**A Project Report**

**on**

**Internet banking system**

**By**

Velaga Sai Surya

AP18110010090

CSE-B

UNDER THE GUIDENCE OF

Dr. Ashu Abdul

**ABSTRACT**

Internet banking refers to system that enables bank customers to access account, bank products and services. The major concern for an Internet -banking is the ‘security’ This will be accessible to all customers who have a valid Account number and Password. There was a time when the only way to open a bank account was in person. But now, using this GUI application it is possible to complete the enrolment from start to end within minutes.

It provides secure access to the account and bank services in a user-friendly interface. It provides the greater opportunity to interact with Account holder. The main aim of the project is transfer money to another account, change password, check account balance, transaction history etc. One called Login Password and the other called transaction password. By using login password, the customer can login to be account and we can perform only some (limited) operations like viewing A/C balance and personal details etc.

In the account creation module, it will provide account number, cif number a password to login and a transaction password. A user can make a fund transfer to another account.

**INTRODUCTION**

Internet banking system is all about organizing and managing the bank products and services that provides to a customer (user). It involves maintaining the database with user details that are provided during the account creation. Every moment/operation of the user is stored into the database.

And the project also allows the user to choose the account type like current account or savings account in the time of account creation. The owner can easily update, delete and insert data in the database with this project. Also, a user can request the details (history) of his/her transactions performed from their account. The user can change/update his/her either login password or transaction password so that it makes the account more secure while doing transactions etc,

The following are some of the features/functions provided by this project: -

1. A new user must create account to get the login credentials and access the bank product and services.

2. User must have a valid user ID and password to login to the system.

3. After the valid user logs in, the system shows my page.

4.My page contains several modules starting with user profile where the user can update the profile.

5. All the modules are shown on different panels in the same java frame.

6. User can perform transactions like deposit and withdrawal from his account along with money transfer to another account.

7.It provides a refresh button whenever it is updating my page content like transaction history etc, for the user.

**Related work/Literature survey**

The general java internet banking system provides only the login password for login and transactions and it doesn’t have any transaction history. But that makes the user account secure less, i.e. anyone knowing the user password can access the entire bank products and services.

User

login

This project provides two passwords. One is for login which is used to login to the system and can access to withdraw and deposit. And the other is transaction password which is used to do transactions. The system only allows the user to perform the transactions if the transaction password is a valid one. It is like having two secure firewalls for a system. And this project also provides transaction history i.e., the transactions that a user performed or received from another user. Both are shown in different panels (let’s say tables).

Also, this project allows the user to reset not only the login password but also the transaction password. So, it makes the user account more secure.

Use of Login password:

1. For login purpose

2. Withdraw / Deposit

3. For resetting transaction password.

4. Balance enquiry.

Diagrammatical representation:

User details

User

Access

Access

Valid user

password

**Database**

response

Registration Info

Login\_ Info

User Details

Request

(d)

transaction password

**PROBLEM STATEMENT/OBJECTIVE**

Internet banking project object is to maintain the user/customer details, manages the user data and their money deposit, withdraw, transactions. Manages includes updating user details, checking received transactions, send transactions and updating the login and transaction password.

This project reduces the manual work instead of transactions. These all things make the user account more secure. The user must use the password to perform/use any action/module. For transactions purpose one must have to use the transaction password. And for withdraw and deposit purpose only password is required. And it makes the easy task the monitor the account for a user.

**The Main Functions of the Internet Banking System:**

1. User can check and update the profile after login.
2. Can make transactions to a known person by simply knowing their account number.
3. Can request details of all the transactions that he/she sent or received.
4. Provides balance enquiry
5. Withdraw amount from account.
6. Deposit amount in the account.
7. Can change Login Password.
8. Can change the Transaction Password.

**PROPOSED SYSTEM**

The project includes various modules. They are: -

User Registration Module:

This is where a user must create an account to login. During the account creation process account, cif number are auto generated which can’t be edited. And the module auto generates the login password and transaction password, but the user can re-write it to their own.

User Login Module:

This module allows the valid user having a bank account to the bank products and services.

My page Module:

It contains all the bank services in the form of module including profile, transactions etc.

**Algorithm**

Step-1: Start

Step-2: Create a database (named bankdb) with tables as (userdetails, balances, transactions) with required table data.

Step-3: Create a Connecting named class with method name ConnerDb() and write the connection in it. Call the method in every Constructor of the new classes.

