

6.1 Kesimpulan

Berdasarkan uraian dan pengujian yang telah dilakukan, kesimpulan dari pembangunan sistem informasi Alta Tour dan Event berbasis web adalah sebagai berikut:

1. Sistem informasi Alta Tour dan Event berbasis web telah berhasil dibangun dan dapat menjadi alat pendukung pengelolaan proses bisnis yang dijalankan oleh Alta Tour dan Event.
2. Memudahkan Alta dalam mengakses data dan informasi yang dibutuhkan.

6.2 Saran

Walaupun sistem informasi alta tour dan event berbasis web telah berhasil dibangun, namun masih terdapat kekurangan yang dilakukan penulis. Oleh sebab itu, penulis memberikan saran:

Sistem informasi ini diharapkan dapat dikembangkan lebih lanjut terkait dengan pengelolaan transaksi yang belum bisa memberikan tanggapan langsung kepada pelanggan, sehingga pengaksesan data dan informasi serta pengelolaan proses bisnis yang dijalankan oleh Alta dapat berjalan secara maksimal dan efisien.

DAFTAR PUSTAKA

- Agustian, V. T. (2016). *SISTEM INFORMASI RESERVASI TOUR & TRAVEL BERBASIS WEB PADA A-RAYA TOUR & TRAVEL JEMBER.*
- Al Fatta, H. (2009). *Analisis dan Perancangan Sistem Informasi untuk keunggulan bersaing Perusahaan dan Organisasi Modern.*
- Alfiaani, A. M. (2012). *SISTEM APLIKASI MANAJEMEN BIRO WISATA TOUR & TRAVEL PADA PT KARSA MANDIRI KARYA BERBASIS DESKTOP.*
- Aryanto, A. (2013). *PEMBANGUNAN SISTEM PENJUALAN ONLINE PADA TOKO INDAH JAYA FURNITURE SURAKARTA.*
- Handita, D. B., Umar, & Fadlillah, U. (2012). *Jurnal Emitor. SISTEM INFORMASI PENJUALAN BERBASIS WEB PADA USANTEX.*
- Handoyo, R. (2013). *Pembangunan Aplikasi Panduan Transportasi Wisata Di Yogyakarta Berbasis Lokasi Pada Perangkat Mobile.*
- Maulidin, R. N. (2015). *SISTEM INFORMASI PAKET WISATA PADA CV.KHANS TOUR BERBASIS WEB.*
- Solichin, A. (2012). *Permodelan Arsitektur Teknologi Informasi berbasis Cloud Computing untuk Institusi Perguruan Tinggi di Indonesia.*

LAMPIRAN

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SIATE

(Sistem Informasi Alta Tour Dan Event Berbasis Web)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Albertus Bagawan Abdi / 100706172

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL-SIATE		1/35
		Revisi		

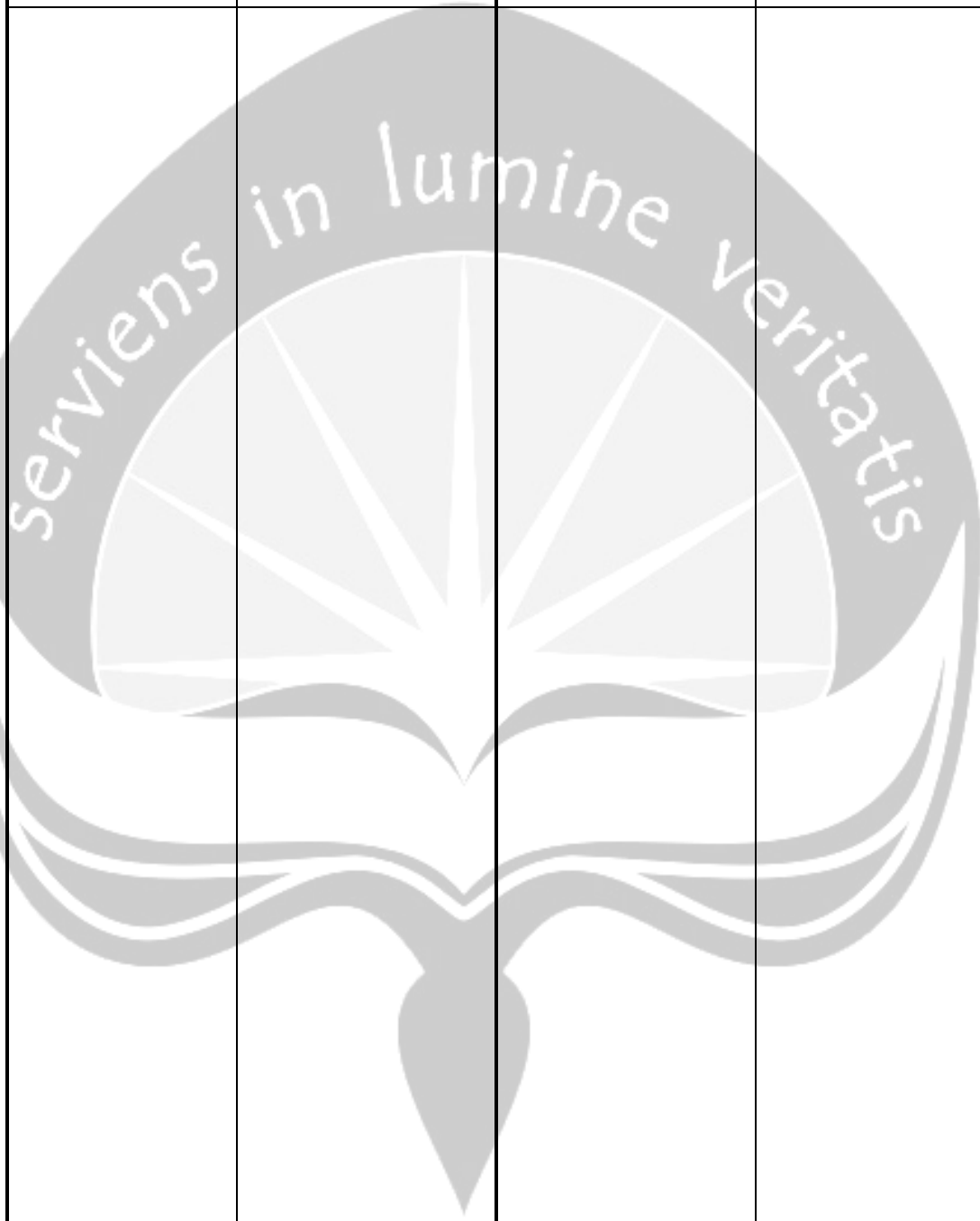
DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui 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	9
1.5	Deskripsi umum (Overview)	9
2	Deskripsi Kebutuhan	9
2.1	Perspektif produk	9
2.2	Fungsi Produk	11
2.3	Karakteristik Pengguna	15
2.4	Batasan-batasan	16
2.5	Asumsi dan Ketergantungan	16
3	Kebutuhan khusus	16
3.1	Kebutuhan antarmuka eksternal	16
3.2	Kebutuhan fungsionalitas Perangkat Lunak	19
4	Spesifikasi Rinci Kebutuhan	19
4.1	Spesifikasi Kebutuhan Fungsionalitas	19
5	Entity Relationship Diagram (ERD)	35

Daftar Gambar

1. Arsitektur Perangkat Lunak SIATE	10
2. Use Case Diagram	20
3. ERD (Entity Relationship Diagram).....	35



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIATE (Sistem Informasi Alta Tour Dan Even Berbasis Web) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain, perangkat lunak dan perangkat keras, dan pengguna), dan atribut (fitur-fitur tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SIATE ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak SIATE dikembangkan dengan tujuan untuk:

1. Menangani Pengelolaan kategori paket wisata.
2. Menangani Pengelolaan Tempat wisata.
3. Menangani Pengelolaan Paket Wisata.
4. Menangani Pelaporan transaksi pemesanan paket wisata.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIATE-XX	Kode yang merepresentasikan kebutuhan pada SIATE (Sistem Informasi Alta Tour Dan Event Berbasis Web) dimana XXX merupakan nomor fungsi produk.
SIATE	Perangkat lunak Alta Tour Dan Event berbasis web.
Admin	Pengguna sistem yang bertugas mengelola data paket, tempat, ref person, kategori dan kategori wisata.

Super Admin	Pengguna sistem yang bertugas untuk melakukan penambahan admin.
Web	<i>Website</i> atau yang biasa disebut <i>Web</i> adalah kumpulan halaman yang memiliki informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman.
Basis Data	Basis Data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi.
Server	Komputer yang menyediakan sumber data bagi <i>Client</i> yang terhubung melalui jaringan.
Code Igniter	Aplikasi open source yang berupa framework dengan model MVC (<i>Model, View, Controller</i>) untuk membangun <i>website</i> dinamis dengan menggunakan PHP.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Spesifikasi Kebutuhan Perangkat Lunak (SKPL) Sistem Informasi Apotek Pasti oleh Arik Eko Prasetyo / 110706608, Universitas Atma Jaya Yogyakarta.

1.5 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 SIATE yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SIATE tersebut.

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

2 Deskripsi Kebutuhan

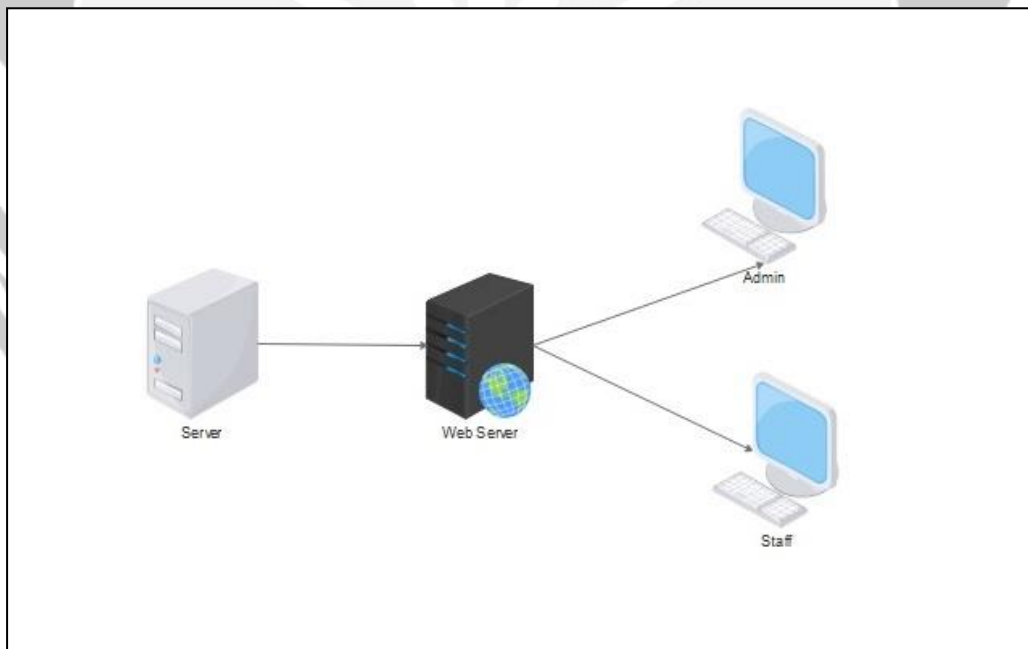
2.1 Perspektif produk

SIATE merupakan perangkat lunak berbasis web yang dikembangkan untuk membantu sebuah usaha yang bergerak dalam bidang jasa tour dan event. Sistem ini membantu

Program Studi Teknik Informatika	SKPL – SIATE	9/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Alta tour dan event untuk mengelola proses bisnis yang dijalankan. Sistem ini menangani pengelolaan data pengguna, pengelolaan tempat wisata, pengelolaan kategori paket wisata, pengelolaan paket wisata, pengelolaan ref person, dan pengelolaan transaksi pemesanan.

Perangkat lunak SIATE ini berjalan pada web browser apapun, dan dibuat menggunakan bahasa pemrograman HTML5, PHP 5.5.9 dengan framework Code Igniter 2.2.0, Javascript dengan framework JQuery 1.9, CSS dengan framework Bootstrap 3.3.0. Sedangkan untuk lingkungan pemrogramannya menggunakan Notepad++. Untuk basis data dipergunakan MySQL 5.6.16 sebagai server basis data dan Apache 2.4.7 sebagai web server.



Gambar 1. Arsitektur Perangkat lunak SIATE

Dapat dilihat pada Gambar 1, yaitu seluruh pengguna sistem dapat mengakses data yang diperlukan melalui web server dan mengacu pada basis data yang sama.

2.2 Fungsi Produk

Fungsi produk perangkat lunak SIATE adalah sebagai berikut:

1. Fungsi *Login* (**SKPL-SIATE-01**)

Merupakan fungsi yang digunakan oleh super admin dan admin untuk dapat masuk ke dalam sistem yang digunakan untuk mengelola data.

2. Fungsi *Pengelolaan Tempat Wisata* (**SKPL-SIATE-02**)

Merupakan fungsi yang digunakan oleh Admin untuk mengelola tempat wisata.

Fungsi *Pengelolaan Tempat Wisata* mencakup :

a. Fungsi *Tambah Tempat Wisata* (**SKPL-SIATE-02-1**)

Merupakan fungsi yang digunakan untuk menambahkan tempat wisata yang baru.

b. Fungsi *Ubah Tempat Wisata* (**SKPL-SIATE-02-2**).

Merupakan fungsi yang digunakan untuk mengubah tempat wisata.

c. Fungsi *Hapus Tempat Wisata* (**SKPL-SIATE-02-3**)

Merupakan fungsi yang digunakan untuk menghapus tempat wisata.

d. Fungsi *Tampil Tempat Wisata* (**SKPL-SIATE-02-4**)

Merupakan fungsi yang digunakan untuk menampilkan tempat wisata.

3. Fungsi *Pengelolaan Data User* (**SKPL-SIATE-03**)

Merupakan fungsi yang digunakan oleh Super Admin untuk mengelola tempat wisata.

Fungsi pengelolaan data pengguna meliputi:

Program Studi Teknik Informatika	SKPL – SIATE	11/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

a. Fungsi *Tambah Data Pengguna* (**SKPL-SIATE-03-1**)
Merupakan fungsi yang digunakan untuk menambahkan data pengguna yang baru.

b. Fungsi *Ubah Data Pengguna* (**SKPL-SIATE-03-2**)
Merupakan fungsi yang digunakan untuk mengubah data pengguna bila terjadi kesalahan data.

c. Fungsi *Hapus Tempat wisata* (**SKPL-SIATE-03-3**)
Merupakan fungsi yang digunakan untuk menghapus tempat wisata.

d. Fungsi *Tampil Tempat wisata* (**SKPL-SIATE-03-4**)
Merupakan fungsi yang digunakan untuk menampilkan tempat wisata.

4. Fungsi *Pengelolaan Kategori Paket* (**SKPL-SIATE-04**)

Merupakan fungsi yang digunakan oleh Pengguna yang memiliki Role sebagai Admin dan Super Admin. Fungsi ini digunakan untuk melakukan pengelolaan kategori paket wisata. Fungsi kategori paket wisata meliputi:

a. Fungsi *Tambah Paket* (**SKPL-SIATE-04-1**)

Fungsi ini merupakan fungsi yang digunakan untuk menambahkan daftar kategori paket wisata yang baru.

b. Fungsi *Ubah Kategori Paket* (**SKPL-SIATE-04-2**)

Fungsi ini digunakan untuk mengubah kategori paket wisata jika terjadi kesalahan pada data.

c. Fungsi *Hapus Kategori Paket* (**SKPL-SIATE-04-3**)

Fungsi ini berguna untuk menghapus data kategori paket jika data tidak diperlukan maupun

Program Studi Teknik Informatika	SKPL – SIATE	12/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

jika data yang dimasukkan tidak sesuai keinginan.

d. Fungsi *Tampil Kategori Paket* (**SKPL-SIATE-04-4**)

Fungsi ini digunakan untuk menampilkan data Kategori Paket dalam bentuk tabel.

5. Fungsi *Pengelolaan Paket* (**SKPL-SIATE-05**)

Merupakan fungsi yang digunakan oleh pengguna yang memiliki role sebagai Admin dan Super Admin. Fungsi ini digunakan untuk mengelola data-data paket wisata. Fungsi pengelolaan paket meliputi:

a. Fungsi *Tambah Paket* (**SKPL-SIATE-05-1**)

Fungsi ini merupakan fungsi yang digunakan untuk menambahkan paket.

b. Fungsi *Ubah Paket* (**SKPL-SIATE-05-2**)

Fungsi ini digunakan untuk mengubah paket wisata jika terjadi kesalahan pada data.

c. Fungsi *Hapus Paket* (**SKPL-SIATE-05-3**)

Fungsi ini berguna untuk menghapus paket jika data tidak dipelukan lagi maupun jika data yang dimasukkan tidak sesuai keinginan.

d. Fungsi *Tampil Paket* (**SKPL-SIATE-05-4**)

Fungsi ini digunakan untuk menampilkan data paket dalam bentuk tabel.

6. Fungsi *Pengelolaan Harga* (**SKPL-SIATE-06**)

Merupakan fungsi yang digunakan oleh pengguna yang memiliki role sebagai Admin dan Super Admin.

Fungsi ini digunakan untuk mengelola data-data Harga. Fungsi pengelolaan paket meliputi:

e. Fungsi *Tambah Harga* (**SKPL-SIATE-06-1**)

Fungsi ini merupakan fungsi yang digunakan untuk menambahkan harga.

f. Fungsi *Ubah Harga* (**SKPL-SIATE-06-2**)

Fungsi ini digunakan untuk mengubah harga jika terjadi kesalahan pada data.

g. Fungsi *Hapus Harga* (**SKPL-SIATE-06-3**)

Fungsi ini berguna untuk menghapus harga jika data tidak diperlukan lagi maupun jika data yang dimasukkan tidak sesuai keinginan.

h. Fungsi *Tampil Harga* (**SKPL-SIATE-06-4**)

Fungsi ini digunakan untuk menampilkan data harga dalam bentuk tabel.

