ATM SIMULATOR SYSTEM

MINI PROJECT REPORT

By

PRATHAM SHRIVASTAV (RA2211026010366)
SHYAM KUMAR SHAH (RA2211026010371)
ANJALI SATIJA (RA2211026010390)

Under the guidance of

Dr. Maivizhi R

In partial fulfilment for the Course

of

21CSC203P – ADVANCED PROGRAMMING PRACTICE

in **CINTEL**

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

NOVEMBER 2023

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this minor project report for the course **21CSC203P ADVANCED PROGRAMMING PRACTICE** entitled in "**ATM SIMULATOR SYSTEM**" is the bonafide work of **Pratham Shrivastav (RA2211026010366) , Shyam Kumar (RA2211026010371)** and **Anjali Satija (RA2211026010390)** who carried out the work under my supervision.

PROJECT GUIDE

Dr Maivizhi R
Assistant Professor
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur

HEAD OF THE DEPARTMENT

Dr Annie Uthra
Professor & Head
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur

ABSTRACT

The Bank Account Management System is a Java-based application designed to streamline and enhance the functionality of banking operations. This project addresses the essential aspects of managing a person's account, offering a user-friendly interface for various banking tasks. It aims to overcome the limitations of manual systems by providing quick and efficient processes. The focus on requirements definition and management ensures compliance, customer satisfaction, and project efficiency. By incorporating content management and interactive features, the system optimizes transaction time, ultimately boosting overall efficiency and user experience in the banking environment.

ACKNOWLEDGEMENT

We express our heartfelt thanks to our honorable **Vice Chancellor Dr. C. MUTHAMIZHCHELVAN**, for being the beacon in all our endeavors. We would like to express my warmth of gratitude to our **Registrar Dr. S. Ponnusamy,** for his encouragement.

We express our profound gratitude to our **Dean (College of Engineering and Technology) Dr. T. V.Gopal,** for bringing out novelty in all executions.

We would like to express my heartfelt thanks to Chairperson, School of Computing **Dr. Revathi Venkataraman,** for imparting confidence to complete my course project

We wish to express my sincere thanks to **Course Audit Professors Dr. Vadivu. G , Professor, Department of Data Science and Business Systems and Dr. Sasikala. E Professor, Department of Data Science and Business Systems** and **Course Coordinators** for their constant encouragement and support.

We are highly thankful to our my Course project Faculty **Dr. Maivizhi R , Assistant Professor , CINTEL,** for his/her assistance, timely suggestion and guidance throughout the duration of this course project.

We extend my gratitude to our **HOD Dr. Annie Uthra, Professor & Head , CINTEL** and my Departmental colleagues for their Support.

Finally, we thank our parents and friends near and dear ones who directly and indirectly contributed to the successful completion of our project. Above all, I thank the almighty for showering his blessings on me to complete my Course project.

TABLE OF CONTENTS

CHAPTER NO	CONTENTS	PAGE NO
1	**INTRODUCTION**	**1**
	1.1 Motivation	**1**
	1.2 Objective	**1**
	1.3 Problem Statement	**2**
	1.4 Challenges	**2**
2	**LITERATURE SURVEY**	**3**
3	**REQUIREMENT ANALYSIS**	**4**
4	**ARCHITECTURE & DESIGN**	**5**
5	**IMPLEMENTATION**	**8**
6	**EXPERIMENT RESULTS & ANALYSIS**	**13**
7	**CONCLUSION**	**15**
8	**REFERENCES**	**16**

1. INTRODUCTION

In the rapidly evolving landscape of banking, the transition from traditional brick-and-mortar structures to virtual platforms is imperative. The motivation behind the "Bank Account Management System" project stems from the desire to embrace this digital transformation. The aim is to empower customers with the convenience of executing basic banking transactions from the comfort of their homes or offices using their PCs or laptops. Internet Banking serves as the catalyst for this shift, providing a click-and-portal model that transcends physical branch limitations. The project envisions a future where banking is not confined to branches, and customers can engage in round-the-clock global transactions..