Step-4: Create a login JFrame class and place the login fields such as account number entry field and password entry field.

Step-5: Add a login button and create account button.

Step-6: Create a jframe named account (To create a account).

Step-7: Place the required components such as account number, cif number, password, transaction password file and write the code to generate these items using random method.

Step-8: Add more labels for person details such as name, gender, village, phone number and Aadhar number, mail id etc.

Step-9: Write the code to write the user details along with account number and passwords into the userdetails table.

Step-10: Create a loading frame with a parameterised constructor (namely string).

Step-11: In login button which is placed in login frame write the action listener so that the details provided by the user are valid (check for the password that having the same account number that user provided)

Step-12: If the user details are valid pass the account number to loading class. If not show a message that the user details are invalid using JOptionPane.

Step-13: Create a mypage frame. With parameterised constructor.

Step-14: In loading class Add a progressbar in and create a thread with run method. If the login details are valid set the progress bar increment using run method.

Step-14: And after 100% pass the user account number (which came from login) as a parameter to mypage.

Step-15: Add 9 panels and using a tabbed pane along with a logout button and refresh button. And two labels to display the username and date.

Step-16: Name the panels as

1. profile

2.deposit

3.withdraw

4.transfer

5.transactions(sent)

6.transactions(received)

7.change password

8.transaction password

9.balance enquiry.

Step-17: In profile panel add required labels and text fields (default to not editable) to display the user button along with edit and save button.

Step-18: In the parameterised constructor of mypage class using account number (which is passed from login class to loading class and then mypage) write the code to display the user details in profile panel by retrieving the user data using account number from database.

Step-19: In the edit button create an action listener to set the text fields editable. (So that the user can update his/her profile).

Step-20: Create another action listener in the save button to update the user details from profile panel to database.

Step-21: In deposit panel add the required labels and text-fields along with add button and deposit button and a password field.

Step-22: Create an action listener in the add to get the user account balance from database and add it to the amount that user wants to deposit.

Step-23: Create another action listener in the deposit button to check the user password that he/she entered to do the deposit action is valid or not.

Step-24: If valid, update the user balance. If not show a message that the password is not valid.

Step-25: In withdraw panel add the required components such as labels, text-fields, password fields, and a withdraw button.

Step-26: create an action listener in the withdraw button as if the user entered password is valid it should minus the amount that user entered from the action account balance.

Step-27: And update the account balance in the database.

Step-28: In the transfer panel add the labels and text-fields, and search, transfer buttons.

Step-29: create an action listener in the search button to search for the destination account holder details by getting the account number form text-field.

Step-30: If account number is found, display the account holder name in a label. If not show a message that no user found.

Step-31: Create another action listener in the transfer button and perform action to add the transfer amount to destination account (only if the given transaction password is valid).

Step-32: And update the balance of both the destination account and user’s account. And insert this transaction into transactions table.

Step-33: Add a table in the transaction(sent) panel to display the user performed transactions.

Step-34: Add another table to the transaction(received) panel to display the user received transactions.

Step-35: In the parametrised constructor of mypage class, get the user sent transactions by searching the user account number in the transaction table of database.

Step-36: Display all the sent transactions in the table of transaction(sent) panel.

Step-37: Get the user received transactions and display them in transaction(received) panel.

Step-38: Add the required password fields in the change password panel and an update button

Step-39: In the update button create an action listener. Check if the user password valid or not.

Step-40: If valid update the new password in the userdetails table. If not display that the password is wrong.

Step-41: And similarly add password fields and a update button in transaction password panel to update the password.

Step-42: create an action listener in the update button to check the given password valid or not.

Step-43: If valid update the transaction password in the userdetails table. If not display that the password is invalid.

Step-44: In the balance enquiry panel add labels and password field and a button named ‘check’.

Step-45: Create an action listener in the check button. If the user given password is valid display the user’s account balance.

Step-46: If not display a message that is not valid.

