



# PETIR

JURNAL  
PENGKAJIAN DAN PENERAPAN  
TEKNIK INFORMATIKA

VOLUME 9 - NOMOR 1

MARET 2016

ISSN 1978-9262

PENENTUAN NASABAH PENERIMA REWARD PRODUK *GOLD* DENGAN METODE *SIMPLE ADDITIVE WEIGHTING* (SAW)  
STUDI KASUS : PT. PINJAM INDONESIA

*Riki Ruli A. Siregar; Faizal Fachrurrozi*

IMPLEMENTASI METODE *BACKWARD CHAINING* PADA DATA *WAREHOUSE* DAOP 1 JAKARTA PT. KAI (PERSERO)

*Rakhmat Arianto; Chandra Bagus Sugiarto*

IMPLEMENTASI METODE DETEKSI TEPI *CANNY* PADA OBJEK SEBAGAI MODEL KEAMANAN APLIKASI PADA  
SMARTPHONE ANDROID

*Abdul Haris; Andi Prasetyo*

ANALISA DATA DAN PERANCANGAN APLIKASI SERVICE PELANGGAN PT. JNE UNTUK PENINGKATAN KUALITAS  
LAYANAN

*Dewi Arianti Wulandari; Sonny Syahrindra Putra*

JARINGAN AD-HOC VEHICULAR (VANET) : TINJAUAN TENTANG ARSITEKTUR, KARAKTERISTIK, APLIKASI, DAN  
PROTOKOL *MEDIUM ACCESS CONTROL* (MAC)

*Rosida Nur Aziza*

PERANCANGAN APLIKASI PENJADWALAN MATA KULIAH MENGGUNAKAN METODE *CONSTRAINT PROGRAMMING*

*Syam Gunawan*

RANCANGAN TATA KELOLA PEREMAJAAN RUANG KELAS DIREKTORAT PEMBINAAN SEKOLAH DASAR

*Ratna Mutu Manikam; Purwanto*

PENGEMBANGAN AMORIK MENGGUNAKAN METODE GARIS SINGGUNG TERHADAP DUA LINGKARAN DAN PERSAMAAN  
KURVA *BEZIER* ORDE DUA.

*Darma Rusjdi*

OPTIMALISASI PENGAMBILAN KEPUTUSAN PENILAIAN KINERJA DOSEN PADA PERGURUAN TINGGI ISLAM XYZ  
MENGGUNAKAN *AHP* (ANALYTICAL HIERARCHY PROCESS)

*Rahma Farah Ningrum*

PENGAMANAN SMS PADA TELEPON SELULER BERBASIS ANDROID MENGGUNAKAN ALGORITMA *TRIPLE DES*

*Raka Yusuf; M. Rival Suheri*

APLIKASI LATIHAN SOAL UJIAN TEORI SURAT IZIN MENGEMUDI BERBASIS WEB

*Harni Kusniyati; Raka Yusuf; Andri Setiawan*

RANCANG BANGUN SIMULASI TERJADINYA LISTRIK DENGAN SUMBER DAYA SAMPAH BERBASIS MULTIMEDIA  
(STUDI KASUS : TPST BANTAR GEBANG)

*Yasni Djamain; Ika Fitriyani Putri*

ISSN 1978-9262



771978 926272

SEKOLAH TINGGI TEKNIK - PLN (STT-PLN)

PETIR

VOL. 9

NO. 1

HAL. 1 - 87

JAKARTA, MARET 2016

ISSN 1978-9262

# APLIKASI LATIHAN SOAL UJIAN TEORI SURAT IZIN MENGEMUDI BERBASIS WEB

Harni Kusniyati; Raka Yusuf; Andri Setiawan

Program Studi Sistem Informasi Fasilkom Universitas Mercu Buana  
E-mail : harni246@gmail.com; raka@mercubuana.ac.id; contact@andry.my.id

## ABSTRAK

Semua pengemudi tentu akan merasa tenang apabila telah memiliki Surat Izin Mengemudi sebagai salah satu syarat mengendarai kendaraan bermotor. Dalam kenyataannya, masih ada masyarakat yang kesulitan mengerjakan ujian teori yang diselenggarakan oleh kepolisian. Diharapkan dengan aplikasi ini dapat dijadikan pembelajaran dan membantu menghadapi ujian teori, khusus untuk masyarakat yang ingin membuat Surat Izin Mengemudi. Aplikasi ini dibangun dengan menggunakan metode pengembangan perangkat lunak Waterfall dan menggunakan UML untuk dokumentasi, spesifikasi, dan pemodelan sistem. Dalam implementasinya, aplikasi ini dibuat dengan menggunakan PHP Framework (CodeIgniter) dan User Interface Framework (Bootstrap) serta menggunakan database MySQL untuk menyimpan data soal-soal latihan. Pengguna aplikasi ini dilengkapi dengan fitur koreksi, jika diaktifkan, pengguna dapat mengoreksi jawaban setiap soal yang ditampilkan.

**Kata Kunci:** Surat Izin Mengemudi, Ujian Teori, Latihan Soal

## I. PENDAHULUAN

### 1.1. Latar Belakang

Surat Izin Mengemudi (SIM) berfungsi sebagai bukti kompetensi (kemampuan) seseorang dalam mengemudi kendaraan bermotor dan juga sebagai registrasi data pengemudi kendaraan bermotor yang digunakan untuk mendukung penyelidikan, penyidikan dan identifikasi forensik kepolisian. Setiap orang wajib memiliki SIM untuk dapat mengemudikan kendaraan bermotor. Prosedur pembuatan SIM terbilang mudah dan praktis, salah satunya adalah mengikuti ujian teori. Namun, tidak sedikit calon pengemudi yang gagal dalam ujian teori ini dikarenakan tidak ada persiapan dengan latihan soal-soal ujian teori untuk mendapatkan SIM.

