Software Requirements Specification

Banking Database Management System

Version 1.0

TEAM:

1. CHAITANYA KODLEKERE
2. K S SHRINIKET
3. SINCHANA R
4. SUMUKH V SHANKAR

Table of Contents

1.Introduction

2. Overall Description

3. System Features

4. External Interface Requirements

5. Other Non-functional Requirements

1. INTRODUCTION

1.1 PURPOSE

The project titled “Banking management system” is a computerized telecommunications device that provides access to financial transactions in a public space without the need for a clerk or bank teller. Primary purpose of this project is to provide a customer to create an account and make transactions. Along with it, customer will also be allowed to apply for loan and pay the instalments of existing loan.

1.2 DOCUMENT CONVENTIONS

This document uses the following conventions.

DB – Database
ER – Entity Relationship
NF – Normal Form
FR – Functional Requirements

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

This project is a prototype for the banking management system and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project intends to ease the work of all the employees in the banking sector as well as to provide ease of access to customer.

1.4 PROJECT SCOPE

The purpose of the online banking management system is to reduce the crowd in bank and save time of both customers as well as employees. The system is based on a relational database with employees and customers. We will have a database server supporting creation of hundreds of accounts across any branch and also enable the customers to transfer funds 24 / 7. Above all, we hope to provide an overall better banking experience.

1.5 REFERENCES

- Fundamentals of Database Systems by Ramez Elmasri and Shamkant B. Navathe
- Software Engineering: A Practitioner's Approach by Roger S. Pressman and Bruce R. Maxim

2. OVERALL DISCRIPTION.

2.1 Product Perspective.

The manual banking system suffers from the following drawbacks:

- The existing system involves a lot of paper work and manual calculations. This has lead to inconsistency and inaccuracy in maintenance of data.
- The data, stored on papers or files, may be lost, stolen or destroyed.

- The existing system is time consuming causing inconvenience to customers and banking staff.
- Since the number of customers have drastically increased therefore maintaining and retrieving detailed records of customers is extremely difficult.

Hence the banking management system is proposed with following Product perspective:-

- The computerization of banking system will reduce a lot of paperwork and hence the load on the banking administrative staff.
- The calculations are performed by machine. Hence the chances of errors are almost nil.
- The customers can retrieve balance, transaction and loan details easily and any required addition, deletion or updation can be performed.
- Each customer has to be validated before using bank services. Hence, unauthorised access is prevented.

2.2 Product Features.

The "Banking Management System" Software is an independent web based applications. The are various interfaces related to the software. These interfaces help the user to interact with the software and provide necessary information for online banking.

The following are the interfaces and product features:

❖ Client Interface of Product

The client will have client interface in which he can interact with the banking system. It is a web-based interface which will be the web page of the banking application. The home page is displayed with the various options and services provided by the bank. The home page has a dashboard which has login option. Choosing login option redirects to login page where the user can enter the login details. If the user is a new customer then he/she can register to become a customer of the bank. If the login particulars are valid then the user is taken to a home page where he has the dashboard with the entire service and transaction list that he can perform with the bank. All the above activities come under the client interface.

❖ Administrative Interface of Product

The administrator will have an administrative interface which is a GUI so that he can view the entire system. He will also have a login page where he can enter the login particulars. On entering correct login details he will be redirected to admin home page where he can perform all his actions. This administrative interface provides different environment such that he can maintain database & provide backups for the information in the database. He can validate new users who have registered to become a new customer of the bank. The he adds the new customer to the database. He can view the loan requests & perform action to issue or reject the loan requests by the clients.

❖ Banking Database

The banking database stores all the information in the following modules:

1) Login Process

This module allows valid customers to access the functionalities provided by the bank.

2) Balance Enquiry

This module maintains the balance details of a particular account.

3) Update Profile

This module allows the customer to update profile of their account.

4) Funds Transfer

This module allows the customers to transfer funds from one account to another within the same bank or other banks.

5) Change of Password

This module allows customers to change their password.

6) Mini Statements

This module allows customers to view their transaction details.

7) Loan Request

This module allows customers to apply for new loan by filling loan application.

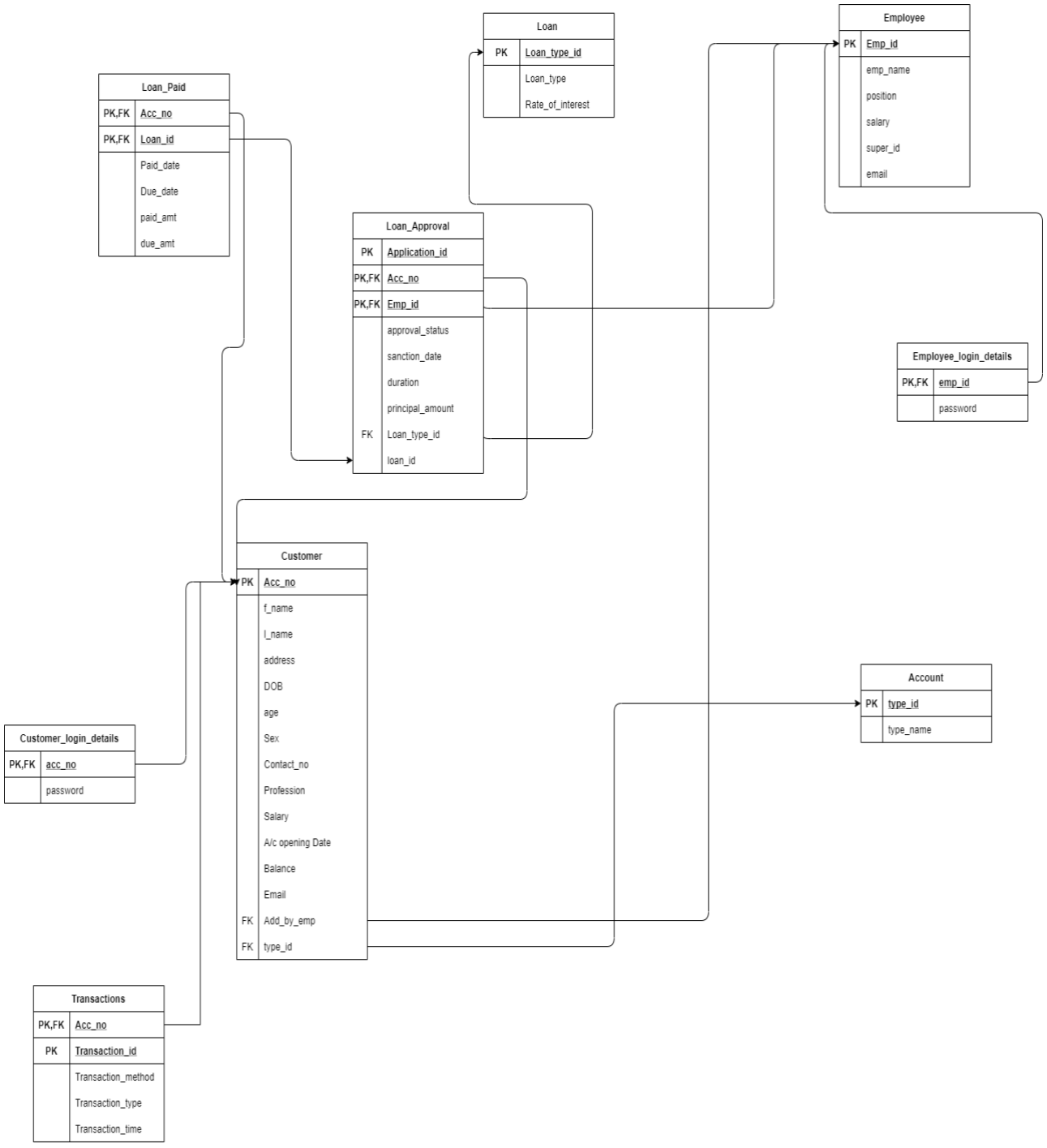
8) Loan Payment

This module allows customers to pay the loan instalments.

9) Employee Module

This module allows employees(administrators) to keep track of every customer and their transactions, loan payments etc.

The following E-R Diagram (Relational Schema) shows the major features of the banking database system:



2.3 User Classes & Characteristic

Customers: The users must have an account. The customers with an account can do the following:

- ✓ Customer should be able to login to respective account any time he/she desires.
- ✓ Customer may apply for loan.
- ✓ Customer may apply for loan.
- ✓ Customer can make transactions.
- ✓ Allow customer to change password according to their choice
- ✓ Allow customers to view their transaction history

Employees: The Employee should have following management functionalities:

- CUSTOMER FUNCTIONS.
  - Get all customers who have an account in the bank.
  - View transaction history of customer.
  - Approve or disapprove loan applications filled by customers.
  - Check if the instalments paid by customers are on time.

- ADMINISTRATIVE
  - Add/Delete a loan type.
  - Update loan interest rates.
  - Add/Delete a transaction type
  - Add/Delete a new employee

2.4 Operating Environment.

- Web based Application with Distributed Database
- Client/ Server system
- Operating system: Windows.
- DBMS: MySQL
- Frame Work: Express JS
- Frontend: JavaScript

2.5 Design & Implementation Constraint.

Design Constraints

- The global schema, fragmentation schema, and allocation schema.
- SQL commands for above queries/applications

Implementation Constraints

- This system works only on a single server.
- This is designed in Express JS.
- Language used is Java Script.
- Limited to HTTP/HTTPS protocols.

2.6 User Documentation.

A registered user can have following facilities:
→ Accounts and accounts status.
→ The balance enquiry.
→ The fund transfer standards.
→ Loan Requests.
→ Password Changing.

2.7 Assumption & Dependencies.

→ Every user should be comfortable of working with computer.
→ Every user should be comfortable with net browsing.
→ He should be aware of the banking system.
→ He must have basic knowledge of English.

3. SYSTEM FEATURES

3.1 FUNCTIONAL REQUIREMENTS

- **Customer Requirements**

FR. No	FR Description
1	Customer should be able to create account.
2	Customer should be able to login to respective account any time he/she desires.
3	Customer may apply for loan.
4	Allow customers to pay instalments.
5	Customer can make transactions 24 / 7.
6	Allow customer to change password according to their choice.
7	Allow customers to view their transaction history

- **Employee Requirements**

FR. No	FR Description
1	Employee should be able to add a customer.
2	Employee may approve / disapprove loan applications of a customer.
3	Employee can view transaction history of customers.
4	Employees may or may not have supervisors.
5	Employees can change their password linked to their ID.
6	Supervisors can view the details of supervisee under them.

3.2 CLIENT / SERVER SYSTEM

The term client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the DBMS (also known as the back-end).

A client/server system is a distributed system in which,

- Some sites are client sites and others are server sites.
- All the data resides at the server sites.
- All applications execute at the client sites.

4. EXTERNAL INTERFACE REQUIREMENT.

4.1 User Interface

There are five different ways for a user to interact with the system:

Viewers:
Many unknown persons or un-authenticated persons visit the Bank official web-site via internet. They collect the information and search what are the schemes are available in the bank web page. Those viewers or visitors may or may not became the customer of the bank.

New User:
These are the people who have visited the Bank webpage or heard about the bank. These people want to open a new account in bank. They become a new customer by filling the bank's new customer registration form, submitting it and start a new account.

Existing User:
The Existing user is the most typical user of the Online Banking system. Each Users have their own account and authorized login access. The Existing user can login in online to their account perform the operation of fund transfer, balance queries and transactions. All the operations of the banking are done online. This is helpful for user because this saves time and is more efficient process than offline banking.

Employee:
These are the employees of the bank that have their own log credentials. They may or may not have account in the bank. They can overview the transactions done by the customer. They can check the list of newly registered customers to verify if the new user trying to register is valid or not. They can accept or reject loan application depending on their role. They can help the customers with any enquires.

Administrator:
Admin is master user of the system because they are main role of the system. Admin grant and maintain the database of the existing user and grant the permissions to users. It over rules all other users.

4.2 Hardware Interface

Client Side:

- *User on Internet* : Web Browser, Operating System(any)
- *Application Server* : WAS
- *Data Base Server* : DB2
- *Network* : Internet
- *Development Tools* : JavaScript, HTML,OS(Windows).

Server Side:

- *Processor* : iCore3 or higher.
- *RAM* : 4 GB or more.
- *Disk Space* : More than 2 TB.
- *DBMS* : MySQL
- *Frame Work* : Express JS and Node JS

4.3 Software Interface.

Operating System	We have chosen WINDOWS operating system for its best support and user-friendliness.
DBMS	MySQL
Programming Language	Java Script
Frame Work	Express JS
Network	Internet

4.4 Communication Interface.

Golden Horizon Bank website i:e www.goldenhorizon.com offers all banking facilities anywhere any time through internet facility.

→ Client on Internet will be using HTTP/HTTPS protocol.

→ Client on Intranet will be using TCP/IP protocol

→ This project works on all types of browsers.

5. NONFUNCTIONAL REQUIREMENTS

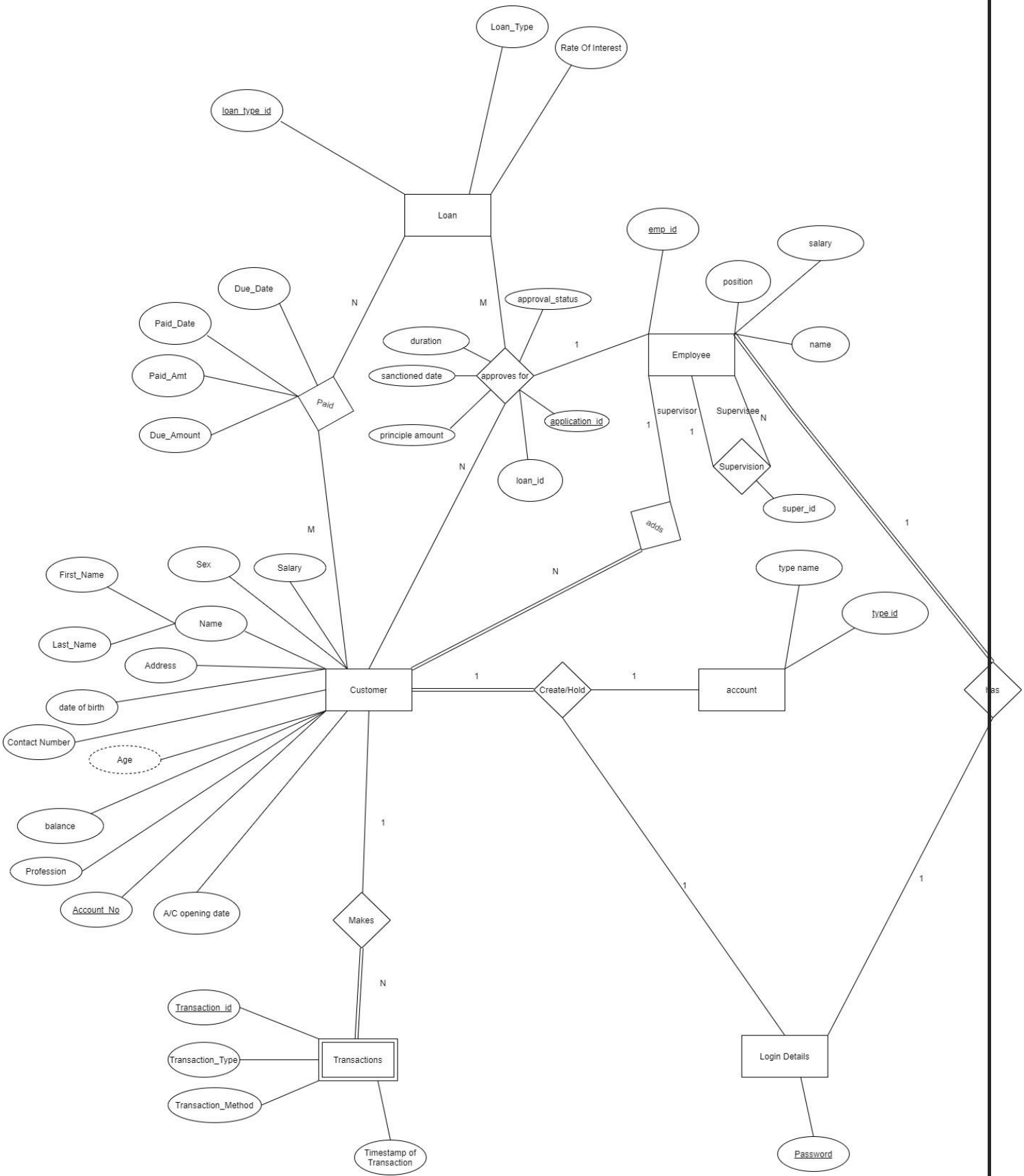
5.1 PERFORMANCE

The steps involved to perform the implementation of the banking management system are listed as below.

