

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan dari pembahasan-pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan dari Tugas Akhir ini :

1. Pengembangan Sistem Informasi Desa Wisata di Daerah Istimewa Yogyakarta berbasis Lokasi yang dikembangkan telah berhasil dibuat dan berjalan baik untuk menampilkan posisi dan informasi desa wisata di wilayah Yogyakarta.
2. Pengembangan Sistem Informasi Desa Wisata di Daerah Istimewa Yogyakarta berbasis Lokasi dapat menampilkan informasi detail berupa foto, video dan keterangan dalam bentuk informasi tertulis.
3. Secara keseluruhan dari kuesioner yang dibagikan kepada 33 responden, berpendapat bahwa aplikasi ini baik dan fungsinya dapat berjalan sesuai dengan yang diharapkan.
4. Masih ada responden yang menginginkan antarmuka dengan variasi warna yang lebih menarik.

6.2 Saran

Beberapa saran dan masukan yang dapat disampaikan penulis terhadap Pengembangan Sistem Informasi Desa Wisata di Daerah Istimewa Yogyakarta berbasis Lokasi ini dimasa yang akan datang adalah :

1. Pengembangan pada platform selain Andorid.
2. Menambahkan fitur *Augmented Reality* dengan *3D Model Design* agar tampilan lebih atraktif.

3. Menambahkan data lokasi untuk tempat-tempat lain yang butuh informasi lebih lengkap seperti wisata kuliner tradisional, pusat oleh-oleh dan sentra batik.



DAFTAR PUSTAKA

- Adrisijanti, I., 2007. *Kota Yogyakarta Sebagai Kawasan Pusaka Budaya: Potensi dan Permasalahannya (The City of Yogyakarta as Cultural Heritage Region, Potencies and Problems*. Yogyakarta, Centre of Yogyakarta Historic Preservation and Traditional Values.
- Arliani, S., 2009. *Ibing Penca Gaya Cikalong pada Paguron Benteng Ksatria di Kabupaten Cianjur*. s.l., FPBS UPI.
- Chen,, C.-W., Montelatici, E. & Crisan , M., 2009. Perivascular multi-lineage progenitor cells in human organs: regenerative units, cytokine sources or both?. p. 429-434.
- Craig, K. & Balkun, M. M., 2011. Teaching On The Virtuality Continuum: Augmented Reality in the. *Transformations: The Journal of Inclusive Scholarship and Pedagogy*, Volume 10(1), pp. 100-145.
- Craig, K. & Mary McAleer, B., 2011. Teaching on the Virtuality Continuum : Augmented Reality in the Classroom. *Transformations* 22.1 , pp. 100-113,145.
- Dunleavy , P., 2014. The impact of the social sciences: how academics and their research make a difference. *Sage Publications*.
- Enyedy, N., Danish, J. A., Delacruz, G. & Kumar, M., 2012. Learning physics through play in an augmented reality environment. 7(3), pp. 347-378.
- Febian, 2015. *Win Poin*. [Online]
Available at: <http://winpoin.com/market-share-android/>
[Accessed 10 1 2016].
- Ghazael, F. & Mahmoudi, S. S., 2012. Talking Business Card Using Augmented Reality. *International Journal of Computer Science and Information Security*, 10(5), pp. 52-58.
- Giap Weng, E. N., Ren, Parhizkar & Lashkari, 2011. Mobile Phone Augmented. *International Journal of*

Computer Science and Information Security, 9(5), pp. 154-164.

Hollere, T. H. & Feiner, S. K., 2004. Mobile Augmented Reality. *Telegoinformation*, Volume 9.

Intana, L., 2012. *Swa Online*. [Online]
Available at: <http://swa.co.id/technology/tahun-ini-pengguna-smartphone-naik-3-kali-lipat>
[Accessed 18 June 2015].

Johnson, C., 2011. Gillard, Rudd and Labor Tradition. *Australian Journal of Politics & History*, 57(4), pp. 562-579.

Kanchanawong, P., Chodok, C. & Songsak, P., 2012. E-Tourism Usage Pattern of Tourism Business in Chiang Mai, Thailand. *Chinese Bussines Review*, Volume 11, pp. 98-193.

Kenteris, M., Gavalas, D. & Economou, D., 2011. Electronic mobile guides: a survey, Personal and Ubiquitous Computing. Volume 15, p. 97-111.

Kessler, K. & Thomson, L. A., 2010. The embodied nature of spatial perspective taking: embodied transformation versus sensorimotor interference. *Cognition*, 114(1), pp. 72-88.

Papaioannou, G. & Deliyiannis, I., 2013. Augmented Reality for Archaeological Environments on mobile devices: a novel open framework. *International Workshop on Virtual Archaeology, Museums and Cultural Tourism*, pp. 25-28.

Silva, O. R., 2003. Taxonomy of Mixed Rality Visual Displays. *Introduction to Augmented Reality*.

Umlauft, M. & Anegg, H., 2003. Usablility of a Location Based UMTS application.. *elektro- & informationstechnik*, 2(2), p. 61-65.

Uzoka, F. M. & Akinuwesi, B., 2009. A Framework of Web Based Fuzzy Expert System for Managing Tourism Information. *Georgian Electronic Scientific Journal: Computer Science and Telecommunications*, 20(3), pp. 77-89.

Yoon, S. & Wang, J., 2014. Making the invisible visible in science museums through augmented reality devices. *Tech Trends*, 58(1), pp. 49-55.



DPPL

**DESKRIPSI PERANCANGAN PERANGKAT
LUNAK**

**PENGEMBANGAN SISTEM INFORMASI DESA WISATA
YOGYAKARTA DENGAN *AUGMENTED REALITY* BERBASIS
LOKASI (JOGJANESIA)**

Untuk
:

**Fakultas Teknologi Industri
Universitas Atma Jaya Yogyakarta**

Dipersiapkan oleh:

Samuel Ady Sanjaya / 12. 07. 07007

**Program Studi Teknik
Informatika Fakultas Teknologi
Industri Universitas Atma Jaya
Yogyakarta**

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL - JOGJANESIA		
		Revisi		1 / 60

DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	
B	
C	
D	
E	
F	

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

DAFTAR HALAMAN PERUBAHAN

Program Studi Teknik Informatika	DPPL – JOGJANESIA	2/ 56
----------------------------------	-------------------	-------

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

Halaman	Revisi	Halaman	Revisi

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 JOGJANESIA dikembangkan dengan tujuan untuk:

1. Menangani penyajian informasi desa wisata informasi penginapan, dan informasi penyewaan kendaraan.
2. Menangani pencarian lokasi desa wisata dengan augmented reality.
3. Menangani navigasi rute perjalanan dari lokasi pengguna ke lokasi desa wisata.
4. Menangani pengelolaan data desa wisata.
5. Menangani pengelolaan data hotel.
6. Menangani pengelolaan data rental kendaraan.

Dan aplikasi ini dapat berjalan pada perangkat berbasis android yang memiliki GPS dan kamera.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
JOGJANESIA	Perangkat lunak mobile untuk Desa Wisata dengan Augmented Reality.
SKPL - JOGJANESIA-XXX	Kode yang merepresentasikan kebutuhan pada JOGJANESIA[Nama Aplikasi Mobile] dimana XXX merupakan nomor fungsi produk.
Augmented Reality	teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata.
Desa Wisata	kawasan pedesaan yang memiliki beberapa karakteristik khusus untuk menjadi daerah tujuan wisata dan merupakan suatu bentuk integrasi antara atraksi, akomodasi dan fasilitas pendukung yang disajikan dalam suatu struktur kehidupan masyarakat yang menyatu dengan tata cara dan tradisi yang berlaku.

Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World wide web.
Framework	Suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks.
PHP	Hypertext Preprocessor merupakan bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Android	adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

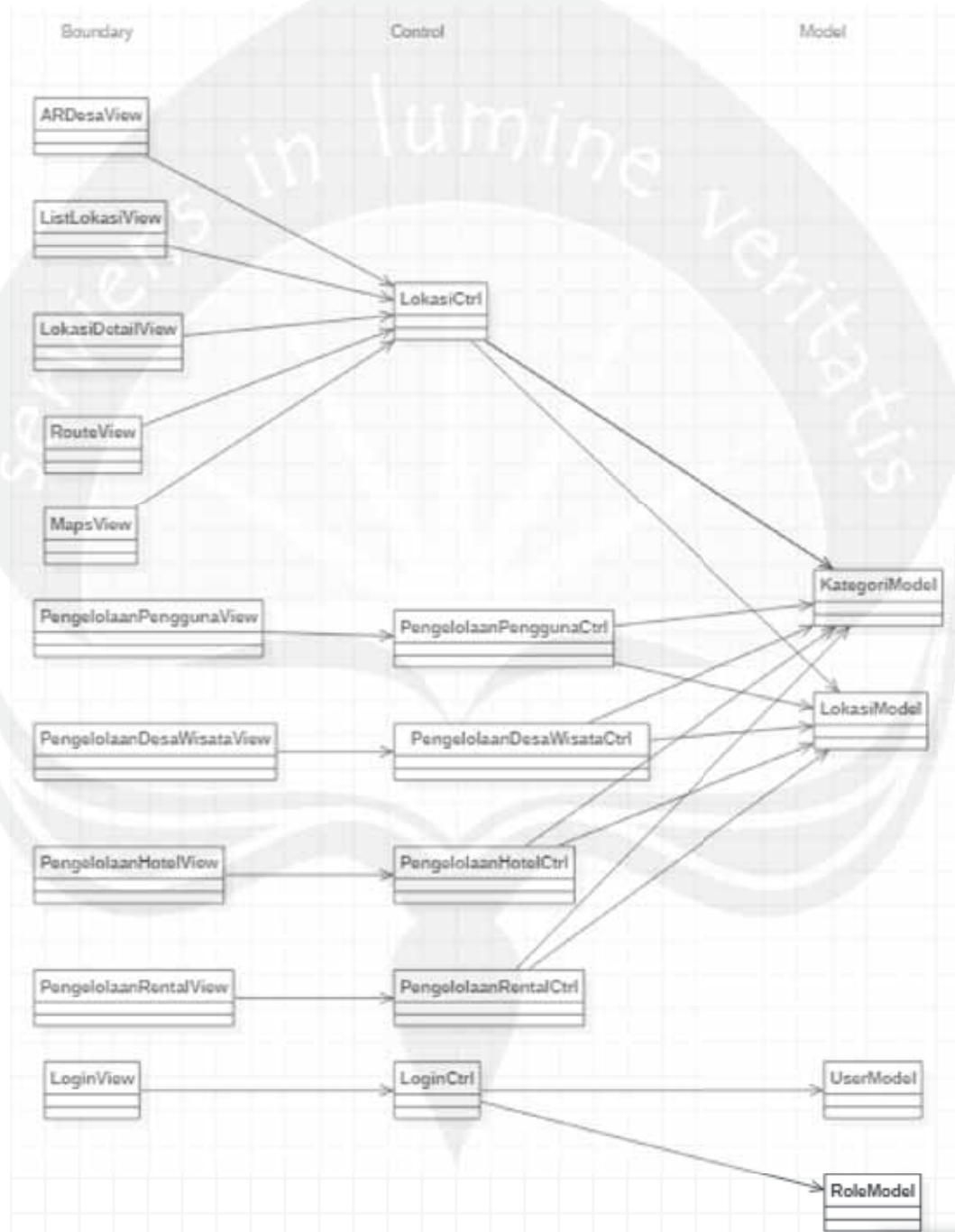
1. Yosafat Novianto Ade Putra, *Deskripsi Perancangan Perangkat Lunak (DPPL) WIYATA*, Universitas Atma Jaya Yogyakarta.

2. Yosep Wihelmus Nabu, *Deskripsi Perancangan Perangkat Lunak (DPPL) VISITMABAR*, Universitas Atma Jaya Yogyakarta.
3. Samuel Ady Sanjaya, 2016, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) JOGJANESIA*, Universitas Atma Jaya Yogyakarta.



2 Perancangan Sistem

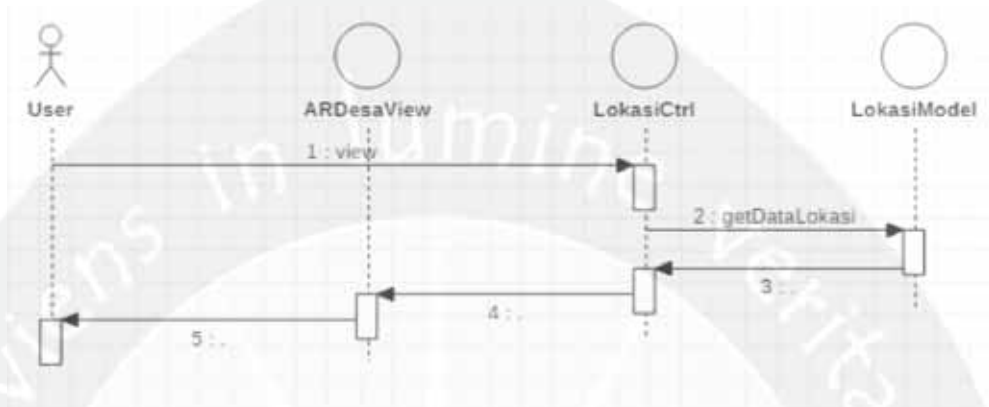
2.1 Perancangan Arsitektur



2.2 Perancangan Rinci

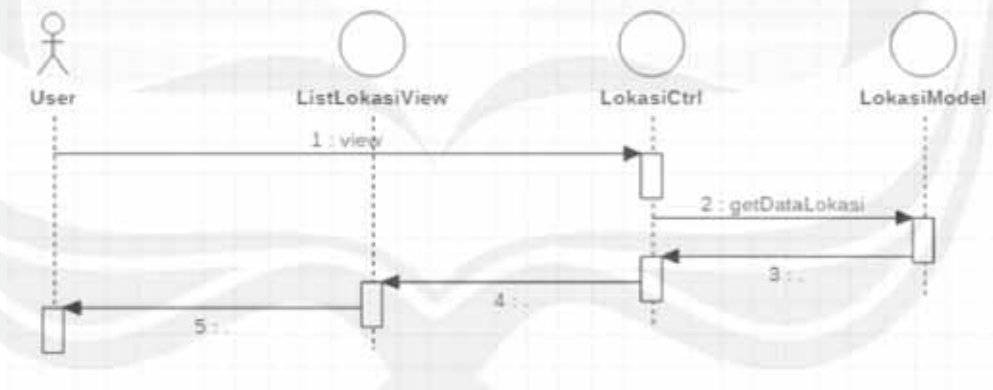
2.2.1. Sequence Diagram

2.2.1.1 Fungsi Show Desa Wisata dengan AR



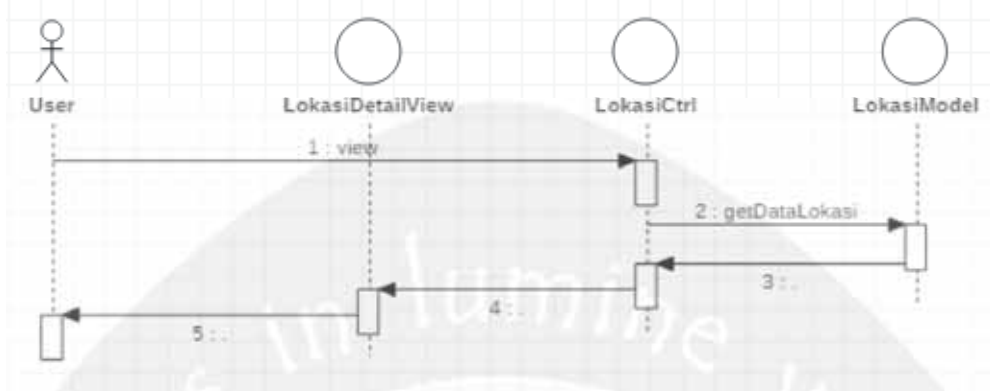
Gambar 2.1 Sequence Diagram : Fungsi Show Desa Wisata dengan AR

2.2.1.2 Menampilkan Lokasi dalam List View



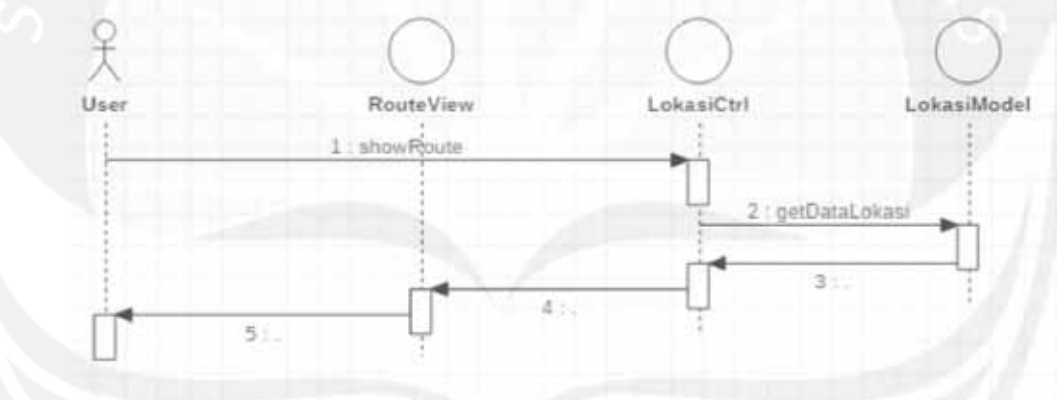
Gambar 2.2 Sequence Diagram : Fungsi Show Lokasi dalam List

2.2.1.3 Menampilkan Detail Lokasi View



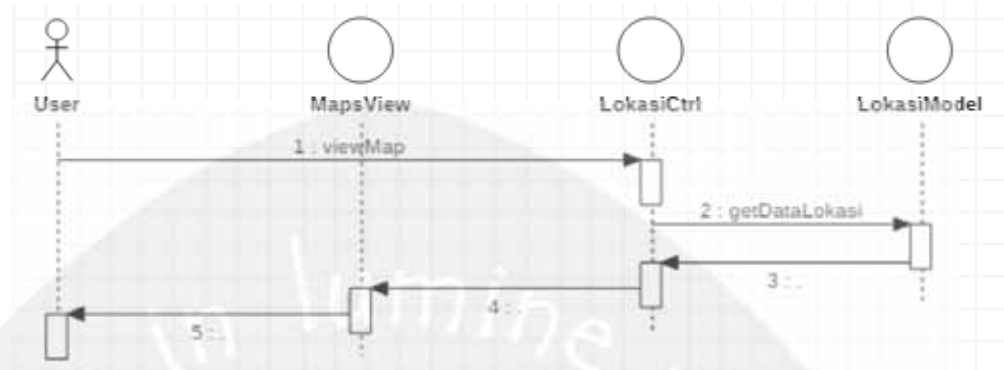
Gambar 2.3 Sequence Diagram : Fungsi Show Detail Lokasi