7. Fungsi *Pengelolaan Ref Person* (**SKPL-SIATE-07**)

Merupakan fungsi yang digunakan oleh Pengguna yang memiliki Role sebagai Admin dan Super Admin. Fungsi ini digunakan untuk melakukan pengelolaan Ref person. Fungsi pengelolaan ref person meliputi:

a. Fungsi *Tambah Ref Peron* (**SKPL-SIATE-07-1**)

Fungsi ini merupakan fungsi yang digunakan untuk menambahkan jumlah orang.

b. Fungsi *Ubah Ref Person* (**SKPL-SIATE-07-2**)

Fungsi ini digunakan untuk jumlah orang jika terjadi kesalahan pada data.

c. Fungsi *Hapus Ref Person* (**SKPL-SIATE-07-3**)

Fungsi ini berguna untuk menghapus jumlah orang jika data tidak diperlukan lagi maupun jika data yang dimasukkan tidak sesuai keinginan.

d. Fungsi *Tampil Ref Person* (**SKPL-SIATE-07-4**)

Fungsi ini digunakan untuk menampilkan data jumlah orang dalam bentuk tabel.

8. Fungsi *Pengelolaan Transaksi* (**SKPL-SIATE-08**)

Merupakan fungsi yang digunakan oleh Pengguna yang memiliki Role sebagai Admin dan Super Admin. Fungsi ini digunakan untuk menampilkan data *booking*. Fungsi pengelolaan transaksi meliputi:

a. Fungsi *Tampil Data Transaksi* (**SKPL-SIATE-08-1**)

Fungsi ini digunakan untuk menampilkan data *booking* dalam bentuk tabel..

9. Fungsi *Logout* (**SKPL-SIATE-9**)

Merupakan fungsi yang digunakan oleh pengguna untuk keluar dari sistem.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SIATE adalah sebagai berikut:

1. Mengerti pengoperasian komputer.
2. Memahami penggunaan website.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak SIATE tersebut adalah:

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak SIATE.

2. Keterbatasan perangkat keras

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

2.5 Asumsi dan Ketergantungan

Sistem ini dapat berjalan pada browser dengan versi terbaru yang mendukung penuh penggunaan bahasa pemrograman HTML5, CSS 3 dan Javascript dengan Framework JQuery 1.9.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak SIATE meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam halaman website.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam aplikasi SIATE adalah:

Program Studi Teknik Informatika	SKPL – SIATE	16/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Perangkat PC/Laptop.
2. Perangkat Server Basis Data.

3.1.3 Antarmuka perangkat lunak

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

1. Nama : Apache ver 2.4.7.
 Sumber : Apache Software Foundation.
 Sebagai sistem menyediakan layanan akses kepada pengguna melalui protocol komunikasi.
2. Nama : MySQL ver 5.6.16.
 Sumber : Oracle Corporation.
 Sebagai manajemen sistem basis data yang digunakan untuk menyimpan data dari sisi server.
3. Nama : Code Igniter.
 Sumber : EllisLab, Inc.
 Sebagai framework bahasa pemrograman PHP untuk menjalankan aplikasi.
4. Nama : IE/Firefox/Chrome/dll
 Sumber : Microsoft/Mozilla/Google/dll.
 Sebagai perangkat lunak yang berguna untuk menampilkan berbagai dokumen yang tersedia pada web server.
5. Nama : Windows XP/7/8/Ubuntu/dll
 Sumber : Microsoft/Cononical/dll.
 Sebagai sistem operasi yang digunakan.
6. Nama : Notepad++
 Sumber : Don Ho.
 Sebagai aplikasi untuk edito program.

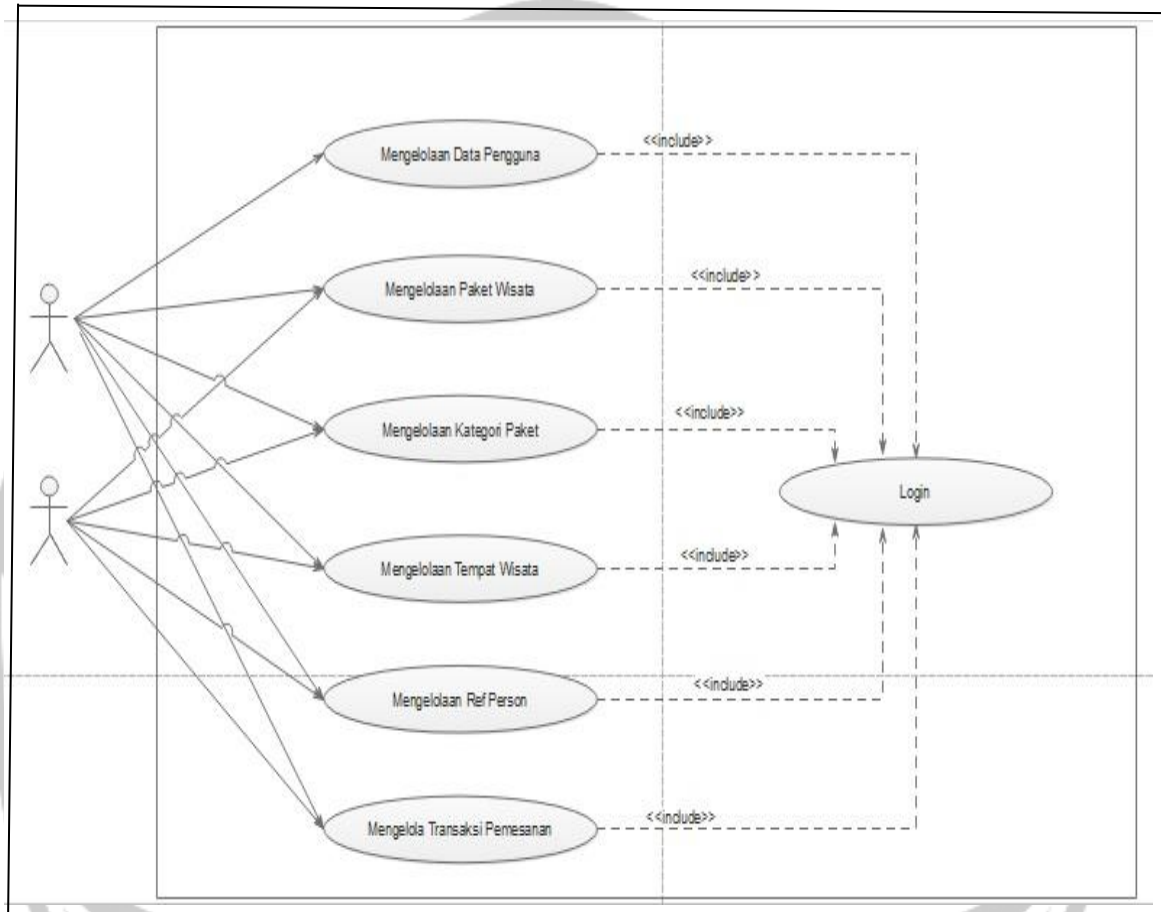
3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIATE menggunakan protocol HTTP.



3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram

4 Spesifikasi Rinci Kebutuhan

4.1 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 email dari user dan password yang berupa rangkaian karakter.

Program Studi Teknik Informatika	SKPL – SIATE	19/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

1. Super admin.
2. Admin.

3. Supporting Actor

None

4. Basic Flow

1. Use case ini dimulai ketika aktor memilih untuk melakukan login.
2. Sistem menampilkan antarmuka untuk login.
3. Actor memasukkan email dan password lalu menekan tombol login.
4. Sistem memeriksa email dan password yang dimasukkan aktor.
E-1. Username atau password tidak sesuai.
5. Sistem memberikan akses ke aktor.
6. Use case ini selesai.

5. Alternative Flow

None

6. Error Flow

E-1 email atau password tidak sesuai.

1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai.
2. Kembali pada Basic Flow langkah ke 3.

7. PreCondition

Aktor masih berada diluar sistem.

8. PostCondition

Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

4.1.2 Use case Spesification : Pengelolaan Data Pengguna

Program Studi Teknik Informatika	SKPL – SIATE	20/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data dari pengguna sistem. Aktor dapat melakukan tambah data pengguna, ubah data pengguna, hapus data pengguna, tampil data pengguna, dan cari data pengguna.

2. Primary Actor

1. Super Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pengguna.
2. Sistem menampilkan antarmuka pengelolaan data pengguna.
3. Sistem memberikan pilihan untuk melakukan tambah data pengguna, ubah data pengguna, hapus data pengguna, cari data pengguna dan tampil data pengguna.
4. Aktor memilih untuk melakukan tambah data pengguna.
 - A-1 Aktor memilih untuk melakukan ubah data pengguna.
 - A-2 Aktor memilih untuk melakukan tampil data pengguna.
 - A-3 Aktor memilih untuk melakukan hapus data pengguna.
5. Aktor memasukkan data pengguna.
6. Sistem memeriksa data pengguna yang telah dimasukkan.

E-1 Data pengguna yang dimasukkan salah atau kurang lengkap.

7. Sistem menyimpan data pengguna kedalam basis data.

8. Use Case selesai

5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data pengguna.

1. Aktor memilih data pengguna yang akan diubah.

2. Aktor mengubah data pengguna.

3. Sistem melakukan pemeriksaan terhadap data pengguna yang telah diubah.

E-2 Data pengguna yang telah diubah salah atau kurang lengkap.

4. Sistem menyimpan data pengguna yang telah diubah ke dalam basis data.

5. Berlanjut ke Basic Flow langkah ke 8.

A-2 Aktor memilih untuk melakukan tampil data pengguna.

1. Sistem menampilkan data atau profil pengguna.

2. Berlanjut pada Basic Flow langkah ke 8.

A-3 Aktor memilih untuk melakukan hapus data pengguna.

1. Aktor memilih data pengguna yang akan dihapus.

2. Sistem memeriksa data pengguna yang akan dihapus.

3. Sistem menghapus data pengguna dari basis data.

4. Kembali pada Basic Flow langkah ke 8.

6. Error Flow

Program Studi Teknik Informatika	SKPL – SIATE	22/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-1 Data pengguna yang dimasukkan salah atau kurang lengkap.

1. Sistem memberikan peringatan bahwa data yang dimasukkan salah.
2. Kembali ke Basic Flow Langkah ke 5.

E-2 Data pengguna yang telah diubah salah atau kurang lengkap.

1. Sistem memberikan peringatan bahwa data yang diubah salah atau kurang lengkap.
2. Kembali ke Alternative Flow A-1 Langkah ke 2.

7. PreConditions

1. Use case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

Sistem memperbaharui basis data.

4.1.3 Use case Spesification : Pengelolaan Tempat Wisata

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data-data tempat wisata. Aktor dapat melakukan tambah tempat wisata, ubah tempat wisata, hapus tempat wisata, tampil tempat wisata.

2. Primary Actor

1. Admin dan Super Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mengelola tempat wisata.

Program Studi Teknik Informatika	SKPL – SIATE	23/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Sistem menampilkan antarmuka pengelolaan tempat wisata.
3. Sistem memberikan pilihan untuk melakukan tambah tempat wisata, ubah tempat wisata, hapus tempat wisata, dan cari tempat wisata.
4. Aktor memilih untuk melakukan tambah tempat wisata.
 - A-1 Aktor memilih untuk melakukan hapus tempat wisata.
 - A-2 Aktor memilih untuk melakukan ubah tempat wisata.
5. Aktor memasukkan tempat wisata.
6. Sistem menyimpan tempat wisata ke dalam basis data.
7. Use Case selesai.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan hapus tempat wisata.
 1. Aktor memilih tempat wisata yang akan dihapus.
 2. Sistem menghapus tempat wisata dari basis data.
 3. Berlanjut ke Basic Flow langkah ke 8.
- A-2 Aktor memilih untuk melakukan ubah tempat wisata.
 1. Aktor memilih tempat wisata yang ingin diubah.
 2. Sistem menampilkan antarmuka ubah tempat wisata.
 3. Aktor mengubah tempat wisata.
 4. Sistem menyimpan perubahan data kedalam basis data.
 5. Berlanjut pada Basic Flow langkah 7.

Program Studi Teknik Informatika	SKPL – SIATE	24/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Error Flow

1. None

7. PreConditions

1. Use Case Login sudah dilakukan
2. Aktor telah memasuki sistem

8. PostConditions

Tempat wisata dalam basis data terbaharui.

4.1.4 Use case Spesification : Pengelolaan Kategori Paket

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola kategori paket. Aktor dapat melakukan tambah kategori paket, ubah kategori paket, hapus kategori paket, dan tampil detail kategori paket.

2. Primary Actor

1. Operator.

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan kategori paket.
2. Sistem menampilkan antarmuka pengelolaan kategori paket dan kategori paket.
3. Sistem memberikan pilihan untuk melakukan tambah kategori paket, ubah kategori paket, hapus kategori paket, tampil detail kategori paket, dan cari kategori paket.
4. Aktor memilih untuk melakukan tambah kategori paket.

A-1 Aktor memilih untuk melakukan tampil detail kategori paket.

A-2 Aktor Memilih untuk melakukan ubah kategori paket.

A-3 Aktor memilih untuk melakukan hapus kategori paket.

5. Sistem menampilkan antarmuka untuk tambah kategori paket.

6. Aktor memasukkan kategori paket.

7. Sistem memeriksa data yang dimasukkan aktor.

E-1 Masukan Aktor tidak lengkap.

8. Sistem menyimpan data yang dimasukkan aktor ke dalam basis data.

9. Use Case selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan tampil detail kategori paket.

1. Aktor memilih salah satu kategori paket yang ingin ditampilkan.

2. Sistem menampilkan antarmuka yang berisi detail kategori paket.

3. Berlanjut ke Basic Flow langkah ke 9.

A-2 Aktor memilih untuk melakukan Ubah kategori paket.

1. Aktor memilih salah satu kategori paket yang ingin diubah.

2. Sistem menampilkan antarmuka ubah kategori paket.

3. Aktor mengubah kategori paket yang dipilih.

4. Sistem memeriksa kategori paket yang telah diubah.

- E-2 Masukan aktor tidak lengkap.
5. Sistem menyimpan perubahan data ke dalam basis data.
 6. Berlanjut ke Basic Flow langkah ke 9.
- A-3 Aktor memilih untuk melakukan hapus kategori paket
1. Aktor memilih salah satu kategori paket yang ingin dihapus.
 2. Kategori paket yang dipilih berhasil dihapus.
 3. Berlanjut pada Basic Flow langkah ke 9.

6. Error Flow

- E-1 Masukan aktor tidak lengkap.
1. Sistem memberikan peringatan bahwa masukan aktor tidak lengkap.
 2. Kembali ke Alternatif Flow A-2 langkah 3.

7. PreConditions

1. Use Case Login sudah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

Sistem memperbaharui basis data.

4.1.5 Use case Spesification: Pengelolaan Paket.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola paket wisata. Aktor dapat melakukan tambah paket, ubah paket, hapus paket, tampil detil paket, dan cari paket.

2. Primary Actor

1. Admin dan Super Admin.

3. Supporting Actor

None.

Program Studi Teknik Informatika	SKPL – SIATE	27/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan paket.
2. Sistem menampilkan antarmuka pengelolaan paket.
3. Sistem memberikan pilihan untuk melakukan tambah paket, ubah paket, hapus paket, tampil detil paket, dan cari paket.
4. Aktor memilih untuk melakukan tambah paket.
 - A-1. Aktor Memilih untuk melakukan ubah paket.
 - A-2. Aktor memilih untuk melakukan hapus paket.
 - A-3. Aktor memilih untuk melakukan cari paket.
5. Sistem menampilkan antarmuka tambah paket.
6. Aktor memasukkan paket.
7. Sistem memeriksa data yang dimasukkan aktor.
8. Sistem menyimpan data yang dimasukkan aktor ke dalam basis data.
9. Use Case selesai

5. Alternatif Flow

- A-1 Aktor memilih untuk melakukan Ubah harga.
1. Aktor memilih paket yang ingin diubah.
 2. Sistem menampilkan antarmuka Ubah paket.
 3. Aktor mengubah paket yang dipilih.
 4. Sistem memeriksa masukan aktor.
 5. Sistem menyimpan perubahan data kedalam basis data.
 6. Berlanjut pada Basic Flow langkah ke 9.
- A-3 Aktor memilih untuk melakukan hapus paket.
1. Aktor memilih paket yang ingin dihapus.
 2. Sistem menghapus paket yang dipilih dari basis data.
 3. Berlanjut pada Basic Flow langkah ke 9.

A-4 Aktor memilih untuk melakukan cari paket.

1. Aktor memasukkan data yang dicari pada kolom pencarian.
 2. Sistem menampilkan paket yang dicari.
- E-3 Data tidak tersedia.
3. Berlanjut pada Basic Flow langkah ke 9.

6. Error Flow

E-1. Data tidak tersedia.

1. Sistem memberikan peringatan bahwa data yang dicari tidak tersedia.
2. Kembali ke Alternatif Flow A-4 langkah ke 1.

7. PreCondition

1. Use case login sudah dilakukan.
2. Aktor sudah memasuki sistem

8. PostCondition

Sistem memperbaharui basis data.

4.1.6 Use case Spesification: Pengelolaan Harga.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola harga. Aktor dapat melakukan tambah harga, ubah harga, hapus harga, tampil harga, dan cari harga.

