CERTIFICATE

This is to certify that the project on **"BANK MANAGEMENT SYSTEM"** is a bonafide work done by M. VINOD KUMAR with Regd.No.0320014055 during the year 2022 in partial fulfilment of the requirements for the award of degree in B.Sc., [M. P. Cs,].

This project work carried by him under our supervision.

K Dhumeeswari.
Project Guide

G. M. Sha
12/7/22
HEAD OF THE DEPARTMENT

"G M SHANMUGAM M.Sc., M.ED., Ph.D."

Head of the department,

Department of Computer Science.

Submitted for viva voce examination held on ________________________.

Examiner.

Acknowledgement

We are very thankful for our college management and our Principal Mr. P. Babu M.Sc., M.Phil., Ph.D. and vice principal, Head of the Department of Computer Science SVCR Govt degree college Mr. G M Shanmugam M.Sc., M.ED., Ph.D." and Mr. M Surya Sekhar Reddy M.Sc., M.Phil., M. ED, (Ph.D.). who has been showing keen interest in our welfare, for providing the necessary facilities for pursuing this project work.

I humble to submit my deep sense of gratitude and sincere thanks to Mrs. k Muneeswari M.C.A, and Mr. K Vijaya Kumar M.Sc., M.C.A and Mrs. N. Madhavi M.C.A, lecturers in Computer Science for giving valuable advice and guidance in my course of study and also in completing my project. With an opportunity to build our career in computer science and information technology.

I express my deep sense of gratitude and high records to Mrs. k Muneeswari M.C.A, and Mr. K Vijaya Kumar M.Sc., M.C.A lecturers in computer science for their valuable instructions encouragement and cooperation extended to my course of study during the year 2022. I also express my sincere thanks to staff members of SVCR Govt degree college for their valuable cooperation and support extended to our study in the college and in the project work.

Last but not least, our whole hearted love to my parents who guided us to choose the right path to join this course, family members and friends for their continuous support, inspiration and encouragement.

Yours

M.VINOD KUMAR

Executive Summary

The Bank Account Management System is an application for maintaining a person's accountin a bank. In this project I tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System. To develop a project for solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks. Also to enable the user's workspace to have additional functionalities which are not provided under a conventional banking project.

The Bank Account Management System undertaken as a project is based on relevant

Technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this Software. This project is developed using PHP, HTML language and MYSQL use for Database connection. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a

Contractual relationship. Organization need to effectively define and manage requirements to

Ensure they are meeting needs of the customer, while proving compliance and staying on the Schedule and within budget. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment.

The project analyses the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications.

The system is then designed in accordance with specifications to satisfy the requirements.

The system design is then implemented with MYSQL, PHP and HTML. The system is designed as an interactive and content management system. The content management system deals

with data entry, validation confirm and updating whiles the interactive system deals with system interaction with the administration and users.

Thus, above features of this project will save transaction time and therefore increase the efficiency of the system.

Contents

Chapter 1

Chapter 2

2.1.4. Product Scope

2.1.5. References

2.2. Overall Description

2.2.1. Product Perspective

2.2.2. Product Functions

2.2.3. User Classes and Characteristics

2.2.4. Operating Environment

2.2.5. Design and Implementation Constraints

2.2.6. User Documentation

2.2.7. Assumptions and Dependencies

2.3. External Interface Requirements

2.3.1. User Interfaces

2.3.2. Hardware Interfaces

2.3.3. Software Interfaces

2.3.4. Communications Interfaces

2.4. System Features

2.4.1. System Feature 1.

2.4.1.1. Description and Priority.

2.4.1.2. Stimulus/Response Sequences

2.4.1.3. Functional Requirements

2.4.2. System Feature

2.4.2.1. Description and Priority

2.4.2.2. Stimulus/Response Sequences.

2.4.2.3. Functional Requirements

2.4.3. System Feature 3 (and so on)

2.5. Other Nonfunctional Requirements

2.5.1. Performance Requirements

2.5.2. Safety Requirements

2.5.3. Security Requirements

2.5.4. Software Quality Attributes

2.5.5. Business Rules

2.6. Other Requirements

Chapter 3

Use Case Analysis

3.1. Use Case Model

3.2. Fully Dressed Use Cases

Chapter 4

System Design

4.1. Architecture Diagram

4.2. Domain Model

4.3. Entity Relationship Diagram with data dictionary

4.4. Class Diagram

Chapter 1:

Introduction

The "Bank Account Management System" project is a model Internet Banking Site. This site enables the customers to perform the basic banking transactions by sitting at their office or at homes through PC or laptop. The system provides the access to the customer to create an account, deposit/withdraw the cash from his account, also to view reports of all accounts present. The customers can access the banks website for viewing their Account details and perform the transactions on account as per their requirements. With Internet Banking, the brick and mortar structure of the traditional banking gets converted into a click and portal model, thereby giving a concept of virtual banking a real shape. Thus today's banking is no longer confined to branches. E-banking facilitates banking transactions by customers round the clock globally.

The primary aim of this "Bank Account Management System" is to provide an improved design methodology, which envisages the future expansion, and modification, which is necessary for a core sector like banking. This necessitates the design to be expandable and modifiable and so a modular approach is used in developing the application software. Anybody who is an Account holder in this bank can become a member of Bank Account Management System. He has to fill a form with his personal details and Account Number.

Bank is the place where customers feel the sense of safety for their property. In the bank, customers deposit and withdraw their money. Transaction of money also is a part where customer takes shelter of the bank. Now to keep the belief and trust of customers, there is the positive need for management of the bank, which can handle all this with comfort and ease.

Smooth and efficient management affects the satisfaction of the customers and staff members, indirectly. And of course, it encourages management committee in taking some needed decision for future enhancement of the bank. Now a day's, managing a bank is tedious job up to certain limit. So software that reduces the work is essential. Also today's world is a genuine computer world and is getting faster and faster day-by-day. Thus, considering above necessities, the software for bank management has become necessary which would be useful in managing the bank more efficiently. All transactions are carried out online by transferring from accounts in the same Bank or international bank. The software is meant to overcome the drawbacks of the manual system.

1.1. Background

Bank Account Management System keeps the day by day tally record as a complete banking system. It can keep the information of Account type, account opening form, Deposit fund,

Withdrawal, and Searching the transaction, Transaction reports, Individual account opening form, Group Account. The existing part of this project is; it displays Transaction reports, Statistical Summary of Account type and Interest Information.

1.2. Motivations and Challenges

The main aim of designing and developing this Internet banking System PHP primarily based Engineering project is to provide secure and efficient net banking facilities to the banking customers over the internet. Apache Server Pages, MYSQL database used to develop this bank application where all banking customers can login through the secured web page by their account login id and password. Users will have all options and features in that application like get money from western union, money transfer to others, and send cash or money to inter banking as well as other banking customers by simply adding them as payees.

1.3. Goals and Objectives

I. Main goals:

- Our motto is to develop a software program for managing the entire bank process related to Administration accounts customer accounts and to keep each every track about their property and their various transaction processes efficiently.
- Hereby, our main objective is the customer's satisfaction considering today's faster in the world.

II. Customer Satisfaction:

- Client can do his operations comfortably without any risk or losing of his privacy.

- Our software will perform and fulfill all the tasks that any customer would desire.

III. Saving Customer Time:

□ Client doesn't need to go to the bank to do small operation.

IV. Protecting the Customer:

□ it helps the customer to be satisfied and comfortable in his choices, this protection contains customer's account, money and his privacy.

V. Transferring Money:

□ Help client transferring money to/or another bank or country.

VI. Protecting The Customer:

1.4. Gap Analysis

The criteria for selecting these banks were their deposit base. We have limited our Service Category to the core services in Retail Banking and a few specialized services. The report is a mixture of Secondary and Primary data, with Questionnaires being our major instrument to collect primary data. Major topics we have attempted to cover in this project are to:

- Explore the services and products offered by the banks to individual customers.
- Understand the perception of the manage ment with respect to services offered by banks.
- Understand the perception of the customers with respect to services offered by banks.

