CHAPTER V CONCLUSIONS AND RECOMMENDATIONS

V.1 Conclusions

1. JOE-Banking which provides online transactions across internet protocol that eases customers doing their transactions and getting account information, has been successfully built by JSP, Servlet, and Java 2 Platform technologies.
2. JOE-banking has implemented object-oriented. Each object-oriented requirement of JOE-Banking, which are boundary class, entity class, and control class, is really implemented. Those classes are united into a *.jar file.
3. JOE-banking has implemented component-based theory. The *.jar file can be treated as a reusable component, which can be simply implemented into another application.

V.2 Recommendations

1. Develop JOE-Banking application for mobile user that increases the mobility.
2. Implement the data encryption for better security of the application.

REFERENCES

1. Jacobson, Ivar, Grady Booch, James Rumbaugh, *The Unified Software Development Process*, Addison Wesley, 2001.
2. Deitel, Advanced *Java 2 Platform How to Program*, Prentice-Hall Inc, 2003
3. NIIT, *Servlets & JSP Student Guide*, 2001.
4. Keegan, Champenois, Crawley, Hunt, Webster, *NetBeans IDE Field Guide*, Prentice Hall, 2005.
5. www.NetBeans.org

SRS

Software Requirements Specification

The Analysis of

Java Object Oriented E-Banking

(JOE-Banking)

Written by:

Sonny Hendrawan

02 07 03578

Department of Informatics Engineering

Faculty of Industrial Technology

Atma Jaya Yogyakarta University

Alteration List

Revision	Description
A	
B	
C	
D	
E	
F	

INDEX DATE	-	A	B	C	D	E
Author						
Examinated by						
Agreed by						

This is a private document, no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

Page Alteration List

Page	Revision	Page	Revision

This is a private document, no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

Table of Contents

This is a private document, no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

List of Figure

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

Java Object Oriented E-Banking

(JOE-BANKING)

1 Preface

1.1 Purpose

This Software Requirements Spesification (SRS) is the requirements analysis of JOE-BANKING (Java Object Oriented E-Banking) which will be implemented. This document will be used for technical reference of JOE-BANKING development process. Java programming language from Sun Microsystem will be used as the tool and Oracle 9i as the database management systems.

1.2 Problems Scope

JOE-BANKING (Java Object Oriented E-Banking) is a web-based application implementing JSP (Java Server Pages) for handling the bank transactions over the internet. JOE-BANKING provides several functions:

a. Web Based
    1. Login Web
    2. Change Password
    3. Transfer
    4. View Acccunt Information
    5. Pay Bill
        i. Pay Credit Card Bill
        ii. Pay Electricity Bill
        iii. Pay Cellular Phone Bill

This is a private document, no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

b. Desktop Based

1. Login
2. Manage Administrator
    i. Add Administrator
    ii. Update Administrator
    iii. Delete Administrator
3. Manage Account Holder
    i. Add Account Holder
    ii. Update Account Holder
4. Display Transactions
    i. Display Daily Transactions
    ii. Display Monthly Transactions
    iii. Display Account Transactions
5. Print Report
    i. Print Daily Report
    ii. Print Monthly Report
    iii. Print Account Report

1.3 Definition, Acronym, and Abstract

- SRS stand for Software Requirements Spesification. SRS is an analysis and spesification of the sortware that will be made.
- JOE-BANKING stand for Java Object Oriented E-Banking. JOE-BANKING is the software that will be made.
- SRS-JOE-BANKING.FPX is code that represents the requeirements of JOE-BANKING. FP is a fase code, and X is a number or digit requirement.
- GUI stand for Graphical User Interface, is used to build the interface of the code.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

1.4 References

This project uses:

- Jacobson, Ivar, Grady Booch, James Rumbaugh, *The Unified Software Development Process*, Addison Wesley, 2001.
- Deitel, Advanced *Java 2 Platform How to Program*, Prentice-Hall Inc, 2003
- NIIT, *Servlets & JSP Student Guide*, 2001.
- Keegan, Champenois, Crawley, Hunt, Webster, *NetBeans IDE Field Guide*, Prentice Hall, 2005.
- www.NetBeans.org

1.5 Overview

This document consists of 3 parts. The first part contains purposes, problems scope, definitions, references, and overview.

The second part contains JOE-BANKING global explanations such as functions, user's characteristic, scope, and assumption that will be applied in this software development.

The third part contains the specific things of software requirements.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2 Descriptions

2.1 Product's Perspectives

JOE-BANKING is a web-based application that uses JSP (Java Server Pages) on Java 2 Platform programming as interface, servlet as business logic, and Oracle 9i as the database management system. This application is made for serving online transactions such as transferring, credit card payment, electricity payment, and mobile phone payment which known as e-banking.

This application can be accessed all around the world, but not all users can access the application. Only registered users can use this application by entering username and password for the validation process.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.2 Product Functions

This application provides several functions:

1) **Login Web (SRS-JOE-BANKING.FP1)**

   This function is used for validation process which limit people who do not have right to use this application.

2) **Change Password (SRS-JOE-BANKING.FP2)**

   This function is used for changing account holder password.

3) **Transfer (SRS-JOE-BANKING.FP3)**

   This function is used for transferring amount of money to another account.

4) **View Account Information (SRS-JOE-BANKING.FP4)**

   This function is used for viewing the account information (balance).

5) **Pay Bill (SRS-JOE-BANKING.FP5)**

   a) **Pay Credit Card Bill (SRS-JOE-BANKING.FP6)**

      This function is used for paying the credit card claim.

   b) **Pay Electricity Bill (SRS-JOE-BANKING.FP7)**

      This function is used for paying the electricity claim.

   c) **Pay Cellular Phone Bill (SRS-JOE-BANKING.FP8)**

      This function is used for paying the cellular phone claim.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

6) Login (SRS-JOE-BANKING.FP9)

This function is used for limiting people who do not have right to use this application.

7) Manage Administrator (SRS-JOE-BANKING.FP10)

a) Add Administrator (SRS-JOE-BANKING.FP11)

This function is used for adding administrator data.

b) Update Administrator (SRS-JOE-BANKING.FP12)

This function is used for updating administrator data.

c) Delete Administrator (SRS-JOE-BANKING.FP13)

This function is used for deleting administrator data (active administrator cannot be deleted).

8) Manage Account Holder (SRS-JOE-BANKING.FP14)

a) Add Account Holder (SRS-JOE-BANKING.FP15)

This function is used for adding account holder data.

b) Update Account Holder (SRS-JOE-BANKING.FP16)

This function is used for updating account holder data.

9) Display Transactions (SRS-JOE-BANKING.FP17)

This function is used for displaying transactions.

a) Display Daily Transactions (SRS-JOE-Banking.FP18)

This function is for displaying daily transactions.

b) Display Monthly Transactions (SRS-JOE-Banking.FP19)

This function is for displaying monthly transactions.

c) Display Account Transactions (SRS-JOE-Banking.FP20)

This function is for displaying transactions per account.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

10) **Print Daily Report (SRS-JOE-BANKING.FP21)**

This function is for printing daily report.

11) **Print Monthly Report (SRS-JOE-BANKING.FP22)**

This function is for printing monthly report.

12) **Print Account Report (SRS-JOE-Banking.FP23)**

This function is for printing account report.

2.3 User's Characteristic

JOE-BANKING's user is person who has a basic knowledge about internet, browser application such as Internet Explorer, NetScape, Opera, etc, and able to use GUI application.

2.4 Scopes

This application is able to run multi-platform (Windows, MacOS, etc) but it is designed on Windows platform. This application connected to Oracle 9i database which stores the user's data. User who does not have data inside the database (data for authentication process) won't be able to use this application.

2.5 Assumptions and Dependencies

For clients, this application runs on multi-platform computers which connected to the internet and have browser application for displaying the interface. The minimum spesification is Pentium III 500 MHz, memory 64 MB.

For server, this application runs on Windows platform which have JVM (Java Virtual Mechine) including JRE (Java Runtime Environment) and requires minimum specification Pentium III 600Mhz, memory 256MB for running the Oracle 9i database server.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3 Spesific Descriptions

3.1 External Interface Requirements

The external interface requirements of JOE-BANKING include user interface requirements, hardware interface requirements, and software interface requirements.

3.1.1 User Interface Requirements

User interacts with this application through monitor in several forms on browser. User can see the output or the result of this application on browser.

3.1.2 Hardware Interface Requirements

JOE-BANKING uses several hardware interface:

1. Personal Computer (PC)
2. Keyboard
3. Monitor
4. Mouse

3.1.3 Software Interface Requirements

Software interfaces support JOE-BANKING is:

1. Operating System, Windows XP
2. Software NetBeans IDE 4.0
3. J2EE 1.5 SDK
4. Database Management Systems, Oracle 9i

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2 Functional Requirements

3.2.1 Use Case Diagram Package Web-Based

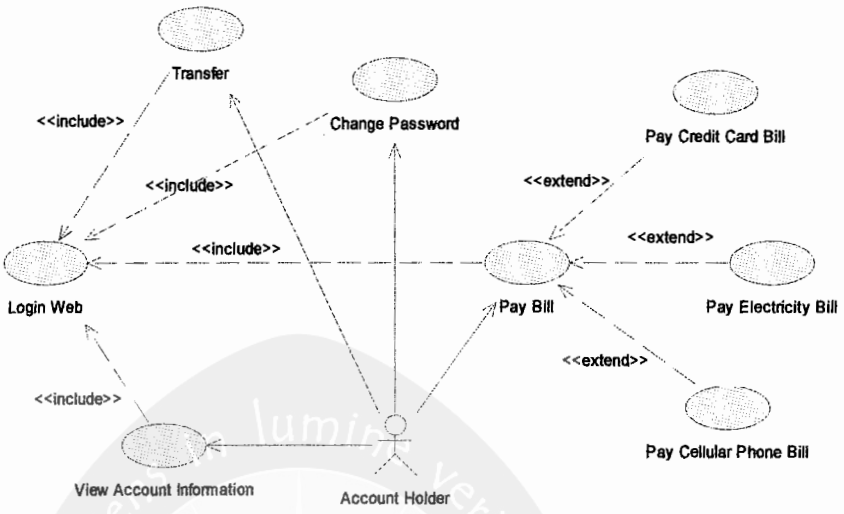

Figure 3.1 Use Case Diagram Package Web-Based

3.2.2 Use Case Diagram Package Desktop-Based

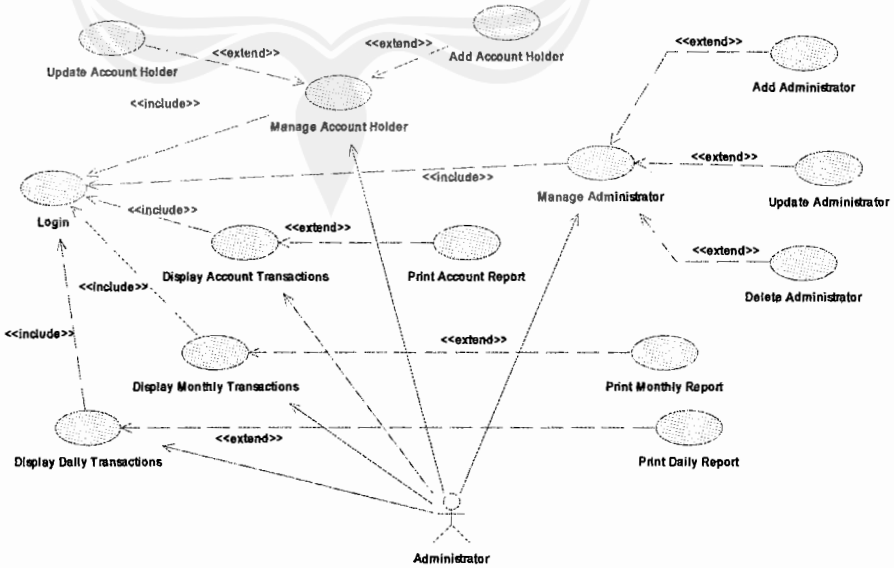

Figure 3.2 Use Case Diagram Package Desktop-Based

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.1 Use Case Login Web Scenario

Identification		
Number	SRS-JOE-BANKING.FP1	
Name	Login Web	
Purpose	Validation process for limiting unregistered account holders.	
Description	Account holder inputs username and password to be validated by system.	
Actor	Account Holder	
Main Scenario		
Initial Condition	-	
Actor Action		**System Reaction**
1. Account holder inputs username and password.		2. System validates the username and password.
Final Condition	Account holder enters the system and can use the functions.	
Alternative Path	-	
Exception Path	Incorrect username and password: 1.Account holder can not use the system. 2.Back to 1st action.	
Include	-	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.2 Use Case Change Password Scenario

Identification	
Number	SRS-JOE-BANKING.FP2
Name	Change Password
Purpose	Changing account holder password.
Description	Account holder inputs old password and new password for changing process.
Actor	Account holder
Main Scenario	
Initial Condition	Account holder is passed through Login.

Actor Action	**System Reaction**
1.Account holder inputs old password and new password.	2.System validates old password.
	3.System changes the password.

Final Condition	Password is changed.
Alternative Path	-
Exception Path	Invalid old password: 1.Account holder can not change the password. 2.Back to 1st action. Input for 2 new password is not equal: 1.Account holder can not change the password. 2.Back to 1st action.
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.3 Use Case Transfer Scenario

Identification	
Number	SRS-JOE-BANKING.FP3
Name	Transfer
Purpose	Transferring the amount of money from one sender's account to receiver's account.
Description	Account holder inputs the amount of money and receiver's account number for transferring process.
Actor	Account Holder
Main Scenario	
Initial Condition	Account Holder is passed through Login.

Actor Action	**System Reaction**
1.Account holder inputs receiver's account and the amount of money transferred.	2. System checks the sender's balance.
	3. System transfers the money.

Final Condition	Sender's balance is reduced and receiver's balance is increased as the amount of money transferred.
Alternative Path	-
Exception Path	Sender's balance is not sufficient: 1. Account holder can not transfer. 2.Back to 1st action.
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.4 Use Case View Account Information Scenario

Identification		
Number	SRS-JOE-BANKING.FP4	
Name	View Account Information	
Purpose	Displaying user's account information.	
Description	Account holder chooses menu for displaying the account information of the current month.	
Actor	Account Holder	
Main Scenario		
Initial Condition	Account holder is passed through Login.	
Actor Action		**System Reaction**
1. Account holder chooses the menu of displaying account information.		2. System gets the data from database and displays it.
Final Condition	Account holder gets account information.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.5 Use Case Pay Bill Scenario

Identification	
Number	SRS-JOE-BANKING.FP5
Name	Pay Bill
Purpose	Choose menu for bill payment.
Description	Account holder chooses menu for bill payment.
Actor	Account Holder
Main Scenario	
Initial Condition	Account holder is passed through Login.
Actor Action	**System Reaction**
1. Account holder chooses the menu of credit card bill payment.	2. System displays the credit card bill payment interface.
Final Condition	Account holder can use the credit card bill payment function.
Alternative Path	-
Exception Path	-
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.6 Use Case Pay Credit Card Bill Scenario

Identification		
Number	SRS-JOE-BANKING.FP6	
Name	Pay Credit Card Bill	
Purpose	Transferring amount of money for paying credit card bill.	
Description	Account holder inputs credit card number for payment process.	
Actor	Account Holder	
Main Scenario		
Initial Condition	Account holder is passed through Login.	
Actor Action		**System Reaction**
1. Account holder inputs the credit card number.		2. System gets the bill data and displays it.
3. Account holder chooses to pay for the claim.		4. System reduces the balance to be transferred for credit card payment.
Final Condition	Account holder's balance is reduced and bill is paid.	
Alternative Path	-	
Exception Path	Invalid credit card number: 1.Payment process is canceled. 2.Back to 1st action. Balance is not sufficient: 1.Payment process is canceled. 2.Back to 1st action.	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.7 Use Case Pay Electricity Bill Scenario

Identification	
Number	SRS-JOE-BANKING.FP7
Name	Pay Electricity Bill
Purpose	Transferring amount of money for paying electricity bill.
Description	Account holder inputs customer number for payment process.
Actor	Account Holder
Main Scenario	
Initial Condition	Account holder is passed through Login.