2. Primary Actor

Admin dan Super Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan harga.
2. Sistem menampilkan antarmuka pengelolaan harga.

3. Sistem memberikan pilihan untuk melakukan tambah harga, ubah harga, hapus harga, tampil harga, dan cari harga.
4. Aktor memilih untuk melakukan tambah harga.
 - A-1. Aktor Memilih untuk melakukan ubah harga.
 - A-2. Aktor memilih untuk melakukan hapus harga.
 - A-3. Aktor memilih untuk melakukan cari harga.
5. Sistem menampilkan antarmuka tambah paket.
6. Aktor memasukkan harga.
7. Sistem memeriksa data yang dimasukkan aktor.
8. Sistem menyimpan data yang dimasukkan aktor ke dalam basis data.
9. Use Case selesai

5. Alternatif Flow

- A-1 Aktor memilih untuk melakukan Ubah harga.
1. Aktor memilih harga yang ingin diubah.
 2. Sistem menampilkan antarmuka Ubah harga.
 3. Aktor mengubah pharga yang dipilih.
 4. Sistem memeriksa masukan aktor.
 5. Sistem menyimpan perubahan data kedalam basis data.
 6. Berlanjut pada Basic Flow langkah ke 9.
- A-3 Aktor memilih untuk melakukan hapus harga.
1. Aktor memilih harga yang ingin dihapus.
 2. Sistem menghapus harga yang diipilih dari basis data.
 3. Berlanjut pada Basic Flow langkah ke 9.
- A-4 Aktor memilih untuk melakukan cari harga.
1. Aktor memasukkan data yang dicari pada kolom pencarian.
 2. Sistem menampilkan paket yang dicari.

Program Studi Teknik Informatika	SKPL – SIATE	30/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-3 Data tidak tersedia.

1. Berlanjut pada Basic Flow langkah ke 9.

9. Error Flow

E-1. Data tidak tersedia.

1. Sistem memberikan peringatan bahwa data yang dicari tidak tersedia.
2. Kembali ke Alternatif Flow A-4 langkah ke 1.

10. PreCondition

1. Use case login sudah dilakukan.
2. Aktor sudah memasuki sistem

11. PostCondition

Sistem memperbaharui basis data.

4.1.7 Use case Spesification : Pengelolaan Ref Person

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola jumlah orang. Aktor dapat melakukan tambah person, ubah person, hapus person, tampil detil person, dan cari person.

2. Primary Actor

3. Admin dan Super Admin.

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan person.
2. Sistem menampilkan antarmuka pengelolaan person.
3. Sistem memberikan pilihan untuk melakukan tambah person, ubah person, hapus person, tampil detil person, dan cari person.
4. Aktor memilih untuk melakukan tambah person.

Program Studi Teknik Informatika	SKPL – SIATE	31/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-1 Aktor Memilih untuk melakukan ubah person.

A-2 Aktor memilih untuk melakukan hapus person.

A-3 Aktor memilih untuk melakukan cari person.

5. Sistem menampilkan antarmuka pengelolaan person.

E-1 person tidak tersedia.

6. Aktor memilih jumlah person.

7. Aktor memasukkan jumlah person.

8. Sistem memeriksa data yang dimasukkan aktor.

E-2 Masukan aktor tidak lengkap.

9. Sistem menyimpan data yang dimasukkan aktor kedalam basis data.

10. Use Case selesai.

5. **Alternatif Flow**

A-1 Aktor memilih untuk melakukan tampil detil jumlah person.

1. Aktor memilih salah satu jumlah person yang akan ditampilkan.

2. Sistem menampilkan antarmuka yang berisi person.

3. Berlanjut pada Basic Flow langkah ke 10.

A-2 Aktor memilih untuk melakukan ubah person.

1. Aktor memilih jumlah person akan diubah.

2. Sistem menampilkan antarmuka ubah person.

3. Aktor mengubah person yang dipilih.

4. Sistem memeriksa masukan aktor.

5. Sistem menyimpan perubahan data kedalam basis data.

6. Berlanjut pada Basic Flow langkah ke 10.

A-3 Aktor memilih untuk melakukan hapus person.

1. Aktor memilih person yang akan dihapus.

2. Person berhasil dihapus.

3. Berlanjut pada Basic Flow langkah ke 10.

A-4 Aktor memilih untuk melakukan cari person.

1. Aktor memasukkan data yang dicari pada kolom pencarian.

2. Sistem menampilkan data yang dicari.

E-5 Data yang dicari tidak tersedia.

3. Berlanjut pada Basic Flow langkah ke 10.

6. **Error Flow**

E-1 Data yang dicari tidak tersedia.

1. Sistem memberi peringatan bahwa data yang dicari tidak ditemukan.

2. Kembali pada Alternatif Flow A-4 langkah 1.

7. **PreCondition**

1. Use case login telah dilakukan.

2. Aktor telah masuk kedalam sistem.

8. **PostCondition**

Sistem memperbaharui basis data.

4.1.8 **Use case Spesification** : Pengelolaan Transaksi Pemesanan

1. **Brief Description**

Use case ini digunakan oleh aktor untuk mengelola data pengelolaan transaksi pemesanan. Aktor dapat mencari transaksi pemesanan.

2. **Primary Actor**

1. Admin dan Super Admin.

Program Studi Teknik Informatika	SKPL – SIATE	33/ 34
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3. Supporting Actor

None.

4. Basic Flow

1. Use case ini dimulai ketika aktor memilih untuk melakukan pengelolaan transaksi pemesanan.
2. Sistem menampilkan antarmuka pengelolaan transaksi pemesanan.
3. Aktor memilih untuk mencari transaksi pemesanan.
4. Sistem menampilkan antarmuka transaksi pemesanan.
E-1. Person tidak tersedia.
5. Sistem memeriksa masukan aktor.
6. Use case selesai.

5. Alternatif Flow

1. None

6. Error Flow

E-1 Data yang dicari tidak tersedia.

1. Sistem memberi peringatan bahwa data yang dicari tidak tersedia.
2. Kembali pada Basic Flow langkah 1.

7. PreCondition

1. Use case login telah dilakukan.
2. Aktor telah masuk kedalam sistem.

8. PostCondition

None.

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SIATE

(Sistem Informasi Alta Tour Dan event Berbasis Web)

Untuk:

Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta


Dipersiapkan Oleh:

Albertus Bagawan Abdi /100706172

Program Studi Teknik Informatika

Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen	Halaman
			1/43

DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

INDEKS TGL	-	A	B	C	D	E	F
DITULIS OLEH							
DIPERIKSA OLEH							
DISETUJUI OLEH							

DAFTAR HALAMAN PERUBAHAN

Halaman	Revisi	Halaman	Revisi

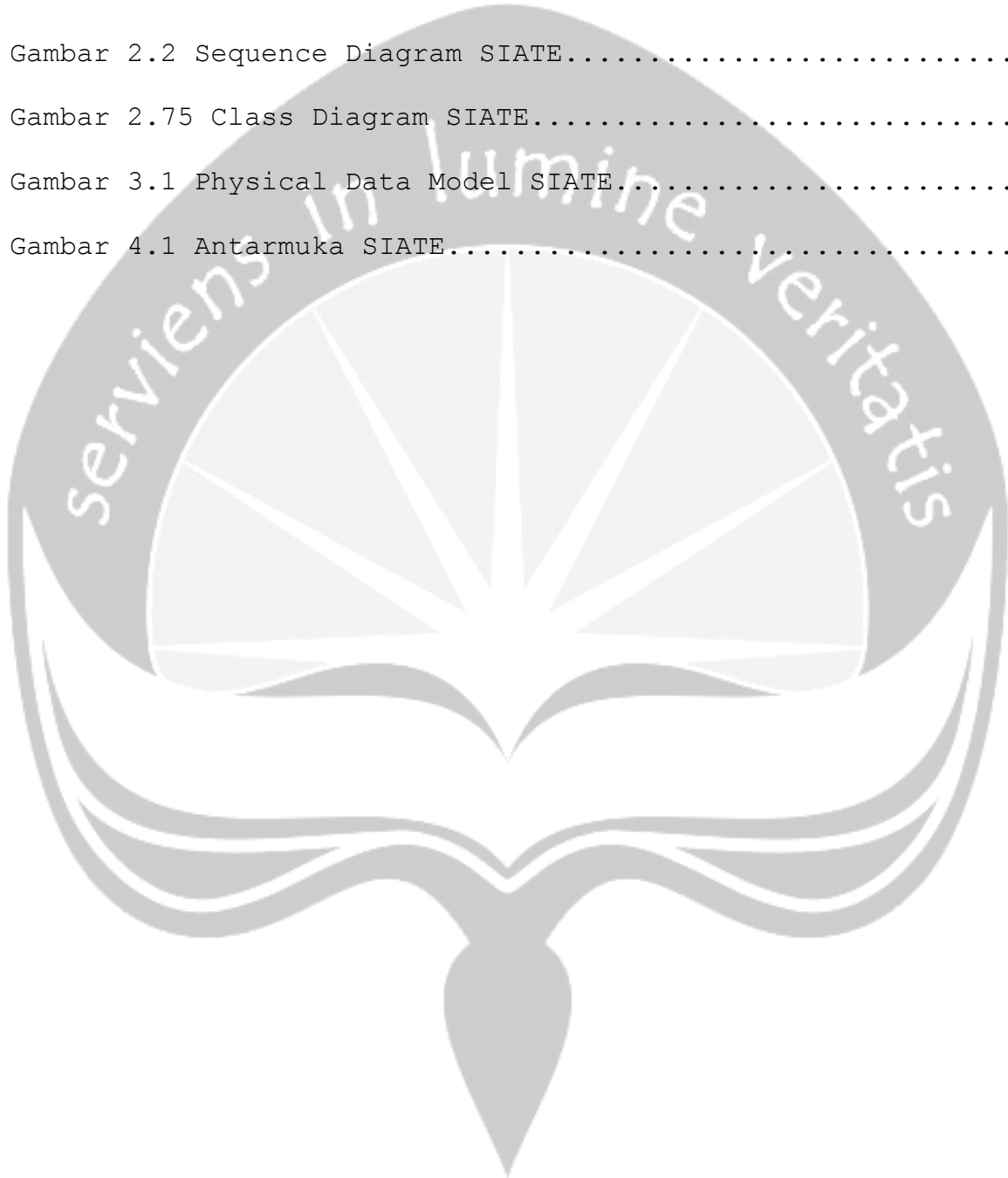


DAFTAR ISI

1	Pendahuluan.....	6
1.1	Tujuan.....	6
1.2	Lingkup Masalah.....	6
1.3	Definisi, Akronim, dan Singkatan.....	6
1.4	Referensi.....	7
2	Perencanaan Sistem	8
2.1	Perancangan Arsitektur.....	8
2.2	Perancangan Rinci.....	9
2.2.1	Sequence Diagram.....	9
2.2.2	Class Diagram.....	22
2.2.3	Class Diagram Spessification.....	24
3	Perancangan Data.....	32
3.1	Dekomposisi Data.....	32
3.2	Physical Data Model.....	36
4	Perancangan Antarmuka.....	37

DAFTAR GAMBAR

Gambar 2.1	Arsitektur SIATE.....	8
Gambar 2.2	Sequence Diagram SIATE.....	9
Gambar 2.75	Class Diagram SIATE.....	22
Gambar 3.1	Physical Data Model SIATE.....	36
Gambar 4.1	Antarmuka SIATE.....	37



1. Pendahuluan

1.1 Tujuan

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

1.2 Lingkup Masalah

Perangkat Lunak SIATE dikembangkan dengan tujuan untuk:

1. Menangani pengelolaan pengguna sistem.
2. Menangani pengelolaan paket wisata.
3. Menangani pengelolaan kategori paket wisata.
4. Menangani pelaporan transaksi pemesanan.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD). Merupakan deskripsi dari perancangan produk / perangkat lunak yang akan dikembangkan.
DPPL-SIATE-XX	Kode yang merepresentasikan kebutuhan pada SIATE (Sistem Informasi Alta Tour Dan Travel Berbasis Web) di mana XX merupakan nomor fungsi produk.
SIATE	Perangkat lunak pengelolaan Alta Tour Dan Event berbasis Web.

Super Admin	Pengguna sistem yang bertugas mengelola data pengguna, paket, tempat, kategori paket, ref person dan transaksi pemesanan dari Alta Tour Dan Travel berbasis Web.
Admin	Pengguna sistem yang bertugas mengelola paket, tempat, kategori paket, ref person dan transaksi pemesanan dari Alta Tour Dan Travel berbasis Web.
Web	<i>Website</i> atau yang biasa disebut <i>Web</i> adalah kumpulan halaman yang memiliki informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman.
Basis Data	Basis Data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi.
Server	Komputer yang menyediakan sumber data bagi <i>Client</i> yang terhubung melalui jaringan.
Code Igniter	Aplikasi open source yang berupa framework dengan model MVC (<i>Model, View, Controller</i>) untuk membangun <i>website</i> dinamis dengan menggunakan PHP.

1.4 Referensi

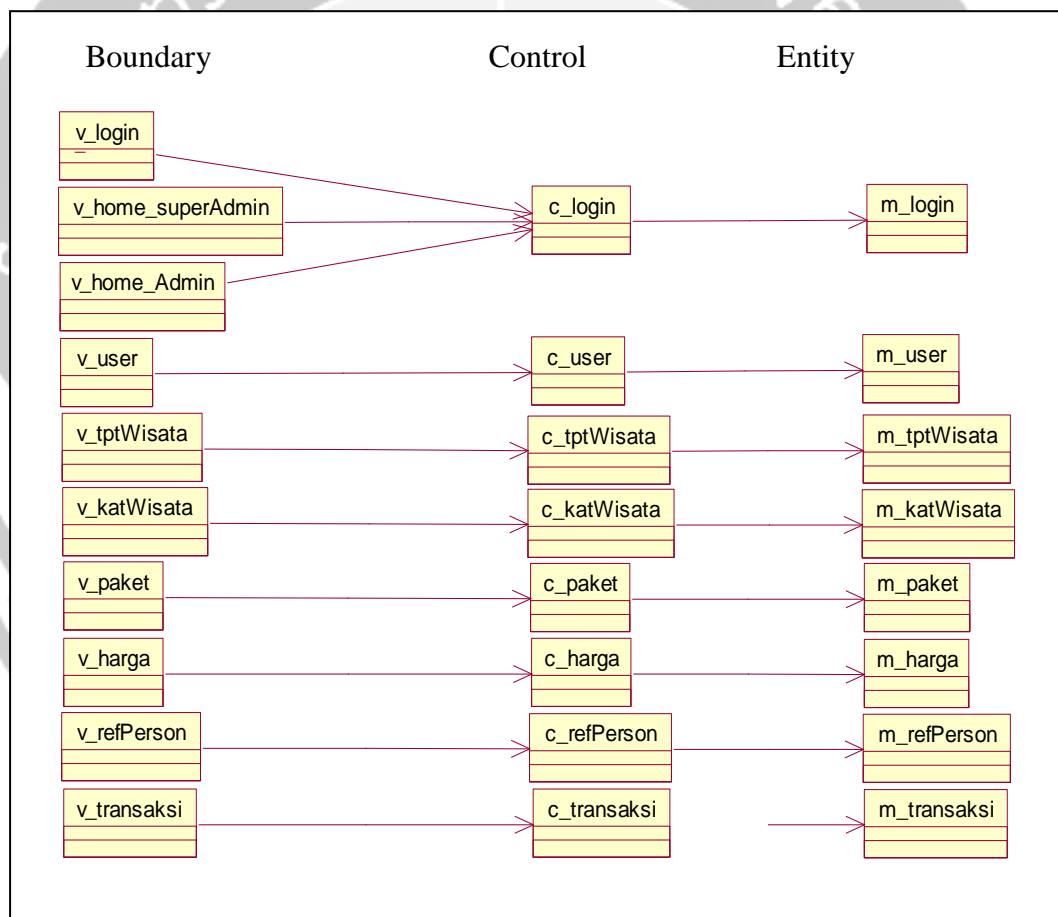
Referensi yang digunakan pada perangkat lunak tersebut adalah:

Program Studi Teknik Informatika	DPPL – SIATE	7/43
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Deskripsi Perancangan Perangkat Lunak (DPPL) Sistem Informasi Apotek Pasti oleh Arik Eko Prasetyo / 110706608, Universitas Atma Jaya Yogyakarta.