1.5. Proposed Solution

The Traditional way of maintaining details of a user in a bank was to enter the details and record them. Every time the user needs to perform some transactions he has to go to bank and perform the necessary actions, which may not be so feasible all the time. It may be a hardhitting task for the users and the bankers too. The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain. Here, we provide automation for banking system through Internet. Online Banking System project captures activities performed by different roles in real life banking which provides enhanced techniques for maintaining the required information up-to-date, which results in efficiency.

The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain.

1.6. Project Plan

Online banking helps you become more of a banker, running your accounts like a small business that you control every day. Once you get started, you'll be hooked. Soon enough you'll be checking your bank account as often as your e-mail.

1.6.1. Work Breakdown Structure

If you want to try out online banking without committing, select our Online Banking. You don't have to register in any way, so it's a good way to check it out first before register.

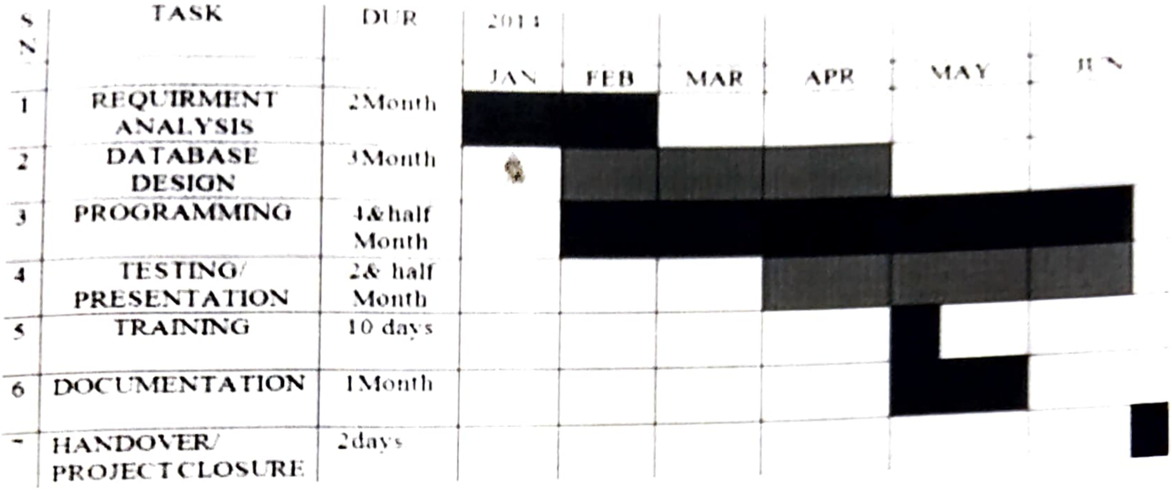
Once you register, you'll have the choice of doing just basic banking and viewing your balance or doing more involved transactions like bill payments and transfers. The choice is yours. It really depends on how you like to bank.

You will get a confirmation number after each transaction and you can always check the session summary to see what you've done. If you make a mistake, customer service is always available for your good kindness help.

1.6.2. Roles & Responsibility Matrix

Categories of Attributes	Responsible	Accountable	Consulted	Informed
Product hierarchies	Merchandising, Finance	Merchandising	Marketing	Store Operations, e-commerce, IT
Global Trade Item Number (GTIN)	Vendor	Merchandising	Finance	Store Operations, e-commerce, IT
Product description	Vendor, Marketing	Merchandising		IT
Product type		Merchandising	Finance	IT
Vendor name and identifier		Merchandising	Finance, Supply Chain	Store Operations, e-commerce, IT
Vendor certifications	Assurance	Merchandising		Store Operations, IT
Hazardous materials/ Recycling content	Vendor	Merchandising	Supply Chain, Assurance	Store Operations, Legal, IT
Weight and dimensions		Merchandising	Supply Chain	Store Operations, Finance, IT
Packaging		Merchandising	Supply Chain, Quality Assurance	IT
Product images	e-commerce, Marketing	Merchandising	Store Operations, Customer Service, Quality Assurance	Legal, IT
Costing and pricing	Merchandising, Finance	Merchandising	Store Operations, e-commerce, Customer Service	Information security, IT
Product relationships	Marketing	Merchandising		IT
Product status		Merchandising	Marketing, Finance	IT
Promotions		Merchandising	Marketing, Finance	IT

1.6.3. Gantt chart



S N	TASK	DUR	2014 JAN	FEB	MAR	APR	MAY	JUN
1	REQUIRMENT ANALYSIS	2Month	■	■				
2	DATABASE DESIGN	3Month		■	■	■		
3	PROGRAMMING	4&half Month		■	■	■	■	■
4	TESTING/ PRESENTATION	2& half Month				■	■	■
5	TRAINING	10 days					■	
6	DOCUMENTATION	1Month					■	
7	HANDOVER/ PROJECTCLOSURE	2days						■

1.7. Report Outline

- User registration for online banking if not register.
- Adding Beneficiary account by customer.
- Transferring amount to the local customer account number.
- Admin must approve the user account activation before it can be used and transferring funds, view statement history.
- Customer gets to know his last login date and time each time he logs in.
- Customer can check all transactions made with their account.
- Customer can check their account statement within a date range.
- Customer can request for ATM and Check Book.

- Admin can add/edit/delete customer accounts
- All two of them (customer & admin) can change their password.
- Admin Login pages are hidden from customer for security purpose.
- Passwords are stored as encrypted hashes with an additional random salt for added security.

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

Aim To develop a software for solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks. Also to enable the user's workspace to have additional functionalities which are not provided under a conventional banking software.

2.1.2. Document Conventions

This project is about the Banking System. The project has the complete information regarding the account details (including create, withdrawal, deposit, summary etc.). It also contains the information about the different customers opening their account in the bank. This project also helps to keep the information about all the details of the various customers who have opened their account. This project is stand alone and the various centers of banking can implement connected through the network (only WAN) and only authenticated Person of the Bank can check the details of the account holder, he can also create saving, fix, current account of the customer. Due to centralized database only authenticated Person of The Bank can handle the application.

2.1.3. Intended Audience and Reading Suggestions

The objective is to prepare a software or application, which could maintain data & provide a user friendly interface for retrieving customer related details just in few seconds, with 100%accuracy. Software is completely computerized, so it is not time consuming process. No paper work required & can be implemented further. The application should also facilitate the addition of new Customer A/c, deletion of A/c& modification of existing customer A/C. To Search for every individual accounts for a particular customer, show all transaction & any account should be opened with minimum Rest. 500 etc.

2.1.4. Product Scope

The scope of the Bank Management System extends to all the users who wish for easy banking facilities. This software product will be used for storing user's account information and the transactions made by them.

2.2. Overall Description

Here are some of the features available through online banking:

1. View balances: Firstly login your account with your account number and password.
2. Then checking your balance doesn't require much work. You simply select Account balances and take a look at your balance and past transactions. If you have more than one account, you can also do transfers between accounts.
3. Transfer funds: When you select Transfer Funds, you'll be asked where to transfer the money to and from, when, and the amount.

4. Set up recurring bill payments or transfers: If you make a regular payment every month, it might be convenient to set up an automatic withdrawal from your account.

5. Monitor CIBC investments: If you have any CIBC investments, you can keep an eye on those stocks or mutual funds here.

6. Pay bills: To pay your bills online, you just need to add to your account the names of the companies you wish to pay bills to.

7. View our VISA* accounts: Always a good place to monitor your spending. You can make your credit card payments online, right from your account.

8. Order Checks: We don't need them much anymore due to online banking and debit purchases, but if you still use checks, you can order them directly from the BAMS website.

2.2.1. Product Perspective

Today, banks are looking beyond the transactions to the full opportunity on how to manage their customers. Accordingly, they are moving beyond managing clients as simple contacts to a whole new level of client relationship management, crafting a superior commercial client experience that gives the bank a competitive advantage and a more loyal, profitable and committed customers. Internet Banking System refers to systems that enable bank customers to access accounts and general information on bank products and services through a personal computer or other intelligent device. But most of these systems do not focus on how best to manage and keep their customer’s data more secured.

2.2.2. Product Functions

The Modules description of Bank Account Management System project. These modules will be developed in PHP source code and MYSQL database.

1. Create New Account: A customer who having the account in the world can create a virtual account through this module. This module receives the customer profile details and the bank account details with the proof of the ownership of the bank account.

