

A

PROJECT REPORT FORM

***“*ATM SIMULATOR MACHINE”**

Submitted to

**SHIVAJI UNIVERSITY, KOLHAPUR**

FOR THE PARTIAL FULLFILLMENT OF

**BACHLOUR OF SCIENCEC ( COMPUTER SCIENCE)**

**(B.SC .-III SEMESTER - VI)**

BY

**Mr. PRATHAMESH MANOHAR MANE**

**Mr. RITESH MARUTI MALVEKAR**

UNDER THE GUIDENCE OF

**Prof. Reshma D. Niduni**

THROUGH

**THE PRINCIPAL**

**Prin. Dr. Nila G. Joshi**

**D.R.MANE MAHAVIDYALAYA, KAGAL**

**2023-2024**



**D.R.MANE MAHAVIDYALAYA, KAGAL**

**CERTIFICATE**

This is to certify that **Mr. Prathamesh Manohar Mane& Mr. Ritesh Maruti Malvekar** have satisfactorily completed the project work on **“ATM SIMULATOR MACHINE”** under the guidance of **Prof. S.P.PATIL** for the partial fulfilment of B.S.C-III (Sem-VI) submitted to **Shivaji University, Kolhapur** during the academic year 2023-2024.

**Place: Kagal**

**Date:**

**External Examiner External Examiner**

**Mr. Shardul P.Patil Dr. P .N. Chougale**

**(H.O.D) (Principal)**



**D.R.MANE MAHAVIDYALAYA,KAGAL**

**GUIDE’S CERTIFICATE**

This is certify that **Mr**.**Prathamesh Manohar Mane & Mr. Ritesh Maruti Malvekar**have satifactional completed the project work on **“ATM SIMULATOR MACHINE”** for the partial fulfilment of BSC–III (sem-VI) submitted to **Shivaji University Kolhapur** during the academic year 2023-2024 under my guidance. To the best of my knowledge and belief the matter presented by then are original and not copied from any source. Also, this report has not submitted earlier for the award of degree of Shivaji University, Kolhapur.

**Place: Kolhapur Prof. S. P. PATIL**

**Date: (project Guide)**

**DECLARATION**

We hereby declare that the project report entitled **“EVENT MANAGEMENT SYSTEM”,** submitted in partial fulfilment of the requirements for the degree BSC-III Semester –VI of **Shivaji University** is our original work and not submitted for the award of any other degree, diploma, fellowship or any other similar titles or prizes.

|  |  |  |
| --- | --- | --- |
| **S.R. No** | **Name** | **Sign** |
| **1** | **Prathamesh Manohar Mane** |  |
| **2** | **Ritesh Maruti Malvekar** |  |

**Place:-Kagal**

**Date:-**

**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 **Prof. Reshma D. Niduni** 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

|  |  |  |
| --- | --- | --- |
| **S.R. No** | **Name** | **Sign** |
| **1** | **Prathamesh Manohar Mane** |  |
| **2** | **Ritesh Maruti Malvekar** |  |

**Index**

|  |  |  |
| --- | --- | --- |
| **Sr.No** | **Particulars** | **Page No.** |

|  |  |  |
| --- | --- | --- |
| **1.** | **Introduction to project**   * Introduction * Existing System * Need & Scope of system * Organization Profile |  |
| **2.** | **Proposed System**   * Objectives * Requirement Engineering. * Requirement Gathering * Fact Finding Techniques * Requirement Analysis   -Feasibility Study  -Platform Selection |  |
| **3.** | **System Diagram**   * DFD * ERD * System Requirement * Hardware & Software Requirement |  |
| **4.** | **System Design**   * Form Design |  |
| **5.** | **Source Code** |  |
| **6.** | **Outputs**   * Output Reports * Software Testing * User Guideline |  |
| **7.** | **Conclusion**   * Conclusion * Bibliography |  |

# INTRODUCTION

**ABOUT PROJECT**

# INTRODUCTION ABOUT PROJECT

### INTRODUCTION

In this project, we've created a virtual ATM experience where users can practice different banking transactions, just like using a real ATM machine.

Our goal with this project is to provide a hands-on learning opportunity for users to understand how ATMs work and become more comfortable with banking transactions. Whether you're new to banking or want to improve your skills, this simulation is designed to help you learn in a fun and interactive way.

We've included clear instructions and guidance to help users navigate the simulation easily. Our hope is that this project enhances your understanding of banking and gives you more confidence in managing your finances.

**EXISTING SYSTEM**

Throughout this project, users can:

* **Withdraw Cash**: Practice taking out money from their account.
* **Deposit Funds:** Learn how to put money into their account, either by cash or check.
* **Check Balances:** See how much money they have in their account.
* **Transfer Funds:** Practice moving money between different accounts.

### NEED AND SCOPE OF SYSTEM

Need of System:

1. Education and Training: Many people are unfamiliar with how to use ATMs, especially those who have never used one before or are transitioning to a new banking system. A simulation system provides a risk-free environment for users to learn and practice ATM transactions before attempting them in real life. This is particularly valuable for individuals who may feel intimidated or anxious about using an ATM for the first time.
2. Accessibility: Not everyone has easy access to physical ATMs, especially in remote or underserved areas. A simulation system allows individuals to learn about and experience ATM transactions without needing to visit a physical location, making banking services more accessible to a wider population.
3. Safety and Security Training: ATM fraud and security breaches are a concern for both banks and customers. A simulation system can be used to educate users about common security threats, such as skimming and phishing, and teach them how to recognize and avoid fraudulent activities. This helps to enhance overall awareness and security in the banking ecosystem.
4. Cost-Effective Development: Developing and maintaining physical ATM machines can be expensive. A simulation system provides a cost-effective alternative for banks and financial institutions to train staff, test new features, and conduct user research without the need for physical infrastructure.
5. Continuous Improvement: As technology evolves, so do ATM systems. A simulation system allows developers to prototype and experiment with new features and functionalities in a controlled environment before rolling them out to real-world ATMs. This iterative approach facilitates continuous improvement and innovation in ATM design and user experience.

SCOPE:

The scope of our ATM simulation system project includes designing and developing a user-friendly virtual environment that replicates the functionalities of a real ATM. This system will allow users to practice basic banking transactions such as withdrawing cash, depositing funds, checking balances, and transferring money between accounts. Additionally, the project will focus on implementing robust security measures, error handling mechanisms, and customization options to enhance user experience and ensure the reliability of the simulation.