2 Perancangan Sistem

2.1 Perancangan Asitektur

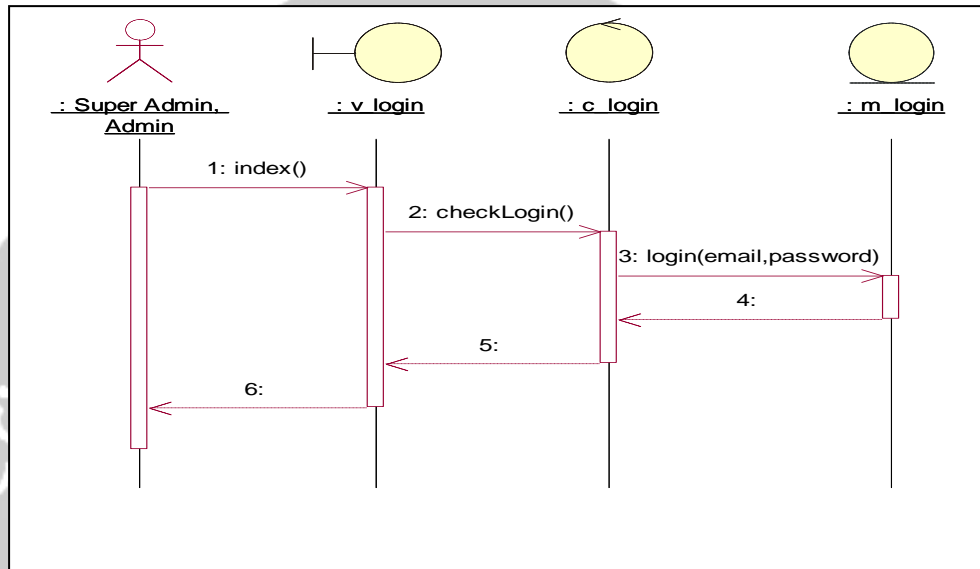


Gambar 2.1 Rancangan Arsitektur SIATE

2.2 Perancangan Rinci

2.2.1. Sequence Diagram

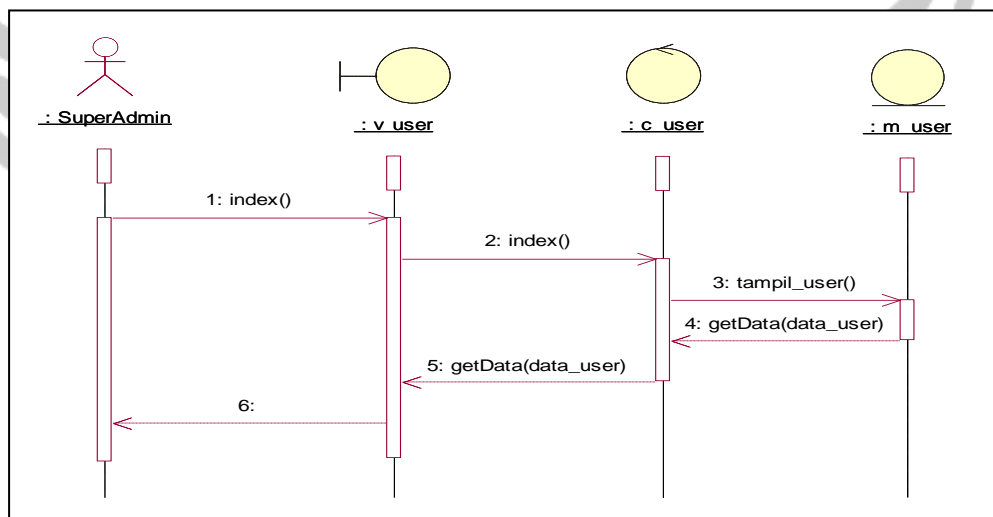
2.2.1.1 Fungsi Login



Gambar 2.2 Sequence Diagram: Login

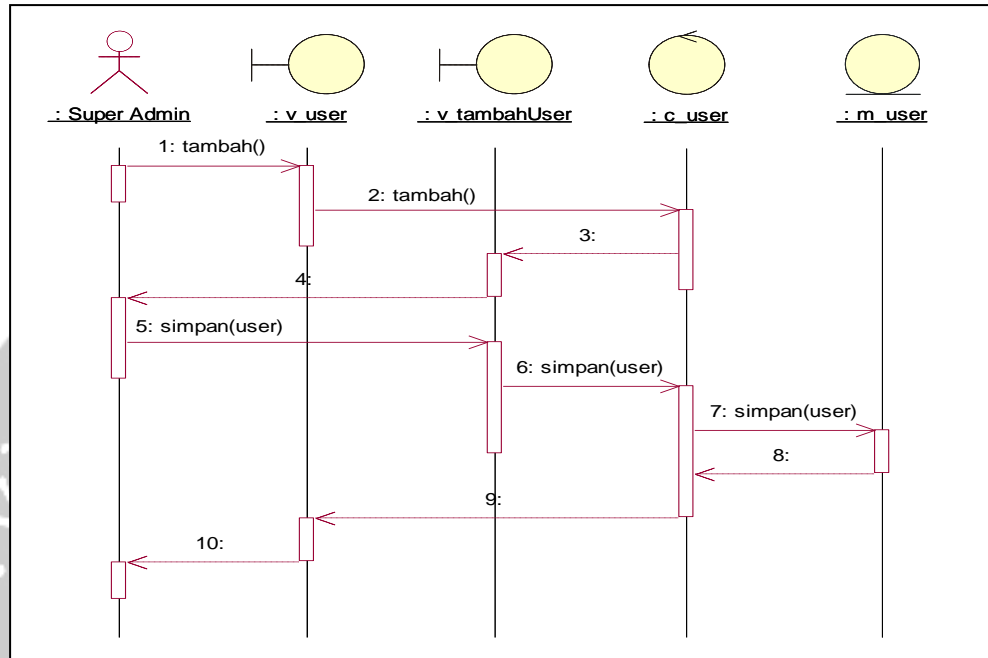
2.2.1.2 Fungsi Pengelolaan User

A. Fungsi Tampil User



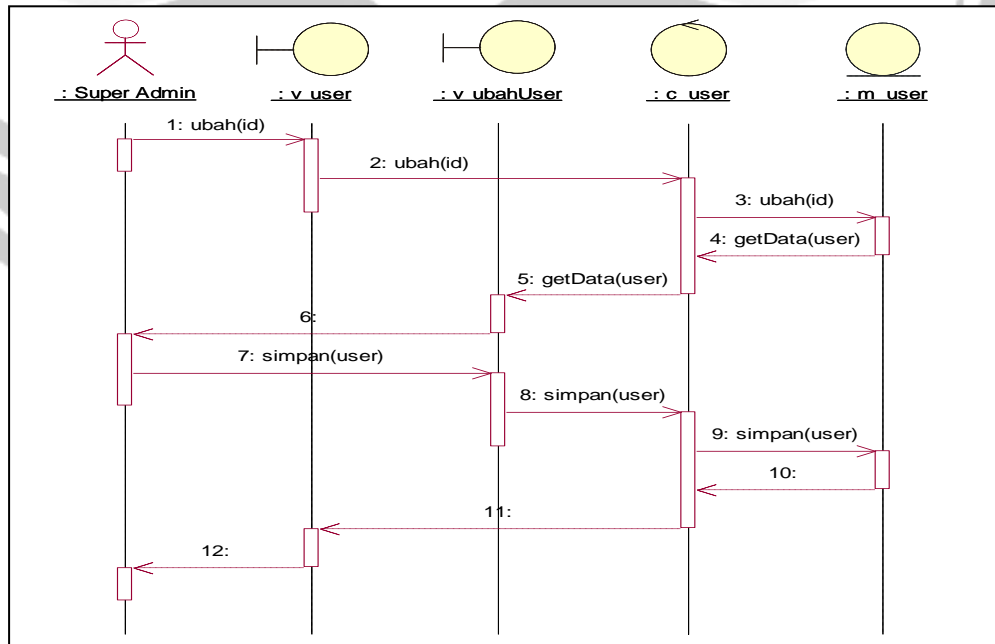
Gambar 2.3 Sequence Diagram: Tampil User

B. Fungsi Tambah User



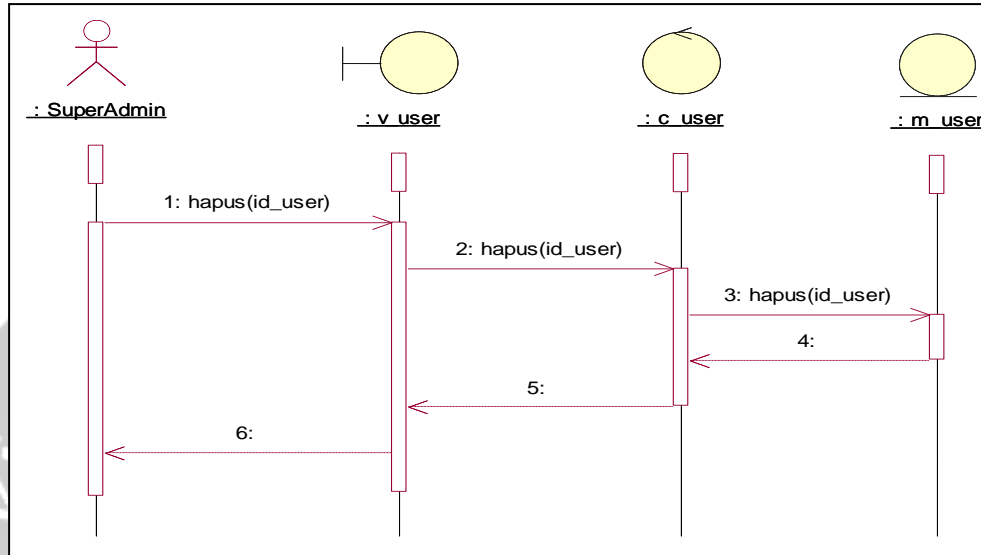
Gambar 2.4 Sequence Diagram: Tambah user

C. Fungsi Ubah Data Unit



Gambar 2.5 Sequence Diagram: Ubah User

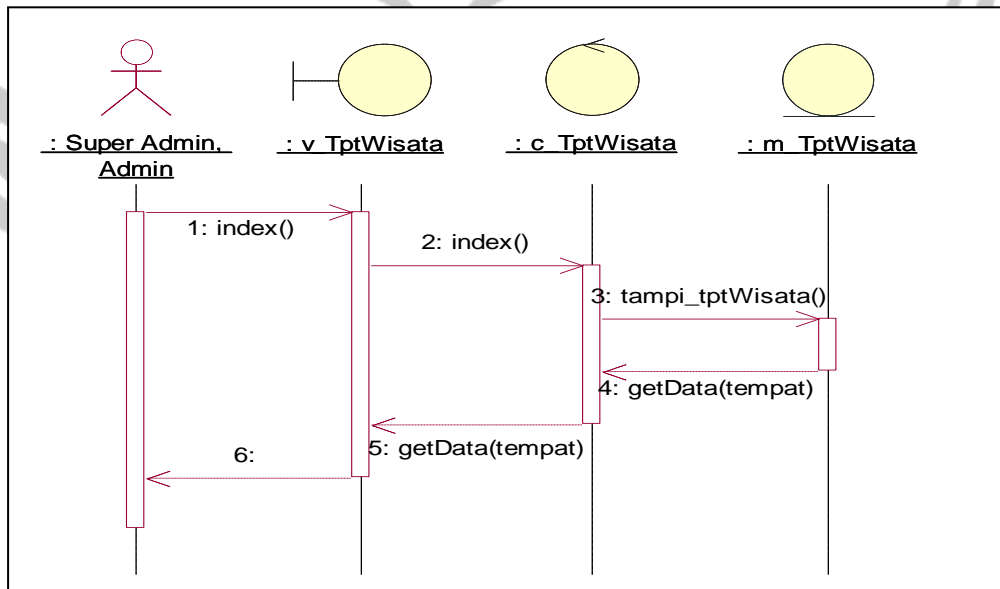
D. Fungsi Hapus Data Unit



Gambar 2.6 Sequence Diagram: Hapus User

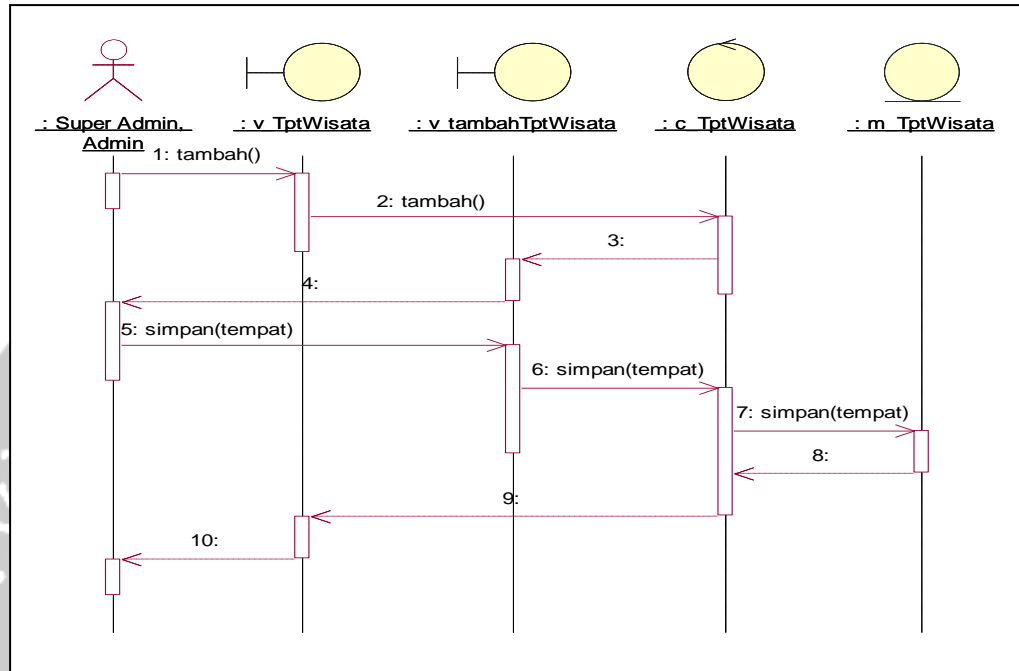
2.2.1.3 Fungsi Pengelolaan Tempat Wisata

A. Fungsi Tampil Tempat Wisata



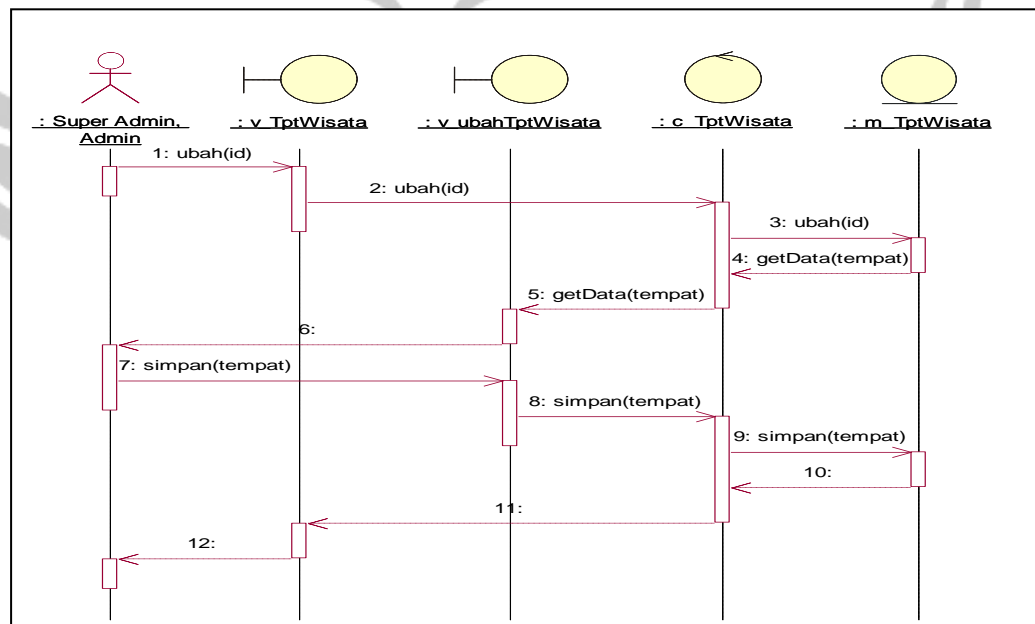
Gambar 2.7 Sequence Diagram: Tampil Tempat Wisata

B. Fungsi Tambah Tempat Wisata



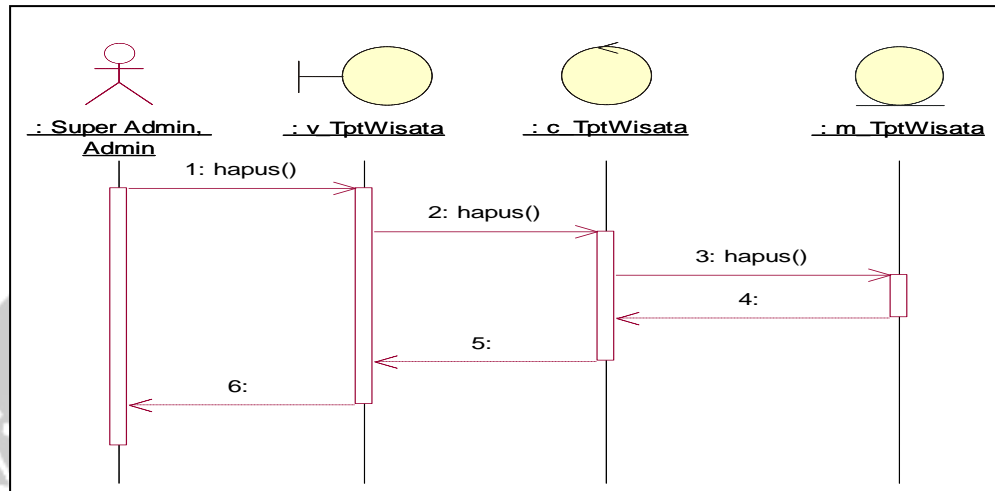
Gambar 2.8 Sequence Diagram: Tambah Tempat Wisata

