

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Setelah aplikasi pencarian tiket termurah dengan nama RESTITER ini selesai diimplementasikan dan berdasarkan dari pembahasan-pembahasan pada bab-bab sebelumnya, maka dapat ditarik beberapa kesimpulan, yaitu:

1. Perangkat Lunak RESTITER (*Reservasi Tiket Termurah*) telah berhasil merekomendasikan tiket termurah dari tiga situs maskapai penerbangan.
2. Perangkat Lunak RESTITER telah berhasil dikembangkan untuk melakukan pemesanan dan pembelian tiket secara efisien, efektif dan akurat.
3. Perangkat Lunak RESTITER telah berhasil dikembangkan menggunakan teknologi *J2ME (Java 2 Micro Edition)*.

6.2 Saran

Penulis ingin memberikan beberapa saran untuk pengembangan lebih lanjut perangkat lunak RESTITER :

1. Mengembangkan aplikasi agar dapat menambahkan lebih dari tiga maskapai penerbangan untuk pencarian tiket termurah.
2. Menambahkan fitur untuk transaksi biaya *reschedule* reservasi jika terjadi perubahan harga/*upgrade class*.
3. Mengembangkan perangkat lunak RESTITER agar dapat diimplementasikan untuk semua operasi sistem piranti *mobile*.

DAFTAR PUSTAKA

- Betha Sidik, Ir. 2001. *Pemrograman WEB dengan PHP*, Bandung:Penerbit Informatika.
- Burmistrov Ivan, 2009. *Mobile Air Ticket Booking*, Moscow State University, pp 10-11.
- Dreamteach Software Team, 2002. *Wireless Programing with J2ME*, New York: Hungry Minds, Inc.
- Microsystems Sun, 2009. *Java™ Platform Micro Edition Software Development Kit*, California: Sun Microsystems, Inc.
- Shalahuddin, M., dan Rosa A.S., 2008. *Pemrograman J2ME Belajar Cepat Pemrograman Perangkat Telekomunikasi Mobile*, Bandung:Informatika.
- Sugiyono, Prof.Dr. 2011. *Metode Penelitian Kuantatif dan Kualitatif dan R&D*, Bandung:Alfabeta.
- Sutarman, S.Kom. 2003. *Membangun Aplikasi Web dengan PHP dan MySQL*, Yogyakarta:Graha Ilmu.
- Wahana Komputer, 2010. *Tutorial 5 Hari Membangun GUI dengan Java Netbeans 6.5*. Yogyakarta:Andi Offset.

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

RESTITER

Reservasi Tiket Termurah

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Pirhotlan Merson Purba / 070705482

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL-RESTITER		1/25
		Revisi		

Program Studi Teknik Informatika	SKPL – RESTITER	1/ 25
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								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1	Pendahuluan	6
1.1	Tujuan	6
1.2	Lingkup Masalah	6
1.3	Definisi, Akronim dan Singkatan	6
1.4	Referensi	7
1.5	Deskripsi umum (Overview)	7
2	Deskripsi Kebutuhan	8
2.1	Perspektif produk	8
2.2	Fungsi Produk	9
2.3	Karakteristik User	12
2.4	Batasan-batasan	12
2.5	Asumsi dan Ketergantungan	12
3	Kebutuhan khusus	13
3.1	Kebutuhan antarmuka eksternal	13
3.2	Kebutuhan fungsionalitas Perangkat Lunak.	14
4	Spesifikasi Rinci Kebutuhan	16
4.1	Spesifikasi Kebutuhan Fungsionalitas	16
5	Entity Relationship Diagram (ERD)	25

Daftar Gambar

1. Arsitektur Perangkat Lunak RESTITER	9
2. Use Case Diagram	14
3. ERD	25

1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak RESTITER (*Reservasi Tiket Termurah*) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka *external* (antarmuka antara sistem dengan perangkat lunak, perangkat keras, dan user), dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-RESTITER ini juga mendefinisikan batasan-batasan dari perancangan perangkat lunak.

1.2 Lingkup Masalah

Adapun batasan sistem adalah sesuai dengan lingkup tujuan pembuatan perangkat lunak RESTITER yaitu:

1. Menangani pencarian tiket termurah pada saat itu (*real-time*)
2. Menangani pemesanan tiket pesawat
3. Menangani pembatalan reservasi tiket pesawat
4. Menampilkan histori reservasi tiket.

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-RESTITER	Kode yang merepresentasikan kebutuhan pada RESTITER (<i>Reservasi Tiket Termurah</i>)

RESTITER	Aplikasi <i>mobile</i> untuk mencari harga tiket termurah dari berbagai situs maskapai penerbangan serta melakukan reservasi tiket.
J2ME	Java 2 Micro Edition Teknologi yang digunakan untuk mengembangkan aplikasi RESTITER.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Kusuma Swedasse Artita, *Pembangunan Sistem Informasi Mobile Ticketing pada Pemesanan Tiket Bioskop*, Universitas Atma Jaya Yogyakarta, 2010.
2. Purba Pirhotlan Merson, *Spesifikasi Kebutuhan Perangkat Lunak ATMAVision*, Universitas Atma Jaya Yogyakarta, 2010.
3. Siadari Lucky Prima Lestari, *Pembangunan Aplikasi Sistem Reservasi Tiket Bus JOGLOSEMAR Berbasis J2ME*, Universitas Atma Jaya Yogyakarta, 2010.

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 5 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL,

Program Studi Teknik Informatika	SKPL – RESTITER	7/ 25
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		

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 RESTITER yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik user, batasan dalam useran perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak RESTITER tersebut.

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

Bagian keempat berisi penjelasan tentang spesifikasi rinci kebutuhan dari tiap use case yang ada pada use case diagram pada bagian ketiga.

Bagian kelima berisi gambaran ERD (*Entity Relationship Diagram*) yang digunakan dalam mengembangkan perangkat lunak RESTITER.

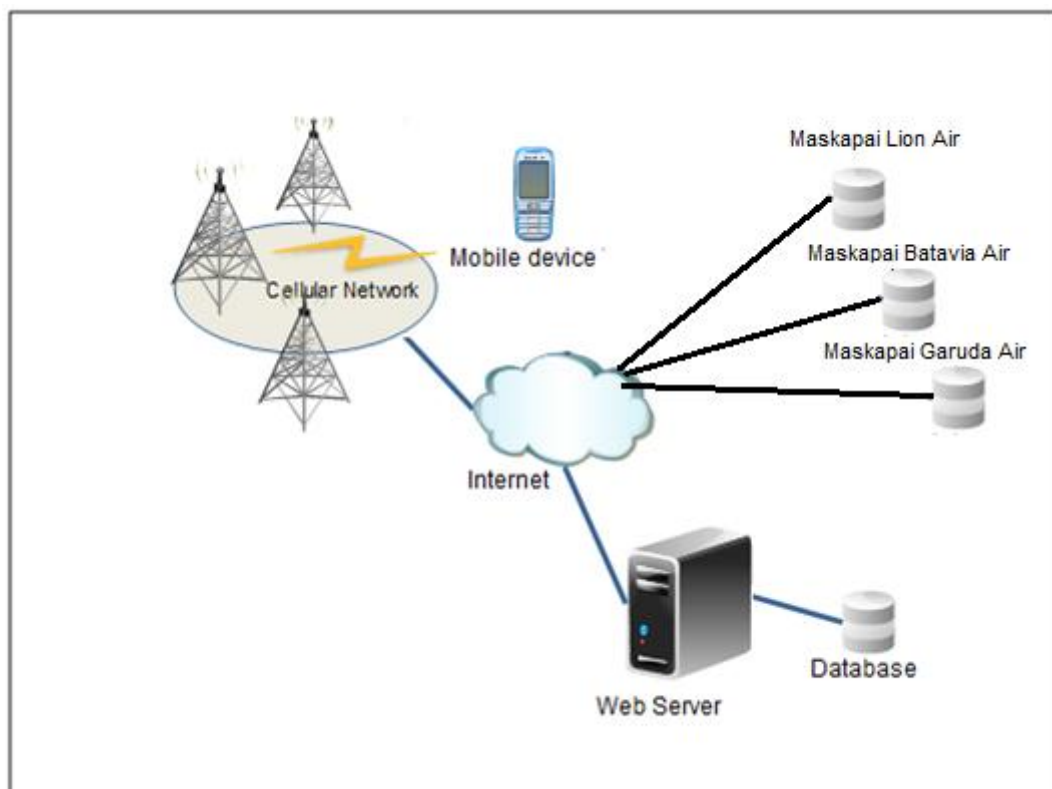
2 Deskripsi Kebutuhan

2.1 Perspektif produk

Aplikasi *Reservasi Tiket Termurah* (RESTITER) merupakan perangkat lunak atau aplikasi yang dikembangkan untuk mencari tiket termurah dari harga tiket yang disediakan oleh beberapa maskapai penerbangan. Aplikasi dapat melakukan pemesanan tiket melalui telepon selular. Reservasi keberangkatan penumpang akan disimpan dan user dapat mengecek histori reservasi pemesanan tersebut.

Aplikasi *RESTITER* ini berjalan pada platform yang mendukung teknologi java untuk perangkat mobile dan dibuat menggunakan bahasa pemrograman J2ME dengan lingkungan emulator menggunakan Netbeans 6.9.1.

User akan berinteraksi dengan sistem melalui antarmuka GUI (*Graphical User Interface*) pada perangkat *mobile*. Pada sistem ini, seperti terlihat pada gambar 1, aplikasi ini bersifat *client-server*, yaitu user mengakses data yang terdapat pada web server dan masukan data disimpan dalam database.



