TEENOVATORS

Core Banking
Software Requirements Specification

College

IT-BHU,Varanasi

Team Members

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

Faculty Guide

Mr. Bhaskar Biswas

E-Mentor

Mr. Vikas Manoria

Contents

Software Requirements Specification

1. Introduction:

A Core Banking Software is the key to the application of modern technology in the banking sector. It provides innumerable ways to perform tasks consistently, accurately and with versatility. Core Banking is the networking of branches which enables a customer to avail the entire bank services irrespective of where he/she maintains his/her account.

1.1 Purpose:

The present scenario views the need of deploying a technological solution that encompasses every aspect of the banking operations and provides a complete suite of delivery channels. It is the turn of the transaction processing systems to dominate the arena and transform banking processes and practices the world over to achieve higher levels of differentiation, agility and operational efficiency. The Core Banking would prove the most fruitful step in this field.

1.2 Scope:

1.2.1 Includes different handling procedures for Indian and NRIs.

1.2.2 Maintainence of the ACID properties of the schema and records.

1.2.3 Proper Encryption of passwords and other confidential records to ensure security.

1.2.4 Special emphasis on the rural sector including support for small scale industries, etc.

1.2.5 Some additional services such as ATM services, Internet Banking, payment of utility bills and others.

1.2.6 Inclusion of some International Banking standards and taking care of RISK Management.

1.2.7 Consideration for the currency issue and stock relations.

1.2.8 Proper database relations to keep track of all the relevant information regarding the customers.

1.2.9 Appropriate search functionality on various criterion.

1.2.10 Functionality of transactions including e-mails and chats.

1.2.11 Creating of proper log files to keep a record of various information.

1.2.12 Provision of separate user-ids and passwords to customers through which the various details regarding their bank accounts can be viewed.

1.3 Definitions:

- **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.

➢ **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.

1.4 References:

➢ Online References:

- www.sbi.co.in
- www.ucobank.com
- www.icicibank.com

➢ Books:

- Database System Concepts by Abraham Silberschatz

➢ IEEE SRS Format

➢ Project Scenario and specification given by IBM

1.5 Technologies:

1.5.1 Eclipse IDE

1.5.2 WASCE-WebSphere Application Server Community Edition

1.5.3 DB2 Express

1.6 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:

2.1 <u>Product Perspective:</u> The overall look to this application can be viewed through the following diagram.

2.1.1 The software provides different sections for each section viz. Personal (for general public), NRI (Non-Resident Indians), Rural Sector (incl. small-scale industries, agriculture banking), International and Corporate Banking and also providing some out of the way services.

2.1.2 It is created in the form of a web application so that anyone can access it through the internet when the total database is kept on a server.

CoreBanking
Personal
Accounts
Loans
Transaction
NRI
NRIAccounts
Finance
Remittance
RuralSector
SmallScale
Agriculture
RuralLoans
International
Trade
Treasury
OBU
Corporate
Services
Bills
ATM
Taxes
Others

2.2 **Product Function:** The various sections of the product are described below with their functions:

2.2.1 **Personal Tab:** This portion shall be useful only for the employees as it shall ensure the administrative work such as maintainence of different types of account and loan details, transaction functions and few other jobs. Owing to such confidential functions, access to this shall be possible only through a secure method of logging in.

2.2.2 **NRI:** This shall have similar functions as personal tab but with a different purpose.

2.2.3 **Rural Sector:** This tab is primarily created so as to cater to the increasing needs of the rural sector in setting up various farming phenomenon. It also has special specification for small-scale industries. Besides, all the Rural Branches are also handled under this section.