### Organization Profile

Introduction:

In today's digital age, Automated Teller Machines (ATMs) have become ubiquitous, offering convenient access to banking services. However, many individuals, especially newcomers to banking or those in underserved areas, may lack the familiarity or confidence to use ATMs effectively. Recognizing this need, our project aims to bridge the gap by developing a virtual ATM simulation system.

**PROPOSED SYSTEM**

###### PROPOSED SYSTEM

Our proposed ATM simulation system will feature an intuitive user interface mimicking the layout of physical ATMs, enabling users to conduct various banking transactions such as withdrawals, deposits, balance inquiries, and transfers. To ensure security, the system will implement robust authentication and encryption mechanisms, while comprehensive error handling will guide users through transaction processes. Customization options will allow users to tailor their experience, and stringent data management practices will safeguard user information. We'll provide extensive documentation and support to assist users, aiming to deliver a secure, user-friendly platform for learning and practicing essential banking skills.

###### OBJECTIVES

* Transaction Management: Implement a system that facilitates searching and tracking transactions based on various factors, such as transaction type (withdrawal, deposit, transfer), date, and amount, providing a comprehensive overview of financial activities.
* Event Management Integration: Integrate an online event management system that allows users to manage cash details related to payment, billing, and other transactions. Enable functionalities for online payment processing, including credit card and QR code transactions.
* Customer Information Management: Develop a robust customer information management module to store and retrieve customer details efficiently. This includes personal information, transaction history, and preferences to enhance customer service and experience.
* Event Information Display: Implement features that showcase detailed information and descriptions of events and services provided by the ATM system. This ensures transparency and clarity for users regarding the available functionalities.
* Efficiency Improvement: Strive to increase the efficiency of managing the system's resources, with a particular focus on optimizing shoebox management, ensuring that the ATM system operates seamlessly and meets user needs promptly.
* Payment Transaction Monitoring: Develop a secure and effective mechanism to monitor payment transactions, ensuring accuracy and security in financial operations. Implement real-time tracking and notifications for users to stay informed about their financial activities.
* Event Information Management: Create a module to manage event-related information, allowing for easy editing, addition, and updating of records. This ensures the accuracy and relevance of event data, contributing to proper resource management.
* Resource Management Enhancement: Improve the system's resource management capabilities by optimizing the editing, adding, and updating of records. This enhances the overall efficiency and effectiveness of managing event data and related information.
* User-Friendly Interface: Design an intuitive and user-friendly interface to ensure ease of use for both customers and administrators. This includes providing a seamless experience for navigating through various features and functionalities.
* Security Measures: Implement robust security measures to safeguard customer information, financial transactions, and system integrity. Incorporate encryption, authentication, and authorization protocols to ensure a secure environment for users.

###### REQUIREMENT GATHERING

Requirement gathering is the process of generating a list of functional, technical and systematic requirements from several project stakeholders, such as clients, IT staff, product users or vendors. This list may likely include features, activities and tasks for a team to execute in order to achieve the goals of a project. Usually, there are two types of requirements to consider: functional and non-functional. Functional requirements include the information, interactions and processes that a client requests. Non-functional requirements are other technical and operational aspects of a project.

Consider starting the process of requirements gathering at the beginning of a project to help ensure effective planning and management. Here are some aspects to consider as you start requirements gathering:

**Timeline:**

This entails estimating the overall length and schedule of a project. This can help you effectively plan for its requirements over the course of an entire project, ensuring that you're prepared throughout its duration.

**People:**

Consider who you want to include in a project and what their various roles might be. Try to determine their individual requirements, as well as how they might collaborate with the entire team.

**Goals:**

Identifying primary goals early can help you determine a project's requirements and ensure that the project specifically focuses on executing these objectives.

###### Fact Finding Techniques

**Fact Finding Techniques**

* **Fact Finding Techniques:-**

In Order to gather an analysed the relevant information following methods or adopted.

1. Observation:-

Find hand information about various activities carried out can be studied through observation. This method used to solve how documents were handled, how processes were carried out and whether those steps were actually followed Thus the current scenario was understood using observation.

1. Record Review: -

The Event Management record maintained to give clear idea as to how actual process carried also management working on whole.

1. Document Searching:-

Various documents which are handled these used formats where studied relation with system, information analysed

**Requirement Analysis**

Requirement analysis is a software engineering task that bridges the gap between system level requirements engineering & software design. Requirement engineering activities results in the specification of software’s operational characteristics include software’s interface with other system elements, and establish constraints that software must meet. Requirement analysis provides the software designer with representation of information, function, & behaviour that can be translated to data, architectural, interface & component-level designs. Finally the requirement specification provides the developer & the customer with the means to access quality ones software.

**Software requirement analysis may be divided into five area of efforts**:

* + Problem recognition
  + Evaluation & synthesis
  + Modelling
  + Specification
  + Review

Initially, the analyst studies the systems specification & the software project plan. It is important to understand software in a system context & to review the software scope that was used to generate planning estimates. Next communication for analysis must be established so that the problem recognition is ensured. The goal is recognition of the basic problem elements as perceived by customers or users.

Problem evaluation & solution synthesis is the next major area of effort for analysis. The analyst must all externally observable data objects, evaluate the flow & content of information, define & elaborate all software functions, understand software behaviour in the context of events that affects the system, establish system interface characteristics, & uncover additional design constraints. Each of these tasks serves to desirable the problem so that an overall approach or solution may be synthesized.

**Feasibility Study**

**Technical Feasibility:**

Firstly, we need to ensure that our SuperStore Insight project is technically possible. This means checking if we have the right technology and skills to make it happen. Luckily, the hardware we need is already available in the store. If there are any missing pieces, we'll figure out how to get them.

**Social Feasibility:**

People usually resist change, but in this case, we're introducing a system to make everyone's work easier. SuperStore Insight aims to reduce the workload by helping users calculate salaries and deductions, generate reports with fewer errors, and generally make operations smoother. So, despite initial resistance, the system is designed to be user-friendly and beneficial.

