

BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan penelitian yang dilakukan oleh penulis dengan mengamati dan menganalisa sistem pengolahan data digunakan serta dilandasi oleh teori-teori dan alat-alat yang digunakan berkaitan dengan penelitian, maka dapat ditarik kesimpulan dari Tugas Akhir ini yaitu :

1. Telah berhasil dibangun aplikasi *mobile* dan web untuk navigasi pada persewaan mobil berdasarkan lokasi pengguna dengan lokasi tujuan yang ditampilkan pada *google maps* dengan rute berbentuk garis merah atau disebut *polyline*.
2. Aplikasi RENTCAR dapat menerapkan sistem navigasi petunjuk jalan pada aplikasi *mobile* dengan memanfaatkan *google API key* yang mengambil koordinat lokasi berdasarkan *latitude* dan *longitude* dari pengguna.
3. Aplikasi RENTCAR dapat mengirim atau *upload* foto bukti pembayaran yang dapat di ambil dari kamera maupun *gallery smartphone* yang digunakan pengguna.
4. Aplikasi RENTCAR dapat mengirim dan menerima email yang terkoneksi dengan akun *gmail* atau akun lainnya.
5. Aplikasi RENTCAR dapat melakukan telepon dengan nomor yang disediakan pada aplikasi dan dapat mengunjungi sosial media *facebook* atau *twitter*.

6.2 Saran

Dari proses analisis, perancangan, implementasi hingga pengujian sistem pada pembuatan Tugas Akhir, didapatkan beberapa saran untuk pengembangan lebih lanjut dari perangkat lunak RENTCAR, yaitu :

1. Aplikasi RENTCAR dapat dikembangkan lebih lanjut untuk bagian *geolocation* agar dapat menghasilkan pengambilan *latitude* dan *longitude* lokasi keberadaan mobil yang lebih akurat yang nantinya digunakan untuk memonitor mobil yang sedang disewa.
2. Perlunya penambahan fungsionalitas pada *mobile* mengenai *history latitude* dan *longitude* lokasi keberadaan mobil yang sedang disewa secara *realtime*.
3. Menambahkan fungsionalitas *chatting* pada *mobile* dan web untuk memudahkan proses tawar-menawar harga sewa mobil yang memungkinkan pelanggan dan petugas rental dapat berkomunikasi *textstual* secara langsung.

DAFTAR PUSTAKA

- Android. (2015). *What Is Android*. Retrieved Mei 26, 2015, from <http://developer.android.com/guide/basics/what-is-android.html>
- Anindito, K., Darma P, I., & Wuryo Putro, S. (2012). Perancangan Aplikasi Mobile City Directory Yogyakarta Berbasis Android. *Seminar Nasional Teknologi Informasi dan Komunikasi 2012 (SENTIKA 2012)* , 7.
- Aryanto, A., & Tjendrowasono, T. (2013). Pembangunan Sistem Penjualan Online pada Toko Indah Jaya Furniture Surakarta. *Indonesian Journal on Computer Science - Speed* , X (1), 55-61.
- Axel, K., & Treu, G. (2005). *From Location to Position Management: User Tracking or Location-based Services*. Germany: Institute for Informatics University of Munich.
- Fatmanto, A. (2013). *Pembangunan Aplikasi Forum Jual Beli Berbasis Mobile*. Skripsi, Universitas Atma Jaya Yogyakarta Universitas Atma Jaya Yogyakarta, Yogyakarta.
- Firdaus, M. R., Hidayat, M. N., & K, S. Y. (2009). *Pengembangan Sistem Mobile Commerce Pada Proses Rental Mobil Menggunakan Rapid Application Development*. Teknik Informatika UNILA and Teknik Informatika UNSOED.
- Karlina W P, D., & Rizky Wicaksono, S. (2014). Sistem Informasi E-Logistik Pada Siklus Proses Order BahanBaku. *Jurnal Buana Informatika*5 , 5 (2), 95-104.
- Nyura, Y. (2010). Pembuatan Aplikasi Pembelajaran Bahasa Inggris Pada Handphone dengan J2ME . *Jurnal Informatika Mulawarman* , 5, 18-27.
- Pakereng, A., Beeh, Y., Basten, M., & M.A.I. (2012). *Perancangan dan Implementasi Aplikasi Navigasi Objek Wisata di Kota Salatiga Berbasis Teknologi Android*.

Thesis, Program Studi Teknik Informatika FTI-UKSW Universitas Kristen Satya Wacana, Salatiga.

Paryanto, D., Nurgiyatna, & Rakhmadi, A. (2014). *Aplikasi Pemesanan Penyewaan Rental Mobil Di Prima Tour*. Surakarta: Universitas Muhammadiyah Surakarta.

Ristian Handoyo, E. (2013). *Pembangunan Aplikasi Panduan Transportasi Wisata Di Yogyakarta Berbasis Lokasi Pada Perangkat Mobile*. Yogyakarta: Universitas Atma Jaya Yogyakarta.

Sasmito, S. M., Suprayogi, S. T., & Hati, G. (2013). Aplikasi Penanda Lokasi Peta Digital Berbasis Mobile GIS Pada Smartphone Android. *Jurnal Geodesi Undip*, 2 (4), 26-40.

Solichin, A. (2012). *Pemrograman Web. In Pengenalan Web Server dan Server Side Scripting*. UPT PIMK STT Ronggolawe Cepu, Cepu.

Stefanus Diptya, A. A. (2012). *Pembangunan Aplikasi Android Untuk Layanan Pesan Antar Barang Atau Jasa Berbasis Lokasi Dan Teknologi Push Notification*. Yogyakarta: Universitas Atma Jaya Yogyakarta.

Yani, M. (2013). *Media Pembelajaran Matematika Untuk Balita Dengan Metode Glenn Doman Berbasis Android*. Tugas Akhir, Universitas Widyatama Universitas Widyatama, Bandung.



DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

PEMBANGUNAN APLIKASI MOBILE UNTUK
NAVIGASI PADA PERSEWAAN MOBIL

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Firmansyah Abdul Rafi / 6695

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		<i>DPPL-RENTCAR</i>		1/72
	Fakultas Teknologi Industri	<i>Revisi</i>		

Program Studi Teknik Informatika	DPPL – RENTCAR	1 / 72
----------------------------------	----------------	--------

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

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.....	8
1.1 Tujuan.....	8
1.2 Lingkup Masalah.....	8
1.3 Definisi, Akronim dan Singkatan.....	9
1.4 Referensi.....	10
2. Perancangan Sistem.....	11
2.1 Perancangan Asitektur.....	11
2.1.1 Perancangan Asitektur Mobile Apss.....	11
2.1.2 Perancangan Asitektur Web Apss.....	12
2.2 Perancangan Rinci.....	13
2.2.1 Sequence Diagram Rentcar Apps.....	13
2.2.1.1 Fungsi Login <i>Mobile</i>	13
2.2.1.2 Fungsi Registrasi <i>Mobile</i>	13
2.2.1.3 Fungsi Kelola Data Pelanggan <i>Mobile</i>	14
2.2.1.4 Fungsi Find Data Mobil <i>Mobile</i>	14
2.2.1.5 Fungsi Booking Mobil <i>Mobile</i>	15
2.2.1.6 Fungsi Geolocation <i>Mobile</i>	15
2.2.1.7 Fungsi Email <i>Mobile</i>	16
2.2.1.8 Fungsi Login Web.....	16
2.2.1.9 Fungsi Reset Password Pelanggan Web.....	17
2.2.1.10 Fungsi Delete Data Pelanggan Web.....	17
2.2.1.11 Fungsi Insert Data Mobil Web.....	18
2.2.1.12 Fungsi Edit Data Mobil Web.....	18
2.2.1.13 Fungsi Delete Data Mobil Web.....	19
2.2.1.14 Fungsi Show Data Mobil Web.....	19
2.2.1.15 Fungsi Update Status Pembayaran Web.....	20
2.2.1.16 Fungsi Create Email Web.....	20
2.2.1.17 Fungsi Reply Email Web.....	21
2.2.1.18 Fungsi Hapus Data Email Web.....	21
2.2.1.19 Fungsi Cari Lokasi Mobil Web.....	22
2.2.2 Class Diagram.....	23
2.2.2.1 Class Diagram Mobile Apps.....	23
2.2.2.2 Class Diagram Web Apps.....	24
2.2.3 Class Diagram Specific Descriptions.....	25
2.2.3.1 <i>Specific Design Class</i> login_layout.....	25
2.2.3.2 <i>Specific Design Class</i> daftar_user_layout.....	25
2.2.3.3 <i>Specific Design Class</i> my_account_layout.....	26
2.2.3.4 <i>Specific Design Class</i> update_account_layout.....	26
2.2.3.5 <i>Specific Design Class</i> fragment_tab_layout.....	26
2.2.3.6 <i>Specific Design Class</i> activity_search_layout.....	27
2.2.3.7 <i>Specific Design Class</i> show_mobil_layout.....	27
2.2.3.8 <i>Specific Design Class</i> booking_mobil_layout.....	27
2.2.3.9 <i>Specific Design Class</i> activity_search_layout.....	28
2.2.3.10 <i>Specific Design Class</i> Class map_user_layout.....	28
2.2.3.11 <i>Specific Design Class</i> map_rentcar_layout.....	28
2.2.3.12 <i>Specific Design Class</i> email_inbox_layout.....	28
2.2.3.13 <i>Specific Design Class</i> email_show_layout.....	29
2.2.3.14 <i>Specific Design Class</i> email_reply_layout.....	29
2.2.3.15 <i>Specific Design Class</i> email_create_layout.....	29
2.2.3.16 <i>Specific Design Class</i> FragmentLogin.....	30
2.2.3.17 <i>Specific Design Class</i> DaftarUser.....	30
2.2.3.18 <i>Specific Design Class</i> FragmentMyAccount.....	30
2.2.3.19 <i>Specific Design Class</i> UpdateAccount.....	31

2.2.3.20	<i>Specific Design Class</i>	TabSwipe.....	31
2.2.3.21	<i>Specific Design Class</i>	MobilSearch.....	32
2.2.3.22	<i>Specific Design Class</i>	MobilShow.....	32
2.2.3.23	<i>Specific Design Class</i>	MobilBooking.....	32
2.2.3.24	<i>Specific Design Class</i>	TransaksiSearch.....	33
2.2.3.25	<i>Specific Design Class</i>	MapUser.....	33
2.2.3.26	<i>Specific Design Class</i>	MapRentcar.....	34
2.2.3.27	<i>Specific Design Class</i>	FragmentEmail.....	34
2.2.3.28	<i>Specific Design Class</i>	EmailShow.....	34
2.2.3.29	<i>Specific Design Class</i>	EmailReply.....	35
2.2.3.30	<i>Specific Design Class</i>	CreateEmail.....	35
2.2.3.31	<i>Specific Design Class</i>	s_m_user.....	35
2.2.3.32	<i>Specific Design Class</i>	s_m_mobil.....	36
2.2.3.33	<i>Specific Design Class</i>	s_m_transaksi.....	36
2.2.3.34	<i>Specific Design Class</i>	s_m_email.....	37
2.2.3.35	<i>Specific Design Class</i>	v_login.....	37
2.2.3.36	<i>Specific Design Class</i>	v_pelanggan.....	37
2.2.3.37	<i>Specific Design Class</i>	v_inputdatamobil.....	38
2.2.3.38	<i>Specific Design Class</i>	v_editdatamobil.....	38
2.2.3.39	<i>Specific Design Class</i>	v_mobil.....	38
2.2.3.40	<i>Specific Design Class</i>	v_transaksi.....	38
2.2.3.41	<i>Specific Design Class</i>	v_detail_transaksi.....	39
2.2.3.42	<i>Specific Design Class</i>	v_emailbroadcast.....	39
2.2.3.43	<i>Specific Design Class</i>	v_emailinbox.....	39
2.2.3.44	<i>Specific Design Class</i>	v_emailread.....	39
2.2.3.45	<i>Specific Design Class</i>	v_emailsend.....	40
2.2.3.46	<i>Specific Design Class</i>	v_emailsendread.....	40
2.2.3.47	<i>Specific Design Class</i>	v_map.....	40
2.2.3.48	<i>Specific Design Class</i>	c_login.....	41
2.2.3.49	<i>Specific Design Class</i>	c_pelanggan.....	41
2.2.3.50	<i>Specific Design Class</i>	c_inputdatamobil.....	41
2.2.3.51	<i>Specific Design Class</i>	c_mobil.....	42
2.2.3.52	<i>Specific Design Class</i>	c_transaksi.....	42
2.2.3.53	<i>Specific Design Class</i>	c_detai_transaksi.....	42
2.2.3.54	<i>Specific Design Class</i>	c_email.....	43
2.2.3.55	<i>Specific Design Class</i>	c_map.....	43
2.2.3.56	<i>Specific Design Class</i>	m_login.....	44
2.2.3.57	<i>Specific Design Class</i>	m_pelanggan.....	44
2.2.3.58	<i>Specific Design Class</i>	m_mobil.....	44
2.2.3.59	<i>Specific Design Class</i>	m_transaksi.....	45
2.2.3.60	<i>Specific Design Class</i>	m_email.....	45
2.2.3.61	<i>Specific Design Class</i>	m_lokasi.....	45
3.	Perancangan Data.....		46
3.1	Dekomposisi Data.....		46
3.1.1	Deskripsi Entitas TBL_Admin.....		46
3.1.2	Deskripsi Entitas TBL_User.....		46
3.1.3	Deskripsi Entitas TBL_Mobil.....		46
3.1.4	Deskripsi Entitas TBL_Transaksi.....		47
3.1.5	Deskripsi Entitas TBL_Email.....		48
3.1.6	Deskripsi Entitas TBL_EmailSend.....		48
3.1.7	Deskripsi Entitas TBL_Lokasi.....		49
3.1.8	Deskripsi Entitas TBL_Image.....		49
3.2	Physical Data Model.....		51
4.	Deskripsi Perancangan Antarmuka		52
4.1	Antarmuka Login Mobile.....		52
4.2	Antarmuka Registrasi Pelanggan.....		53

4.3	Antarmuka Pengelolaan Data Pelanggan.....	54
4.4	Antarmuka List Home Mobil.....	55
4.5	Antarmuka Show Detail Mobil.....	56
4.6	Antarmuka Transaksi Mobil.....	57
4.7	Antarmuka Detail Transaksi Mobil.....	58
4.8	Antarmuka Konfirmasi Pembayaran.....	59
4.9	Antarmuka Pengelolaan Data Email.....	60
4.10	Antarmuka Geolocation.....	61
4.11	Antarmuka Login Web.....	62
4.12	Antarmuka Input New Mobil.....	63
4.13	Antarmuka Show Pelanggan.....	64
4.14	Antarmuka Show Mobil Web.....	65
4.15	Antarmuka Show Pemesanan.....	66
4.16	Antarmuka Show Detail Pemesanan.....	67
4.17	Antarmuka Show Konfirmasi Pembayaran.....	68
4.18	Antarmuka Email Inbox Web.....	69
4.19	Antarmuka Read Email Web.....	70
4.20	Antarmuka Email Reply Web.....	71
4.21	Antarmuka Cari Lokasi Mobil.....	72

DAFTAR GAMBAR

Gambar 2.1 Perancangan Arsitektur Rentcar Mobile Apps.....	11
Gambar 2.2 Perancangan Arsitektur Rentcar Web Apps.....	12
Gambar 2.3 <i>Sequence Diagram</i> : Fungsi Login Mobile.....	13
Gambar 2.4 <i>Sequence Diagram</i> : Fungsi Registrasi Mobile.....	13
Gambar 2.5 <i>Sequence Diagram</i> : Fungsi Data Pelanggan Mobile.....	14
Gambar 2.6 <i>Sequence Diagram</i> : Fungsi Find Data Mobil Mobile....	14
Gambar 2.7 <i>Sequence Diagram</i> : Fungsi Booking Mobil Mobile.....	15
Gambar 2.8 <i>Sequence Diagram</i> : Fungsi Geolocation Mobile.....	15
Gambar 2.9 <i>Sequence Diagram</i> : Fungsi Email Mobile.....	16
Gambar 2.10 <i>Sequence Diagram</i> : Fungsi Login Web.....	16
Gambar 2.11 <i>Sequence Diagram</i> : Reset Password Pelanggan Web....	17
Gambar 2.12 <i>Sequence Diagram</i> : Fungsi Delete Pelanggan Web....	17
Gambar 2.13 <i>Sequence Diagram</i> : Fungsi Insert Data Mobil Web....	18
Gambar 2.14 <i>Sequence Diagram</i> : Fungsi Edit Data Mobil Web.....	18
Gambar 2.15 <i>Sequence Diagram</i> : Fungsi Delete Data Mobil Web....	19
Gambar 2.16 <i>Sequence Diagram</i> : Fungsi Show Data Mobil Web.....	19
Gambar 2.17 <i>Sequence Diagram</i> : Fungsi Update Status Pembayaran.	20
Gambar 2.18 <i>Sequence Diagram</i> : Fungsi Create Email Web.....	20
Gambar 2.19 <i>Sequence Diagram</i> : Fungsi Reply Email Web.....	21
Gambar 2.20 <i>Sequence Diagram</i> : Fungsi Hapus Data Email Web....	21
Gambar 2.21 <i>Sequence Diagram</i> : Fungsi Cari Lokasi Mobil.....	22
Gambar 2.22 Class Diagram Mobile Apps.....	23
Gambar 2.23 Class Diagram Web Apps.....	24
Gambar 3.1 Physical Data Model.....	51
Gambar 4.1 Antarmuka Login Mobile.....	52
Gambar 4.2 Antarmuka Registrasi User.....	53
Gambar 4.3 Antarmuka Pengelolaan Data Pelanggan.....	54
Gambar 4.4 Antarmuka List Home Mobil.....	55
Gambar 4.5 Antarmuka Show Detail Mobil.....	56
Gambar 4.6 Antarmuka Transaksi Mobil.....	57
Gambar 4.7 Antarmuka Detail Transaksi Mobil.....	58
Gambar 4.8 Antarmuka Konfirmasi Pembayaran.....	59
Gambar 4.9 Antarmuka Pengelolaan Data Email.....	60
Gambar 4.10 Rancangan Geolocation.....	61
Gambar 4.11 Antarmuka Login Web.....	62
Gambar 4.12 Antarmuka Input New Mobil.....	63
Gambar 4.13 Antarmuka Show Pelanggan.....	64
Gambar 4.14 Antarmuka Show Mobil Web.....	65
Gambar 4.15 Antarmuka Show Pemesanan.....	66
Gambar 4.16 Antarmuka Show Detail Pemesanan.....	67
Gambar 4.17 Antarmuka Show Konfirmasi Pembayaran.....	68
Gambar 4.18 Antarmuka Email Inbox Web.....	69
Gambar 4.19 Antarmuka Read Email Web.....	70
Gambar 4.20 Antarmuka Email Reply Web.....	71
Gambar 4.21 Antarmuka Cari Lokasi Mobil.....	72

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 RENTCAR dikembangkan dengan tujuan untuk :

1. Pengelolaan data pelanggan sebagai member.
2. Pengelolaan pencatatan data persewaan mobil.
3. Memberikan informasi mobil yang disediakan.
4. Pengelolaan pembayaran persewaan mobil dengan pelanggan.
5. Navigasi petunjuk arah menggunakan *geolocation* untuk pengambil atau pengantar mobil.
6. mencari lokasi mobil yang sedang di sewa.
7. Email sebagai penghubung antara pelanggan dan petugas untuk penawaran harga sewa mobil.

Perangkat Lunak RENTCAR bisa berjalan pada lingkungan dengan *mobile* dan *web*.

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-RENTCAR-XXX	Kode yang merepresentasikan kebutuhan pada Aplikasi <i>Mobile</i> Untuk Navigasi Pada Persewaan Mobil di mana XXX merupakan nomor fungsi produk.
RENTCAR	Perangkat lunak pengelolaan data yang berjalan di perangkat <i>mobile</i> dan web dalam kasus persewaan mobil.
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 <i>e-mail</i> , <i>FTP</i> , dan <i>World Wide Web</i> .
Administrator	Petugas yang mengolah data master rental mobil Nayara Transport.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.

Rent Car Nayara Transport	Nayara Transport adalah sebuah perusahaan yang bergerak di bidang jasa transportasi, perusahaan ini menawarkan harga sewa mobil yang murah, kompetitif, dan selalu mengutamakan kualitas kenyamanan pelayanan kepada pelanggan. Nayara Transport bertujuan untuk menjadikan perusahaannya sebagai sebuah perusahaan persewaan mobil yang terpercaya bagi pelanggan yang mengutamakan pelayanan guna bersaing dengan perusahaan persewaan mobil lainnya.
Pelanggan	Pelanggan dari Rent Car Nayara Transport yang terdaftar sebagai member untuk menyewa mobil di rental ini melalui aplikasi <i>mobile</i> .

1.4 Referensi

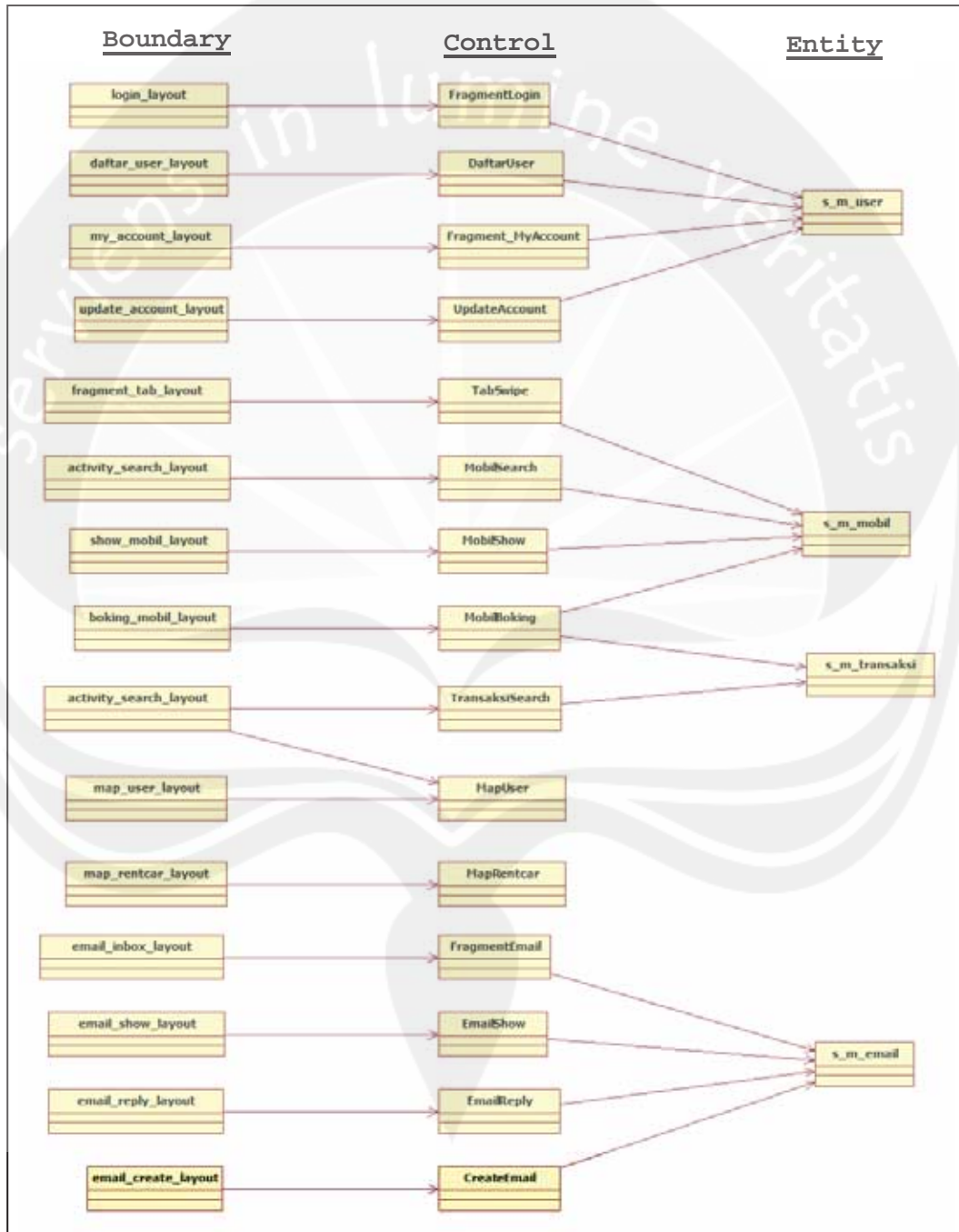
Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Firmansyah Abdul Rafi / 6695, *Deskripsi Perancangan Perangkat Lunak (DPPL) EMERSY*, Universitas Atma Jaya Yogyakarta.
2. Firmansyah Abdul Rafi / 6695, *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) EMERSY*, Universitas Atma Jaya Yogyakarta.

