VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAUM-590014

A DBMS Mini Project Report On

"BANK MANAGEMENT SYSTEM"

Submitted in the partial fulfillment of the requirements for the award of the Degree of

Bachelor of Engineering in Computer Science and Engineering

Submitted by

R VIKRANTH (1OX15CS081)

SAHANA SHREEDHAR KULKARNI (1OX15CS093)

TALARI SURYA TEJA (1OX15CS114)

Under the guidance of

Ms. Deepa

Project Guide

Department of Computer Science and Engineering

The Oxford College of Engineering

Bommanahalli, Bangalore-68

2017-2018

THE OXFORD COLLEGE OF ENGINEERING

Bommanahalli,Hosur Road,Bangalore – 560068

(Affiliated To Visvesvaraya Technological University, Belgaum)

Department of Computer Science and Engineering

CERTIFICATE

Certified that the project work entitled **"BANK MANAGEMENT SYSTEM"** carried out by **R Vikranth (1OX15CS081), Sahana Shreedhar Kulkarni (1OX15CS093), Talari Surya Teja (1OX15CS114)** bonafide students of The Oxford College of Engineering, Bangalore in partial fulfillment for the award of the Degree of **Bachelor of Engineering in Computer Science and Engineering** of **Visvesvaraya Technological University, Belgaum** during the year 2017-2018. The Mini project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Ms.Deepa	**Dr. R.J.Anandhi**	**Dr. R.V Praveena Gowda**
Project Guide	**H.O.D,Dept. of CSE**	**Principal, TOCE**

External Viva

Name of the Examiners	**Signature with Date**
1._____	_____
2._____	_____

ACKNOWLEDGEMENT

A project is a job of great enormity and it can't be accomplished by an individual all by them. Eventually, we are grateful to a number of individuals whose professional guidance, assistance and encouragement have made it a pleasant endeavor to undertake this project.

It gives us great pleasure in expressing our deep sense of gratitude to our respected Founder Chairman **Late Shri S. Narasa Raju** and to the respected Chairman **Shri S.N.V.L Narasimha Raju** for having provided us with great infrastructure and well furnished labs.

We take this opportunity to express our profound gratitude to our respected Principal **Dr. R.V Praveena Gowda** for his support.

We are graceful to the Vice Principal and Head of the Department **Dr. R.J. Anandhi** for her unfailing encouragement and suggestion given to us in the course of our project work.

Guidance and deadlines play a very important role in successful completion of the project on time. We convey our gratitude to **Ms.Deepa,** Project Guide for having constantly guided and monitored the development of the project.

A note of thanks to the Department of Computer Science Engineering, both teaching and non-teaching staff for their co-operation extended to us.

We thank our parents for their constant support and encouragement. Last, but not the least, we would like to thank our peers and friends.

R VIKRANTH
SAHANA SHREEDHAR KULKARNI
TALARI SURYA TEJA

ABSTRACT

The project is based on the bank management system. This application is used by the administrator of the bank which will help him to maintain records of the bank. Using this application the administrator can create an account for the user, make deposits in the customer's account, withdraw amount from the customer's account and display the passbook details accordingly. The administrator can also delete a user's account on the customer's demand. This application has a GUI (graphical user interface) which will make the user easily understand the application and feed data accordingly. The data that is feed in the front end will be stored in the backend. The frontend is created using java(NetBeans) and the backend is created using MySQL. The frontend and backend are connected using java connectivity code. A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server. It consists of trigger the converts the customer's name from lower case to upper case. A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs. It consists of a stored procedure that displays the current balance of the customer when he mensions his aadhar number. Care is taken while creating an account or while making any bank transactions that the balance cannot go below Rs.1000.The applications cautions the user with dialog boxes when the any invalid data is given by the user. It also pops up dialog boxes on successful modifications made on request.

ACKNOWLEDGEMENT

A project is a job of great enormity and it can't be accomplished by an individual all by them. Eventually, we are grateful to a number of individuals whose professional guidance, assistance and encouragement have made it a pleasant endeavor to undertake this project.

It gives us great pleasure in expressing our deep sense of gratitude to our respected Founder Chairman **Late Shri S. Narasa Raju** and to the respected Chairman **Shri S.N.V.L Narasimha Raju** for having provided us with great infrastructure and well furnished labs.

We take this opportunity to express our profound gratitude to our respected Principal **Dr. R.V Praveena Gowda** for his support.

We are graceful to the Vice Principal and Head of the Department **Dr. R.J. Anandhi** for her unfailing encouragement and suggestion given to us in the course of our project work.

Guidance and deadlines play a very important role in successful completion of the project on time. We convey our gratitude to **Ms.Deepa,** Project Guide for having constantly guided and monitored the development of the project.

A note of thanks to the Department of Computer Science Engineering, both teaching and non-teaching staff for their co-operation extended to us.

We thank our parents for their constant support and encouragement. Last, but not the least, we would like to thank our peers and friends.

SAHANA SHREEDHAR KULKARNI

TALARI SURYA TEJA

R.VIKRANTH

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Preamble

Bank:

A bank is a financial institution that accepts deposits from the public and creates credit.

Banking:

Banking in its modern sense evolved in the 14th century in the prosperous cities of renaissance Italy but in many ways was a continuation of ideas and concepts of credit and leading that had their roots in the ancient world.

Types of banking:

Retail banking: Dealing directly with individuals and small businesses.

Business banking: Providing services to mid-market business.

Corporate banking: Directly at large business entities.

Private banking: providing wealth management services to high-net-worth individuals .

1.2 Problem statement

The previous banking system which was non-computerized is done manually, by writing the transaction details in a book.This process takes long time for any banking process.This method of non-computerized banking is less productive.Sometime the work might remain incomplete which is waste of time.If paper of the bank is lost,you may also lose the right of you bank properties.