Gambar 1. Arsitektur Perangkat lunak RESTITER

2.2 Fungsi Produk

Adapun fungsionalitas yang terdapat pada *RESTITER* ini adalah sebagai berikut:

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

Program Studi Teknik Informatika	SKPL – RESTITER	9/ 25
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 oleh member agar dapat masuk ke dalam sistem RESTITER.

2. Fungsi *Sign Up* (**SKPL-RESTITER-002**)

Merupakan fungsi yang digunakan oleh calon member untuk melakukan pendaftaran menjadi member.

3. Fungsi *Search Flight Schedule* (**SKPL-RESTITER-003**)

Merupakan fungsi yang digunakan untuk membantu member untuk mencari jadwal keberangkatan pesawat. Adapun fungsionalitas *search flight schedule* adalah sebagai berikut:

a. Fungsi *cari tiket termurah class bisnis* (**SKPL-RESTITER-003-01**)

Merupakan fungsi yang digunakan oleh member untuk mencari jadwal penerbangan untuk class bisnis dengan harga tiket yang disortir mulai dari harga termurah sampai yang harga yang termahal.

b. Fungsi *cari tiket termurah class ekonomi* (**SKPL-RESTITER-003-02**)

Merupakan fungsi yang digunakan oleh member untuk mencari jadwal penerbangan untuk class ekonomi dengan harga tiket yang disortir mulai dari harga termurah sampai yang harga yang termahal.

4. Fungsi *Kelola Reservasi* (**SKPL-RESTITER-004**)

Merupakan fungsi yang digunakan oleh member untuk mengelolah reservasi tiket pesawat. Adapun fungsionalitas *kelola reservasi* adalah:

a. Fungsi *Reservasi Tiket* (**SKPL-RESTITER-004-01**)

Program Studi Teknik Informatika	SKPL – RESTITER	10/ 25
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 oleh member untuk melakukan reservasi tiket pesawat.

b. Fungsi *Batal Reservasi Tiket* **(SKPL-RESTITER-004-02)**

Merupakan fungsi yang digunakan oleh member untuk melakukan pembatalan reservasi tiket

5. Fungsi *Display History Reservasi* **(SKPL-RESTITER-005)**

Merupakan fungsi yang dapat digunakan oleh member untuk menampilkan histori semua reservasi.

5. Fungsi *Pembayaran* **(SKPL-RESTITER-006)**

Merupakan fungsi yang dapat digunakan oleh member untuk melakukan pembayaran melalui ATM atau Credit Card.

6. Fungsi *Login Web* **(SKPL-RESTITER-007)**

Merupakan fungsi yang digunakan oleh operator untuk dapat masuk kedalam sistem web.

7. Fungsi *Kelola Member* **(SKPL-RESTITER-008)**

Merupakan fungsi yang digunakan oleh operator untuk mengelola member. Adapun fungsionalitas kelola member adalah:

a. Fungsi *Approve Member* **(SKPL-RESTITER-008-01)**

Merupakan fungsi yang digunakan oleh operator untuk menambah member.

b. Fungsi *Delete Member* **(SKPL-RESTITER-008-02)**

Merupakan fungsi yang digunakan oleh operator untuk menghapus data member.

2.3 Karakteristik User

User perangkat lunak RESTITER terdiri dari:

1. User

Karakteristik User sebagai berikut:

- a. Mampu mengoperasikan telepon selular dengan sistem operasi yang mendukung teknologi java.

2. Operator

Karakteristik Operator sebagai berikut:

- a. Memahami pengoperasian telepon selular dengan sistem operasi yang mendukung teknologi java.
- b. Memahami pengelolaan data pada web menggunakan PHP dan MySql.

2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak RESTITER tersebut adalah:

1. Kebijakan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak RESTITER.

2. Keterbatasan Perangkat Lunak

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

2.5 Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada perangkat mobile (telepon selular) yang menggunakan sistem operasi yang mendukung teknologi java.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak RESTITER meliputi kebutuhan antarmuka pemakai, antarmuka perangkat mobile, dan antarmuka perangkat lunak.

3.1.1 Antarmuka pemakai

User berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form yang ada pada layar telepon selular.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak RESTITER adalah:

1. Perangkat mobile (telepon selular).
2. PC

3.1.3 Antarmuka perangkat lunak

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

1. Nama : MIDP 2.0
Sumber : Sun Microsystem
Sebagai platform environment pada perangkat mobile.
2. Nama : Mysql
Sumber : Mysql
Sebagai *Data Base Management System* (DBMS).
3. Nama : Apache
Sumber : Apache
Sebagai web server local

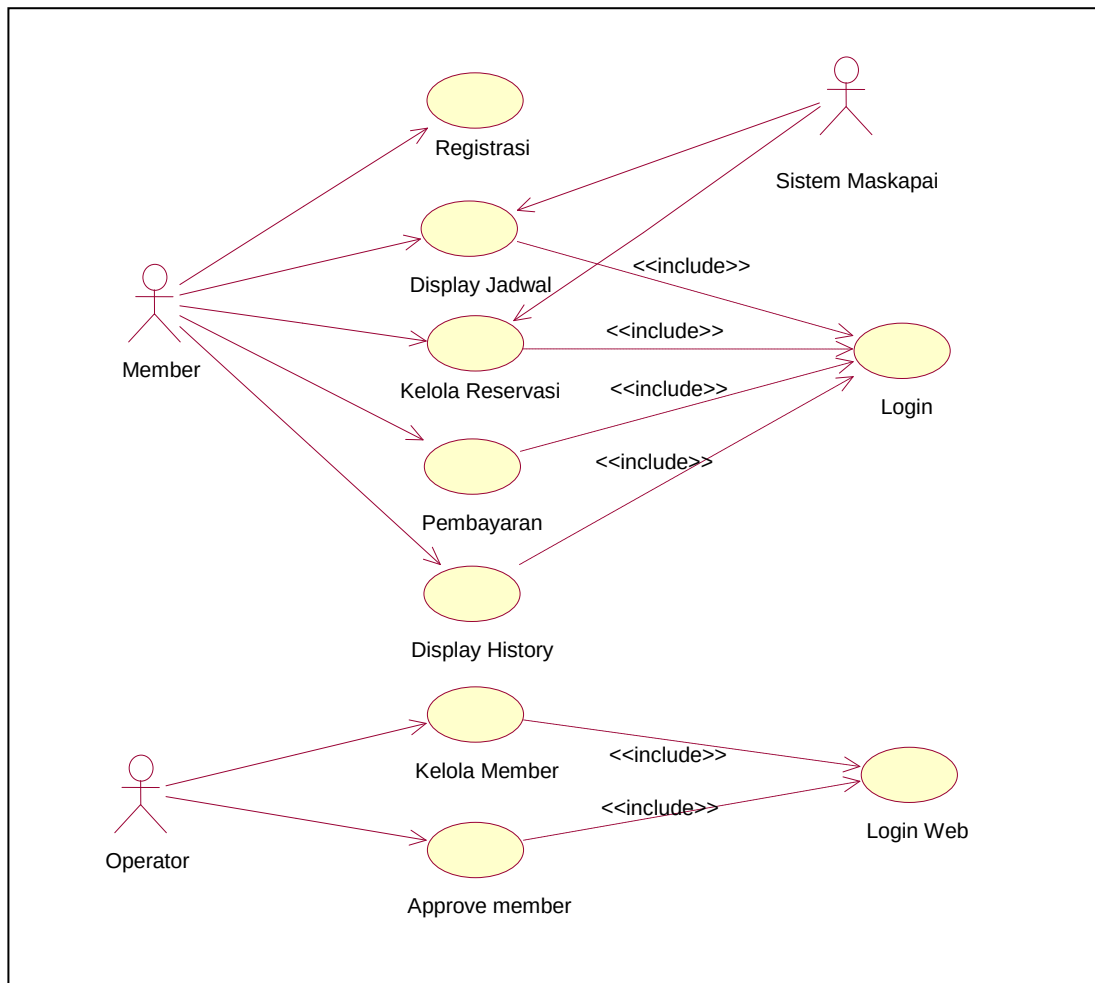
4. Nama : Netbeans 6.9.1
Sumber : Sun Microsystem
Sebagai platform environment pada perangkat mobile.
5. Nama : Windows 7
Sumber : Microsoft
Sebagai system operasi yang digunakan.

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak RESTITER menggunakan protokol http.

3.2 Kebutuhan fungsionalitas Perangkat Lunak.

3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification : *Login*

1. Brief Description

Use Case ini digunakan oleh member untuk memperoleh akses ke sistem. *Login* didasarkan pada sebuah id unik yaitu username dan password dari user.

2. Primary Actor

1. Member

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 Password atau username user tidak sesuai.
5. Sistem memberikan akses ke aktor.
6. Use Case ini selesai.

5. Alternative Flow

none.

6. Error Flow

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

7. PreConditions

none

Program Studi Teknik Informatika	SKPL – RESTITER	16/ 25
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		

8. PostConditions

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

4.1.2 Use case Spesification: *Sign Up*

1. Brief Description

Use Case ini memungkinkan user untuk melakukan pendaftaran untuk menjadi member.

2. Primary Actor

1. Member.

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pendaftaran menjadi member.
2. Sistem menampilkan *form input* data member.
3. Aktor memasukkan data member.
4. Aktor meminta sistem untuk menyimpan data user.
5. Sistem mengecek data user yang telah diinputkan.
E-1 Data username yang diinputkan salah.
6. Sistem menyimpan data user ke database.
7. Use Case selesai.

