Atm Simulator with Role Based Access Control

A CS814 Course Project Report

Under the guidance of
Dr. Mahendra Pratap Singh

Submitted By

Satyam Prakash (202CS027)
Joan Lusanji Imbwaga (202CS034)

Department of Computer Science and Engineering

National Institute of Technology Karnataka

P.O. Srinivasnagar, Surathkal, Mangalore-575025 Karnataka, India

January 2021

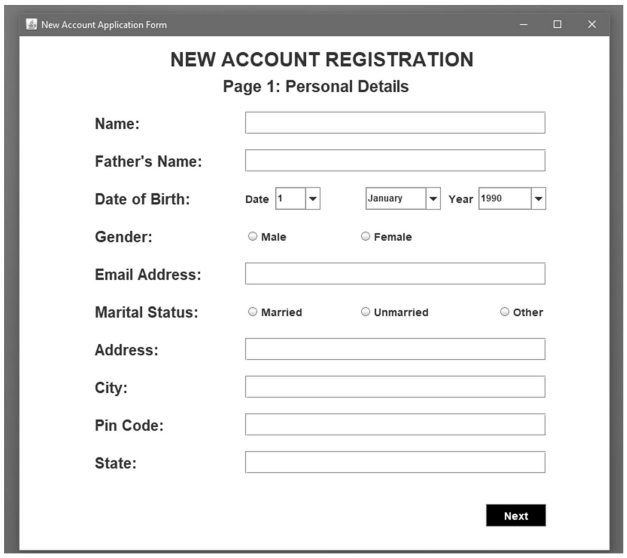
TABLE OF CONTENTS

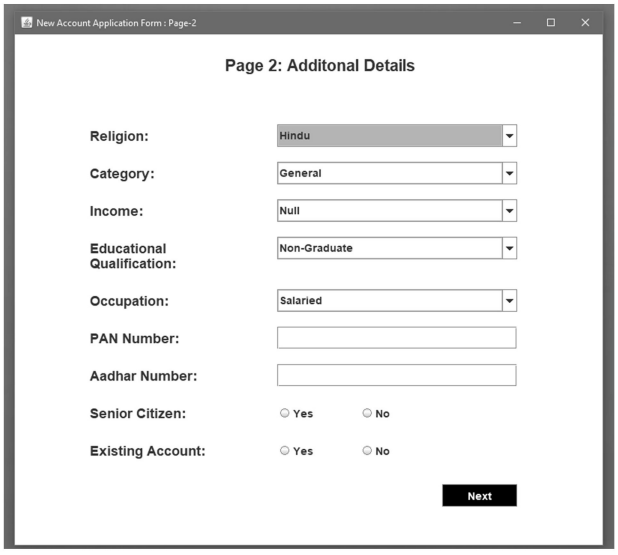
INTRODUCTION

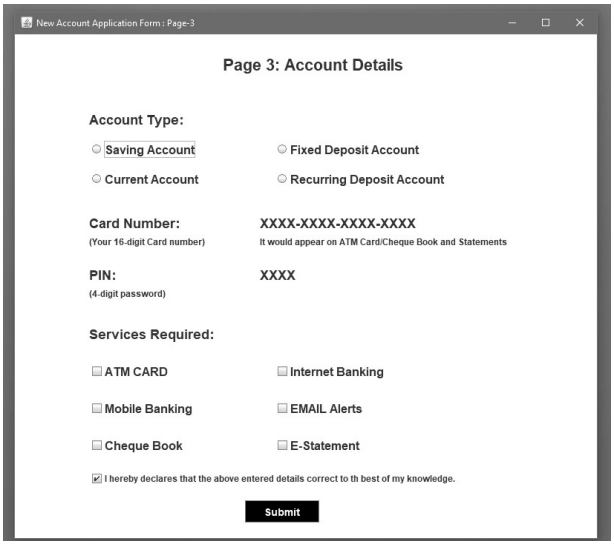
The aim of the ATM Simulation System project is to build a Java based ATM (Automated Teller Machine) Simulation System using Role Based Access Control. The introduction of ATM's by various banks has brought about freedom from the endless queues in front of withdrawal counters at banks. This ATM Simulation System requires the constant updating of records between the bank servers and a spread out network of ATM's.

Security is the foundation of a good ATM system. This system will provide for secure authenticated connections between users and the bank servers. The whole process will be automated right from PIN (Personal Identification Number) validation to transaction completion. ATM Simulation System will enable two important features of an ATM, reduction of human error in the banking system and the possibility of 24 hour personal banking.