1.3Proposed solution

All banking activities are done using computer system.Banking process will shorten the time of creating an account, depositing , withdrawing and the deletion of account.Thus, the above operations they are more productive than the previous method of banking.Convenience,fewer bank visits,faster transactions.Uniform platform.Paperless banking.Less crowding at branch counters.Bulk data can be stored easily by the use of database systems.Software can be easily understood by the user with the help of GUI,necessary warnings and guildlies.The Aadhaar platform helps service providers authenticate identity of residents electronically, in a safe and quick manner, making service delivery more cost effective and efficient.

CHAPTER 2

Analysis and System Requirements

2.1 Literature Survey

Firstly, we created a database using MySQL[1]. The name of the database is banking. We created five tables that is cust_info, amount,deposit,withdraw and passbook [2]. The front end is created using (java)NetBeans [2]. The GUI has a tabbed pane where there are tables for creation of account,deposit,withdraw ,deletion of account and to view the passbook details. The creation of account has different labels with text fields next to them which tells the user clearly where to enter the data. The data that is entered in these field will be retrieved and can be placed accordingly in the backend. We have used trigger for converting the name entered by the user from lower case to upper case [5]. On clicking the submit button the data will be placed in the backend. The coding is done for the buttons. The buttons contain the connectivity code. That is nothing but JDBC code [3]. This connectivity code contains the queries which are declared as string. These queries will be executed in the backend.

Similarly, using the next tabbed plan we can make deposit by giving the aadhar number of the customer. We have coded for the “open” button in the above way. The open button calls a stored procedure [4]. The user can enter the amount to be deposited and submit it. This will update the tables in the backend. Similarly, the same “open” is present in the withdraw,passbook and delete account tab.

On clicking submit button in withdraw tab it will update the tables in the backend. In passbook tab we have used a text Area to display the details [6]. It is a string builder function that will append the data retrieved from the backend.

In delete account tab the open button will display the first name,middle name and last name along with the current balance of the customer. On clicking the “delete” button the particular account will be deleted from the database. This is because we have used on delete cascade condition while creating the table [1].

2.2 Hardware & Software Requirements

Hardware requirements:

The hardware requirements are very minimal and the program can be run on most of the machines.

Processor: intel core i3

Processor speed: 2.0 Ghz

RAM:4 GB

Storage space: 1 GB

Resolution: 1204*768

Software requirements:

Operating System: windows 10

IDE: NetBeans 8.2

Database:MySql

CHAPTER 3

System Design and Modeling

3.1 Preliminary Design:

A logical data model should be used as the blueprint for designing and creating a physical database. But the physical database cannot be created properly with a simple logical to physical mapping. Many physical design decisions need to be made by the DBA before implementing physical database structures. This may necessitate deviating from the logical data model. But such deviation should occur only based on in-depth knowledge of the DBMS and the physical environment in which the database will exist.

The aadhar number attribute in the cust_info table is made a primary key and the aadhar number attribute in the other tables is made as foreign key's. After deleting a tuple from a table in order to delete it from other tables, we have used on delete cascade.

The front end of this project is java(NetBeans). The GUI consists of tabbed panes like the create an account, deposit, withdrawal, deletion of the users account. The queries tab includes data about the particular query. The front end and the back end is connected using the jdbc(java database connectivity).

Trigger is being used in this project for converting the alphabets from lower case to upper case which includes the first name, middle name and last name attributes. This trigger is automatically being activated when the above mentioned attributes are provided with input in lower case.

Stored procedure is being used in this project to open a particular account of the corresponding aadhar number being provided by the customer. This stored procedure is used to open an account in the deposit tabbed pane.

3.1.1 E R diagram

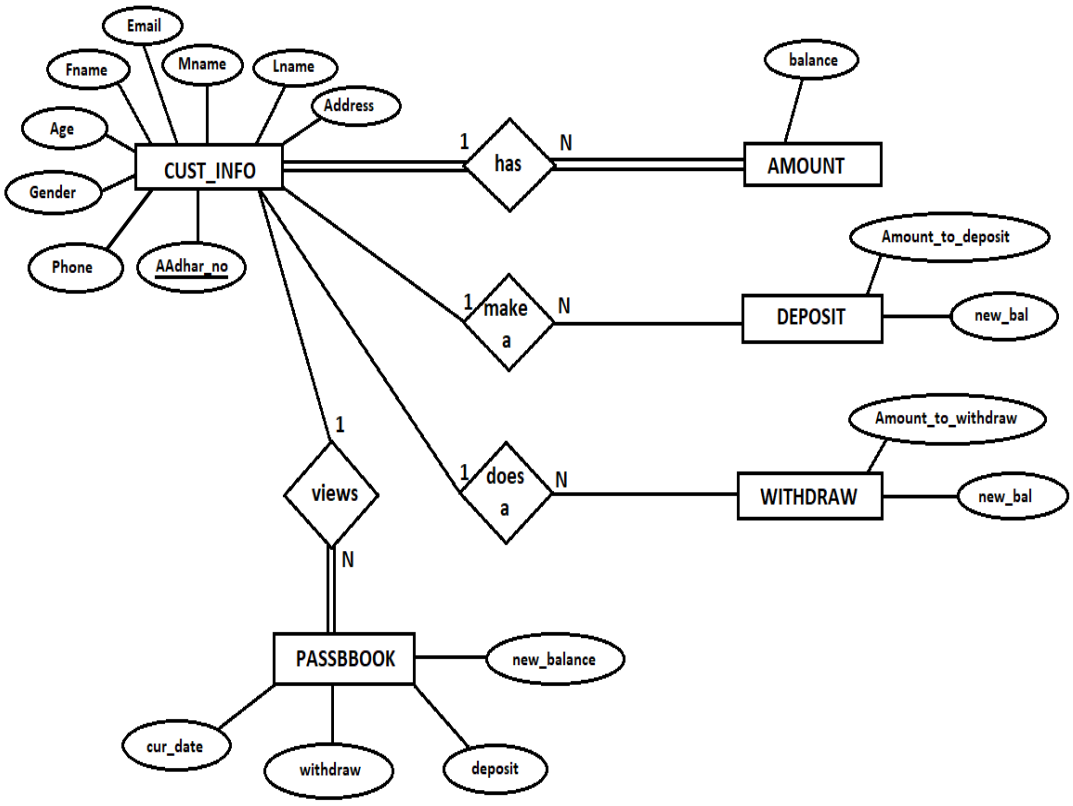


Figure 3.1 ER Diagram of Bank Management System

There are five entities in the ER diagram. The entities are Customer information, Amount, Withdraw, Deposit, Passbook as shown in figure 3.1

Customer information has attributes like First name, Middle name, Last name, Age, Address, Gender, Email, Phone and Aadhar number (primary key).

Amount has attributes like and balance.

Withdraw has attributes like Amount to be withdrawn and New balance.

Deposit has attributes like Amount to be deposit, and New balance.

Passbook has attributes like current date, withdraw, deposit and New balance.

Customer information is related to amount with 1:n cardinality ratio.

Customer information is related to deposit with 1:n cardinality ratio.

Customer information is related to withdraw with 1:n cardinality ratio.

Customer information is related to passbook with 1:n cardinality ratio.

Customer information table is totally dependent on amount table.

Every customer will have minimum of one passbook entry.

3.1.2 Schema diagram

There are five tables in the Schema diagram. The tables are Customer information, Amount, Withdraw, Deposit, Passbook as shown in Figure 3.2.

Customer information has attributes like First name, Middle name, Last name, Age, Address, Gender, Email, Phone and Aadhaar number (primary key).

Amount has attributes like Aadhar number (foreign key) and balance.

Withdraw has attributes like Aadhar number (foreign key), Amount to be withdrawn and New balance.

Deposit has attributes like Aadhar number (foreign key), Amount to be deposit, and New balance.

Passbook has attributes like current date, Aadhar number (foreign key), withdraw, deposit and New balance.

After deleting a tuple from a table in order to delete it from other tables, we have used on delete cascade.

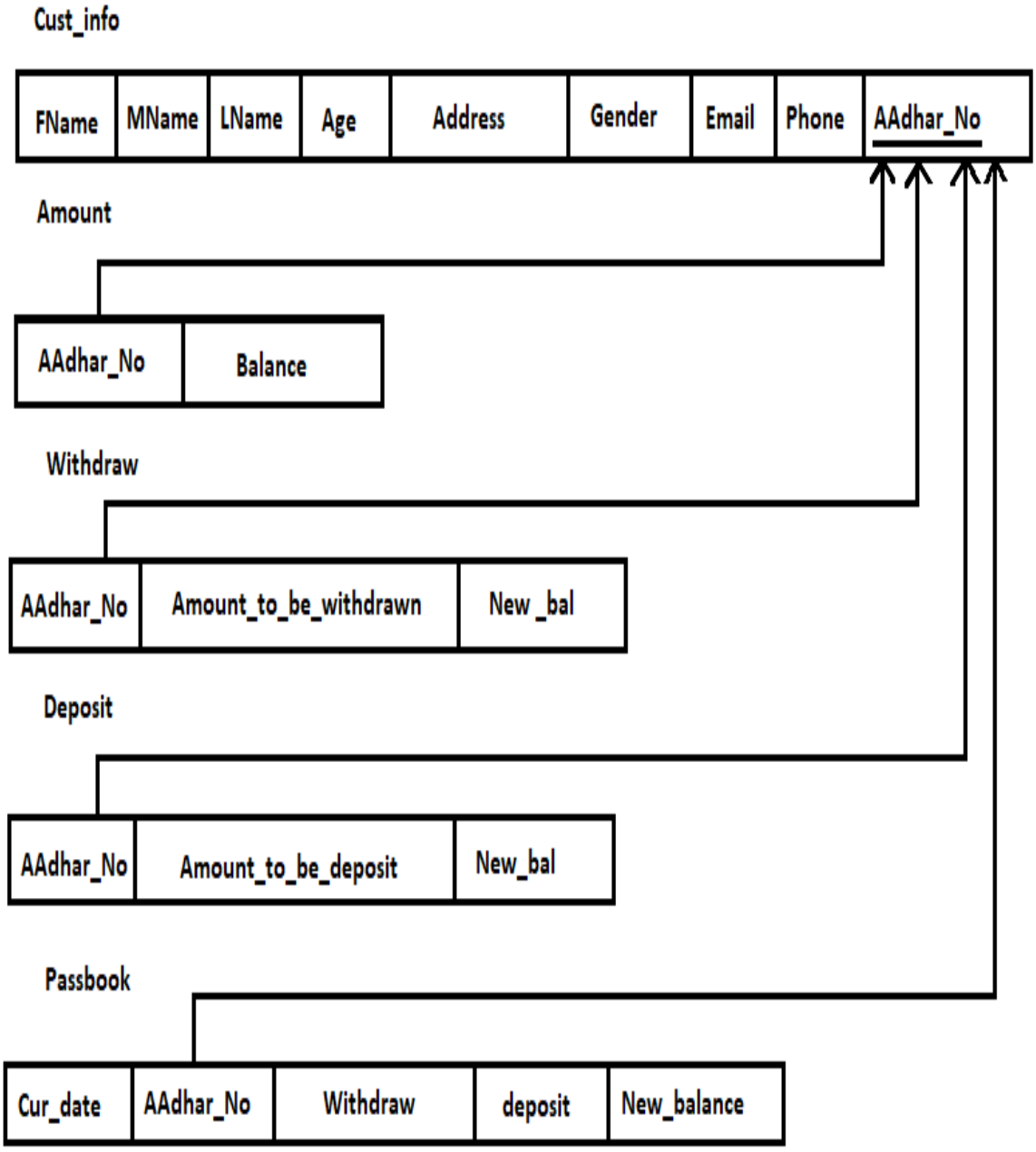


Figure 3.2 Schema Diagram of Bank Management System

3.2 Normalization

Normalization is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly. Let's discuss about anomalies first then we will discuss normal forms with examples.

Anomalies in DBMS

There are three types of anomalies that occur when the database is not normalized. These are – Insertion, update and deletion anomaly.

3.2.1. First normal form (1NF)

As per the rule of first normal form, an attribute (column) of a table cannot hold multiple values. It should hold only atomic values.

Cust_info

Table 3.2 First Normal Form

FName	MName	LName	Age	Address	Gender	Email	Phone	AAdhar_No

The above table is in first normal form.

3.2.2. Second normal form (2NF)

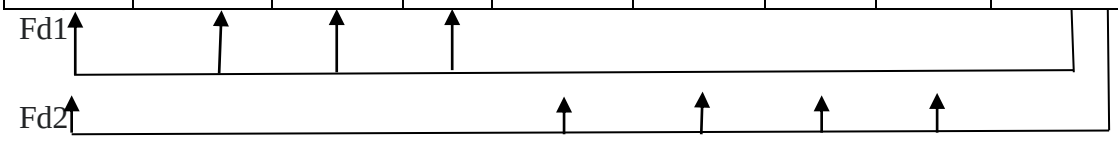
A table is said to be in 2NF if both the following conditions hold:

- Table is in 1NF (First normal form)
- No non-prime attribute is dependent on the proper subset of any candidate key of table.

An attribute that is not part of any candidate key is known as non-prime attribute.

Table 3.3 Second Normal Form

FName	MName	LName	Age	Address	Gender	Email	Phone	AAdhar_No



FD1

AAdhar_No	FName	MName	LName	Age

FD2

AAdhar_No	FName	Address	Gender	Email	Phone

3.2.3.Third Normal form (3NF)

A table design is said to be in 3NF if both the following conditions hold:

- Table must be in 2NF
- Transitive functional dependency of non-prime attribute on any super key should be removed.
- An attribute that is not part of any candidate key is known as non-prime attribute.

Cust_info

Table 3.4 Third Normal Form

FName	MName	LName	Age	Address	Gender	Email	Phone	AAdhar_No

The above table is in Third normal form.

CHAPTER 4

Implementation

4.1 Implementation

Operations:

1. **Creating an account:**Make sure you're eligible to open an account.Visit the bank and ask to open an account.Ask important questions before you finalize your account.Supply the necessary information to create your account.
2. **Deposit:** When you want to put more money into your bank accounts you need to make a deposit.This is by provide your account numbers (or aadhar number).
3. **Withdrawal of money:** You will usually need your account number (or aadhar number)
4. **Deletion of an account:**Look up your bank account closing procedure.Request verification that your account is closed and collect your cash.
5. **Passbook:**A passbook or bankbook is a paper book used to record bank transactions on a deposit account and withdraw account.

SQL statements:

Insert statement:The INSERT INTO statement is used to insert new records in a table.The INSERT INTO syntax would be as follows:INSERT INTO *table_name*VALUES (*value1*, *value2*, *value3*, ...);

The following SQL statement insert's a new record in the "Cust_info" table:Insert into cust_info VALUES('sahana','shreedhar','kulkarni',20,'bangalore','female','sahana271196@yahoo.co.in','9972492202',12345);

Update statement: An SQL UPDATE statement changes the data of one or more records in a table. Either all the rows can be updated, or a subset may be chosen using a condition.

The UPDATE syntax would be as follows:

UPDATE *table_name* SET *column_name = value* [, *column_name = value* ...]

[WHERE *condition*]The following SQL statement update's a record in the "amount" table:

UPDATE amount set balance=balance+1000 where aadhar_no=12345;

Delete statement:The DELETE statement is used to delete existing records in a table.The

DELETE syntax would be as follows:DELETE FROM *table_name*WHERE *condition*;

The following SQL statement delete's a record in the "cust_info" table:

delete from cust_info where aadhar_no=12345;

Create statement: The CREATE TABLE Statement is used tocreate tables to store data. Integrity Constraints like primary key, unique key, foreign key can be defined for the columns while creating the table. The CREATE syntax would be as follows:

CREATETABLE table_name(column1datatype,column2 datatype,column3 datatype,....columnN datatype,PRIMARY KEY(one or more columns));

The following SQL statement creates a table "amount":

Create table amount (aadhar_no int(5)),balance float(10,2),foreign key(aadhar_no) references cust_info(aadhar_no) on delete cascade);

Trigger statement:A trigger is a special kind of stored procedure that automatically executes when an event occurs in the database server.The trigger syntax would be as follows:

Create trigger trigger_name before insert or update of <parameters> on <table_name> for each row set <condition>;

The following SQL statement triggers records in the "cust_info" table:

Create trigger inform before insert on cust_info for each row set new.fname=upper(new.fanme);

Stored procedure:A stored procedure is a set of SQLstatements with an assigned name, which are stored in a RDBMS as a group, so it can be reused and shared by multiple programs.The stored procedure syntax would be as follows:Create procedure procedure_name(input parameters,output parameters)as begin <sql statement used in stored procedure>end

The following SQL statement creates a procedure:

Delimiter //

Create procedure proc_dep(IN aadhar INT)begin select balance from amount where aadhar_no=12345;

End//

Java:

Java is ageneral-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere".Java applications are typically compiled to bytecode that can run on any JVM regardless of computer architecture.

4.1.1 Explanation with pseudo code /flowchart

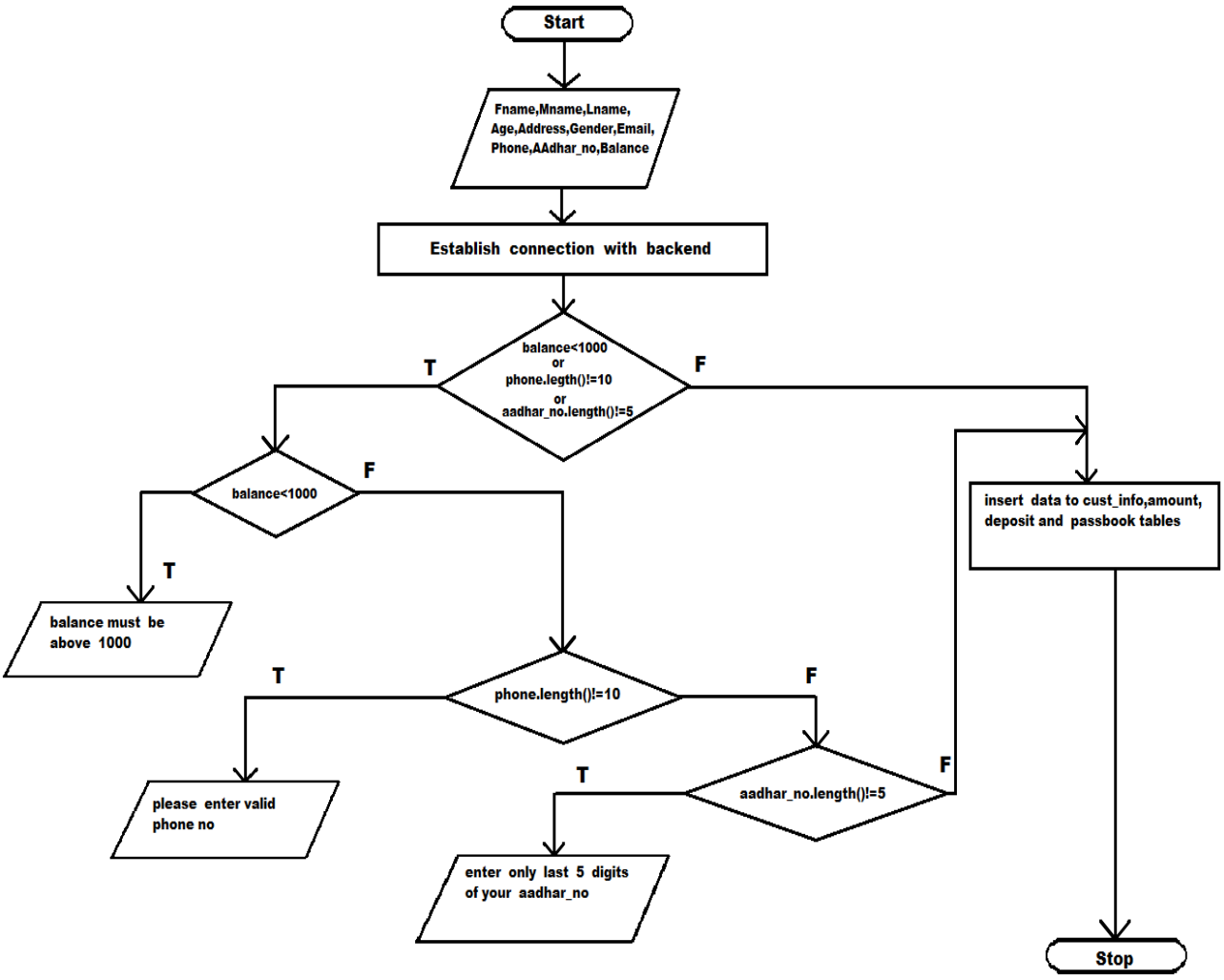


Figure 4.1 Flowchart of Bank Management System

4.1.2 Module1 Description

Algorithm to Create and show Trigger in SQL Server

1.START

2.CREATE TRIGGER inform.

3.{BEFORE AND INSERT}.

4.[ON CUST_INFO].

5. [FOR EACH ROW].

6.[SET NEW.FNAME=UPPER (NEW.FNAME), NEW.MNAME=UPPER (NEW.MNAME), NEW.LNAME=UPPER (NEW.LNAME)]

7.SHOW TRIGGERS

8.END

4.1.3 Module2 Description

Algorithm To Create s stored Procedure

1.START

2.1. Create procedure in MySQL

2.2. Declare DELIMITER //

2.3. CREATE PROCEDURE proc_dep3

2.4. Pass parameter (IN aadhar INT)

2.5. BEGIN

2.6. SELECT BALANCE FROM AMOUNT WHERE AADHAR_NO=aadhar;

2.7.END//

3.END

CHAPTER 5

TESTING

This chapter gives the outline of all the testing methods that are carried out to get a bug free application. Quality can be achieved by testing the product using different techniques at different phases of the project development.

5.1 Testing process

Testing is an integral part of software development. Testing process, in a way certifies, whether the product, that is developed, compiles with the standards, that it was designed to. Testing process involves building of test cases, against which, the product has to be tested. In some cases, test cases are done based on the system requirements specified for the product/software, which is to be developed.

5.2 Testing objectives

The main objectives of testing process are as follows:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probability of finding an as yet undiscovered error.
- A successful test is one that uncovers an as yet undiscovered error.

5.3 Levels of Testing

Different levels of testing are used in the testing process; each level of testing aims to test different aspects of the system. The basic levels are unit testing, integration testing, system testing and acceptance testing.

5.3.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design the module. The software built, is a collection of individual modules.

In this kind of testing exact flow of control for each module was verified. With detailed design consideration used as a guide, important control paths are tested to uncover errors within the boundary of the module.

Table 5.1:Negative test case for create account

Function Name	Input	Expected Output	Error	Resolved
Create account	Sahana123 as first name	Must take only sahana as input	Numbers are being taken as input for name	consume()

Table 5.2:Positive test case for create account

Function Name	Input	Expected Output	Error	Resolved
Create account	Sahana123 as first name	Expected output is seen	-	-

Table 5.3:Negative test case for input phone number

Function Name	Input	Expected Output	Error	Resolved
Input phone number	9663487888abc	Must take only 9663487888 as input	alphabets are being taken as input for phone number	consume()

Table 5.4:Positive test case for input phone number

Function Name	Input	Expected Output	Error	Resolved
Input phone number	9663487888abc	Expected output is seen	-	-

Table 5.5:Negative test case for initial balance

Function Name	Input	Expected Output	Error	Resolved
Input initial balance	100	Must take only 1000 and above	100 is also being taken as an input	if(balance<1000) then display a dialog box that balance must be above 1000

Table 5.6:Positive test case for initial balance

Function Name	Input	Expected Output	Error	Resolved
Input initial balance	100	Balance must be above Rs.1000	-	-

5.3.2 Integration testing

The second level of testing is called integration testing. In this, many class-tested modules arecombined into subsystems, which are then tested. The goal here is to see if all the modules can be integrated properly. We have are identified and debugged.

Table 5.7: Test case on basics of creation of account and transactions

Function Name	Input	Expected Output	Error	Resolved
Negative-Checking for account using aadhar number	99998	Must display that the aadhar number is not valid	Output not seen	proc_dep(aadhar_no)
Positive-Checking for account using aadhar number	12345	Must display that the current balance	-	-

5.3.3 System testing

Here the entire application is tested. the reference document for this process is the requirement document, and the goal is to see IF the application meets its requirements.

Each module and component of ethereal was thoroughly tested to remove bugs through a system testing strategy. Test cases were generated for all possible input sequences and the output was verified for its correctness.

Table 5.8: Test cases for the project

Steps	Action	Expected output
Step1: **choice**	The screen appears when the users runs the program	A page with different tabbed panes appears.
Step2: **Selection1**	The screen appears when the user selects any one of the tabbed panes from the click of the mouse 1.create an account 2.deposit 3.withdrawal 4.passbook 5.delete account 6.queries	A window for filling the application of the customer with option '1',a window for depositing and withdrawing of the respective account with option '2' and '3' and a window for displaying the transactions of the account with option '4' and option '5' for deleting the particular account . Option '6' for displaying output of the queries
Step3: **Selection2**	The screen appears when the user enters the respective aadhar_no of his account 1.deposit 2.withdrawal 3.passbook 4.delete account	The label that shows the current balance of the respective aadhar_no

CHAPTER 6

CONCLUSION

In this application the administrator of the bank will maintain the records of the bank. Using this application, the administrator will successfully add the account information like first name, middle name, last name, age, address, gender, email, phone and aadhar_no from the customer successfully. The administrator can make the deposits and withdrawal from the customer's account using aadhar_no. If the customer wants to view his transaction details, then he can access his passbook details accordingly. All customer bank transactions are being linked to the aadhar_no. the administrator can also delete the customer's account on the respective customer demand. The data which is being entered through the front end(java) will be store into the backend(MySQL). The frontend and the backend is successfully being connected using jdbc. It contains the stored procedure that displays the current balance of the customer when he mentions his aadhar_no. The project successfully checks the initial balance which should not be lesser than RS.1000. The user is instructed with dialog boxes when any invalid entry or successful modifications is being made by the user.

REFERENCES

[1] http://iips.icci.edu.iq/images/exam/databases-ramaz.pdf

[2]http://www.kviitkanpur.org/admin/downloads/1401126064ncert_informatics_practices_class_xi.pdf

[3] http://www.cbseportal.com/download/text-books/informatics-practices-class-xii.pdf

[4]https://stackoverflow.com/questions/15786240/mysql-create-stored-procedure-syntax-with-delimiter.html

[5] https://stackoverflow.com/questions/15975877/mysql-trigger-on-insert-update-events.html

[6]https://stackoverflow.com/questions/12216494/how-to-add-text-to-a-textarea-instead-of-replacing-it.html

Appendix A: Snapshots

Figure A.1:Create account

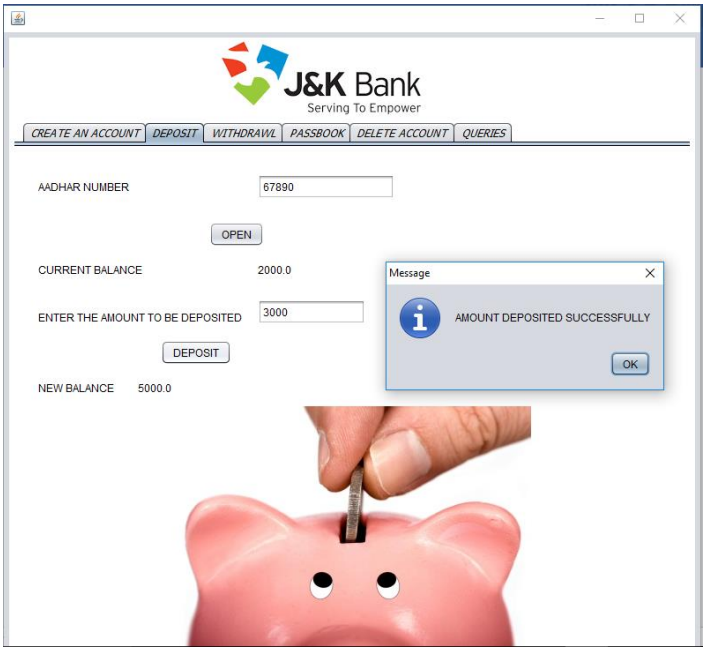


Figure A.2:Deposit

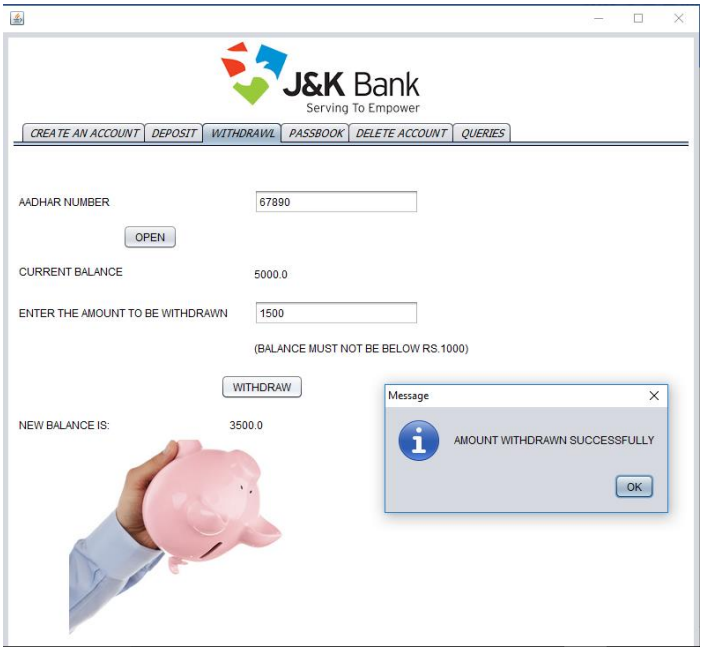


Figure A.3:Withdraw

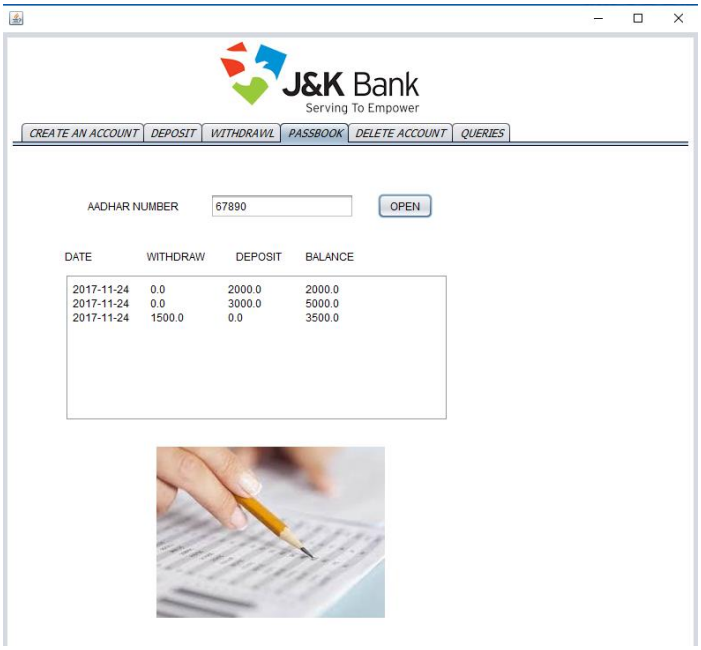


Figure A.4:Passbook

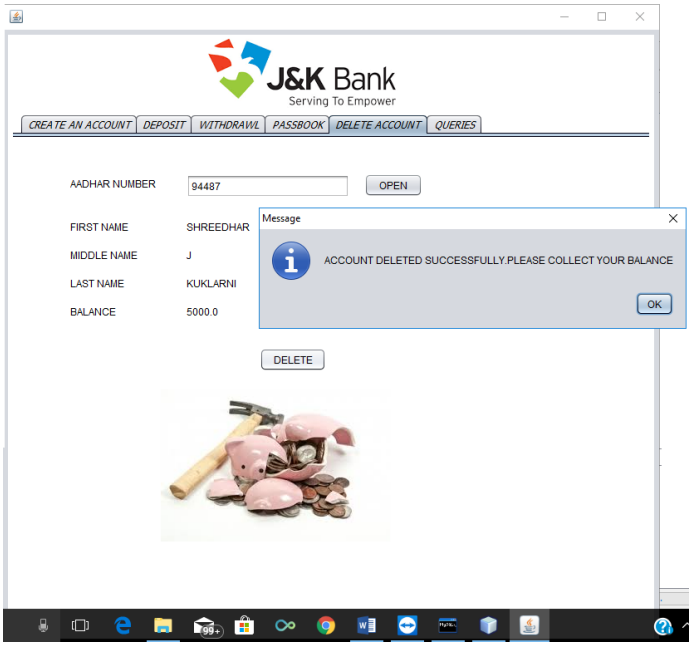


Figure A.5:Delete account

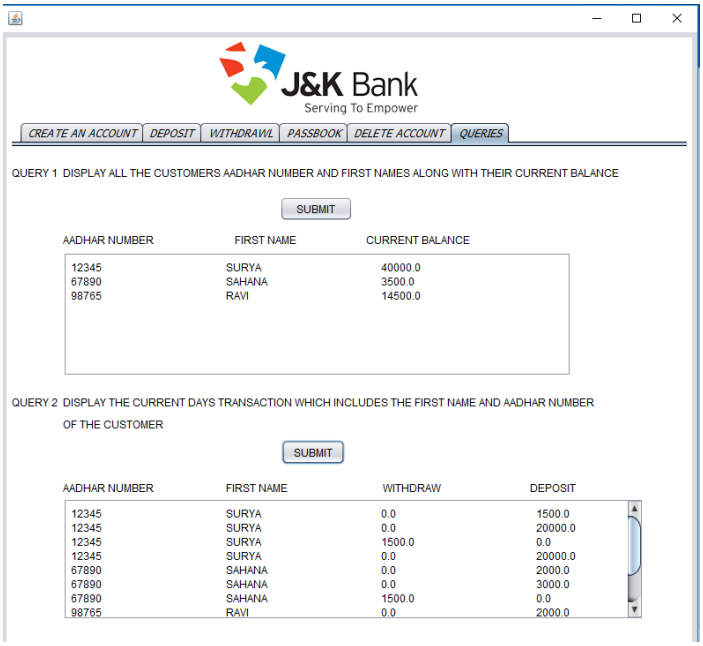


Figure A.6:Queries

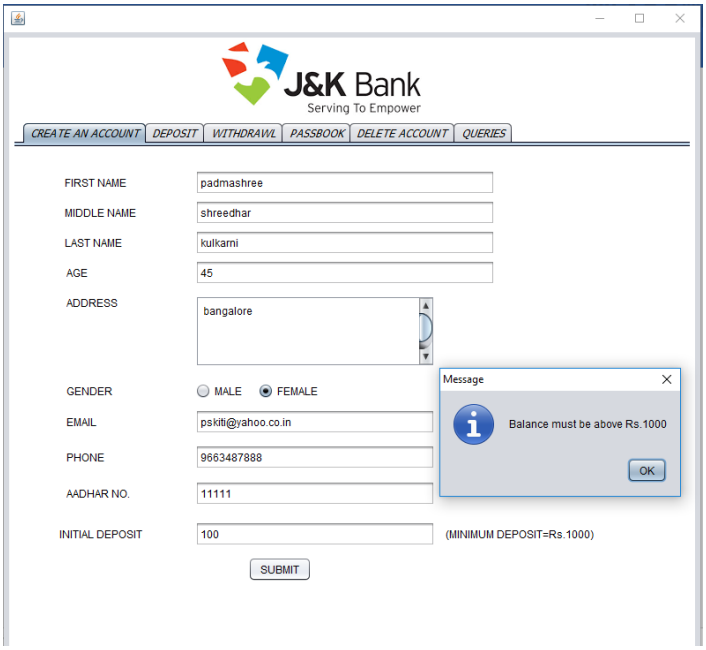


Figure A.7:Error for initial deposit

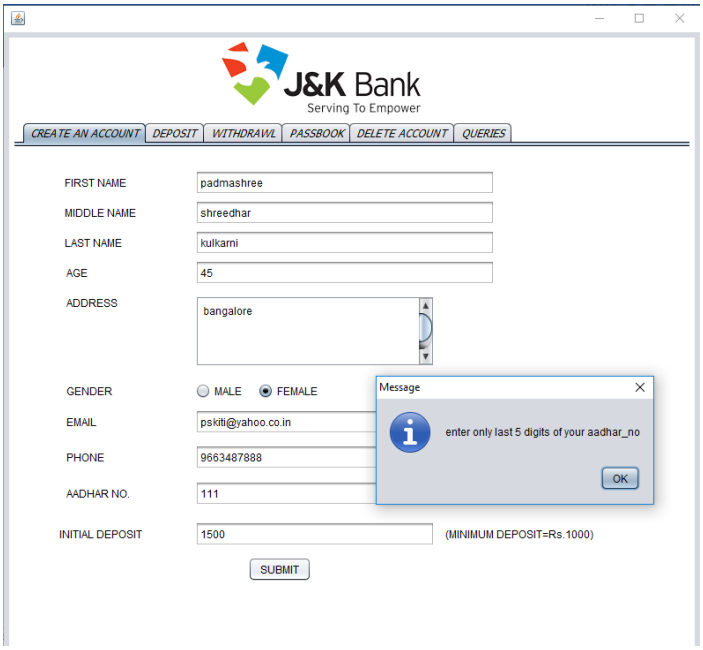


Figure A.8:Error for aadhar number

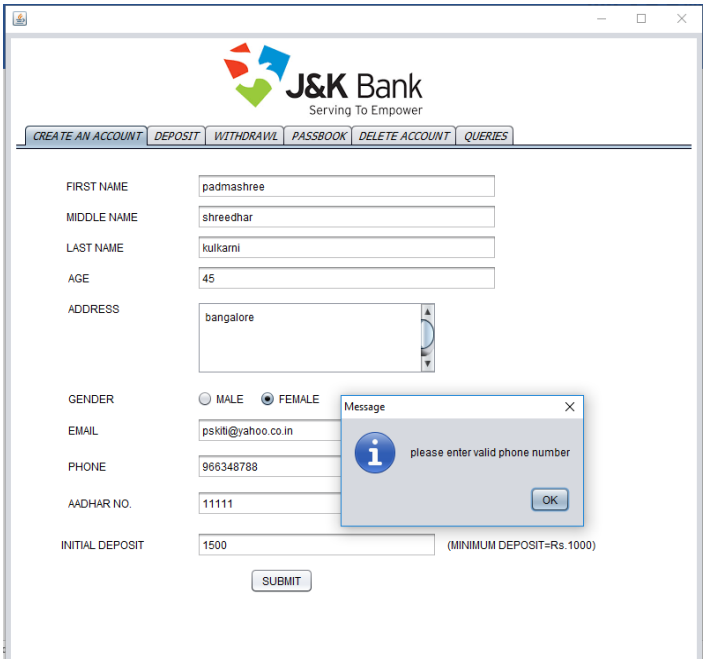


Figure A.9:Error for phone number

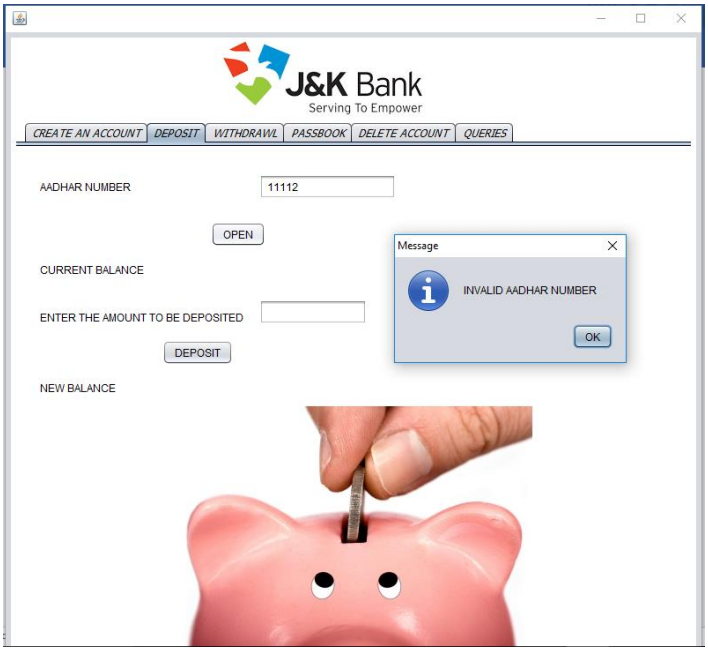


Figure A.10:Error for aadhar number in transaction