1.1 Motivation:

In the rapidly evolving landscape of banking, the transition from traditional brick-and-mortar structures to virtual platforms is imperative. The motivation behind the "Bank Account Management System" project stems from the desire to embrace this digital transformation. The aim is to empower customers with the convenience of executing basic banking transactions from the comfort of their homes or offices using their PCs or laptops. Internet Banking serves as the catalyst for this shift, providing a click-and-portal model that transcends physical branch limitations. The project envisions a future where banking is not confined to branches, and customers can engage in round-the-clock global transactions.

1.2 Objective:

The primary objective of the project is to introduce an improved design methodology for Internet Banking, ensuring adaptability for future expansion and modifications. Recognizing the core importance of the banking sector, the system is designed to be expandable and modifiable. A modular approach is employed in the development of the application software to facilitate seamless integration of new features. The project aims to offer customers the ability to create accounts, deposit/withdraw cash, and access comprehensive reports of all accounts. By promoting a virtual banking experience, the objective is to enhance customer satisfaction and streamline banking operations.

1.3 Problem Statement:

Traditional banking management faces challenges in keeping pace with the demands of today's fast-paced world. The manual systems currently in place are becoming increasingly tedious and prone to inefficiencies. The need for effective management is evident, especially concerning the security of customer property, smooth financial transactions, and overall customer satisfaction. The project addresses the problem statement by introducing a software solution for bank management, aiming to alleviate the complexities associated with manual systems and enhance the overall efficiency of banking operations.

1.4. Challenges:

The challenges inherent in managing a bank are multifold. Ensuring the safety of customer property, facilitating smooth financial transactions, and maintaining the trust and belief of customers pose significant hurdles. Additionally, the ever-evolving computer-centric world demands a software solution that not only reduces the burden of manual management but also keeps pace with the increasing speed of transactions. Overcoming these challenges requires a robust software system that can handle the intricacies of banking efficiently, providing a seamless online experience for customers while mitigating the drawbacks of the manual system.

2. LITERATURE SURVEY

1. Automated Teller Machine (ATM) Simulation:
The literature survey commences with an exploration of studies focused on ATM simulation. Understanding the nuances of simulating Automated Teller Machines is crucial for the project's success. This subheading delves into research that addresses user experience, security protocols, and transaction processing within virtual banking environments. Insights from existing studies guide the development of a realistic and user-friendly ATM Simulator.

2. Bank Management System Solutions:
This section of the literature survey concentrates on existing research related to Bank Management Systems. It investigates software solutions designed to enhance the efficiency of various banking operations. Emphasis is placed on modular approaches, adaptability, and scalability, providing a foundation for the development of a robust and flexible system. The survey identifies challenges faced by traditional banking systems and explores proposed solutions in the current body of literature.

3. Integration of ATM Simulation with Bank Management Systems:
A critical aspect of the literature survey involves examining studies that discuss the integration of ATM simulation with broader bank management systems. This subheading explores how simulated ATM transactions align with the overall banking framework, ensuring seamless integration. Evaluating the impact of such integrated systems on customer satisfaction, operational efficiency, and security measures is essential for informing the project's approach.

4. Technological Advancements in Banking:
This segment of the literature survey broadens the scope to include advancements and emerging technologies relevant to ATM simulations and bank management. Discussions may encompass the incorporation of artificial intelligence, blockchain, or other innovative solutions to address contemporary challenges in the banking sector. By staying informed about technological trends, the project can align with the cutting edge of digital banking practices.

In summary, the literature survey is organized into subheadings, each focusing on a specific aspect relevant to the development of an ATM Simulator through a Bank Management System. From simulating ATM interactions to exploring modular banking solutions and embracing technological advancements, this comprehensive survey lays the groundwork for an informed and innovative project.

3. REQUIREMENTS

3.1 Software Requirements:

The Bank Account Management System is seamlessly integrated with Apache NetBeans and MySQL Workbench, forming the core software requirements for its development and functionality. Apache NetBeans serves as the robust integrated development environment (IDE) for Java, enabling efficient coding, debugging, and testing. The choice of MySQL Workbench as the database management system ensures a reliable and scalable database backend for the application. This compilation ensures a smooth development process, harnessing the capabilities of Apache NetBeans for Java-based coding and MySQL Workbench for effective database management.