2 Perancangan Sistem

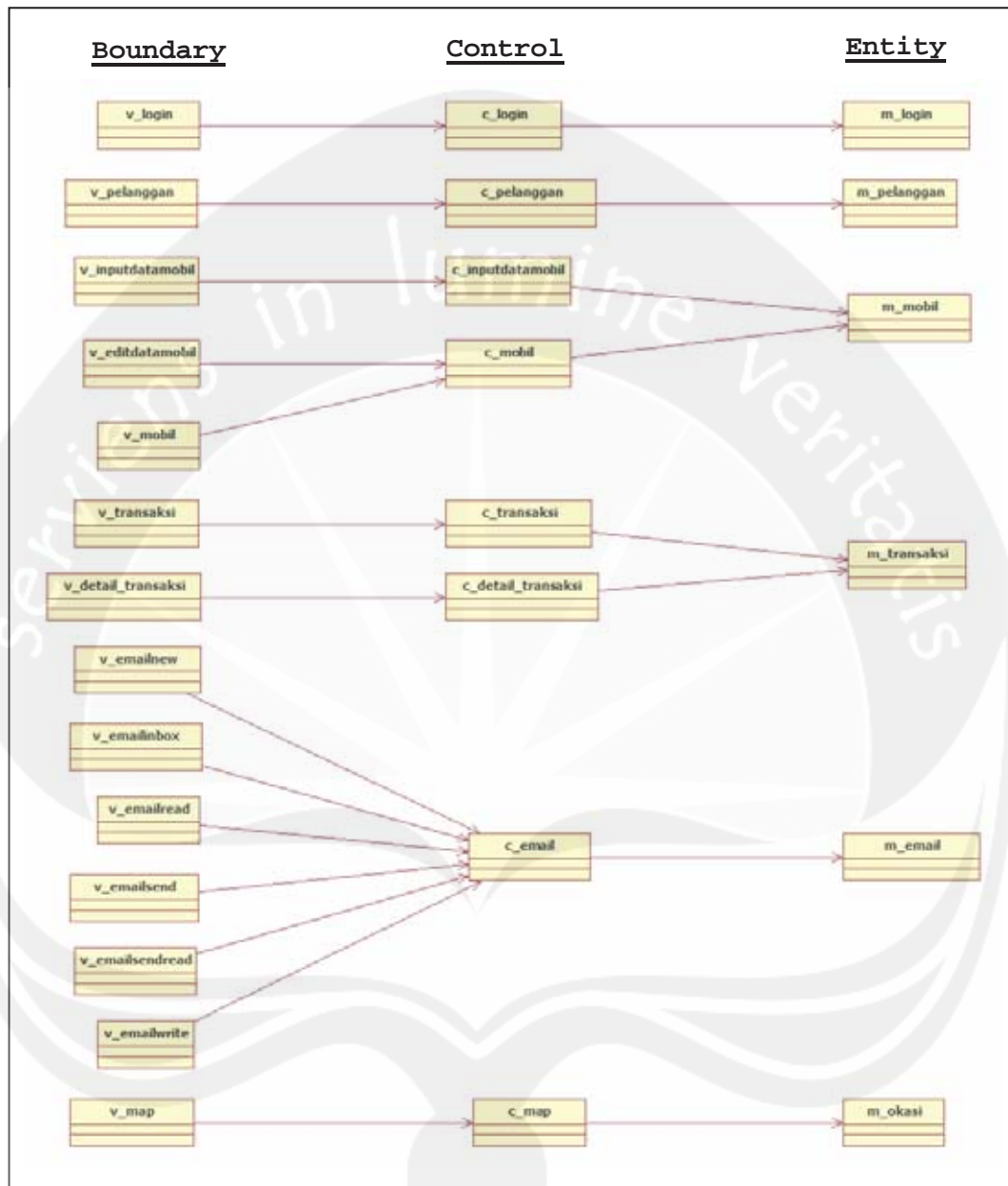
2.1 Perancangan Asitektur

2.1.1 Mobile Apps



Gambar 2.1 Perancangan Arsitektur RENTCAR Mobile Apps

2.1.2 Web Apps

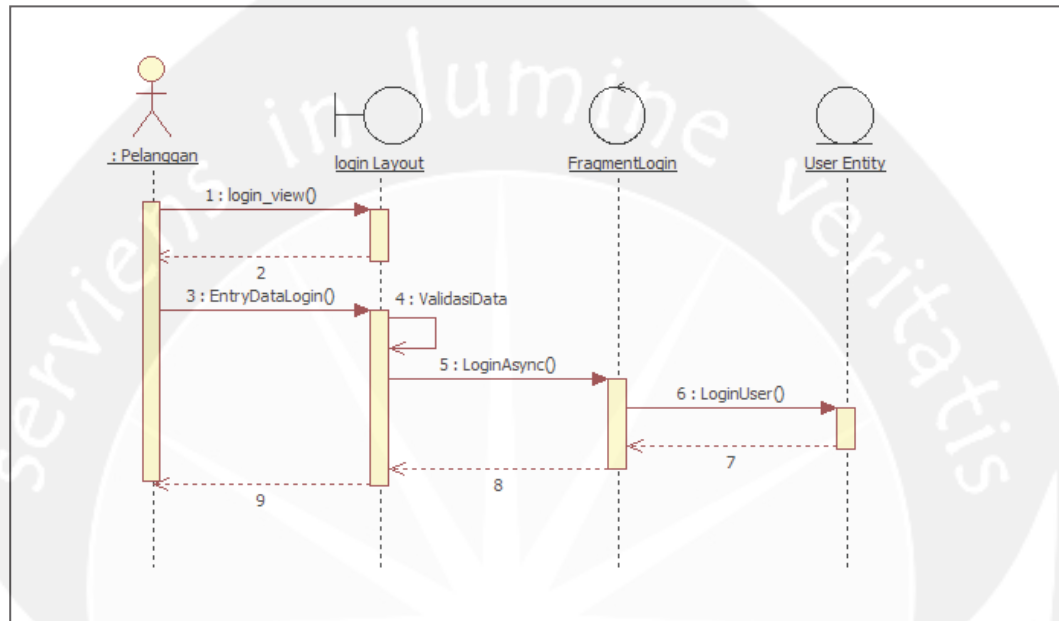


Gambar 2.2 Perancangan Arsitektur RENTCAR Web Apps

2.2 Perancangan Rinci

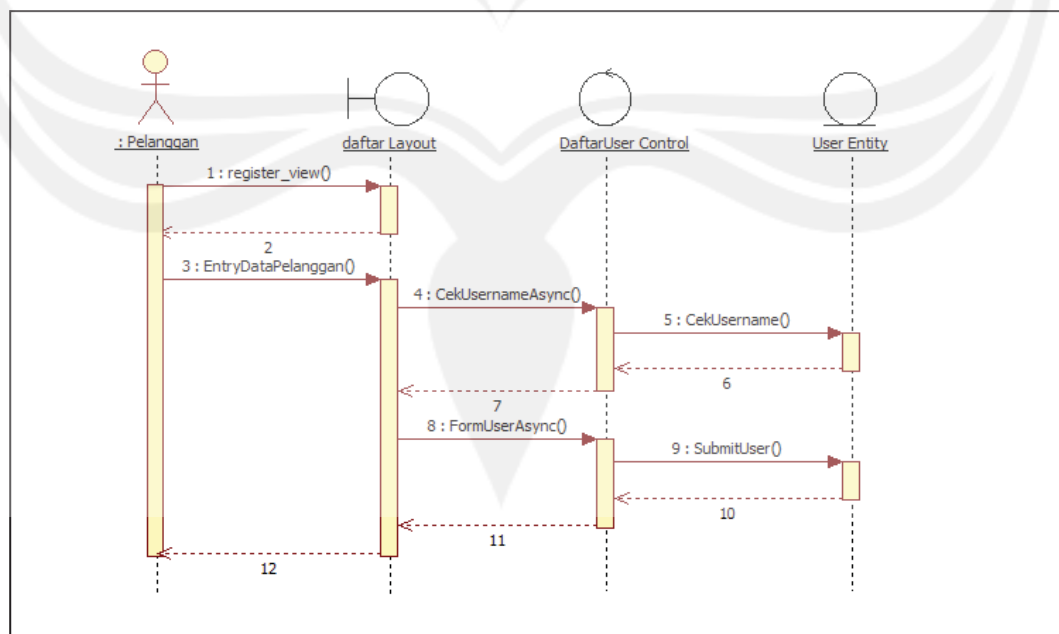
2.2.1 Sequence Diagram Rentcar Apps

2.2.1.1 Fungsi Login Mobile



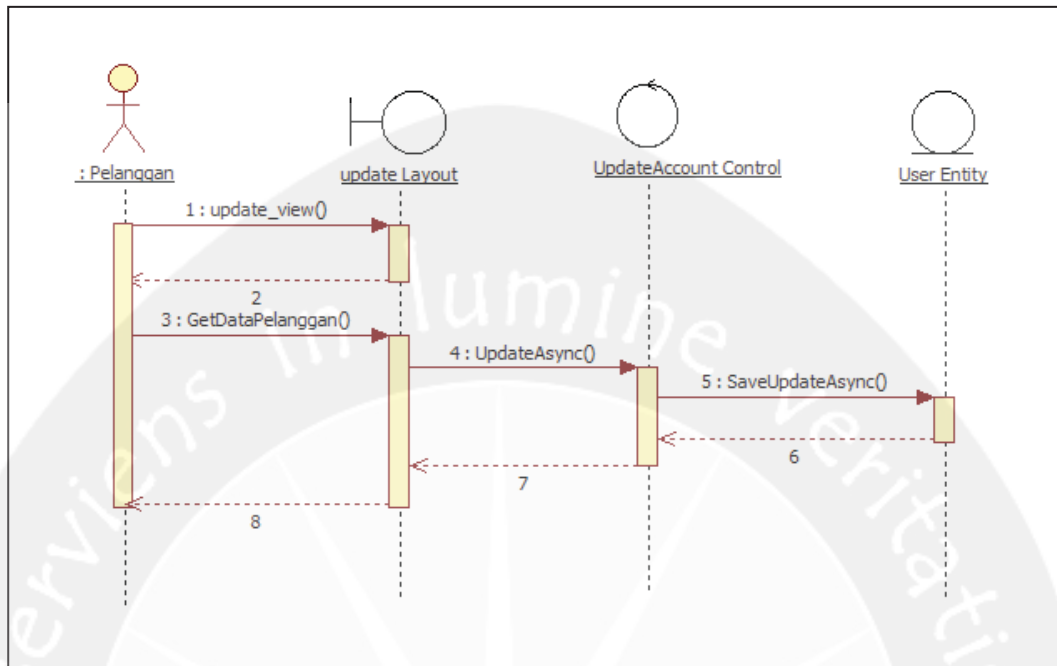
Gambar 2.3 Fungsi Login Mobile

2.2.1.2 Fungsi Registrasi Mobile



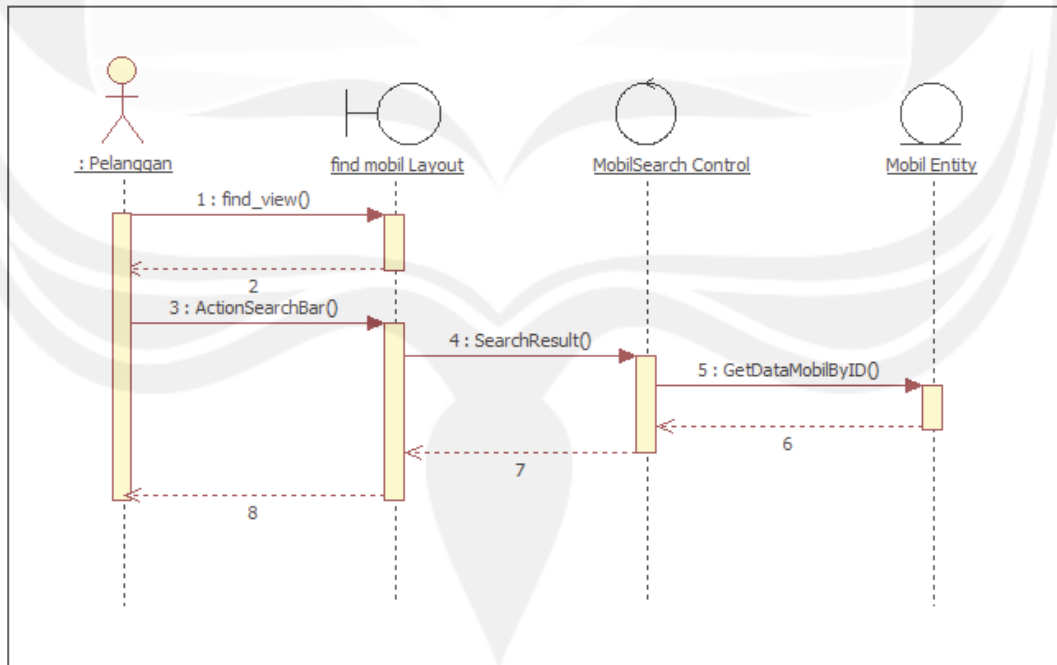
Gambar 2.4 Fungsi Registrasi Mobile

2.2.1.3 Fungsi Kelola Data Pelanggan Mobile



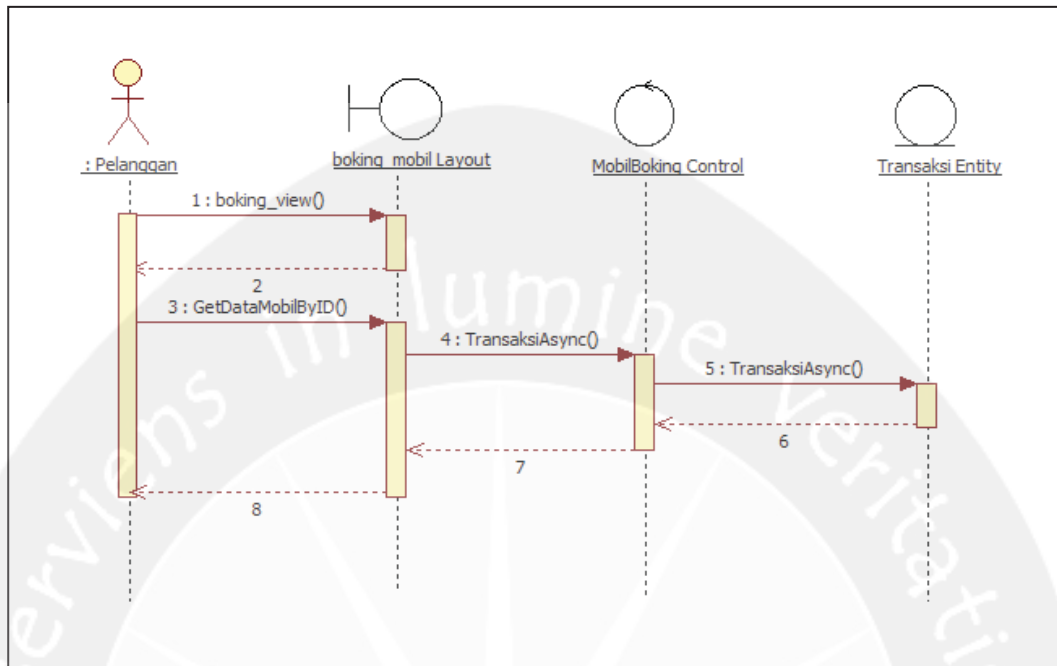
Gambar 2.5 Fungsi Kelola Data Pelanggan Mobile

2.2.1.4 Fungsi Find Data Mobil Mobile



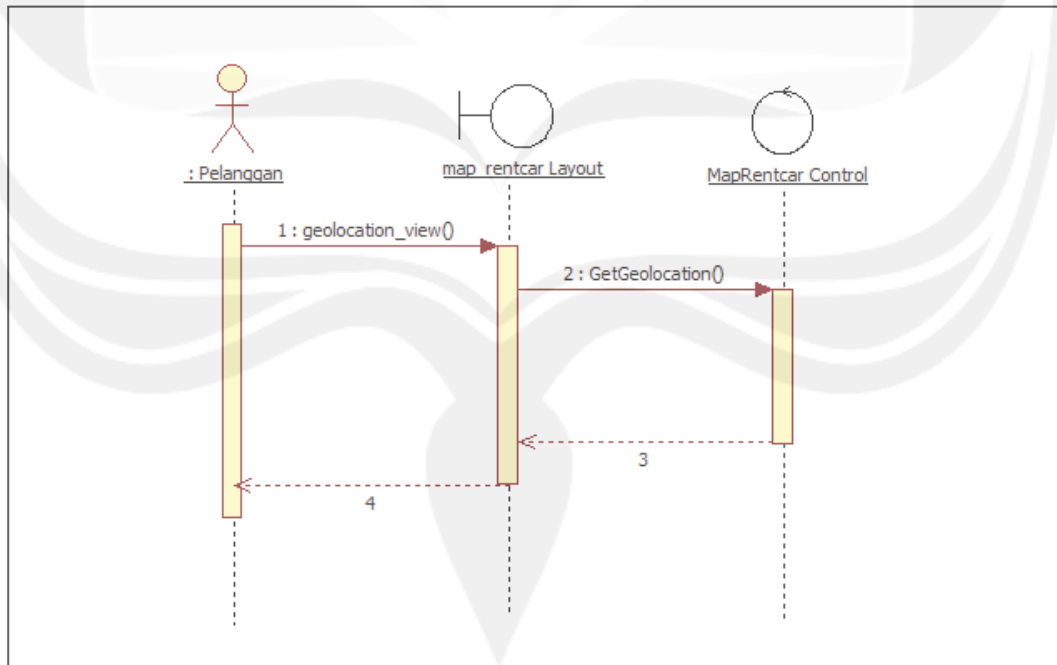
Gambar 2.6 Fungsi Find Data Mobil Mobile