2.2.1.4 Menampilkan Rute perjalanan



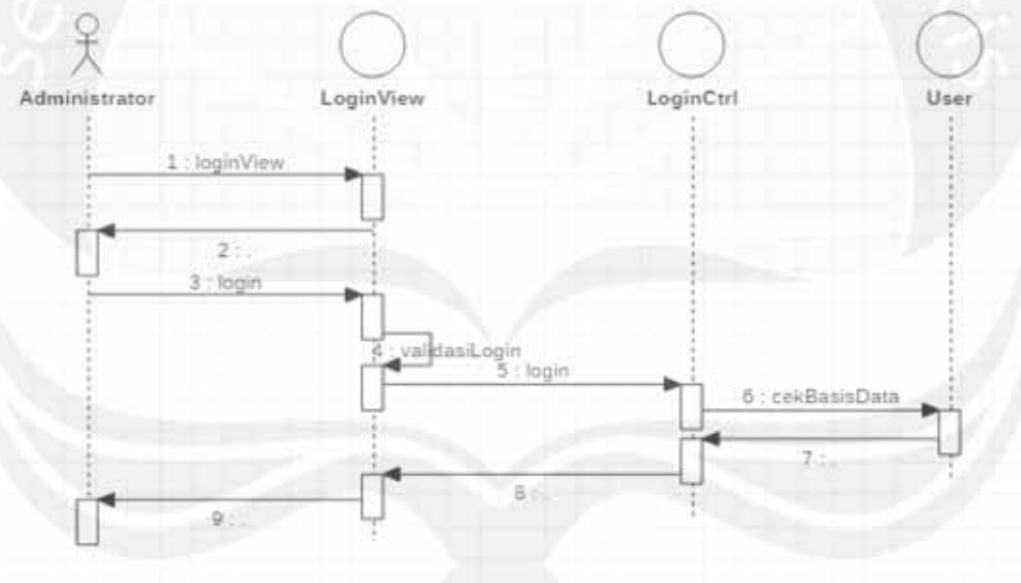
Gambar 2.4 Sequence Diagram : Fungsi Show Route

2.2.1.5 Menampilkan Lokasi dalam Maps



Gambar 2.5 Sequence Diagram : Fungsi Show Lokasi dalam Maps

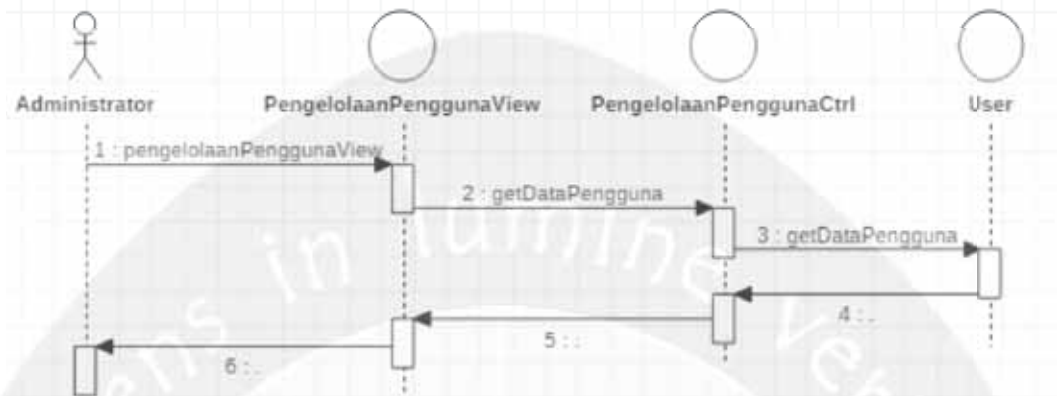
2.2.1.6 Fungsi Login



Gambar 2.6 Sequence Diagram : Fungsi Login

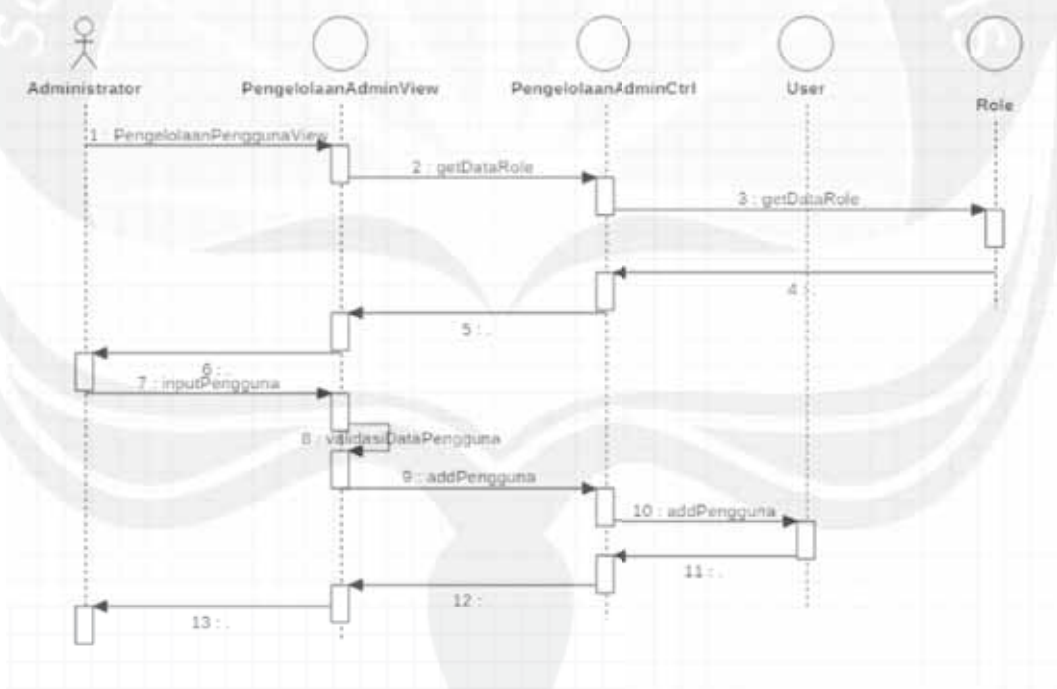
2.2.1.7 Pengelolaan Pengguna

2.2.1.7.1 Fungsi Show Pengguna



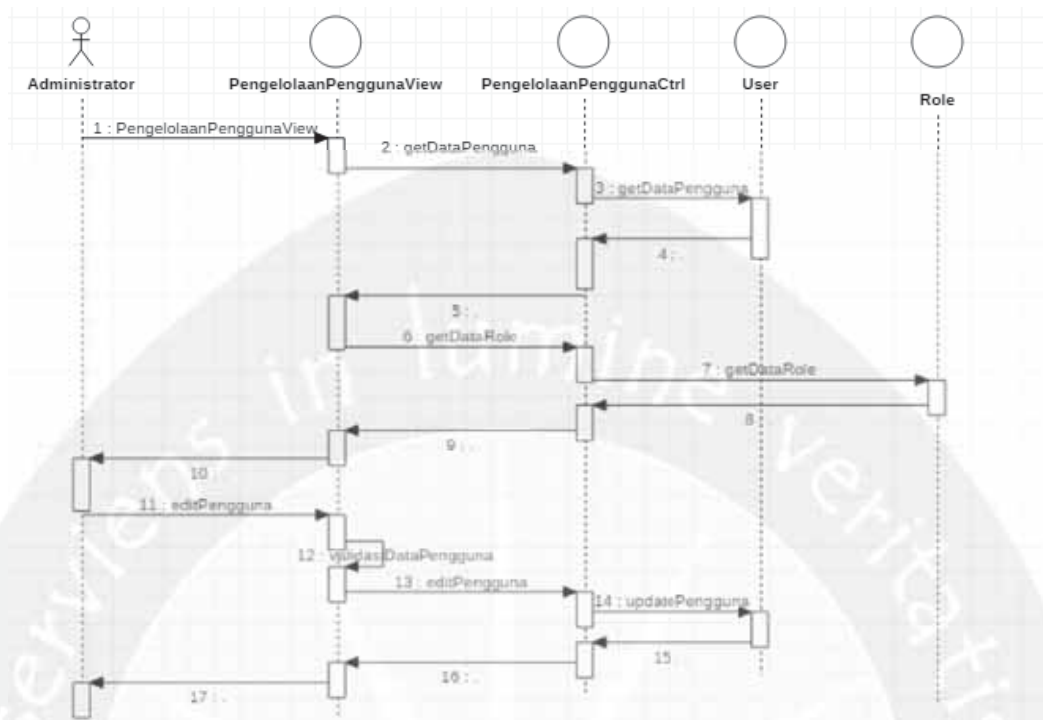
Gambar 2.7 Sequence Diagram : Fungsi Show Pengguna

2.2.1.7.2 Fungsi Add Pengguna



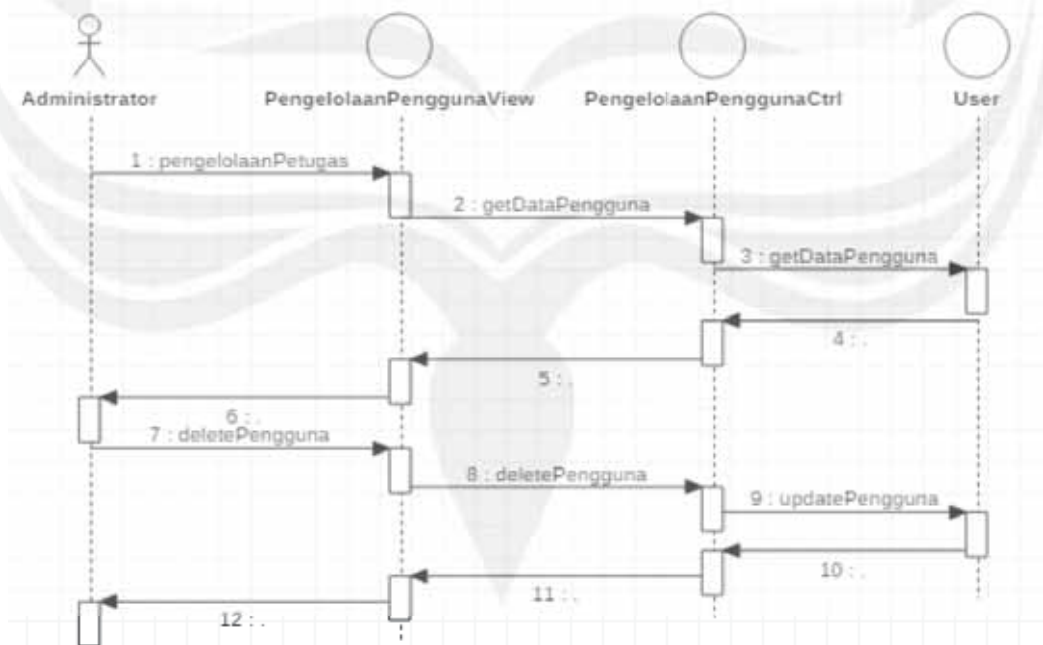
Gambar 2.7 Sequence Diagram : Fungsi Add Pengguna

2.2.1.7.3 Fungsi Edit Pengguna



Gambar 2.8 Sequence Diagram : Fungsi Edit Pengguna

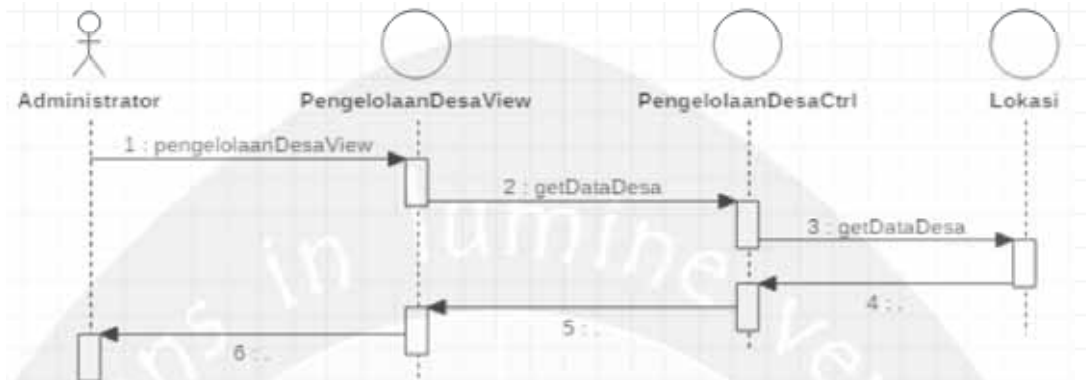
2.2.1.7.4 Fungsi Delete Pengguna



Gambar 2.9 Sequence Diagram : Fungsi Delete Pengguna

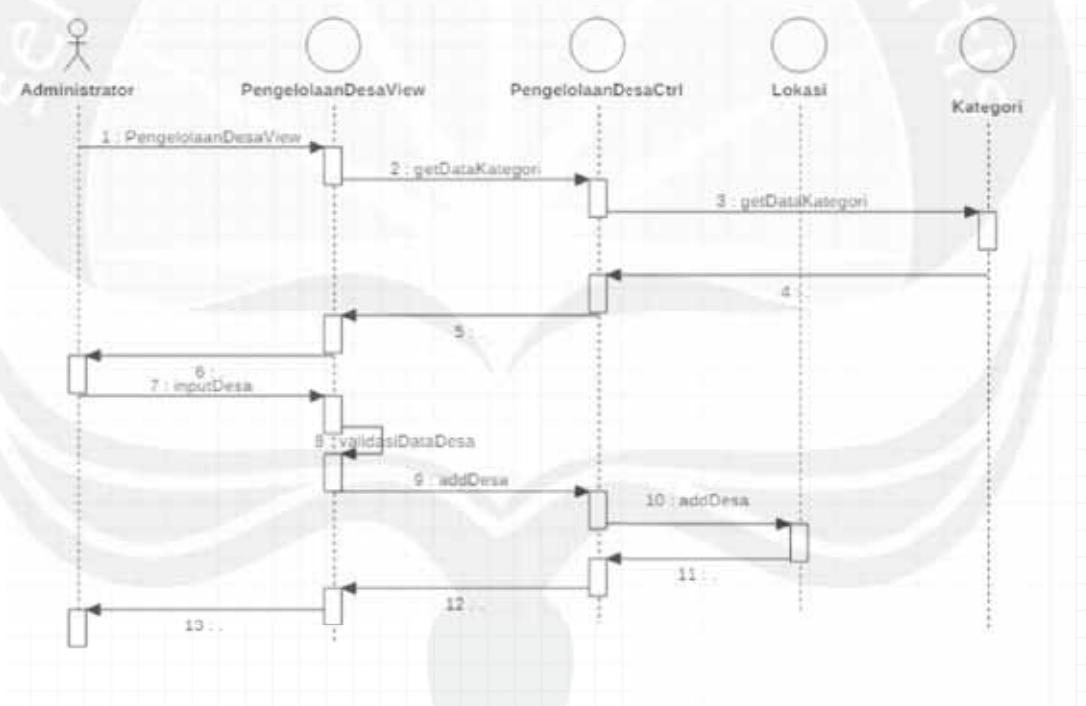
2.2.1.8 Pengelolaan Desa Wisata

2.2.1.8.1 Fungsi Show Desa Wisata



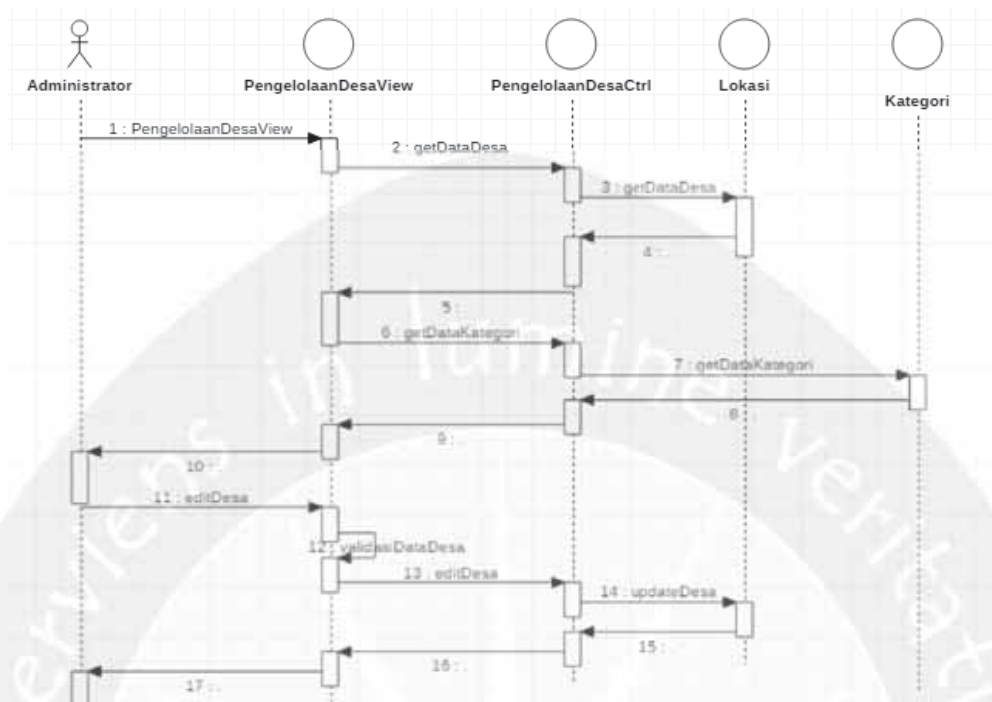
Gambar 2.10 Sequence Diagram : Fungsi Show Desa Wisata

2.2.1.8.2 Fungsi Add Desa Wisata



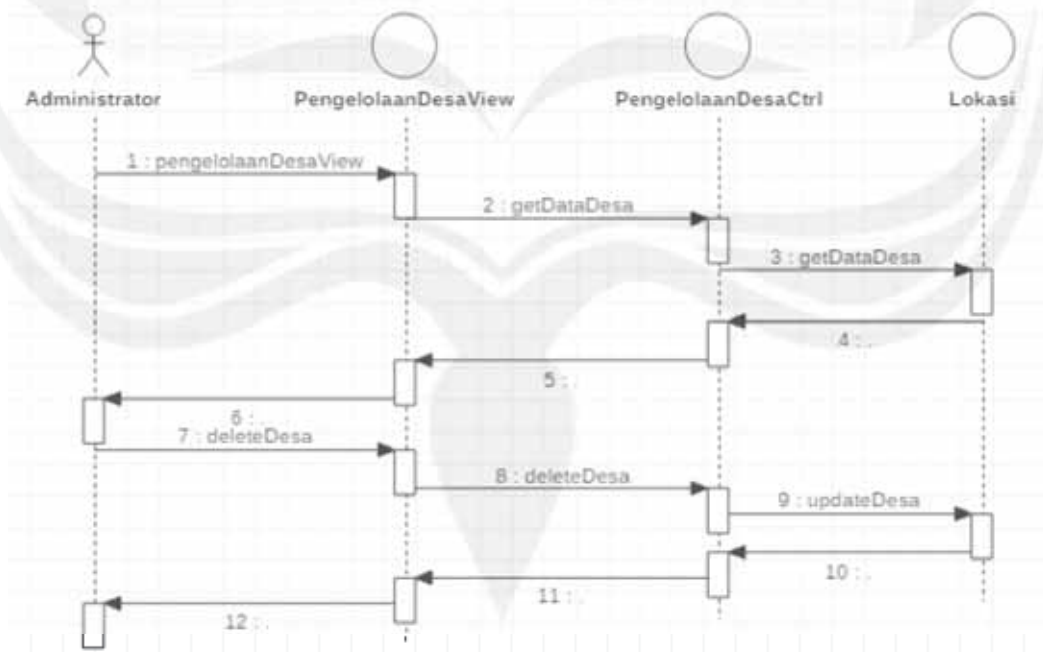
Gambar 2.11 Sequence Diagram : Fungsi Add Desa Wisata