5. Alternative Flow

None.

6. Error Flow

E-1 Data user yang diinputkan aktor salah.

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan oleh aktor salah.

Program Studi Teknik Informatika	SKPL – RESTITER	17/ 25
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 Langkah ke 3.

7. PreConditions

1. Aktor pertama kali memasuki sistem.

8. PostConditions

1. Aktor telah memasukkan data user yang diperlukan dan data user tersimpan dalam database.

4.1.3 Use case Spesification: *Search Flight Schedule*

1. Brief Description

Use Case ini digunakan untuk menampilkan jadwal penerbangan.

2. Primary Actor

1. Member

3. Supporting Actor

None.

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk mencari harga tiket termurah.
2. Sistem menampilkan *form* untuk memilih rute, kategori class dan tanggal keberangkatan.
3. User memilih rute, tanggal keberangkatan dan kategori class yang diinginkan.
4. Sistem menampilkan informasi jadwal keberangkatan pesawat.

E-1 Sistem meminta untuk memasukkan kembali Data pencarian.

5. Use Case ini selesai

5. Alternative Flow

none

Program Studi Teknik Informatika	SKPL – RESTITER	18/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Error Flow

E-1 rute yang diinginkan dari user tidak ada

1. Sistem menampilkan peringatan bahwa tidak ada rute yang diinginkan oleh user.
2. Kembali ke Basic Flow langkah ke 3.

7. PreConditions

none

8. PostConditions

1. Informasi harga tiket termurah ditampilkan.

4.1.4 Use case Spesification: *Kelola Reservasi*

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data reservasi. Aktor dapat menambahkan reservasi, cari reservasi dan cancel reservasi.

2. Primary Actor

1. Member

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor melakukan pengelolaan reservasi.
2. Sistem memberikan pilihan untuk melakukan penambahan, pembatalan atau cari reservasi.
3. Aktor memilih untuk melakukan penambahan reservasi.
4. Sistem meminta aktor menginput data penumpang
5. Aktor memasukkan data penumpang
6. Sistem melakukan pengecekan terhadap data yang diinputkan

E-1 Data yang diinput salah

Program Studi Teknik Informatika	SKPL – RESTITER	19/ 25
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. Aktor meminta sistem menyimpan data reservasi penumpang kedalam basis data
8. Sistem menyimpan reservasi
9. Use case selesai

5. Alternative Flow

A-1 Aktor memilih untuk melakukan search reservasi

1. Sistem meminta aktor memasukkan kode booking
2. Aktor memasukkan kode booking reservasi
3. Sistem melakukan pengecekan terhadap kode booking

E-1 kode booking tidak ditemukan

4. Sistem menampilkan reservasi sesuai kode booking.
5. Berlanjut ke Basic Flow pad langkah ke-9

A-2 Aktor memilih untuk melakukan pembatalan reservasi

1. Sistem meminta aktor memasukkan data pembatal reservasi.
2. Aktor memasukkan data pembatal reservasi
3. Sistem melakukan pengecekan terhadap data pembatal
4. Sistem mengubah data reservasi
5. Berlanjut ke Basic Flow pad langkah ke-9

6. Error Flow

E-1 kode booking tidak ditemukan

1. Sistem memberikan peringatan bahwa kode booking yang diinput tidak ditemukan
2. Kembali ke Alternatif Flow A-1 langkah ke-1

7. PreConditions

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

Program Studi Teknik Informatika	SKPL – RESTITER	20/ 25
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		

8. PostConditions

1. Reservasi penumpang terupdate pada database.

4.1.5 Use case Spesification: *Pembayaran*

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pembayaran tiket dengan meminta sistem untuk membuat kode pembayaran ATM atau membayar melalui Credit Card.

2. Primary Actor

1. User

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pembayaran tiket.
2. Aktor memilih untuk melakukan pembayaran tiket
 - A-1 Aktor memilih pembayaran melalui ATM
 - A-2 Aktor memilih pembayaran melalui Credit Card
3. Sistem melakukan pengecekan terhadap basis data
4. Sistem menambahkan data pembayaran kedalam database
5. Use Case selesai

5. Alternative Flow

A-1Aktor memilih pembayaran melalui ATM

1. Sistem mencetak kode pembayaran tiket
2. Kembali ke Basic Flow langkah 3.

A-1 Aktor memilih pembayaran melalui Credit Card

Program Studi Teknik Informatika	SKPL – RESTITER	21/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem meminta aktor untuk memasukkan data Credit Card
2. Aktor memasukkan data Credit Card.
3. Kembali Basic Flow langkah ke-3.

E-1 Data Credit Card tidak berlaku

6. Error Flow

E-1 Data yang dimasukkan tidak berlaku

1. Sistem memberikan peringatan bahwa data credit card tidak berlaku.
2. Kembali ke Alternative Flow langkah ke-2

7. PreConditions

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

8. PostConditions

1. Sistem menampilkan kode pembayaran melalui ATM atau sistem memasukkan data pembayaran Credit Card ke dalam basis data.

4.1.6 Use case Spesification : *Login Web*

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem web. *Login* didasarkan pada username dan password dari admin.

2. Primary Actor

2. Operator

3. Supporting Actor

None

4. Basic Flow

3. Use Case ini dimulai ketika aktor memilih untuk melakukan *login*.

Program Studi Teknik Informatika	SKPL – RESTITER	22/ 25
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. Sistem menampilkan antarmuka untuk *login*.
5. Aktor memasukkan username dan password.
6. Sistem memeriksa username dan password yang diinputkan aktor.
E-1 Password atau username user tidak sesuai.
7. Sistem memberikan akses ke aktor.
8. Use Case ini selesai.

5. Alternative Flow

none.

6. Error Flow

E-1 Password atau username 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

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

4.1.7 Use case Spesification : Approve Member

1. Brief Description

Use Case ini digunakan oleh aktor untuk menambah member.

2. Primary Actor

1. Operator

3. Supporting Actor

none

4. Basic Flow

Program Studi Teknik Informatika	SKPL – RESTITER	23/ 25
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use Case ini dimulai ketika aktor memilih untuk melakukan approve member.
2. Sistem menampilkan seluruh data calon member.
3. Aktor memilih data calon member yang akan dihapus.
4. Sistem melakukan pengecekan terhadap data yang dihapus
5. Sistem mengupdate data member
6. Use case selesai

5. Alternative Flow

6. Error Flow

7. PreConditions

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

8. PostConditions

Data member terupdate

4.1.8 Use case Spesification : Approve Member

1. Brief Description

Use Case ini digunakan oleh aktor untuk menghapus data member.

2. Primary Actor

1. Operator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan delete member.
2. Sistem menampilkan seluruh data member.
3. Aktor memilih data calon member yang akan dihapus.

Program Studi Teknik Informatika	SKPL – RESTITER	24/ 25
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. Sistem melakukan pengecekan terhadap data yang dihapus
5. Sistem menghapus data member
6. Use case selesai

5. Alternative Flow

6. Error Flow

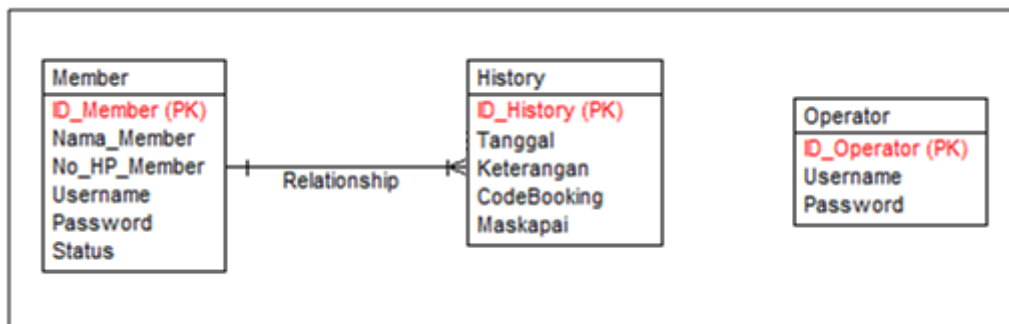
7. PreConditions

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

8. PostConditions

Data member dalam basis data terhapus.