Actor Action	**System Reaction**
1. Account holder inputs the customer number.	2. System gets the bill data and displays it.
3. Account holder chooses to pay for the claim.	4. System reduces the balance to be transferred for electricity payment.

Final Condition	Account holder's balance is reduced and bill is paid.
Alternative Path	-
Exception Path	Invalid customer number: 1.Payment process is canceled. 2.Back to 1st action. Balance is not sufficient: 1.Payment process is canceled. 2.Back to 1st action.
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.8 Use Case Pay Cellular Phone Bill Scenario

Identification	
Number	SRS-JOE-BANKING.FP8
Name	Pay Cellular Phone Bill
Purpose	Transferring amount of money for paying cellular phone bill.
Description	Account holder inputs mobile phone number for payment process.
Actor	Account Holder
Main Scenario	
Initial Condition	Account holder is passed through Login.

Actor Action	**System Reaction**
1. Account holder inputs the mobile phone number.	2. System gets the bill data and displays it.
3. Account holder chooses to pay for the claim.	4. System reduces the balance to be transferred for mobile phone payment.

Final Condition	Account holder's balance will be reduced and bill is paid.
Alternative Path	-
Exception Path	Invalid cellular phone number: 1.Payment process is canceled. 2.Back to 1st action. Balance is not sufficient: 1.Payment process is canceled. 2.Back to 1st action.
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.9 Use Case Login Scenario

Identification		
Number	SRS-JOE-BANKING.FP9	
Name	Login	
Purpose	Validation process for limiting unregistered administrators.	
Description	Administrator inputs username and password to be validated by system.	
Actor	Administrator	
Main Scenario		
Initial Condition	-	
Actor Action		**System Reaction**
1. Administrator inputs username and password.		2. System validates the username and password.
Final Condition	Administrator enters the system and can use the functions.	
Alternative Path	-	
Exception Path	Incorrect username and password: 1. Administrator can not use the system. 2.Back to 1st action.	
Include	-	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.10 Use Case Manage Administrator Scenario

Identification		
Number	SRS-JOE-BANKING.FP10	
Name	Manage Administrator	
Purpose	Managing application administrator.	
Description	Administrator chooses menu for adding, updating, and deleting.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses menu for adding user.		2. System displays the adding user interface.
Final Condition	Administrator is able to use the adding function.	
Alternative Path	1. Administrator chooses menu for updating user. 2. Administrator chooses menu for deleting user.	
Exception Path	-	
Include	Login Use Case	
Extend	Adding Administrator Use Case Updating Administrator Use Case Deleting Administrator Use Case	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.11 Use Case Add Administrator Scenario

Identification		
Number	SRS-JOE-BANKING.FP11	
Name	Add Administrator	
Purpose	Adding new administrator data into the database.	
Description	Administrator inputs new administrator data for storing into database.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator fills new user data and takes save action.		2. System stores the data into the database.
Final Condition	New administrator data is saved.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.12 Use Case Update Administrator Scenario

Identification	
Number	SRS-JOE-BANKING.FP12
Name	Update Administrator
Purpose	Updating administrator data on the database.
Description	Administrator chooses administrator data for updating process.
Actor	Administrator

Main Scenario	
Initial Condition	Administrator is passed through Login.

Actor Action	**System Reaction**
1. Administrator chooses user data for updating.	2. System displays the data choosen.
3. Administrator changes the data and takes save action.	4. System updates the user data on the database.

Final Condition	Administrator data is updated.
Alternative Path	-
Exception Path	-
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.13 Use Case Delete Administrator Scenario

Identification		
Number	SRS-JOE-BANKING.FP13	
Name	Delete Administrator	
Purpose	Deleting administrator data from database.	
Description	Administrator chooses administrator data for deleting process.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses user data for deleting.		2. System displays the data choosen.
3. Administrator takes delete action.		4. System displays confirmation message.
5. Administrator chooses to delete the user data.		6. System deletes the user data from database
Final Condition	User data is deleted.	
Alternative Path	Actor chooses to cancel the deletion process and data is still on the database.	
Exception Path	Active administrator cannot be deleted: 1. Administrator cannot delete the data 2.Back to 1st action.	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.14 Use Case Manage Account Holder Scenario

Identification	
Number	SRS-JOE-BANKING.FP14
Name	Manage Account Holder
Purpose	Managing account holder data.
Description	Administrator chooses menu for adding, updating, and deleting.
Actor	Administrator
Main Scenario	
Initial Condition	Administrator is passed through Login.
Actor Action	**System Reaction**
1. Administrator chooses menu for adding account holder.	2. System displays the adding account holder interface.
Final Condition	Administrator is able to use the adding function.
Alternative Path	1. Administrator chooses menu for updating account holder. 2. Administrator chooses menu for deleting account holder.
Exception Path	-
Include	Login Use Case
Extend	Adding Account Holder Use Case Updating Account Holder Use Case

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.15 Use Case Add Account Holder Scenario

Identification		
Number	SRS-JOE-BANKING.FP15	
Name	Add Account Holder	
Purpose	Adding new account holder data into the database.	
Description	Administrator inputs new account holder data for storing into database.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator fills new account holder data and takes save action.		2. System stores the data into the database.
Final Condition	New account holder data is saved.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.16 Use Case Update Account Holder Scenario

Identification	
Number	SRS-JOE-BANKING.FP16
Name	Update Account Holder
Purpose	Updating account holder data on the database.
Description	Administrator chooses account holder data for updating process.
Actor	Administrator
Main Scenario	
Initial Condition	Administrator is passed through Login.
Actor Action	**System Reaction**
1. Administrator chooses account holder data for updating.	2. System displays the data choosen.
3. Administrator changes the data and takes save action.	4. System updates the account holder data on the database.
Final Condition	Account holder data is updated.
Alternative Path	-
Exception Path	-
Include	Login Use Case
Extend	-

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.17 Use Case Display Transactions Scenario

Identification		
Number	SRS-JOE-BANKING.FP17	
Name	Display Transactions	
Purpose	Choose menu for displaying transactions.	
Description	Administrator chooses menu for displaying the transactions (daily, monthly, or per account).	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses menu of displaying transactions.		2. System starts the displaying interface.
Final Condition	System displays the interface.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.18 Use Case Display Daily Transactions Scenario

Identification		
Number	SRS-JOE-BANKING.FP18	
Name	Display Daily Transactions	
Purpose	Displaying daily transactions.	
Description	Administrator chooses the date of the transactions for displaying transactions.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses the date of transactions.		2. System displays the transactions by date.
Final Condition	System displays the transactions by date.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.19 Use Case Display Monthly Transactions Scenario

Identification		
Number	SRS-JOE-BANKING.FP19	
Name	Display Monthly Transactions	
Purpose	Displaying monthly transactions.	
Description	Administrator chooses the month of the transactions for displaying transactions.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses the month of transactions.		2. System displays the transactions by month.
Final Condition	System displays the date by month.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.20 Use Case Display Account Transactions Scenario

Identification		
Number	SRS-JOE-BANKING.FP20	
Name	Display Account Transactions	
Purpose	Displaying transactions per account.	
Description	Administrator inputs the account number and chooses the date of the transactions.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator inputs the account number and chooses the date of transactions.		2. System displays the transactions by date of the selected account number.
Final Condition	System displays the transactions of selected account by date.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	-	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.21 Use Case Print Daily Report Scenario

Identification		
Number	SRS-JOE-BANKING.FP21	
Name	Print Daily Report	
Purpose	Printing daily report.	
Description	Administrator prints the daily report.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses to print daily report.		2. System prints the daily report.
Final Condition	Actor gets the print the daily report.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	Display Daily Transactions Use Case	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.22 Use Case Print Monthly Report Scenario

Identification		
Number	SRS-JOE-BANKING.FP22	
Name	Print Monthly Report	
Purpose	Print monthly report.	
Description	Administrator prints the monthly report.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses to print monthly report.		2. System prints monthly report.
Final Condition	Actor gets the monthly report.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	Display Monthly Transactions Use Case	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.2.2.23 Use Case Print Account Report Scenario

Identification		
Number	SRS-JOE-BANKING.FP23	
Name	Print Account Report	
Purpose	Print report per account.	
Description	Administrator prints the account report.	
Actor	Administrator	
Main Scenario		
Initial Condition	Administrator is passed through Login.	
Actor Action		**System Reaction**
1. Administrator chooses to print account report.		2. System prints account report.
Final Condition	Actor gets the monthly report.	
Alternative Path	-	
Exception Path	-	
Include	Login Use Case	
Extend	Display Account Transactions Use Case	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.3 Entity Relationship Diagram (ERD)

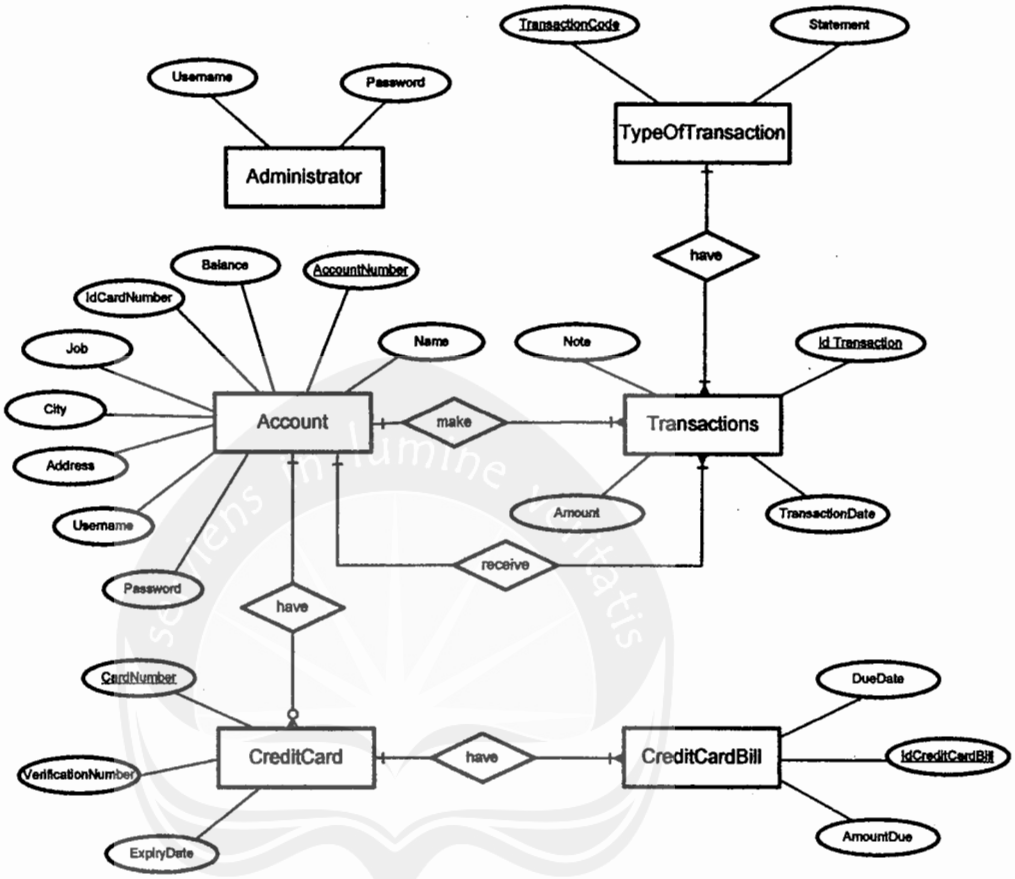

Figure 3.3 Entity Relationship Diagram

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

SDD

Software Design Description

The Design of

Java Object Oriented E-Banking

(JOE-BANKING)

Written by:

Sonny Hendrawan

02 07 03578

Department of Informatics Engineering

Faculty of Industrial Technology

Atma Jaya Yogyakarta University

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

List of Figures

This is a private document; no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

Java Object Oriented E-Banking

(JOE-BANKING)

1 Preface

1.1 Purpose

Software Design Description (SDD) is used to design the JOE-BANKING interfaces. This document is used by the programmer as a technical reference of JOE-BANKING implementation.

1.2 Problems Scope

JOE-BANKING (Java Object Oriented E-Banking) is a web-based application implementing JSP (Java Server Pages) for handling bank transactions over the internet. JOE-BANKING provides several functions which are:

a. Web Based
    1. Login Web
    2. Change Password
    3. Transfer
    4. View Account Information
    5. Pay Bill
        i. Pay Credit Card Bill
        ii. Pay Electricity Bill
        iii. Pay Cellular Phone Bill

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

b. Desktop Based

1. Login
2. Manage Administrator
    i. Add Administrator
    ii. Update Administrator
    iii. Delete Administrator
3. Manage Account Holder
    i. Add Account Holder
    ii. Update Account Holder
4. Display Transactions
    i. Display Daily Transactions
    ii. Display Monthly Transactions
    iii. Display Account Transactions
5. Print Report
    i. Print Daily Report
    ii. Print Monthly Report
    iii. Print Account Report

1.3 Definition and Acronym

- SRS stand for Software Requirements Specification. SRS is an analysis and specification of the application that will be made.
- JOE-BANKING stand for Java Object Oriented E-Banking. JOE-BANKING is the software that will be made.
- GUI stand for Graphical User Interface, is used for building the interface of the code.
- SDD stand for Software Design Description. SDD is the description of application design.
- SDD-JOE-BANKING.FPx is code that represents collaboration diagram of JOE-BANKING.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

1.4 References

The references that are used by this document:

- Hendrawan, Sonny. *SRS Java Object Oriented E-Banking*, UAJY, 2006.
- Jacobson, Ivar, Grady Booch, James Rumbaugh, *The Unified Software Development Process*, Addison Wesley, 2001.
- Deitel, Advanced *Java 2 Platform How to Program*, Prentice-Hall Inc, 2003
- NIIT, *Servlets & JSP Student Guide*, 2001.
- Keegan, Champenois, Crawley, Hunt, Webster, *NetBeans IDE Field Guide*, Prentice Hall, 2005.
- www.NetBeans.org

1.5 Overview

This document consists of 3 parts. The first part contains the purpose of SDD, problems scope, definitions, reference, and overview.

The second part contains architectural description from class diagram and classes specification. The classes derive from JOE-BANKING specification inside SRS.

The third part contains software requirement in a specific ways.