2.2.1.8.3 Fungsi Edit Desa Wisata



Gambar 2.12 Sequence Diagram : Fungsi Edit Desa Wisata

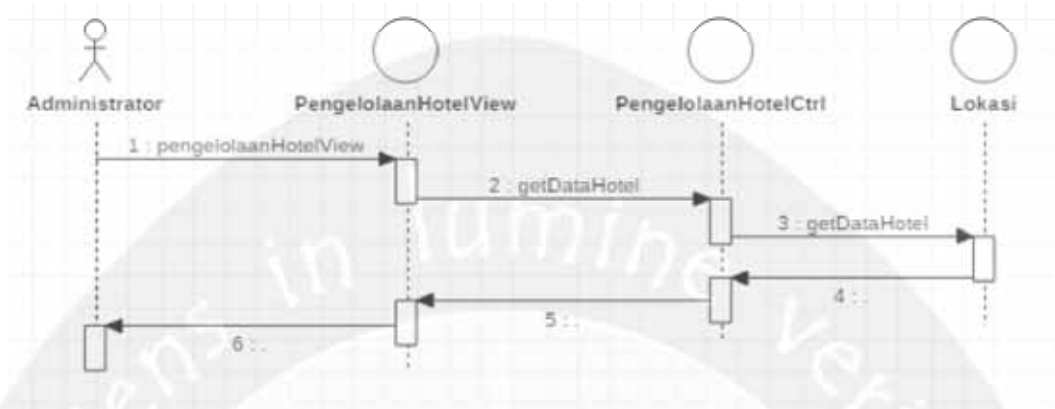
2.2.1.8.4 Fungsi Delete Desa Wisata



Gambar 2.13 Sequence Diagram : Fungsi Delete Desa Wisata

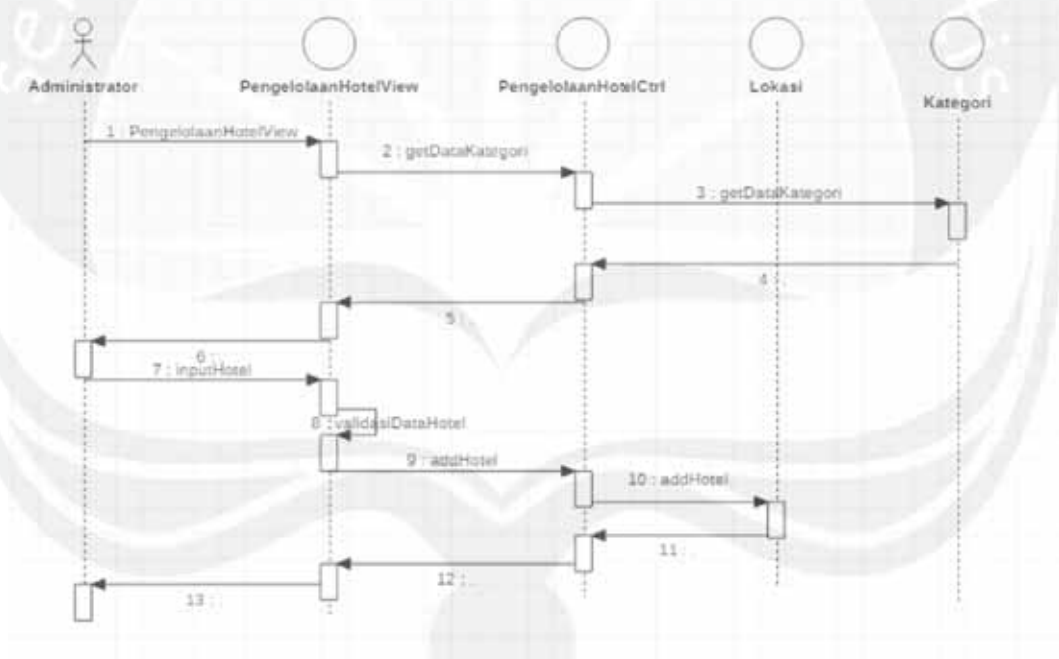
2.2.1.9 Pengelolaan Hotel

2.2.1.9.1 Fungsi Show Hotel



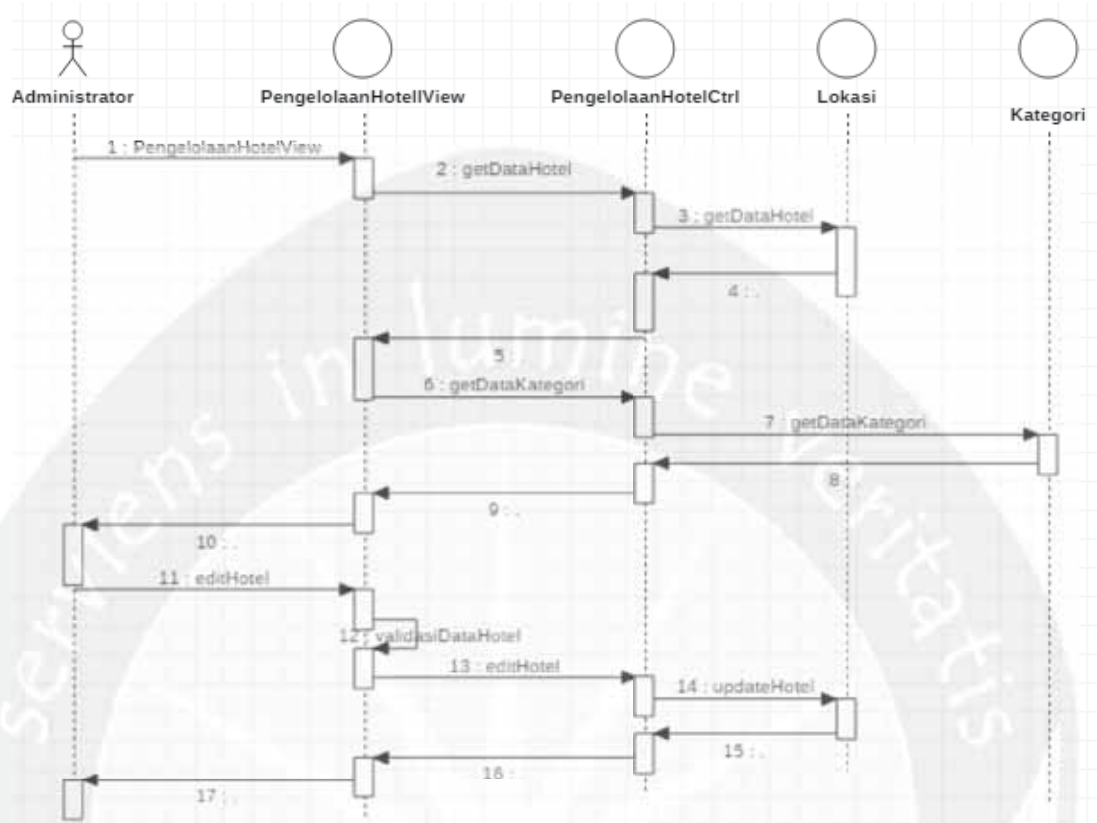
Gambar 2.14 Sequence Diagram : Fungsi Show Hotel

2.2.1.9.2 Fungsi Add Hotel



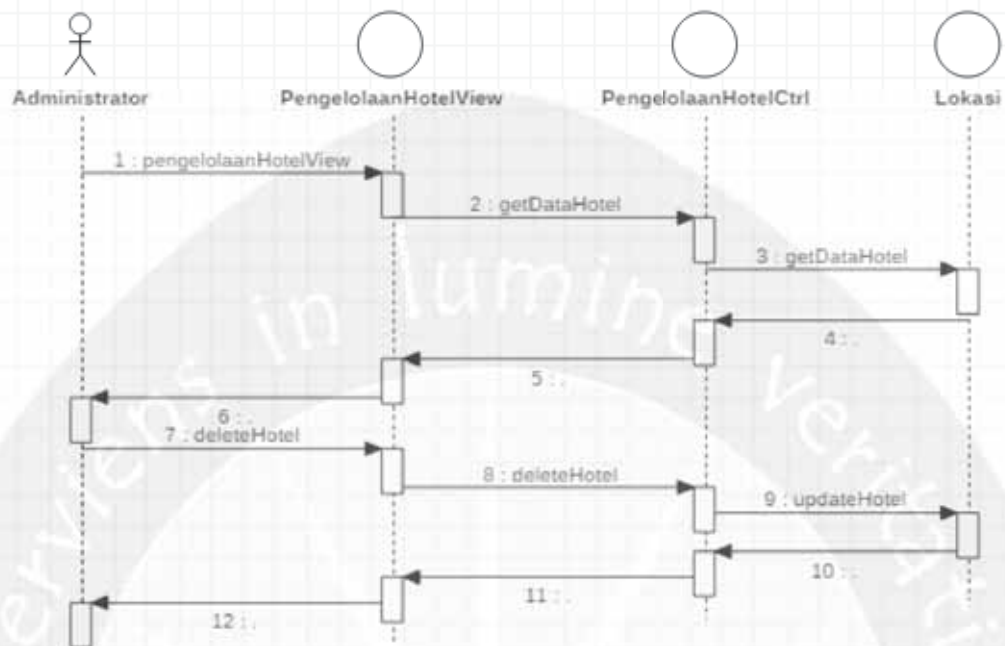
Gambar 2.15 Sequence Diagram : Fungsi Add Hotel

2.2.1.9.3 Fungsi Edit Hotel



Gambar 2.16 Sequence Diagram : Fungsi Edit Hotel

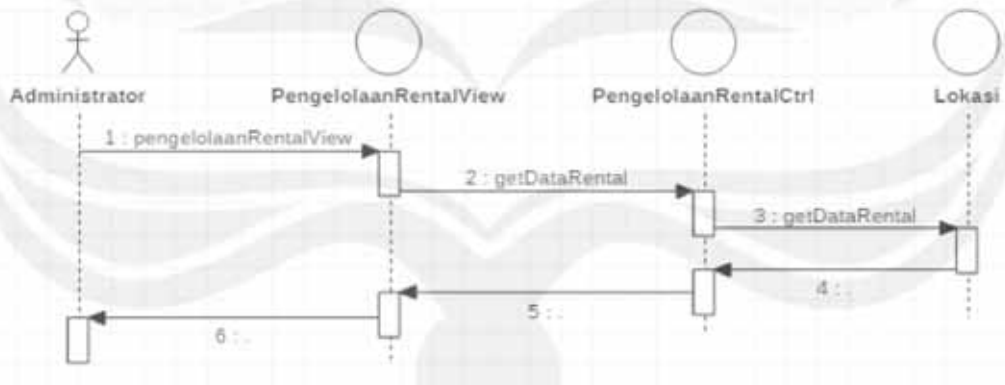
2.2.1.9.4 Fungsi Delete Hotel



Gambar 2.17 Sequence Diagram : Fungsi Delete Hotel

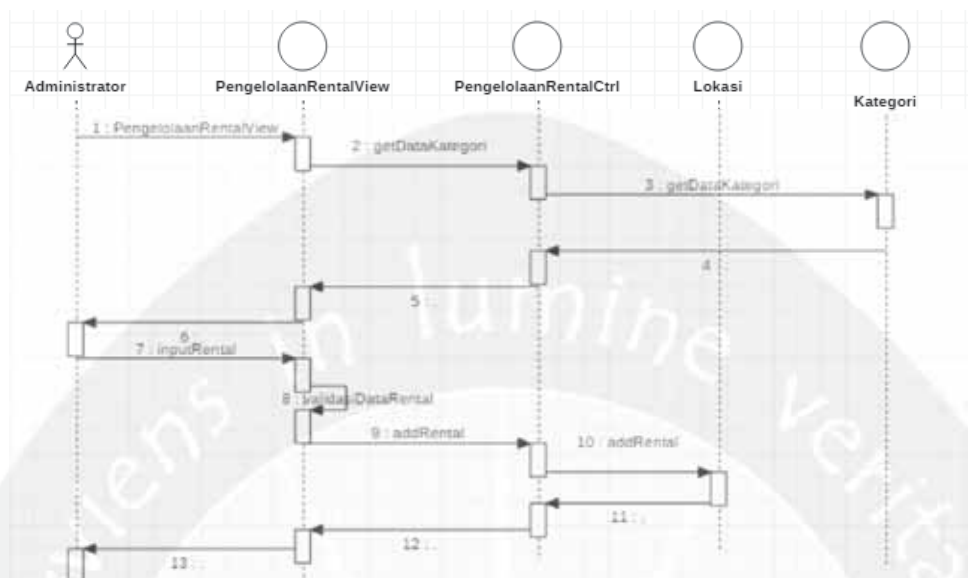
2.2.1.10 Pengelolaan Rental Kendaraan

2.2.1.10.1 Fungsi Show Rental Kendaraan



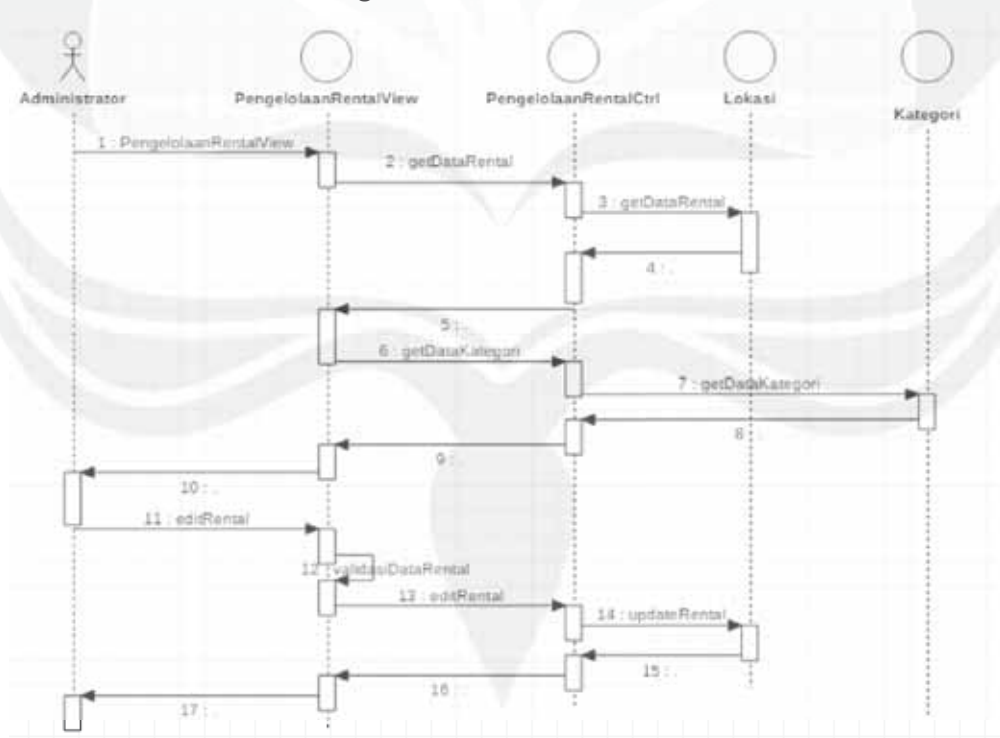
Gambar 2.18 Sequence Diagram : Fungsi Show Rental Kendaraan

2.2.1.10.2 Fungsi Add Rental Kendaraan



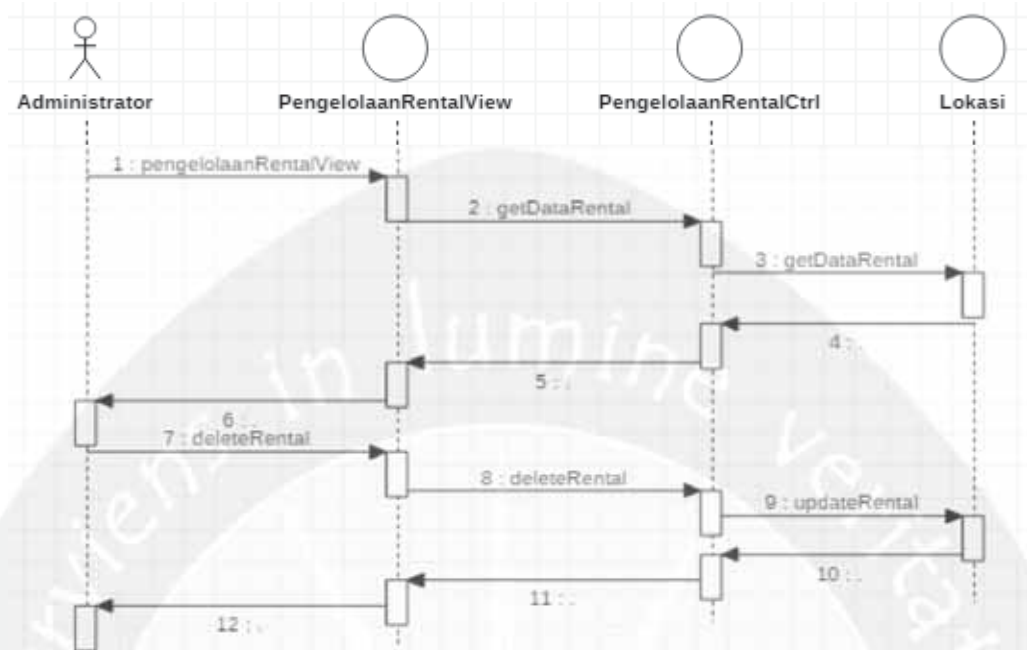
Gambar 2.19 Sequence Diagram : Fungsi Add Rental Kendaraan