5. Entity Relationship Diagram (ERD)



Gambar 3. ERD

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

RESTITER

Reservasi Tiket Termurah

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Pirhotlan Merson Purba / 070705482

Program Studi Teknik Informatika – Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL-RESTITER		1/32

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

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

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1	Pendahuluan.....	7
1.1	Tujuan.....	7
1.2	Ruang Lingkup.....	7
1.3	Definisi dan Akronim.....	7
1.4	Referensi.....	8
2	Perancangan Sistem.....	9
2.1	Perancangan Arsitektur.....	9
2.2	Perancangan Rinci.....	10
2.2.1	Sequence Diagram	10
2.2.1.1	Login	10
2.2.1.2	Registrasi	11
2.2.1.3	Search Schedule	11
2.2.1.4	Book Flight	12
2.2.1.3	Search Schedule	12
2.2.1.6	Cancel Keberangkatan	13
2.2.1.7	Display Histori	13
2.2.2	Class Diagram	14
2.2.3	Spesifikasi Deskripsi Kelas Diagram ...	15
2.2.3.1	Spesifikasi Design Class RestiterMain .	16
2.2.3.2	Spesifikasi Design Class ControlLogin .	16
2.2.3.3	Spesifikasi Design Class ControlRegistrasi	16
2.2.3.4	Spesifikasi Design Class ControlSearchSchedule.....	17
2.2.3.5	Spesifikasi Design Class ControlMyBooking	17
2.2.3.6	Spesifikasi Design Class ControlPembayaran.....	17
2.2.3.7	Spesifikasi Design Class ControlHistori	18
2.2.3.6	Spesifikasi Design Kelas Member.....	18
2.2.3.7	Spesifikasi Design Kelas FlightSchedule	19
2.2.3.8	Spesifikasi Design Kelas Reservasi....	19
2.2.3.9	Spesifikasi Design Kelas Penumpang....	20
3	Perancangan Data.....	21
3.1	Dekomposisi Data.....	21
3.1.1	Deskripsi Entitas Data Member	21
3.1.2	Deskripsi Entitas Data Operator	21
3.1.3	Deskripsi Entitas Data Histori	22
3.1.4	Physical Data Model	23
4	Perancangan Antarmuka.....	24
4.1	Sketsa Antarmuka dan Deskripsinya.....	24
4.1.1	Antarmuka Login	24
4.1.2	Antarmuka Menu	25

4.1.3	Antarmuka Registrasi	25
4.1.4	Antarmuka Search Flight Schedule	26
4.1.5	Antarmuka List Flight Schedule	26
4.1.6	Antarmuka Book Flight	27
4.1.7	Antarmuka Pembayaran	24
4.1.8	Antarmuka Display Reservasi	24
4.1.9	Antarmuka Cancel Reservasi	24
4.1.10	Antarmuka Search Reservasi	29
4.1.11	Antarmuka Login Operator	30
4.1.12	Antarmuka Approve Member	31
4.1.13	Antarmuka Delete Member	32

Daftar Gambar

Gambar 2.1	Rancangan Arsitektur RESTITER	9
Gambar 2.2	Rancangan Arsitektur RESTITER	10
Gambar 2.3	Sequence Diagram : Login	11
Gambar 2.4	Sequence Diagram : Registrasi	11
Gambar 2.5	Sequence Diagram : Cari Tiket Termurah	12
Gambar 2.6	Sequence Diagram : Pesan Tiket	12
Gambar 2.7	Sequence Diagram : Search Reservasi	13
Gambar 2.8	Class Diagram RESTITER	14
Gambar 3.1	Physical Data Model	23
Gambar 4.1	Rancangan Antarmuka Login	24
Gambar 4.2	Rancangan Antarmuka Menu	24
Gambar 4.3	Rancangan Antarmuka Registrasi	25
Gambar 4.4	Rancangan Antarmuka Search Flight Schedule	25
Gambar 4.5	Rancangan Antarmuka List Flight Schedule	26
Gambar 4.6	Rancangan Antarmuka Book Flight	26
Gambar 4.7	Rancangan Antarmuka Pembayaran	27
Gambar 4.8	Rancangan Antarmuka Display Reservasi	28
Gambar 4.9	Rancangan Antarmuka Cancel Reservasi	29
Gambar 4.10	Rancangan Antarmuka Search Reservasi	29
Gambar 4.11	Rancangan Antarmuka Login Operator	30
Gambar 4.12	Rancangan Antarmuka Approve Member	31
Gambar 4.13	Rancangan Antarmuka Delete Member	32

1 Pendahuluan

1.1 Tujuan

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

1.2 Ruang Lingkup

Perangkat Lunak RESTITER dikembangkan dengan tujuan untuk :

1. Pencarian tiket termurah pada saat itu (*real-time*).
2. Menangani pemesanan tiket.
3. Berjalan pada lingkungan dengan platform teknologi Java.

1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-RESTITER	Kode yang merepresentasikan kebutuhan pada RESTITER (RESERVASI TIKET TERMURAH)
RESTITER	Aplikasi <i>mobile</i> untuk mencari tiket termurah.
J2ME	Sebuah teknologi yang dirancang untuk pengembangan Perangkat Lunak.
Internet	Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang

	terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.
--	--

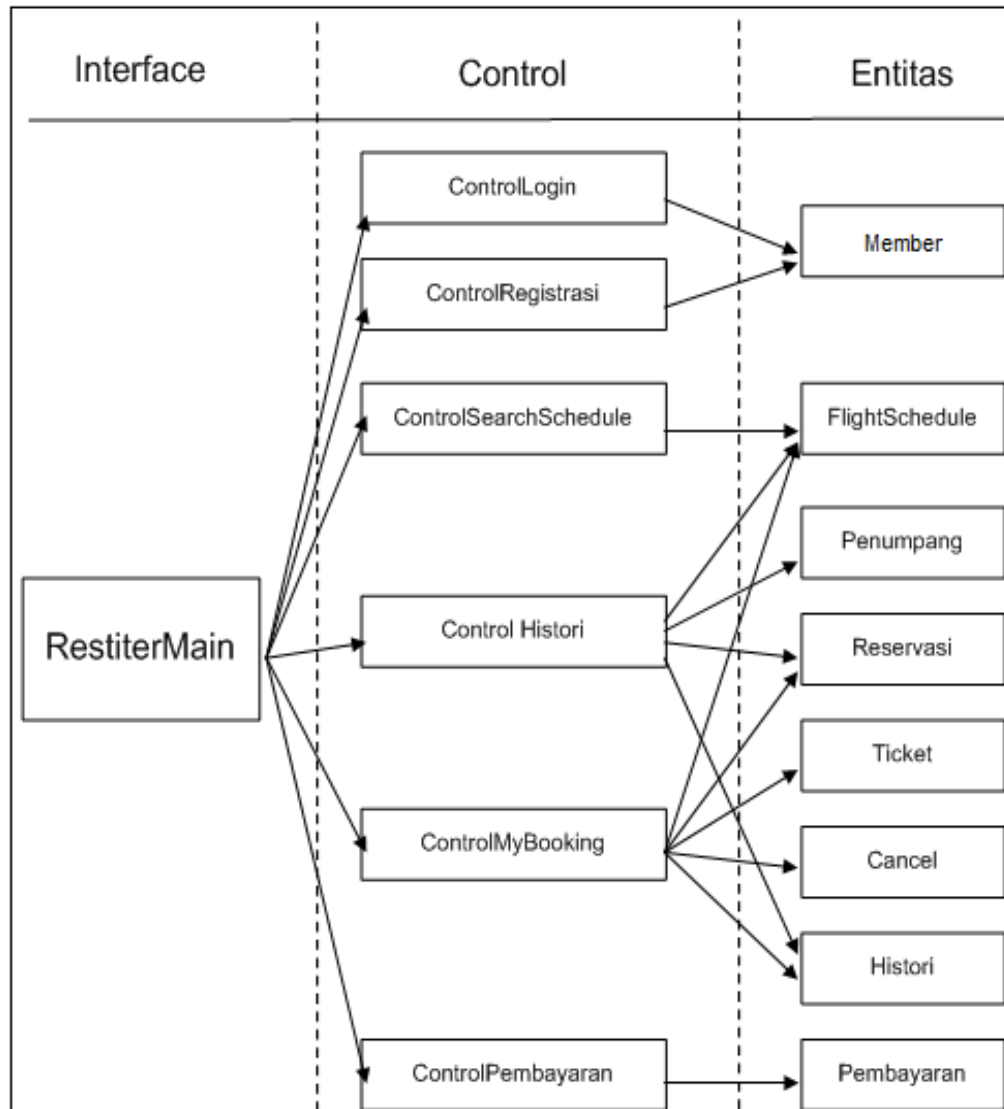
1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Sapta, Juli. *Deskripsi Perancangan Perangkat Lunak SC3*, Universitas Atma Jaya Yogyakarta, 2006.