**Economic Feasibility:**

We need to analyze the costs and benefits of our SuperStore Insight project. This involves looking at how much it will cost versus the benefits it will bring. This analysis helps decision-makers see if the project makes economic sense and provides positive benefits to the store. It's like making sure the investment is worthwhile and credible.

**Scheduling Feasibility:**

One of the most crucial aspects of our project's success is getting it done on time. Scheduling feasibility means estimating how much time it will take to complete the SuperStore Insight project. We want to ensure that everything runs smoothly and on schedule. This way, we avoid any delays that could potentially cause the project to fail.

Once we look into these areas, we'll also check for any constraints the SuperStore Insight project might face. These could be internal, like technical or budget constraints, corporate constraints related to finance or marketing, and external constraints like dealing with laws and regulations or environmental factors. By understanding and addressing these constraints, we'll increase the chances of our project's success.

###### Platform Selection

* + Front End Tool [C#]:

**Java NetBeans as a Front-End Tool:**

**Similar to how we've utilized C# as a front-end tool, Java NetBeans serves as a powerful and user-friendly platform for creating applications. Java NetBeans offers a clear and intuitive approach to coding, enhancing our understanding and making the coding process more accessible.**

**In the context of windows applications, Java NetBeans proves to be an efficient choice. It provides a robust set of tools that simplifies the development of windows applications, just like C#. The platform offers a comprehensive toolkit, making it quick and straightforward to create code tailored for Windows applications.**

**Java, the programming language underlying NetBeans, is renowned for its versatility. With Java NetBeans, we have the capability to design and implement programs, including Graphical User Interfaces (GUIs), to fulfill specific tasks. This aligns with the dual nature of computer programs, which are broadly categorized as packaged or custom solutions.**

**In the realm of window programs developed with Java NetBeans, a notable advantage lies in the high degree of user interaction facilitated by graphical elements. These elements, seamlessly integrated into the application's window, enhance the overall user experience and engagement. The Java NetBeans environment ensures that creating GUIs for windows applications is not only efficient but also offers flexibility and a wide range of design options.**

**To summarize, Java NetBeans, as both a platform and a programming language, provides a robust and versatile foundation for developing windows applications. Its user-friendly nature, coupled with a rich set of tools, makes it an excellent choice for designing and implementing programs with graphical interfaces, aligning with the specific needs and objectives of our project.What is DBMS?**

Database is collection ofrelated data.A database management System is collection of program that enables user to create andmaintain a database.

The database is haven a general purpose software system that facilitates the process of defining construction and manipulationdatabase for various applications. Defining the database means toCreate data. Constructing the database to store the data and manipulation of database means to retrieve and update from datato database.

It is a coherent collection of a data with some inheritancemeaning, designed, built and populated with data for a specificpurpose. A database stored data that is useful to us. This data I onlya part of the entire data available in the world around us.

To be able to success fully designed and maintained databasewe have to do the following:

* Identify which parts of the words data is of interested to us.
* Identify what’s specific in the part of the words data are to interestto us.
* Identifya relationship between the objects.

Hence, the objects their attributes and relationship between themthat are of interest to use are stored in the database. That is adesigned, built and populated with data with a specific purpose.

Characteristics of Database Management System:

1. It keeps complex relation between data.
2. Keeps a tight control of data redundancy.
3. Enforces user defined rules to insure the integrity of table data.
4. Ensures that data can be shared across applications.
5. Enforces data access authorization.
6. Has and 2utornatic intelligent backup and recovery procedure for Data.
7. Have different interfaces via which user can manipulated.
8. **SQL Server**

##### Backend Tool:

This is a client-server database management system developed byMicrosoft for large scale database. It comes into RDBMS (RelationalDatabase Management System) category. It works based on SQL (StructuredQuery Language) a latest and advanced database language.

A user need to login to this software system in-order to manage the database. SQL server allows us to logon in two ways.

* + Windows authentication.
  + SQL Server authentication.

In windows authentication, system recognizes the user through his/herwindows login credentials. That means, if a user logs into a windowsoperating system, he/she becomes an authorized user to use all resources thataccept windows authentication and available in the system. In SQL server authentication, user will be provided a user id and password. A user from anySystem in the network can use the user id and password to connect to thedatabase server.

**Features of SQL Server:-**

* + License cost is very lower than any RDBMS Systems.
  + Runs under Windows all windows server (NT/2000/2003)
  + Runs under client versions Windows 95/98/XP…(to install services, we need to install MSDE – Microsoft SQL Desktop Engine)
  + Supports data replication
  + Supports Data Marts and Data Warehouses
  + Provide OLDP service
  + E Data transmission services enable easy exchange of data
  + Data transmission services enable easy exchange of data