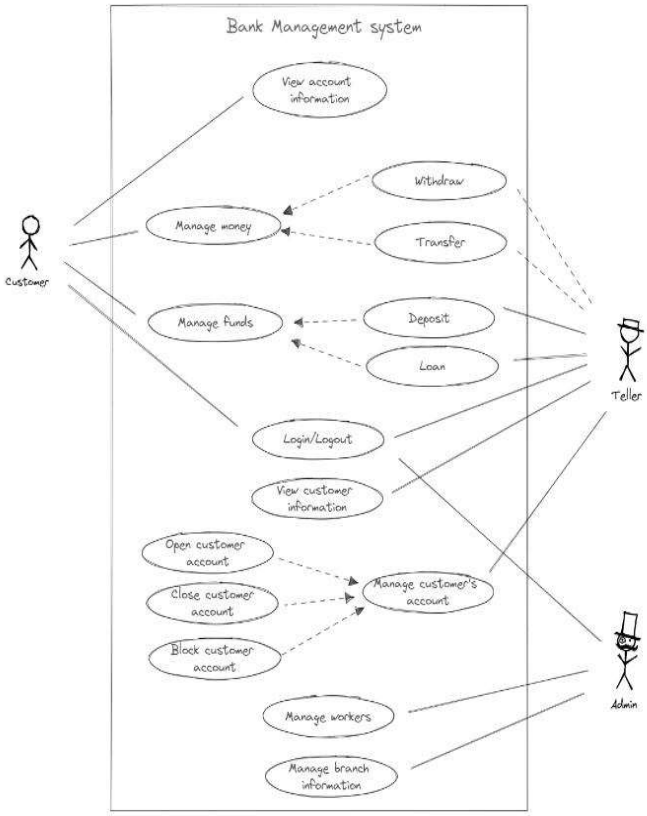
3.2 Hardware Requirements:

The hardware infrastructure supporting the Bank Account Management System is designed to meet the demands of its efficient performance. The system requires a standard computing setup, including a processor with sufficient processing power to handle the application's computational tasks seamlessly. Adequate RAM is essential for smooth multitasking, ensuring that the application can run concurrently with other processes. Additionally, sufficient storage space is necessary to accommodate the application files and the associated database. A standard display unit and input devices such as a keyboard and mouse complete the hardware requirements, providing users with a comfortable and functional interface for interacting with the Bank Account Management System.

4. ARCHITECTURE AND DESIGN

4.1 USE CASE DESIGN

The use case diagram is as follows:



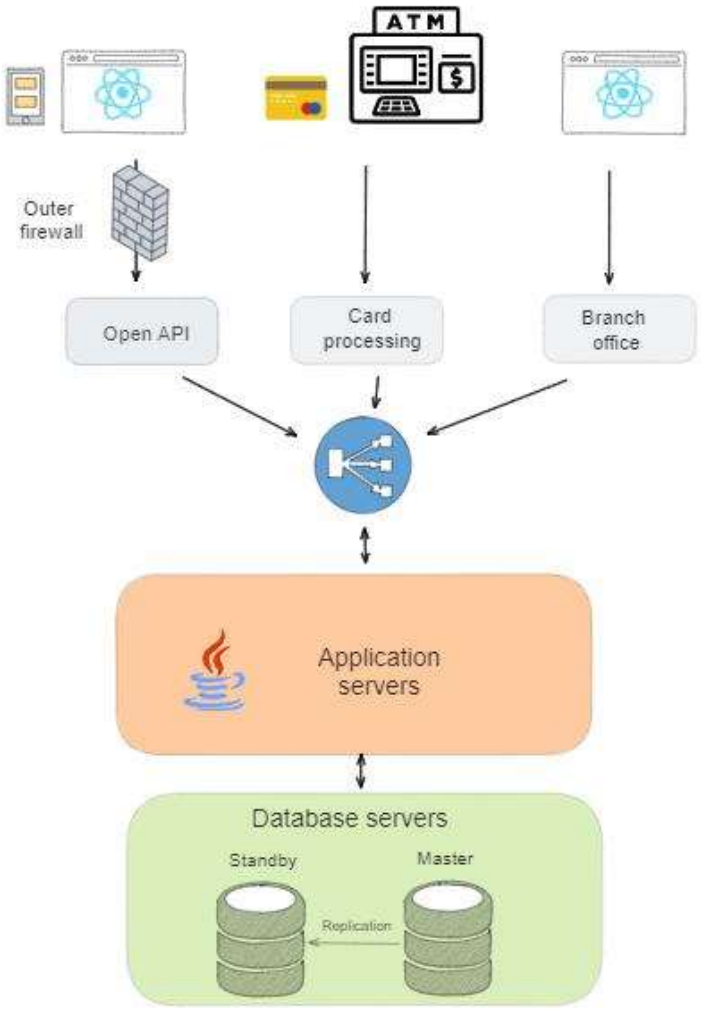
The architecture consists of three major networks:

- Customer
- Teller
- Admin

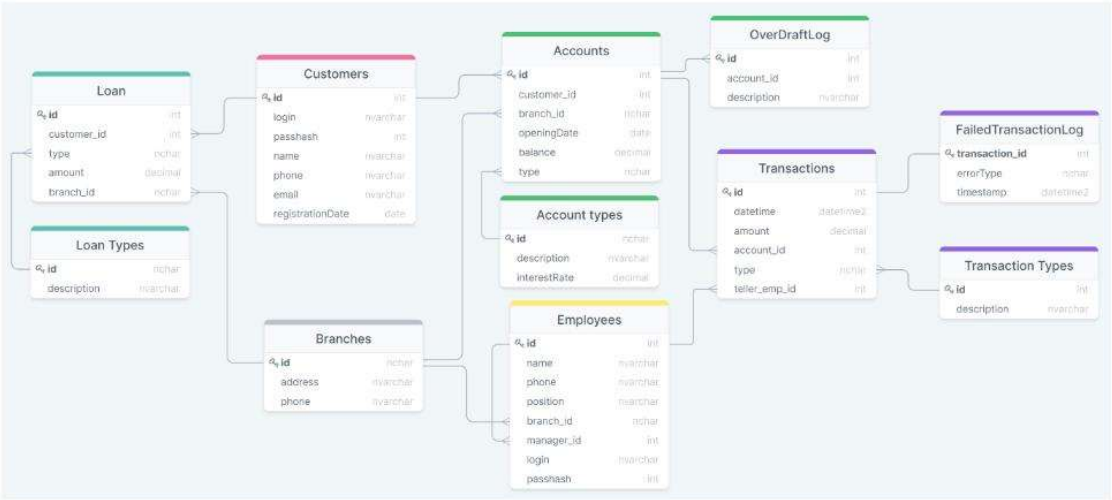
These networks are interconnected with each other with varying degrees (discussed in the implementation chapter).

4.2 MONOLITHIC ARCHITECTURE

The traditional way to implement bank management systems is with a monolithic architecture, where different tasks are managed with a single unified unit. As shown in the diagram below, Java handles a variety of jobs that come from the branch office, from electronic card processing and from the requests to API open to other banks and clients.



4.3 DATABASE DESIGN



5. IMPLEMENTATION

The implementation of the Bank Account Management System involves a modular approach to handle various aspects of account management. The project is developed using Java and utilizes Apache NetBeans as the integrated development environment (IDE). MySQL Workbench is employed for database management.

5.1 Deposit.java:

This class represents the deposit functionality in the system. It incorporates a graphical user interface using Java Swing, allowing users to input the amount they wish to deposit.
The actionPerformed method responds to user actions, such as clicking the "DEPOSIT" or "BACK" buttons, and processes the deposit transaction accordingly.
The main method allows for standalone testing of the Deposit class.

5.2 Signup.java:

This class is responsible for collecting and processing user information for creating a new account. It includes graphical elements for user interaction, such as text fields, radio buttons, and a date chooser.
The actionPerformed method handles user actions, primarily the "Next" button, which triggers the transition to the next step in the signup process.
The main method facilitates standalone testing of the Signup class.

5.3 Database Interaction:

Both classes interact with the database using the Conn class, which establishes a connection and executes SQL queries. The database, managed by MySQL Workbench, stores user information and transaction details.