This project implemented using Java language and MySQL database. Customer and Administrator can login into system using their Card Number and PIN. There will be need of a person who can manage database as adding or removing a administrator can be performed by that person. New customer account can be created by administrator because administrator can verify all details and enter in system accurately. The proposed system constantly updates the bank records. The Java based construction of the system will enable transactions at any bank or ATM to be registered within a matter of seconds.







Customer and Administrator have same login page. They can login to their respective interfaces with their card number and PIN.

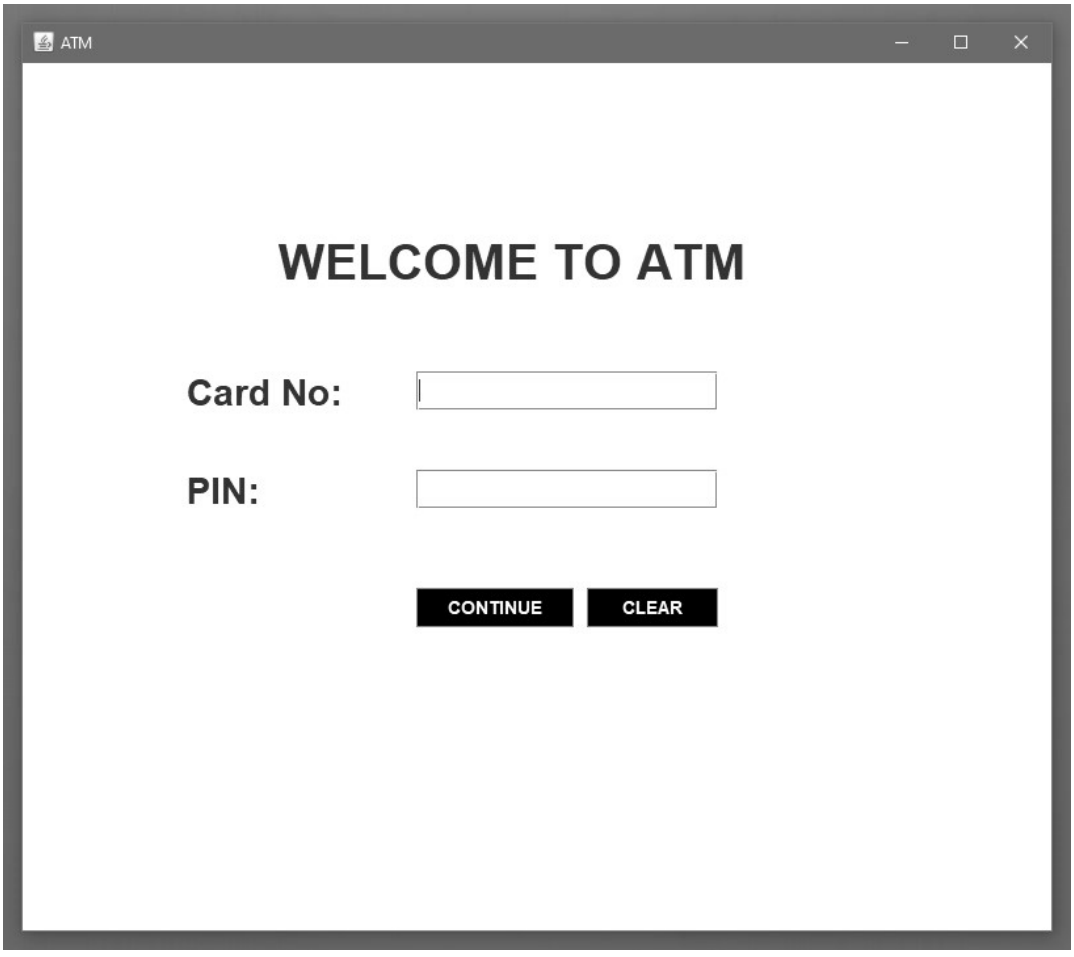


Figure 1 : Login Page

AUTHORIZATION

Role-based access control (RBAC) is a policy-neutral access-control mechanism defined around roles and privileges. The components of RBAC such as role-permissions, user-role and role-role relationships make it simple to perform user assignments. A study by NIST has demonstrated that RBAC addresses many needs of commercial and government organizations. RBAC can be used to facilitate administration of security in large organizations with hundreds of users and thousands of permissions. Although RBAC is different from MAC and DAC access control frameworks, it can enforce these policies without any complication.

Within an organization, roles are created for various job functions. The permissions to perform certain operations are assigned to specific roles. Members or staff (or other system users) are assigned particular roles, and through those role assignments acquire the permissions needed to perform particular system functions. Since users are not assigned permissions directly, but only acquire them through their role (or roles), management of individual user rights becomes a matter of simply assigning appropriate roles to the user's account; this simplifies common operations, such as adding a user, or changing a user's department.