2.2.1.5 Fungsi Booking Mobil Mobile



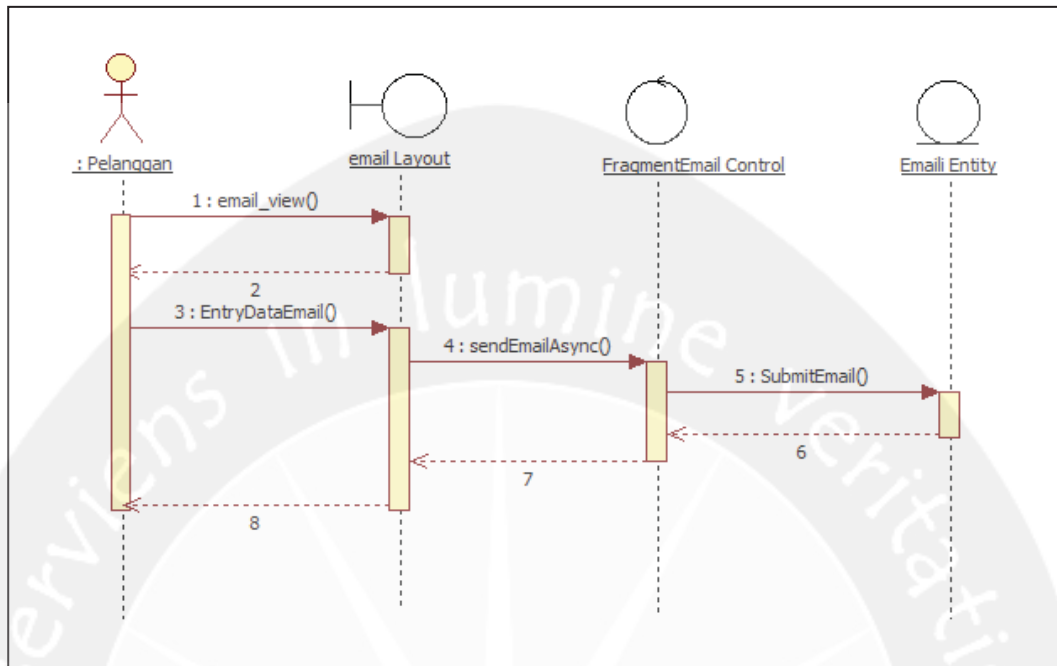
Gambar 2.7 Fungsi Booking Mobil Mobile

2.2.1.6 Fungsi Geolocation Mobile



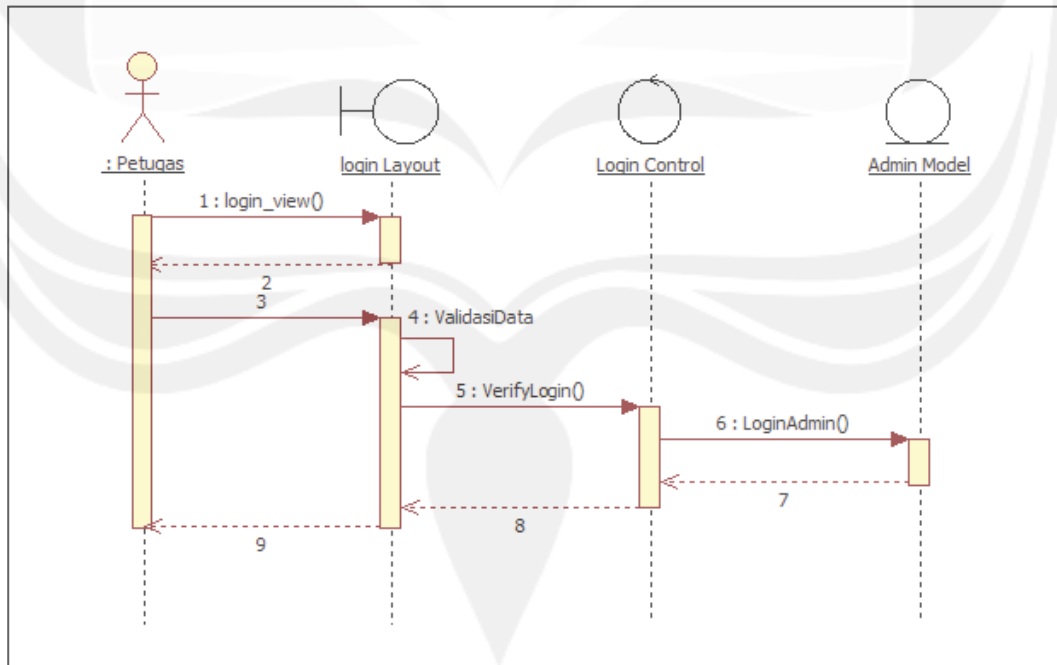
Gambar 2.8 Fungsi Geolocation Mobile

2.2.1.7 Fungsi Email Mobile



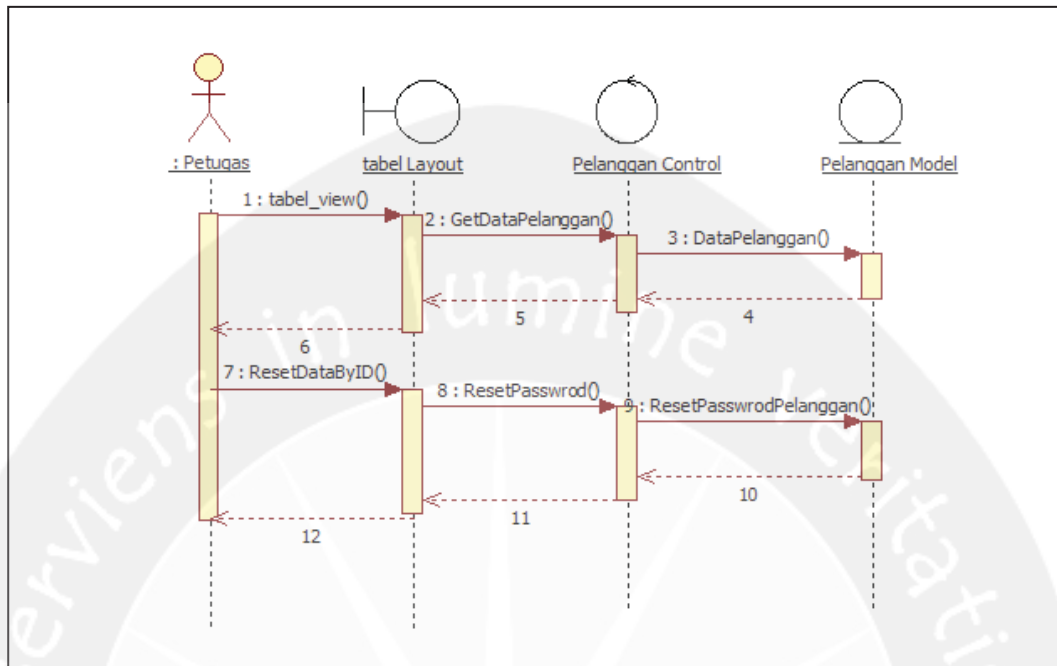
Gambar 2.9 Fungsi Email Mobile

2.2.1.8 Fungsi Login Web



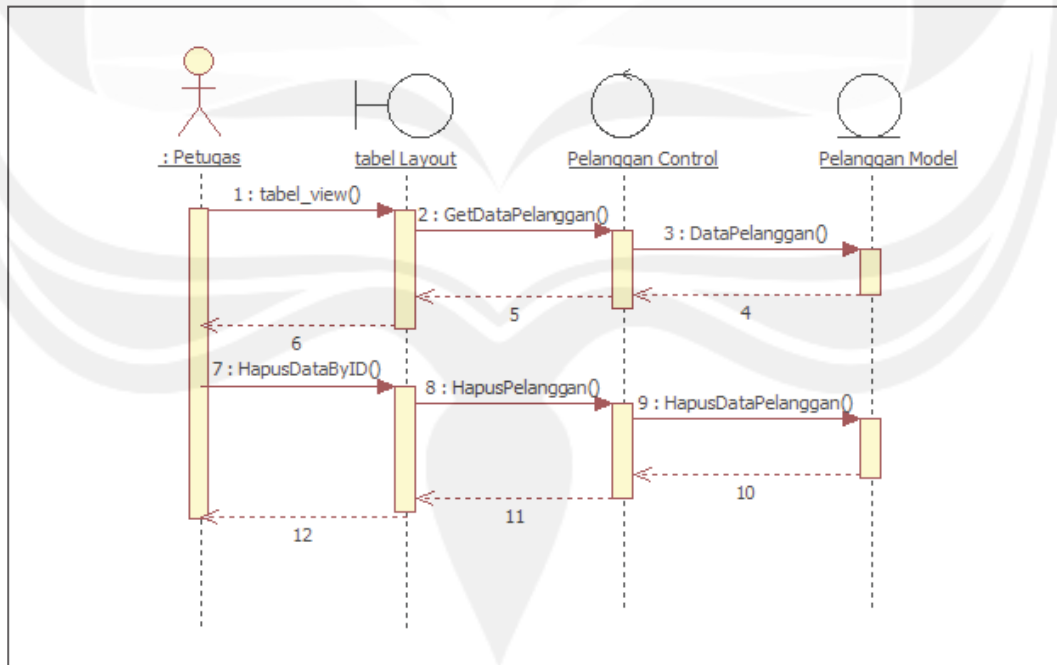
Gambar 2.10 Fungsi Login Web

2.2.1.9 Fungsi Reset Password Pelanggan Web



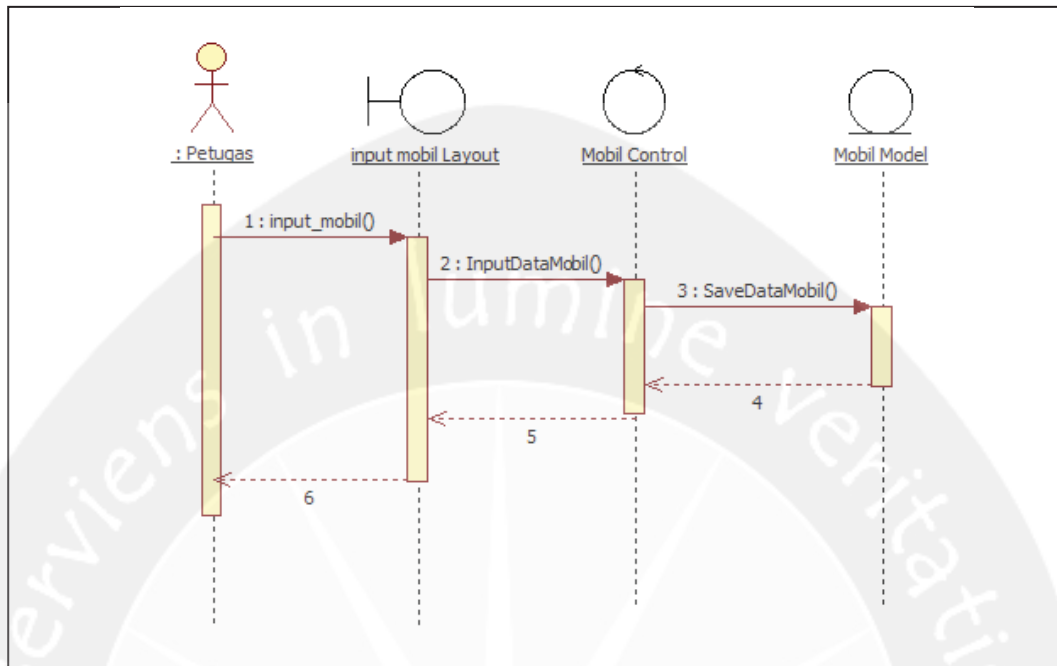
Gambar 2.11 Fungsi Reset Password Pelanggan Web

2.2.1.10 Fungsi Delete Data Pelanggan Web



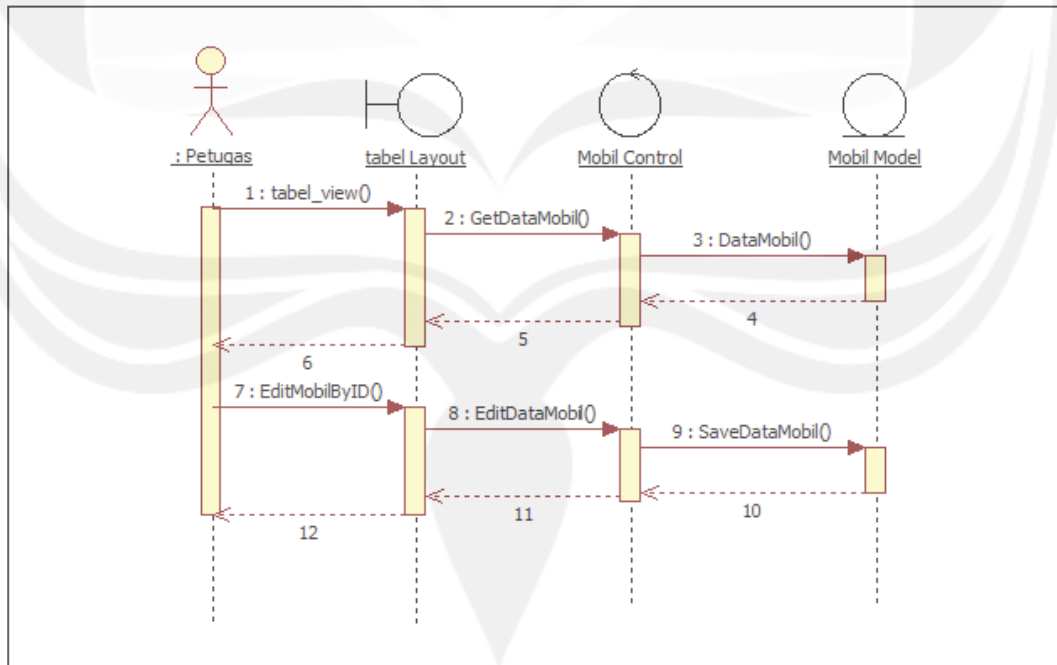
Gambar 2.12 Fungsi Delete Data Pelanggan Web

2.2.1.11 Fungsi Insert Data Mobil Web



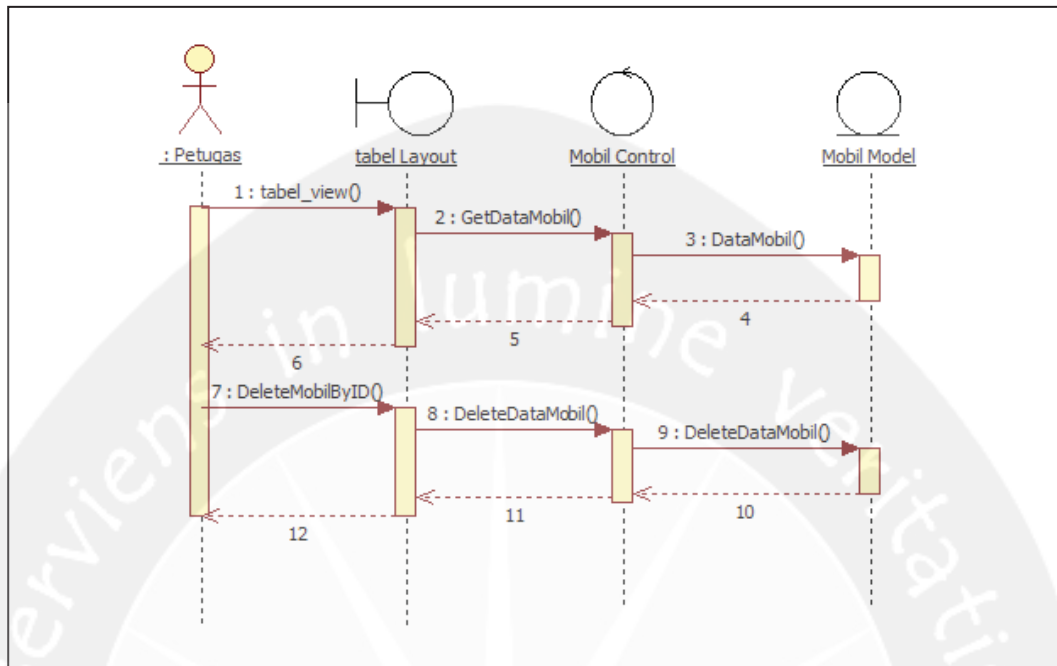
Gambar 2.13 Fungsi Insert Data Mobil Web

2.2.1.12 Fungsi Edit Data Mobil Web



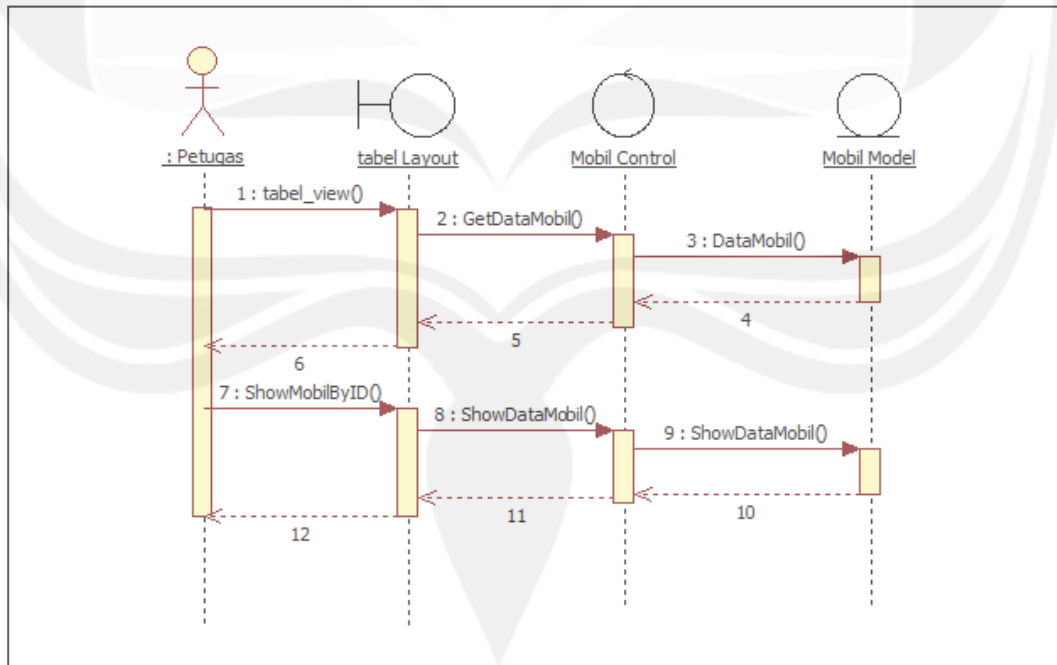
Gambar 2.14 Fungsi Edit Data Mobil Web

2.2.1.13 Fungsi Delete Data Mobil Web



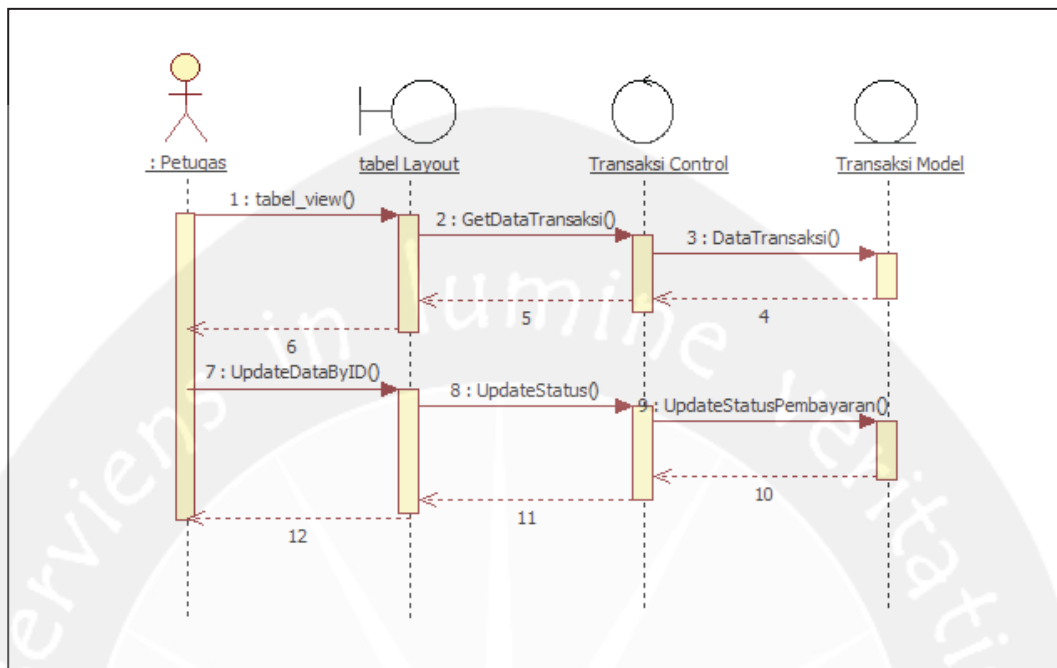
Gambar 2.15 Fungsi Delete Data Mobil Web

2.2.1.14 Fungsi Show Data Mobil Web



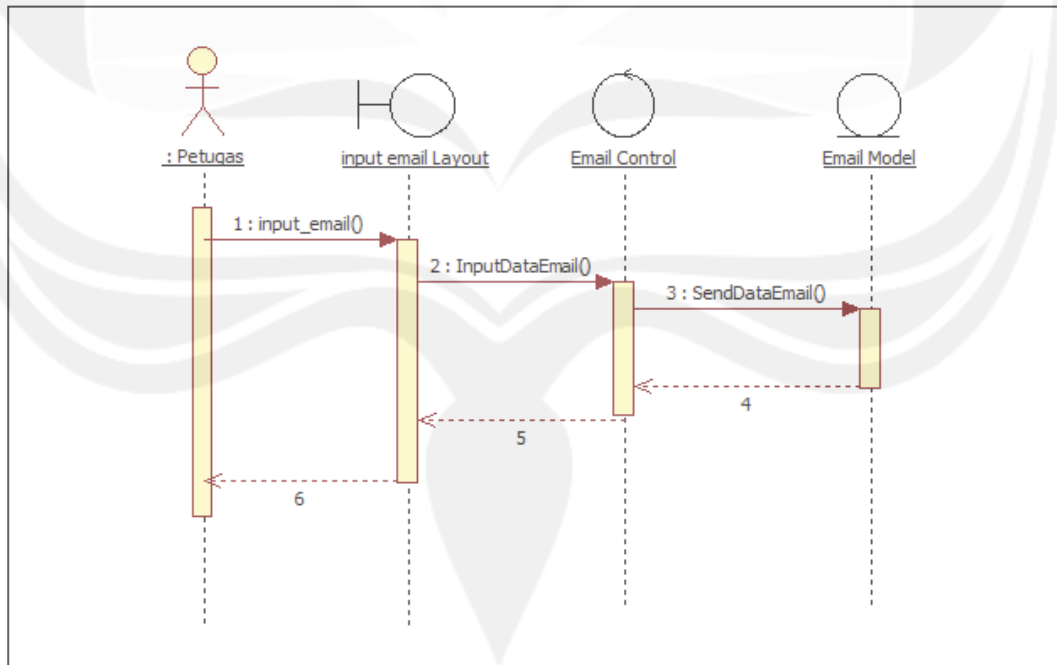
Gambar 2.16 Fungsi Show Data Mobil Web

2.2.1.15 Fungsi Update Status Pembayaran Web



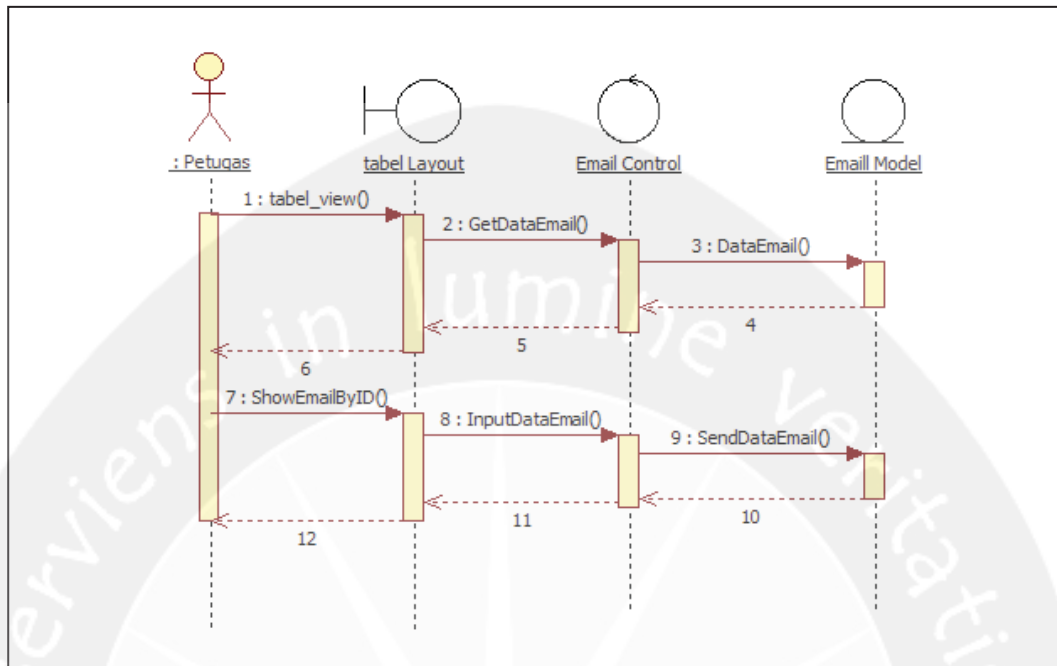
Gambar 2.17 Fungsi Update Status Pembayaran Web

