

BAB VI

PENUTUP

VI.1 Kesimpulan

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

8. Sistem informasi penjualan tiket (*ticketing*) yang mampu mencegah penyalahgunaan peruntukan tiket telah berhasil dirancang dengan adanya pencatatan data nomor detail transaksi yang sekaligus merupakan data tiket yang telah di-*write* maupun *read* melalui pengintegrasian sistem dengan *RFID access control* yang telah terpasang di lokasi *counter* dan *gate*.
9. Sistem informasi penjualan tiket telah memberi fasilitas bagi user untuk mencatat transaksi baik yang dilakukan secara langsung di lokasi objek wisata, melalui email maupun melalui *Agent*.
10. Sistem informasi penjualan tiket telah memberikan fasilitas bagi pihak Manajemen dalam melakukan pencetakan laporan omzet penjualan tiket pada periode tertentu, tahun tertentu maupun berdasarkan *Agent* tertentu.

VI.2 Saran

Saran dan masukan yang dapat disampaikan penulis terhadap pembuatan SIPETIK ini dimasa yang akan datang adalah:

1. Sistem diharapkan dapat dikembangkan lebih lanjut dengan memaksimalkan fitur *online booking*, contohnya dengan adanya pengelolaan website dan kerjasama dengan pihak perbankan terkait proses pembayaran tiket sehingga memudahkan customer dalam melakukan transaksi online serta serta data input *online customer* yang dapat terupdate secara otomatis pada *database*.

DAFTAR PUSTAKA

- Aipasha M., 2012. *Efektivitas Radio Frequency Identification (RFID) di Kelompok Layanan Terbuka Perpustakaan Nasional Republik Indonesia*. Dalam situs <http://ejournal-s1.undip.ac.id/index.php/jip/article/view/546>, diakses pada tanggal 12 Desember 2012, pukul 21.36 WIB.
- Bandal, A., Nawale, S., 2012. RFID Security using Lightweight Mutual Authentication and Ownership Transfer Protocol. *International Journal of Ad hoc, Sensor, and Ubiquitous Computing (IJASUC)* 3, 123-133.
- Cahyadi, D., 2009. *Desain Sistem Absensi PNS berbasis Teknologi RFID*. *Jurnal Informatika Mulawarman* 4, 29-36.
- Hamid, W., Hong, L. C., 2008. *Development and Implementation of Radio-Frequency Identification (RFID) Technology for Inventory Management System: a Case Study*. *Mechanical Journal* 27, 51, 68.
- Harpal, S., Satinder, J., Ravinder, P., 2012. *Red Light Violation Detection using RFID*. *International Journal of Computing and Business Research*. Dalam situs <http://www.researchmanuscripts.com/isociety2012/16>

.pdf, diakses pada tanggal 12 Desember 2012, pukul 21.36 WIB.

Hidayat, R., 2010. *Teknologi Wireless RFID untuk Perpustakaan Polnes : suatu Peluang*. Jurnal Informatika Mulawarman 5, 42-49.

Kamaladevi, B., 2010. *RFID-The Best Technology in Supply Chain Management*. International Journal of Innovation, Management and Technology 1 No. 2, 198-204.

Konidala, D., Kin, D., Yeun, C., 2011. *Security Framework for RFID-based Applications in Smart Home Environment*. Journal of Information Processing Systems 7, 111-120.

Nabila, F., 2009. *Teknik Kompresi XML Data Rekam Medis pada Smartcard RFID*). Jurnal Konvergensi 5, 9-18.

Saputra, D., Cahyadi, D., Kridalaksana, A. H., 2010. *Sistem Otomasi Perpustakaan dengan Menggunakan Radio Frequency Identification (RFID)*. Jurnal Informatika Mulawarman 5, 1-11.

Supriyono, 2010. *Penerapan Aplikasi RFID di Bidang Perpustakaan*. Dalam situs <http://prisekip.blog.ugm.ac.id/files/2009/08/11.pdf>, diakses pada tanggal 12 Desember 2012, pukul 21.13 WIB.

Tarigan, Z. J., 2004. *Integrasi Teknologi RFID dengan Teknologi ERP untuk Otomatisasi Data (Studi Kasus pada Gudang Barang Jadi Perusahaan Furniture)*. Jurnal Teknik Industri 6, 134-141.

Vanany, I., Shahrour, A., 2009. *Pengadopsian Teknologi RFID di Rumah Sakit Indonesia, Manfaat dan Hambatannya*. Jurnal Teknik Industri 11, 82-94.

Wahyu, Y., Rahardian, D., Rustamaji, 2003. *Radio-Frequency Identification (RFID) dengan Active Integrated Antennas (AIAs)*. Jurnal Elektronika dan Telekomunikasi III, 37-42.

Yudastoro, T. R., 2012. *Rancang Bangun Sistem Pendeteksi Pencurian Buku Perpustakaan Berbasis RFID*. Journal of Technical Education and Training 1, 8-14.

Yusianto, R., 2010. *Implementasi Teknologi RFID dalam Perencanaan dan Pengendalian Persediaan Sistem Distribusi Barang*. Techno Science Journal 4, 554-560.

Yusianto, R., 2010. *Rancang Bangun Teknologi Radio Frequency Identification (RFID) untuk Efisiensi Waktu dalam Layanan Check Out Supermarket*. Jurnal Riptek 4, 55-60.

Yusianto, R., 2011. *Pengembangan Prototype Teknologi RFID untuk Optimalisasi Stok dalam Rantai Pasok*



LAMP IRAN

SKPL

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

SIPETIK

(Sistem Informasi Penjualan Tiket dengan
Memanfaatkan Teknologi *Radio Frequency
Identification (RFID)*)

Untuk :


Mandara Wisata Wenara Wana Monkey Forest
Ubud, Bali

Dipersiapkan oleh:

Gusti Ayu Sinta Adnyani / 090706029

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		SKPL-SIPETIK		1/40
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

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

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

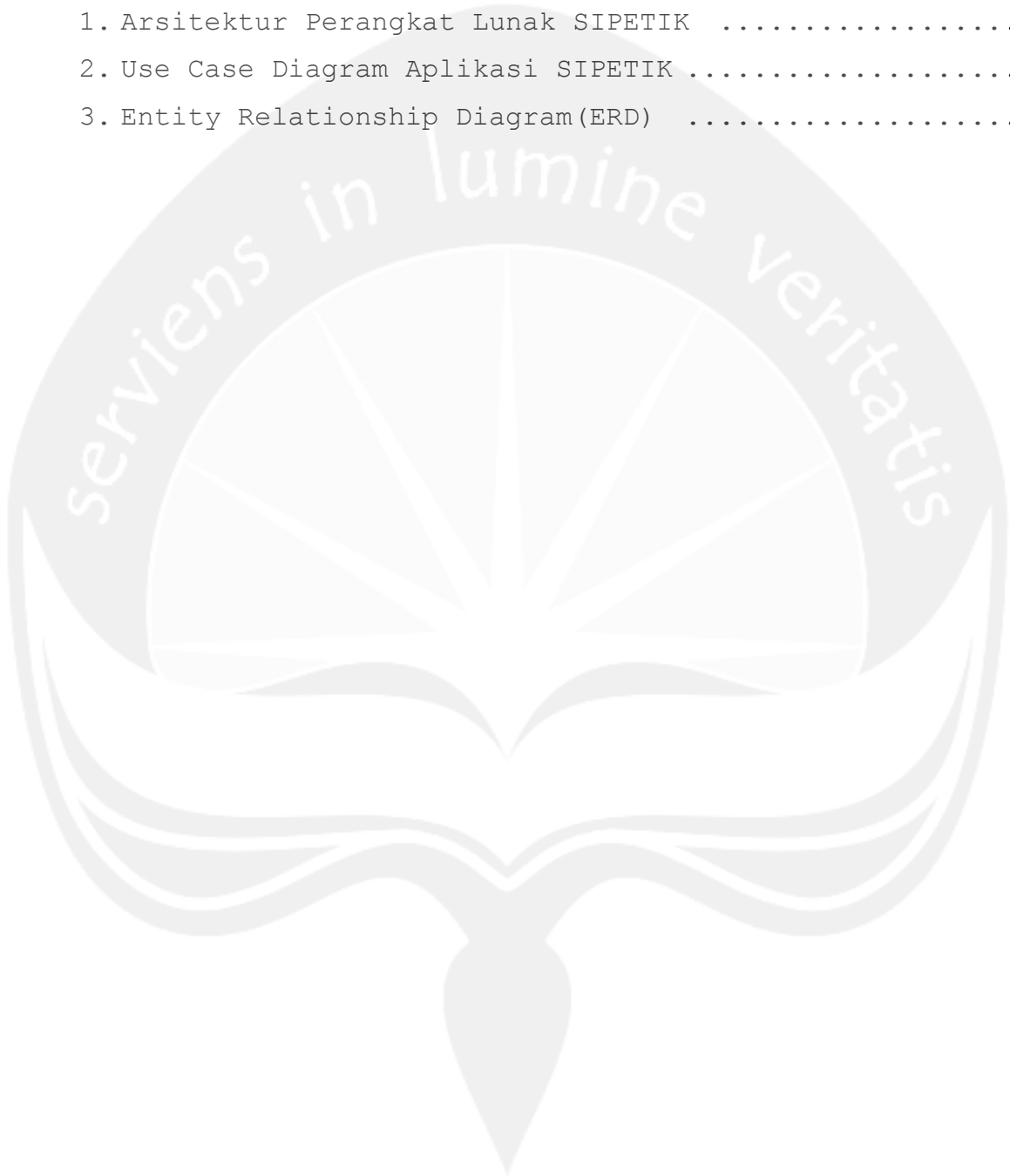
1.	Pendahuluan	7
1.1.	Tujuan	7
1.2.	Lingkup Masalah	7
1.3.	Definisi, Akronim, dan Singkatan	8
1.4.	Referensi	8
1.5.	Deskripsi Umum (Overview)	9
2.	Deskripsi Kebutuhan	9
2.1.	Perspektif Produk	9
2.2.	Kebutuhan Fungsionalitas Produk	11
2.3.	Karakteristik Pengguna	16
2.4.	Batasan - Batasan	16
2.5.	Asumsi dan Ketergantungan	16
3.	Kebutuhan Khusus	16
3.1.	Kebutuhan Antar Muka Eksternal	16
3.1.1.	Antarmuka Pemakai	17
3.1.2.	Antarmuka Perangkat Keras	17
3.1.3.	Antarmuka Perangkat Lunak	17
3.1.4.	Antarmuka Komunikasi	18
3.2.	Kebutuhan Fungsionalitas Perangkat Lunak	18
3.2.1.	Use Case Diagram	18
4.	Spesifikasi Rinci Kebutuhan	19
4.1.	Spesifikasi Kebutuhan Fungsionalitas	19
4.1.1.	Use Case Spesification: Login	18
4.1.2.	Use Case Spesification: Ubah Username.	20
4.1.3.	Use Case Spesification: Ubah Password	21
4.1.4.	Use Case Spesification: Pengelolaan Data Pengguna	22
4.1.5.	Use Case Spesification: Pengelolaan Data Agent	25
4.1.6.	Use Case Spesification: Pengelolaan Data Customer	28
4.1.7.	Use Case Spesification: Pengelolaan Data Tipe Tiket	30
4.1.8.	Use Case Spesification: Penyediaan Laporan Transaksi Penjualan Tiket Periode tertentu	33
4.1.9.	Use Case Spesification: Penyediaan Laporan Transaksi Penjualan Tiket per Bulan dalam Tahun tertentu	34
4.1.10.	Use Case Spesification: Penyediaan Laporan Transaksi Penjualan Tiket berdasarkan Agent	36
4.1.11.	Use Case Spesification: Pengelolaan	

Transaksi Penjualan Tiket	37
4.1.12. Use Case Spesification: Validasi Tiket ...	38
5. Entity Relationship Diagram (ERD)	40



Daftar Gambar

1. Arsitektur Perangkat Lunak SIPETIK	10
2. Use Case Diagram Aplikasi SIPETIK	18
3. Entity Relationship Diagram(ERD)	37



1. Pendahuluan

Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIPETIK (Sistem Informasi Penjualan Tiket dengan Memanfaatkan Teknologi *Radio Frequency Identification (RFID)*) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan pengguna), layanan-layanan yang diberikan dalam sistem, atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SIPETIK ini juga mendefinisikan batasan perancangan perangkat lunak.

Lingkup Masalah

Perangkat Lunak SIPETIK dikembangkan dengan tujuan untuk :

4. Menyediakan sistem informasi penjualan tiket (*ticketing*) yang mampu mencegah penyalahgunaan peruntukan tiket.
5. Menyediakan sistem informasi penjualan tiket yang memberi fasilitas bagi *user* untuk melakukan pencatatan transaksi, baik secara langsung di lokasi, melalui *email* maupun melalui *Agent*.
6. Menyediakan pelaporan (*reporting*) terkait dengan transaksi penjualan tiket tersebut.

Program Studi Teknik Informatika	SKPL – SIPETIK	162/ 274
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		

Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIPETIK-XXX	Kode yang merepresentasikan kebutuhan pada SIPETIK (Sistem Informasi Penjualan Tiket dengan Memanfaatkan Teknologi <i>RFID</i>) dimana XXX merupakan nomor fungsi produk.
SIPETIK	Perangkat lunak yang digunakan untuk menyediakan sistem informasi penjualan tiket di Mandara Wisata Wenara Wana Monkey Forest, Bali.
Server	Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.
<i>RFID</i>	<i>RFID</i> adalah singkatan dari <i>Radio Frequency Identification</i> yang merupakan proses identifikasi seseorang atau objek menggunakan frekuensi transmisi radio.

Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

2. Cahyadi, D., 2009. *Desain Sistem Absensi PNS berbasis Teknologi RFID*. Jurnal Informatika Mulawarman 4, 29-36.

Program Studi Teknik Informatika	SKPL – SIPETIK	163/ 274
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. Hamid, W., Hong, L. C., 2008. *Development and Implementation of Radio-Frequency Identification (RFID) Technology for Inventory Management System: a Case Study*. Mechanical Journal 27, 51, 68.

Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian pertama 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 SIPETIK yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pelanggan, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SIPETIK tersebut.

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

Deskripsi Kebutuhan

Perspektif produk

SIPETIK merupakan perangkat lunak yang dikembangkan untuk membantu pihak manajemen Mandara Wisata Wenara Wana Monkey Forest, Ubud-Bali dalam menyediakan sistem informai penjualan tiket yang mampu

Program Studi Teknik Informatika	SKPL – SIPETIK	164/ 274
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		

mencegah penyalahgunaan peruntukan tiket, memudahkan pencatatan transaksi secara langsung di lokasi objek wisata, melalui email maupun melalui agent, serta dapat menampilkan *report* transaksi penjualan tiket. Sistem ini dapat menangani pengelolaan data yang meliputi proses *Entry, Update, Delete* dan *Display* data pengguna serta data transaksi yang dilengkapi dengan pencetakan laporan hasil transaksi.

Perangkat lunak SIPETIK ini berjalan pada *platform* Windows untuk desktop, dan dibuat menggunakan bahasa pemrograman Microsoft Visual C#. Sedangkan untuk lingkungan pemrogramannya menggunakan Microsoft Visual Studio 2005. SIPETIK juga memanfaatkan teknologi *Radio Frequency Identification (RFID)* yang terintegrasi dengan sistem. *Card* berfungsi sebagai tiket fisik untuk pelanggan dan *Reader* berfungsi sebagai pembaca validitas *card*.

Pelanggan akan berinteraksi dengan sistem melalui antarmuka *GUI (Graphical User Interface)*. Pada sistem ini, seperti terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa *client server*, di mana *desktop software* dapat mengakses aplikasi melalui *server* yang terhubung dengan basis data.



Gambar 1 Arsitektur Perangkat Lunak SIPETIK

Program Studi Teknik Informatika	SKPL – SIPETIK	165/ 274
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		

Kebutuhan Fungsionalitas Produk

Fungsi produk perangkat lunak SIPETIK adalah sebagai berikut :

1. Fungsi Login (SKPL-SIPETIK-001)

Fungsi ini digunakan oleh pengguna untuk dapat masuk dalam sistem yang akan digunakan sesuai dengan role yang dimiliki masing-masing pengguna. Fungsi ini mencakup :

a. Fungsi Input Data Login (SKPL-SIPETIK-001-01)

Fungsi ini digunakan user untuk menginputkan username serta password.

b. Fungsi Validasi Data Login (SKPL-SIPETIK-001-02)

Fungsi ini melakukan proses pengecekan apakah data login yang diinputkan valid atau tidak.

2. Fungsi Ubah Username (SKPL-SIPETIK-002) .

Fungsi ini digunakan untuk mengubah username yang digunakan untuk login. Fungsi ini dapat diakses semua pengguna.

3. Fungsi Ubah Password (SKPL-SIPETIK-003) .

Fungsi ini digunakan untuk mengubah password yang digunakan untuk login. Fungsi ini dapat diakses semua pengguna.

4. Fungsi Pengelolaan Data Pengguna (SKPL-SIPETIK-004) .

Fungsi ini digunakan untuk mengelola data pengguna SIPETIK. Fungsi ini hanya dapat diakses oleh user yang memiliki role sebagai Administrator. Fungsi Pengelolaan Data Pengguna meliputi :

Program Studi Teknik Informatika	SKPL – SIPETIK	166/ 274
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

a. Fungsi Tambah Data Pengguna (SKPL-SIPETIK-004-01)

Fungsi Tambah Data Pengguna ini digunakan untuk menambahkan data pengguna baru.

b. Fungsi Ubah Data Pengguna (SKPL-SIPETIK-004-02)

Fungsi ini digunakan untuk memperbaiki kesalahan atau mengubah data pengguna.

c. Fungsi Hapus Data Pengguna (SKPL-SIPETIK-004-03)

Fungsi Hapus Data Pengguna ini digunakan untuk menghapus data pengguna yang tidak diperlukan.

d. Fungsi Display Data Pengguna (SKPL-SIPETIK-004-04)

Fungsi ini digunakan untuk menampilkan data pengguna yang tersimpan, baik seluruh atau berdasarkan kriteria tertentu yang dipilih.

5. Fungsi Pengelolaan Data Agent (SKPL-SIPETIK-005).

Fungsi ini digunakan untuk mengelola data Agent Monkey Forest. Fungsi ini hanya dapat diakses oleh user yang memiliki role sebagai Manager/Administrator. Fungsi Pengelolaan Data Agent meliputi :

a. Fungsi Tambah Data Agent (SKPL-SIPETIK-005-01)

Fungsi ini digunakan untuk menambahkan data agent baru.

b. Fungsi Ubah Data Agent (SKPL-SIPETIK-005-02)

Fungsi ini digunakan untuk memperbaiki kesalahan atau mengubah data agent.

c. Fungsi Hapus Data Agent (SKPL-SIPETIK-005-03)

Fungsi ini digunakan untuk menghapus data agent yang tidak diperlukan.

Program Studi Teknik Informatika	SKPL – SIPETIK	167/ 274
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		

d. Fungsi Display Data Agent (SKPL-SIPETIK-005-04)

Fungsi ini digunakan untuk menampilkan data agent yang tersimpan, baik seluruh atau berdasarkan kriteria tertentu yang dipilih.

6. Fungsi Pengelolaan Data Customer (SKPL-SIPETIK-006).

Fungsi ini digunakan untuk mengelola data Customer Monkey Forest. Fungsi ini hanya dapat diakses oleh user yang memiliki role sebagai Manager/Administrator. Fungsi Pengelolaan Data Customer meliputi :

a. Fungsi Tambah Data Customer (SKPL-SIPETIK-006-01)

Fungsi ini digunakan untuk menambahkan data customer baru.

b. Fungsi Ubah Data Customer (SKPL-SIPETIK-006-02)

Fungsi ini digunakan untuk memperbaiki kesalahan atau mengubah data customer.

c. Fungsi Hapus Data Customer (SKPL-SIPETIK-006-03)

Fungsi ini digunakan untuk menghapus data customer yang tidak diperlukan.

d. Fungsi Display Data Customer (SKPL-SIPETIK-006-04)

Fungsi ini digunakan untuk menampilkan data customer yang tersimpan, baik seluruh atau berdasarkan kriteria tertentu yang dipilih.

7. Fungsi Pengelolaan Data Tipe Tiket (SKPL-SIPETIK-007).

Fungsi ini digunakan untuk mengelola data tipe tiket masuk Objek Wisata Monkey Forest. Fungsi ini

Program Studi Teknik Informatika	SKPL – SIPETIK	168/ 274
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		

hanya dapat diakses oleh user yang memiliki role sebagai Manager/ Administrator. Fungsi Pengelolaan Data Tipe Tiket meliputi :

a. Fungsi Tambah Data Tipe Tiket (SKPL-SIPETIK-007-01)

Fungsi ini digunakan untuk menambahkan data tipe tiket baru.

b. Fungsi Ubah Data Tipe Tiket (SKPL-SIPETIK-007-02)

Fungsi ini digunakan untuk memperbaiki kesalahan atau mengubah data agent.

c. Fungsi Hapus Data Tipe Tiket (SKPL-SIPETIK-007-03)

Fungsi ini digunakan untuk menghapus data tipe tiket yang tidak diperlukan.

d. Fungsi Display Data Tipe Tiket (SKPL-SIPETIK-007-04)

Fungsi ini digunakan untuk menampilkan data tipe tiket yang tersimpan, baik seluruh atau berdasarkan kriteria tertentu yang dipilih.

8. Fungsi Cetak Laporan Transaksi Periode tertentu (SKPL-SIPETIK-008) .

Fungsi ini digunakan untuk mencetak laporan hasil penjualan tiket per periode tertentu.

9. Fungsi Cetak Laporan Transaksi per Bulan dalam Periode tertentu (SKPL-SIPETIK-009) .

Fungsi ini digunakan untuk mencetak laporan hasil penjualan tiket per bulan dalam periode tertentu yang dipilih oleh pengguna.

Program Studi Teknik Informatika	SKPL – SIPETIK	169/ 274
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		

10. Fungsi Cetak Laporan Transaksi berdasarkan (SKPL-SIPETIK-010) .

Fungsi ini digunakan untuk mencetak laporan hasil penjualan tiket berdasarkan agent tertentu.

11. Fungsi Pengelolaan Data Transaksi (SKPL-SIPETIK-011) .

Fungsi ini digunakan untuk input transaksi penjualan tiket. Fungsi ini meliputi:

a. Fungsi Tambah Data Transaksi (SKPL-SIPETIK-011-01)

Fungsi ini digunakan untuk menambahkan data transaksi baru.

b. Fungsi Ubah Data Transaksi (SKPL-SIPETIK-011-02)

Fungsi ini digunakan untuk memperbaiki kesalahan data transaksi.

c. Fungsi Hapus Data Transaksi (SKPL-SIPETIK-011-03)

Fungsi ini digunakan untuk menghapus data transaksi yang tidak diperlukan.

d. Fungsi Display Data Transaksi (SKPL-SIPETIK-011-04)

Fungsi ini digunakan untuk menampilkan data transaksi yang tersimpan seluruhnya atau berdasarkan kriteria tertentu.

