

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **VI.1. Kesimpulan**

Setelah perangkat lunak Inkost selesai diuji, maka dapat ditarik beberapa kesimpulan, yaitu:

1. Perangkat lunak Inkost telah berhasil dibangun menjadi *website* sistem informasi yang dapat memberikan layanan beriklan dengan baik.
2. Perangkat lunak Inkost telah berhasil dibangun menjadi *website* iklan kost yang dapat menjadi media promosi melalui internet.

## **VI.2. Saran**

Penulis ingin memberikan beberapa saran untuk pengembangan lebih lanjut perangkat lunak Inkost antara lain:

1. Sistem dapat dikembangkan agar dapat melakukan pembayaran secara online.
2. Sistem dapat dikembangkan tidak hanya terpatok pada daerah jogja saja lebih dikembangkan untuk daerah yang lebih luas mencakup seluruh pulau jawa atau bahkan Indonesia.
3. Juga bisa dikembangkan web versi mobile atau perangkat lainnya.
4. Pesan ditambah dengan notifikasi sehingga lebih mudah untuk diketahui.

## DAFTAR PUSTAKA

- Hartanto, Steven., *Pembangunan Sistem Informasi Keluarga Berencana Indonesia (SIPKBI)*. Universitas Atma Jaya, Yogyakarta, 2010.
- Whisnu, Andreas., *Pembangunan Sistem Informasi Rental Mobil Berbasis Web Dengan Framework Codeigniter (EXCARRENTIS)*. Universitas Atma Jaya, Yogyakarta, 2013.
- Saputra, Agus., *Trik Kolaborasi Codeigniter dan jQuery*, Lokomedia, 2012.
- Basuki, Awan Pribadi., *Membangun Web Berbasis PHP dengan Framework Codeigniter*, Locomedia, 2010.
- Saputra, Agus., Feni Agustin., *Pemrograman CSS Untuk Pemula*, PT. Elex Media Komputindo, 2011.
- Hakim, Lukmanul., *Trik Dasyat Menguasai Ajax dengan jQuery*, Lokomedia, 2011.
- Werdana, S.Hut., M.Si., *Menjadi Master PHP dengan Framework Codeigniter*, PT. Elex Media Komputindo, 2010.
- Wiskawarma, Komang., *9 Langkah menjadi Master Framework Codeigniter*, Lokomedia, 2011.

# SKPL

## SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK


### INKOST (INFORMATION KOST)

Untuk:  
Tempat Beriklan Kost

Dipersiapkan oleh:  
Ferry Sukarto [06 07 05106]

Program Studi Teknik Informatika - Fakultas Teknologi  
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<i>SKPL-INKOST</i>		
		Revisi		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

## Daftar Isi

1. Pendahuluan.....	6
1.1 Tujuan.....	6
1.2 Lingkup Masalah.....	6
1.3 Definisi, Akronim, dan Singkatan.....	7
1.4 Referensi.....	8
1.5 Deskripsi umum ( <i>Overview</i> ).....	9
2. Deskripsi Kebutuhan.....	9
2.1 Perspektif produk.....	9
2.2 Fungsi Produk.....	11
2.3 Karakteristik <i>User</i> .....	16
2.4 Batasan-batasan.....	16
2.5 Asumsi dan Ketergantungan.....	16
3. Kebutuhan khusus.....	16
3.1 Kebutuhan antarmuka eksternal.....	16
3.1.1 Antarmuka pemakai.....	17
3.1.2 Antarmuka perangkat keras.....	17
3.1.3 Antarmuka perangkat lunak.....	17
3.1.4 Antarmuka Komunikasi.....	18
3.2 Kebutuhan fungsionalitas Perangkat Lunak.....	18
3.2.1 <i>Use Case Diagram</i> .....	18
4. Spesifikasi Rinci Kebutuhan.....	19
4.1 Spesifikasi Kebutuhan Fungsionalitas.....	19
4.1.1 <i>Use Case Specification: Login</i> .....	19
4.1.2 <i>Use Case Specification:</i> Reset Password.....	20
4.1.3 <i>Use Case Specification:</i> Registrasi.....	21
4.1.4 <i>Use Case Specification:</i> Pengelolaan User.....	22
4.1.5 <i>Use Case Specification:</i> Pengelolaan iklan.....	23
4.1.6 <i>Use Case Specification:</i> Pengelolaan pesan.....	25
4.1.7 <i>Use Case Specification:</i> Pengelolaan berita pemberitahuan.....	27
4.1.8 <i>Use Case Specification:</i> Pengelolaan profile.....	29
4.1.9 <i>Use Case Specification:</i> Pengelolaan pembayaran.....	31
5. <i>Conceptual Data Model (CDM)</i> .....	33

## Daftar Gambar

1. Gambar 1	Arsitektur Perangkat Lunak INKOST.....	10
2. Gambar 2	<i>Use Case</i> Diagram.....	18
3. Gambar 3	<i>Conceptual</i> Data Model (CDM).....	33



## **1 Pendahuluan**

### **Tujuan**

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak INKOST (INFORMATION KOST) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain, perangkat lunak, perangkat keras, dan pengguna), perfomansi (kemampuan perangkat lunak dari segi kecepatan, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut (fitur tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-INKOST ini juga mendefinisikan batasan perancangan perangkat lunak.

### **Lingkup Masalah**

Perangkat Lunak INKOST dikembangkan dengan tujuan untuk:

10. Menangani pengelolaan data member.
11. Menangani pengelolaan data iklan.
12. Menangani pengelolaan data pesan.
13. Menangani pengelolaan data pemberitahuan.
14. Menangani pengelolaan profil pengguna.
15. Menangani pengelolaan konfirmasi pembayaran.
16. Menangani pencarian kost.
17. Menangani pengelolaan kritik dan saran.
18. Menangani pengelolaan contactus

## Definisi, Akronim, dan Singkatan

Daftar definisi akronim dan singkatan:

<i>Keyword/Phrase</i>	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-INKOST-XXX	Kode yang merepresentasikan kebutuhan pada INKOST (Information Kost) dimana XXX merupakan nomor fungsi produk.
INKOST	Perangkat lunak sistem informasi berbasis web untuk menangani pendataan semua informasi kost-kostan.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Web server	Software yang memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan browser web dan mengirimkan kembali hasilnya dalam bentuk halaman - halaman web yang umumnya berbentuk dokumen HTML.
Web Browser	Aplikasi perangkat lunak yang memungkinkan pengguna untuk menayangkan dan berinteraksi dengan tulisan, gambar, video, musik dan berbagai informasi lainnya yang terdapat pada halaman Web di sebuah situs di World Wide Web atau di jaringan LAN lokal.

## **Referensi**

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Hartanto, Steven., *Pembangunan Sistem Informasi Keluarga Berencana Indonesia (SIPKBI)*. Universitas Atma Jaya, Yogyakarta, 2010.
2. Whisnu, Andreas., *Pembangunan Sistem Informasi Rental Mobil Berbasis Web Dengan Framework Codeigniter (EXCARRENTIS)*. Universitas Atma Jaya, Yogyakarta, 2013.
3. Saputra, Agus., *Trik Kolaborasi Codeigniter dan jQuery*, Lokomedia, 2012.
4. Basuki, Awan Pribadi., *Membangun Web Berbasis PHP dengan Framework Codeigniter*, Locomedia, 2010.
5. Saputra, Agus., Feni Agustin., *Pemrograman CSS Untuk Pemula*, PT. Elex Media Komputindo, 2011.
6. Hakim, Lukmanul., *Trik Dasyat Menguasai Ajax dengan jQuery*, Lokomedia, 2011.
7. Werdana, S.Hut., M.Si., *Menjadi Master PHP dengan Framework Codeigniter*, PT. Elex Media Komputindo, 2010.

## **Deskripsi umum (Overview)**

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak INKOST yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik *User*, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak INKOST tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak INKOST yang akan dikembangkan.

## **Deskripsi Kebutuhan**

### **Perspektif produk**

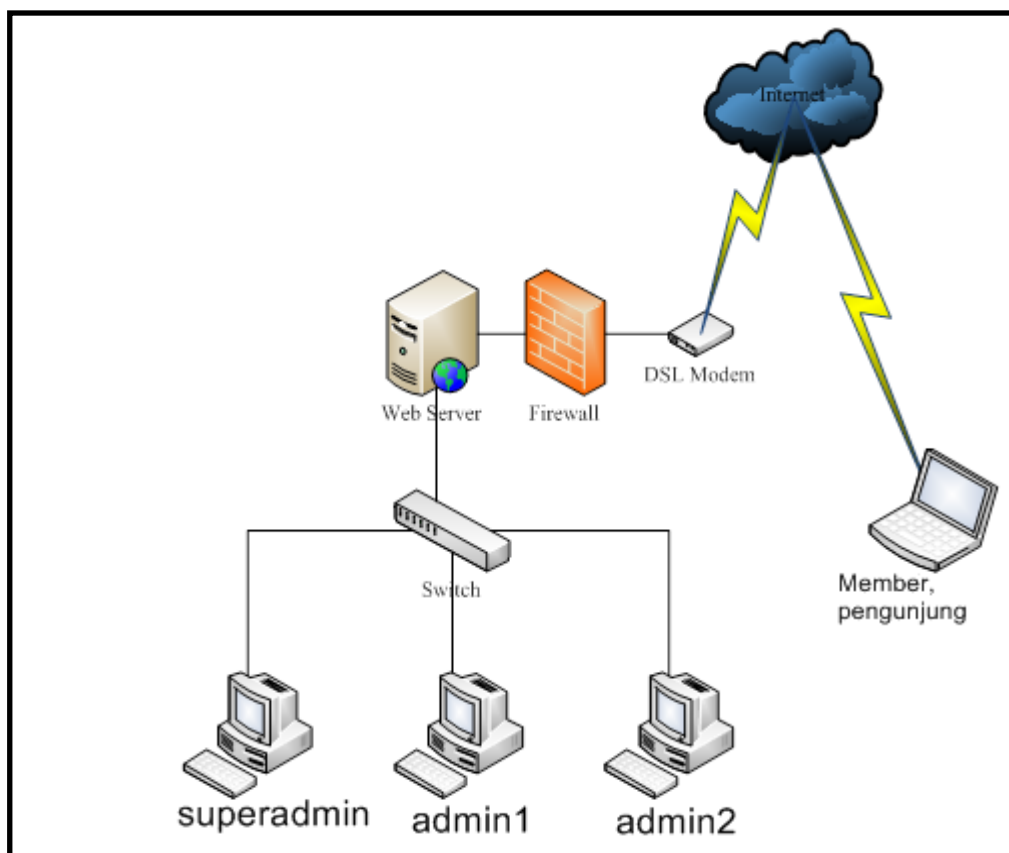
INKOST merupakan perangkat lunak yang dikembangkan untuk membantu pemilik kost dan para mahasiswa baru yang akan mencari kost. Sistem ini menangani pengelolaan member, pengelolaan kost, pengelolaan berita, pengelolaan pesan, pengelolaan data iklan, pencarian kost dan pengelolaan profil pengguna.

Perangkat lunak INKOST ini merupakan *web service* yang dapat diakses pada computer-computer yang terkoneksi dengan internet, dan dibuat menggunakan bahasa pemrograman PHP. Sedangkan untuk lingkungan pemrogramannya menggunakan *framework* Codeigniter dengan *tools* pemrograman *Eclipse*.

*User* akan berinteraksi dengan sistem melalui antarmuka GUI (*Graphical User Interface*). Pada sistem ini, seperti terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa *web server*, di mana semua data disimpan di *web server*. *User* dapat

mengakses data yang ada di *server* tersebut melalui *web browser* pada jaringan *intranet* yang tersedia.

Inputan data yang dimasukkan akan disimpan dalam *database web server*, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke *database web server* yang selanjutnya dikirimkan ke *client* yang *request* melalui jaringan.



Gambar 1. Arsitektur Perangkat Lunak INKOST

### **Fungsi Produk**

Fungsi produk perangkat lunak INKOST adalah sebagai berikut:

1. Fungsi *Login* (**SKPL-INKOST-001**).

Merupakan fungsi yang digunakan oleh semua pengguna untuk dapat masuk dalam sistem yang akan digunakan.

2. Fungsi *reset password* **(SKPL-INKOST-002)** .

Merupakan fungsi yang digunakan oleh pengguna yang Ketika lupa *password*.

3. Fungsi *Registrasi* **(SKPL-INKOST-003)** .

Merupakan fungsi yang digunakan oleh semua pengguna untuk mendaftar. Dengan menyimpan nama, email, foto, tanggal\_lahir dan password.

4. Fungsi *Pengelolaan profile* **(SKPL-INKOST-004)** .

Merupakan fungsi yang digunakan untuk mengelola data user sendiri.

Fungsi pengelolaan member mencakup:

a. Fungsi ganti password **(SKPL-INKOST-004-01)** .

Merupakan fungsi yang digunakan untuk mengganti password user.

b. Fungsi edit profile **(SKPL-INKOST-004-02)** .

Merupakan fungsi yang digunakan untuk mengubah data.

c. Fungsi ganti foto **(SKPL-INKOST-004-03)** .

Merupakan fungsi yang digunakan untuk emngganti foto user itu sendiri.

5. Fungsi *Pengelolaan user* **(SKPL-INKOST-005)** .

Merupakan fungsi yang digunakan untuk mengelola data user yang dilakukan oleh admin.

Fungsi pengelolaan user mencakup:

a. Fungsi hapus user **(SKPL-INKOST-005-01)** .

Merupakan fungsi yang digunakan untuk menghapus user.

b. Fungsi view user **(SKPL-INKOST-005-02)** .

Fungsi view user digunakan untuk melihat detail user.

6. Fungsi Pengelolaan iklan **(SKPL-INKOST-006)** .

Merupakan fungsi yang digunakan untuk mengelola data iklan baik dari pihak admin maupun dari sisi user.

Fungsi pengelolaan iklan mencakup:

a. Fungsi tambah iklan **(SKPL-INKOST-006-01)** .

Merupakan fungsi yang digunakan untuk menambahkan data iklan baru, fungsi ini hanya bias diakses oleh member.

b. Fungsi edit data iklan **(SKPL-INKOST-006-02)** .

Merupakan fungsi yang digunakan untuk mengubah data iklan, fungsi ini hanya bisa diakses oleh.

c. Fungsi delete iklan **(SKPL-INKOST-006-03)** .

Merupakan fungsi yang digunakan untuk mendelete iklan, hanya bias diakses oleh admin.

d. Fungsi view iklan **(SKPL-INKOST-006-04)** .

Merupakan fungsi yang digunakan untuk melihat iklan secara detail, fungsi ini bias diakses oleh admin dan member.

7. Fungsi berita pemberitahuan **(SKPL-INKOST-007)** .

Merupakan fungsi yang digunakan untuk mengelola berita. Berita ini menyampaikan pemberitahuan baru atau peraturan (rule baru) dari admin.

Fungsi pengelolaan berita mencakup:

a. Fungsi tambah berita **(SKPL-INKOST-007-01)**.

Merupakan fungsi yang digunakan untuk menambahkan berita.

b. Fungsi edit berita **(SKPL-INKOST-007-02)**.

Merupakan fungsi yang digunakan untuk mengubah data berita.

c. Fungsi view berita **(SKPL-INKOST-007-03)**.

Merupakan fungsi yang digunakan untuk melihat berita secara detail.

d. Fungsi hapus berita **(SKPL-INKOST-007-04)**.

Merupakan fungsi yang digunakan untuk menghapus data berita.

8. Fungsi Pengelolaan pesan **(SKPL-INKOST-008)**.

Merupakan fungsi yang digunakan untuk mengelola data pesan.

Fungsi pengelolaan pesan mencakup:

a. Fungsi tambah pesan **(SKPL-INKOST-008-01)**.

Merupakan fungsi yang digunakan untuk menambahkan data pesan atau mengirim pesan.

b. Fungsi delete pesan **(SKPL-INKOST-008-02)**.

Merupakan fungsi yang digunakan untuk menghapus pesan.



c. Fungsi balas pesan (**SKPL-INKOST-008-03**).

Merupakan fungsi yang digunakan untuk membalas pesan.

d. Fungsi view pesan (**SKPL-INKOST-008-04**).

Merupakan fungsi yang digunakan untuk melihat pesan secara detail.

9. Fungsi Pencarian (**SKPL-INKOST-009**).

Merupakan fungsi yang digunakan untuk mengelola data pencarian.

Fungsi Pencarian mencakup:

a. Fungsi cari berdasarkan daerah (**SKPL-INKOST-009-01**).

Merupakan fungsi yang digunakan untuk mencari kost berdasarkan daerah.

b. Fungsi cari berdasarkan kampus terdekat (**SKPL-INKOST-009-02**).

Merupakan fungsi yang digunakan untuk mencari kost berdasarkan kampus terdekat.

c. Fungsi cari jenis kost (**SKPL-INKOST-009-03**).

Merupakan fungsi yang digunakan untuk mencari kost berdasarkan jenis kost. Misalnya laki-laki, wanita dan campur.

d. Fungsi cari berdasarkan kamar mandi (**SKPL-INKOST-009-04**).

Fungsi ini digunakan untuk mencari kost berdasarkan kamar mandi, misalnya kamar mandi luar atau dalam.

e. Fungsi cari berdasarkan pemakaian AC (**SKPL-INKOST-009-05**).

Fungsi ini digunakan untuk mencari kost berdasarkan pemakaian ac atau tidak.

f. Fungsi cari berdasarkan lama sewa (**SKPL-INKOST-009-06**).

Merupakan fungsi yang digunakan untuk mencari kost berdasarkan lama sewa kost yang kita cari. Misalnya harian, bulanan dan tahunan.

10. Fungsi konfirmasi pembayaran (**SKPL-INKOST-010**).

Merupakan fungsi yang digunakan untuk mengelola data pembayaran.

Fungsi Pencarian mencakup:

a. Fungsi konfirmasi pembayaran (**SKPL-INKOST-010-01**).

Merupakan fungsi yang digunakan untuk konfirmasi pembayaran yaitu member.

b. Fungsi confirm request konfirmasi (**SKPL-INKOST-010-02**).

Merupakan fungsi yang digunakan untuk konfirmasi pembayaran member dari admin.

### **Karakteristik User**

Karakteristik dari *User* perangkat lunak INKOST adalah sebagai berikut:

1. Memahami pengoperasian Komputer.

2. Mengerti tentang internet dan memahami pengoperasian *web browser*.

### **Batasan-batasan**

Batasan-batasan dalam pengembangan perangkat lunak INKOST tersebut adalah:

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak INKOST.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

### **Asumsi dan Ketergantungan**

Sistem ini dapat dijalankan pada computer (PC, Laptop, Netbook, dan *smartphone*) yang memiliki aplikasi *web browser*.

### **Kebutuhan khusus**

#### **Kebutuhan antarmuka eksternal**

Kebutuhan antarmuka eksternal pada perangkat lunak INKOST meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

#### **Antarmuka pemakai**

*User* berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form pada *web browser*.

### **Antarmuka perangkat keras**

Antarmuka perangkat keras yang digunakan dalam perangkat lunak INKOST adalah:

1. Perangkat keras (PC, Laptop, Netbook, dan *smartphone*) yang memiliki aplikasi *web browser*.

### **Antarmuka perangkat lunak**

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak INKOST adalah sebagai berikut:

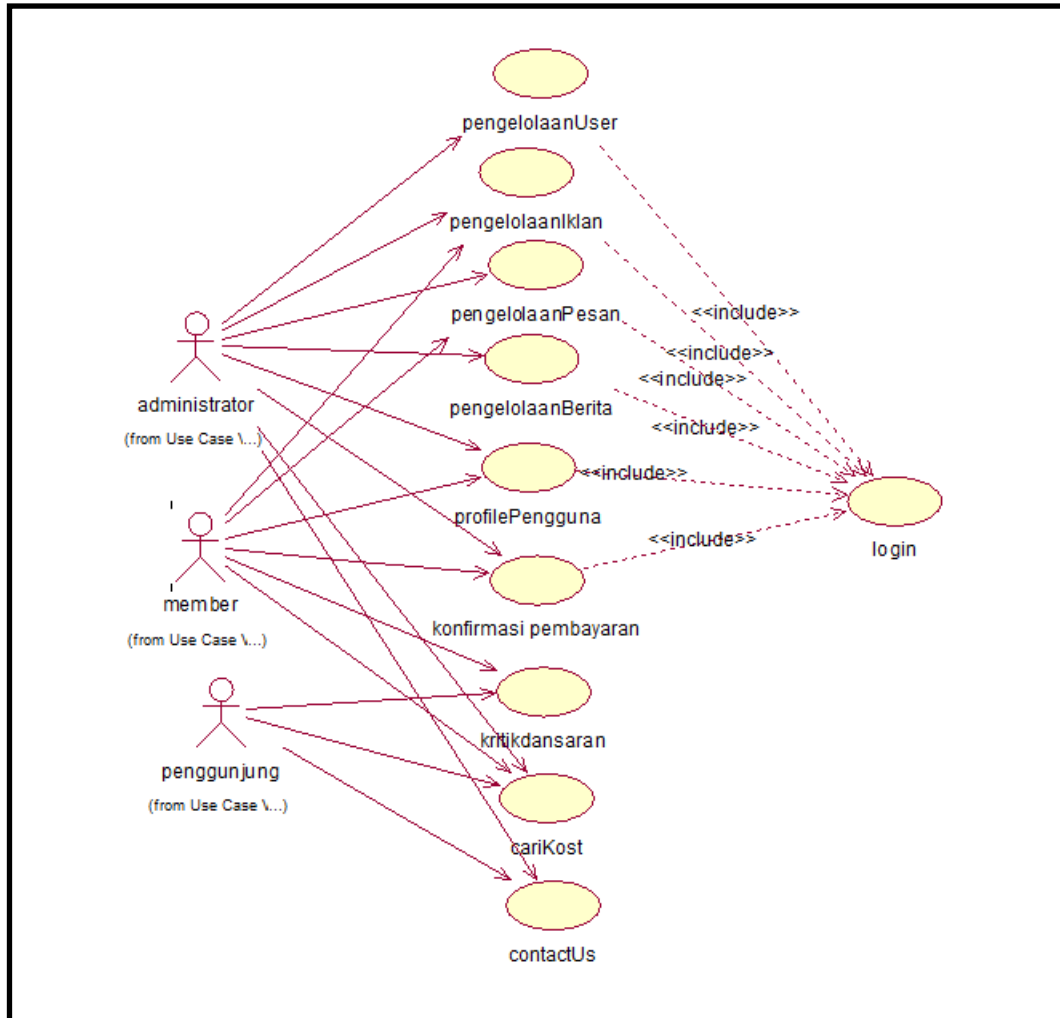
1. Nama : XAMPP For Windows  
Sumber : Apachefriends  
Sebagai *web service* yang digunakan untuk akses data pada jaringan intranet dan internet, yang didalamnya sudah tertanam *MySQL database management system (DBMS)* sebagai penyimpan data di sisi server.
2. Nama : Windows.  
Sumber : Microsoft.  
Sebagai sistem operasi.
3. *Web Browser* Sebagai antarmuka untuk mengakses sistem.

### **Antarmuka Komunikasi**

Antarmuka komunikasi perangkat lunak INKOST menggunakan *mysql connections*.

## Kebutuhan Fungsionalitas Perangkat Lunak

### Use Case Diagram



Gambar 2. Use Case Diagram

## Spesifikasi Rinci Kebutuhan

### Spesifikasi Kebutuhan Fungsionalitas

#### 4.1.1 Use Case Specification: Login

##### 1. Brief Description

*Use Case* ini digunakan oleh Aktor untuk memperoleh akses ke sistem. *Login* didasarkan pada sebuah *Username* dan *password* yang berupa rangkaian karakter

##### 2. Primary Aktor

1. *User*

##### 3. Supporting Aktor

*None*

##### 4. Basic Flow

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan *Login*
2. Sistem menampilkan antarmuka untuk *Login*
3. Aktor memasukkan *Email* dan *password*
4. Sistem memeriksa *Email* dan *password* yang diinputkan Aktor  
E-1 *Password* atau id *User* tidak sesuai
5. Sistem memberikan akses ke Aktor
6. *Use Case* ini selesai

##### 5. Alternative Flow

*None*

##### 6. Error Flow

E-1 *Password* atau nama *User* tidak sesuai

1. Sistem menampilkan peringatan bahwa *Email* atau *password* tidak sesuai
2. Kembali ke *Basic Flow* langkah ke-3

##### 7. Pre Conditions

*None*

#### 8. *Post Conditions*

1. Aktor masuk ke sistem dan dapat menggunakan fungsi-fungsi pada sistem

### **4.1.2 Use Case Specification: Reset Password**

#### 1. *Brief Description*

*Use Case* ini digunakan oleh Aktor untuk mereset password. *Reset password* yang dilakukan oleh user itu sendiri.

#### 2. *Primary Aktor*

1. *Administrator*
2. *Member*

#### 3. *Supporting Aktor*

*None*

#### 4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan *reset password*.
2. Sistem menampilkan antarmuka untuk *reset password*.
3. Aktor memasukkan *Email*.
4. Sistem memeriksa *Email* yang diinputkan Aktor.  
E-1 *Email* yang di inputkan tidak terdaftar di system.
5. Sistem memberikan informasi ke Aktor bahwa reset password berhasil.
6. *Use Case* ini selesai.

#### 5. *Alternative Flow*

*None*

#### 6. *Error Flow*

- E-1 *Email* yang di inputkan tidak terdaftar di sistem
- a. Sistem menampilkan peringatan bahwa *Email* tidak terdaftar
  - b. Kembali ke *Basic Flow* langkah ke-3

#### *Pre Conditions*

*None*

7. *Post Conditions*

1. Aktor berhasil mereset password

#### **4.1.3 Use Case Specification: Registrasi**

1. *Brief Description*

*Use Case* ini digunakan oleh Aktor untuk registrasi. data yang dibutuhkan saat registrasi yaitu nama, email, password, gambar untuk profile, dan tanggal lahir.

2. *Primary Aktor*

1. *member*

3. *Supporting Aktor*

*None*

4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan *registrasi*.
2. Sistem menampilkan antarmuka untuk *registrasi*
3. Aktor memasukkan nama, email, password, gambar profile dan tanggal lahir
4. Sistem memeriksa nama, email, password, gambar profile dan tanggal lahir yang diinputkan Aktor.  
E-1 Nama yang di inputkan sudah terdaftar di system  
E-2 Email yang di inputkan sudah terdaftar di sistem  
E-3 password dan konfirmasi password yang di inputkan tidak sama.
5. Sistem memberikan informasi ke Aktor bahwa registrasi berhasil
6. *Use Case* ini selesai

5. *Alternative Flow*

*None*

6. *Error Flow*

- E-1 Nama yang di inputkan sudah terdaftar di system



1. Sistem menampilkan peringatan bahwa *nama* sudah terdaftar.
  2. Kembali ke *Basic Flow* langkah ke-3.
- E-2 Email yang di inputkan tidak terdaftar di sistem
1. Sistem menampilkan peringatan bahwa *Email* sudah terdaftar.
  2. Kembali ke *Basic Flow* langkah ke-3.
- E-3 password dan konfirmasi password yang di inputkan tidak sama.
1. Sistem menampilkan peringatan bahwa *password dan konfirmasi password* tidak sama.
  2. Kembali ke *Basic Flow* langkah ke-3.

#### 7. *Pre Conditions*

*None*

#### 8. *Post Conditions*

1. Aktor berhasil registrasi

### 1.1.4 **Use Case Specification: Pengelolaan User**

#### 1. Brief Description

*Use Case* ini digunakan oleh Aktor untuk mengelola user. Aktor dapat melakukan *view user*, *remove user*.

#### 2. Primary Aktor

1. Administrator

#### 3. *Supporting* Aktor

*None*

#### 4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan user.
2. Sistem menampilkan semua data user, dan memberikan pilihan untuk melakukan *view user*, dan *delete user*.