Three primary rules are defined for RBAC:

1. **Role assignment:** A subject can have permission only if the subject has selected or been assigned a role.

2. **Role authorization:** A subject's active role must be authorized for the subject. With rule 1 above, this rule ensures that users can take on only roles for which they are authorized.

3. **Permission authorization:** A subject can have permission only if the permission is authorized for the subject's active role. With rules 1 and 2, this rule ensures that users can exercise only permissions for which they are authorized.

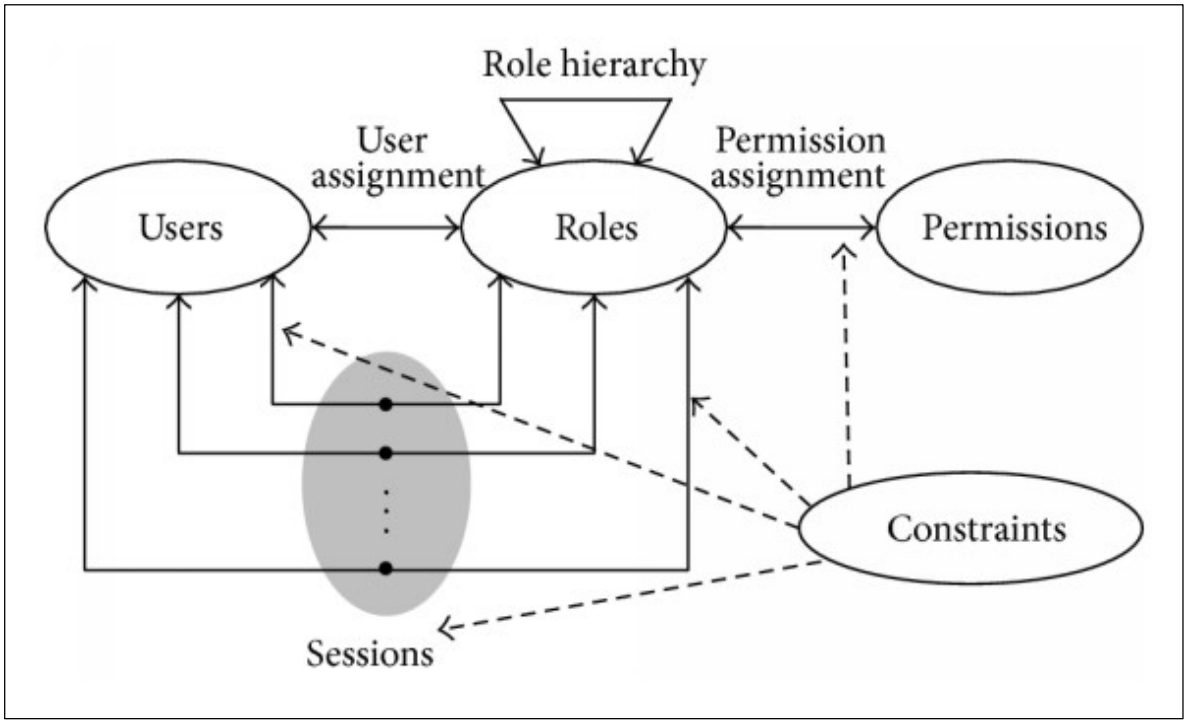


Figure 2: Role Base Access Control Model

Roles

There are two types of roles associated with the users:

1. **Administrator**
2. **Customer**

Permissions

There are multiple permissions associated with roles:

1. **Administrator permissions**
   a) Registration of new customer account
   b) Deletion of a customer account
   c) Access Logs
   d) Set Withdrawal Limit
   e) Show customer accounts
2. **Customer permissions**
   a) Deposit
   b) Withdrawal
   c) Mini statement
   d) PIN change