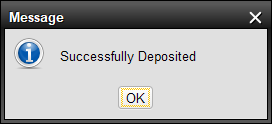
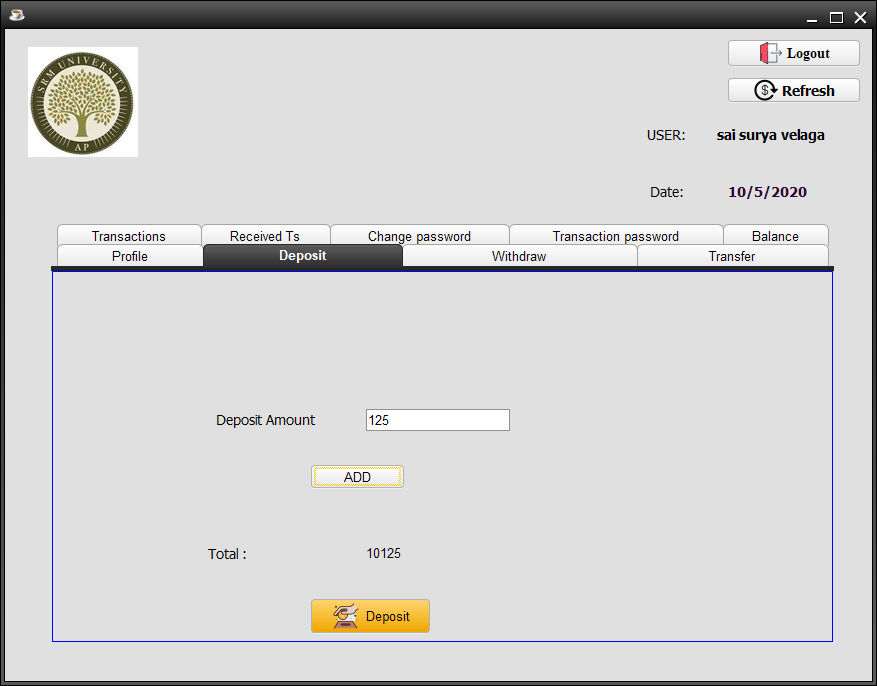
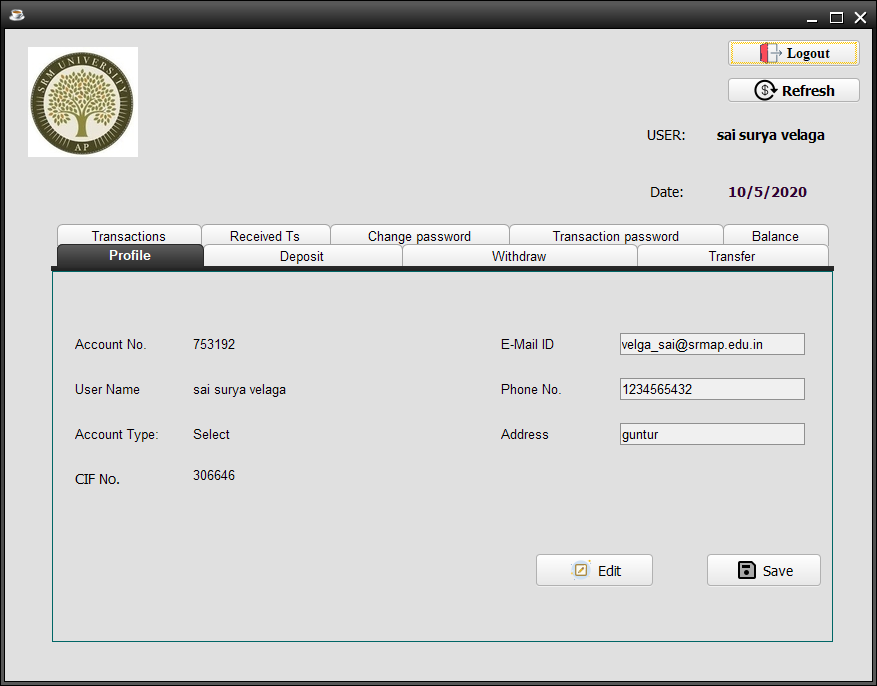
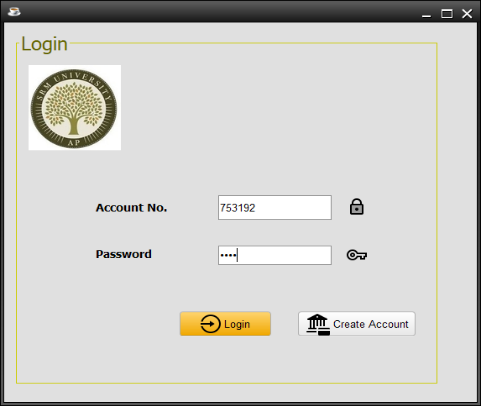