The owner of this document and the information inside is informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2 Architectural Description

2.1 Architectural Design

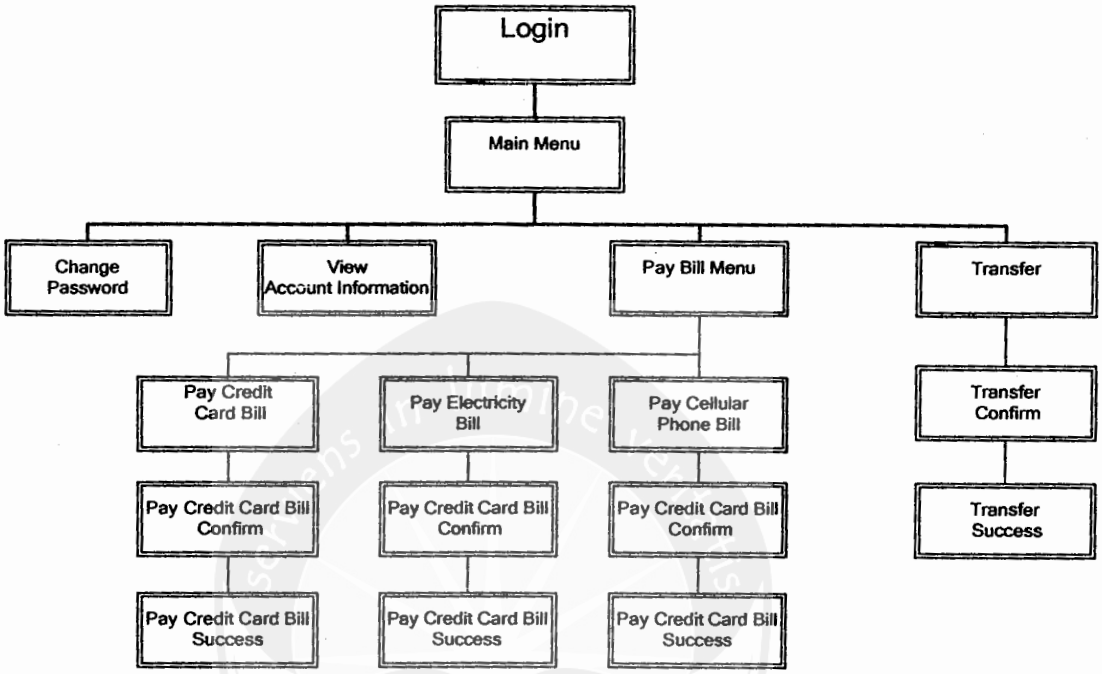

Figure 2.1 Architectural Web Design

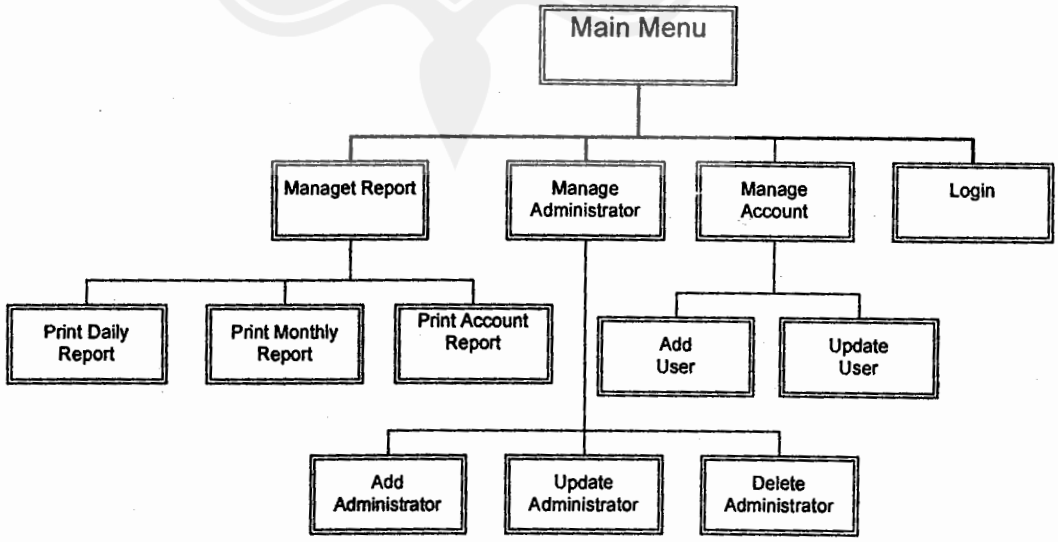

Figure 2.2 Architectural Administrator Design

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.2 Deployment Diagram

The Deployment diagram shows the physical layout of the network where the various component will reside. The deployment diagram for this application shown in below:

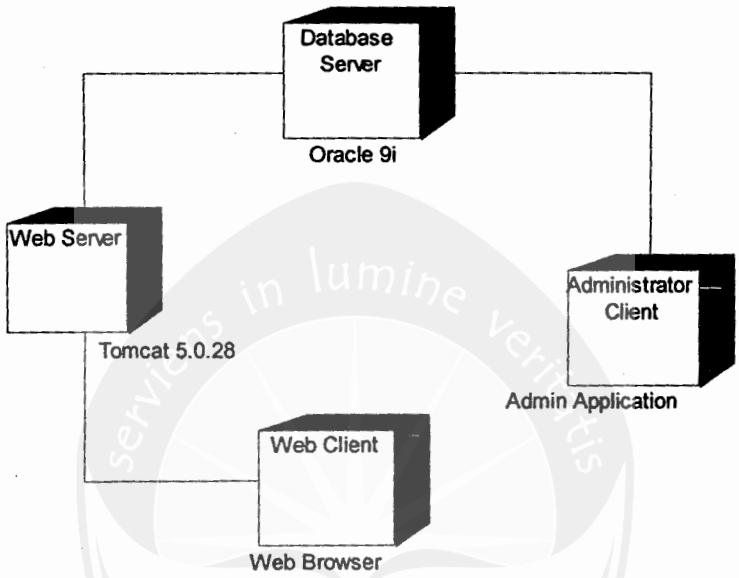

Figure 2.3 Deployment Diagram

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

The deployment diagram above shows that the DBMS, Oracle 9i is in the server side. Both web client and admin application handles will take care the client requests and access the database as a response.

2.2.1 *Database Server Node*

Node that contains Oracle 9i database which serves as storage mechanism of JOE-BANKING.

2.2.2 *Web Server Node*

Node that contains Tomcat 5.0.28 as a web-server of application.

2.2.3 *Web Client Node*

Node that is used by account holder to access the functionality of JOE-BANKING application over internet protocol. Web client node is a browser such as Internet Explorer, Netscape Navigator, Opera, etc.

2.2.4 *Administrator Client Node*

Node that is used by administrator to maintain the users of JOE-BANKING application. Administrator node is a desktop application node.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3 Use Case Realizations

2.3.1 Static Structure Diagrams

2.3.1.1 Class Diagram Package Web-Based

This is a private document; no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

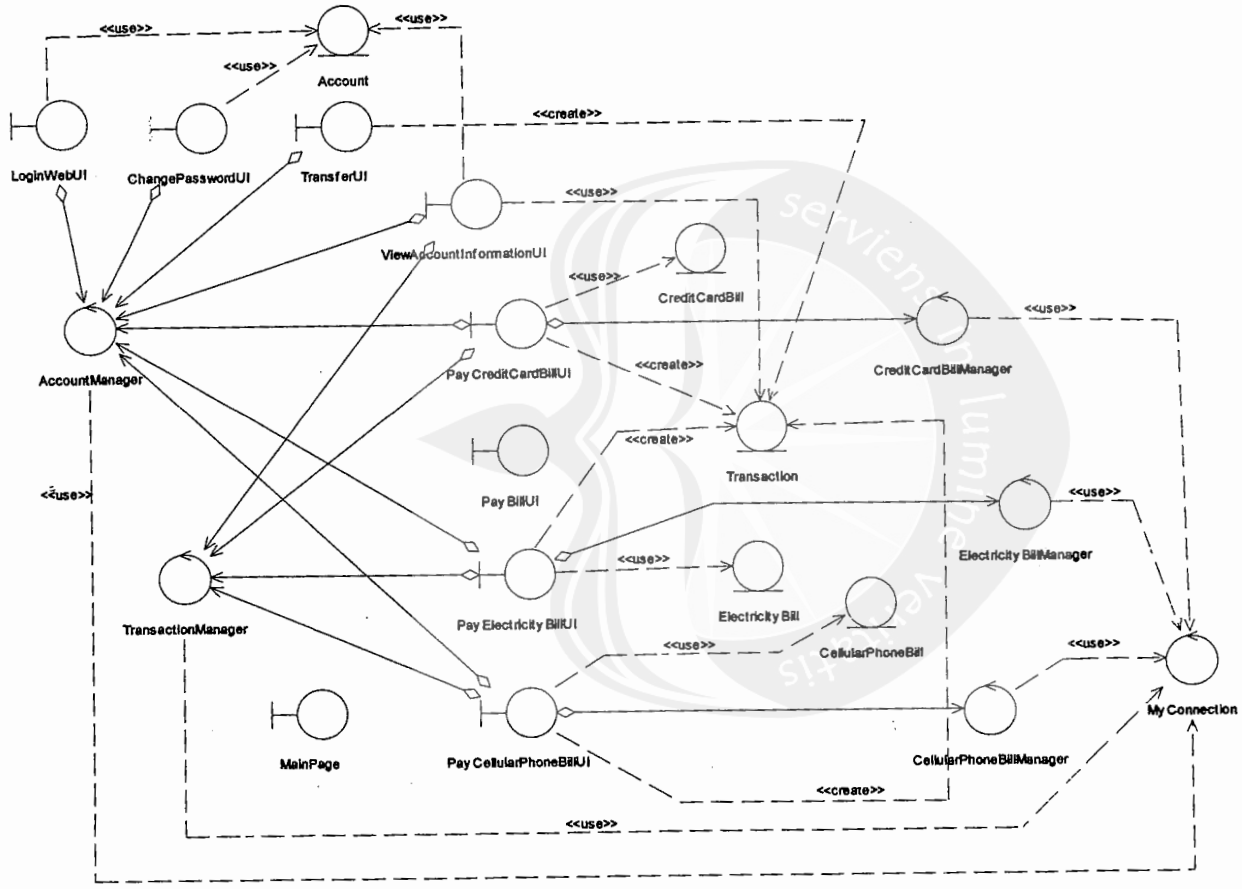

Figure 2.4 Class Diagram Package Web-Based

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.1.2 Class Diagram Package Desktop-Based

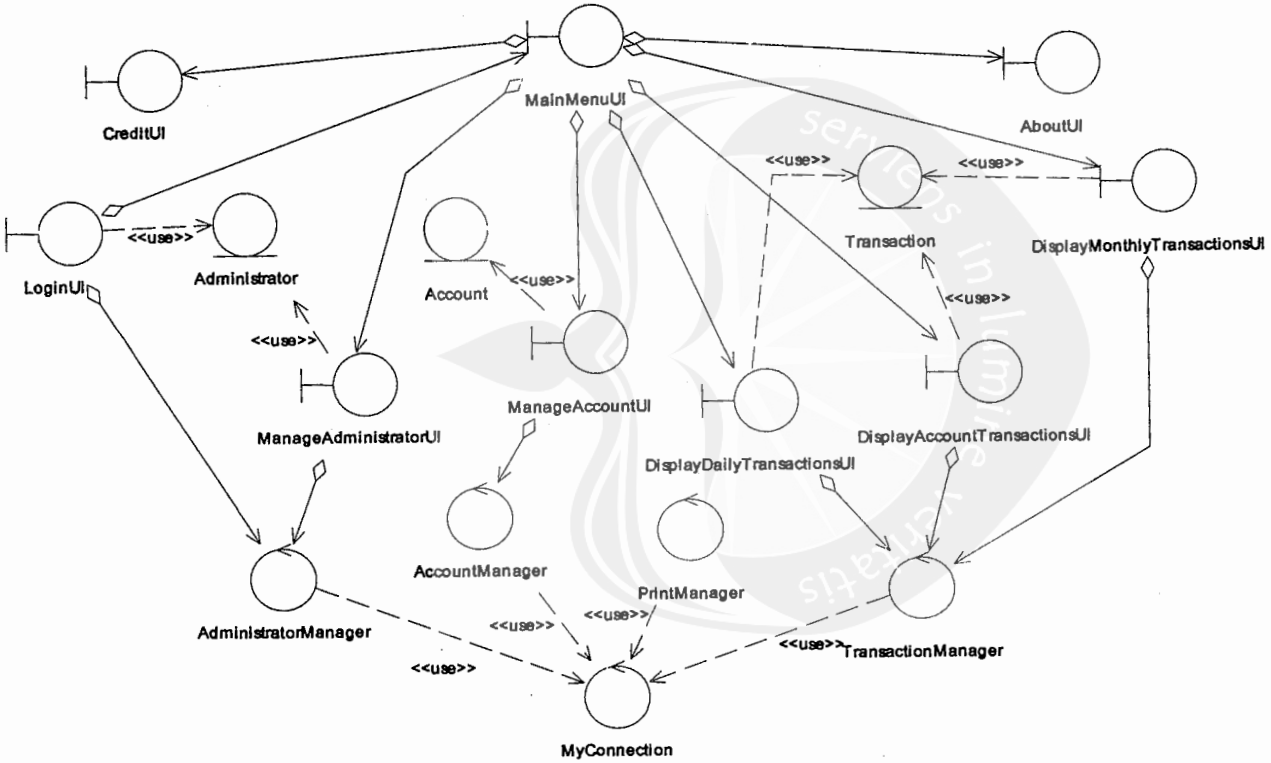

Figure 2.5 Class Diagram Package Desktop-Based

This is a private document; no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

2.3.2 Interaction Diagrams

2.3.2.1 Collaboration Diagram Design: Use Case Login Web (SDD-JOE-BANKING.FP1)

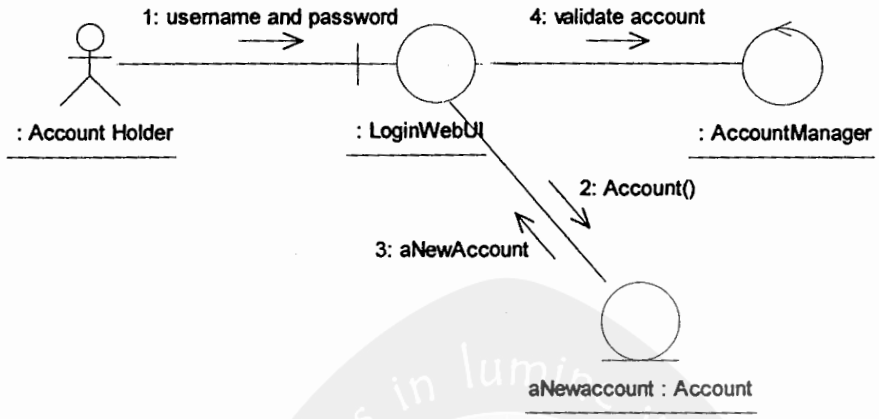

Figure 2.6 Collaboration Diagram: Use Case Login Web

2.3.2.2 Collaboration Diagram Design: Use Case Change Password (SDD-JOE-BANKING.FP2)

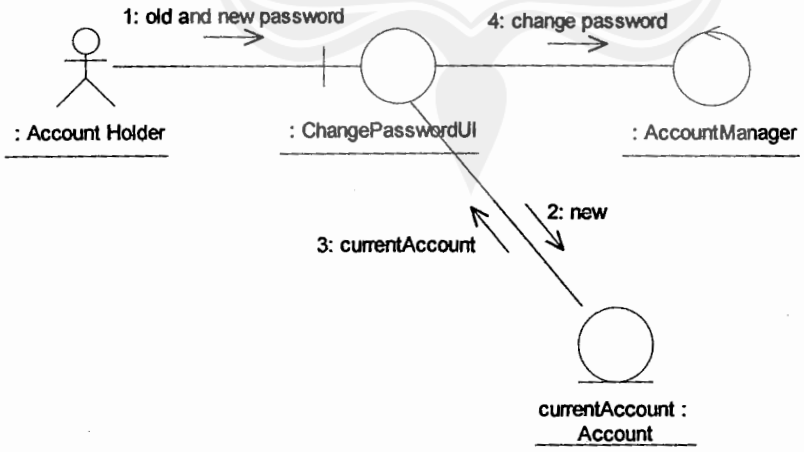

Figure 2.7 Collaboration Diagram: Use Case Change Password

This is a private document; no one can copy or reproduce it without license from Department of Informatics Engineering UAJY.