**DFD**

**(Data Flow Diagram)**

**DFD:**

**ERD**

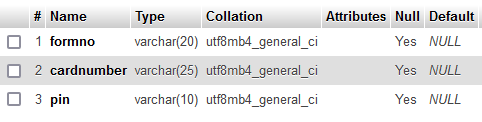
**(Entity Relationship Diagram)**

**ERD:**

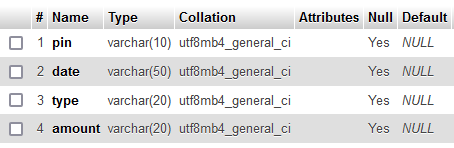
**SYSTEM DESIGN**

**DATABASE DESIGN**

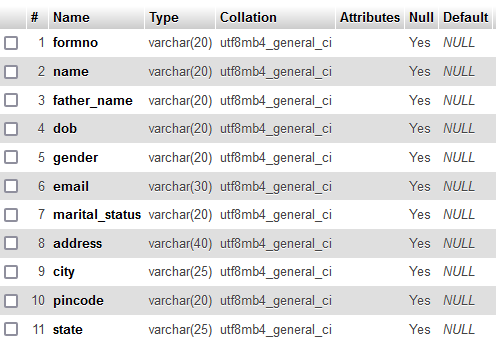
**LOGIN:**

****

**BANK DETAILS:**

****

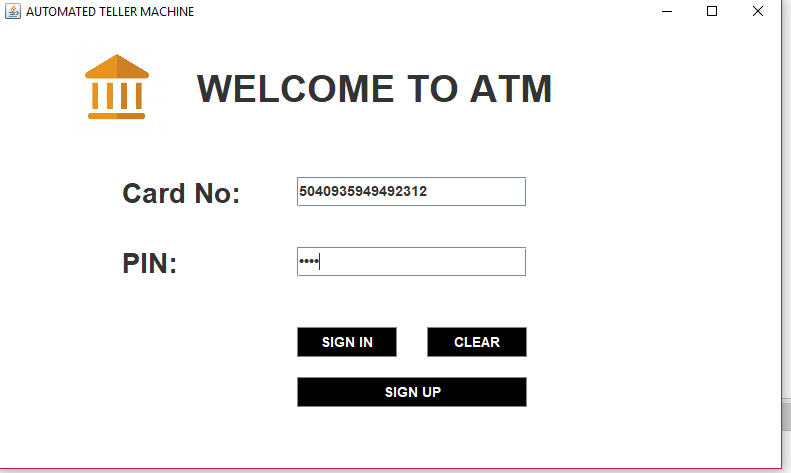