12. Fungsi Validasi Tiket (SKPL-SIPETIK-012) .

Fungsi ini digunakan untuk memvalidasi tiket atau hak akses memasuki area Monkey Forest.

Program Studi Teknik Informatika	SKPL – SIPETIK	170/ 274
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		

Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SIPETIK adalah sebagai berikut :

1. Memahami pengoperasian *Personal Computer (PC)*.
2. Memahami pengoperasian fungsi-fungsi yang terdapat dalam SIPETIK.

Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak SIPETIK tersebut adalah :

1. Kebijakan Umum
Berpedoman pada tujuan dari pengembangan perangkat lunak SIPETIK.
2. Keterbatasan perangkat keras
Keterbatasan perangkat keras dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

Asumsi dan Ketergantungan

Sistem ini dapat dijalankan pada aplikasi desktop dan hanya bisa dijalankan pada PC dengan sistem operasi Windows XP, Windows Vista atau Windows7.

Kebutuhan khusus

Kebutuhan antarmuka eksternal

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

Program Studi Teknik Informatika	SKPL – SIPETIK	171/ 274
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		

Antarmuka pemakai/pelanggan

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form. Pada proses cetak laporan hasil penjualan tiket, antarmuka ditampilkan dalam bentuk data tabel serta tampilan *line-graph* untuk pencarian hasil penjualan tiket per bulan pada periode tertentu.

Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SIPETIK adalah:

1. Perangkat PC (sistem operasi Windows XP, Windows Vista atau Windows 7).
2. CPU
3. Monitor
4. Keyboard
5. Mouse
6. Printer

Antarmuka perangkat lunak

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

4. Nama : SQL Server Express 2005

Sumber : Microsoft

Sebagai database management sistem (DBMS) yang digunakan untuk menyimpan data di sisi server.

5. Nama : .NET Framework

Sumber : Microsoft

Sebagai framework aplikasi.

6. Nama : Windows XP, Windows Vista, Windows7

Sumber : Microsoft

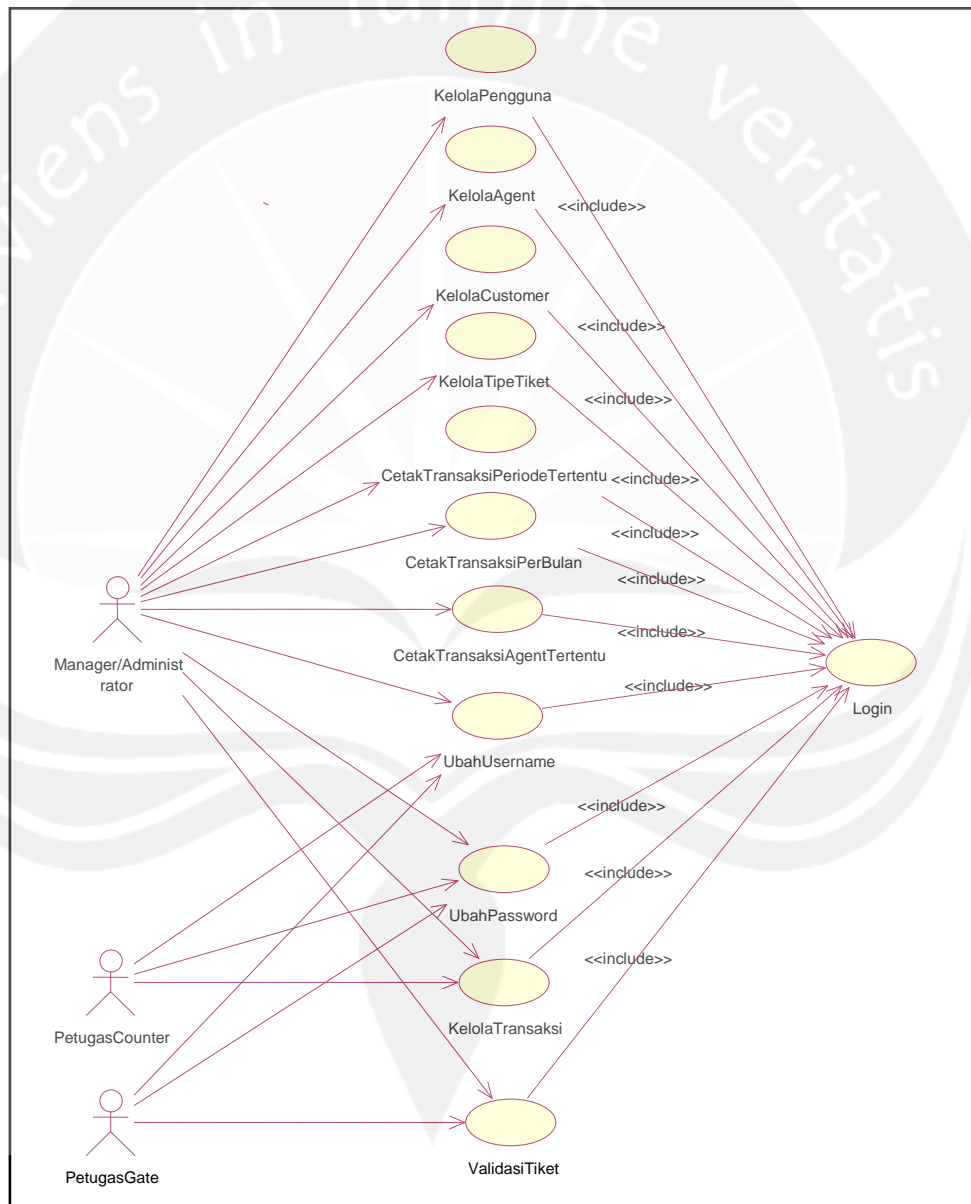
Sebagai sistem operasi.

Program Studi Teknik Informatika	SKPL – SIPETIK	172/ 274
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		

Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIPETIK menggunakan protocol TCP/IP.

Kebutuhan Fungsionalitas Perangkat Lunak
Use Case Diagram



Gambar 2 Use Case Diagram

Spesifikasi Rinci Kebutuhan

Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification : Login

9. Brief Description

Use Case ini digunakan oleh aktor untuk masuk ke dalam sistem. Login didasarkan pada sebuah id unik yaitu username dan password.

10. Primary Aktor

1. Manager/ Administrator
2. PetugasCounter
3. PetugasGate

11. Supporting Aktor

none

12. 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 melakukan validasi terhadap username dan password yang diinputkan aktor
E-1 Username atau password user tidak sesuai
5. Sistem memberikan akses kepada aktor
6. Use Case ini selesai

13. Alternative Flow

None

14. Error Flow

E-1 Password atau nama user tidak sesuai

1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai

Program Studi Teknik Informatika	SKPL – SIPETIK	174/ 274
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

15. PreConditions

none

16. PostConditions

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

4.1.2 Use case Spesification : Ubah Username

9. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan ubah data username yang digunakan untuk masuk ke dalam sistem.

10. Primary Aktor

1. Manager/ Administrator

11. Supporting Aktor

none

12. Basic Flow

7. Use case ini dimulai ketika aktor memilih untuk melakukan pengubahan username
8. Sistem menampilkan data username aktor
9. Aktor memasukkan data username baru
10. Sistem melakukan validasi terhadap data username baru yang dimasukkan aktor
 - E-1 Username baru yang dimasukkan aktor sudah terdaftar
11. Sistem menampilkan peringatan bahwa username aktor telah sukses di ubah
12. Use case ini selesai

13. Alternative Flow

None

Program Studi Teknik Informatika	SKPL – SIPETIK	175/ 274
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		

14. Error Flow

E-1 Username baru yang dimasukkan aktor sudah terdaftar

3. Sistem memberikan peringatan bahwa username baru yang dimasukkan aktor sudah terdaftar

4. Kembali ke basic flow langkah ke 6

15. PreConditions

2. User telah melakukan use case login dan telah memasuki sistem

16. PostConditions

1. Data yang ada pada basis data telah terupdate

Use case Spesification : Ubah Password

2. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan ubah data password yang digunakan untuk masuk ke dalam sistem.

3. Primary Aktor

1. Manager/ Administrator
2. PetugasCounter
3. PetugasGate

4. Supporting Aktor

none

5. Basic Flow

7. Use case ini dimulai ketika aktor memilih untuk melakukan pengubahan password

8. Sistem menampilkan data username aktor

9. Aktor memasukkan data password baru dan konfirmasi password baru

Program Studi Teknik Informatika	SKPL – SIPETIK	176/ 274
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		

10. Sistem melakukan validasi terhadap data password baru yang dimasukkan aktor

E-1 Password baru yang dimasukkan aktor tidak diterima

11. Sistem menampilkan peringatan bahwa password aktor telah sukses di ubah

12. Use case ini selesai

6. Alternative Flow

none

7. Error Flow

E-1 Password baru yang dimasukkan aktor tidak diterima

3. Sistem memberikan peringatan bahwa password baru yang dimasukkan aktor tidak diterima

4. Kembali ke basic flow langkah ke 6

8. PreConditions

2. User telah melakukan use case login dan telah memasuki sistem

9. PostConditions

1. Data yang ada pada basis data telah terupdate

4.1.4 Use case Spesification : Pengelolaan Data Pengguna

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pengguna yang menggunakan sistem.

2. Primary Aktor

1. Manager/ Administrator

3. Supporting Aktor

none

Program Studi Teknik Informatika	SKPL – SIPETIK	177/ 274
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 pengguna
2. Sistem menampilkan semua data pengguna yang ada pada basis data
3. Sistem memberikan pilihan menu dalam pengelolaan pengguna
4. Aktor memilih menu tambah data pengguna
 - A-1 Aktor memilih menu ubah data pengguna
 - A-2 Aktor memilih menu hapus data pengguna
 - A-3 Aktor memilih menu tampil data pengguna
5. Aktor melakukan pengisian data pengguna baru
6. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah
7. Sistem memberikan peringatan bahwa data pengguna baru telah sukses disimpan
8. Use case ini selesai

5. Alternative Flow

- A-1 Aktor memilih menu ubah data pengguna
1. Aktor memilih data pengguna yang akan diubah
 2. Aktor melakukan pengisian data pengguna baru
 3. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah

Program Studi Teknik Informatika	SKPL – SIPETIK	178/ 274
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 memberikan peringatan bahwa data telah berhasil diubah

5. Kembali ke basic flow langkah ke 8

A-2 Aktor memilih menu hapus data pengguna

3. Aktor memilih data pengguna yang akan dihapus

4. Sistem melakukan validasi terhadap data pengguna yang akan dihapus

E-2 Data tidak dapat dihapus

5. Sistem memberikan peringatan bahwa data telah berhasil dihapus

6. Kembali ke basic flow langkah ke 8

A-3 Aktor memilih menu cari data pengguna

2. Aktor memilih pencarian berdasarkan kriteria tertentu

3. Aktor melakukan pengisian data pengguna yang akan dicari

4. Sistem melakukan pencarian berdasarkan kriteria yang dipilih aktor

E-3 Data yang dicari tidak ditemukan

5. Sistem menampilkan data hasil pencarian

6. Kembali ke basic flow langkah ke 8

7. Error Flow

E-1 Data yang dimasukkan aktor salah

1. Sistem memberikan peringatan bahwa data yang dimasukkan aktor salah

2. Kembali ke basic flow langkah ke 8

E-2 Data tidak dapat dihapus

1. Sistem memberikan peringatan bahwa data yang dipilih aktor tidak dapat dihapus

Program Studi Teknik Informatika	SKPL – SIPETIK	179/ 274
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 8
- E-3 Data yang dicari tidak ditemukan
1. Sistem memberikan peringatan bahwa data yang dicari oleh user tidak ditemukan
 2. Kembali ke basic flow langkah ke 8

8. PreConditions

1. User telah melakukan use case login dan telah memasuki sistem

9. PostConditions

1. Data yang ada pada basis data telah terupdate

4.1.5 Use case Spesification : Pengelolaan Data Agent

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data agent yang menggunakan sistem.

6. Primary Aktor

1. Manager/ Administrator

7. Supporting Aktor

none

8. Basic Flow

1. Use case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data agent
2. Sistem menampilkan semua data agent yang ada pada basis data
3. Sistem memberikan pilihan menu dalam pengelolaan agent
4. Aktor memilih menu tambah data agent
 - A-1 Aktor memilih menu ubah data agent
 - A-2 Aktor memilih menu hapus data agent
 - A-3 Aktor memilih menu tampil data agent

Program Studi Teknik Informatika	SKPL – SIPETIK	180/ 274
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. Aktor melakukan pengisian data agent baru
6. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah
7. Sistem memberikan peringatan bahwa data agent baru telah sukses disimpan
8. Use case ini selesai

9. Alternative Flow

A-1 Aktor memilih menu ubah data agent

1. Aktor memilih data agent yang akan diubah
2. Aktor melakukan pengisian data agent baru
3. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah

4. Sistem memberikan peringatan bahwa data telah berhasil diubah
5. Kembali ke basic flow langkah ke 8

A-2 Aktor memilih menu hapus data agent

1. Aktor memilih data agent yang akan dihapus
2. Sistem melakukan validasi terhadap data agent yang akan dihapus
 - E-2 Data tidak dapat dihapus

3. Sistem memberikan peringatan bahwa data telah berhasil dihapus
4. Kembali ke basic flow langkah ke 8

A-3 Aktor memilih menu cari data agent

Program Studi Teknik Informatika	SKPL – SIPETIK	181/ 274
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. Aktor memilih pencarian berdasarkan kriteria tertentu
2. Aktor melakukan pengisian data agent yang akan dicari
3. Sistem melakukan pencarian berdasarkan kriteria yang dipilih aktor
 - E-3 Data yang dicari tidak ditemukan
4. Sistem menampilkan data hasil pencarian
5. Kembali ke basic flow langkah ke 8

6. Error Flow

E-1 Data yang dimasukkan aktor salah

1. Sistem memberikan peringatan bahwa data yang dimasukkan aktor salah
2. Kembali ke basic flow langkah ke 8

E-2 Data tidak dapat dihapus

1. Sistem memberikan peringatan bahwa data yang dipilih aktor tidak dapat dihapus
2. Kembali ke basic flow langkah ke 8

E-3 Data yang dicari tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang dicari oleh user tidak ditemukan
9. Kembali ke basic flow langkah ke 8

7. PreConditions

1. User telah melakukan use case login dan telah memasuki sistem

8. PostConditions

1. Data yang ada pada basis data telah terupdate

Use case Spesification : Pengelolaan Data Customer

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data customer yang menggunakan sistem.

2. Primary Aktor

1. Manager/ Administrator

3. Supporting Aktor

None

4. Basic Flow

1. Use case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data customer
2. Sistem menampilkan semua data customer yang ada pada basis data
3. Sistem memberikan pilihan menu dalam pengelolaan customer
4. Aktor memilih menu tambah data customer
 - A-1 Aktor memilih menu ubah data customer
 - A-2 Aktor memilih menu hapus data customer
 - A-3 Aktor memilih menu tampil data customer
5. Aktor melakukan pengisian data customer baru
6. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah
7. Sistem memberikan peringatan bahwa data customer baru telah sukses disimpan
8. Use case ini selesai

Program Studi Teknik Informatika	SKPL – SIPETIK	183/ 274
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 menu ubah data customer

1. Aktor memilih data customer yang akan diubah
2. Aktor melakukan pengisian data customer baru
3. Sistem melakukan validasi terhadap data yang dimasukkan aktor

E-1 Data yang dimasukkan aktor salah

4. Sistem memberikan peringatan bahwa data telah berhasil diubah
5. Kembali ke basic flow langkah ke 8

A-2 Aktor memilih menu hapus data customer

1. Aktor memilih data customer yang akan dihapus
2. Sistem melakukan validasi terhadap data customer yang akan dihapus

E-2 Data tidak dapat dihapus

3. Sistem memberikan peringatan bahwa data telah berhasil dihapus
4. Kembali ke basic flow langkah ke 8

A-3 Aktor memilih menu cari data customer

1. Aktor memilih pencarian berdasarkan kriteria tertentu
2. Aktor melakukan pengisian data customer yang akan dicari
3. Sistem melakukan pencarian berdasarkan kriteria yang dipilih aktor

E-3 Data yang dicari tidak ditemukan

4. Sistem menampilkan data hasil pencarian
5. Kembali ke basic flow langkah ke 8

6. Error Flow

E-1 Data yang dimasukkan aktor salah

1. Sistem memberikan peringatan bahwa data yang dimasukkan aktor salah
2. Kembali ke basic flow langkah ke 8

E-2 Data tidak dapat dihapus

1. Sistem memberikan peringatan bahwa data yang dipilih aktor tidak dapat dihapus
2. Kembali ke basic flow langkah ke 8

E-3 Data yang dicari tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang dicari oleh user tidak ditemukan
10. Kembali ke basic flow langkah ke 8

7. PreConditions

1. User telah melakukan use case login dan telah memasuki sistem

8. PostConditions

1. Data yang ada pada basis data telah terupdate

4.1.7 Use case Spesification : Pengelolaan Data Tipe Tiket

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data tipe tiket yang menggunakan sistem.

2. Primary Aktor

1. Manager/ Administrator

3. Supporting Aktor

None

Program Studi Teknik Informatika	SKPL – SIPETIK	185/ 274
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 tipe tiket
2. Sistem menampilkan semua data tipe tiket yang ada pada basis data
3. Sistem memberikan pilihan menu dalam pengelolaan agent
4. Aktor memilih menu tambah data agent
 - A-1 Aktor memilih menu ubah data tipe tiket
 - A-2 Aktor memilih menu hapus data tipe tiket
 - A-3 Aktor memilih menu tampil data tipe tiket
5. Aktor melakukan pengisian data tipe tiket baru
6. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah
7. Sistem memberikan peringatan bahwa data tipe tiket baru telah sukses disimpan
8. Use case ini selesai

10. Alternative Flow

- A-1 Aktor memilih menu ubah data tipe tiket
1. Aktor memilih data tipe tiket yang akan diubah
 2. Aktor melakukan pengisian data tipe tiket baru
 3. Sistem melakukan validasi terhadap data yang dimasukkan aktor
 - E-1 Data yang dimasukkan aktor salah

Program Studi Teknik Informatika	SKPL – SIPETIK	186/ 274
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 memberikan peringatan bahwa data telah berhasil diubah

5. Kembali ke basic flow langkah ke 8

A-2 Aktor memilih menu hapus data tipe tiket

1. Aktor memilih data agent yang akan dihapus

2. Sistem melakukan validasi terhadap data tipe tiket yang akan dihapus

E-2 Data tidak dapat dihapus

3. Sistem memberikan peringatan bahwa data tipe tiket telah berhasil dihapus

4. Kembali ke basic flow langkah ke 8

A-3 Aktor memilih menu cari data tipe tiket

1. Aktor memilih pencarian berdasarkan kriteria tertentu

2. Aktor melakukan pengisian data tipe tiket yang akan dicari

3. Sistem melakukan pencarian berdasarkan kriteria yang dipilih aktor

E-3 Data yang dicari tidak ditemukan

4. Sistem menampilkan data hasil pencarian

5. Kembali ke basic flow langkah ke 8

6. Error Flow

E-1 Data yang dimasukkan aktor salah

1. Sistem memberikan peringatan bahwa data yang dimasukkan aktor salah

2. Kembali ke basic flow langkah ke 8

E-2 Data tidak dapat dihapus

1. Sistem memberikan peringatan bahwa data yang dipilih aktor tidak dapat dihapus

Program Studi Teknik Informatika	SKPL – SIPETIK	187/ 274
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 8
E-3 Data yang dicari tidak ditemukan

1. Sistem memberikan peringatan bahwa data yang dicari oleh user tidak ditemukan
2. Kembali ke basic flow langkah ke 8

7. PreConditions

1. User telah melakukan use case login dan telah memasuki sistem

8. PostConditions

1. Data yang ada pada basis data telah terupdate

Use case Spesification : Penyediaan Laporan Transaksi Penjualan Tiket untuk Periode tertentu

1. Brief Description

Use case ini digunakan oleh aktor untuk menyediakan laporan penjualan tiket untuk periode tertentu

2. Primary Aktor

1. Administrator/ Manager

3. Supporting Aktor

none

4. Basic Flow

8. Use Case ini dimulai ketika aktor memilih untuk melakukan pencetakan hasil penjualan tiket untuk periode tertentu
9. Sistem menampilkan menu pilihan pencetakan berdasarkan *range* waktu tertentu
10. Aktor memilih *range* waktu hasil pencetakan

E-1 Range waktu yang dipilih aktor tidak valid

Program Studi Teknik Informatika	SKPL – SIPETIK	188/ 274
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		

11. Sistem menampilkan data berdasarkan *range* periode waktu yang dipilih aktor
12. Aktor menekan button print
13. Sistem melakukan pencetakan laporan penjualan tiket sesuai dengan periode yang dipilih aktor
14. Use case ini selesai

5. Alternative Flow

none

6. Errow Flow

- E-1 Range waktu yang dipilih aktor tidak valid
3. Sistem memberikan peringatan bahwa *range* waktu yang dipilih aktor tidak valid.
 4. Kembali ke basic flow langkah ke 3

7. PreConditions

3. Use Case Login telah dilakukan
4. Aktor telah memasuki sistem

8. PostConditions

2. Data laporan transaksi penjualan tiket per periode tertentu telah dicetak

Use case Spesification : Penyediaan Laporan Transaksi Penjualan Tiket per Bulan dalam Tahun tertentu

1. Brief Description

Use case ini digunakan oleh aktor untuk menyediakan laporan penjualan tiket per bulan dalam tahun tertentu

2. Primary Aktor

1. Administrator/ Manager

Program Studi Teknik Informatika	SKPL – SIPETIK	189/ 274
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 Aktor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pencetakan hasil penjualan tiket per bulan dalam tahun tertentu
2. Sistem menampilkan antarmuka untuk memilih tahun yang ingin dilihat hasil penjualan tiketnya
3. Aktor menginputkan tahun yang ingin disediakan laporan penjualan per bulannya
E-1 Tahun yang diinputkan aktor salah
4. Sistem menampilkan antarmuka yang berisi laporan penjualan per bulan dalam tahun yang dipilih oleh user
5. Aktor menekan button print
6. Sistem melakukan pencetakan laporan penjualan tiket sesuai dengan tahun yang dipilih aktor
7. Use case ini selesai

7. Alternative Flow

none

8. Errow Flow

E-1 Range waktu yang dipilih aktor tidak valid

1. Sistem memberikan peringatan bahwa tahun yang dipilih aktor tidak valid
2. Kembali ke basic flow langkah ke 3

9. PreConditions

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

Program Studi Teknik Informatika	SKPL – SIPETIK	190/ 274
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. Data laporan transaksi penjualan tiket per bulan dalam tahun tertentu telah dicetak

Use case Spesification : Penyediaan Laporan Transaksi Penjualan Tiket berdasarkan Agent

1. Brief Description

Use case ini digunakan oleh aktor untuk menyediakan laporan penjualan tiket berdasarkan Agent.