e) Fast cash withdrawal
f) Balance enquiry

Database Implementation

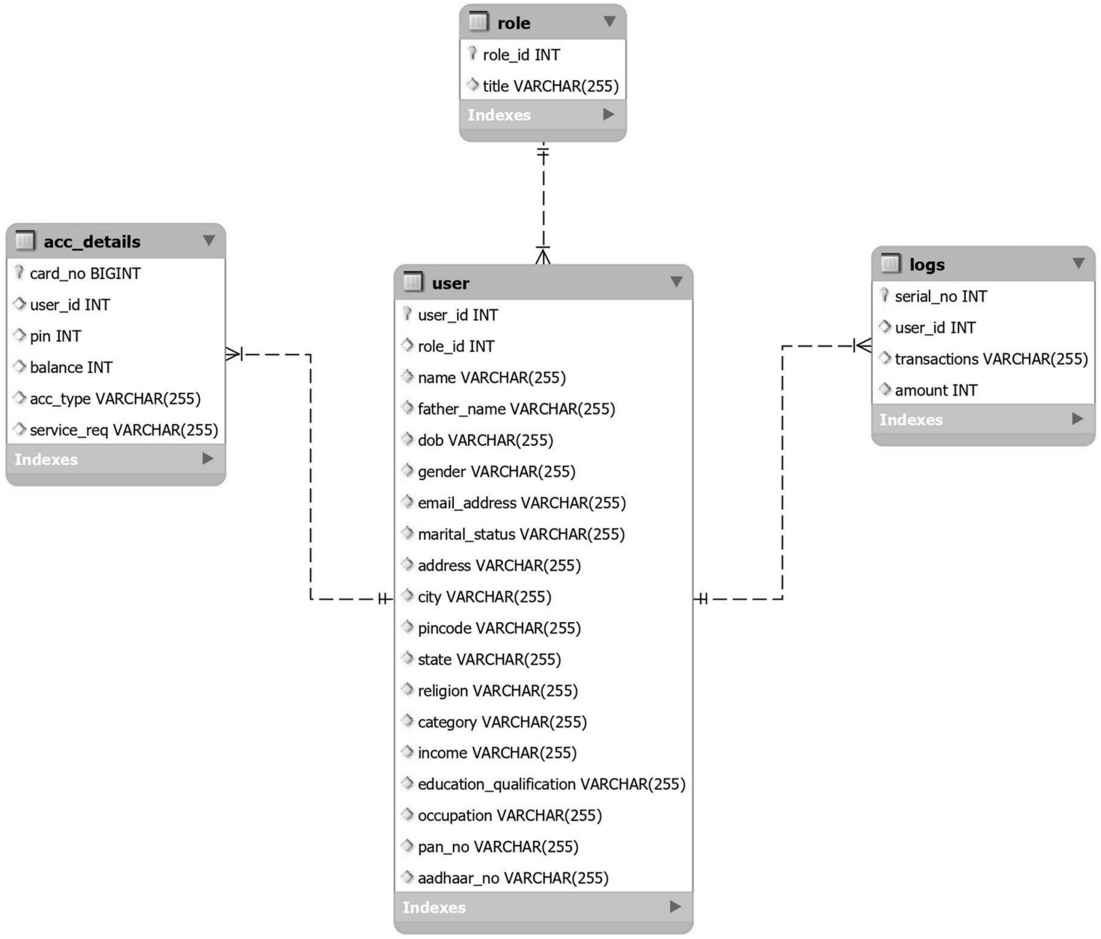


Figure 3 : Entity Relationship Diagram

In case of Database Implementation, we first analyse our requirement and try to come up with database schema. Here we have URA table (table name: user) where we store the user information along with its role.

We have ROLE table which contains role ID and role TITLE .Here role id is the referred key and role value from URA table is the referring key. It creates a foreign key relationship with ON DELETE CASCADE property.

The entire role - permission assignment are performed in the program code itself.