C. Fungsi Ubah Tempat Wisata



Gambar 2.9 Sequence Diagram: Ubah Tempat Wisata

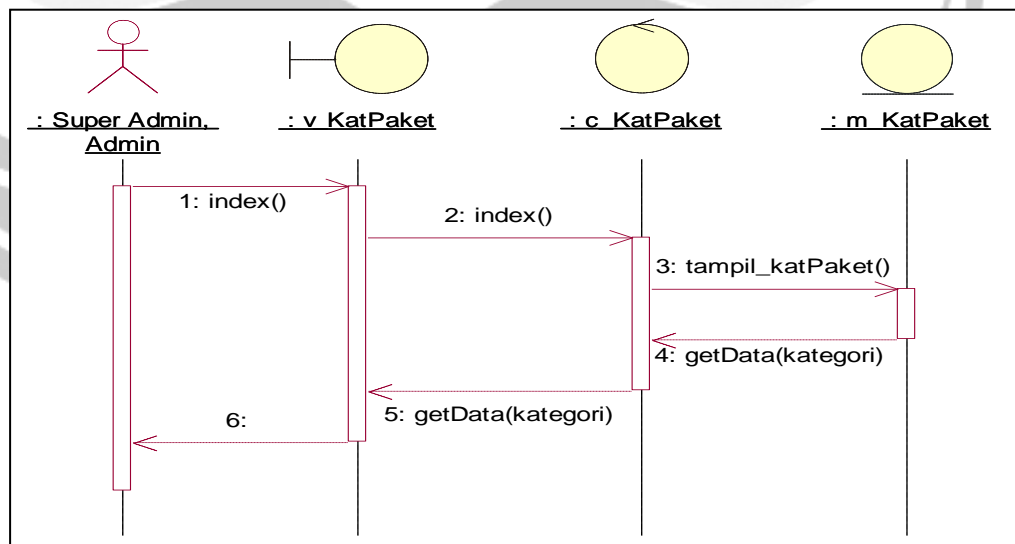
D. Fungsi Hapus Tempat Wisata



Gambar 2.10 Sequence Diagram: Hapus Tempat Wisata

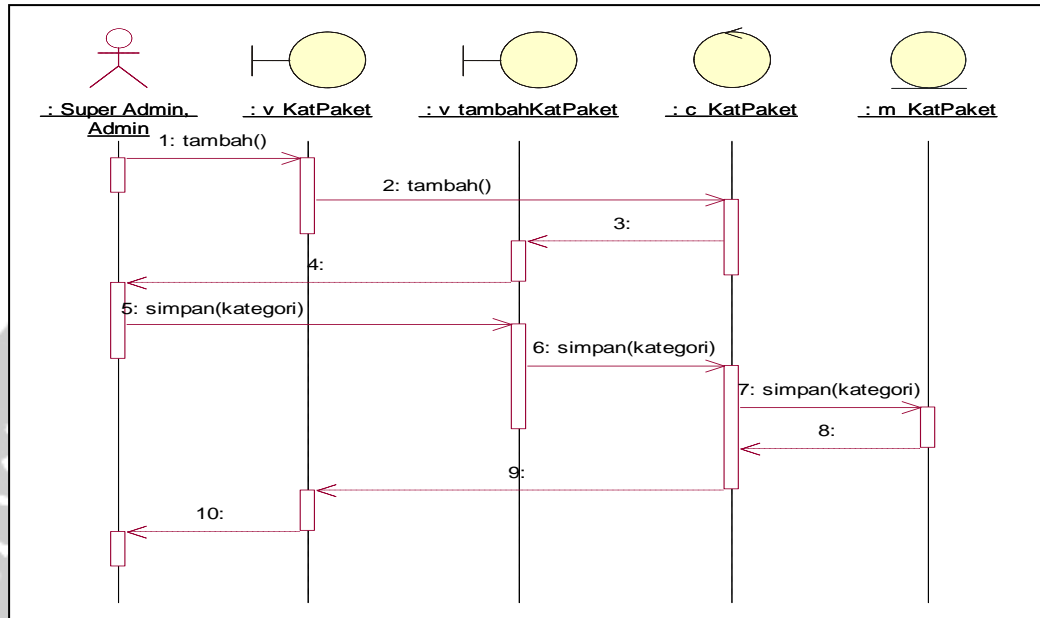
2.2.1.4 Fungsi Pengelolaan Kategori Paket

A. Fungsi Tampil Kategori Paket



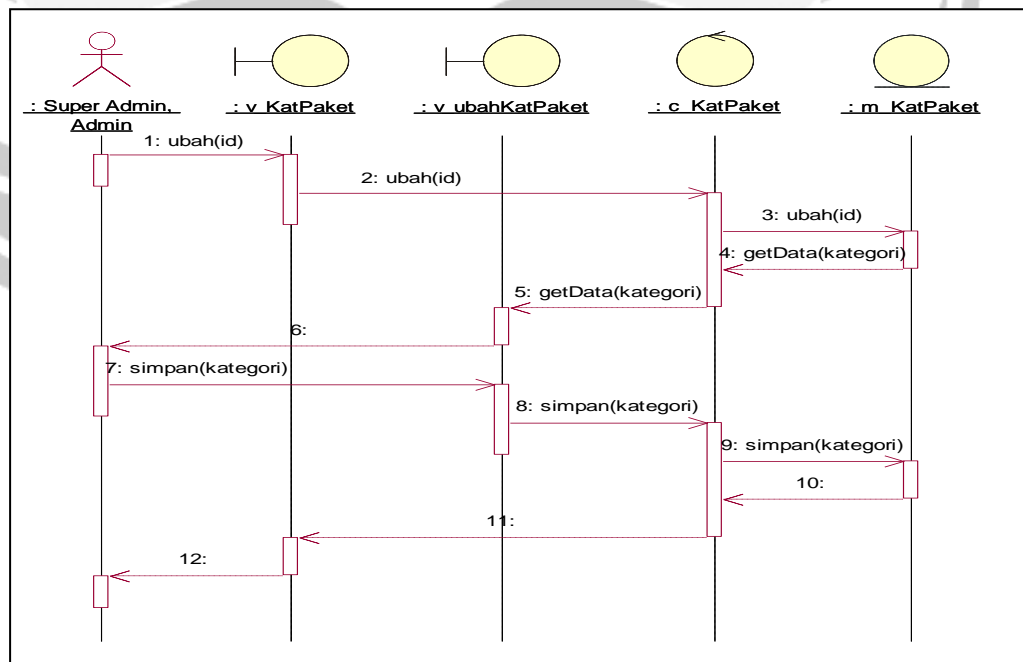
Gambar 2.12 Sequence Diagram: Tampil Kategori

B. Fungsi Tambah Kategori Paket



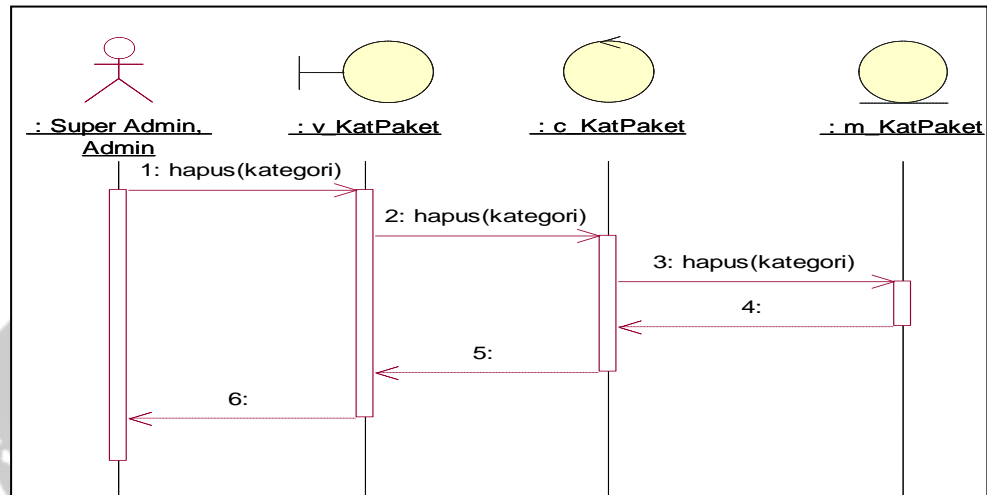
Gambar 2.13 Sequence Diagram: Tambah Kategori

C. Fungsi Ubah Kategori Paket



Gambar 2.14 Sequence Diagram: Ubah Kategori Paket

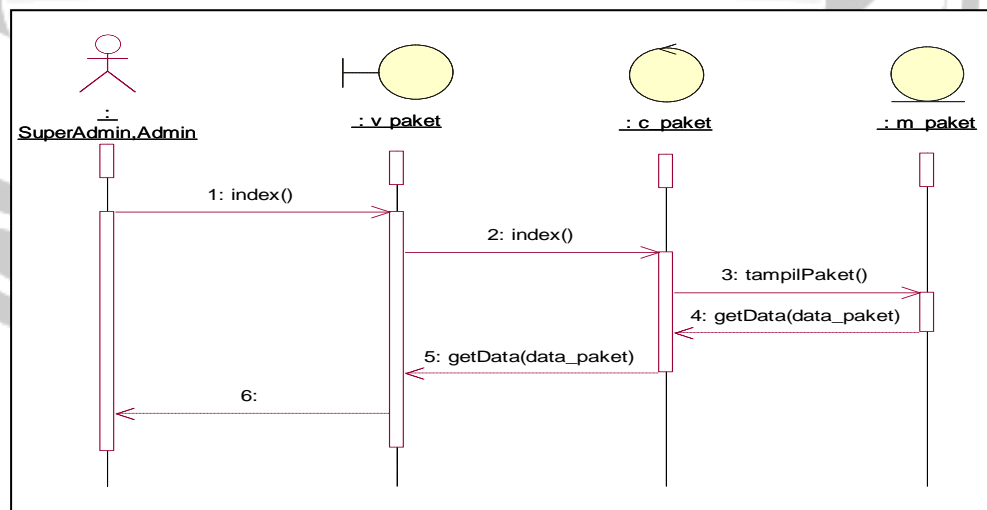
D. Fungsi Hapus Kategori Paket



Gambar 2.15 Sequence Diagram: Hapus Kategori

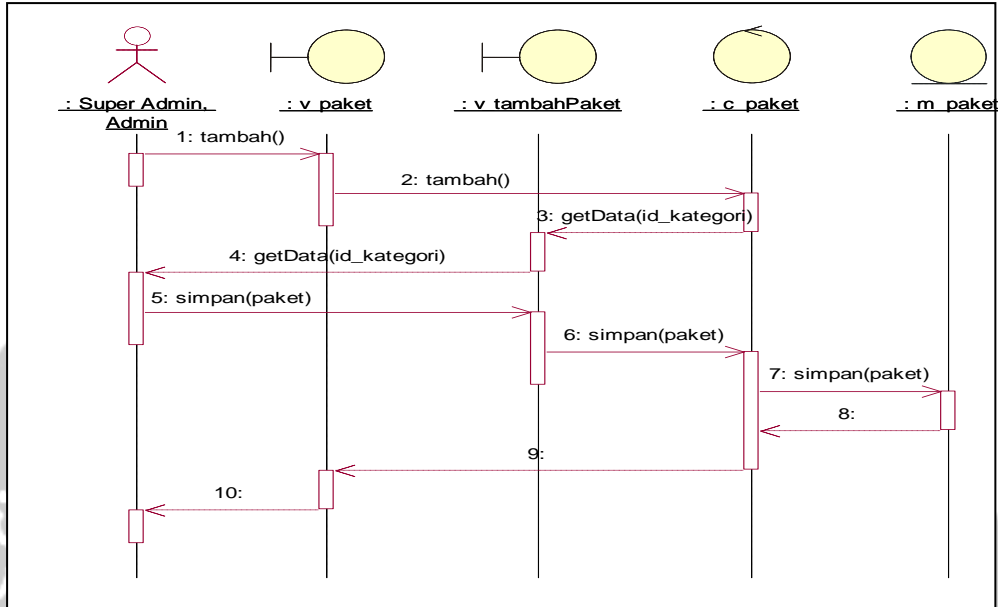
2.2.1.5 Fungsi Pengelolaan Paket

A. Tampil Paket



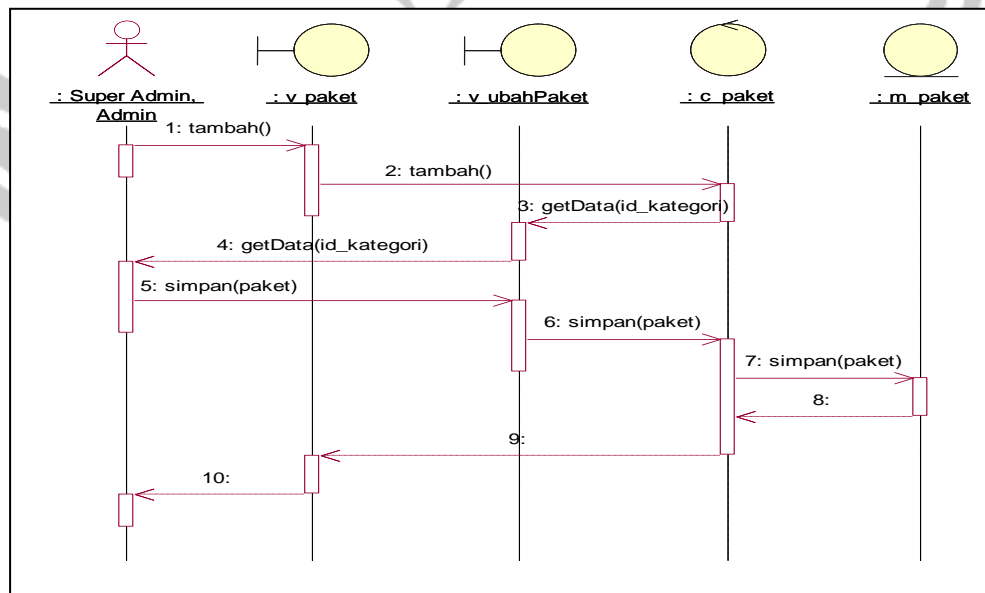
Gambar 2.16 Sequence Diagram: Tampil Paket

B. Tambah Paket



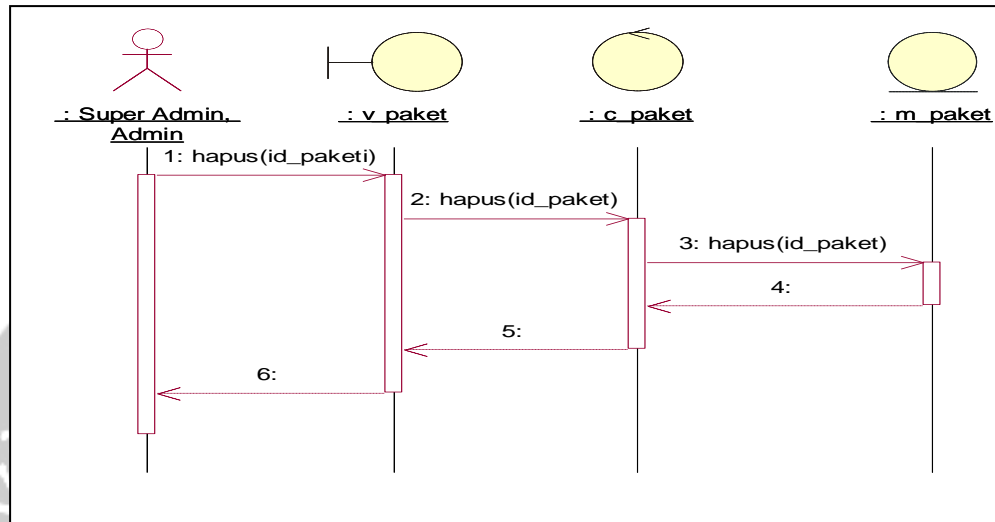
Gambar 2.17 Sequence Diagram: Tambah Paket