2. Login: Virtual account holders can login in to the system using this module. Thus this is the secured login page for the customers in the website.

3. Virtual Account: After the approval of new virtual account creation, the customer assigned a unique virtual account number to make the online money transactions. This module views the details of the logged customer's virtual account.

4. Bank Accounts: A customer may have more than one bank account in various banks, in this case, the customer prompted to decide which bank account should reflect in the account debit or amount credit. For these operations customers can add their owned bank accounts here and it will be approved by the administrations of the system.

5. Fund Transfer: This is the module to make fund transfer to the virtual bank account holders or the usual bank account holders from the customer's specified bank account.
6. Beneficiary: Beneficiary is a person who receives money. Here the customer can add the beneficiaries to make fund transfer in the future.

7. Transactions: This module displays the transactions made by the customer in the particular date with the transaction details.

8. Administrative Control: This module contains the administrative functions such as view all virtual account, transactions, approve bank accounts, approve virtual accounts etc.

2.2.3. User Classes and Characteristics

<u>System specification</u>

Hardware requirements:

Processor	: Intel Pentium III or later
Main Memory (RAM)	: 256 MB
Cache Memory	: 512 KB
Monitor	: 14 inch Color Monitor
Keyboard	: 108 Keys
Mouse	: Optical Mouse
Hard Disk	: 160 GB

Software requirements:

Front End/Language	: PHP
Back End/Database	: MYSQL
Additional Tools	: XAPM Server
Operating System	: Windows 7, 8, 9, 10, XP

2.2.4. Operating Environment

Software interface is supported by the main control panels and operating system in which hosts the algorithms for calculating distributed travel and wait time information. Additionally, the algorithms define and export system commands for main control panels, and communication mediums. For testing purposes the software shall be capable of interfacing with software simulators on a PC computer using GUI applications of webpages.

2.2.5. Design and Implementation Constraints

The information of all the users must be stored in a database that is accessible by the Online System. The Online Banking System is connected and is running all 24 hours a day. The users access the Online System from any computer that has Internet browsing capabilities and an Internet connection.

The users must have their correct usernames and passwords to enter into the Online Dictionary System. The project is safety critical. Under no circumstances shall a user of the system be harmed or harm others through proper or improper use of the online. The project shall conform to any rules for Online Banking in place in the United States of America.

2.2.6. User Documentation

All system interfaces communicate in order to activate ordered requests. The communication mediums (wired or wireless) are the external interface that communicates with the control panel of the Online Banking System. This communication allows for failure messages, and requests to be sent and received by the main system.

2.2.7. Assumptions and Dependencies

2.3. External Interface Requirements

Some external requirement are as follows:

2.3.1. User Interfaces

A simple user can access their account and can deposit/withdraw money from their account. User can also transfer money from their account to any other bank account. User can see their transaction report and balance enquiry too. registration (local/international/domestic) User login, use PIN system View statements transaction Funds transfer User account details Change Password and PINCreating/open new account

- User login, use PIN system
- Creating/open new account registration
- Funds transfer (local/international/domestic)
- View statements transaction
- User account details
- Change Password and PIN
- View about developer details

2.3.2. Hardware Interfaces

The software shall interface with the electromechanical that controls the online connection systems. The software shall interface with a breaking mechanism in case of emergencies. The transactions and accesses shall be controlled by the software based on command and graphical user inputs. The hardware interface is supported by the main control panels (buttons, keyboard, mouse and communication mediums)

2.3.3. Software Interfaces

Software interface is supported by the main control panels and operating system in which hosts the algorithms for calculating distributed travel and wait time information. Additionally, the algorithms define and export system commands for main control panels, and communication mediums. For testing purposes the software shall be capable of interfacing with software simulators on a PC computer using GUI applications of webpages.

2.3.4. System Feature

2.3.4.1. Description and Priority

There are other features and actions that can be performed on a back account but we are not going to look at bank accounts in their entirety only the basics, this way we avoid over complicating the exercise. The purpose of this whole exercise is to show the usefulness of object oriented programming as opposed to really wanting to create a banking system.

2.3.4.2. Functional Requirements

- **Customer** The valid customer on internet banking has a set of requirements he/she does on internet banking. These requirements are offered on next pointes.

- **Login** A customer to be able to use this system, he/she has to enter username and password which he/she has created before and been saved in the database in the Login page. This function might be a customer or an Admin also

- **View Account** View Account allows to a customer to view today's up-to the minute balance information on deposit (saving/current), credit card, etc. The customer can also view transaction history with retention period up to a maximum of 90 days. Within this feature, the customer can request for account such as "view online, by e-mail or by post option. But the customer most be logged in the internet banking.

- **Transfer Funds** The customer can save up to a maximum of 10 accounts and update or delete the account details. All the outstanding future transfers are recorded in a table. The customer can enquire whether there is any funds transfer pending and. when the customer selects the Transfer funds, the system will display Menu to select Transfer

Funds function for transfer funds or Transfer History function for display the transaction he/she done.

- **Pay Bills** The customer most be logged into Banking System. With internet banking, customers can make payments to corporations that include utilities, assessments, Insurance, telecommunications, and other services. He /she doesn't have to enter his/her bill account number anymore. And remove bills from list of "Registered Payment" by using the Bill Deregistration function.

- **Check Services** The customer most be logged into Banking System. The customer may enquiries check status, whether it is paid, unpaid, stopped or returned. It also allows customer to stop check payment and to request for a check book online.

- **Utility** The customer most be logged into Banking System. Utility allows customer to change password and the secure delivery contact information. Within this feature, the customer can also change the online profile personal information that is retained by the internet banking system only. And the customer can cancel the ATM facilities.

- **Logout** The customer most be logged into Banking System. This function is used when a logged in user finishes his/her job and wants to be logged out so that no one can abuse his username. The system will state the user has been logged out successfully.

- **Administrator** An administrator is that person who makes some editing for the internet banking system like add/cancel customer, check the transactions etc. Therefore the administrator must have a username and password. In the project we will not go deep in an administrator because we will focus on the

2.4. Other Nonfunctional Requirements

Non-functional requirements are requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability, response time and store occupancy. They may specify system performance, security, availability, and other emergent properties. This means that they are often more critical than individual functional requirements. System users can usually find ways to work around a system function that doesn't really meet their needs. However, failing to meet a nonfunctional requirement can mean that the whole system is unusable. Non-functional requirements needed in this internet banking system are identified as performance requirements, safety requirements, security requirements and software quality attributes.

2.4.1. Performance Requirements

The Online Banking System shall be built upon an internet connection of server. The processor must be capable of handling real-time functionality activated by the defined users and communication medium. In addition, the system must be safety-critical. All failures reported by the communication medium must be handled instantaneously to allow for user and system safety.

The software shall control n-user in a building with m-services. The maximum number of commands the software shall handle is (m*n) +2*(m-1) +n, where m is the number of services and n is the number of users. The software shall have a communication time variable of x seconds, based on signal or web based inputs, which if exceeded, the software shall recognize an error and take corrective action

2.4.2. Safety Requirements

The Online Banking System shall run on an embedded system that handles safety-critical functionality. The system shall use a real-time processor with dynamic memory allocation in order to handle continuous activity. Also, user and software interfaces shall be simple and userfriendly, and comply with the following:

- **Standards Compliance:**
- The software shall adhere to Account Department codes and regulations, and Building codes related to public accounts safety.
- **Hardware Limitations:**
- This software shall run only on an internet, it must be easily transferable to the field. Admin perform the operation in online either offline.

2.4.3. Software Quality Attributes

Admin can access this project there is an authorization process. If you login as an Admin then you will be redirected to the Admin Home Page and if you are a simple user you will be redirected to your Account Home Page. This performs the following functions: Create Individual Accounts, Manage existing accounts, View all transactions, Balance enquiry,

Delete/close account etc.

- Admin login
- Add/delete/update account
- Withdrawal/deposit/statements transaction
- Account Information
- User details list
- Active/Inactive account
- View transaction histories

2.4.4. Business Rules

We understand that there is nothing more important than knowing that transactions are private and secure. Therefore, we have applied the very latest in technology when creating the Internet Banking security architecture. The best way to understand the security architecture within the Internet Banking is to take it one step at a time. These security features are described briefly below.