### 1.2. Rumusan Masalah

Berdasarkan latar belakang permasalahan di atas dapat dirumuskan menjadi beberapa masalah sebagai berikut:

1. Bagaimana merancang dan membangun sistem perangkat lunak berbasis web menghadapi ujian teori pembuatan Surat Izin Mengemudi?
2. Bagaimana menerapkan fitur pengecekan jawaban pada soal-soal yang ditampilkan?

### 1.3. Batasan Masalah

Berdasarkan rumusan masalah, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Rancang bangun aplikasi latihan soal ujian teori untuk mendapatkan Surat Izin Mengemudi hanya terbatas pada satu golongan, yaitu Golongan SIM A.
2. Halaman pengguna tidak memiliki fitur login. Pengguna dapat mengoreksi jawaban benar atau salah di setiap soal yang ditampilkan, serta menampilkan nilai persentase. Dimana

nilai yang diperoleh tidak dimasukkan dalam database.

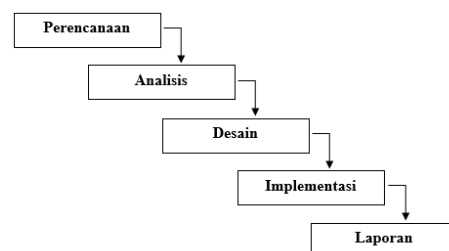
3. Halaman admin memiliki fitur login, penambahan, perubahan, dan penghapusan soal-soal latihan.
4. Soal-soal pada aplikasi ini berupa teks maupun gambar dan sistem menampilkan soal secara acak (*random*).
5. Aplikasi berbasis web ini hanya bisa berjalan di peramban internet (*browser*). Tampilan lebih sempurna jika dijalankan pada *browser desktop*, bukan *browser mobile*.

### 1.4. Tujuan dan Manfaat

Tujuan dari penelitian ini melakukan rancang bangun aplikasi latihan soal yang dilengkapi fitur pengecekan jawaban. Adapun manfaat dari penelitian ini untuk memudahkan seseorang atau calon pengemudi kendaraan bermotor dalam menghadapi ujian teori untuk mendapatkan Surat Izin Mengemudi dalam bentuk soal-soal latihan berbasis web.

### 1.5. Metodologi Penelitian

Dalam melakukan penelitian, penulis menggunakan metode rekayasa perangkat lunak model *waterfall development*. *Waterfall development* merupakan salah satu metode pengembangan perangkat lunak yang banyak digunakan.



Gambar 1. Model Waterfall Development

Adapun uraian dari masing-masing tahapan sistem adalah sebagai berikut:

1. Perencanaan  
Merupakan tahapan untuk mengetahui kenapa sistem tersebut dibangun dan menjelaskan pengalokasian sumber daya untuk membangun sistem tersebut. Kegiatan yang dilakukan dalam tahapan ini adalah inisiasi proyek, analisa kelayakan dan manajemen proyek.
2. Analisis  
Merupakan tahapan untuk mengetahui apa yang akan dilakukan oleh sistem, siapa yang menggunakannya, dimana dan kapan sistem tersebut digunakan. Dan dalam hal ini juga dianalisa kebutuhan sistem secara keseluruhan dan rinci.
3. Desain  
Merupakan tahapan untuk mengetahui bagaimana sistem akan beroperasi, meliputi perangkat keras (*hardware*), perangkat lunak (*software*), basis data (*database*), program, antarmuka pengguna (*user interface*) yang dibutuhkan oleh sistem tersebut. Kegiatan yang dilakukan dalam tahapan ini adalah strategi desain, arsitektur sistem, spesifikasi basis data dan file, serta kode program.
4. Implementasi  
Tahapan proses untuk mengeksekusi sistem secara nyata. Kegiatan yang dilakukan dalam tahapan ini adalah konstruksi, instalasi dan support.
5. Laporan  
Dalam tahap ini penyusun melakukan dokumentasi untuk semua bahan, data modul-modul yang telah selesai dikerjakan dari tahap awal sampai dengan tahap akhir pengerjaan yang berhubungan dengan sistem yang dikerjakan dalam bentuk laporan tugas akhir.

## II. LANDASAN TEORI

### 2.1. Pengertian Website

Web merupakan media informasi berbasis jaringan komputer yang dapat diakses di mana saja dengan biaya relative murah. Web merupakan bentuk implementasi dari bahasa pemrograman web (*web programming*). Sejarah perkembangan bahasa pemrograman web diawali dengan munculnya HTML (*Hypertext Markup Language*), yang kemudian dikembangkan dengan munculnya CSS (*Cascading Style Sheet*) yang bertujuan untuk memperindah tampilan website [9].

Informasi yang disajikan dalam halaman web menggunakan konsep multimedia, informasi dapat disajikan dengan menggunakan banyak media (teks, gambar, animasi, suara (*audio*), dan atau video. Dalam suatu halaman web, informasi akan dapat disajikan dalam kombinasi media teks, gambar, animasi, suara, atau video, semuanya dapat disajikan dalam satu halaman.

### 2.1.1. Jenis-jenis Website

Secara umum, situs web digolongkan menjadi 3 jenis yaitu sebagai berikut:

- a. Website Statis  
Website Statis adalah web yang mempunyai halaman tidak berubah. Artinya adalah untuk melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengedit code yang menjadi struktur dari situs tersebut.
- b. Website Dinamis  
Website Dinamis merupakan website yang secara struktur diperuntukan untuk update sesering mungkin. Biasanya selain utama yang bisa diakses oleh user pada umumnya, juga disediakan halaman admin untuk mengedit konten.
- c. Website Interaktif  
Website Interaktif adalah web yang saat ini memang sedang *booming*. Salah satu contoh website interaktif adalah blog dan forum. Di website ini user bisa berinteraksi dan beradu argument mengenai apa yang menjadi pemikiran mereka.