**APPLICATION:**

****

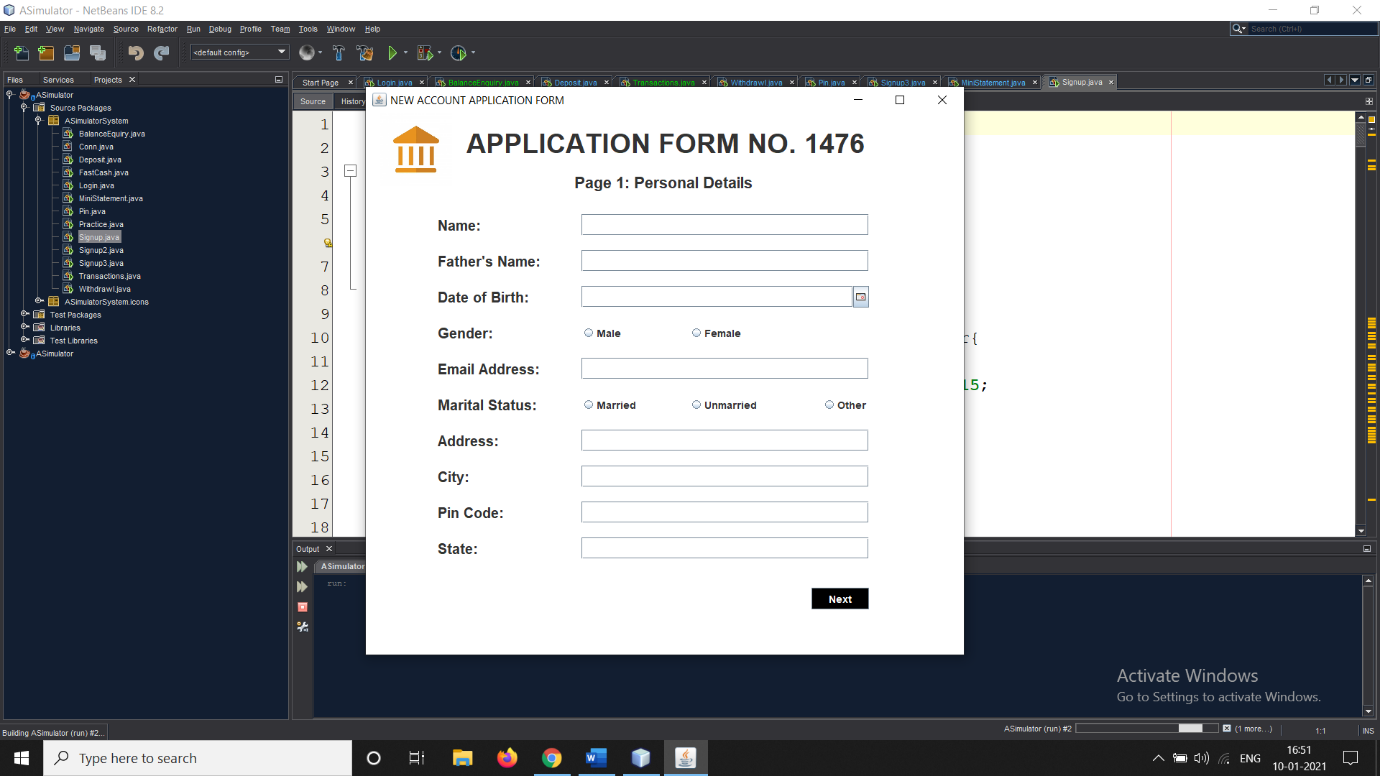
# OUTPUT

**SCREENSHOTS**

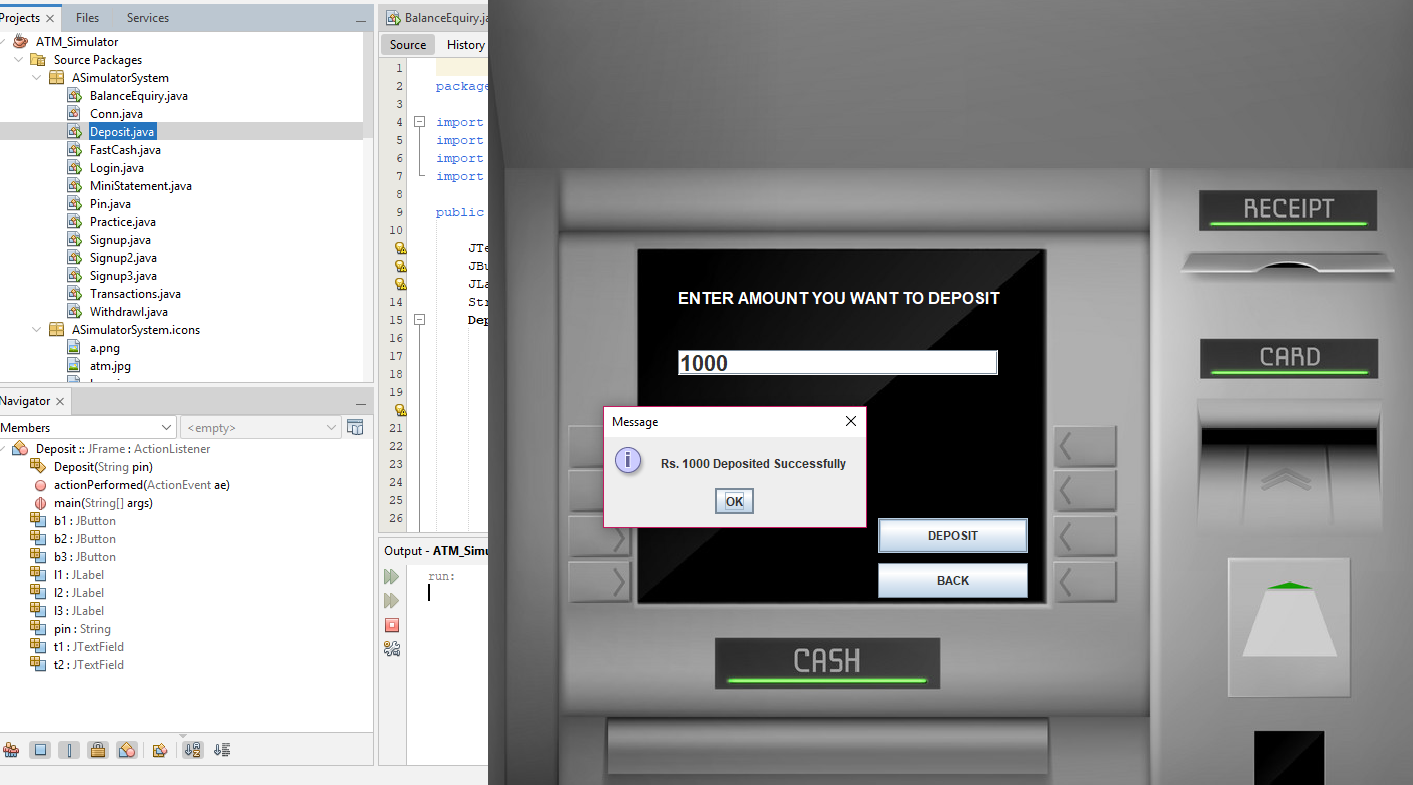
**LOGIN-**



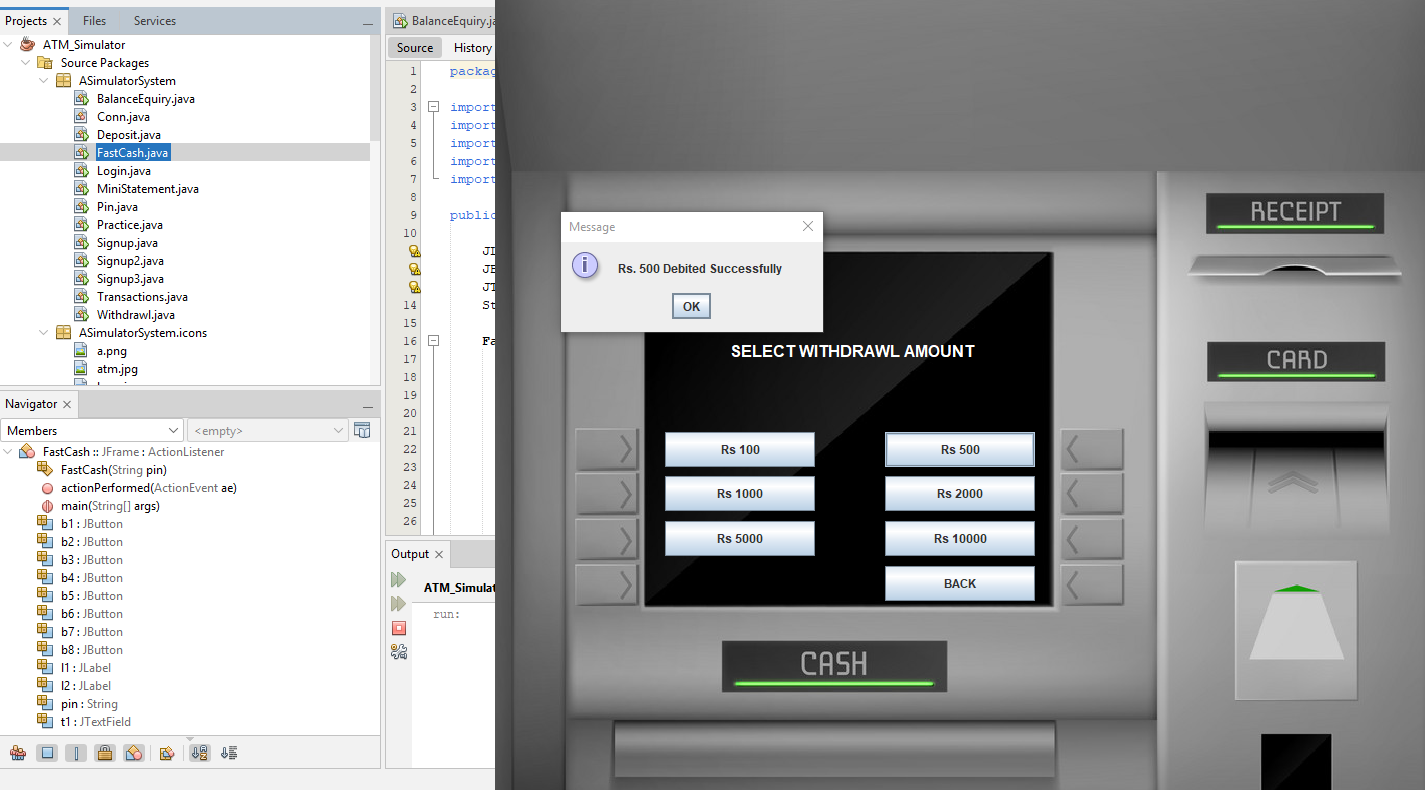
**APPLICATION FROM**



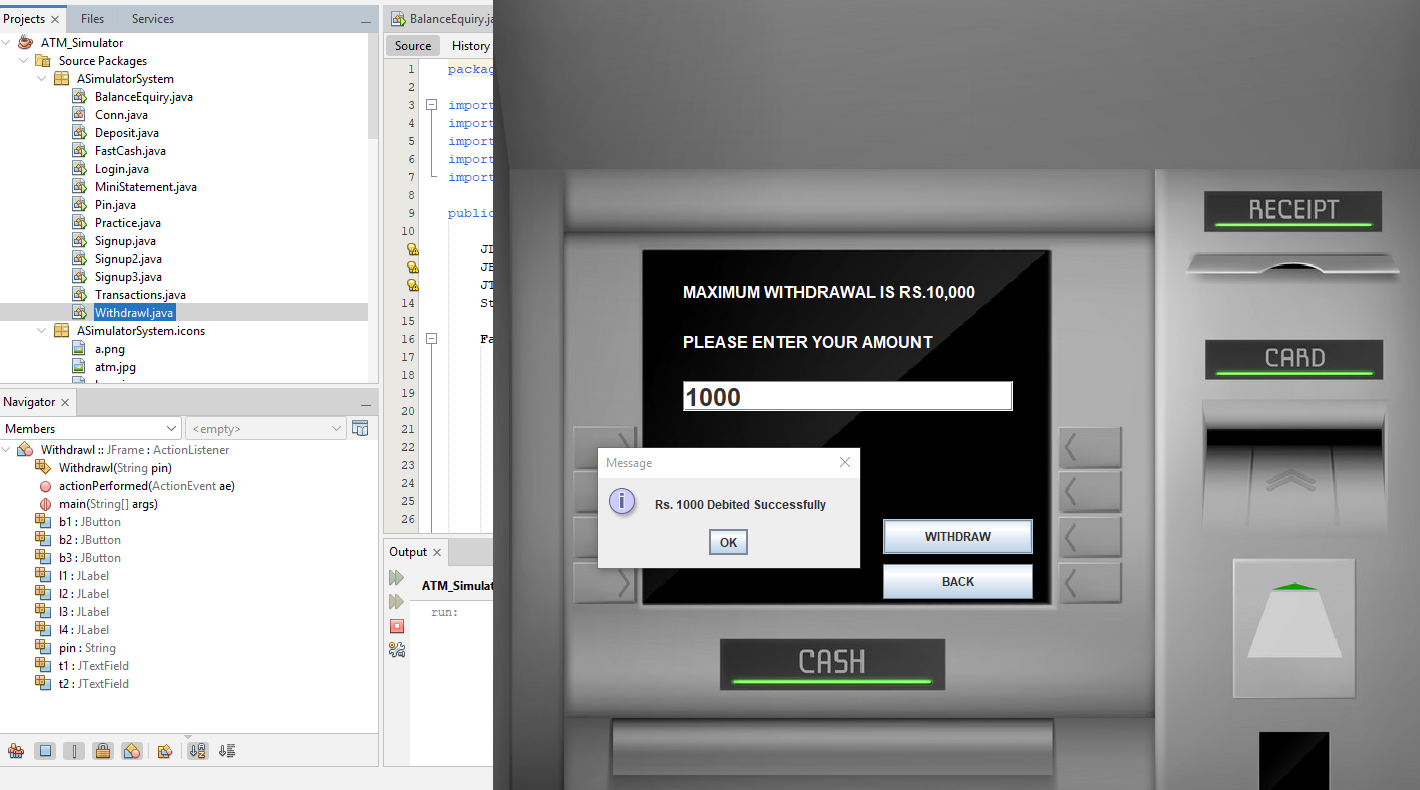
**DEPOSITE**

****

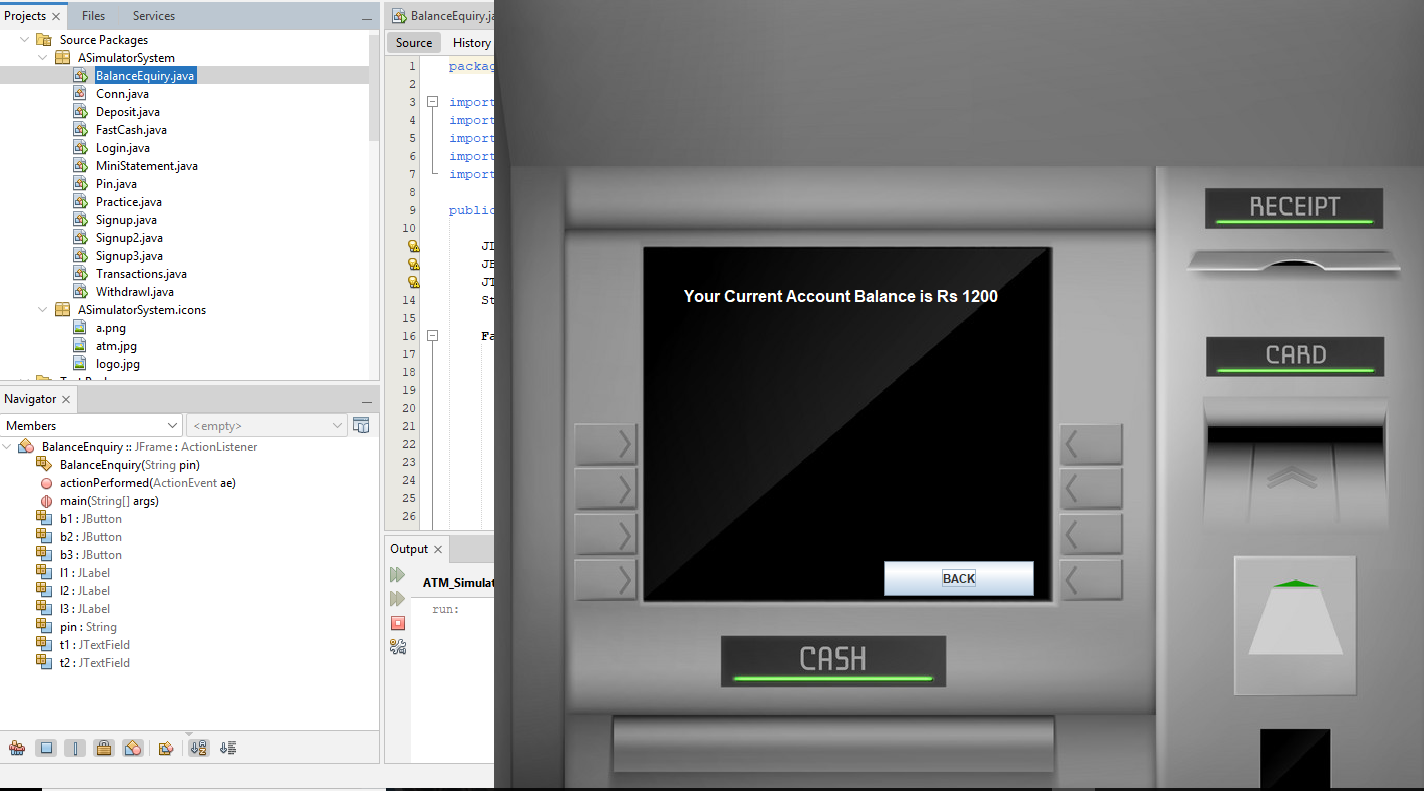
### FAST CASH



**WITH DRAWAL**

****

**ACCOUNT BALANCE**

****

**SOURCE CODE**

**Code:**[C:\Users\Piyush Jadhav\Downloads\deref\http:\stackoverflow.com\questions\3937513\javascript-validation-for-empty-input-field](file:///C:\Users\Piyush%20Jadhav\Downloads\deref\http:\stackoverflow.com\questions\3937513\javascript-validation-for-empty-input-field)