2.5. Other Requirements

A degraded mode of operation should be possible in which each user can operate independently of central scheduling. The software shall have power failure and error recognition codes acting as a safety net, thus keeping the software from performing any major catastrophic functions.

In the next stage it is to be decided that which medium is the most appropriate for the output. The main considerations when deciding about the output media are:

- The suitability for the device to the particular application.
- The need for a hard copy.
- The response time required.
- The location of the users

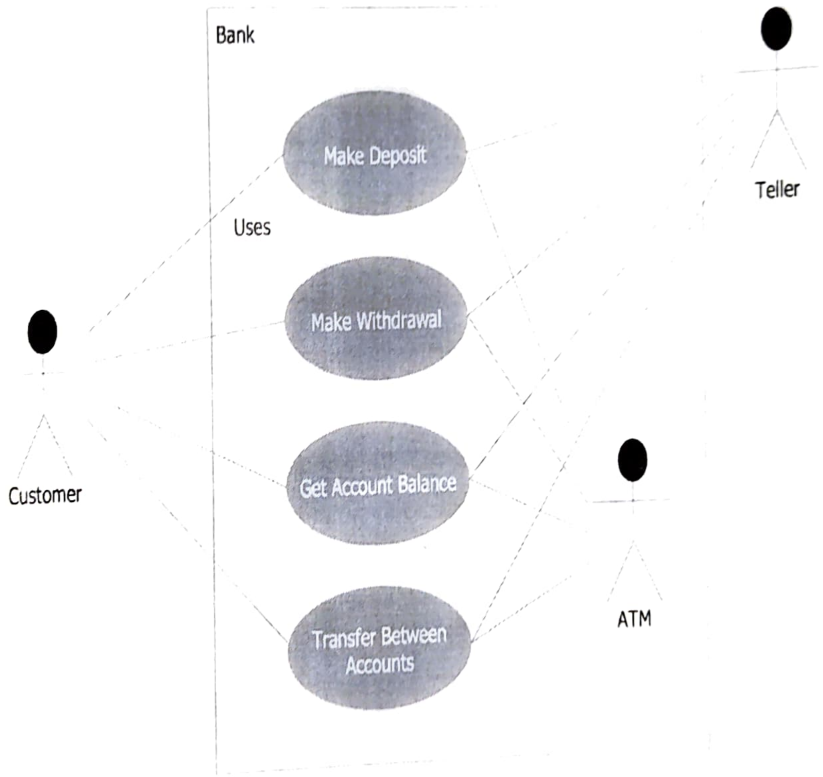
The software and hardware available. Keeping in view the above description the project is to have outputs mainly coming under the category of internal outputs. The main outputs desired according to the requirement specification are: The outputs were needed to be generated as a hot copy and as well as queries to be viewed on the screen. Keeping in view these outputs, the format for the output is taken from the outputs, which are

currently being obtained after manual processing. The standard printer is to be used as output media for hardcopies.

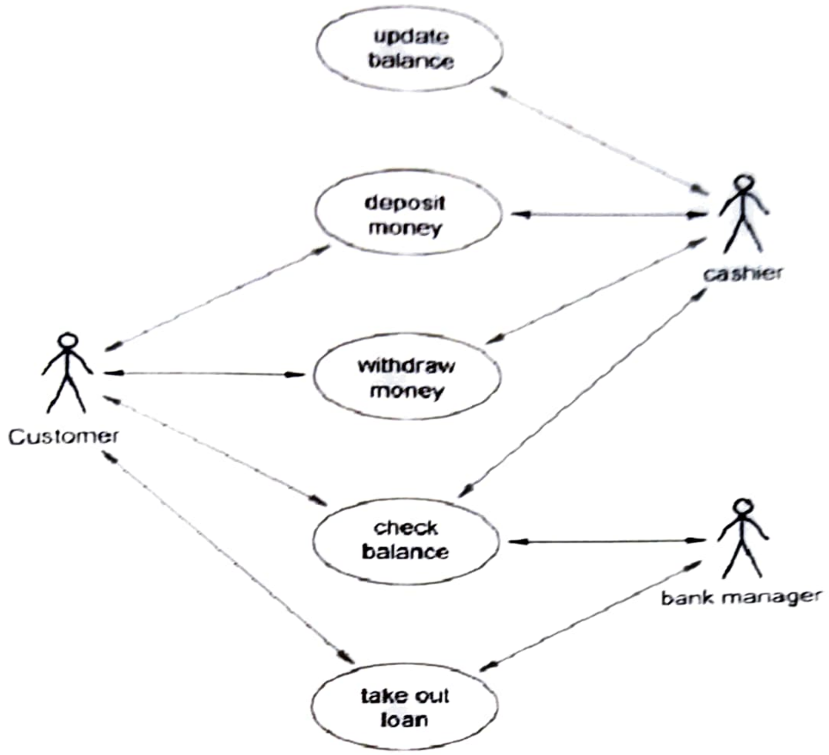
Chapter 3: System Analysis

This document details the software requirements for the Online Banking system project. It defines what the problem is and what problems a complete solution has to solve. The intended audiences for this document are the development team, the team manager, the customer and all other stakeholders in the system.