### 2.1.2. Aplikasi Web

Aplikasi web adalah aplikasi yang disimpan dan dieksekusi di lingkungan web server. Setiap permintaan yang dilakukan oleh user melalui aplikasi klien (*web browser*) akan direspon oleh aplikasi web dan hasilnya akan dikembalikan lagi ke hadapan user [5].

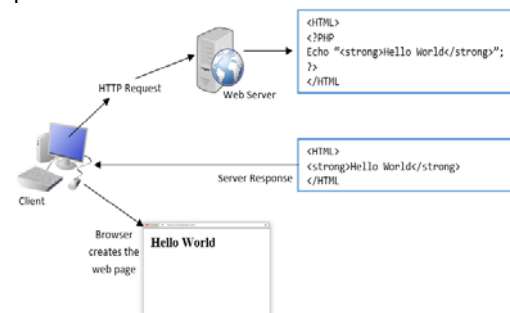


Gambar 2. Komunikasi Antara Web Browser dan Aplikasi Web

## 2.2. Bahasa Pemrograman PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman berbasis web yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis. PHP dapat dikatakan sebagai sebuah *server-side embedded script language*, artinya semua sintaks dan perintah program akan sepenuhnya dijalankan oleh server, tetapi dapat disertakan pada halaman HTML biasa [9].

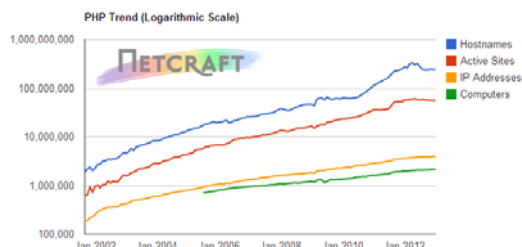
Pada prinsipnya, server akan bekerja apabila ada permintaan dari client. Dalam hal ini, client menggunakan kode-kode PHP akan mengirim permintaan ke server.



Gambar 3. Siklus PHP dalam Web Server

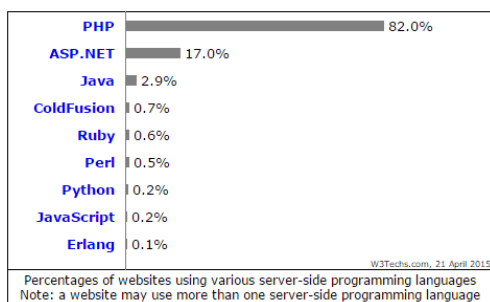
### 2.2.1. Statistik Penggunaan PHP

Berdasarkan laporan Netcraft, pada Januari 2013 bahasa pemrograman PHP digunakan kurang lebih 244 juta oleh situs web dari jumlah sekitar 629 juta hasil survey Netcraft.



Gambar 4. Grafik Penggunaan PHP

Diagram di bawah ini menunjukkan persentase website yang menggunakan berbagai bahasa pemrograman *server-side*.



Gambar 5 Diagram Penggunaan PHP

### 2.3. Basis Data

Basis data atau Database adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap database mempunyai API (*Application Programming Interface*) tertentu untuk membuat, mengakses, mengatur, mencari dan menyalin data yang ada di dalamnya [2].

MySQL merupakan suatu database. MySQL dapat juga dikatakan sebagai database yang sangat cocok bila dipadukan dengan PHP. Secara umum, database berfungsi sebagai tempat atau wadah untuk menyimpan, mengklasifikasikan data secara profesional. MySQL bekerja menggunakan SQL Language (*Structure Query Language*).

### 2.4. Kerangka Kerja (Framework)

Kerangka Kerja atau *Framework* merupakan banyak kode yang disimpan dalam beberapa file yang terpisah, dan memudahkan dalam penggunaan kode yang digunakan secara berulang-ulang [2].

#### 2.4.1. PHP Framework (CodeIgniter)

CodeIgniter adalah aplikasi close source yang berupa framework dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis dengan menggunakan PHP. *CodeIgniter* memudahkan *developer* untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal.

### 2.4.2. User Interface Framework (Bootstrap)

Bootstrap merupakan *framework* untuk membangun desain web secara responsif. Artinya, tampilan web yang dibuat oleh *bootstrap* akan menyesuaikan ukuran layar dari *browser* yang kita gunakan baik di desktop, tablet ataupun *mobile device*. *Bootstrap* dibangun dengan teknologi HTML dan CSS yang dapat membuat *layout* halaman website, tabel, tombol, form, navigasi, dan komponen lainnya dalam sebuah website hanya dengan memanggil fungsi CSS (*class*) dalam berkas HTML yang telah didefinisikan.

### 2.5. Unified Modeling Language (UML)

*Unified Modeling Language* merupakan salah satu alat bantu yang dapat digunakan dalam bahasa pemrograman berorientasi objek, saat ini UML sudah menjadi standar masa depan bagi industri pengembangan sistem/perangkat lunak yang akan berorientasi objek, sebab pada dasarnya UML telah digunakan oleh banyak perusahaan raksasa seperti IBM, Microsoft, dan sebagainya.


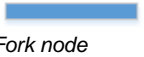
#### 2.5.1. Notasi Dasar UML

Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan.

- Actor**  
*Actor* adalah segala sesuatu yang berinteraksi langsung dengan sistem seperti orang, benda atau lainnya.
- Class**  
Notasi utama dan yang paling mendasar pada diagram UML adalah notasi untuk mempresentasikan suatu class beserta dengan atribut dan operasinya.
- Use Case**  
Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan sistem.
- Activity Diagram**  
*Activity diagram* menggambarkan berbagai alur aktifitas dalam sebuah sistem yang sedang dirancang.