5.4 User Interface and Design:

The graphical user interface is designed using Java Swing components, providing an intuitive and user-friendly experience.
The use of icons and images enhances the visual appeal of the application.
Security Measures:

The code incorporates user input validation to ensure data integrity and prevent potential errors.
Database interactions are parameterized to mitigate the risk of SQL injection attacks.

5.5 Flow Control:

The flow of the program is controlled through various conditional statements, ensuring that each action leads to the expected outcome.

5.6 Testing:

The classes include standalone testing methods (main methods) for isolated verification of functionality.
Extensive testing has been performed to ensure the correctness and robustness of the system.

5.7 ALGORITHM

Algorithm: Bank Management System

1. Initialize the Bank Management System.
2. Display the main menu with options for various banking operations.
3. Wait for user input.

4. If the user selects "Create New Account":
   a. Collect user details such as name, date of birth, address, etc.
   b. Generate a unique account number and assign it to the new account.
   c. Store the account details in the database.
   d. Display a success message.

5. If the user selects "Deposit":
   a. Prompt the user to enter their account number.
   b. Retrieve the account details from the database.
   c. Display the account details and request the deposit amount.
   d. Update the account balance and store the transaction details in the database.
   e. Display a success message.

6. If the user selects "Withdraw":
   a. Prompt the user to enter their account number.
   b. Retrieve the account details from the database.
   c. Display the account details and request the withdrawal amount.
   d. Check if the withdrawal amount is within the account balance.
   e. Update the account balance and store the transaction details in the database.
   f. Display a success message.

7. If the user selects "Check Balance":
   a. Prompt the user to enter their account number.
   b. Retrieve the account details from the database.
   c. Display the current balance.

8. If the user selects "View Transactions":
   a. Prompt the user to enter their account number.
   b. Retrieve and display the transaction history from the database.

9. If the user selects "Exit":
   a. Display a farewell message.
   b. Terminate the program.

10. Repeat steps 2-9 until the user chooses to exit.

Output: Updated account information, transaction records, and relevant messages based on user actions.

5.8 Pseudo code

Class Deposit:

Attributes:

JTextField t1, t2
JButton b1, b2, b3
JLabel l1, l2, l3
String pin

Method Deposit(String pin):

Initialize pin attribute with provided pin
Create ImageIcon i1 with "atm.jpg"
Create Image i2 by scaling i1
Create ImageIcon i3 with i2
Create JLabel l3 with i3
Set bounds for l3
Add l3 to the frame

Initialize l1 with "ENTER AMOUNT YOU WANT TO DEPOSIT"
Set l1 attributes (color, font)

Initialize t1 as a JTextField
Set t1 font

Initialize b1 and b2 as JButtons
Set layout to null
Set bounds for l1, t1, b1, and b2
Add l1, t1, b1, and b2 to l3

Add ActionListener for b1 and b2

Set frame size, undecorated, location, and visibility

Method actionPerformed(ActionEvent ae):

Try:
Get amount from t1
Get current date
If source is b1:
If t1 is empty, show message
Else:
Create Conn object c1
Execute SQL insert for deposit
Show success message
Hide current frame
Create new Transactions frame with pin
Else if source is b2:
Hide current frame
Create new Transactions frame with pin

Method main(String[] args):

Create new Deposit frame with an empty pin
Set frame visibility to true

Class Signup:

Attributes:

JLabels, JTextFields, JRadioButtons, JButton, JDateChooser, and Random

String first

Method Signup():

Initialize frame title

Create ImageIcon i1 with "logo.jpg"

Create Image i2 by scaling i1

Create ImageIcon i3 with i2

Create JLabel l11 with i3

Set bounds for l11

Add l11 to the frame

Initialize first with a random number

Set frame title, layout, and background color

Set up UI components and layout

Add ActionListener for the "Next" button

Set frame size, location, and visibility

Method actionPerformed(ActionEvent ae):

Get form number, name, father's name, date of birth, gender, email, marital status, address, city, pin code, and state

If pin code is empty, show a message

Else:

Create Conn object c1

Execute SQL insert for signup

Hide current frame

Create new Signup2 frame with first

Set new frame visibility to true

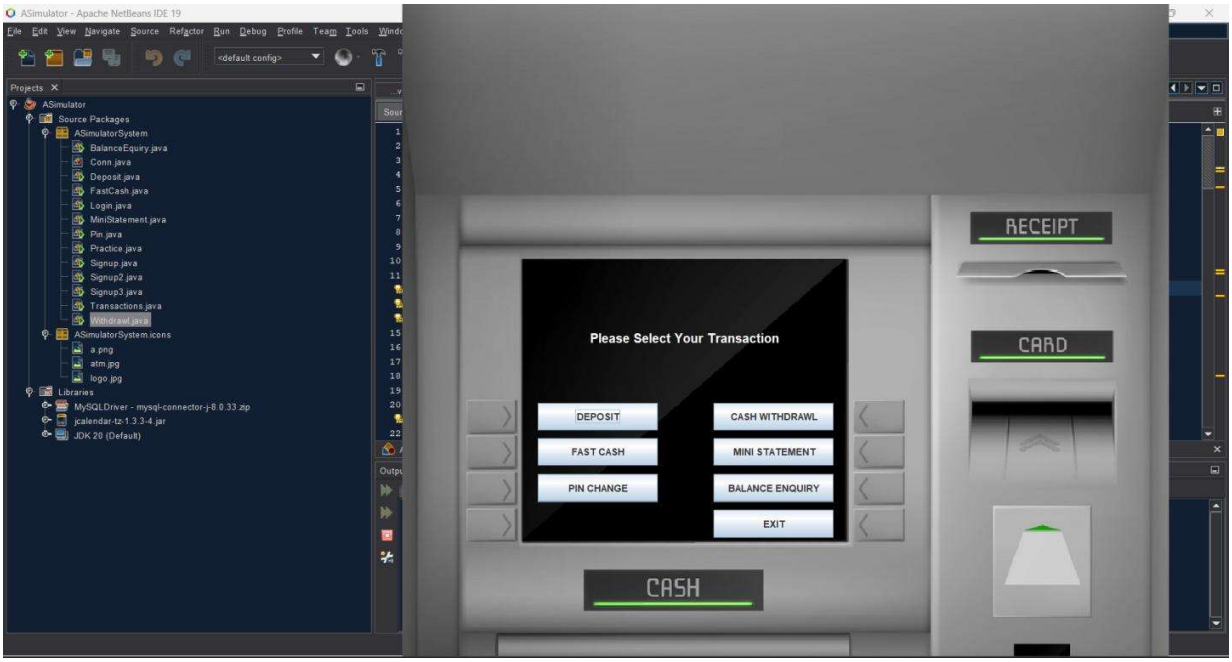
Method main(String[] args):