3.1. Use Case Model

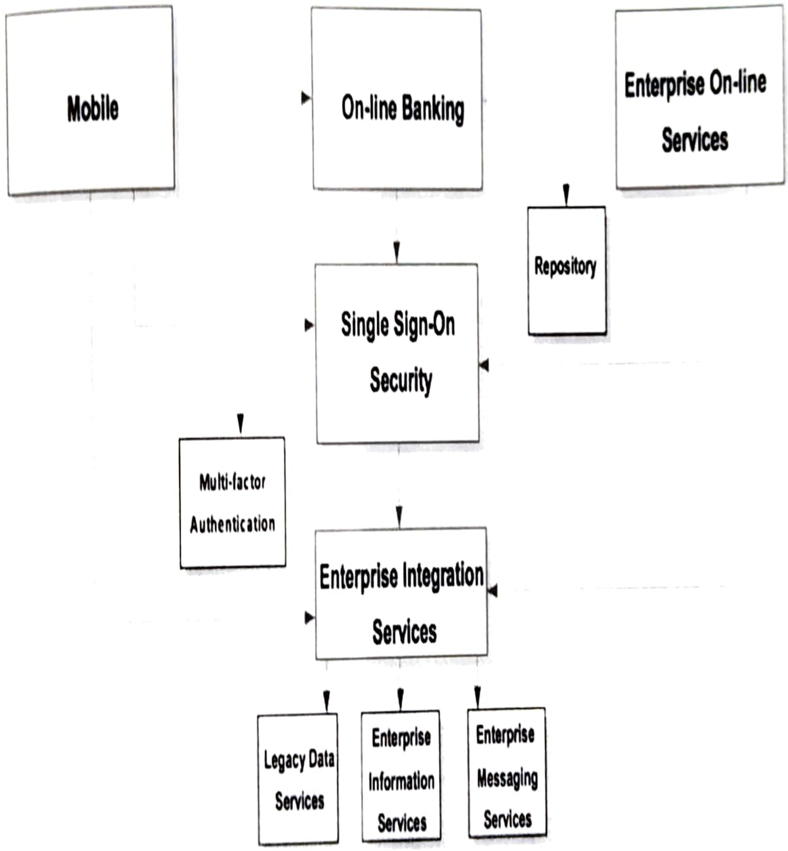


3.2. Fully Dressed Use Cases

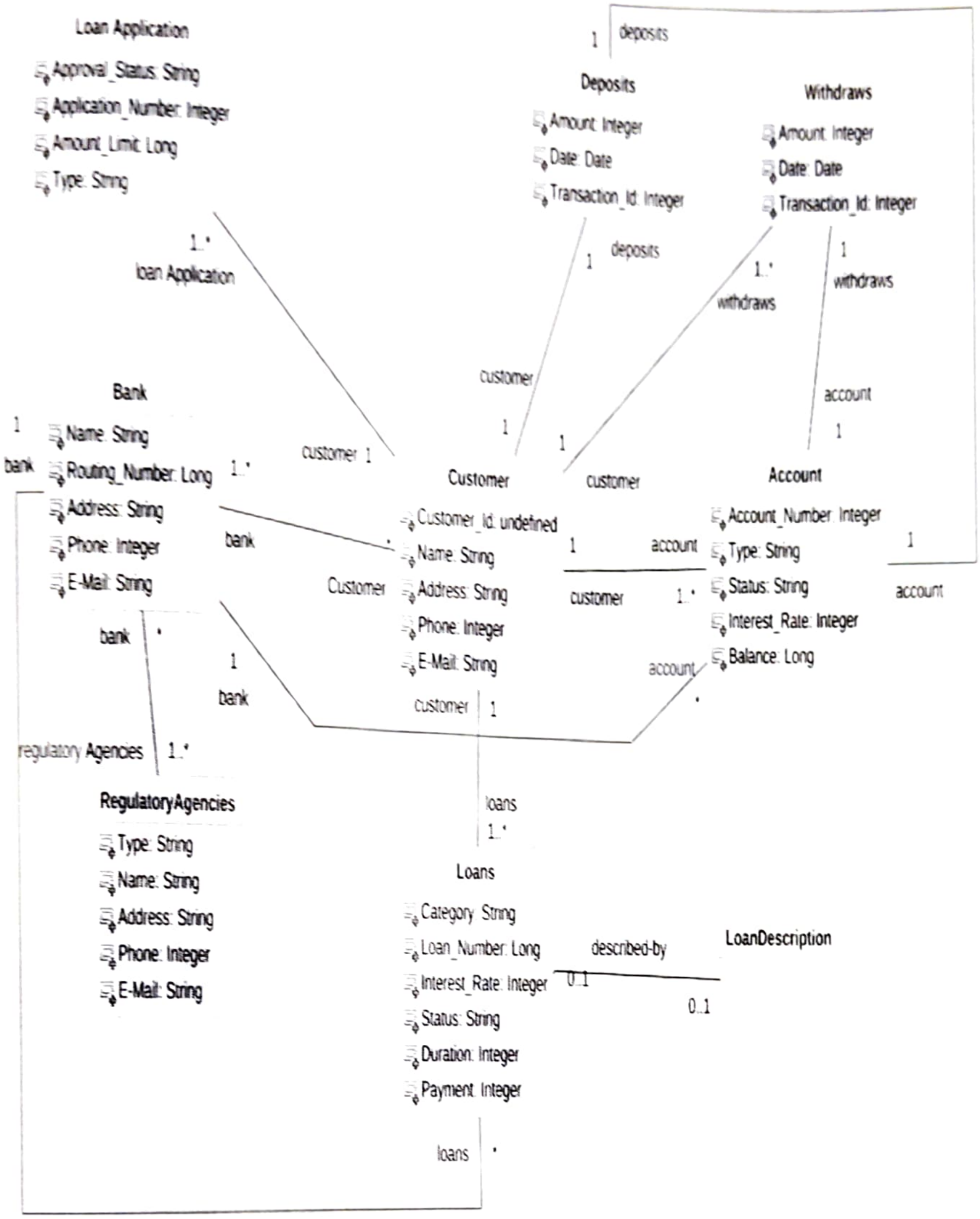


Chapter 4: System Design

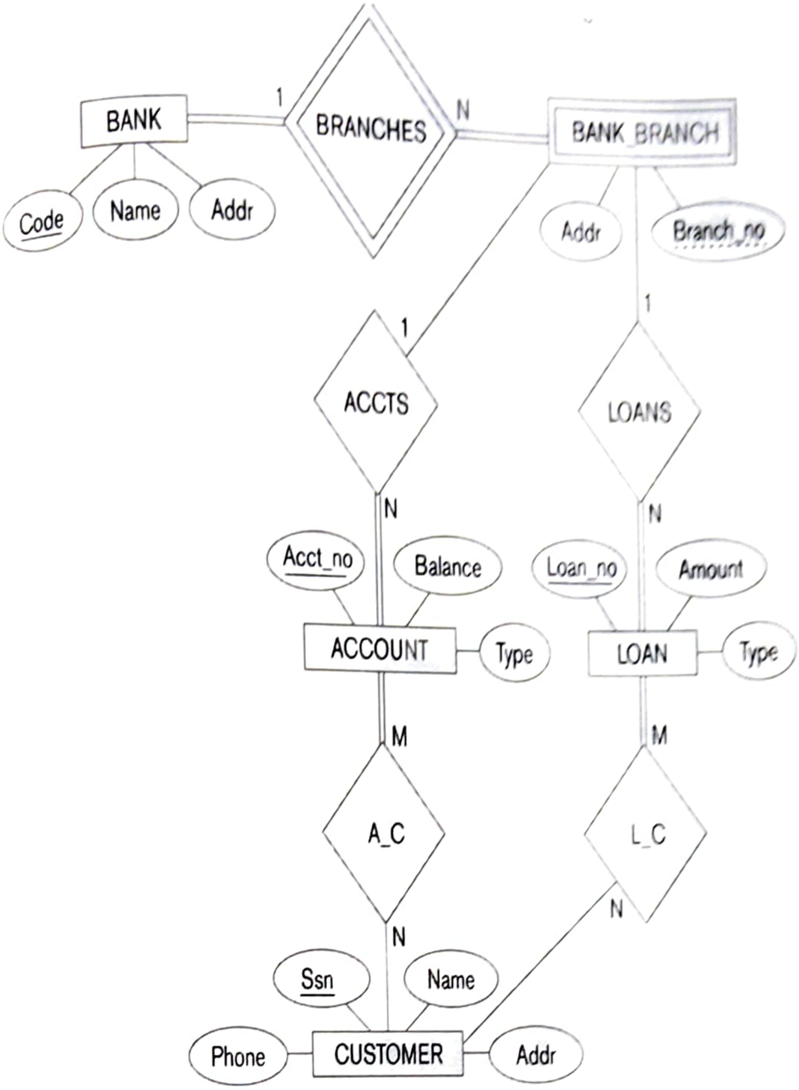
4.1. Architecture Diagram


Mobile
On-line Banking
Enterprise On-line Services
Repository
Single Sign-On Security
Multi-factor Authentication
Enterprise Integration Services
Legacy Data Services
Enterprise Information Services
Enterprise Messaging Services

