VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAUM



**DATABASE MANAGEMENT SYSTEM**

**LABORATORY MINI-PROJECT REPORT**

**ON**

**ATM/CDM SYSTEM**

SUBMITTED BY

**DEEPAK V GAUTHAM N**

**1AT15CS020 1AT15CS023**

Submitted in partial fulfillment of the requirement for the award of the

Degree of Bachelor of Engineering

In

Computer Science &Engineering

UNDER THE GUIDANCE OF

**Prof. Rajendra M**

**Asst. Professor**

**Atria Institute of Technology**

**Anandanagar, Bangalore-560024**

****

**Atria Institute of Technology**

**Anandanagar, Bangalore-560024**

# CERTIFICATE

This is to certify that the project entitled “**ATM/CDM SYSTEM**” has been successfully completed by

**DEEPAK V GAUTHAM N**

**1AT15CS020 1AT15CS023**

In partial fulfillment for the data base management system laboratory with mini-project report[15CSL58] –V semester, BE Computer Science during2016-2017.

Signature of guide Signature of HOD

Prof. Rajendra M Dr. P. Aishwarya

**External Viva:**

(Name of Internal/External Examiner with signature & Date)

Examiner 1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner 2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends, who believe that the individual can excel and put in their every bit in every endeavor he/she embarks on, at every stage in life. And the success is derived when opportunity meets preparation, also supported by a well-coordinated approach and attitude.

The joy and satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people who made it possible and also whose efforts gave this project its final shape.

We would like to express our gratitude to our respected principal **Dr. K V Narayanaswamy** for providing a congenial environment and surrounding to work in. We would like to express our sincere gratitude to **Dr. P. Aishwarya, Head of the Department, Computer Science& Engineering**, for his continuous support and encouragement.

We are indeed indebted to **Prof. Rajendra M,** project guide and lab in-charge, for is support, advice and inputs in the course of this project. We would also like to thank various faculty members of the Computer Science & Engineering Department for their valuable suggestions and inputs.

Last, but not the least we would like to thank our parents, who have acted as a beacon of light throughout life.

Our sincere gratitude goes out to all our comrades and well-wishers who have supported us through all our ventures.

**DEEPAK V GAUTHAM N**

**1AT15CS020 1AT15CS023**

# ABSTRACT

**ATM/CDM SYSTEM**

The **ATM/CDM System** is the project which is used by customers to access their bank accounts in order to make cash withdrawals and cash deposits. Whenever the user need to make cash withdraws, they can enter their PIN number ***(Personal Identification Number)*** and it will display the amount to be withdrawn in the form of 100’s. Once their withdrawal is successful, the amount will be debited in their account and a Transaction ID will be displayed.

The ATM/CDM System is developed in front-end as PHP and back-end database as My SQL. PHP ***(Hypertext Preprocessor)*** is a widely used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Hence we used this software in our project.

The ATM/CDM will service one customer at a time. A customer will be required to enter Bank Account number and PIN – both of which will be sent to the database for validation as part of each transaction. The customer will then be able to perform one or more transactions. Also customer must be able to make a balance inquiry and generate a Mini Statement of any account linked to the Account number.

The ATM will communicate each transaction to the database and obtain verification that it was allowed by the database. In the case of a cash withdrawal, an alert with the Transaction ID will be displayed which the user can take note of. The same goes for Deposit transaction also. Any debits or credits made will be sent to the Database and changes will be reflected in the database as well.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then alert the customer of the error and the customer will be redirected to the Login page.

The ATM will provide the customer with an alert for each successful transaction, showing the transaction ID. When balance enquiry is selected the balance of the account of the customer is displayed. When Mini statement is selected a table showing the date, time, type of transaction, amount and available balance of the affected account

.

# 

# 1. INTRODUCTION

# 1.1 INTRODUCTION TO DBMS

A database is simply an organized collection of related data, typically stored on disk, and accessible by possibly many concurrent users. Databases are generally separated into application areas. For example, one database may contain Human Resource (employee and payroll) data; another may contain sales data; another may contain accounting data; and so on. Databases are managed by a DBMS. Many Database Systems are being used which are in turn managed by many other Database Management Systems. A Database Management System (DBMS) is a set of programs that manages any number of databases. Basically DBMS is a software tool to organize (create, retrieve, update and manage) data in a database. The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Database systems are meant to handle large collection of information. Management of data involves both defining structures for storage of information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

**1.1.1 SQL PROGRAMMING**

SQL (Structured Query Language) is a standard database programming language used to accessing and manipulating data in a database. SQL was developed In 1970 by Donald D. Chamberlin and Raymond F. Boyce at IBM. For handling database and database related programming, programmers need to have some medium or you can say interface to particularize a set of commands or codes to deal with database or to access database’s data. It is pronounced as “sequel”. SQL is the standard language for Relation Database System. SQL is used by many databases like MySQL, Oracle, SQL Server, PostgreSQL etc.