2 Perancangan Sistem

2.1 Perancangan Arsitektur

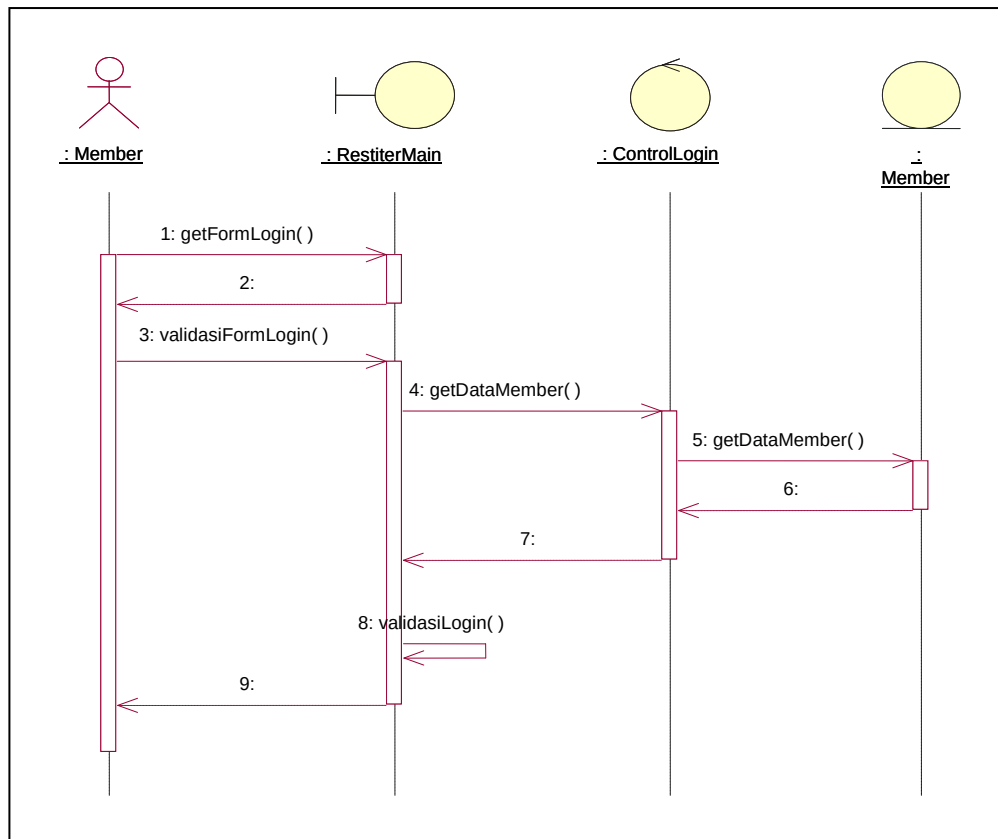


Gambar 2.1 Rancangan Arsitektur RESTITER

2.2 Perancangan Rinci

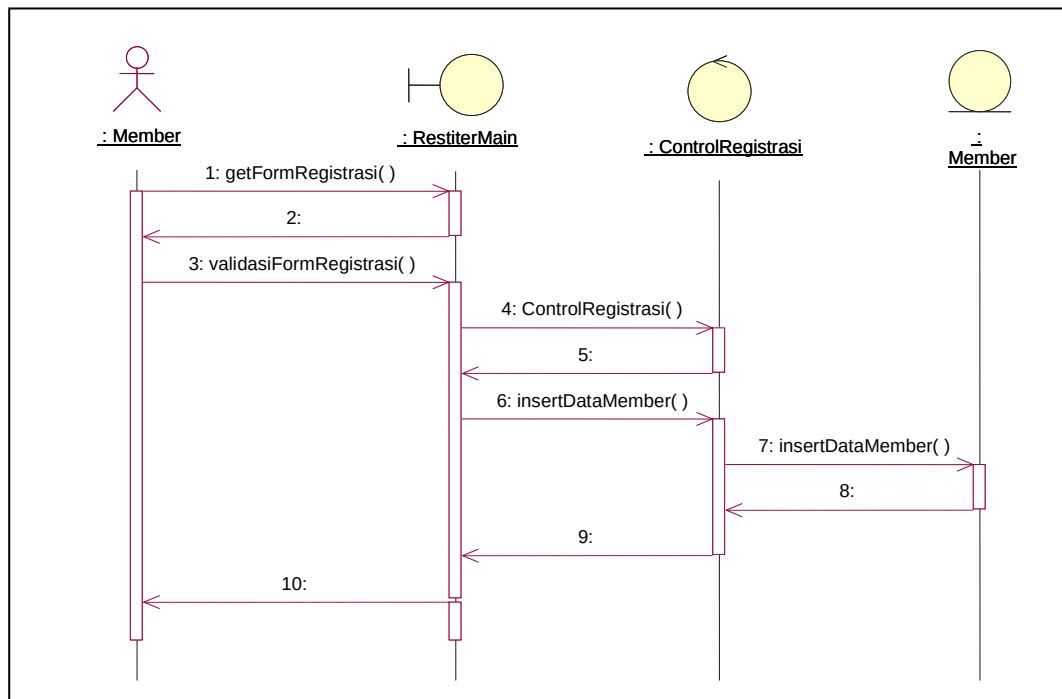
2.2.1 Sequence Diagram

2.2.1.1 Login



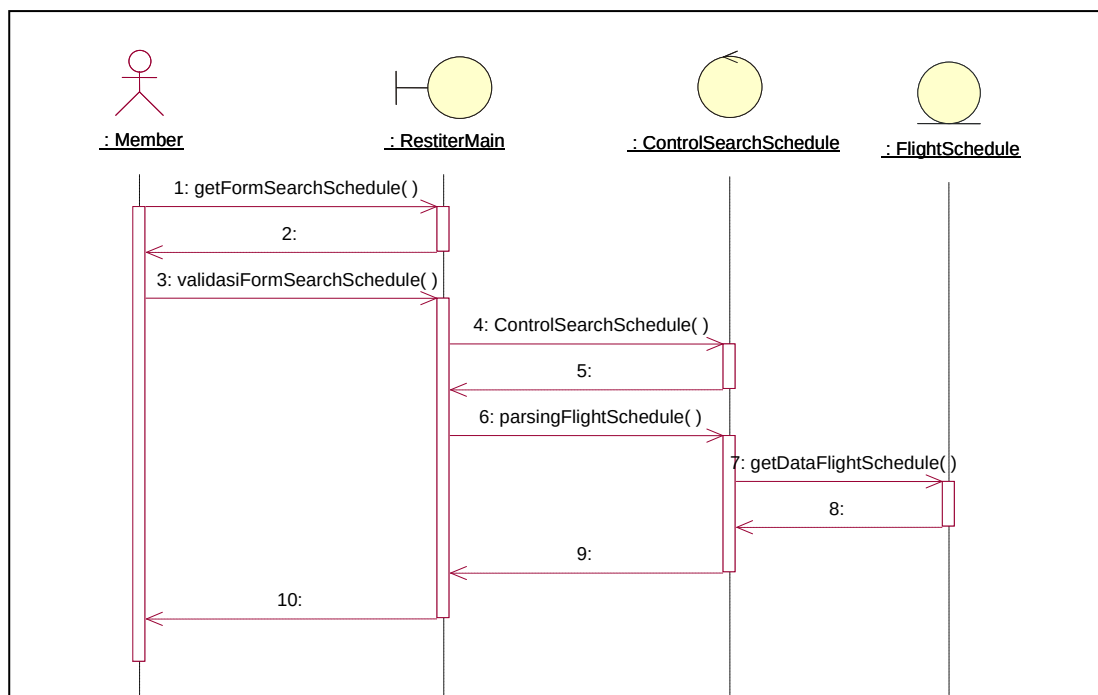
Gambar 2.3 Sequence Diagram : Login

2.2.1.2 Registrasi



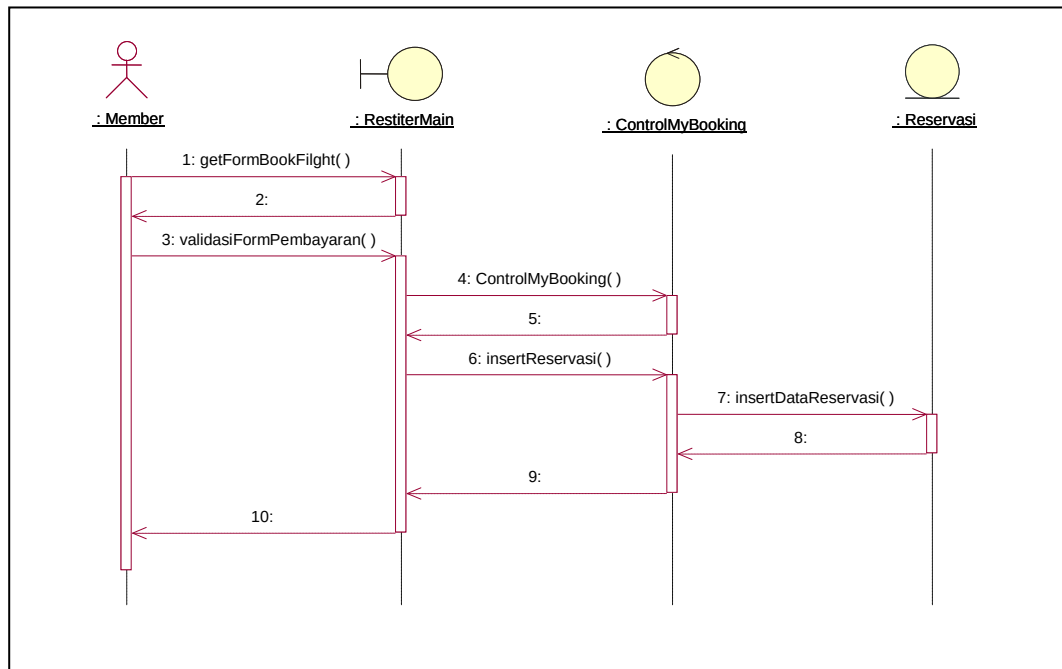
Gambar 2.4 Sequence Diagram : Registrasi

2.2.1.3 Search Flight Schedule



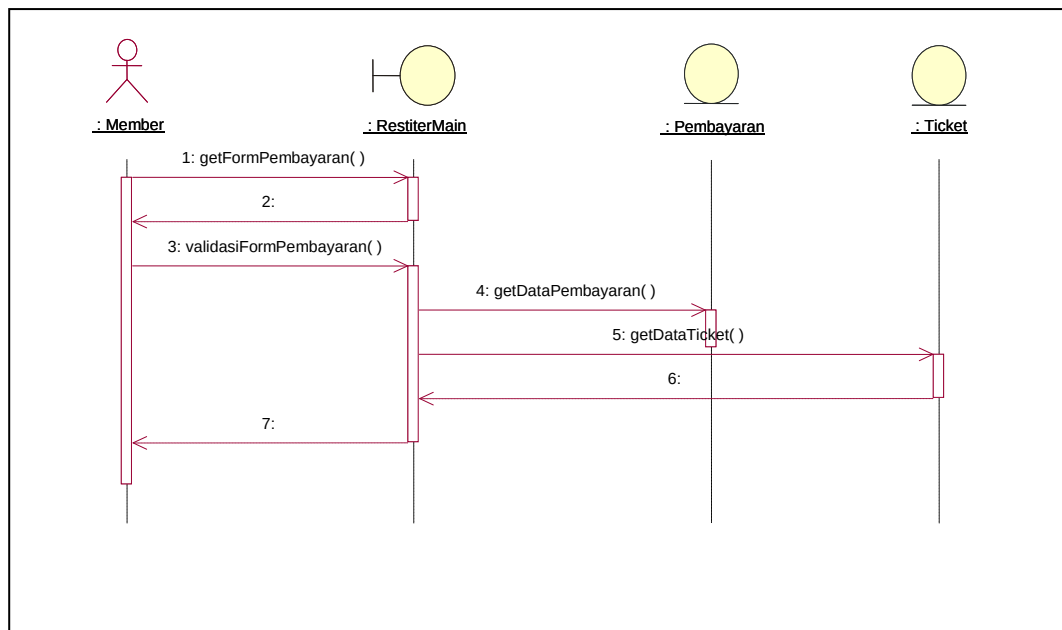
Gambar 2.5 Sequence Diagram : Search Flight Schedule