Tabel 1 Notasi *Activity Diagram* pada UML

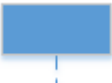



No.	Notasi	Keterangan
1	 <i>Initial node</i>	Titik awal.
2	 <i>Final node</i>	Menunjukkan titik akhir dari activity diagram.
3	 <i>Decision</i>	Pilihan untuk mengambil keputusan. Mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.

4	 Action	Menggambarkan sebuah pekerjaan/aktivitas dalam workflow.
5	 Fork node	Menunjukkan adanya percabangan secara paralel dari aktivitas.

### 5. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

Tabel 2. Notasi Sequence Diagram pada UML

No.	Notasi	Keterangan
1	 Object lifeline	Sebuah objek yang berpartisipasi secara berurutan dengan mengirimkan atau menerima pesan.
2	 Activation	Menandakan ketika suatu objek mengirim atau menerima pesan.
3	 Message	Menyatakan bahwa suatu objek mengirimkan masukan ke objek lainnya.
4	 Return message	Objek menghasilkan suatu kembalikan ke objek tertentu.

## III. ANALISIS DAN PERANCANGAN

### 3.1. Analisis

Mengendarai kendaraan tidak sembarangan, ada aturan-aturan yang harus ditaati dan juga syarat-syarat tertentu yang harus dipenuhi sebelum berkendara di jalanan umum. Karena jalan umum adalah milik bersama maka wajib bagi semua pengendara mematuhi aturan-aturan yang telah ditetapkan. Tentu saja aturan dibuat ada tujuannya, misalnya untuk meminimalisir hal-hal buruk yang bisa terjadi sehingga akan tercipta keselamatan di jalan. Sebelum berkendara, salah satu syarat yang harus dimiliki oleh pengendara adalah Surat Ijin Mengemudi (SIM).

Salah satu syarat untuk mendapatkan Surat Ijin Mengemudi (SIM) di setiap golongan adalah lulus ujian atau tes teori yang diadakan oleh kepolisian. Ujian teori yang dilakukan menyangkut pengetahuan tentang berlalu-lintas, ketentuan umum tentang lalu-lintas hingga peraturan lalu-lintas lainnya.

### 3.2. Pemodelan

Pemodelan (*modeling*) adalah proses merancang sistem atau perangkat lunak sebelum pengkodean (*coding*). Bahasa pemodelan standar untuk sistem atau perangkat lunak adalah UML (*Unified Modeling Language*). UML digunakan untuk menggambarkan, mendeskripsikan,

merancang, dan mendokumentasikan sistem perangkat lunak.

### 3.2.1. Diagram Use Case

Diagram Use Case menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Diagram use case merepresentasikan sebuah interaksi antara aktor dengan sistem. Aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan tertentu.

### 3.2.2. Deskripsi Use Case

Setelah menentukan use case yang ada pada sistem, maka penjelasan masing-masing use case yang terdapat pada Diagram Use Case dapat dilihat pada tabel-tabel di bawah ini.

Tabel 3 Deskripsi Use Case Mengerjakan Soal

Use Case	: Mengerjakan Soal	ID : 001
Aktor	: Pengguna	
Deskripsi :		
1. Pengguna membuka halaman utama		
2. Pengguna mengklik tombol mulai pada halaman utama		
3. Sistem akan menampilkan soal secara acak dan tombol pilihan jawaban		
4. Pengguna mengerjakan soal dengan memilih jawaban		
5. Pengguna mengklik tombol lanjut untuk beralih ke soal selanjutnya		

Tabel 4 Deskripsi Use Case Mengoreksi Jawaban

Use Case	: Mengoreksi Jawaban	ID : 002
Aktor	: Pengguna	
Deskripsi :		
1. Pengguna membuka halaman utama		
2. Pengguna mengaktifkan fitur pengecekan dan mengklik tombol mulai pada halaman utama		
3. Sistem akan menampilkan soal secara acak dan tombol pilihan jawaban serta mengaktifkan tombol koreksi		
4. Pengguna memilih jawaban dan mengklik tombol koreksi		

Tabel 5 Deskripsi Use Case Melihat Hasil

Use Case	: Melihat Hasil	ID : 003
Aktor	: Pengguna	
Deskripsi :		
1. Pengguna membuka halaman utama		
2. Pengguna mengklik tombol mulai pada halaman utama		
3. Sistem akan menampilkan soal secara acak dan tombol pilihan jawaban		
4. Pengguna mengerjakan soal dan mengklik tombol lanjut untuk beralih ke soal selanjutnya		
5. Setelah mengerjakan soal nomor 25, pengguna mengklik tombol lanjut dan sistem akan menampilkan hasil		

Tabel 6 Deskripsi Use Case Menambah Soal

Use Case	: Menambah Soal	ID : 004
Aktor	: Administrator	
Deskripsi :	<ol style="list-style-type: none"> <li>1. Administrator melakukan login ke dalam sistem melalui halaman login administrator.</li> <li>2. Sistem menampilkan halaman indeks soal yang berisi daftar soal-soal</li> <li>3. Administrator mengklik tombol tambah soal</li> <li>4. Sistem menampilkan form menambah soal</li> <li>5. Administrator mengisi form dan mengklik tombol simpan</li> </ol>	