SQL statements always start with the keywords. SQL statement ends with semicolon. SQL is not case sensitive, means update is the same as UPDATE.

Structured Query Language is perhaps the most common way to extract data from a relational database system. The Basic Syntax (simplified) Format is:

SELECT field name(s)

FROM databases table name(s)

WHERE conditional clause

This is the way in which SQL programming can be used to access many number of Databases.

**1.1.2 SQL COMMANDS**

|  |  |
| --- | --- |
| **SQL Command** | **Description** |
| [CREATE DATABASE](https://www.w3schools.in/sql/create-database/) | Creates a new database. |
| [CREATE TABLE](https://www.w3schools.in/sql/create-table/) | Creates a new table. |
| ALTER DATABASE | Modifies a database. |
| ALTER TABLE | Modifies a table. |
| DROP TABLE | Deletes a table. |
| CREATE INDEX | Creates an index. |
| DROP INDEX | Deletes an index. |
| SELECT | Fetch data from a database table. |
| UPDATE | Modifies data in a database table. |
| DELETE | Deletes data from a database table. |
| INSERT INTO | Inserts new data into a database table. |
|  |  |

# 1.2 BRIEF OUTLINE OF THE PROJECT

The ATM/CDM System is a simulated working of an **Automated teller machine/Cash deposit machine**.

The front end tool used to simulate this software database is My SQL. When a user enters his account number and his PIN he will be able to log into his Account. This is achieved by sending the login information to the database and verifying whether the credentials are matching the database records. If it matches the records then the user is able to login to his account to perform any transaction. Else he’ll be required to enter his credentials properly again.

After he has access to his account he can choose between 5 options which are:

1. Withdraw cash.

2. Deposit cash.

3. Enquire balance.

4. Generate a Mini Statement.

5. Change PIN Code.

Any of the option can be chosen choose based on his requirement. If he chooses Withdraw cash option then he will be redirected to a page where he can select if he wants to Withdraw from his Savings account or a Current account. Then he can enter the amount he wishes to withdraw and if the balance is sufficient in his account the transaction will be successful, else he will be redirected back to the login page with an alert that he has insufficient funds in his account. The amount entered should be in multiples of 100s only. The amount which is withdrawn will be updated in the database of the account holder. A transaction ID will be generated and presented to the user, which he can make note of. This is the same case of operation if he chooses the 2nd option which is the Deposit operation in which case he can deposit money into his account.

When the 3rd option that is, Enquire Balance is chosen, the user is presented with a page to choose between his savings account and his current account and that account’s balance is displayed on a new page. This balance will be the current balance pulled from the database.

When the 4th option that is, Mini Statement is chosen, the most recent 8 transactions of that account holder will be presented in the form of a table. The tuples of the table are Date, Time, Transaction number, Account Number, Account Type, Amount, Balance and Transaction Type. All of this tuple details will be pulled from the database and presented to the user.

The final 5th option is Change PIN option in which the user can change his account PIN Code by entering his Old PIN code, New PIN code and verifying his new PIN code. This information is sent to the database and the PIN code change is reflected in the database as well.

This a Brief outline of how the project operates.

# 2. REQUIREMENT SPECIFICATIONS

# 2.1 EXTERNAL INTERFACE REQUIREMENTS

**2.1.1 USER REQUIREMENTS**

The interface provided to the user should be a very user-friendly one and it should provide an optional interactive help for each of the service listed. The interface provided is a menu driven one and the following screens will be provided:-

1. A login screen is provided in the beginning for entering the required username/pin no. and account number.

2. An unsuccessful login leads to a reattempt screen to again enter the proper information. The successful login leads to a screen displaying a list of supported options from which a user can select any one.

3. After the login, a screen with a number of options is then shown to the user. It contains all the options along with their brief description to enable the user to understand their functioning and select the proper option.

4. A screen will be provided for the user to perform various transactions in his account such as cash withdrawals/deposits, check balance, change PIN etc.

5. A screen will be provided for user to withdraw cash or another screen to deposit cash which will also generate a Transaction ID post a successful transaction.

6. A screen will be provided for user to check his account balance.

7. A screen will be provided where the user should be able to generate a Mini Statement with about 8 recent transactions listed.

The following reports will be generated after each session dealt with in the machine: 1. Login time and Logout time along with the user’s pin no and account number is registered in the database.

2. The ATM’s branch ID through which the session is established is also noted down in the database.

3. Various changes in the user’s account after the transactions, if any, are reported in the database.

4. An alert is generated for the user displaying all the transactions he performed.

**2.1.2 SOFTWARE REQUIREMENTS**

The software that needs to be installed in order to run this software are XAMPP server software(In which Apache server is needed to run the PHP/HTML interface and My SQL database to store all the information), any web browser such as Google Chrome to open the website interface. Using these 2 software’s is sufficient to run this software system.