2.2.1.4 *Book Flight*



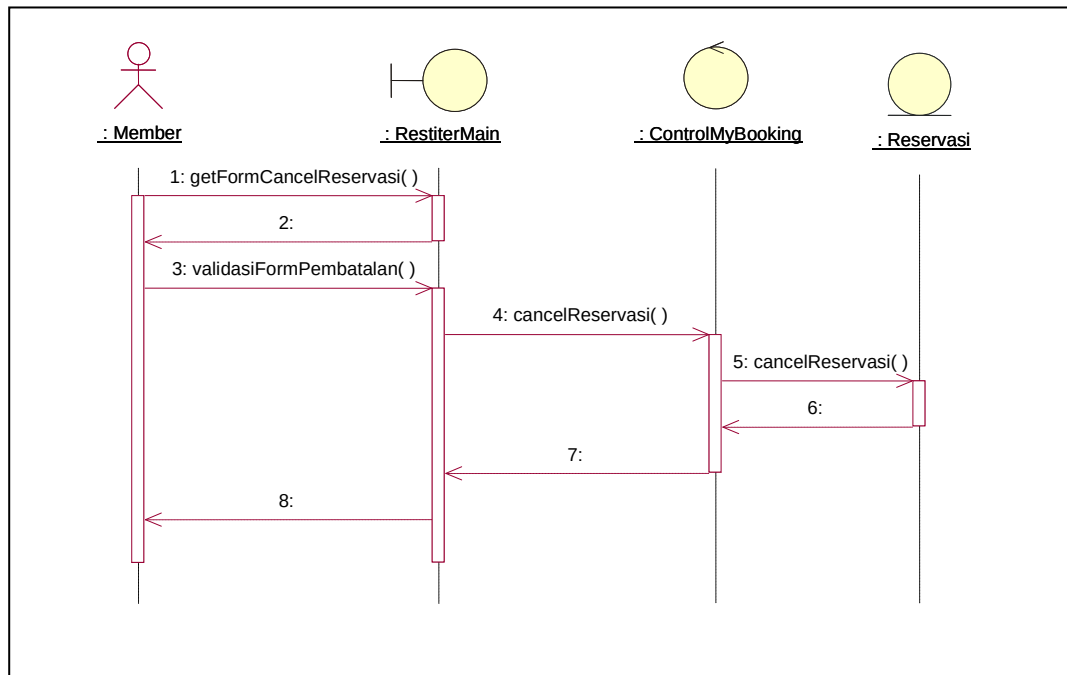
Gambar 2.6 Sequence Diagram : *Book Flight*

2.2.1.5 *Pembayaran*



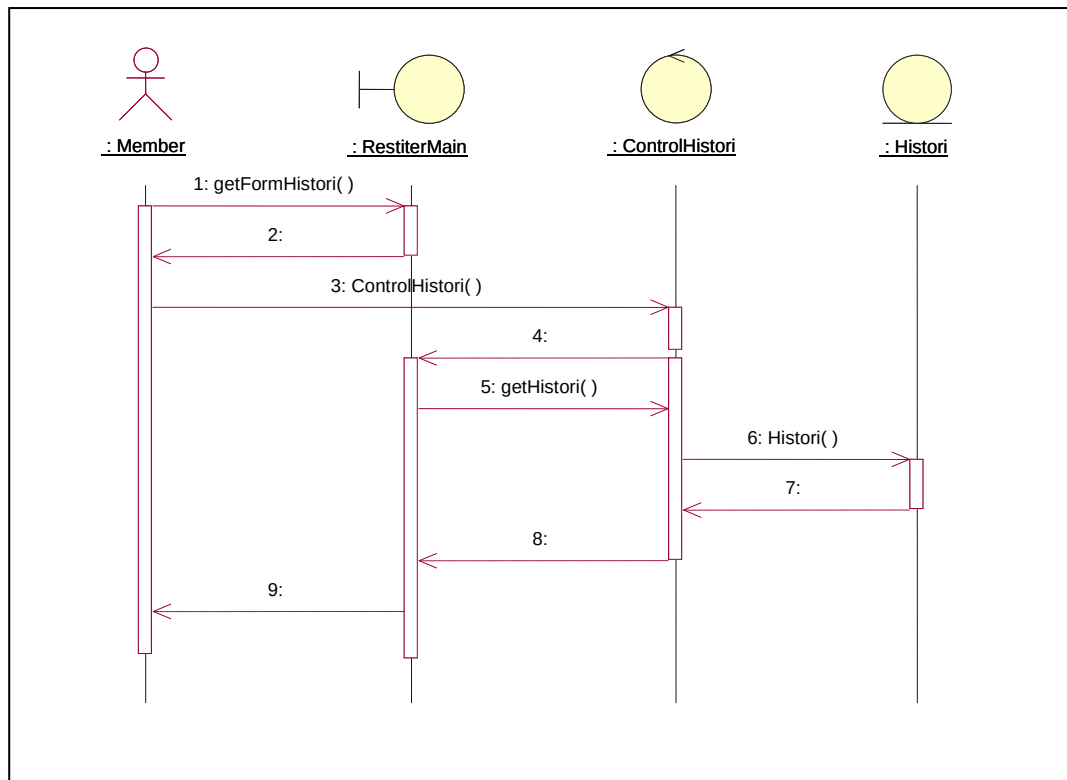
Gambar 2.7 Sequence Diagram : *Pembayaran*

2.2.1.6 Cancel Keberangkatan



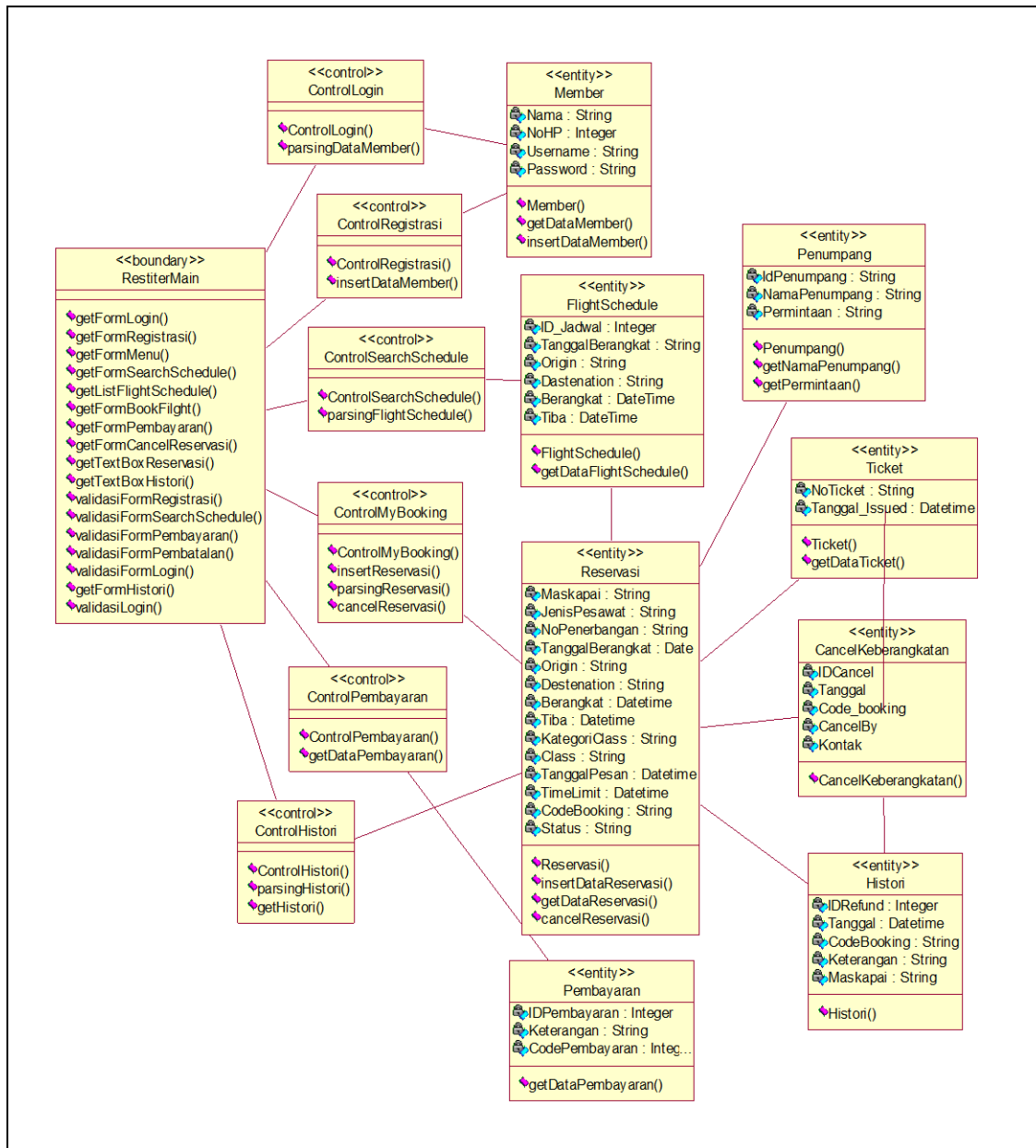
Gambar 2.7 Sequence Diagram : *Cancel Keberangkatan*