Tabel 7 Deskripsi Use Case Mengubah Soal

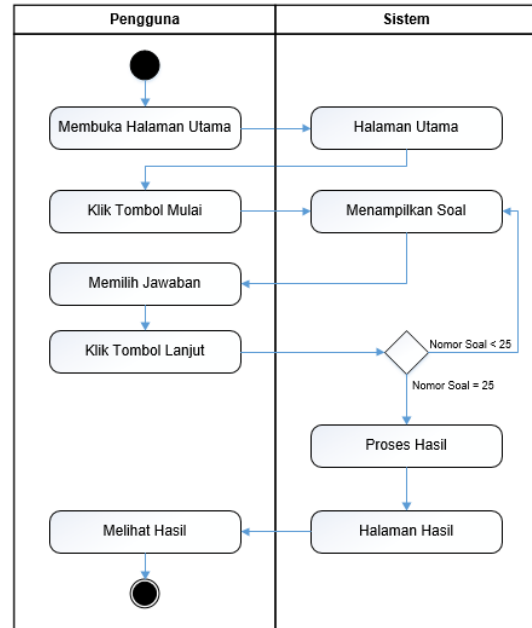
Use Case	: Mengubah Soal	ID : 005
Aktor	: Administrator	
Deskripsi :	<ol style="list-style-type: none"> <li>1. Administrator melakukan login ke dalam sistem melalui halaman login administrator.</li> <li>2. Sistem menampilkan halaman indeks soal yang berisi daftar soal-soal</li> <li>3. Administrator mengklik tombol ubah pada daftar soal yang dipilih</li> <li>4. Sistem menampilkan halaman mengubah soal</li> <li>5. Administrator mengubah soal dan mengklik tombol simpan</li> </ol>	

Tabel 8 Deskripsi Use Case Menghapus Soal

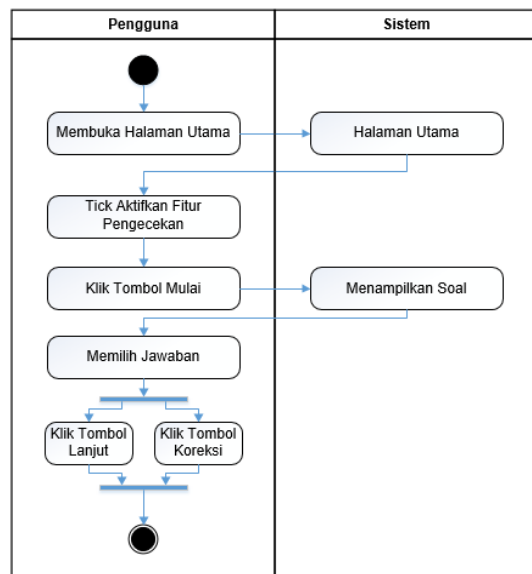
Use Case	: Menghapus Soal	ID : 006
Aktor	: Administrator	
Deskripsi :	<ol style="list-style-type: none"> <li>1. Administrator melakukan login ke dalam sistem melalui halaman login administrator.</li> <li>2. Sistem menampilkan halaman indeks soal yang berisi daftar soal-soal</li> <li>3. Administrator mengklik tombol hapus pada daftar soal yang dipilih</li> </ol>	

### 3.2.3. Activity Diagram

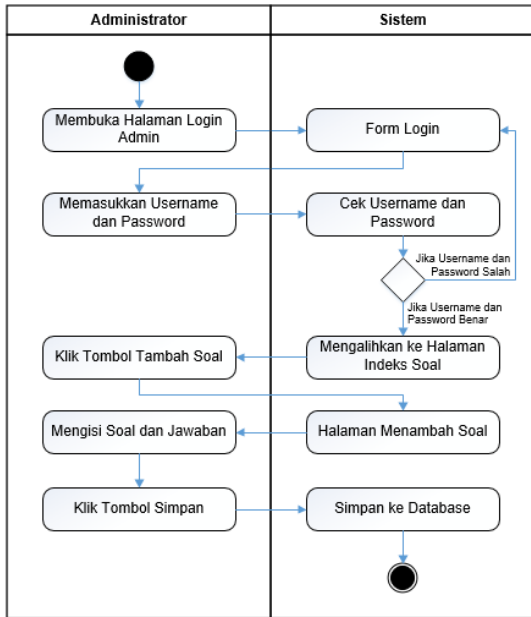
*Activity Diagram* memodelkan alur kerja (*workflow*) sebuah urutan aktivitas pada suatu proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan prosedur logika, proses bisnis dan alur kerja. Dalam hal ini, *activity diagram* akan menggambarkan urutan aktifitas yang terjadi pada proses dalam aplikasi soal latihan ujian teori SIM. Berikut masing-masing *activity diagram* dari setiap proses:



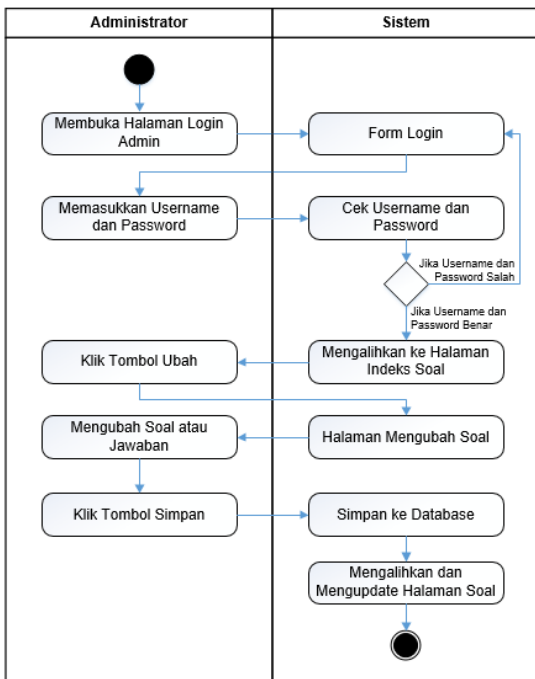
Gambar 6 Activity Diagram Mengerjakan Soal dan Melihat Hasil