C. Ubah Data Paket



Gambar 2.18 Sequence Diagram: Ubah Paket

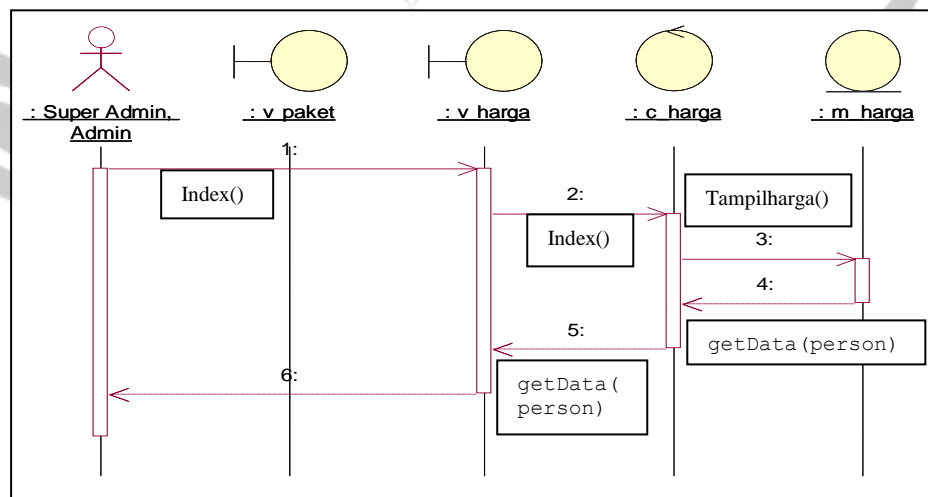
D. Hapus Paket



Gambar 2.19 Sequence Diagram: Hapus Paket

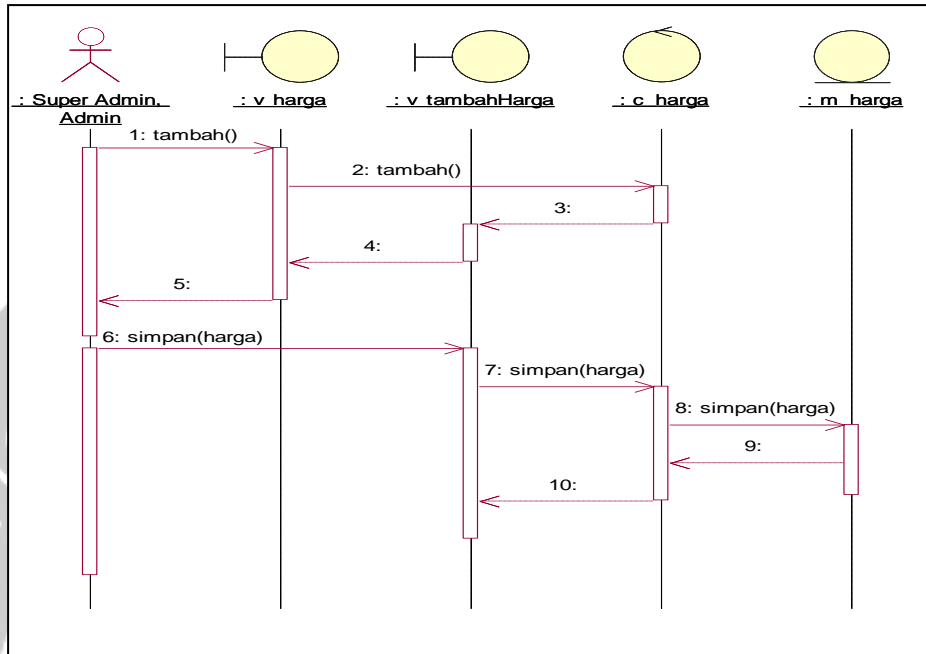
2.2.1.6 Fungsi Pengelolaan Harga

A. Tampil Harga



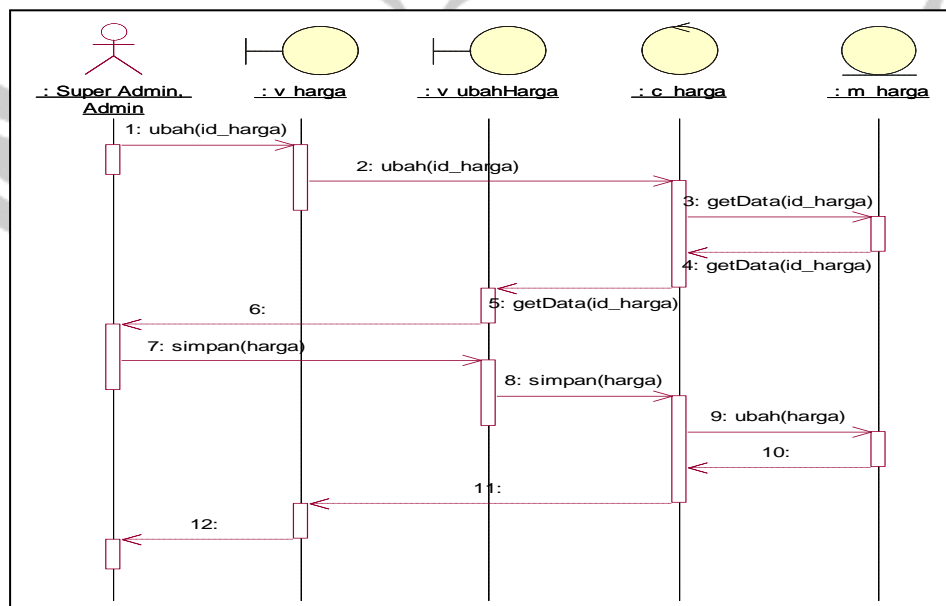
Gambar 2.20 Sequence Diagram: Tampil Harga

B. Tambah Harga



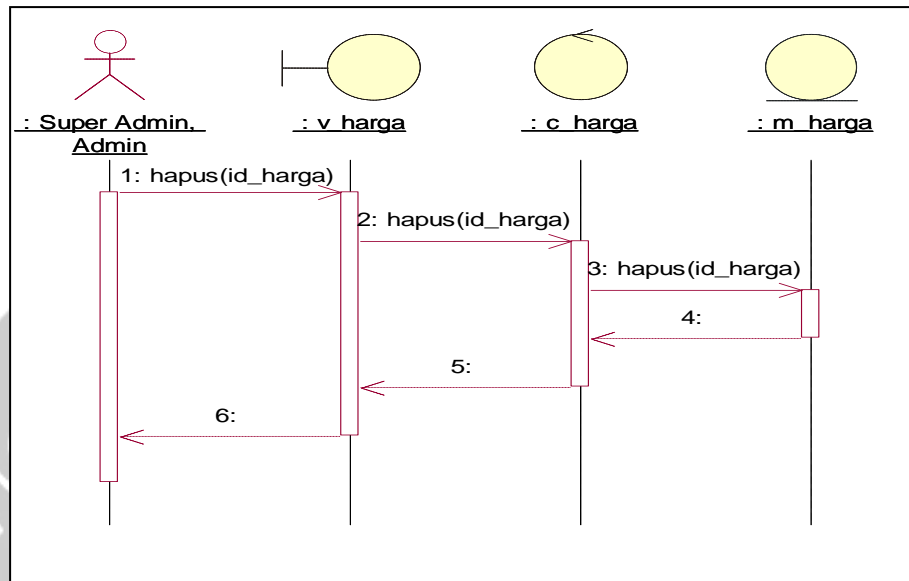
Gambar 2.21 Sequence Diagram: Tambah Harga

C. Ubah Data Pengadaan



Gambar 2.22 Sequence Diagram: Ubah Pengadaan

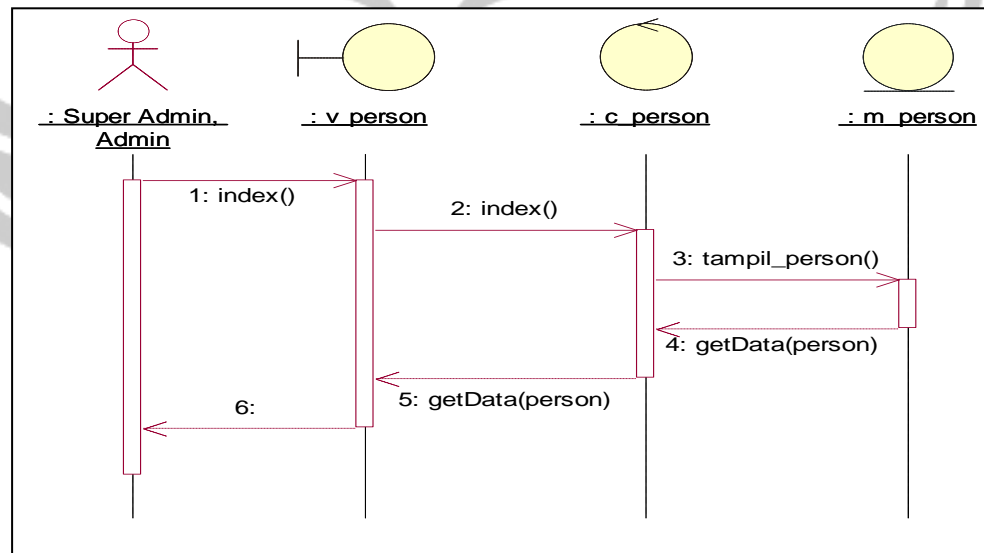
D. Hapus Data Harga



Gambar 2.23 Sequence Diagram: Hapus Harga

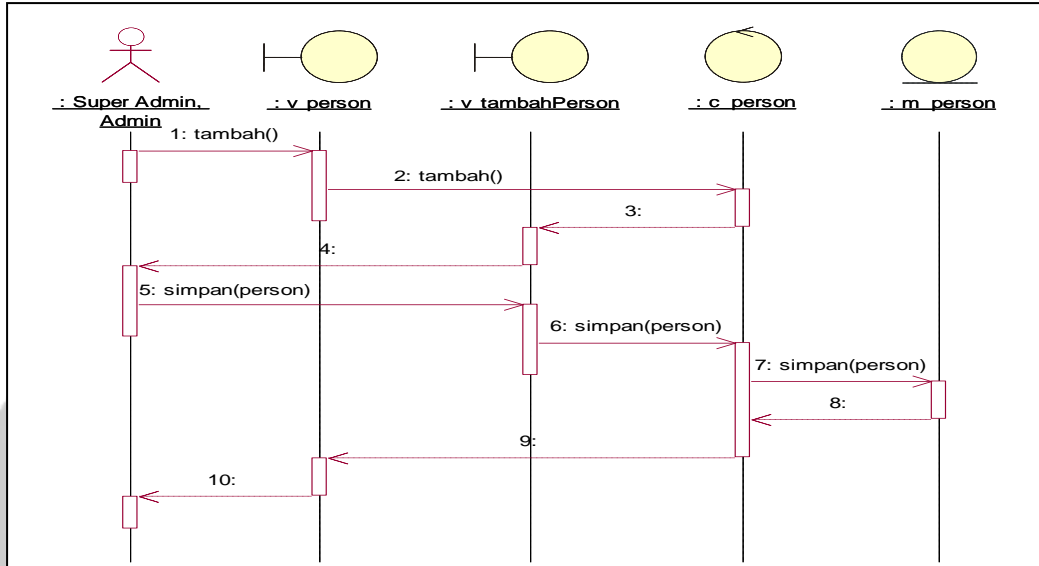
2.2.1.7 Fungsi Pengelolaan Ref Person

A. Tampil Ref Person



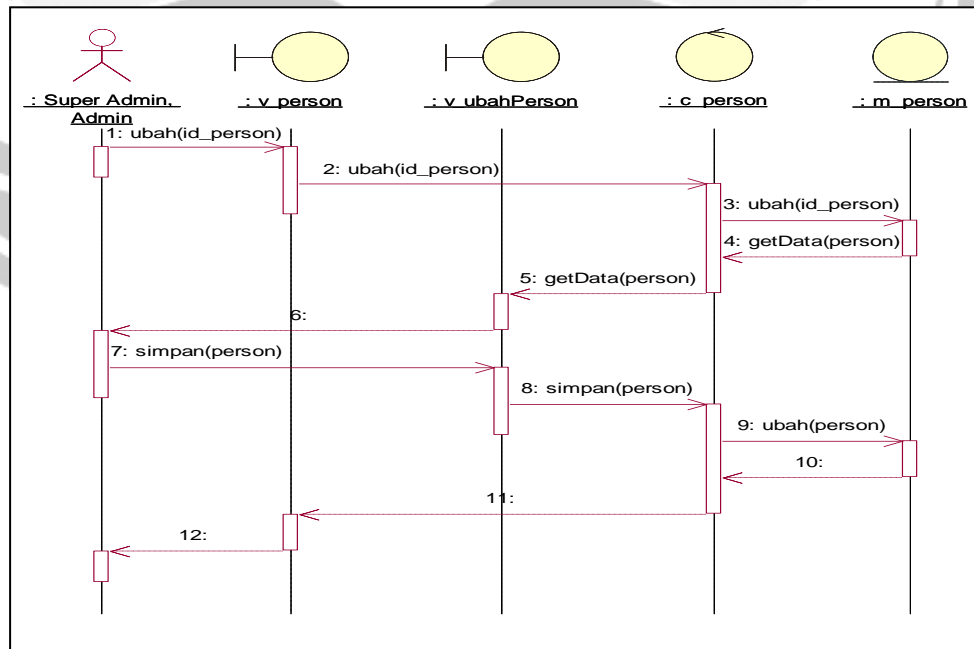
Gambar 2.24 Sequence Diagram: Tampil Ref Person

B. Tambah Ref Person



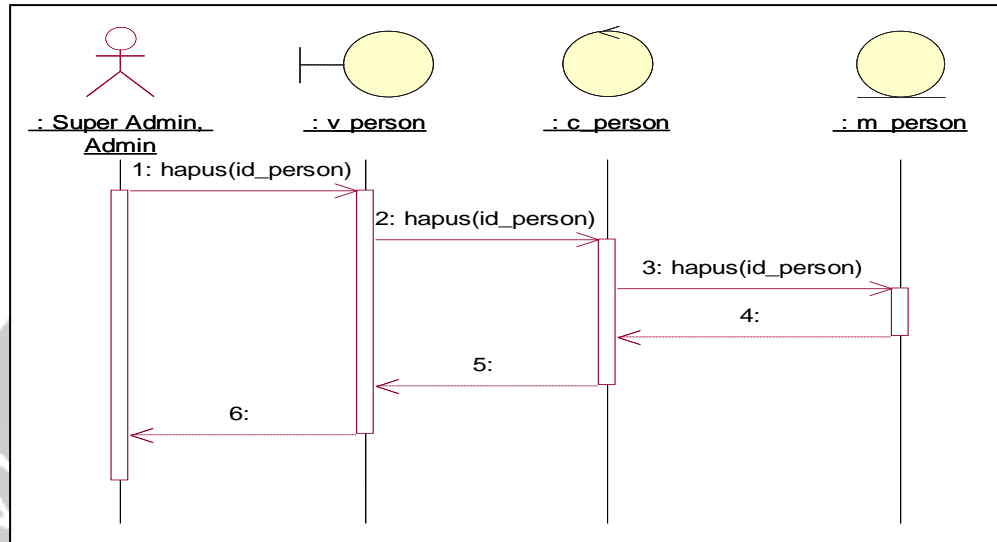
Gambar 2.25 Sequence Diagram: Tambah Ref Person

C. Ubah Ref Person



Gambar 2.26 Sequence Diagram: Ubah Ref Person

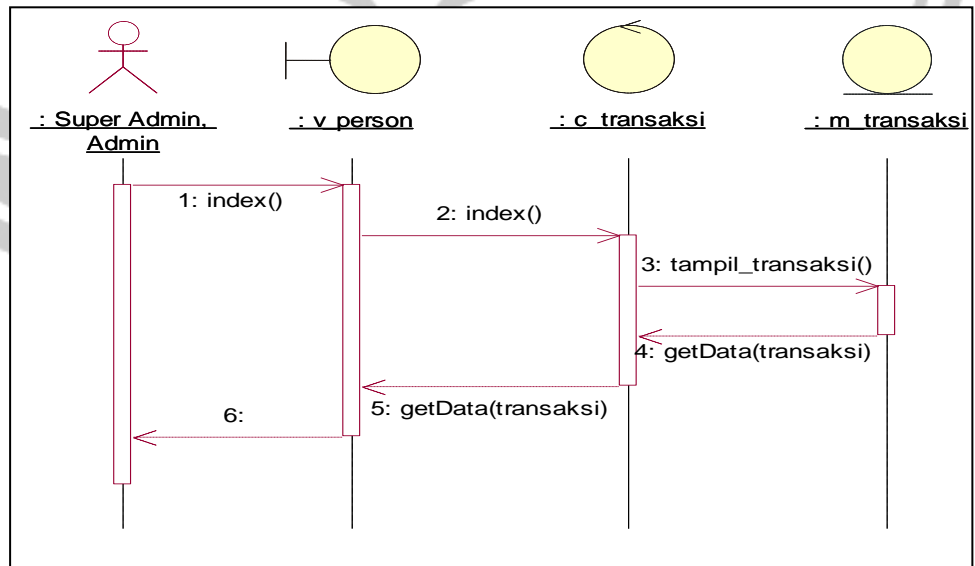
D. Hapus Ref Person



Gambar 2.27 Sequence Diagram: Hapus Ref Person

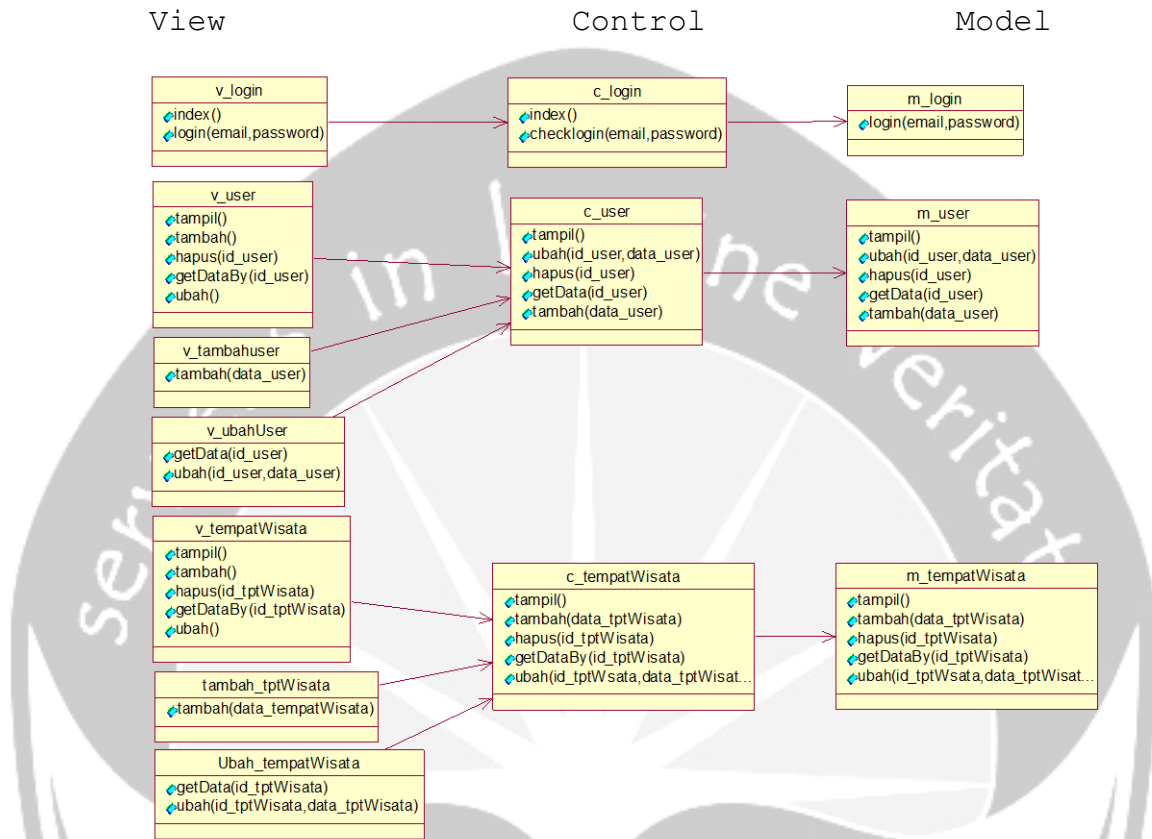
2.2.1.8 Fungsi Pengelolaan Transaksi Pemesanan