2.2.1.10.3 Fungsi Edit Rental Kendaraan



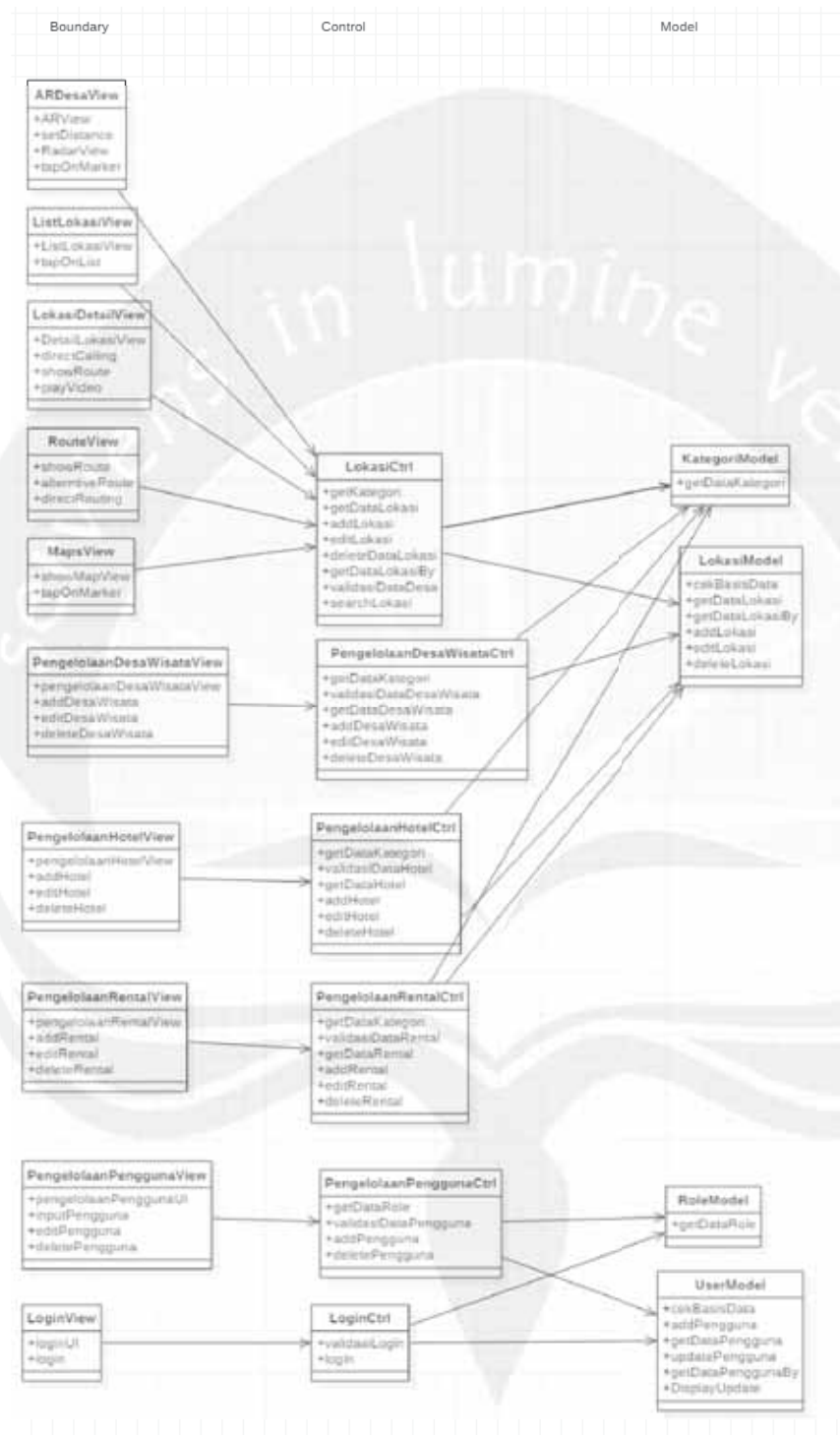
Gambar 2.20 Sequence Diagram : Fungsi Edit Rental Kendaraan

2.2.1.10.4 Fungsi Delete Rental Kendaraan



Gambar 2.21 Sequence Diagram : Fungsi Delete Rental Kendaraan

2.2.2 Class Diagram



Gambar 2.22 Class Diagram

2.2.3 Class Diagram Specific Descriptions

2.2.3.1 Specific Design Class ARDesaView

ARDesaView	<<boundary>>
ARView() Fungsi ini digunakan untuk menampilkan desa wisata sekitar dalam bentuk marker setDistance() Fungsi ini digunakan untuk mengatur jarak jangkauan lokasi yang akan ditampilkan radarView() Fungsi ini digunakan untuk melihat keseluruhan lokasi yang ada dalam jangkauan untuk memudahkan pengguna menemukan lokasi disekitarnya onMarkerTap() Fungsi ini digunakan untuk menampilkan detail desa wisata yang dipilih melalui marker	

2.2.3.2 Specific Design Class ListLokasiView

ListLokasiView	<<boundary>>
ListLokasiView() Fungsi ini digunakan untuk menampilkan semua lokasi yang ada di database dan menampilkannya berdasarkan kategori lokasi tapOnList() Fungsi ini digunakan untuk menuju halaman detail Lokasi	

2.2.3.3 Specific Design Class RouteView

RouteView	<<boundary>>
showRoute() Fungsi ini digunakan untuk menampilkan rute dari posisi Pengguna menuju ke Lokasi yang akan dituju	
alternativeRoute() Fungsi ini digunakan untuk memberikan pilihan alternative untuk jalan menuju lokasi	
directRouting() Fungsi ini digunakan untuk menunjukan jalan secara langsung dari sudut pandang pengguna setelah memilih salah satu jalur alternatif	

2.2.3.4 Specific Design Class MapView

MapView	<<boundary>>
ShowMapView() Fungsi ini digunakan untuk menampilkan lokasi berdasarkan kategori yaitu desa wisata, hotel atau rental kendaraan dengan menggunakan marker dalam peta.	
tapOnMarker() Fungsi ini digunakan untuk menampilkan keterangan singkat dalam bentuk toast ketika pengguna men-tap salah satu marker lokasi	

2.2.3.5 Specific Design Class PengelolaanDesaWisataView

PengelolaanDesaWisataView	<<boundary>>
---------------------------	--------------

`pengelolaanDesaWisataView()`
 sebagai default konstruktor dimana semua atribut pada class ini diinisialisasi.

`addDesaWisata()`
 digunakan untuk menambahkan data desa wisata ke dalam database.

`deleteDesaWisata()`
 digunakan untuk menghapus data desa wisata yang ada di database.

`editDesaWisata()`
 digunakan untuk mengubah data desa wisata yang ada di database.

2.2.3.6 Specific Design Class PengelolaanHotelView

PengelolaanHotelView	<<boundary>>
<code>pengelolaanHotelView()</code> sebagai default konstruktor dimana semua atribut pada class ini diinisialisasi. <code>addHotel()</code> digunakan untuk menambahkan data hotel ke dalam database. <code>deleteHotel()</code> digunakan untuk menghapus data hotel yang ada di database. <code>editHotel()</code> digunakan untuk mengubah data hotel yang ada di database.	

2.2.3.7 Specific Design Class PengelolaanRentalView

PengelolaanRentalView	<<boundary>>
<pre>pengelolaanRentalView() sebagai default konstruktor dimana semua atribut pada class ini diinisialisasi. addRental() digunakan untuk menambahkan data lokasi rental kendaraan ke dalam database. deleteRental() digunakan untuk menghapus data lokasi rental kendaraan yang ada di database. editRental() digunakan untuk mengubah data lokasi rental kendaraan yang ada di database.</pre>	

2.2.3.8 Specific Design Class PengelolaanPenggunaView

PengelolaanPenggunaView	<<boundary>>
<pre>pengelolaanPenggunaView() sebagai default konstruktor dimana semua atribut pada class ini diinisialisasi. addPengguna() digunakan untuk menambahkan data pengguna ke dalam database. deleteRental() digunakan untuk menghapus data pengguna yang ada di database. editRental()</pre>	

digunakan untuk mengubah data pengguna yang ada di database.

2.2.3.9 Specific Design Class LoginView

LoginView	<<boundary>>
LoginUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. login() Operasi ini digunakan untuk mengambil data login yang diinputkan oleh user, yaitu login username dan password.	

2.2.3.10 Specific Design Class LokasiCtrl

LokasiCtrl	<<control>>
addLokasi() digunakan untuk menambahkan data lokasi ke dalam database. getDataLokasi() digunakan untuk mengambil data lokasi dari database. getDataLokasi() digunakan untuk mengambil data lokasi dari database berdasarkan id_lokasi atau id_kategori. searchLokasi() digunakan untuk mencari data lokasi berdasarkan keyword tertentu.	

2.2.3.11 Specific Design Class PengelolaanDesaWisataCtrl

PengelolaanDesaWisataCtrl	<<control>>
<pre>validasiDataDesaWisata() digunakan untuk mengecek data desa wisata inputan user sebelum diteruskan ke database addLokasi() digunakan untuk menambahkan data desa wisata ke dalam database. editLokasi() digunakan untuk mengedit data desa wisata yang ada di dalam database getDataLokasi() digunakan untuk mengambil data lokasi desa wisata dari database. getDataLokasi() digunakan untuk mengambil data lokasi desa wisata dari database berdasarkan id_lokasi atau id_kategori.</pre>	

2.2.3.12 Specific Design Class PengelolaanHotelCtrl

PengelolaanHotelCtrl	<<control>>
<pre>validasiDataHotel() digunakan untuk mengecek data hotel inputan user sebelum diteruskan ke database addLokasi() digunakan untuk menambahkan data hotel ke dalam database. editLokasi() digunakan untuk mengedit data hotel yang ada di dalam database</pre>	

getDataLokasi()
digunakan untuk mengambil data lokasi hotel dari database.

getDataLokasi()
digunakan untuk mengambil data lokasi hotel dari database berdasarkan id_lokasi atau id_kategori.

2.2.3.13 Specific Design Class PengelolaanRentalCtrl

PengelolaanRentalCtrl	<<control>>
validasiDataRental() digunakan untuk mengecek data lokasi rental kendaraan inputan user sebelum diteruskan ke database addLokasi() digunakan untuk menambahkan data lokasi rental kendaraan ke dalam database. editLokasi() digunakan untuk mengedit data lokasi rental kendaraan yang ada di dalam database getDataLokasi() digunakan untuk mengambil data lokasi rental kendaraan dari database. getDataLokasi() digunakan untuk mengambil data lokasi rental kendaraan dari database berdasarkan id_lokasi atau id_kategori.	

2.2.3.14 Specific Design Class PengelolaanPenggunaCtrl

PengelolaanPenggunaCtrl	<<control>>
validasiDataPengguna() digunakan untuk mengecek data pengguna inputan user sebelum diteruskan ke database addPengguna() digunakan untuk menambahkan data pengguna ke dalam database. editPengguna() digunakan untuk mengedit data pengguna yang ada di dalam database getDataPengguna() digunakan untuk mengambil data pengguna dari database. getDataPengguna() digunakan untuk mengambil data pengguna dari database berdasarkan id_lokasi atau id_kategori.	

2.2.3.15 Specific Design Class LoginCtrl

LoginCtrl	<< Control >>
validasiLogin() digunakan untuk mencocokkan data inputan user dengan database. login() operasi ini digunakan untuk meneruskan hasil pencocokan data inputan ke database.	

2.2.3.16 Specific Design Class User

USER	<<Entity>>
-id_user: int, digunakan untuk menyimpan data identitas	

user.

-id_role: int, digunakan untuk menyimpan data identitas role.

-nama_lengkap: varchar, digunakan untuk menyimpan data nama lengkap.

-username: varchar, digunakan untuk menyimpan data username.

-password: varchar, digunakan untuk menyimpan data password.

-no_hp : varchar, digunakan untuk menyimpan data nomor *handphone* user.

-status: integer, digunakan untuk menyimpan jika data user terhapus.

+cekBasisData() digunakan untuk mengecek data username password inputan user dengan database.

+addPengguna() digunakan untuk menambahkan data pengguna ke dalam database.

+getDataPengguna() digunakan untuk mendapatkan data pengguna.

+updatePengguna() digunakan untuk mengubah data pengguna dalam database dengan data pengguna yang sedang diubah.

+getDataPenggunaBy() digunakan untuk mendapatkan data Pengguna yang sesuai dengan id atau nama pengguna tertentu.

+DisplayUpdate() digunakan untuk menampilkan data pengguna yang akan diubah.

+addPengguna() digunakan untuk menambahkan data pengguna baru.

+getDataPengguna() digunakan untuk mendapatkan data pengguna dari database.

+updatePengguna() digunakan untuk mengubah data pengguna dalam database dengan data pengguna yang telah diubah.

+getDataPenggunaBy() digunakan untuk mendapatkan data pengguna berdasarkan id atau nama pengguna tertentu.

2.2.3.17 Specific Design Class Role

ROLE	<<Entity>>
-id_role: int, digunakan untuk menyimpan data identitas role.	
-keterangan: varchar, digunakan untuk menyimpan data keterangan role.	
+getDataRole() digunakan untuk mendapatkan data role dari database.	

2.2.3.18 Specific Design Class Kategori

Kategori	<<Entity>>
-id kategori: int, digunakan untuk menyimpan data identitas kategori.	
-keterangan: varchar, digunakan untuk menyimpan data keterangan kategori.	
+getDataKategori() digunakan untuk mendapatkan data kategori dari database.	

2.2.3.19 Specific Design Class Lokasi

LokasiModel	<<entity>>
Id_lokasi : int	
Atribut ini digunakan untuk menyimpan nilai id lokasi	
+id_wilayah : int	

Atribut ini digunakan untuk menyimpan nilai id wilayah

+id_kategori : int

Atribut ini digunakan untuk menyimpan nilai id kategori

+latitude : double

Atribut ini digunakan untuk menyimpan latitude desa wisata

+longitude : double

Atribut ini digunakan untuk menyimpan longitude desa wisata

+nama : string

Atribut ini digunakan untuk menyimpan nama atribut lokasi

+alamat : string

Atribut ini digunakan untuk menyimpan alamat lokasi

+telepon : string

Atribut ini digunakan untuk menyimpan nomor telepon

+arah_lokasi : string

Atribut ini digunakan untuk menyimpan detail arah menuju lokasi

+imageDesa : string

Atribut ini digunakan untuk menyimpan image/gambar dari lokasi

+keterangan : string

Atribut ini digunakan untuk menyimpan keterangan tentang lokasi

+keterangan2 : string

Atribut ini digunakan untuk menyimpan keterangan tambahan tentang lokasi

cekBasisData()

digunakan untuk mengecek data username password inputan user dengan database.

`addLokasi()`

digunakan untuk menambahkan data lokasi ke dalam database.

`getDataLokasi()` digunakan untuk mendapatkan data lokasi.

`updateLokasi()`

digunakan untuk mengubah data lokasi dalam database dengan data lokasi yang sedang diubah.

`getDataLokasiBy()`

digunakan untuk mendapatkan data lokasi yang sesuai dengan id atau nama lokasi tertentu.

`DisplayUpdate()`

digunakan untuk menampilkan data lokasi yang akan diubah.

`getDataLokasiBy()`

digunakan untuk mendapatkan data lokasi berdasarkan id atau nama lokasi tertentu.

3. PERANCANGAN DATA

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Lokasi

Nama	Tipe	Panjang	Keterangan
Id_desa	Int	100	Id dari lokasi , primary key.
Id_wilayah	Int	100	Id wilayah, foreign key.
Id_kategori	Int	100	Id_kategori, foreign key
Lattitude	Double	100	Lattitude lokasi

Longitude	Double	100	Longitude lokasi
Nama	Varchar	255	Nama lokasi
Alamat	Varchar	255	Alamat lokasi
Telepon	Varchar	255	Telepon lokasi
Arah_lokasi	Varchar	255	Arah lokasi dengan menggunakan petunjuk tempat-tempat yang diketahui secara umum
Image	Varchar	100	Gambar lokasi
keterangan	Varchar	255	Keterangan tambahan untuk lokasi

3.1.1 Deskripsi Entitas Hotel

Nama	Tipe	Panjang	Keterangan
id_user	integer	-	Kode dari user, primary key
Id_role	Integer	-	Kode dari role, foreign key
Nama_lengkap	varchar	100	Nama lengkap user
username	varchar	50	Username yang dimiliki user
password	varchar	50	Password yang dimiliki user
no_hp	varchar	50	Nomor ponsel user
status	integer	-	Tanda jika terisi 1 maka data user telah terhapus

3.1.1 Deskripsi Entitas Role

Nama	Tipe	Panjang	Keterangan
Id_role	integer	-	Kode dari role, primary key
keterangan	varchar	1024	Keterangan / jenis role

3.1.2 Deskripsi Entitas Kategori

Nama	Tipe	Panjang	Keterangan
Id_kategori	integer	-	Kode dari

			kategori, primary key
Keterangan	Varchar	1024	Keterangan / jenis kategori



4. Perancangan Antarmuka

APLIKASI MOBILE

4.1 Main Menu



Gambar 4.1 : Mockup Home

Antarmuka Gambar 4.1 digunakan oleh pengguna untuk menampilkan semua desa wisata, rental mobil dan hotel yang ada di dalam database. Data ditampilkan dalam bentuk List. Pengguna dapat melihat detail desa wisata, rental atau hotel dengan men-tap salah satu item yang ada dalam List.

4.2 Menu Slider

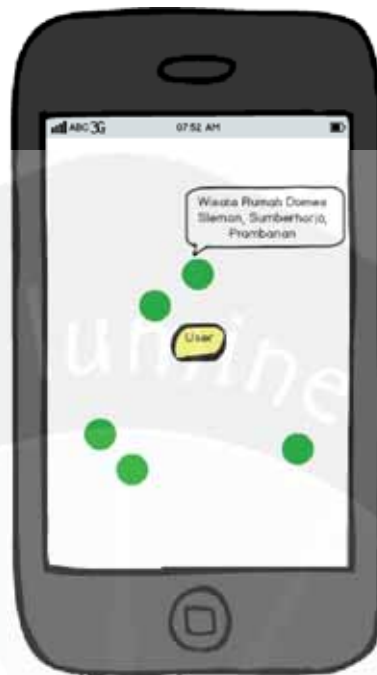


Gambar 4.2 : Mockup Menu SLider

Antarmuka Gambar 4.2 adalah Menu Slider yang digunakan sebagai Menu navigasi utama. Dari menu navigasi utama ini, pengguna dapat berpindah ke sub-menu seperti show maps [Gambar 4.3], Ar desa wisata [Gambar 4.4] dan tentang jogja.