2. Primary Aktor

1. Manager

3. Supporting Aktor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pencetakan hasil penjualan tiket berdasarkan Agent
2. Sistem menampilkan antarmuka untuk memilih tahun yang ingin dilihat hasil penjualan tiketnya
3. Aktor menginputkan tahun yang ingin disediakan laporan penjualan tiket berdasarkan Agent
E-1 Tahun yang diinputkan aktor salah
4. Sistem menampilkan antarmuka yang berisi laporan penjualan tiket berdasarkan Agent.
5. Aktor menekan button print
6. Sistem melakukan pencetakan laporan penjualan tiket sesuai dengan tahun yang dipilih aktor
7. Use case ini selesai

Program Studi Teknik Informatika	SKPL – SIPETIK	191/ 274
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 Range waktu yang dipilih aktor tidak valid

1. Sistem memberikan peringatan bahwa tahun yang dipilih aktor tidak valid
2. Kembali ke basic flow langkah ke 3

7. PreConditions

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

8. PostConditions

1. Data laporan transaksi penjualan tiket berdasarkan agent tertentu

Use case Spesification : Pengelolaan Transaksi Penjualan Tiket

1. Brief Description

Use case ini digunakan oleh aktor untuk melakukan pengelolaan terhadap transaksi penjualan tiket.

2. Primary Aktor

1. PetugasCounter

3. Supporting Aktor

none

4. Basic Flow

8. Use Case ini dimulai ketika aktor memilih untuk mengelola transaksi penjualan tiket
9. Sistem menampilkan menu utama form transaksi penjualan tiket
10. Aktor menginputkan data penjualan pada form yang ditampilkan sistem

Program Studi Teknik Informatika	SKPL – SIPETIK	192/ 274
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		

11. Aktor mengindikasikan menyimpan data penjualan tiket yang telah diinputkan
12. Sistem melakukan validasi input data penjualan
E-1 Data penjualan yang diinputkan salah
13. Sistem menyimpan data penjualan yang diinputkan aktor ke database
14. Use case ini selesai

5. Alternative Flow

none

6. Error Flow

E-1 Username baru yang dimasukkan aktor sudah terdaftar

1. Sistem memberikan peringatan bahwa input data penjualan salah
2. Kembali ke basic flow langkah ke 6

7. PreConditions

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

8. PostConditions

1. Data transaksi penjualan pada basis data telah terupdate

4.1.12 Use case Spesification : Validasi Tiket

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan validasi tiket untuk akses masuk ke area Monkey Forest

3. Primary Aktor

1. PetugasGate

Program Studi Teknik Informatika	SKPL – SIPETIK	193/ 274
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. Supporting Aktor

none

5. Basic Flow

1. Use case ini dimulai ketika aktor memilih untuk melakukan validasi tiket
2. Sistem menampilkan form validasi
3. Aktor memasukkan card ke reader
4. Sistem melakukan validasi terhadap data yang tersimpan dalam card
5. Sistem menampilkan keterangan card valid atau tidak valid
6. Use case ini selesai

6. Alternative Flow

none

7. Error Flow

none

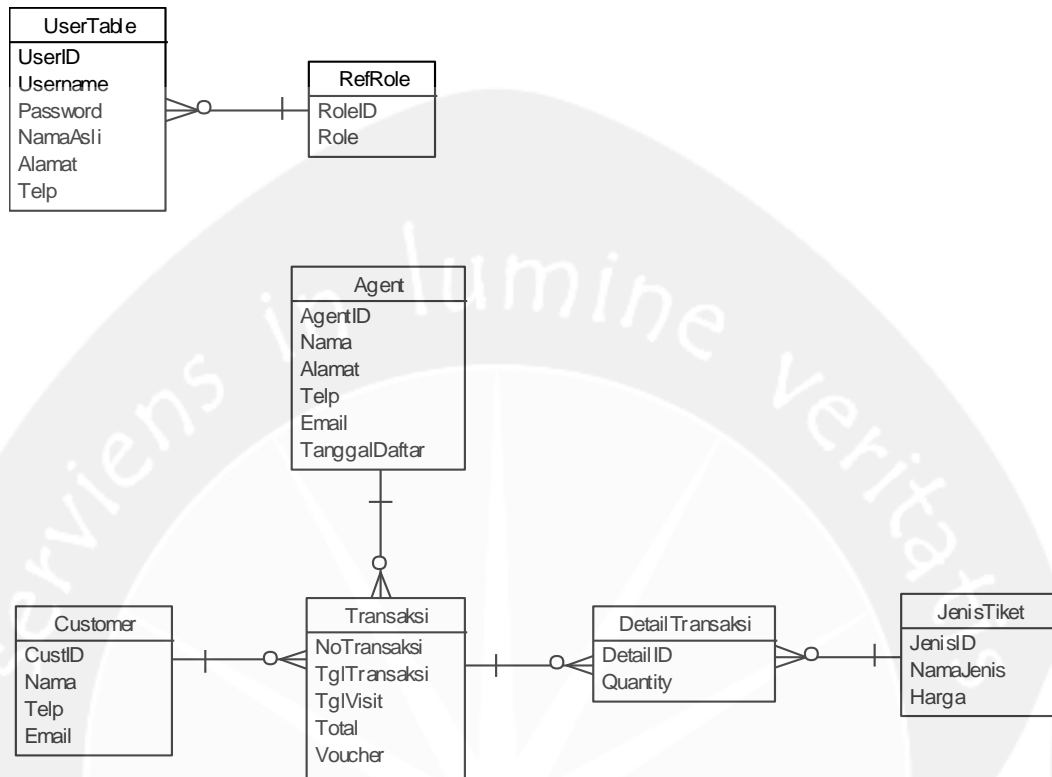
8. PreConditions

1. User telah melakukan use case login dan telah memasuki sistem

9. PostConditions

1. Diketahui bahwa card pengunjung valid atau tidak

Entity Relationship Diagram (ERD)



Gambar 4 Entity Relationship Diagram

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SIPETIK

(Sistem Informasi Penjualan Tiket dengan
Memanfaatkan Teknologi *Radio Frequency
Identification (RFID)*)

Untuk :


Mandara Wisata Wenara Wana Monkey Forest
Ubud, Bali

Dipersiapkan oleh:

Gusti Ayu Sinta Adnyani / 090706029

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

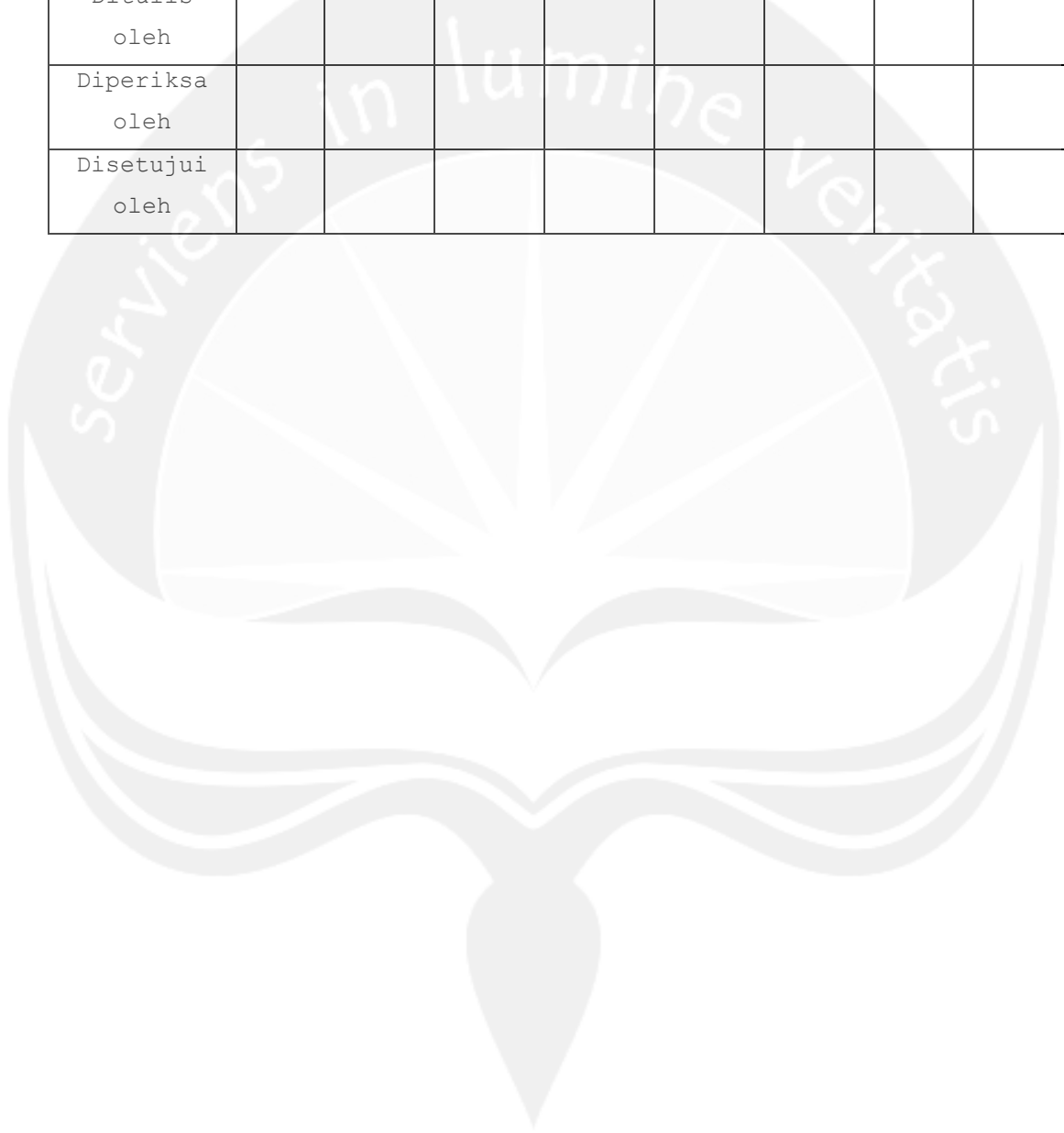
Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		DPPL-SIPETIK		1/56
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

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	9
1.1. Tujuan	9
1.2. Ruang Lingkup	9
1.3. Definisi dan Akronim	9
1.4. Referensi	10
2. Perancangan Sistem	11
2.1. Perancangan Arsitektur	11
2.2. Perancangan Rinci	12
2.2.1. Sequence Diagram	12
2.2.1.1. <i>Login</i>	12
2.2.1.2. Ubah Username	12
2.2.1.3. Ubah <i>Password</i>	13
2.2.1.4. Pengelolaan Data Pengguna	13
2.2.1.4.1. Tambah Data Pengguna	13
2.2.1.4.2. Ubah Data Pengguna	14
2.2.1.4.3. Hapus Data Pengguna	14
2.2.1.4.4. <i>Display</i> Data Pengguna	15
2.2.1.5. Pengelolaan Data Agent	15
2.2.1.5.1. Tambah Data Agent	15
2.2.1.5.2. Edit Data Agent	16
2.2.1.5.3. <i>Delete</i> Data Agent	16
2.2.1.5.4. <i>Display</i> Data Agent	17
2.2.1.6. Pengelolaan Data Customer	17
2.2.1.6.1. Tambah Data Customer	12
2.2.1.6.2. Edit Data Customer	18
2.2.1.6.3. <i>Delete</i> Data Customer	18
2.2.1.6.4. <i>Display</i> Data Customer	19
2.2.1.7. Pengelolaan Data Tipe Tiket	19
2.2.1.7.1. Tambah Data Tipe Tiket	19
2.2.1.7.2. Edit Data Tipe Tiket	20
2.2.1.7.3. <i>Delete</i> Data Tipe Tiket	20
2.2.1.7.4. <i>Display</i> Data Tipe Tiket	21
2.2.1.8. Pengelolaan Transaksi	21
2.2.1.8.1. <i>Entry</i> Transaksi	21
2.2.1.8.2. Edit Transaksi	22
2.2.1.8.3. <i>Delete</i> Transaksi	22
2.2.1.8.4. <i>Display</i> Transaksi	23
2.2.1.9. Pengelolaan Laporan Transaksi	23
2.2.1.9.1. Laporan transaksi Periode tertentu	23
2.2.1.9.2. Laporan Transaksi per Bulan pada Tahun tertentu	24
2.2.1.9.3. Laporan Transaksi berdasarkan Agent tertentu	24

2.2.1.10.	Validasi Tiket	25
2.2.2.	Class Diagram	26
2.2.3.	Spesifikasi Deskripsi Kelas Diagram	27
2.2.3.1.	Spesifikasi Design Kelas <i>LoginUI</i>	27
2.2.3.2.	Spesifikasi Design Kelas PengelolaanPegguna	27
2.2.3.3.	Spesifikasi Design Kelas <i>UbahUsername</i>	27
2.2.3.4.	Spesifikasi Design Kelas <i>UbahPassword</i>	28
2.2.3.5.	Spesifikasi Design Kelas PengelolaanAgent	28
2.2.3.6.	Spesifikasi Design Kelas PengelolaanCustomer	29
2.2.3.7.	Spesifikasi Design Kelas PengelolaanTipeTiket	29
2.2.3.8.	Spesifikasi Design Kelas PengelolaanTransaksi	30
2.2.3.9.	Spesifikasi Design Kelas PengelolaanLaporanTransaksi	30
2.2.3.10.	Spesifikasi Design Kelas <i>ValidasiTiket</i> ..	30
2.2.3.11.	Spesifikasi Design Kelas <i>LoginControl</i>	31
2.2.3.12.	Spesifikasi Design Kelas <i>UserControl</i>	31
2.2.3.13.	Spesifikasi Design Kelas <i>AgentControl</i>	32
2.2.3.14.	Spesifikasi Design Kelas <i>CustomerControl</i> .	33
2.2.3.15.	Spesifikasi Design Kelas <i>TiketControl</i>	34
2.2.3.16.	Spesifikasi Design Kelas PengelolaanTransaksiControl	35
2.2.3.17.	Spesifikasi Design Kelas PengelolaanLaporanTransaksiControl	36
2.2.3.18.	Spesifikasi Design Kelas <i>ValidasiTiketControl</i>	37
2.2.3.19.	Spesifikasi Design Kelas <i>DataUser</i>	37
2.2.3.20.	Spesifikasi Design Kelas <i>DataAgent</i>	38
2.2.3.21.	Spesifikasi Design Kelas <i>DataCustomer</i>	38
2.2.3.22.	Spesifikasi Design Kelas <i>DataTiket</i>	39
2.2.3.23.	Spesifikasi Design Kelas <i>DataTransaksi</i> ..	39
2.2.3.24.	Spesifikasi Design Kelas <i>DataDetailTiket</i>	40
3.	Perancangan Data	40
3.1.	Dekomposisi Data	40
3.1.1.	Deskripsi Entitas Data <i>UserTable</i>	40
3.1.2.	Deskripsi Entitas Data <i>Customer</i>	40
3.1.3.	Deskripsi Entitas Data <i>Agent</i>	41
3.1.4.	Deskripsi Entitas Data <i>Transaksi</i>	41
3.1.5.	Deskripsi Entitas Data <i>DetailTransaksi</i>	41
3.1.6.	Deskripsi Entitas Data <i>JenisTiket</i>	42
3.1.7.	Deskripsi Entitas Data <i>RefRole</i>	42

3.2.	Physical Data Model	43
4.	Perancangan Antarmuka	44
4.1.	Sketsa UI dan Deskripsinya	44
4.1.1.	Antarmuka Halaman <i>Login</i>	44
4.1.2.	Antarmuka Halaman Utama	45
4.1.3.	Antarmuka Halaman Ubah <i>Username</i>	47
4.1.4.	Antarmuka Halaman Ubah <i>Password</i>	48
4.1.5.	Antarmuka Halaman Pengelolaan Data User....	49
4.1.6.	Antarmuka Halaman Pengelolaan Data Agent...	50
4.1.6.	Antarmuka Halaman Pengelolaan Data Customer	51
4.1.6.	Antarmuka Halaman Pengelolaan Data Tiket...	52
4.1.7.	Antarmuka Halaman Transaksi Penjualan	53
4.1.8.	Antarmuka Halaman Validasi Tiket.....	54
4.1.9.	Antarmuka Halaman Cetak Laporan Transaksi Periode tertentu.....	55
4.1.10.	Antarmuka Halaman Cetak Laporan Transaksi per Bulan pada Tahun tertentu	55
4.1.11.	Antarmuka Halaman Cetak Laporan Transaksi berdasarkan Agent.....	56

Daftar Gambar

Gambar 2.1	Rancangan Arsitektur SIPETIK.....	11
Gambar 2.2	Sequence Diagram : <i>Login</i>	12
Gambar 2.3	Sequence Diagram : Ubah Username.....	12
Gambar 2.4	Sequence Diagram : Ubah <i>Password</i>	13
Gambar 2.5	Sequence Diagram : Tambah Data Pengguna	13
Gambar 2.6	Sequence Diagram : Ubah Data Pengguna .	14
Gambar 2.7	Sequence Diagram : Hapus Data Pengguna .	14
Gambar 2.8	Sequence Diagram : <i>Display</i> Data Pengguna	15
Gambar 2.9	Sequence Diagram : Tambah Data Agent ..	15
Gambar 2.10	Sequence Diagram : Edit Data Agent.....	16
Gambar 2.11	Sequence Diagram : <i>Delete</i> Data Agent...	16
Gambar 2.12	Sequence Diagram : <i>Display</i> Data Agent..	17
Gambar 2.13	Sequence Diagram : Tambah Data Customer	17
Gambar 2.14	Sequence Diagram : Edit Data Customer..	18
Gambar 2.15	Sequence Diagram : <i>Delete</i> Data Customer	18
Gambar 2.16	Sequence Diagram : <i>Display</i> Data Customer	19
Gambar 2.17	Sequence Diagram : Tambah Data Tiket...	19
Gambar 2.18	Sequence Diagram : Edit Data Tipe Tiket	20
Gambar 2.19	Sequence Diagram : <i>Delete</i> Data Tiket...	20
Gambar 2.20	Sequence Diagram : <i>Display</i> Data Tiket..	21
Gambar 2.21	Sequence Diagram : <i>Entry</i> Transaksi.....	21
Gambar 2.22	Sequence Diagram : Edit Transaksi	22
Gambar 2.23	Sequence Diagram : <i>Delete</i> Transaksi.....	22
Gambar 2.24	Sequence Diagram : <i>Display</i> Transaksi...	23
Gambar 2.25	Sequence Diagram : Laporan Transaksi Periode tertentu	23
Gambar 2.26	Sequence Diagram :Laporan Transaksi per Bulan pada Tahun tertentu	24
Gambar 2.27	Sequence Diagram : Laporan Transaksi berdasarkan Agent	24
Gambar 2.28	Sequence Diagram : Validasi Tiket.....	25
Gambar 2.29	Class Diagram	26
Gambar 3.1	Physical Data Model	43
Gambar 4.1	Rancangan Antarmuka <i>Login</i>	44
Gambar 4.2	Rancangan Antarmuka Halaman Utama	45
Gambar 4.2.1	Rancangan Antarmuka Halaman Utama- Tab Home	45
Gambar 4.2.2	Rancangan Antarmuka Halaman Utama- Tab <i>Manage</i>	46
Gambar 4.2.3	Rancangan Antarmuka Halaman Utama- Tab <i>Transaction</i>	46
Gambar 4.3	Rancangan Antarmuka Ubah <i>Username</i>	47
Gambar 4.4	Rancangan Antarmuka Ubah <i>Password</i>	48
Gambar 4.5	Rancangan Antarmuka Pengelolaan Data User	49

Gambar 4.6	Rancangan Antarmuka Pengelolaan Data Agent	50
Gambar 4.7	Rancangan Antarmuka Pengelolaan Data Customer	51
Gambar 4.8	Rancangan Antarmuka Pengelolaan Data Tipe Tiket	52
Gambar 4.9	Rancangan Antarmuka Transaksi Penjualan Tiket	53
Gambar 4.10	Rancangan Antarmuka Validasi Tiket	54
Gambar 4.11	Rancangan Antarmuka Cetak Laporan Transaksi Periode tertentu	55
Gambar 4.11	Rancangan Antarmuka Cetak Laporan Transaksi per Bulan di Tahun tertentu ..	55
Gambar 4.12	Rancangan Antarmuka Cetak Laporan Transaksi berdasarkan Agent	56

Pendahuluan

Tujuan

Tujuan dari dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) SIPETIK (Sistem Informasi Penjualan Tiket dengan Memanfaatkan Teknologi *Radio Frequency Identification (RFID)*) adalah untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL ini digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

Ruang Lingkup

Perangkat Lunak SIPETIK dikembangkan dengan tujuan untuk:

7. Menyediakan sistem informasi penjualan tiket (*ticketing*) yang mampu mencegah penyalahgunaan peruntukan tiket.
8. Menyediakan sistem informasi penjualan tiket yang memberi kemudahan bagi user dalam mencatat transaksi, baik yang dilakukan secara langsung di lokasi objek wisata, melalui email maupun melalui *agent*.
9. Menyediakan pelaporan (*reporting*) terkait dengan transaksi penjualan tiket tersebut.

Perangkat lunak ini berjalan pada lingkungan dengan platform Windows untuk aplikasi desktop dan web.

Definisi dan Akronim

Daftar definisi akronim dan singkatan:

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak

Program Studi Teknik Informatika	DPPL – SIPETIK	205/ 274
----------------------------------	----------------	----------

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		
--	--	--

	disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.
DPPL-SIPETIK-XXX	Kode yang merepresentasikan kebutuhan pada SIPETIK (Sistem Informasi Penjualan Tiket dengan Memanfaatkan Teknologi <i>Radio Frequency Identification (RFID)</i>) dimana XXX merupakan nomor fungsi produk.
SIPETIK	Perangkat lunak yang digunakan untuk membantu manajemen Mandara Wisata Wenara Wana Monkey Forest terkait dengan penjualan tiket.
<i>RFID</i>	<i>RFID</i> adalah singkatan dari <i>Radio Frequency Identification</i> yang merupakan proses identifikasi seseorang atau objek menggunakan frekuensi transmisi radio.