2.3.2.3 Collaboration Diagram Design: Use Case Transfer (SDD-JOE-BANKING.FP3)

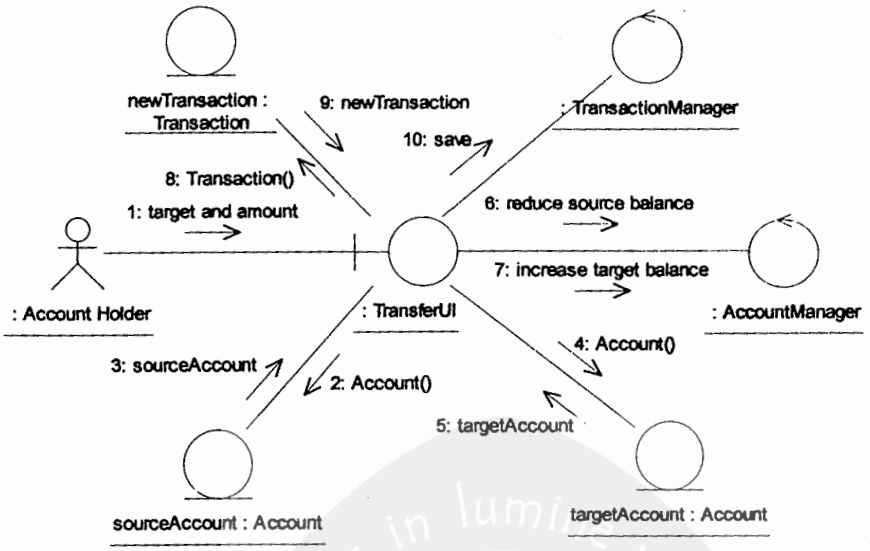

Figure 2.8 Collaboration Diagram: Use Case Transfer

2.3.2.4 Collaboration Diagram Design: Use Case View Account Information (SDD-JOE-BANKING.FP4)

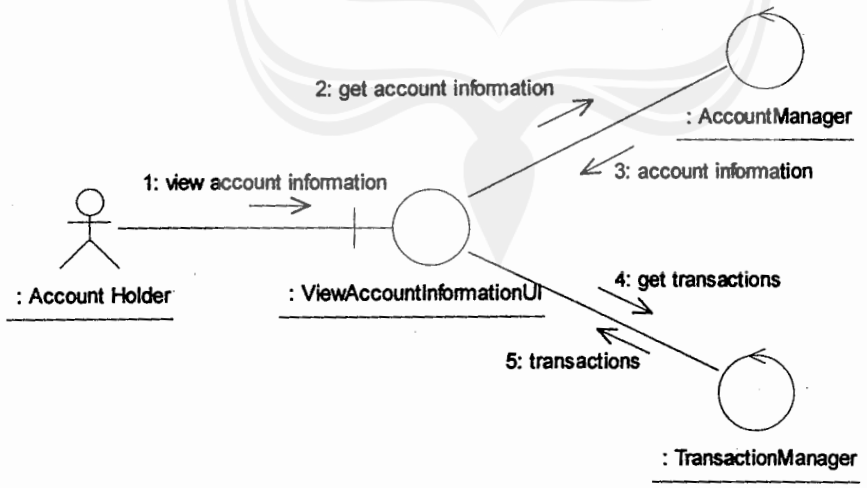

Figure 2.9 Collaboration Diagram: Use Case View Account Information

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.5 Collaboration Diagram Design: Use Case Pay Bill (SDD-JOE-BANKING.FP5)

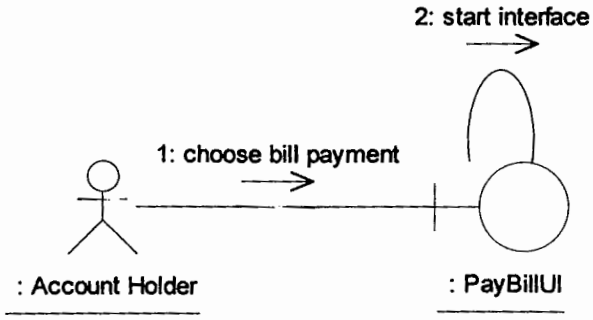

Figure 2.10 Collaboration Diagram: Use Case Pay Bill

2.3.2.6 Collaboration Diagram Design: Use Case Pay Credit Card Bill (SDD-JOE-BANKING.FP6)

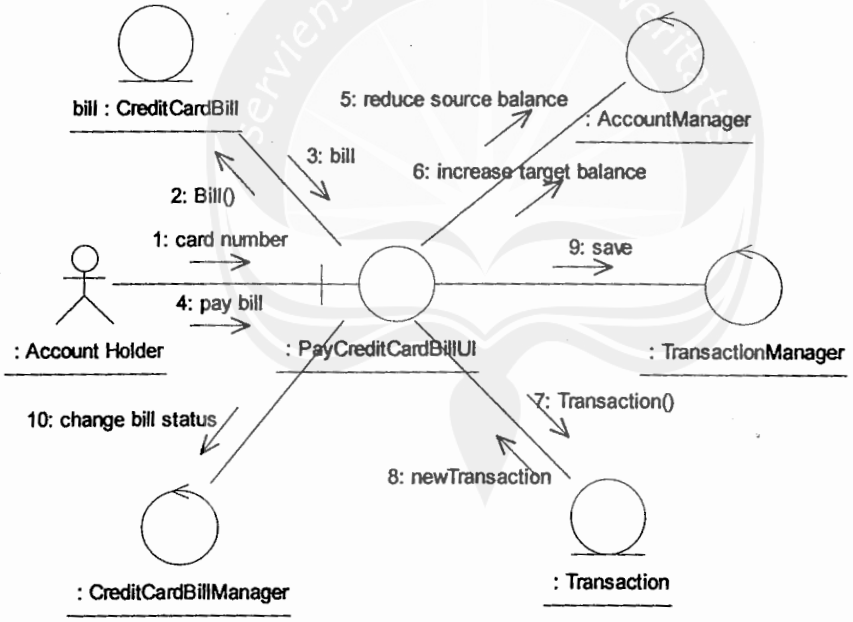

Figure 2.11 Collaboration Diagram: Use Case Pay Credit Card Bill

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.7 Collaboration Diagram Design: Use Case Pay Electricity Bill (SDD-JOE-BANKING.FP7)

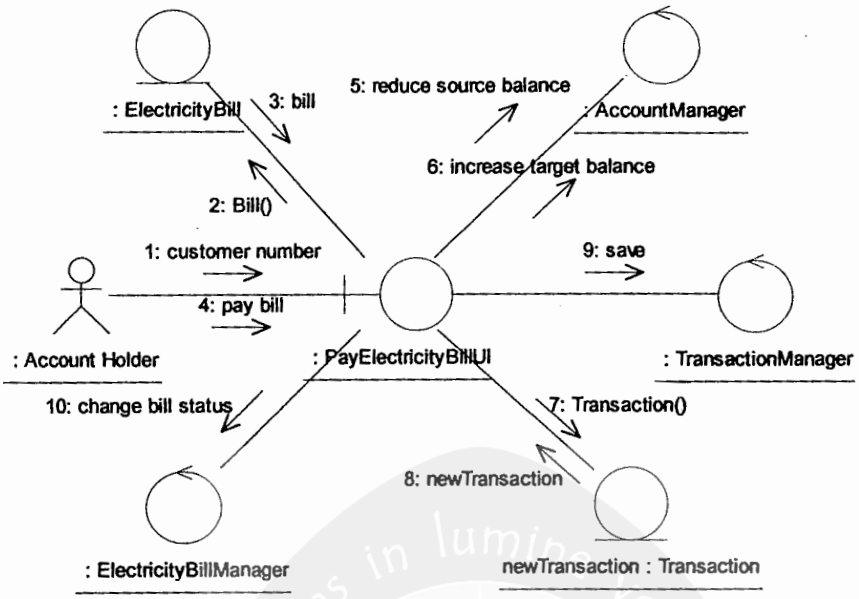

Figure 2.12 Collaboration Diagram: Use Case Pay Electricity Bill

2.3.2.8 Collaboration Diagram Design: Use Case Pay Cellular Phone Bill (SDD-JOE-BANKING.FP8)

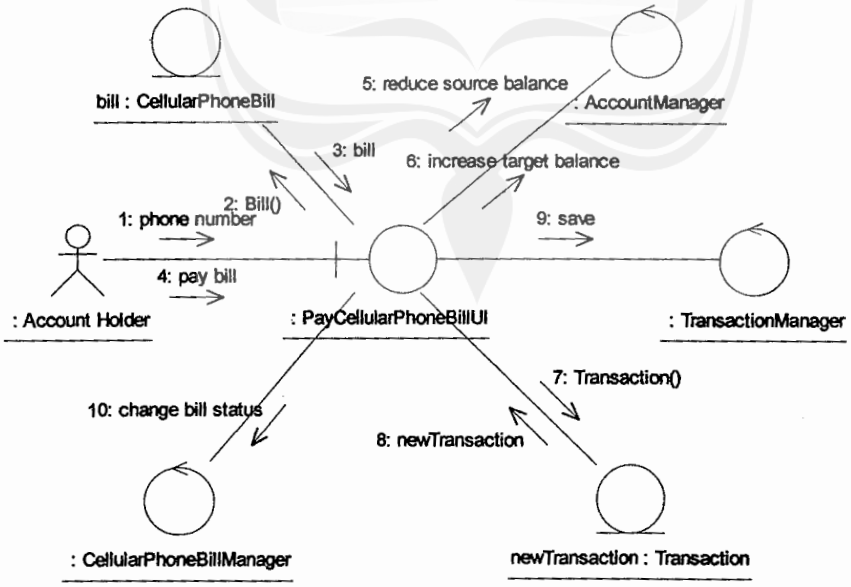

Figure 2.13 Collaboration Diagram: Use Case Pay Cellular Phone Bill

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.9 Collaboration Diagram Design: Login (SDD-JOE-Banking.FP9)

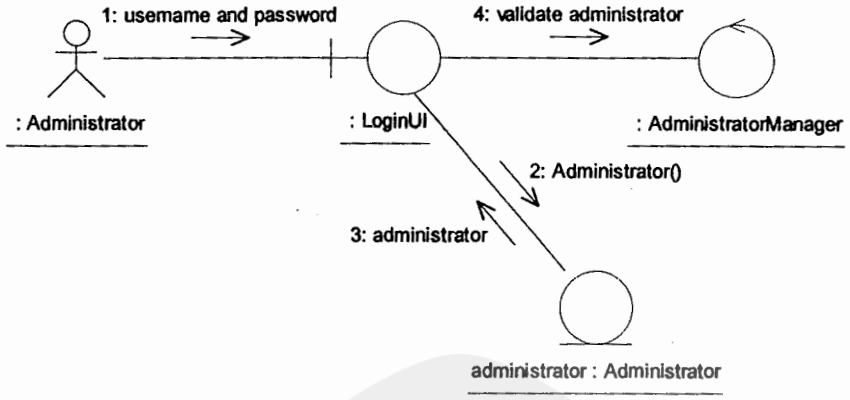

Figure 2.14 Collaboration Diagram: Use Case Login

2.3.2.10 Collaboration Diagram Design: Manage Administrator (SDD-JOE-BANKING.FP10)

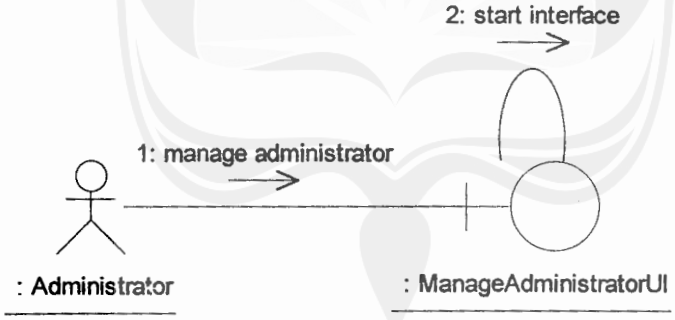

Figure 2.15 Collaboration Diagram: Use Case Manage Administrator

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.11 Collaboration Diagram Design: Use Case Add Administrator (SDD-JOE-BANKING.FP11)

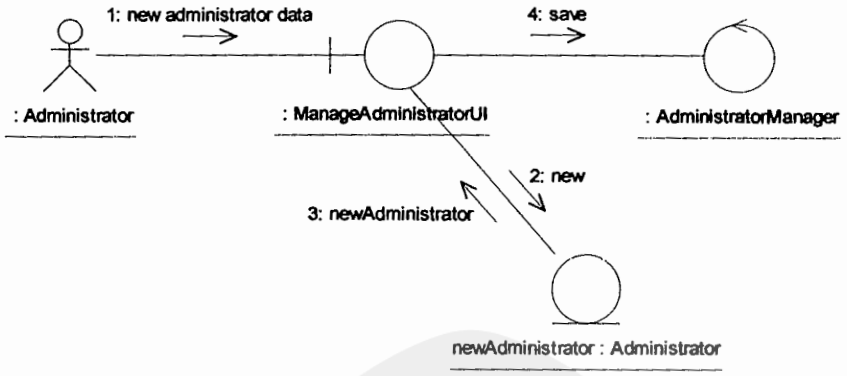

Figure 2.16 Collaboration Diagram: Use Case Add Administrator

2.3.2.12 Collaboration Diagram Design: Use Case Update Administrator (SDD-JOE-BANKING.FP5)

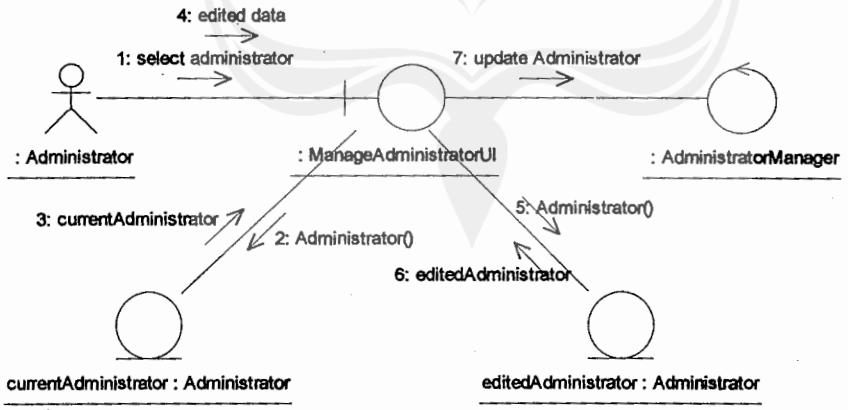

Figure 2.17 Collaboration Diagram: Use Case Update Administrator

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.13 Collaboration Diagram Design: Use Case Delete Administrator (SDD-JOE-BANKING.FP13)

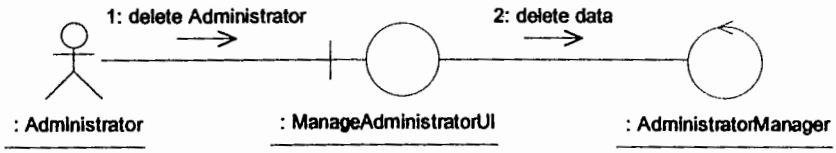

Figure 2.18 Collaboration Diagram: Use Case Delete Administrator

2.3.2.14 Collaboration Diagram Design: Use Case Manage Account Holder (SDD-JOE-Banking.FP14)

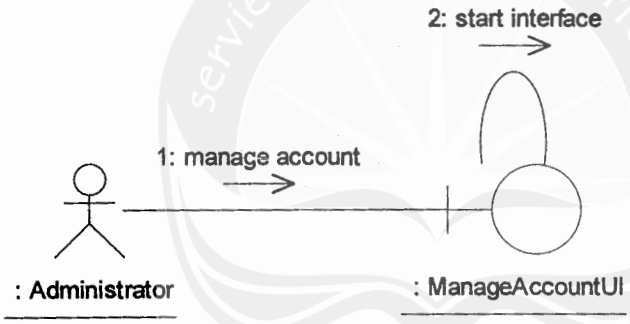

Figure 2.19 Collaboration Diagram: Use Case Manage Account Holder

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.15 Collaboration Diagram Design: Use Case Add Account Holder (SDD-JOE-Banking.FP15)