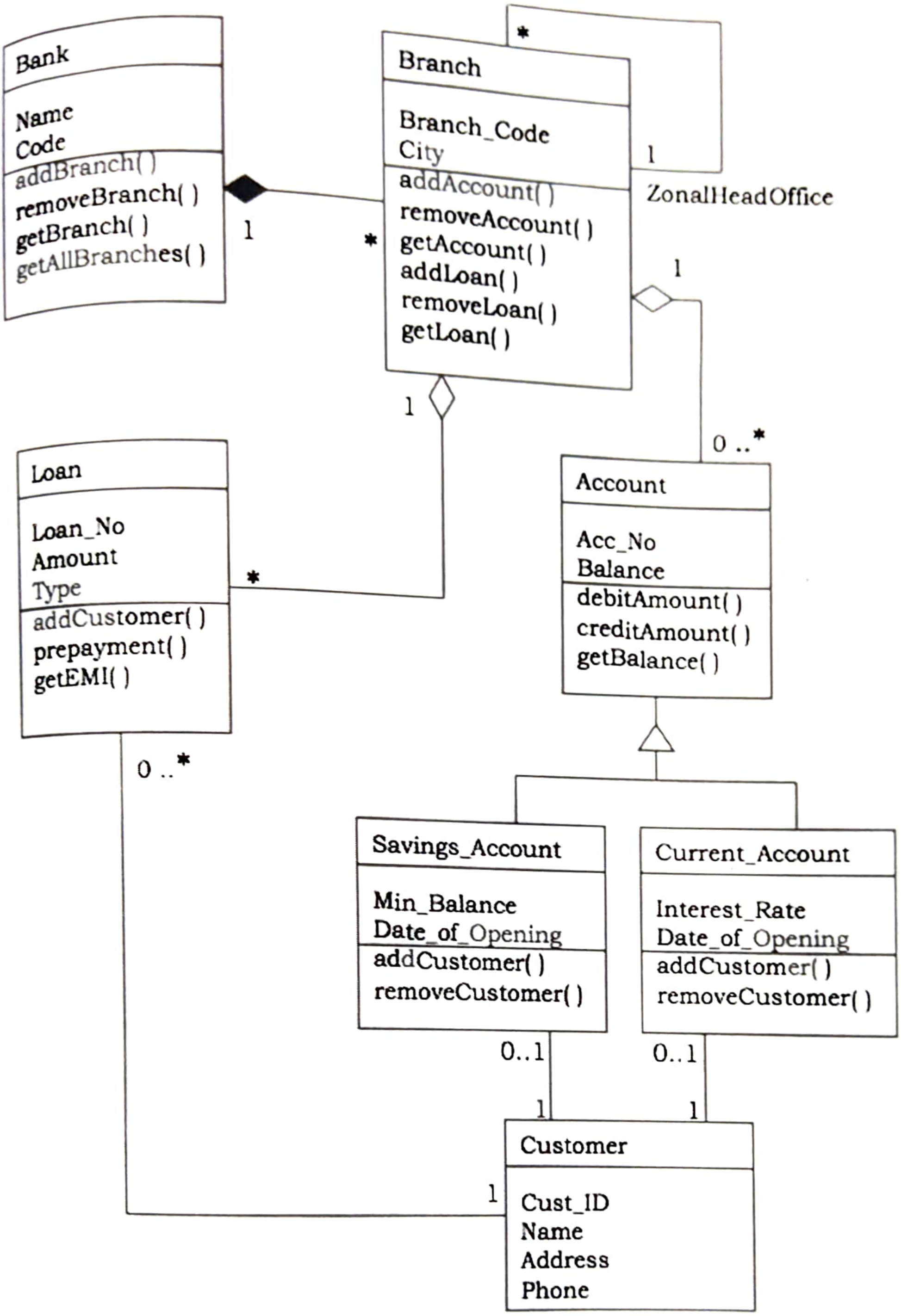
4.2. Domain Model



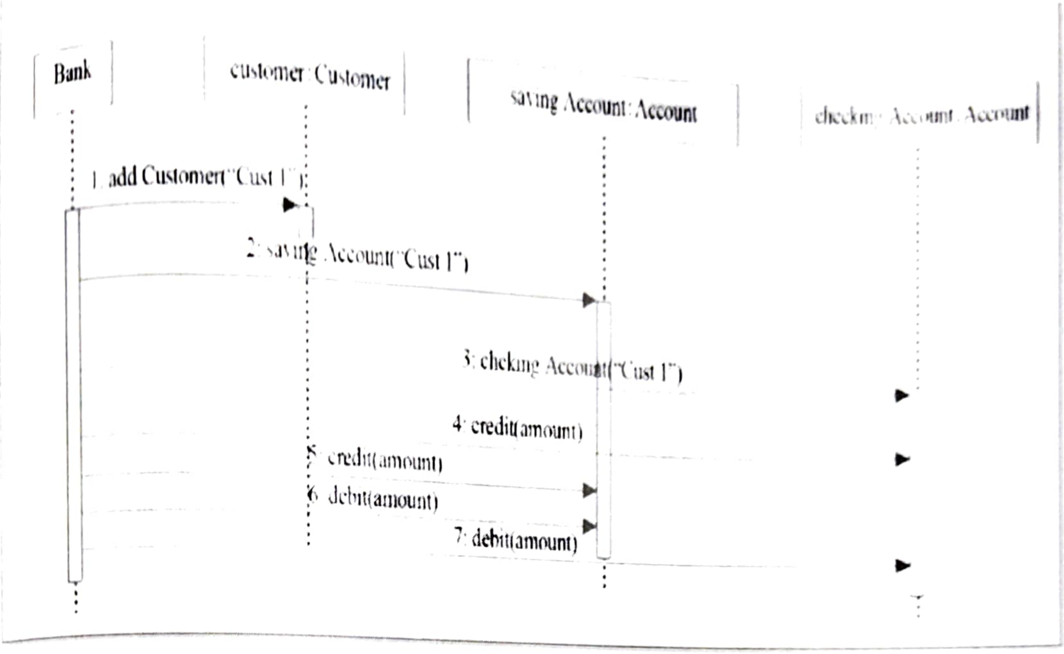
4.3. Entity Relationship Diagram with data dictionary