Referensi

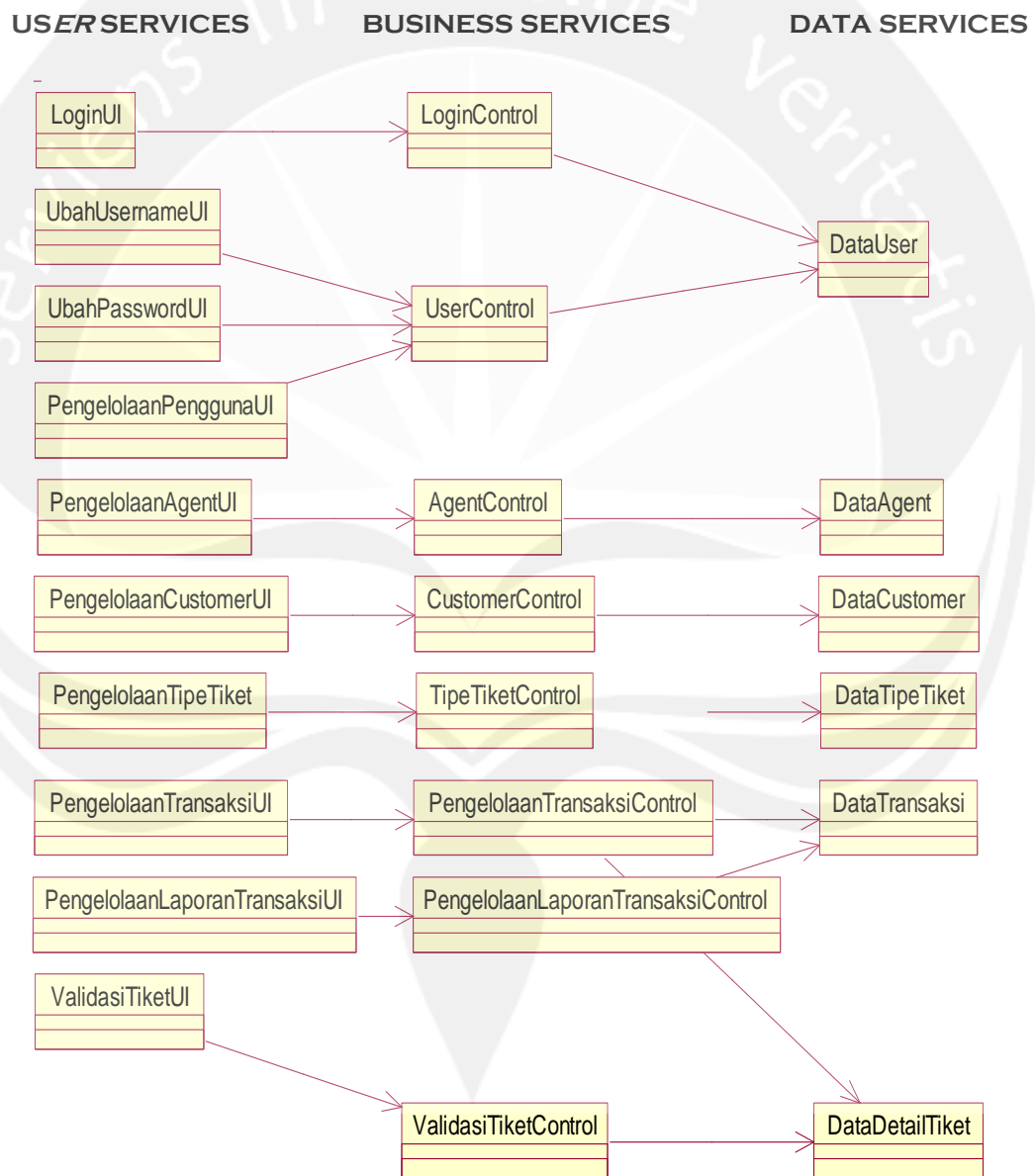
Referensi yang digunakan pada perangkat lunak tersebut adalah:

4. Adnyani, Gusti Ayu Sinta, *Spesifikasi Kebutuhan Perangkat Lunak SIPETIK*, Universitas Atma Jaya Yogyakarta, 2013, Yogyakarta.
5. Purnomo, Andreanus Agung, *Deskripsi Perancangan Perangkat Lunak SIGPosyandu*, Universitas Atma Jaya Yogyakarta, 2009, Yogyakarta.

6. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.

Perancangan Sistem

Perancangan Arsitektur

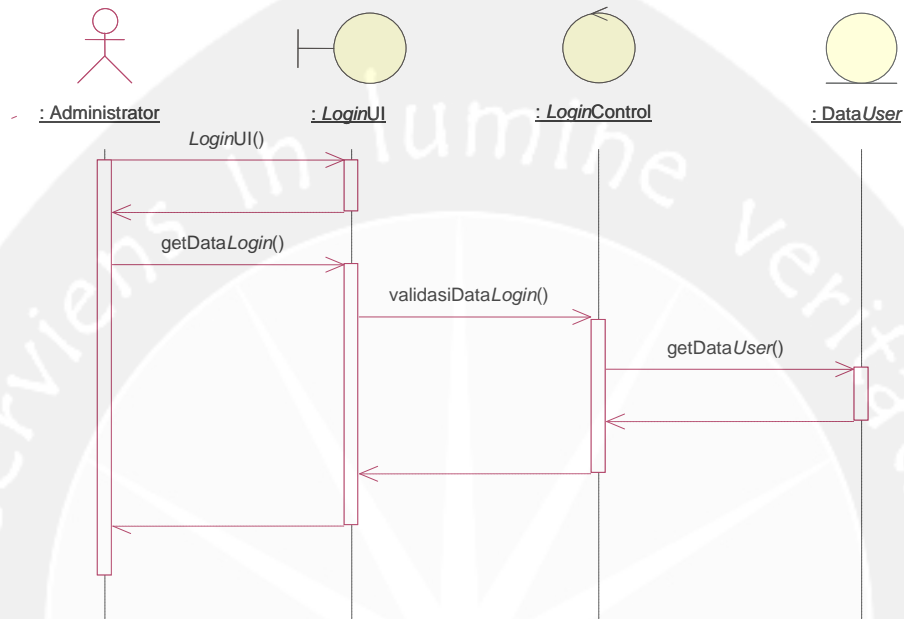


Gambar 2.1 Rancangan Arsitektur SIPETIK

Perancangan Rinci

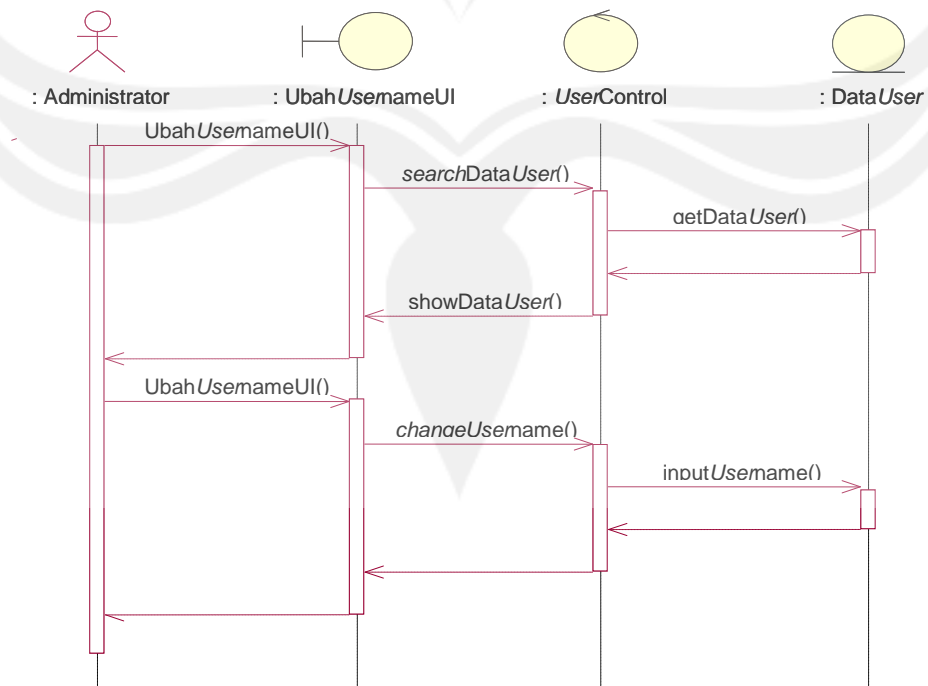
Sequence Diagram

Login



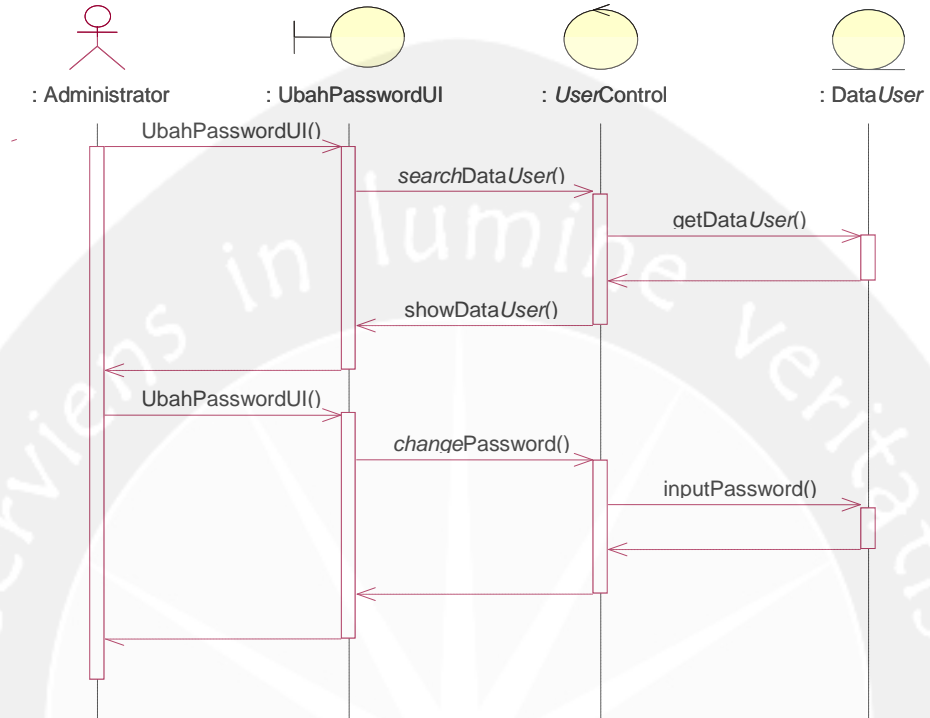
Gambar 2.2 Sequence Diagram: Login

Ubah Username



Gambar 2.3 Sequence Diagram: Ubah Username

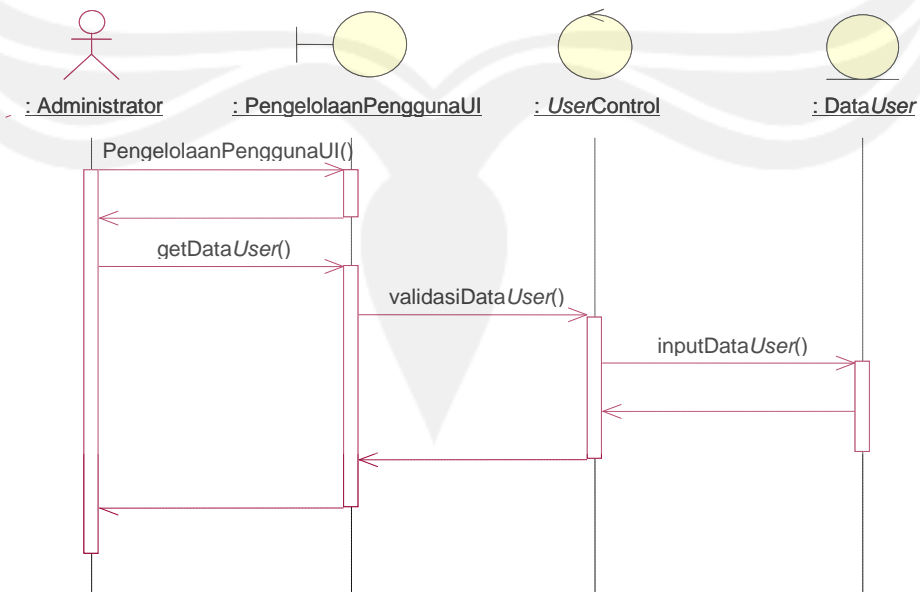
Ubah Password



Gambar 2.4 Sequence Diagram: Ubah Password

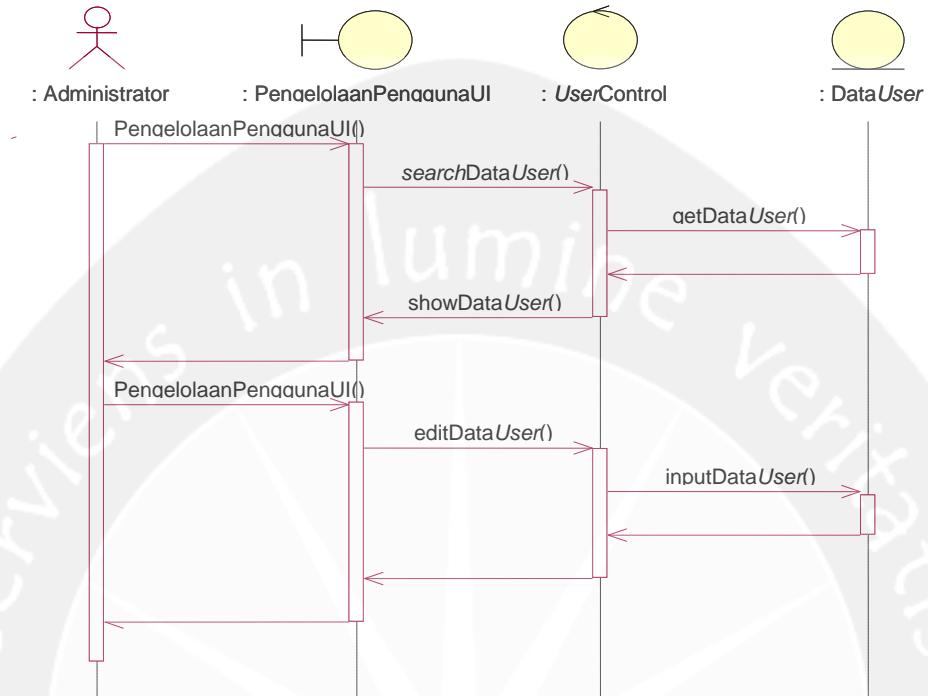
Pengelolaan Data Pengguna

Tambah Data Pengguna



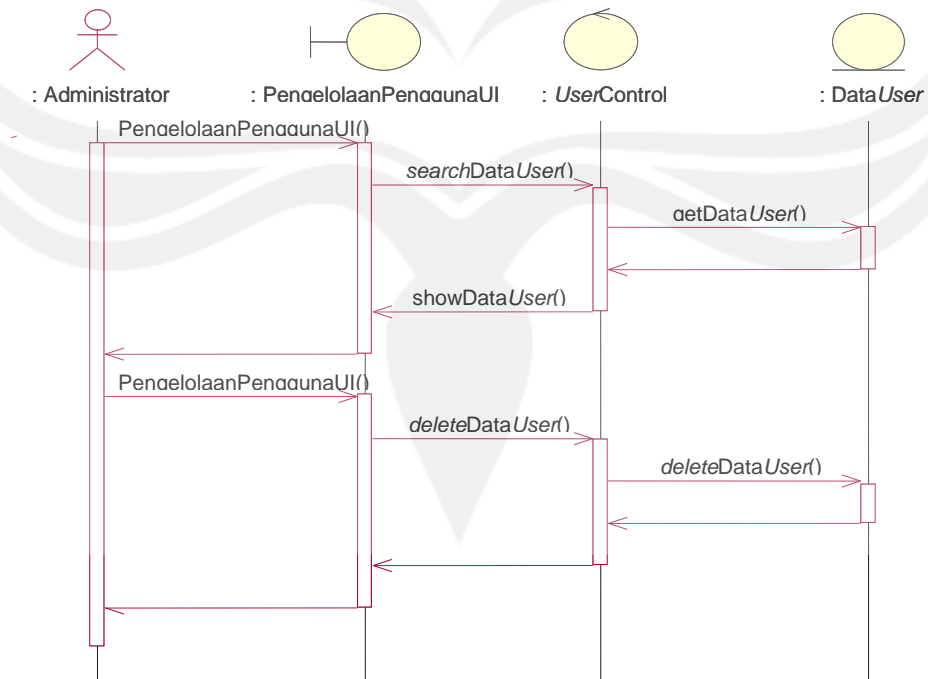
Gambar 2.5 Sequence Diagram: Tambah Data Pengguna

Ubah Data Pengguna



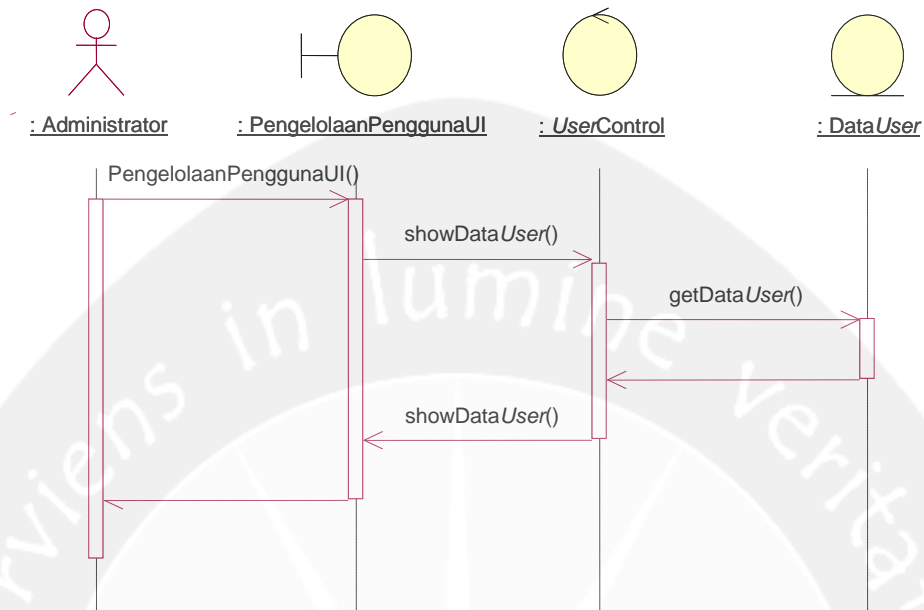
Gambar 2.6 Sequence Diagram: Ubah Data Pengguna

Hapus Data Pengguna



Gambar 2.7 Sequence Diagram: Hapus Data Pengguna

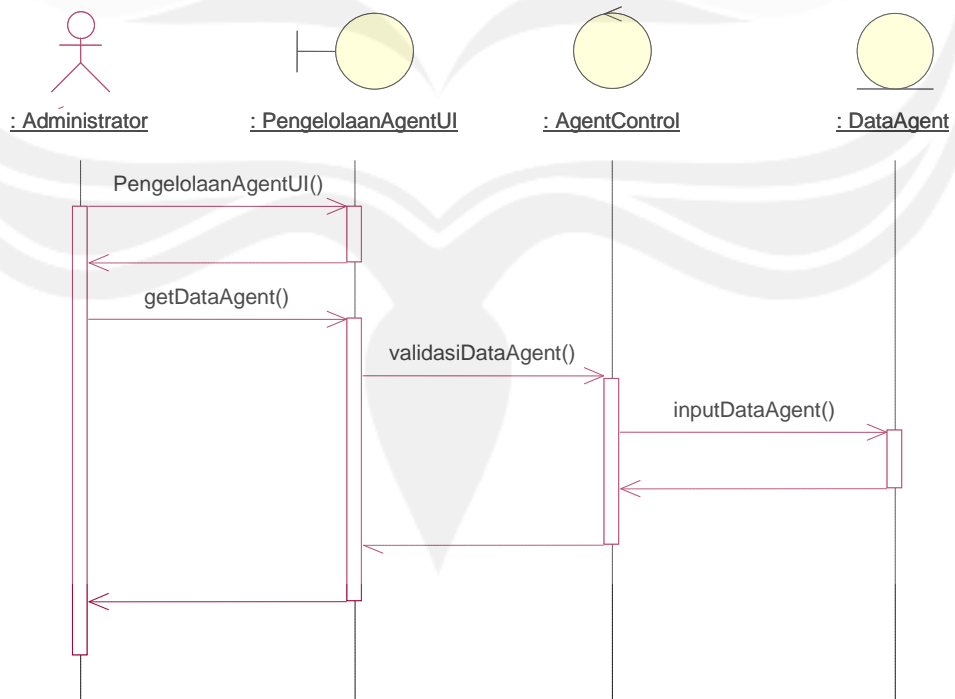
Display Data Pengguna



Gambar 2.8 Sequence Diagram: *Display Data Pengguna*

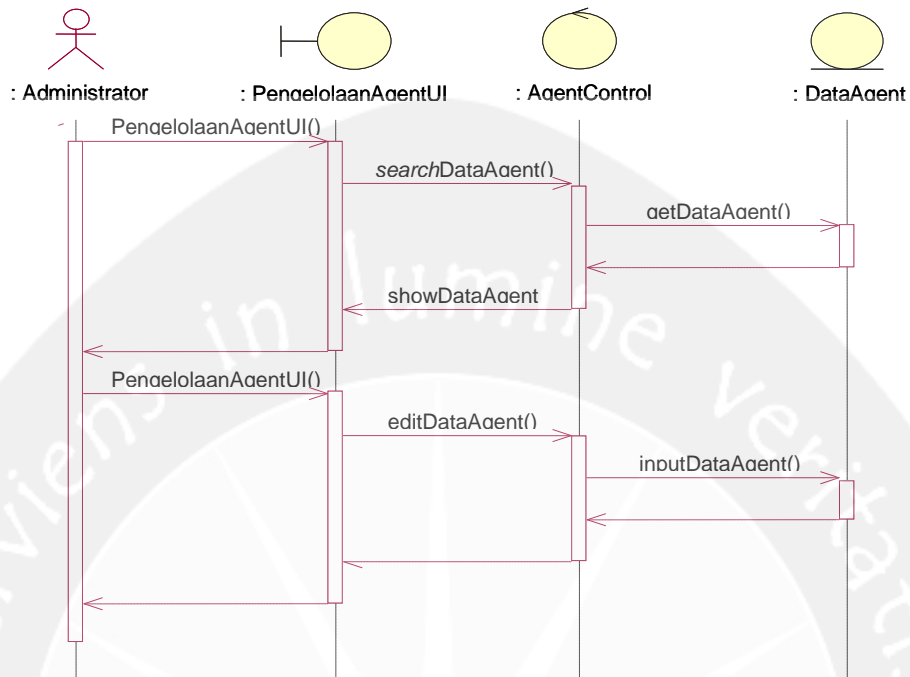
Pengelolaan Data Agent

2.2.1.5.1 Tambah Data Agent



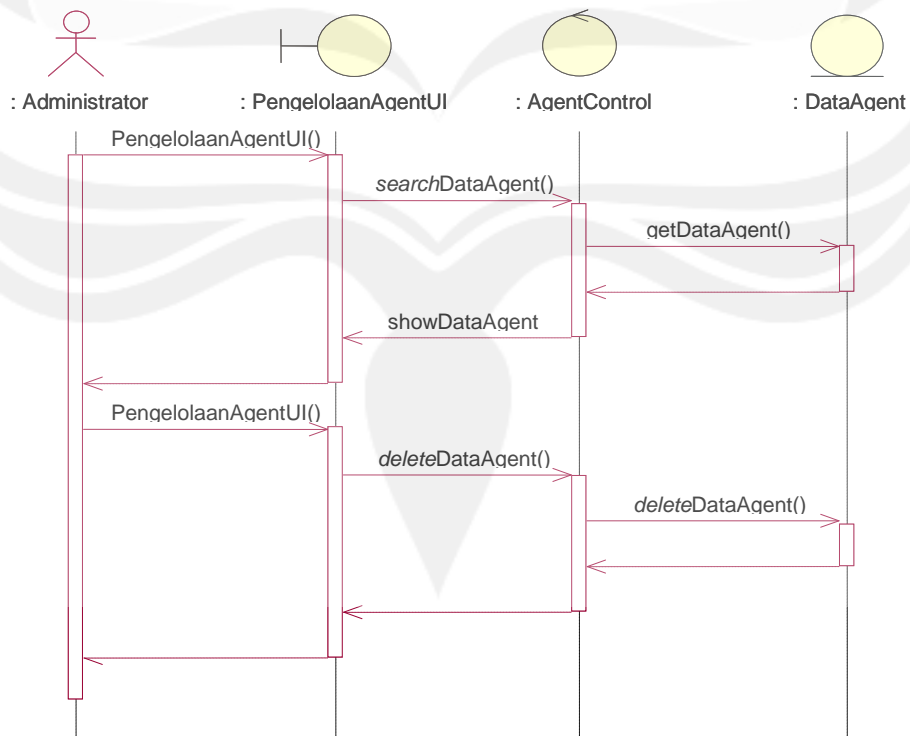
Gambar 2.9 Sequence Diagram: *Tambah Data Agent*