2.2.1.7 Display Histori



Gambar 2.7 Sequence Diagram : *Display Histori*

2.2.2 Class Diagram



Gambar 2.8 Class Diagram RESTITER

2.2.3 Spesifikasi Deskripsi Class Diagram

2.2.3.1 Spesifikasi Design Class RestiterMain

RestiterMain	<<boundary>>
<pre>+ RestiterMain() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + getFormLogin Operasi ini digunakan untuk membentuk form Login dan komponen-komponen yang akan ditampilkan didalamnya. + getFormRegistrasi() Operasi ini digunakan untuk membentuk form Registrasi dan komponen-komponen yang akan ditampilkan didalamnya. + getFormRestiterMenu() Operasi ini digunakan untuk membentuk form Restiter Menu dan komponen-komponen yang akan ditampilkan didalamnya. + getFormSearchSchedule() Operasi ini digunakan untuk membentuk form Search Schedule dan komponen-komponen yang akan ditampilkan didalamnya. + getFormBookFlight() Operasi ini digunakan untuk membentuk form Book Ticket dan komponen-komponen yang akan ditampilkan didalamnya. + getFormMyBooking() Operasi ini digunakan untuk membentuk form MyBooking dan komponen-komponen yang akan ditampilkan didalamnya. +getFormListFlightSchedule() Operasi ini digunakan untuk membentuk form List Flight Schedule dan komponen-komponen yang ditampilkan didalamnya. +getTextBoxReservasi() Operasi ini digunakan membentuk Text Box dan komponen- komponen yang ada didalamnya. getTextBoxHistori()</pre>	

Operasi ini digunakan untuk membentuk Text Box Histori dan komponen-komponen yang ada didalamnya.

+getFormHistori()

Operasi ini digunakan untuk membentuk Form Histori dan komponen-komponen yang ada didalamnya.

+validasiFormRegistrasi()

Operasi ini digunakan untuk memeriksa inputan dari User pada form registrasi.

+validasiFormSearchSchedule()

Operasi ini digunakan untuk memeriksa inputan dari User pada form search schedule.

+validasiFormPembayaran()

Operasi ini digunakan untuk memeriksa inputan dari User pada form registrasi.

+validasiFormPembatalan()

Operasi ini digunakan untuk memeriksa inputan dari User dari form pembatalan keberangkatan.

+validasiLogin()

Operasi ini digunakan untuk memeriksa inputan berupa identitas user pada form login.

2.2.3.2 Spesifikasi Design class ControlLogin

ControlLogin	<<control>>
+ ControlLogin() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + parsingDataLogin() Operasi ini digunakan untuk membaca data user.	

2.2.3.3 Spesifikasi Design Class ControlRegistrasi

ControlRegistrasi	<<control>>

+ ControlRegistrasi()

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

+ insertDataMember()

Operasi ini digunakan untuk memasukkan data Member baru ke dalam database.

2.2.3.4 Spesifikasi Design Class ControlSearchSchedule

ControlSearchSchedule	<<control>>
+ ControlSearchSchedule() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + parsingDataFlightSchedule() Operasi ini digunakan untuk membaca data flight schedule.	

2.2.3.5 Spesifikasi Design Class ControlMyBooking

ControlBookFlight	<<control>>
+ ControlBookFlight() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + insertReservasi() Operasi ini digunakan untuk memasukkan data reservasi kedalam database penumpang yang diinputkan user. + parsingReservasi() Operasi ini digunakan untuk membaca data reservasi. + cancelReservasi() Operasi ini digunakan untuk membatalkan reservasi.	

2.2.3.6 Spesifikasi Design Class ControlPembayaran

ControlPembayaran	<<control>>
+ ControlPembayaran()	

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

+ getDataPembayaran()

Operasi ini digunakan untuk melakukan pembayaran tiket dan digunakan untuk mendapatkan kode pembayaran.

2.2.3.7 Spesifikasi Design Class ControlHistori

ControlHistori	<<control>>
<pre>+ ControlHistori() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. + insertHistori() Operasi ini digunakan untuk menambahkan data Histori kedalam database. + getHistori() Operasi ini digunakan untuk membaca data histori.</pre>	

2.2.3.8 Spesifikasi Design Class Member

Member	<<entity>>
<pre>-ID_member: int Atribut ini digunakan untuk menyimpan id member -Nama_member: String Atribut ini digunakan untuk menyimpan nama member -NoTelepon: int Atribut ini digunakan untuk menyimpan nomor telepon member -Username : String Atribut ini digunakan untuk menyimpan username member -Password : String Atribut ini digunakan untuk menyimpan password pengguna.</pre>	
<pre>+ userModel() Default konstruktor, digunakan untuk inisialisasi semua</pre>	

atribut dari kelas ini.

2.2.3.9 Spesifikasi Design Class FlightSchedule

FlightSchedule	<<entity>>
-tanggalBerangkat: String Atribut ini digunakan untuk menyimpan tanggal berangkat -KotaAsal: String Atribut ini digunakan untuk menyimpan kota tujuan penerbangan -KotaTujuan : String Atribut ini digunakan untuk menyimpan tempat tujuan penerbangan -JamBerangkat : String Atribut ini digunakan untuk menyimpan jam berangkat. -JamTiba : String Atribut ini digunakan untuk menyimpan jam tiba. -class: String Atribut ini digunakan untuk menyimpang class.	
+ FlightSchedule() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	

2.2.3.10 Spesifikasi Design Class Reservasi

Reservasi	<<entity>>
-namaMaskapai: String Atribut ini digunakan untuk menyimpan nama maskapai. -nomorPenerbangan : DateTime Atribut ini digunakan untuk menyimpan nomor penerbangan. -jenisPesawat : String Atribut ini digunakan untuk menyimpan jenis pesawat. -tanggalBerangkat: DateTime Atribut ini digunakan untuk menyimpan tanggal keberangkatan -Origin : DateTime Atribut ini digunakan untuk menyimpan kota asal	

keberangkatan

-Depart : String

Atribut ini digunakan untuk menyimpan kota tujuan.

-tanggalBooking: DateTime

Atribut ini digunakan untuk menyimpan tanggal pemesanan tiket pesawat.

-class: String

Atribut ini digunakan untuk menyimpan class

-kategoriClass: String

Atribut ini digunakan untuk menyimpan kategori class

-hargaClass: int

Atribut ini digunakan untuk harga class

-hargaBayi: int

Atribut ini digunakan untuk menyimpan harga bayi

-jumlahPenumpang: int

Atribut ini digunakan untuk menyimpan jumlah penumpang

-limit : DateTime

Atribut ini digunakan untuk menyimpan batas pengambilan tiket.

-codeBooking : String

Atribut ini digunakan untuk menyimpan code booking reservasi.

+ Reservasi()

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

2.2.3.11 Spesifikasi Design Class Penumpang

Penumpang	<<entity>>
-ID_penumpang: int Atribut ini digunakan untuk menyimpan id reservasi -namaDepan: String Atribut ini digunakan untuk menyimpan nama depan penumpang. -namaBelakang: String Atribut ini digunakan untuk menyimpan nama belakang	

penumpang. -Permintaan : String Atribut ini digunakan untuk menyimpan permintaan penumpang.
+Penumpang() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

3 Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Data Member

Nama	Tipe	Panjang	Keterangan
ID_Member	Integer	-	ID member, Primary Key
Nama_member	VarChar	30	Nama member
NoTelepon_member	VarChar	20	Nomor telepon member
Username_member	VarChar	20	Username yang digunakan member untuk login
Password_member	VarChar	20	Password digunakan member untuk login

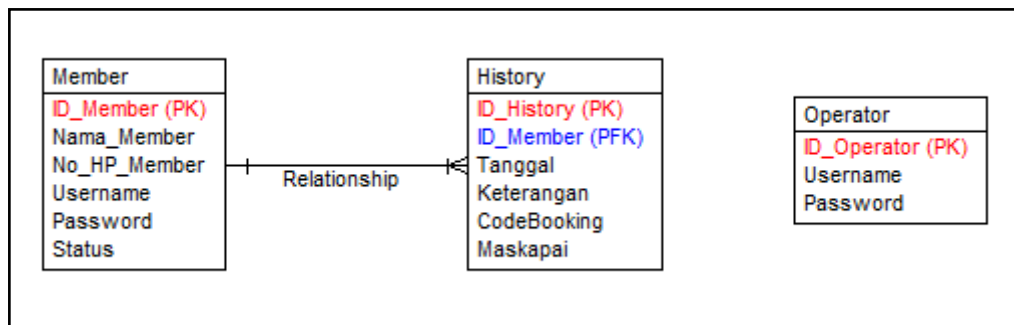
3.1.2 Deskripsi Entitas Data Operator

Nama	Tipe	Panjang	Keterangan
ID_operator	Integer	-	ID admin, Primary Key
Username_operator	VarChar	20	Username yang digunakan operator untuk login
Password_opertor	VarChar	20	Password yang digunakan operator untuk login

3.1.3 Deskripsi Entitas Data Histori

Nama	Tipe	Panjang	Keterangan
ID_History	Integer	-	ID history, Primary Key
ID_Member	Integer	-	ID Member, Foreign Key
Tanggal	Datetime	-	Attribut yang digunakan untuk menyimpan Tanggal jika penumpang mengubah reservasi
Keterangan	VarChar	50	Attribut yang digunakan untuk menyimpan keterangan jika ada perubahan reservasi
Code_Booking	VarChar	6	Attribut yang digunakan untuk menyimpan keterangan kode pemesanan setiap reservasi.
Maskapai	VarChar	20	Attribut yang digunakan untuk menyimpan keterangan Maskapai.