2.2.1.16 Fungsi Create Email Web



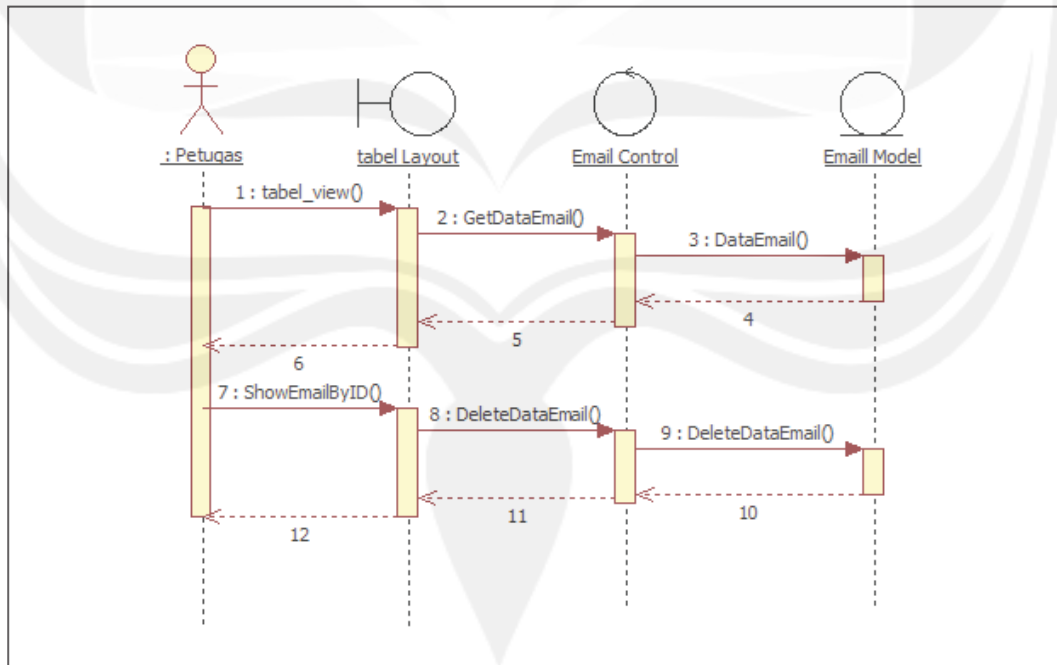
Gambar 2.18 Fungsi Create Email Web

2.2.1.17 Fungsi Reply Email Web



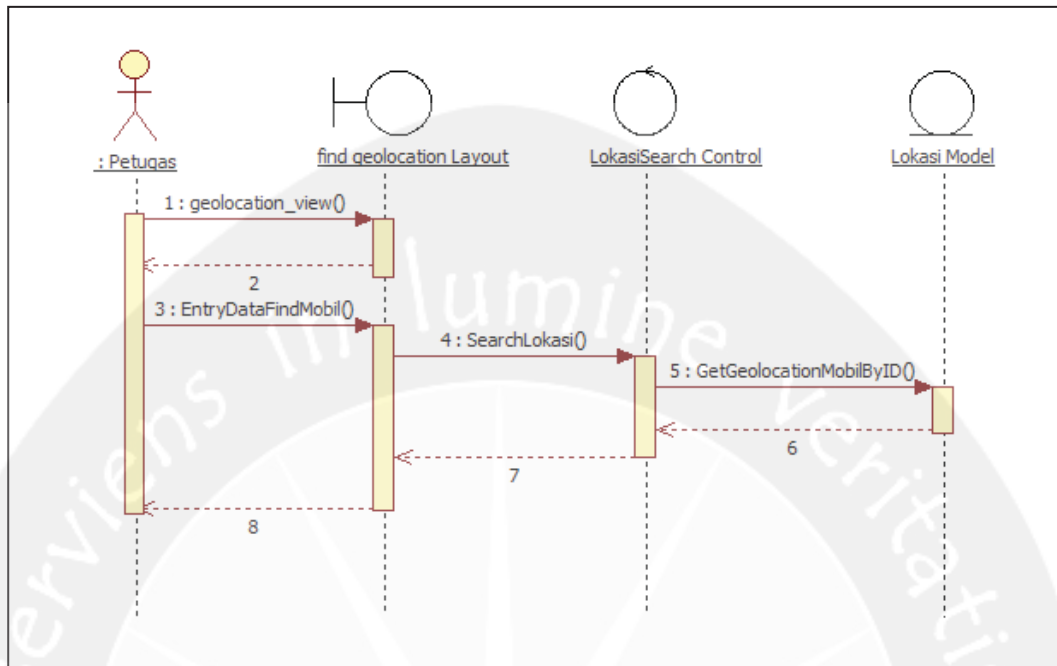
Gambar 2.19 Fungsi Reply Email Web

2.2.1.18 Fungsi Hapus Data Email Web



Gambar 2.20 Fungsi Hapus Data Email Web

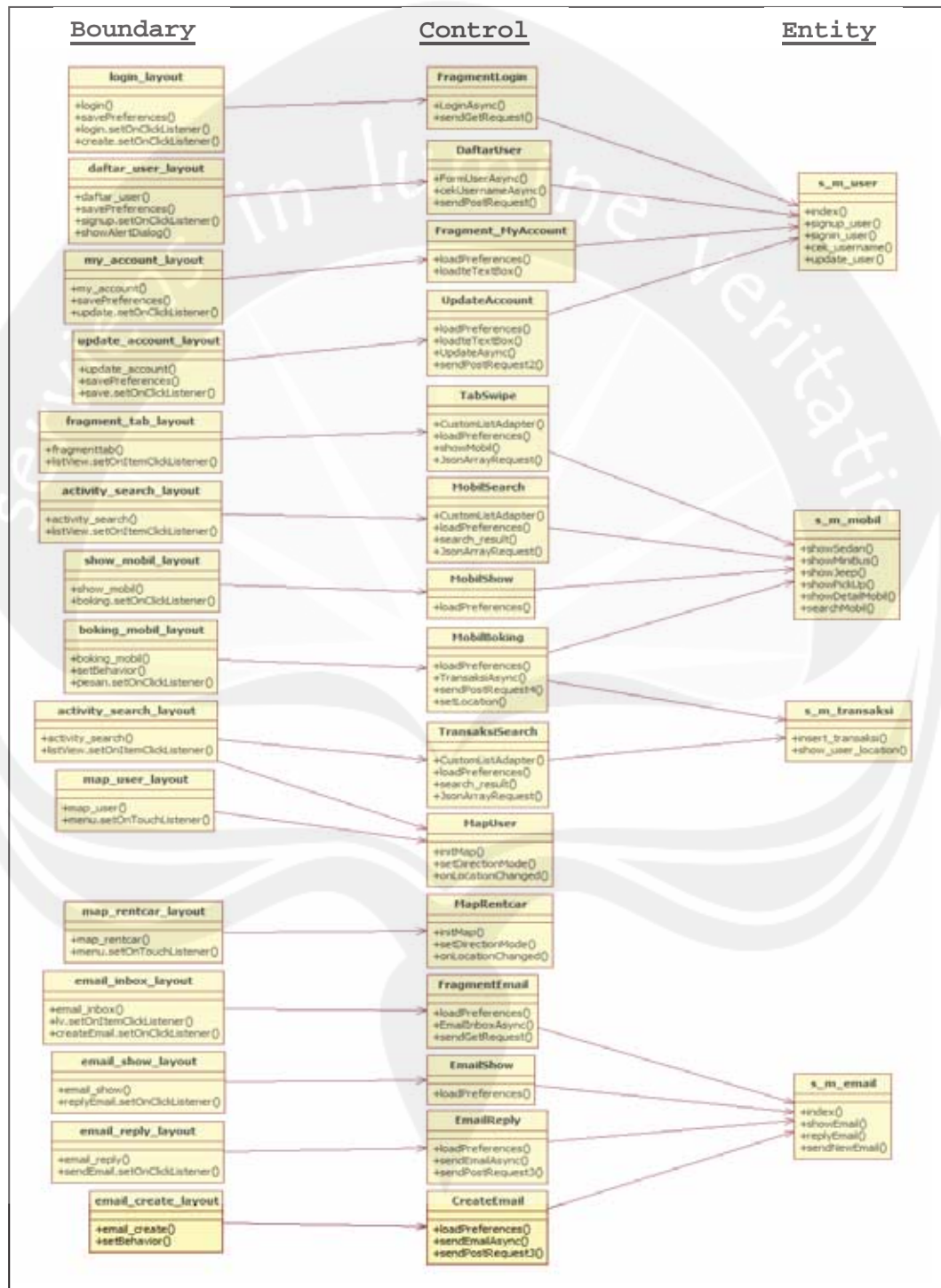
2.2.1.19 Fungsi Cari Lokasi Mobil Web



Gambar 2.21 Fungsi Cari Lokasi Mobil Web

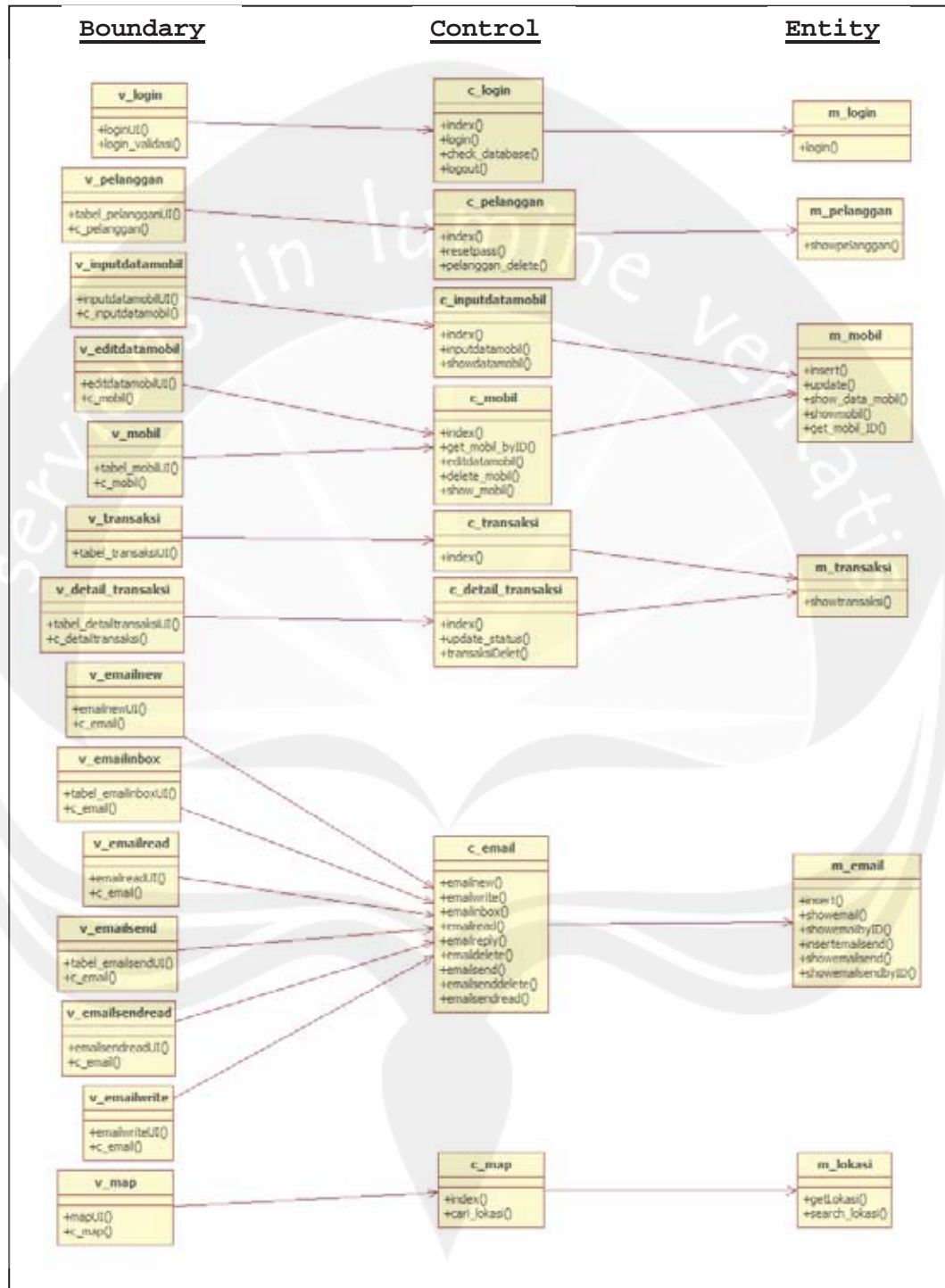
2.2.2 Class Diagram

2.2.2.1 Mobile Apps



Gambar 2.22 Class Diagram Mobile Apps

2.2.2.2 Web Apss



Gambar 2.23 Class Diagram Web Apss

2.2.3 Class Diagram Specific Descriptions

Mobile Apps

2.2.3.1 Specific Design Class login_layout

login_layout	<<boundary>>
<p>+login() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+savePreferences() : Prosedur ini digunakan untuk menyimpan data ke <i>memory</i> penyimpanan <i>device</i>.</p> <p>+login.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau event yang lain.</p> <p>+create.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau event yang lain.</p>	

2.2.3.2 Specific Design Class daftar_user_layout

daftar_user_layout	<<boundary>>
<p>+daftar_user() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+savePreferences() : Prosedur ini digunakan untuk menyimpan data ke <i>memory</i> penyimpanan <i>device</i>.</p> <p>+sign_up.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau event yang lain.</p> <p>+showAlertDialog() : Prosedur ini digunakan untuk membuat <i>AlertDialog</i> yang muncul saat tombol <i>login</i> ditekan</p>	

2.2.3.3 Specific Design Class my_account_layout

my_account_layout	<<boundary>>
<pre>+my_account() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +savePreferences() : Prosedur ini digunakan untuk menyimpan data ke memory penyimpanan device. +update.setOnClickListener() : Prosedur ini digunakan untuk mengeset widget agar widget mempunyai event OnClickListener atau event yang lain.</pre>	

2.2.3.4 Specific Design Class update_account_layout

update_account_layout	<<boundary>>
<pre>+update_account() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +savePreferences() : Prosedur ini digunakan untuk menyimpan data ke memory penyimpanan device. +save.setOnClickListener() : Prosedur ini digunakan untuk mengeset widget agar widget mempunyai event OnClickListener atau event yang lain.</pre>	

2.2.3.5 Specific Design Class fragment_tab_layout

fragment_tab_layout	<<boundary>>
<pre>+frgmenttab() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +listview.setOnClickListener() : Prosedur ini digunakan untuk mengeset widget agar widget mempunyai event OnClickListener atau event yang lain.</pre>	

2.2.3.6 Specific Design Class `activity_search_layout`

<code>activity_search_layout</code>	<code><<boundary>></code>
<code>+activity_search() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</code>	
<code>+listview.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <code>OnClickListener</code> atau <i>event</i> yang lain.</code>	

2.2.3.7 Specific Design Class `show_mobil_layout`

<code>show_mobil_layout</code>	<code><<boundary>></code>
<code>+show_mobil() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</code>	
<code>+boking.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <code>OnClickListener</code> atau <i>event</i> yang lain.</code>	

2.2.3.8 Specific Design Class `boking_mobil_layout`

<code>boking_mobil_layout</code>	<code><<boundary>></code>
<code>+boking_mobil() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</code>	
<code>+setBehavior() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <code>OnClickListener</code> atau <i>event</i> yang lain.</code>	
<code>+pesan.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <code>OnClickListener</code> atau <i>event</i> yang lain.</code>	

2.2.3.9 Spesifikasi Design Class

activity_search_layout

activity_search_layout	<<boundary>>
<p>+activity_search() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+listview.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau <i>event</i> yang lain.</p>	

2.2.3.10 Spesifikasi Design Class map_user_layout

map_user_layout	<<boundary>>
<p>+map_user() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+menu.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau <i>event</i> yang lain.</p>	

2.2.3.11 Spesifikasi Design Class map_rentcar_layout

map_rentcar_layout	<<boundary>>
<p>+map_rentcar() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+menu.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event <i>OnClickListener</i> atau <i>event</i> yang lain.</p>	

2.2.3.12 Spesifikasi Design Class email_inbox_layout

email_inbox_layout	<<boundary>>
<p>+email_inbox() : Default konstruktor, digunakan untuk</p>	

inisialisasi semua *attribute* dari kelas ini.
 +lv.setOnClickListener() : Prosedur ini digunakan untuk mengeset *widget* agar *widget* mempunyai event OnClickListener atau event yang lain.
 +createEmail.setOnClickListener() : Prosedur ini digunakan untuk mengeset *widget* agar *widget* mempunyai event OnClickListener atau event yang lain.

2.2.3.13 Specific Design Class email_show_layout

email_show_layout	<<boundary>>
<p>+email_show() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +replyEmail.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event OnClickListener atau event yang lain.</p>	

2.2.3.14 Specific Design Class email_reply_layout

email_reply_layout	<<boundary>>
<p>+email_reply() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +sendEmail.setOnClickListener() : Prosedur ini digunakan untuk mengeset <i>widget</i> agar <i>widget</i> mempunyai event OnClickListener atau event yang lain.</p>	

2.2.3.15 Specific Design Class email_create_layout

email_create_layout	<<boundary>>
<p>+email_create() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p>	

+setBehavior.setOnClickListener() : Prosedur ini digunakan untuk mengeset *widget* agar *widget* mempunyai event *OnClickListener* atau event yang lain.

2.2.3.16 Specific Design Class FragmentLogin

FragmentLogin	<<control>>
<p>+LoginAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi <i>login</i> secara <i>doInBackground</i>.</p> <p>+sendGetRequest() : Operasi ini digunakan untuk melakukan pengambilan data dari API <i>web service</i> menggunakan JSON.</p>	

2.2.3.17 Specific Design Class DaftarUser

DaftarUser	<<control>>
<p>+FormUserAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi <i>daftar</i> secara <i>doInBackground</i>.</p> <p>+CekUsernameAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi <i>cek username</i> secara <i>doInBackground</i>.</p> <p>+sendPostRequest() : Operasi ini digunakan untuk melakukan pengiriman data ke API <i>web service</i> dengan mengirimkan nilai-nilai yang sudah diset.</p>	

2.2.3.18 Specific Design Class FragmentMyAccount

FragmentMyAccount	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p>	

+loadkeTextBox() : Prosedur ini digunakan untuk meletakkan data yang sudah di load kedalam *Widget EditText*.

2.2.3.19 Specific Design Class UpdateAccount

UpdateAccount	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+loadkeTextBox() : Prosedur ini digunakan untuk meletakkan data yang sudah di load kedalam <i>Widget EditText</i>.</p> <p>+UpdateAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi <i>update</i> secara <i>doInBackground</i>.</p> <p>+sendPostRequest2() : Operasi ini digunakan untuk melakukan pengiriman data ke API <i>web service</i> dengan mengirimkan nilai-nilai yang sudah diset.</p>	

2.2.3.20 Specific Design Class TabSwipe

TabSwipe	<<control>>
<p>+CustomListAdapter() : Prosedur ini digunakan untuk passing data list yang akan ditampilkan.</p> <p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+show_mobil() : Prosedur ini digunakan untuk menampilkan data dengan <i>JSONArrayRequest</i>.</p> <p>+JSONArrayRequest() : Operasi ini digunakan untuk meminta JSON secara <i>array</i> melalui <i>variable</i> yang sudah diset <i>static</i> sebelumnya.</p>	

2.2.3.21 Specific Design Class MobilSearch

MobilSearch	<<control>>
<p>+CustomListAdapter() : Prosedur ini digunakan untuk passing data list yang akan ditampilkan.</p> <p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+search_result() : Prosedur ini digunakan untuk mengambil data berdasarkan inputan.</p> <p>+JSONArrayRequest() : Operasi ini digunakan untuk meminta JSON secara <i>array</i> melalui <i>variable</i> yang sudah diset <i>static</i> sebelumnya.</p>	

2.2.3.22 Specific Design Class MobilShow

MobilShow	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p>	

2.2.3.23 Specific Design Class MobilBoking

MobilBoking	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+TransaksiAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi transaksi secara <i>doInBackground</i>.</p> <p>+sendPostRequest4() : Operasi ini digunakan untuk melakukan pengiriman data ke API <i>web service</i> dengan mengirimkan nilai-nilai yang sudah diset.</p>	

+setLocation() : Operasi ini digunakan untuk mengambil *latitude* dan *longitude* pelanggan sekarang berada dan menyimpannya ke dalam *database*.

2.2.3.24 Specific Design Class TransaksiSearch

TransaksiSearch	<<control>>
<p>+CustomListAdapter() : Prosedur ini digunakan untuk passing data list yang akan ditampilkan.</p> <p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+search_result() : Prosedur ini digunakan untuk mengambil data berdasarkan inputan.</p> <p>+JSONArrayRequest() : Operasi ini digunakan untuk meminta JSON secara <i>array</i> melalui <i>variable</i> yang sudah diset <i>static</i> sebelumnya.</p>	

2.2.3.25 Specific Design Class MapUser

MapUser	<<control>>
<p>+initMap() : Operasi ini digunakan untuk membuat garis berwarna merah menggunakan teknik <i>polyline</i> dari <i>latitude</i> dan <i>longitude</i> tujuan dengan <i>latitude</i> dan <i>longitude</i> sekarang berada.</p> <p>+setDirectionMode() : Operasi ini digunakan untuk mengeset mode <i>direction</i> (<i>driving, bicycling, walking</i>).</p> <p>+onLocationChanged() : Operasi ini digunakan untuk mengeset <i>latitude</i> dan <i>longitude</i> tujuan dengan <i>latitude</i> dan <i>longitude</i> sekarang berada.</p>	

2.2.3.26 Specific Design Class MapRentcar

MapRentcar	<<control>>
<p>+initMap() : Operasi ini digunakan untuk membuat garis berwarna merah menggunakan teknik <i>polyline</i> dari <i>latitude</i> dan <i>longitude</i> tujuan dengan <i>latitude</i> dan <i>longitude</i> sekarang berada.</p> <p>+setDirectionMode() : Operasi ini digunakan untuk mengeset mode <i>direction</i> (<i>driving, bicycling, walking</i>).</p> <p>+onLocationChanged() : Operasi ini digunakan untuk mengeset <i>latitude</i> dan <i>longitude</i> tujuan dengan <i>latitude</i> dan <i>longitude</i> sekarang berada.</p>	

2.2.3.27 Specific Design Class FragmentEmail

FragmentEmail	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+EmailInboxAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi email inbox secara <i>doInBackground</i>.</p> <p>+sendGetRequest() : Operasi ini digunakan untuk melakukan pengambilan data dari API <i>web service</i> menggunakan JSON.</p>	

2.2.3.28 Specific Design Class EmailShow

EmailShow	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p>	

2.2.3.29 Specific Design Class EmailReply

EmailReply	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+sendEmailAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi email <i>reply</i> secara <i>doInBackground</i>.</p> <p>+sendPostRequest3() : Operasi ini digunakan untuk melakukan pengiriman data ke API <i>web service</i> dengan mengirimkan nilai-nilai yang sudah diset.</p>	

2.2.3.30 Specific Design Class CreateEmail

CreateEmail	<<control>>
<p>+loadPreferences() : Prosedur ini digunakan untuk mengambil data dari <i>memory</i> penyimpanan <i>device</i>.</p> <p>+sendEmailAsync() : Operasi ini digunakan untuk melakukan pemanggilan fungsi email <i>create</i> secara <i>doInBackground</i>.</p> <p>+sendPostRequest3() : Operasi ini digunakan untuk melakukan pengiriman data ke API <i>web service</i> dengan mengirimkan nilai-nilai yang sudah diset.</p>	

2.2.3.31 Specific Design Class s_m_user

s_m_user	<<entity>>
<p>+index() : Digunakan untuk mengambil data semua pelanggan dari <i>database</i>.</p> <p>+signup_user() Digunakan untuk memasukkan data</p>	

pelanggan yang mendaftar sebagai member melalui aplikasi *mobile* dan menyimpannya ke dalam *database*.

+signin_user() : Digunakan untuk melakukan pengecekan berdasarkan *username* dan *password* untuk dapat masuk ke sistem.

+cek_username() : Digunakan untuk mengecek *username* sudah ada atau belum ketika pelanggan mendaftarkan diri sebagai member.

+update_user() : Digunakan untuk melakukan *update* data pelanggan yang telah terdaftar sebagai member.

2.2.3.32 Specific Design Class s_m_mobil

s_m_mobil	<<entity>>
<p>+showSedan() : Digunakan untuk mengambil data mobil berdasarkan jenis mobil sedan.</p> <p>+showMiniBus() : Digunakan untuk mengambil data mobil berdasarkan jenis mobil mini bus.</p> <p>+showJeep() : Digunakan untuk mengambil data mobil berdasarkan jenis mobil jeep.</p> <p>+showPickUp() : Digunakan untuk mengambil data mobil berdasarkan jenis mobil pick up.</p> <p>+showDetailMobil() : Digunakan untuk mengambil data berdasarkan id_mobil.</p> <p>+searchMobil() : Digunakan untuk mengambil data mobil dengan pemfilteran data berdasarkan <i>keyword</i>.</p>	

2.2.3.33 Specific Design Class s_m_transaksi

s_m_transaksi	<<entity>>
<p>+insert_transaksi() : Digunakan untuk operasi <i>insert</i></p>	

transaksi pemesanan mobil dan menyimpannya ke dalam *database*.

+show_user_location() : Digunakan untuk mengambil lokasi user dengan pemfilteran data berdasarkan *keyword*

2.2.3.34 Specific Design Class s_m_email

s_m_email	<<entity>>
<p>+index() : Digunakan untuk mengambil data semua pelanggan dari <i>database</i>.</p> <p>+shoEmail() : Digunakan untuk mengambil data mobil berdasarkan <i>id_user</i>.</p> <p>+replyEmail() : Digunakan untuk operasi untuk membalas email dari petugas.</p> <p>+sendNewEmail() : Digunakan untuk operasi untuk mengirim email baru kepada petugas dan menyimpannya ke dalam <i>database</i>.</p>	

Web Apps

2.2.3.35 Specific Design Class v_login

v_login	<<boundary>>
<p>+loginUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+login_validasi() : Fungsi yang digunakan untuk pemanggilan class <i>c_login</i>.</p>	

2.2.3.36 Specific Design Class v_pelanggan

v_pelanggan	<<boundary>>

+table_pelangganUI() : Default konstruktor, digunakan untuk inisialisasi semua *attribute* dari kelas ini.
 +c_pelanggan() : Fungsi yang digunakan untuk pemanggilan class c_pelanggan.

2.2.3.37 Specific Design Class v_inputdatamobil

v_inputdatamobil	<<boundary>>
<p>+inputdatamobilUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +c_inputdatamobil() : Fungsi yang digunakan untuk pemanggilan class c_inputdatamobil.</p>	

2.2.3.38 Specific Design Class v_editdatamobil

v_editdatamobil	<<boundary>>
<p>+editdatamobilUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +c_editdatamobil() : Fungsi yang digunakan untuk pemanggilan class c_editdatamobil.</p>	

2.2.3.39 Specific Design Class v_mobil

v_mobil	<<boundary>>
<p>+table_mobilUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +c_mobil() : Fungsi yang digunakan untuk pemanggilan class c_mobil.</p>	

2.2.3.40 Specific Design Class v_transaksi

v_transaksi	<<boundary>>
--------------------	---------------------------------

+table_transaksiUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.

2.2.3.41 Specific Design Class v_detail_transaksi

v_detail_transaksi	<<boundary>>
<p>+table_detail_transaksiUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+c_transaksi() : Fungsi yang digunakan untuk pemanggilan class c_transaksi.</p>	

2.2.3.42 Specific Design Class v_emailnew

v_emailnew	<<boundary>>
<p>+emailnewUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+c_email() : Fungsi yang digunakan untuk pemanggilan class c_email.</p>	

2.2.3.43 Specific Design Class v_emailinbox

v_emailinbox	<<boundary>>
<p>+table_emailinboxUI() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+c_email() : Fungsi yang digunakan untuk pemanggilan class c_email.</p>	

2.2.3.44 Specific Design Class v_emailread

v_emailread	<<boundary>>
--------------------	---------------------------------

<pre>+emailreadUI() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +c_email() : Fungsi yang digunakan untuk pemanggilan class c_email.</pre>

2.2.3.45 Specific Design Class v_emailsend

v_emailsend	<<boundary>>
<pre>+table_emailsendUI() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +c_email() : Fungsi yang digunakan untuk pemanggilan class c_email.</pre>	

2.2.3.46 Specific Design Class v_emailsendread

v_emailsendread	<<boundary>>
<pre>+ emailsendreadUI() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +c_email() : Fungsi yang digunakan untuk pemanggilan class c_email.</pre>	

2.2.3.47 Specific Design Class v_map

v_map	<<boundary>>
<pre>+mapUI() : Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +c_map() : Fungsi yang digunakan untuk pemanggilan class c_map.</pre>	

2.2.3.48 Specific Design Class c_login

c_login	<<control>>
<p>+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+logout() : Fungsi yang digunakan untuk memulai session pada web.</p> <p>+logout() : Fungsi yang digunakan untuk menghapus session pada web.</p> <p>+check_database() : Fungsi ini digunakan untuk mengecek <i>username</i> dan <i>password</i> yang ada dalam <i>database</i>.</p>	

2.2.3.49 Specific Design Class c_pelanggan

c_pelanggan	<<control>>
<p>+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+resetpassword() : Fungsi ini digunakan untuk mereset password pelanggan.</p> <p>+pelanggan_delete() : Fungsi ini digunakan untuk menghapus data pelanggan yang ada dalam <i>database</i>.</p>	

2.2.3.50 Specific Design Class c_inputdatamobil

c_inputdatamobil	<<control>>
<p>+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+inputdatamobil() : Fungsi ini digunakan untuk insert data mobil baru dan akan disimpan ke dalam <i>database</i>.</p> <p>+showdatamobil() : Fungsi ini digunakan untuk</p>	

menampilkan data mobil yang tersimpan pada *database*.

2.2.3.51 Specific Design Class *c_mobil*

c_mobil	<<control>>
+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+get_mobil_byID() : Fungsi ini digunakan untuk menampilkan atau memanggil data mobil berdasarkan <i>id_mobil</i> .	
+editdatamobil() : Fungsi ini digunakan untuk mengubah data mobil yang tersimpan pada <i>database</i> .	
+delete_mobil() : Fungsi ini digunakan untuk menghapus sementara atau menyembunyikan data mobil dengan asumsi <i>is_delete = 0</i> .	
+show_mobil() : Fungsi ini digunakan untuk menampilkan kembali data mobil dengan asumsi <i>is_delete = 1</i> .	

2.2.3.52 Specific Design Class *c_transaksi*

c_transaksi	<<control>>
+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	

2.2.3.53 Specific Design Class *c_detai_transaksi*

c_detai_transaksi	<<control>>
+index() : Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+update_status() : Fungsi ini digunakan untuk mengubah status pembayaran sewa mobil dari belum dibayar menjadi	

terbayar.

+transaksiDelete() : Fungsi ini digunakan untuk menghapus data transaksi yang tersimpan pada *database*.

2.2.3.54 Specific Design Class c_email

c_email	<<control>>
+emailnew() : Fungsi ini digunakan untuk mengirim email ke pelanggan.	
+emailwrite() : Fungsi ini digunakan untuk menginputkan <i>content</i> atau isi email.	
+emailinbox() : Fungsi ini digunakan untuk menampilkan email yang dikirim pelanggan.	
+emailread() : Fungsi ini digunakan untuk menampilkan <i>content</i> atau isi data email berdasarkan <i>id_user</i> .	
+emailreply() : Fungsi ini digunakan untuk membalas email dari pelanggan.	
+emaildelete() : Fungsi ini digunakan untuk menghapus data email dari pelanggan yang tersimpan pada <i>database</i> .	
+emailsend() : Fungsi ini digunakan untuk menampilkan email yang dikirim petugas.	
+emailsenddelete() : Fungsi ini digunakan untuk menghapus data email dari petugas yang tersimpan pada <i>database</i> .	
+emailsendread() : Fungsi ini digunakan untuk menampilkan <i>content</i> atau isi data email berdasarkan <i>id_user</i> .	

2.2.3.55 Specific Design Class c_map

c_map	<<control>>

+index() : Default konstruktor, digunakan untuk inisialisasi semua *attribute* dari kelas ini.

+cari_lokasi() : Fungsi ini digunakan untuk menampilkan lokasi pelanggan dengan pemfilteran data berdasarkan keyword.

2.2.3.56 Specific Design Class m_login

m_login	<<entity>>
+login() : Fungsi ini digunakan untuk mencocokkan <i>username</i> dan <i>password</i> yang dikirim oleh c_login.	

2.2.3.57 Specific Design Class m_pelanggan

m_pelanggan	<<entity>>
+showpelanggan() : Fungsi ini digunakan untuk menampilkan atau mengambil data pelanggan	

2.2.3.58 Specific Design Class m_mobil

m_mobil	<<entity>>
+insert() : Fungsi ini digunakan untuk menambahkan data mobil dan akan disimpan ke dalam <i>database</i> . +update() : Fungsi ini digunakan untuk mengubah data mobil yang tersimpan pada <i>database</i> . +show_data_mobil() : Fungsi ini digunakan untuk menampilkan setelah tambah data selesai. +showmobil() : Fungsi ini digunakan untuk menampilkan atau mengambil data mobil. +get_mobil_ID() : Fungsi ini digunakan untuk menampilkan atau mengambil data mobil berdasarkan	

id_mobil.

2.2.3.59 Specific Design Class m_transaksi

m_transaksi	<<entity>>
+showtransaksi() : Fungsi ini digunakan untuk menampilkan atau mengambil data transaksi.	

2.2.3.60 Specific Design Class m_email

m_email	<<entity>>
+insert() : Fungsi ini digunakan untuk menambahkan data email yang dikirim pelanggan dan akan disimpan ke dalam <i>database</i> .	
+showemail() : Fungsi ini digunakan untuk menampilkan data email dari pelanggan.	
+showemailbyID() : Fungsi ini digunakan untuk menampilkan atau mengambil data email yang dikirim pelanggan berdasarkan id_email.	
+insertemailsend() : Fungsi ini digunakan untuk menambahkan data email yang dikirim petugas dan akan disimpan ke dalam <i>database</i> .	
+showemailsend() : Fungsi ini digunakan untuk menampilkan data email dari petugas.	
+showemailsendbyID() : Fungsi ini digunakan untuk menampilkan atau mengambil data email yang dikirim petugas berdasarkan id_email.	

2.2.3.61 Specific Design Class m_lokasi

m_lokasi	<<entity>>

+getLokasi() : Fungsi ini digunakan untuk menampilkan atau mengambil data lokasi pelanggan.

+search_lokasi() : Fungsi ini digunakan untuk mencocokkan data lokasi yang dikirim oleh c_map.

3 Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas TBL_Admin			
Nama	Tipe	Panjang	Keterangan
Id_admin	Int	-	Id Admin, Primary Key
Username	Varchar	45	Username Admin
Password	Varchar	45	Password Admin
Email_admin	Varchar	45	Email Admin

3.1.2 Deskripsi Entitas TBL_User			
Nama	Tipe	Panjang	Keterangan
Id_user	Int	-	Id User, Primary Key
Nama_lengkap	Varchar	45	Nama Lengkap Pelanggan
Username	Varchar	45	Username Pelanggan
Password	Varchar	45	Password Pelanggan
No_telp	Varchar	45	Nomor Telpon Pelanggan
Alamat	Varchar	45	Alamat Pelanggan
Email	Varchar	45	Email Pelanggan
Rekening	Varchar	45	Rekening Pelanggan

3.1.3 Deskripsi Entitas TBL_Mobil			
Nama	Tipe	Panjang	Keterangan

Id_mobil	Int	-	Id Mobil, Primary Key
No_plat	Varchar	45	Nomor Plat Mobil
Merk_mobil	Varchar	45	Merk Mobil
Jenis_mobil	Varchar	45	Jenis Mobil
Tarif	Int	-	Tarif Mobil / Hari
Deskripsi	Varchar	1024	Deskripsi Mobil
Foto	Varchar	1024	Foto Mobil
Is_deleted	Int	-	

3.1.4 Deskripsi Entitas TBL_Transaksi

Nama	Tipe	Panjang	Keterangan
Id_transaksi	Int	-	Id Transaksi, Primary Key
Id_user	Int	-	Id User, Foreign key dari TBL_User
Id_mobil	Int	-	Id Mobil, Foreign key dari TBL_Mobil
Lama_sewa	Int	-	Lama Sewa Mobil / Hari
Tgl_transaksi	Date		Tanggal Transaksi Peminjaman Mobil
Tgl_pinjam	Date	-	Tanggal Peminjaman Mobil
Tgl_kembali	Date	-	Tanggal Pengembalian Mobil
Supir_mobil	Varchar	45	Asumsi Peminjaman Mobil (Dengan Supir / Tanpa Supir)
Supir_nama	Varchar	100	Nama Supir Jika

			Asumsi Peminjaman "Dengan Supir"
Antar_mobil	Varchar	45	Asumsi Pengantaran Mobil (Diantarkan / Ambil Sendiri)
Biaya_sewa	Int	-	Total Biaya Peminjaman Mobil
Status	Varchar	45	Status transaksi (Belum Dibayar / Terbayar)
Longitude	Double	-	Koordinat Lokasi Penyewa
Latitude	Double	-	Koordinat Lokasi Penyewa

3.1.5 Deskripsi Entitas TBL_Email

Nama	Tipe	Panjang	Keterangan
Id_email	Int	-	Id Email, Primary Key
Id_admin	Int	-	Id Admin, Foreign key dari TBL_Admin
Id_user	Int	-	Id User, Foreign key dari TBL_User
Judul_email	Varchar	50	Judul Email Yang Dikirim Pelanggan
Isi_email	Varchar	1024	Isi dari Email yang Dikirim Pelanggan
Tgl_email	Date	-	Tanggal Pengiriman Email

3.1.6 Deskripsi Entitas TBL_EmailSend

Program Studi Teknik Informatika	DPPL – RENTCAR	48/ 72
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		

Nama	Tipe	Panjang	Keterangan
Id_email	Int	-	Id Email, Primary Key
Id_admin	Int	-	Id Admin, Foregin key dari TBL_Admin
Id_user	Int	-	Id User, Foregin key dari TBL_User
Judul_email	Varchar	50	Judul Email Yang Dikirim Admin
Isi_email	Varchar	1024	Isi dari Email yang Dikirim Admin
Tgl_email	Date	-	Tanggal Pengiriman Email

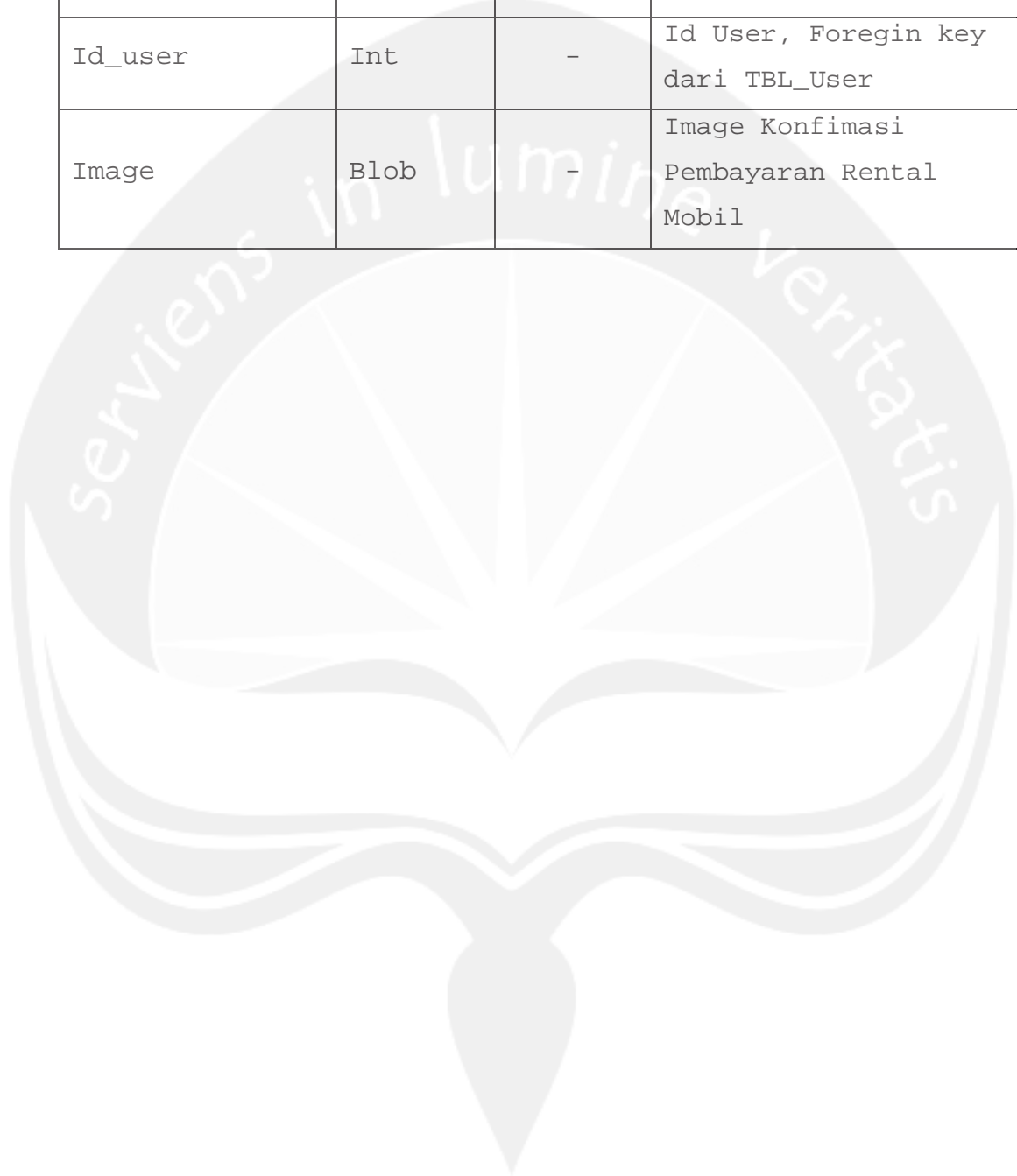
3.1.7 Deskripsi Entitas TBL_Lokasi

Nama	Tipe	Panjang	Keterangan
Id_lokasi	Int	-	Id Lokasi, Primary Key
Id_transaksi	Int	-	Id Transaksi, Foregin key dari TBL_Transaksi
Id_user	Int	-	Id User, Foregin key dari TBL_User
Id_mobil	Int	-	Id Mobil, Foregin key dari TBL_Mobil
Longitude	Double	-	Koordinat Lokasi Penyewa
Latitude	Double	-	Koordinat Lokasi Penyewa

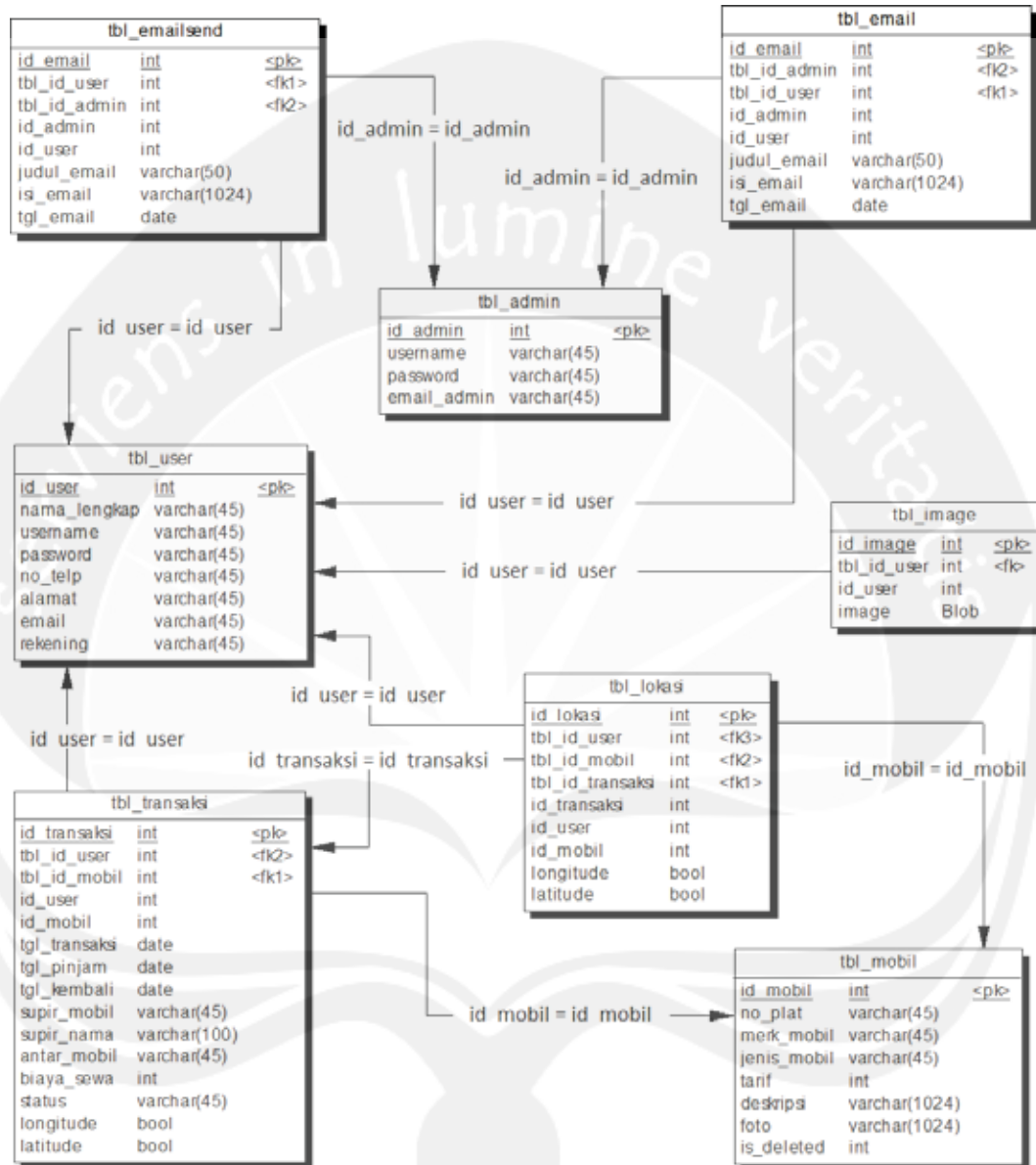
3.1.8 Deskripsi Entitas TBL_Image

Program Studi Teknik Informatika	DPPL – RENTCAR	49/ 72
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		

Nama	Tipe	Panjang	Keterangan
Id_image	Int	-	Id Image, Primary Key
Id_user	Int	-	Id User, Foregin key dari TBL_User
Image	Blob	-	Image Konfirmasi Pembayaran Rental Mobil



3.2 Physical Data Model



Gambar 3.1 Physical Data Model

4. Deskripsi Perancangan Antarmuka

4.1 Antarmuka Login *Mobile*



Gambar 4.1 Antarmuka Login *Mobile*

Antarmuka ini digunakan untuk melakukan proses *login* ke dalam sistem aplikasi *mobile*, pelanggan harus memasukkan *username* dan *password* dengan benar pada *textbox* yang telah disediakan pada aplikasi. Pada saat tombol *login* ditekan, sistem akan mengecek *username* dan *password* yang di masukkan dengan data *username* dan *password* pelanggan yang telah tersimpan di *database*. Jika data *username* dan *password* benar maka pelanggan akan masuk ke dalam sistem yang dilanjutkan ke antarmuka *list home* mobil, sebaliknya jika *username* dan *password* salah maka akan diberikan pesan peringatan untuk memasukkan kembali data *username* dan *password*.

Program Studi Teknik Informatika	DPPL – RENTCAR	52/ 72
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		

4.2 Antarmuka Registrasi Pelanggan



Gambar 4.2 Antarmuka Registrasi Pelanggan

Antarmuka ini digunakan untuk melakukan proses registrasi ke dalam sistem aplikasi *mobile*, untuk mendapat akses masuk ke dalam sistem pelanggan harus registrasi atau mendaftarkan diri sebagai *member* dengan mengisi kolom-kolom data diri dengan benar pada *textbox* yang telah disediakan. Pada saat tombol "*Sign Up*" ditekan, maka sistem akan mengecek *username* sudah terdaftar dan tersimpan di *database*. Jika data belum terdaftar akan disimpan di *database*, sebaliknya jika jika ada yang tidak cocok maka akan diberikan pesan peringatan untuk mengulanginya.

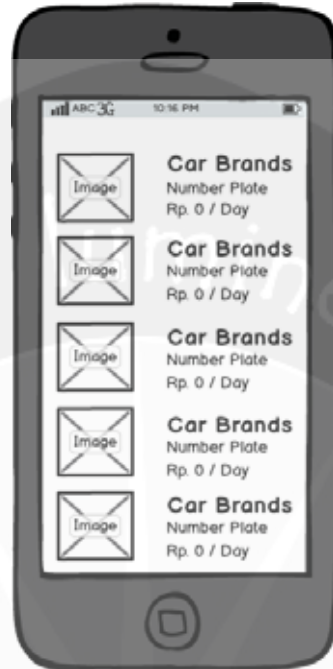
4.3 Antarmuka Pengelolaan Data Pelanggan



Gambar 4.3 Antarmuka Pengelolaan Data Pelanggan

Antarmuka ini digunakan untuk melakukan proses *update account* pelanggan yang sudah terdaftar sebagai *member* pada sistem aplikasi *mobile*. Pada *form* pertama data pelanggan akan ditampilkan setelah melakukan *login*, jika tombol "*Update*" ditekan maka akan dilanjutkan ke antarmuka *update data*. Terdapat kolom-kolom dalam bentuk *textbox* untuk mengganti data pelanggan yang sebelumnya sudah tersimpan dalam *database*, setelah data di *update* kemudian tombol "*Save*" ditekan maka sistem akan menyimpan data yang baru ke dalam *database*.

4.4 Antarmuka List Home Mobil



Gambar 4.4 Antarmuka List Home Mobil

Antarmuka ini digunakan untuk melihat mobil yang ditawarkan pada sistem aplikasi *mobile*, antarmuka ini merupakan antarmuka utama yang berisi daftar mobil dalam bentuk *listview*, pelanggan dapat memilih mobil yang akan disewa sesuai keinginan. *Listview* yang ditampilkan berupa *images car*, *car brands*, *number plate*, *cost / day*. Setelah dipilih salah satu mobil dan di tekan pada *listview*, maka akan di lanjutkan ke antarmuka *show detail* mobil.

4.5 Antarmuka Show Detail Mobil



Gambar 4.5 Antarmuka Show Detail Mobil

Antarmuka ini digunakan untuk melihat detail mobil yang ditawarkan pada sistem aplikasi *mobile* yang sebelumnya telah memilih pada *list home* mobil. Pada antarmuka ini pelanggan dapat melihat informasi mobil secara detail, data mobil yang ditampilkan diantaranya : *images car*, *car brands*, *number plate*, *kind of car*, *rental costs / day*, dan *description* mengenai mobil yang ditawarkan. Apabila tombol "Booking Now" ditekan, maka selanjutnya akan ditampilkan antarmuka untuk pemesanan mobil tersebut.

4.6 Antarmuka Transaksi Mobil

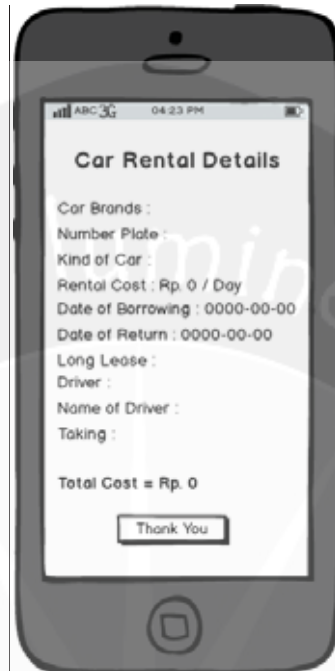


Gambar 4.6 Antarmuka Transaksi Mobil

Antarmuka ini digunakan untuk melakukan proses transaksi pemesanan mobil pada sistem aplikasi *mobile*, pelanggan harus mengisi kolom data yang disediakan untuk memesan mobil, diantaranya : *date of borrowing* dan *date of return* yang kemudian akan *men-generate* perkalian antara *car of cost* dan *long lease*, *checkbox* untuk memilih peminjaman *with driver* atau *without driver*, jika *with driver* maka pelanggan dapat memilih *name of driver* yang ada, *checkbox* pengambilan mobil untuk memilih apakah *delivered* atau *take to rent*. Kemudian setelah data dimasukan secara otomatis akan *men-generate total cost* yang ditampilkan dalam bentuk *textview*, jika pelanggan menekan tombol "*Booking*" akan ditampilkan *dialog* bahwa pesanan telah diterima dan akan dilanjutkan ke detail pemesanan.

Program Studi Teknik Informatika	DPPL – RENTCAR	57/ 72
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		

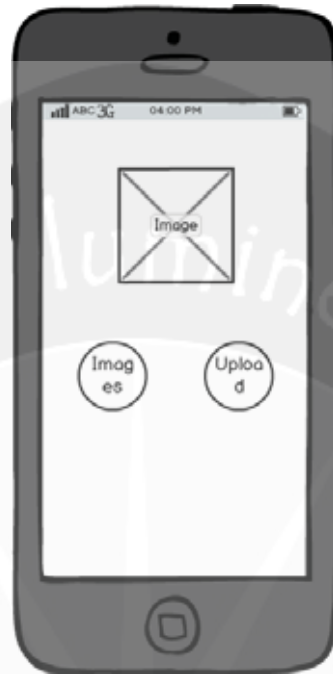
4.7 Antarmuka Detail Transaksi Mobil



Gambar 4.7 Antarmuka Detail Transaksi Mobil

Antarmuka ini digunakan untuk melihat detail transaksi pemesanan mobil pada sistem aplikasi *mobile*, pelanggan dapat melihat detail transaksi pemesanan mobil diantaranya : *car brands*, *number plate*, *kind of car*, *rental costs per day*, *date of borrowing*, *date of return*, *long lease*, *driver* jika menggunakan, *name of driver*, *taking car*, dan *total cost* keseluruhan. Kemudian ketika tombol "Thank You" ditekan, maka akan dilanjutkan ke menu *home*.

4.8 Antarmuka Konfirmasi Pembayaran



Gambar 4.8 Antarmuka Konfirmasi Pembayaran

Antarmuka ini digunakan untuk meng-*upload* foto untuk mengkonfirmasi pembayaran sewa mobil yang sebelumnya telah dipesan oleh pelanggan pada sistem aplikasi *mobile*. Pelanggan dapat mengambil gambar melalui kamera atau melalui *gallery* pada *smartphone*, maka gambar akan ditampilkan sebelum di *upload*. Ketika tombol "*Upload*" ditekan maka gambar akan dikirim ke petugas rental dan disimpan dalam *database* dan kemudian akan di lanjutkan ke antarmuka menu *home*.

4.9 Antarmuka Pengelolaan Data Email



Gambar 4.9 Antarmuka Pengelolaan Data Email

Antarmuka ini digunakan untuk mengelola data *email* pada sistem aplikasi *mobile*, pelanggan yang sebelumnya telah *login* dapat melihat *email inbox* yang telah dikirim petugas ataupun sebaliknya. *Email inbox* yang ditampilkan dalam bentuk *listview* dapat ditekan untuk melihat detail *email* yang masuk, setelah detail *email* ditampilkan dan pelanggan menekan tombol "Reply" maka akan dilanjutkan ke antarmuka *reply email*. Terdapat dua kolom untuk membalas *email*, diantaranya : *title* atau judul yang telah diambil dari *email inbox* dan *content* yang akan di kirim pelanggan. Pelanggan juga dapat membuat *email* baru dengan menekan tombol "Create New Email" yang kemudian akan ditampilkan antarmuka untuk menulis email baru.

4.10 Antarmuka Geolocation



Gambar 4.10 Antarmuka Geolocation

Antarmuka ini digunakan untuk melakukan proses *geolocation* atau navigasi penunjuk jalan pada sistem aplikasi *mobile*, terdapat dua *options* atau dua tombol pada antarmuka ini. Jika pelanggan menekan tombol "Rent" maka sistem akan menampilkan navigasi penunjuk jalan menuju ke rental mobil Nayara Transport dengan *latitude* dan *longitude* yang sudah terdefiniskan pada sistem, kemudian untuk tombol "Car" hanya digunakan untuk petugas sebagai sarana petunjuk jalan ke pelanggan yang memesan mobil dengan asumsi diantarkan ke lokasi pelanggan.

4.11 Antarmuka Login Web



Gambar 4.11 Antarmuka Login Web

Antarmuka ini digunakan untuk melakukan proses *login* ke dalam sistem aplikasi web yang hanya dapat dilakukan oleh petugas. Petugas harus memasukkan *username* dan *password* dengan benar pada *textbox* yang telah disediakan pada aplikasi. Pada saat tombol *login* ditekan, sistem akan mengecek *username* dan *password* yang di masukkan dengan data *username* dan *password* petugas yang telah tersimpan di *database*. Jika data *username* dan *password* benar maka petugas akan masuk ke dalam sistem yang dilanjutkan ke antarmuka *input new mobil*, sebaliknya jika *username* dan *password* salah maka akan diberikan pesan peringatan untuk memasukkan data kembali.

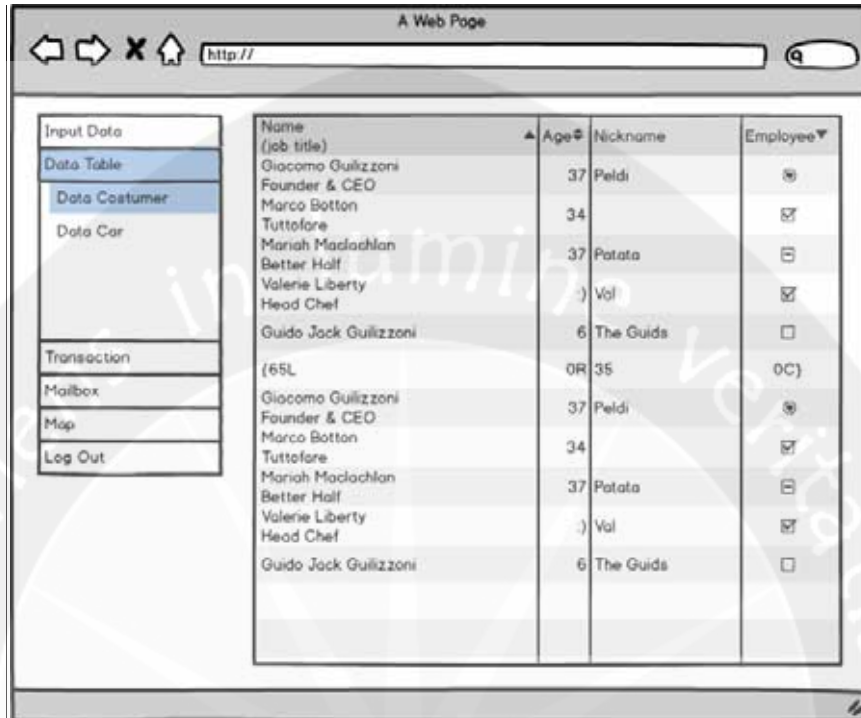
4.12 Antarmuka Input New Mobil

The image shows a web browser window titled "A Web Page" with a search bar containing "http://". On the left, a sidebar menu lists "Input Data", "Data Table", "Transaction", "Mailbox", "Map", and "Log Out", with "Input Data" highlighted. The main content area contains a form with the following fields: "Number Plate" (text input), "Car Brands" (text input), "Kind of Car" (ComboBox), "Cost of Car" (text input), and "Car Description" (text area). Below the description field are a "Choose File" button and a "Submit" button.

Gambar 4.12 Antarmuka Input New Mobil

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk memasukkan data mobil yang akan disimpan ke dalam *database*, yang kemudian dapat dilihat oleh pelanggan melalui aplikasi *mobile*. Petugas harus melengkapi data yang pada *form* yang disediakan untuk dapat disimpan ke dalam *database*, data yang harus diisi diantaranya : *number plate*, *car brands*, *kind of car*, *rental costs per day*, *description*, dan *images car*. Jika semua data telah dimasukkan dan tombol "*Submit*" ditekan, maka data telah berhasil disimpan ke dalam *database* dan akan dilanjutkan ke antarmuka *show* mobil web untuk melihat data mobil yang telah berhasil disimpan, namun sebaliknya jika salah satu data belum diisi maka akan diberikan peringatan bahwa ada data yang belum diisi.

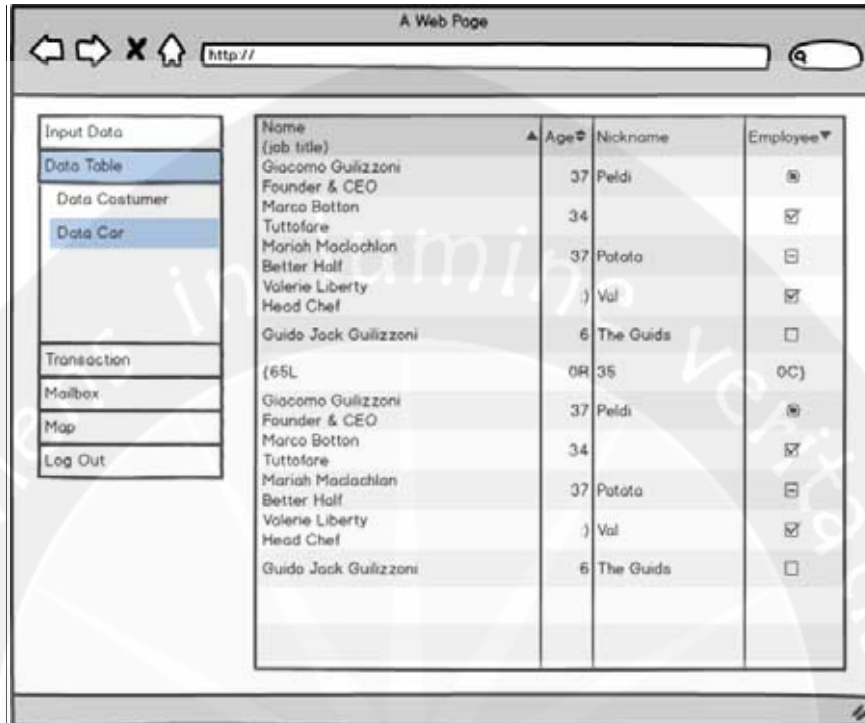
4.13 Antarmuka Show Pelanggan



Gambar 4.13 Antarmuka Show Pelanggan

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data profil pelanggan yang telah terdaftar sebagai *member* dan datanya tersimpan pada *database*, data akan ditampilkan pada tabel yang diberikan fungsi *pagination*. Pada antarmuka ini petugas dapat mencari data pelanggan untuk memudahkan dalam pencarian data pelanggan berdasarkan data yang dimasukkan, petugas juga dapat melakukan beberapa aksi : pertama petugas dapat mereset *password* pelanggan yang sebelumnya *password* ditentukan oleh pelanggan menjadi "*userpass*" atau *deffault*, kedua petugas dapat menghapus data pelanggan dari *database*.

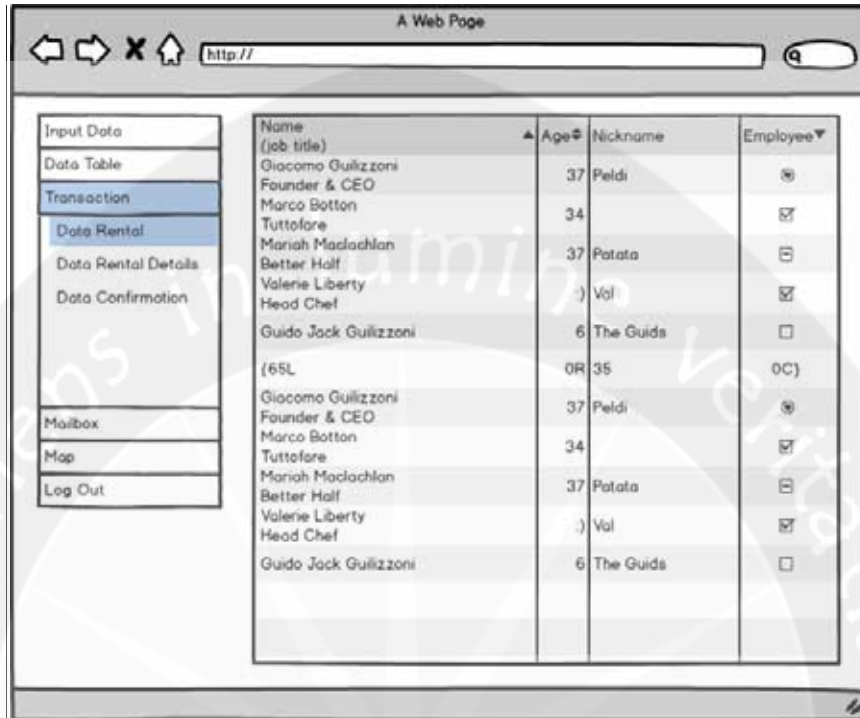
4.14 Antarmuka Show Mobil Web



Gambar 4.14 Antarmuka Show Mobil Web

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data mobil yang telah dimasukkan sebelumnya dan datanya tersimpan pada *database*, data akan ditampilkan pada tabel yang diberikan fungsi *pagination*. Pada antarmuka ini petugas dapat mencari data mobil untuk memudahkan dalam pencarian data mobil berdasarkan data yang dimasukkan, petugas juga dapat melakukan beberapa aksi : pertama petugas dapat mengubah data mobil untuk yang sebelumnya tersimpan pada *database*, kedua petugas dapat menghapus data mobil secara tidak permanen yang nantinya data mobil tersebut dapat ditampilkan lagi.

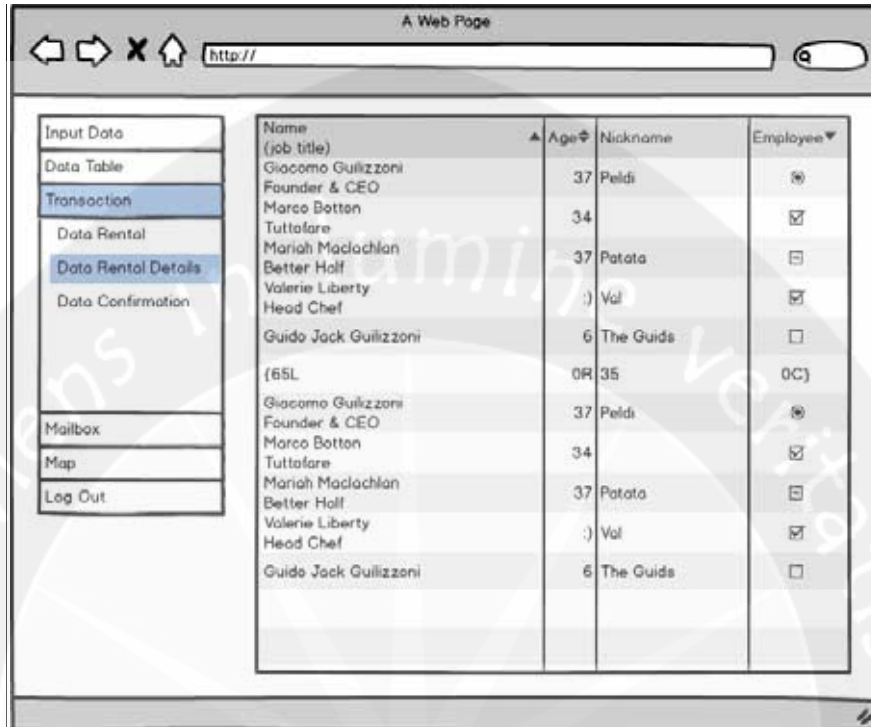
4.15 Antarmuka Show Pemesanan



Gambar 4.15 Antarmuka Show Pemesanan

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data pemesanan mobil dari pelanggan, data ditampilkan pada tabel yang diberikan fungsi *pagination*. Pada antarmuka ini petugas dapat mencari data pesanan untuk memudahkan dalam pencarian data mobil berdasarkan data yang dimasukkan.

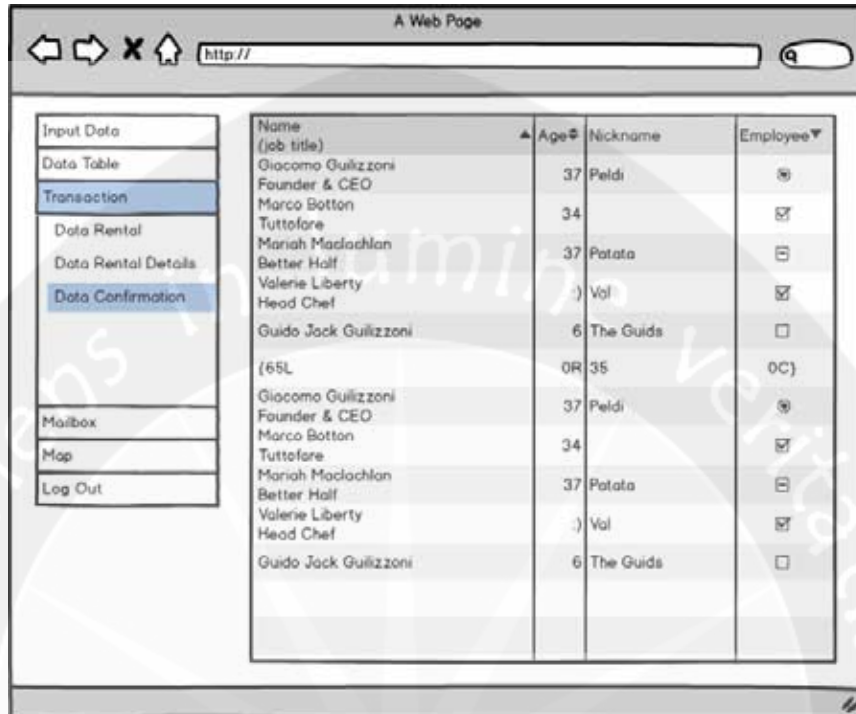
4.16 Antarmuka Show Detail Pemesanan



Gambar 4.16 Antarmuka Show Detail Pemesanan

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data detail pemesanan mobil, data ditampilkan pada tabel yang diberikan fungsi *pagination*. Pada antarmuka ini petugas dapat mencari data detail pesanan untuk memudahkan dalam pencarian data mobil berdasarkan data yang dimasukkan, petugas juga dapat menghapus data transaksi dan mengubah status pemesanan mobil dari pelanggan yang telah di konfirmasi sebelumnya. Pembayaran dapat melalui transfer rekening ataupun langsung datang ke rental.

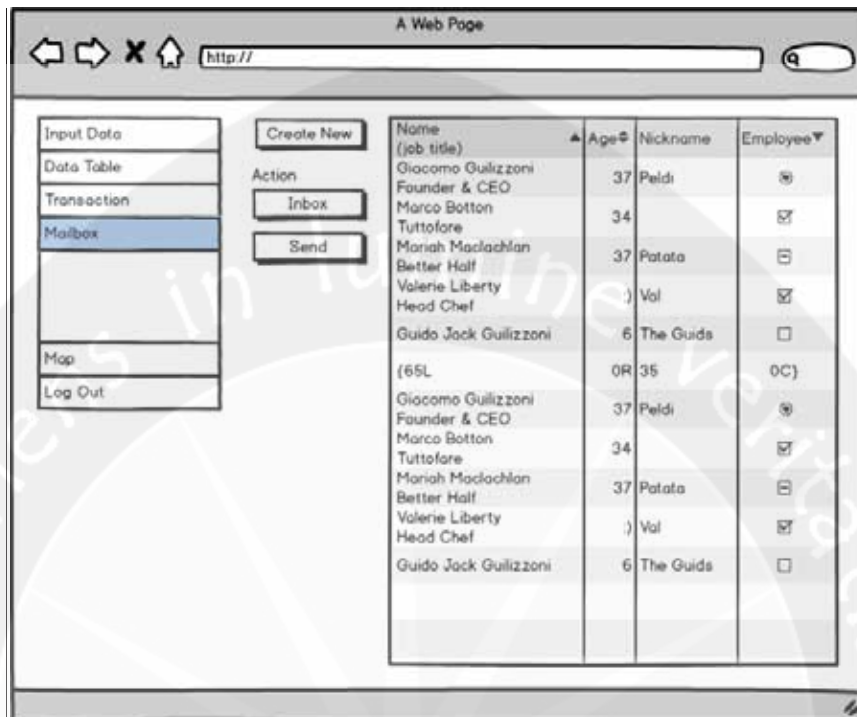
4.17 Antarmuka Show Konfirmasi Pembayaran



Gambar 4.17 Antarmuka Show Konfirmasi Pembayaran

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data konfirmasi pembayaran pemesanan mobil, data ditampilkan pada tabel yang diberikan fungsi *pagination*. Pada antarmuka ini petugas dapat mencari data konfirmasi pembayaran pesanan untuk memudahkan dalam pencarian data mobil berdasarkan data yang dimasukkan.

4.18 Antarmuka Email Inbox Web



Gambar 4.18 Antarmuka Email Inbox Web

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data *email* yang telah dikirimkan pelanggan melalui aplikasi *mobile*, data ditampilkan pada tabel yang diberikan fungsi *pagination*. Petugas dapat mencari data *email inbox* untuk memudahkan dalam pencarian *email inbox* berdasarkan data yang dimasukkan, petugas juga dapat melakukan beberapa aksi : pertama petugas dapat membuat pesan baru ketika menekan tombol "*Creat New*", kedua petugas dapat melihat detail *email*, ketiga petugas dapat membalas *email* yang dikirimkan oleh pelanggan, keempat petugas dapat melihat data *email* yang dikirimkan untuk pelanggan ketika menekan tombol "*Send Email*", kelima petugas dapat menghapus data *email* dari *database*.

4.19 Antarmuka Read Email Web



Gambar 4.19 Antarmuka Read Email Web

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk melihat data detail *email inbox* yang telah dikirimkan pelanggan melalui aplikasi *mobile* yang tersimpan pada *database*, data akan ditampilkan dalam bentuk *form*. Ketika petugas menekan tombol "*Reply*" maka akan dilanjutkan ke antarmuka *reply* web dan disitu petugas dapat mengisikan konten yang akan dikirim kepada pelanggan.

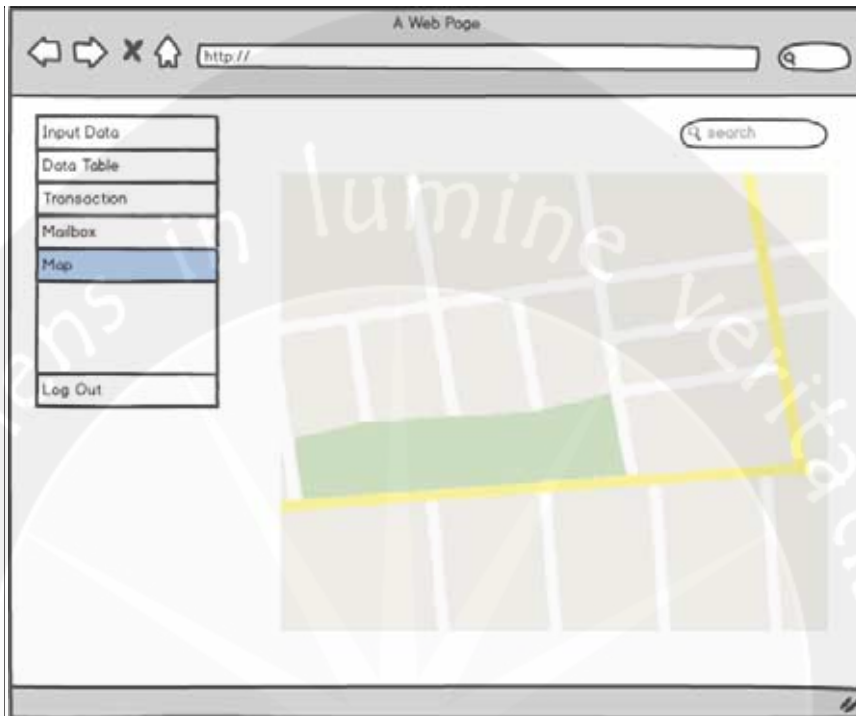
4.20 Antarmuka Email Reply Web



Gambar 4.20 Antarmuka Email Reply Web

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk membalas *email* yang telah dikirimkan pelanggan melalui aplikasi *mobile* yang tersimpan pada *database*. Petugas harus mengisi kolom deskripsi atau isi *email* yang disediakan pada *form*, setelah diisi dan kemudian menekan tombol "Send" maka data *reply email* akan disimpan ke dalam *database* agar dapat dilihat pelanggan melalui aplikasi *mobile*.

4.21 Antarmuka Cari Lokasi Mobil



Gambar 4.21 Antarmuka Cari Lokasi Mobil

Antarmuka ini digunakan oleh petugas yang sebelumnya telah *login* ke dalam sistem untuk mencari lokasi mobil yang telah dipinjam oleh pelanggan melalui *maps*. Petugas dapat mencari mobil untuk mencari lokasi keberadaan mobil yang sedang dibawa oleh pelanggan. Pada *form* ini data lokasi akan ditampilkan berdasarkan *longitude* dan *latitude* lokasi keberadaan mobil.

SKPL

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

PEMBANGUNAN APLIKASI MOBILE UNTUK
NAVIGASI PADA PERSEWAAN MOBIL

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Firmansyah Abdul Rafi / 6695

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

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

Program Studi Teknik Informatika	SKPL – RENTCAR	1/37
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		

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	8
1.5	DESKRIPSI UMUM (OVERVIEW)	8
2	DESKRIPSI KEBUTUHAN	9
2.1	PERSPEKTIF PRODUK	9
2.2	FUNGSI PRODUK	11
2.3	KARAKTERISTIK PENGGUNA	16
2.4	BATASAN-BATASAN	17
2.5	ASUMSI DAN KETERGANTUNGAN	17
3	KEBUTUHAN KHUSUS	17
3.1	KEBUTUHAN ANTARMUKA EKSTERNAL	17
3.1.1	Antarmuka pemakai	17
3.1.2	Antarmuka perangkat keras	18
3.1.3	Antarmuka perangkat lunak	18
3.1.4	Antarmuka Komunikasi	19
4	SPEKIFIKASI RINCI KEBUTUHAN	21
4.1	SPEKIFIKASI KEBUTUHAN FUNGSIONALITAS	21
4.1.1	Use case Spesification : Login Mobile	21
4.1.2	Use case Spesification : Registrasi Pelanggan	22
4.1.3	Use case Spesification : Pengelolaan Data Pelanggan	23
4.1.4	Use case Spesification : Find Data Mobil	24
4.1.5	Use case Spesification : Booking Mobil	25
4.1.6	Use case Spesification : Geolocation	27
4.1.7	Use case Spesification : Email Mobile	28
4.1.8	Use case Spesification : Login Web	29
4.1.9	Use case Spesification : Pengelolaan Data Pelanggan	30
4.1.10	Use case Spesification : Pengelolaan Data Mobil	31
4.1.11	Use case Spesification : Pengelolaan Data Transaksi	33
4.1.12	Use case Spesification : Email Web	34
4.1.13	Use case Spesification : Mencari Lokasi Mobil	35

Daftar Gambar

Gambar 2.1	Arsitektur Perangkat Lunak RENTCAR.....	11
Gambar 3.1	Use Case Diagram.....	20
Gambar 5.1	ERD.....	37



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak "Aplikasi *Mobile* Untuk Navigasi Pada Persewaan Mobil" 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-RENTCAR ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak RENTCAR dikembangkan dengan tujuan untuk :

1. Pengelolaan data pelanggan sebagai member.
2. Pengelolaan pencatatan data persewaan mobil.
3. Memberikan informasi mobil yang disediakan.
4. Pengelolaan pembayaran persewaan mobil oleh pelanggan.
5. Navigasi petunjuk arah menggunakan *geolocation* untuk pengambil atau pengantar mobil.
6. Mencari lokasi mobil yang sedang di sewa.
7. *Email* sebagai penghubung antara pelanggan dan petugas untuk penawaran harga sewa mobil.

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-RENTCAR-XXX	Kode yang merepresentasikan kebutuhan pada Aplikasi <i>Mobile</i> Untuk Navigasi Pada Persewaan Mobil di mana XXX merupakan nomor fungsi produk.
RENTCAR	Perangkat lunak pengelolaan data yang berjalan di perangkat <i>mobile</i> dan web dalam kasus persewaan mobil.
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 <i>e-mail, FTP, dan World Wide Web</i> .
Administrator	Petugas yang mengolah data master rental mobil Nayara Transport.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.

Rent Car Nayara Transport	Nayara Transport adalah sebuah perusahaan yang bergerak di bidang jasa transportasi, perusahaan ini menawarkan harga sewa mobil yang murah, kompetitif, dan selalu mengutamakan kualitas kenyamanan pelayanan kepada pelanggan. Nayara Transport bertujuan untuk menjadikan perusahaannya sebagai sebuah perusahaan persewaan mobil yang terpercaya bagi pelanggan yang mengutamakan pelayanan guna bersaing dengan perusahaan persewaan mobil lainnya.
Pelanggan	Pelanggan dari Rent Car Nayara Transport yang terdaftar sebagai member untuk menyewa mobil di rental ini melalui aplikasi <i>mobile</i> .

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Firmansyah Abdul Rafi / 6695 *Spesifikasi Kebutuhan Perangkat Lunak (SKPL) EMERSY*, 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 RENTCAR 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 RENTCAR tersebut.

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

2 Deskripsi Kebutuhan

2.1 Perspektif produk

RENTCAR merupakan perangkat lunak yang dikembangkan untuk membantu pengelolaan sistem informasi pada *mobile* dan web. Sistem ini membantu "Nayara Transport" untuk membangun sebuah sistem informasi yang dapat meng-*cover* semua proses bisnis yang terjadi di perusahaan tersebut. Sistem ini dapat dipergunakan oleh perusahaan Nayara Transport sendiri maupun dipergunakan oleh pelanggan dari perusahaan Nayara Transport dan sistem sendiri, pada sistem ini dibagi menjadi dua jenis, aplikasi *mobile* dan aplikasi web.

Pada aplikasi *mobile* digunakan oleh pihak pelanggan Nayara Transport. Aplikasi *mobile* ini menangani antara lain: pengelolaan data pelanggan sebagai member, informasi persediaan mobil, pemesanan mobil, navigasi lokasi rental maupun lokasi pelanggan,

Program Studi Teknik Informatika	SKPL – RENTCAR	9/ 37
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		

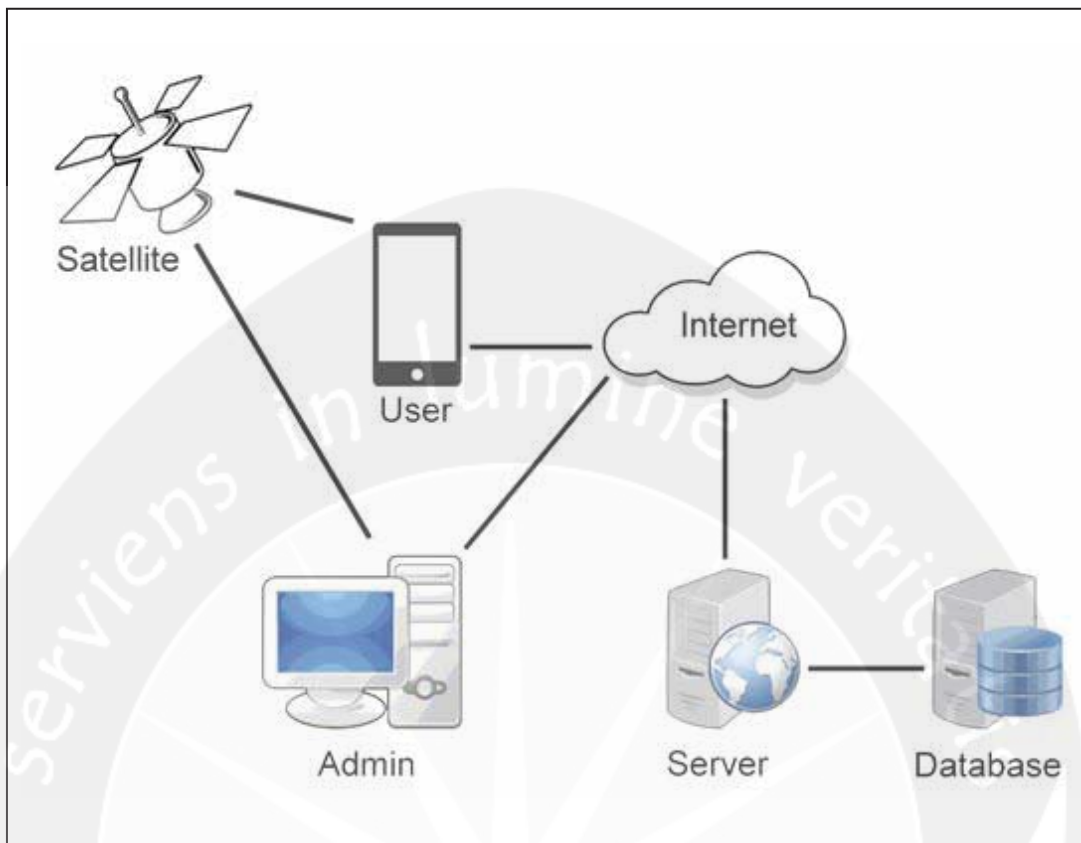
dan *email*. Untuk aplikasi web, sistem ini hanya dapat digunakan untuk petugas Nayara Transport. Melalui aplikasi web tersebut petugas dapat mengelola data mobil, *reset password* pelanggan, konfirmasi pembayaran sewa, mencari lokasi mobil, dan *email*.

Perangkat lunak *mobile* RENTCAR ini berjalan pada *platform mobile device*, dan dibuat menggunakan bahasa pemrograman *Java*. Untuk lingkungan pemrogramannya menggunakan *Eclipse Mars*. Untuk web bisa berjalan pada *web browser* apapun, dan dibuat menggunakan bahasa pemrograman *CI (Code Igniter)*. Sedang untuk database, akan dipergunakan *MySQL* untuk kedua aplikasi *mobile* dan web, akan mengacu kepada *database* yang sama.

Pelanggan akan berinteraksi dengan aplikasi melalui antarmuka *GUI (Graphical Pelanggan Interface)*. Pada aplikasi seperti terlihat pada gambar 2.1, arsitektur perangkat lunak yang digunakan berupa *client server*, dimana data disimpan di server. Pelanggan dapat mengakses data yang ada di server tersebut secara *online* dengan memanggil *web service* pada *website* yang tersedia di *web server*.

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

Program Studi Teknik Informatika	SKPL – RENTCAR	10/ 37
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		



Gambar 2.1 Arsitektur Perangkat Lunak RENTCAR

2.2 Fungsi Produk

Fungsi produk perangkat lunak RENTCAR adalah sebagai berikut :

APLIKASI MOBILE

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

Merupakan fungsi yang digunakan oleh semua pelanggan yang terdaftar sebagai member untuk dapat masuk ke dalam sistem dan memperoleh hak akses sesuai dengan peran yang dimiliki oleh pelanggan tersebut.

2. Fungsi *Registrasi Pelanggan* (**SKPL-RENTCAR-002**).

Merupakan fungsi yang digunakan untuk mendaftarkan diri sebagai member dari Nayara Transport.

3. Fungsi *Pengelolaan Data Pelanggan* (**SKPL-RENTCAR-003**).

Merupakan fungsi yang digunakan pelanggan yang telah terdaftar sebagai member untuk mengelola data pelanggan.

Fungsi *Pengelolaan Data Pelanggan* meliputi :

a. Fungsi *Display Account* (**SKPL-RENTCAR-003-01**).

Merupakan fungsi yang digunakan untuk melihat data pelanggan yang telah terdaftar sebagai member.

b. Fungsi *Edit Data Pelanggan* (**SKPL-RENTCAR-003-02**). Merupakan fungsi yang digunakan untuk mengubah data pelanggan, serta dapat ubah *password* sesuai keinginan pelanggan.

4. Fungsi *Find Data Mobil* (**SKPL-RENTCAR-004**).

Merupakan fungsi yang digunakan pelanggan yang telah terdaftar sebagai member untuk mencari data mobil.

5. Fungsi *Booking Mobil* (**SKPL-RENTCAR-005**).

Merupakan fungsi yang digunakan pelanggan yang telah terdaftar sebagai member untuk memesan mobil.

a. Fungsi *Show Mobil* (**SKPL-RENTCAR-005-01**).

Merupakan fungsi yang digunakan untuk melihat detail data mobil.

b. Fungsi *Booking Mobil* (**SKPL-RENTCAR-005-02**).

Merupakan fungsi yang digunakan oleh pelanggan untuk memesan mobil.

c. Fungsi *Upload Foto* (**SKPL-RENTCAR-005-03**).

Merupakan fungsi yang digunakan untuk mengupload foto konfirmasi pembayaran rental mobil.

6. Fungsi *Navigasi Geolocation* (**SKPL-RENTCAR-006**).

Merupakan fungsi yang digunakan untuk navigasi atau *geolocation* petunjuk arah.

Fungsi *Navigasi Geolocation* meliputi :

a. Fungsi *Find Rent* (**SKPL-RENTCAR-006-01**).

Merupakan fungsi yang digunakan untuk mencari jalur tercepat dari posisi pelanggan menuju rental mobil.

b. Fungsi *Find Pelanggan* (**SKPL-RENTCAR-006-02**).

Merupakan fungsi yang digunakan untuk mencari jalur pengantaran mobil dari posisi rental menuju pemesan atau pelanggan.

7. Fungsi *Email Mobile* (**SKPL-RENTCAR-007**)

Merupakan fungsi yang digunakan pelanggan yang terdaftar sebagai member untuk mengelola *email*.

Fungsi *Email Mobile* meliputi :

a. Fungsi *Create New Email* (**SKPL-RENTCAR-007-01**).

Program Studi Teknik Informatika	SKPL – RENTCAR	13/ 37
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		

Merupakan fungsi yang digunakan untuk membuat pesan baru ke petugas rental.

b. Fungsi *Reply Email* (**SKPL-RENTCAR-007-02**).

Merupakan fungsi yang digunakan untuk membalas *email* dari petugas rental.

APLIKASI WEB

8. Fungsi *Fungsi Login* (**SKPL-RENTCAR-008**)

Merupakan fungsi yang digunakan oleh petugas untuk masuk ke dalam sistem dan memperoleh hak akses sistem.

9. Fungsi *Pengelolaan Data Pelanggan Web* (**SKPL-RENTCAR-009**)

Merupakan fungsi yang digunakan oleh petugas untuk mengelola data pelanggan yang terdaftar sebagai member.

a. Fungsi *Reset Password* (**SKPL-RENTCAR-009-01**).

Merupakan fungsi yang digunakan oleh petugas untuk mereset *password* pelanggan.

b. Fungsi *Delete Pelanggan* (**SKPL-RENTCAR-009-02**).

Merupakan fungsi yang digunakan oleh petugas untuk menghapus data pelanggan.

c. Fungsi *Find Pelanggan* (**SKPL-RENTCAR-009-03**).

Merupakan fungsi yang digunakan oleh petugas untuk mencari data pelanggan.

10. Fungsi *Pengelolaan Data Mobil* (**SKPL-RENTCAR-010**)

Merupakan fungsi yang digunakan oleh petugas untuk mengelola data mobil.

Fungsi *Pengelolaan Data Mobil* meliputi:

Program Studi Teknik Informatika	SKPL – RENTCAR	14/ 37
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 *Entri Data Mobil*(**SKPL-RENTCAR-010-01**).
Merupakan fungsi yang digunakan oleh petugas untuk menambahkan data mobil.
- b. Fungsi *Edit Data Mobil*(**SKPL-RENTCAR-010-02**).
Merupakan fungsi yang digunakan oleh petugas untuk mengubah data mobil.
- c. Fungsi *Delete Data Mobil*(**SKPL-RENTCAR-010-03**).
Merupakan fungsi yang digunakan oleh petugas untuk menghapus data mobil.
- d. Fungsi *Find Data Mobil*(**SKPL-RENTCAR-010-04**).
Merupakan fungsi yang digunakan oleh petugas untuk mencari data mobil.

11. Fungsi *Transaksi Mobil* (**SKPL-RENTCAR-011**)

Merupakan fungsi yang digunakan oleh petugas untuk mengelola data transaksi atau pemesanan mobil.

Fungsi *Pengelolaan Data Transaksi Mobil* meliputi:

- a. Fungsi *Ubah Status* (**SKPL-RENTCAR-011-01**).
Merupakan fungsi yang digunakan oleh petugas untuk mengubah status pemesanan mobil.
- b. Fungsi *Delete Transaksi* (**SKPL-RENTCAR-011-02**).
Merupakan fungsi yang digunakan oleh petugas untuk menghapus data pemesanan mobil.
- c. Fungsi *Find Transaksi*(**SKPL-RENTCAR-011-03**).
Merupakan fungsi yang digunakan oleh petugas untuk mencari data transaksi.

12. Fungsi *Mecari Lokasi Mobil* (**SKPL-RENTCAR-012**)

Merupakan fungsi yang digunakan oleh petugas untuk mencari lokasi mobil yang ditampilkan dari *google maps*.

13. Fungsi *Email Web* (**SKPL-RENTCAR-013**)

Merupakan fungsi yang digunakan oleh petugas untuk mengelola *email* dari *member* *Nayara Transport*.

Fungsi *Pengelolaan Email Web* meliputi:

- a. Fungsi *Create New Email* (**SKPL-RENTCAR-013-01**).
Merupakan fungsi yang digunakan oleh petugas untuk mengirim *email* ke pelanggan.
- b. Fungsi *Read Email*(**SKPL-RENTCAR-013-02**).
Merupakan fungsi yang digunakan oleh petugas untuk menampilkan *email* masuk atau *email* terkirim.
- c. Fungsi *Reply Email*(**SKPL-RENTCAR-013-03**).
Merupakan fungsi yang digunakan oleh petugas untuk membalas *email* dari pelanggan.
- d. Fungsi *Delete Email*(**SKPL-RENTCAR-013-04**).
Merupakan fungsi yang digunakan oleh petugas untuk menghapus data *email*.
- e. Fungsi *Find Data Email*(**SKPL-RENTCAR-013-05**).
Merupakan fungsi yang digunakan oleh petugas untuk mencari data *email* pelanggan.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak RENTCAR adalah sebagai berikut :

1. Memahami pengoperasian perangkat *mobile*.
2. Mengerti tentang internet dan web.
3. Memahami penggunaan aplikasi RENTCAR.

Program Studi Teknik Informatika	SKPL – RENTCAR	16/ 37
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		

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak RENTCAR tersebut adalah :

1. Kebijaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak RENTCAR.

2. Keterbatasan perangkat keras

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

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat *Mobile Smartphone* yang menggunakan sistem operasi versi android 4.1.2 keatas, serta aplikasi *web browser* apapun.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

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

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk *form-form* yang tersedia.