Edit Data Agent



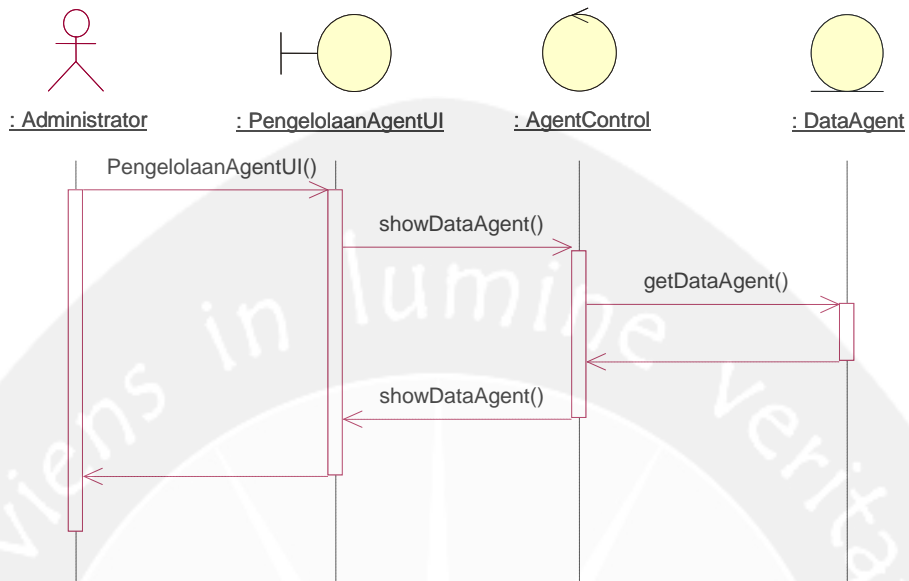
Gambar 2.10 Sequence Diagram: Edit Data Agent

Delete Data Agent



Gambar 2.11 Sequence Diagram: Delete Data Agent

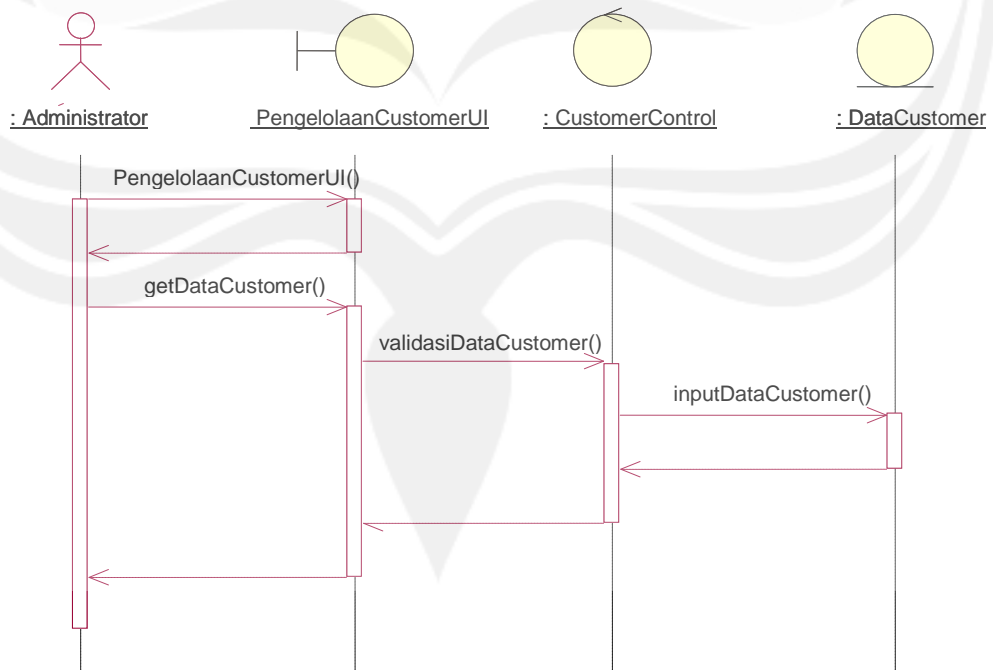
Display Data Agent



Gambar 2.12 Sequence Diagram: *Display Data Agent*

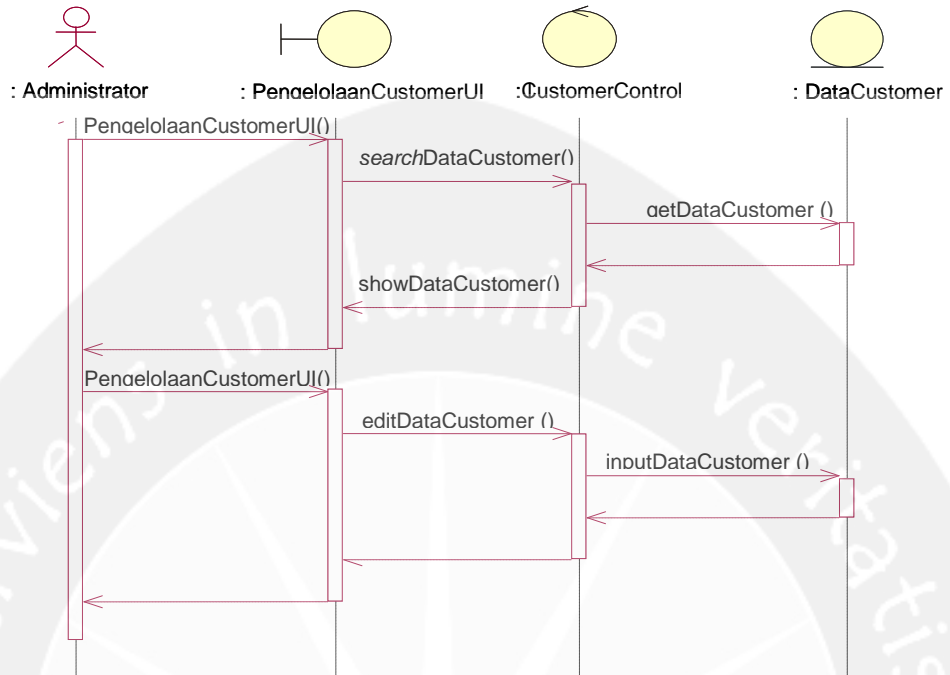
Pengelolaan Data Customer

2.2.1.6.1 Tambah Data Customer



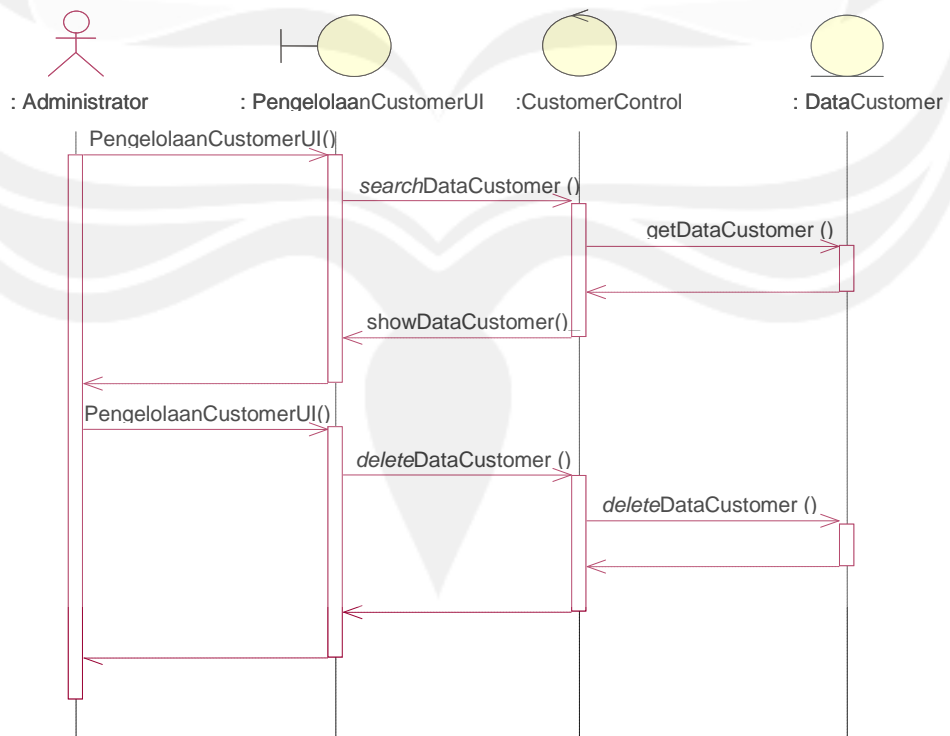
Gambar 2.13 Sequence Diagram: *Tambah Data Customer*

Edit Data Customer



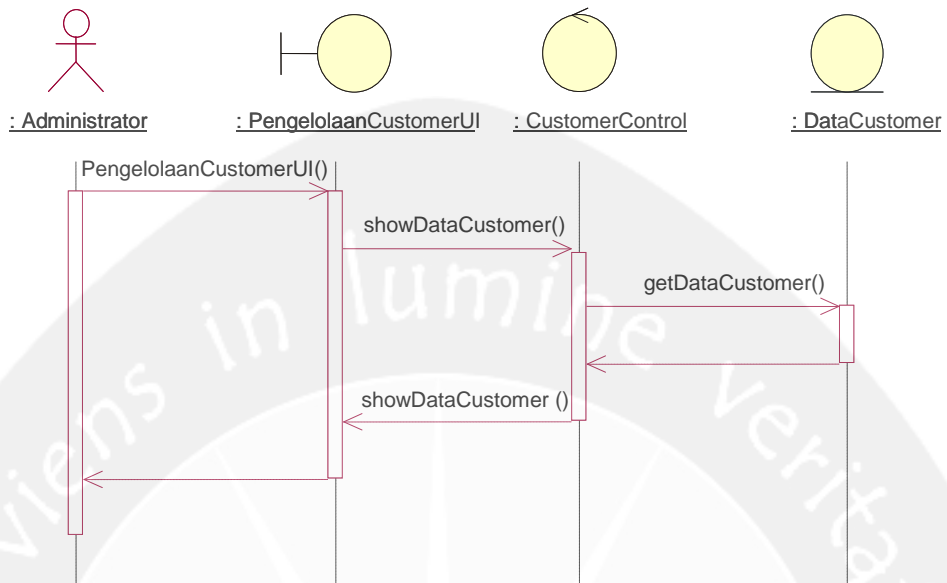
Gambar 2.14 Sequence Diagram: Edit Data Customer

Delete Data Customer



Gambar 2.15 Sequence Diagram: Delete Data Agent

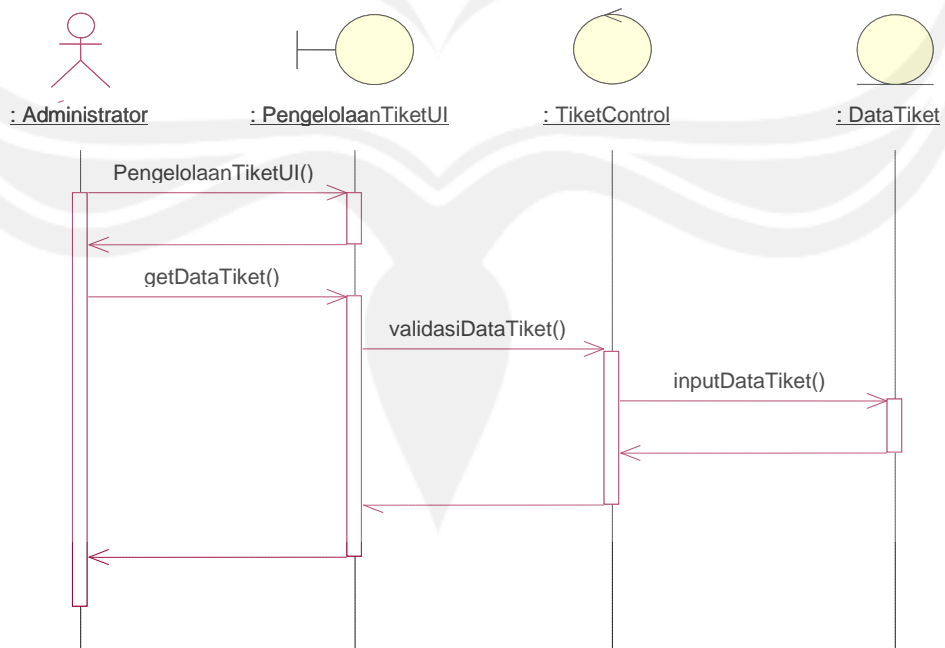
Display Data Customer



Gambar 2.16 Sequence Diagram: Display Data Agent

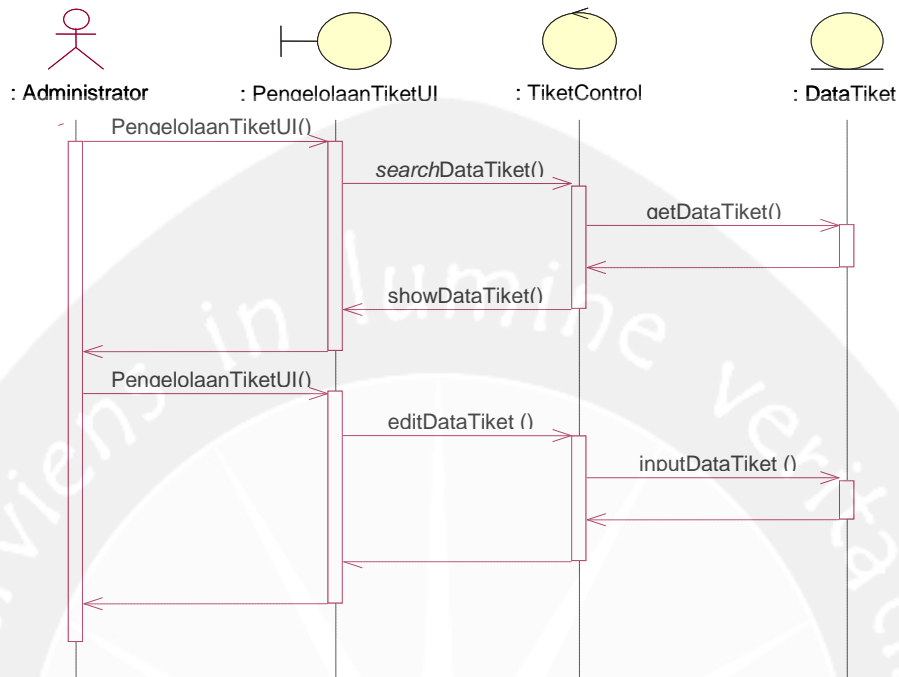
Pengelolaan Data Tipe Tiket

2.2.1.7.1 Tambah Data Tipe Tiket



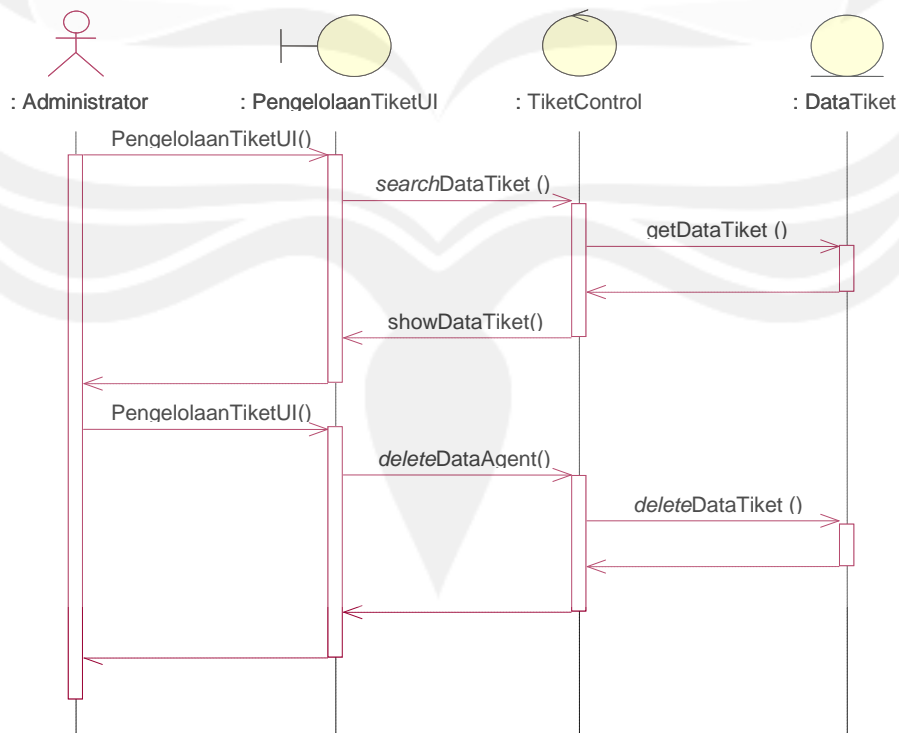
Gambar 2.17 Sequence Diagram: Tambah Data Agent

Edit Data Tipe Tiket



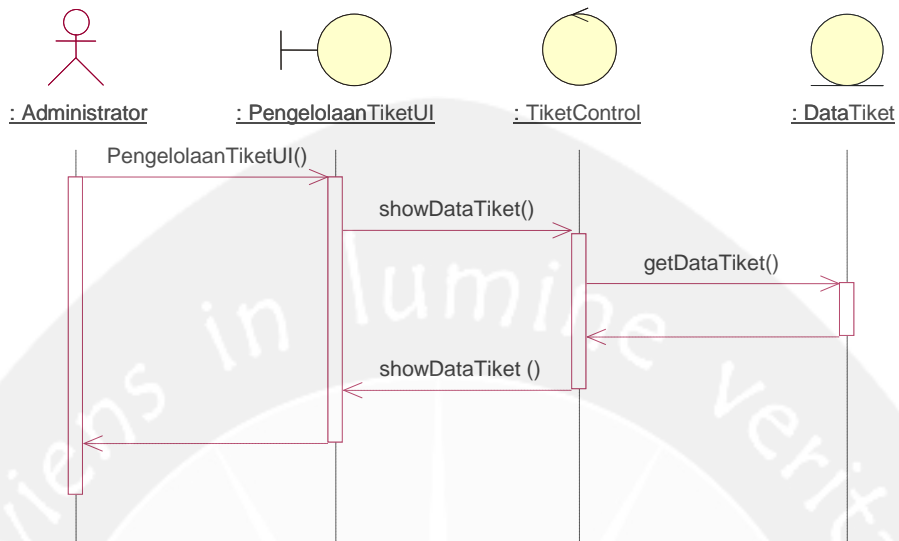
Gambar 2.18 Sequence Diagram: Edit Data Agent