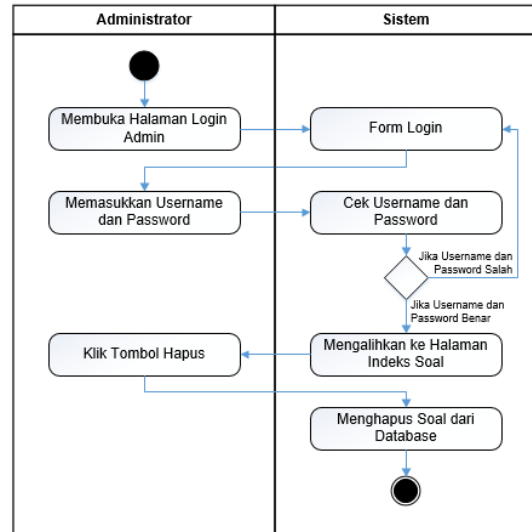
Gambar 7 Activity Diagram Mengoreksi Jawaban



Gambar 8 Activity Diagram Menambah Soal



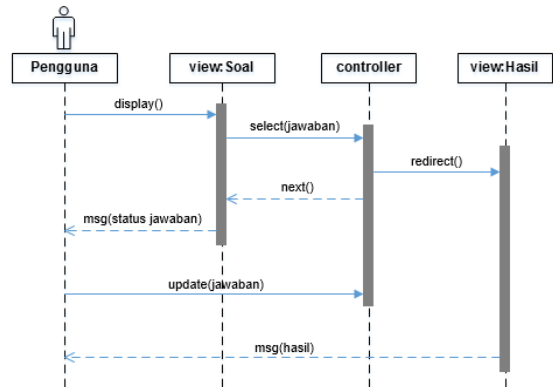
Gambar 9 Activity Diagram Mengubah Soal



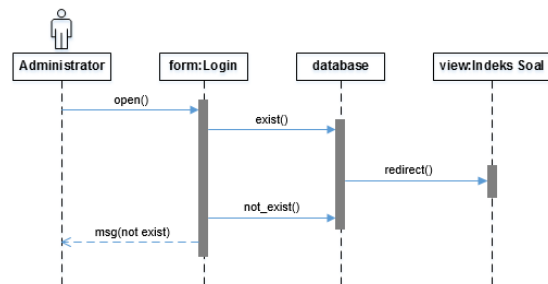
Gambar 10 Activity Diagram Menghapus Soal

### 3.2.4. Sequence Diagram

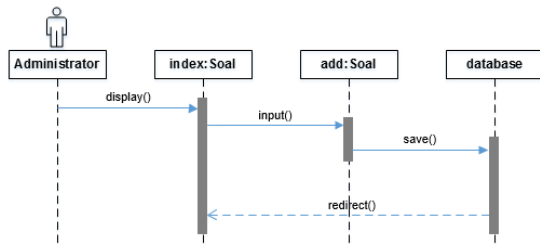
Sequence Diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem berupa message yang digambarkan terhadap waktu. Diagram ini secara khusus berasosiasi dengan diagram use case dan memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam use case.



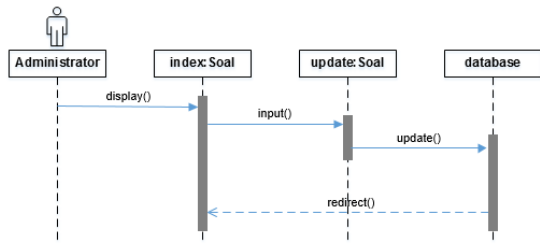
Gambar 11 Sequence Diagram Pengguna Menjerakan, Mengoreksi dan Melihat Hasil



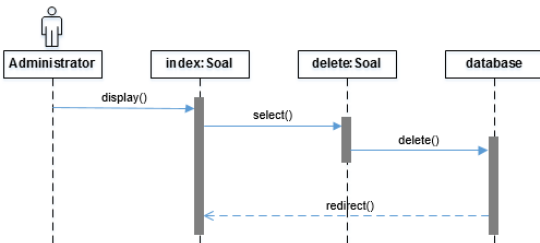
Gambar 12 Sequence Diagram Login



Gambar 13 Sequence Diagram Menambah Soal



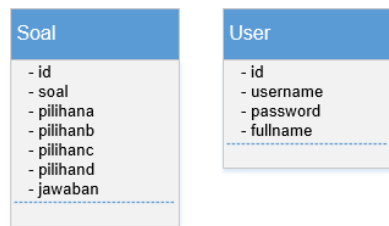
Gambar 14 Sequence Diagram Mengubah Soal



Gambar 15 Sequence Diagram Menghapus Soal

### 3.2.5. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Pada umumnya, class diagram mempresentasikan hubungan (*relationship*) antar kelas. Dalam kasus ini tidak ditemukan adanya hubungan relasi antar kelas.



Gambar 16 Class Diagram Aplikasi Latihan Soal Ujian Teori

### 3.3. Struktur Tabel Basis Data

Struktur Tabel merupakan penjabaran dan penjelasan dari suatu database. Dalam struktur Tabel dijelaskan fungsi dari masing-masing Tabel hingga fungsi masing-masing *field* yang ada di dalam Tabel.

#### 3.3.1. Tabel soal

Nama Tabel : soal  
 Primary Key : id  
 Foreign Key : -  
 Fungsi : Untuk menyimpan data soal

Tabel 9 Struktur Database Tabel Soal

No	Field	Type Data	Keterangan
1	id	integer	Kode identitas soal
2	soal	text	Materi soal latihan
3	pilihana	text	Pilihan jawaban dari soal
4	pilihana	text	Pilihan jawaban dari soal
5	pilihanc	text	Pilihan jawaban dari soal
6	pilihand	text	Pilihan jawaban dari soal
7	jawaban	char	Jawaban yang tepat dari soal

#### 3.3.2. Tabel user

Nama Tabel : user  
 Primary Key : id  
 Foreign Key : -  
 Fungsi : Untuk menyimpan data user

Tabel 10 Struktur Database Tabel User

No	Field	Type Data	Keterangan
1	id	integer	Kode identitas user
2	username	varchar	Nama login user
3	password	varchar	Password user
4	fullname	varchar	Nama lengkap user

### 3.4. Perancangan Antar Muka

Setelah disusun dan dirancang struktur basis data, berikutnya adalah rancangan antar muka (*user-interface*) dari sistem yang dibangun. Dengan adanya rancangan antar muka ini, akan segera diketahui gambaran mendetail dari aplikasi latihan soal ujian teori SIM. Perancangan antar muka ini juga akan mewakili keadaan sebenarnya dari aplikasi yang akan dibangun.