Program Studi Teknik Informatika	SKPL – RENTCAR	17/ 37
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		

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak RENTCAR adalah:

1. Perangkat *mobile* Android.

3.1.3 Antarmuka perangkat lunak

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

1. Nama : MySQL

Sumber : MySQL

Sebagai *database management system* (DBMS) yang digunakan untuk penyimpanan data di sisi server.

2. Nama : Windows 7/64 Bit

Sumber : Microsoft

Sebagai sistem operasi untuk pembuatan aplikasi RENTCAR.

3. Nama : IE/Firefox/Chrome/Opera/dll

Sumber : berbagai sumber

Sebagai aplikasi *web browser* untuk membuka sistem web.

4. Nama : Eclipse Mars

Sumber : Free and Open Source Community

Sebagai *tool* perancangan yang dibutuhkan untuk membuat RENTCAR.

5. Nama : Android SDK

Sumber : Google Inc

Sebagai *library* dalam perancangan RENTCAR.

6. Nama : Java

Sumber : Oracle Corporation

Program Studi Teknik Informatika	SKPL – RENTCAR	18/ 37
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		

Sebagai bahasa pemrograman *mobile* dalam pembuatan aplikasi RENTCAR.

7. Nama : PHP / CI (Code Igniter)

Sumber : The PHP Group

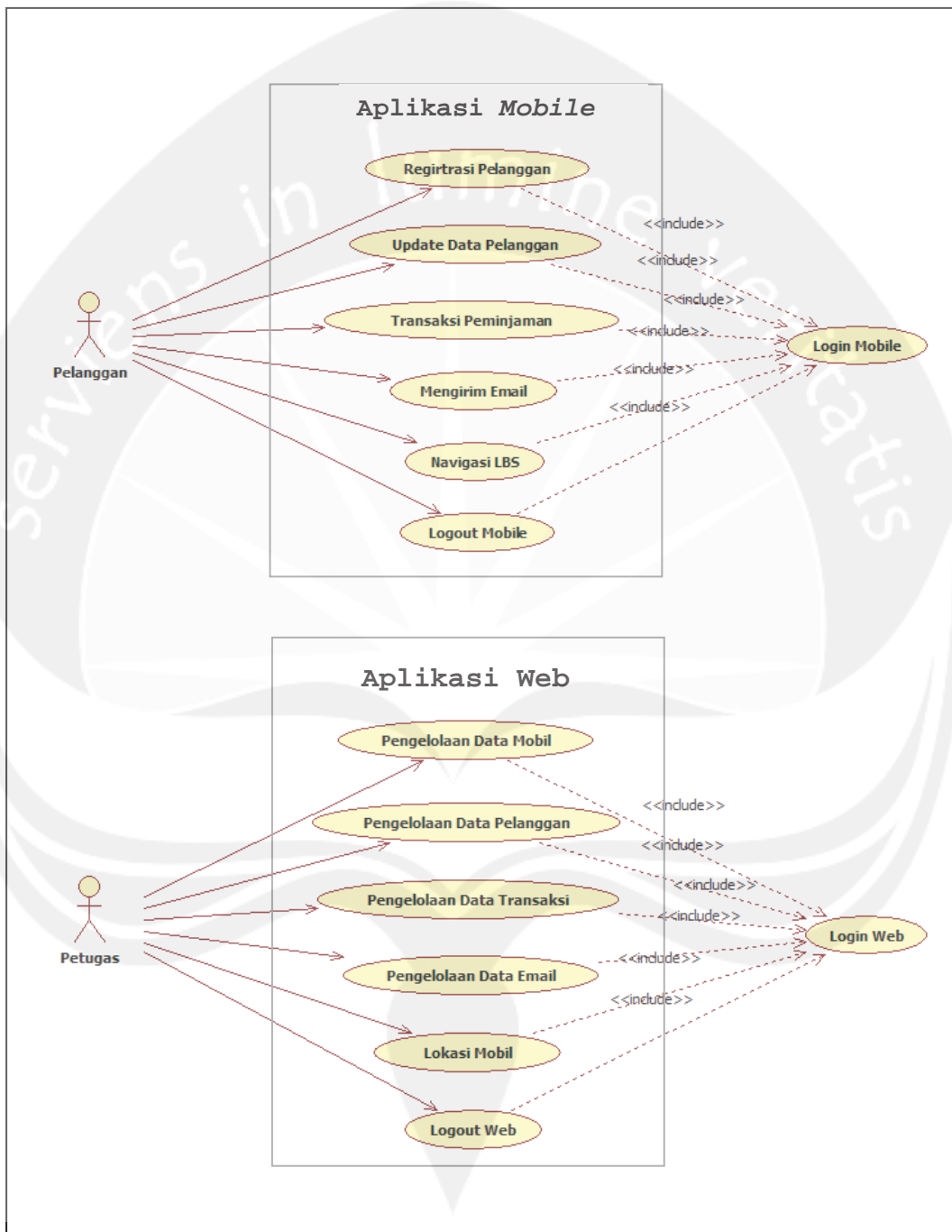
Sebagai bahasa pemrograman *web service* yang digunakan untuk membuat aplikasi RENTCAR.

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak RENTCAR menggunakan *protocol* HTTP.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 3.1 Use Case Diagram RENTCAR

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

APLIKASI MOBILE

4.1.1 Use case Spesification : Login Mobile

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. *Login* didasarkan pada sebuah *username* dari pelanggan dan *password* yang berupa rangkaian karakter.

2. Primary Actor

1. Pelanggan

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. Aktor memasukkan *username* dan *password*.
4. Sistem memeriksa *username* dan *password* yang diinputkan 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 *username* atau *password* tidak sesuai.

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

Program Studi Teknik Informatika	SKPL – RENTCAR	21/37
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		

7. PreConditions

none

8. PostConditions

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

4.1.2 Use case Spesification : Registrasi Pelanggan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mendaftarkan diri sebagai member Nayara Transport, yang nantinya akan disimpan ke dalam database oleh sistem.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan registrasi.
2. Sistem menampilkan antarmuka untuk memasukkan data diri.
3. Aktor menginputkan data diri pada *form* yang disediakan sistem.
4. Aktor meminta sistem menyimpan data diri pelanggan yang telah di inputkan.
5. Sistem mengecek data diri pelanggan yang telah di masukkan aktor.
E-1 Masukkan data diri pelanggan salah.
6. Data di simpan ke database oleh sistem.
7. *Use case* telah selesai.

Program Studi Teknik Informatika	SKPL – RENTCAR	22/ 37
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		

5. Alternative Flow

none

6. Error Flow

E-1 Terdapat kolom yang tidak sesuai atau belum diisi.

1. Sistem memberikan peringatan bahwa terdapat inputan yang salah.
2. Kembali ke *Basic Flow* langkah 3.

7. PreConditions

none

8. PostConditions

1. Data pelanggan dalam database telah ter-update.

4.1.3 Use case Spesification : Pengelolaan Data Pelanggan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data data pelanggan yang terdaftar sebagai member, pada *use case* ini aktor dapat melakukan *display* data dan *edit* data pelanggan.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pelanggan.
2. Sitem menampilkan data pelanggan
3. Sistem memberikan pilihan untuk melakukan *update* data pelanggan.
4. Aktor memilih untuk melakukan *edit* data pelanggan.

5. Aktor memasukkan data diri pada *form* yang disediakan sistem.
6. Aktor meminta sistem untuk menyimpan data yang telah di masukkan.
7. Sistem mengecek data pelanggan yang telah di masukkan aktor.
 - E-1 Masukkan data pelanggan salah.
8. Sistem menyimpan data pelanggan yang selesai di *edit* ke *database*.
9. *Use Case* telah selesai.

5. Alternative Flow

none

6. Error Flow

E-1 Terdapat kolom yang tidak sesuai atau belum diisi.

1. Sistem memberikan peringatan bahwa terdapat masukkan yang salah.
2. Kembali ke *Basic Flow* langkah 4.

7. PreConditions

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

8. PostConditions

1. Data pelanggan dalam *database* telah ter-*update*.

4.1.4 Use case Spesification : Find Data Mobil

1. Brief Description

Use Case ini digunakan oleh aktor untuk mencari data mobil, pada use case ini aktor dapat melakukan *find* data mobil berdasarkan masukkan data dari pelanggan.

2. Primary Actor

1. Pelanggan

Program Studi Teknik Informatika	SKPL – RENTCAR	24/ 37
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 *find* data mobil.
2. Sistem memberikan pilihan untuk melakukan *find* data.
3. Aktor memasukkan data mobil yang akan dicari.
4. Sistem mengecek dan menampilkan data mobil yang dicari.
E-1 Masukkan data mobil tidak ada.
5. Aktor memilih data mobil yang dicari
6. *Use Case* selesai.

5. Alternative Flow

none

6. Error Flow

- E-1 Data mobil dimasukkan aktor tidak ada.
1. Sistem memberikan pesan peringatan bahwa data yang di masukkan tidak ada.
 2. Kembali ke *Basic Flow* Langkah ke 3.

7. PreConditions

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

8. PostConditions

1. Aktor memilih data mobil yang dicari

4.1.5 Use case Spesification : Booking Mobil

1. Brief Description

Use Case ini digunakan oleh aktor untuk memesan mobil, pada *use case* ini aktor dapat melakukan *display* data mobil dan *booking* mobil.

Program Studi Teknik Informatika	SKPL – RENTCAR	25/ 37
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. Pelanggan

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melihat data mobil.
2. Sistem menampilkan data mobil dalam bentuk *list*.
3. Sistem memberikan pilihan untuk melakukan *booking* mobil.
4. Aktor memilih mobil yang akan di *booking*.
5. Sistem menampilkan detail mobil.
6. Aktor memilih *booking* mobil.
 - A-1 Aktor membatalkan *booking* mobil.
7. Sistem menampilkan UI *booking* mobil.
8. Aktor menginputkan ketentuan *booking* mobil.
9. Sistem mengecek data pesanan yang di masukkan.
 - E-1 Masukkan data pesanan salah.
10. Sistem menyimpan data *booking* atau pesanan ke *database*
11. *Use Case* selesai.

5. Alternative Flow

A-1 Aktor memilih membatalkan *booking* mobil

1. Aktor memilih membatalkan pesanan yang ada.
2. Berlanjut ke *Basic Flow* langkah ke 4.

6. Error Flow

E-1 Data yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa masukkan data salah.
2. Kembali ke *Basic Flow* Langkah ke 9.

Program Studi Teknik Informatika	SKPL – RENTCAR	26/ 37
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		

7. PreConditions

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

8. PostConditions

1. Data *booking* atau pesanan mobil telah *ter-update*.

4.1.6 Use case Spesification : Geolocation

1. Brief Description

Use Case ini digunakan oleh aktor untuk navigasi petunjuk arah, pada *use case* ini aktor dapat melakukan navigasi ke rental.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melihat rute jalan ke rental.
2. Sistem menampilkan antarmuka *geolocation*.
3. Sistem menampilkan *maps* atau peta.
4. Sistem mengambil data lokasi pelanggan.
5. Sistem memberikan petunjuk arah kepada aktor dalam *maps* atau peta.
6. Aktor mengikuti petunjuk jalan ke rental
7. *Use Case* selesai.

5. Alternative Flow

none

6. Error Flow

none

7. PreConditions

1. *Use Case* Login telah dilakukan.

Program Studi Teknik Informatika	SKPL – RENTCAR	27/ 37
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. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor mendapatkan petunjuk jalan ke rental.

4.1.7 Use case Spesification : Email Mobile

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola *email*, pada *use case* ini aktor dapat melakukan *create new email* dan *reply email*.

2. Primary Actor

1. Pelanggan

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk mengelola data *email*.
2. Sistem menampilkan antarmuka *email*.
3. Sistem memberikan pilihan untuk melakukan *create* atau *reply*.
4. Aktor memilih *create new email*.
A-1 Aktor memilih untuk *reply email*.
5. Aktor menginputkan data *email*.
6. Sistem mengecek data *email* yang dimasukkan.
E-1 Masukkan data *email* salah.
7. Sistem menyimpan data *email* ke *database*
8. *Use Case* selesai.

5. Alternative Flow

A-1 Aktor memilih *reply email*.

1. Sistem menampilkan antarmuka *reply email*
2. Aktor menginputkan data *email*.
3. Berlanjut ke *Basic Flow* langkah ke 4.

6. Error Flow

Program Studi Teknik Informatika	SKPL – RENTCAR	28/ 37
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 *email* yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa masukkan salah.
2. Kembali ke *Basic Flow* Langkah ke 5.

7. PreConditions

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

8. PostConditions

1. Data *email* telah ter-update.

APLIKASI WEB

4.1.8 Use case Spesification : Login Web

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. *Login* didasarkan pada sebuah *username* dari petugas dan *password* yang berupa rangkaian karakter.

2. Primary Actor

1. Petugas

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. Aktor memasukkan *username* dan *password*.
4. Sistem memeriksa *username* dan *password* yang dimasukkan aktor.

E-1 *username* atau *password* tidak sesuai.

5. Sistem memberikan akses ke aktor.
6. *Use Case* selesai.

Program Studi Teknik Informatika	SKPL – RENTCAR	29/ 37
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		

5. Alternative Flow

none

6. Error Flow

E-1 *username* atau *password* tidak sesuai.

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

7. PreConditions

none

8. PostConditions

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

4.1.9 Use case Spesification : Pengelolaan Data Pelanggan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pelanggan. Aktor dapat melakukan *reset password* pelanggan, *delete* data pelanggan, *find* data pelanggan.

2. Primary Actor

1. Petugas

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pelanggan.
2. Sistem menampilkan antarmuka untuk pengelolaan data pelanggan.
3. Sistem memberikan pilihan pada aktor untuk melakukan *reset password* pelanggan, *delete* data pelanggan, atau *find* data pelanggan.

Program Studi Teknik Informatika	SKPL – RENTCAR	30/ 37
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. Aktor memilih untuk melakukan *reset password* pelanggan.

A-1 Aktor memilih untuk melakukan *delete* data pelanggan

A-2 Aktor memilih untuk melakukan *find* data pelanggan

5. Aktor melakukan *reset password* pada form yang disediakan sistem.

6. Sistem menyimpan data data pelanggan.

7. *Use Case* selesai.

5. Alternative Flow

none

6. Error Flow

none

7. PreConditions

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

8. PostConditions

1. Data pelanggan telah tersimpan ke dalam database.

4.1.10 Use case Spesification : Pengelolaan Data Mobil

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data mobil. Aktor dapat melakukan *entri* mobil, *edit* data mobil, *delete* data mobil, *find* data mobil.

2. Primary Actor

1. Petugas

3. Supporting Actor

none

Program Studi Teknik Informatika	SKPL – RENTCAR	31/37
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 data mobil.
2. Sistem menampilkan antarmuka untuk pengelolaan data mobil.
3. Sistem memberikan pilihan pada aktor untuk melakukan *entri* data mobil, *edit* data mobil, *delete* data mobil, atau *find* data mobil.
4. Aktor memilih untuk melakukan *entri* data mobil.
 - A-1 Aktor memilih untuk melakukan *edit* data mobil
 - A-2 Aktor memilih untuk melakukan *delete* data mobil
 - A-3 Aktor memilih untuk melakukan *find* data mobil
5. Aktor memasukkan data mobil pada form yang disediakan sistem.
6. Aktor meminta sistem untuk menyimpan data mobil yang telah di masukkan.
7. Sistem mengecek data mobil yang telah di masukkan aktor.
 - E-1 Masukkan data mobil salah.
8. Data di simpan ke *database* oleh sistem
9. *Use Case* selesai.

5. Alternative Flow

none

6. Error Flow

E-1 Terdapat kolom yang tidak sesuai atau belum diisi.

1. Sistem memberikan peringatan bahwa terdapat masukkan yang salah.

Program Studi Teknik Informatika	SKPL – RENTCAR	32/ 37
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. Kembali ke *Basic Flow* 5.

7. PreConditions

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

8. PostConditions

1. Data mobil telah tersimpan ke dalam *database*.

4.1.11 Use case Spesification : Pengelolaan Data Transaksi

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data transaksi pemesanan mobil. Aktor dapat melakukan *update* status pemesanan dan *delete* data transaksi.

2. Primary Actor

1. Petugas

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk melakukan *update* status pemesanan mobil.
2. Sistem menampilkan antarmuka kelola transaksi pemesanan mobil.
3. Aktor memilih melakukan *update* status konfirmasi pembayaran.
A-1 Aktor memilih *delete* data transaksi pemesanan mobil
4. Aktor memilih data yang akan di *update* berdasarkan konfirmasi pembayaran.
5. Sistem menyimpan data transaksi pemesanan mobil ke *database*.
6. *Use Case* ini selesai.

Program Studi Teknik Informatika	SKPL – RENTCAR	33/ 37
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		

5. Alternative Flow

A-1 Aktor memilih *delete* data transaksi pemesanan mobil

1. Aktor data transaksi pemesanan yang akan dihapus.
2. Kembali ke *Basic Flow* langkah ke 4.

6. Error Flow

none

7. PreConditions

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

8. PostConditions

1. Data transaksi pemesanan mobil telah *ter-update*.

4.1.12 Use case Spesification : Email Web

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola *email*, pada *use case* ini aktor dapat melakukan *create new email*, *reply email*, dan *delete email*.

2. Primary Actor

1. Petugas

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk mengelola data *email*.
2. Sistem menampilkan antarmuka *email*.
3. Sistem memberikan pilihan untuk melakukan *create* atau *reply*.
4. Aktor memilih *create new email*.

Program Studi Teknik Informatika	SKPL – RENTCAR	34/ 37
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 *reply email*.

A-2 Aktor memilih untuk *delete email*.

5. Aktor memasukkan data *email*.

6. Sistem mengecek data *email* yang di masukkan.

E-1 Masukkan data *email* salah.

7. Sistem menyimpan data *email* ke *database*

8. *Use Case* selesai.

5. **Alternative Flow**

A-1 Aktor memilih *reply email*.

1. Sistem menampilkan antarmuka *reply email*.

2. Aktor memasukkan data *email*.

3. Berlanjut ke *Basic Flow* langkah ke 4.

A-2 Aktor memilih *delete email*.

1. Sistem menampilkan antarmuka *delete email*

2. Aktor menghapus data *email*.

3. Berlanjut ke *Basic Flow* langkah ke 4.

6. **Error Flow**

E-1 Data *email* yang dimasukkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa masukkan salah.

2. Kembali ke *Basic Flow* Langkah ke 5.

7. **PreConditions**

1. *Use Case* Login telah dilakukan.

2. Aktor telah memasuki sistem.

8. **PostConditions**

1. Data *email* telah ter-*update*.

4.1.13 Use case Spesification : Mencari Lokasi Mobil

1. **Brief Description**

Use Case ini digunakan oleh aktor untuk mencari lokasi mobil yang disewa pelanggan, pada *use case*

Program Studi Teknik Informatika	SKPL – RENTCAR	35/ 37
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		

ini aktor dapat mencari mobil untuk mengetahui lokasi mobil.

2. Primary Actor

1. Petugas

3. Supporting Actor

none

4. Basic Flow

1. *Use Case* ini dimulai ketika aktor memilih untuk mencari lokasi mobil yang sedang disewa.
2. Sistem menampilkan antarmuka *geolocation* mobil.
3. Sistem menampilkan *maps* atau peta.
4. Aktor menginputkan data mobil yang akan di cari.
5. Sistem mengecek dan mengambil data lokasi mobil yang sedang di pinjam.
6. Sistem menampilkan lokasi mobil atau *current locations*.
7. *Use Case* selesai.

5. Alternative Flow

none

6. Error Flow

E-1 Data mobil yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa masukkan salah.
2. Kembali ke *Basic Flow* Langkah ke 4.

7. PreConditions

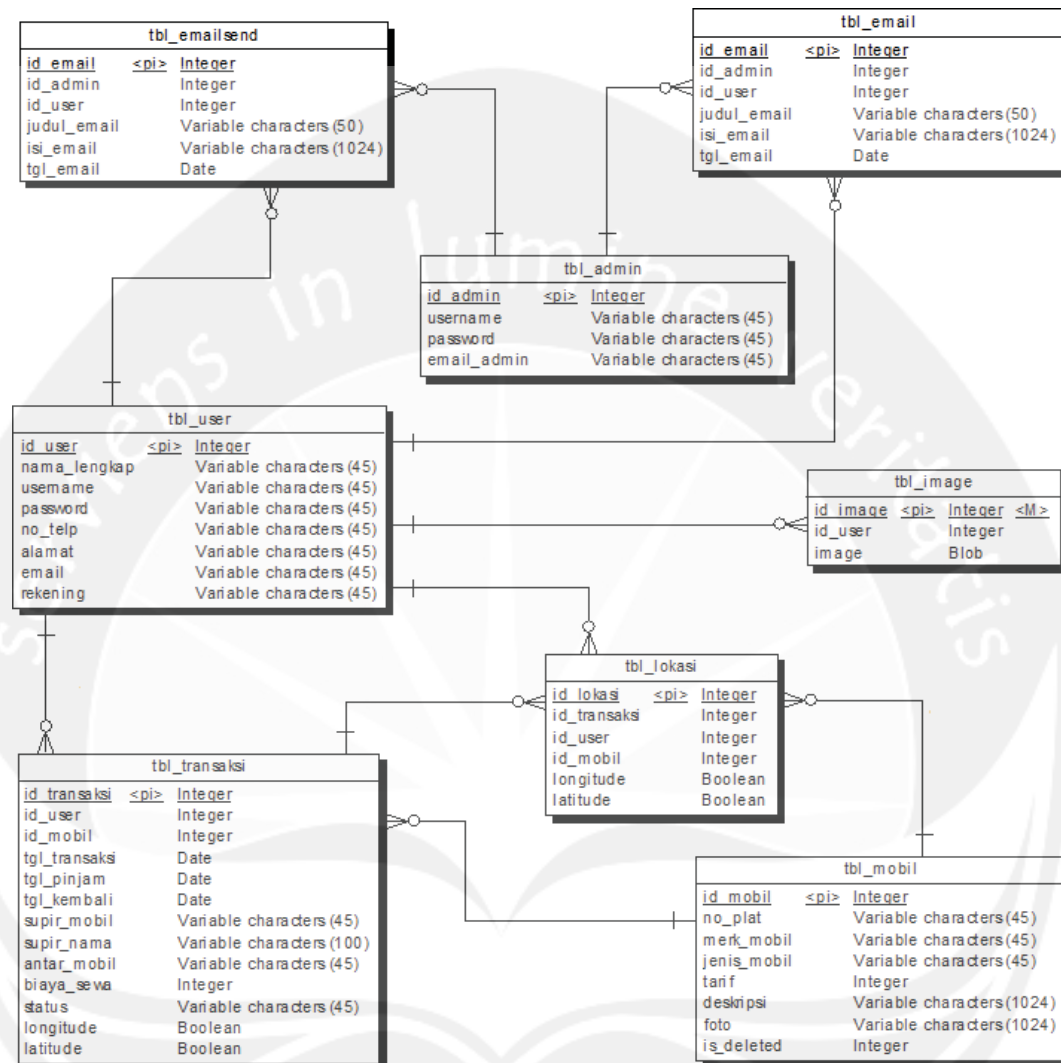
1. *Use Case* Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Aktor memonitoring mobil yang sedang disewa.

Program Studi Teknik Informatika	SKPL – RENTCAR	36/ 37
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		

5. ERD RENTCAR



Gambar 5.1 Entity Relationship Diagram RENTCAR