",
                    Toast.LENGTH_SHORT).show(); }
            else if(sim.getSelectedItem().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan sim",
                    Toast.LENGTH_SHORT).show(); }
            else if(nomerkendaraan.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan twit",
                    Toast.LENGTH_SHORT).show(); }
            else if(nomerstnk.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan nomer stnk",
                    Toast.LENGTH_SHORT).show(); }
            else if(jeniskendaraan.getSelectedItem().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan
                twit"Toast.LENGTH_SHORT).show(); }
            else if(pasal.getSelectedItem().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan pasal",
                    Toast.LENGTH_SHORT).show(); }
            else if(nomerhandphone.getText().toString().equals("")) {

                Toast.makeText(Tilang.this, "Masukan nomer hp",
                    Toast.LENGTH_SHORT).show(); }
            else if(lokasi.getText().toString().equals("")) {
                Toast.makeText(Tilang.this, "Masukan lokasi",
                    Toast.LENGTH_SHORT).show(); }
            else {
                process();}}});
        }
        protected void process() {
            ArrayList<NameValuePair> tilang = new ArrayList<NameValuePair>();

```

```

tilang.add(new BasicNameValuePair("function", "tilang"));
tilang.add(new BasicNameValuePair("idpolisi",
idpolisi.getText().toString()));
tilang.add(new BasicNameValuePair("password",
password.getText().toString()));
tilang.add(new BasicNameValuePair("nama",
nama.getText().toString()));
tilang.add(new BasicNameValuePair("gender",
gender.getSelectedItem().toString()));
tilang.add(new BasicNameValuePair("alamat",
alamat.getText().toString()));
tilang.add(new BasicNameValuePair("pekerjaan",
pekerjaan.getText().toString()));
tilang.add(new BasicNameValuePair("sim",
sim.getSelectedItem().toString()));
tilang.add(new BasicNameValuePair("nomerkendaraan",
nomerkendaraan.getText().toString()));
tilang.add(new BasicNameValuePair("nomerstnk",
nomerstnk.getText().toString()));
tilang.add(new BasicNameValuePair("jeniskendaraan",
jeniskendaraan.getSelectedItem().toString()));
tilang.add(new BasicNameValuePair("pasal",
pasal.getSelectedItem().toString()));
tilang.add(new BasicNameValuePair("nomerhandphone",
nomerhandphone.getText().toString()));
tilang.add(new BasicNameValuePair("lokasi",
lokasi.getText().toString()));
String respon = null;
try {
respon = koneksi.eksekusiHttpPost(url, tilang);
String res = respon.toString().trim();
if(res.equals("0")){
Toast.makeText(this, "Gagal", Toast.LENGTH_SHORT).show();
}
else{
Toast.makeText(this, "Sukses", Toast.LENGTH_SHORT).show();
Intent i = new Intent(this, Utama.class);
startActivity(i);
finish(); } }
catch(Exception e) {
e.printStackTrace();
Toast.makeText(this, e.toString(), Toast.LENGTH_SHORT).show();
}}
}

```



### 13. Regis.java

```
package halo.source;
import java.util.ArrayList;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;
public class Regis extends Activity {
    EditText idpolisi;
    EditText password;
    EditText nama;
    Spinner spinner1;
    EditText daerahtugas;
    EditText jabatan;
    Button submit;
    Button cancel;
    String url = koneksi.url;
    @Override
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView(R.layout.regis);
        Database dbUser = new Database(this);
        dbUser.open();
        dbUser.createUser();
        dbUser.close();
        idpolisi = (EditText) findViewById(R.id.idpolisi);
        password = (EditText) findViewById(R.id.password);
        nama = (EditText) findViewById(R.id.nama);
        daerahtugas =
(EditText) findViewById(R.id.daerahtugas);
        jabatan = (EditText) findViewById(R.id.jabatan);
        spinner1 = (Spinner) findViewById(R.id.gender);
        submit = (Button) findViewById(R.id.submit);
        cancel = (Button) findViewById(R.id.cancel);
        buttonListener();
    }
    private void buttonListener() {
        submit.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if(idpolisi.getText().toString().equals("")) {
                    Toast.makeText(Regis.this, "Masukan ID",
                    Toast.LENGTH_SHORT).show();
                }
                else if(password.getText().toString().equals("")) {
```

```

Toast.makeText(Regis.this, "Masukan Password",
Toast.LENGTH_SHORT).show();}
else if(nama.getText().toString().equals("")) {
Toast.makeText(Regis.this, "Masukan Nama",
Toast.LENGTH_SHORT).show();}
else if(daerahtugas.getText().toString().equals("")) {
Toast.makeText(Regis.this, "Masukan daerah tugas",
Toast.LENGTH_SHORT).show(); }
else if(jabatan.getText().toString().equals("")) {
Toast.makeText(Regis.this, "Masukan Jabatan",
Toast.LENGTH_SHORT).show();}
else if(spinner1.getSelectedItem().toString().equals("")) {
Toast.makeText(Regis.this, "Masukan gender",
Toast.LENGTH_SHORT).show();} else {
process(); } } } } }
protected void process() {
ArrayList<NameValuePair> regis = new ArrayList<NameValuePair>();
regis.add(new BasicNameValuePair("function", "regis"));
regis.add(new BasicNameValuePair("idpolisi",
idpolisi.getText().toString()));
regis.add(new BasicNameValuePair("password",
password.getText().toString()));
regis.add(new BasicNameValuePair("nama",
nama.getText().toString()));
regis.add(new BasicNameValuePair("daerahtugas",
daerahtugas.getText().toString()));
regis.add(new BasicNameValuePair("jabatan",
jabatan.getText().toString()));
regis.add(new BasicNameValuePair("spinner1",
spinner1.getSelectedItem().toString()));
String respon = null;
    try {
        respon = koneksi.eksekusiHttpPost(url, regis);
        String res = respon.toString().trim();
        if(res.equals("0")){
Toast.makeText(this, "Gagal", Toast.LENGTH_SHORT).show();}
else{
        Toast.makeText(this, "Sukses",
Toast.LENGTH_SHORT).show();
Intent i = new Intent(this, Utama.class);
startActivity(i);
finish();}}
    catch(Exception e) {
e.printStackTrace();
Toast.makeText(this, e.toString(), Toast.LENGTH_SHORT).show();} }
}

```