Delete Data Tipe Tiket



Gambar 2.19 Sequence Diagram: Delete Data Agent

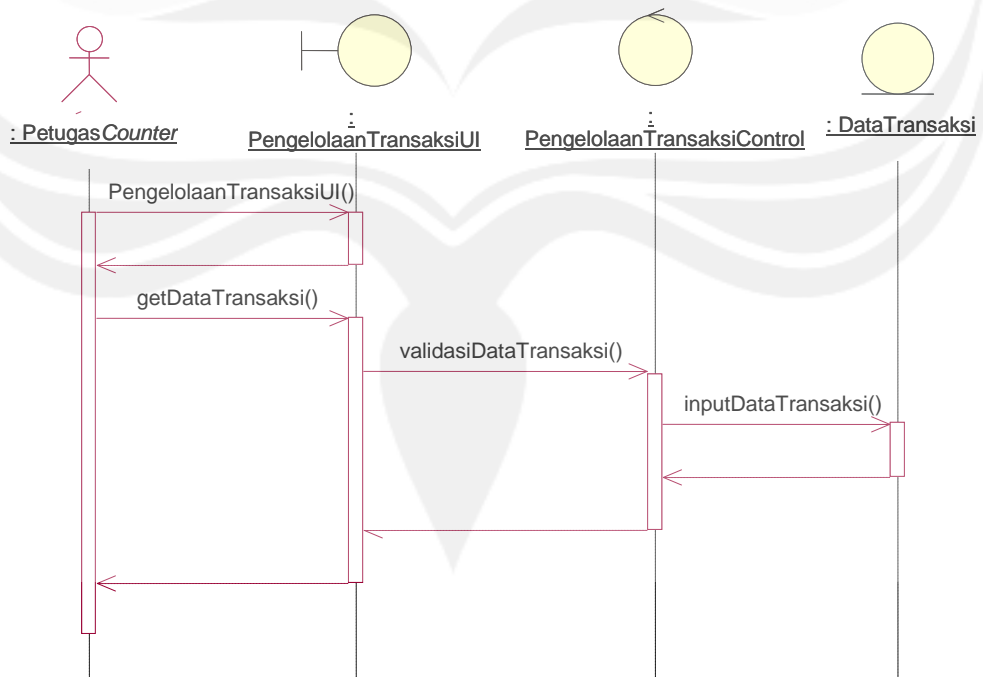
Display Data Tiket



Gambar 2.20 Sequence Diagram: *Display Data Agent*

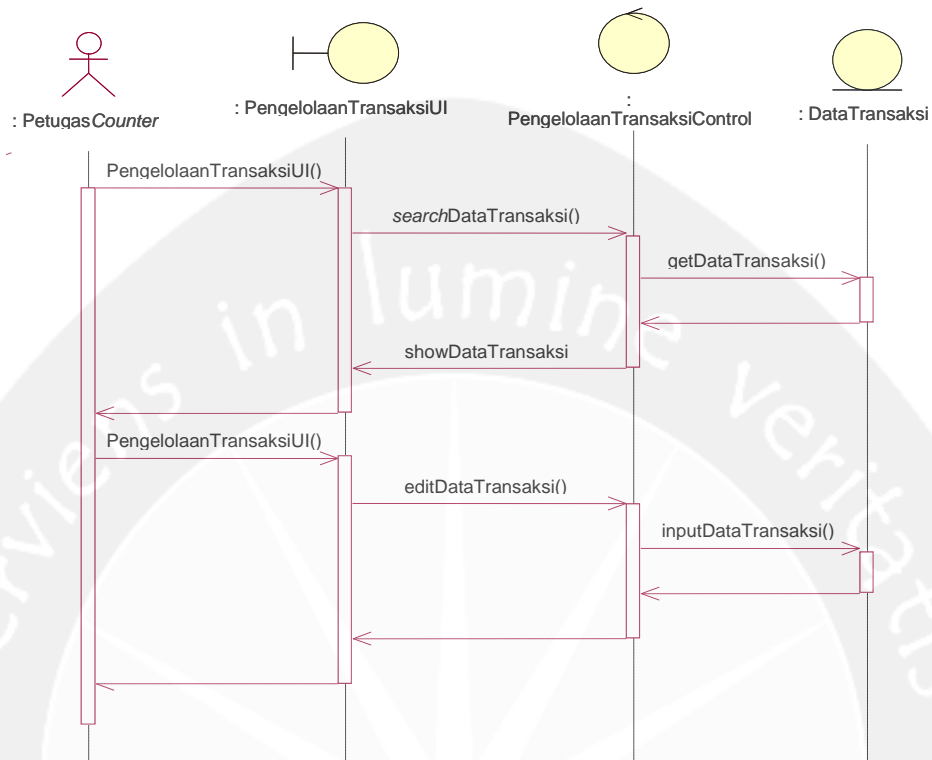
Pengelolaan Transaksi

Entry Transaksi



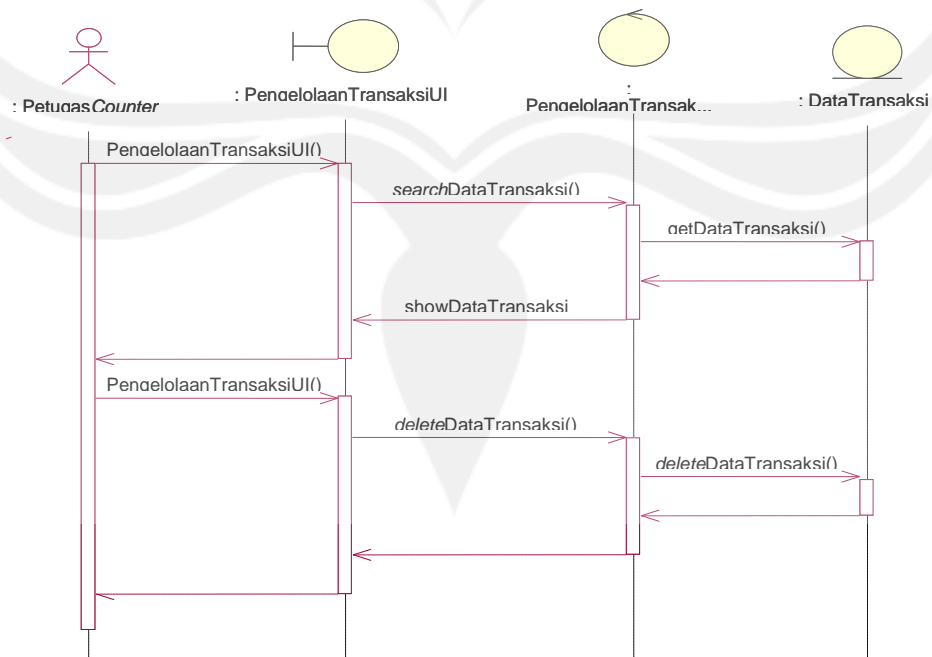
Gambar 2.21 Sequence Diagram: *Entry Transaksi*

Edit Transaksi



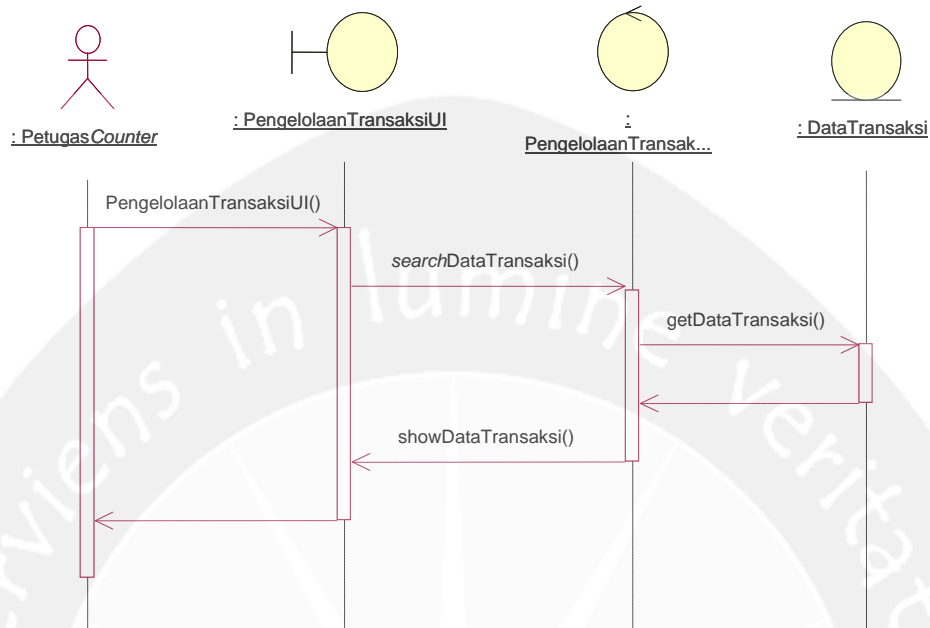
Gambar 2.22 Sequence Diagram: Edit Transaksi

Delete Transaksi



Gambar 2.23 Sequence Diagram: Delete Transaksi

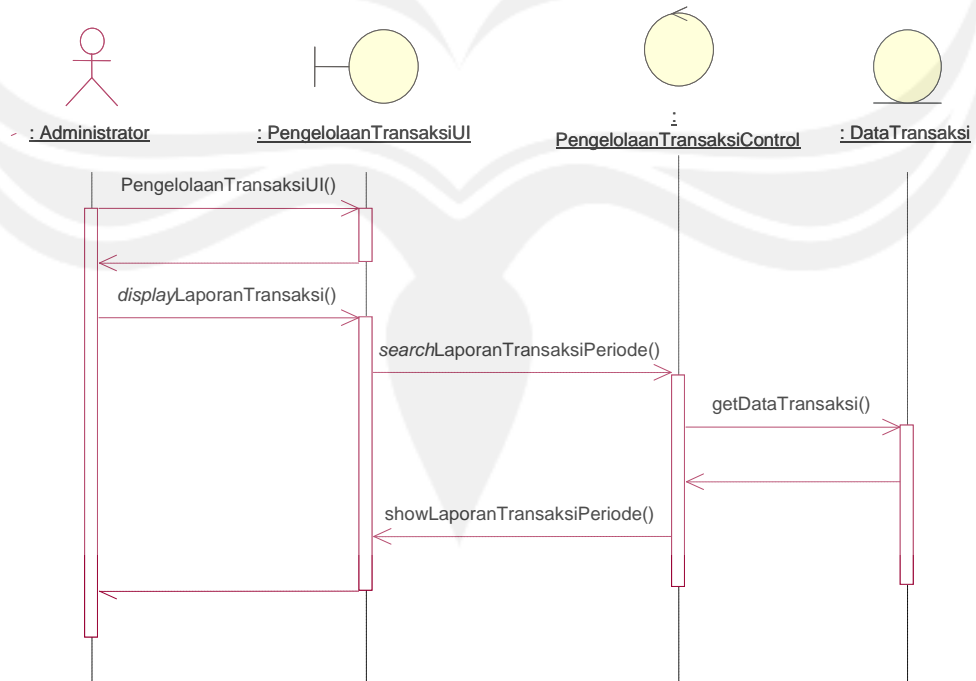
Display Transaksi



Gambar 2.24 Sequence Diagram: *Display Transaksi*

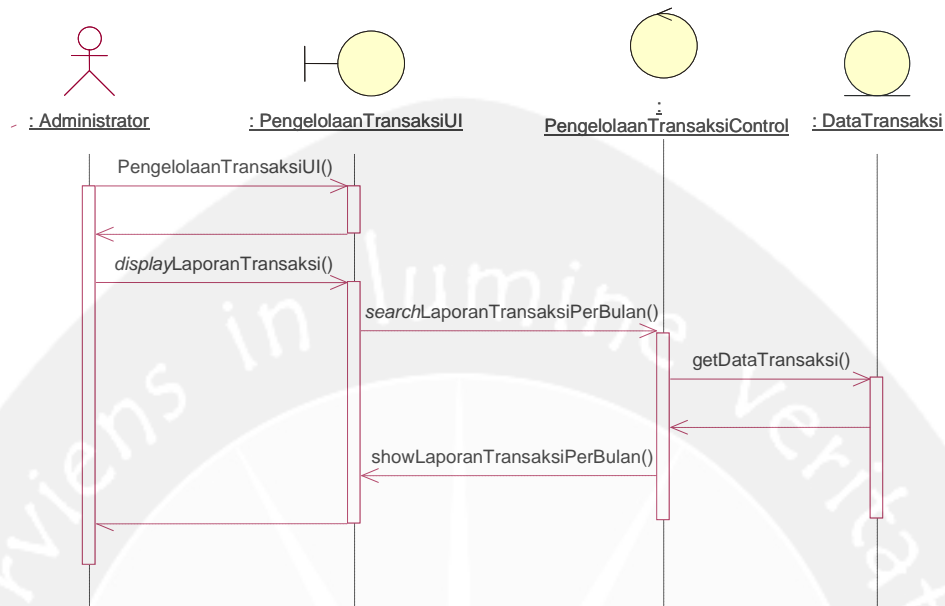
Penyediaan Laporan Transaksi

Laporan Transaksi Periode tertentu



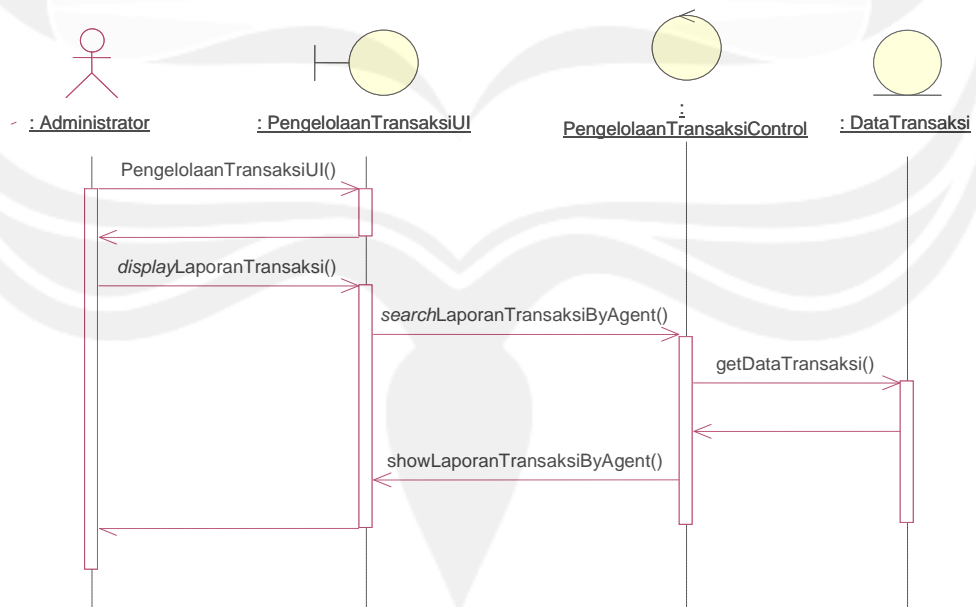
Gambar 2.25 Sequence Diagram: Laporan Transaksi Periode tertentu

Laporan Transaksi per Bulan pada Tahun tertentu



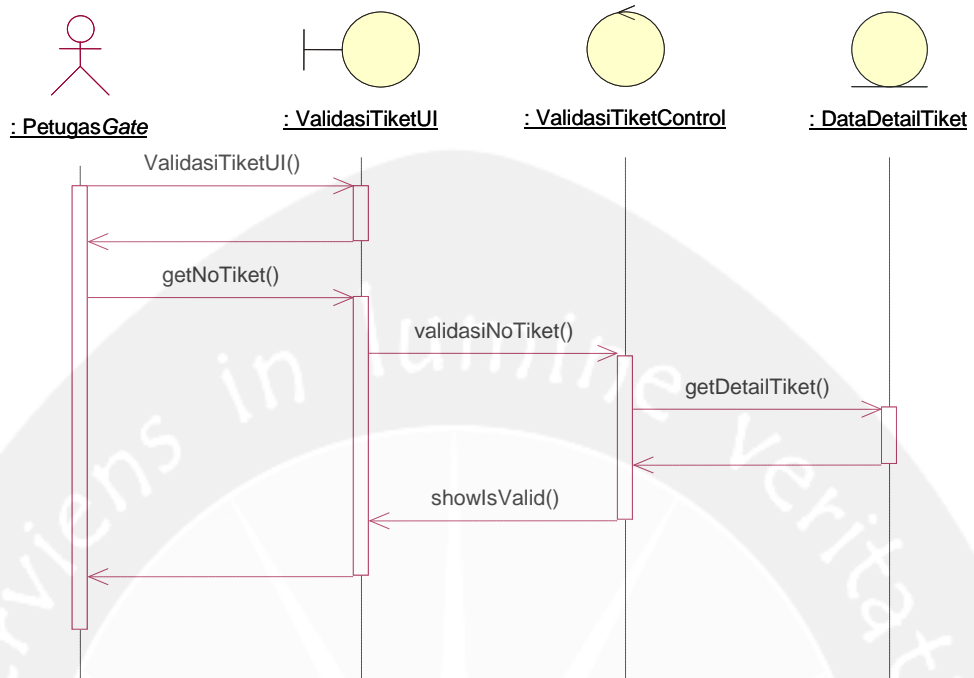
Gambar 2.26 Sequence Diagram: Laporan Transaksi per Bulan pada Tahun tertentu

Laporan Transaksi berdasarkan Agent tertentu



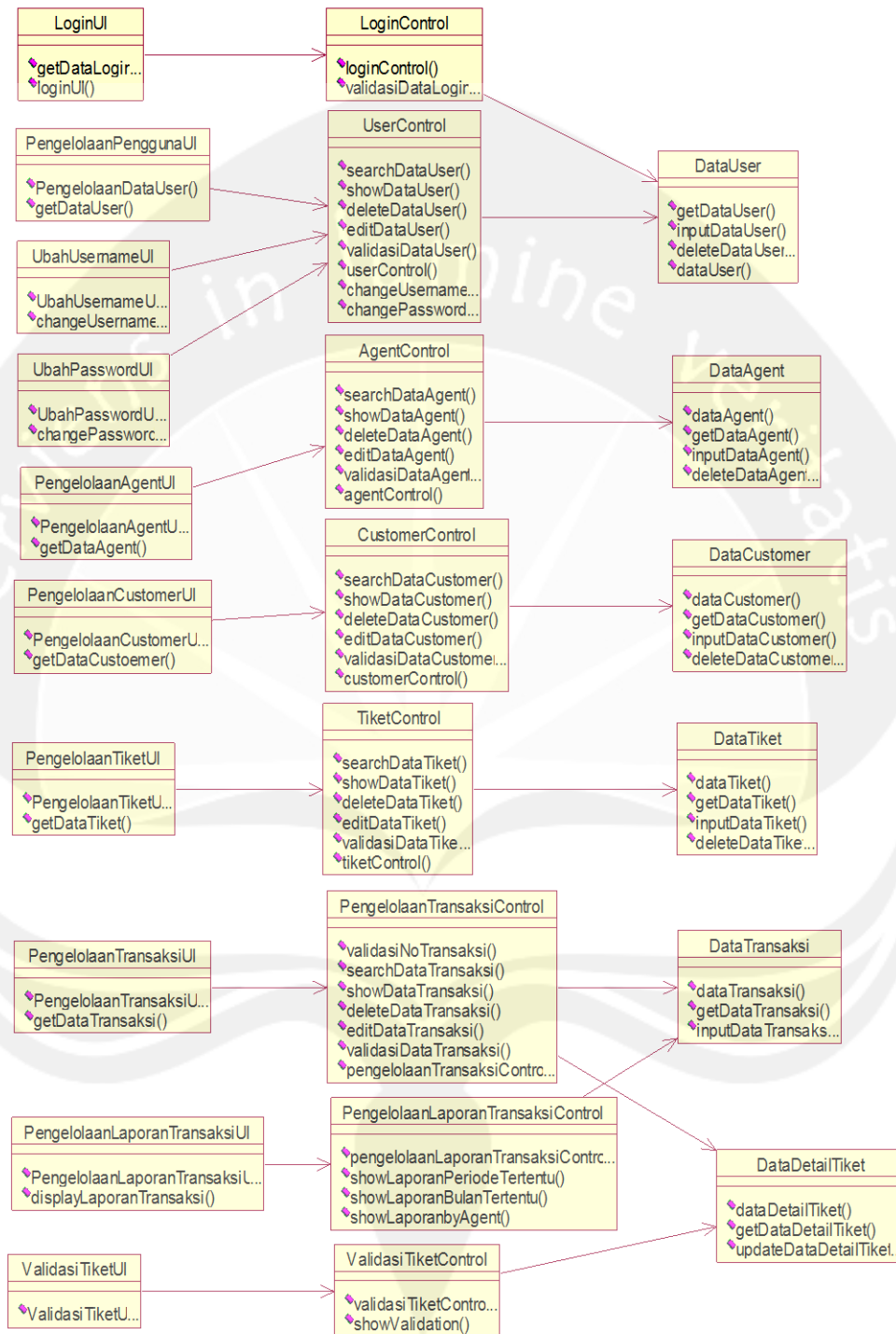
Gambar 2.27 Sequence Diagram: Laporan Transaksi berdasarkan Agent tertentu

2.2.1.10 Validasi Tiket



Gambar 2.28 Sequence Diagram: Validasi Tiket

2.1.1 Class Diagram



Gambar 2.29 Class Diagram

2.2.3 Spesifikasi Deskripsi Kelas Diagram

2.2.3.1 Spesifikasi Design Kelas LoginUI

LoginUI	<<boundary>>
<pre>+loginUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataLogin(): void Operasi ini digunakan untuk mengambil data login yang diinputkan oleh user, yaitu name, password dan role user.</pre>	

2.2.3.2 Spesifikasi Design Kelas PengelolaanPegguna

PengelolaanPegguna	<<boundary>>
<pre>+pengelolaanPegguna() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataUser() Operasi ini digunakan untuk mengambil data user yang diinputkan oleh administrator, yaitu username, nama, password, alamat, noTelp.</pre>	

2.2.3.3 Spesifikasi Design Kelas UbahUsername

UbahUsername	<<boundary>>
<pre>+ubahUsername() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +changeUsername(string idUser, string username) Operasi ini digunakan untuk mencari data pengguna sesuai dengan idUser pengguna untuk diganti dengan username baru.</pre>	

2.2.3.4 Spesifikasi Design Kelas *UbahPassword*

<i>UbahPassword</i>	<<boundary>>
<pre>+ubahPasswordUI() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +changePassword(string idUser, string oldPass, string newPass) Operasi ini digunakan untuk mencari data pengguna sesuai dengan idUser pengguna untuk diganti dengan password baru dengan terlebih dahulu dicek apakah password lama input pengguna telah sesuai dengan password yang ada di dalam basis data.</pre>	

2.2.3.5 Spesifikasi Design Kelas *PengelolaanAgent*

<i>PengelolaanAgent</i>	<<boundary>>
<pre>+pengelolaanAgent() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataVendor() Operasi ini digunakan untuk mengambil data agent yang diinputkan oleh administrator, yaitu AgentId, Nama, Alamat, Telp, Email dan TanggalDaftar</pre>	

2.2.3.6 Spesifikasi Design Kelas PengelolaanCustomer

PengelolaanCustomer	<<boundary>>
<pre>+pengelolaanCustomer() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataCustomer() Operasi ini digunakan untuk mengambil data customer yang diinputkan oleh administrator, yaitu CustId, Nama, Telp, dan Email</pre>	

2.2.3.7 Spesifikasi Design Kelas PengelolaanTiket

PengelolaanTipeTiket	<<boundary>>
<pre>+pengelolaanTipeTiket() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataTipeTiket() Operasi ini digunakan untuk mengambil data customer yang diinputkan oleh administrator, yaitu JenisId, Nama dan Harga</pre>	

2.2.3.8 Spesifikasi Design Kelas

PengelolaanTransaksi

PengelolaanTransaksi	<<boundary>>
+pengelolaanTransaksi() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+getDataTransaksi() Operasi ini digunakan untuk mengambil data transaksi penjualan yang diinputkan oleh petugas <i>counter</i> , yaitu <i>NoTransaksi</i> , <i>CustId</i> , <i>AgentID</i> , <i>TglTransaksi</i> , <i>TglVisit</i> , <i>Total</i> dan jumlah <i>Voucher</i> yang dibawa oleh pengunjung.	

2.2.3.9 Spesifikasi Design Kelas

PengelolaanLaporanTransaksi

PengelolaanLaporanTransaksi	<<boundary>>
+pengelolaanLaporanTransaksi() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+displayLaporanTransaksi() Operasi ini digunakan untuk menampilkan laporan transaksi.	