**Deposit.java**

package ASimulatorSystem;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.\*;

public class Deposit extends JFrame implements ActionListener{

JTextField t1,t2;

JButton b1,b2,b3;

JLabel l1,l2,l3;

String pin;

Deposit(String pin){

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("ASimulatorSystem/icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(0, 0, 960, 1080);

add(l3);

l1 = new JLabel("ENTER AMOUNT YOU WANT TO DEPOSIT");

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

t1 = new JTextField();

t1.setFont(new Font("Raleway", Font.BOLD, 22));

b1 = new JButton("DEPOSIT");

b2 = new JButton("BACK");

setLayout(null);

l1.setBounds(190,350,400,35);

l3.add(l1);

t1.setBounds(190,420,320,25);

l3.add(t1);

b1.setBounds(390,588,150,35);

l3.add(b1);

b2.setBounds(390,633,150,35);

l3.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);

setSize(960,1080);

setUndecorated(true);

setLocation(500,0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

try{

String amount = t1.getText();

Date date = new Date();

if(ae.getSource()==b1){

if(t1.getText().equals("")){

JOptionPane.showMessageDialog(null, "Please enter the Amount to you want to Deposit");

}else{

Conn c1 = new Conn();

c1.s.executeUpdate("insert into bank values('"+pin+"', '"+date+"', 'Deposit', '"+amount+"')");

JOptionPane.showMessageDialog(null, "Rs. "+amount+" Deposited Successfully");

setVisible(false);

new Transactions(pin).setVisible(true);

}

}else if(ae.getSource()==b2){

setVisible(false);

new Transactions(pin).setVisible(true);

}

}catch(Exception e){

e.printStackTrace();

}

}

public static void main(String[] args){

new Deposit("").setVisible(true);

}

}

**Signup.java**

package ASimulatorSystem;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import com.toedter.calendar.JDateChooser;

import java.util.\*;

public class Signup extends JFrame implements ActionListener{

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13,l14,l15;

JTextField t1,t2,t3,t4,t5,t6,t7;

JRadioButton r1,r2,r3,r4,r5;

JButton b;

JDateChooser dateChooser;

Random ran = new Random();

long first4 = (ran.nextLong() % 9000L) + 1000L;

String first = "" + Math.abs(first4);

Signup(){

setTitle("NEW ACCOUNT APPLICATION FORM");

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("ASimulatorSystem/icons/logo.jpg"));

Image i2 = i1.getImage().getScaledInstance(100, 100, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l11 = new JLabel(i3);

l11.setBounds(20, 0, 100, 100);

add(l11);

l1 = new JLabel("APPLICATION FORM NO. "+first);

l1.setFont(new Font("Raleway", Font.BOLD, 38));

l2 = new JLabel("Page 1: Personal Details");

l2.setFont(new Font("Raleway", Font.BOLD, 22));

l3 = new JLabel("Name:");

l3.setFont(new Font("Raleway", Font.BOLD, 20));

l4 = new JLabel("Father's Name:");

l4.setFont(new Font("Raleway", Font.BOLD, 20));

l5 = new JLabel("Date of Birth:");

l5.setFont(new Font("Raleway", Font.BOLD, 20));

l6 = new JLabel("Gender:");

l6.setFont(new Font("Raleway", Font.BOLD, 20));

l7 = new JLabel("Email Address:");

l7.setFont(new Font("Raleway", Font.BOLD, 20));

l8 = new JLabel("Marital Status:");

l8.setFont(new Font("Raleway", Font.BOLD, 20));

l9 = new JLabel("Address:");

l9.setFont(new Font("Raleway", Font.BOLD, 20));

l10 = new JLabel("City:");

l10.setFont(new Font("Raleway", Font.BOLD, 20));

l11 = new JLabel("Pin Code:");

l11.setFont(new Font("Raleway", Font.BOLD, 20));

l12 = new JLabel("State:");

l12.setFont(new Font("Raleway", Font.BOLD, 20));

l13 = new JLabel("Date");

l13.setFont(new Font("Raleway", Font.BOLD, 14));

l14 = new JLabel("Month");

l14.setFont(new Font("Raleway", Font.BOLD, 14));

l15 = new JLabel("Year");

l15.setFont(new Font("Raleway", Font.BOLD, 14));

t1 = new JTextField();

t1.setFont(new Font("Raleway", Font.BOLD, 14));

t2 = new JTextField();

t2.setFont(new Font("Raleway", Font.BOLD, 14));

t3 = new JTextField();

t3.setFont(new Font("Raleway", Font.BOLD, 14));

t4 = new JTextField();

t4.setFont(new Font("Raleway", Font.BOLD, 14));

t5 = new JTextField();

t5.setFont(new Font("Raleway", Font.BOLD, 14));

t6 = new JTextField();

t6.setFont(new Font("Raleway", Font.BOLD, 14));

t7 = new JTextField();

t7.setFont(new Font("Raleway", Font.BOLD, 14));

b = new JButton("Next");

b.setFont(new Font("Raleway", Font.BOLD, 14));

b.setBackground(Color.BLACK);

b.setForeground(Color.WHITE);

r1 = new JRadioButton("Male");

r1.setFont(new Font("Raleway", Font.BOLD, 14));

r1.setBackground(Color.WHITE);

r2 = new JRadioButton("Female");

r2.setFont(new Font("Raleway", Font.BOLD, 14));

r2.setBackground(Color.WHITE);