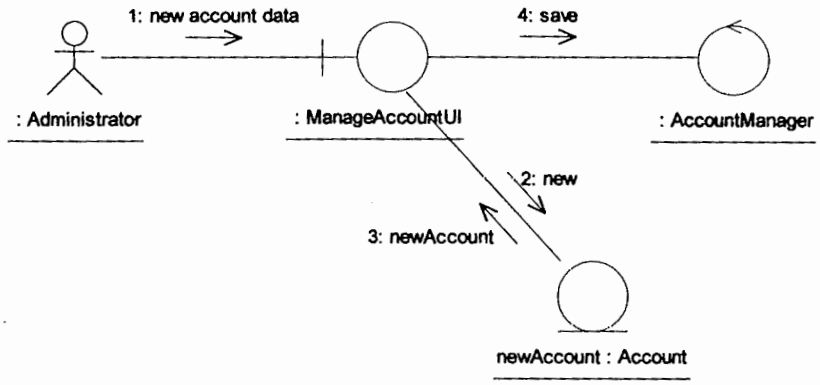

Figure 2.20 Collaboration Diagram: Use Case Add Account Holder

2.3.2.16 Collaboration Diagram Design: Use Case Update Account Holder (SDD-JOE-Banking.FP16)

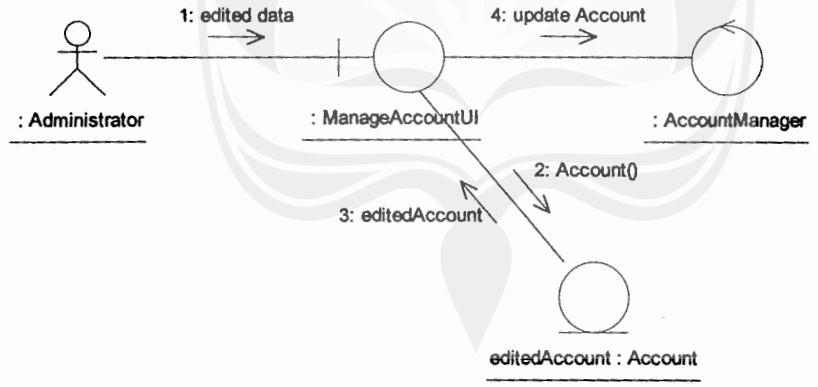

Figure 2.21 Collaboration Diagram: Use Case Update Account Holder

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.17 Collaboration Diagram Design: Use Case Display Transactions (SDD-JOE-BANKING.FP17)

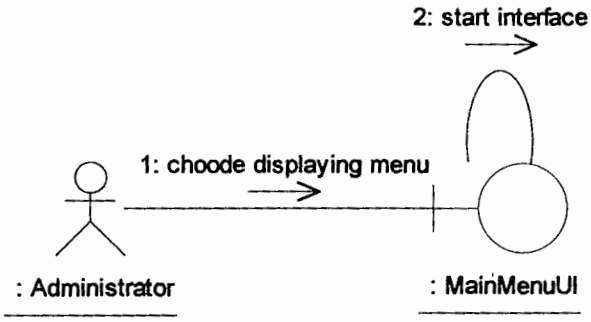

Figure 2.22 Collaboration Diagram: Use Case Display Transactions

2.3.2.18 Collaboration Diagram Design: Use Case Display Daily Transactions (SDD-JOE-BANKING.FP18)

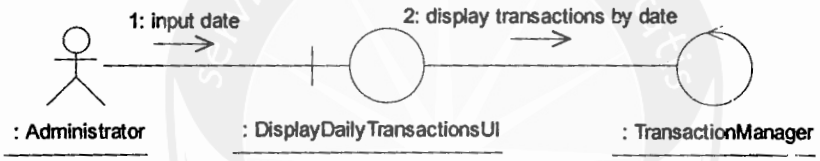

Figure 2.23 Collaboration Diagram: Use Case Display Daily Transactions

2.3.2.19 Collaboration Diagram Design: Use Case Display Monthly Transactions (SDD-JOE-BANKING.FP19)

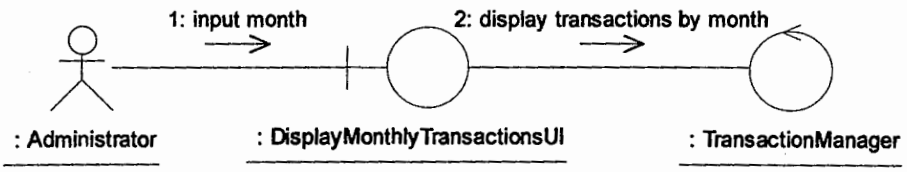

Figure 2.24 Collaboration Diagram: Use Case Display Monthly Transactions

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.20 Collaboration Diagram Design: Use Case Display Account Transactions (SDD-JOE-BANKING.FP20)

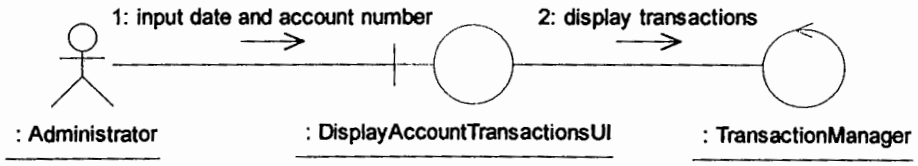

Figure 2.25 Collaboration Diagram: Use Case Display Account Transactions

2.3.2.21 Collaboration Diagram Design: Use Case Print Daily Report (SDD-JOE-BANKING.FP21)

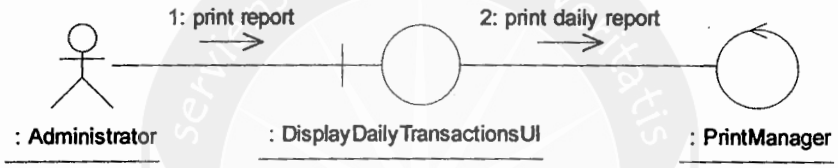

Figure 2.26 Collaboration Diagram: Use Case Print Daily Report

2.3.2.22 Collaboration Diagram Design: Use Case Print Monthly Report (SDD-JOE-BANKING.FP22)

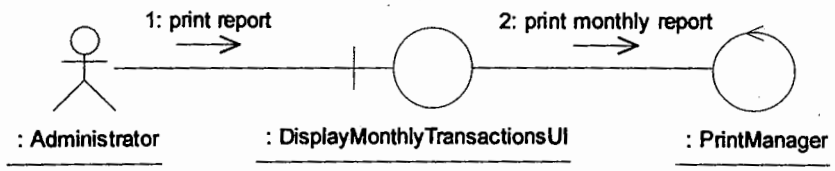

Figure 2.27 Collaboration Diagram: Use Case Print Monthly Report

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.3.2.23 Collaboration Diagram Design: Use Case Print Account Report (SDD-JOE-BANKING.FP21)

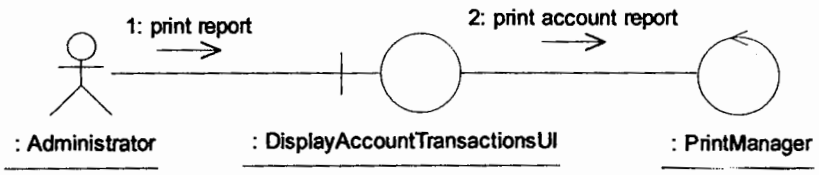

Figure 2.28 Collaboration Diagram: Use Case Print Account Report

2.4 Class Diagram Specific Descriptions

2.4.1 Specific Design Class LoginWebUI

LoginWebUI	**<<boundary>>**
- manager : AccountManager Manager which validates the account holder data.	
- LoginWebUI() Default constrictor for LoginWebUI class.	

2.4.2 Specific Design Class ChangePasswordUI

ChangePasswordUI	**<<boundary>>**
- manager : AccountManager Manager which changes the account holder data.	
- ChangePasswordUI() Default constrictor for ChangePasswordUI class.	

2.4.3 Specific Design Class TransferUI

TransferUI	**<<boundary>>**
- manager 1 : AccountManager Manager which manipulates the account data on the database. - manager 2 : TransactionManager Manager which saves the transaction.	
- TransferUI() Default constrictor for TransferUI class.	

2.4.4 Specific Design Class ViewAccountInformationUI

ViewAccountInformationUI	**<<boundary>>**

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

- manager 1 : AccountManager Manager which displays the account data. - manager 2 : TransactionManager Manager which displays the transaction.
- ViewAccountInformationUI() Default constrictor for ViewAccountInformationUI class.

2.4.5 Specific Design Class PayBillUI

PayBillUI	**<<boundary>>**
- PayBillUI() Default constrictor for PayBillUI class.	

2.4.6 Specific Design Class PayCreditCardBillUI

PayCreditCardBillUI	**<<boundary>>**
- manager 1 : CreditCardBillManager Manager which displays the bill data. - manager 2 : AccountManager Manager which reduces the source balance and increases the target balance for credit card bill payment. - manager 3 : TransactionManager Manager which saves the transaction.	
- PayCreditCardBillUI() Default constrictor for PayCreditCardBillUI class.	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.7 Specific Design Class PayElectricityBillUI

PayElectricityBillUI	<<boundary>>
- manager 1 : ElectricityBillManager Manager which displays the bill data. - manager 2 : AccountManager Manager which reduces the source balance and increases the target balance for electricity bill payment. - manager 3 : TransactionManager Manager which saves the transaction.	
- PayElectricityBillUI() Default constrictor for PayElectricityBillUI class.	

2.4.8 Specific Design Class PayCellularPhoneBillUI

PayCellularPhoneBillUI	<<boundary>>
- manager 1 : CellularPhoneBillManager Manager which displays the bill data. - manager 2 : AccountManager Manager which reduces the source balance and increases the target balance for cellular phone bill payment. - manager 3 : TransactionManager Manager which saves the transaction.	
- PayCellularPhoneBillUI() Default constrictor for PayCellularPhoneBillUI class.	

2.4.9 Specific Design Class ManageAdministratorUI

ManageUserUI	<<boundary>>
- manager : AdministratorManager Manager which manages administrator data manipulation.	
- ManageAdministratorUI() Default constrictor for manageUserUI class.	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.10 Specific Design Class ManageAccountUI

ManageAccountUI	**<<boundary>>**
- manager : AccountManager Manager which displays the transaction.	
- ManageAccountUI() Default constrictor for ManageAccountUI class.	

2.4.11 Specific Design Class DisplayDailyTransactionsUI

DisplayDailyTransactionsUI	**<<boundary>>**
- manager : TransactionManager Manager which displays the transaction.	
- DisplayDailyTransactionsUI() Default constrictor for DisplayDailyTransactionsUI class.	

2.4.12 Specific Design Class DisplayMonthlyTransactionsUI

DisplayMonthlyTransactionsUI	**<<boundary>>**
- manager : TransactionManager Manager which displays the transaction.	
- DisplayMonthlyTransactionsUI() Default constrictor for DisplayMonthlyTransactionsUI class.	

2.4.13 Specific Design Class DisplayAccountTransactionsUI

DisplayAccountTransactionsUI	**<<boundary>>**
- manager : TransactionManager Manager which displays the transaction.	
- DisplayAccountTransactionsUI() Default constrictor for DisplayAccountTransactionsUI class.	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.14 Specific Design Class Administrator

Administrator	**<<entity>>**
- username : String The attribute for user name. - password : String The attribute for user password.	
- Administrator() The default constructor for User class. - Administrator (String username, String password) Parameterized constructor for Administrator class. Accessor method : - getUserName() : String - getPassword() : String Mutator method : - setUserName(String username) - setPassword(String password)	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.15 Specific Design Class Account

Account	<<entity>>
- accountNumber : String The attribute for account number. - name : String The attribute for account holder's name. - address : String The attribute for account holder's address. - city : String The attribute for account holder's city. - job : String The attribute for account holder's job. - idCardNumber : String The attribute for account holder's id card number. - balance : Double The attribute for account balance. - username : String The attribute for account holder's username. - password : String The attribute for account holder's password.	
- Account() The default constructor for Account class. - Account (String accountNumber, String name, String address, string city, String job, String idCardNumber, Double balance, String username, String password) Parameterized constructor for Account class. Accessor method : - getAccountNumber() : String - getName() : String - getAddress() : String - getCity() : String - getJob() : String - getIdCardNumber() : String - getBalance() : Double - getUserName() : String - getPassword() : String Mutator method :	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

- setAccountNumber(String accountNumber) - setName(String name) - setAddress(String address) - setCity(String city) - setJob(String job) - setIdCardNumber(String idCardNumber) - setBalance(Double balance) - setUserName(String username) - setPassword(String password)

2.4.16 Specific Design Class Transaction

Transaction	**<<entity>>**
- transactionId : String The attribute for transaction id. - transactionDate : Date The attribute for transaction date. - source : String The attribute for transaction source. - target : String The attribute for transaction target. - amount : Double The attribute for transaction amount. - note : String The attribute for transaction note.	
- Transaction() The default constructor for Transaction class. - Transaction(String transactionId, Date transactiondate, String source, String target, Double amount, String note) Parameterized constructor for Transaction class. Accessor method : - getTransactionId() : String - getTransactionDate() : Date - getSource() : String - getTarget() : String - getAmount() : Double - getNote() : String Mutator method : - setTransactionId(String transactionId)	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

- setTransactionDate(Date transactionDate) - setSource(String source) - setTarget(String target) - setAmount(Double amount) - setNote(String note)

2.4.17 Specific Design Class CreditCardBill

CreditCardBill	**<<entity>>**
- IdCreditCardBill : String The attribute for credit card bill id. - cardNumber : String The attribute for card number. - dueDate : String The attribute for bill's due date. - amountDue : Double The attribute for bill amount. - status : String The attribute for bill's status	
- CreditCardBill() The default constructor for credit card bill class. - CreditCardBill(String IdCreditCardBill, String cardNumber, String dueDate, Double amountDue, String status) Parameterized constructor for credit card bill class. Accessor method : - getIdCreditCardBill() : String - getCardNumber() : String - getDueDate() : String - getAmountDue() : Double - getStatus() : String Mutator method : - setIdCreditCardBill(String IdCreditCardBill) - setCardNumber(String cardNumber) - setDueDate(String dueDate) - setAmountDue(Double amountDue) - setStatus(string status)	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.18 Specific Design Class ElectricityBill

ElectricityBill	**<<entity>>**
- IdElectricityBill : String The attribute for electricity bill id. - custNumber : String The attribute for customer number. - dueDate : String The attribute for bill's due date. - amountDue : Double The attribute for bill amount. - status : String The attribute for bill's status	
- ElectricityBill() The default constructor for electricity bill class. - ElectricityBill(String IdCreditCardBill, String cardNumber, String dueDate, Double amountDue, String status) Parameterized constructor for electricity bill class. Accessor method : - getIdElectricityBill() : String - getCustNumber() : String - getDueDate() : String - getAmountDue() : Double - getStatus() : String Mutator method : - setIdElectricityBill(String IdElectricityBill) - setCustNumber(String custNumber) - setDueDate(String dueDate) - setAmountDue(Double amountDue) - setStatus(string status)	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.19 Specific Design Class CellularPhoneBill

CellularPhoneBill	**<<entity>>**
- IdCellularPhoneBill : String The attribute for cellular phone bill id. - phoneNumber : String The attribute for phone number. - dueDate : String The attribute for bill's due date. - amountDue : Double The attribute for bill amount. - status : String The attribute for bill's status	
- CellularPhoneBill() The default constructor for cellular phone bill class. - CellularPhoneBill(String IdCellularPhoneBill, String phoneNumber, String dueDate, Double amountDue, String status) Parameterized constructor for cellular phone bill class. Accessor method : - getIdCellularPhoneBill() : String - getPhoneNumber() : String - getDueDate() : String - getAmountDue() : Double - getStatus() : String Mutator method : - setIdCellularPhoneBill(String IdCellularPhoneBill) - setPhoneNumber(String phoneNumber) - setDueDate(String dueDate) - setAmountDue(Double amountDue) - setStatus(string status)	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.20 Specific Design Class AdministratorManager

AdministratorManager	<<control>>
- AdministratorManager() The default constructor for AdministratorManager class. - addAdministrator(Administrator admin) Method for saving new administrator data into database. - updateAdministrator(Administrator currentadmin, Administrator editedAdmin) Method for updating administrator data. - deleteAdministrator(String username) Method for deleting administrator data from database. - validateAdministrator(String username, String password) Method for validating administrator data.	