User Interface

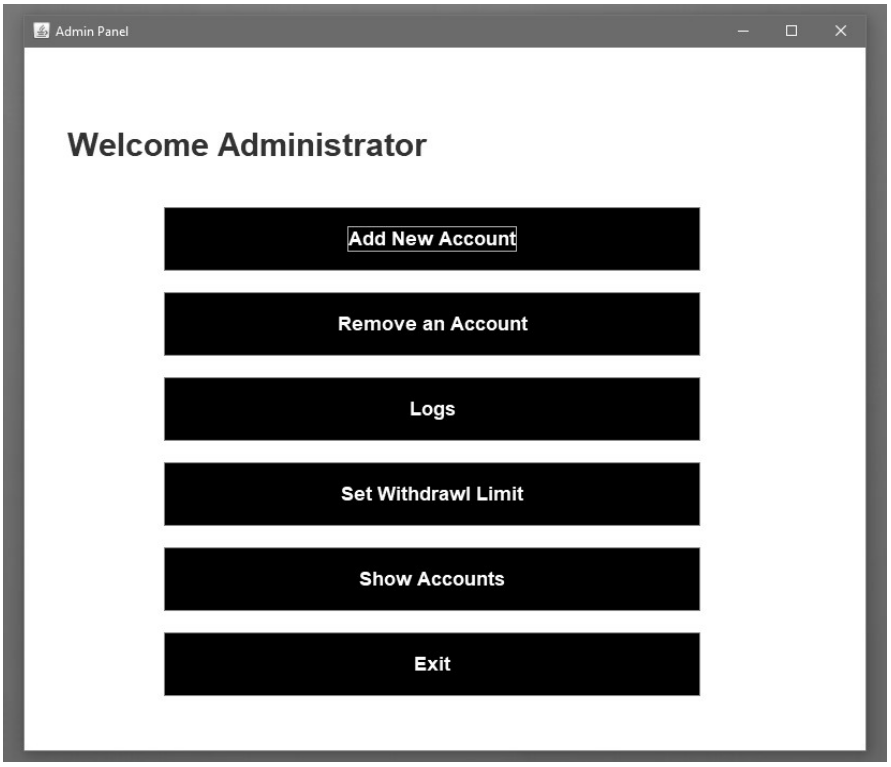


Figure 4 : Administrator User Interface

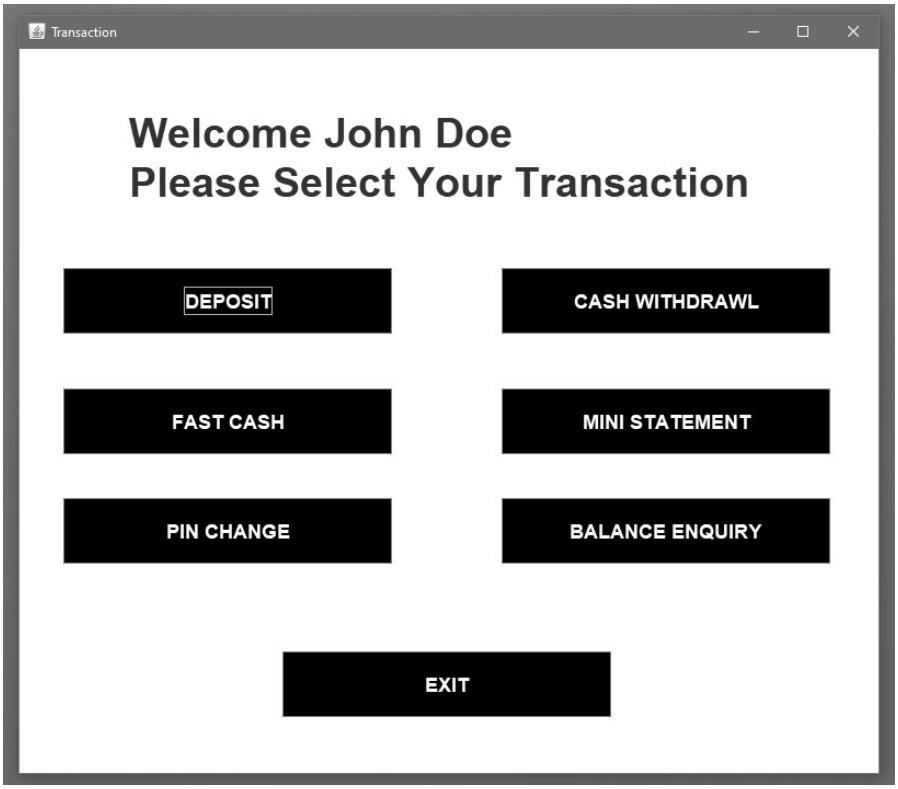


Figure 5 : Customer User Interface

CONCLUSION

The ATM Simulation System with RBAC has many features for customer like cash withdrawal, cash deposit, pin change, mini statement, balance enquiry and administrator can add new account, remove an account, see all accounts, see logs, set withdrawal limit. Despite that there are still many features to add and problems which are crucial.

RBAC also has some issues with it like it cannot use contextual information e.g. time, user location etc., it ignores resource meta-data, the permissions and privileges can be assigned to user roles but not to operations and objects, role explosion. Admin can be added by a database administrator which has right to access the central database.

BIBLIOGRAPHY

1. SANDHU , R. AND B HAMIDIPATI , V. 1997. The URA97 model for role-based administration of user-role assignment. In Database Security XI: Status and Prospect, T. Y. Lin and X. Qian, Eds. Elsevier North-Holland, Inc., Amsterdam, The Netherlands.

2. SANDHU , R. 1997. Roles versus groups. In Proceedings of the 2nd ACM Workshop on Role-Based Access Control (Fairfax, VA, Nov. 6-7). ACM Press, New York, NY.

3. https://www.javatpoint.com/java-swing

4. https://www.javatpoint.com/mysql-tutorial

5. https://www.indiastudychannel.com/resources/169331-What-is-an-ATM-and-what-are-the-services-and-facilities-available-at-ATMs.aspx