4.4. Class Diagram



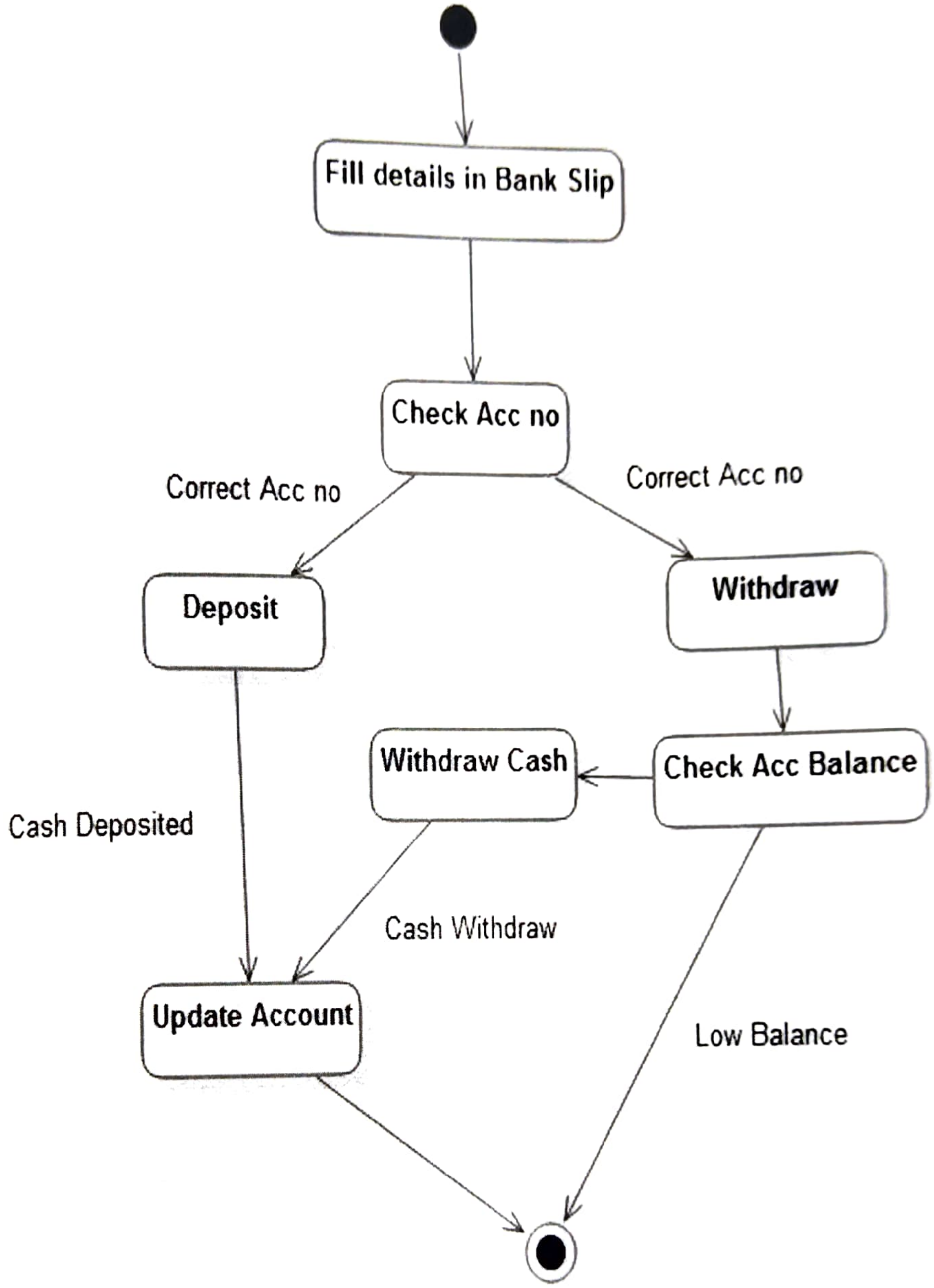
4.5. Sequence / Collaboration Diagram



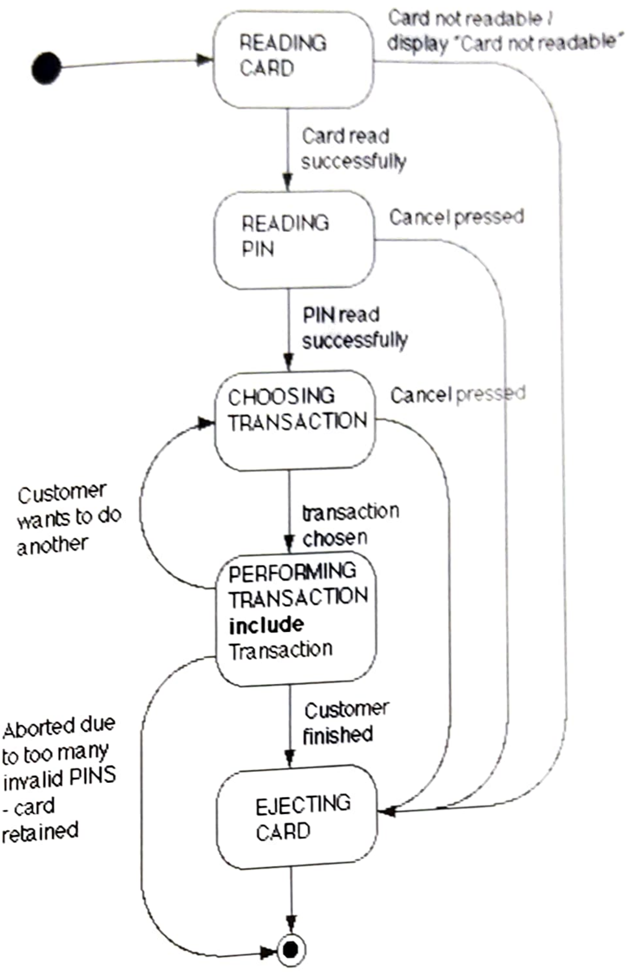
4.6. Operation contracts

- Contractor to manage a range of activities.
- Generally short term, usually for two to five years.
- Traditionally been favored as transitional arrangements for introducing the private sector into managing infrastructure but longer term operation and maintenance agreements are becoming more common in the water and energy sectors where more extensive participation by the private sector through a lease, afterimage or concession arrangement in these "essential services" is deemed to be too politically sensitive or impractical.
- Commonly found in the water sector and, to a more limited extent, energy sector.
- Limited potential for improvements in efficiency and performance although more sophisticated management contracts (which are often called operation and maintenance contracts) may introduce some incentives for efficiency or improved bill collection, by defining performance targets and basing a portion of the remuneration on their fulfillment (and cover longer time periods).
- Operator is usually paid a fixed fee to cover its staff and expenses. There may also be a performance based fee and liquidated damages for failure to achieve performance parameters.
- Operator may be required to collect bills on behalf of the utility and may accept some collection risk in terms of performance standards but is unlikely to collect bills on its own behalf.
- Can be useful where condition of assets is uncertain where the private sector would be unwilling to accept more extensive risk.

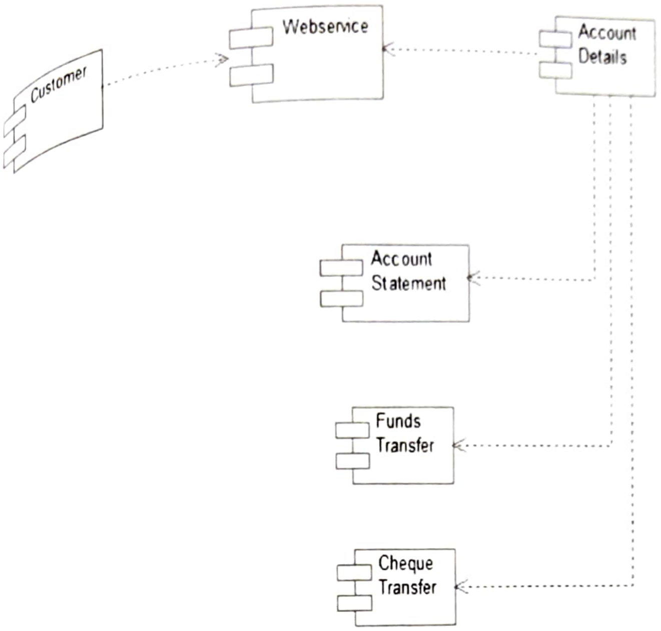
4.7. Activity Diagram



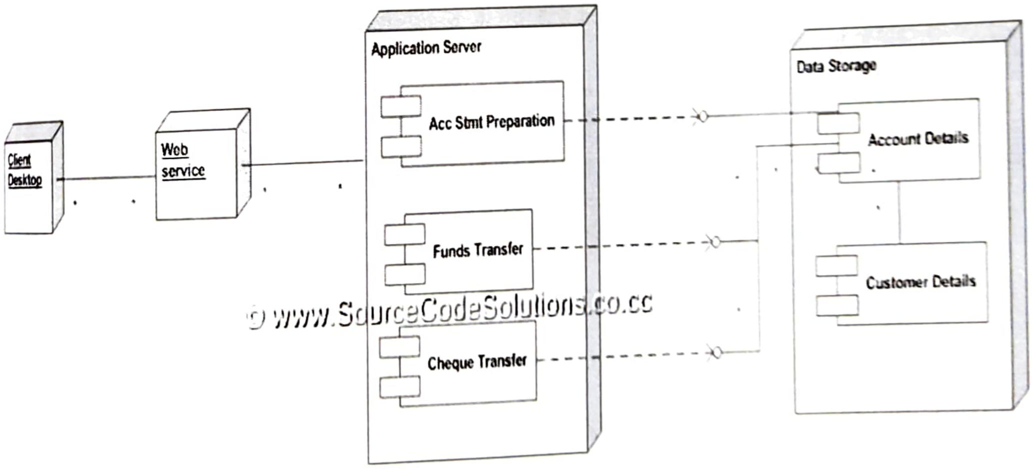
4.8. State Transition Diagram


READING CARD
Card not readable / display "Card not readable"
Card read successfully
READING PIN
Cancel pressed
PIN read successfully
CHOOSING TRANSACTION
Cancel pressed
Customer wants to do another
transaction chosen
PERFORMING TRANSACTION include Transaction
Aborted due to too many invalid PINS - card retained
Customer finished
EJECTING CARD

4.9. Component Diagram



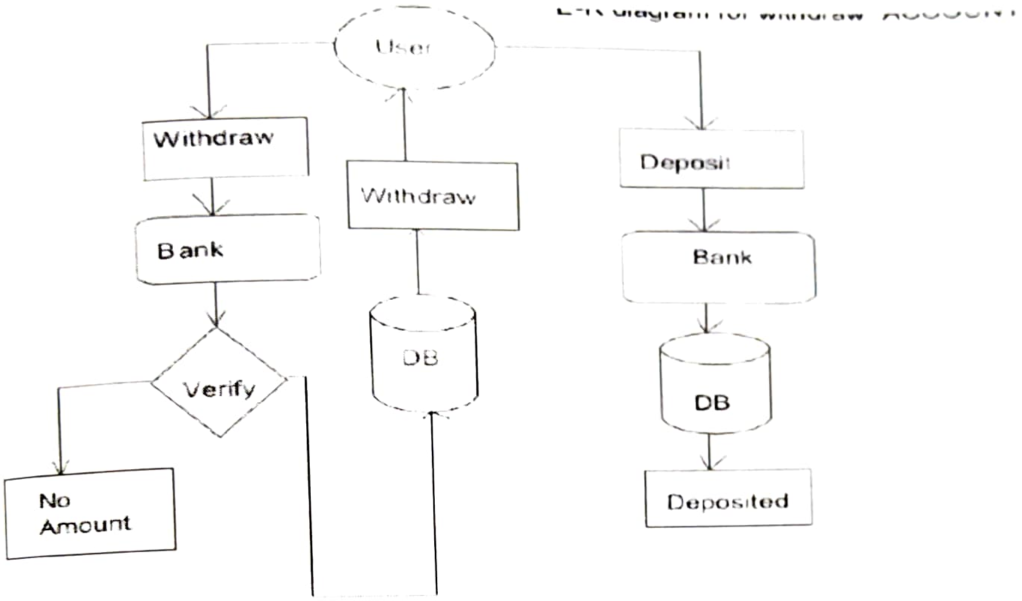
4.10. Deployment Diagram



Chapter 5: Implementation

Now we will implement the system which is detailed as above now the implementation details are as follows:

5.1. Important Flow Control/Pseudo codes



5.2. Deployment Environment

Banks are liable for whatever is done with their customers' savings and investments. Therefore, changes in the IT infrastructure are carefully controlled in order to deeply understand and measure the implications of the change. The life-cycle of systems in the banking sector is significantly longer than other business markets, such as telecommunications or theconsumer electronics markets.