A) ER DIAGRAM

The E-R Diagram constitutes a technique for representing the logical structure of a database in a pictorial manner. This analysis is then used to organize data as a relation, normalizing relation and finally obtaining a relation database.

- ENTITIES: Which specify distinct real-world items in an application.
- PROPERTIES/ATTRIBUTES: Which specify properties of an entity and relationships.
- RELATIONSHIPS: Which connect entities and represent meaningful dependencies between them.


Loan_Type
Rate Of Interest
loan_type_id
Loan
emp_id
salary
position
name
Employee
Due_Date
Paid_Date
Paid_Amt
Due_Amount
Paid
N
M
approval_status
duration
sanctioned date
principle amount
approves for
application_id
loan_id
1
supervisor
Supervisee
N
1
Supervision
super_id
adds
M
N
1
First_Name
Sex
Salary
Name
Last_Name
Address
date of birth
Contact Number
Age
balance
Profession
Account_No
A/C opening date
Customer
1
Create/Hold
1
account
type name
type id
has
1
Makes
N
1
1
Transaction_id
Transaction_Type
Transaction_Method
Transactions
Timestamp of Transaction
Login Details
Password

B) NORMALIZATION

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored.

If a database is not properly designed it can give rise to modification anomalies. Modification anomalies arise when data is added to, changed or deleted from a database table. Similarly, in traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database.

Normalization is the process of breaking down a table into smaller tables. So that each table deals with a single theme. There are three different kinds of modifications of anomalies and formulated the first, second and third normal forms (3NF) is considered sufficient for most practical purposes. It should be considered only after a thorough analysis and complete understanding of its implications.

This project includes Normalization upto 3NF, hence making the database more efficient and compact as ever.

C) PERFORMANCE DEPLOYEMENT

- System can withstand even if many no. of customers request the service. Access is given to only valid users of bank who requires the services such as balance enquiry, update profile, funds transfer, mini statements, and loan installment payments.
- It is available during whole week for all 24 hours.

5.2 SAFETY REQUIREMENTS

- ✓ If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backup log, up to the time of failure.
- ✓ By incorporating a secure database and proven network protocol into the system, reliable performance and integrity of data is ensured.
- ✓ There must be a power backup for server system. Since the product is of 24x7 availability there should be power backup for server which provides the information.

5.3 SECURITY REQUIREMENTS

Security is often an afterthought during software development. Realizing security early, especially in the requirement phase, is important so that security problems can be tackled early enough before going further in the process and avoid rework. A more effective approach for security requirement engineering is needed to provide a more systematic way for eliciting adequate security requirements. In the banking database, we have provided customers as well as employees a unique login id so that only they can have access to their particular accounts.

5.4 SOFTWARE QUALITY ATTRIBUTES

- AVAILABILITY:

  The website will be up 24 /7. Hence, customers as well as employees can login to their respective accounts and perform desired operations whenever required.

- CORRECTNESS:

  The data of the customers will be displayed correctly as long as they have provided accurate data. Operations such as transactions or loan payment are handled carefully so that the funds flow to the right place.

- MAINTAINABILITY:

  Server checking and maintenance will be done regularly to make sure that there is no loss of data or to speed up the transactions and make the system more efficient.

- USABILITY:

  - The website's interface has to be user-friendly and easy to use.
  - Efficiency of use: goals are easy to accomplish quickly and with few or no user errors
  - Intuitiveness: the interface is easy to learn and navigate; buttons, headings, and help/error messages are simple to understand
  - Low perceived workload: the interface appears easy to use, rather than intimidating, demanding and frustrating.