TEENOVATORS

Internet Banking System
Software Requirements Specification

College
IT-BHU, Varanasi

Team Members
1) Ajay Kr. Gautam
2) Gaurav Gupta
3) Karan Jain
4) Rohit Jaiswal

Faculty Guide
Prof. A.K.Agarwal

Contents

Software Requirements Specification

1. **Introduction:** Online Banking or Internet Banking allows customers to conduct financial transactions on a secure website operated by their retail or virtual bank, credit union or building society.

   i. **Purpose:** With the advent of technology and introduction of computerization in almost every aspect of life, the Banks do aim at providing ease to one's life. Hence, a demand for an online comprehensive solution to manage Internet Banking exists. It shall be a system accessible to all its customers through a valid userID and password allotted to them. They can avail facilities such as viewing their balance & account statements; transferring funds from one account to another; request for cheque book, stop payments of cheques, bill payments and many more.

   ii. **Scope:**

   a. Includes a Virtual Keyboard option for prevention of Keylogger attacks.
   b. Provision of facility of blocking user's account on 3 successive invalid attempts to login.
   c. View/Select/Map an account to your userID and view its corresponding information.
   d. View/Edit your profile details.
   e. Change password and other settings of the account.
   f. Map a loan account to your userID.
   g. Pay/Print/Save bills & corresponding receipts and others services such as mobile recharge, card to card transfer, etc.
   h. Issue Standing Instructions.
   i. Request Cheque Book/Draft.

j. View status of Cheques/Drafts; Stop payment of Cheques/Drafts.
k. View Transaction Reports; View Account statements for any desired period in form of statistics & graphs.
l. Enable high security option where SMS is sent to mobile for confirmation.
m. Transfer funds from one account to another and also to 3rd party accounts.
n. A very useful search functionality.
o. Providing letter of Credit/Guarantee, etc.
p. Blocking & Unblocking of account (facility of admin)
q. Proper logs are created and accessible by Administrator.
r. Customizable themes & templates.
s. A separate customizable JAVA based client for Administrator.

iii. Definitions:

a. ACID Properties:

- **A**tomicity: Either all operations of a transaction are reflected properly in a database or none are.
- **C**onsistency: Execution of a transaction in isolation i.e. with no other transaction executing concurrently preserves the consistency of the database.
- **I**solation: Even though multiple transactions may execute concurrently, the system guarantees that for every pair of transaction T_i and T_j , it appears to T_i that either T_j finished execution before T_i started or T_j started execution after T_i finished. Thus, each transaction is unaware of other transactions executing concurrently in the system.
- **D**urability: After a transaction completes successfully, the changes it has made to the database persist, even if there are system failures.

b. E-R Model: Based on the perception of a real world that consists of basic objects called Entities and Relations between these. The

set of entities of the same type are called Entity Set and relations of same type are called Relationship Set. E-R model places a constraint called mapping cardinalities which expresses the number of entities to which another entity can be associated.

c. **Remittance:** The process of sending money to remove an obligation. This is most often done through an electronic network, wire transfer or mail. The term also refers to the amount of money being sent to remove the obligation.

d. **Standing Instructions:** Alerts that a customer can give to the bank to permit it to debit a fixed amount of money from his account for a particular reason. The bank charges some commission for this purpose.

e. **Keylogger Attacks:** This is a type of attack where a malicious user installs software on the system which logs/records all the keys pressed through the keyboard and can record relevant information.

iv. References:

- Online References:
  - http://www.chase.com/
  - http://www.anz.com/
  - http://www.standardchartered.com.sg/
  - https://www.bankofamerica.com/
  - http://www.citibank.com/us/index.htm
  - http://www.usbank.com/
  - http://www.marketwatch.com/
  - http://www.creditcards.com/

- http://www.stumbleupon.com/demo/?review=1#url=http://www.favoritesworld.com/
- http://www.britishairways.com/travel/informationhubpage/public/en_us?source=TOP_information
- http://www.worldbank.org/

➢ IEEE SRS Format

➢ Project Scenario and specification given by IBM

v. Technologies:

i. Eclipse IDE

ii. WASCE-WebSphere Application Server Community Edition

iii. DB2 Express

iv. AJAX

v. XML

vi. RSA – Rational Software Architect

vi. Overview: The remaining portion of SRS is divided into two sections:

➢ *Overall Description* will describe the major components of the system, interconnection and external interfaces.

➢ *Specific Requirements* will describe the function of actors, their role in the system and constraints.

2. Overall Description:

I. **Product Perspective:** There are 3 aspects of perspection to this project:

i. Customer View

ii. Industrialists'/Organizations'/Entrepreneurs' View

iii. Administrator View

II. **Product Function:** The various sections of the product are described below with their function:

i. **Login Module:**

This portion is meant for authenticating a valid user. Since this is the only entry point into the system, proper security measures like encryption of data, virtual keyboards, prevention of script execution by image verification, blocking account on 3 successive invalid login attempts have been implemented.

ii. **MyProfile Module:**

This section shall primarily focus on my personal information. It has the options of View/Select/Map accounts to the userID, View/Edit account information, changing Account Settings such as the security level, password, etc. One can also map his loan account to the userID.

iii. **Services Modules:**

It includes features such as bill payment facilities, mobile phone recharge, sending money to some address, card to card transfer, etc.

iv. **Standing Instructions:**

Under this section, we provide customers a facility to issue an instruction to the bank to debit/transfer money from his/her account and credit it to someone else's account on a particular date repeatedly after a fixed interval of time.

v. **Cheques/Drafts:**

Under this section we deal with viewing the status of Cheques/Drafts. We can also request to stop the payment of cheques/drafts. Besides, there are also options for requesting to issue a Cheque Book.

vi. **Letter of Credit/Guarantee:**

This is a facility of most use to the entrepreneurs/organizations. This is meant in order to transfer large amounts of money which cannot be taken in hand for business purposes.

vii. **Funds Transfer:**

Under this one can transfer money between 2 accounts mapped to a single userID as well as transfer money to a 3rd party as well.

III. **Data Flow Diagram:** The following diagram indicates the flow of information data through our system.

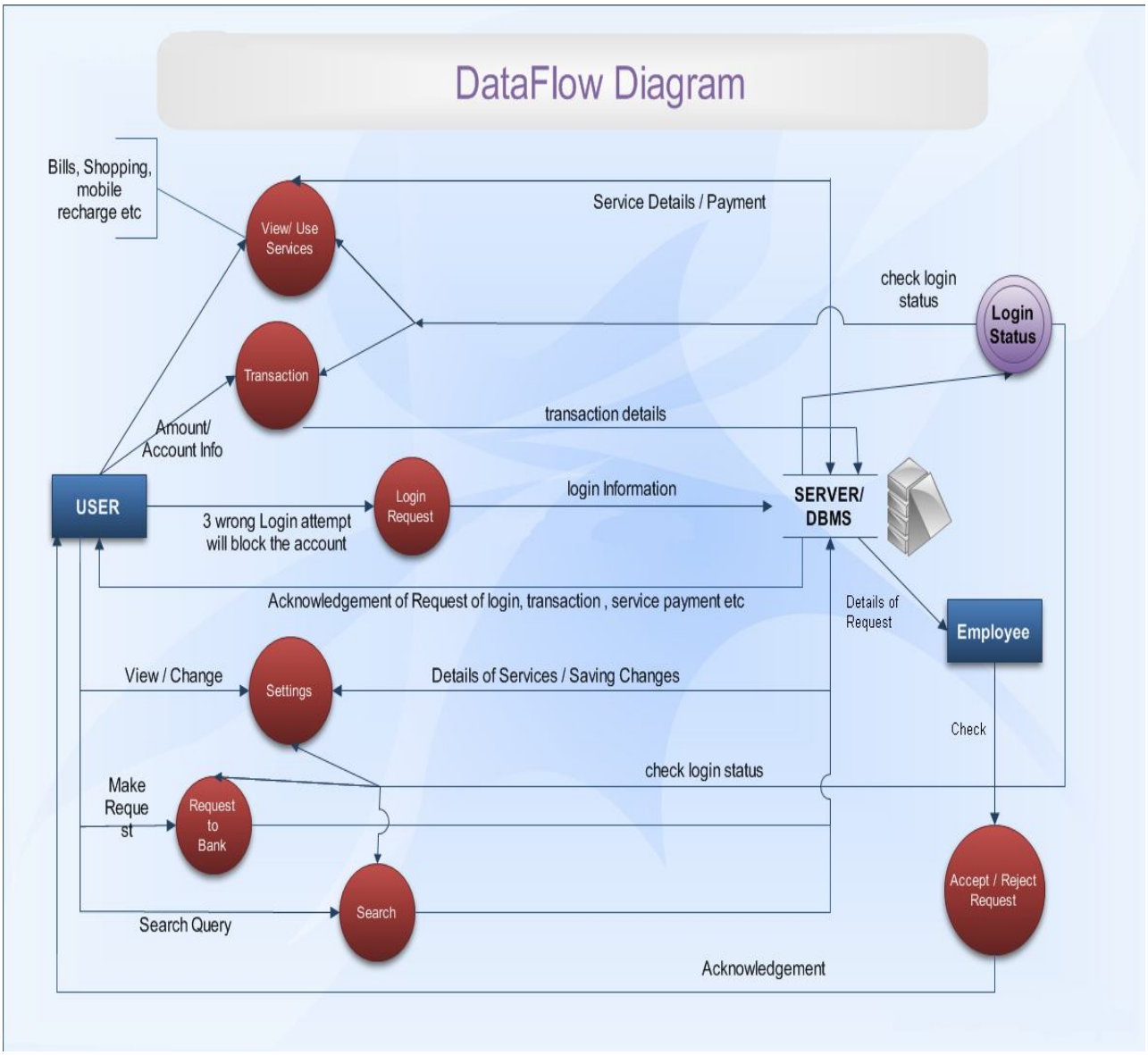


Figure 1 : Data Flow Diagram for Internet Banking System

IV. Class Diagram:

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes and the relationships among the classes.