3. Aktor memilih untuk melakukan view user user.
  - A-2 Aktor memilih untuk melakukan delete user.
4. sistem menampilkan data user yang dipilih oleh aktor.
5. *Use Case* selesai
5. *Alternative Flow*
  - A-2 Aktor memilih untuk melakukan delete user
    1. Sistem menampilkan data user
    2. Actor memilih user yang akan didelete
    3. Sistem mengkonfirmasi apakah pelanggan akan didelete atau tidak.
    4. kembali langkah ke-8
6. *Error Flow*
  1. none
2. *Pre Conditions*
  1. *Use Case Login* telah dilakukan
  2. Aktor telah masuk ke sistem
3. *Post Conditions*
  1. Data User di *database* telah ter-update

### 1.1.5 ***Use Case Spesification: Pengelolaan Iklan***

1. Brief Description
 

*Use Case* ini digunakan oleh Aktor untuk mengelola iklan. Aktor dapat melakukan tambah data iklan, edit data iklan, lihat iklan, delete data iklan.
2. Primary Aktor
  1. Administrator.
  2. Member.
3. *Supporting* Aktor
 

None
4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan iklan
  2. Sistem menampilkan semua menu iklan, dan memberikan pilihan untuk melakukan tambah data iklan, edit data iklan, lihat iklan dan delete iklan.
  3. Aktor memilih untuk melakukan tambah iklan.
    - A-1 Aktor memilih untuk melakukan edit data iklan.
    - A-2 Aktor memilih delete data iklan.
    - A-3 Aktor memilih view data iklan.
  4. Aktor memilih untuk buat iklan.
  5. Aktor meminta sistem untuk menyimpan data iklan telah diinputkan.
  6. Sistem mengecek data iklan yang telah diinputkan.
    - E-1 Data iklan yang diinputkan Aktor belum lengkap
  7. Sistem menyimpan data iklan ke *database*
  8. *Use Case* selesai
5. *Alternative Flow*
- A-1 Aktor memilih untuk melakukan edit iklan.
    1. Sistem menampilkan data atau profile iklan
      - A-3 Aktor memilih untuk melakukan *delete iklan*
    2. Aktor mengedit data iklan yang sudah ditampilkan
    3. Aktor meminta sistem untuk menyimpan data iklan yang telah diedit
    4. Sistem melakukan pengecekan terhadap data iklan yang telah diedit
      - E-2 Data iklan yang telah diedit tidak lengkap
    5. Sistem menyimpan data iklan yang telah diedit ke *database*
    6. Berlanjut ke *Basic Flow* langkah ke-8
  - A-2 Aktor memilih delete data iklan
    1. Sistem menampilkan data atau profile iklan
    2. Aktor memilih iklan yang akan dihapus
    3. Sistem mengkonfirmasi apakah yakin akan menghapus iklan atau tidak

4. kembali *Alternative Flow* A-1 langkah ke-8
- A-3 Aktor memilih delete data iklan
1. Sistem menampilkan data atau profile iklan
  2. Aktor memilih iklan yang akan ditampilkan
  3. Sistem menampilkan data iklan yang dipilih
  4. kembali *Alternative Flow* A-1 langkah ke-8

## 2. *Error Flow*

E-1 Data iklan yang diinputkan Aktor tidak lengkap

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan tidak lengkap
2. Kembali ke *Basic Flow* Langkah ke-5

### 1. *Pre Conditions*

1. *Use Case Login* telah dilakukan
2. Aktor telah masuk ke sistem

### 2. *Post Conditions*

1. Data iklan di *database* telah ter-update

## ***Use Case Specification: Pengelolaan pesan***

### 1. Brief Description

*Use Case* ini digunakan oleh Aktor untuk pesan. Aktor dapat melakukan tambah pesan, edit data pesan, dan delete pesan.

### 2. Primary Aktor

1. Administrator
2. Member

### 3. Supporting Aktor

1. Operator

### 4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan pesan

2. Sistem menampilkan semua data pesan, dan memberikan pilihan untuk melakukan tambah pesan, edit data pesan, dan delete data pesan.
  3. Aktor memilih untuk melakukan tambah data pesan
    - A-1 Aktor memilih untuk melakukan edit data pesan
    - A-2 Aktor memilih untuk menghapus data pesan
  4. Aktor menginputkan data pesan
  5. Aktor meminta sistem untuk menyimpan data pesan yang telah diinputkan
  6. Sistem mengecek data pesan yang telah diinputkan.
    - E-1 Data pesan yang diinputkan Aktor belum lengkap
  7. Sistem menyimpan data sopir ke *database*
  8. *Use Case* selesai
5. *Alternative Flow*
- A-1 Aktor memilih untuk melakukan edit data pesan
1. Sistem menampilkan data pesan
  2. Aktor mengedit data pesan yang sudah ditampilkan
  3. Aktor meminta sistem untuk menyimpan data pesan yang telah diedit
  4. Sistem melakukan pengecekan terhadap data pesan yang telah diedit
    - E-2 Data pesan yang telah diedit tidak lengkap
  5. Sistem menyimpan data pesan yang telah diedit ke *database*
  6. Berlanjut ke *Basic Flow* langkah ke-8
- A-2 menghapus data pesan
1. Aktor memilih data pesan yang akan dihapus
  2. Sistem menampilkan data pesan dan konfirmasi apakah akan menghapus pesan atau tidak
  3. kembali ke *Basic Flow* langkah ke-8
6. *Error Flow*
- E-1 Data pesan yang diinputkan Aktor tidak lengkap
1. Sistem memberikan pesan peringatan bahwa data yang diinputkan tidak lengkap

- 2. Kembali ke *Basic Flow* Langkah ke-7
- E-2 Data pesan yang diinputkan Aktor tidak lengkap
  - 2. Sistem memberikan pesan peringatan bahwa data yang diinputkan tidak lengkap
  - 3. Kembali ke *Alternative Flow A-1* Langkah ke-5
- 3. *Pre Conditions*
  - 1. *Use Case Login* telah dilakukan
  - 2. Aktor telah masuk ke sistem
- 4. *Post Conditions*
  - 1. Data pesan di *database* telah *ter-update*

**Use Case Spesification: Pengelolaan Berita Pemberitahuan**

- 1. Brief Description
 

*Use Case* ini digunakan oleh Aktor untuk berita. Aktor dapat melakukan tambah berita, edit data berita, dan hapus data berita.
- 2. Primary Aktor
  - 1. Administrator
- 3. *Supporting* Aktor
 

None
- 4. *Basic Flow*
  - 1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan berita
  - 2. Sistem menampilkan semua data berita dan memberikan pilihan untuk melakukan tambah berita, edit berita, dan hapus berita.
  - 3. Aktor memilih untuk melakukan tambah data berita
    - A-1 Aktor memilih untuk melakukan edit data berita
    - A-2 Aktor memilih untuk melakukan hapus berita
  - 4. Aktor menginputkan data berita

5. Aktor meminta sistem untuk menyimpan data berita yang telah diinputkan
  6. Sistem mengecek data berita yang telah diinputkan.
    - E-1 Data berita yang diinputkan Aktor belum lengkap
  7. Sistem menyimpan data berita ke *database*
  8. *Use Case* selesai
5. *Alternative Flow*
- A-1 Aktor memilih untuk melakukan edit berita
1. Sistem menampilkan data berita
  2. Aktor mengedit data berita yang sudah ditampilkan
  3. Aktor meminta sistem untuk menyimpan data berita yang telah diedit
  4. Sistem melakukan pengecekan terhadap data berita yang telah diedit
    - E-2 Data berita yang telah diedit tidak lengkap
  5. Sistem menyimpan data berita yang telah diedit ke *database*
  6. Berlanjut ke *Basic Flow* langkah ke-8
- A-2 Aktor memilih untuk melakukan hapus data berita
1. Sistem mengkonfirmasi hapus berita
  2. Data berita di *database* telah terhapus
  3. Berlanjut ke *Basic Flow* langkah ke-8
1. *Error Flow*
- E-1 Data berita yang diinputkan Aktor tidak lengkap
1. Sistem memberikan pesan peringatan bahwa data yang diinputkan tidak lengkap
  2. Kembali ke *Basic Flow* Langkah ke-7
1. *Pre Conditions*
1. *Use Case Login* telah dilakukan
  2. Aktor telah masuk ke sistem
2. *Post Conditions*
1. Data berita di *database* telah ter-update

## **Use Case Specification: Pengelolaan Profil**

### 1. Brief Description

*Use Case* ini digunakan mengubah data profil pribadi pengguna. Aktor dapat melakukan ubah data pengguna, ubah foto pengguna, dan ubah *password* pengguna.

### 2. Primary Aktor

1. Admin
2. member

### 3. Supporting Aktor

*None*

### 4. Basic Flow

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan data profil
2. Sistem menampilkan data profil beserta semua pilihan menu profil
3. Aktor memilih untuk melakukan ubah data pengguna
  - A-1 Aktor memilih untuk ubah foto pengguna
  - A-2 Aktor memilih untuk ubah *data* pengguna
  - A-3 Aktor memilih untuk ubah *password* pengguna
4. Aktor mengubah data pengguna
5. Aktor meminta sistem untuk menyimpan data pengguna yang telah diinputkan
6. Sistem mengecek data pengguna yang telah diinputkan.
  - E-1 Data pengguna yang diinputkan Aktor belum lengkap
7. Sistem menyimpan data pengguna ke *database*
8. *Use Case* selesai

### 5. Alternative Flow

- A-1 Aktor memilih untuk ubah foto pengguna
1. Sistem menampilkan foto pengguna
  2. Aktor memilih foto
  3. Aktor meminta sistem untuk mengupload foto baru
  4. Sistem menyimpan data foto pengguna ke *database*



5. Berlanjut ke *Basic Flow* langkah ke-8

A-2 Aktor memilih untuk ubah *data* pengguna

1. Sistem menampilkan form ubah *data*
2. Aktor menginputkan *data* baru
3. Aktor meminta sistem untuk menyimpan *data* yang telah diedit
4. Sistem melakukan pengecekan terhadap *data* yang telah diedit
5. Sistem menyimpan *data* yang telah diedit ke *database*
6. Berlanjut ke *Basic Flow* langkah ke-8

A-3 Aktor memilih untuk ubah *password* pengguna

1. Sistem menampilkan form ubah *password*
2. Aktor menginputkan *password* lama, *password* baru, dan *password* baru lagi
3. Aktor meminta sistem untuk menyimpan *data password* yang telah diedit
4. Sistem melakukan pengecekan terhadap *data password* yang telah diedit

E-4 *password* kosong

E-5 *password* lama salah

E-6 *password* baru dan konfirmasi *password* baru tidak sama

5. Sistem menyimpan *data password* yang telah diedit ke *database*

Berlanjut ke *Basic Flow* langkah ke-8

6. *Error Flow*

E-1 *Data* pengguna yang diinputkan Aktor tidak lengkap

1. Sistem memberikan pesan peringatan bahwa *data* yang diinputkan tidak lengkap
2. Kembali ke *Basic Flow* Langkah ke-8

E-3 *Password* sudah ada

1. Sistem memberikan pesan peringatan bahwa *username* sudah ada
2. Kembali ke *Alternative Flow* A-2 Langkah ke-6

E-4 *Password* kosong

1. Sistem memberikan pesan peringatan bahwa *password* tidak boleh kosong
2. Kembali ke *Alternative Flow* A-3 Langkah ke-6

E-5 *Password* lama salah

3. Sistem memberikan pesan peringatan bahwa *password* lama salah
4. Kembali ke *Alternative Flow* A-3 Langkah ke-6

E-6 *Password* baru dan konfirmasi *password* baru tidak sama

1. Sistem memberikan pesan peringatan bahwa *password* baru dan konfirmasi *password* baru tidak sama
2. Kembali ke *Alternative Flow* A-3 Langkah ke-6

7. *Pre Conditions*

1. *Use Case Login* telah dilakukan
2. Aktor telah masuk ke sistem

8. *Post Conditions*

Data profil pengguna di *database* telah *ter-update*

### ***Use Case Spesification: Pengelolaan Pembayaran***

1. *Brief Description*

*Use Case* ini digunakan oleh Aktor untuk mengelola Pembayaran. Aktor dapat mengirim pesan kepada admin Bahwa telah membayar

2. *Primary Aktor*

1. Member
2. admin

3. *Supporting Aktor*

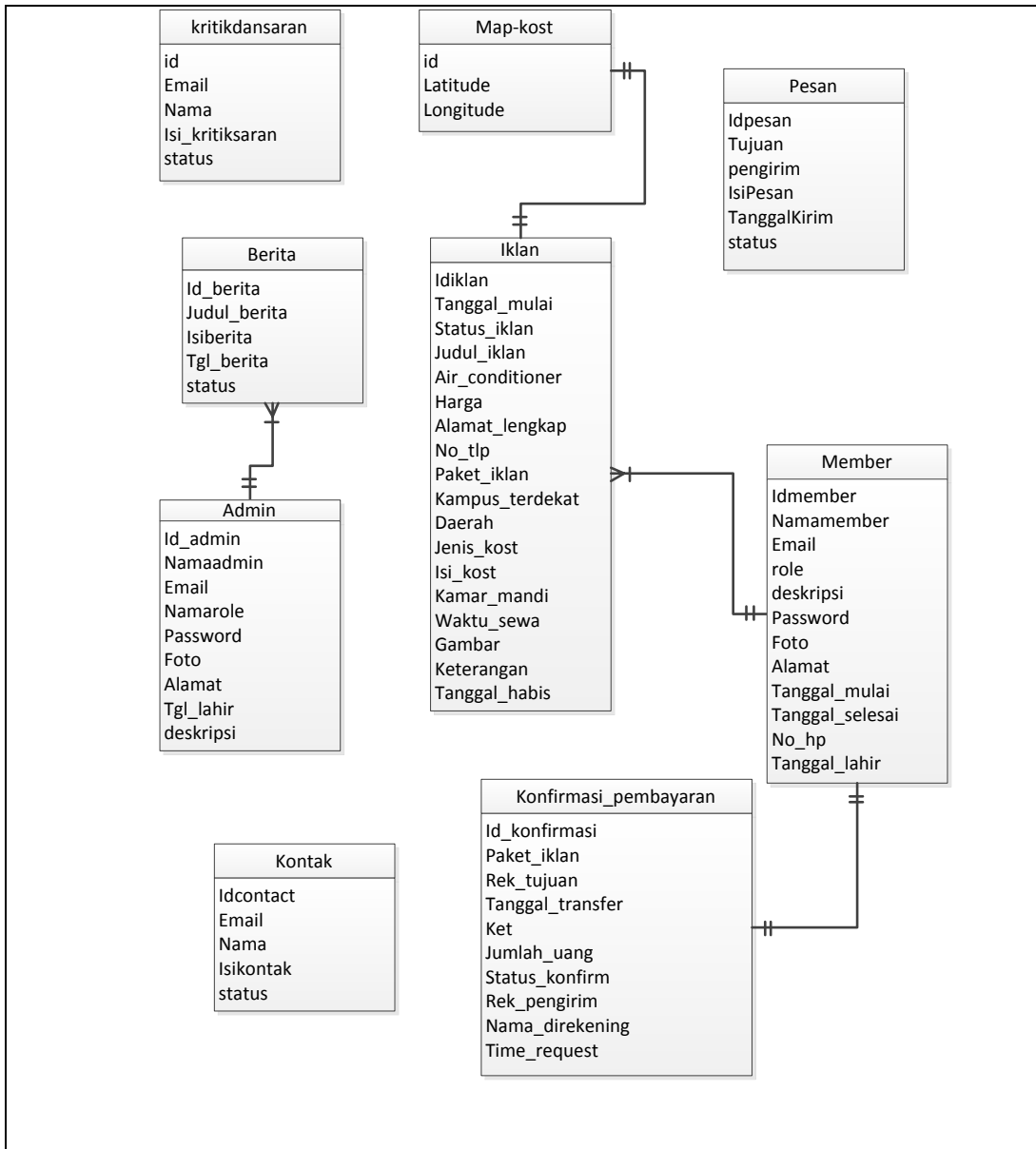
*None*

4. *Basic Flow*

1. *Use Case* ini dimulai ketika Aktor memilih untuk melakukan pengelolaan pembayaran
2. Sistem menampilkan semua data pembayaran,
3. Aktor memilih untuk melakukan tambah pembayaran

4. Aktor menginputkan data konfirmasi pembayaran.
5. Aktor meminta sistem untuk menyimpan data pembayaran yang telah diinputkan.
6. Sistem mengecek data pembayaran yang telah diinputkan.  
E-1 Data pembayaran yang diinputkan Aktor belum lengkap
7. Sistem menyimpan data pembayaran ke *database*
8. *Use Case* selesai
5. *Alternative Flow*  
None
6. *Error Flow*  
E-1 Data komentar yang diinputkan Aktor tidak lengkap
  1. Sistem memberikan pesan peringatan bahwa data yang diinputkan tidak lengkap.
  2. Kembali ke *Basic Flow* Langkah ke-8
7. *Pre Conditions*
  1. *Use Case Login* telah dilakukan
  2. Aktor telah masuk ke sistem
8. *Post Conditions*
  1. Data pembayaran di *database* telah ter-*update*

### Conceptual Data Model (CDM)



Gambar 3. Conceptual Data Model

# DPPL

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK


### INKOST (INFORMATION KOST)

Untuk:  
Tempat Beriklan Kost

Dipersiapkan oleh:  
Ferry Sukarto [06 07 05106]

Program Studi Teknik Informatika - Fakultas Teknologi  
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika  Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		<i>DPPL-INKOST</i>		
		Revisi		

## DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

**Daftar Halaman Perubahan**

<b>Halaman</b>	<b>Revisi</b>	<b>Halaman</b>	<b>Revisi</b>

## Daftar Isi

1	Pendahuluan.....	9
1.1	Tujuan .....	9
1.2	Ruang Lingkup .....	9
1.3	Definisi dan Akronim .....	10
1.4	Referensi .....	11
2	Rancangan Arsitektur.....	12
3	Sequence Diagram.....	14
3.1	User .....	14
3.1.1	Login User.....	14
3.1.2	Register User.....	14
3.1.3	Reset Password User.....	15
3.2	Profile .....	15
3.2.1	ubah foto user.....	15
3.2.2	Edit data user.....	16
3.2.3	Ganti Password User.....	16
3.3	Pengelolaan user .....	17
3.3.1	Hapus User.....	17
3.3.2	Lihat User.....	17
3.4	Iklan .....	18
3.4.1	Tambah Iklan.....	18
3.4.2	Edit Iklan.....	19
3.4.3	Lihat Iklan.....	18
3.4.4	Hapus Iklan.....	19
3.5	Pesan .....	20
3.5.1	Tambah Pesan.....	20
3.5.2	Lihat Pesan.....	20
3.5.3	Balas Pesan.....	21
3.5.4	Delete Pesan.....	21
3.6	Berita Pemberitahuan .....	22
3.6.1	Tambah Berita Pemberitahuan.....	22
3.6.2	Edit Berita Pemberitahuan.....	22
3.6.3	Delete Berita Pemberitahuan.....	23
3.6.4	Lihat Berita Pemberitahuan.....	23
3.7	Konfirmasi Pembayaran .....	24
3.7.1	Member Konfirmasi.....	24
3.7.1	Admin Konfirmasi.....	24
3.8	Kritik dan Saran .....	25
3.8.1	Tambah Kritik dan Saran.....	25
3.8.2	Lihat Kritik dan Saran.....	25
3.8.3	Balas Kritik dan Saran.....	26
3.8.4	Hapus Kritik dan Saran.....	26
3.9	ContactUs .....	27
3.9.1	Tambah ContactUs.....	27
3.9.2	Lihat ContactUs.....	27
3.9.3	Balas ContactUs.....	28
3.9.4	Hapus ContactUs.....	28
3.10	Cari Kost.....	29
3.10.1	Cari Berdasarkan Daerah.....	29



3.10.2	Cari Berdasarkan Kampus Terdekat.....	29
3.10.3	Cari Berdasarkan Jenis Kost.....	30
3.10.4	Cari Berdasarkan Lama Sewa.....	30
3.10.3	Cari Berdasarkan Kamar Mandi.....	31
3.10.4	Cari Berdasarkan Pemakaian AC.....	31
3.10.4	Cari Berdasarkan Isi Kost.....	32
4	Class Diagram Specific Descriptions.....	33
4.1	Class Diagram .....	33
4.2	Deskripsi Kelas .....	35
4.2.1	Deskripsi Kelas <i>Entity</i> .....	35
4.2.1.1	<i>Admin</i> .....	35
4.2.1.2	<i>Member</i> .....	36
4.2.1.3	<i>Iklan</i> .....	37
4.2.1.4	<i>Pesan</i> .....	39
4.2.1.5	<i>Berita</i> .....	40
4.2.1.6	<i>Konfirmasi</i> .....	41
4.2.1.7	<i>Map</i> .....	42
4.2.1.8	<i>KritikSaran</i> .....	42
4.2.1.9	<i>Contactus</i> .....	43
4.2.2	Deskripsi Kelas <i>Control</i> .....	44
4.2.2.1	<i>User</i> .....	44
4.2.2.2	<i>c_iklan</i> .....	45
4.2.2.3	<i>c_pesanan</i> .....	45
4.2.2.4	<i>c_berita</i> .....	46
4.2.2.5	<i>c_konfirmasi</i> .....	47
4.2.2.6	<i>c_kritiksaran</i> .....	47
4.2.2.7	<i>c_contactus</i> .....	48
4.2.3	Deskripsi Kelas <i>Boundary</i> .....	48
4.2.3.1	<i>login</i> .....	48
4.2.3.2	<i>registrasi</i> .....	48
4.2.3.3	<i>resetpassword</i> .....	49
4.2.3.4	<i>ubahprofile</i> .....	49
4.2.3.5	<i>gantipassword</i> .....	49
4.2.3.6	<i>gantifoto</i> .....	49
4.2.3.7	<i>iklanui</i> .....	49
4.2.3.8	<i>beritauai</i> .....	50
4.2.3.9	<i>pesanui</i> .....	50
4.2.3.10	<i>konfirmasi_pembayaranui</i> .....	50
5	Perancangan Data.....	52
5.1	<i>Physical</i> Data Model .....	52
5.2	Dekomposisi Data .....	53
5.2.1	Deskripsi Entitas Data Pesan .....	53
5.2.2	Deskripsi Entitas Data Berita.....	53
5.2.3	Deskripsi Entitas Data <i>admin</i> .....	53
5.2.4	Deskripsi Entitas Data member.....	54
5.2.5	Deskripsi Entitas Data iklan.....	55
5.2.6	Deskripsi Entitas Data pembayaran.....	56
5.2.7	Deskripsi Entitas Data kritiksaran.....	56
5.2.8	Deskripsi Entitas Data contactus.....	57
5.3	Deskripsi Perancangan Antarmuka .....	58
5.3.1	halaman Luar.....	58
5.3.1.1	Halaman Pertama.....	58
5.3.1.2	Cari kost Page.....	59
5.3.1.3	Beriklan Page.....	60
5.3.1.4	ContactUs page.....	61
5.3.2	<i>User</i> .....	62

5.3.2.1	<i>Login</i> .....	62
5.3.2.2	Registrasi.....	63
5.3.2.3	Reset <i>Password</i> .....	64
5.3.2.4	edit profile.....	64
5.3.2.5	ganti <i>Password</i> .....	65
5.3.3	Pengelolaan User.....	66
5.3.3.1	Delte user.....	66
5.3.3.2	lihat user.....	67
5.3.4	Pengelolaan Iklan.....	67
5.3.4.1	Tambah iklan.....	68
5.3.4.2	Edit iklan.....	69
5.3.4.3	lihat iklan.....	70
5.3.5	Pengelolaan Pesan.....	71
5.3.5.1	Tambah Pesan.....	72
5.3.5.2	Lihat Pesan.....	73
5.3.5.3	Balas Pesan.....	74
5.3.6	Pengelolaan Berita Pemberitahuan.....	74
5.3.6.1	Tambah Berita Pemberitahuan.....	75
5.3.6.2	Hapus Berita Pemberitahuan.....	76
5.3.6.3	Edit Berita Pemberitahuan.....	77
5.3.7	Pengelolaan konfirmasi pembayaran.....	78
5.3.7.1	member konfirmasi.....	78
5.3.7.2	admin konfirmasi.....	79
5.3.8	Kritik dan saran.....	80
5.3.8.1	Tambah Kritik dan Saran.....	80
5.3.8.2	Lihat Kritik dan Saran.....	81
5.3.8.3	Balas Kritik dan Saran.....	82
5.3.8.4	Hapus Kritik dan Saran.....	83
5.3.9	ContactUS.....	84
5.3.9.1	Tambah ContactUs.....	84
5.3.9.2	Lihat ContactUs.....	85
5.3.9.3	Balas ContactUs.....	86
5.3.9.4	Hapus ContactUs.....	87

## Daftar Gambar

Gambar 2	Rancangan Arsitektur INKOST.....	12
Gambar 3.1	<i>Sequence</i> Diagram: <i>Login User</i> .....	14
Gambar 3.2	<i>Sequence</i> Diagram: Registrasi.....	14
Gambar 3.3	<i>Sequence</i> Diagram: Reset Password.....	15
Gambar 3.4	<i>Sequence</i> Diagram: ubah foto user.....	15
Gambar 3.5	<i>Sequence</i> Diagram: Ubah profil user.....	16
Gambar 3.6	<i>Sequence</i> Diagram: Ubah password User.....	16
Gambar 3.7	<i>Sequence</i> Diagram: Hapus User.....	17
Gambar 3.8	<i>Sequence</i> Diagram: Lihat User.....	17
Gambar 3.9	<i>Sequence</i> Diagram: <i>Tambah Iklan</i> .....	18
Gambar 3.10	<i>Sequence</i> Diagram: Edit Iklan.....	18
Gambar 3.11	<i>Sequence</i> Diagram: Lihat Iklan.....	19
Gambar 3.12	<i>Sequence</i> Diagram: <i>Delete Iklan</i> .....	19
Gambar 3.13	<i>Sequence</i> Diagram: Pesan.....	20
Gambar 3.14	<i>Sequence</i> Diagram: Lihat Pesan.....	20
Gambar 3.15	<i>Sequence</i> Diagram: Balas Pesan.....	21
Gambar 3.16	<i>Sequence</i> Diagram: Delete Pesan.....	21
Gambar 3.17	<i>Sequence</i> Diagram: Tambah Berita.....	22
Gambar 3.18	<i>Sequence</i> Diagram: Edit Berita.....	22
Gambar 3.19	<i>Sequence</i> Diagram: Hapus Berita.....	23
Gambar 3.20	<i>Sequence</i> Diagram: Lihat Berita.....	23
Gambar 3.21	<i>Sequence</i> Diagram: Member Konfirmasi.....	24
Gambar 3.22	<i>Sequence</i> Diagram: Admin Konfirmasi.....	24
Gambar 3.23	<i>Sequence</i> Diagram: Tambah Kritik saran.....	25
Gambar 3.24	<i>Sequence</i> Diagram: Lihat kritik saran.....	25
Gambar 3.25	<i>Sequence</i> Diagram: balas kritik saran.....	26
Gambar 3.26	<i>Sequence</i> Diagram: Delete Kritik saran.....	26
Gambar 3.27	<i>Sequence</i> Diagram: tambah contactus.....	27
Gambar 3.28	<i>Sequence</i> Diagram: lihat contactus.....	27
Gambar 3.29	<i>Sequence</i> Diagram: balas contactus.....	28
Gambar 3.30	<i>Sequence</i> Diagram: hapus contactus.....	28
Gambar 3.31	<i>Sequence</i> Diagram: cari berdasarkan daerah.....	29
Gambar 3.32	<i>Sequence</i> Diagram: cari berdasarkan kampus terdekat 29	
Gambar 3.33	<i>Sequence</i> Diagram: cari berdasarkan lama sewa.....	30
Gambar 3.34	<i>Sequence</i> Diagram: cari berdasarkan jenis kost....	30
Gambar 3.35	<i>Sequence</i> Diagram: cari berdasarkan kamar mandi...	31
Gambar 3.36	<i>Sequence</i> Diagram: cari berdasarkan pemakaian ac..	31
Gambar 3.37	<i>Sequence</i> Diagram: cari kost berdasarkan isi kost.	32
Gambar 4	Class Diagram.....	33
Gambar 5	<i>Physical</i> Data Model.....	52
Gambar 5.1	Rancangan Antarmuka Halaman Pertama.....	58
Gambar 5.2	Rancangan Antarmuka Cari Kost.....	59
Gambar 5.3	Rancangan Antarmuka Beriklan.....	60
Gambar 5.4	Rancangan Antarmuka ContactUs.....	61
Gambar 5.5	Rancangan Antarmuka <i>Login</i> .....	62
Gambar 5.6	Rancangan Antarmuka Registrasi.....	63
Gambar 5.7	Rancangan Antarmuka Reset Password.....	64
Gambar 5.8	Rancangan Antarmuka Ganti password.....	64
Gambar 5.9	Rancangan Antarmuka <i>Edit Profile</i> .....	65
Gambar 5.10	Rancangan Antarmuka User.....	66
Gambar 5.11	Rancangan Antarmuka User - Hapus User.....	66

Gambar 5.12	Rancangan Antarmuka User - Lihat User.....	67
Gambar 5.13	Rancangan Antarmuka Iklan.....	67
Gambar 5.14	Rancangan Antarmuka Iklan - Tambah iklan.....	68
Gambar 5.15	Rancangan Antarmuka Iklan - Edit iklan.....	69
Gambar 5.16	Rancangan Antarmuka Iklan - lihat iklan.....	70
Gambar 5.17	Rancangan Antarmuka pesan.....	71
Gambar 5.18	Rancangan Antarmuka pesan - Tambah pesan.....	72
Gambar 5.19	Rancangan Antarmuka pesan - Lihat pesan.....	73
Gambar 5.20	Rancangan Antarmuka pesan - Balas pesan.....	74
Gambar 5.21	Rancangan Antarmuka Berita pemberitahuan.....	74
Gambar 5.22	Rancangan Antarmuka Berita pemberitahuan - tambah berita pemberitahuan.....	75
Gambar 5.23	Rancangan Antarmuka Berita pemberitahuan - Hapus berita pemberitahuan.....	76
Gambar 5.24	Rancangan Antarmuka Berita pemberitahuan - Edit berita pemberitahuan.....	77
Gambar 5.25	Rancangan Antarmuka Konfirmasi Member.....	78
Gambar 5.26	Rancangan Antarmuka Konfirmasi Admin.....	79
Gambar 5.27	Rancangan Antarmuka Tambah Kritik dan Saran.....	80
Gambar 5.28	Rancangan Antarmuka Lihat Kritik dan Saran.....	81
Gambar 5.29	Rancangan Antarmuka Balas Kritik dan Saran.....	82
Gambar 5.30	Rancangan Antarmuka Hapus Kritik dan Saran.....	83
Gambar 5.31	Rancangan Antarmuka Tambah ContactUs.....	84
Gambar 5.32	Rancangan Antarmuka Lihat ContactUs.....	85
Gambar 5.33	Rancangan Antarmuka Balas ContactUs.....	86
Gambar 5.34	Rancangan Antarmuka Hapus ContactUs.....	87

## 1. Pendahuluan

### Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

### Ruang Lingkup

Perangkat Lunak INKOST dikembangkan dengan tujuan untuk:

19. Menangani pengelolaan data user.
20. Menangani pengelolaan data iklan.
21. Menangani pengelolaan data pesan.
22. Menangani pengelolaan data berita pemberitahuan.
23. Menangani pengelolaan profil pengguna.
24. Menangani pengelolaan konfirmasi pembayaran.
25. Menangani pencarian kost.
26. Menangani pengelolaan kritik dan saran.
27. Menangani pengelolaan contactus

### Definisi dan Akronim

Daftar definisi akronim dan singkatan:

<i>Keyword/Phrase</i>	<i>Definisi</i>
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga <i>Perangkat lunak Design Description</i> (SDD) merupakan deskripsi dari perancangan produk/perangkat

	lunak yang akan dikembangkan.
INKOST	Perangkat lunak Sistem informasi berbasis <i>web</i> untuk menangani iklan kost.
<i>Server</i>	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
<i>Web Server</i>	Perangkat lunak yang memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan <i>web browser</i> dan mengirimkan kembali hasilnya dalam bentuk halaman - halaman <i>web</i> yang umumnya berbentuk dokumen HTML.
Web Browser	Aplikasi perangkat lunak yang memungkinkan pengguna untuk menayangkan dan berinteraksi dengan tulisan, gambar, video, musik dan berbagai informasi lainnya yang terdapat pada halaman web di sebuah situs di World Wide Web atau di jaringan LAN lokal.