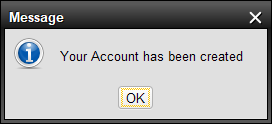
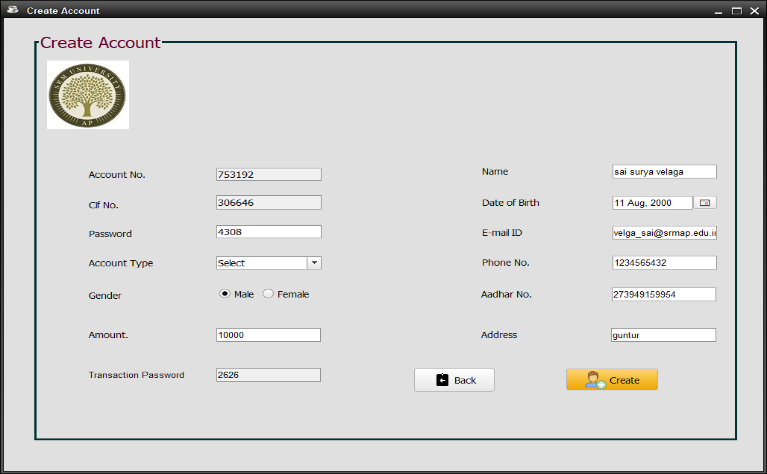
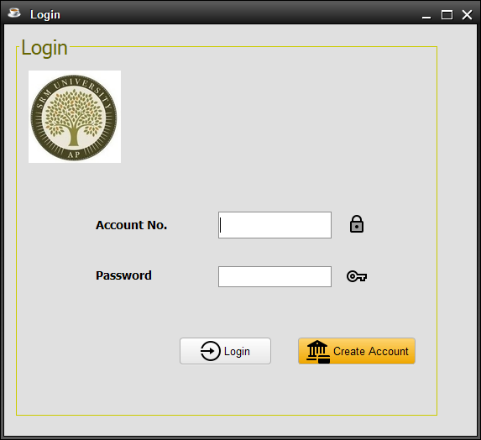
Step-47: Add two buttons (Logout, Refresh) to the mypage frame.