2.2.3.10 Spesifikasi Design Kelas ValidasiTiket

ValidasiTiket	<<boundary>>
+validasiTiket() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	

2.2.3.11 Spesifikasi Design Kelas *LoginControl*

<i>LoginControl</i>	<<control>>
<pre>+loginControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +validasiLogin(): Boolean Operasi ini digunakan untuk mengecek format login yang diinputkan user.</pre>	

2.2.3.12 Spesifikasi Design Kelas *UserControl*

<i>UserControl</i>	<<control>>
<pre>+UserControl(): Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +validasiDataUser(): Boolean Operasi ini digunakan untuk mengecek format data user yang diinputkan administrator. +deleteDataUser() Operasi ini digunakan untuk menghapus data user yang diterima dari database. +showDataUser(): void Operasi ini digunakan untuk menampilkan data user yang sudah tersimpan di database. +searchDataUser(): void Operasi ini digunakan untuk mencari data user yang ada di dalam database. +editDataUser(): void Operasi ini digunakan untuk mengubah data user yang ada di dalam database.</pre>	

2.2.3.13 Spesifikasi Design Kelas AgentControl

AgentControl	<<control>>
<p>+AgentControl(): Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+validasiDataAgent(): Boolean Operasi ini digunakan untuk mengecek format data agent yang diinputkan administrator.</p> <p>+deleteDataAgent() Operasi ini digunakan untuk menghapus data agent yang diterima dari <i>database</i>.</p> <p>+showDataAgent(): void Operasi ini digunakan untuk menampilkan data agent yang sudah tersimpan di <i>database</i>.</p> <p>+searchDataAgent(): void Operasi ini digunakan untuk mencari data agent yang ada di dalam <i>database</i>.</p> <p>+editDataAgent(): void Operasi ini digunakan untuk mengubah data agent yang ada di dalam <i>database</i>.</p>	

2.2.3.14 Spesifikasi Design Kelas CustomerControl

CustomerControl	<<control>>
<p>+CustomerControl(): Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+validasiDataCustomer(): Boolean Operasi ini digunakan untuk mengecek format data customer yang diinputkan administrator.</p> <p>+deleteDataCustomer() Operasi ini digunakan untuk menghapus data customer yang diterima dari <i>database</i>.</p> <p>+showDataCustomer(): void Operasi ini digunakan untuk menampilkan data customer yang sudah tersimpan di <i>database</i>.</p> <p>+searchDataCustomer(): void Operasi ini digunakan untuk mencari data customer yang ada di dalam <i>database</i>.</p> <p>+editDataCustomer(): void Operasi ini digunakan untuk mengubah data customer yang ada di dalam <i>database</i>.</p>	

2.2.3.15 Spesifikasi Design Kelas TiketControl

TiketControl	<<control>>
<p>+TiketControl(): Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+validasiDataTiket(): Boolean Operasi ini digunakan untuk mengecek format data tiket yang diinputkan administrator.</p> <p>+deleteDataTiket() Operasi ini digunakan untuk menghapus data tiket yang diterima dari <i>database</i>.</p> <p>+showDataTiket(): void Operasi ini digunakan untuk menampilkan data tiket yang sudah tersimpan di <i>database</i>.</p> <p>+searchDataTiket(): void Operasi ini digunakan untuk mencari data tiket yang ada di dalam <i>database</i>.</p> <p>+editDataTiket(): void Operasi ini digunakan untuk mengubah data tiket yang ada di dalam <i>database</i>.</p>	

2.2.3.16 Spesifikasi Design Kelas

PengelolaanTransaksiControl

PengelolaanTransaksiControl	<<control>>
<pre>+pengelolaanTransaksiControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +validasiDataTransaksi(): Boolean Operasi ini digunakan untuk mengecek format data transaksi yang diinputkan Petugas Counter. +deleteDataTransaksi() Operasi ini digunakan untuk menghapus data transaksi yang diterima dari database. +showDataTransaksi(): void Operasi ini digunakan untuk menampilkan data transaksi yang sudah tersimpan di database. +searchDataTransaksi(): void Operasi ini digunakan untuk mencari data transaksi yang ada di dalam database. +editDataTransaksi(): void Operasi ini digunakan untuk mengubah data transaksi yang ada di dalam database.</pre>	

2.2.3.17 Spesifikasi Design Kelas

PengelolaanLaporanTransaksiControl

PengelolaanLaporanTransaksiControl	<<control>>
<pre>+pengelolaanLaporanTransaksiControl() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +showLaporanTransaksiPeriodeTertentu() Operasi ini digunakan untuk menampilkan laporan transaksi pada periode tertentu. +showLaporanTransaksiPerBulanTahunTertentu() Operasi ini digunakan untuk menampilkan laporan transaksi per bulan pada tahun tertentu. +showLaporanTransaksiAgentTertentu() Operasi ini digunakan untuk menampilkan laporan transaksi berdasarkan agent tertentu +showLaporanOmzetPerBulanWithLineGraph() Operasi ini digunakan untuk menampilkan laporan transaksi per bulan pada tahun tertentu dengan menggunakan line-graph.</pre>	

2.2.3.18 Spesifikasi Design Kelas

ValidasiTiketControl

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

2.2.3.19 Spesifikasi Design Kelas DataUser

DataUser	<<entity>>
+dataUser() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini. +getDataUser() Operasi ini digunakan untuk mengambil data user yang tersimpan di <i>database</i> . +inputDataUser() Operasi ini digunakan untuk memasukan data user ke <i>database</i> . +deleteDataUser() Operasi ini digunakan untuk menghapus data user dalam <i>database</i> .	

2.2.3.20 Spesifikasi Design Kelas DataAgent

DataAgent	<<entity>>
<p>+dataAgent () Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+getDataAgent () Operasi ini digunakan untuk mengambil data agent yang tersimpan di <i>database</i>.</p> <p>+inputDataAgent () Operasi ini digunakan untuk memasukan data agent ke <i>database</i>.</p> <p>+deleteDataAgent () Operasi ini digunakan untuk menghapus data agent dalam <i>database</i>.</p>	

2.2.3.21 Spesifikasi Design Kelas DataCustomer

DataCustomer	<<entity>>
<p>+dataCustomer () Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.</p> <p>+getDataCustomer () Operasi ini digunakan untuk mengambil data customer yang tersimpan di <i>database</i>.</p> <p>+inputDataCustomer () Operasi ini digunakan untuk memasukan data customer ke <i>database</i>.</p> <p>+deleteDataCustomer () Operasi ini digunakan untuk menghapus data customer dalam <i>database</i>.</p>	

2.2.3.22 Spesifikasi Design Kelas DataTiket

DataTiket	<<entity>>
+dataTiket() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+getDataTiket() Operasi ini digunakan untuk mengambil data tiket yang tersimpan di <i>database</i> .	
+inputDataTiket() Operasi ini digunakan untuk memasukan data tiket ke <i>database</i> .	
+deleteDataTiket() Operasi ini digunakan untuk menghapus data tiket dalam <i>database</i> .	

2.2.3.23 Spesifikasi Design Kelas DataTransaksi

DataTransaksi	<<entity>>
+dataTransaksi() Default konstruktor, digunakan untuk inisialisasi semua <i>attribute</i> dari kelas ini.	
+getDataTransaksi() Operasi ini digunakan untuk mengambil data transaksi yang tersimpan di <i>database</i> .	
+inputDataTransaksi() Operasi ini digunakan untuk memasukan data transaksi ke dalam <i>database</i> .	

2.2.3.24 Spesifikasi Design Kelas DataDetailTiket

DataDetailTiket	<<entity>>
<pre>+dataDetailTiket () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataDetailTiket () Operasi ini digunakan untuk mengambil data detail tiket yang tersimpan di database. +updateDataDetailTiket () Operasi ini digunakan untuk mengupdate data detail tiket ke dalam database.</pre>	

3 Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Data UserTable

Nama	Tipe	Panjang	Keterangan
UserID	Character	10	Id user, primary key
RoleID	Character	10	Id role, foreign key
Username	Character	20	Username
Password	Character	20	Password user
NamaAsli	Character	40	Nama asli user
Alamat	Character	100	Alamat user
Telp	Character	20	No telepon user

3.1.2 Deskripsi Entitas Data Customer

Nama	Tipe	Panjang	Keterangan
CustID	Character	10	Id customer, primary key
Nama	Character	40	Nama customer
Email	Character	50	Email customer
Telp	Character	20	No telepon customer

3.1.3 Deskripsi Entitas Data Agent

Nama	Tipe	Panjang	Keterangan
AgentId	Character	10	Id agent, primary key
Nama	Character	40	Nama agent
Alamat	Character	100	Alamat agent
Telp	Character	20	No telepon agent
Email	Character	50	Email agent
TglDaftar	Date		Tanggal daftar agent

3.1.4 Deskripsi Entitas Data Transaksi

Nama	Tipe	Panjang	Keterangan
NoTransaksi	Character	12	No transaksi, primary key
CustID	Character	10	Id customer, foreign key
AgentID	Character	10	Id agent, foreign key
TglTransaksi	Date		Tanggal transaksi
TglVisit	Date		Tanggal berkunjung
Total	Number	10,2	Total transaksi
Voucher	Int	5	Jumlah voucher

3.1.5 Deskripsi Entitas Data DetailTransaksi

Nama	Tipe	Panjang	Keterangan
DetailID	Character	12	Id detail transaksi, primary key
NoTransaksi	Character	12	No transaksi, foreign key
JenisID	Character	10	Id jenis, foreign key
Quantity	Int	5	Jumlah pengunjung (anak atau dewasa)

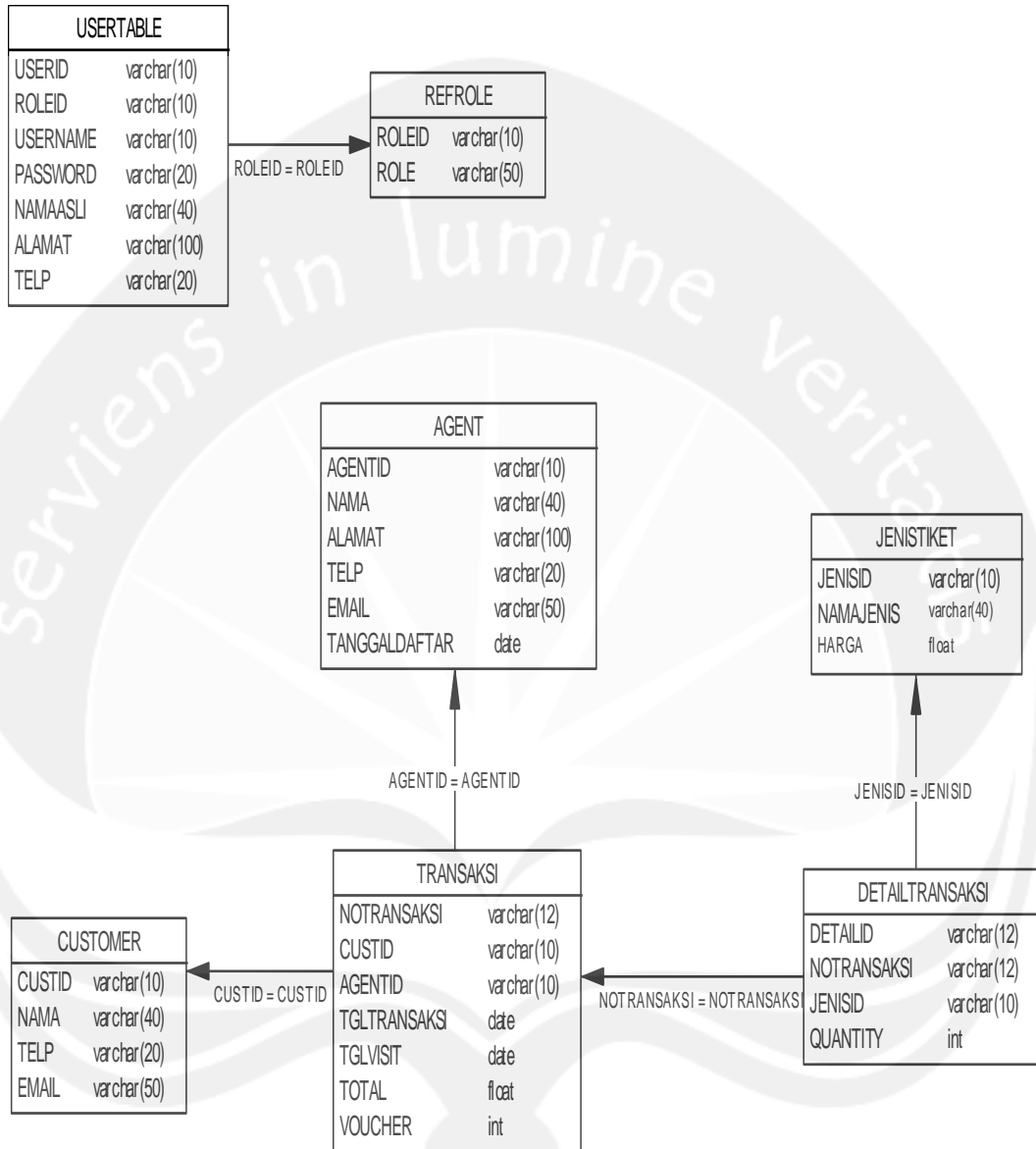
3.1.6 Deskripsi Entitas Data JenisTiket

Nama	Tipe	Panjang	Keterangan
JenisID	Character	10	Id jenis, primary key
Nama Jenis	Character	40	Nama jenis tiket
Harga	Number	10,2	Harga berdasarkan jenis tiket

3.1.7 Deskripsi Entitas Data RefRole

Nama	Tipe	Panjang	Keterangan
RoleID	Int	5	Id role, primary key
Role	Character	40	Nama role

3.2 Physical Data Model

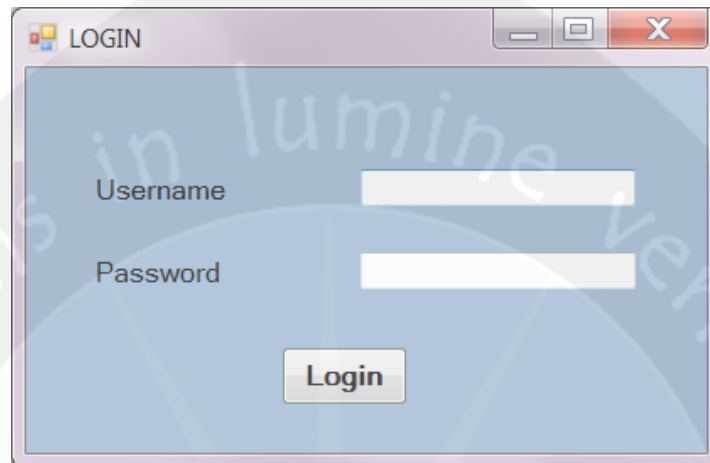


Gambar 3.1 Physical Data Model

4 Perancangan Antarmuka

4.1 Sketsa UI dan Deskripsinya

4.1.1 Antarmuka Halaman Login

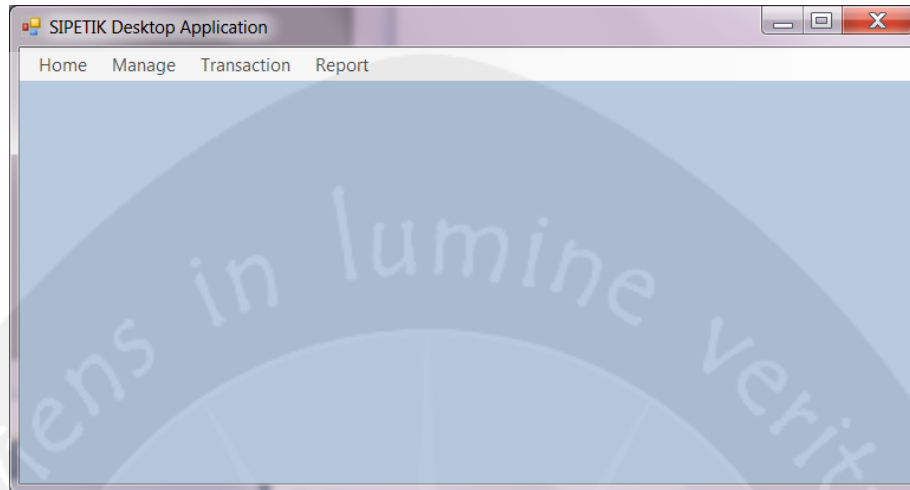


Gambar 4.1 Rancangan Antarmuka Login

Antarmuka ini digunakan untuk melakukan proses login ke dalam sistem. Untuk mendapat akses masuk ke dalam sistem, user harus menginputkan username dan password dengan benar pada textbox yang telah disediakan. Pada saat tombol login ditekan, sistem akan mengecek username dan password yang diinputkan dengan data id dan password yang telah tersimpan di database. Jika data id dan password benar atau cocok maka user akan masuk ke dalam sistem, sebaliknya jika id dan password salah atau tidak cocok maka akan diberikan pesan peringatan. Sistem juga akan mengecek role dari user yang login.

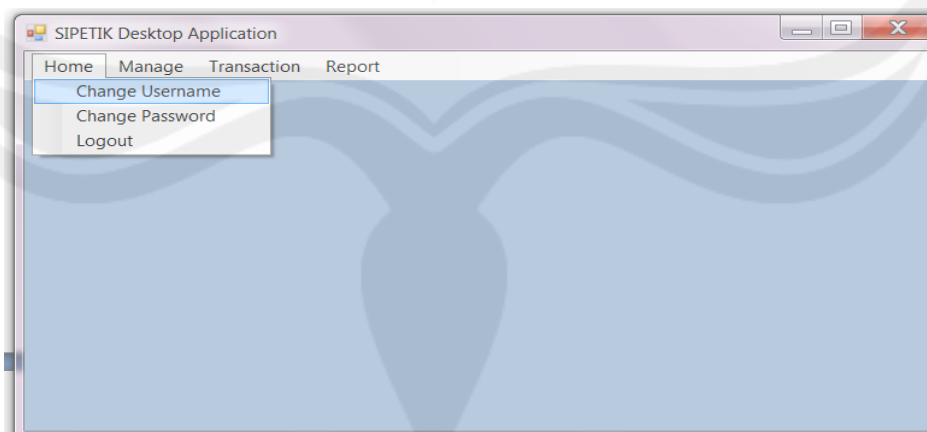
Apabila user mengindikasikan menekan *button Login*, maka proses akan dilanjutkan ke antarmuka halaman utama. Namun bila *button Exit* yang ditekan, maka akan keluar dari sistem ini.

4.1.2 Antarmuka Halaman Utama



4.2 Rancangan Antarmuka Halaman Utama

Antarmuka ini merupakan inti dari sistem yang berisi menu untuk masuk ke antarmuka-antarmuka lainnya. Menu Home berisi sub menu *Change Username*, *Change Password* dan *Logout*. Menu *Manage* berisi sub menu untuk Pengelolaan User dan Agent. Menu Transaksi berisi sub menu *Write Transaction* dan *Read Ticket*.

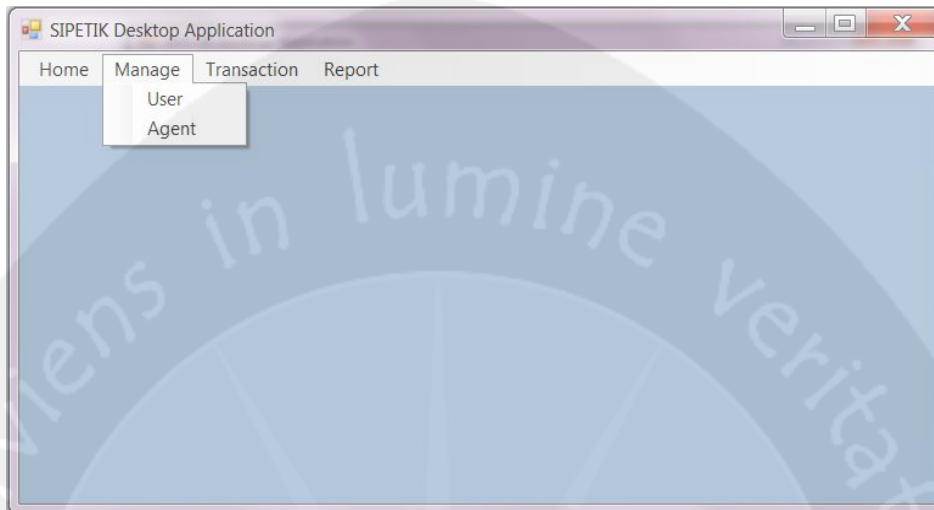


4.2.1 Rancangan Antarmuka Halaman Utama-Tab Home

Saat user memilih sub menu *Change Username* maka sistem akan masuk ke antarmuka *Ubah Password*. Kemudian jika user memilih sub menu *Change Password*, maka sistem akan

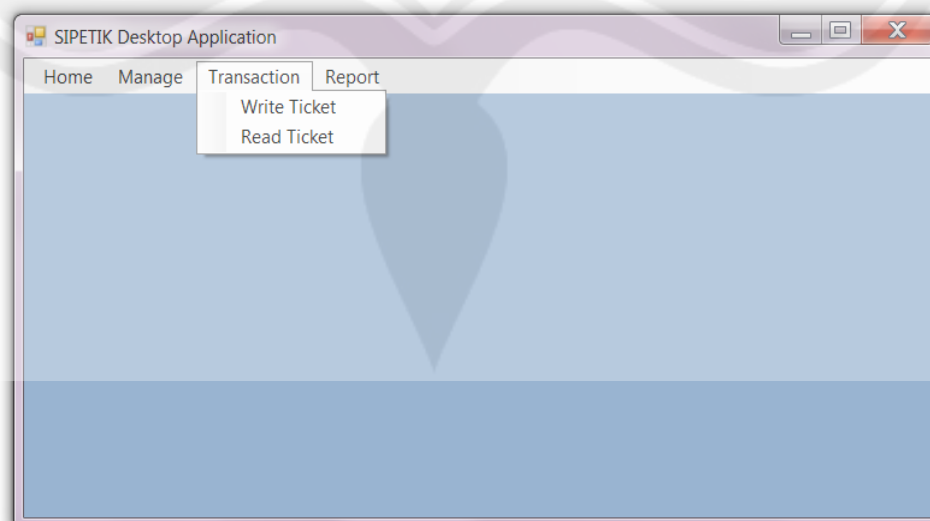
Program Studi Teknik Informatika	DPPL – SIPETIK	241/ 274
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

menampilkan antarmuka Ubah *Password*. Bila user memilih Log Out maka akan kembali ke antarmuka *Login*.



4.2.2 Rancangan Antarmuka Halaman Utama-Tab *Manage*

Jika user mengindikasikan masuk ke tab menu *Manage*, dan memilih *User*, maka sistem akan menampilkan antarmuka untuk Pengelolaan Data *User*, sedangkan jika dipilih sub menu *Agent*, sistem akan menampilkan antarmuka Pengelolaan Data *Agent*. Kedua sub menu ini hanya dapat diakses oleh user dengan *role* sebagai Administrator atau *Manager*.