# 2.2 FUNCTIONAL REQUIREMENTS

There are a few functional requirements for this system to work. They are:

1. The account information must be validated before the user can access his account information to carry out any transactions.
2. Valid errors must be provided if account number or PIN is incorrect.
3. A cancel button must be present in every screen so that the user can terminate the session at any point.
4. A menu with all the available transactions should be displayed on a screen for a user to select from.
5. The keypad must accept input from the user.
6. The withdraw option must check if sufficient funds are available in a user’s account before proceeding with the withdrawal transaction.
7. The amount entered must be in multiples of 100s and this must also be checked by the system.
8. A transaction ID must be generated for each transaction that takes place.
9. Account balance information must be updated after each transaction takes place

10) Mini Statement must be generated of at least 5 recent transactions and must be

presented in the form of a table.

11) User must be able to change his PIN code successfully which must also be

updated in the database.

12) After a successful transaction, it must be redirected back to the start page.

These are few of the functional requirements of the project.

# 3. SYSTEM DESIGN

## 3.1 E-R DIAGRAM

## 3.2 USER INTERFACE DESIGN

# 4. DETAILED DESIGN

## 4.1 DATABASE CREATION

### 4.1.1 DATA DEFINITIONS

### 4.1.2 CONSTRAINTS

# TABLE OF CONTENTS

[Certificate 1](#_Toc419247224)

[Acknowledgement 2](#_Toc419247225)

[Abstract](#_ABSTRACT) 3

[1. INTRODUCTION 4](#_1._INTRODUCTION)

[1.1 INTRODUCTION TO DBMS 4](#_1.1_INTRODUCTION_TO)

[1.1.1 SQL PROGRAMMING 4](#_1.1_INTRODUCTION_TO)

[1.1.2 Working with SQL COMMANDS 5](#_1.1_INTRODUCTION_TO)

[1.2 BRIEF OUTLINE OF THE PROJECT 6](#_1.2_BRIEF_OUTLINE_1)

[2. REQUIREMENT SPECIFICATIONS](#_2._REQUIREMENT_SPECIFICATIONS) 7

[2.1 EXTERNAL INTERFACE REQUIREMENTS 7](#_2.1_EXTERNAL_INTERFACE)

[2.2 FUNCTIONAL REQUIREMENTS](#_2.2_FUNCTIONAL_REQUIREMENTS) 8

3.SYSTEM DESIGN……………………………………………………………9

[3.1 ER DIAGRAM 1](#_Toc419247244)5

[3.3 USER INTERFACE DESIGN 18](#_Toc419247247)

[4. DETAILED DESIGN 19](#_Toc419247248)

[4.1 DATABASE CREATION 19](#_Toc419247249)

[4.1.1 Data Definitions 19](#_Toc419247250)

4.1.2.constraints………………………………………………………………….

[4.2.STORED PROCEDURES 20](#_Toc419247252)

4.3.TRIGGERS

**5.DATABASE CONNECTIONS & CODE IMPLEMENTATION**………………………….14

5.1 Retrieving data from the database……………………………….......................................19

5.2 Inserting data into the database…………………………………….................................22

5.3 Updating records into the database……………………………………………………….24

5.4 Deleting data from the database…………………………………..................................26

[6. System and Database Testing](#_Toc419247253) 26

[6.1](#_Toc419247254)  [TEST RESULTS/SNAPSHOTS](#_Toc419247257) 28

[7. CONCLUSION AND SCOPE FOR FURTHER ENHANCEMENTS](#_Toc419247260) 34

[8. APPENDIX 3](#_Toc419247261)5

[8.1APPENDIX-I 3](#_Toc419247262)5

[8.1.1 Bibliography 3](#_Toc419247263)5

[8.2 APPENDIX-II 3](#_Toc419247264)6

[8.2.1 Development Tools 3](#_Toc419247265)6

[8.2.2 Software Environment 3](#_Toc419247266)6

[8.2.3 Hardware Environment 3](#_Toc419247267)6

# 8. APPENDIX

**8.1APPENDIX-I**

### 8.1.1 Bibliography

**8.1.1.1 Reference Books/Papers**

**8.1.1.2 Web Sites**

**8.2 APPENDIX-II**

### 8.2.1 Development Tools

### 8.2.2 Software Environment

### 8.2.3 Hardware Environment

|  |  |
| --- | --- |
| **System** | |
| Manufacturer | HP |
| Model | HP Pavilion g6 Notebook PC |
| Total amount of system memory | 4.00 GB RAM |
| System type | 64-bit Operating System |
| Processor | Intel® Core i5-3230 CPU 4 x 2.60 GHz |
| **Storage** | |
| Total size of hard disk | 500 GB |
| 1. Disk partition (C:) | 40.8 GB Free (371 GB Total) |
| Disk partition (D:) | 2.10 GB Free (16.7 GB Total) |
| Disk partition (H:) | 5 GB Free (34 GB Total) |
|  | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |