PROJECT REPORT
ON
"BANK MANAGEMENT SYSTEM"

Final year project has been accepted in partial fulfilment of the requirements for the degree of Bachelor's of Computer Application (Semester VI)

SUBMITTED BY

ROHAN RAJ GUPTA

UNIVERSITY ROLL NO: NSU2103112

Session: 2021-2024 (BCA)

Under the guidance of

L.K SIR

RITESH SIR

DEEP SIR

DEPTT: IT DEPARTMENT

SUBMITTED TO

NETAJI SUBHAS UNIVERSITY

DEPARTMENT OF COMPUTER APPLICATION, JAMSHEDPUR

POKAHRI,NEAR BHILAI PAHADI JAMSHEDPUR

CERTIFICATE

THIS IS TO CERTIFY THAT MR. ROHAN RAJ GUPTA BEARING ROLL NO. NSU2103112 HAVE DEVELOPED A SOFTWARE PROJECT TITLED "BANK MANAGEMENT SYSTEM" FOR THEIR FINAL YEAR PROJECT AS A PARTIAL FULFILLMENT FOR THE AWARD OF THE DEGREE OF BACHELOR'S OF COMPUTER APPLICATION.

EXTERNAL

HEAD OF DEPARTMENT
DEPARTMENT OF IT & COMPUTER SCIENCE

TABLE OF CONTENTS [TOC]

ACKNOWLEDGEMENT

This Project report was completed as a result of support from many people, although not all of them can be mentioned.

We wish to express our sincere gratitude to God for his protection, providence, guidance and above all, for sustaining us.

We are greatly indebted to our good supervisor L.K Sir & Ritesh Sir for his useful and necessary observation, suggestions, contribution and corrections. We would not have been able to achieve anything in this research without your supervision. May God enrich you greatly in every area of life.

Finally, we wish to express our appreciation to our parents for their love and support.

Rohan Raj Gupta

Introduction

The “Bank Account Management System” project is a model Internet Banking Site. This site enables the customers to perform the basic banking transactions by sitting at their office or at homes through PC or laptop. The system provides the access to the customer to create an account, deposit/withdraw the cash from his account, also to view reports of all accounts present. The customers can access the banks website for viewing their Account details and perform the transactions on account as per their requirements. With Internet Banking, the brick and mortar structure of the traditional banking gets converted into a click and portal model, thereby giving a concept of virtual banking a real shape. Thus, today's banking is no longer confined to branches. E-banking facilitates banking transactions by customers round the clock globally.

The primary aim of this “Bank Account Management System” is to provide an improved design methodology, which envisages the future expansion, and modification, which is necessary for a core sector like banking. This necessitates the design to be expandable and modifiable and so a modular approach is used in developing the application software.

Anybody who is an Account holder in this bank can become a member of Bank Account
Management System. He has to fill a form with his personal details and Account Number.
Bank is the place where customers feel the sense of safety for their property. In the bank, customers deposit and withdraw their money. Transaction of money also is a part where customer takes shelter of the bank. Now to keep the belief and trust of customers, there is the positive need for management of the bank, which can handle all this with comfort and ease.
Smooth and efficient management affects the satisfaction of the customers and staff members, indirectly. And of course, it encourages management committee in taking some needed decision for future enhancement of the bank.

Now a day’s, managing a bank is tedious job up to certain limit. So software that reduces the work is essential. Also, today’s world is a genuine computer world and is getting faster and faster day-by-day. Thus, considering above necessities, the software for bank management has become necessary which would be useful in managing the bank more efficiently.

All transactions are carried out online by transferring from accounts in the same Bank or international bank. The software is meant to overcome the drawbacks of the manual system.

Abstract

The Bank Account Management System is an application for maintaining a person's account in a bank. In this project I tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System. To develop a project for solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks. Also, to enable the user's work space to have additional functionalities which are not provided under a conventional banking project.

The Bank Account Management System undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this software. This project is developed using Java language. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting needs of the customer, while proving compliance and staying on the schedule and within budget.

The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is then implemented with Java. The system is designed as an interactive and content management system. The content management system deals with data entry, validation confirm and updating whiles the interactive system deals with system interaction with the administration and users. Thus, above features of this project will save transaction time and therefore increase the efficiency of the system

AIM of this project

The main aim of designing and developing this Internet banking System Java Core (Swing & AWT) primarily based Engineering project is to provide secure and efficient net banking facilities to the banking customers over the internet. IntelliJ IDEA Community Edition 2024.1.2, MYSQL Workbench database used to develop this bank application where all banking customers can login through the secured web page by their account login id and password. Users will have all options and features in that application like get money from western union, money transfer to others, and send cash or money to inter banking as well as other banking customers by simply adding them as payees.

Main Purpose

The Traditional way of maintaining details of a user in a bank was to enter the details and record them. Every time the user needs to perform some transactions he has to go to bank and perform the necessary actions, which may not be so feasible all the time. It may be a hard-hitting task for the users and the bankers too. The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain. Here, we provide automation for banking system through Internet. Online Banking System project captures activities performed by different roles in real life banking which provides enhanced techniques for maintaining the required information up-to-date, which results in efficiency. The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain

Main Goal

1. Motto: Our motto is to develop a software program for managing the entire bank process related to Administration accounts customer accounts and to keep each every track about their property and their various transaction processes efficiently. Hereby, our main objective is the customer's satisfaction considering today's faster in the world.

2. Customer Satisfaction: Client can do his operations comfortably without any risk or losing of his privacy. Our software will perform and fulfill all the tasks that any customer would desire.

3. Saving Customer Time: Client doesn't need to go to the bank to do small operation.

4. Protecting the Customer: It helps the customer to be satisfied and comfortable in his choices, this protection contains customer's account, money and his privacy.

5. Transferring Money: Help client transferring money to/or another bank or country.

Modules Description

The Modules description of Bank Account Management System project. These modules will be developed in PHP source code and MYSQL database.

1. **Create New Account:** A customer who having the account in the world can create a virtual account through this module. This module receives the customer profile details and the bank account details with the proof of the ownership of the bank account.
2. **Login:** Virtual account holders can login in to the system using this module. Thus this is the secured login page for the customers in the website.
3. **Virtual Account:** After the approval of new virtual account creation, the customer assigned a unique virtual account number to make the online money transactions. This module views the details of the logged customer's virtual account.
4. **Bank Accounts:** A customer may have more than one bank account in various banks, in this case, the customer prompted to

decide which bank account should reflect in the account debit or amount credit. For these operations customers can add their owned bank accounts here and it will be approved by the administrations of the system.

5. **Fund Transfer:** This is the module to make fund transfer to the virtual bank account holders or the usual bank account holders from the customer's specified bank account.

6. **Beneficiary:** Beneficiary is a person who receives money. Here the customer can add the beneficiaries to make fund transfer in the future.

7. **Transactions:** This module displays the transactions made by the customer in the particular date with the transaction details.

8. **Administrative Control:** This module contains the administrative functions such as view all virtual account, transactions, approve bank accounts, approve virtual accounts etc.

There are other features and actions that can be performed on a back account but we are not going to look at bank accounts in their entirety only the basics, this way we avoid over complicating the exercise. The purpose of this whole exercise is to show the usefulness of object oriented programming as opposed to really wanting to create a banking system.

Translating the above points into software is easy when you think of a bank account as an object:

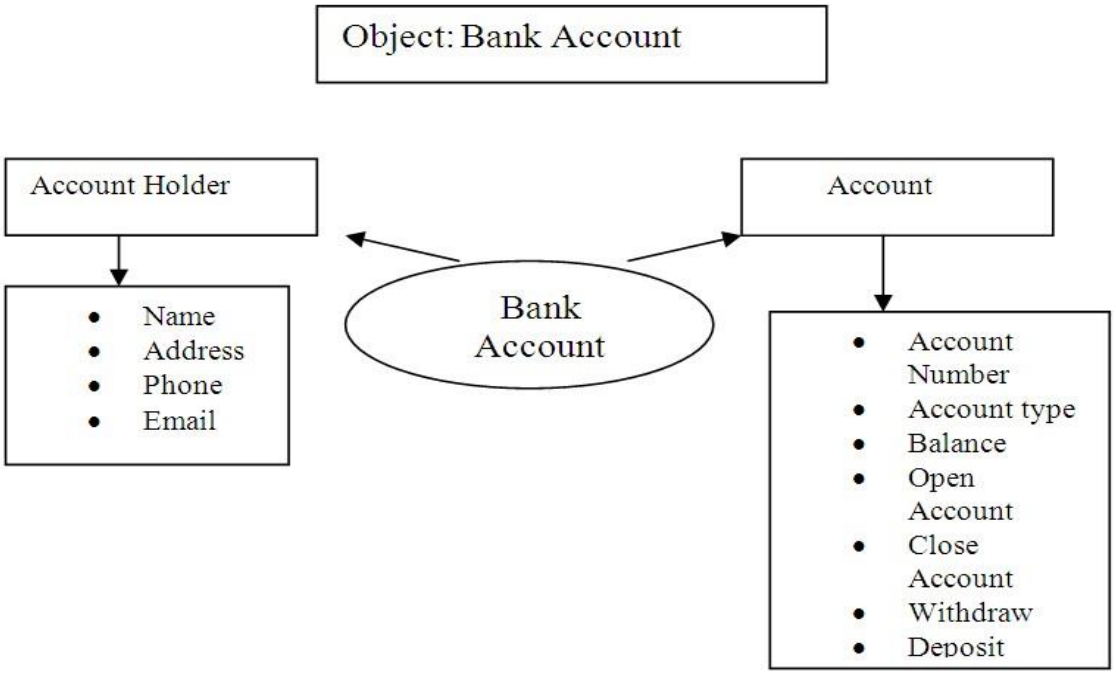


Figure-1: Bank Account System

Just by looking at the above picture, we can work out what methods we need our class to have:

Methods

- We need to be able to generate an account number
- Account types: Savings or Current Account
- Maintain/update Balance
- Open/Close Account
- Withdraw/Deposit

Administrative Modules

Here in my project there are two types of modules. This module is the main module which performs all the main operations in the system. The major operations in the system are:

Admin Module

Admin can access this project there is an authorization process. If you login as an Admin then you will be redirected to the Admin Home Page and if you are a simple user you will be redirected to your Account Home Page. This performs the following functions: Create
Individual Accounts, manage existing accounts, View all transactions, Balance enquiry,
Delete/close account etc.

1- Admin login
2- Add/delete/update account
3- Withdrawal/deposit/statements transaction
4- Account Information
5- User details list
6- Active/Inactive account
7- View transaction histories

User Module

A simple user can access their account and can deposit/withdraw money from their account.
User can also transfer money from their account to any other bank account. User can see their transaction report and balance enquiry too.

1- User login, use PIN system
2- Creating/open new account registration
3- Funds transfer (local/international/domestic)
4- View statements transaction
5- User account details
6- Change Password and Pin

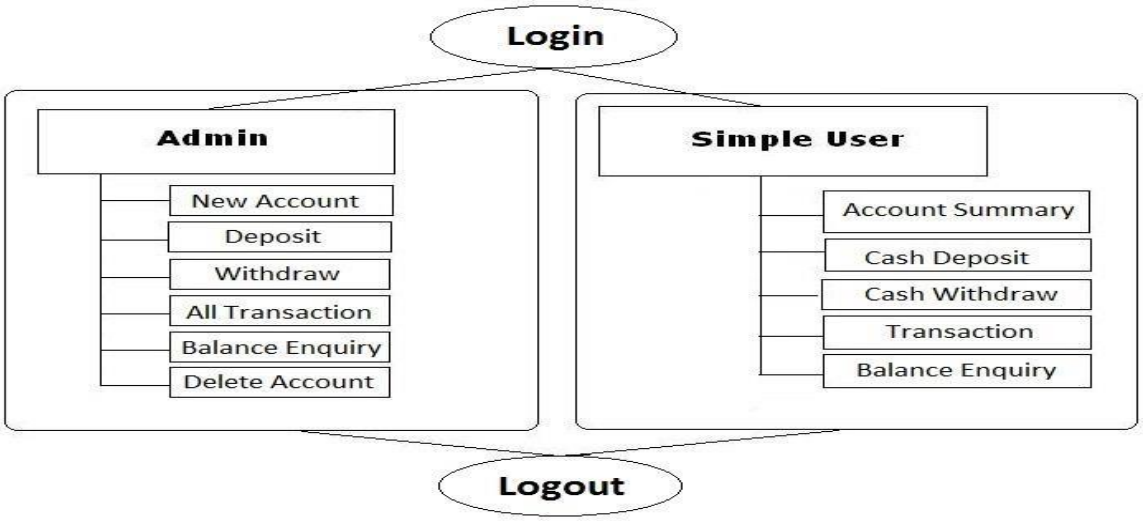


Figure-2: Module of project

Hardware Requirements Specification

Processor	:Intel Core i5
Main Memory(RAM)	: 512 GB
Cache Memory	: 512 KB
Monitor	: 14 inch Color Monitor
Keyboard	: 108 Keys
Mouse	: Optical Mouse
Hard Disk	: 160 GB

Software Requirements Specification

Front End/Language	: Core Java (Swing & AWT)
Back End/Database	: MySQL Workbench
Additional Tools	: IntelliJ IDEA Community Edition 2024.1.2
Operating System	: Windows 7, 8, 9, 10, XP

System Design

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term "design" is defined as "the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization". It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user oriented document to a document to the programmers or database personnel.

System design goes through two phases of development:

- ✓ Logical Design and
- ✓ Physical Design.

Logical Design

The logical flow of a system and define the boundaries of a system. It includes the following steps:

- ✓ Reviews the current physical system – its data flows, file content, volumes, frequencies etc.

- ✓ Prepares output specifications – that is, determines the format, content and frequency of reports.
- ✓ Prepares input specifications – format, content and most of the input functions.
- ✓ Prepares edit, security and control specifications.
- ✓ Specifies the implementation plan.
- ✓ Prepares a logical design walk through of the information flow, output, input, controls and implementation plan.
- ✓ Reviews benefits, costs, target dates and system constraints.

Physical Design

Physical system produces the working systems by define the design specifications that tell the programmers exactly what the candidate system must do. It includes the following steps.

- ✓ Design the physical system.
- ✓ Specify input and output media.
- ✓ Design the database and specify backup procedures.
- ✓ Design physical information flow through the system and a physical design Walk through.
- ✓ Plan system implementation.
- ✓ Prepare a conversion schedule and target date.
- ✓ Determine training procedures, courses and timetable.
- ✓ Devise a test and implementation plan and specify any new hardware/software.
- ✓ Update benefits, costs, and conversion date and system constraints.

Database design

The database, called a bank, will have two tables, one called accounts and the other called customer. Each will hold information

about either the account or the customer. The two tables will be linked through a foreign key. The customer table has the following fields: **Account User Table-3.1**

Field	Description
cusid	Creates a unique customer id for each new customer
name	Stores the customer name
address	Stores the customer address
acc_id	Links the customer to a account in the accounts table

Accounts Table-3.2

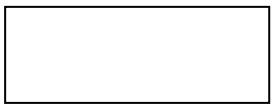
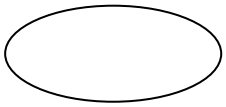
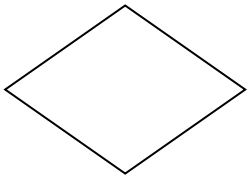
Field	Description
accid	Creates a unique account number for each new account
accno	Stores the account number
type	Stores the account type
balance	Stores the account balance
active	Shows the account status

Since one customer can have many accounts, I thought it only right to insert a foreign key acc_id into the customer table. In addition, instead of having fields such as date created and date closed, I simply use the active field to check if the account is active or not. This will enable us to focus more on the programming than on particulars of the database.

ENTITY-RELATIONSHIP DIAGRAMS

E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

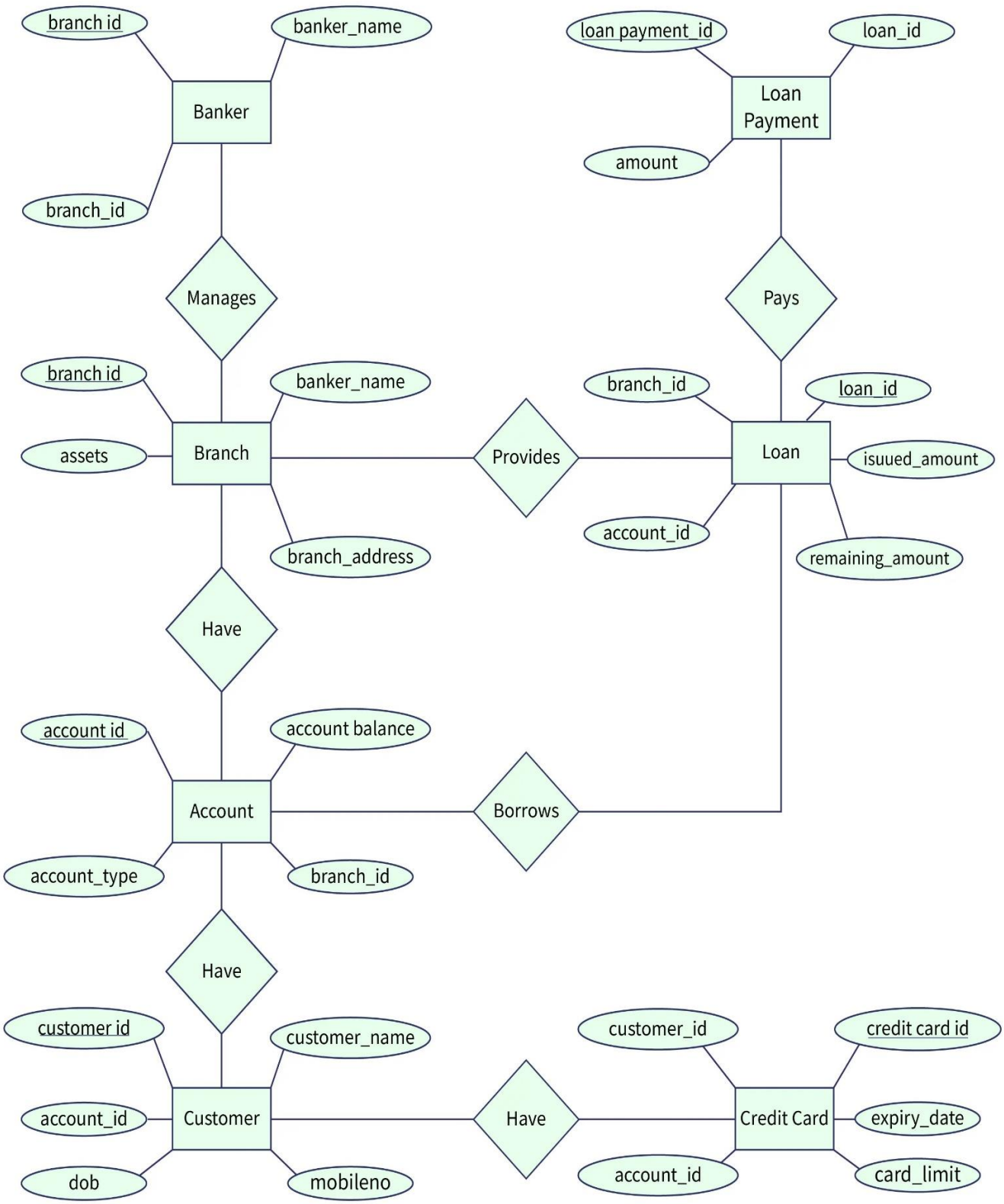
The symbols used in E-R diagrams are:

SYMBOL	PURPOSE
	Represents Entity sets.
	Represent attributes.
	Represent Relationship Sets.
	Line represents flow

Structured analysis is a set of tools and techniques that the analyst.

To develop a new kind of a system:

The traditional approach focuses on the cost benefit and feasibility analysis, Project management, and hardware and software selection a personal considerations.


branch id
banker_name
Banker
branch_id
Manages
loan payment_id
loan_id
Loan Payment
amount
Pays
branch id
banker_name
assets
Branch
branch_address
Provides
branch_id
loan_id
Loan
isuued_amount
account_id
remaining_amount
Have
account id
account balance
Account
account_type
branch_id
Borrows
Have
customer id
customer_name
account_id
Customer
dob
mobileno
Have
customer_id
credit card id
Credit Card
expiry_date
account_id
card_limit

Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

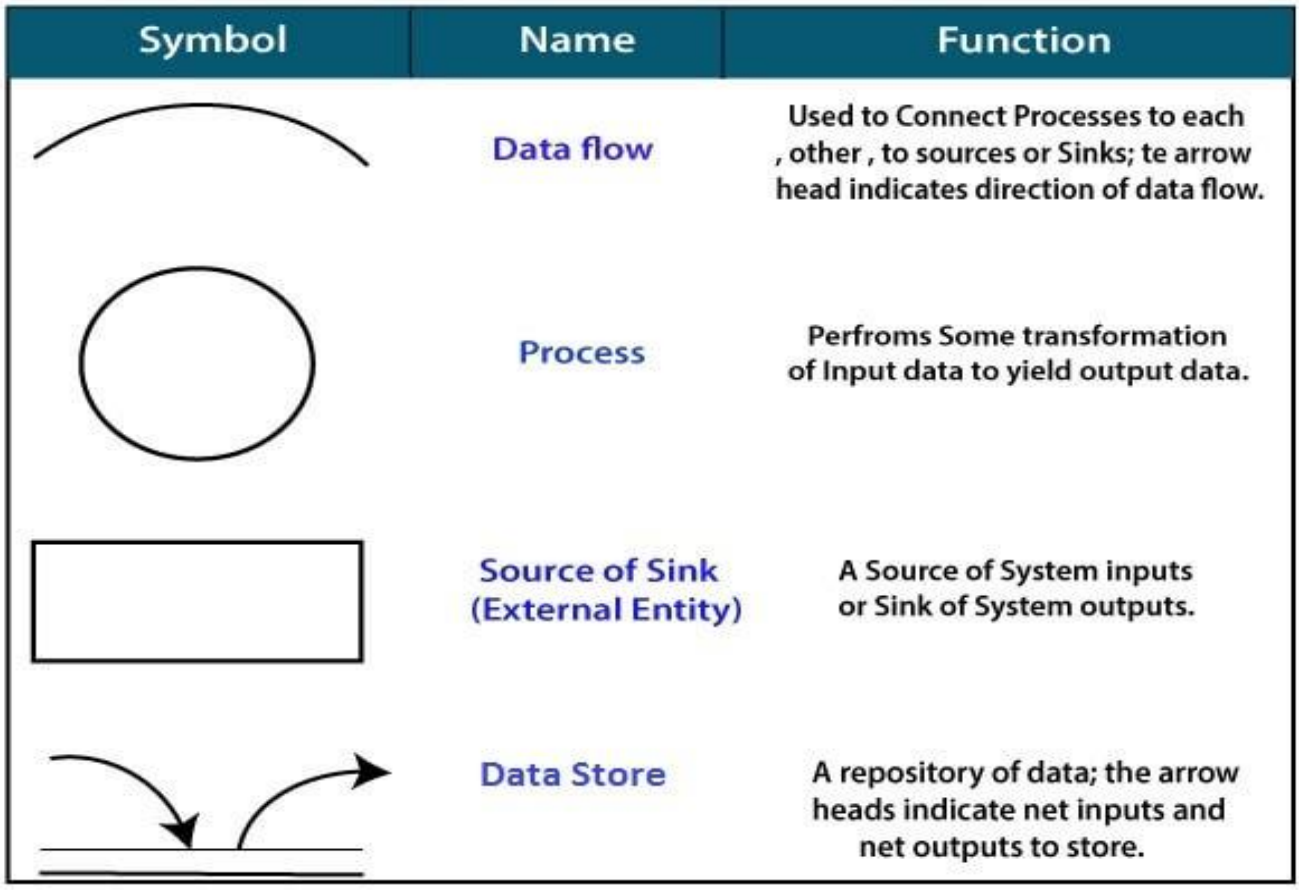
It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

The following observations about DFDs are essential:

1. All names should be unique. This makes it easier to refer to elements in the DFD.
2. Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.
3. Suppress logical decisions. If we ever have the urge to draw a diamond-shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.
4. Do not become bogged down with details. Defer error conditions and error handling until the end of the analysis.

Standard symbols for DFDs are derived from the electric circuit diagram analysis and are shown in fig:



Symbols for Data Flow Diagrams

Circle: A circle (bubble) shows a process that transforms data inputs into data outputs.

Data Flow: A curved line shows the flow of data into or out of a process or data store.

Data Store: A set of parallel lines shows a place for the collection of data items. A data store indicates that the data is stored which can be used at a later stage or by the other processes in a different order. The data store can have an element or group of elements.

Source or Sink: Source or Sink is an external entity and acts as a source of system inputs or sink of system outputs.

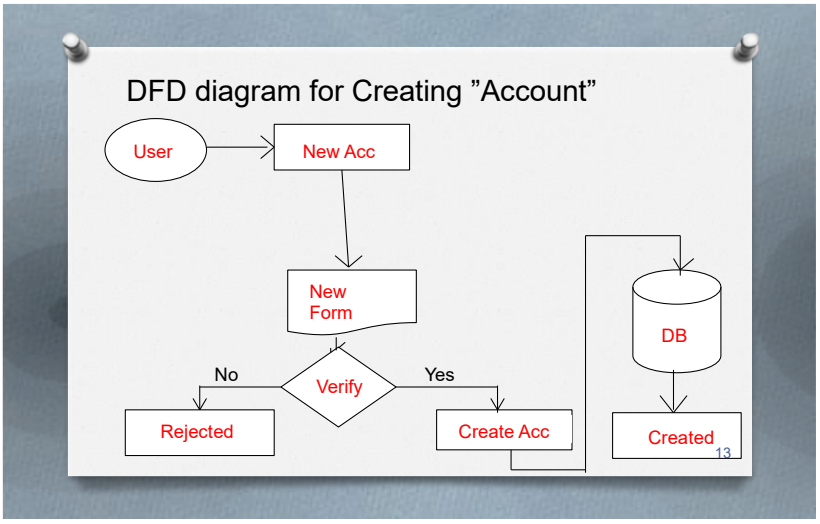


Figure-4.1: Create new account DFD

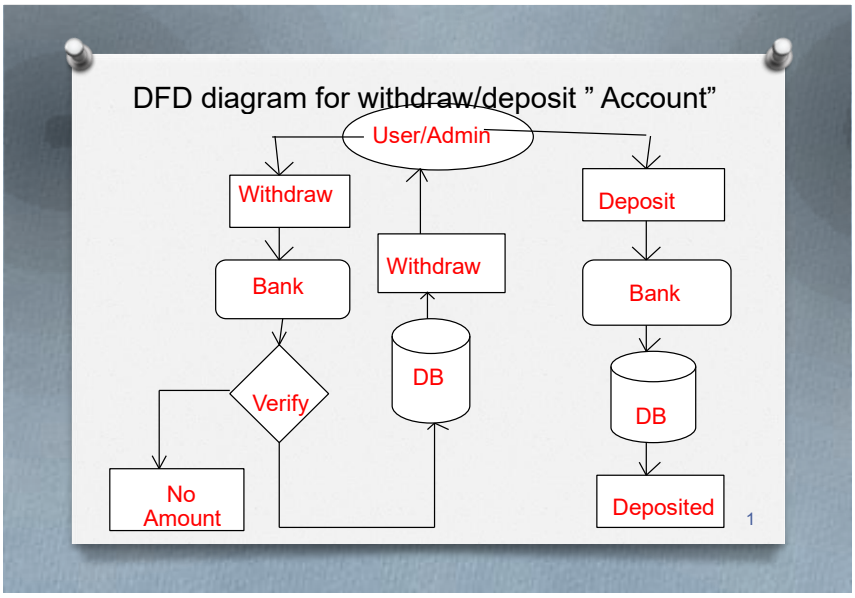


Figure-4.2: Withdraw/deposit account DFD

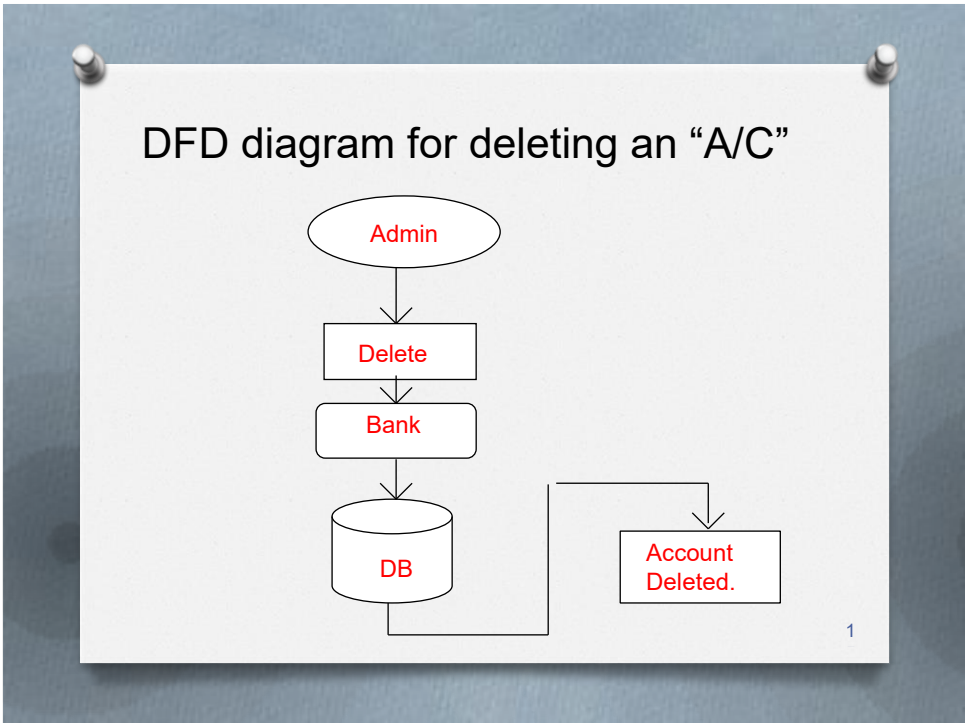
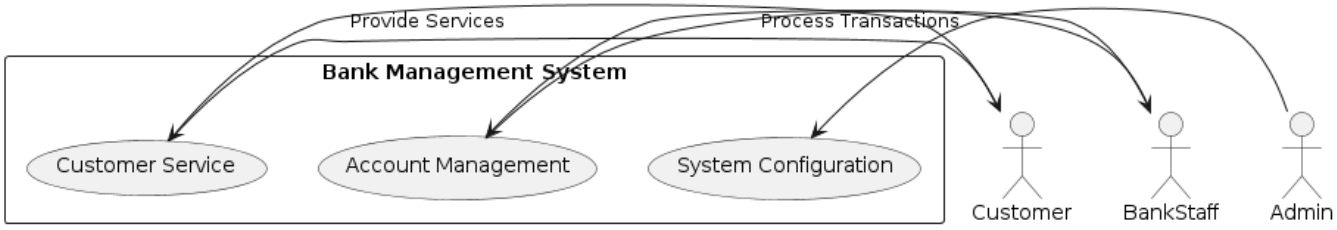
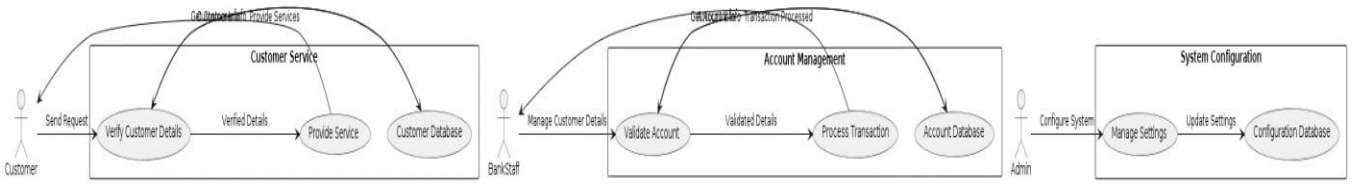


Figure-4.3: Deleting an account DFD

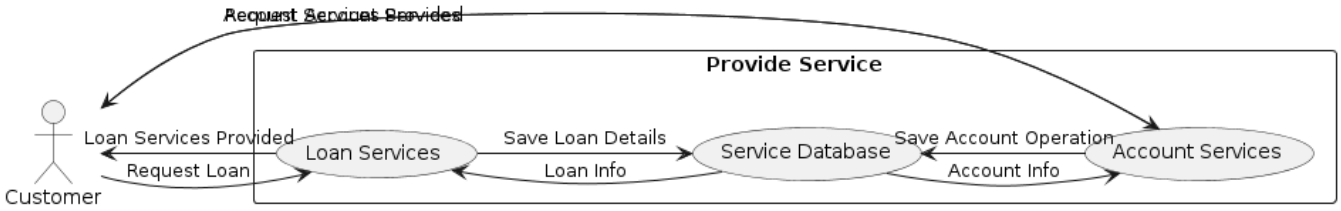
Zero Level DFD



First Level DFD



Second Level DFD



Banks terms:

1. All requests received from customers are logged for backend fulfillment and are effective from the time they are recorded at the branch.
2. Rules and regulations applicable to normal banking transactions in India will be applicable mutatis mutandis for the transactions executed through this site.
3. The BAMS Bank service cannot be claimed as a right. The bank may also convert this into a discretionary service anytime.
4. Dispute between the customer and the Bank in this service is subject to the jurisdiction of the courts in the Republic of India and governed by the laws prevailing in India.
5. The Bank reserves the right to modify the services offered or the Terms of service of BAMS Bank. The changes will be notified to the customers through a notification on the Site.

Customer's obligations

1. The customer has an obligation to maintain secrecy in regard to Username & Password registered with the Bank. The bank presupposes that login using valid Username and Password is a valid session initiated by none other than the customer.
2. Transaction executed through a valid session will be construed by RR to have emanated from the registered customer and will be binding on him/her.
3. The customer will not attempt or permit others to attempt accessing the BAMS Bank through any unlawful means.

Benefits of online banking

Many of us lead busy lives. Some of us are up before the crack of dawn, getting ourselves prepared so we can in turn get our families ready for the day. We rush to work, rush to get the kids to school, and at the end of the day we rush home only to brace ourselves for the next day. After a hectic day, the last thing you want to do is spend time waiting in line at the bank, or even the post office. That's where Online Banking comes in. Many of the benefits of doing our banking online are obvious:

1- You don't have to wait in line.
2- You don't have to plan your day around the bank's hours.
3- You can look at your balance whenever you want, not just when you get a statement.

There are some hidden benefits too. As a young bank customer, you're just learning how to manage your money and observe your spending patterns.

Online banking allows you to watch your money on a daily basis if you want to. By keeping close tabs on your funds, you'll always be aware of what's happening in your bank account.
For those experienced spenders, this option is far more appealing than the sudden discovery that you're broke!

It's also helpful to watch how much interest you're gathering on investments and savings or what service charges you have incurred.
Most available benefits

1. Online banking with key bank is fast, secure, convenient and free.
2. Quick, simple, authenticated access to accounts via the web application.
3. Simply scalable to grow with changing system requirement.
4. Global enterprise wide access to information.
5. Improved data security, restricting unauthorized access.
6. Minimize Storage Space.

Future Look

The "Banking Online System is a big and ambitious project. I am thankful for being provided this great opportunity to work on it. As already mentioned, this project has gone through extensive research work. On the basis of the research work, we have successfully designed and implemented banking online System. To know what the future of online banking looks like, it's probably worth looking at the present – online banking isn't new. When you think of online banking, you probably think about a computer (either a desktop or laptop), a three or four step security process and then an interface that lets you view the balance of your various bank accounts and credit cards, whilst permitting you to transfer money and pay bills. And you're not wrong either. The most valuable future looks are following below:
1- More branches of the bank, maybe it will be international, that means more ATM machines outside.
2- Customer issues development based on their needs, so the help desk will be aware of their needs and easy to use.
3- Developing a mobile App for banking system that help users to do the obtained his operations without go to the bank only he needs to sign in using his A/C NO. And password and then use your own PIN. Finally the system will update automatically.

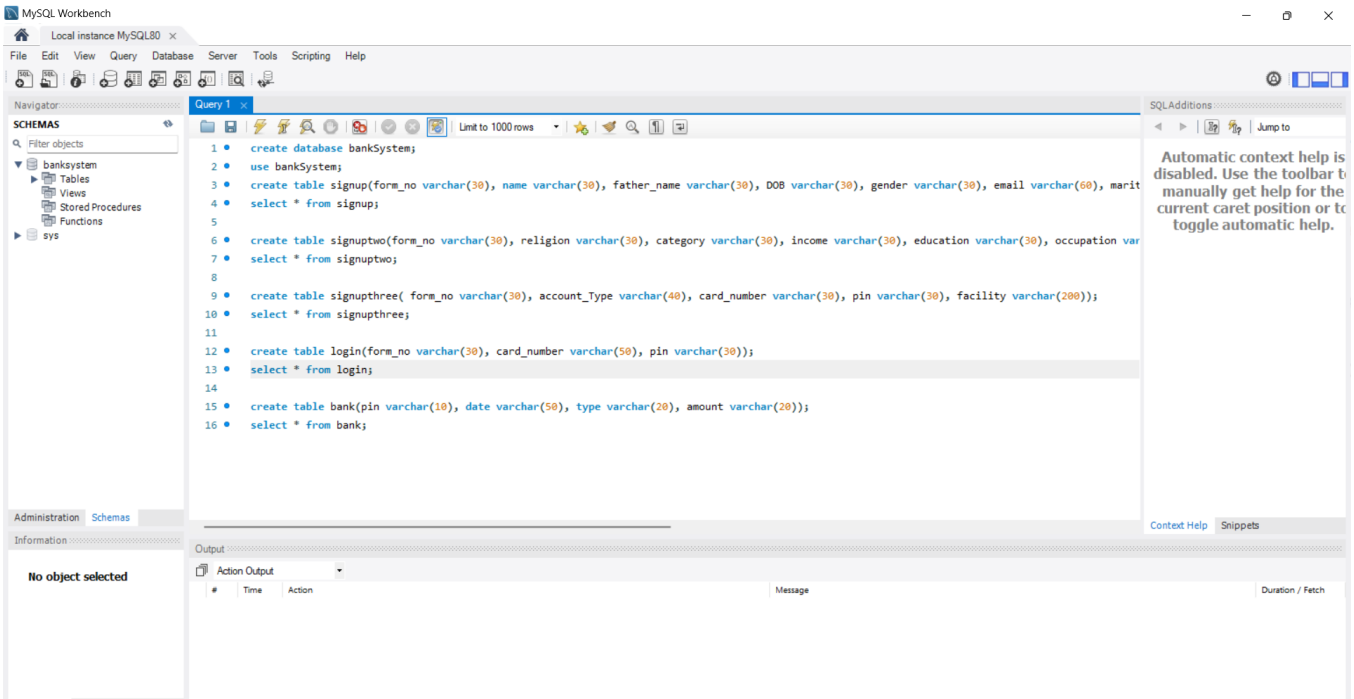
Database Design

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together

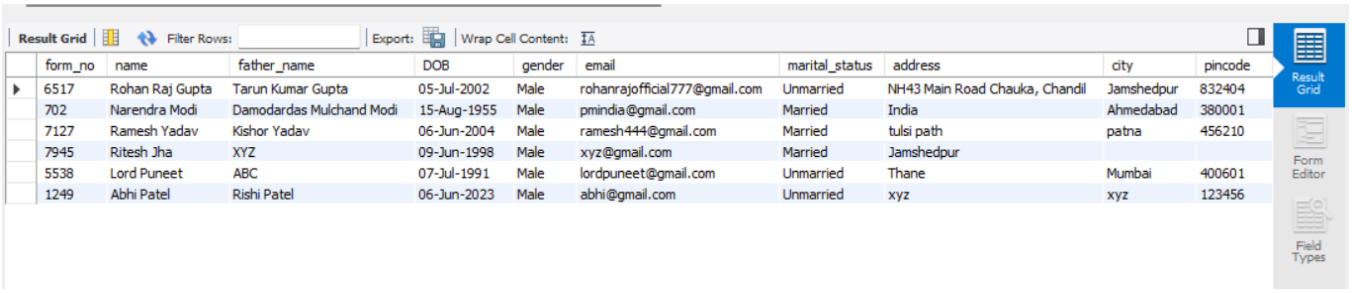
to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MySQL Workbench database has been chosen for developing the relevant databases.

Bank Management System (BMS) contains 5 MySQL tables :

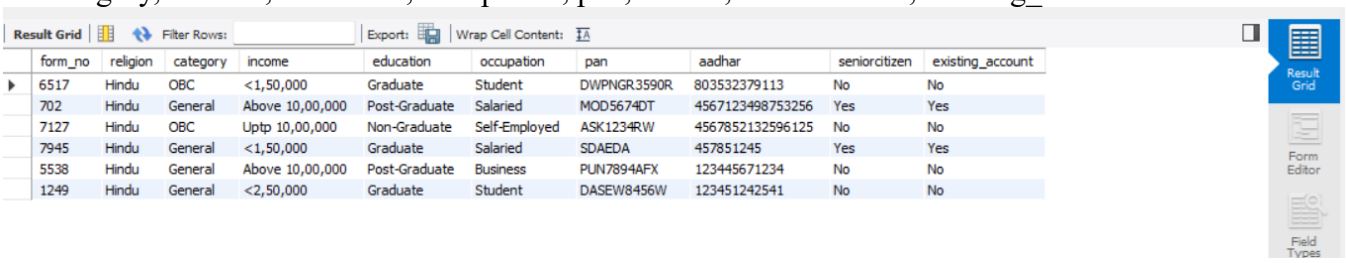


1. **Admin table Structure** : This table store the admin login and personal Details like name, father_name, DOB, gender, email, marital_status, address, city, pincode.



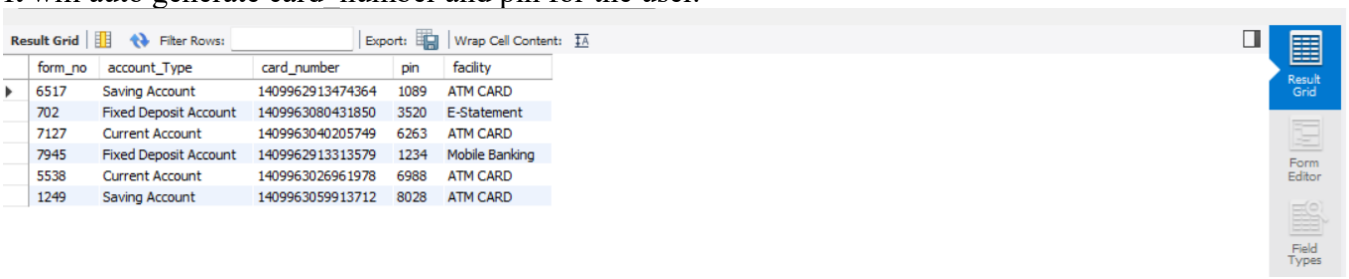
form_no	name	father_name	DOB	gender	email	marital_status	address	city	pincode
6517	Rohan Raj Gupta	Tarun Kumar Gupta	05-Jul-2002	Male	rohanrajofficial777@gmail.com	Unmarried	NH43 Main Road Chauka, Chandil	Jamshedpur	832404
702	Narendra Modi	Damodardas Mulchand Modi	15-Aug-1955	Male	pmindia@gmail.com	Married	India	Ahmedabad	380001
7127	Ramesh Yadav	Kishor Yadav	06-Jun-2004	Male	ramesh444@gmail.com	Married	tulsi path	patna	456210
7945	Ritesh Jha	XYZ	09-Jun-1998	Male	xyz@gmail.com	Married	Jamshedpur		
5538	Lord Puneet	ABC	07-Jul-1991	Male	lordpuneet@gmail.com	Unmarried	Thane	Mumbai	400601
1249	Abhi Patel	Rishi Patel	06-Jun-2023	Male	abhi@gmail.com	Unmarried	xyz	xyz	123456

2. **Personal Information Structure**: This table store Personal Details related to Religion, Category, Income, education, occupation, pan, aadhar, Seniorcitizen, existing_account.



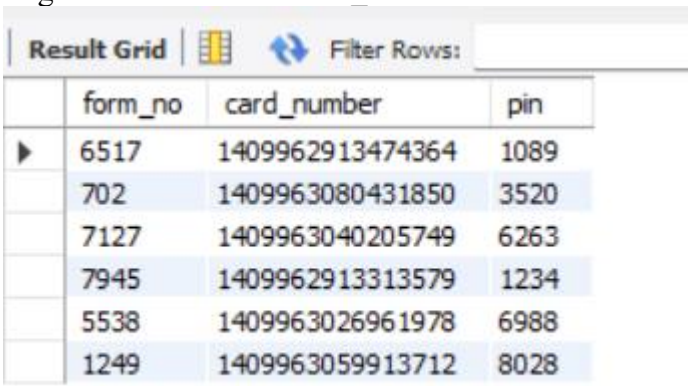
form_no	religion	category	income	education	occupation	pan	aadhar	seniorcitizen	existing_account
6517	Hindu	OBC	<1,50,000	Graduate	Student	DWPNGR3590R	803532379113	No	No
702	Hindu	General	Above 10,00,000	Post-Graduate	Salaried	MOD5674DT	4567123498753256	Yes	Yes
7127	Hindu	OBC	Uptp 10,00,000	Non-Graduate	Self-Employed	ASK1234RW	456785213259612 5	No	No
7945	Hindu	General	<1,50,000	Graduate	Salaried	SDAEDA	457851245	Yes	Yes
5538	Hindu	General	Above 10,00,000	Post-Graduate	Business	PUN7894AFX	123445671234	No	No
1249	Hindu	General	<2,50,000	Graduate	Student	DASEW8456W	123451242541	No	No

3. **Account Choice Structure:** This table store the details of choosing account_type and facility. It will auto generate card_number and pin for the user.



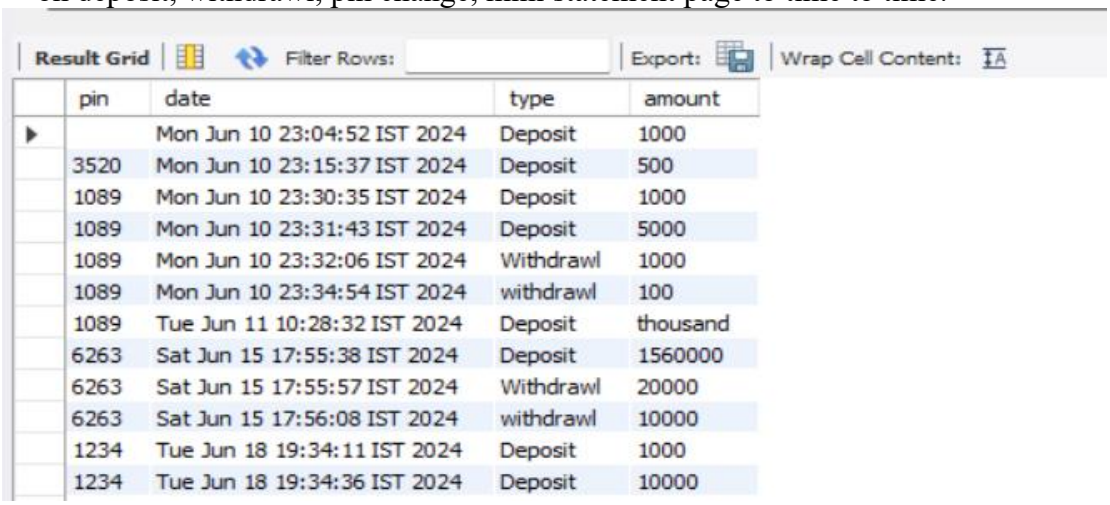
form_no	account_Type	card_number	pin	facility
6517	Saving Account	1409962913474364	1089	ATM CARD
702	Fixed Deposit Account	1409963080431850	3520	E-Statement
7127	Current Account	1409963040205749	6263	ATM CARD
7945	Fixed Deposit Account	1409962913313579	1234	Mobile Banking
5538	Current Account	1409963026961978	6988	ATM CARD
1249	Saving Account	1409963059913712	8028	ATM CARD

4. **Account Created Information Structure:** This Table shows what card_number and pin is generated to which form_no of the customer.



form_no	card_number	pin
6517	1409962913474364	1089
702	1409963080431850	3520
7127	1409963040205749	6263
7945	1409962913313579	1234
5538	1409963026961978	6988
1249	1409963059913712	8028

5. **Customer Transaction Information Structure**: This table shows customer daily transaction on deposit, withdrawl, pin change, mini statement page to time to time.



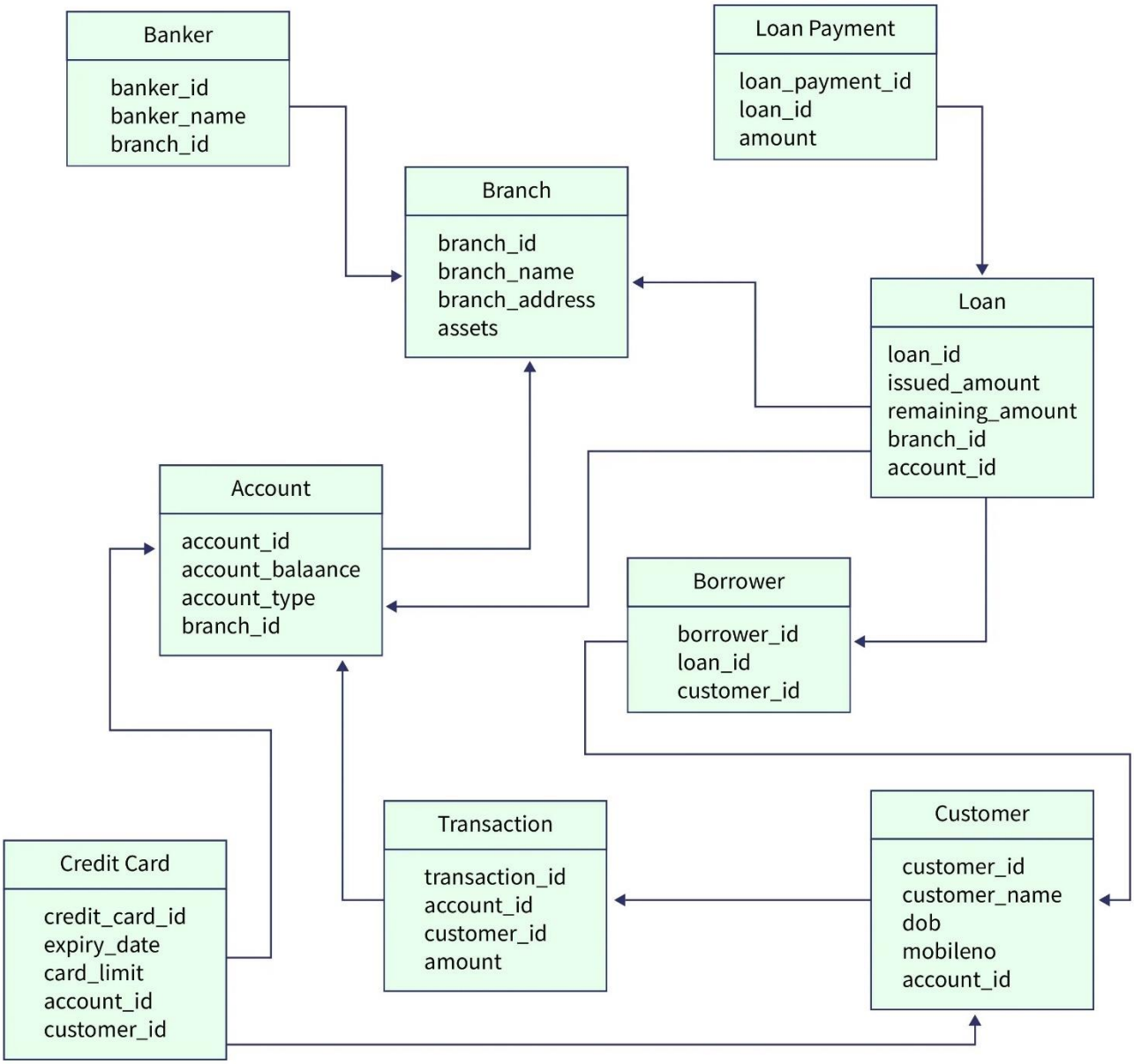
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

pin	date	type	amount
	Mon Jun 10 23:04:52 IST 2024	Deposit	1000
3520	Mon Jun 10 23:15:37 IST 2024	Deposit	500
1089	Mon Jun 10 23:30:35 IST 2024	Deposit	1000
1089	Mon Jun 10 23:31:43 IST 2024	Deposit	5000
1089	Mon Jun 10 23:32:06 IST 2024	Withdrawl	1000
1089	Mon Jun 10 23:34:54 IST 2024	withdrawl	100
1089	Tue Jun 11 10:28:32 IST 2024	Deposit	thousand
6263	Sat Jun 15 17:55:38 IST 2024	Deposit	1560000
6263	Sat Jun 15 17:55:57 IST 2024	Withdrawl	20000
6263	Sat Jun 15 17:56:08 IST 2024	withdrawl	10000
1234	Tue Jun 18 19:34:11 IST 2024	Deposit	1000
1234	Tue Jun 18 19:34:36 IST 2024	Deposit	10000

SCHEMA DIAGRAM:

A database schema is a **logical representation of data** that shows how the data in a database should be stored logically. It shows how the data is organized and the relationship between the tables.
Database schema contains table, field, views and relation between different keys like primary key, foreign key.
Data are stored in the form of files which is unstructured in nature which makes accessing the data difficult. Thus to resolve the issue the data are organized in structured way with the help of database schema.
Database schema provides the organization of data and the relationship between the stored data.
Database schema defines a set of guidelines that control the database along with that it provides information about the way of accessing and modifying the data.

SCHEMA DIAGRAM



JAVA CORE (Using SWING & AWT) Source Code:

1. Login.java :

```
package bank.management.system;

import javax.swing.*;
```

```
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;

public class Login extends JFrame implements ActionListener {
   JLabel label1, label2, label3;
   JTextField textField2;
   JPasswordField passwordField3;

   JButton button1,button2,button3;
   Login(){
      super("Bank Management System");
      ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icon/bank.png"));
      Image i2 = i1.getImage().getScaledInstance(100,100,Image.SCALE_DEFAULT);
      ImageIcon i3 = new ImageIcon(i2);
      JLabel image = new JLabel(i3);
      image.setBounds(350,10,100,100);
      add(image);

      ImageIcon ii1 = new ImageIcon(ClassLoader.getSystemResource("icon/card.png"));
      Image ii2 = ii1.getImage().getScaledInstance(100,100,Image.SCALE_DEFAULT);
      ImageIcon ii3 = new ImageIcon(ii2);
      JLabel iimage = new JLabel(ii3);
      iimage.setBounds(630,350,100,100);
      add(iimage);

      label1 = new JLabel("WELCOME TO ATM");
      label1.setForeground(Color.WHITE);
      label1.setFont(new Font("AvantGarde", Font.BOLD, 38));
      label1.setBounds(230,125,450,40);
      add(label1);

      label2 = new JLabel("Card No:");
      label2.setFont(new Font("Ralway", Font.BOLD, 28));
      label2.setForeground(Color.WHITE);
      label2.setBounds(150,190,375,30);
      add(label2);

      textField2 = new JTextField(15);
      textField2.setBounds(325,190,230,30);
      textField2.setFont(new Font("Arial", Font.BOLD,14));
      add(textField2);

      label3 = new JLabel("PIN: ");
      label3.setFont(new Font("Ralway", Font.BOLD, 28));
      label3.setForeground(Color.WHITE);
      label3.setBounds(150,250,375,30);
      add(label3);
```

```
    passwordField3 = new JPasswordField(15);
    passwordField3.setBounds(325,250,230,30);
    passwordField3.setFont(new Font("Arial", Font.BOLD, 14));
    add(passwordField3);

    button1 = new JButton("SIGN IN");
    button1.setFont(new Font("Arial", Font.BOLD, 14));
    button1.setForeground(Color.WHITE);
    button1.setBackground(Color.BLACK);
    button1.setBounds(300,300,100, 30);
    button1.addActionListener(this);
    add(button1);

    button2 = new JButton("CLEAR");
    button2.setFont(new Font("Arial", Font.BOLD, 14));
    button2.setForeground(Color.WHITE);
    button2.setBackground(Color.BLACK);
    button2.setBounds(430,300,100, 30);
    button2.addActionListener(this);
    add(button2);

    button3 = new JButton("SIGN UP");
    button3.setFont(new Font("Arial", Font.BOLD, 14));
    button3.setForeground(Color.WHITE);
    button3.setBackground(Color.BLACK);
    button3.setBounds(300,350,230, 30);
    button3.addActionListener(this);
    add(button3);

    ImageIcon iii1 = new ImageIcon(ClassLoader.getSystemResource("icon/backbg.png"));
    Image iii2 = iii1.getImage().getScaledInstance(850,480,Image.SCALE_DEFAULT);
    ImageIcon iii3 = new ImageIcon(iii2);
    JLabel iiimage = new JLabel(iii3);
    iiimage.setBounds(0,0,850,480);
    add(iiimage);


    setLayout(null);
    setSize(850,480);
    setLocation(450,200);
    setUndecorated(true);
    setVisible(true);
}

@Override
public void actionPerformed(ActionEvent e) {
    try{
        if (e.getSource()==button1){
```

```
            Connn c = new Connn();
            String cardno = textField2.getText();
            String pin = passwordField3.getText();
            String q = "select * from login where card_number = '"+cardno+"' and  pin = '"+pin+"'";
            ResultSet resultSet = c.statement.executeQuery(q);
            if (resultSet.next()){
               setVisible(false);
               new main_Class(pin);
            }else {
               JOptionPane.showMessageDialog(null,"Incorrect Card Number or PIN");
            }


         }else if (e.getSource() == button2){
            textField2.setText("");
            passwordField3.setText("");
         }else if (e.getSource() == button3){
            new Signup();
            setVisible(false);
         }
      }catch (Exception E){
         E.printStackTrace();
      }

   }

   public static void main(String[] args) {
      new Login();
   }
}
```

2. Signup.java :

```
package bank.management.system;

import com.toedter.calendar.JDateChooser;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.Random;

public class Signup extends JFrame implements ActionListener {
   JRadioButton r1,r2,m1,m2,m3;
   JButton next;

   JTextField textName ,textFname, textEmail,textAdd,textcity,textState,textPin;
   JDateChooser dateChooser;
```

```
Random ran = new Random();
long first4 =(ran.nextLong() % 9000L) +1000L;
String first = " " + Math.abs(first4);
Signup(){
   super ("APPLICATION FORM");

   ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icon/bank.png"));
   Image i2 = i1.getImage().getScaledInstance(100,100,Image.SCALE_DEFAULT);
   ImageIcon i3 = new ImageIcon(i2);
   JLabel image = new JLabel(i3);
   image.setBounds(25,10,100,100);
   add(image);

   JLabel label1 = new JLabel("APPLICATION FORM NO."+ first);
   label1.setBounds(160,20,600,40);
   label1.setFont(new Font("Raleway",Font.BOLD,38));
   add(label1);

   JLabel label2 = new JLabel("Page 1");
   label2.setFont(new Font("Ralway",Font.BOLD, 22));
   label2.setBounds(330,70,600,30);
   add(label2);

   JLabel label3 = new JLabel("Personal Details");
   label3.setFont(new Font("Raleway", Font.BOLD,22));
   label3.setBounds(290,90,600,30);
   add(label3);

   JLabel labelName = new JLabel("Name :");
   labelName.setFont(new Font("Raleway", Font.BOLD, 20));
   labelName.setBounds(100,190,100,30);
   add(labelName);

   textName = new JTextField();
   textName.setFont(new Font("Raleway",Font.BOLD, 14));
   textName.setBounds(300,190,400,30);
   add(textName);

   JLabel labelfName = new JLabel("Father's Name :");
   labelfName.setFont(new Font("Raleway", Font.BOLD, 20));
   labelfName.setBounds(100,240,200,30);
   add(labelfName);

   textFname = new JTextField();
   textFname.setFont(new Font("Raleway",Font.BOLD, 14));
   textFname.setBounds(300,240,400,30);
   add(textFname);

   JLabel DOB = new JLabel("Date of Birth");
```

```
DOB.setFont(new Font("Raleway", Font.BOLD, 20));
DOB.setBounds(100,340,200,30);
add(DOB);

dateChooser = new JDateChooser();
dateChooser.setForeground(new Color(105,105,105));
dateChooser.setBounds(300,340,400,30);
add(dateChooser);

JLabel labelG = new JLabel("Gender");
labelG.setFont(new Font("Raleway", Font.BOLD, 20));
labelG.setBounds(100,290,200,30);
add(labelG);

r1 = new JRadioButton("Male");
r1.setFont(new Font("Raleway", Font.BOLD,14));
r1.setBackground(new Color(222,255,228));
r1.setBounds(300,290,60,30);
add(r1);

r2 = new JRadioButton("Female");
r2.setBackground(new Color(222,255,228));
r2.setFont(new Font("Raleway", Font.BOLD,14));
r2.setBounds(450,290,90,30);
add(r2);

ButtonGroup buttonGroup = new ButtonGroup();
buttonGroup.add(r1);
buttonGroup.add(r2);

JLabel labelEmail = new JLabel("Email address :");
labelEmail.setFont(new Font("Raleway", Font.BOLD, 20));
labelEmail.setBounds(100,390,200,30);
add(labelEmail);

textEmail = new JTextField();
textEmail.setFont(new Font("Raleway",Font.BOLD, 14));
textEmail.setBounds(300,390,400,30);
add(textEmail);


JLabel labelMs = new JLabel("Marital Status :");
labelMs.setFont(new Font("Raleway", Font.BOLD, 20));
labelMs.setBounds(100,440,200,30);
add(labelMs);

m1 = new JRadioButton("Married");
m1.setBounds(300,440,100,30);
m1.setBackground(new Color(222,255,228));
```

```
m1.setFont(new Font("Raleway", Font.BOLD,14));
add(m1);

m2 = new JRadioButton("Unmarried");
m2.setBackground(new Color(222,255,228));
m2.setBounds(450,440,100,30);
m2.setFont(new Font("Raleway", Font.BOLD,14));
add(m2);

m3 = new JRadioButton("Other");
m3.setBackground(new Color(222,255,228));
m3.setBounds(635,440,100,30);
m3.setFont(new Font("Raleway", Font.BOLD,14));
add(m3);

ButtonGroup buttonGroup1 = new ButtonGroup();
buttonGroup1.add(m1);
buttonGroup1.add(m2);
buttonGroup1.add(m3);

JLabel labelAdd = new JLabel("Address :");
labelAdd.setFont(new Font("Raleway", Font.BOLD, 20));
labelAdd.setBounds(100,490,200,30);
add(labelAdd);

textAdd = new JTextField();
textAdd.setFont(new Font("Raleway",Font.BOLD, 14));
textAdd.setBounds(300,490,400,30);
add(textAdd);

JLabel labelCity = new JLabel("City :");
labelCity.setFont(new Font("Raleway", Font.BOLD, 20));
labelCity.setBounds(100,540,200,30);
add(labelCity);

textcity = new JTextField();
textcity.setFont(new Font("Raleway",Font.BOLD, 14));
textcity.setBounds(300,540,400,30);
add(textcity);

JLabel labelPin = new JLabel("Pin Code :");
labelPin.setFont(new Font("Raleway", Font.BOLD, 20));
labelPin.setBounds(100,590,200,30);
add(labelPin);

textPin = new JTextField();
textPin.setFont(new Font("Raleway",Font.BOLD, 14));
textPin.setBounds(300,590,400,30);
add(textPin);
```

```
    JLabel labelstate = new JLabel("State :");
    labelstate.setFont(new Font("Raleway", Font.BOLD, 20));
    labelstate.setBounds(100,640,200,30);
    add( labelstate);

    textState = new JTextField();
    textState.setFont(new Font("Raleway",Font.BOLD, 14));
    textState.setBounds(300,640,400,30);
    add(textState);

    next = new JButton("Next");
    next.setFont(new Font("Raleway",Font.BOLD, 14));
    next.setBackground(Color.BLACK);
    next.setForeground(Color.WHITE);
    next.setBounds(620,710,80,30);
    next.addActionListener(this);
    add(next);

    getContentPane().setBackground(new Color(222,255,228));
    setLayout(null);
    setSize(850,800);
    setLocation(360,40);
    setVisible(true);

}

@Override
public void actionPerformed(ActionEvent e) {

    String formno = first;
    String name = textName.getText();
    String fname = textFname.getText();
    String dob = ((JTextField) dateChooser.getDateEditor().getUiComponent()).getText();
    String gender = null;
    if(r1.isSelected()){
        gender = "Male";
    }else if (r2.isSelected()){
        gender = "Female";
    }
    String email = textEmail.getText();
    String marital =null;
    if (m1.isSelected()){
        marital = "Married";
    } else if (m2.isSelected()) {
        marital = "Unmarried";
    } else if (m3.isSelected()) {
        marital = "Other";
    }
```

```
        String address = textAdd.getText();
        String city = textcity.getText();
        String pincode = textPin.getText();
        String state = textState.getText();

        try{
            if (textName.getText().equals("")){
                JOptionPane.showMessageDialog(null, "Fill all the fields");
            }else {
                Connn c = new Connn();
                String q = "insert into signup values('"+formno+"',
'"+name+"','"+fname+"','"+dob+"','"+gender+"','"+email+"','"+marital+"', '"+address+"',
'"+city+"','"+pincode+"','"+state+"' )";
                c.statement.executeUpdate(q);
                new Signup2(formno);
                setVisible(false);
            }

        }catch (Exception E){
            E.printStackTrace();
        }

    }

    public static void main(String[] args) {
        new Signup();
    }
}
```

3. **Signup2.java;**

```
package bank.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class Signup2 extends JFrame implements ActionListener {
    JComboBox comboBox,comboBox2,comboBox3,comboBox4,comboBox5;
    JTextField textPan,textAadhar;
    JRadioButton r1,r2, e1,e2;
    JButton next;
    String formno;
    Signup2(String formno){
        super("APPLICATION FORM");

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icon/bank.png"));
        Image i2 = i1.getImage().getScaledInstance(100,100,Image.SCALE_DEFAULT);
```

```
ImageIcon i3 = new ImageIcon(i2);
JLabel image = new JLabel(i3);
image.setBounds(150,5,100,100);
add(image);

this.formno = formno;

JLabel l1 = new JLabel("Page 2 :-");
l1.setFont(new Font("Raleway", Font.BOLD,22));
l1.setBounds(300,30,600,40);
add(l1);

JLabel l2 = new JLabel("Additonal Details");
l2.setFont(new Font("Raleway", Font.BOLD,22));
l2.setBounds(300,60,600,40);
add(l2);

JLabel l3 = new JLabel("Religion :");
l3.setFont(new Font("Raleway", Font.BOLD,18));
l3.setBounds(100,120,100,30);
add(l3);

String religion[] = {"Hindu","Muslim","Sikh", "Christian", "Other"};
comboBox = new JComboBox(religion);
comboBox.setBackground(new Color(252,208,76));
comboBox.setFont(new Font("Raleway",Font.BOLD,14));
comboBox.setBounds(350,120,320,30);
add(comboBox);

JLabel l4 = new JLabel("Category : ");
l4.setFont(new Font("Raleway", Font.BOLD,18));
l4.setBounds(100,170,100,30);
add(l4);

String Category [] = {"General","OBC","SC", "ST", "Other"};
comboBox2 = new JComboBox(Category);
comboBox2.setBackground(new Color(252,208,76));
comboBox2.setFont(new Font("Raleway",Font.BOLD,14));
comboBox2.setBounds(350,170,320,30);
add(comboBox2);

JLabel l5 = new JLabel("Income : ");
l5.setFont(new Font("Raleway", Font.BOLD,18));
l5.setBounds(100,220,100,30);
add(l5);

String income [] = {"Null","<1,50,000","<2,50,000", "5,00,000", "Uptp 10,00,000","Above 10,00,000"};
comboBox3 = new JComboBox(income);
comboBox3.setBackground(new Color(252,208,76));
```

```
comboBox3.setFont(new Font("Raleway",Font.BOLD,14));
comboBox3.setBounds(350,220,320,30);
add(comboBox3);

JLabel l6 = new JLabel("Educational : ");
l6.setFont(new Font("Raleway", Font.BOLD,18));
l6.setBounds(100,270,150,30);
add(l6);

String educational [] = {"Non-Graduate","Graduate","Post-Graduate", "Doctrate", "Others"};
comboBox4 = new JComboBox(educational);
comboBox4.setBackground(new Color(252,208,76));
comboBox4.setFont(new Font("Raleway",Font.BOLD,14));
comboBox4.setBounds(350,270,320,30);
add(comboBox4);


JLabel l7 = new JLabel("Occupation : ");
l7.setFont(new Font("Raleway", Font.BOLD,18));
l7.setBounds(100,340,150,30);
add(l7);

String Occupation [] = {"Salaried","Self-Employed","Business", "Student", "Retired", "Other"};
comboBox5 = new JComboBox(Occupation);
comboBox5.setBackground(new Color(252,208,76));
comboBox5.setFont(new Font("Raleway",Font.BOLD,14));
comboBox5.setBounds(350,340,320,30);
add(comboBox5);

JLabel l8 = new JLabel("PAN Number : ");
l8.setFont(new Font("Raleway", Font.BOLD,18));
l8.setBounds(100,390,150,30);
add(l8);

textPan = new JTextField();
textPan.setFont(new Font("Raleway", Font.BOLD,18));
textPan.setBounds(350,390,320,30);
add(textPan);

JLabel l9 = new JLabel("Aadhar Number : ");
l9.setFont(new Font("Raleway", Font.BOLD,18));
l9.setBounds(100,440,180,30);
add(l9);

textAadhar = new JTextField();
textAadhar.setFont(new Font("Raleway", Font.BOLD,18));
textAadhar.setBounds(350,440,320,30);
add(textAadhar);
```

```
JLabel l10 = new JLabel("Senior Citizen : ");
l10.setFont(new Font("Raleway", Font.BOLD,18));
l10.setBounds(100,490,180,30);
add(l10);

r1 = new JRadioButton("Yes");
r1.setFont(new Font("Raleway", Font.BOLD,14));
r1.setBackground(new Color(252,208,76));
r1.setBounds(350,490,100,30);
add(r1);
r2 = new JRadioButton("No");
r2.setFont(new Font("Raleway", Font.BOLD,14));
r2.setBackground(new Color(252,208,76));
r2.setBounds(460,490,100,30);
add(r2);

JLabel l11 = new JLabel("Existing Account : ");
l11.setFont(new Font("Raleway", Font.BOLD,18));
l11.setBounds(100,540,180,30);
add(l11);

e1 = new JRadioButton("Yes");
e1.setFont(new Font("Raleway", Font.BOLD,14));
e1.setBackground(new Color(252,208,76));
e1.setBounds(350,540,100,30);
add(e1);
e2 = new JRadioButton("No");
e2.setFont(new Font("Raleway", Font.BOLD,14));
e2.setBackground(new Color(252,208,76));
e2.setBounds(460,540,100,30);
add(e2);

JLabel l12 = new JLabel("Form No : ");
l12.setFont(new Font("Raleway", Font.BOLD,14));
l12.setBounds(700,10,100,30);
add(l12);

JLabel l13 = new JLabel(formno);
l13.setFont(new Font("Raleway", Font.BOLD,14));
l13.setBounds(760,10,60,30);
add(l13);

next = new JButton("Next");
next.setFont(new Font("Raleway",Font.BOLD,14));
next.setBackground(Color.WHITE);
next.setForeground(Color.BLACK);
next.setBounds(570,640,100,30);
next.addActionListener(this);
```

```
        add(next);


        setLayout(null);
        setSize(850,750);
        setLocation(450,80);
        getContentPane().setBackground(new Color(252, 208, 76));
        setVisible(true);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String rel = (String) comboBox.getSelectedItem();
        String cate = (String) comboBox2.getSelectedItem();
        String inc = (String) comboBox3.getSelectedItem();
        String edu = (String) comboBox4.getSelectedItem();
        String occ = (String) comboBox5.getSelectedItem();

        String pan = textPan.getText();
        String addhar = textAadhar.getText();

        String scitizen = " ";
        if ((r1.isSelected())){
            scitizen = "Yes";
        } else if (r2.isSelected()) {
            scitizen ="No";
        }
        String eAccount = " ";
        if ((r1.isSelected())){
            eAccount = "Yes";
        } else if (r2.isSelected()) {
            eAccount ="No";
        }

        try{
            if (textPan.getText().equals("") || textAadhar.getText().equals("")){
                JOptionPane.showMessageDialog(null,"Fill all the fields");
            }else {
                Connn c = new Connn();
                String q = "insert into Signuptwo values('"+formno+"', '"+rel+"',
'"+cate+"','"+inc+"','"+edu+"','"+occ+"','"+pan+"','"+addhar+"','"+scitizen+"','"+eAccount+"')";
                c.statement.executeUpdate(q);
                new Signup3(formno);
                setVisible(false);
            }


        }catch (Exception E){
            E.printStackTrace();
```

```
        }


    }

    public static void main(String[] args) {
        new Signup2("");
    }
}
```

4. Signup3.java :

```
package bank.management.system;

import javax.print.attribute.standard.JobHoldUntil;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.rmi.server.ExportException;
import java.util.Random;

public class Signup3 extends JFrame implements ActionListener {

    JRadioButton r1,r2,r3,r4;
    JCheckBox c1,c2,c3,c4,c5,c6;
    JButton s,c;
    String formno;
    Signup3(String formno){

        this.formno = formno;

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icon/bank.png"));
        Image i2 = i1.getImage().getScaledInstance(100,100,Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(i2);
        JLabel image = new JLabel(i3);
        image.setBounds(150,5,100,100);
        add(image);

        JLabel l1 = new JLabel("Page 3:");
        l1.setFont(new Font("Raleway",Font.BOLD,22));
        l1.setBounds(280,40,400,40);
        add(l1);

        JLabel l2 = new JLabel("Account Details");
        l2.setFont(new Font("Raleway",Font.BOLD,22));
        l2.setBounds(280,70,400,40);
        add(l2);
```

```
JLabel l3 = new JLabel("Account Type:");
l3.setFont(new Font("Raleway",Font.BOLD,18));
l3.setBounds(100,140,200,30);
add(l3);

r1 = new JRadioButton("Saving Account");
r1.setFont(new Font("Raleway",Font.BOLD,16));
r1.setBackground(new Color(215,252,252));
r1.setBounds(100,180,150,30);
add(r1);

r2 = new JRadioButton("Fixed Deposit Account");
r2.setFont(new Font("Raleway",Font.BOLD,16));
r2.setBackground(new Color(215,252,252));
r2.setBounds(350,180,300,30);
add(r2);

r3 = new JRadioButton("Current Account");
r3.setFont(new Font("Raleway",Font.BOLD,16));
r3.setBackground(new Color(215,252,252));
r3.setBounds(100,220,250,30);
add(r3);

r4 = new JRadioButton("Recurring Deposit Account");
r4.setFont(new Font("Raleway",Font.BOLD,16));
r4.setBackground(new Color(215,252,252));
r4.setBounds(350,220,250,30);
add(r4);

ButtonGroup buttonGroup = new ButtonGroup();
buttonGroup.add(r1);
buttonGroup.add(r2);
buttonGroup.add(r3);
buttonGroup.add(r4);

JLabel l4 = new JLabel("Card Number:");
l4.setFont(new Font("Raleway",Font.BOLD,18));
l4.setBounds(100,300,200,30);
add(l4);

JLabel l5 = new JLabel("(Your 16-digit Card Number)");
l5.setFont(new Font("Raleway",Font.BOLD,12));
l5.setBounds(100,330,200,20);
add(l5);

JLabel l6 = new JLabel("XXXX-XXXX-XXXX-4841");
l6.setFont(new Font("Raleway",Font.BOLD,18));
l6.setBounds(330,300,250,30);
add(l6);
```

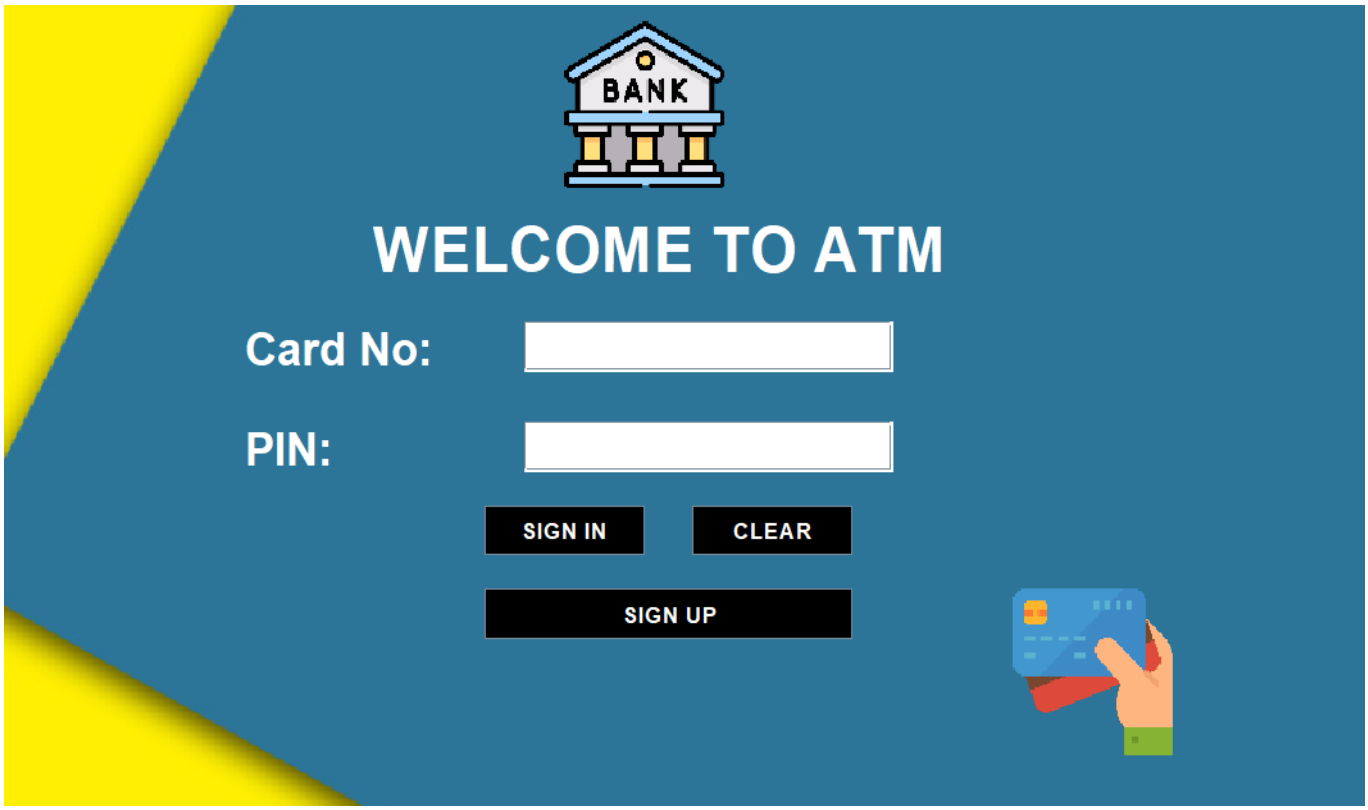
```
JLabel l7 = new JLabel("(It would appear on atm card/cheque Book and Statements)");
l7.setFont(new Font("Raleway",Font.BOLD,12));
l7.setBounds(330,330,500,20);
add(l7);

JLabel l8 = new JLabel("PIN:");
l8.setFont(new Font("Raleway",Font.BOLD,18));
l8.setBounds(100,370,200,30);
add(l8);

JLabel l9 = new JLabel("XXXX");
l9.setFont(new Font("Raleway",Font.BOLD,18));
l9.setBounds(330,370,200,30);
add(l9);

JLabel l10 = new JLabel("(4-digit Password)");
l10.setFont(new Font("Raleway",Font.BOLD,12));
l10.setBounds(100,400,200,20);
add(l10);

JLabel l11 = new JLabel("Services Required:");
l11.setFont(new Font("Raleway",Font.BOLD,18));
l11.setBounds(100,450,200,30);
add(l11);

c1 = new JCheckBox("ATM CARD");
c1.setBackground(new Color(215,252,252));
c1.setFont(new Font("Raleway",Font.BOLD,16));
c1.setBounds(100,500,200,30);
add(c1);

c2 = new JCheckBox("Internet Banking");
c2.setBackground(new Color(215,252,252));
c2.setFont(new Font("Raleway",Font.BOLD,16));
c2.setBounds(350,500,200,30);
add(c2);

c3 = new JCheckBox("Mobile Banking");
c3.setBackground(new Color(215,252,252));
c3.setFont(new Font("Raleway",Font.BOLD,16));
c3.setBounds(100,550,200,30);
add(c3);

c4 = new JCheckBox("EMAIL Alerts");
c4.setBackground(new Color(215,252,252));
c4.setFont(new Font("Raleway",Font.BOLD,16));
c4.setBounds(350,550,200,30);
add(c4);
```

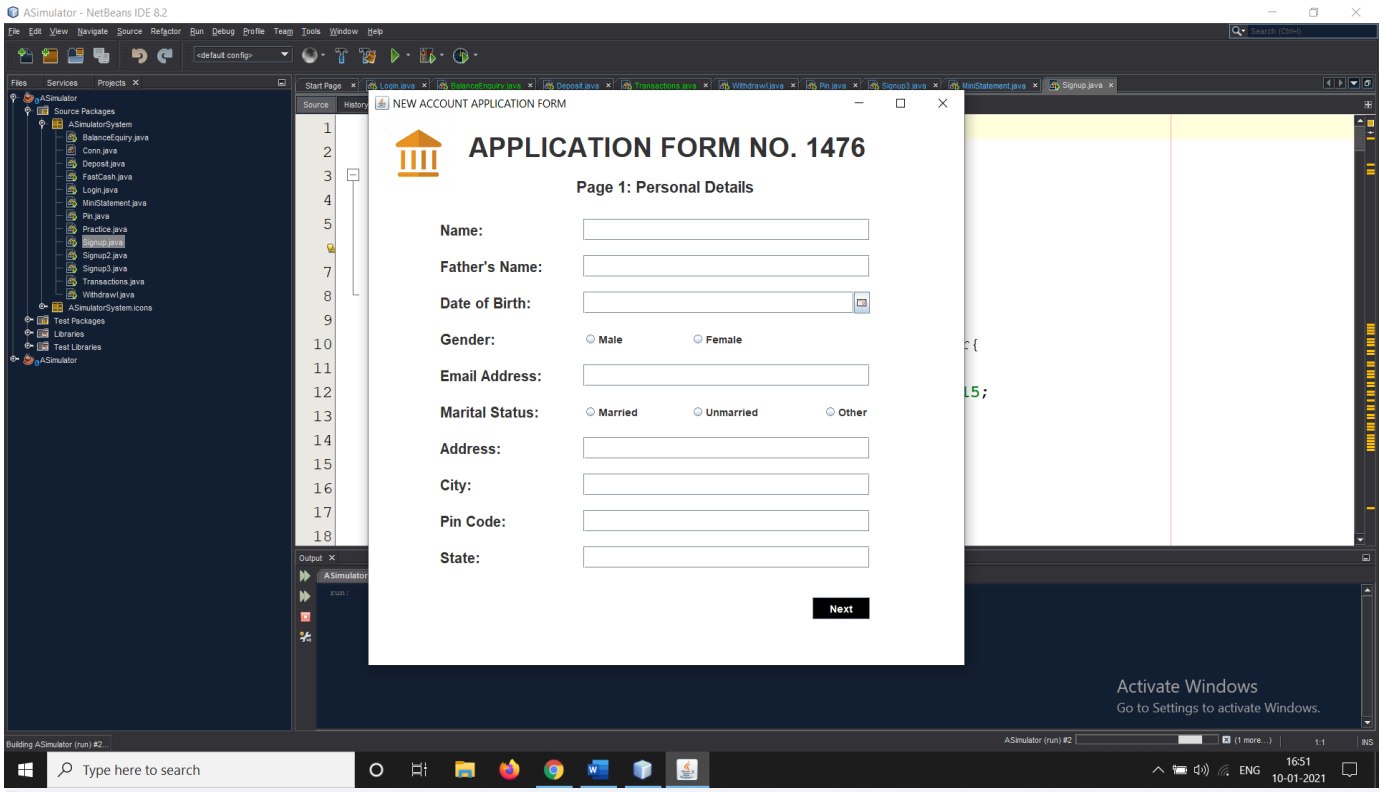
```
c5 = new JCheckBox("Cheque Book");
c5.setBackground(new Color(215,252,252));
c5.setFont(new Font("Raleway",Font.BOLD,16));
c5.setBounds(100,600,200,30);
add(c5);

c6 = new JCheckBox("E-Statement");
c6.setBackground(new Color(215,252,252));
c6.setFont(new Font("Raleway",Font.BOLD,16));
c6.setBounds(350,600,200,30);
add(c6);

JCheckBox c7 = new JCheckBox("I here by decleares that the above entered details correct to the best of my knlowledge.",true);
c7.setBackground(new Color(215,252,252));
c7.setFont(new Font("Raleway",Font.BOLD,12));
c7.setBounds(100,680,600,20);
add(c7);

JLabel l12 = new JLabel("Form No : ");
l12.setFont(new Font("Raleway", Font.BOLD,14));
l12.setBounds(700,10,100,30);
add(l12);

JLabel l13 = new JLabel(formno);
l13.setFont(new Font("Raleway", Font.BOLD,14));
l13.setBounds(760,10,60,30);
add(l13);


s = new JButton("Submit");
s.setFont(new Font("Raleway", Font.BOLD,14));
s.setBackground(Color.BLACK);
s.setForeground(Color.WHITE);
s.setBounds(250,720,100,30);
s.addActionListener(this);
add(s);

c = new JButton("Cancel");
c.setFont(new Font("Raleway", Font.BOLD,14));
c.setBackground(Color.BLACK);
c.setForeground(Color.WHITE);
c.setBounds(420,720,100,30);
c.addActionListener(this);
add(c);

getContentPane().setBackground(new Color(215,252,252));
setSize(850,800);
```

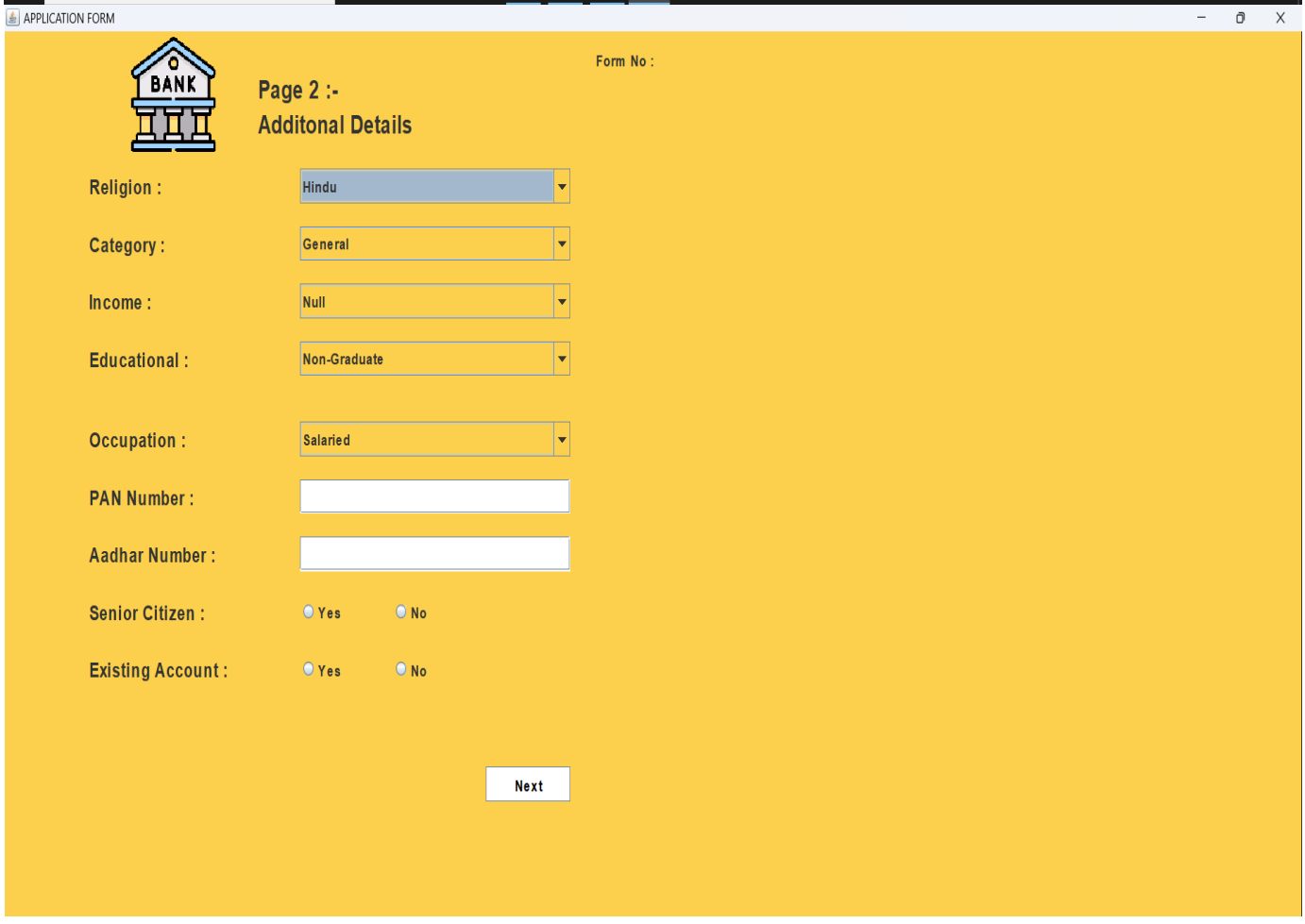
```
    setLayout(null);
    setLocation(400,20);
    setVisible(true);
}

@Override
public void actionPerformed(ActionEvent e) {
    String atype = null;
    if (r1.isSelected()){
        atype = "Saving Account";
    } else if (r2.isSelected()) {
        atype ="Fixed Deposit Account";
    }else if (r3.isSelected()){
        atype ="Current Account";
    }else if (r4.isSelected()){
        atype = "Recurring Deposit Account";
    }

    Random ran = new Random();
    long first7 = (ran.nextLong() % 90000000L) + 1409963000000000L;
    String cardno = "" + Math.abs(first7);

    long first3 = (ran.nextLong() % 9000L)+ 1000L;
    String pin = "" + Math.abs(first3);

    String fac = "";
    if(c1.isSelected()){
        fac = fac+"ATM CARD ";
    } else if (c2.isSelected()) {
        fac = fac+"Internet Banking";
    } else if (c3.isSelected()) {
        fac = fac+"Mobile Banking";
    } else if (c4.isSelected()) {
        fac = fac+"EMAIL Alerts";
    } else if (c5.isSelected()) {
        fac=fac+"Cheque Book";
    } else if (c6.isSelected()) {
        fac=fac+"E-Statement";
    }

    try {
        if (e.getSource()==s){
            if (atype.equals("")){
                JOptionPane.showMessageDialog(null,"Fill all the fields");
            }else {
                Connn c1 = new Connn();
                String q1 = "insert into signupthree values('"+formno+"', '"+atype+"','"+cardno+"','"+pin+"','"+fac+"')";
                String q2 = "insert into login values('"+formno+"','"+cardno+"','"+pin+"')";
                c1.statement.executeUpdate(q1);
```

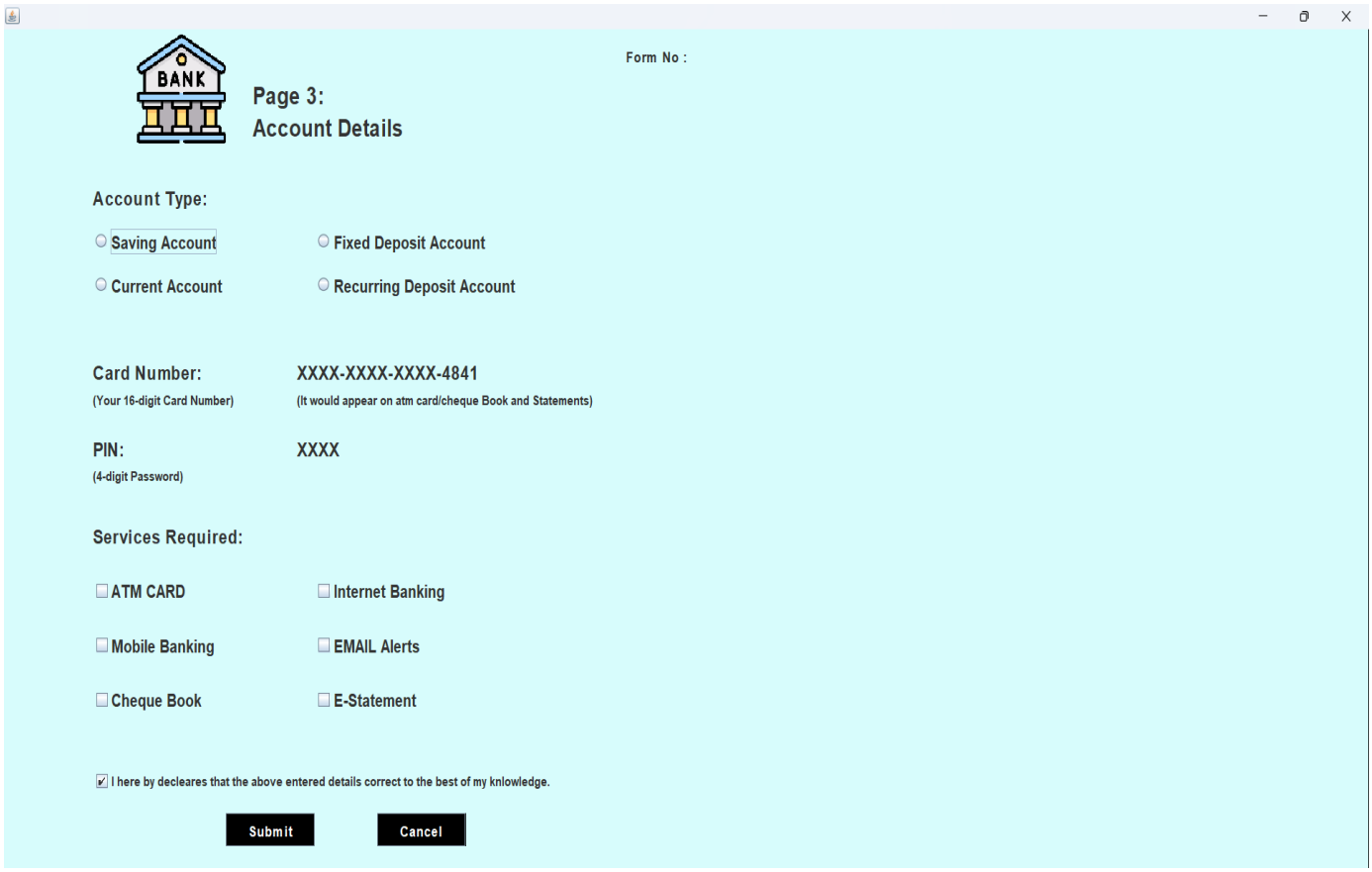
```
            c1.statement.executeUpdate(q2);
            JOptionPane.showMessageDialog(null,"Card Number : "+cardno+"\n Pin : "+pin );
            new Deposit(pin);
            setVisible(false);
          }
        } else if (e.getSource()==c) {
          System.exit(0);
        }

      }catch (Exception E){
        E.printStackTrace();
      }

    }

    public static void main(String[] args) {
      new Signup3("");
    }
}
```