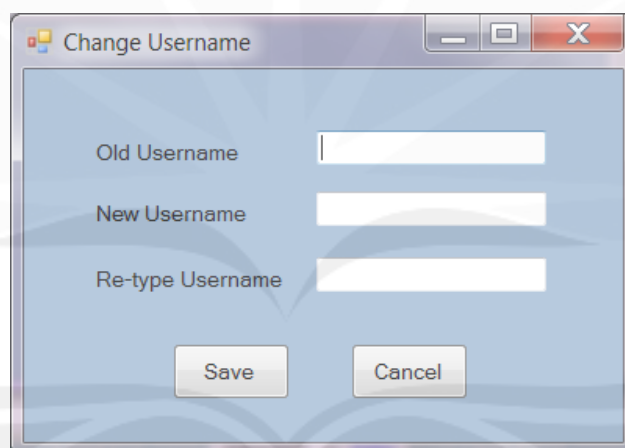
Gambar 4.2.3 Antarmuka Halaman Utama-Tab *Transaction*

Bila user masuk ke tab menu *Transaction* dan memilih sub menu *Write Ticket*, maka sistem akan menampilkan antarmuka Transaksi Penjualan. Sub menu ini hanya dapat diakses oleh user dengan *role* sebagai *Petugas Counter*.

Bila user masuk ke sub menu *Read Ticket*, maka sistem akan menampilkan antarmuka untuk Validasi Tiket yang dapat diakses oleh user dengan *role* sebagai *Petugas Gate*.

Jika user mengindikasikan memilih tab menu report, maka sistem akan menampilkan antarmuka Cetak Laporan Transaksi, dimana menu ini hanya dapat diakses oleh user dengan *role* sebagai *Administrator/ Manager*.

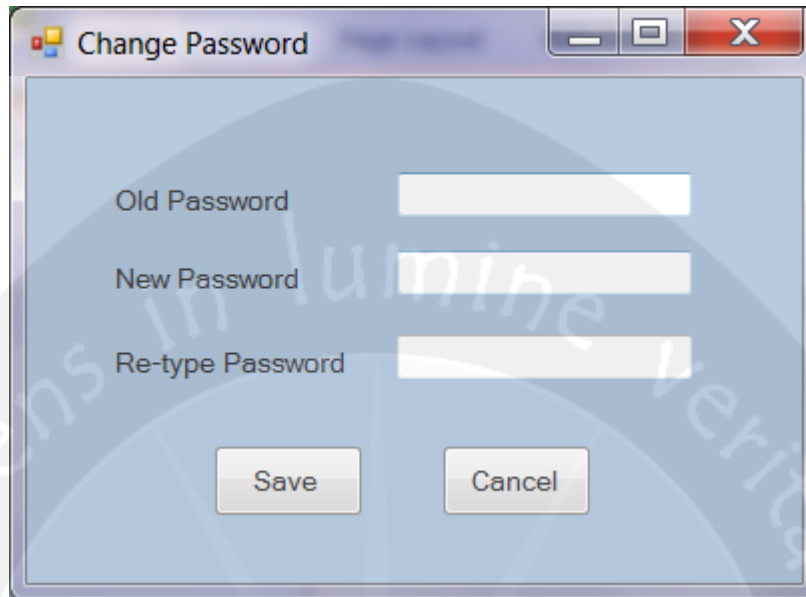
4.1.3 Antarmuka Halaman Ubah Username



Gambar 4.3 Antarmuka Ubah Username

Antarmuka ini digunakan jika user ingin mengubah *username* untuk *Login*. Pada halaman ini user diminta untuk menginputkan *username* yang lama, kemudian *username* yang baru dan diminta mengulang input *username* yang baru untuk validasi. Jika user memilih *button Save*, maka sistem akan menyimpan perubahan *username* yang dilakukan user. Jika yang dipilih *button Cancel*, maka akan kembali ke antarmuka halaman *Login*.

4.1.4 Antarmuka Halaman Ubah *Password*



The image shows a standard Windows-style dialog box titled "Change Password". It features a light blue background and a white border. At the top, there is a title bar with the text "Change Password" and three control buttons: minimize, maximize, and close. The main area of the dialog contains three text input fields, each with a label to its left: "Old Password", "New Password", and "Re-type Password". Below these fields are two buttons: "Save" and "Cancel". The dialog is centered on the screen, and a large, faint watermark of a university logo is visible in the background.

Gambar 4.4 Antarmuka Ubah *Password*

Antarmuka ini digunakan jika user ingin mengubah *password* untuk *Login*. Pada halaman ini user diminta untuk menginputkan *password* yang lama, kemudian *password* yang baru dan diminta mengulang input *password* yang baru untuk validasi. Jika user memilih *button Save*, maka sistem akan menyimpan perubahan *password* yang dilakukan user. Jika yang dipilih *button Cancel*, maka akan kembali ke antarmuka halaman *Login*.

4.1.5 Antarmuka Halaman Pengelolaan Data User

The screenshot shows a web application window titled "Pengelolaan User". The interface is divided into two main sections. The left section contains a form for adding or editing user data, with fields for "Id User", "Id Role" (a dropdown menu), "Username", "Password", "Nama Asli", "Alamat", and "Telepon". Below the form are three buttons: "Entry", "Edit", and "Delete". Further down are search fields: "Search by Id", "Search by Username", and "Search by Role" (a dropdown menu), followed by a "Search" button. The right section is a large rectangular area labeled "Data User", which is currently empty, representing a data grid view.

Gambar 4.5 Antarmuka Pengelolaan Data User

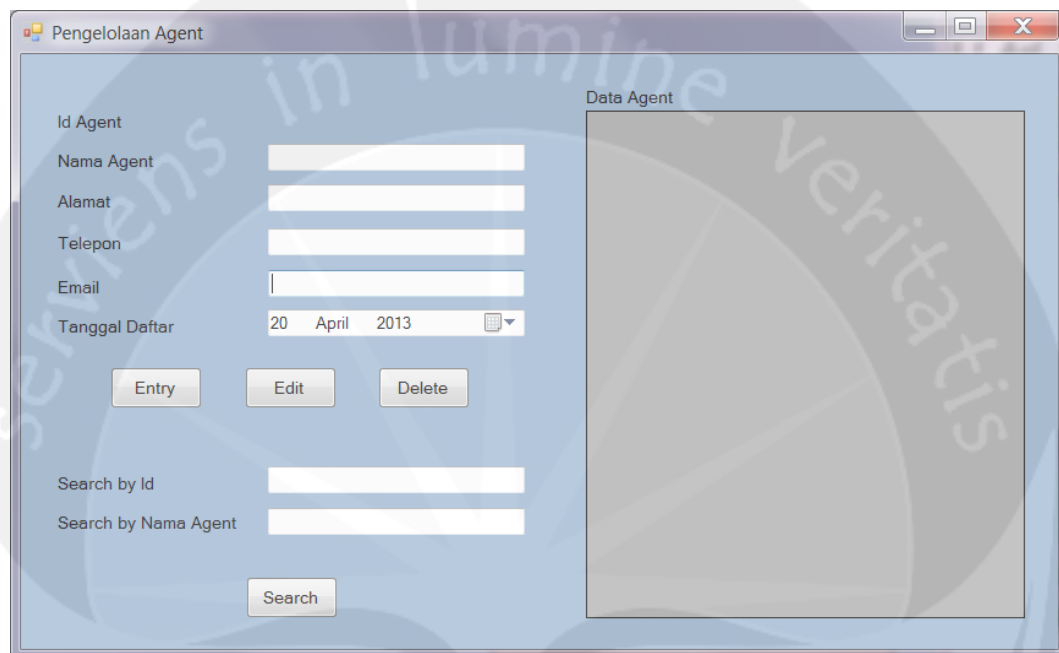
Antarmuka ini digunakan untuk mengelola data-data user oleh Administrator/ *Manager*. User diminta mengisi data-data mengenai detail user seperti *id role* yang dapat dipilih melalui *combobox*, *username*, *password*, nama asli, alamat, telepon. Dan tampilan pada sebelah kanan merupakan data-data user yang sudah ada dalam *database*.

Antarmuka ini juga memiliki 3 buah *button* untuk melakukan pengelolaan yaitu *entry* untuk menambah data user dalam *database*, *edit* untuk mengubah data user dalam *database*, *delete* untuk menghapus data user dalam *database*. *Display* dilakukan secara langsung pada data grid view saat user menekan salah satu *button* tersebut.

User juga dapat melakukan pilihan menampilkan data user berdasarkan *Id*, *username* atau *role*. Setelah memilih salah

satunya dan mengindikasikan menekan *button search*, maka data akan segera difilter berdasarkan pilihannya dan ditampilkan pada grid view.

4.1.6 Antarmuka Halaman Pengelolaan Data Agent



Gambar 4.6 Antarmuka Pengelolaan Data Agent

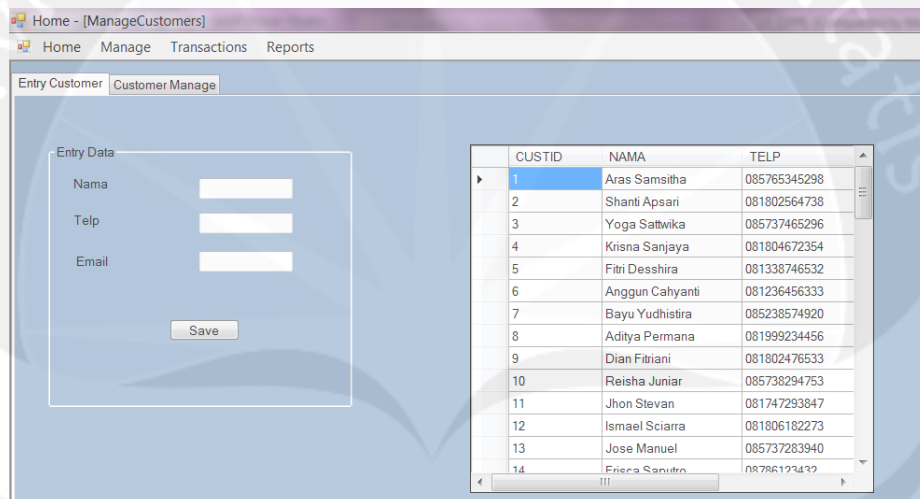
Antarmuka ini digunakan untuk mengelola data-data agent oleh Administrator. User diminta menginputkan data-data mengenai detail data agent seperti nama agent, alamat, no telepon, email serta tanggal daftar agent. Id agent akan digenerate otomatis oleh sistem. Tampilan pada sebelah kanan merupakan data-data vendor yang sudah ada dalam *database*.

Antarmuka ini juga memiliki 3 buah *button* untuk melakukan pengelolaan yaitu *entry* untuk menambah data agent dalam *database*, *edit* untuk mengubah data agent dalam *database*, dan *delete* untuk menghapus data agent dalam *database*. Data grid view di sebelah kanan berfungsi untuk menampilkan data agent yang sudah ada dalam *database*.

Display dilakukan secara langsung pada data grid view saat user menekan salah satu *button* tersebut.

User juga dapat melakukan pencarian agent berdasarkan Id maupun nama agent. Setelah memilih salah satunya dan mengindikasikan menekan *button search*, maka data akan segera difilter berdasarkan pilihannya dan ditampilkan pada grid view.

4.1.7 Antarmuka Halaman Pengelolaan Data Customer



Gambar 4.7 Antarmuka Pengelolaan Data Customer

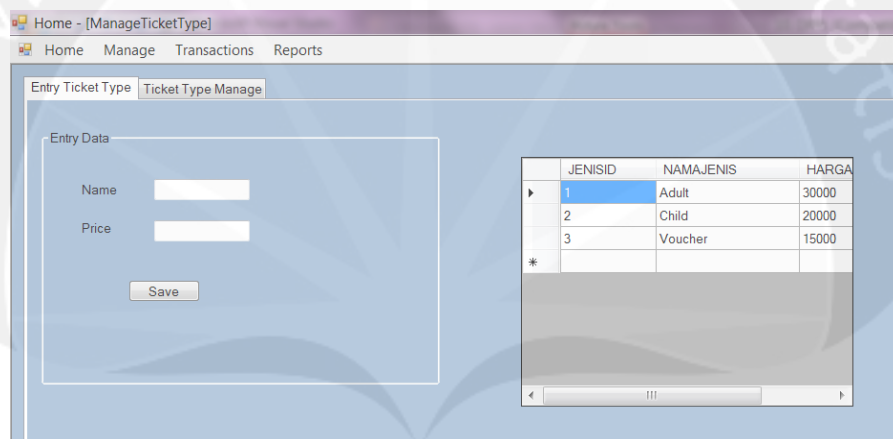
Antarmuka ini digunakan untuk mengelola data-data customer oleh Administrator. User diminta menginputkan data-data mengenai detail data customer seperti nama customer, no telepon serta email customer. Id customer akan digenerate otomatis oleh sistem. Tampilan pada sebelah kanan merupakan data-data customer yang sudah ada dalam *database*.

Antarmuka ini juga memiliki 3 buah *button* untuk melakukan pengelolaan yaitu *entry* untuk menambah data customer dalam *database*, *edit* untuk mengubah data customer dalam *database*, dan *delete* untuk menghapus data customer dalam *database*. Data grid view di sebelah kanan berfungsi

untuk menampilkan data customer yang sudah ada dalam *database*. *Display* dilakukan secara langsung pada data grid view saat user menekan salah satu *button* tersebut.

User juga dapat melakukan pencarian customer berdasarkan Id maupun nama customer. Setelah memilih salah satunya dan mengindikasikan menekan *button search*, maka data akan segera difilter berdasarkan pilihannya dan ditampilkan pada grid view.

4.1.8 Antarmuka Halaman Pengelolaan Data Tiket



Gambar 4.8 Antarmuka Pengelolaan Data Tipe Tiket

Antarmuka ini digunakan untuk mengelola data-data tipe tiket oleh Administrator. User diminta menginputkan data-data mengenai detail data tiket seperti nama dan harga. Id tipe tiket akan digenerate otomatis oleh sistem. Tampilan pada sebelah kanan merupakan data-data tipe tiket yang sudah ada dalam *database*.

Antarmuka ini juga memiliki 3 buah *button* untuk melakukan pengelolaan yaitu *entry* untuk menambah data tipe tiket dalam *database*, *edit* untuk mengubah data tipe tiket dalam *database*, dan *delete* untuk menghapus data tipe tiket dalam *database*. Data grid view di sebelah kanan berfungsi

untuk menampilkan data tipe tiket yang sudah ada dalam *database*. *Display* dilakukan secara langsung pada data grid view saat user menekan salah satu *button* tersebut.

User juga dapat melakukan pencarian tipe tiket berdasarkan Id maupun nama tipe. Setelah memilih salah satunya dan mengindikasikan menekan *button search*, maka data akan segera difilter berdasarkan pilihannya dan ditampilkan pada grid view.

4.1.9 Antarmuka Halaman Transaksi Penjualan Tiket

The screenshot shows a window titled "Transaksi" with a light blue background. At the top, there are labels for "No Transaksi" and "Tanggal Transaksi". Below these, there is a dropdown menu for "Jenis Transaksi" and a text input field for "[Customer name or Agent Id]". The main area contains a table with columns "Jumlah", "Harga", and "Total". The rows are for "Dewasa", "Anek", and "Voucher", each with a corresponding input field and placeholder text like "[nominal harga]". Below the table is an "Add" button. To the right of the table is a large grey rectangular area labeled "Tampil Tiket". At the bottom, there are input fields for "Bayar" and "Kembali", a "Save" button, and a "Write Ticket" button. Summary labels like "Subtotal", "Total Voucher", and "Grand Total" are also present with their respective placeholder values.

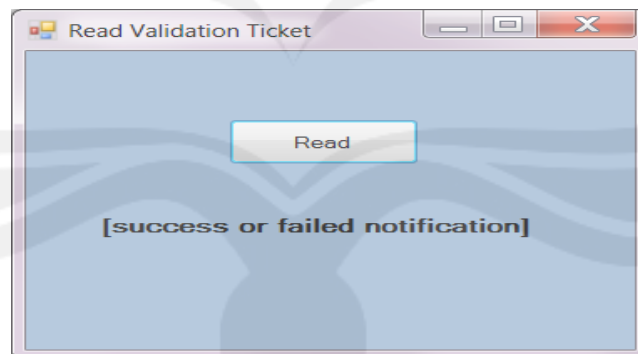
Gambar 4.9 Antarmuka Transaksi Penjualan Tiket

Antarmuka ini digunakan untuk mengelola transaksi penjualan oleh Petugas *Counter*. User (Petugas *Counter*) diminta menginputkan jenis penjualan terlebih dahulu melalui combobox. Terdapat 3 jenis pilihan penjualan, yaitu *direct personal*, *email personal* dan *by agent*. Ketika dipilih *direct*

personal atau transaksi langsung di lokasi, mana tidak perlu input *customer name*. Jika dipilih by email, maka perlu input nama *customer* dan jika by agent maka user diminta menginput id agent. Selanjutnya user No order degenerate secara otomatis. Kemudian user diminta menginput jumlah orang dewasa dan anak yang ingin masuk ke area tempat wisata. Jika pelanggan membawa voucher, maka user menginput jumlah voucher yang dibawa tersebut. Kemudian ketika *button Add* diklik, sistem akan menampilkan secara otomatis berapa total bayar pelanggan beserta potongan harga jika membawa voucher. Berikutnya user menginput nominal bayar oleh pelanggan dan jika ada sisa uang kembali, akan terinput secara otomatis oleh sistem.

Ketika *button Save* diklik, maka transaksi akan tersimpan di *database* dan disiapkan tiketnya. *Button write* tiket digunakan saat ingin mengisi (*write*) *ticket* RFID.

4.1.10 Antarmuka Halaman Validasi Tiket



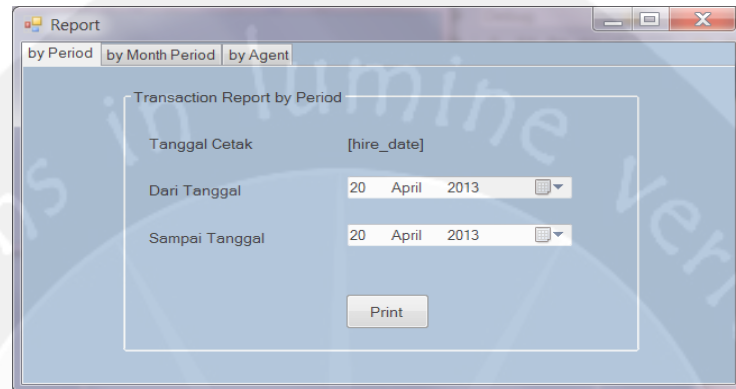
Gambar 4.10 Antarmuka Validasi Tiket

Antarmuka ini digunakan untuk melakukan validasi tiket oleh Petugas *Gate*. Dalam skenarionya, pelanggan akan menyerahkan *card* yang telah di-*write* oleh Petugas *Counter* saat transaksi. Lalu Petugas *Gate* mendekatkan *card* tersebut ke *Reader* dan menekan *button Read*. Jika validasi berhasil akan tampil pesan bahwa tiket sukses tervalidasi. Jika gagal

Program Studi Teknik Informatika	DPPL – SIPETIK	250/ 274
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		

akan menampilkan pesan bahwa tiket tidak berhasil tervalidasi.

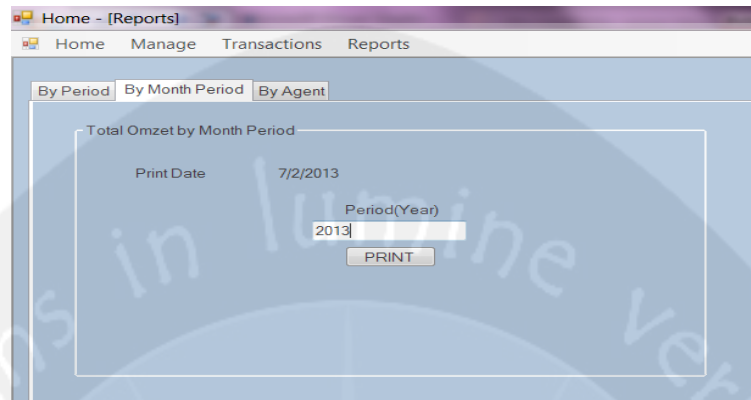
4.1.11 Antarmuka Halaman Cetak Laporan Transaksi Periode tertentu



Gambar 4.11 Antarmuka Cetak Laporan Transaksi Periode tertentu

Antarmuka ini digunakan untuk menyediakan laporan transaksi penjualan tiket untuk periode tertentu oleh *manager*. *User (manager)* diminta menginputkan rentang waktu yang ingin dicetak laporannya. Setelah melakukan input tanggal mulai hingga tanggal selesai dan mengindikasikan menekan *button* Cetak Laporan, maka system akan menampilkan hasil laporan tersebut berupa *crystal report*.

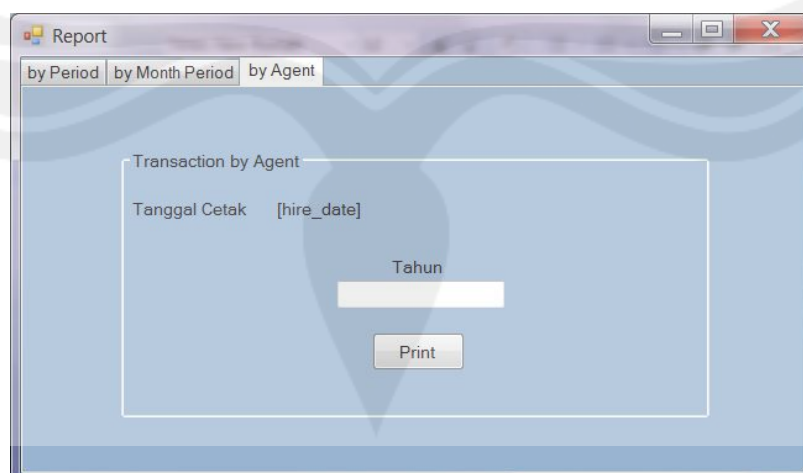
4.1.12 Antarmuka Halaman Cetak Laporan Transaksi per Bulan pada Tahun tertentu



Gambar 4.12 Antarmuka Cetak Laporan per Bulan di Tahun tertentu

Antarmuka ini digunakan untuk menyediakan laporan transaksi penjualan tiket per bulan dalam tahun tertentu oleh *manager*. User (*manager*) menginputkan tahun yang ingin dicetak laporannya. Setelah dipilih dan mengindikasikan menekan *button* Print, maka system akan menampilkan hasil laporan tersebut berupa crystal report.

4.1.13 Antarmuka Halaman Cetak Laporan Transaksi Agent



Gambar 4.13 Antarmuka Cetak Laporan Transaksi berdasarkan Agent

Antarmuka ini digunakan untuk menyediakan laporan transaksi penjualan tiket berdasarkan Agent oleh *manager*. User (*manager*) diminta memilih tahun yang ingin dicetak laporan transaksi per Agentnya. Setelah melakukan input tahun dan mengindikasikan menekan *button* Print, maka sistem akan menampilkan hasil laporannya pada *crystal report*.