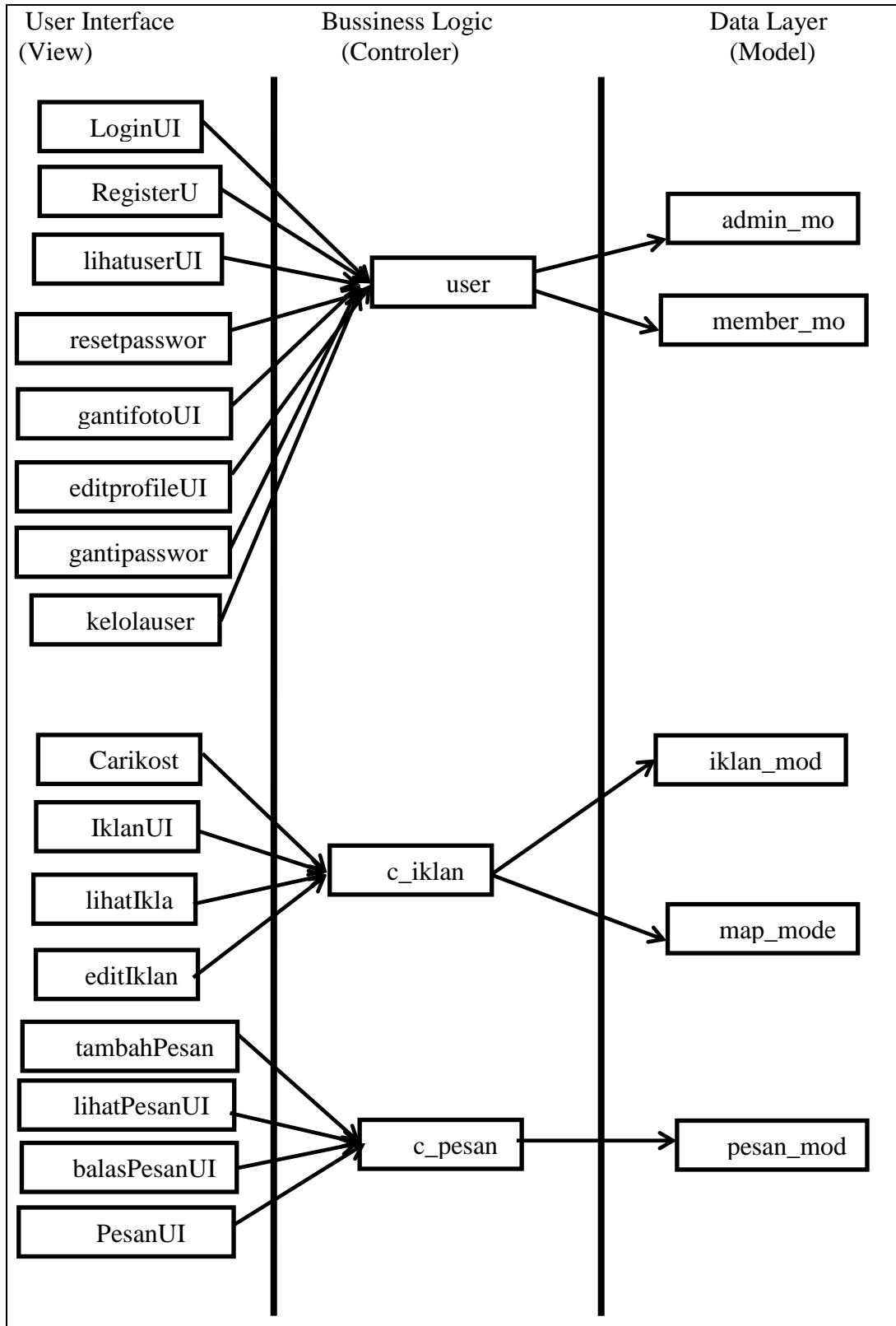
### Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

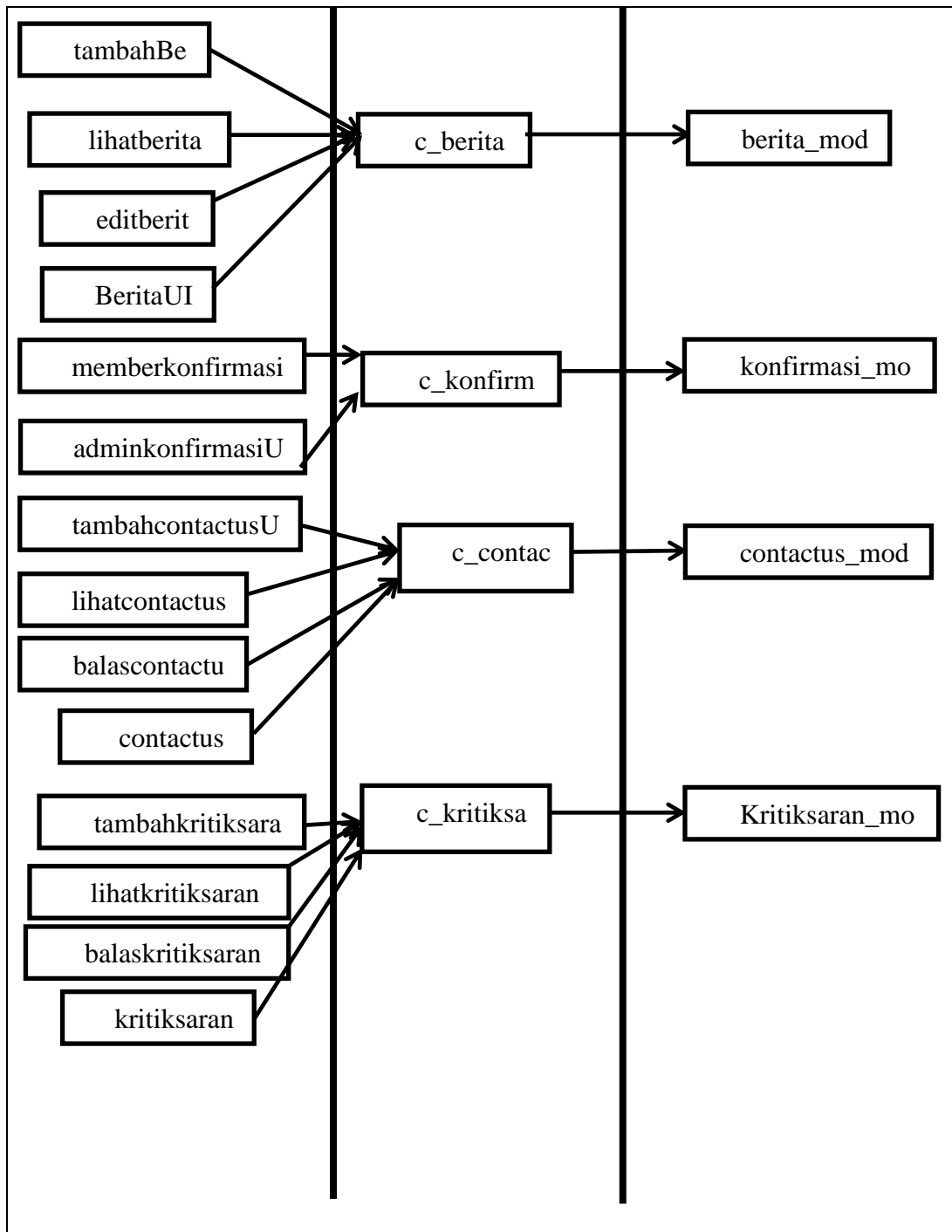
8. Hartanto, Steven., *Pembangunan Sistem Informasi Keluarga Berencana Indonesia (SIPKBI)*. Universitas Atma Jaya, Yogyakarta, 2010.

9. Whisnu, Andreas., *Pembangunan Sistem Informasi Rental Mobil Berbasis Web Dengan Framework Codeigniter (EXCARRENTIS)*. Universitas Atma Jaya, Yogyakarta, 2013.
10. Saputra, Agus., *Trik Kolaborasi Codeigniter dan jQuery*, Lokomedia, 2012.
11. Basuki, Awan Pribadi., *Membangun Web Berbasis PHP dengan Framework Codeigniter*, Locomedia, 2010.
12. Saputra, Agus., Feni Agustin., *Pemrograman CSS Untuk Pemula*, PT. Elex Media Komputindo, 2011.
13. Hakim, Lukmanul., *Trik Dasyat Menguasai Ajax dengan jQuery*, Lokomedia, 2011.
14. Werdana, S.Hut., M.Si., *Menjadi Master PHP dengan Framework Codeigniter*, PT. Elex Media Komputindo, 2010.

## 2. Rancangan Arsitektur





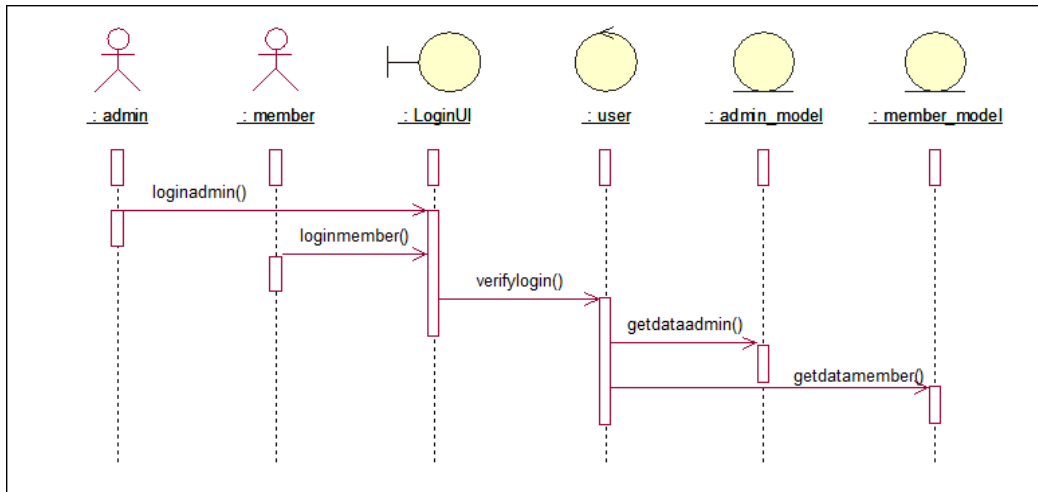


Gambar 2 Rancangan Arsitektur INKOST

### 3. Sequence Diagram

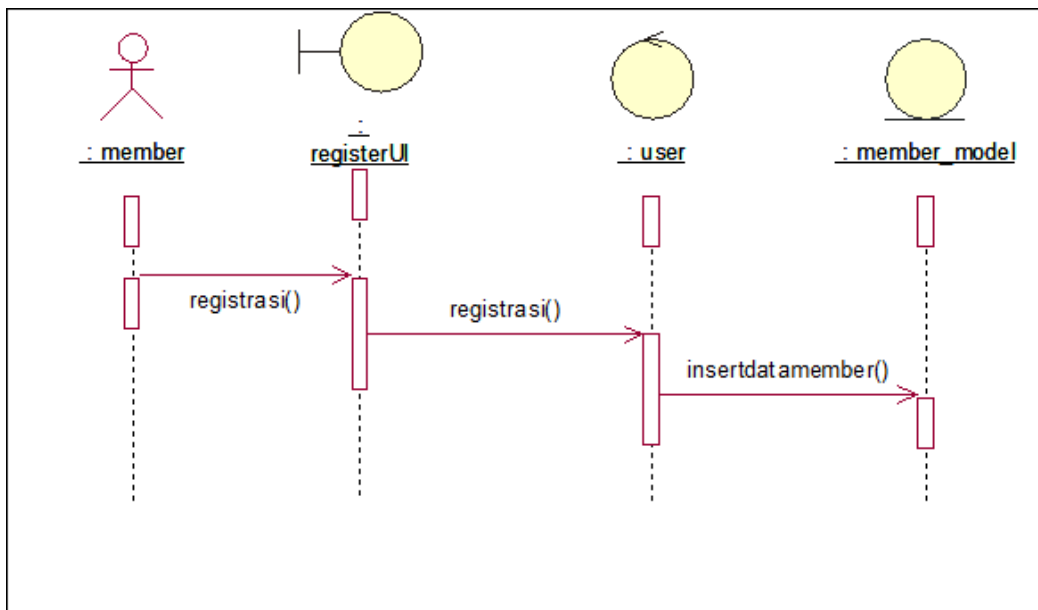
#### 3.1 User

##### 3.1.1 Login User



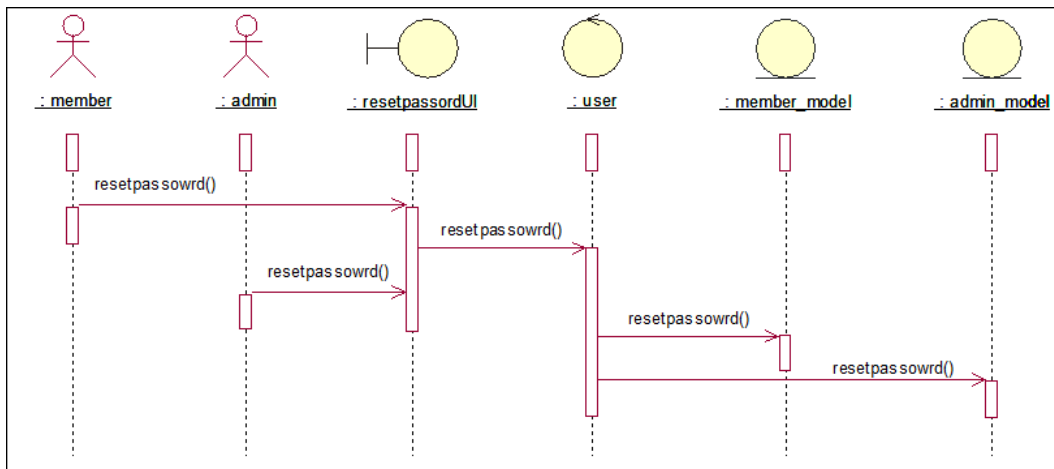
Gambar 3.1 Sequence Diagram: Login user

##### 3.1.2 Register



Gambar 3.2 Sequence Diagram: Register

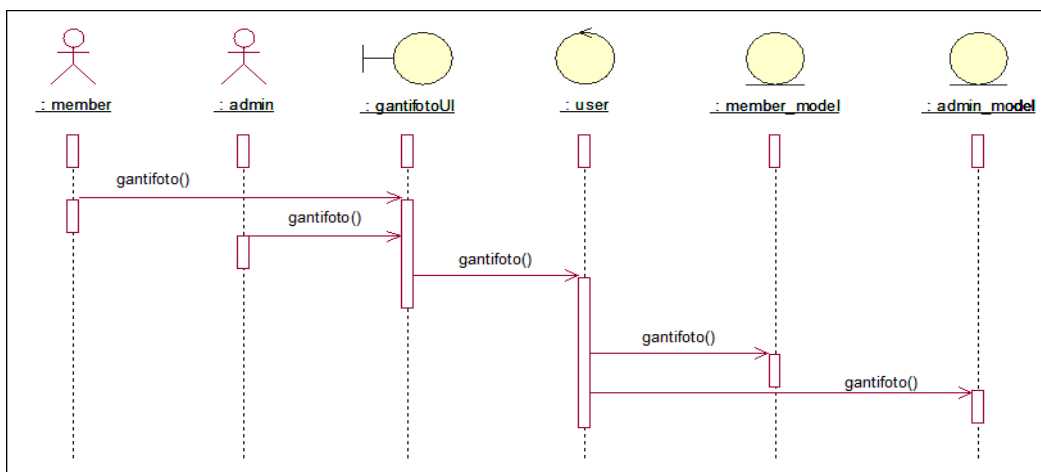
### 3.1.3 Reset Password



Gambar 3.3 *Sequence Diagram*: reset password

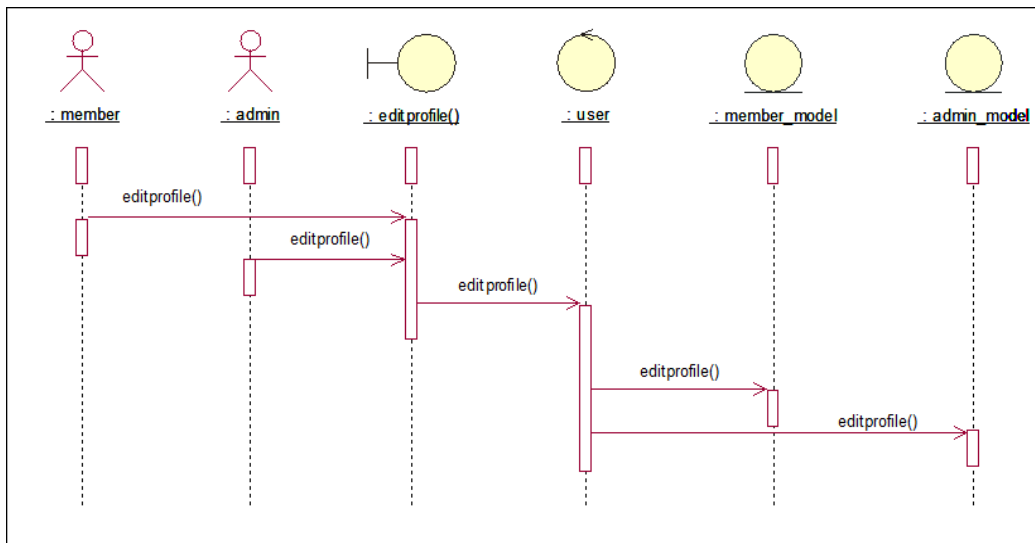
## 3.2 Profile

### 3.2.1 Ubah Foto User



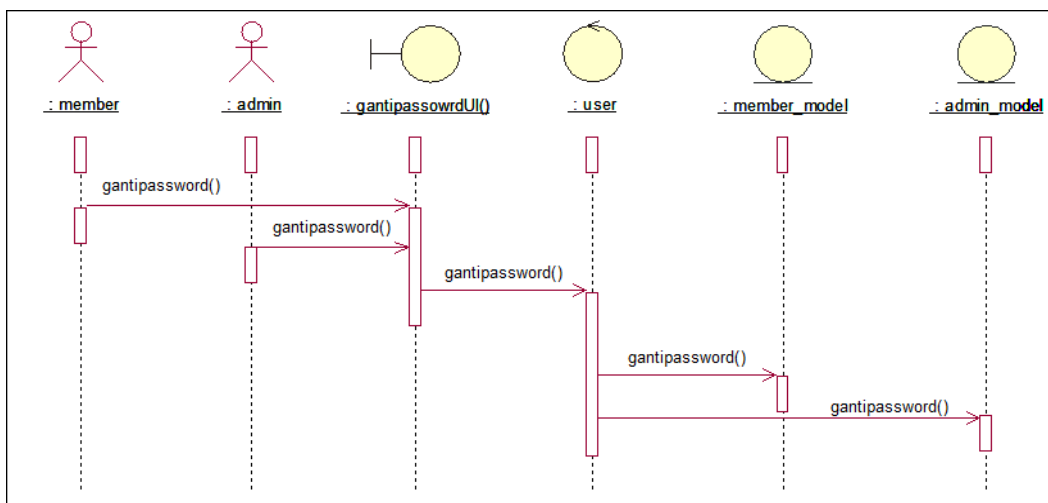
Gambar 3.4 *Sequence Diagram*: Ubah foto

### 3.2.2 Ubah Data Profil User



Gambar 3.5 *Sequence Diagram*: Ubah profil user

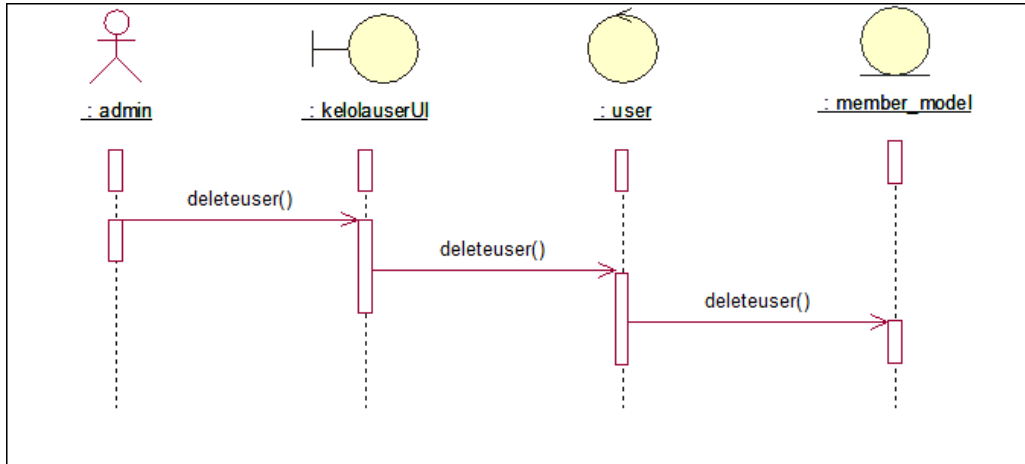
### 3.2.3 Ubah Password User



Gambar 3.6 *Sequence Diagram*: Ubah password user

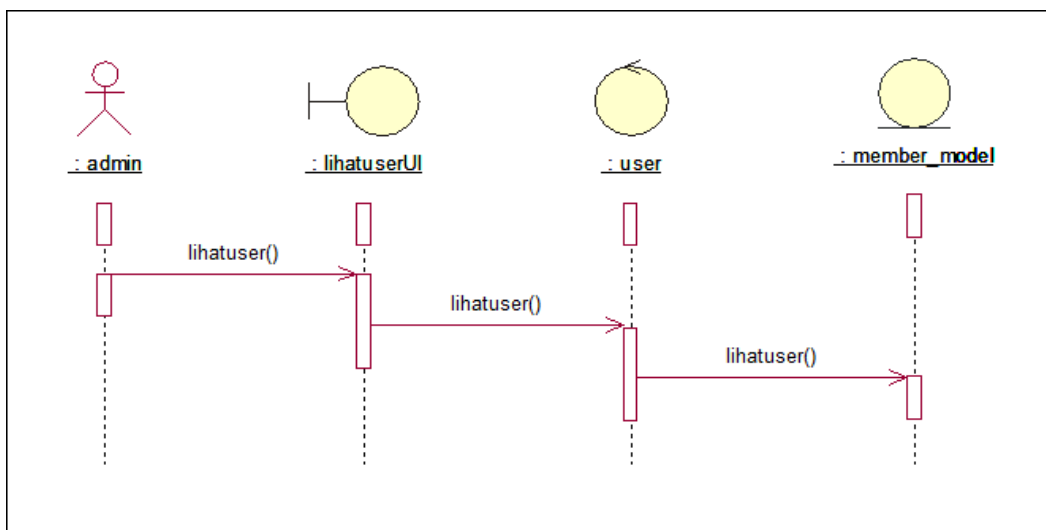
### 3.3 Pengelolaan User

#### 3.3.1 Hapus User



Gambar 3.7 *Sequence Diagram: hapus user*

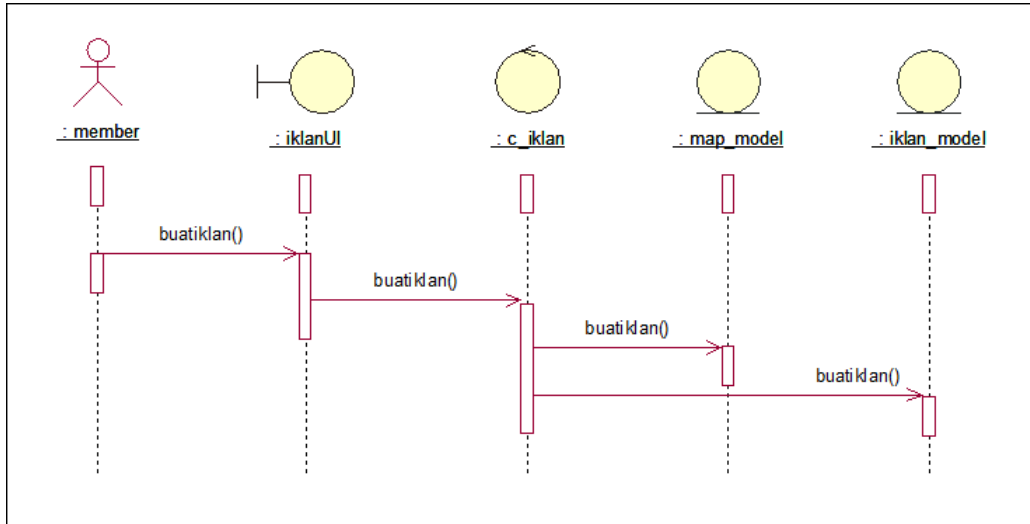
#### 3.3.2 Lihat User



Gambar 3.8 *Sequence Diagram: lihat user*

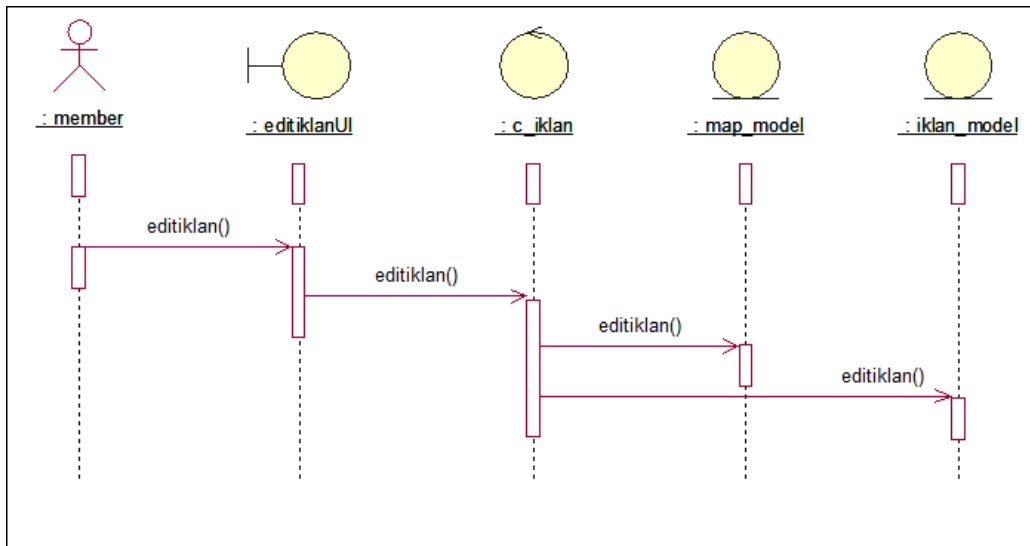
### 3.4 Iklan

#### 3.4.1 Tambah Iklan



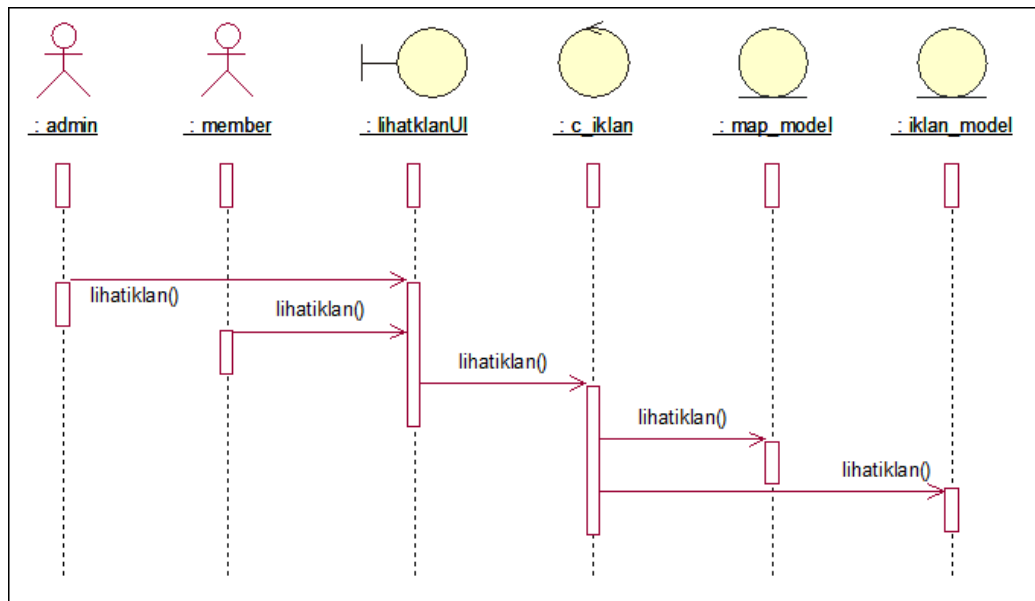
Gambar 3.9 *Sequence Diagram*: Tambah iklan