2.4.21 Specific Design Class AccountManager

AccountManager	<<control>>
- AccountManager() The default constructor for AccountManager class. - increaseTargetBalance(String accountNumber) Method for increasing account balance. - reduceSourceBalance(String accountNumber) Method for decreasing account balance. - displayAccountInformation(String accountNumber) Method for displaying account information.	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.22 Specific Design Class TransactionManager

TransactionManager	<<control>>
- TransactionManager() The default constructor for TransactionManager class. - addTransaction(Transaction transaction) Method for saving a new transaction into database. - displayDailyTransactions(String date) Method for displaying daily transaction. - displayMonthlyTransactions(String month) Method for displaying monthly transaction. - displayDailyTransactions(String date, String accNumber) Method for displaying per account transaction.	

2.4.23 Specific Design Class CreditCardBillManager

CreditCardBillManager	<<control>>
- BillManager() The default constructor for CreditCardBillManager class. - getBill(String cardNumber) Method for getting bill data from database. - changeBillStatus() Method for changing bill status (paid or not).	

2.4.24 Specific Design Class ElectricityBillManager

ElectricityBillManager	<<control>>
- BillManager() The default constructor for ElectricityBillManager class. - getBill(String custNumber) Method for getting bill data from database. - changeBillStatus() Method for changing bill status (paid or not).	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

2.4.25 Specific Design Class CellularPhoneBillManager

CellularPhoneBillManager	<<control>>
- CellularPhoneBillManager() The default constructor for CellularPhoneBillManager class. - getBill(String custNumber) Method for getting bill data from database. - changeBillStatus() Method for changing bill status (paid or not).	

2.4.26 Specific Design Class PrintManager

PrintManager	<<control>>
- PrintManager() The default constructor for PrintManager class. - printDailyReport() Method for printing a daily report. - printMonthlyReport() Method for printing a monthly report. - printAccountReport() Method for printing an account report.	

2.4.27 Specific Design Class MyConnection

MyConnection	<<control>>
- MyConnection() The default constructor for MyConnection class. - getConnection() Method for connecting application to database.	

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3 Interface Descriptions

3.1 Main Menu Interface Description

This Interface is the main form of JOE-BANKING (for Administrator). It has forms such as Login Interface, User Management Interface, Administrator Management Interface, Report Management Interface, Credit Interface, and About Interface. These are the functions on the menu bar:

1. Menu
   a. If submenu Login clicked, then Login Interface will appear.
   b. If submenu Admin Management clicked, then the Admin Management Interface will appear.
   c. If submenu User Management clicked, then new User Management Interface will appear.
   d. If submenu Report Management clicked, then the submenu will appear to choose daily report, monthly report, or account report.
   e. If submenu Logout clicked, then the administrator will be logout from the system.
   f. If submenu Exit clicked, then the application will be terminated.
2. Help
   a. If submenu About clicked, then About Interface will appear.
   b. If submenu Credits clicked, then Credit Interface will appear.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

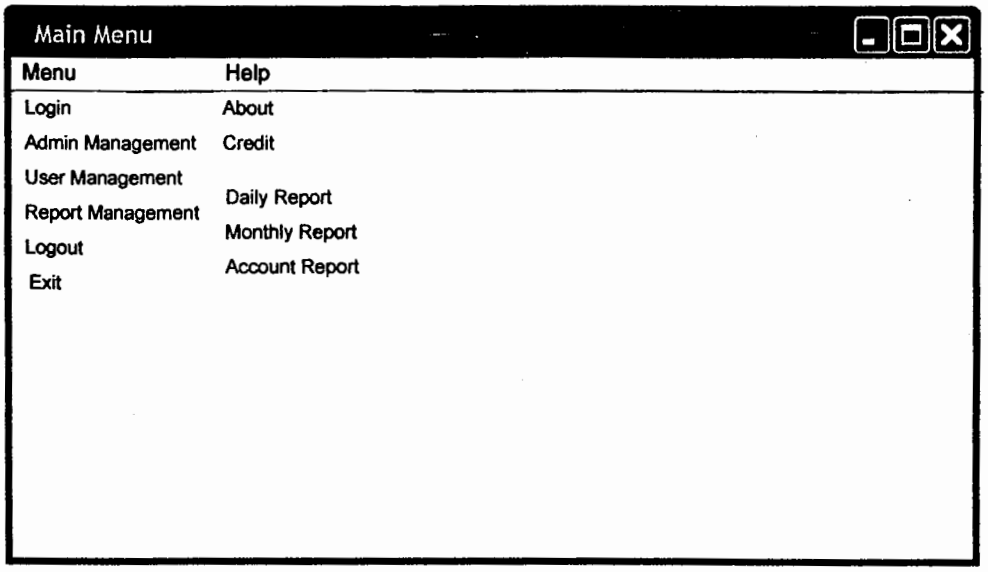

Figure 3.1 Main Menu Interface

3.2 About Interface Description

This interface is intended to give information about the system.

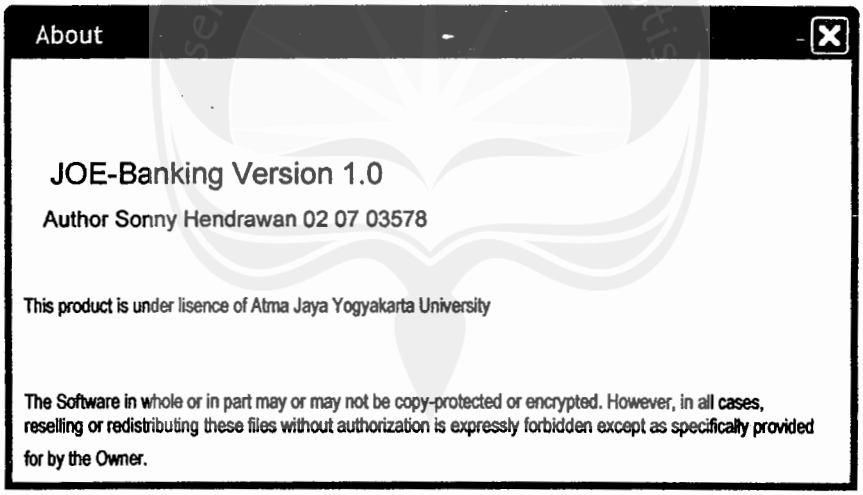

Figure 3.2 About Interface

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.3 Credits Interface Description

This interface is intended to give information about the programmer.

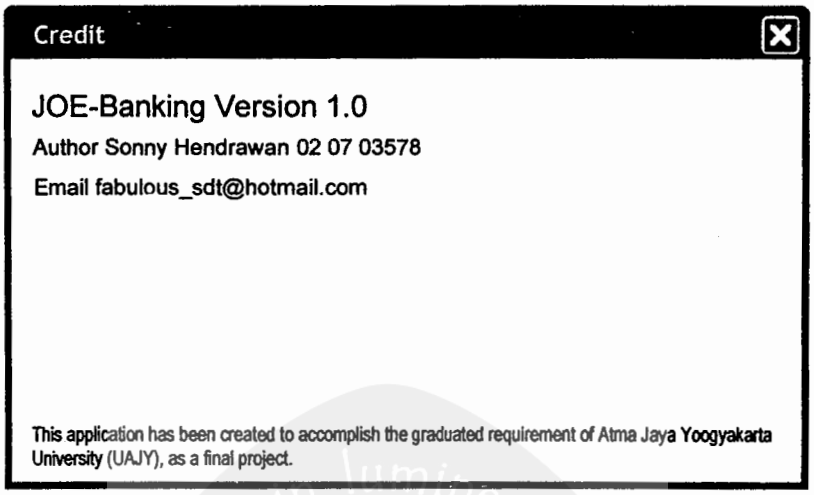

Figure 3.3 Credits Interface

3.4 Login Interface Description

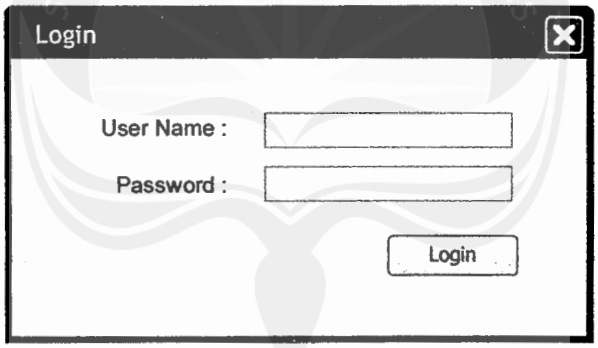

Figure 3.4 Login Interface

This interface is intended to limit unregistered administrator. Administrator has to fill in username and password to be validated by system after button Login is clicked. If inputs are valid then system can be used. Otherwise, system cannot be used and error message is shown.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.5 Administrator Management Interface Description

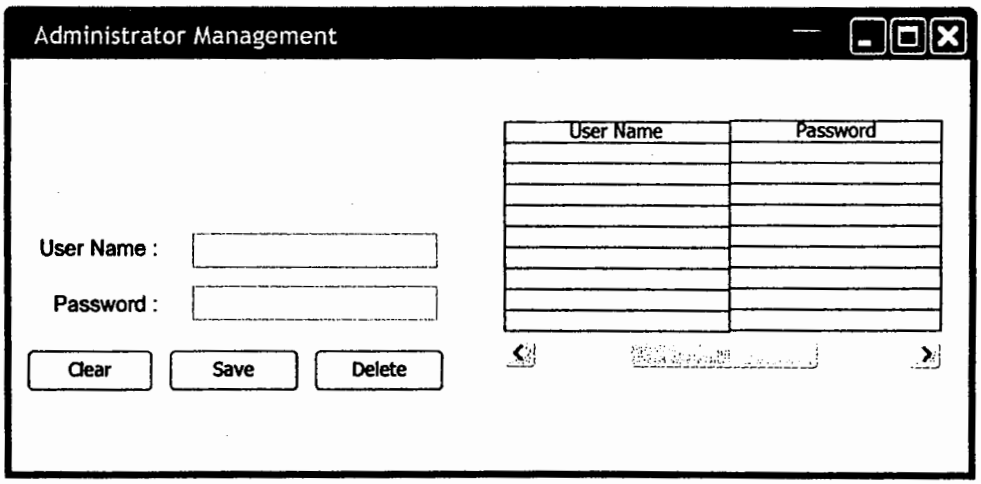

Figure 3.5 Administrator Management Interface

This interface is intended for administrator management process. It allows administrator to add another new administrator, edit, and delete the existing administrator. A new administrator data can be added by entering new username and password then clicking Save button. For editing existing administrator, clicked on the data in the data grid and the selected data will appear in textboxes on the left side. Change the value and then clicked Save button, the existing data will be updated. For deleting, the process is the same as editing process, but clicking Delete button and data will be deleted.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.6 Account Holder Management Interface Description

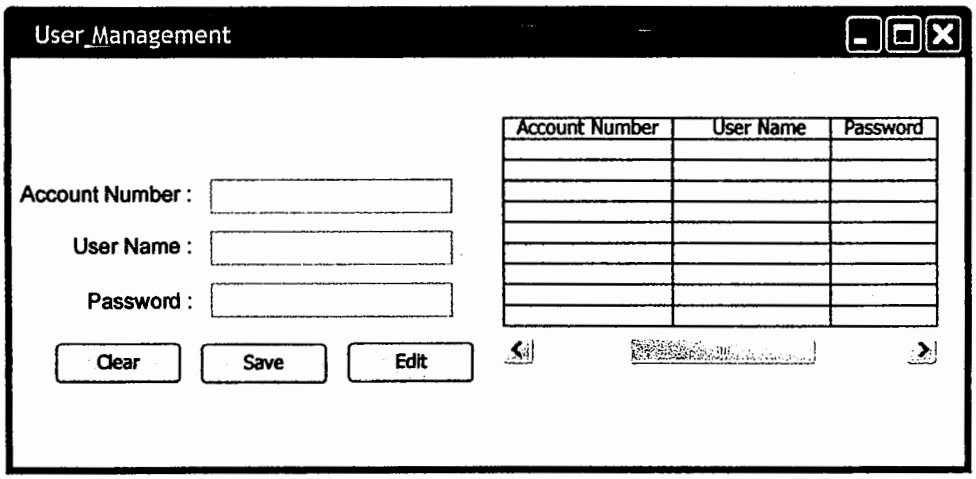

Figure 3.6 Account Holder Management Interface

This interface is intended to manage the account holder data which use the web-based application. Administrator will give the username and password for each account holder so that they can use the web-based application by entering their username and password. Administrator chooses the account holder on the data grid and fills the username and password in the textboxes. To save, click Save button. For editing, just like the first step, change the value and click Save button to update.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.7 Daily Report Interface Description

This interface is intended for reporting process. The administrator chooses the date and clicks Search button. To print the report, click Print button.

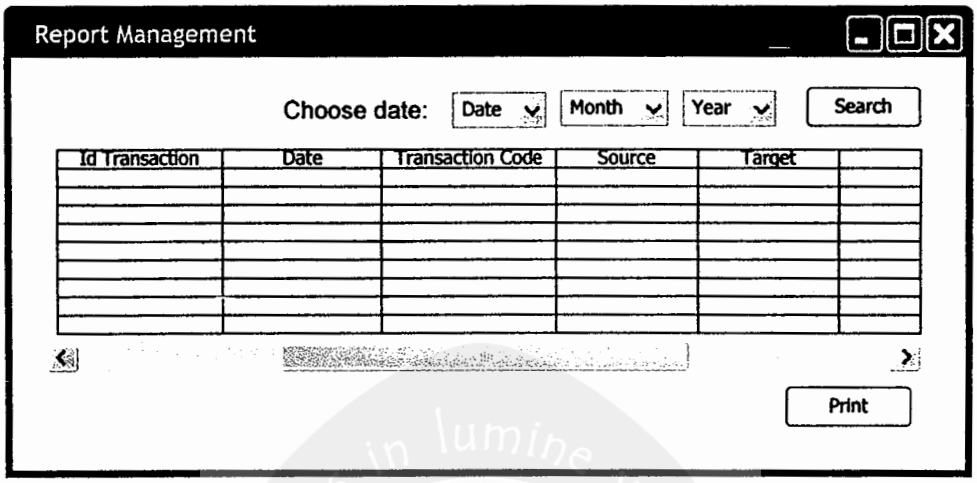

Figure 3.7 Daily Report Interface

3.8 Monthly Report Interface Description

This interface is intended for reporting process. The administrator chooses the month and clicks Search button. To print the report, click Print button.

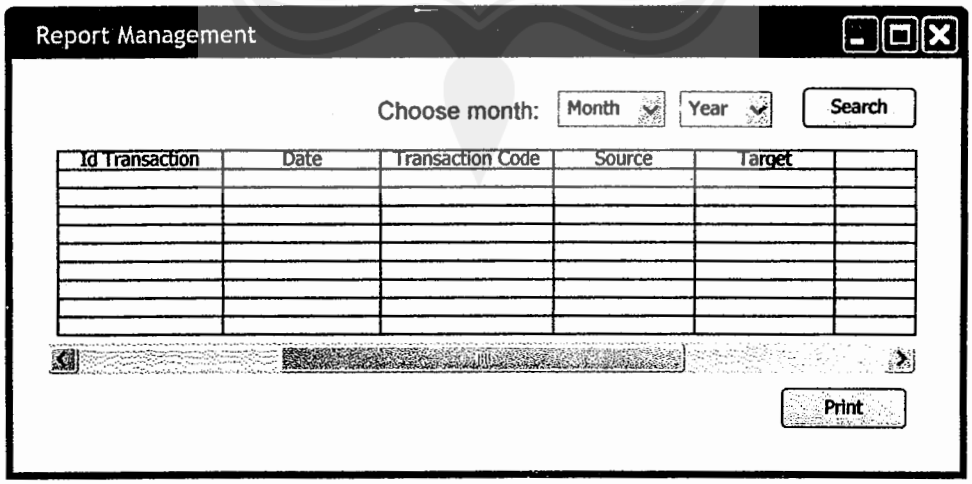

Figure 3.8 Monthly Report Interface

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.9 Account Report Interface Description

This interface is intended for reporting process. The administrator inputs the account number and chooses the date and clicks Search button. To print the report, click Print button.

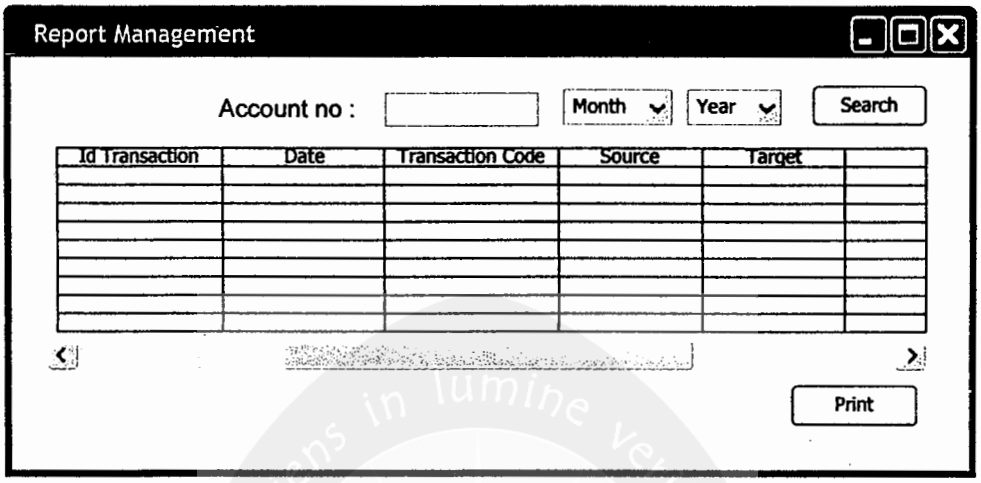

Figure 3.9 Account Report Interface

3.10 Login Interface Description (Web-Based Application)

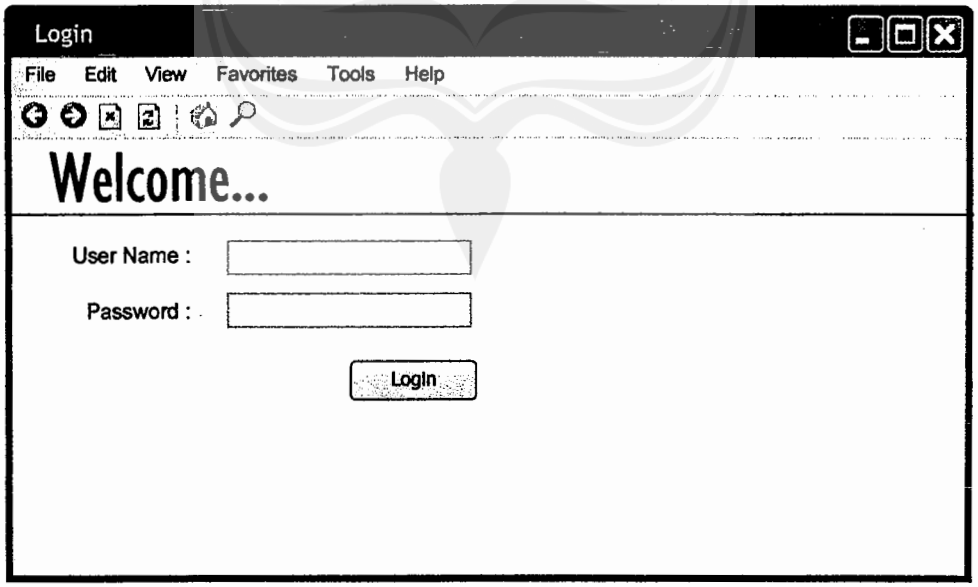

Figure 3.10 Login Interface (Web-Based)

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

This interface is intended to limit unregistered account holder. The account holders as the user will input their username and password to be validated by the system. If inputs are valid, account holder can used the system. Otherwise, account holder cannot use the system and its functionalities.

3.11 Main Menu Interface Description (Web-Based)

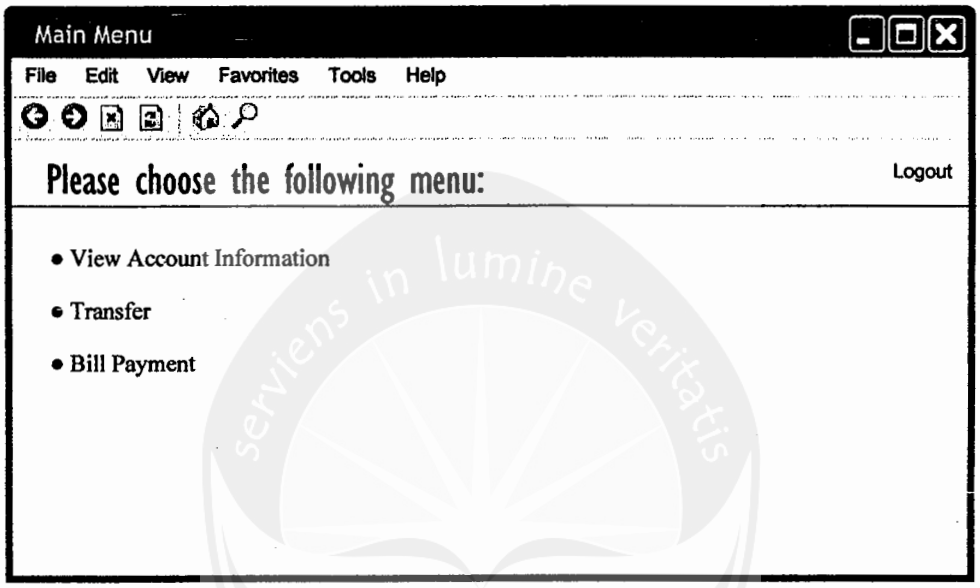

Figure 3.11 Main Menu Interface (Web-Based)

This interface is intended to display the functionalities of the web-based application. This application contains 3 main functions which are:

1. View account information which displays the transaction of account and its balance
2. Transfer which is used for transferring amount of money inter account
3. Bill payment for paying credit card, electricity, and cellular phone bill

User can click the selected label to use the function.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.12 View Account Information Interface Description (Web-Based)

This interface is intended to display the account information along with the transaction during given month by selecting month and year from the drop down lists. The transaction will be displayed on the data grid. User chooses the month and to display the information.

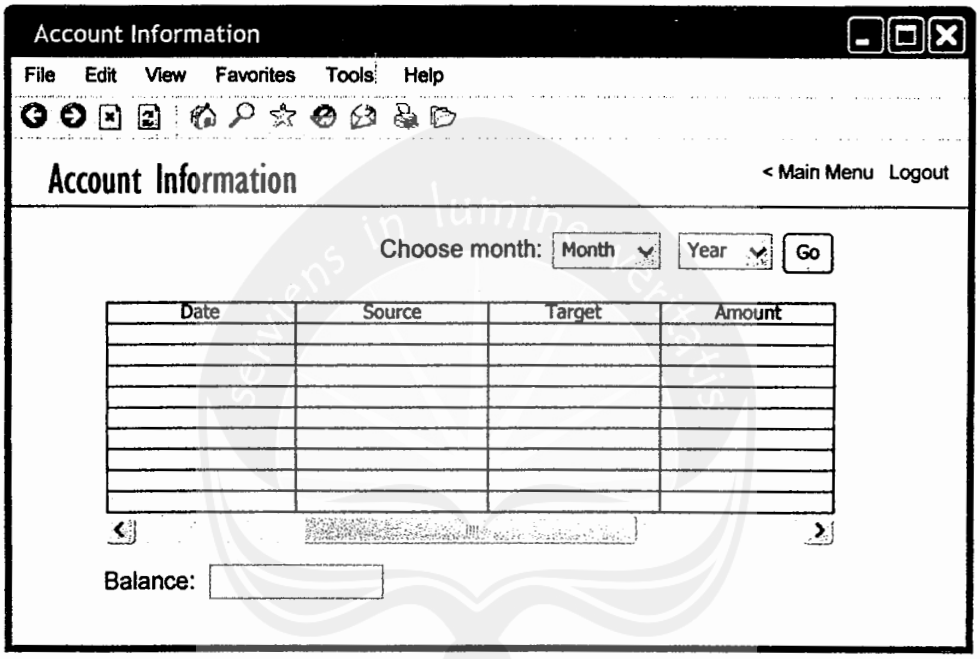

Figure 3.12 View Account Information Interface (Web-Based)

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.13 Transfer Interface Description

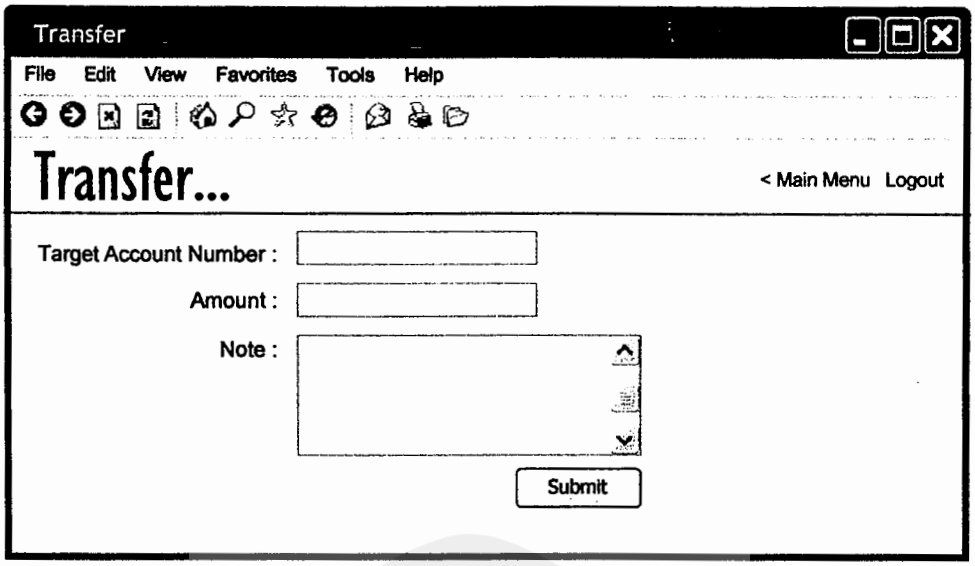

Figure 3.13 Transfer Interface (Web-Based)

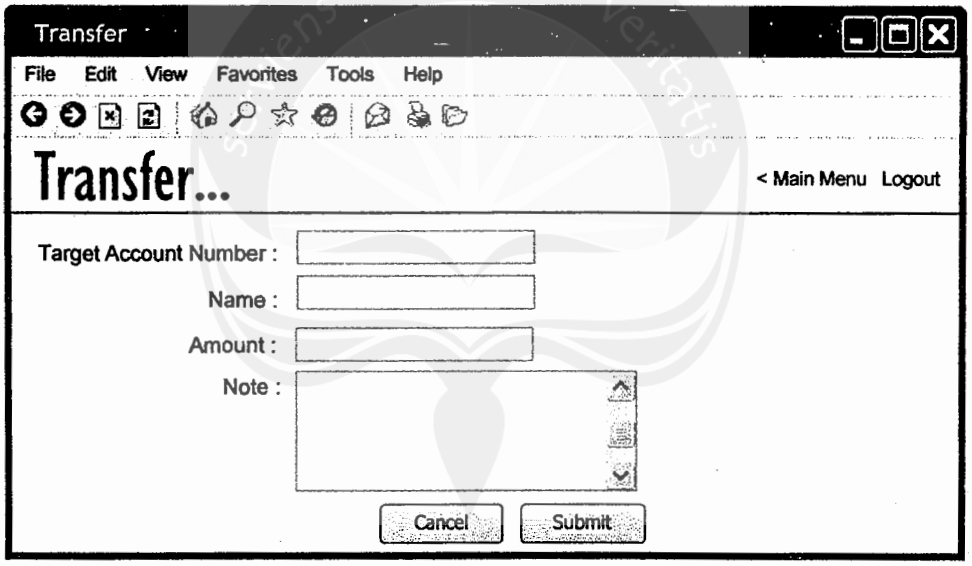

Figure 3.14 Transfer Interface Confirmation (Web-Based)

This interface is intended to do transfer inter-account. This function allows an account holder to transfer some amount of money to another account. Fill in the target account number, amount, and note then click Submit button to go to confirmation page that

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

displays the name of the account target. Click Submit to commence process that account sender balance will be reduced and target account balance will be increased. Click Cancel to back to the previous page.

3.14 Bill Payment Menu Interface Description (Web-Based)

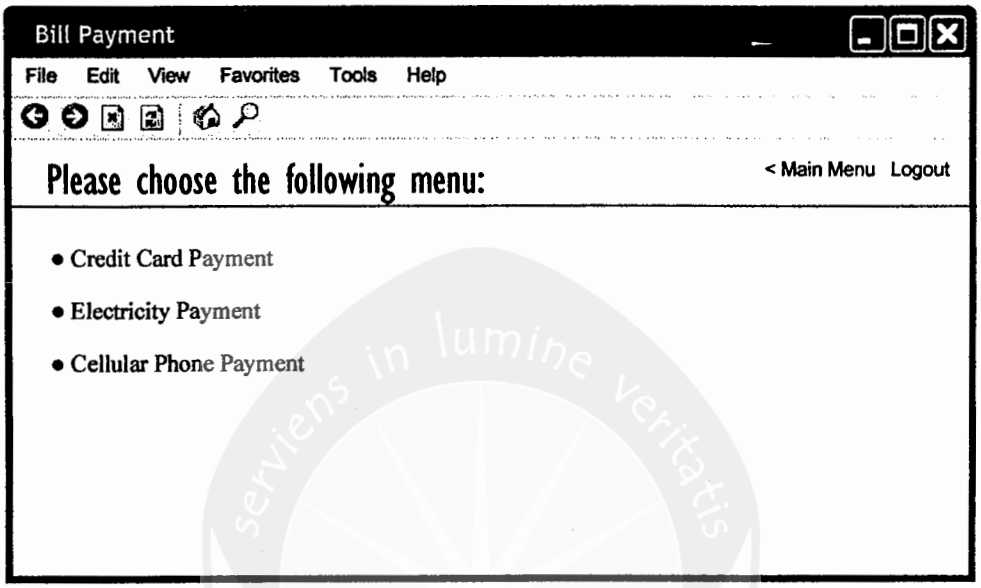

Figure 3.15 Bill Payment Interface (Web-Based)

This interface is intended to display the payment menu. This menu contains 3 functions which are:

1. Paying credit card bill
2. Electricity bill
3. Cellular phone bill

User can click the selected label to use the function. If the user clicks on the credit card payment menu then the credit card payment interface will appear and so does the other.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

3.15 Credit Card Bill Payment Interface Description (Web-Based)

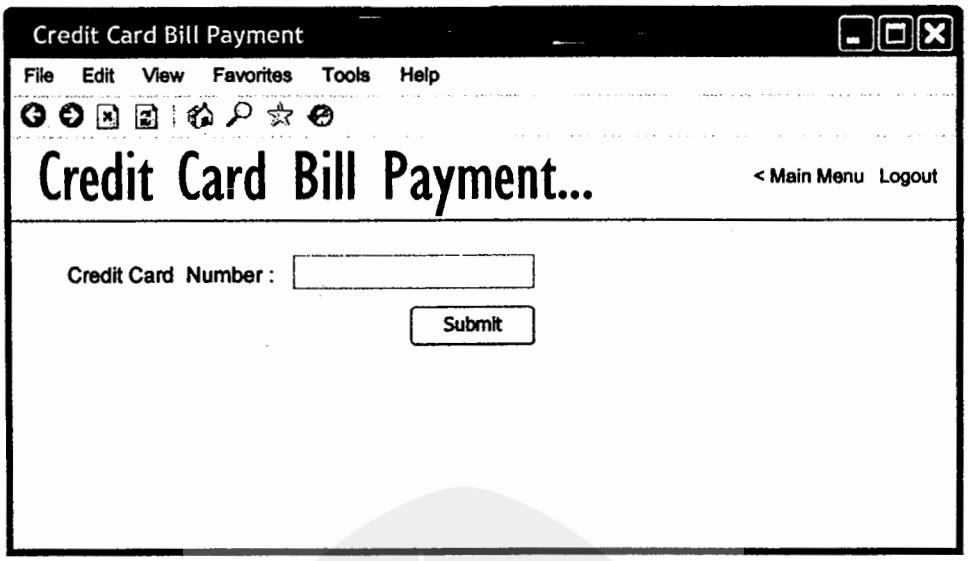

Figure 3.16 Credit Card Bill Payment Interfaces 1 (Web-Based)

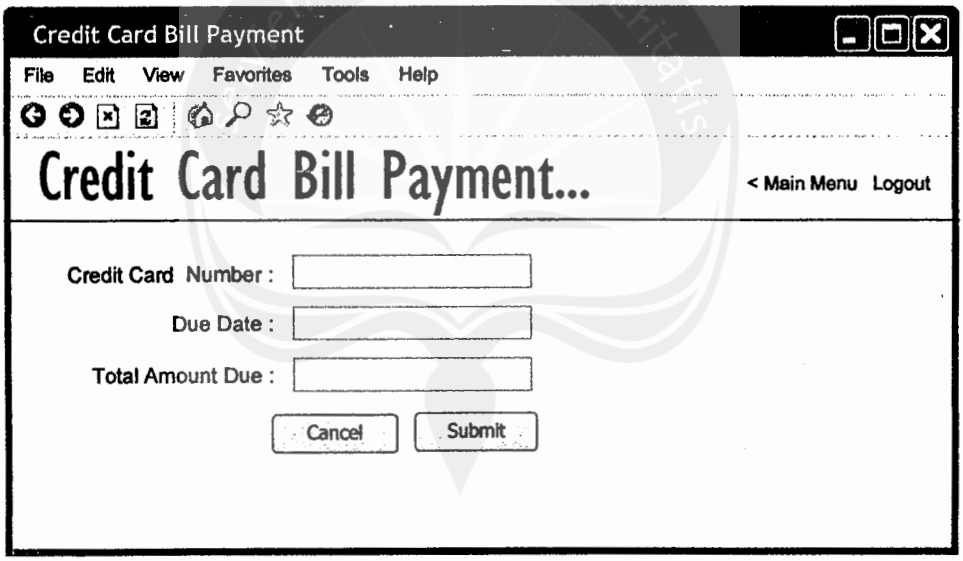

Figure 3.17 Credit Card Bill Payment Interfaces 2 (Web-Based)

These interfaces are intended to do credit card bill payment. The account holder will fill the credit card number in the textbox on the first interface the system will get the bill data (due date and total

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

amount due) to be shown on the second interface. Otherwise, system will display the error message. The account holder can pay the bill by clicking Submit button and system will process the payment or click Cancel to back to previous page.

3.16 Electricity Bill Payment Interface Description (Web-Based)

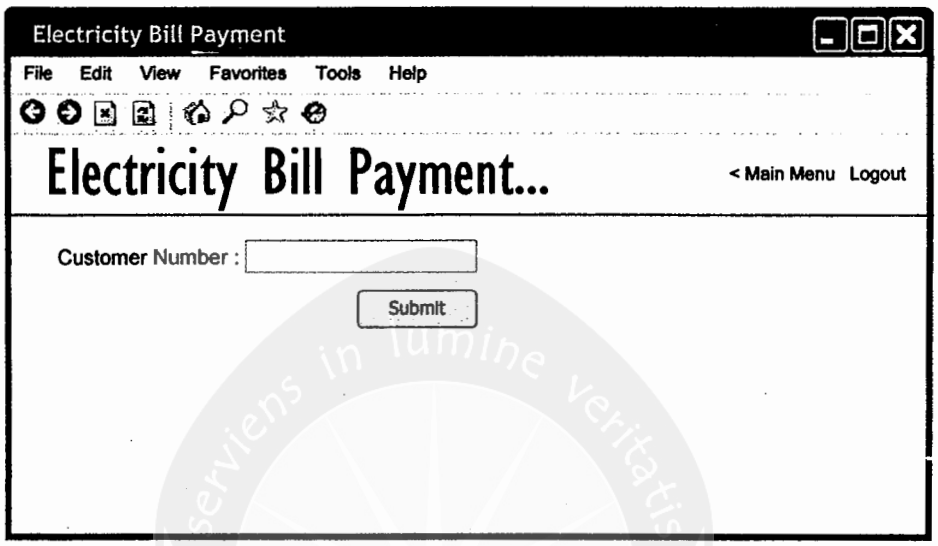

Figure 3.18 Electricity Bill Payment Interfaces 1 (Web-Based)

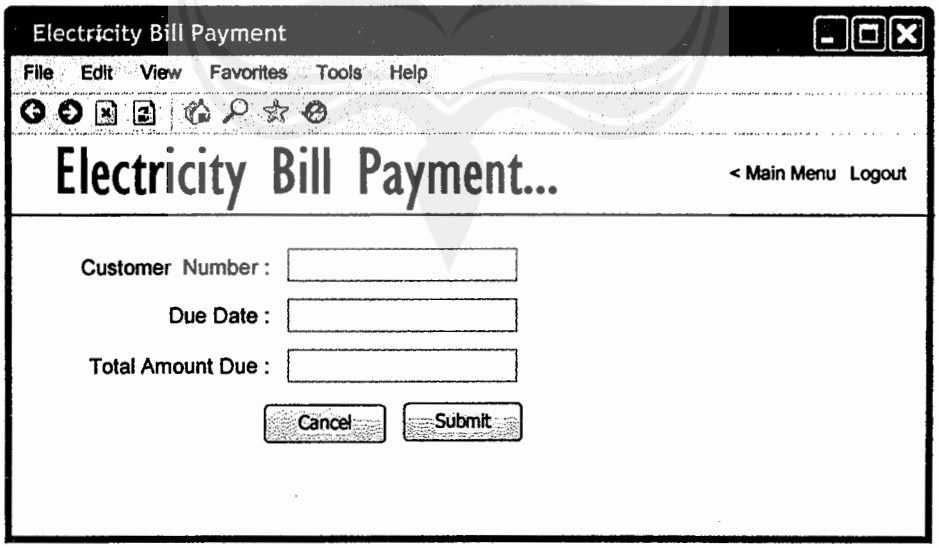

Figure 3.19 Electricity Bill Payment Interfaces 2 (Web-Based)

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

These interfaces are intended to do electricity bill payment. The account holder will fill the customer number in the textbox on the first interface the system will get the bill data (due date and total amount due) to be shown on the second interface. Otherwise, system will display the error message. The account holder can pay the bill by clicking Submit button and system will process the payment or click Cancel to back to previous page.

3.17 Cellular Phone Bill Payment Interface Description (Web-Based)

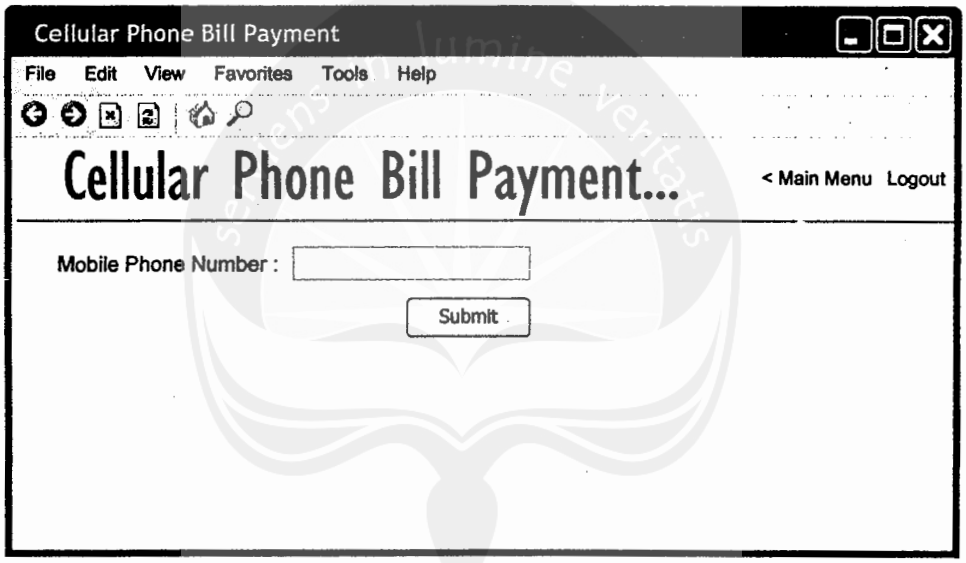

Figure 3.20 Cellular Phone Bill Payment Interfaces 1 (Web-Based)

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

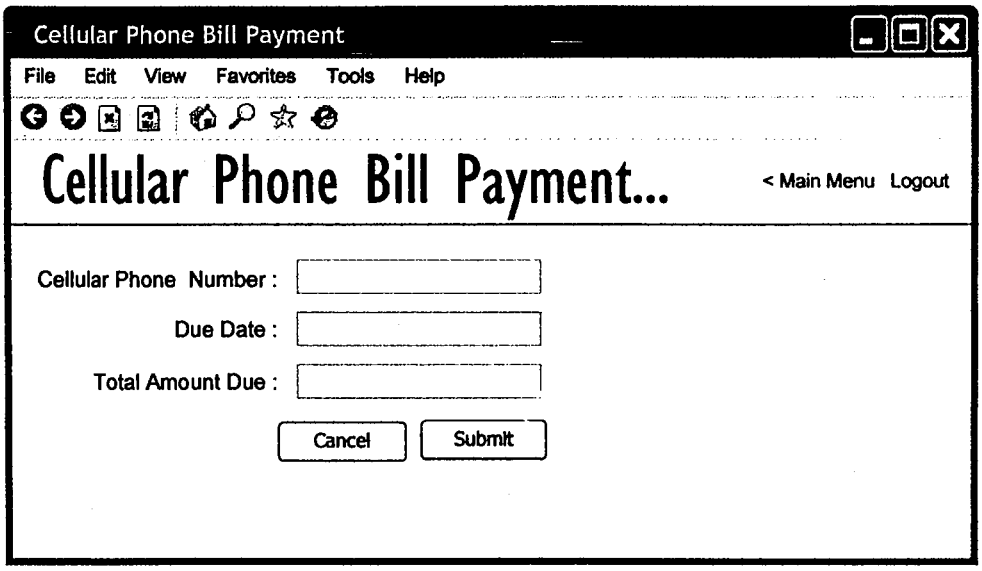

Figure 3.21 Cellular Phone Bill Payment Interfaces 2 (Web-Based)

These interfaces are intended to do cellular bill payment. The account holder will fill the cellular phone number in the textbox on the first interface the system will get the bill data (due date and total amount due) to be shown on the second interface. Otherwise, system will display the error message. The account holder can pay the bill by clicking Submit button and system will process the payment or click Cancel to back to previous page.

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

4 Physical Database Design

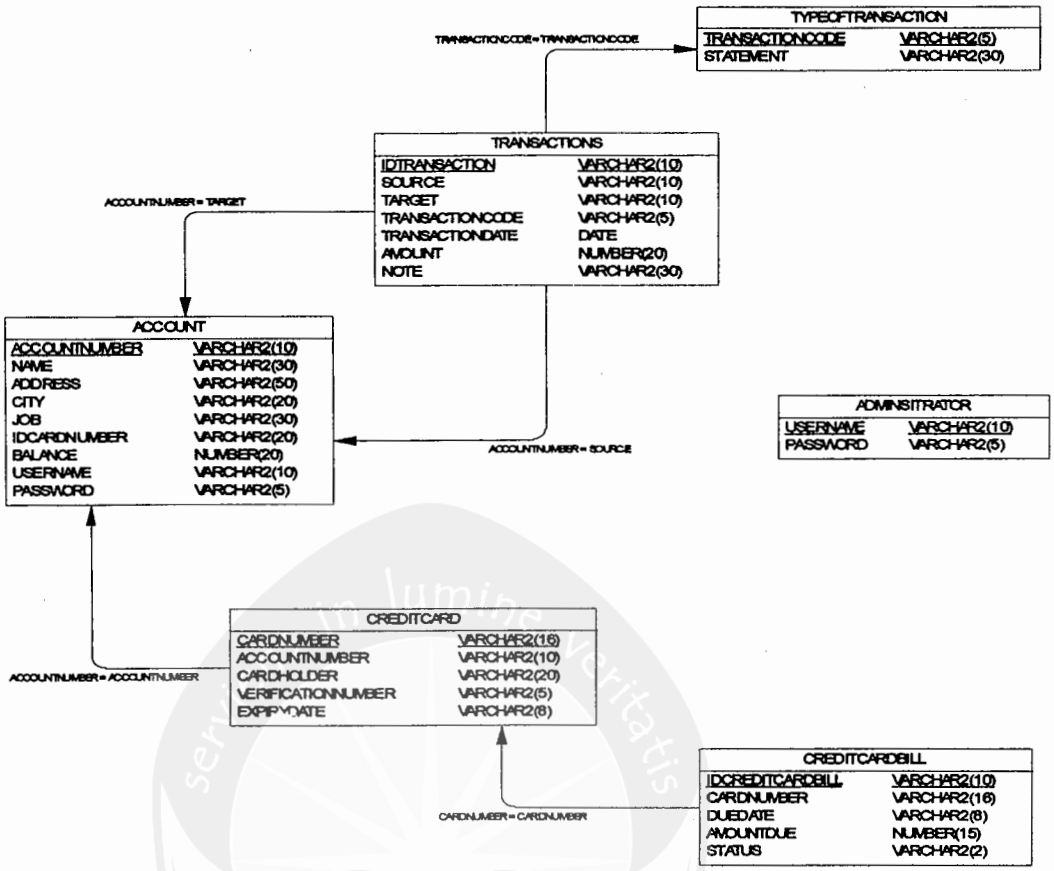

Figure 4.1 Physical Database Design

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

4.1 Administrator Description

Username	Varchar2(10)	Administrator username, primary key
Password	Varchar2(10)	Administrator password

4.2 Account Description

AccountNumber	Varchar2(10)	Account number, primary key
Name	Varchar2(30)	Account holder name
Address	Varchar2(50)	Account holder address
City	Varchar2(20)	Account holder city of live
Job	Varchar2(30)	Account holder job
IdCardNumber	Varchar2(20)	Account holder id card number
Balance	Number(20)	Account balance
Username	Varchar2(10)	Account holder username
Password	Varchar2(5)	Account holder password

4.3 TypeOfTransaction Description

TransactionCode	Varchar2(5)	Code of transaction, primary key
Statement	Varchar2(30)	Transaction type statement

4.4 Transactions Description

IdTransaction	Varchar2(10)	Id transaction, primary key
TransactionDate	Date	Transaction date
TransactionCode	Varchar2(5)	Transaction code
Source	Varchar2(10)	Transaction source (sender)
Target	Varchar2(10)	Transaction target (receiver)
Amount	Number(20)	Amount of transaction
Note	Varchar2(30)	Transaction note

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).

4.5 CreditCard Description

CardNumber	Varchar2(16)	Credit card number, primary key
AccountNumber	Varchar2(10)	Account as a source of credit card
CardHolder	Varchar2(20)	Credit card holder name
VerificationNumber	Varchar2(10)	Credit card verification number
ExpiryDate	Varchar2(8)	Credit card expiry date

4.6 CreditCardBill Description

IdCreditCardBill	Varchar2(10)	Id credit card bill, primary key
CreditCardNumber	Varchar2(16)	Credit card number
DueDate	Varchar2(8)	Credit card bill due date
AmountDue	Number(15)	Credit card bill amount due
Status	Varchar2(2)	Credit card bill status

The owner of this document and the information inside is Informatics Engineering of Atma Jaya Yogyakarta University (UAJY).