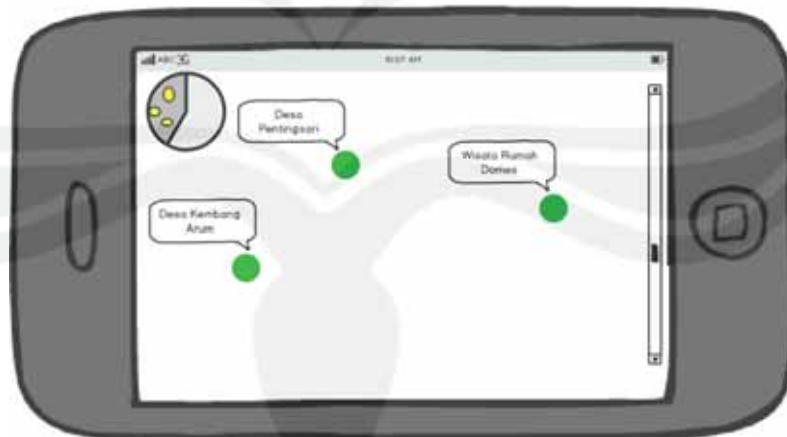
4.3 Show Maps

Antarmuka Gambar 4.3 adalah Show Maps dimana dalam halaman ini akan ditampilkan semua lokasi desa wisata, hotel atau rental kendaraan. Di bagian bawah terdapat tombol zoom in dan zoom out yang digunakan untuk mengatur jarak pandang pengguna terhadap marker lokasi.



Gambar 4.3 : Mockup Menu Show Maps

4.4 AR Desa Wisata



Gambar 4.4 : Mockup Desa Wisata Sekitar

Antarmuka Gambar 4.4 digunakan oleh pengguna untuk menampilkan desa wisata di sekitar pengguna dalam bentuk marker. Untuk mendapatkan lokasi marker pengguna

men-scan area sekitar dengan menggunakan kamera. Marker yang tertampil dapat di tap untuk mendapatkan detail informasi tentang desa wisata yang dipilih.

4.5 Route to Desa Wisata



Gambar 4.5 : Mockup Route to Desa Wisata

Antarmuka pada gambar 4.5 ini digunakan untuk menampilkan rute dari posisi pengguna ke posisi desa wisata yang telah dipilih pada menu sebelumnya.

4.6 Detail Desa Wisata

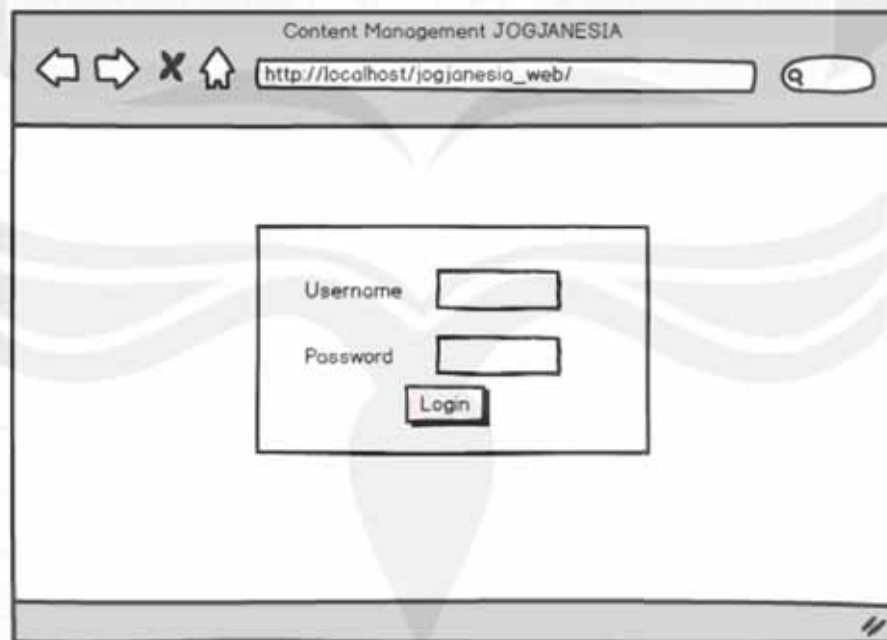
Antarmuka Gambar 4.6 digunakan oleh pengguna untuk menampilkan detail desa wisata yang dipilih pada marker sebelumnya. Dalam detail informasi di dapatkan gambar desa wisata dan atribut-atribut lain mengenai desa wisata tersebut. dalam antarmuka ini didapatkan juga tombol route here yang digunakan untuk menampilkan rute.



Gambar 4.6 : Mockup Detail Desa Wisata

APLIKASI WEB

4.7 Login Page

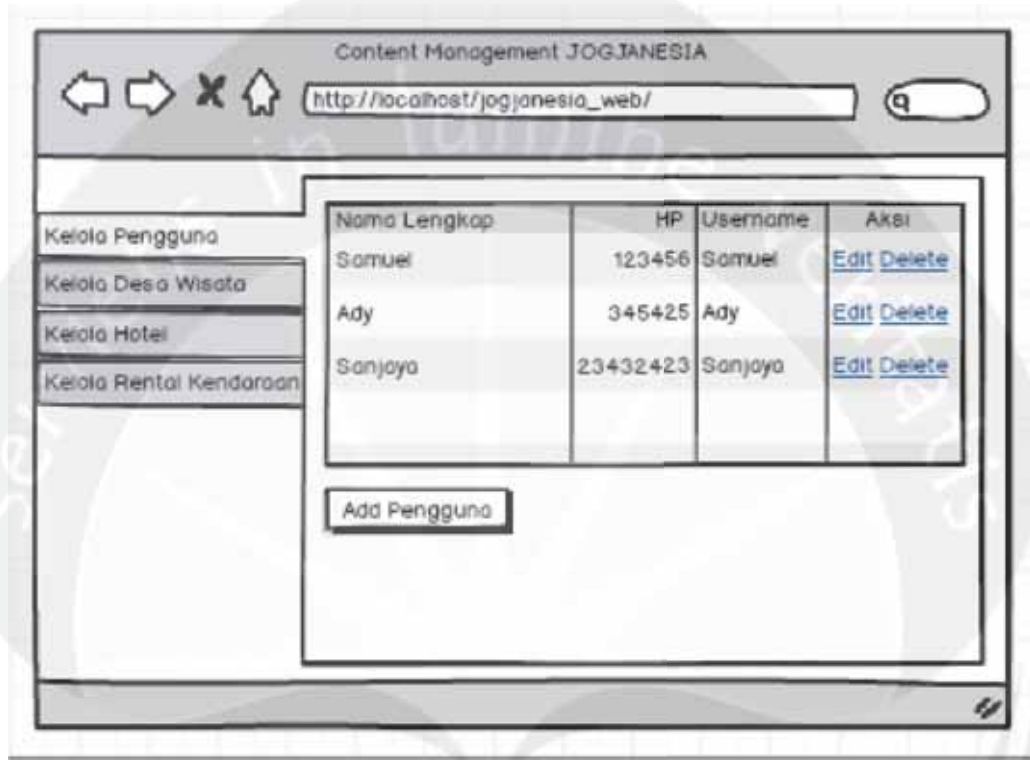


Gambar 4.7 Halaman Login

Pada gambar 4.7 digunakan untuk melakukan proses login ke dalam sistem aplikasi web. Untuk bias login ke

dalam aplikasi web, pengguna harus memasukkan username password yang telah ada dalam database.

4.8 Pengelolaan Data Pengguna



Gambar 4.8 Halaman Pengelolaan Data Pengguna

Halaman ini digunakan untuk mengelola data pengguna yang meliputi fungsi show data pengguna, add data pengguna, edit data pengguna dan delete data pengguna. Tabel dalam halaman tersebut menampilkan semua data pengguna yang ada dalam database.

4.9 Add data Pengguna

Content Management JOGJANESIA

http://localhost/jogjanesia_web/

Kelola Pengguna

Kelola Desa Wisata

Kelola Hotel

Kelola Rental Kendaraan

Nama Lengkap

Username

Password

No HP

Add Pengguna

Gambar 4.9 Halaman Add data pengguna

Halaman ini muncul saat user yang login memilih untuk melakukan tambah data pengguna. Dalam halaman ini terdapat form yang digunakan untuk mengisi informasi detail pengguna baru yang meliputi nama lengkap, username, password dan nomor HP. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol add pengguna.

4.10 Edit Data Pengguna

Content Management JOGJANESIA

http://localhost/jogjanesia_web/

Kelola Pengguna
Kelola Desa Wisata
Kelola Hotel
Kelola Rental Kendaraan

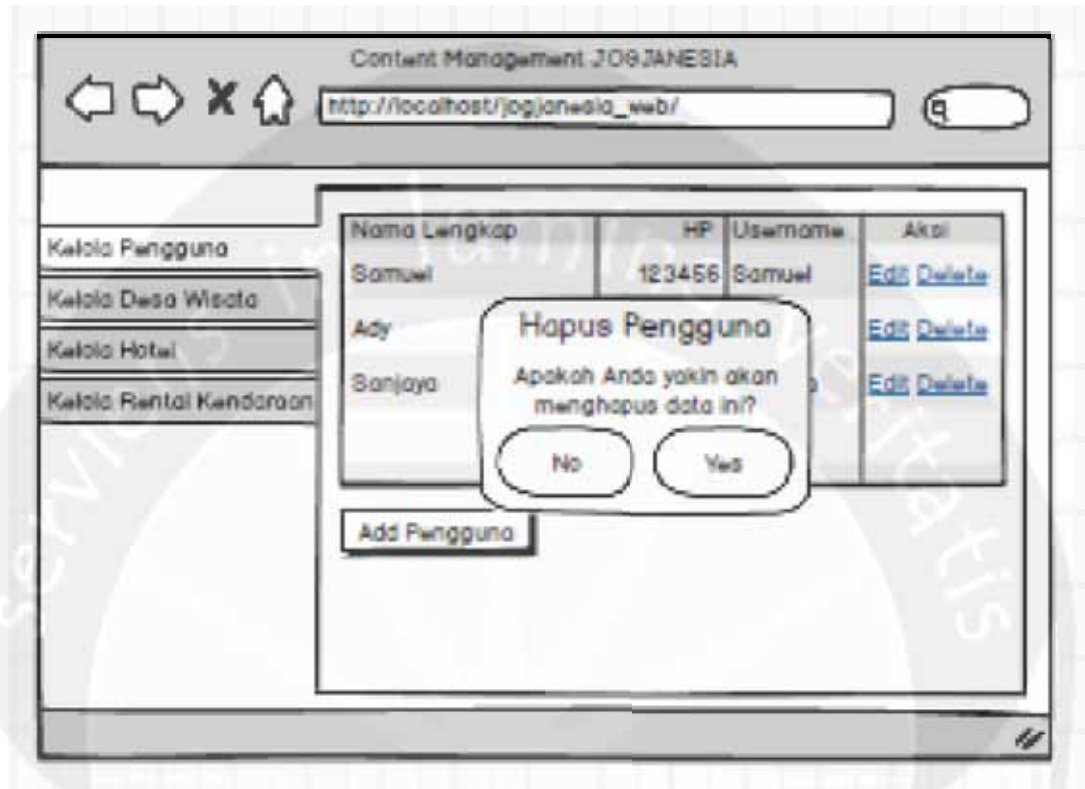
Nama Lengkap: Samuel
Username: Sam
Password:
No HP: 123123

Simpan Batal

Gambar 4.10 Halaman Edit Data Pengguna

Halaman ini muncul saat user yang login memilih untuk melakukan edit data pengguna. Untuk melakukan edit data, user cukup memilih tombol edit yang ada di bagian paling kanan dalam suatu baris. Dalam halaman ini terdapat form yang digunakan untuk mengupdate informasi detail pengguna yang meliputi nama lengkap, username, dan nomor HP. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol simpan, jika user ingin membatalkan proses dengan mengklik tombol batal.

4.11 Delete Data Pengguna



Gambar 4.11 Halaman Delete Data Pengguna

Pesan peringatan ini muncul saat user yang sedang login ingin menghapus user lain yang ada di database. Jika user telah yakin untuk menghapus data pengguna, maka pilih "Yes" dan data pengguna akan terhapus dari database. Untuk membatalkan penghapusan pengguna memilih "No". untuk menghapus satu data, pengguna harus memilih satu data yang akan dihapus, jika pengguna belum memilih maka akan keluar pesan bahwa pengguna harus memilih salah satu data.

4.12 Pengelolaan data Desa Wisata



Gambar 4.12 Halaman Pengelolaan Data Desa Wisata

Halaman ini digunakan untuk mengelola data desa wisata yang meliputi fungsi show data desa wisata, add data desa wisata, edit data desa wisata dan delete data desa wisata. Tabel dalam halaman tersebut menampilkan semua data desa wisata yang ada dalam database.

4.13 Add data Desa Wisata

The screenshot shows a web browser window titled 'Content Management JOGJANESIA' with the address bar displaying 'http://localhost/jogjanesia_web/'. On the left is a sidebar menu with four items: 'Kelola Pengguna', 'Kelola Desa Wisata', 'Kelola Hotel', and 'Kelola Rental Kendaraan'. The main content area contains a form for adding a new village tourism location. The form fields are: 'Nama Lokasi' (filled with 'Desa Wisata Tembi'), 'Alamat' (filled with 'Sewon Bantul'), 'Telepon' (filled with '12321412'), 'Arah' (filled with 'di selatan'), and 'Keterangan' (filled with 'ada kebun tehnya'). At the bottom of the form are two buttons: 'Simpan' and 'Batal'.

Gambar 4.13 Halaman Add data desa wisata

Halaman ini muncul saat user yang login memilih untuk melakukan tambah data desa wisata. Dalam halaman ini terdapat form yang digunakan untuk mengisi informasi detail desa wisata baru yang meliputi nama lokasi, alamat, nomor telepon, arah dan keterangan. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol add desa wisata.

4.14 Edit Data Desa Wisata

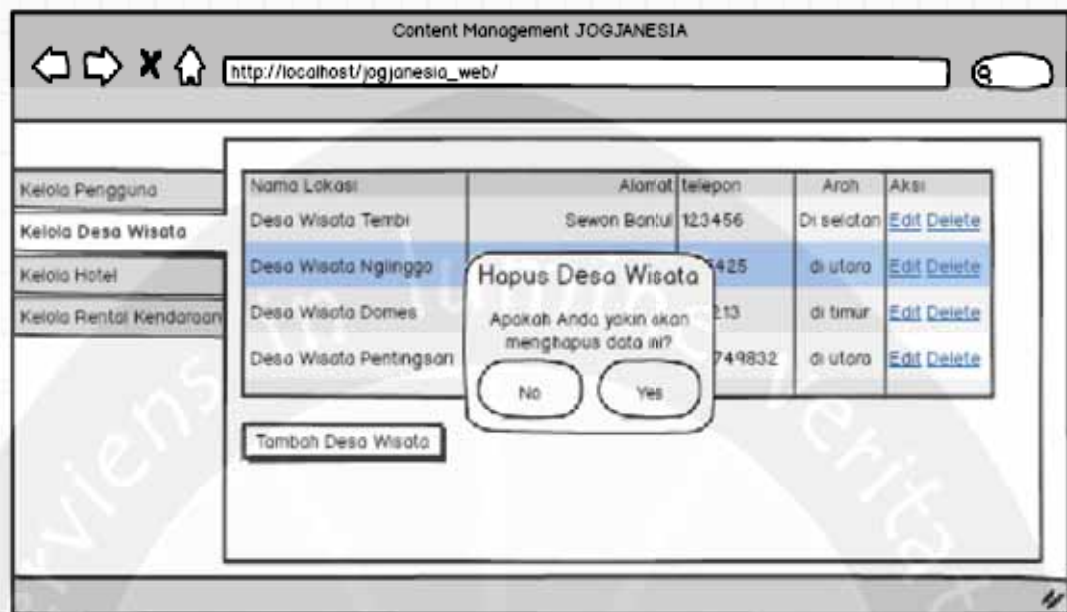


The screenshot shows a web browser window titled "Content Management JOGJANESIA" with the address bar displaying "http://localhost/jogjanesia_web/". On the left is a sidebar menu with four items: "Kelola Pengguna", "Kelola Desa Wisata", "Kelola Hotel", and "Kelola Rental Kendaraan". The main content area is titled "Edit Desa Wisata" and contains a form with the following fields: "Nama Lokasi" (filled with "Desa Wisata Tembi"), "Alamat" (filled with "Sewon Bantul"), "Telepon" (filled with "12321412"), "Arah" (filled with "di selatan"), and "Keterangan" (filled with "ada kebun tehnya"). At the bottom of the form are two buttons: "Simpan" and "Batal".

Gambar 4.14 Halaman Edit Data Desa wisata

Halaman ini muncul saat user yang login memilih untuk melakukan edit data desa wisata. Untuk melakukan edit data, user cukup memilih tombol edit yang ada di bagian paling kanan dalam suatu baris. Dalam halaman ini terdapat form yang digunakan untuk mengupdate informasi detail desa wisata. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol simpan, jika user ingin membatalkan proses dengan mengklik tombol batal.

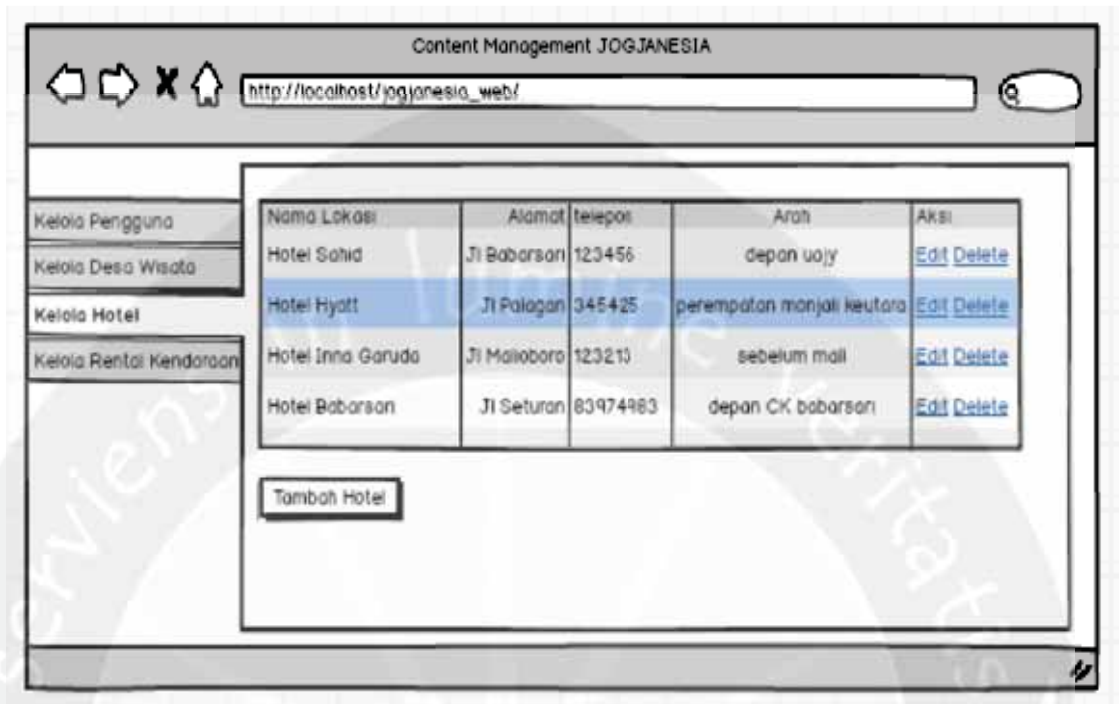
4.15 Delete Data Desa Wisata



Gambar 4.15 Halaman Delete Data Desa Wisata

Pesan peringatan ini muncul saat user yang sedang login ingin menghapus data desa wisata yang ada di database. Jika user telah yakin untuk menghapus data desa wisata, maka pilih "Yes" dan data pengguna akan terhapus dari database.