## IV. IMPLEMENTASI DAN PENGUJIAN

### 4.1. Implementasi Sistem

Implementasi merupakan kelanjutan dari kegiatan perancangan sistem dan dapat dipandang sebagai usaha untuk mewujudkan sistem yang dirancang. Langkah-langkah dari proses implementasi adalah urutan dari kegiatan awal sampai kegiatan akhir yang harus dilakukan dalam mewujudkan sistem yang dirancang. Hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik.

#### 4.1.1. Spesifikasi Kebutuhan Sistem

Spesifikasi kebutuhan sistem pada area kerja dan pengujian terbagi menjadi 2 (dua) yaitu perangkat keras dan perangkat lunak yang dapat dilihat pada keterangan di bawah ini:

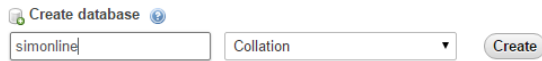
- Perangkat Keras
  - Prosesor Intel® Core™ i5-4210M (3M Cache, 2.6GHz)
  - Memori 4GB, PC3-12800 1600MHz DDR3L
  - Hardisk 500GB 7200 RPM
  - VGA Intel HD Graphics 4600
- Perangkat Lunak
  - Sistem Operasi, Windows 8.1 64-bit



- Peramban Internet, Google Chrome 44.0.2403.125
- XAMPP Web Server v5.6.8
  - Apache 2.4.12 (http server)
  - MySQL 5.6.24
  - PHP 5.6.8
  - phpMyAdmin 4.4.3
- Framework CodeIgniter 3.0.0 dan Bootstrap 3.3.5
- Text Editor CKEditor 4.5.1

#### 4.1.2. Pembuatan Database

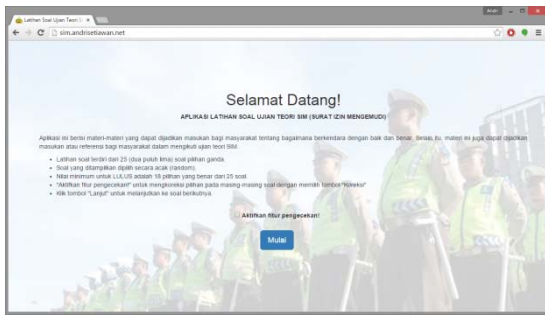
Pada tahap pertama dalam membangun aplikasi ini adalah pembuatan databasenya terlebih dahulu.



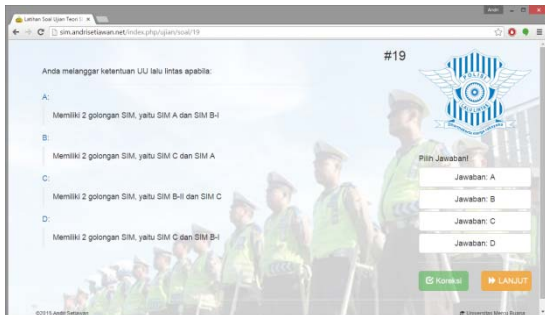
Gambar 17 Membuat Database dengan phpMyAdmin

#### 4.1.3. Antar Muka Sistem (Interface)

Berikut ini adalah implementasi antar muka dari aplikasi yang dibuat:



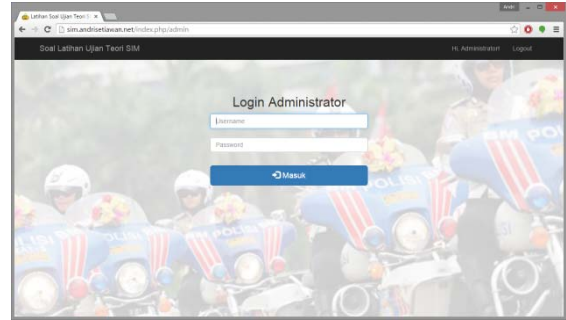
Gambar 18 Implementasi Halaman Utama Pengguna



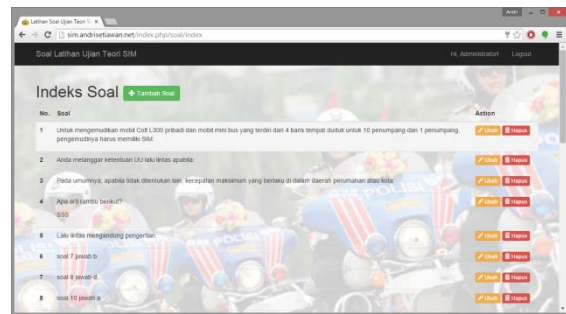
Gambar 19 Implementasi Halaman Soal Latihan



Gambar 20 Implementasi Halaman Hasil



Gambar 21 Implementasi Halaman Login Admin



Gambar 22 Implementasi Halaman Indeks Soal

#### 4.2. Pengujian Sistem

Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

##### 4.2.1. Pengujian Antar Muka Pengguna

Tabel 11 Daftar Pengujian pada Antar Muka Pengguna

Case ID	Data Masukan	Pengamatan	Kesimpulan
1	Klik tombol "Mulai".	Dapat menampilkan soal latihan.	[v] Diterima [ ] Ditolak
2	Check pada "Aktifkan fitur pengecekan!"	Dapat menampilkan soal latihan dan tombol "Koreksi" dapat digunakan	[v] Diterima [ ] Ditolak
3	Klik tombol "Lanjut" tanpa memilih jawaban.	Dapat menampilkan pesan kesalahan "Silakan pilih jawaban untuk melanjutkan!"	[v] Diterima [ ] Ditolak
4	Memilih jawaban dan klik tombol "Lanjut".	Dapat menampilkan soal selanjutnya.	[v] Diterima [ ] Ditolak
5	Klik tombol "Koreksi" tanpa memilih jawaban	Dapat menampilkan pesan kesalahan	[v] Diterima [ ] Ditolak