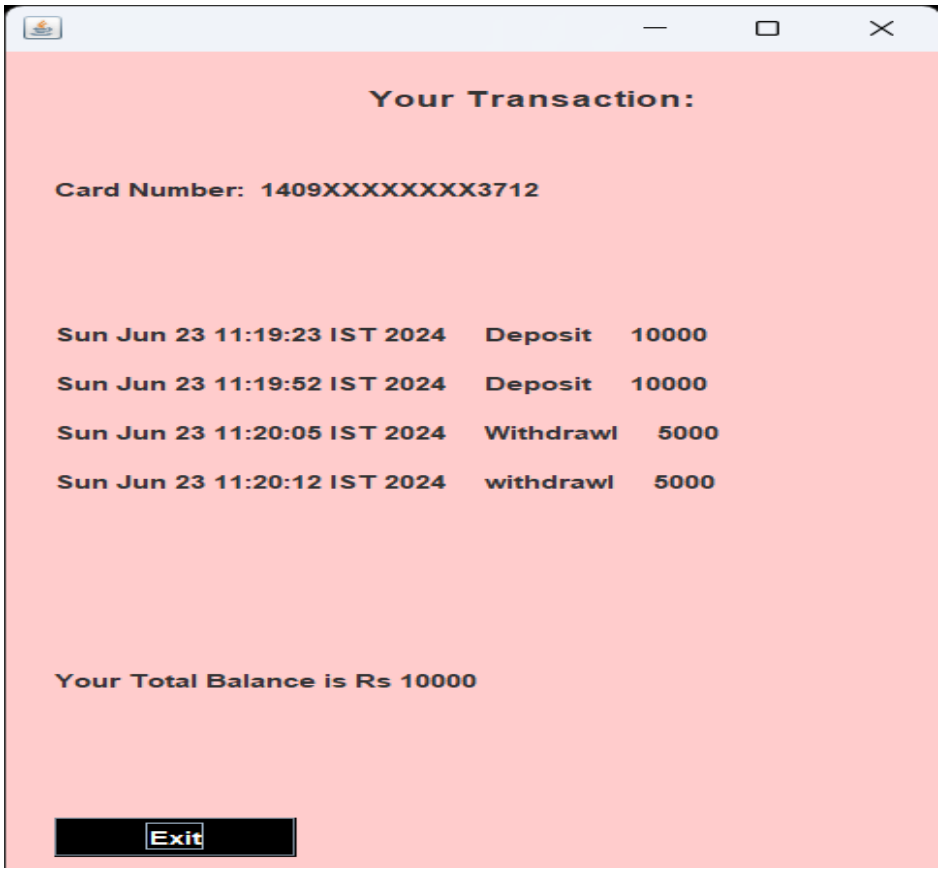
Output Screen of Project

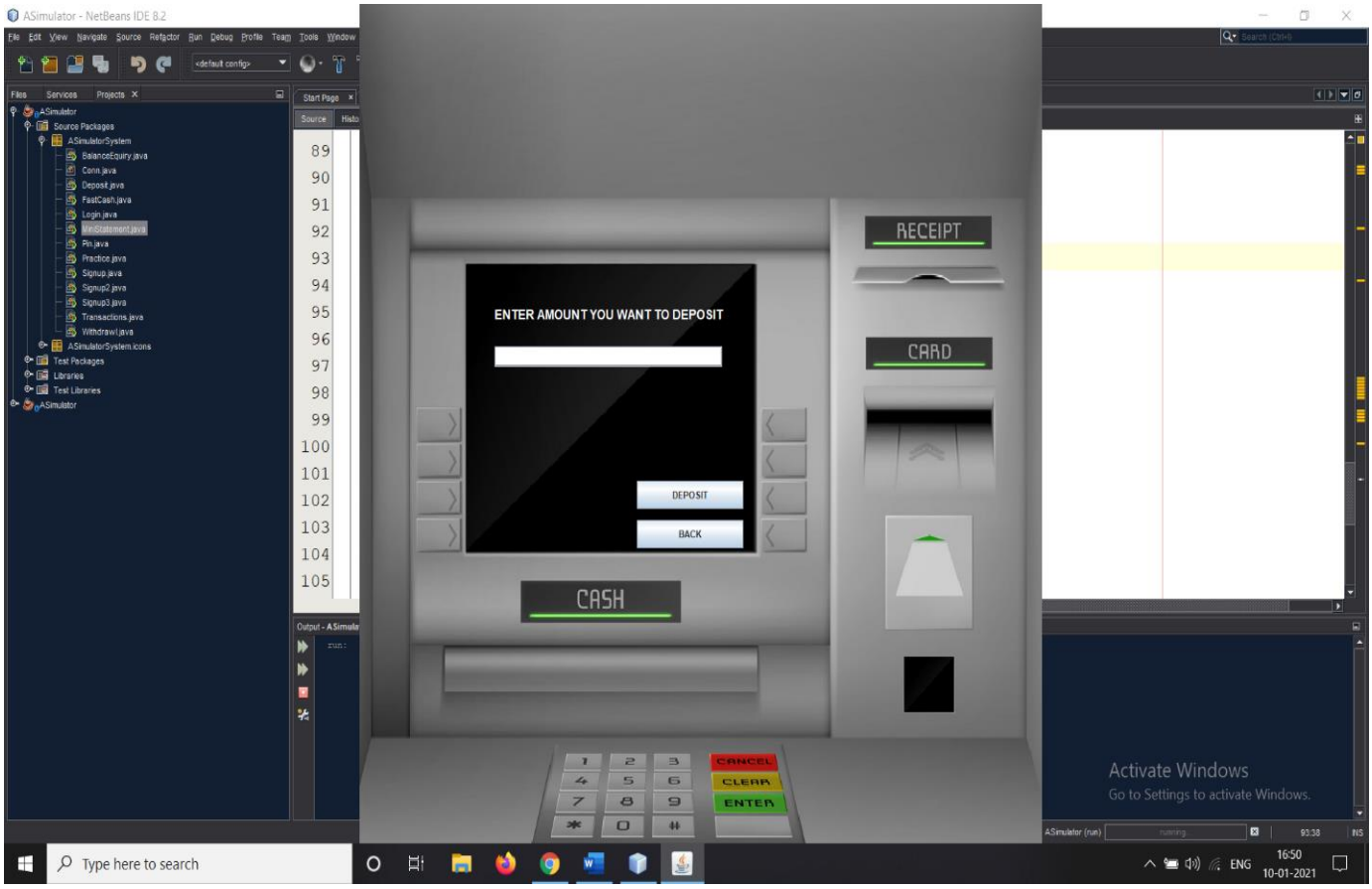


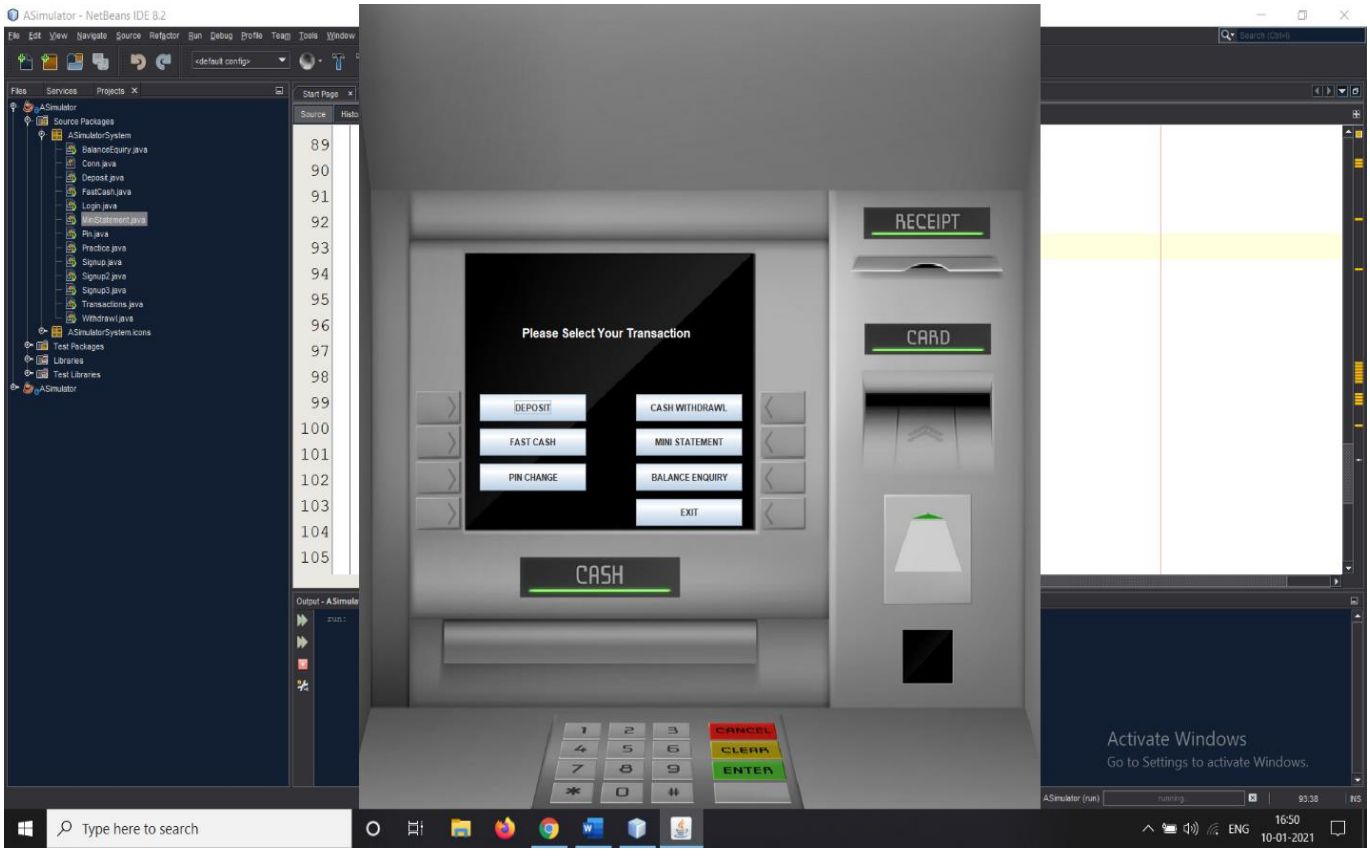

NEW ACCOUNT APPLICATION FORM
APPLICATION FORM NO. 1476
Page 1: Personal Details
Name:
Father's Name:
Date of Birth:
Gender:
Male
Female
Email Address:
Marital Status:
Married
Unmarried
Other
Address:
City:
Pin Code:
State:
Next


APPLICATION FORM
Form No :
BANK
Page 2 :-
Additonal Details
Religion :
Hindu
Category :
General
Income :
Null
Educational :
Non-Graduate
Occupation :
Salaried
PAN Number :
Aadhar Number :
Senior Citizen :
Yes
No
Existing Account :
Yes
No
Next


Form No :
Page 3:
Account Details
Account Type:
Saving Account
Fixed Deposit Account
Current Account
Recurring Deposit Account
Card Number:
(Your 16-digit Card Number)
XXXX-XXXX-XXXX-4841
(It would appear on atm card/cheque Book and Statements)
PIN:
(4-digit Password)
XXXX
Services Required:
ATM CARD
Internet Banking
Mobile Banking
EMAIL Alerts
Cheque Book
E-Statement
I here by decleares that the above entered details correct to the best of my knlowledge.
Submit
Cancel


Your Transaction:
Card Number: 1409XXXXXXXX3712
Sun Jun 23 11:19:23 IST 2024 Deposit 10000
Sun Jun 23 11:19:52 IST 2024 Deposit 10000
Sun Jun 23 11:20:05 IST 2024 Withdrawl 5000
Sun Jun 23 11:20:12 IST 2024 withdrawl 5000
Your Total Balance is Rs 10000
Exit


ENTER AMOUNT YOU WANT TO DEPOSIT
DEPOSIT
BACK
RECEIPT
CARD
CASH
CANCEL
CLEAR
ENTER
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
16:50
10-01-2021


Please Select Your Transaction
DEPOSIT
CASH WITHDRAWL
FAST CASH
MINI STATEMENT
PIN CHANGE
BALANCE ENQUIRY
EXIT
RECEIPT
CARD
CASH
CANCEL
CLEAR
ENTER
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
16:50
10-01-2021

Conclusion

This project is developed to nurture the needs of a user in a banking sector by embedding all the tasks of transactions taking place in a bank. Future version of this project will still be much enhanced than the current version. Writing and depositing checks are perhaps the most fundamental ways to move money in and out of a checking account, but advancements in technology have added ATM and debit card transactions. All banks have rules about how long it takes to access your deposits, how many debit card transactions you're allowed in a day, and how much cash you can withdraw from an ATM. Access to the balance in your checking account can also be limited by businesses that place holds on your funds.

Banks are providing internet banking services also so that the customers can be attracted. By asking the bank employs we came to know that maximum numbers of internet bank account holders are youth and business man. Online banking is an innovative tool that is fast becoming a necessity. It is a successful strategic weapon for banks to remain profitable in a volatile and competitive marketplace of today. If proper training should be given to customer by the bank employs to open an account will be beneficial secondly the website should be made friendlier from where the customers can directly make and access their accounts.
Thus, the Bank Management System it is developed and executed successfully.

Reference

1. Fundamentals of database systems by (**Elmasri Navathe, 2000**),
Website:https://archive.org/stream/FundamentalsOfDatabaseSystemselmasrinavathe# page/n51/mode/2up,
Page: From **52** to more.

1. Article: **Online banking,** Website: https://en.wikipedia.org/wiki/Online_banking **June 29, 2015, 12.30 am.**
2. Online Bank Account Management System
Website: http://www.slideshare.net (Collect some info for report documents, **20142015**)
3. Learning **MYSQL Workbench,** Java core (SWING & AWT), Website: http://www.w3schools.comm, **2014-2015**
4. Java core and MySQL video tutorials, Oct **2014-2015**
Website: http://www.freebanglatutorial.com, http://www.youtube.com
5. Veneeva, V. (2006), "E-Banking (Online Banking) and Its Role in Today's Society", Ezine articles, **June 30, 2015**
, (**May 10, 2015**)