#### 3.4.2 Edit Iklan



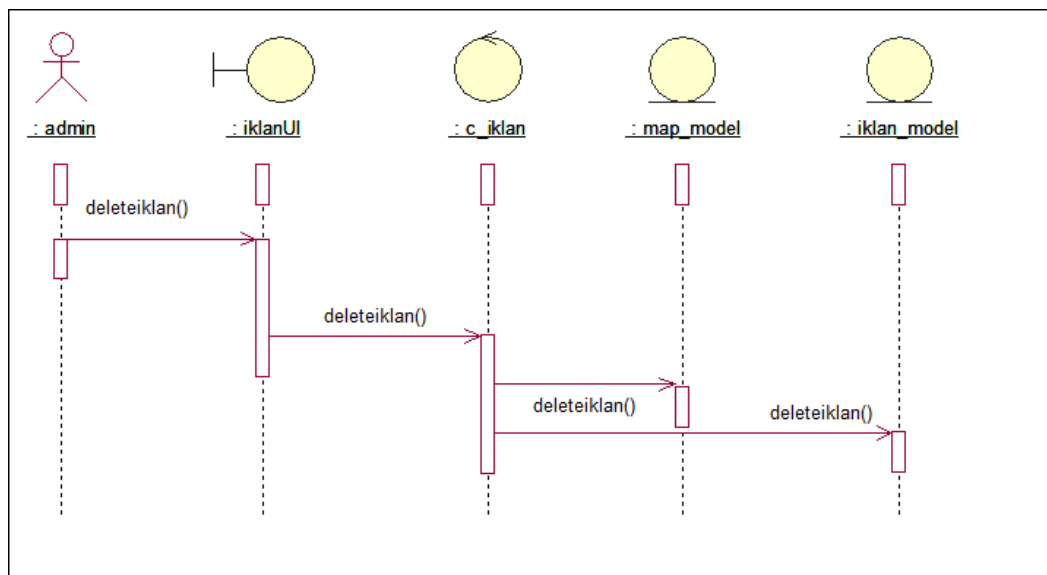
Gambar 3.10 *Sequence Diagram*: edit iklan

### 3.4.3 Lihat Iklan



Gambar 3.11 Sequence Diagram: *lihat iklan*

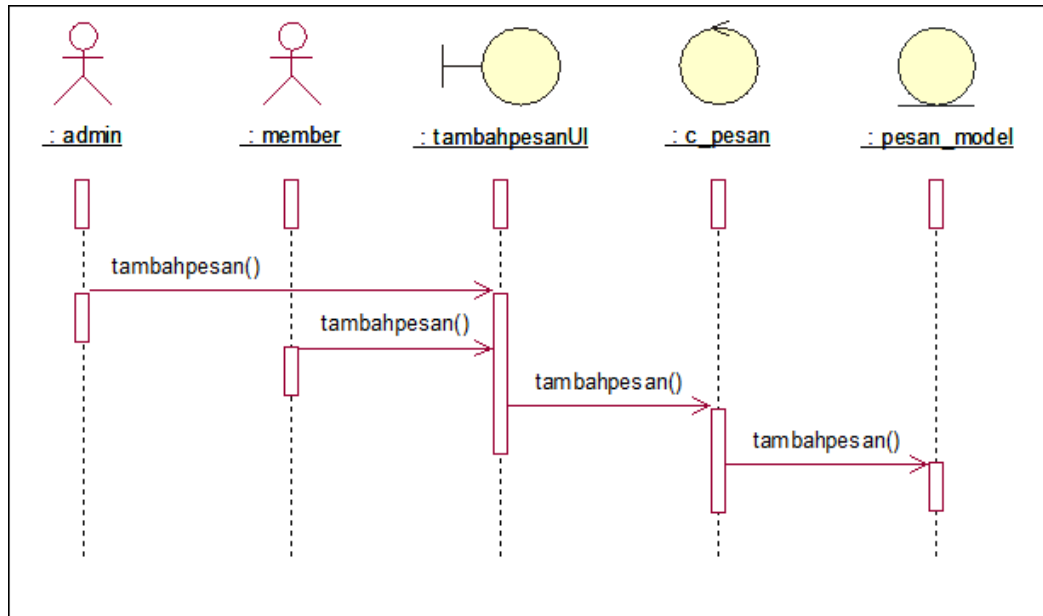
### 3.4.4 Delete Iklan



Gambar 3.12 Sequence Diagram: *delete iklan*

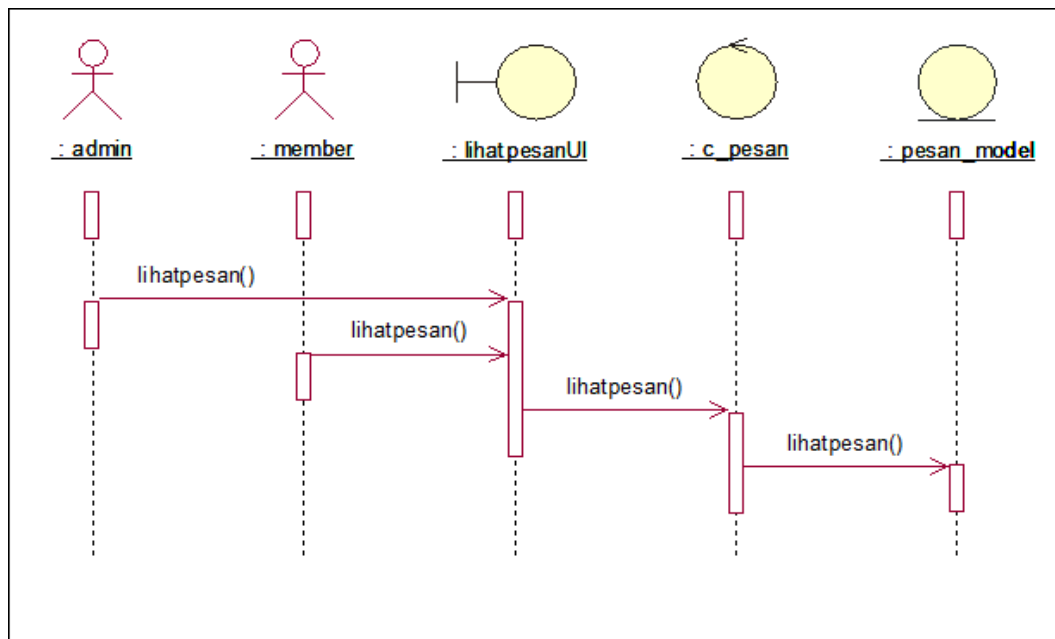
### 3.5 Pesan

#### 3.5.1 Tambah Pesan



Gambar 3.13 *Sequence Diagram*: Tambah pesan

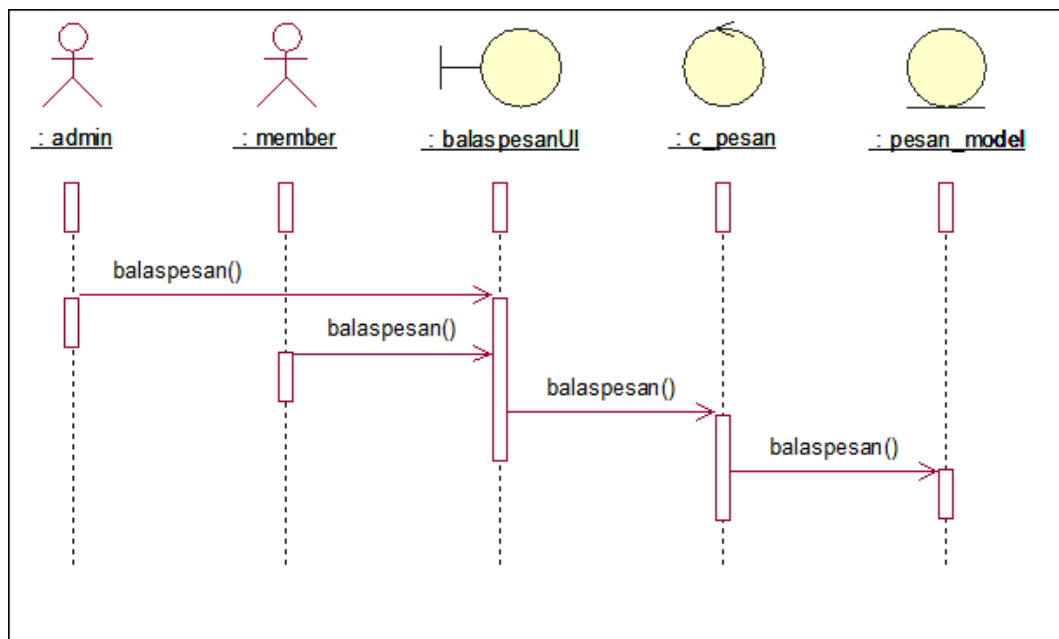
#### 3.5.2 Lihat Pesan



Gambar 3.14 *Sequence Diagram*: lihat pesan

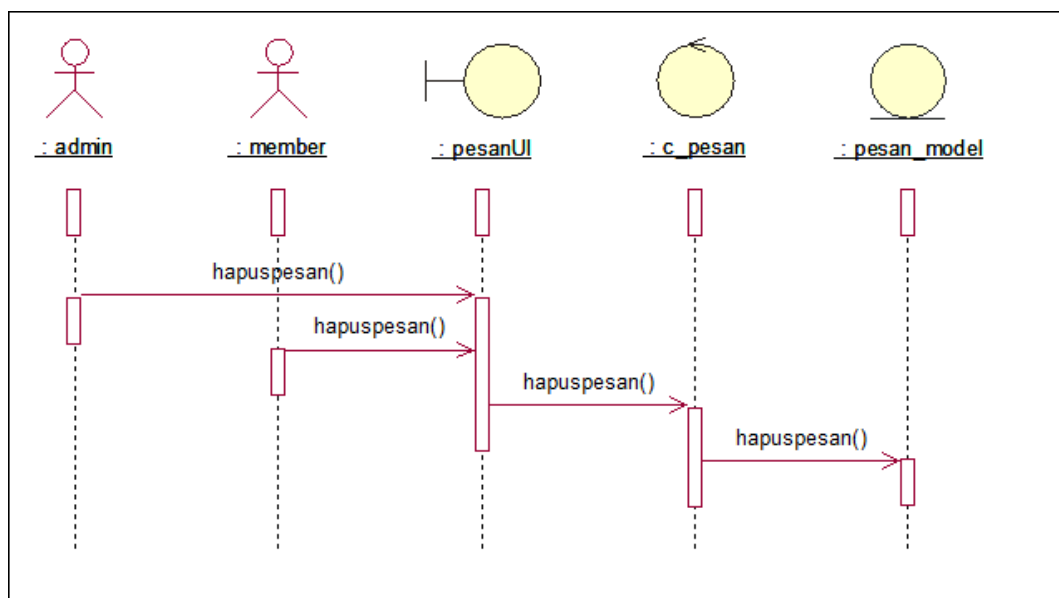


### 3.5.3 Balas Pesan



Gambar 3.15 *Sequence Diagram*: balas pesan

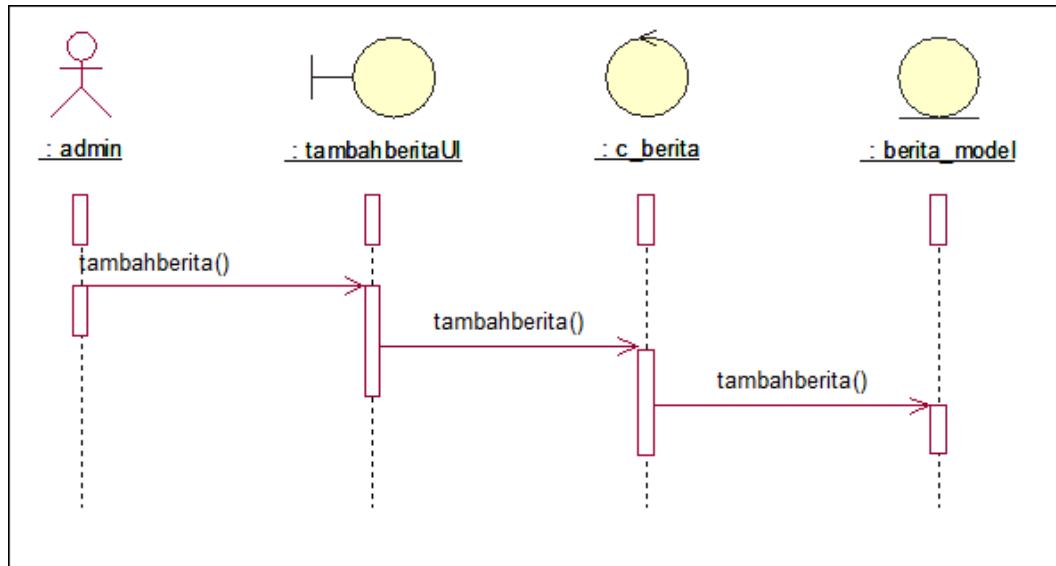
### 3.5.4 Delete Pesan



Gambar 3.16 *Sequence Diagram*: delete pesan

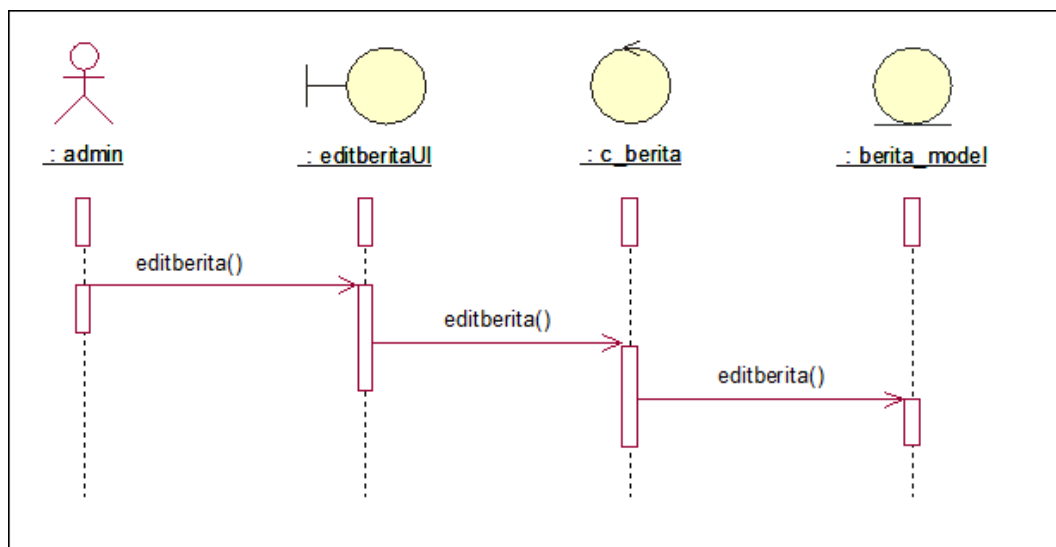
### 3.6 Pengelolaan Berita pemberitahuan

#### 3.6.1 Tambah Berita



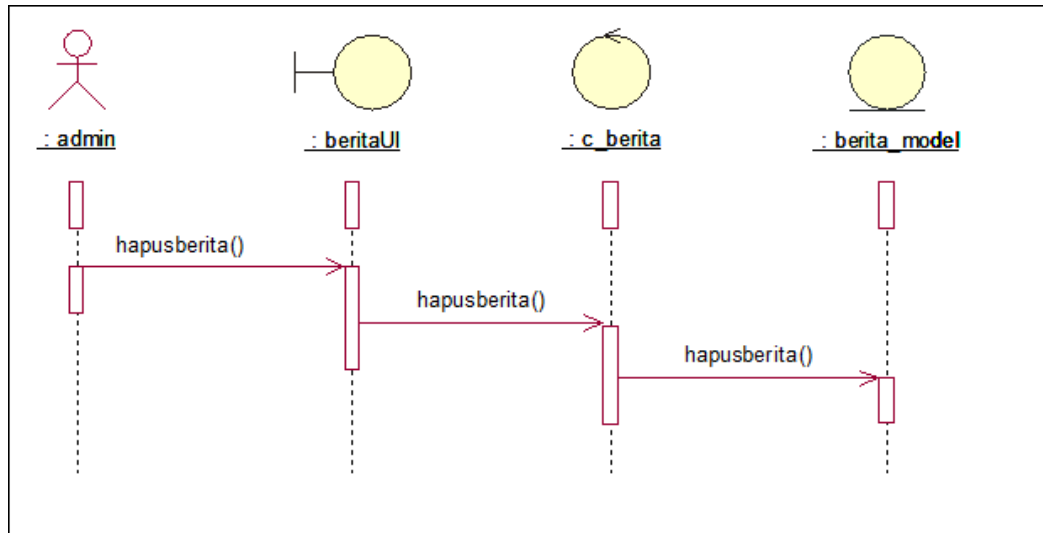
Gambar 3.17 *Sequence Diagram*: tambah berita

#### 3.6.2 Edit Berita



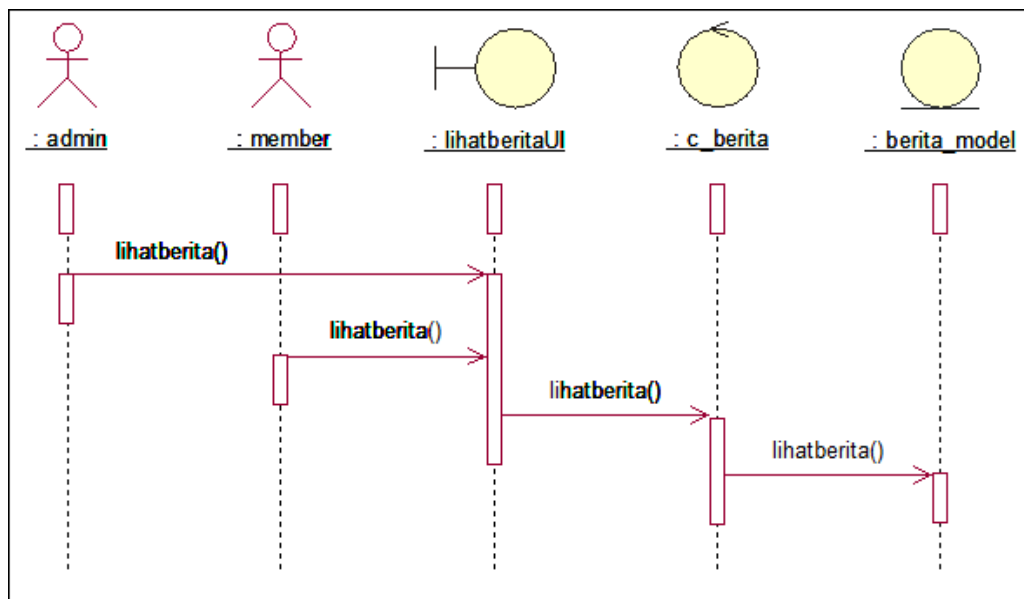
Gambar 3.18 *Sequence Diagram*: Edit berita

### 3.6.3 Hapus Berita



Gambar 3.19 *Sequence Diagram*: Hapus berita

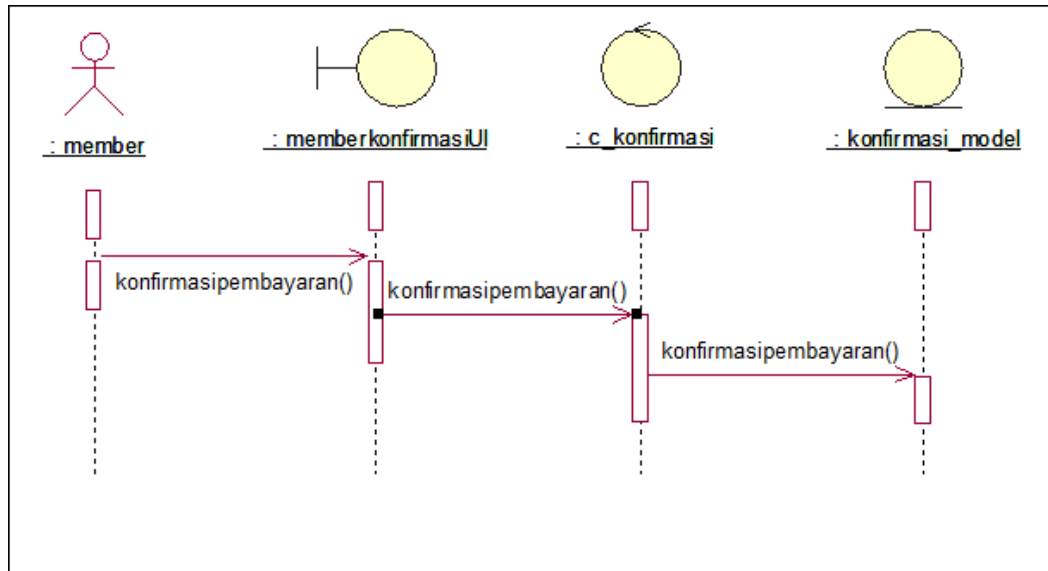
### 3.6.4 Lihat Berita



Gambar 3.20 *Sequence Diagram*: lihat berita

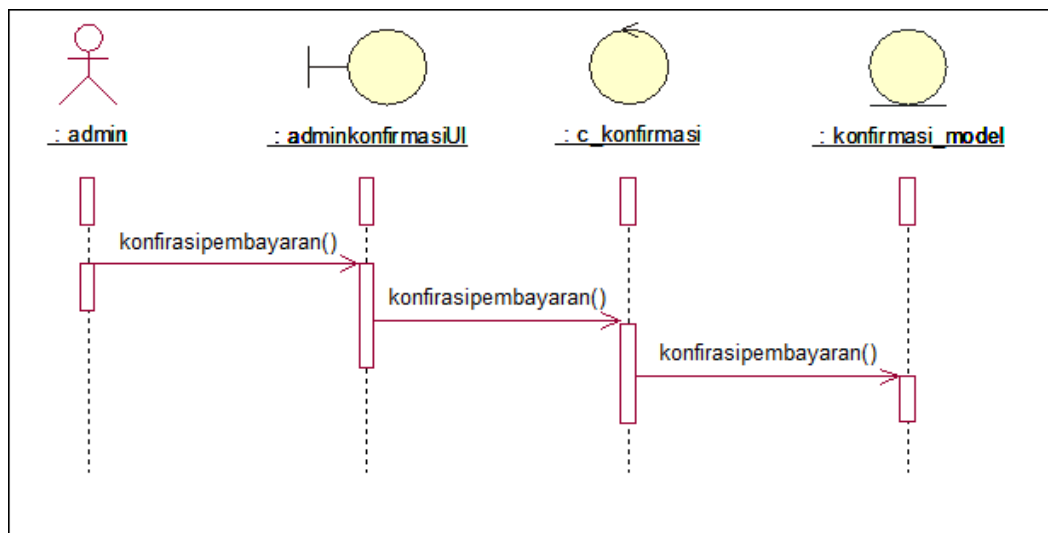
### 3.7 konfirmasi pembayaran

#### 3.7.1 Konfirmasi Pembayaran Dari Member



Gambar 3.21 *Sequence Diagram*: Konfirmasi pembayaran

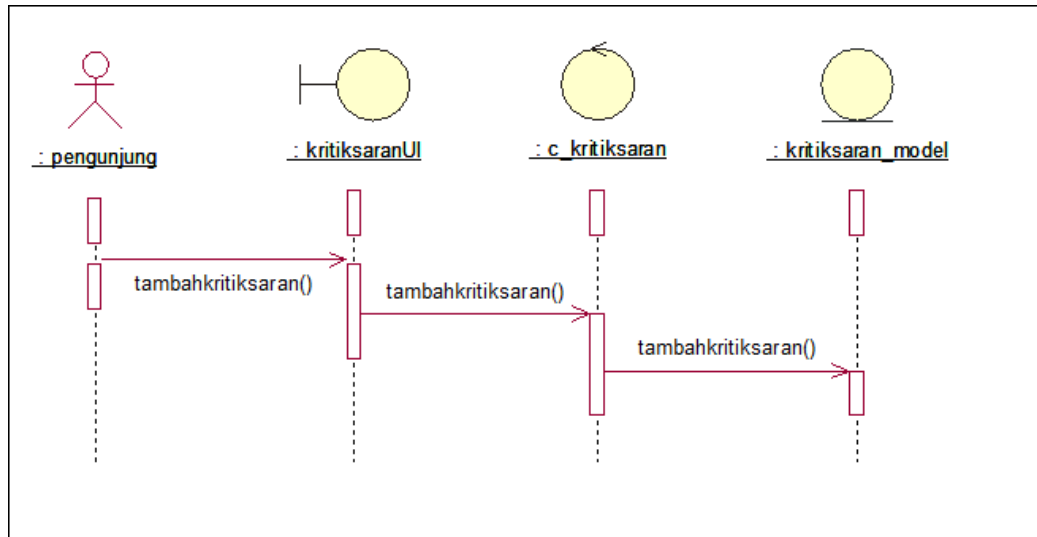
#### 3.7.2 Konfirm Pembayaran Dari Admin



Gambar 3.22 *Sequence Diagram*: admin konfirm

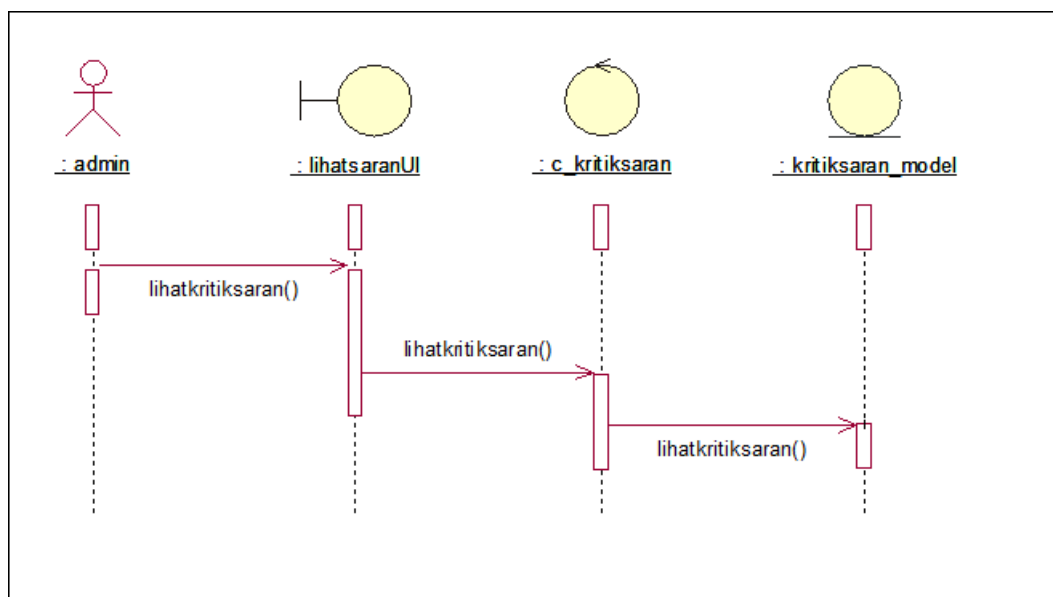
### 3.8 Kritik dan Saran

#### 3.8.1 Tambah Kritiksaran



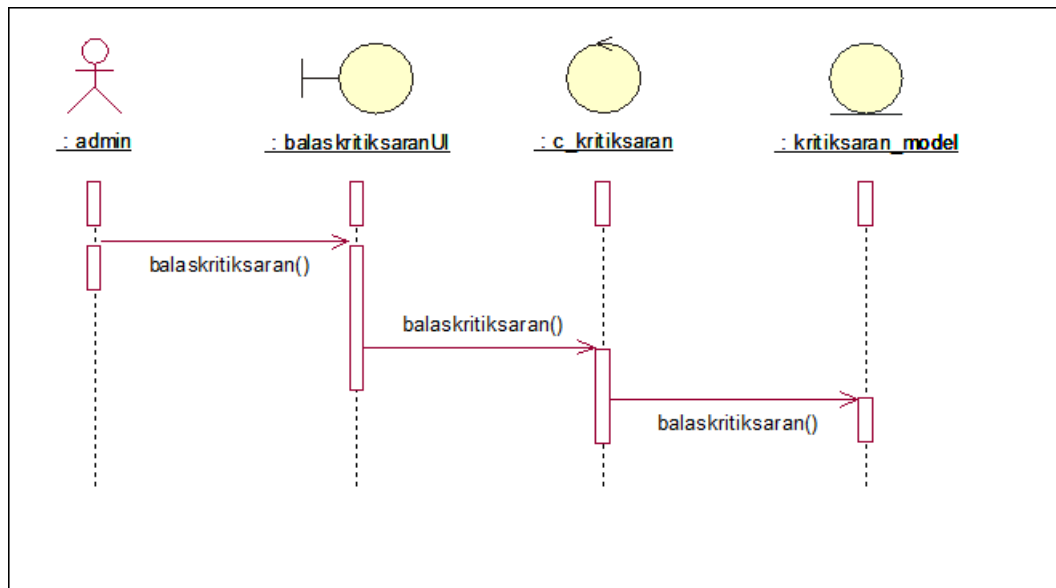
Gambar 3.23 Sequence Diagram: *tambah kritik saran*