		"Anda belum memilih jawaban!"	
6	Klik tombol "Koreksi" dengan memilih jawaban yang salah.	Dapat menampilkan pesan "Jawaban SALAH"	[v] Diterima [ ] Ditolak
7	Klik tombol "Koreksi" dengan memilih jawaban yang benar.	Dapat menampilkan pesan "Jawaban BENAR"	[v] Diterima [ ] Ditolak
8	Klik tombol "Lanjut" pada soal nomor 25.	Dapat menampilkan hasil.	[v] Diterima [ ] Ditolak
9	Klik tombol "Mulai ulang" pada halaman hasil	Dapat kembali ke halaman utama pengguna	[v] Diterima [ ] Ditolak

		berhasil ditambahkan.	
15	Klik tombol "Batal" pada halaman tambah soal	Dapat kembali ke halaman Indeks Soal.	[v] Diterima [ ] Ditolak
16	Klik tombol "Ubah" pada halaman Indeks Soal	Dapat menampilkan halaman ubah soal.	[v] Diterima [ ] Ditolak
17	Klik tombol "Simpan" pada halaman ubah soal	Dapat kembali ke halaman Indeks Soal dan soal berhasil diubah.	[v] Diterima [ ] Ditolak
18	Klik tombol "Batal" pada halaman ubah soal	Dapat kembali ke halaman Indeks Soal.	[v] Diterima [ ] Ditolak
19	Klik tombol "Hapus" pada halaman Indeks Soal	Soal yang dipilih dapat terhapus dari halaman Indeks Soal	[v] Diterima [ ] Ditolak

#### 4.2.2. Pengujian Antar Muka Administrator

Tabel 12 Daftar Pengujian pada Antar Muka Administrator

Case ID	Data Masukan	Pengamatan	Kesimpulan
10	Klik tombol "Masuk" tanpa memasukkan username dan password	Dapat menampilkan pesan kesalahan "Username dan Password Salah!" pada halaman login administrator	[v] Diterima [ ] Ditolak
11	Memasukkan username dan password yang salah	Dapat menampilkan pesan kesalahan "Username dan Password Salah!" pada halaman login administrator	[v] Diterima [ ] Ditolak
12	Memasukkan username dan password. Klik tombol "Masuk"  Username: admin Password: AdminUMB13	Login sukses, dapat menampilkan halaman indeks soal	[v] Diterima [ ] Ditolak
13	Klik tombol "Tambah Soal" pada halaman Indeks Soal	Dapat menampilkan halaman menambah soal	[v] Diterima [ ] Ditolak
14	Klik tombol "Simpan" pada halaman tambah soal	Dapat kembali ke halaman Indeks Soal dan soal	[v] Diterima [ ] Ditolak

## V. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Dari uraian diatas, dapat diambil kesimpulan sebagai berikut:

1. Aplikasi latihan soal ujian teori Surat Izin Mengemudi berjalan dengan baik pada perambah internet (*browser*). Aplikasi ini dirancang dan dibangun menggunakan bahasa pemrograman PHP.
2. Pengguna dapat mengoreksi jawaban pada setiap soal yang ditampilkan secara acak oleh sistem dengan mengaktifkan fitur pengecekan jawaban di halaman utama pengguna.

### 5.2. Saran

Berikut ini adalah saran untuk pengembangan selanjutnya:

1. Penambahan kategori golongan Surat Izin Mengemudi, sehingga lebih tepat sasaran kepada pengguna yang ingin memahami soal-soal ujian teori dengan golongan SIM tertentu.
2. Tampilan dibuat lebih menarik dan interaktif. Misalnya penggunaan media suara (*sound*) pada saat pengguna memilih jawaban dan koreksi jawaban.
3. Aplikasi dapat dijalankan pada perangkat mobile (*mobile devices*). Meskipun aplikasi ini bisa diakses melalui *mobile browser*, akan lebih sempurna jika dibuatkan aplikasi berbasis mobile sehingga pengguna bisa mempelajari kapanpun dan dimanapun.

## DAFTAR PUSTAKA

- [1] Alatas, Husein. 2013. *Responsive Web Design dengan PHP dan Bootstrap*. Yogyakarta: Lokomedia
- [2] Jubilee Enterprise. 2015. *Membuat Website PHP dengan CodeIgniter*. Jakarta: Elex Media Komputindo
- [3] Lassoﬀ, Mark. 2014. *PHP and MySQL for Beginners*. Connecticut USA: LearnToProgram, Incorporated
- [4] Pratama, Antonius Nugraha Widhi. 2010. *CodeIgniter: Cara Mudah Membangun Aplikasi PHP*. Jakarta: Media Kita
- [5] Raharjo, Budi, Imam Heryanto, dan E. Rosdiana K. 2014. *Modul Pemrograman WEB (HTML, PHP, & MySQL)*. Bandung: Penerbit Modula
- [6] Sianipar, R.H. 2015. *Membangun Web PHP dan MySQL*. Bandung: Informatika Bandung
- [7] Sidik, Betha. 2012. *Framework CodeIgniter*. Bandung: Informatika Bandung
- [8] Siswosoediro, Henry S. 2009. *Buku Pintar Mengurus Surat & Dokumen Kendaraan Bermotor*. Jakarta: Visimedia
- Wahana Komputer. 2009. *ShortCourse Series: PHP Programming*. Yogyakarta: Andi Publisher