A. Tampil Transaksi Pemesanan

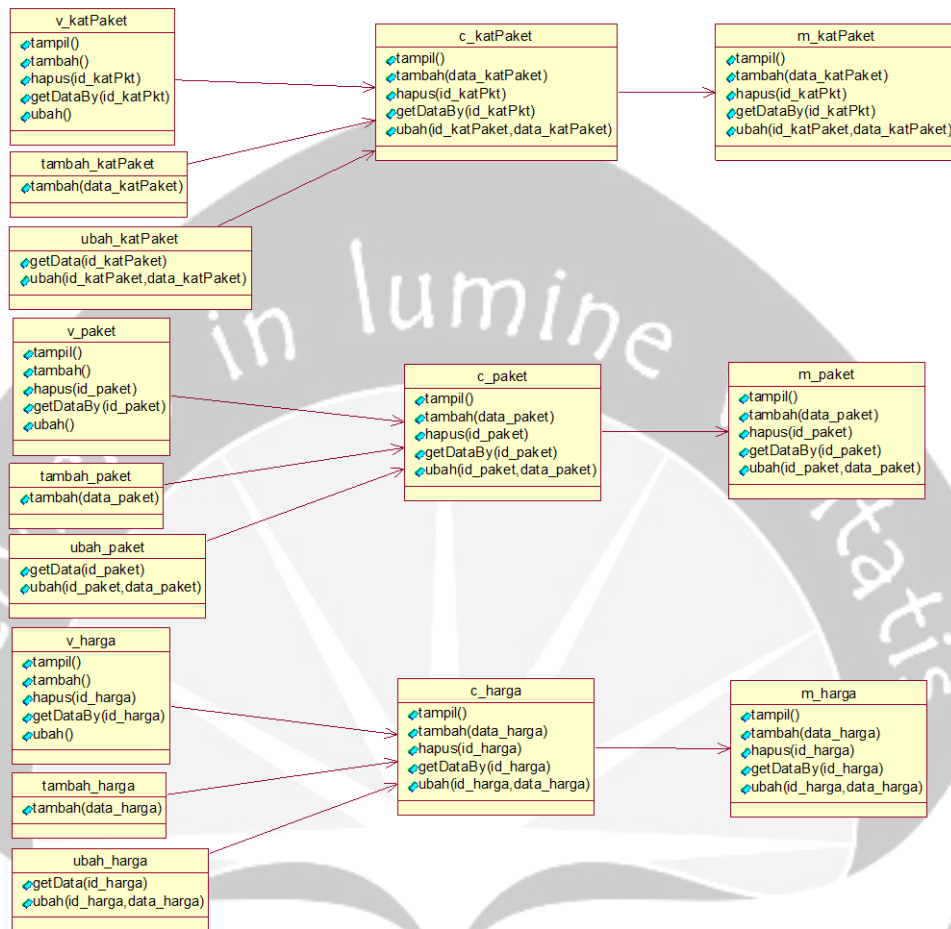


Gambar 2.28 Sequence Diagram: Tampil Transaksi Pemesanan

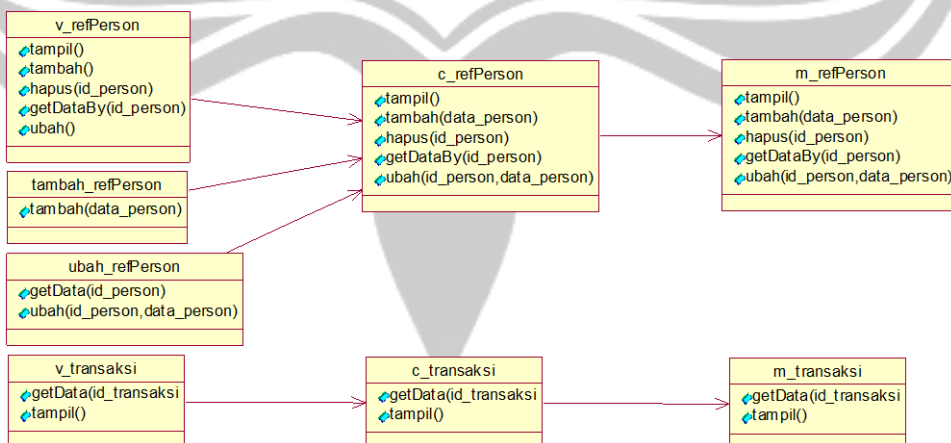
2.2.2 ClassDiagram



Gambar 2.75 Class Diagram SIATE



Lanjutan Gambar 2.75 Class Diagram SIATE



Lanjutan Gambar 2.75 Class Diagram SIATE

2.2.3 Class Diagram Specific Descriptions

2.2.3.1 Specific Design Class v_login

v_login	<<boundary>>
+index() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+login(email,password) Operasi ini digunakan untuk memeriksa username dan password pengguna.	

2.2.1.2 Specific Design Class v_user

v_user	<<boundary>>
+tampil():void Operasi ini digunakan untuk menampilkan data pengguna dengan format tabel.	
+tambah():void Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data pengguna baru.	
+getDataBy(id_user):void Operasi ini digunakan untuk menampilkan form ubah beserta data pengguna yang spesifik sesuai dengan id yang diterima.	
+hapus(id_user):void Operasi ini digunakan untuk menghapus data pengguna sesuai dengan id yang diterima.	
+ubah():void Operasi ini digunakan untuk menampilkan form ubah yang	

digunakan untuk mengubah data pengguna.

2.2.1.3 Specific Design Class v_tambahuser

v_tambahuser	<<boundary>>
+tambah(data_user):void Operasi ini digunakan untuk menyimpan data pengguna yang baru.	

2.2.1.4 Specific Design Class v_ubahuser

v_ubahuser	<<boundary>>
+getDataBy(id_user):void Operasi ini digunakan untuk mengambil data pengguna yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia. +ubah(id_user,data_user):void Operasi ini digunakan untuk menyimpan data pengguna yang telah mendapatkan perubahan.	

2.2.1.5 Specific Design Class v_tempatWisata

v_tempatWisata	<<boundary>>
+tampil():void Operasi ini digunakan untuk menampilkan data tempat wisata dengan format tabel. +tambah():void Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data tempat wisata baru.	

```
+getDataBy(id_tptWista):void
```

Operasi ini digunakan untuk menampilkan form ubah beserta data tempat wisata yang spesifik sesuai dengan id yang diterima.

```
+hapus(id_tptWisata):void
```

Operasi ini digunakan untuk menghapus data tempat wisata sesuai dengan id yang diterima.

2.2.1.6 Specific Design Class tambah_tptWisata

Tambah_tptWisata	<<boundary>>
<pre>+tambah(data_tptWisata):void</pre> <p>Operasi ini digunakan untuk menyimpan data tempat wisata yang baru.</p>	

2.2.1.7 Specific Design Class ubah_tptWisata

Ubah_tptWisata	<<boundary>>
<pre>+getDataBy(id_tptWisata):void</pre> <p>Operasi ini digunakan untuk mengambil data tempat wisata yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia.</p> <pre>+ubah(id_tptWisata,data_tptWisata):void</pre> <p>Operasi ini digunakan untuk menyimpan data tempat wisata yang telah mendapatkan perubahan.</p>	

2.2.1.8 Specific Design Class v_katPaket

v_katPaket	<<boundary>>
+tampil():void Operasi ini digunakan untuk menampilkan data kategori paket dengan format tabel.	
+tambah():void Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data kategori paket baru.	
+getDataBy(id_katPaket):void Operasi ini digunakan untuk menampilkan form ubah beserta data kategori paket yang spesifik sesuai dengan id yang diterima.	
+hapus(id_katPaket):void Operasi ini digunakan untuk menghapus data kategori paket sesuai dengan id yang diterima.	

2.2.1.9 Specific Design Class tambah_katPaket

Tambah_katPaket	<<boundary>>
+tambah(data_katPaket):void Operasi ini digunakan untuk menyimpan data kategori paket yang baru.	

2.2.1.10 Specific Design Class ubah_katPaket

Ubah_katPaket	<<boundary>>
+getDataBy(id_katPaket):void Operasi ini digunakan untuk mengambil data kategori	

paket yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia.

+ubah(id_katPaket,data_katPaket):void

Operasi ini digunakan untuk menyimpan data kategori paket yang telah mendapatkan perubahan.

2.2.1.11 Specific Design Class v_paket

v_paket	<<boundary>>
<p>+tampil():void Operasi ini digunakan untuk menampilkan data paket dengan format tabel.</p> <p>+tambah():void Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data paket baru.</p> <p>+getDataBy(id_paket):void Operasi ini digunakan untuk menampilkan form ubah beserta data paket yang spesifik sesuai dengan id yang diterima.</p> <p>+hapus(id_paket):void Operasi ini digunakan untuk menghapus data pengadaan sesuai dengan id yang diterima.</p>	

2.2.1.12 Specific Design Class tambah_paket

tambah_paket	<<boundary>>
<p>+tambah(data_paket):void Operasi ini digunakan untuk menyimpan data paket yang baru.</p>	

2.2.1.13 Specific Design Class ubah_paket

ubah_paket	<<boundary>>
<pre>+getDataBy(id_paket):void</pre> <p>Operasi ini digunakan untuk mengambil data paket yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia.</p> <pre>+ubah(id_paket,data_paket):void</pre> <p>Operasi ini digunakan untuk menyimpan data paket yang telah mendapatkan perubahan.</p>	

2.2.1.14 Specific Design Class v_harga

v_harga	<<boundary>>
<pre>+tampil():void</pre> <p>Operasi ini digunakan untuk menampilkan data harga dengan format tabel.</p> <pre>+tambah():void</pre> <p>Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data harga baru.</p> <pre>+getDataBy(id_harga):void</pre> <p>Operasi ini digunakan untuk menampilkan form ubah beserta data harga yang spesifik sesuai dengan id yang diterima.</p> <pre>+hapus(id_harga):void</pre> <p>Operasi ini digunakan untuk menghapus data harga sesuai dengan id yang diterima.</p>	

2.2.1.15 Specific Design Class tambah_harga

tambah_harga	<<boundary>>
+tambah(data_harga):void Operasi ini digunakan untuk menyimpan data harga yang baru.	

2.2.1.16 Specific Design Class ubah_harga

ubah_harga	<<boundary>>
+getDataBy(id_harga):void Operasi ini digunakan untuk mengambil data harga yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia. +ubah(id_harga,data_harga):void Operasi ini digunakan untuk menyimpan data harga yang telah mendapatkan perubahan.	

2.2.1.17 Specific Design Class v_refPerson

v_refPerson	<<boundary>>
+tampil():void Operasi ini digunakan untuk menampilkan data person dengan format tabel. +tambah():void Operasi ini digunakan untuk menampilkan form tambah yang digunakan untuk membuat data person baru. +getDataBy(id_person):void Operasi ini digunakan untuk menampilkan form ubah	

beserta data penyaluran yang spesifik sesuai dengan id yang diterima.

+hapus(id_person):void

Operasi ini digunakan untuk menghapus data person sesuai dengan id yang diterima.

2.2.1.18 Specific Design Class tambah_refPerson

tambah_refPerson	<<boundary>>
<p>+tambah(data_penyaluran):void</p> <p>Operasi ini digunakan untuk menyimpan data penyaluran barang yang baru.</p>	

2.2.1.19 Specific Design Class ubah_refPerson

ubah_refPerosn	<<boundary>>
<p>+getDataBy(id_person):void</p> <p>Operasi ini digunakan untuk mengambil data person yang spesifik sesuai dengan id yang diterima dan ditampilkan pada kolom-kolom yang tersedia.</p> <p>+ubah(id_person,data_person):void</p> <p>Operasi ini digunakan untuk menyimpan data person yang telah mendapatkan perubahan.</p>	

2.2.1.20 Specific Design Class v_transaksi

v_transaksi	<<boundary>>
<p>+tampil():void</p> <p>Operasi ini digunakan untuk menampilkan data transkasi</p>	

pemesanan barang dari pihak ketiga dengan format tabel.
`+getDataBy(id_transaksi):void`
 Operasi ini digunakan untuk menampilkan data transaksi sesuai spesifikasi id yang diterima.

2. Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Data `tbl_user`

Nama	Tipe	Panjang	Keterangan
<code>id_user</code>	Integer	11	ID pengguna, Primary Key.
<code>nama_user</code>	Varchar	90	Nama pengguna sistem.
<code>email_user</code>	Varchar	40	Email dari pengguna system yang digunakan pada saat login.
<code>pass_user</code>	Varchar	200	Password dari pengguna system yang digunakan pada saat login
<code>id_role</code>	Integer	20	ID pengguna, Primary Key.

3.1.2 Deskripsi Entitas Data `ref_role`

Nama	Tipe	Panjang	Keterangan
<code>id_role</code>	Integer	11	ID role, Primary

			Key.
nama_role	Varchar	50	Nama hak akses untuk pada pengguna.

3.1.3 Deskripsi Entitas Data harga_tbl

Nama	Tipe	Panjang	Keterangan
id_harga	Integer	11	ID harga, Primary Key.
id_refperson	Integer	11	ID refperson dari ref_person, Foreign key.
id_paket	Integer	11	ID paket dari paket_tbl, Foreign key.
harga	double	-	Harga yang digunakan untuk banyaknya person pada setiap paket wisata.

3.1.4 Deskripsi Entitas Data ref_person

Nama	Tipe	Panjang	Keterangan
id_refperson	Integer	11	ID refperson, Primary Key.
jml_refperson	Integer	11	Banyaknya person pada setiap paket wisata.

3.1.5 Deskripsi Entitas Data kategori_paket_tbl

Nama	Tipe	Panjang	Keterangan
id_kategori_paket	Integer	11	ID kategori paket, Primary Key.
nama_kategori_paket	varchar	50	Nama untuk kategori paket wisata.

3.1.6 Deskripsi Entitas Data paket_tbl

Nama	Tipe	Panjang	Keterangan
id_paket	Integer	11	ID kategori paket, Primary Key.
nama_paket	Varchar	50	Nama untuk paket wisata.
id_kategori_paket	Integer	11	ID kategori paket dari kategori_paket_tbl, Foreign key.
deskripsi_paket	text	-	Deskripsi untuk paket wisata.

3.1.7 Deskripsi Entitas Data tempat_wisata_tbl

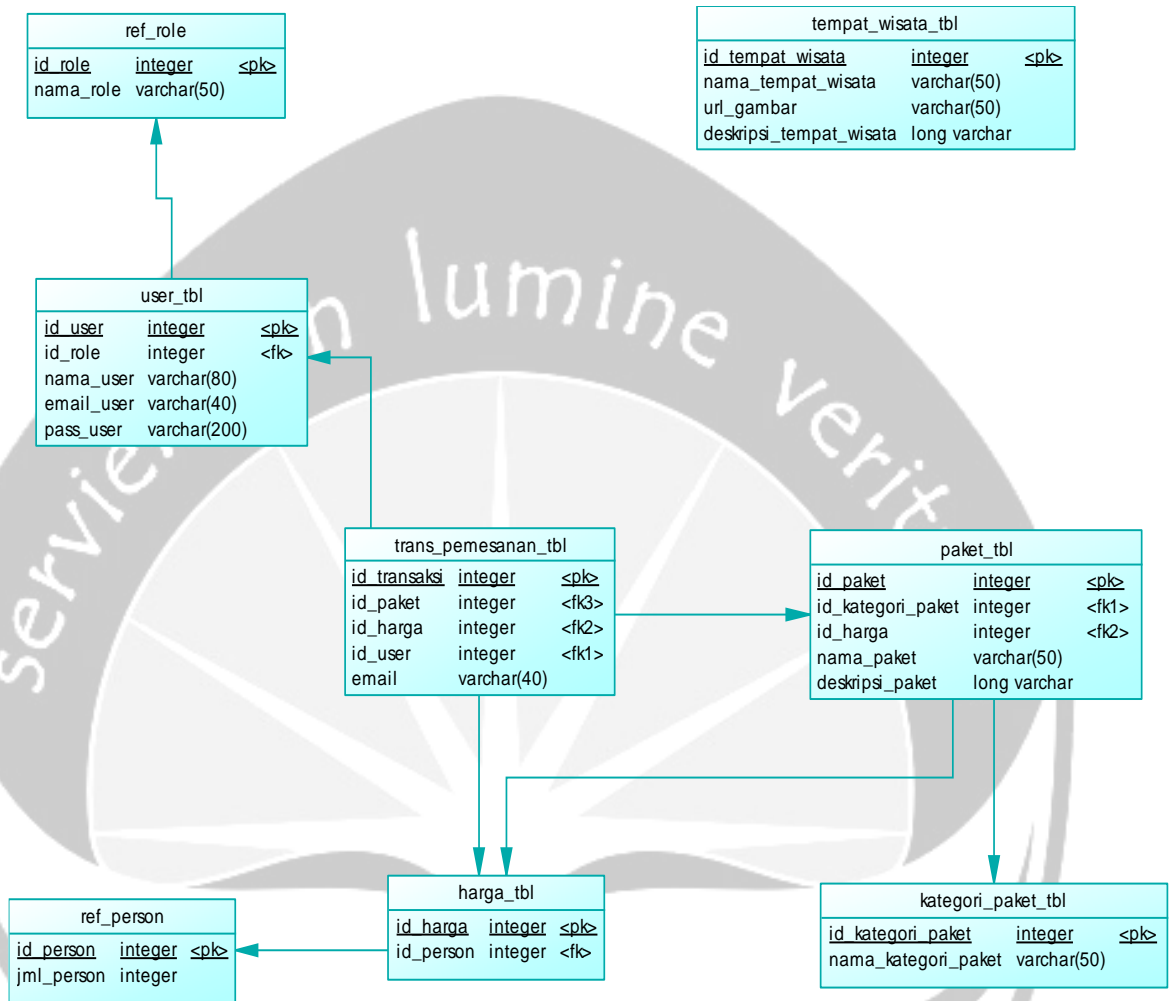
Nama	Tipe	Panjang	Keterangan
id_tempat_wisata	Integer	11	ID tempat wisata, Primary Key.
Nama_tempat_wisata	Varchar	50	nama dari

			tempat tempat wisata.
url_gambar	Varchar	100	url untuk memasukkan foto tempat wisata.
Deskripsi_tempat_wisata	text	50	Deskripsi dari tempat wisata.