#### 3.8.2 Lihat Kritiksaran



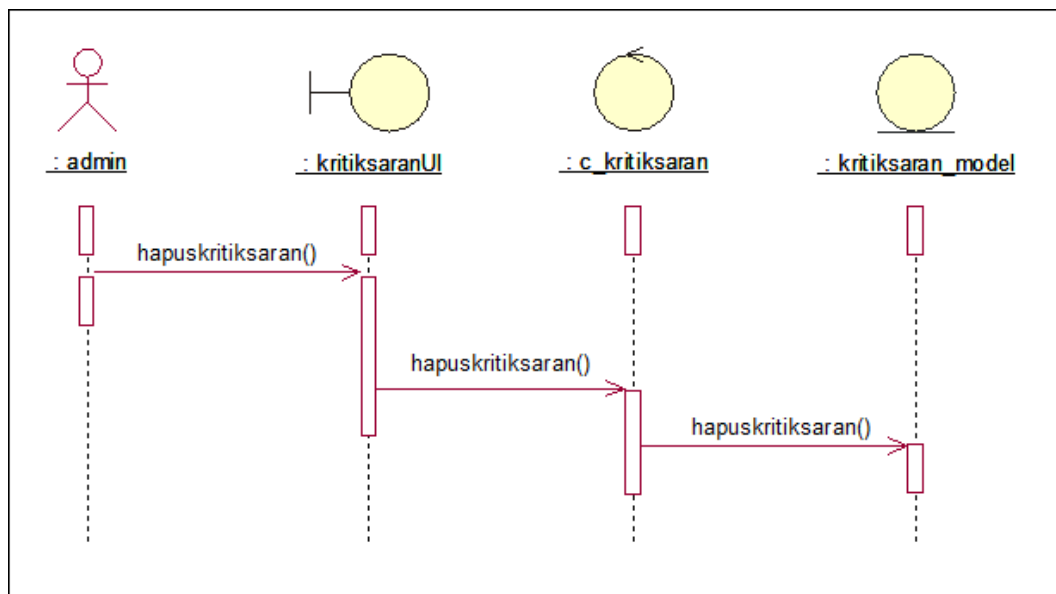
Gambar 3.24 Sequence Diagram: *lihat kritik saran*

### 3.8.3 Balas Kritiksaran



Gambar 3.25 Sequence Diagram: *balas kritik saran*

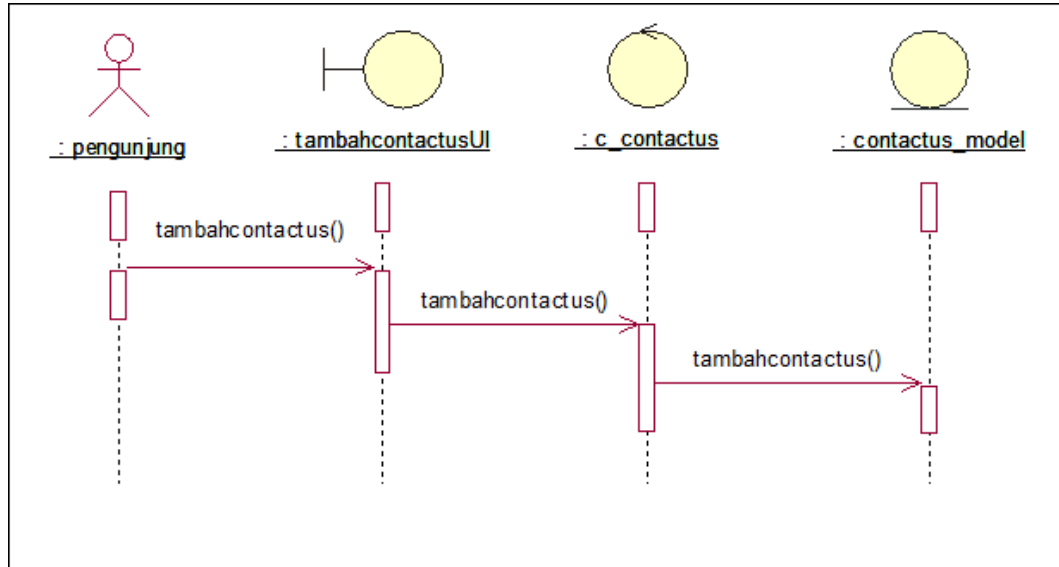
### 3.8.4 Delete Kritiksaran



Gambar 3.26 Sequence Diagram: *hapus kritik saran*

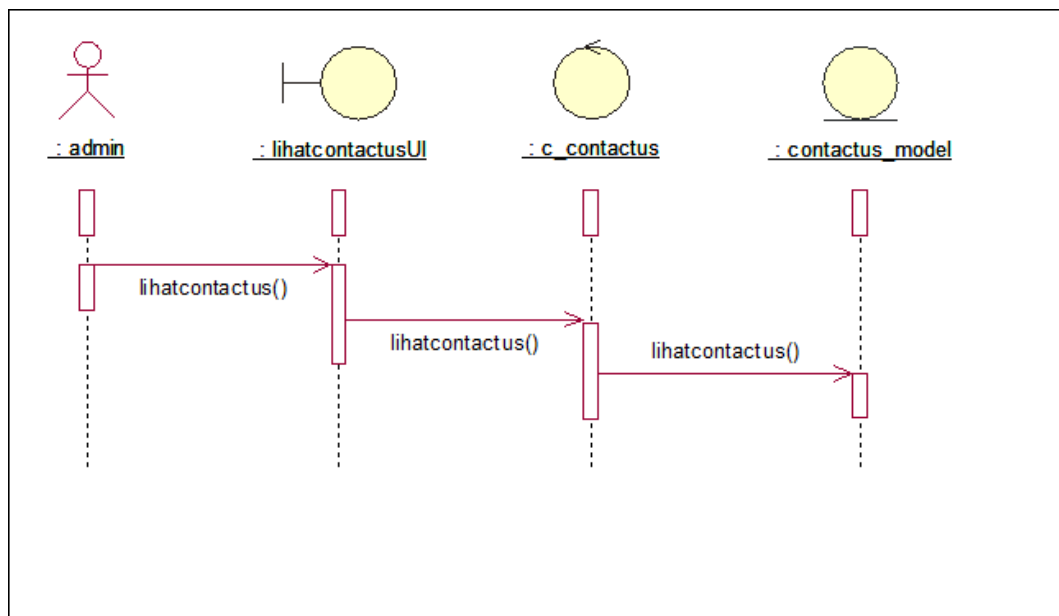
### 3.9 Kontak Kami

#### 3.9.1 Tambah Contactus



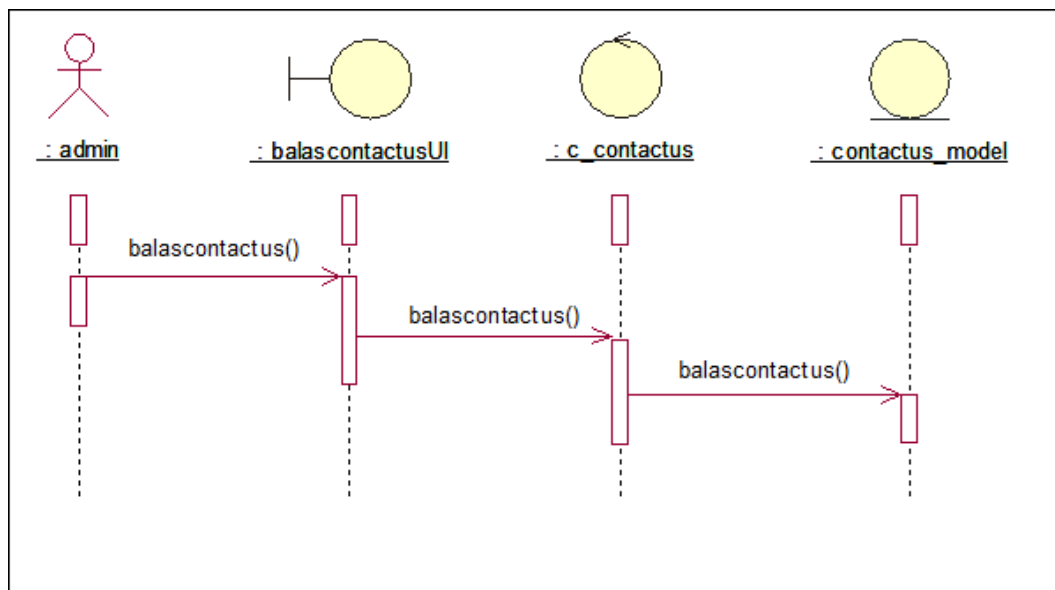
Gambar 3.27 Sequence Diagram: *tambah contactus*

#### 3.9.2 Lihat Contactus



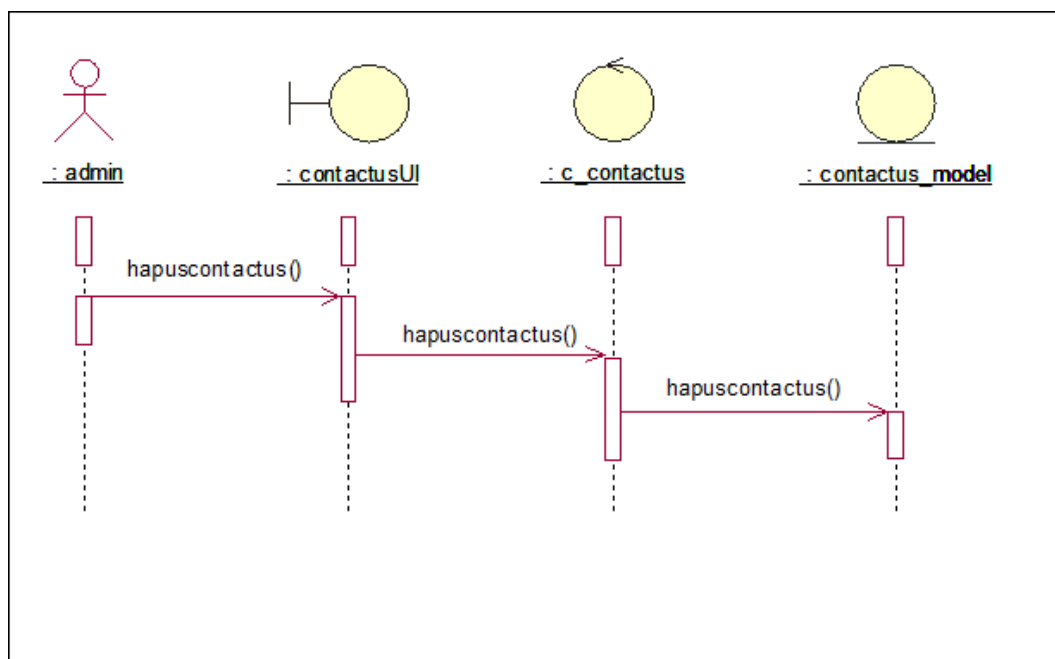
Gambar 3.28 Sequence Diagram: *lihat contactus*

### 3.9.3 Balas Contactus



Gambar 3.29 Sequence Diagram: *balas contactus*

### 3.9.4 Delete Contactus

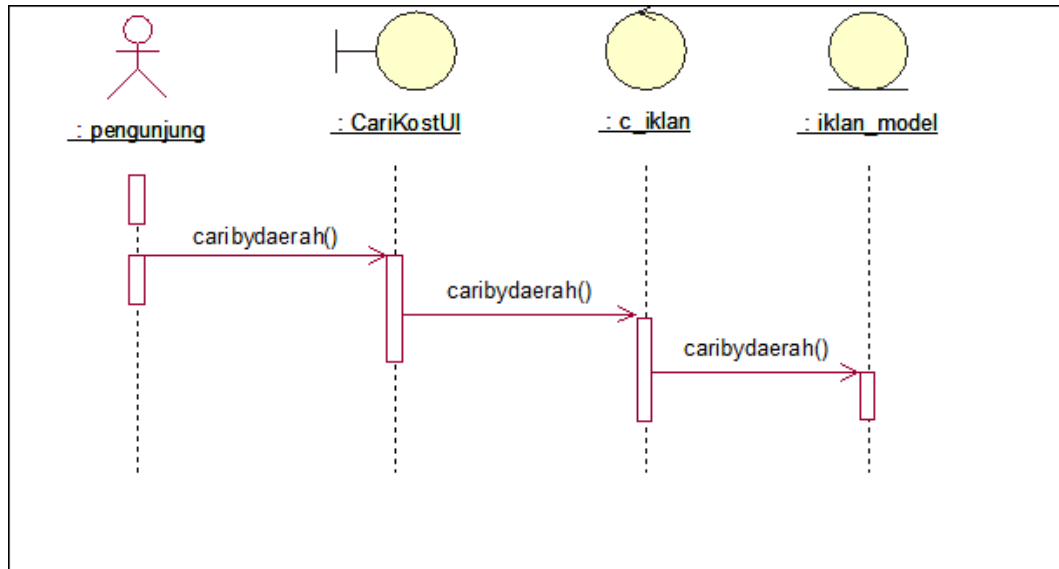


Gambar 3.30 Sequence Diagram: *hapus contactus*



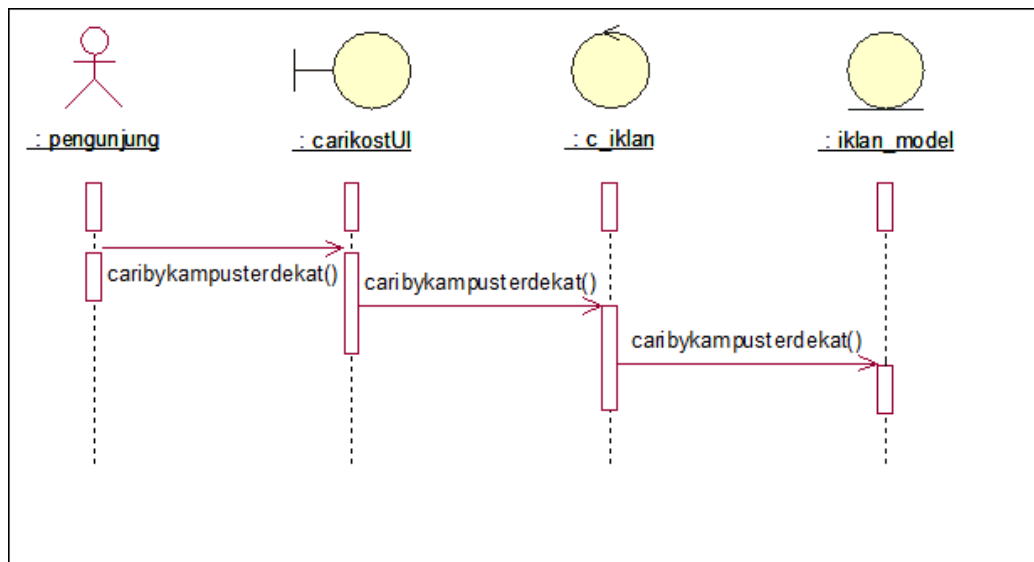
### 3.10 Cari Kost

#### 3.10.1 Cari Berdasarkan Daerah.



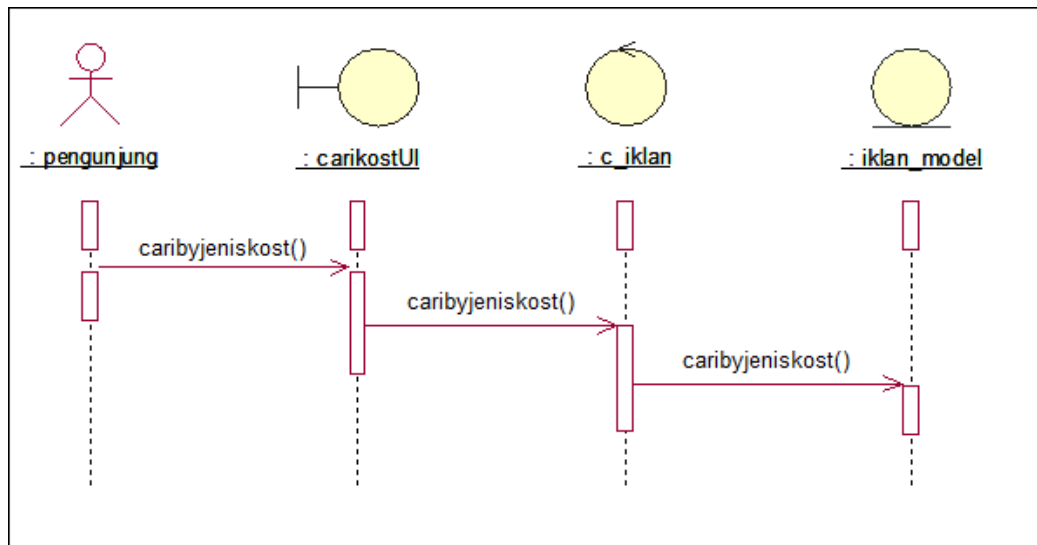
Gambar 3.31 Sequence Diagram: pencarian berdasarkan daerah

#### 3.10.2 Cari Berdasarkan Kampus Terdekat.



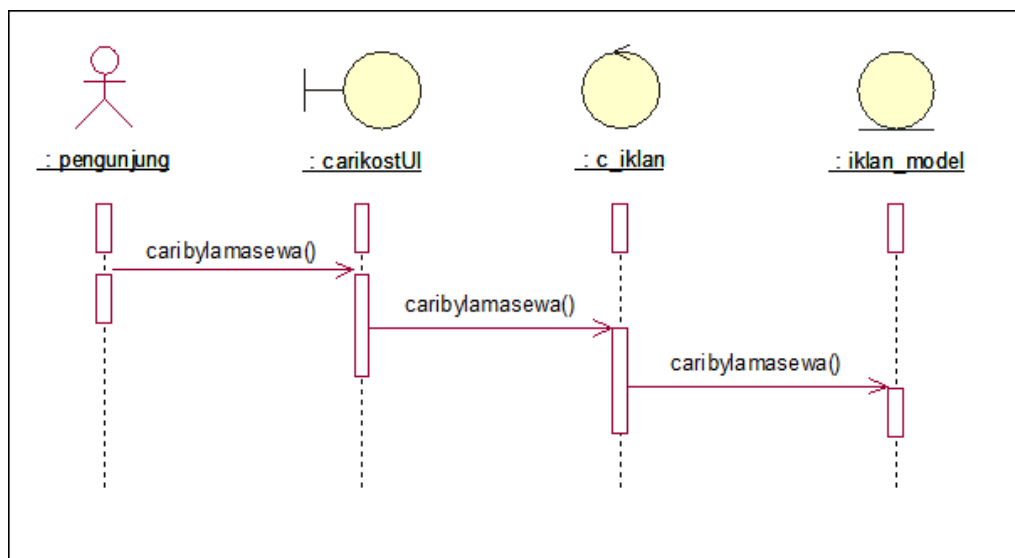
Gambar 3.32 Sequence Diagram: pencarian berdasarkan kampus terdekat

### 3.10.3 Cari Berdasarkan Jenis Kost (gender) .



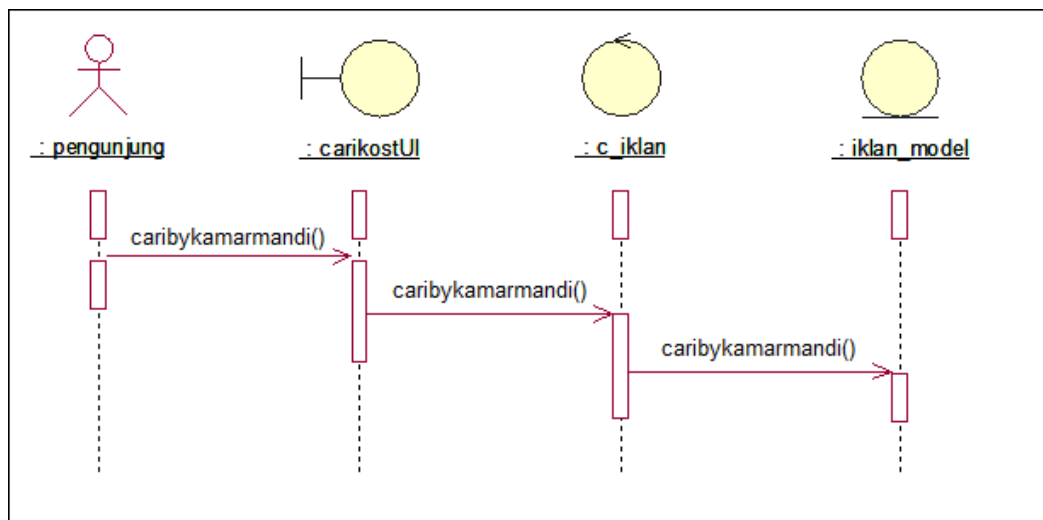
Gambar 3.33 Sequence Diagram: pencarian berdasarkan jenis kost (gender)

### 3.10.4 Cari Berdasarkan Lama Sewa .



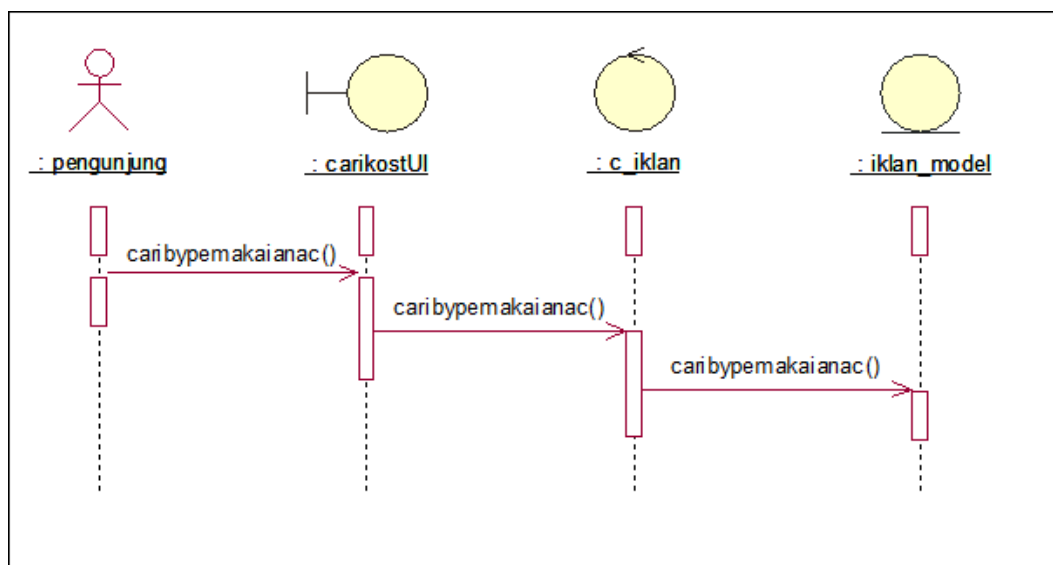
Gambar 3.34 Sequence Diagram: pencarian berdasarkan lama sewa

### 3.10.5 Cari Berdasarkan Kamar Mandi.



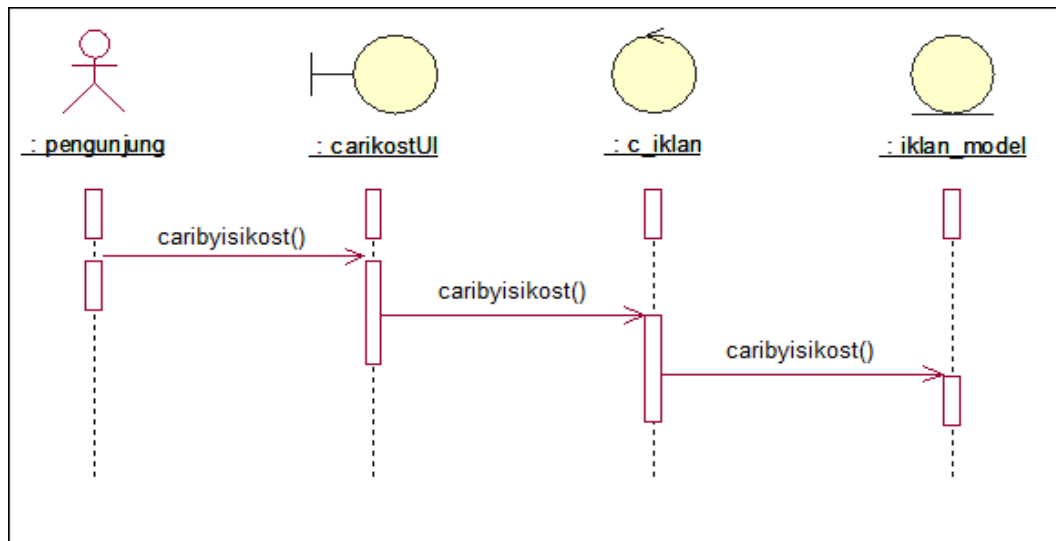
Gambar 3.35 Sequence Diagram: pencarian berdasarkan kamar mandi

### 3.10.6 Cari Berdasarkan Pemakaian Ac.



Gambar 3.36 Sequence Diagram: pencarian berdasarkan pemakaian ac

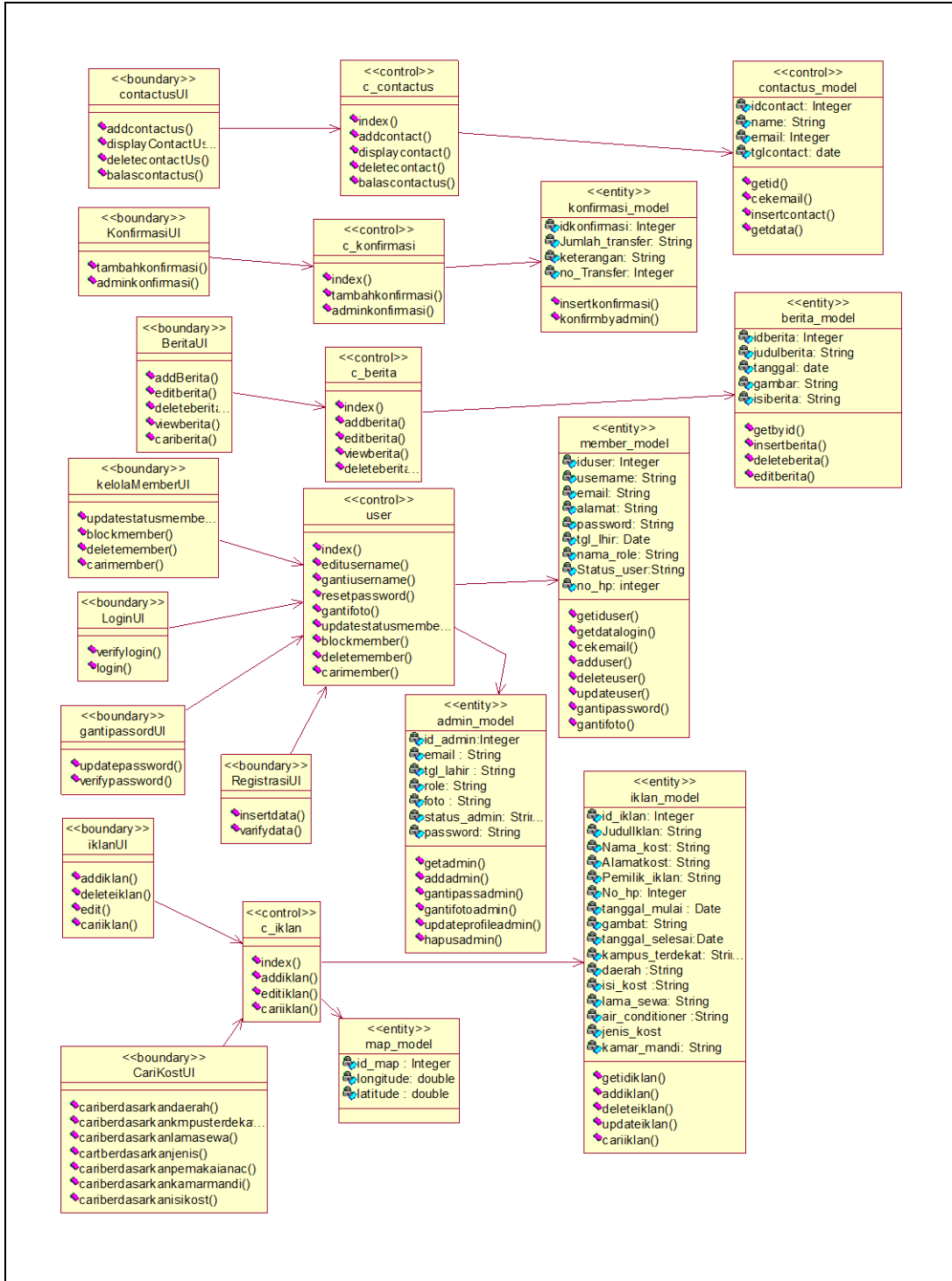
### 3.10.7 Cari Berdasarkan Isi Kost.