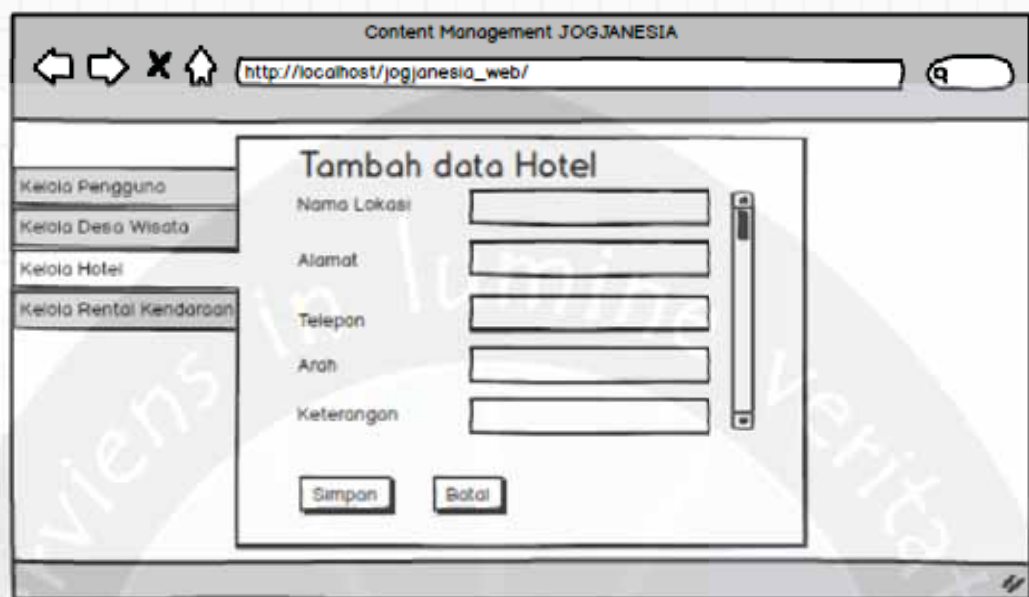
4.16 Pengelolaan Data Hotel



Gambar 4.16 Halaman Pengelolaan Hotel

Halaman ini digunakan untuk mengelola data hotel yang meliputi fungsi show data hotel, add data hotel, edit data hotel dan delete data hotel. Tabel dalam halaman tersebut menampilkan semua data hotel yang ada dalam database.

4.17 Add Data Hotel

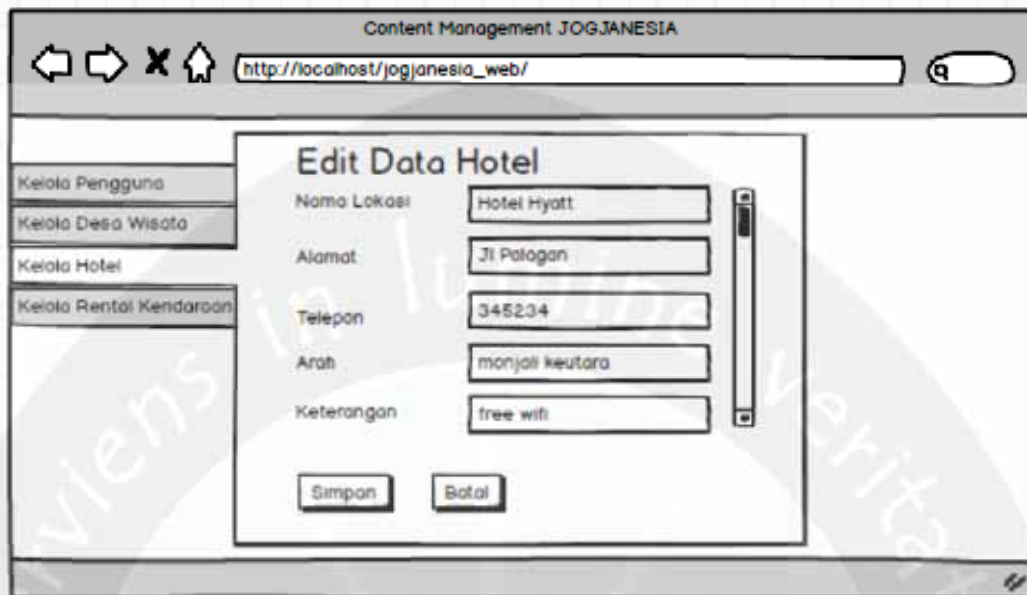


The screenshot shows a web browser window titled "Content Management JOGJANESIA". The address bar displays "http://localhost/jogjanesia_web/". On the left side, there is a vertical menu with four items: "Kelola Pengguna", "Kelola Desa Wisata", "Kelola Hotel", and "Kelola Rental Kendaraan". The "Kelola Hotel" item is selected. The main content area is titled "Tambah data Hotel" and contains a form with the following fields: "Nama Lokasi", "Alamat", "Telepon", "Arah", and "Keterangan". Each field is represented by a text input box. To the right of these input boxes is a vertical scrollbar. At the bottom of the form, there are two buttons: "Simpan" (Save) and "Batal" (Cancel).

Gambar 4.17 Halaman Add Data Hotel

Halaman ini muncul saat user yang login memilih untuk melakukan tambah data hotel. Dalam halaman ini terdapat form yang digunakan untuk mengisi informasi detail hotel baru yang meliputi nama lokasi, alamat, nomor telepon, arah dan keterangan. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol add hotel.

4.18 Edit Data Hotel

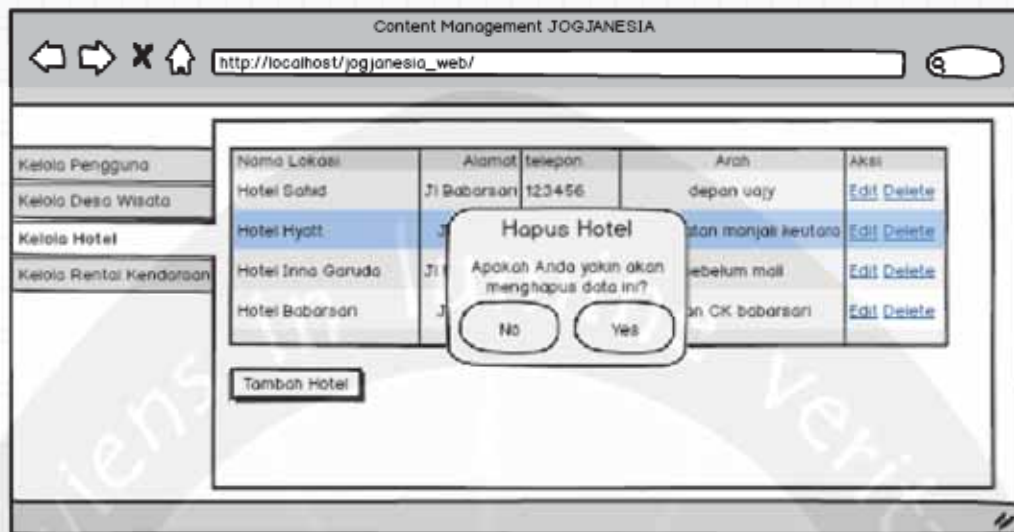


The screenshot shows a web browser window titled "Content Management JOGJANESIA". The address bar displays "http://localhost/jogjanesia_web/". On the left side, there is a vertical menu with four items: "Kelola Pengguna", "Kelola Desa Wisata", "Kelola Hotel", and "Kelola Rental Kendaraan". The "Kelola Hotel" item is selected. The main content area is titled "Edit Data Hotel" and contains a form with the following fields: "Nama Lokasi" (Hotel Hyatt), "Alamat" (Jl Palagan), "Telepon" (345234), "Arah" (menjari keutara), and "Keterangan" (free wifi). At the bottom of the form are two buttons: "Simpan" and "Batal".

Gambar 4.18 Halaman Edit Data Hotel

Halaman ini muncul saat user yang login memilih untuk melakukan edit data hotel. Untuk melakukan edit data, user cukup memilih tombol edit yang ada di bagian paling kanan dalam suatu baris. Dalam halaman ini terdapat form yang digunakan untuk mengupdate informasi detail hotel. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol simpan, jika user ingin membatalkan proses dengan mengklik tombol batal.

4.19 Delete Data Hotel



Gambar 4.19 Halaman Delete Data Hotel

Pesan peringatan ini muncul saat user yang sedang login ingin menghapus data hotel yang ada di database. Jika user telah yakin untuk menghapus data hotel, maka pilih "Yes" dan data pengguna akan terhapus dari database. Untuk membatalkan penghapusan pengguna memilih "No". untuk menghapus satu data, pengguna harus memilih satu data yang akan dihapus, jika pengguna belum memilih maka akan keluar pesan bahwa pengguna harus memilih salah satu data.

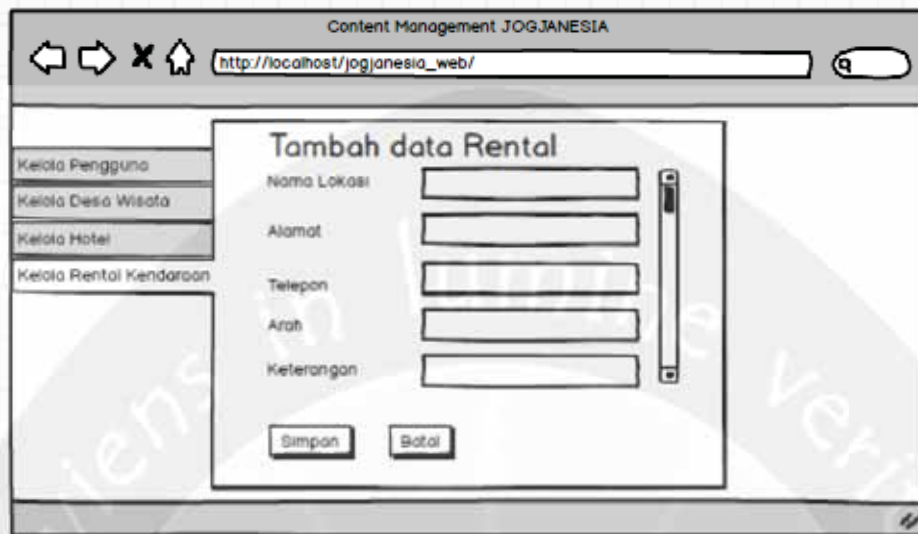
4.20 Pengelolaan Data Rental Kendaraan



Gambar 4.20 Halaman Pengelolaan Data Rental Kendaraan

Halaman ini digunakan untuk mengelola data rental kendaraan yang meliputi fungsi show data rental kendaraan, add data rental kendaraan, edit data rental kendaraan dan delete data rental kendaraan. Tabel dalam halaman tersebut menampilkan semua data rental kendaraan yang ada dalam database.

4.21 Add Data Rental Kendaraan



The screenshot shows a web browser window titled "Content Management JOGJANESIA" with the address bar displaying "http://localhost/jogjanesia_web/". On the left is a sidebar menu with four items: "Kelola Pengguna", "Kelola Desa Wisata", "Kelola Hotel", and "Kelola Rental Kendaraan". The main content area is titled "Tambah data Rental" and contains a form with the following fields: "Nama Lokasi", "Alamat", "Telepon", "Arah", and "Keterangan". Each field has a corresponding text input box. To the right of these fields is a vertical scrollbar. At the bottom of the form are two buttons: "Simpan" and "Batal".

Gambar 4.21 Halaman Add Data Rental Kendaraan

Halaman ini muncul saat user yang login memilih untuk melakukan tambah data rental kendaraan. Dalam halaman ini terdapat form yang digunakan untuk mengisi informasi detail rental kendaraan baru yang meliputi nama lokasi, alamat, nomor telepon, arah dan keterangan. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol add desa wisata.

4.22 Edit Data Rental Kendaraan

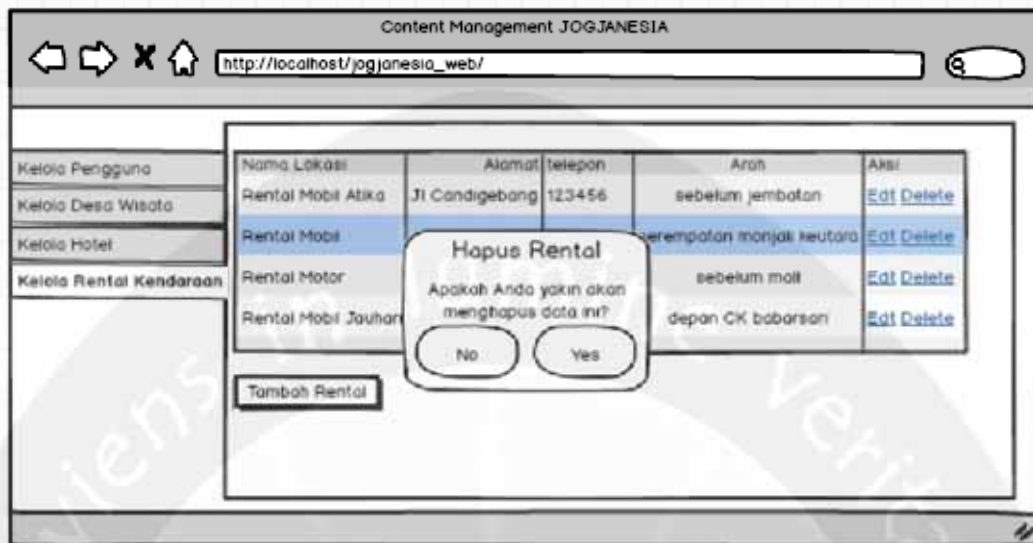


The screenshot shows a web browser window titled "Content Management JOGJANESIA" with the address bar displaying "http://localhost/jogjanesia_web/". On the left, there is a sidebar menu with four items: "Kelola Pengguna", "Kelola Desa Wisata", "Kelola Hotel", and "Kelola Rental Kendaraan", with the last one being selected. The main content area is titled "Edit Data Rental" and contains a form with the following fields: "Nama Lokasi" (filled with "Rental Mobil Atika"), "Alamat" (filled with "Jl Palagan"), "Telepon" (filled with "045234"), "Arah" (filled with "menjari keutara"), and "Keterangan" (filled with "free wifi"). At the bottom of the form are two buttons: "Simpan" and "Batal".

Gambar 4.22 Halaman Edt Data Rental Kendaraan

Halaman ini muncul saat user yang login memilih untuk melakukan edit data rental kendaraan. Untuk melakukan edit data, user cukup memilih tombol edit yang ada di bagian paling kanan dalam suatu baris. Dalam halaman ini terdapat form yang digunakan untuk mengupdate informasi detail rental kendaraan. Setelah user melengkapi field dalam form tersebut, untuk menyimpannya dengan mengklik tombol simpan, jika user ingin membatalkan proses dengan mengklik tombol batal.

4.23 Delete Data Rental Kendaraan



Gambar 4.23 Halaman Delete Data Rental Kendaraan

Pesan peringatan ini muncul saat user yang sedang login ingin menghapus data desa wisata yang ada di database. Jika user telah yakin untuk menghapus data desa wisata, maka pilih "Yes" dan data pengguna akan terhapus dari database.

SKPL

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK PENGEMBANGAN SISTEM INFORMASI DESA WISATA YOGYAKARTA DENGAN *AUGMENTED REALITY* BERBASIS LOKASI (JOGJANESIA)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Samuel Ady Sanjaya / 12. 07. 07007

**Program Studi Teknik Informatika – Fakultas Teknologi
Industri**

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL - JOGJANESIA		1/37
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh	SAS							
Diperiksa oleh	TS BYD							
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak JOGJANESIA untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-JOGJANESIA ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak JOGJANESIA dikembangkan dengan tujuan untuk:

1. Menangani penyajian informasi desa wisata informasi penginapan, dan informasi penyewaan kendaraan.
2. Menangani pencarian lokasi desa wisata dengan augmented reality.
3. Menangani navigasi rute perjalanan dari lokasi pengguna ke lokasi desa wisata.
4. Menangani pengelolaan data desa wisata.
5. Menangani pengelolaan data hotel.
6. Menangani pengelolaan data rental kendaraan.

Dan aplikasi ini dapat berjalan dengan smartphone berbasis android.

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
JOGJANESIA	Perangkat lunak mobile untuk Desa Wisata dengan Augmented Reality.
SKPL - JOGJANESIA-XXX	Kode yang merepresentasikan kebutuhan pada JOGJANESIA[Nama Aplikasi Mobile] dimana XXX merupakan nomor fungsi produk.
SKPL - JOGJANESIA-WEB-XXX	Kode yang merepresentasikan kebutuhan pada JOGJANESIA-WEB[web untuk pengelolaan konten untuk aplikasi mobile] dimana XXX merupakan nomor fungsi produk.
Augmented Reality	Teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata.
Desa Wisata	kawasan pedesaan yang memiliki beberapa karakteristik khusus untuk menjadi daerah tujuan wisata dan merupakan suatu bentuk integrasi antara atraksi, akomodasi dan fasilitas pendukung yang disajikan dalam suatu struktur kehidupan masyarakat yang menyatu dengan tata cara dan tradisi yang berlaku.

Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
Framework	Suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks.
PHP	Hypertext Preprocessor merupakan bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML dan banyak dipakai untuk memprogram situs web dinamis.
Android	adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Yosafat Novianto Ade Putra, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL)* WIYATA, Universitas Atma Jaya Yogyakarta.

2. Yosep Wihelmus Nabu, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) VISITMABAR*, 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 JOGJANESIA 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 JOGJANESIA tersebut.

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

2 Deskripsi Kebutuhan

2.1 Perspektif produk

JOGJANESIA merupakan perangkat lunak yang dikembangkan untuk membantu penyebaran informasi desa wisata yang ada di Yogyakarta menggunakan teknologi Augmented Reality dan memanfaatkan Layanan Berbasis Lokasi. Selain itu sistem ini diharapkan dapat menjadi

sarana untuk mendapatkan informasi mengenai desa wisata Yogyakarta yang interaktif sehingga dapat menarik wisatawan untuk mencari tahu tentang desa wisata.

Perangkat lunak ini dikembangkan untuk ditanamkan pada sisi client di lingkungan sistem operasi Android OS. Perangkat lunak ini dikembangkan menggunakan bahasa pemrograman Java dengan framework Mixare dan menggunakan lingkungan pemrograman Eclipse Luna. Perangkat lunak web sistem ini berjalan pada browser apapun. Aplikasi web ini dibuat menggunakan bahasa pemrograman PHP dengan framework CI. Database yang digunakan pada aplikasi ini terpusat pada server yaitu dengan database MySQL.