Step-48: Create an action listener in the refresh button to refresh the mypage by taking it to loading page.

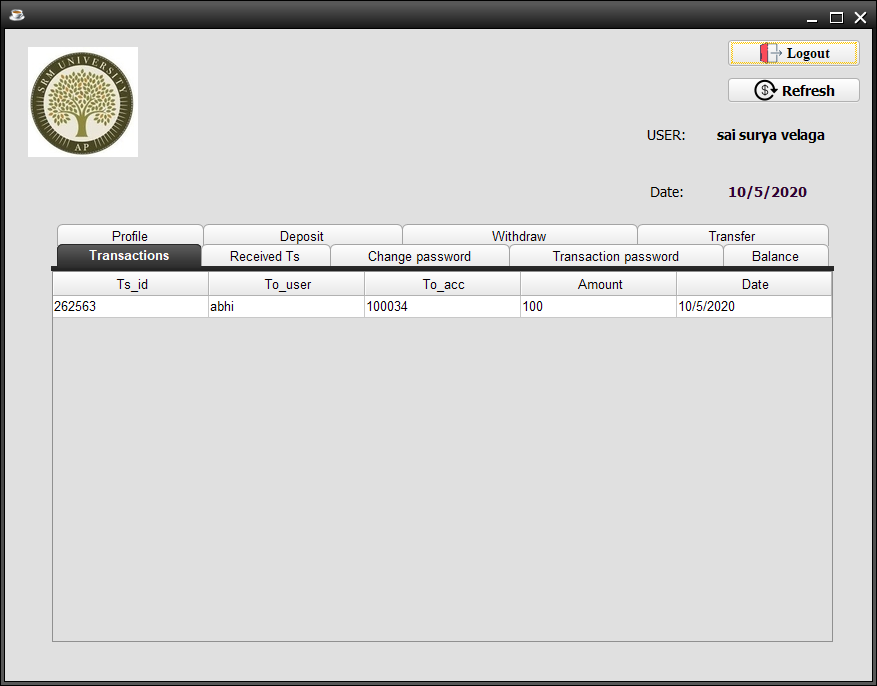
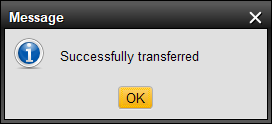
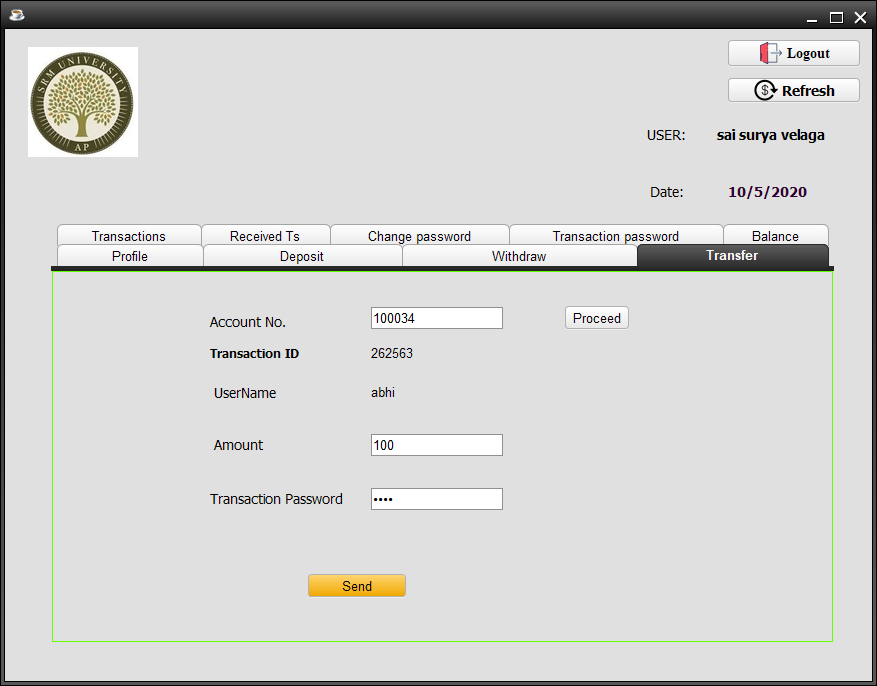
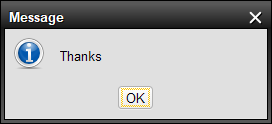
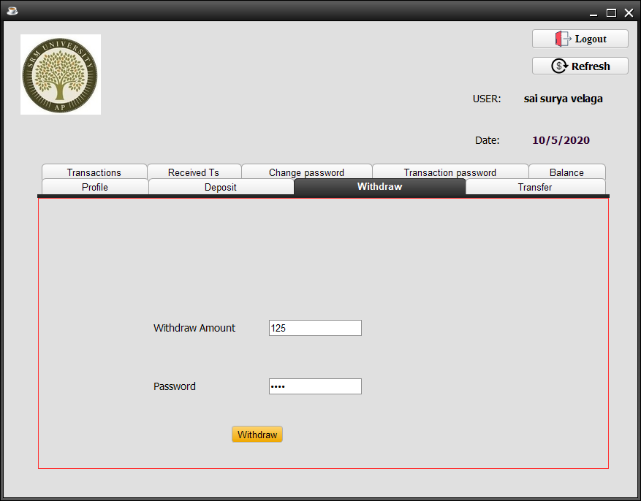
Step-49: Create an action listener in the logout button. If action is performed it should logout and return to authentication frame.

Step-50: Stop.

**RESULTS/PERFORMANCE ANALYSIS**



**Transactions:**



**CONCLUSION**

Here we provide security and authentication for the online banking account, the customer using the online banking service would be given user id along with two other passwords. One called Login Password and the other called transaction password. It planned to be having the view of distributed architecture, with centralized storage of the database. The project provides security to the customer while using the service in the following ways. The customer using the online banking service would be given user id along with two other passwords.

By using login password, the customer can login to be account and we can perform only some (limited) operations like viewing A/C balance and personal details etc. But to perform transfer or online transaction, the user needs to provide the transaction password and to withdraw or deposit the user needs to provide login password.

**Hardware requirements of Internet banking system:**

Operating System: Windows 7 (or above)

RAM: 1 GB or above.

ROM: 50 GB

Software Requirements:

Net Beans IDE or My Eclipse

MySQL

Sql connector

**REFERENCES**

1. For database connections

<https://www.codejava.net/java-se/jdbc/connect-to-mysql-database-via-jdbc>

2. For designing templets

<https://w3layouts.com/banking/>

3. For building GUI applications

<https://netbeans.org/kb/docs/java/gui-functionality.html>

**The Complete Reference Java (Seventh Edition)**

**-HERBERT SCHILDT**

**-- Thank you --**