3.1.4 Physical Data Model



Gambar 3.1 Physical Data Model

4 Perancangan Antarmuka

4.1 Sketsa Antarmuka dan Deskripsinya

4.1.1 Antarmuka *Login Member*

Login Member	
Username	
Password	
	Login
	Sign Up
Exit	Menu

Gambar 4.1 Rancangan Antarmuka *Login Member*

Antarmuka Gambar 4.1 digunakan untuk melakukan proses login ke dalam system. Jika dalam basis data terdapat data username dan password maka proses login akan menuju ke antarmuka menu *Restiter* jika tidak maka tampil *alert*.

4.1.2 Antarmuka *Menu Restiter*

Menu Restiter	
1. Search Flight Schedule	
2. MyBooking	
3. My History	
4. Logout	
Exit	Select

Gambar 4.2 Rancangan Antarmuka *Menu Restiter*

Antarmuka Gambar 4.2 merupakan antarmuka yang berisi menu untuk menggunakan aplikasi ini. Jika salah

satu menu list dipilih atau tekan tombol select maka akan menuju antarmuka yang lain.

4.1.3 Antarmuka Registrasi

Registrasi	
Nama	
No HP	
Username	
Password	
Re-Password	
Back Submit	

Gambar 4.3 Rancangan Antarmuka Registrasi

Antarmuka Gambar 4.3 merupakan antarmuka registrasi yang digunakan untuk memasukkan data calon member berupa nama, nomor telepon selular, username, dan password. Jika pengguna menekan tombol "submit" maka data member akan disimpan dalam basis data.

4.1.4 Antarmuka Search Flight Schedule

Search Flight Schedule	
Asal	:
Tujuan	:
Tanggal Keberangkatan	:
Kategori Class	:
Jumlah penumpang	:
Back Search	

Gambar 4.4 Rancangan Antarmuka Search Flight Schedule

Antarmuka Gambar 4.4 merupakan antarmuka yang digunakan pengguna untuk mencari tiket termurah, dimana

pengguna terlebih dahulu memilih rute asal, rute tujuan, kategori class, tanggal keberangkatan dan jumlah penumpang untuk rencana keberangkatan. Jika pengguna menekan tombol "search" maka akan ditampilkan rekomendasi tiket termurah pada antarmuka selanjutnya.

4.1.5 Antarmuka *List Flight Schedule*

List Flight Schedule	
Informasi jadwal penerbangan	
Informasi jadwal penerbangan	
Informasi jadwal penerbangan	
Informasi jadwal penerbangan	
Informasi jadwal penerbangan	
	Select Next >> << Previous
Back	Menu

Gambar 4.5 Rancangan Antarmuka *List Flight Schedule*

Antarmuka Gambar 4.5 merupakan antarmuka yang berisi informasi detail jadwal penerbangan. Informasi yang ditampilkan berupa jenis pesawat, nomor penerbangan, jam keberangkatan, jam tiba dan harga tiket pesawat. Tampilan tersebut merupakan rekomendasi tiket termurah dari tiga maskapai penerbangan. Jika pengguna menekan tombol "Next >>", maka akan ditampilkan rekomendasi tiket termurah untuk hari selanjutnya, dan jika pengguna menekan tombol "Previous", maka akan ditampilkan rekomendasi tiket termurah untuk hari sebelumnya.

4.1.6 Antarmuka *Book Flight*

Book Flight	
Title	
Nama Depan	
Nama Belakang	
Permintaan	
BackContinue	

Gambar 4.6 Rancangan Antarmuka *Book Flight*

Antarmuka Gambar 4.6 merupakan antarmuka yang digunakan member untuk mengisi nama penumpang serta permintaan khusus dari setiap penumpang. Jika pengguna menekan tombol "continue" maka nama depan, nama belakang dan permintaan penumpang akan disimpan di basis data.

4.1.7 Antarmuka Pembayaran

Book Flight	
Pilih Pembayaran	
Credit Card Type	
Name on as Card	
Credit Card Number	
Expiry Date	
CVN no.	
BackContinue	

Gambar 4.7 Rancangan Antarmuka Pembayaran

Antarmuka gambar 4.7 merupakan antarmuka yang digunakan pengguna untuk melakukan pembayaran tiket

melaui ATM atau Credit Card. Pada *choicegroup* "Pilih pembayaran" terdapat dua pilihan yaitu pembayaran melalu ATM atau Credit Card. Jika pengguna menekan tombol "continue" maka pembayaran tiket akan disimpan dalam basis data.

4.1.8 Antarmuka *Display Reservasi*

Tampil Reservasi	
Passenger Details	
Nama	
Reservasi Details	
Code booking	
Date of booking	
Time limit	
Itinerary Details	
Flight	
Asal	
Tujuan	
Class	
Cancel	
Selesai	Pilih

Gambar 4.8 Rancangan Antarmuka *Display Reservasi*

Antarmuka gambar 4.8 merupakan antarmuka yang berisi tentang reservasi tiket yang sudah dibuat. Reservasi yang ditampilkan berupa *Passenger Details*, *Reservasi Details*, dan *Itenary Details*. Jika pengguna menekan tombol "cancel", akan berpindah pada antarmuka reservasi dan jika pengguna menekan tombol "selesai" maka akan berpindah ke antarmuka "menu".

4.1.9 Antarmuka *Cancel Reservasi*

Cancel Reservasi	
Data Pembatal	
Name:	
Contact:	
Remarks:	
No	Yes

Gambar 4.9 Rancangan Antarmuka *Cancel Reservasi*

Antarmuka gambar 4.9 merupakan antarmuka yang digunakan pengguna untuk memasukkan data pembatal reservasi tiket. Member harus mengisi *name*, *contact*, dan *remarks* sebelum melakukan pembatalan. Jika pengguna menekan tombol "No" maka pembatalan reservasi akan ditunda, dan jika pengguna menekan tombol "yes", maka status reservasi pada basis data sudah berubah.

4.1.10 Antarmuka *Search Reservasi*

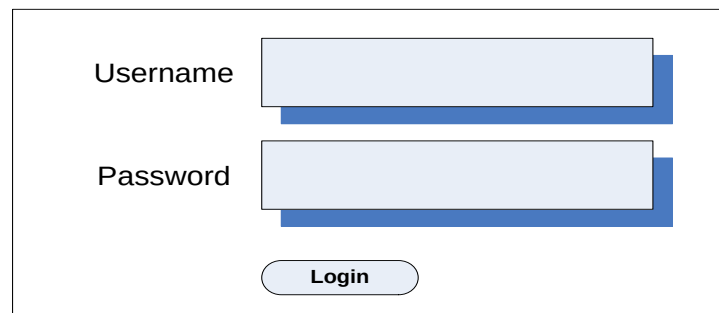
Check Reservasi	
Maskapi Penerbangan	
Code Booking	
Back	Get Status

Gambar 4.10 Rancangan Antarmuka *Search Reservasi*

Antarmuka gambar 4.10 merupakan antarmuka yang digunakan member untuk memeriksa status reservasi

tiket. Jika penumpang ingin melihat informasi status pemesanan tiket maka member harus memilih maskapai penerbangan dan mengisi *code booking* terlebih dahulu. J

4.1.11 Antarmuka *Login Operator*



Gambar 4.11 Rancangan Antarmuka Login Operator

Antarmuka gambar 4.11 digunakan untuk melakukan proses login ke dalam sistem. Jika dalam basis data terdapat data username dan password maka proses login akan menuju ke halaman Home jika tidak maka tampil pesan error.

4.1.12 Antarmuka *Approve Member*

[Home](#)
[Lihat Permintaan](#)
[Lihat Member](#)
[Lihat Pembayaran](#)
[Logout](#)

Lihat Permintaan

Lihat Permintaan Member : No ID

ID Member
Nama
Telepon Selular

Konfirmasi

Gambar 4.12 Rancangan Antarmuka *Approve Member*

Antarmuka gambar 4.12 merupakan antarmuka yang digunakan operator untuk mengkonfirmasi permintaan dari calon member. Tombol konfirmasi merupakan tombol yang digunakan untuk menyetujui permintaan untuk menjadi member. Status member akan berubah dari 0 menjadi 1 pada basis data.

4.1.13 Antarmuka *Delete Member*

Home

Lihat Permintaan

Lihat Member

Lihat Pembayaran

Logout

Lihat Member

ID Member

Nama

Telepon Selular

Hapus

ID Member

Nama

Telepon Selular

Hapus

Gambar 4.13 Rancangan Antarmuka Delete Member

Antarmuka gambar 4.13 merupakan antarmuka yang digunakan operator untuk menghapus data member dari basis data. Jika Operator mengklik tombol "Hapus", maka data member akan terhapus dari basis data sesuai dengan ID Member seperti pada antarmuka.