Gambar 1. Arsitektur Perangkat lunak JOGJANESIA

Pada aplikasi ini, seperti terlihat pada Gambar 1, arsitektur perangkat lunak yang digunakan berupa *client server*. Pada arsitektur ini, inputan data yang dimasukkan akan disimpan dalam basis data *server*, sehingga saat terjadi pengaksesan data pada perangkat *mobile client*, maka permintaan *client* akan dikirimkan ke *server* dan diproses di basis data *server*, kemudian hasil pemrosesan data selanjutnya dikirimkan ke perangkat *mobile client*.

2.2 Fungsi Produk

Fungsi produk perangkat lunak JOGJANESIA adalah sebagai berikut :

APLIKASI MOBILE

1. Fungsi Show Desa Wisata, Hotel dan Rental Kendaraan dalam List (**SKPL-JOGJANESIA-001**)

Merupakan fungsi yang digunakan untuk melihat daftar desa wisata, hotel dan rental pengguna di sekitar pengguna dalam bentuk List.

2. Fungsi Show Desa Wisata dengan Augmented Reality Location Based (**SKPL-JOGJANESIA-002**)

Merupakan fungsi yang digunakan untuk menampilkan lokasi Desa Wisata dalam bentuk Augmented Reality.

3. Fungsi Show Detail Desa Wisata (**SKPL-JOGJANESIA-003**)

merupakan fungsi yang digunakan untuk melihat detail informasi tentang desa wisata yang dipilih.

4. Fungsi Show Rute Desa Wisata (**SKPL-JOGJANESIA-004**)

Merupakan fungsi yang digunakan untuk menampilkan rute dari posisi pengguna menuju ke desa wisata yang dipilih dengan menggunakan support Google Maps.

5. Fungsi Show Desa Wisata, Hotel dan Rental Kendaraan dengan menggunakan Maps(**SKPL-JOGJANESIA-005**)

Merupakan fungsi yang digunakan untuk menampilkan lokasi Desa wisata, Hotel dan Rental Kendaraan dalam bentuk koordinat-koordinat dalam Maps.

6. Fungsi Search Desa Wisata, Hotel dan Rental Kendaraan(**SKPL-JOGJANESIA-006**)

Merupakan fungsi yang digunakan untuk menampilkan semua data Desa Wisata, Hotel dan Rental kendaraan sesuai dengan keyword yang dimasukkan user.

APLIKASI WEB

7. Fungsi Login (**SKPL-JOGJANESIA-WEB-001**)

Merupakan fungsi yang digunakan oleh pengguna khusus yaitu admin untuk dapat masuk ke dalam aplikasi web.

8. Fungsi Pengelolaan Pengguna (**SKPL-JOGJANESIA-WEB-002**)

Merupakan fungsi yang digunakan untuk mengelola data pengguna.

Fungsi *Pengelolaan Pengguna* meliputi:

- a. Fungsi *Add Pengguna* (**SKPL-JOGJANESIA-WEB-002-01**).

Merupakan fungsi yang digunakan untuk menambahkan data pengguna.

- b. Fungsi *Edit Pengguna* (**SKPL-JOGJANESIA-WEB-002-02**).

Program Studi Teknik Informatika	SKPL – JOGJANESIA	10/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk mengubah data pengguna.

c. Fungsi *Delete Pengguna* (**SKPL-JOGJANESIA-WEB-002-03**).

Merupakan fungsi yang digunakan untuk menghapus data pengguna.

d. Fungsi *Display Pengguna* (**SKPL-JOGJANESIA-WEB-002-04**).

Merupakan fungsi yang digunakan untuk menampilkan data pengguna.

9. Fungsi Pengelolaan Desa Wisata (**SKPL-JOGJANESIA-WEB-003**)

Merupakan fungsi yang digunakan untuk mengelola data desa wisata.

Fungsi *Pengelolaan Desa Wisata* meliputi:

a. Fungsi *Add Desa Wisata* (**SKPL-JOGJANESIA-WEB-003-01**).

Merupakan fungsi yang digunakan untuk menambahkan data desa wisata.

b. Fungsi *Edit Desa Wisata* (**SKPL-JOGJANESIA-WEB-003-02**).

Merupakan fungsi yang digunakan untuk mengubah data desa wisata.

c. Fungsi *Delete Desa Wisata* (**SKPL-JOGJANESIA-WEB-003-03**).

Merupakan fungsi yang digunakan untuk menghapus data desa wisata.

d. Fungsi *Display Desa Wisata* (**SKPL-JOGJANESIA-WEB-003-04**).

Merupakan fungsi yang digunakan untuk menampilkan data desa wisata.

10. Fungsi Pengelolaan Hotel (**SKPL-JOGJANESIA-WEB-004**)

Merupakan fungsi yang digunakan untuk mengelola data hotel.

Fungsi *Pengelolaan Hotel* meliputi:

a. Fungsi *Add Hotel* (**SKPL-JOGJANESIA-WEB-004-01**).

Merupakan fungsi yang digunakan untuk menambahkan data hotel.

b. Fungsi *Edit Hotel* (**SKPL-JOGJANESIA-WEB-004-02**).

Merupakan fungsi yang digunakan untuk mengubah data desa wisata.

c. Fungsi *Delete Hotel* (**SKPL-JOGJANESIA-WEB-004-03**).

Merupakan fungsi yang digunakan untuk menghapus data Hotel.

d. Fungsi *Display Hotel* (**SKPL-JOGJANESIA-WEB-004-04**).

Merupakan fungsi yang digunakan untuk menampilkan data hotel.

11. Fungsi Pengelolaan Hotel (**SKPL-JOGJANESIA-WEB-005**)

Program Studi Teknik Informatika	SKPL – JOGJANESIA	12/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan untuk mengelola data rental kendaraan.

Fungsi *Pengelolaan Rental Kendaraan* meliputi:

a. Fungsi *Add Rental Kendaraan* (**SKPL-JOGJANESIA-WEB-005-01**).

Merupakan fungsi yang digunakan untuk menambahkan data hotel.

b. Fungsi *Edit Rental Kendaraan* (**SKPL-JOGJANESIA-WEB-005-02**).

Merupakan fungsi yang digunakan untuk mengubah data desa wisata.

c. Fungsi *Delete Rental Kendaraan* (**SKPL-JOGJANESIA-WEB-005-03**).

Merupakan fungsi yang digunakan untuk menghapus data Hotel.

d. Fungsi *Display Rental Kendaraan* (**SKPL-JOGJANESIA-WEB-005-04**).

Merupakan fungsi yang digunakan untuk menampilkan data hotel.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak JOGJANESIA adalah sebagai berikut :

1. Memahami penggunaan smartphone.
2. Memahami pengoperasian Microsoft Windows.
3. Memahami pengoperasian Android.
4. Memahami pengoperasian XAMPP.

5. Mengerti tentang internet dan web.

6. Memahami penggunaan aplikasi JOGJANESIA.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak JOGJANESIA adalah :

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak JOGJANESIA.

2. Keterbatasan perangkat keras

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

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada web browser dan perangkat mobile dengan sistem operasi Android yang dilengkapi GPS.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

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

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk page atau form.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	14/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak JOGJANESIA adalah:

1. Perangkat *Dekstop*.
2. Perangkat *Database Server*.
3. Perangkat *Web Server*.
4. Perangkat *Mobile*.

3.1.3 Antarmuka perangkat lunak

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

- a. Nama : Android ver 4.0 keatas
Sumber : Android Open Source Project (AOSP)
Sebagai sistem operasi untuk smartphone.
- b. Nama : MySQL
Sumber : MySQL
Sebagai database management system (DBMS) yang digunakan untuk penyimpan data di sisi server.
Nama : Mixare edited by Justin Wetherell
Sumber : Code Google
Sebagai Framework dalam pembuatan Aplikasi Mobile.
- c. Nama : Apache
Sumber : Apache Software Foundation
Sebagai web server.
- d. Nama : PHP
Sumber : PHP
Sebagai penghubung aplikasi dengan database.
- e. Nama : IE/Firefox/Chrome/Opera/dll

Sumber : berbagai sumber

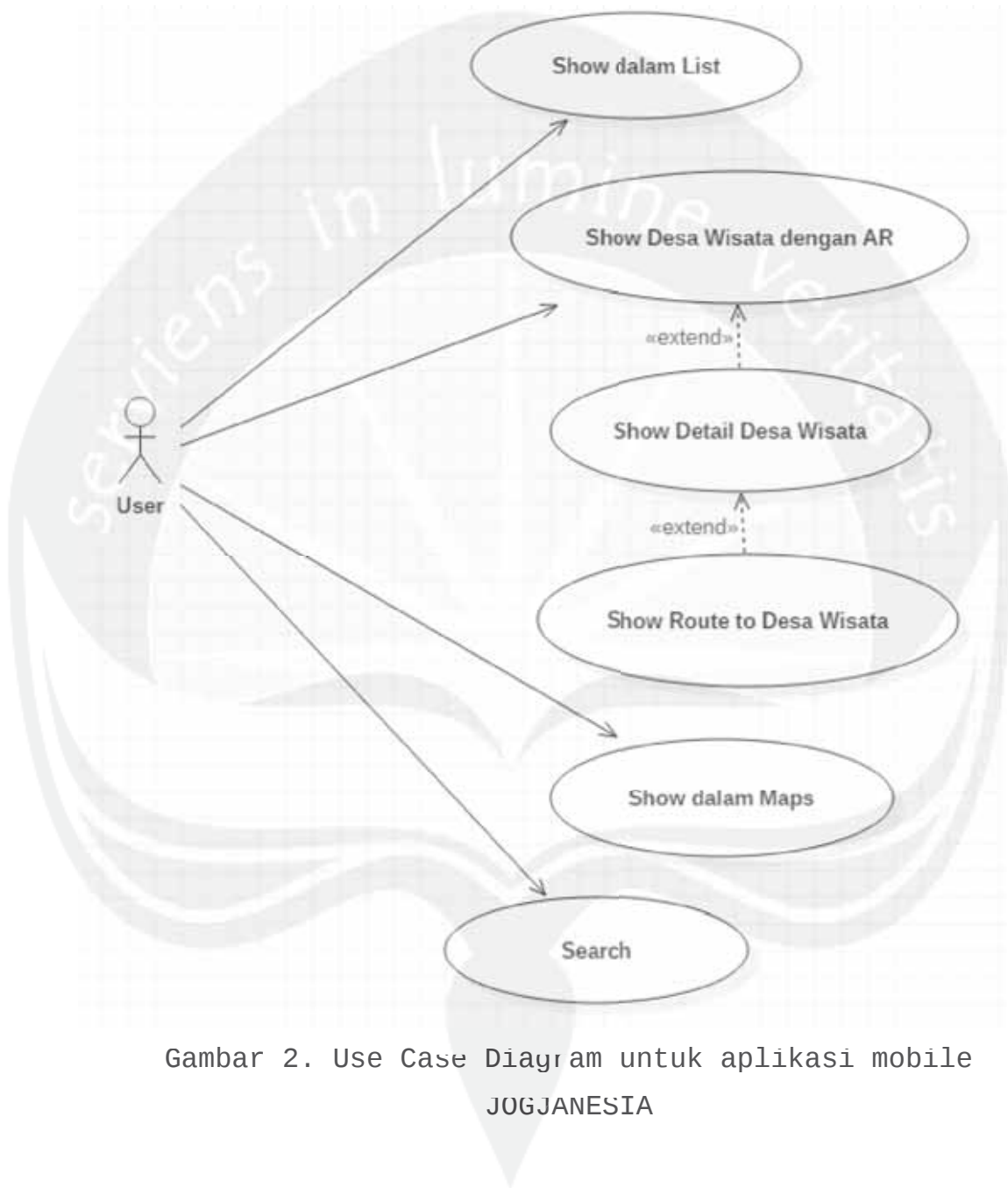
Sebagai aplikasi web browser untuk membuka sistem web.

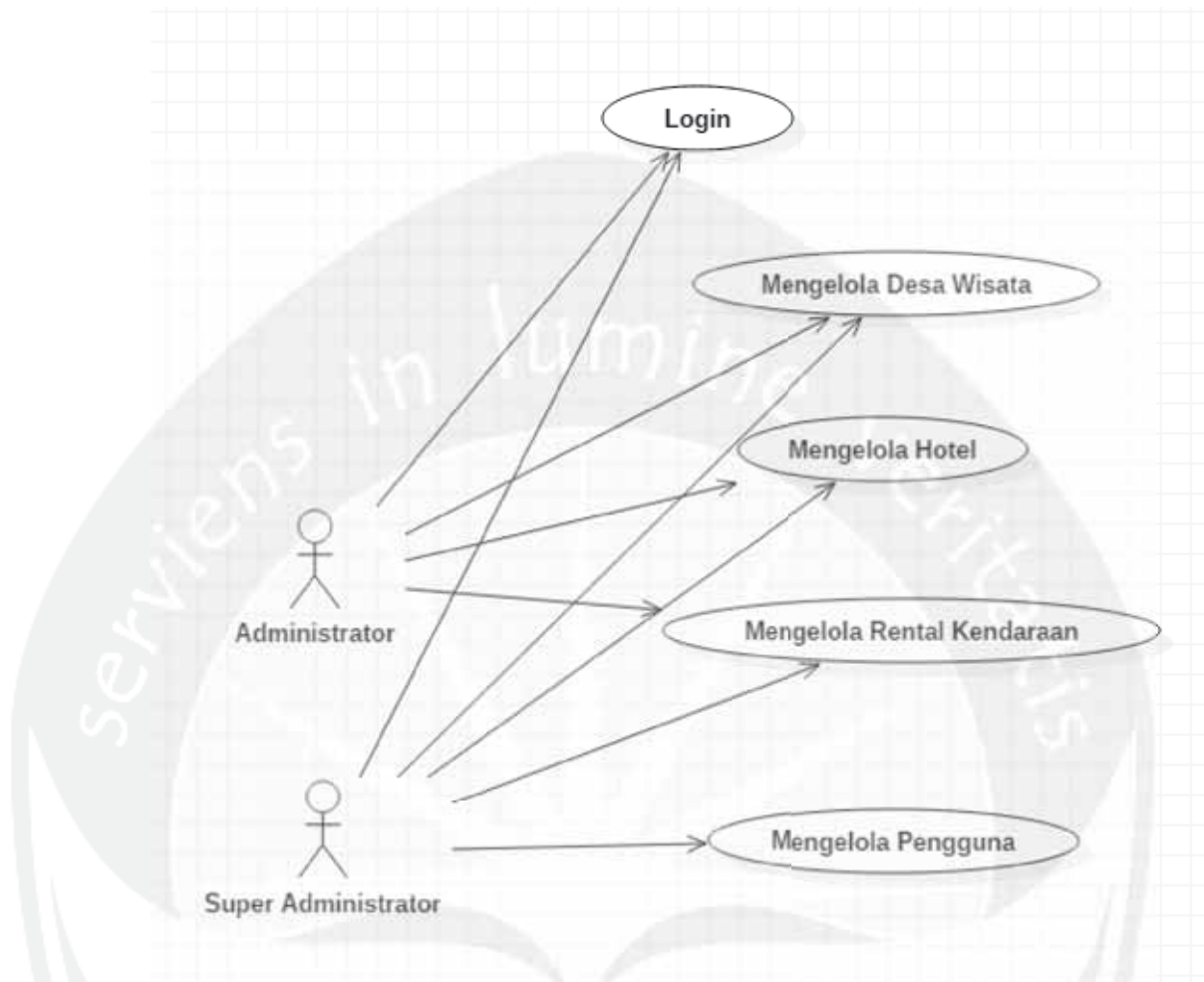
3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak JOGJANESIA menggunakan protocol TCP/IP, sedangkan untuk aplikasi web menggunakan protocol HTTP.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram





Gambar 3. Use Case Diagram untuk aplikasi web
JOGJANESIA-WEB

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

APLIKASI-MOBILE

4.1.1 Use case Specification : Show dalam List

1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan informasi desa wisata, hotel dan rental kendaraan dalam List.

2. Primary Actor

1. Pengguna

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan informasi desa wisata, hotel atau rental kendaraan yang ada dalam database.
2. Aplikasi memberikan pilihan untuk menampilkan informasi desa wisata, hotel atau rental kendaraan.
3. Aktor memilih untuk menampilkan informasi desa wisata.
 - A-1 Aktor memilih untuk menampilkan informasi hotel.
 - A-2 Aktor memilih untuk menampilkan informasi rental kendaraan.
4. Aplikasi menampilkan informasi desa wisata dalam List.
5. Use Case ini selesai.

5. Alternative Flow

- A-1 Aktor memilih untuk menampilkan informasi hotel
 1. Aplikasi menampilkan informasi hotel dalam List
 2. Usecase selesai.
- A-2 Aktor memilih untuk menampilkan informasi hotel
 1. Aplikasi menampilkan informasi hotel dalam List
 2. Usecase selesai.

6. Error Flow

-

7. PreConditions

1. Aplikasi berjalan dengan baik dan lancar.
2. Aktor memiliki koneksi internet sehingga dapat terhubung dengan sistem.
3. Sistem pada server berjalan dengan baik dan lancar.
4. Aktor memiliki koneksi GPS.

8. PostConditions

Informasi Desa wisata, hotel dan rental kendaraan sekitar telah tertampil.

4.1.2 Use case Specification : Show Desa Wisata dengan Augmented Reality

1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan lokasi desa wisata di sekitar pengguna dengan menggunakan fitur Augmented Reality.

2. Primary Actor

Pengguna

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk menampilkan lokasi desa wisata yang ada disekitar pengguna dengan menggunakan Augmented Reality.

2. Aplikasi menyediakan seek bar di bagian kanan untuk mengatur jarak antara pengguna dengan lokasi desa wisata.
3. Aplikasi memeriksa lokasi user.
E-1 Data lokasi user tidak didapatkan.
4. Aplikasi menampilkan desa wisata sekitar dengan Augmented Reality.
Extend Flow 1 – Show Detail Desa Wisata
Extend Flow 2 – Show Route to Desa Wisata
5. Use Case ini selesai.

5. Extend Flow

- Extend Flow 1 – Show Detail Desa Wisata
Detail desa wisata ditampilkan
- Extend Flow 2 – Show Route to Desa Wisata
Menampilkan rute dari posisi pengguna menuju desa wisata yang dipilih user
- Use Case ini selesai.

6. Error Flow

- E-1 Data lokasi user tidak didapatkan .
- a. Aplikasi menampilkan peringatan bahwa data lokasi tidak dapat didapatkan.
 - b. Kembali ke Basic Flow langkah ke-2.

7. PreConditions

- a. Aplikasi berjalan dengan baik dan lancar.
- b. Aktor memiliki koneksi internet sehingga dapat terhubung dengan sistem.
- c. Sistem pada server berjalan dengan baik dan lancar.
- d. Aktor memiliki koneksi GPS.

8. PostConditions

1. Informasi Desa wisata sekitar telah tertampil dengan menggunakan Augmented Reality.
2. Detail informasi desa wisata ditampilkan.
3. Rute dari pengguna menuju desa wisata telah ditampilkan

4.1.3 Use case Specification : Show dalam Maps

1. Brief Description

Merupakan fungsi yang digunakan oleh pengguna untuk menampilkan desa wisata, hotel atau rental kendaraan dalam Maps.

2. Primary Actor

Pengguna

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan menampilkan desa wisata, hotel atau rental kendaraan dalam Maps.
2. Aplikasi memeriksa lokasi user.
E-1 Data lokasi user tidak didapatkan.
3. Aplikasi menampilkan lokasi desa wisata dengan bentuk marker dalam maps.
A-1 Aplikasi menampilkan lokasi hotel dengan bentuk marker dalam maps.
A-2 Aplikasi menampilkan lokasi rental kendaraan dengan bentuk marker dalam maps.
4. Aplikasi memberikan keterangan singkat tentang desa wisata dalam bentuk pop-up jika di-touch

5. Use Case ini selesai.

5. Alternative Flow

A-1 Aplikasi menampilkan lokasi hotel dengan bentuk marker dalam maps.

1. Aplikasi memberikan keterangan singkat tentang desa wisata dalam bentuk pop-up jika di-touch

2. Use Case ini selesai

A-2 Aplikasi menampilkan lokasi rental kendaraan dengan bentuk marker dalam maps.

1. Aplikasi memberikan keterangan singkat tentang desa wisata dalam bentuk pop-up jika di-touch

2. Use Case ini selesai

6. Error Flow

E-1 Data lokasi user tidak didapatkan .

a. Aplikasi menampilkan peringatan bahwa data lokasi tidak dapat didapatkan.

b. Kembali ke Basic Flow langkah ke-2.

7. PreConditions

1. Aplikasi berjalan dengan baik dan lancar.
2. Aktor memiliki koneksi internet sehingga dapat terhubung dengan sistem.
3. Aplikasi pada server berjalan dengan baik dan lancar.

8. PostConditions

1. Data deskripsi lokasi telah tertampil.

4.1.4 Use case Spesification : Search

1. Brief Description

Program Studi Teknik Informatika	SKPL – JOGJANESIA	23/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Use Case ini digunakan oleh Pengguna untuk mencari desa wisata, hotel atau rental mobil berdasarkan keyword yang diinputkan user.

2. Primary Actor

1. Pengguna

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika pengguna memilih untuk mencari desa wisata, hotel atau rental mobil berdasarkan keyword tertentu.
2. Pengguna memasukkan keyword ke dalam field search
3. Aplikasi akan menampilkan hasil pencarian user dalam bentuk List
- E-1 keyword yang diinputkan user tidak cocok dengan data yang ada dalam database
4. Use Case ini selesai

5. Alternative Flow

none

6. Error Flow

- E-1 keyword yang diinputkan user tidak cocok dengan data yang ada dalam database.
1. Aplikasi menampilkan pesan bahwa data yang dicari tidak ada dalam database.
 2. Kembali ke Basic Flow langkah ke-2.

7. PreConditions

1. Aplikasi berjalan dengan baik dan lancar.
2. Aktor memiliki koneksi internet sehingga dapat terhubung dengan sistem.
3. Sistem pada server berjalan dengan baik dan lancar.

4. Aktor memiliki koneksi GPS.

8. PostConditions

Data yang dicari pengguna tertampil.

APLIKASI-WEB

4.1.5 Use case Spesification : Login

1. Brief Description

Use Case ini digunakan oleh Administrator untuk Login ke dalam web pengelolaan.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika pengguna memilih untuk Login ke dalam web pengeleolaan JOGJANESIA-WEB.
2. Aktor memasukkan nama pengguna dan password untuk login.
E-1 nama pengguna atau password yang dimasukkan salah
3. Aktor mengklik tombol login untuk dapat login ke dalam system.
4. Aplikasi menampilkan halaman utama web pengelolaan JOGJANESIA-WEB.
5. Use Case ini selesai

5. Alternative Flow

none

6. Error Flow

E-1 nama pengguna atau password yang dimasukkan salah.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	25/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Aplikasi menampilkan pesan bahwa nama atau password yang dimasukkan salah.

2. Kembali ke Basic *Flow* langkah ke-2.

7. PreConditions

1. Aplikasi berjalan dengan baik dan lancar.
2. Aktor memiliki koneksi internet sehingga dapat terhubung dengan sistem.
3. Sistem pada server berjalan dengan baik dan lancar.

8. PostConditions

Aktor dapat masuk ke halaman web pengelolaan JOGJANESIA-WEB.

4.1.6 Use case Spesification : Pengelolaan Data Pengguna.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Pengguna JOGJANESIA-WEB. Aktor dapat melakukan *add* data pengguna, *edit* data pengguna, *delete* data pengguna, *display* data pengguna, and *search* data pengguna.

2. Primary Actor

1. Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pengguna.
2. Aktor memilih untuk melakukan *add* data pengguna.

A-1 Aktor memilih untuk melakukan *edit* data pengguna.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	26/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-2 Aktor memilih untuk melakukan *display* data pengguna.

A-3 Aktor memilih untuk melakukan *delete* data pengguna.

3. Aktor menginputkan data pengguna.

□ Aktor meminta sistem untuk menyimpan data pengguna yang telah diinputkan.

5. Sistem mengecek data pengguna yang telah diinputkan.

E-1 Data pengguna yang diinputkan aktor salah/kurang lengkap.

6. Sistem menyimpan data pengguna ke database.

7. Use Case ini selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan *edit* data pengguna.

1. Sistem menampilkan data pengguna.

2. Aktor mengedit data pengguna yang sudah ditampilkan.

3. Aktor meminta sistem untuk menyimpan data pengguna yang telah diedit.

4. Sistem melakukan pengecekan terhadap data pengguna yang telah diedit

E-2 Data pengguna yang telah diedit salah.

5. Sistem menyimpan data pengguna yang telah diedit ke database.

6. Berlanjut ke Basic Flow langkah ke 7.

A-2 Aktor memilih untuk melakukan *display* data pengguna.

1. Sistem menampilkan data pengguna.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	27/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Berlanjut ke Basic Flow langkah ke 7.

A-3 Aktor memilih untuk melakukan *delete* data pengguna.

1. Sistem menampilkan data pengguna.
2. Aktor memilih untuk menghapus data pengguna.
3. Sistem menampilkan pertanyaan konfirmasi apakah ingin menghapus data pengguna?
4. Aktor mengkonfirmasi ingin menghapus data pengguna.
5. Sistem mengupdate data pengguna yang ada ke database.
6. Berlanjut ke Basic Flow langkah 7

A-4

6. Error Flow

E-1 Data pengguna yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah.
2. Sistem memberi penanda pada field yang salah.
3. Kembali ke Basic Flow Langkah ke 4.

E-2 Data pengguna yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diedit salah.
2. Sistem memberi penanda pada field yang salah.
3. Kembali ke Alternative Flow A-1 Langkah ke 2.

7. PreConditions

1. Use Case *Login* telah dilakukan.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	28/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Aktor telah memasuki sistem.

8. PostConditions

Data pengguna dalam database telah terupdate.

4.1.7 Use case Spesification : Pengelolaan Data Desa Wisata.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Desa wisata JOGJANESIA-WEB. Aktor dapat melakukan *add* data desa wisata, *edit* data desa wisata, *delete* data desa wisata, *display* data desa wisata, and *search* data desa wisata.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data desa wisata.

2. Aktor memilih untuk melakukan *add* data desa wisata.

A-1 Aktor memilih untuk melakukan *edit* data desa wisata.

A-2 Aktor memilih untuk melakukan *display* data desa wisata.

A-3 Aktor memilih untuk melakukan *delete* data desa wisata.

3. Aktor menginputkan data desa wisata.

□□ Aktor meminta sistem untuk menyimpan data desa wisata yang telah diinputkan.

5. Sistem mengecek data desa wisata yang telah diinputkan.

E-1 Data desa wisata yang diinputkan aktor salah/kurang lengkap.

6. Sistem menyimpan data desa wisata ke database.

7. Use Case ini selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan *edit* data desa wisata.

1. Sistem menampilkan data desa wisata.

2. Aktor mengedit data desa wisata yang sudah ditampilkan.

3. Aktor meminta sistem untuk menyimpan data desa wisata yang telah diedit.

4. Sistem melakukan pengecekan terhadap data desa wisata yang telah diedit

E-2 Data desa wisata yang telah diedit salah.

5. Sistem menyimpan data desa wisata yang telah diedit ke database.

6. Berlanjut ke Basic Flow langkah ke 7.

A-2 Aktor memilih untuk melakukan *display* data desa wisata.

1. Sistem menampilkan data desa wisata.

2. Berlanjut ke Basic Flow langkah ke 7.

A-3 Aktor memilih untuk melakukan *delete* data desa wisata.

3. Sistem menampilkan data desa wisata.

4. Aktor memilih untuk menghapus data desa wisata.

5. Sistem menampilkan pertanyaan konfirmasi apakah ingin menghapus data desa wisata.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	30/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Aktor mengkonfirmasi ingin menghapus data desa wisata.
7. Sistem mengupdate data desa wisata yang ada ke database.
8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Data desa wisata yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah.
2. Sistem memberi penanda pada field yang salah.
3. Kembali ke Basic Flow Langkah ke 4.

E-2 Data desa wisata yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diedit salah.
2. Sistem memberi penanda pada field yang salah.
3. Kembali ke Alternative Flow A-1 Langkah ke 2.

7. PreConditions

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

8. PostConditions

Data desa wisata dalam database telah terupdate.

4.1.8 Use case Spesification : Pengelolaan Data Hotel.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Hotel JOGJANESIA-WEB. Aktor dapat melakukan *add* data hotel, *edit* data hotel, *delete*

Program Studi Teknik Informatika	SKPL – JOGJANESIA	31/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

data hotel, *display* data hotel, and *search* data hotel.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

3. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data hotel.

8. Aktor memilih untuk melakukan *add* data hotel.

A-1 Aktor memilih untuk melakukan *edit* data hotel.

A-2 Aktor memilih untuk melakukan *display* data hotel.

A-3 Aktor memilih untuk melakukan *delete* data hotel.

9. Aktor menginputkan data hotel.

□□ Aktor meminta sistem untuk menyimpan data hotel yang telah diinputkan.

11. Sistem mengecek data hotel yang telah diinputkan.

E-1 Data hotel yang diinputkan aktor salah/kurang lengkap.

12. Sistem menyimpan data hotel ke database.

13. Use Case ini selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan *edit* data hotel.

1. Sistem menampilkan data hotel.

2. Aktor mengedit data hotel yang sudah ditampilkan.

3. Aktor meminta sistem untuk menyimpan data hotel yang telah diedit.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	32/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. Sistem melakukan pengecekan terhadap data hotel yang telah diedit

E-2 Data hotel yang telah diedit salah.

5. Sistem menyimpan data hotel yang telah diedit ke database.

6. Berlanjut ke Basic Flow langkah ke 7.

A-2 Aktor memilih untuk melakukan *display* data hotel.

1. Sistem menampilkan data hotel.

2. Berlanjut ke Basic Flow langkah ke 7.

A-3 Aktor memilih untuk melakukan *delete* data hotel.

3. Sistem menampilkan data hotel.

4. Aktor memilih untuk menghapus data hotel.

5. Sistem menampilkan pertanyaan konfirmasi apakah ingin menghapus data hotel.

6. Aktor mengkonfirmasi ingin menghapus data hotel.

7. Sistem mengupdate data hotel yang ada ke database.

8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Data hotel yang diinputkan aktor salah.

2. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah.

4. Sistem memberi penanda pada field yang salah.

3. Kembali ke Basic Flow Langkah ke 4.

E-2 Data hotel yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diedit salah.

2. Sistem memberi penanda pada field yang salah.

3. Kembali ke Alternative Flow A-1 Langkah ke 2.

7. PreConditions

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

8. PostConditions

Data hotel dalam database telah terupdate.

4.1.9 asdas

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Rental kendaraan JOGJANESIA-WEB. Aktor dapat melakukan *add* data rental kendaraan, *edit* data rental kendaraan, *delete* data rental kendaraan, *display* data rental kendaraan, and *search* data rental kendaraan.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

5. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data rental kendaraan.

14. Aktor memilih untuk melakukan *add* data rental kendaraan.

A-1 Aktor memilih untuk melakukan *edit* data rental kendaraan.

A-2 Aktor memilih untuk melakukan *display* data rental kendaraan.

A-3 Aktor memilih untuk melakukan *delete* data rental kendaraan.

15. Aktor menginputkan data rental kendaraan.

□□ Aktor meminta sistem untuk menyimpan data rental kendaraan yang telah diinputkan.

17. Sistem mengecek data rental kendaraan yang telah diinputkan.

E-1 Data rental kendaraan yang diinputkan aktor salah/kurang lengkap.

18. Sistem menyimpan data rental kendaraan ke database.

19. Use Case ini selesai.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan *edit* data rental kendaraan.

1. Sistem menampilkan data rental kendaraan.

2. Aktor mengedit data rental kendaraan yang sudah ditampilkan.

3. Aktor meminta sistem untuk menyimpan data rental kendaraan yang telah diedit.

4. Sistem melakukan pengecekan terhadap data rental kendaraan yang telah diedit

E-2 Data rental kendaraan yang telah diedit salah.

5. Sistem menyimpan data rental kendaraan yang telah diedit ke database.

6. Berlanjut ke Basic Flow langkah ke 7.

A-2 Aktor memilih untuk melakukan *display* data rental kendaraan.

1. Sistem menampilkan data rental kendaraan.

2. Berlanjut ke Basic Flow langkah ke 7.

A-3 Aktor memilih untuk melakukan *delete* data rental kendaraan.

3. Sistem menampilkan data rental kendaraan.

4. Aktor memilih untuk menghapus data rental kendaraan.

5. Sistem menampilkan pertanyaan konfirmasi apakah ingin menghapus data rental kendaraan.

6. Aktor mengkonfirmasi ingin menghapus data rental kendaraan.

7. Sistem mengupdate data rental kendaraan yang ada ke database.

8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Data rental kendaraan yang diinputkan aktor salah.

3. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah.

6. Sistem memberi penanda pada field yang salah.

3. Kembali ke Basic Flow Langkah ke 4.

E-2 Data rental kendaraan yang diinputkan aktor salah

1. Sistem memberikan pesan peringatan bahwa data yang diedit salah.

2. Sistem memberi penanda pada field yang salah.

3. Kembali ke Alternative Flow A-1 Langkah ke 2.

7. PreConditions

1. Use Case *Login* telah dilakukan.

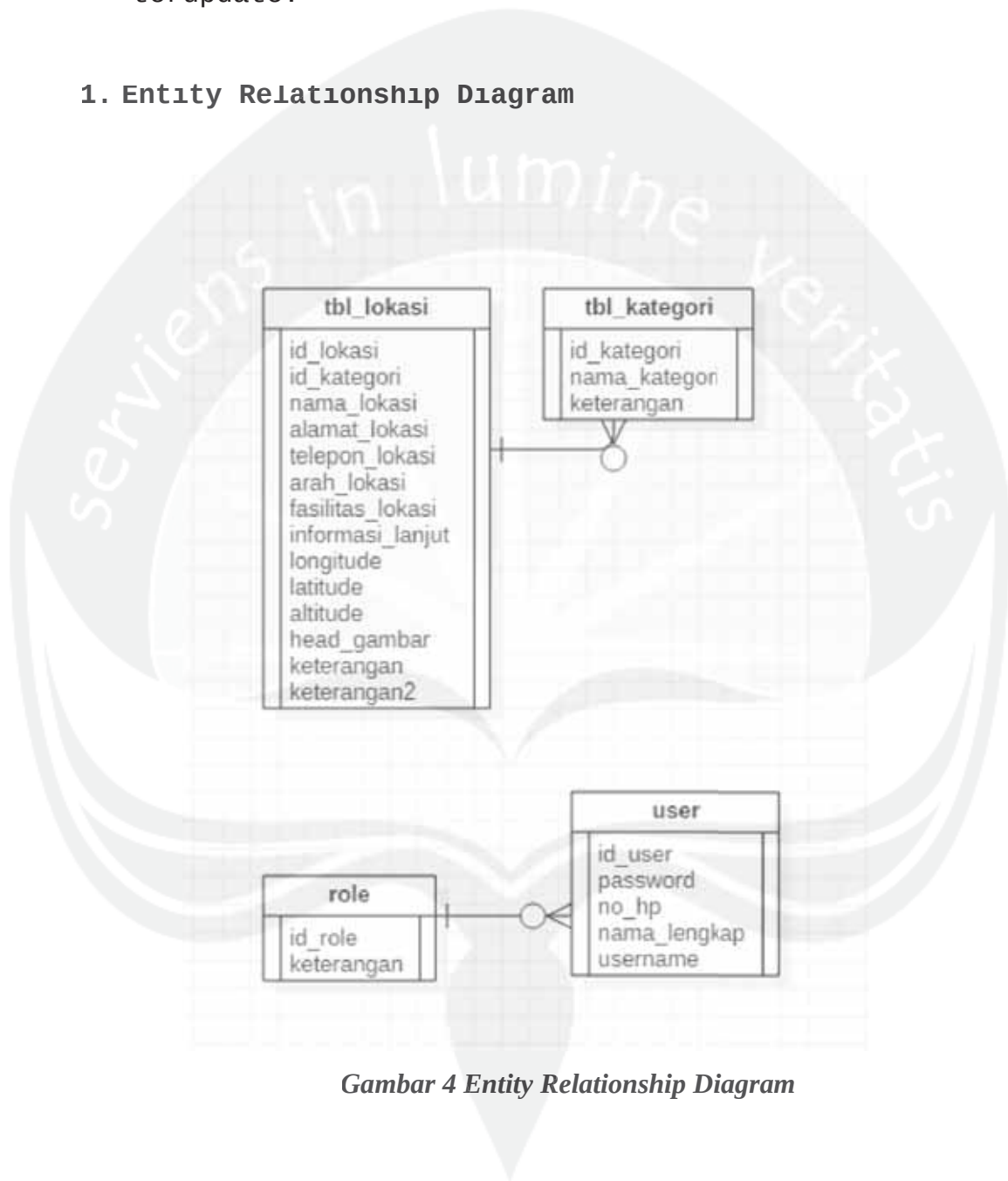
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – JOGJANESIA	36/ 37
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bJOGJANESIAifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

Data rental kendaraan dalam database telah terupdate.

1. Entity Relationship Diagram



Gambar 4 Entity Relationship Diagram