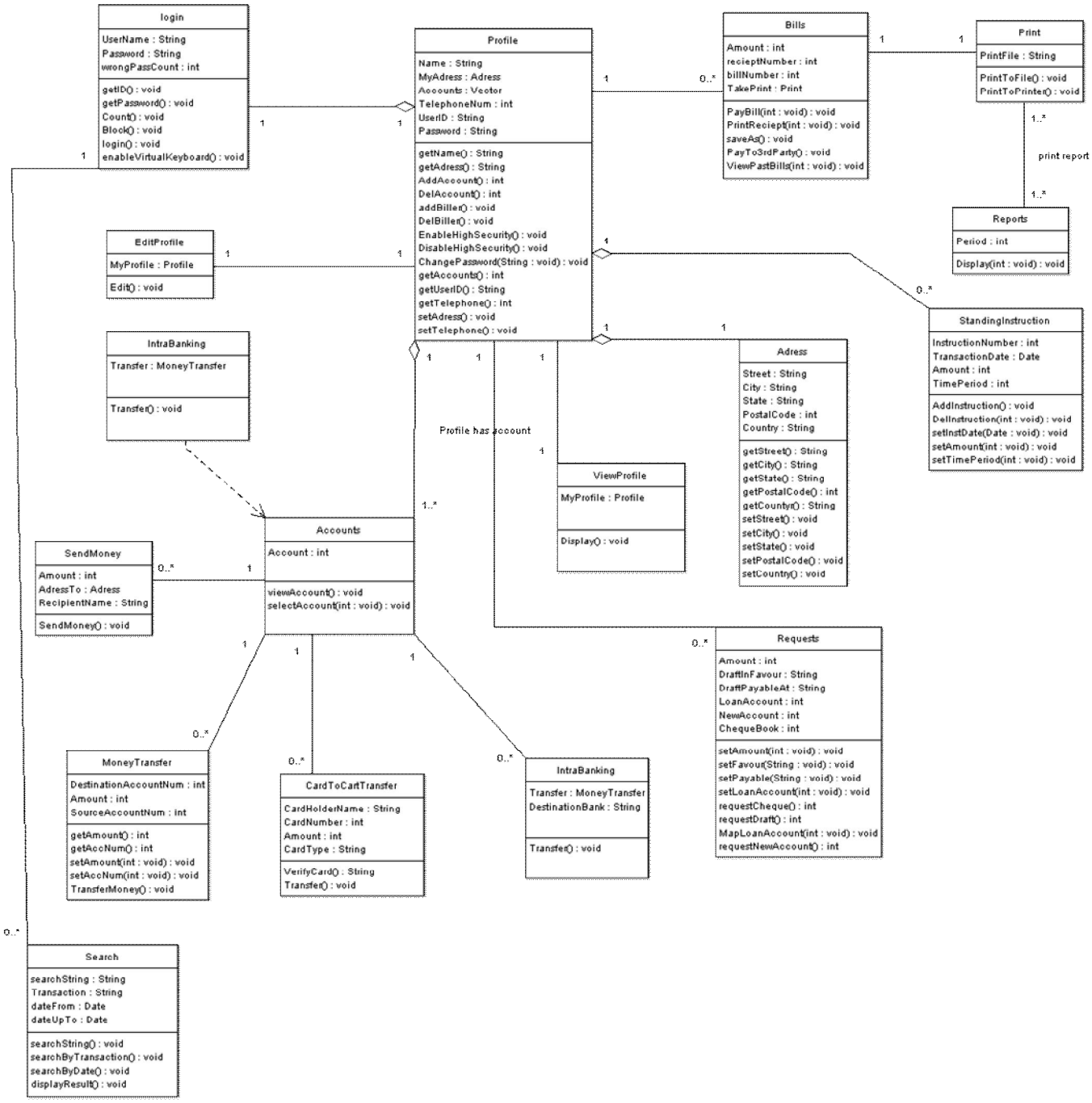
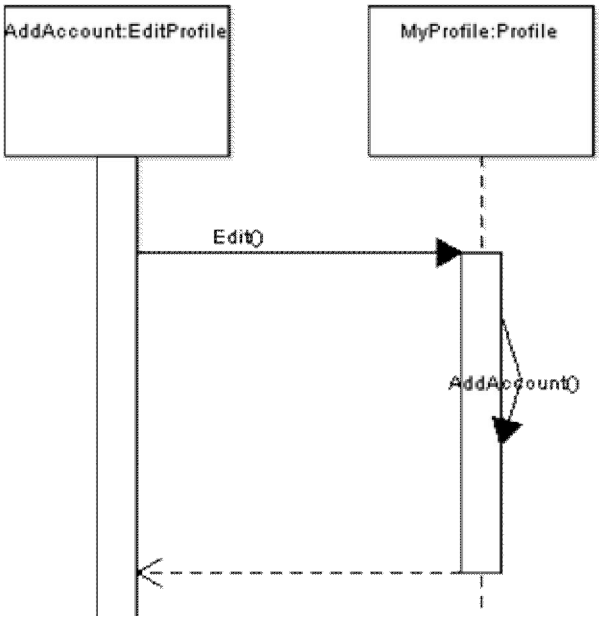


Figure 2: Class Diagram for Internet Banking System

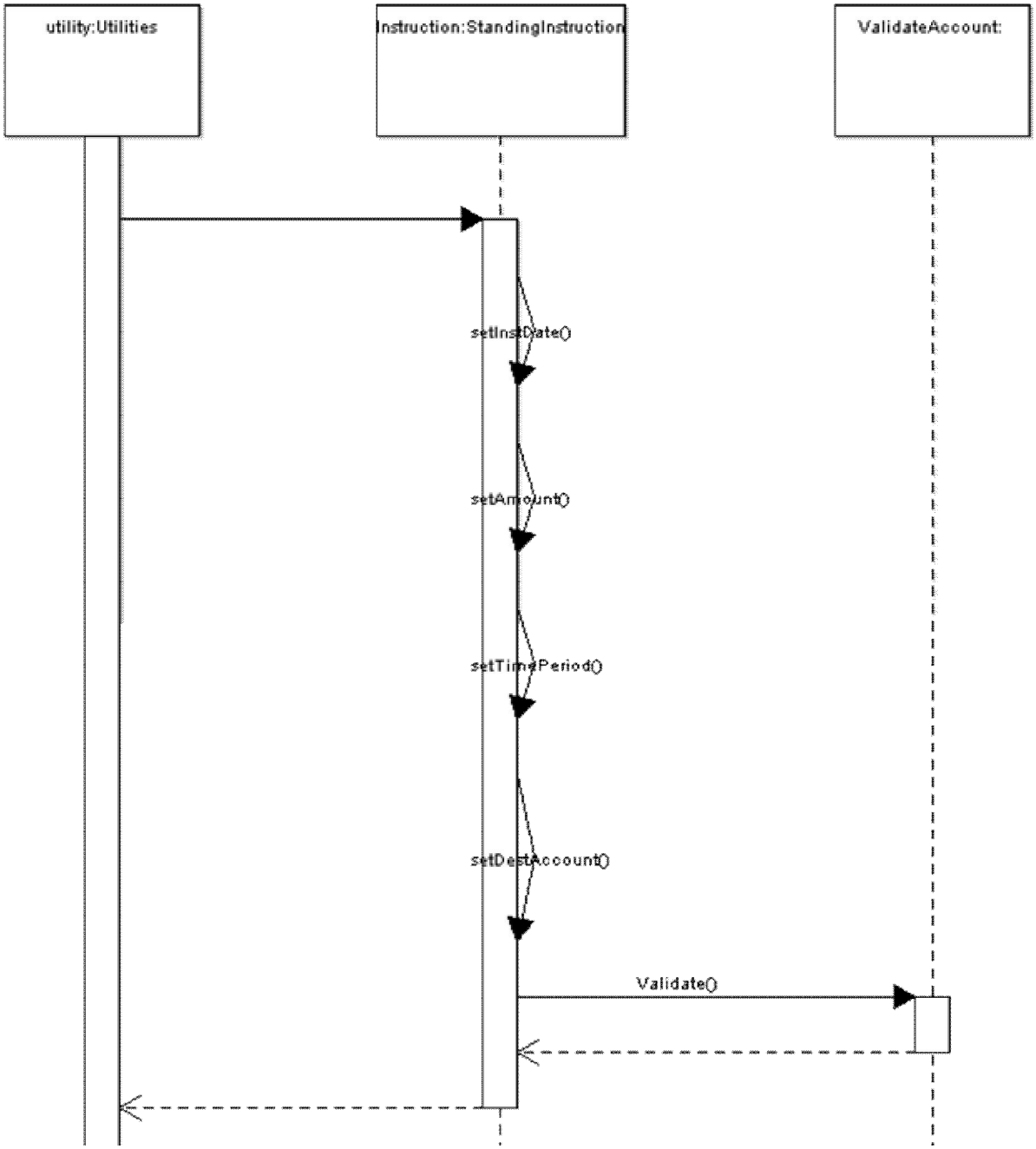
V. Sequence Diagrams:

A sequence diagram (also called interaction diagram) is a UML construct of a Message Sequence Chart. It shows how processes operate one with another and in what order.

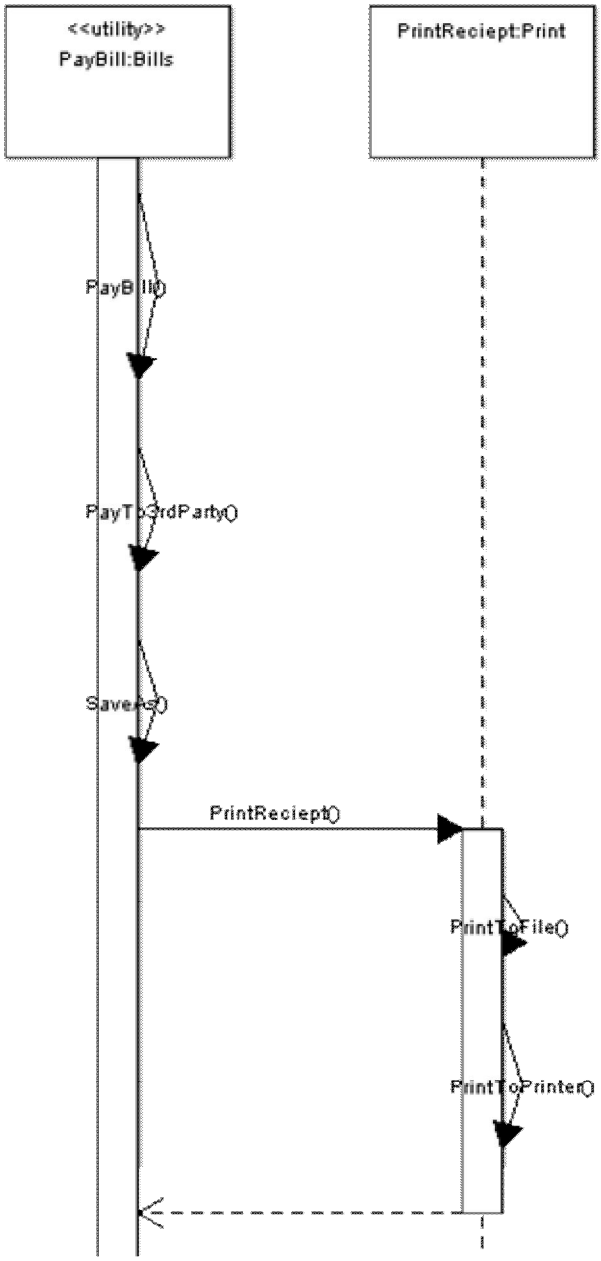
(i) ADD ACCOUNT TO USERID & EDIT PROFILE



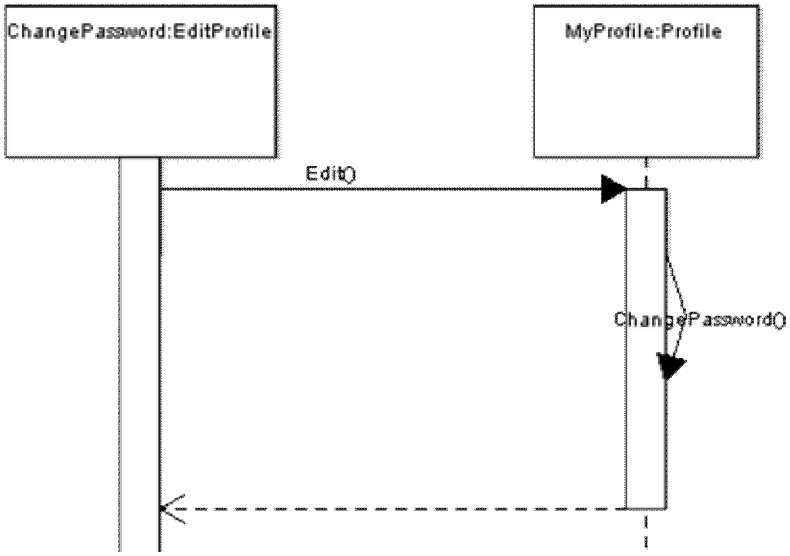
(ii) ISSUE STANDING INSTRUCTION TO THE SYSTEM



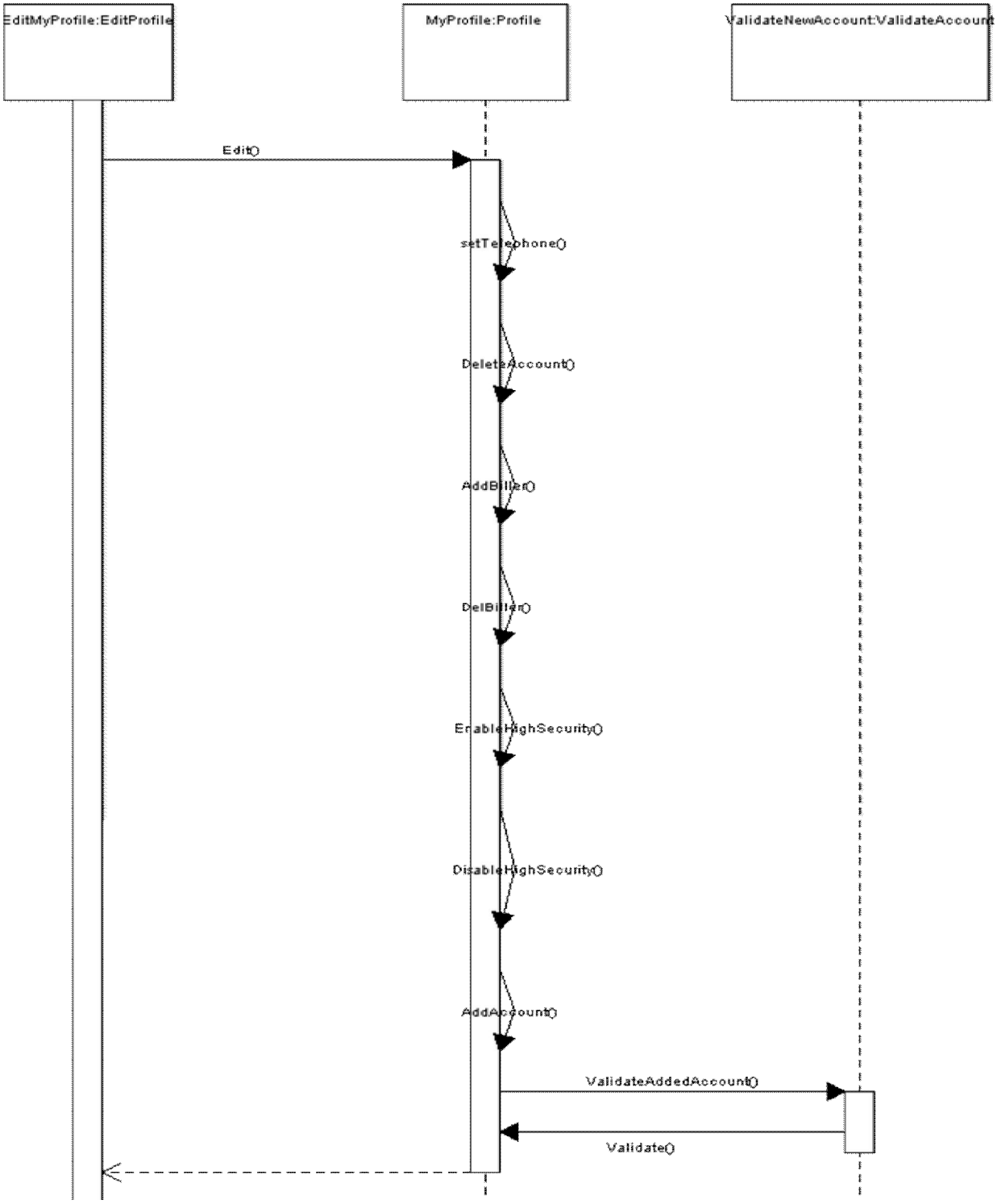
(iii) PAYMENT OF BILLS



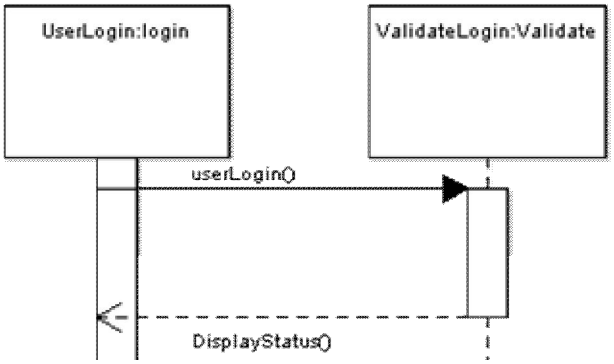
(iv) CHANGE PASSWORD



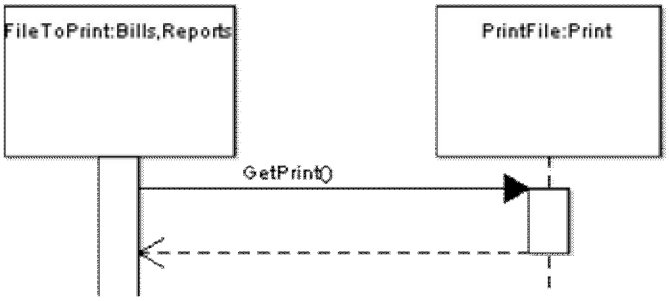
(v) EDIT USER PROFILE & ACCOUNT SETTINGS



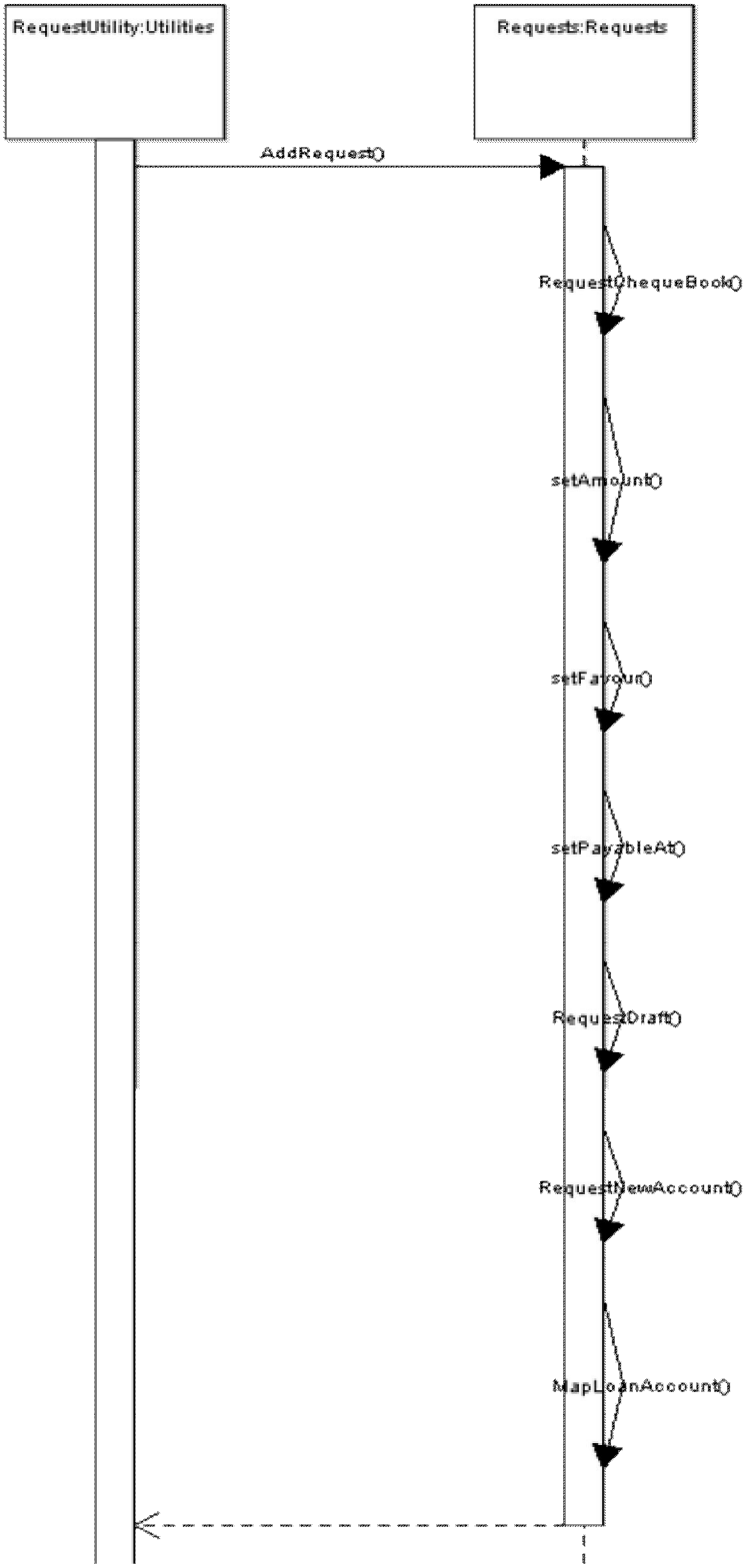
(vi) CLIENT LOGIN TO SYSTEM



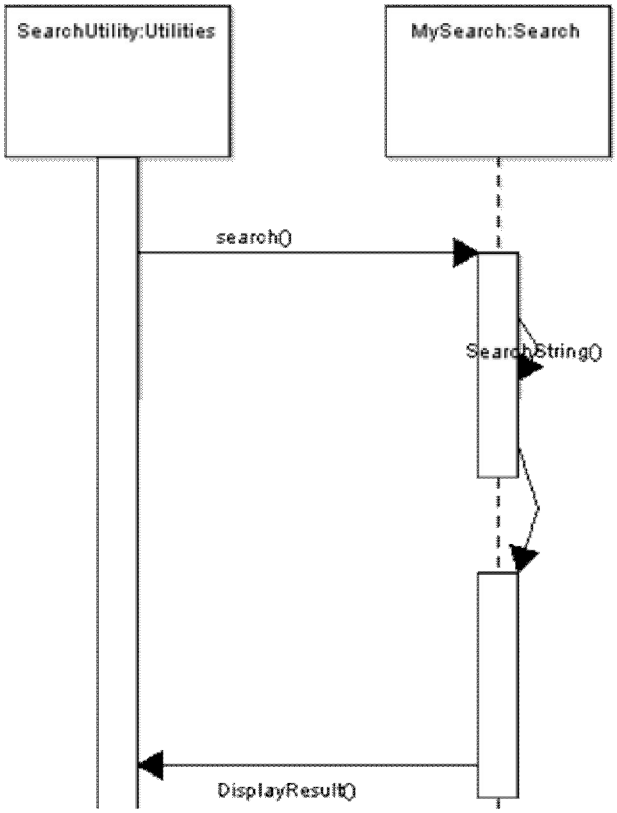
(vii) PRINT REPORTS, RECEIPTS & OTHER ITEMS



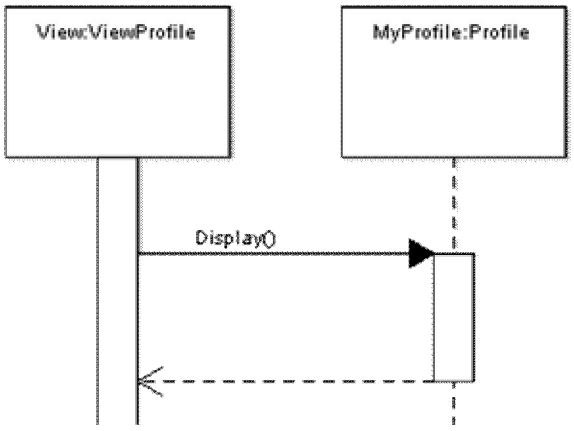
(viii) REQUEST CHEQUEBOOK/DRAFTS



(ix) SEARCH UTILITY



(x) VIEW USER PROFILE

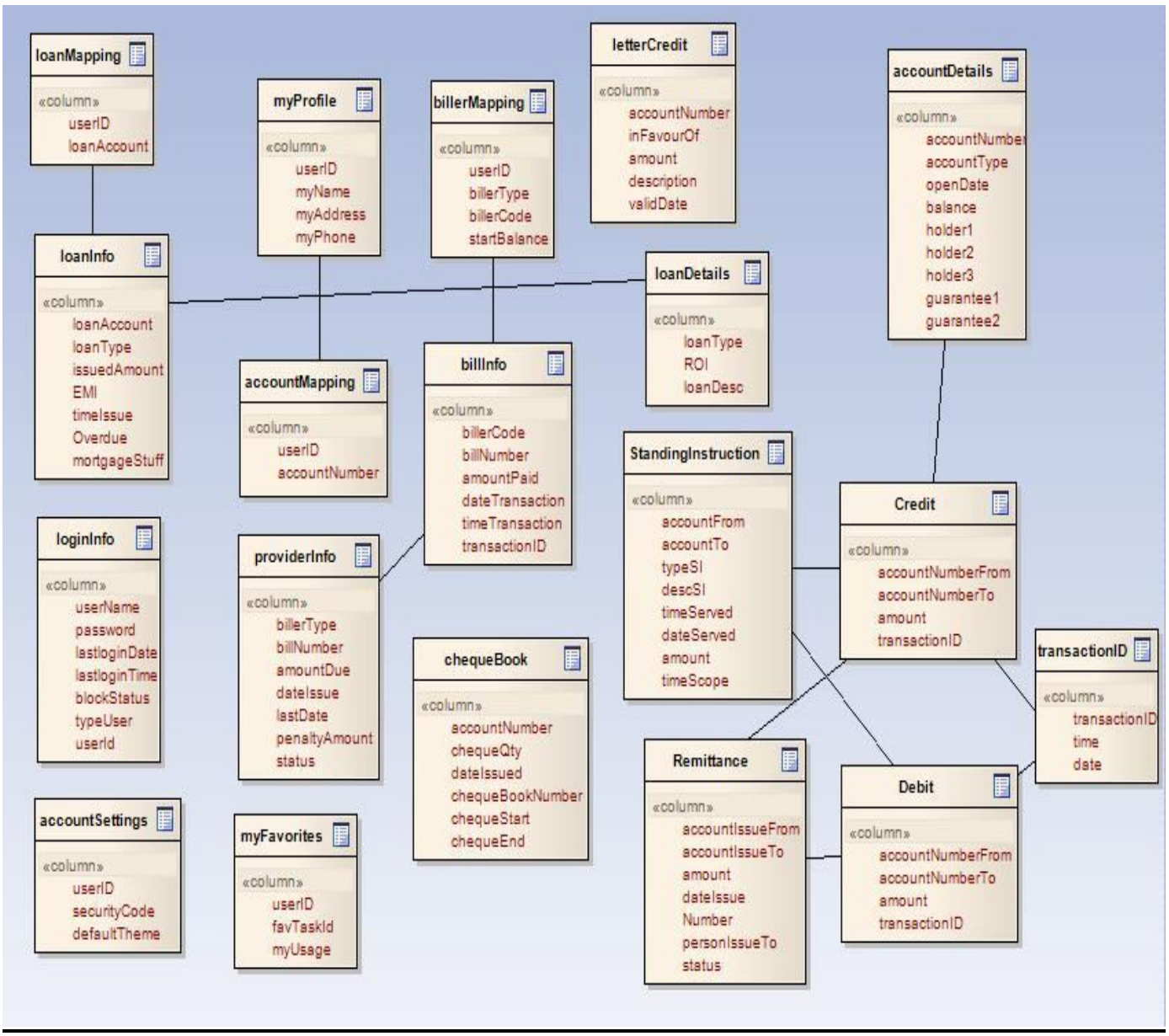


VI. **Database Design:** This essentially consists of two sections:

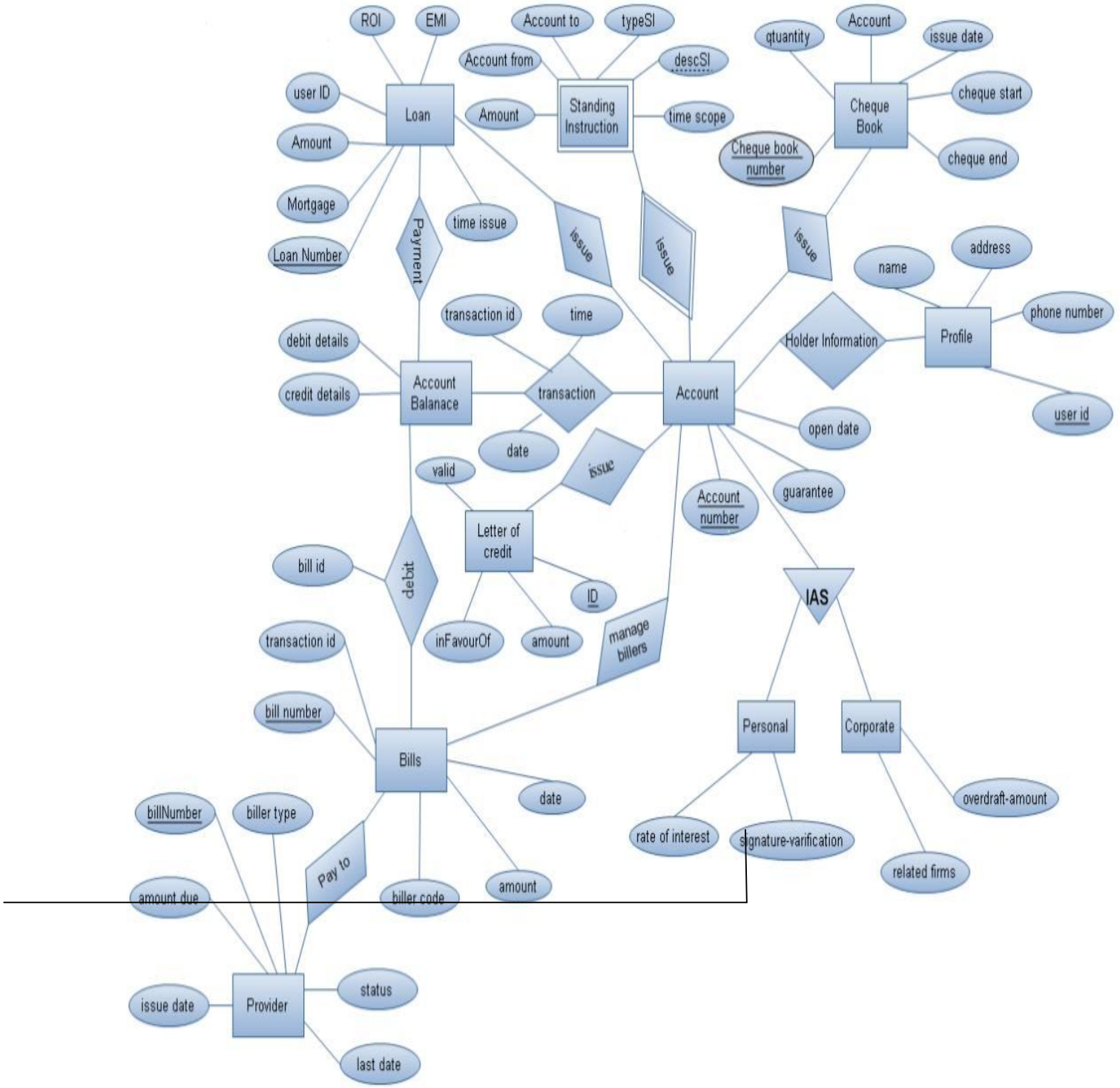
i. Schema Design

ii. E-R Diagram

Schema Design:



E-R Diagram: The Entity-Relationship model for various relations and entity sets is illustrated through the following diagram. The standard notations are followed in the diagram.



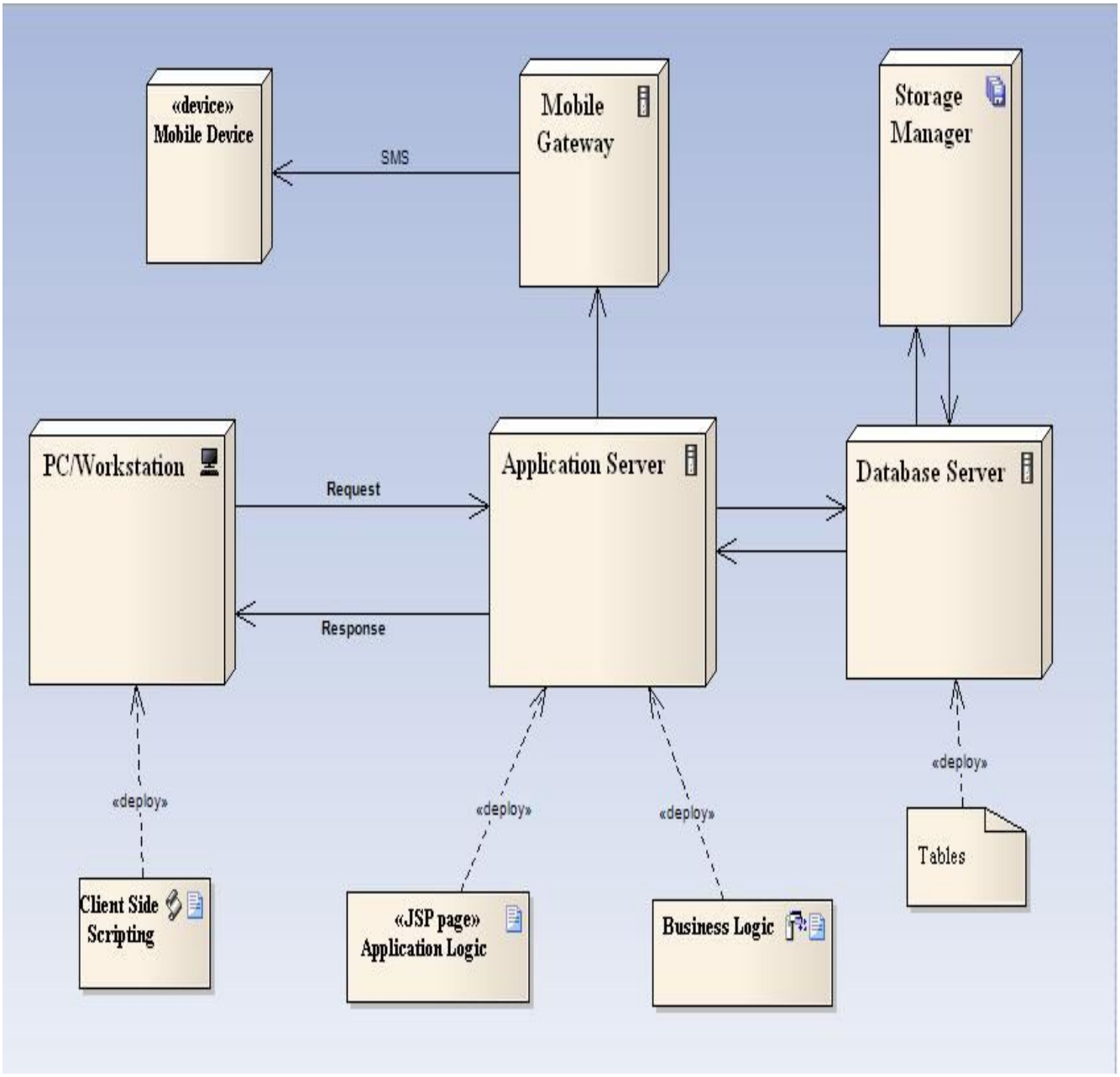
VII. Architecture Diagrams

Two types of architecture diagrams are proposed:

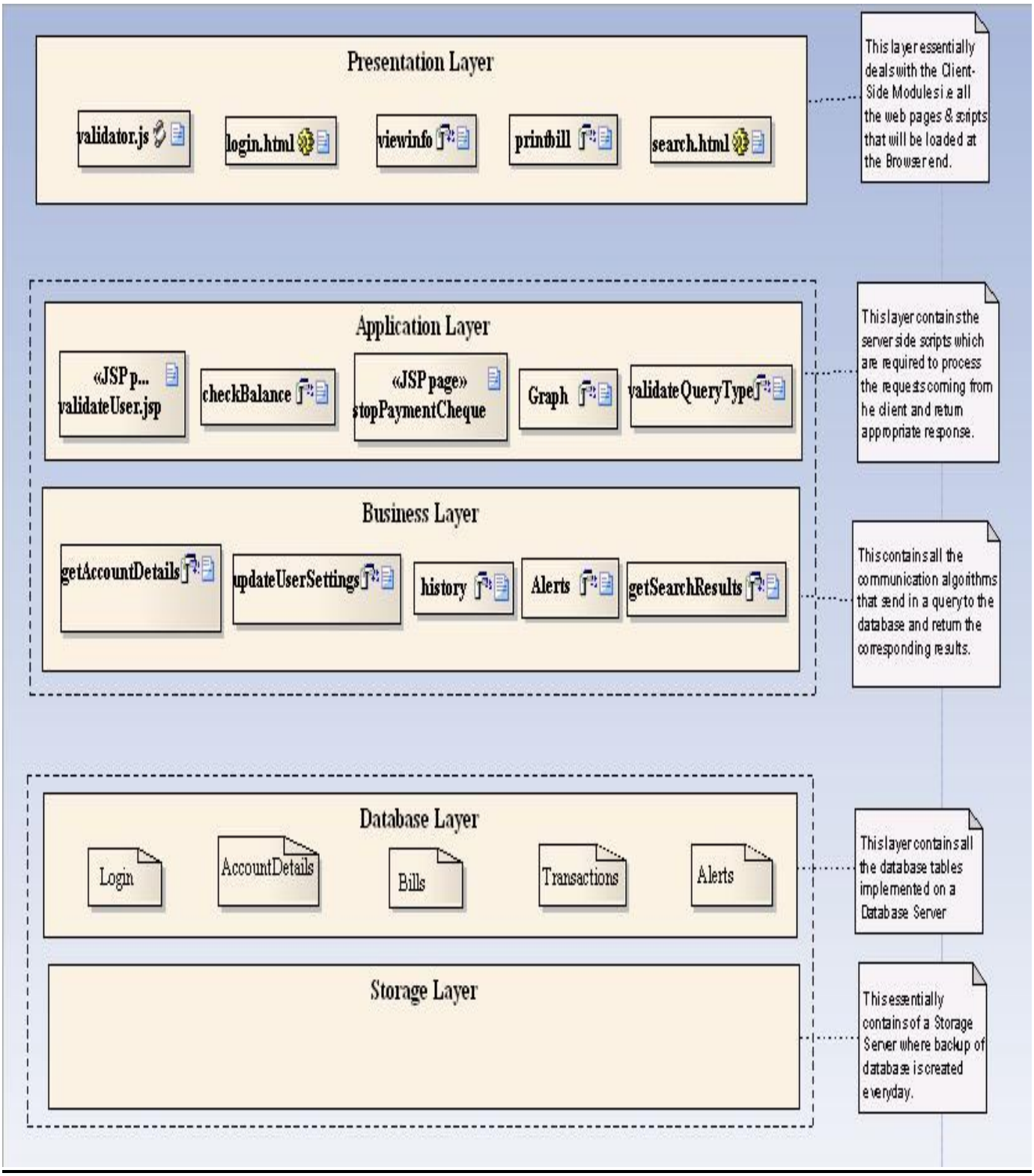
i. Deployment Model

ii. Multi-Layer Architecture

Deployment Model:



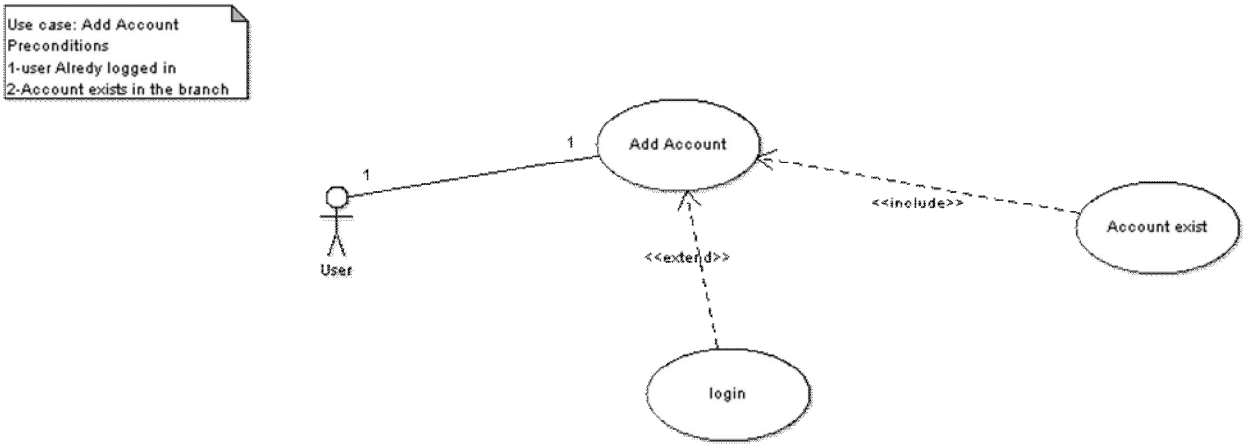
Multi-Layer Architecture / 3-Tier Client Server Structure:



Specific Requirements:

I. Use-case Reports:

a) Name of Use Case : **ADD ACCOUNT**



Description: A customer can map his pre-existing accounts to a single userID.

Pre-Conditions:

i. User is already logged in.

ii. Account that is being added already exists.

Flow of Events:

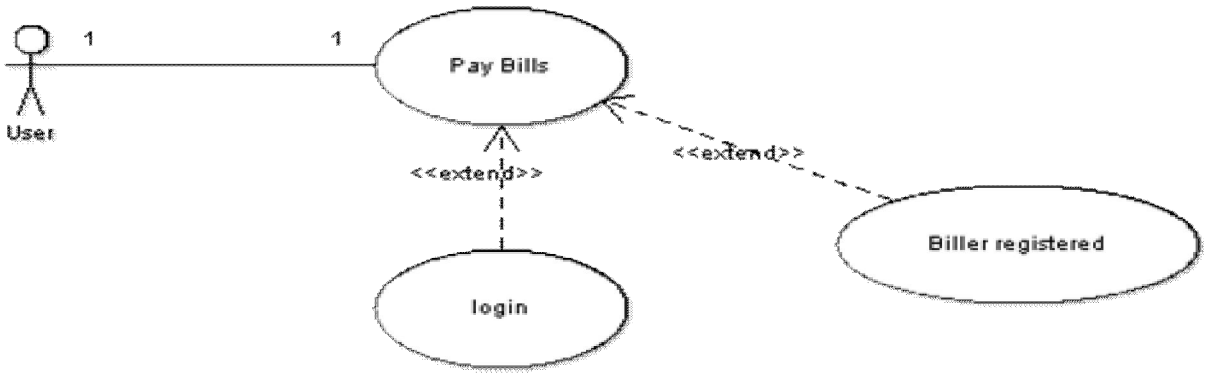
i. User enters the account number of the corresponding account.

ii. System searches for the account number.

iii. If it exists, it maps it to the userID.

iv. If it does not exist, error is returned.

b) Name of Use Case: **BILL PAYMENT**



Description: One can pay his various types of bills through this module.

Pre-Conditions:

i. User is logged in

ii. A biller corresponding to that bill type has already been added to his account.

Flow of Events:

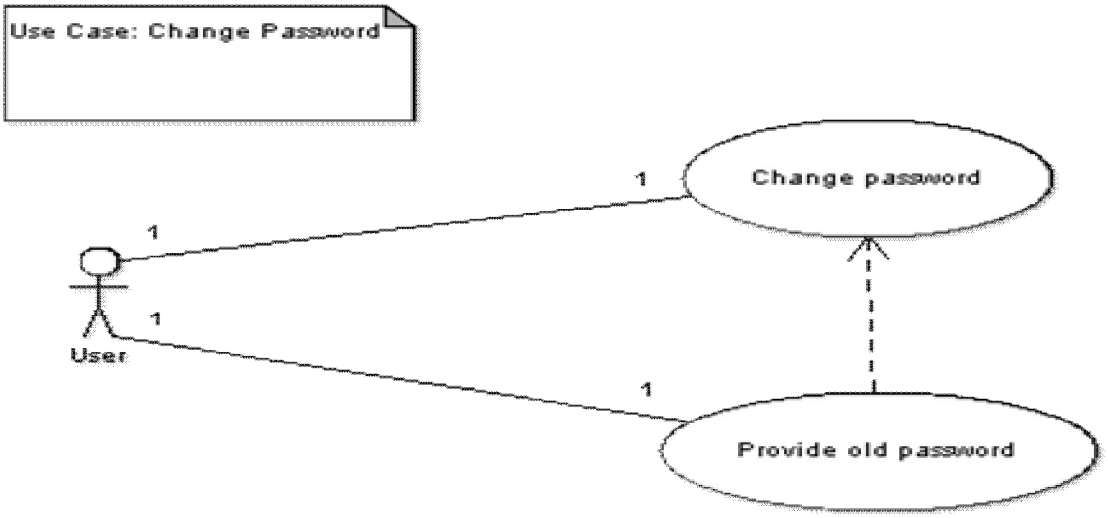
i. User opens the biller to pay the bill and enters his details.

ii. On paying the bill, the system checks whether his account has sufficient balance.

iii. If yes, bill is paid; receipt is issued and balance id deducted along with commission charges (optional).

iv. If no, appropriate error message is displayed.

c) Name of Use Case: **CHANGE PASSWORD**



Description: Through this a user can change his/her password.

Pre-Conditions:

i. User is already logged in

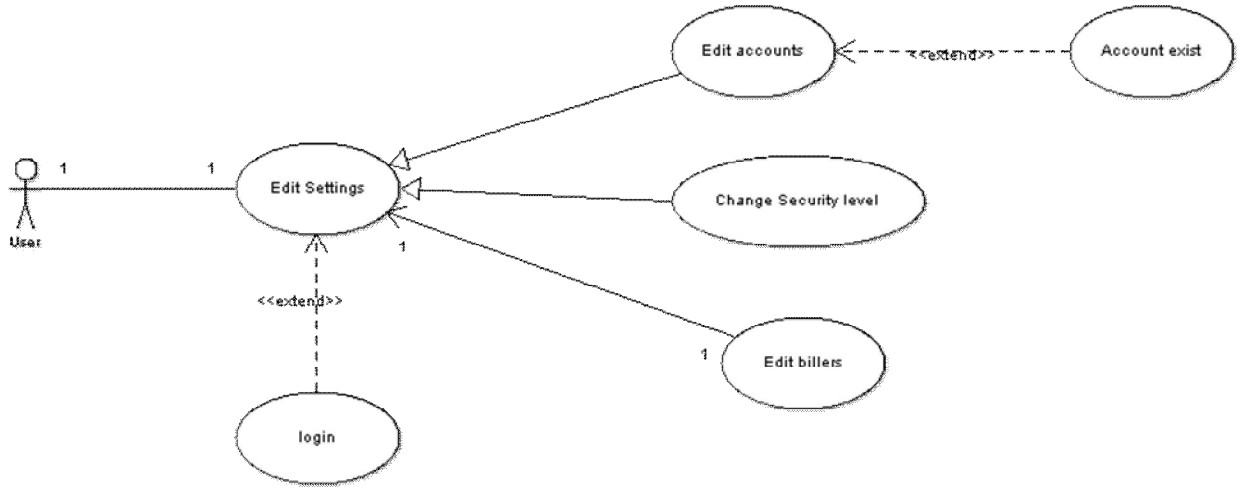
Flow of Events:

i. User enters his old password along with new password.

ii. If old password matches the Database, the password is successfully changed.

iii. If password mismatch occurs between the system and provided by the user, no password change occurs.

d) Name of Use Case: **CHANGE ACCOUNT SETTINGS**



Description: Through this, a user can modify his account settings like changing security level, editing profile details, changing theme, etc.

Pre-Conditions:

i. User is already logged in

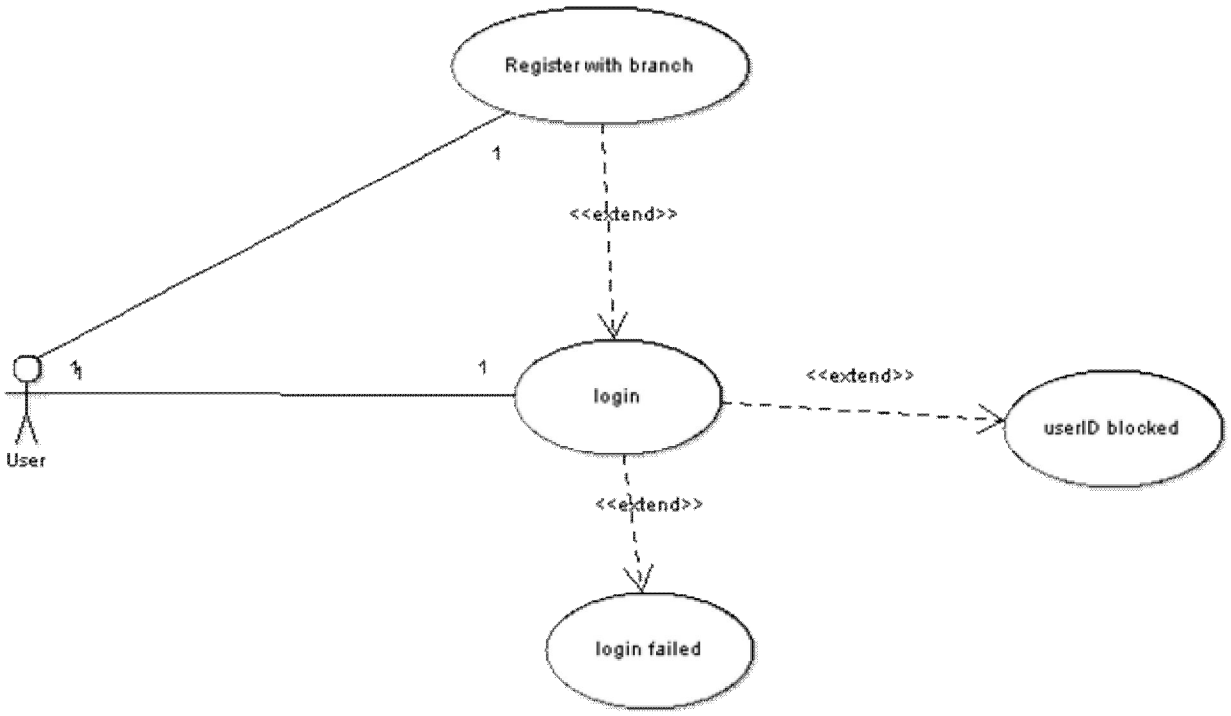
Flow of Events:

i. User selects the option which he wants to edit and gives the corresponding new information.

ii. For the option selected, the pre-conditions must hold in current system state.

iii. On clicking update, the system settings are updated.

e) Name of Use Case: **LOGIN INTO THE SYSTEM**



Description: This module denotes the entry of any user into the system and provides means of securing the authenticated users' data.

Pre-conditions:

i. User is registered with the branch for Internet Banking Service and has been issued a userID & password.

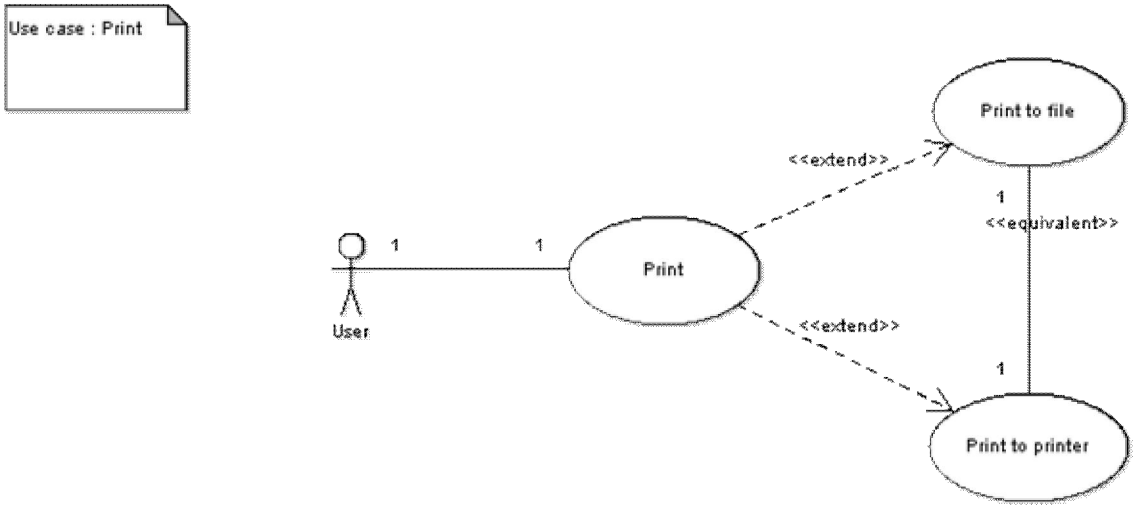
Flow of Events:

i. User enters his username & password allotted to him.

ii. If the information is valid, he is logged in.

iii. If invalid information is provided 3 times in succession, his account is blocked.

f) Name of Use Case: **PRINT BILLS, RECEIPTS, ETC.**



Description: This module is made use of at many places to offer the service of printing in a particular format.

Pre-conditions:

i. User is already logged in.

ii. Printer is attached to the system.

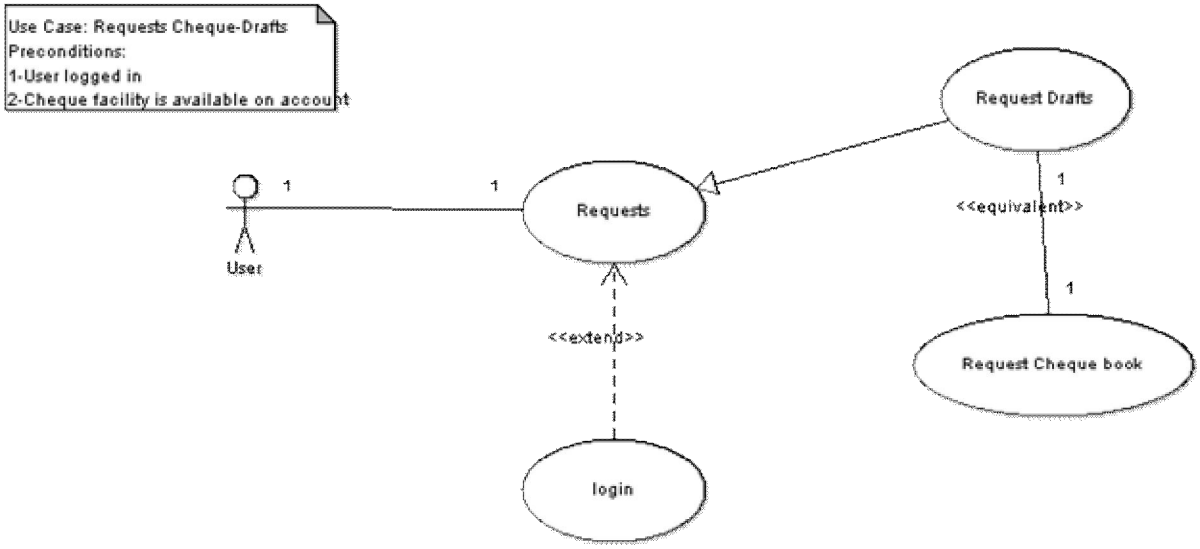
Flow of Events:

i. User avails the service for which he is issued printing instruction.

ii. He is asked to select between printing and saving to a particular file format.

iii. On selecting the desired task is done.

g) Name of Use Case: **REQUEST CHEQUEBOOK/DRAFTS**



Description: Through this module, a user can request bank to issue a cheque book to his account or can also request bank to issue a draft to him.

Pre-conditions:

i. User is already logged in.

Flow of Events:

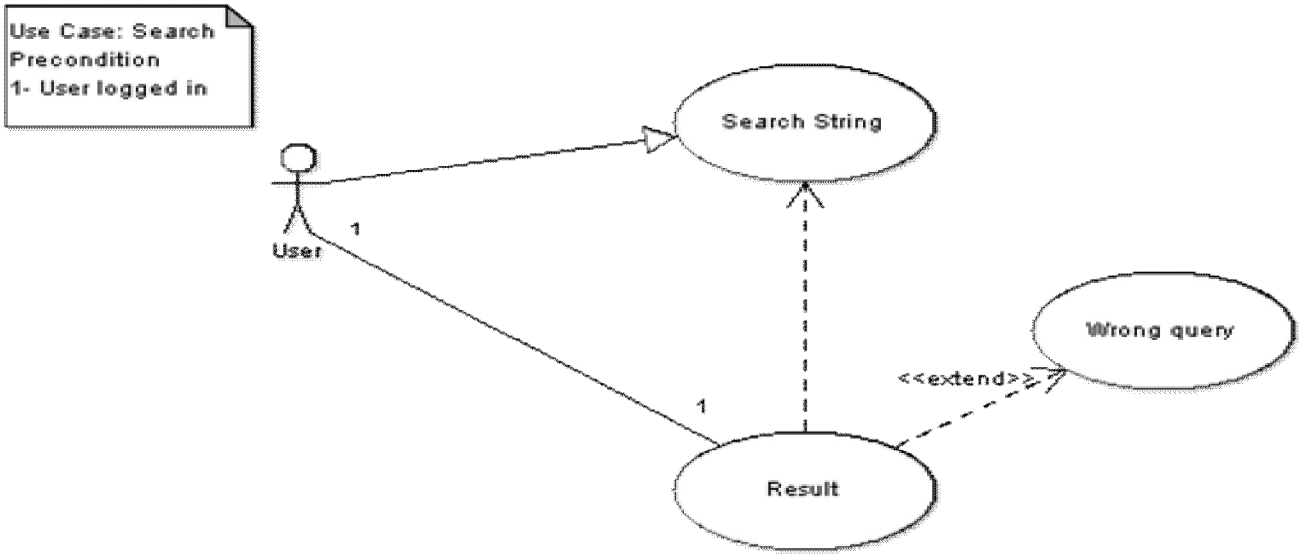
i. User requests for cheque-book or draft.

ii. If cheque book is requested, he is issued a cheque book and is asked for his mode of receiving the cheque book.

iii. If draft is requested, all the details are taken from him and mode of receiving the draft is asked.

iv. User receives his requested cheque-book/draft.

h) Name of Use Case: **SEARCH UTILITY**



Description: Using this module, a user can search for a desired set of outputs such as his previous transactions, etc.

Pre-conditions:

i. User is already logged in.

Flow of Events:

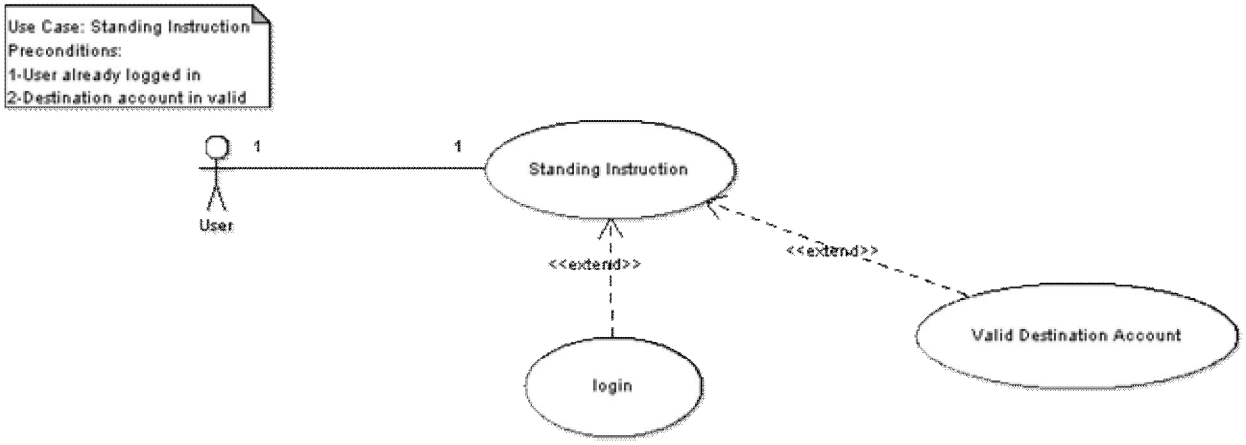
i. User enters the query or the search string.

ii. He selects the type of query.

iii. System searches for the query.

iv. If found, system displays the result in particular format else displays data not found error.

i) Name of Use Case: **ISSUE STANDING INSTRUCTIONS**



Description: A user can issue an instruction to the bank wherein he can instruct the bank to transfer money from his account to some other account periodically.

Pre-conditions:

i. User is already logged in.

ii. Destination Account exists and is valid

Flow of Events:

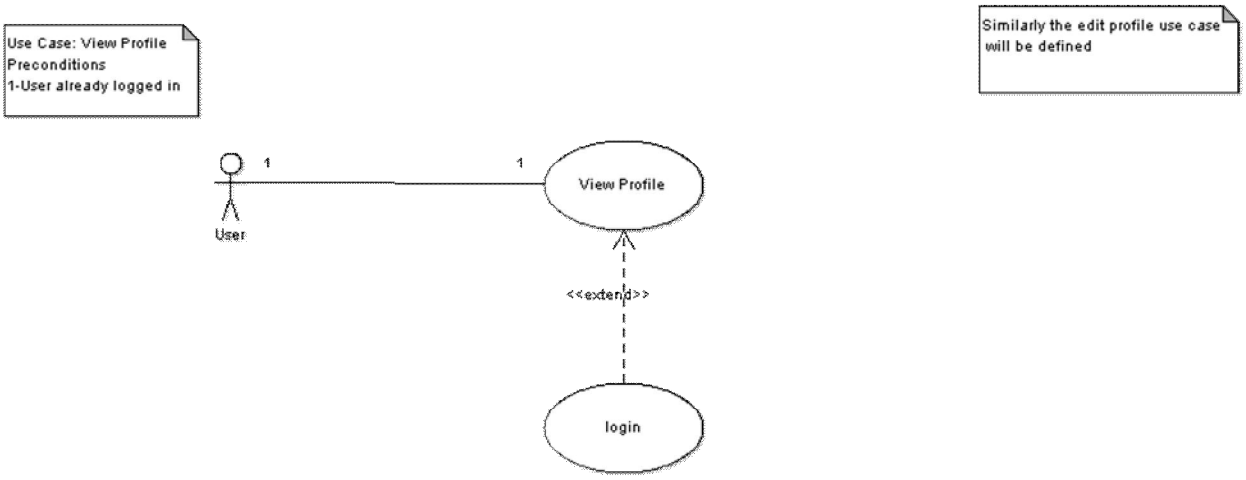
i. User enters his account number and the destination account number and other details.

ii. System stores the information.

iii. A background script running on server checks daily whether the standing instruction has to be followed or not.

iv. If it has to be followed, the corresponding transaction is done.

j) Name of Use Case: **VIEW PROFILE INFORMATION**



Description: Through this, one can view his profile information.

Pre-conditions:

i. User is already logged in

Flow of Events:

i. User clicks the module to view his information.

ii. The system searches the corresponding record and retrieves the information.

iii. It finally displays it in particular format.

II. Supplementary Requirements:

- **Have hours of operation that are 24 x 7 -** Because system can be an automated process, so it can stay open for 24 hours a day. If the base is now the entire world, staying open 24 hours a day becomes critical. System is required to be available 24x7 so UPS support must be on server site for at least 8 hours in case of power failure. System will remain inaccessible to users at 2:00 to 4:00 am for backup and maintenance purpose.

- **Make the existing Web site more dynamic in nature -** Many early Web implementations consisted of static HTML pages. This becomes very difficult to manage if the number of pages gets too large. An effective system should be largely dynamic taking advantage of technology that automates this process rather than relying on manual processes. Application should serve dynamic user based customized web pages to its clients from server.

- **Tie the existing Web site into existing enterprise systems –** Any existing Web site that relies on the manual duplication of data from another system is one that can be improved. Most of the business data in the world today exists in enterprise servers that can be connected to the Web servers to make this process far more effective.

- **Provide good performance and the ability to scale the server –** The Web
  Application Server should provide good performance and the ability to manage performance with techniques, such as support for caching, clustering, and load balancing.

- **Providing session management capability -** Web application developers should not spend valuable time worrying about how to maintain sessions within the application. The Web Application Server should provide these services.