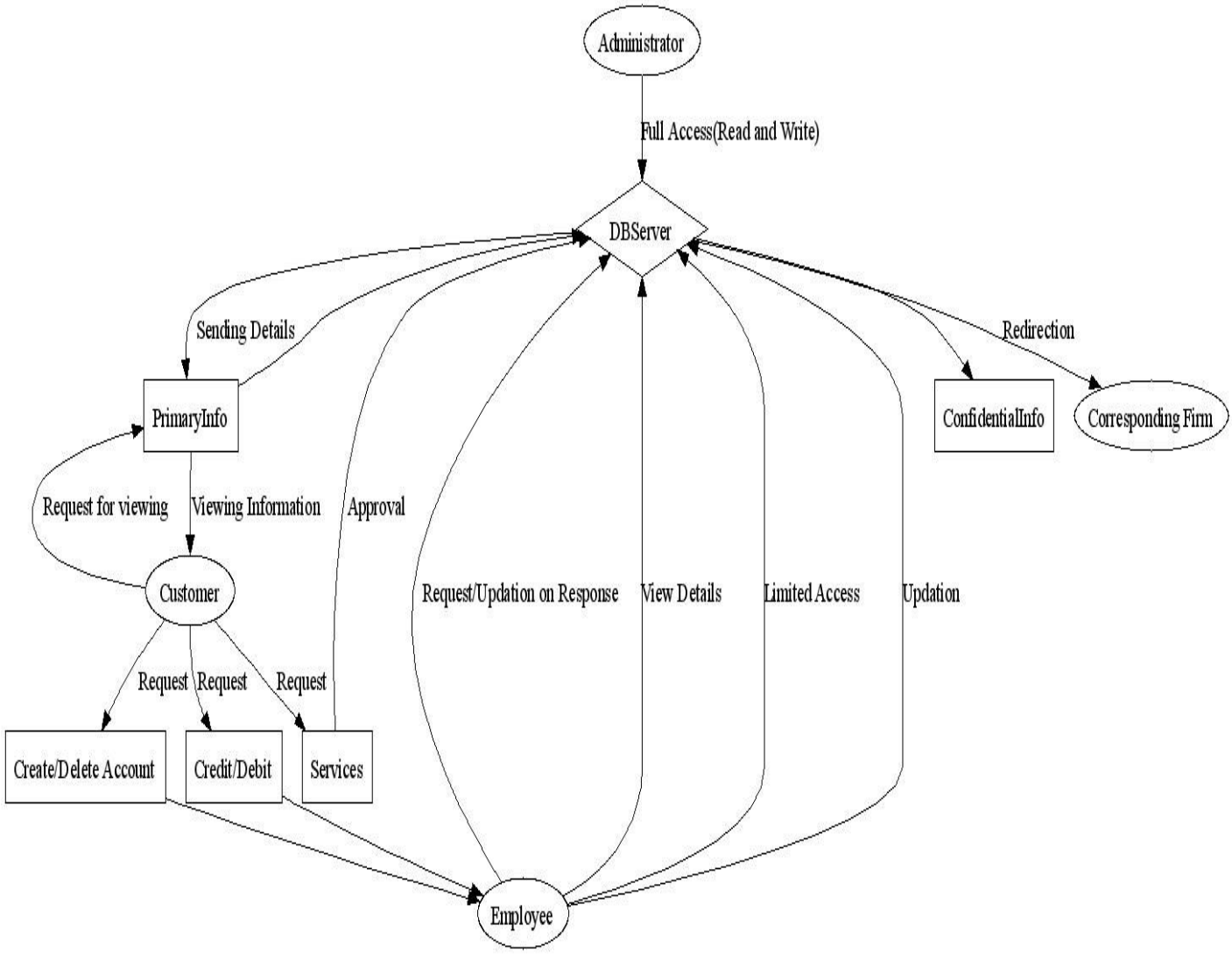
2.2.4 **International Banking:** It includes features such as provision of foreign currency loans, financial services for the purpose of import/export, Treasury services, etc.

2.2.5 **Corporate Banking:** Under this section, we provide capital finance to both domestic as well as international users. It also includes many term loans, deferred payment guarantees, etc.

2.2.6 **Services:** Under this tab we handle many exclusive features which are totally for the use of customers. This includes ATM services, e-Banking, Bill payment, Tax Payments, Stock Purchases, etc.

2.2.7 A special technique for RISK Management has also been implemented.

2.3 Use-Case Model Survey: The following diagram indicates the use-case modelling simulation of a basic Banking Project. The description follows the diagram.



To explain the overall theory of the project, we have decided three vantage points as shown in the above diagram.

- Customer point of View: A customer could have various demands from an application as this-

- Creation or deletion of account.
- Credit or debit balance from account.
- Loan Hiring or Loan Payment.
- Viewing his account details.
- Availing the extra services provided by a banking system.

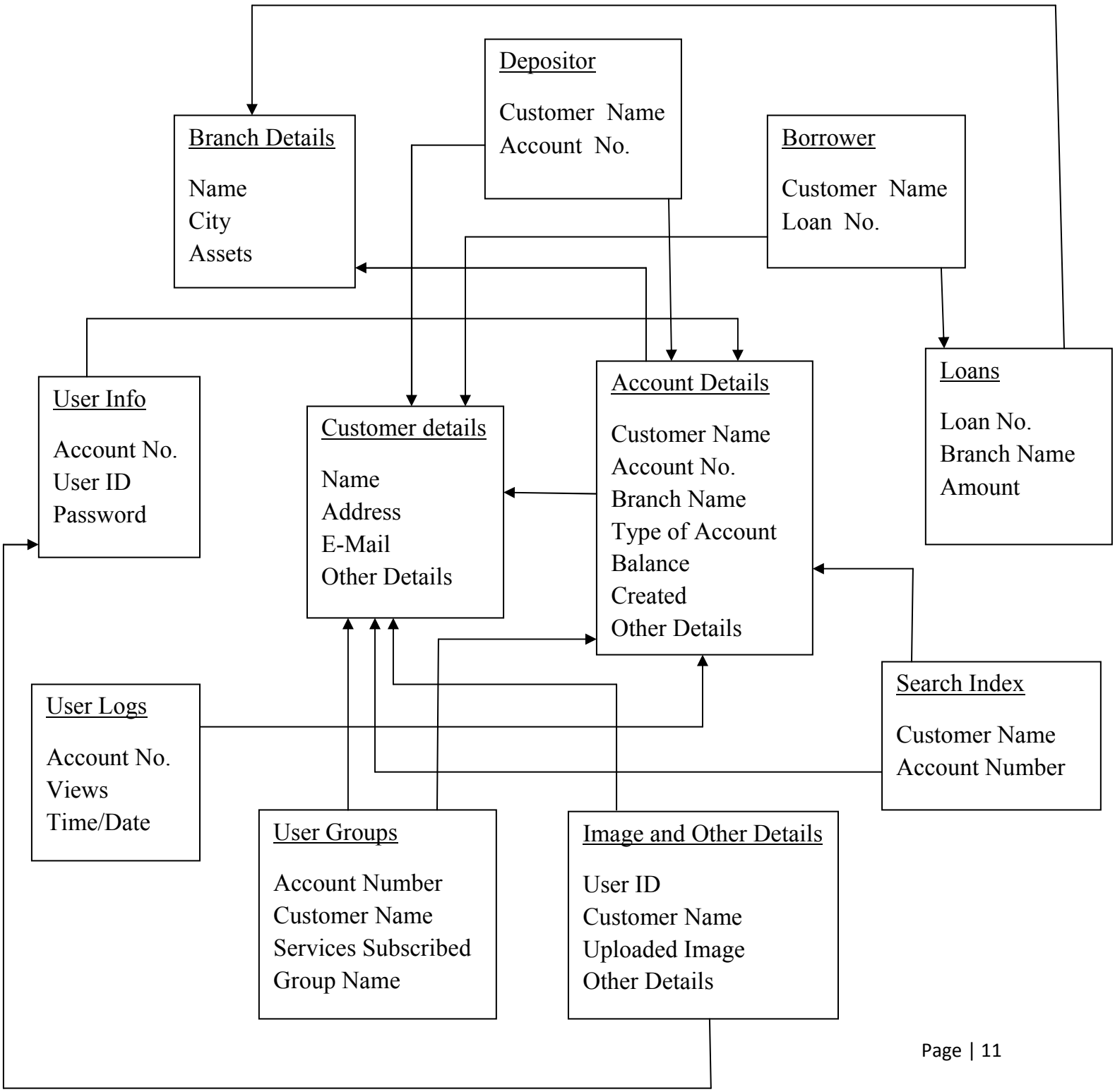
➢ Employee point of View: The following are the jobs of an employee in a banking system.

- Accept various requests of creation/deletion of accounts.
- Crediting or debiting the amount from one's account.
- Forwarding the request of a loan sanction to the Administrator.
- Updation of records in the Database after any transaction or operation.

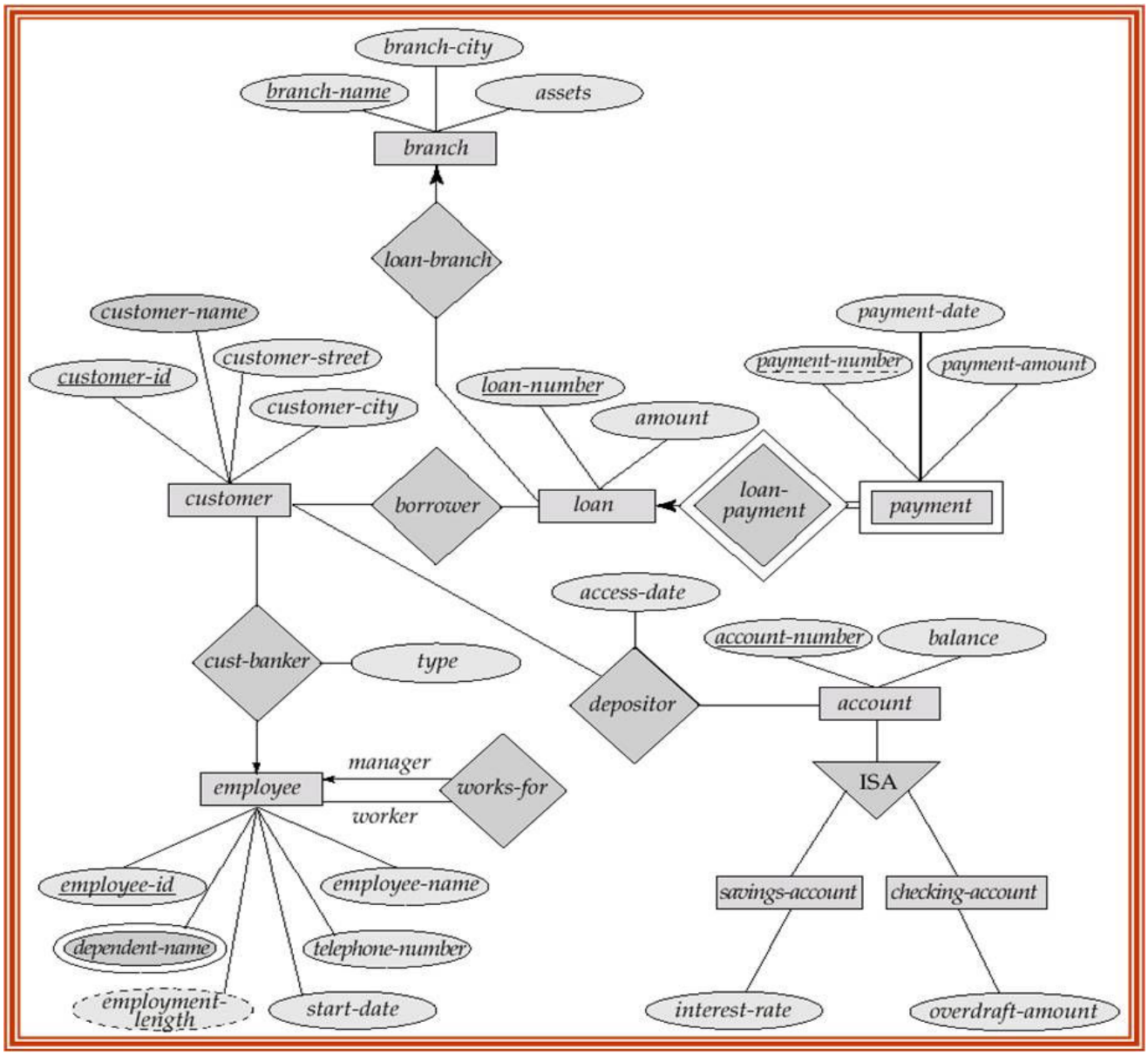
➢ Administrator point of View: The administrator needs to check the following:

- Inter-operatibility between various sections of the system.
- Ensure proper system functioning.
- Take care of the RISK issue and take appropriate management techniques.

2.4 Schema Design: The schema design shown below is self-explanatory. This is a basic Schema or Database design for any Banking Enterprise. However, we have included a few more relations in it to provide some extra features.


Depositor
Customer Name
Account No.
Branch Details
Name
City
Assets
Borrower
Customer Name
Loan No.
User Info
Account No.
User ID
Password
Customer details
Name
Address
E-Mail
Other Details
Account Details
Customer Name
Account No.
Branch Name
Type of Account
Balance
Created
Other Details
Loans
Loan No.
Branch Name
Amount
User Logs
Account No.
Views
Time/Date
Search Index
Customer Name
Account Number
User Groups
Account Number
Customer Name
Services Subscribed
Group Name
Image and Other Details
User ID
Customer Name
Uploaded Image
Other Details

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



2.6 Assumptions:

- The current values for the Database are provided manually and has no real connection with any system database. However, the database can be recreated easily.
- The administrator account is created by default in the project.

3 Specific Requirements:

3.1 Use-case Reports:

- **Administrator**: The person in this designation enjoys full access over the DBServer and thereby has full privileges over the system.
  - Managing System Users- view, create, delete and update information pertaining to them.
  - Viewing Logs.

Name of Use Case:	Viewing User Information
Description:	View the list of customers and employees and all other system users who are assigned some role in the system.
Preconditions:	(i) Administrator is already logged in. (ii) System Users have been already assigned tasks.
Flow of Events:	(i) A particular system user is selected. (ii)The required query is submitted to the DBServer. (iii) The corresponding output is displayed.

Name of Use Case:	Creating User Accounts
Description:	Create accounts of various customers and employees and all other system users.
Preconditions:	(i) Administrator is already logged in.

Flow of Events: (i) All the details like login name,password and other relevant details are collected.
(ii)The required information is submitted to the DBServer with a request of creating account.
(iii) The corresponding account is created.

Alternate Flow of Events: (i) A message for duplicate login name appears.
(ii) The administrator has to enter all the details again.

Name of Use Case: Updating User Information
Description: Update the details of customers and employees and all other system users.
Preconditions: (i) Administrator is already logged in.
(ii) System Users have been already created.
Flow of Events: (i) A particular system user is selected.
(ii)Assign or Revoke the tasks, permissions, etc.
(iii)The required query is submitted to the DBServer.
(iv) The corresponding changes are updated on the server.

Name of Use Case: Viewing Log Information
Description: View the activites (log) of customers and employees and all other system users.
Preconditions: (i) Administrator is already logged in.
(ii) System Users have been already created.
Flow of Events: (i) A particular system user is selected.
(ii) The required date is selected.
(iii)The required query is submitted to the DBServer.
(iv) The corresponding log file is displayed.

- **Employee**: The next subordinate to the position of Administrator.

- He entertains and processes the requests of customers to create/delete account, credit/debit money and other jobs.
- He is responsible for updation of product/service details.

Name of Use Case:	Crediting Money to Account
Description:	To credit the money to a user's account in case he deposits the money through any means.
Preconditions:	(i) Employee is already logged in. (ii) Money has been deposited by customer.
Flow of Events:	(i) The particular account details are viewed. (ii)The amount is credited by increasing the balance. (iii) The details are updated on the DBServer.

Name of Use Case:	Debiting Money from Account
Description:	To debit the money from a user's account in case he wishes to withdraw the money through any means.
Preconditions:	(i) Employee is already logged in. (ii) Request for withdrawal has been submitted by customer.
Flow of Events:	(i) The particular account details are viewed. (ii) Constraints such as minimum balance, etc are verified (iii) After verification, the amount is debit from the account by decreasing the balance. (iv) The details are updated on the DBServer.

Name of Use Case:	Creating Product/Services catalogue.
Description:	The product and services details are maintained in the form of catalogue.
Preconditions:	(i) Employee is already logged in.
Flow of Events:	(i) All the product and service details are entered. (ii)The required information is submitted to the DBServer. (iii) The corresponding service creating is entered in the log.

3.2 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.