ButtonGroup groupgender = new ButtonGroup();

groupgender.add(r1);

groupgender.add(r2);

r3 = new JRadioButton("Married");

r3.setFont(new Font("Raleway", Font.BOLD, 14));

r3.setBackground(Color.WHITE);

r4 = new JRadioButton("Unmarried");

r4.setFont(new Font("Raleway", Font.BOLD, 14));

r4.setBackground(Color.WHITE);

r5 = new JRadioButton("Other");

r5.setFont(new Font("Raleway", Font.BOLD, 14));

r5.setBackground(Color.WHITE);

ButtonGroup groupstatus = new ButtonGroup();

groupstatus.add(r3);

groupstatus.add(r4);

groupstatus.add(r5);

dateChooser = new JDateChooser();

//dateChooser.setBorder(new LineBorder(new Color(0, 0, 0), 1, true));

dateChooser.setForeground(new Color(105, 105, 105));

dateChooser.setBounds(137, 337, 200, 29);

add(dateChooser);

setLayout(null);

l1.setBounds(140,20,600,40);

add(l1);

l2.setBounds(290,80,600,30);

add(l2);

l3.setBounds(100,140,100,30);

add(l3);

t1.setBounds(300,140,400,30);

add(t1);

l4.setBounds(100,190,200,30);

add(l4);

t2.setBounds(300,190,400,30);

add(t2);

l5.setBounds(100,240,200,30);

add(l5);

dateChooser.setBounds(300, 240, 400, 30);

l6.setBounds(100,290,200,30);

add(l6);

r1.setBounds(300,290,60,30);

add(r1);

r2.setBounds(450,290,90,30);

add(r2);

l7.setBounds(100,340,200,30);

add(l7);

t3.setBounds(300,340,400,30);

add(t3);

l8.setBounds(100,390,200,30);

add(l8);

r3.setBounds(300,390,100,30);

add(r3);

r4.setBounds(450,390,100,30);

add(r4);

r5.setBounds(635,390,100,30);

add(r5);

l9.setBounds(100,440,200,30);

add(l9);

t4.setBounds(300,440,400,30);

add(t4);

l10.setBounds(100,490,200,30);

add(l10);

t5.setBounds(300,490,400,30);

add(t5);

l11.setBounds(100,540,200,30);

add(l11);

t6.setBounds(300,540,400,30);

add(t6);

l12.setBounds(100,590,200,30);

add(l12);

t7.setBounds(300,590,400,30);

add(t7);

b.setBounds(620,660,80,30);

add(b);

b.addActionListener(this);

getContentPane().setBackground(Color.WHITE);

setSize(850,800);

setLocation(500,120);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String formno = first;

String name = t1.getText();

String fname = t2.getText();

String dob = ((JTextField) dateChooser.getDateEditor().getUiComponent()).getText();

String gender = null;

if(r1.isSelected()){

gender = "Male";

}else if(r2.isSelected()){

gender = "Female";

}

String email = t3.getText();

String marital = null;

if(r3.isSelected()){

marital = "Married";

}else if(r4.isSelected()){

marital = "Unmarried";

}else if(r5.isSelected()){

marital = "Other";

}

String address = t4.getText();

String city = t5.getText();

String pincode = t6.getText();

String state = t7.getText();

try{

if(t6.getText().equals("")){

JOptionPane.showMessageDialog(null, "Fill all the required fields");

}else{

Conn c1 = new Conn();

String q1 = "insert into signup values('"+formno+"','"+name+"','"+fname+"','"+dob+"','"+gender+"','"+email+"','"+marital+"','"+address+"','"+city+"','"+pincode+"','"+state+"')";

c1.s.executeUpdate(q1);

new Signup2(first).setVisible(true);

setVisible(false);

}

}catch(Exception e){

e.printStackTrace();

}

}

public static void main(String[] args){

new Signup().setVisible(true);

}

}

REPORTS:

### SOFTWARE TESTING

**System Testing:-**

Testing presents an interesting anomaly for the software engineer. During earlier software activities, the engineer attempts to build software from an abstract concept to a tangible product. The engineer creates a series of test cases that are intended to “demolish” the software that has been built, testing is conducted at various levels of the project.

**Module Testing:-**

After doing unit testing, the whole system is tested whether it is working properly or not. The screen together works as per requirement without affecting working of controls. In this we made sure that all the fields that should not be left blank or little by end-user or not.

**Integrated System Testing:-**

All modules are integrated and connected to the main menu and then main module testing is done to check the following: After selecting some option, a proper module should be called. Integrated system is checked to ensure control flow as per the predefined flow of control.

**Validation Testing:-**

At the culmination of integration testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software tests-validation testing-may begin. Validation succeeds when software functions in a manner that can be reasonably expected by the customer.

**System Testing:-**

Software is incorporated with other system element (i.e. Hardware, People, information) and a series of system integration and validation tests are not conducted solely by software engineers.

**SYSTEM REQUIREMENTS**

**SYSTEM REQUIREMENTS**

**Hardware:**

* Hard disk :500GB
* RAM :4.00 GB
* Processor :Intel(R) Core TM i3-8100 CPU @ 3.60GHz
* System type :64 bit
* Operating System : Windows 10

**Software:**

* Front-end : Net Beans
* Back-End : SQL Server

**CONCLUSION**

**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.

**BIBLIOGRAPHY:**

**BIBLIOGRAPHY:**

1.Code for Interview YouTube Channel (https://www.youtube.com/playlist?list=PL5BFcXE899zxVrWaO3Ul6ly2SVJMnJFOr)

2.Online Bank Account Management System

Website: http://www.slideshare.net (Collect some info for report documents)

3.Learning MYSQL, JavaScript, jQuery, PHP, HTML, CSS3,

Website: http://www.w3schools.com

4.PHP and MySQL video tutorials

Website: http://www.freehinditutorial.com, http://www.youtube.com

5. Veneeva, V. (2006), “E-Banking (Online Banking) and Its Role in Today's Society”,

Ezine articles

6.JavaScript validation for empty input field

Website:http://stackoverflow.com/questions/3937513/javascript-validation-for-empty-

input-field ,

7.JavaScript form validation: Validate Password, Validate Email, Validate Phone

Number, http://webcheatsheet.com/javascript/form\_validation.php