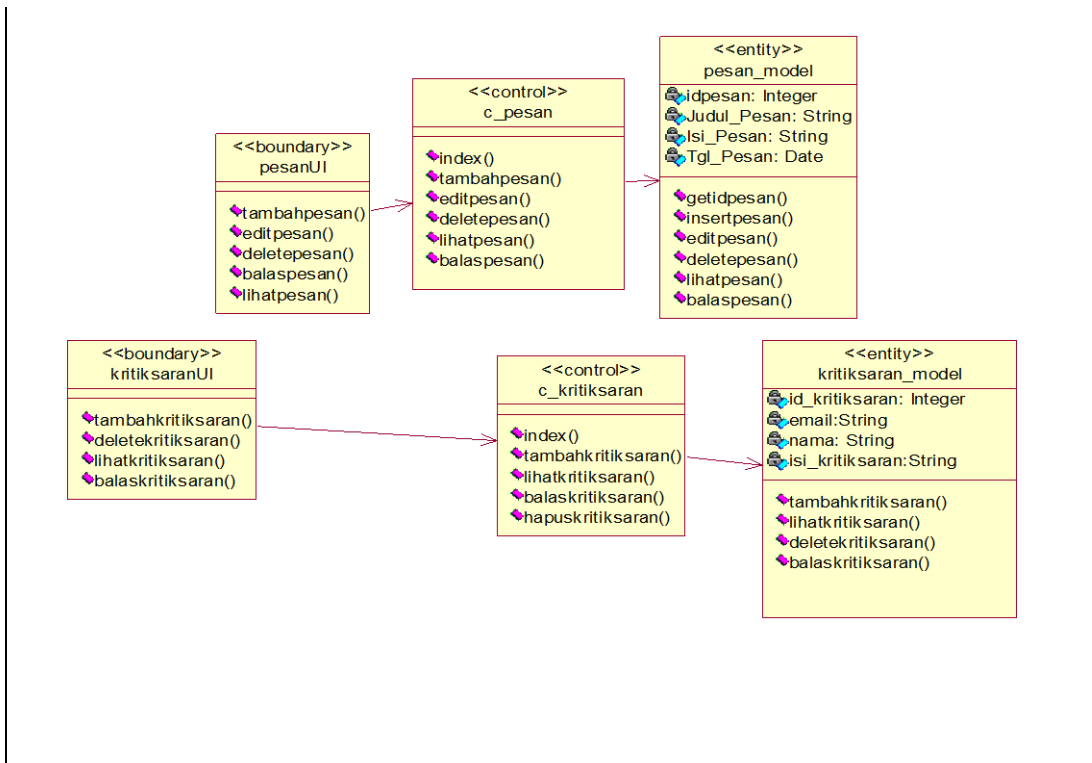


Gambar 3.37 Sequence Diagram: pencarian berdasarkan isi kost

## 4 Class Diagram Specific Descriptions

### 4.1 Class Diagram





Gambar 4 Class Diagram

## 4.2 Deskripsi Kelas

### 4.2.1 Deskripsi Kelas *Entity*

#### 4.2.1.1 *admin*

<i>admin</i>	<<Entity>>
<p>-<i>Idadmin</i>: INT(11) Atribut ini ID <i>admin</i> yang unik, di-generate oleh sistem dan digunakan untuk mengenali data admin.</p> <p>-<i>Nama_admin</i>: Varchar(10) Atribut ini digunakan untuk mengetahui nama admin.</p> <p>-<i>email</i>: Varchar(20) Atribut Ini digunakan identitas <i>User</i> untuk <i>Login</i> ke sistem.</p> <p>-<i>Password</i>: Varchar(20) Digunakan untuk <i>Login</i> ke sistem.</p> <p>-<i>role</i> : Varchar(20) Atribut ini digunakan untuk membedakan role login member dan role login admin.</p> <p>-<i>foto</i> : Varchar(20) Atribut ini digunakan untuk menyimpan data gambar dari admin</p> <p>-<i>status_admin</i> : Varchar(20) Atribut ini digunakan untuk membedakan admin aktif dan admin tidak aktif dan diatur oleh superadmin.</p>	
<p>+<i>admin_model</i>() <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+<i>maxIdadmin</i>(): <i>admin</i> Operasi ini digunakan untuk <i>generate</i> id <i>admin</i> dengan cara mencari nomor id terbesar dari <i>admin</i>.</p> <p>+<i>cekdb</i>(): <i>admin</i> Operasi ini digunakan akan memberi nilai <i>login user</i> berhasil atau gagal.</p> <p>+<i>getPassword</i>():<i>admin</i> Operasi ini digunakan akan mengambil data <i>password admin</i> berdasarkan id <i>admin</i>.</p>	

```

+getdatabystatus(): admin
Operasi ini digunakan akan mengambil data admin berdasarkan
status_admin.
mengambil data tanggal lahir member berdasarkan id admin.
+tambahadmin(): admin
Operasi ini digunakan akan menambah data admin.
+ubahPassword(): admin
Operasi ini digunakan akan mengubah password admin.
+ubahprofile(): admin
Operasi ini digunakan akan mengubah data profile admin.
+gantifoto(): admin
Operasi ini digunakan akan mengganti foto profile admin.

```

#### 4.2.1.2 member

<i>member</i>	<<Entity>>
<p>-Idmember: INT(11) Atribut ini ID <i>member</i> yang unik, di-generate oleh sistem dan digunakan untuk mengenali data <i>member</i>.</p> <p>-Nama_member: Varchar(10) Atribut ini digunakan untuk mengetahui nama <i>member</i>.</p> <p>-email: Varchar(20) Atribut Ini digunakan identitas <i>User</i> untuk <i>Login</i> ke sistem.</p> <p>-Password: Varchar(20) Digunakan untuk <i>Login</i> ke sistem.</p> <p>-role : Varchar(20) Atribut ini digunakan untuk membedakan role login member dan role login admin.</p> <p>-foto : Varchar(20) Atribut ini digunakan untuk menyimpan data gambar dari admin</p> <p>-status_ member: Varchar(20) Atribut ini digunakan untuk membedakan <i>member</i> aktif dan <i>member</i> tidak aktif dan diatur oleh admin dan superadmin.</p>	
<p>+ <i>member_model</i>()</p>	



*Default* konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+maxIdadmin():*member*

Operasi ini digunakan untuk *generate* id *member* dengan cara mencari nomor id terbesar dari *admin*.

+cekdb():*member*

Operasi ini digunakan akan memberi nilai *login user* berhasil atau gagal.

+getPassword():*member*

Operasi ini digunakan akan mengambil data *password admin* berdasarkan id *member*.

+getdatabystatus():*member*

Operasi ini digunakan akan mengambil data *admin* berdasarkan *status\_member*.

mengambil data tanggal lahir *member* berdasarkan id *admin*.

+insertmember(): *member*

Operasi ini digunakan akan menambah data *member*.

+ubahPassword(): *member*

Operasi ini digunakan akan mengubah *password member*.

+ubahprofile(): *member*

Operasi ini digunakan akan mengubah data *profile member*.

+gantifoto(): *member*

Operasi ini digunakan akan mengganti foto *profile member*.

#### 4.2.1.3 iklan

Iklan	<<Entity>>
-Idiklan: INT(11) Atribut ini ID iklan yang unik, di-generate oleh sistem dan digunakan untuk mengenali data iklan.	
-iklan.id_member: INT(11) Id ini digunakan untuk penghubung antara table member dan table iklan	

-iklan.id\_map: INT(11)  
Id ini digunakan untuk penghubung antara table map\_kost dengan table iklan

-Nama\_kost: Varchar(20)  
Atribut ini adalah informasi nama kost pada iklan.

-alamat\_kost: Varchar(20)  
Atribut ini adalah informasi untuk alamat kost.

-gambar\_kost: Varchar(20)  
Atribut ini adalah informasi untuk gambar kost.

-keterangan: varchar(20)  
Atribut ini adalah informasi keterangan.

-Alamat: Varchar(100)  
Atribut ini adalah informasi alamat.

-jlnh\_kmr\_kosong: Varchar(40)  
Atribut ini adalah informasi jumlah kamar kosong.

-kelas\_kost: Varchar(20)

+iklan\_model()  
Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+maxIdiklan(): iklan  
Operasi ini digunakan untuk generate id iklan dengan cara mencari nomor id terbesar dari iklan.

+allDatakost (): iklan  
Operasi ini digunakan untuk mengambil semua data kost.

+tambahiklan(): iklan  
Operasi ini digunakan akan menambah data iklan.

+editiklan(): iklan  
Operasi ini digunakan akan mengubah data iklan.

+deleteiklan(): iklan  
Operasi ini digunakan akan menghapus data iklan.

+lihatiklan(): iklan  
Operasi ini digunakan akan melihat data iklan secara detail.

#### 4.2.1.4 pesan

Pesan	<<Entity>>
<p>-Idpesan: INT(11) Atribut ini ID pesan yang unik, di-generate oleh sistem dan digunakan untuk mengenali data pesan.</p> <p>-pesan.id_member : INT(11) Untuk menghubungkan table pesan dengan table member, dimana pesan mengambil data member untuk menampilkan nama dipesan.</p> <p>-pesan.id_admin : INT(11) Untuk menghubungkan table pesan dengan table admin, dimana pesan mengambil data admin untuk menampilkan nama dipesan.</p> <p>-nama_pengirim: Varchar(20) Atribut ini adalah informasi nama pengirim pesan.</p> <p>-nama_tujuan: Varchar(20) Atribut ini adalah informasi nama tujuan pesan.</p> <p>-isi_pesan: Varchar(20) Atribut ini adalah informasi isi pesan yang akan disampaikan ke tujuan.</p>	
<p>+pesan_model() <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+maxIdpesan(): pesan Operasi ini digunakan untuk <i>generate</i> id member dengan cara mencari nomor id terbesar dari pesan.</p> <p>+getoutbox(): Pesan Operasi ini digunakan untuk mengambil semua outbox data pesan sesuai dengan nama userlogin.</p> <p>+getoutinbox(): Pesan Operasi ini digunakan untuk mengambil semua inbox data pesan sesuai dengan nama userlogin.</p> <p>+tambahpesan(): pesan Operasi ini digunakan akan menambah data pesan.</p> <p>+balaspesan(): Pesan Operasi ini digunakan untuk membalas datapesan.</p>	

```
+deletePesan(): Pesan
```

Operasi ini digunakan akan menghapus data Pesan.

#### 4.2.1.5 Berita

Berita-Aturan	<<Entity>>
<pre>-IdBerita: INT(11)</pre> <p>Atribut ini ID Berita yang unik, di-generate oleh sistem dan digunakan untuk mengenali data Berita.</p> <pre>-berita.id_admin: INT(11)</pre> <p>Atribut ini merupakan ID admin untuk ditampilkan di penulis berita peraturan</p> <pre>-Tgl_Post: Datetime</pre> <p>Atribut ini adalah informasi tanggal posting berita-aturan.</p> <pre>-Judul_Berita: Varchar(100)</pre> <p>Atribut ini adalah informasi judul berita-aturan.</p> <pre>Isi_Berita: Varchar(10)</pre> <p>Atribut ini adalah informasi isi berita-aturan.</p> <pre>-Status: Varchar(10)</pre> <p>Atribut ini adalah informasi status berita-aturan.</p>	
<pre>+berita_model()</pre> <p>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <pre>+maxIdBerita(): Berita-Aturan</pre> <p>Operasi ini digunakan untuk generate id berita-aturan dengan cara mencari nomor id terbesar dari berita-aturan.</p> <pre>+ViewBerita(): Berita-Aturan</pre> <p>Operasi ini digunakan untuk mengambil semua data berita dan dapat juga mengambil data Berita tertentu.</p> <pre>+allBeritaAturan(): Berita-Aturan</pre> <p>Operasi ini digunakan untuk mengambil semua data berita-aturan dan dapat juga mengambil data berita-aturan tertentu.</p> <pre>+tambahBeritaAturan(): Berita-Aturan</pre> <p>Operasi ini digunakan akan menambah data berita-aturan.</p>	

```
+editBeritaAturan(): Berita-Aturan
Operasi ini digunakan akan mengubah data berita-aturan.
+hapusBeritaAturan(): Berita-Aturan
Operasi ini digunakan akan menghapus data berita-aturan.
```

#### 4.2.1.6 Konfirmasi

Konfirmasi	<<Entity>>
<pre>-Id_konfirmasi: INT(11) Atribut ini ID konfirmasi yang unik, di-generate oleh sistem dan digunakan untuk mengenali data konfirmasi. Rek_tujuan Konfirmasi_pembayaran.id_member: INT(11) Atribut ini digunakan untuk mengambil data konfirmasi pembayaran. -Tgl_transfer: Datetime Atribut ini mengetahui tanggal konfirmasi pembayaran. -jumlah_uang : INT(30) Atribut ini digunakan untuk mengetahui jumlah uang yang ditransfer Nama_direkening: String Atribut ini digunakan untuk menyimpan nama rekening pengirim. -rek_tujuan: INT(30) Atribut ini digunakan untuk menyimpan data rek tujuan yang dipilih -rek_pengirim: INT(30) Atribut ini digunakan untuk menyimpan rek pengirim Jenis_paket : String Atribut ini digunakan untuk menyimpan data paket yang dipilih oleh member</pre>	
<pre>+ Konfirmasi_model () Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</pre>	

```

+tambahkonfirmasi(): Konfirmasi
Operasi ini digunakan oleh member untuk melakukan menambah
konfirmasi
+konfirmasirequestmember(): Konfirmasi
Operasi ini digunakan oleh admin untuk melakukan approve
konfirmasi yang diminta oleh member.

```

#### 4.2.1.7 Map

map	<<Entity>>
-Id_map: INT(11) Atribut ini ID map yang unik, di-generate oleh sistem dan digunakan untuk mengenali data map.	
-longitude :double Atribut ini digunakan untuk menyimpan latitude dari map	
-longitude :double Atribut ini digunakan untuk menyimpan longitude dari map	
+ Map_model () Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

#### 4.2.1.8 kritiksaran

kritiksaran	<<Entity>>
-Id_kritiksaran: INT(11) Atribut ini ID kritiksaran yang unik, di-generate oleh sistem dan digunakan untuk mengenali data kritiksaran.	
-nama :String Atribut ini digunakan untuk menyimpan nama pengirim kritiksaran	
-email : String Atribut ini digunakan untuk menyimpan email pengirim kritik saran	
-isi_kritiksaran : String Atribut ini digunakan untuk menyimpan isi kritiksaran	

```

+ kritiksaran_model ()
Default konstruktor, digunakan untuk inisialisasi semua
atribut dari kelas ini.
+tambahkritiksaran(): Kritiksaran
Operasi ini bisa digunakan oleh pengunjung untuk mengirimkan
kritik dan saran.
+balaskritiksaran(): Kritiksaran
Operasi ini bisa digunakan oleh admin untuk membalas kritik
dan saran.
+lihatkritiksaran(): Kritiksaran
Operasi ini bisa digunakan oleh admin untuk melihat kritik
dan saran.
+deletekritiksaran(): Kritiksaran
Operasi ini bisa digunakan oleh admin untuk menghapus kritik
dan saran.

```

#### 4.2.1.8 contactus

<b>kritiksaran</b>	<b>&lt;&lt;Entity&gt;&gt;</b>
<p>-Id_ contactus: INT(11)  Atribut ini ID contactus yang unik, di-generate oleh sistem dan digunakan untuk mengenali data contactus.</p> <p>-nama : String  Atribut ini digunakan untuk menyimpan nama pengirim contactus</p> <p>-email : String  Atribut ini digunakan untuk menyimpan email pengirim kritik saran</p> <p>-isi_ contactus: String  Atribut ini digunakan untuk menyimpan isi contactus</p>	
<pre> + contactus_model () Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. </pre>	

<p>+tambahcontactus():contactus Operasi ini bisa digunakan oleh pengunjung untuk mengirimkan contactus.</p> <p>+balascontactus():contactus Operasi ini bisa digunakan oleh admin untuk membalas contactus.</p> <p>+lihatcontactus():contactus Operasi ini bisa digunakan oleh admin untuk melihat contactus.</p> <p>+deletecontactus():contactus Operasi ini bisa digunakan oleh admin untuk menghapus contactus.</p>
---

#### 4.2.2 Deskripsi Kelas *Control*

##### 4.2.2.1 *User*

<b>User</b>	<b>&lt;&lt;control&gt;&gt;</b>
<p>+User() <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+VerifyLogin(): User Operasi ini digunakan untuk memverifikasi data <i>user</i> yang tersimpan di <i>database</i>.</p> <p>+Profil(): User Operasi ini digunakan untuk melihat data <i>user</i> yang tersimpan di <i>database</i>.</p> <p>+UbahPassword(User) Operasi ini digunakan untuk mengubah <i>password</i> <i>User</i> tertentu kedalam <i>database</i>.</p> <p>+ResetPassword(User) Operasi ini digunakan untuk me-reset <i>password</i> <i>User</i> tertentu</p>	



kedalam *database*.

+*Registrasi(User)*  
 Operasi ini digunakan untuk mendaftar

+*gantifoto(User)*  
 Operasi ini digunakan untuk mengganti foto profile

+*Logout()*  
 Operasi ini digunakan untuk menghapus sesion *login User*.

#### 4.2.2.2 c\_iklan

<b>C_iklan</b>	<b>&lt;&lt;control&gt;&gt;</b>
<p>+<i>Index()</i>  <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+<i>Tambahiklan(iklan)</i>          Operasi ini digunakan untuk menambah data iklan kedalam <i>database</i>.</p> <p>+<i>Editiklan(iklan)</i>          Operasi ini digunakan untuk mengubah data iklan kedalam <i>database</i>.</p> <p>+<i>hapusiklan(iklan)</i>          Operasi ini digunakan untuk hapus data iklan kedalam <i>database</i>.</p> <p>+<i>lihatiklan(iklan)</i>          Operasi ini digunakan untuk menampilkan iklan.</p>	

#### 4.2.2.3 c\_pesanan

<b>C_pesanan</b>	<b>&lt;&lt;control&gt;&gt;</b>
<p>+<i>Index()</i>  <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+<i>Tambahpesanan(pesanan)</i></p>	

```

Operasi ini digunakan untuk menambah data pesan kedalam
database.
+Editpesan(pesan)
Operasi ini digunakan untuk mengubah data pesan kedalam
database.
+hapusipesan(pesan)
Operasi ini digunakan untuk menghapus data pesan kedalam
database.
+balaspesan(pesan)
Operasi ini digunakan untuk membalas data pesan kedalam
database.

```

#### 4.2.2.4 c\_berita

c_berita	<<control>>
<pre> +Index() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +TambahBerita (Berita) Operasi ini digunakan untuk menambah data BeritaAturan kedalam database. +EditBerita (Berita) Operasi ini digunakan untuk mengubah data BeritaAturan kedalam database. +HapusBerita (Berita) Operasi ini digunakan untuk menghapus data BeritaAturan kedalam database. +lihatberita (Berita) Operasi ini digunakan untuk melihat data. </pre>	

#### 4.2.2.5 c\_konfirmasi

c_konfirmasi	<<control>>
<p>+Index() <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+TambahPembayaran(konfirmasi) Operasi ini digunakan untuk menambah data pembayaran.</p> <p>+adminkonfirmasi(konfirmasi) Operasi ini digunakan oleh admin untuk approve konfirmasi dari member.</p>	

#### 4.2.2.6 c\_kritik saran

c_kritik saran	<<control>>
<p>+Index() <i>Default</i> konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+Tambahkritik saran(konfirmasi) Operasi ini digunakan untuk menambah data kritik saran.</p> <p>+balaskritik saran(konfirmasi) Operasi ini digunakan untuk membalas kritik saran.</p> <p>+editkritik saran(konfirmasi) Operasi ini digunakan untuk mengedit kritik saran.</p> <p>+Tambahkritik saran(konfirmasi) Operasi ini digunakan untuk menghapus kritik saran.</p>	

#### 4.2.2.7 c\_contactus

<b>c_contactus</b>	<b>&lt;&lt;control&gt;&gt;</b>
<pre>+Index() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +Tambahcontactus (konfirmasi) Operasi ini digunakan untuk menambah contactus. +balascontactus (konfirmasi) Operasi ini digunakan untuk membalas contactus. +editcontactus (konfirmasi) Operasi ini digunakan untuk mengedit contactus. +Tambahcontactus (konfirmasi) Operasi ini digunakan untuk menghapus contactus.</pre>	

#### 4.2.3 Deskripsi Kelas *Boundary*

##### 4.2.3.1 login

<b>loginUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>-login() Menampilkan form login yang digunakan untuk memasuki sistem.</pre>	

##### 4.2.3.2 registrasi

<b>registrasiUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
<pre>-DisplayProfil() Menampilkan form registrasi yang digunakan untuk registrasi.</pre>	

#### 4.2.3.3 resetpassword

<b>resetpasswordUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
-ResetPassword() Form untuk me-reset password user.	

#### 4.2.3.4 ubahprofile

<b>ubahprofileUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
-Ubah Profile() Form untuk mengubah data profil user.	

#### 4.2.3.5 gantipassword

<b>gantipasswordUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
-gantiPassword() Form untuk mengubah password user.	

#### 4.2.3.1 gantifoto

<b>gantifotoUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
-gantiFoto() Form untuk mengubah foto profil user.	

#### 4.2.3.5 iklan

<b>iklanUI</b>	<b>&lt;&lt;boundary&gt;&gt;</b>
-Displayiklan() Menampilkan data semua iklan. -Tambahiklan() Form untuk menambah data iklan baru. -Editiklan()	

*Form* untuk mengubah data iklan.

-hapusiklan()

*Form* untuk menghapus iklan.

#### 4.2.3.3 Berita

**BeritaAturanUI**

**<<boundary>>**

-DisplayBerita()

Menampilkan data semua beritaaturan.

-TambahBerita()

*Form* untuk menambah data beritaaturan baru.

-EditBerita()

*Form* untuk mengubah data beritaaturan.

-HapusBerita()

Konfirmasi untuk menghapus data beritaaturan.

#### 4.2.3.4 Pesan

**pesanUI**

**<<boundary>>**

-Displaypesan()

Menampilkan data semua pesan.

-Tambahpesan()

*Form* untuk menambah data pesan baru.

-Hapuspesan()

*Form* untuk menghapus pesan.

-Lihatpesan()

*Form* untuk melihat pesan.

#### 4.2.3.5 Konfirmasi Pembayaran

**PembayaranUI**

**<<boundary>>**

-Tambahkonfirmasi()

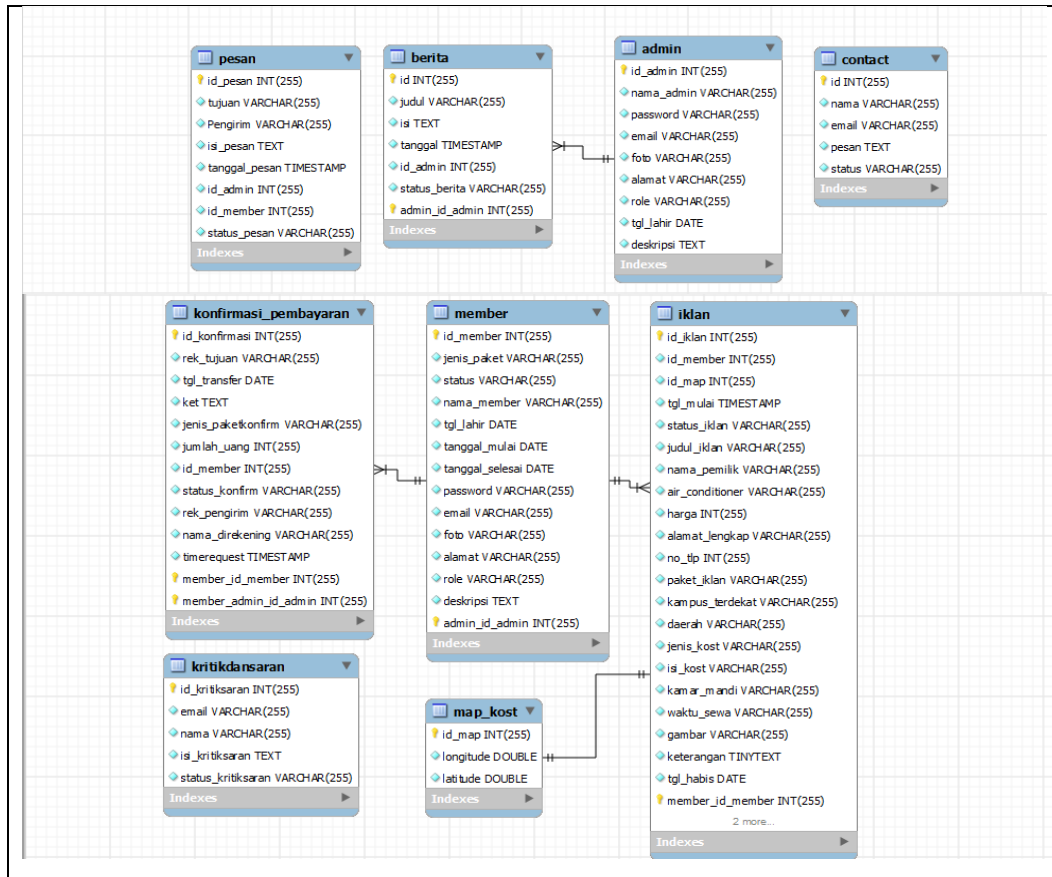
*Form* untuk menambah data konfirmasi pembayaran.

-Adminkonfirmasi()

*Form* untuk approve data konfirmasi pembayaran.

## 5. Perancangan Data

### 5.1 Physical Data Model



Gambar 5 Physical Data Model



## 5.2 Dekomposisi Data

### 5.2.1 Deskripsi Entitas Data Pesan

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDPESAN	INT	-	Id Pesan, <i>Primary key</i>
PESAN.ID_MEMBER	INT	-	Id Member, <i>Foreign Key</i> dari member
PESAN.ID_ADMIN	INT	-	Id ADMIN, <i>Foreign Key</i> dari admin
TGL_Pesan	<i>Datetime</i>	-	Tanggal Kirim Pesan
ISI PESAN	<i>Varchar</i>	255	Isi Pesan

### 5.2.2 Deskripsi Entitas Data BeritaAturan

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDBERITA	INT	11	Id Berita-Aturan, <i>Primary key</i>
BERITA.ID_ADMIN	INT	-	Id ADMIN, <i>Foreign Key</i> dari admin
TGL_POST	<i>Datetime</i>	-	Tanggal posting berita-aturan
JUDUL_BERITA	<i>Varchar</i>	30	Judul berita-aturan
ISI_BERITA	<i>Varchar</i>	200	Isi berita-aturan
STATUS	<i>Varchar</i>	10	Status dari berita-aturan, Show atau Hide

### 5.2.3 Deskripsi Entitas Data admin

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
ID ADMIN	INT	11	Id admin, <i>Primary key</i>
NAMA_ROLE	<i>Varchar</i>	10	Nama Role pengguna

USERNAME	Varchar	20	Username yang digunakan pengguna untuk mengenali user, Unik
EMAIL	VARVHAR	40	email yang digunakan pengguna untuk Login, Unik
PASSWORD	Varchar	20	Password yang digunakan pengguna untuk Login

#### 5.2.4 Deskripsi Entitas Data *member*

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
ID MEMBER	INT	11	Id admin, <i>Primary key</i>
NAMA_ROLE	Varchar	10	Nama Role pengguna
USERNAME	Varchar	20	Username yang digunakan pengguna untuk mengenali user, Unik
EMAIL	VARVHAR	40	email yang digunakan pengguna untuk Login, Unik
PASSWORD	Varchar	20	Password yang digunakan pengguna untuk Login

### 5.2.5 Deskripsi Entitas Data Iklan

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDIKLAN	INT	11	Id Member, <i>Primary key</i>
IKLAN.ID_MEMBER	INT	-	Id Member, <i>Foreign Key</i> dari member
JUDUL_IKLAN	Varchar	20	Judul iklan
NAMA_PEMILIK	Varchar	20	Nama pemilik
ALAMAT_KOST	Varchar	20	Alamat kost
KETERANGAN	Varchar	1000	keterangan
EMAIL	Varchar	40	Email
TLP	Varchar	20	No telepon aktif
FOTO	Varchar	20	Path foto profil
KAMAR_KOSONG	INT	-	Kamar kosong
KAMPUS_TERDEKAT	Varchar	20	Kampus terdekat
KAMAR_MANDI	Varchar	20	Kamar mandi
PEMAKAIAN AC	Varchar	20	Pemakaian ac
LAMA_SEWA	Varchar	20	Lama sewa
JENIS_KOST	Varchar	20	Jenis kost
ISI_KOST	Varchar	20	Isi kost
DAERAH	Varchar	20	daerah
TANGGAL_MULAI	Date	-	Tanggal mulai
TANGGAL_SELESAI	Date	-	Tanggal selesai
STATUS	Varchar	10	Status kost

### 5.2.6 Deskripsi Entitas Data Pembayaran

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDPEMBAYARAN	INT	11	Id Pembayaran, <i>Primary key</i>
PEMBAYARAN.ID_MEMBER	INT	-	Id Member, <i>Foreign Key</i> dari member
NAMA	Varchar	20	Nama lengkap Pengirim
REK_PENGIKIRIM	Varchar	20	Rekening pengirim
REK_TUJUAN	Varchar	20	Rekening tujuan
PILIHAN_PAKET	Varchar	20	Pilihan paket
TANGGAL_TRANSFER	Date	-	Tanggal transfer
Jumlah	Varchar	20	Jumlah uang yang dikirim

### 5.2.7 Deskripsi Entitas Data Kritiksaran

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDKRITIKSARAN	INT	11	Id kritksaran, <i>Primary key</i>
KRITIKSARAN.ID_MEMBER	INT	-	Id Member, <i>Foreign Key</i> dari member
NAMA	Varchar	20	Nama lengkap Pengirim
EMAIL	Varchar	255	Email
ISI_KRITIKSARAN	Varchar	255	Isi kritik saran
STATUS	Varchar	255	Status kritik saran

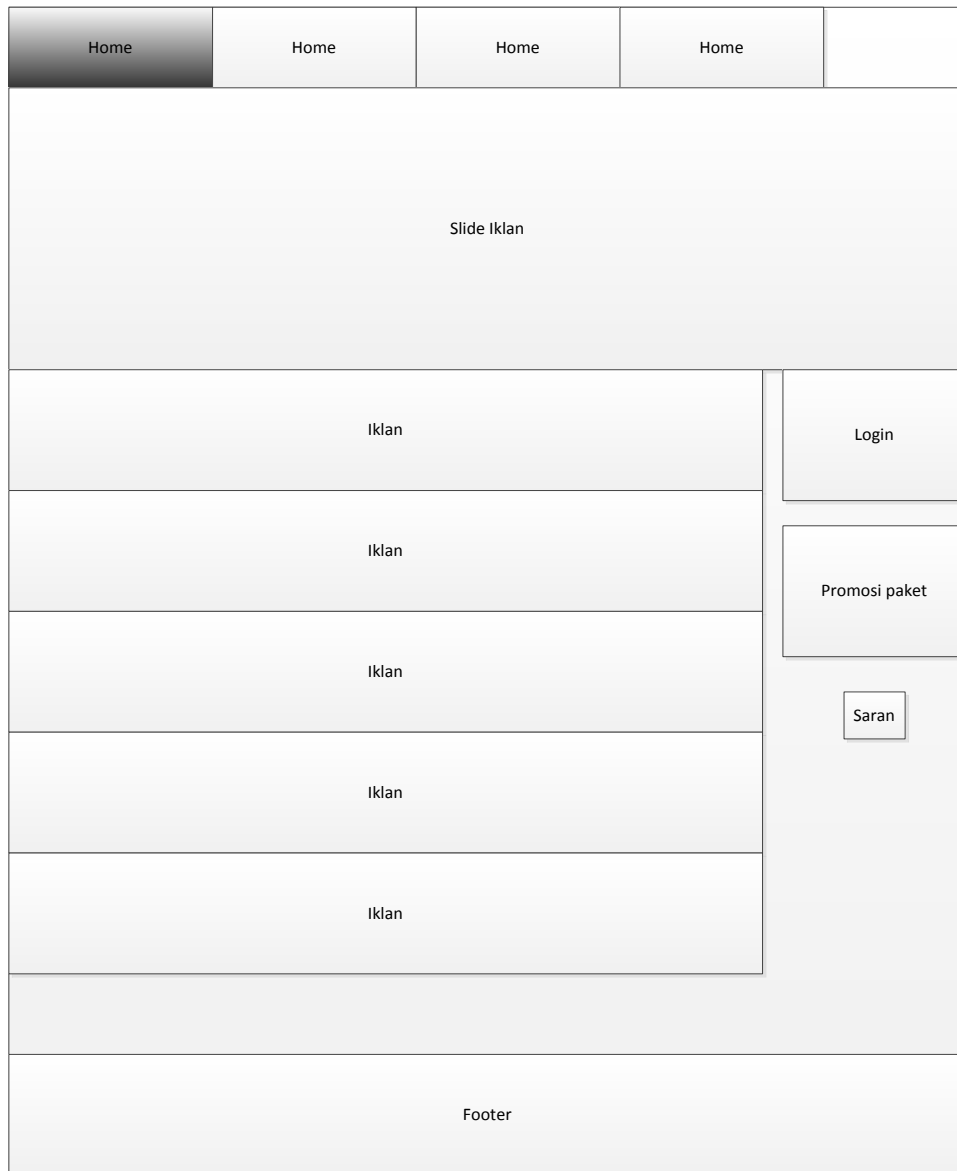
### 5.2.7 Deskripsi Entitas Data contactus

<b>Nama</b>	<b>Tipe</b>	<b>Panjang</b>	<b>Keterangan</b>
IDCONTACTUS	INT	11	Id contactus, <i>Primary key</i>
CONTACTUS.ID_MEMBER	INT	-	Id Member, <i>Foreign Key</i> dari member
NAMA	Varchar	20	Nama lengkap Pengirim
EMAIL	Varchar	255	Email
ISI_CONTACTUS	Varchar	255	Isi contactus
STATUS	Varchar	255	Status contactus

## 5.3 Deskripsi Perancangan Antarmuka

### 5.3.1 Halaman Luar

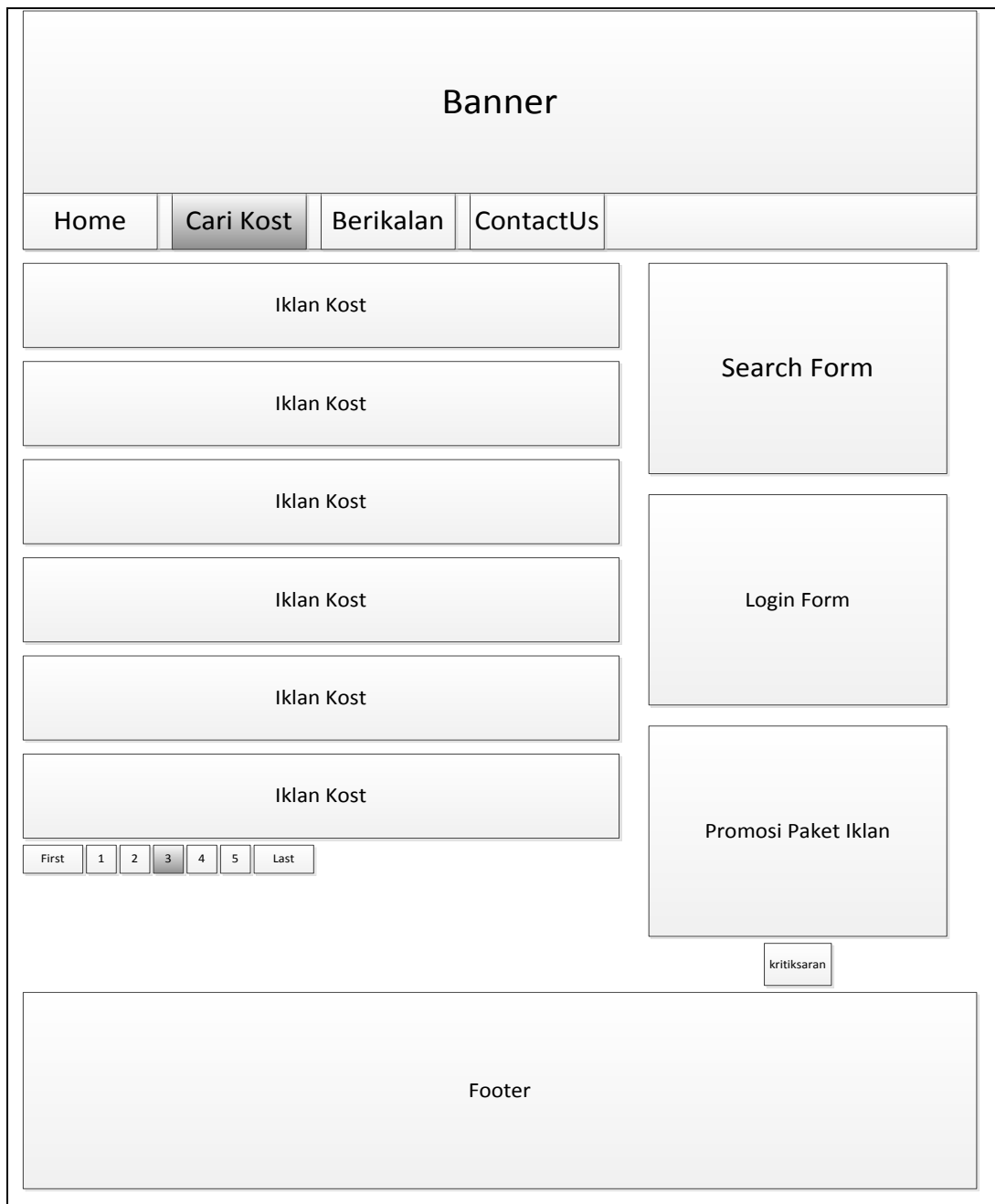
#### 5.3.1.1 Halaman Pertama



Gambar 5.1 Rancangan Antarmuka Halaman Pertama

Pada gambar 5.1 terdiri dari menu, slide iklan, promosi paket, dihalaman contentnya berisi iklan, dibagian aside terdiri dari login form, promosi paket dan keuntungan beriklan disini, ada box kritik dan saran lalu footer.

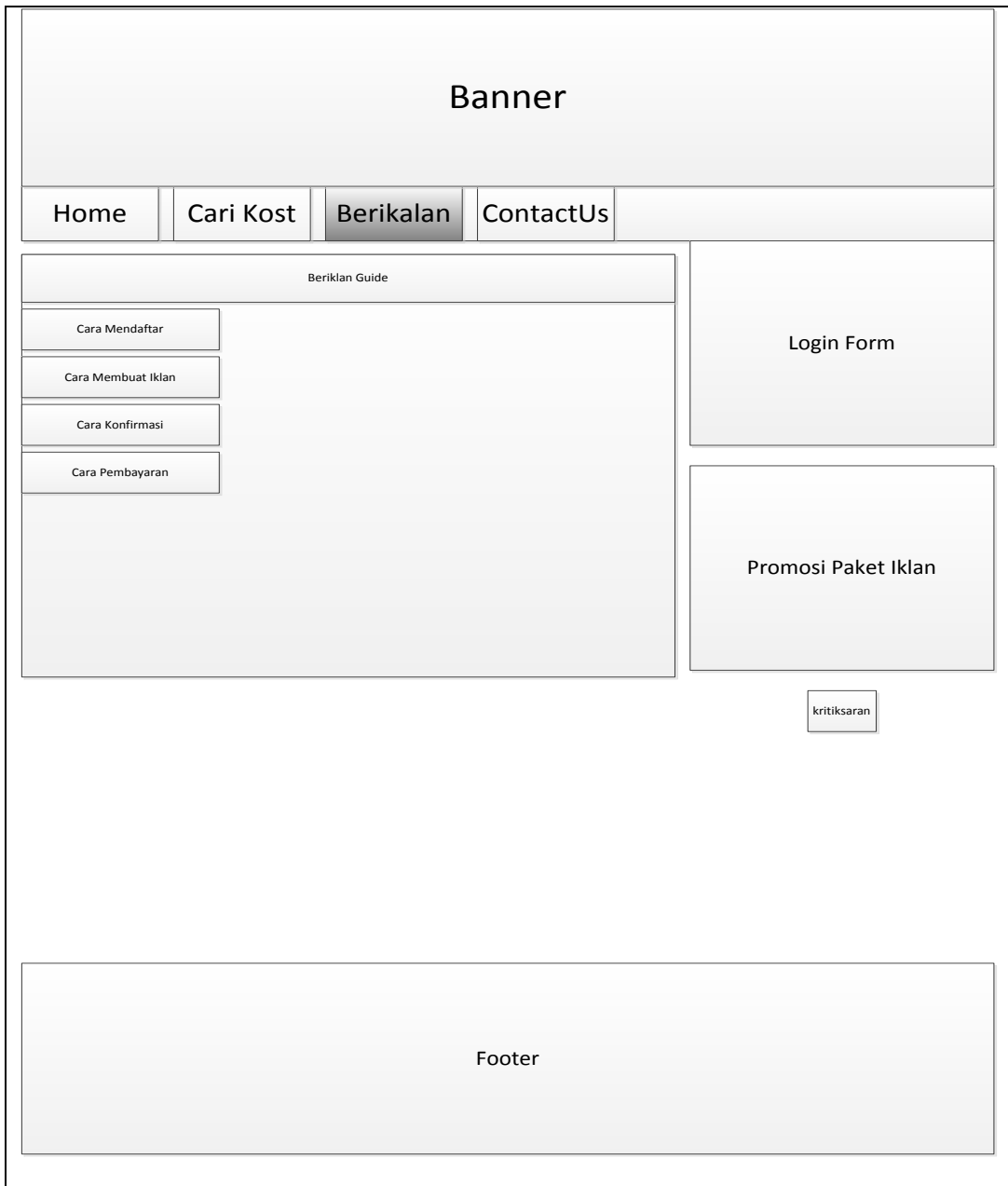
### 5.3.1.2 Cari Kost



Gambar 5.2 Rancangan Antarmuka Cari Kost

Pada gambar 5.2 terdiri dari banner, menu, dihalaman contentnya berisi iklan, dibagian aside terdiri dari search form, login form, promosi paket dan keuntungan beriklan disini, ada box kritik dan saran lalu footer.

### 5.3.1.3 Beriklan



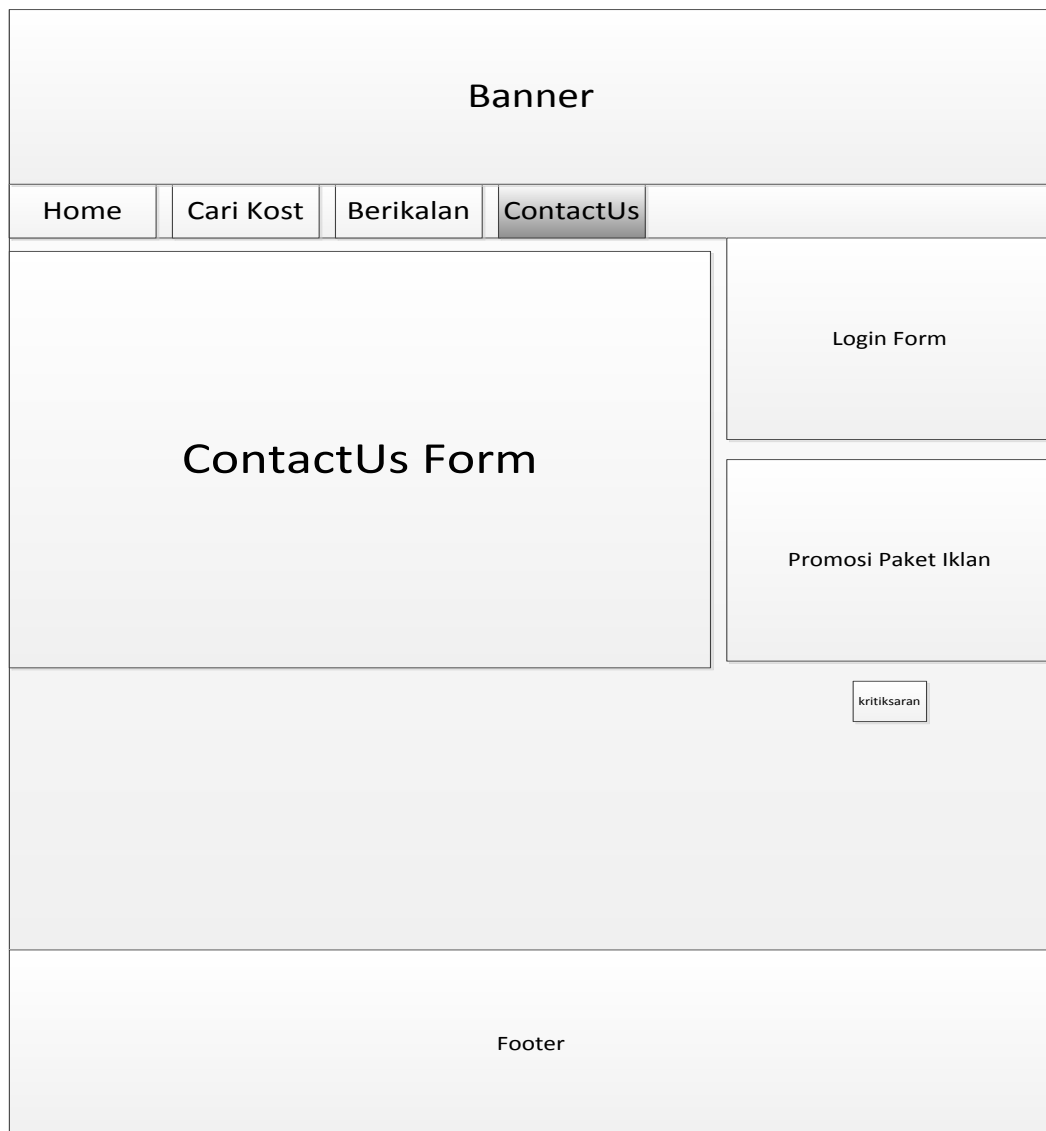
Gambar 5.3 Rancangan Antarmuka Beriklan

Pada gambar 5.3 terdiri dari banner, menu, dibagian content adalah guide untuk mendaftar, bayar, dan konfirmasi. dihalaman contentnya berisi iklan, dibagian aside terdiri dari login form, promosi paket dan



keuntungan beriklan disini, ada box kritik dan saran lalu footer.

#### 5.3.1.4 ContactUs

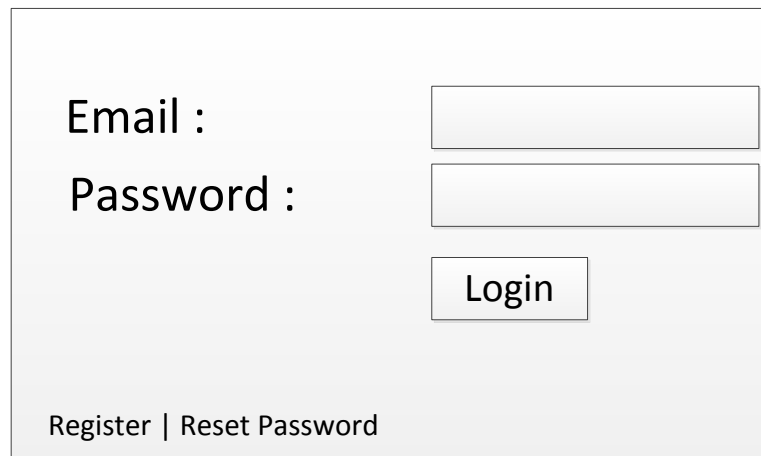


Gambar 5.4 Rancangan Antarmuka contactus

Halaman gambar 5.4 terdiri dari banner, menu. dihalaman contentnya form contactus, dibagian aside terdiri dari login form, promosi paket dan keuntungan beriklan disini, ada box kritik dan saran lalu footer.

#### 5.3.2 User

### 5.3.2.1 *Login*



The image shows a user login form with a light gray background and a thin black border. It contains the following elements:

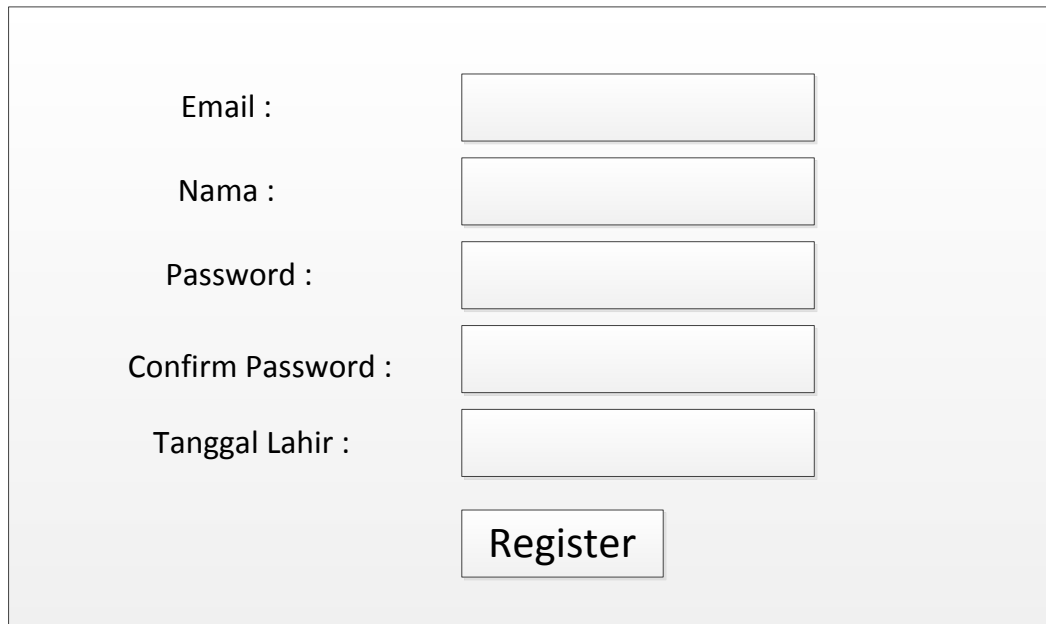
- An "Email :" label followed by a rectangular input field.
- A "Password :" label followed by a rectangular input field.
- A "Login" button centered below the password field.
- At the bottom left, the text "Register | Reset Password" is displayed.

Gambar 5.5 Rancangan Antarmuka *User - Login*

Antarmuka gambar 5.5 digunakan untuk masuk ke halaman home INKOST. *Login* akan berhasil jika pengguna memasukkan *Username* dan *password*-nya dengan benar, jika salah atau ada tidak diisi, maka sistem akan menampilkan pesan *Login* gagal. Tombol *login* diklik saat *username* dan *password* disisikan dengan benar.

Kemudian kode php akan menunjukkan apakah *login* *valid* atau tidak ditentukan dengan apakah fungsi mengembalikan nilai atau *null*.

### 5.3.2.2 Registrasi



The image shows a registration form layout. It consists of five rows of labels and text boxes, followed by a 'Register' button. The labels are 'Email :', 'Nama :', 'Password :', 'Confirm Password :', and 'Tanggal Lahir :'. Each label is followed by a rectangular text box. The 'Register' button is a rectangular box with the text 'Register' inside, positioned below the 'Tanggal Lahir' text box.

Gambar 5.6 Rancangan antarmuka Registrasi

Antarmuka gambar 5.6 digunakan untuk registrasi pengguna. Pada bagian textbox diatas diisi untuk registrasi, Textbox email untuk email pelanggan digunakan untuk login, username, untuk mengetahui nama pengguna atau username, password digunakan untuk login tanpa ada orang lain mengetahui apa isi dari password, konfirmasi password digunakan untuk membandingkan apakah sama dengan password yang dimasukan diawal.

### 5.3.2.3 Reset Password



The image shows a user interface for resetting a password. It consists of a light gray rectangular box. On the left side, the text "Email :" is displayed. To the right of this text is a horizontal rectangular input field. Below the input field is a rectangular button with the word "Reset" centered on it.

Gambar 5.7 Rancangan Antarmuka *User* - Ubah Profil

Antarmuka gambar 5.7 digunakan untuk mereset password user, form diatas user disuruh memasukan email lalu sistem melakukan pengecekan apakah email terdaftar atau tidak. Jika email tidak terdaftar maka sistem menampilkan informasi gagal, jika berhasil informasi reset password akan ditampilkan oleh sistem.

### 5.3.2.4 Ubah Password



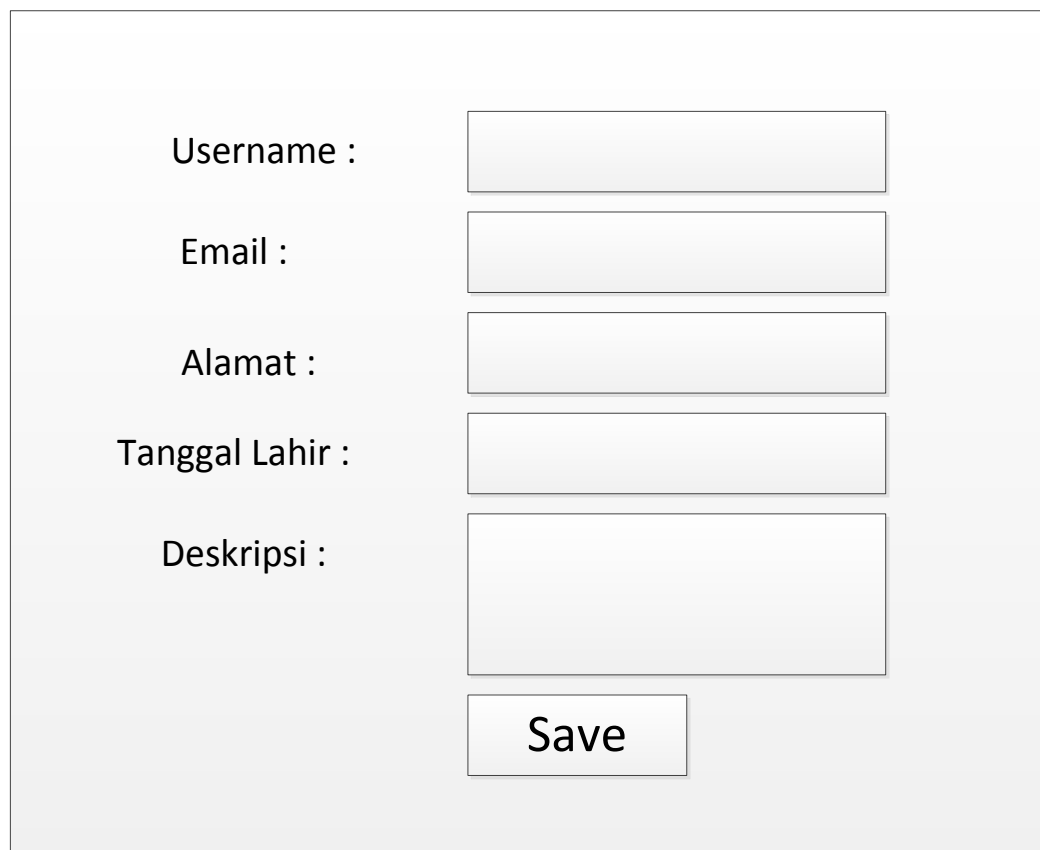
The image shows a user interface for changing a password. It consists of a light gray rectangular box. On the left side, the text "Password :" is displayed. To the right of this text is a horizontal rectangular input field. Below the first input field is the text "Confirm Password :". To the right of this text is another horizontal rectangular input field. Below the second input field is a rectangular button with the word "Save" centered on it.

Gambar 5.8 Rancangan Antarmuka *User* - Ubah Password

Antarmuka gambar 5.8 digunakan untuk mengubah *password user*. Pada *input password* lama diisi dengan *password lama*. Untuk mengubah *password*, maka masukan *password* baru pada kolom *input password* baru, dan masukan konfirmasi password untuk membandingkan password yang

di inputkan. kemudian klik tombol simpan untuk menyimpan perubahan.

#### 5.3.2.5 Edit Profile



The image shows a user interface for editing a profile. It consists of a light gray rectangular container. Inside, there are five rows of input fields, each preceded by a label: 'Username :', 'Email :', 'Alamat :', 'Tanggal Lahir :', and 'Deskripsi :'. The 'Username', 'Email', and 'Tanggal Lahir' fields are represented by single-line text boxes. The 'Alamat' field is a text area. The 'Deskripsi' field is a larger text area. Below these fields is a rectangular button labeled 'Save'.

Gambar 5.9 Rancangan Antarmuka *Edit Profile*

Antarmuka gambar 5.9 digunakan untuk mengedit profile, form menampilkan Username lama pada textbox username, email lama pada textbox email, no hp pada textbox no hp, alamat pada text area alamat, tanggal pada datatime picker dan gambar. Setelah mengubah data. Lalu menyimpan dengan menekan button submit.

### 5.3.3 Pengelolaan User

Nama	Email	Paket	status	view	
Ferry	Ferry@gmail.com	Paket Premium	aktif	view	delete
andre	andre@gmail.com	Paket Biasa	aktif	view	delete

Gambar 5.10 Rancangan Antarmuka User

Antarmuka gambar 5.10 menampilkan semua user ditampilkan dihalaman Admin. Disini admin bebas memilih untuk melakukan edit, delete, block dan view.

#### 5.3.3.1 Delete

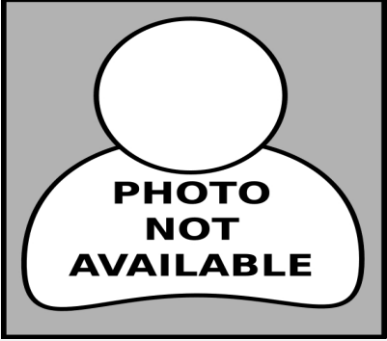
Nama	Email	Paket	status	view	
Ferry	Ferry@			ew	delete
andre	andre@			ew	delete

Apakah anda yakin menghapus user ini?

Gambar 5.11 Rancangan Antarmuka Admin - update status User

Antarmuka gambar 5.11 digunakan untuk mengupdate status user. Status user ini untuk membedakan aktif dan tidak aktif, user aktif jika user sudah membayar dan user tidak aktif jika sudah habis masa paket iklan yang dia beli atau belum membayar. Antarmuka ini hanya bisa diakses oleh admin

### 5.3.3.2 View User

Nama : sdk Pangkat : member Email : sdk@gmail.com Alamat : Deskripsi : Status : aktif Paket : Paket Premium Tanggal Lahir : 2014-03-20 Tanggal Habis Masa Aktif : 2015-03-10	 <input type="button" value="Kirim Pesan"/>
--	--

Gambar 5.12 Rancangan Antarmuka Admin - View User

Antarmuka ini digunakan untuk melihat detail data member secara lengkap. Antarmuka ini terdapat dihalaman pengelolaan user dan hanya bisa diakses oleh admin.

### 5.3.4 Pengelolaan Iklan

judul	paket	status	edit	<input type="button" value="Add iklan"/>
Kost delima	Paket premium	aktif	edit	
Kost mawar	Paket khusus	aktif	edit	

Gambar 5.13 Rancangan Antarmuka Iklan - Pengelolaan Iklan

Antarmuka gambar 5.13 digunakan untuk mengelola iklan. Pengelolaan iklan mempunyai fungsi tambah iklan, edit iklan, hapus iklan, view iklan, dan cari iklan. Ada beberapa fungsi yang hanya bisa digunakan oleh member dan nada beberapa fungsi yang hanya bisa digunakan oleh admin.

#### 5.3.4.1 Tambah Iklan

Judul iklan	<input type="text"/>
Harga Pertahun	<input type="text"/>
No tlp	<input type="text"/>
Alamat	<input type="text"/>
Daerah	<input type="text" value="Select"/>
Kampus Terdekat	<input type="text" value="Select"/>
Kelengkapan	<input type="text" value="Select"/>
AC	<input type="text" value="Select"/>
Kampus Terdekat	<input type="text" value="Select"/>
Waktu Sewa	<input type="text" value="Select"/>
Jenis Kost	<input type="text" value="Select"/>

## Map

latitude	longitude
----------	-----------

Gambar 5.14 Rancangan Antarmuka Iklan - Tambah Iklan

Antarmuka gambar 5.14 digunakan untuk menambah iklan. Fungsi ini hanya bisa digunakan oleh member. Dari form diatas terdapat judul iklan atau bisa diisi dengan nama kost. Keterangan bisa diisi dengan deskripsi singkat tentang keamanan dan sebagainya disekitar kost.



### 5.3.4.2 Edit Iklan

Judul iklan	Kost Delima
Harga Pertahun	2300000
No tlp	085729284581
Alamat	Delima 3A
Daerah	Depok
Kampus Terdekat	Uajy
Kelengkapan	Kosongan
AC	AC
Kamar Mandi	Dalam
Waktu Sewa	Tahunan
Jenis Kost	Wanita

Map

latitude	longitude
----------	-----------

Gambar 5.15 Rancangan Antarmuka Edit Iklan

Antarmuka gambar 5.15 digunakan untuk mengedit iklan. Setelah tampil semua data iklan lama didalam form, maka user mengubah data tersebut atau menambah data tersebut, lalu setelah diedit maka disimpan.

#### 5.3.4.3 Lihat Iklan

<h1>Gambar</h1>
<p>Harga : 10000000 Alamat : jln ns TB V1 no 11 A Jenis Kost : Wanita Isi Kost : Kosongan Kampus Terdekat : Universitas Ahmad Dahlan Waktu Sewa: Harian Nama Pemilik : sdk Kamar Mandi : Kamar Mandi Dalam Daerah : Bambanglipuro, Bantul No tlp : 910234989 Keterangan : testqweqwe</p>
<h1>Map</h1>

Gambar 5.16 Rancangan Antarmuka Lihat Iklan

Antarmuka gambar 5.16 digunakan untuk melihat iklan atau view iklan lebih detail.

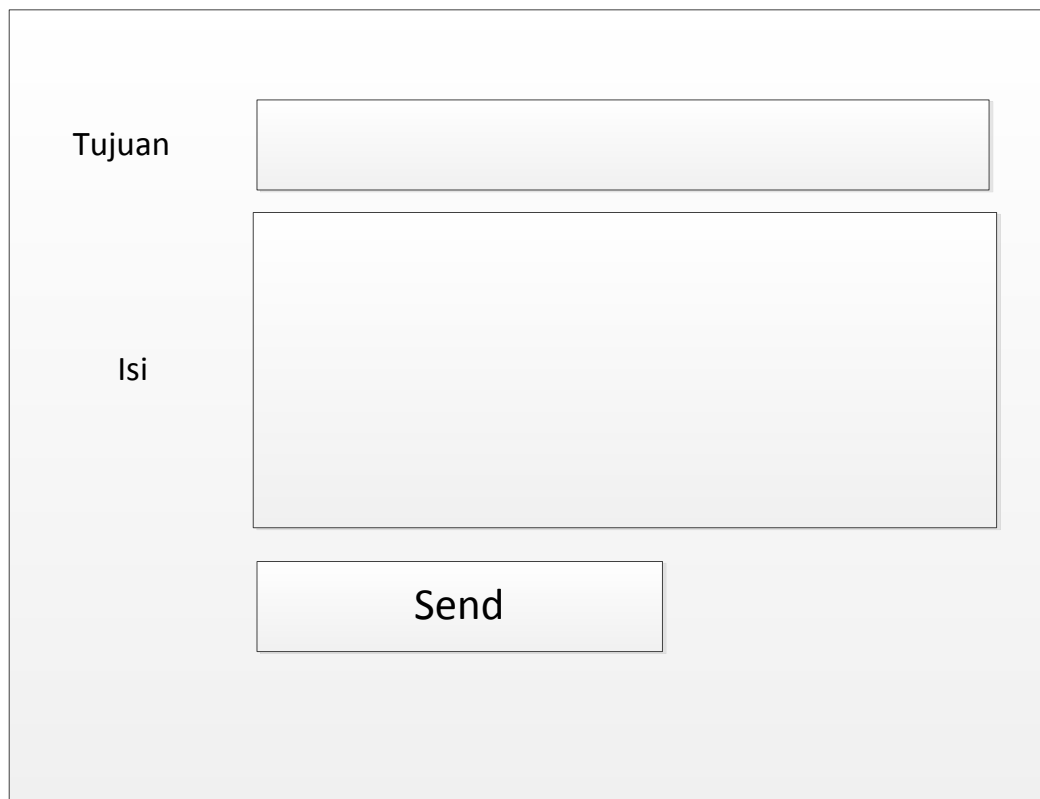
### 5.3.5 Pengelolaan Pesan

Inbox		Add pesan
pengirim	Isi	Delete
Ferry	Test ngirim pesan	Delete
andre	Test ngirim pesan	Delete
andi	Test ngirim pesan	Delete
OutBox		
penerima	Isi	Delete
Ferry	Test ngirim pesan	Delete
andre	Test ngirim pesan	Delete
andi	Test ngirim pesan	Delete

Gambar 5.17 Rancangan Antarmuka Pesan

Antarmuka Gambar 5.17 ini digunakan untuk mengirim pesan dari admin ke member atau Member ke admin, juga bisa mengirim pesan ke member lain.

#### 5.3.5.1 Tambah Pesan



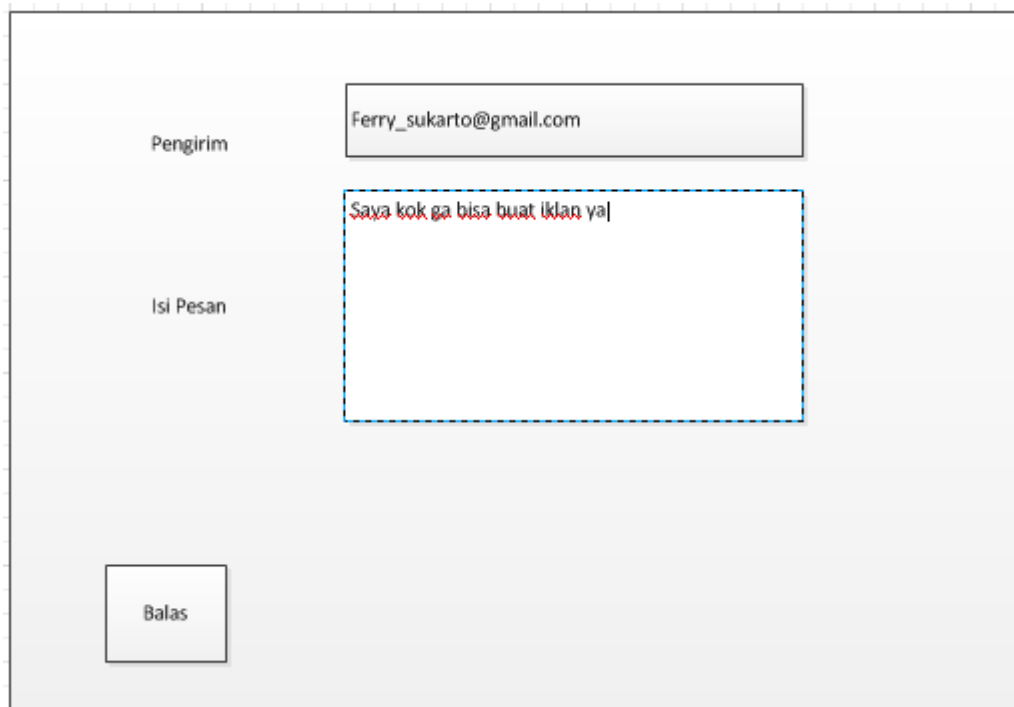
The image shows a user interface for adding a message. It consists of a light gray rectangular container. On the left side, there are two labels: 'Tujuan' (Destination) and 'Isi' (Content). To the right of 'Tujuan' is a single-line text input field. To the right of 'Isi' is a larger, multi-line text area. At the bottom center of the container is a button labeled 'Send'.

Gambar 5.18 Rancangan Antarmuka Tambah Pesan

Antarmuka gambar 5.18 digunakan untuk menambah pesan. Pada form ini terdapat textbox tujuan diisi dengan nama tujuan, isi pesan diisi dengan isi pesan yang akan disampaikan.

Button send digunakan untuk mengirim pesan.

### 5.3.5.2 Lihat Pesan



Gambar 5.19 Rancangan Antarmuka Tambah Pesan

Antarmuka gambar 5.19 digunakan untuk melihat pesan secara detail. Ketika user memilih untuk melihat dan memilih pesan tertentu untuk ditampilkan maka tampilah seperti gambar 5.19 diatas yang ditampilkan ada pengirim dan isi pesan. Button balas digunakan untuk membalas pesan tersebut.

### 5.3.5.3 Balas Pesan

The image shows a user interface for replying to a message. It consists of a light gray rectangular container. On the left side, there are two labels: 'Tujuan' and 'Isi'. To the right of 'Tujuan' is a text input field containing the name 'Ferry Sukarto'. To the right of 'Isi' is a larger text area containing the text 'Cara Buat Iklanya Gimana Neh Pak?'. Below these fields is a rectangular button labeled 'Kirim'.

Gambar 5.20 Rancangan Antarmuka balas pesan

Antarmuka gambar 5.20 digunakan untuk mengirim pesan atau membuat pesan baru. Yang bisa di tujukan kepada siapapun.

### 5.3.6 Pengelolaan Berita Pemberitahuan

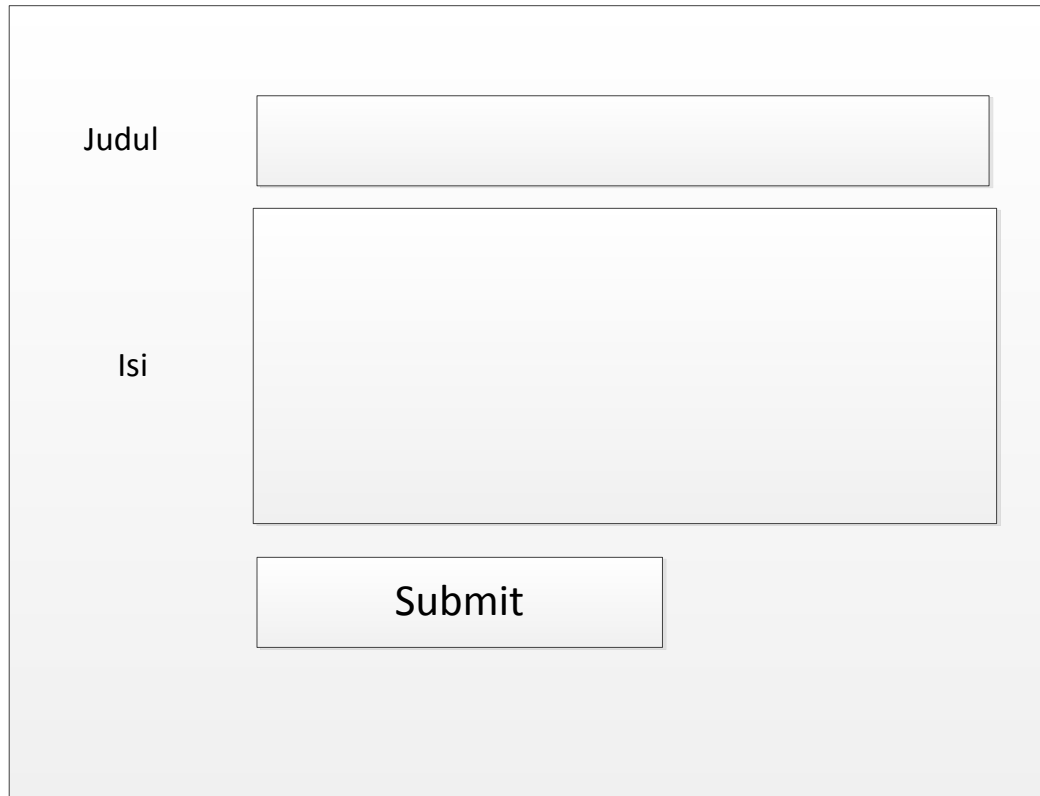
Judul	isi	Edit	Delete	View
Berita 1	isi	Edit	Delete	View
Berita 1	isi	Edit	Delete	View
Berita 1	isi	Edit	Delete	View

Gambar 5.21 Rancangan Antarmuka berita pemberitahuan

Antarmuka gambar 5.21 dibuat untuk berita pemberitahuan kepada seluruh user. Antarmuka ini hanya

bisa digunakan oleh admin, berita pemberitahuan tersebut akan tampil dihome member.

#### **5.3.6.1 Add Berita Pemberitahuan**



The image shows a web form for adding a notice. It consists of three main components: a text input field for the title, a larger text area for the content, and a submit button. The labels 'Judul' and 'Isi' are positioned to the left of their respective input fields. The 'Submit' button is centered below the content area.

Gambar 5.22 Rancangan Antarmuka berita pemberitahuan

Antarmuka gambar 5.22 digunakan untuk tambah berita pemberitahuan. Halaman ini hanya bisa diakses oleh admin. Di form ini ada textbox judul digunakan untuk diisi judul, isi di isi dengan isi berita yang akan dimuat, dan button submit digunakan untuk mempublish berita.

### 5.3.6.2 Hapus Berita

Judul	isi	Edit	Delete	View
Berita 1				
Berita 1				
Berita 1	isi	Edit	Delete	View

Apakah Anda Yakin Menghapus Berita Ini?

Gambar 5.23 Rancangan Antarmuka Hapus Berita

Antarmuka gambar 5.23 digunakan untuk menghapus berita, antarmuka ini hanya bisa digunakan oleh admin. Pemberitahuan bisa dihapus jika admin ingin menghapus pemberitahuan tersebut. Berita tersebut sebenarnya tidak dihapus hanya diubah status.



### 5.3.6.3 Edit Berita



The image shows a user interface for editing a news item. It consists of a light gray rectangular container with a thin black border. On the left side, there are two labels: 'Judul' and 'Isi'. To the right of 'Judul' is a rectangular text input field containing the text 'Pemberitahuan'. To the right of 'Isi' is a larger rectangular text area containing the text 'Bla bla bla'. At the bottom center of the container is a rectangular button labeled 'Submit'.

Gambar 5.24 Rancangan Antarmuka edit berita

Antarmuka gambar 5.24 digunakan untuk mengedit berita pemberitahuan, pada form terdapat textbox judul berita, isi berita dan gambar. Dan button simpan untuk menyimpan data yang telah diubah dari form diatas.

### 5.3.7 Konfirmasi Pembayaran

#### 5.3.7.1 Member Konfirmasi

Rek Tujuan	<input type="text" value="103-000-2349234234 (Bpk Ferry Sukarto)"/>
Jumlah uang	<input type="text" value="100000"/>
Pilihan paket	<input type="text" value="Paket Khusus"/>
Rek Pengirim	<input type="text" value="109-928-239489283"/>
Nama Direk	<input type="text" value="Deni STH"/>
Tgl Transfer	<input type="text" value="01/04/2014"/>
Keterangan	<input type="text" value="Sudah Saya Transfer Pak"/>
	<input type="button" value="Submit"/> <input type="button" value="Reset"/>

Gambar 5.25 Rancangan Antarmuka Member konfirmasi

Antarmuka gambar 5.25 digunakan oleh member untuk melakukan konfirmasi pembayaran. Form ini ada combobox rekening tujuan, textbox jumlah uang yang ditransfer, pilihan paket, rek pengirim, nama di rek, tanggal transfer dan keterangan. Ada dua button di gambar 5.20 ini yaitu submit dan reset.

### 5.3.7.2 Admin Konfirmasi

Nama	Rek Tujuan	Pilihan paket	Jumlah uang	Konfirm
Ferry Sukarto	103-000-2234233	Paket Khusus	100000	<input type="button" value="Konfirm"/>

Gambar 5.26 Rancangan Antarmuka Member konfirmasi

Antarmuka gambar 5.26 digunakan oleh admin untuk konfirmasi pembayaran dari member. Data yang ditampilkan yaitu nama tujuan, rek tujuan, pilihan paket, jumlah uang dan konfirm. Button konfirm itu untuk melakukan konfirmasi oleh admin.

### 5.3.8 Pengelolaan Kritik dan Saran

#### 5.3.8.1 Tambah Kritik dan saran



The image shows a web form for adding feedback and suggestions. It consists of three input fields and two buttons. The first field is labeled 'Nama' and contains the text 'Ferry Sukarto'. The second field is labeled 'Email' and contains the text 'Ferry\_Sukarto@yahoo.com'. The third field is labeled 'Isi Pesan' and contains the text 'Bro. itu tampilan halaman utama slinye jelek amat... ubah dong'. Below the input fields are two buttons: 'Kirim' and 'Cancel'.

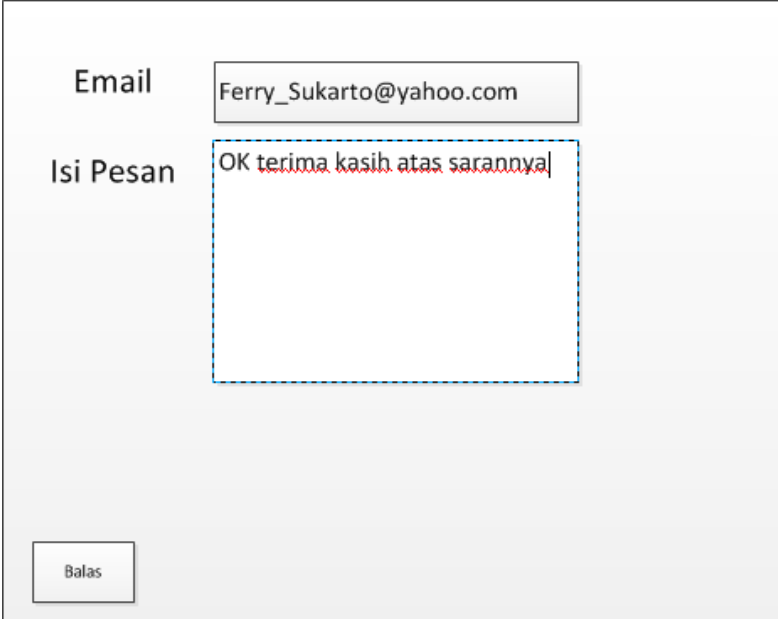
Nama	Ferry Sukarto
Email	Ferry_Sukarto@yahoo.com
Isi Pesan	Bro. itu tampilan halaman utama slinye jelek amat... ubah dong

Kirim Cancel

Gambar 5.27 Rancangan Antarmuka tambah kritik dan saran

Antarmuka gambar 5.27 digunakan oleh pengunjung untuk melakukan tambah kritik dan saran. Form ini terdiri dari textbox email, nama, dan isi pesan yang akan disampaikan.

### 5.3.8.2 Lihat Kritik dan saran



The image shows a user interface for viewing an email. On the left, there are two labels: "Email" and "Isi Pesan". To the right of "Email" is a text box containing the address "Ferry\_Sukarto@yahoo.com". To the right of "Isi Pesan" is a larger text area containing the message "OK terima kasih atas sarannya". At the bottom left of the interface is a button labeled "Balas".

Gambar 5.28 Rancangan Antarmuka Lihat kritik dan saran

Antarmuka gambar 5.28 digunakan untuk melihat kritik dan saran secara detail. Di form gambar ini ditampilkan email dan isi pesan yang disampaikan, lalu ada button balas digunakan untuk membalas pesan ini.

### 5.3.8.3 Balas Kritik dan Saran



The image shows a web form for replying to an email. It consists of three main components: a recipient field, a message body field, and a submit button. The recipient field is labeled 'Dikirim ke' and contains the email address 'Ferry\_Sukarto@yahoo.com'. The message body field is labeled 'Isi Pesan' and contains the text 'Terima kasih atas sarannya|'. Below the message body field is a button labeled 'Kirim'.

Gambar 5.29 Rancangan Antarmuka balas kritik dan saran  
Antarmuka gambar 5.29 digunakan untuk membalas kritik dan saran. Pada form ini terdapat tujuan pengiriman akan terisi sendiri oleh email yang telah di pilih untuk dibalas, isi pesan diisi dengan pesan yang akan disampaikan. Button kirim digunakan untuk mengirim pesan.

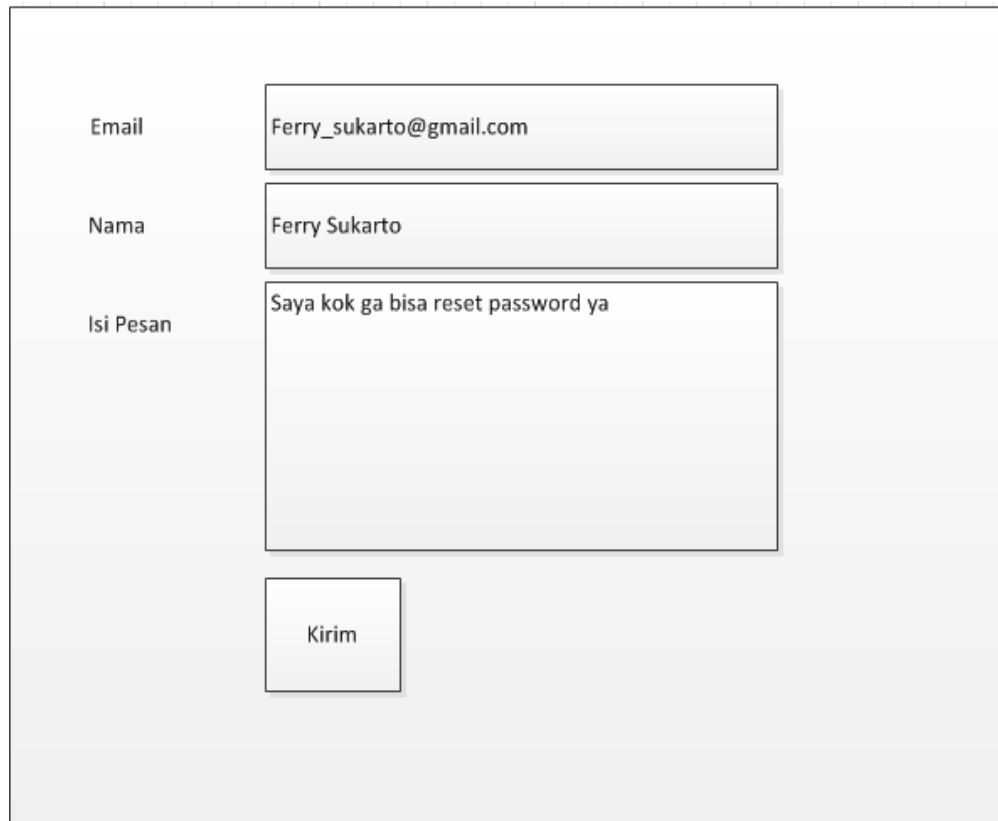
#### 5.3.8.4 Hapus Kritik dan Saran



Gambar 5.30 Rancangan Antarmuka hapus kritik dan saran  
Antarmuka gambar 5.30 digunakan untuk menghapus kritik dan saran dari pengunjung. User memilih data kritik saran yang akan kita hapus, setelah itu sistem akan memberi konfirmasi apakah yakin akan menghapus data yang sudah dipilih.

### 5.3.9 Pengelolaan ContactUs

#### 5.3.9.1 Tambah ContactUs



The image shows a contact form with the following fields and content:

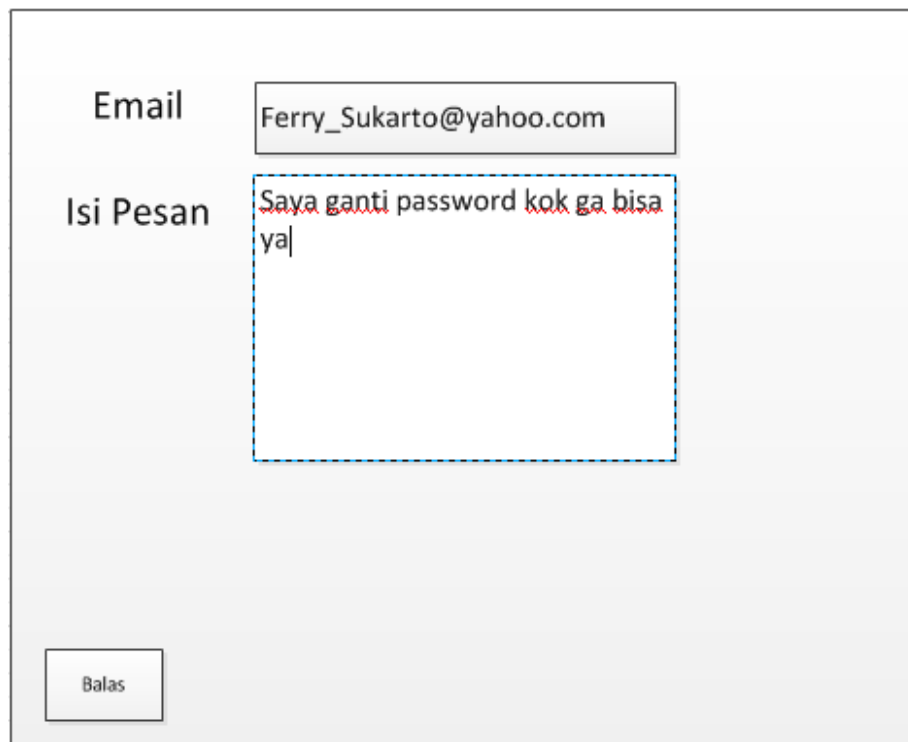
Email	Ferry_sukarto@gmail.com
Nama	Ferry Sukarto
Isi Pesan	Saya kok ga bisa reset password ya
<input type="button" value="Kirim"/>	

Gambar 5.31 Rancangan Antarmuka tambah contactus

Antarmuka gambar 5.31 digunakan untuk menambah contactus. Di form ini terdapat nama, email dan isi pesan dan button kirim digunakan untuk melakukan pengiriman.



### 5.3.9.2 Lihat ContactUs



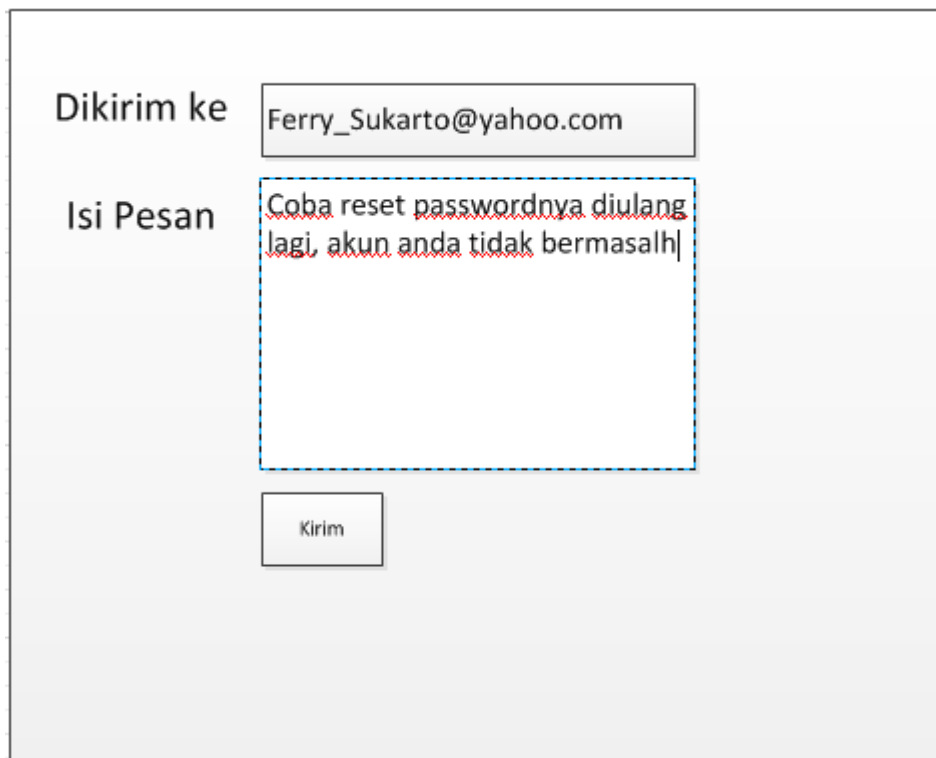
The image shows a contact form with the following elements:

- Email:** A text input field containing the email address `Ferry_Sukarto@yahoo.com`.
- Isi Pesan:** A text area containing the message `Saya ganti password kok ga bisa ya|`. The text is underlined in red.
- Balas:** A button labeled "Balas" located at the bottom left of the form.

Gambar 5.32 Rancangan Antarmuka lihat contactus

Antarmuka gambar 5.32 digunakan untuk melihat contactus, disini ditampilkan email dan isi pesan. Dan button balas digunakn untuk mengirim pesan. Gambar hanya dapat diakses oleh admin.

### 5.3.9.3 Balas ContactUs



The image shows a web form for replying to a contact. It features two main input areas: a text box for the recipient's email address and a larger text area for the message content. A 'Kirim' button is positioned below the message area.

Dikirim ke	Ferry_Sukarto@yahoo.com
Isi Pesan	Coba reset passwordnya diulang lagi, akun anda tidak bermasalh
	Kirim

Gambar 5.33 Rancangan Antarmuka Member konfirmasi

Antarmuka 5.33 digunakan untuk melakukan balas contact use. Di form ini terdapat textbox tujuan akan terisi sendiri oleh email dari contactus yang dipilih untuk di balas. Sedangkan isi pesan diisi oleh pesan yang akan di sampaikan kepada pengirim.

#### 5.3.9.4 Hapus ContactUs



Gambar 5.34 Rancangan Antarmuka hapus contactus

Antarmuka gambar 5.34 digunakan untuk menghapus contactus, user memilih untuk melakukan penghapusan contactus, user memilih contactus yang ingin dihapus, lalu sistem memberikan informasi apakah yakin akan menghapus pesan ini atau tidak. Antarmuka hanya dapat diakses oleh admin.