3.1.8 Deskripsi Entitas Data trans_pemesanan_tbl

Nama	Tipe	Panjang	Keterangan
id_transaksi	Integer	11	ID transaksi, Primary Key.
id_paket	Integer	11	ID paket dari paket_tbl, Foreign key.
id_harga	Integer	11	ID harga dari harga_tbl, Foreign key.
Id_user	Integer	11	ID user dari user_tbl, Foreign key.
email	Varchar	40	Email untuk pelanggan yang melakukan pemesanan.

3.2 Physical Data Model

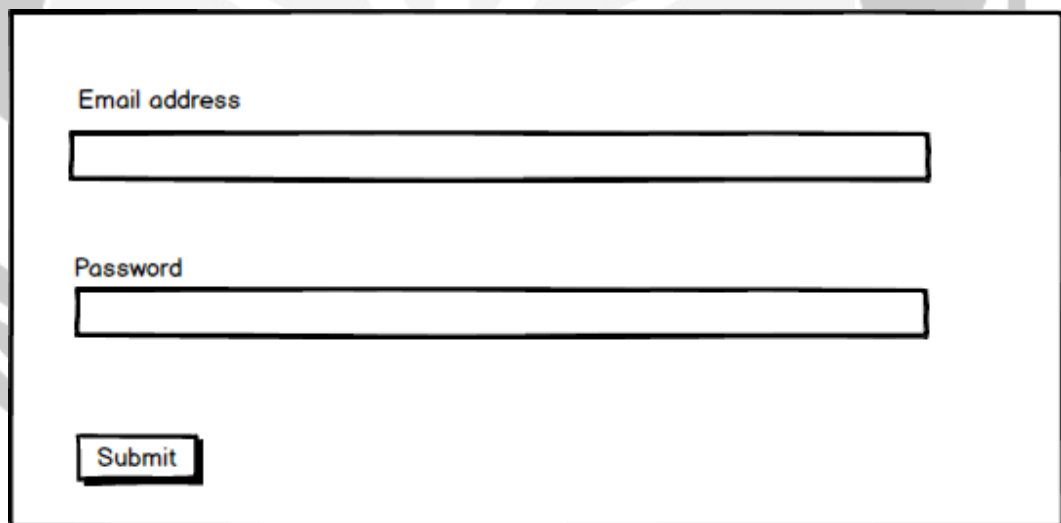


Gambar 3.1 Physical Data Model

4. Perancangan Antarmuka

4.1 Antarmuka Login

Antarmuka ini merupakan antarmuka yang pertama kali muncul ketika aplikasi dijalankan. Antarmuka ini digunakan oleh pengguna untuk mendapatkan hak akses kedalam system. Pengguna diberi 2 masukan yaitu masukan email *address* dan password. Jika username atau password yang dimasukkan oleh pengguna salah maka pengguna tidak dapat masuk kedalam system dan pengguna diberi peringatan bahwa username atau password yang dimasukkan salah. Jika masukan username dan password pengguna benar maka system akan menampilkan halaman utama berdasarkan peranan pengguna yang ada.



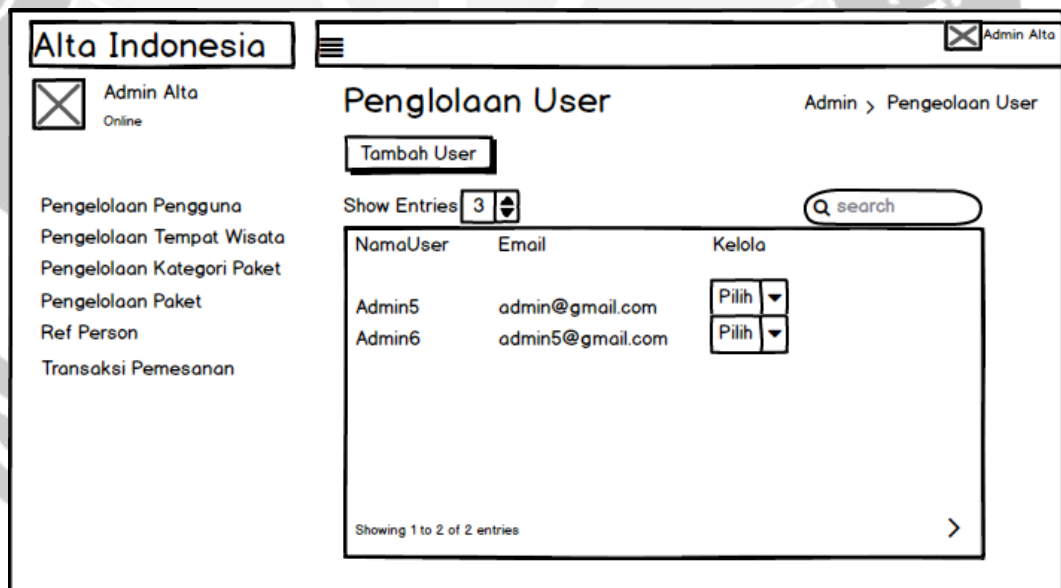
The image shows a login form with two input fields and a submit button. The first field is labeled "Email address" and the second is labeled "Password". Below the fields is a button labeled "Submit".

Gambar 4.1 Antarmuka Login.

4.2 Antarmuka Halaman Utama

Antarmuka ini merupakan antarmuka yang akan tampil ketika pengguna berhasil melakukan login. Antarmuka ini menampilkan menu-menu pengelolaan data yang terdapat pada sebelah kiri halaman. Menu-menu

tersebut adalah fungsi yang terdapat didalam system informasi ini. Fungsi-fungsi yang ada didalam system dapat digunakan sesuai role yang dimiliki oleh pengguna. Pengguna yang memiliki role Super admin akan diberi menu pengelolaan data pengguna. Pengguna yang memiliki role super admin dan admin akan diberi menu untuk mengelola tempat wisata, kategori paket, paket, ref person, dan transaksi pemesanan. Pengguna yang mempunyai role sebagai super admin dapat menambah data user dengan memilih button tambah user, mengubah dan menghapus data user dengan memilih combo box(pilih) pada kolom kelola.



Gambar 4.2 Antarmuka Halaman Utama.

4.3 Antarmuka Pengelolaan Tempat Wisata

Antarmuka ini merupakan antarmuka yang digunakan untuk menampilkan data dengan bentuk tabel. Antarmuka ini digunakan pengguna yang memiliki role

sebagai super admin dan admin pada pengelolaan tempat wisata untuk melakukan tambah, ubah dan hapus data tempat wisata. Untuk melakukan penambahan data tambah tempat wisata pengguna harus memilih button tambah data. Untuk mengubah dan menghapus data tempat wisata pengguna harus memilih combo box (pilih).

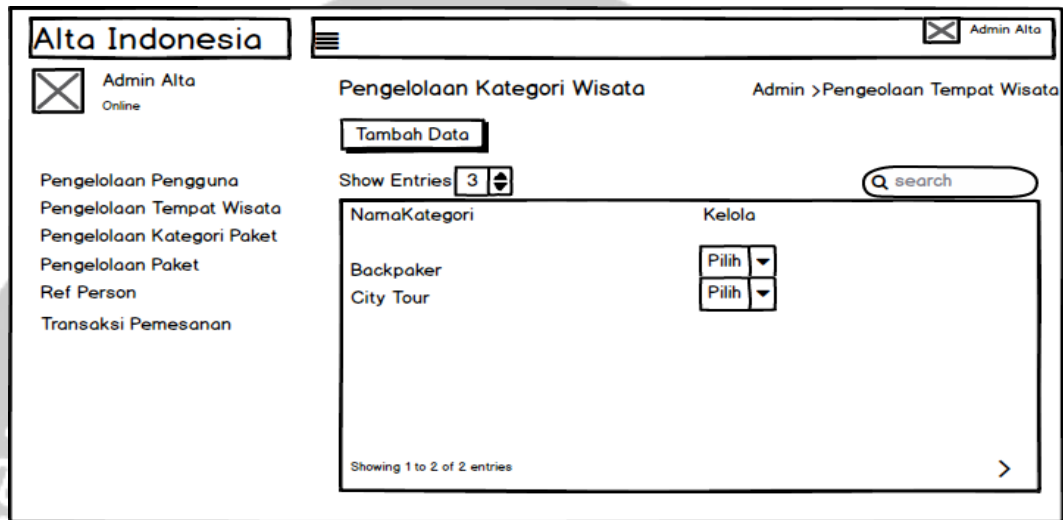


Gambar 4.3 Antarmuka Pengelolaan Tempat Wisata.

4.4 Antarmuka Kategori Wisata

Antarmuka ini merupakan antarmuka yang digunakan pengguna yang mempunyai role sebagai super admin dan admin untuk mengelola tambah, ubah dan hapus kategori wisata. Pengguna dapat melakukan penambahan data kategori wisata dengan memilih

button tambah data. Untuk mengubah dan menghapus data kategori wisata, pengguna harus memilih combo box(pilih) yang terdapat pada kolom kelola

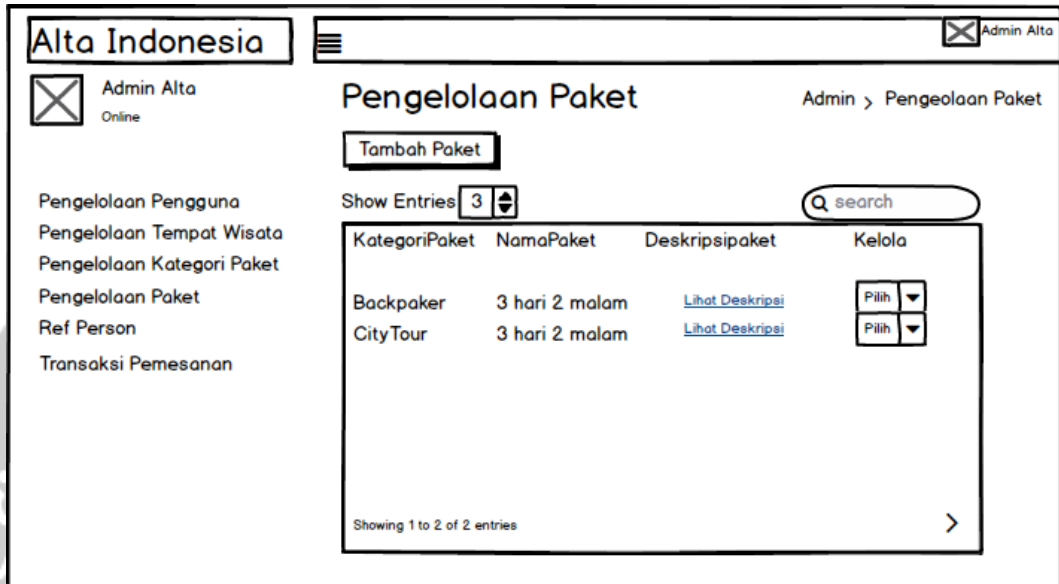


Gambar 4.4 Antarmuka Pengelolaan Kategori Paket Wisata.

4.5 Antarmuka Pengelolaan Paket

Antarmuka ini merupakan antarmuka yang digunakan oleh pengguna yang memiliki role sebagai super admin dan admin untuk menambah, mengelola harga, ubah, dan hapus paket. Pengguna dapat menambah data paket dengan memilih button tambah paket. Untuk mengubah, mengelola harga dan menghapus

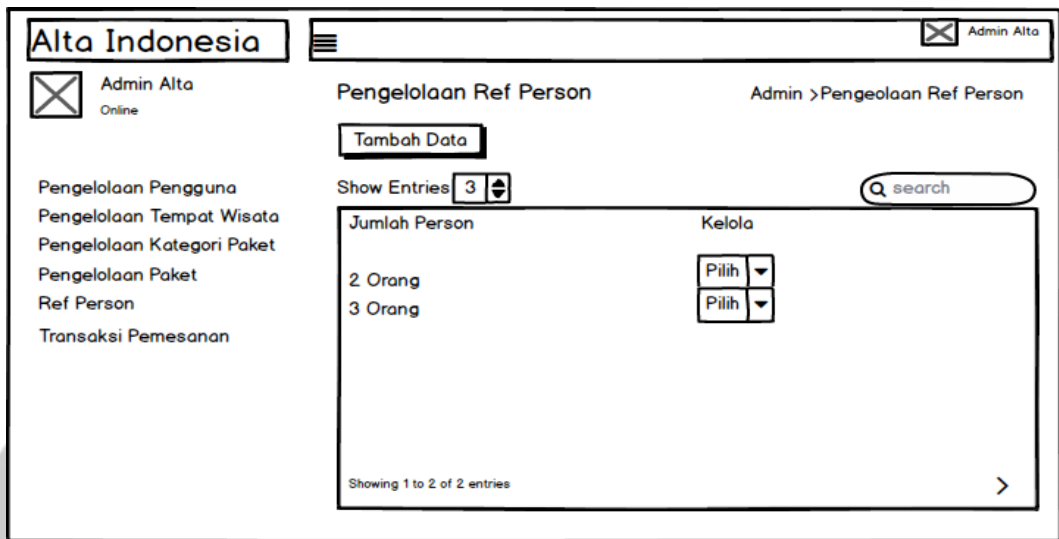
data paket, pengguna harus memilih combo box (pilih) pada kolom kelola.



Gambar 4.5 Antarmuka Pengelolaan Paket.

4.6 Antarmuka Pengelolaan Ref Person

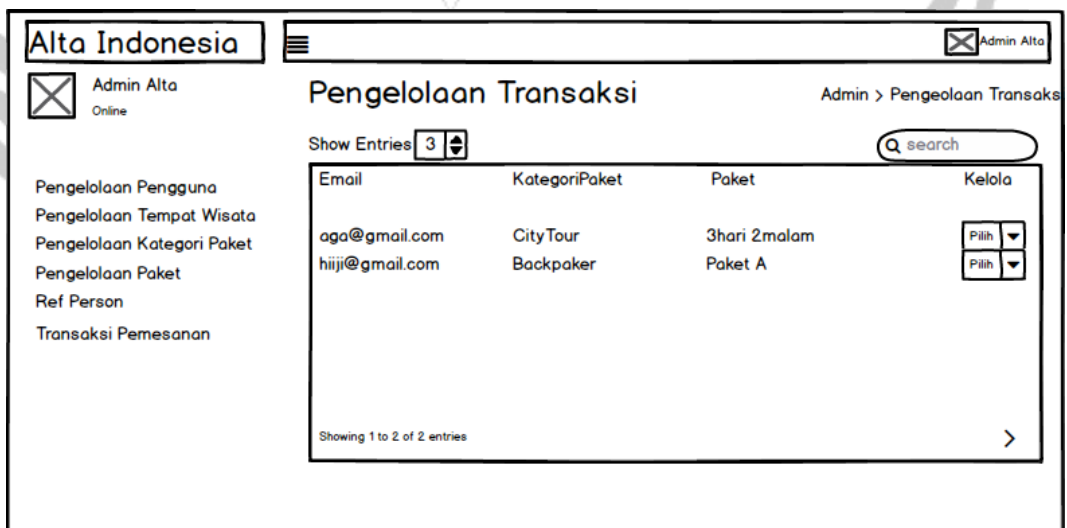
Antarmuka ini merupakan antarmuka yang digunakan oleh pengguna yang memiliki role sebagai super admin dan admin untuk melakukan tambah, ubah dan hapus ref person. Pengguna dapat melakukan tambah ref person dengan memilih button tambah data, dan pengguna bisa melakukan ubah dan hapus data dengan memilih combo box (pilih) yang terdapat pada kolom kelola.



Gambar 4.6 Antarmuka Pengelolaan Ref Person.

4.7 Antarmuka Pengelolaan Transaksi Pemesanan

Antarmuka ini merupakan antarmuka yang digunakan oleh pengguna yang memiliki role sebagai super admin dan admin untuk melihat data transaksi pemesanan yang sudah dipesan oleh pelanggan.



Gambar 4.7 Antarmuka Pengelolaan Transaksi Pemesanan.

4.8 Antarmuka Pengelolaan Harga

Antarmuka ini merupakan antarmuka yang digunakan oleh pengguna yang memiliki role sebagai super admin dan admin untuk melakukan tambah, ubah dan hapus harga dari paket wisata. Untuk melakukan tambah data pengguna harus memilih tombol button tambah data. Jika pengguna ingin melakukan ubah dan hapus data, pengguna harus memilih combo box(pilih) yang berisikan ubah atau hapus pada kolom kelola.

Jumlah Person	HargaPaket	Kelola
2 Orang	300.000	Pilih
3 Orang	400.000	Pilih

Gambar 4.8 Antarmuka Pengelolaan Harga.