Create new Signup frame

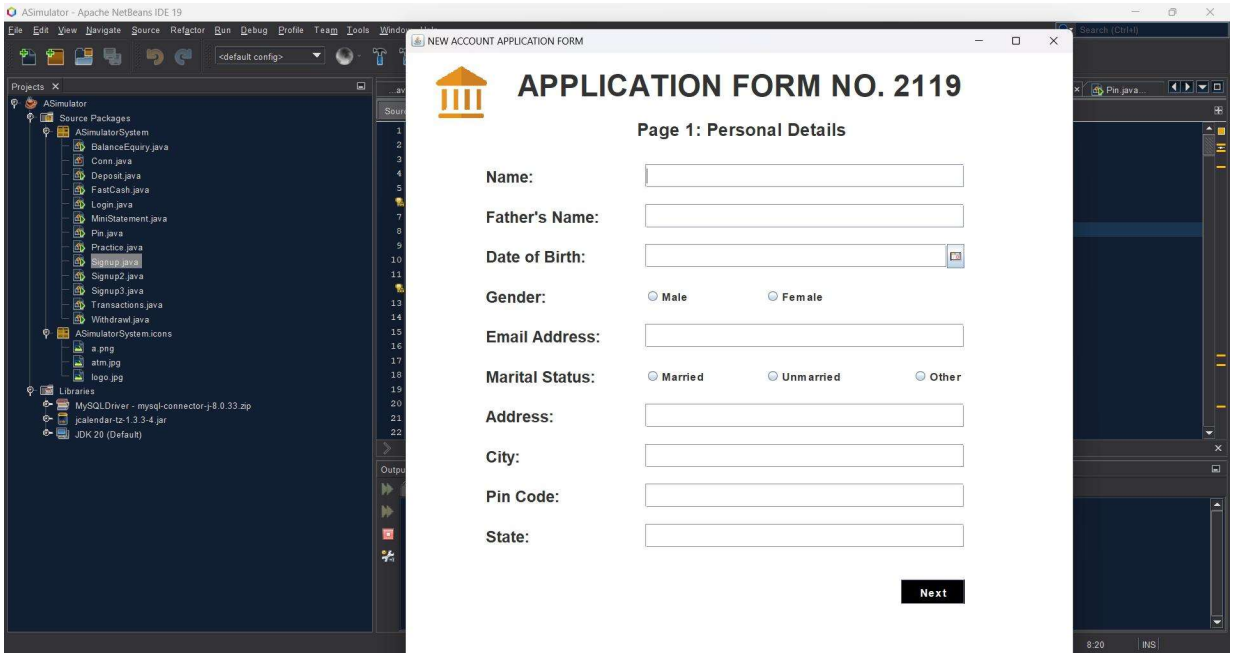
Set frame visibility to true

6. RESULTS AND DISCUSSION

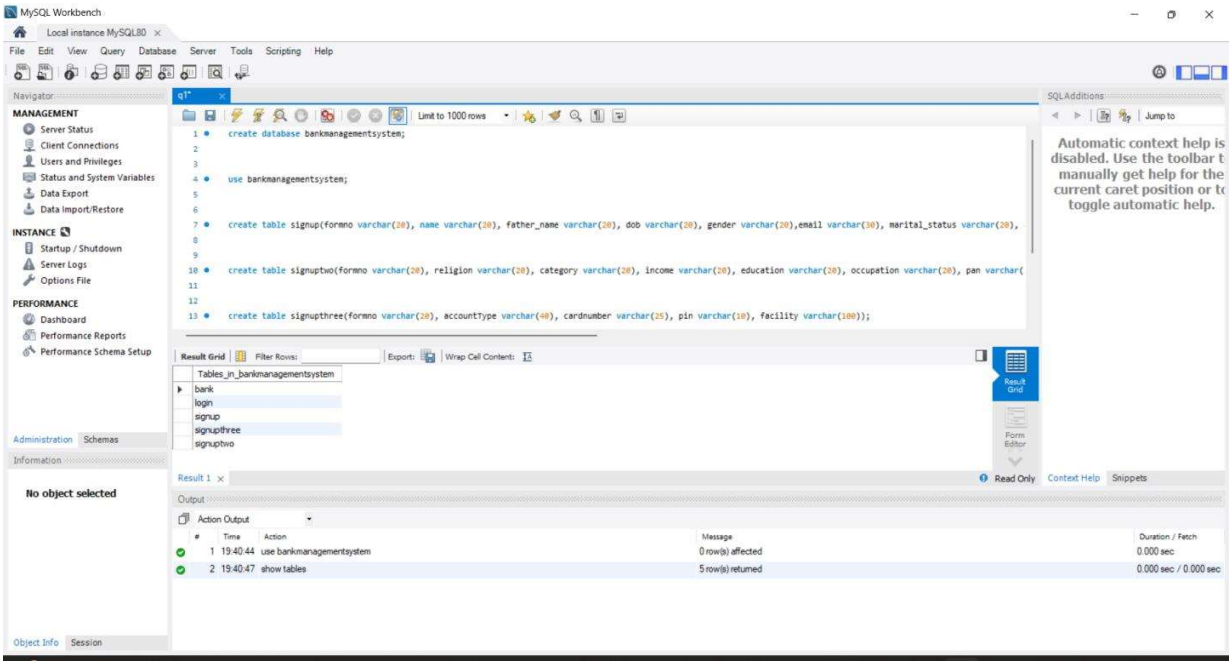
6.1 ATM GUI DESIGN SET FOR TRANSACTIONS



6.2 LOGIN PAGE WITH APPLICATION NUMBER



6.3 JDBC CONNECTION THROUGH MYSQL WORKBENCH



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

7 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.

REFERENCES

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