5.3. Version Control

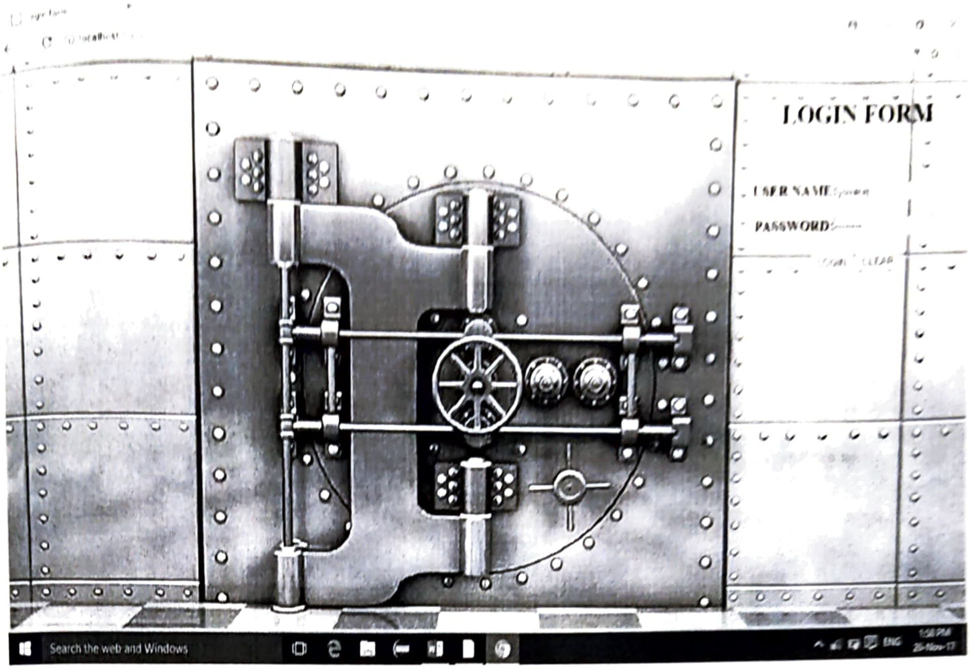
Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. For the examples in this book, you will use software source code as the files being version controlled, though in reality you can do this with nearly any type of file on a computer.

So, for version control we will use:

- Team Foundation Server.
- GitHub.
- Jedi VCS.

Screenshots of running Environment

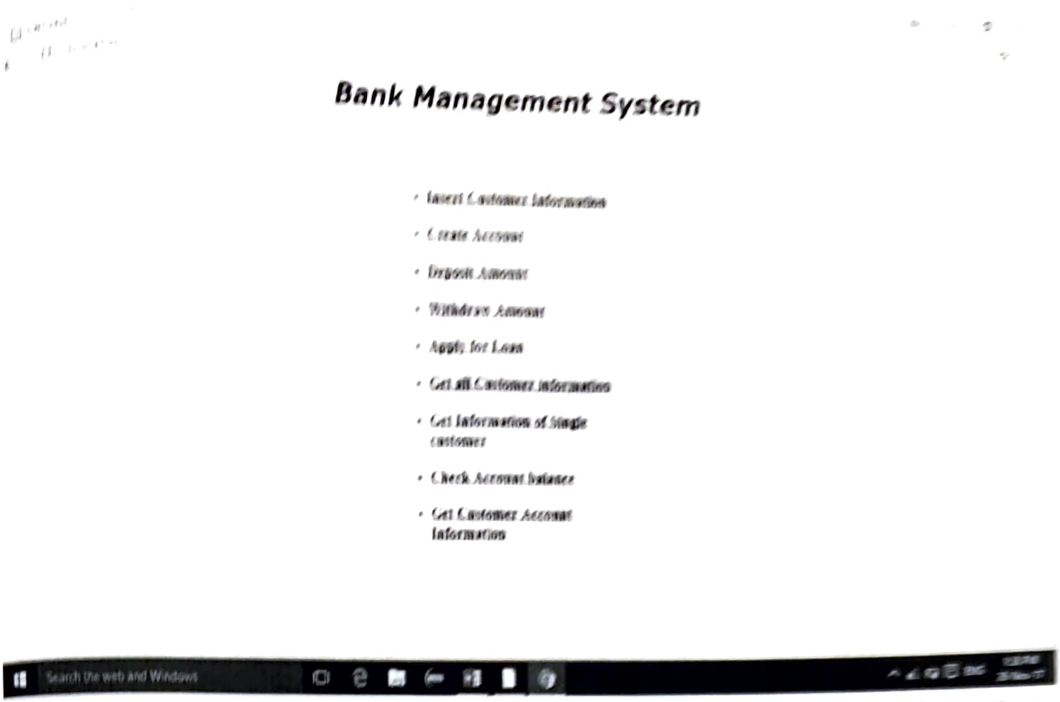
Snapshots



LOGIN FORM

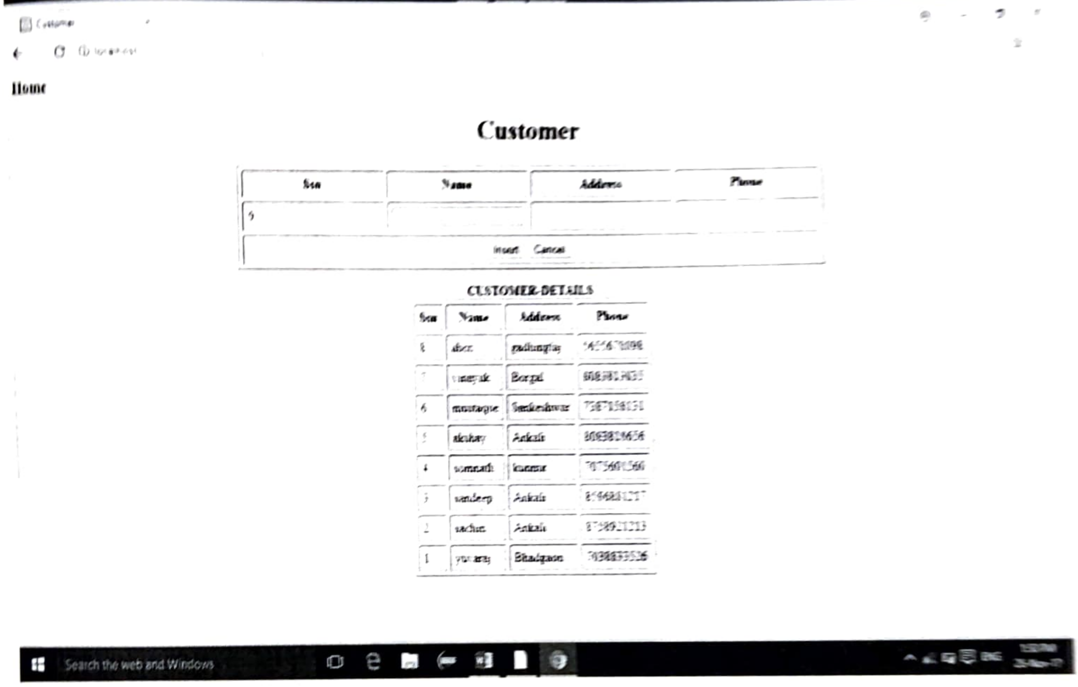
USER NAME: [illegible]

PASSWORD: [illegible]



Bank Management System

- Insert Customer Information
- Create Account
- Deposit Amount
- Withdraw Amount
- Apply for Loan
- Get all Customer information
- Get Information of Single customer
- Check Account balance
- Get Customer Account Information

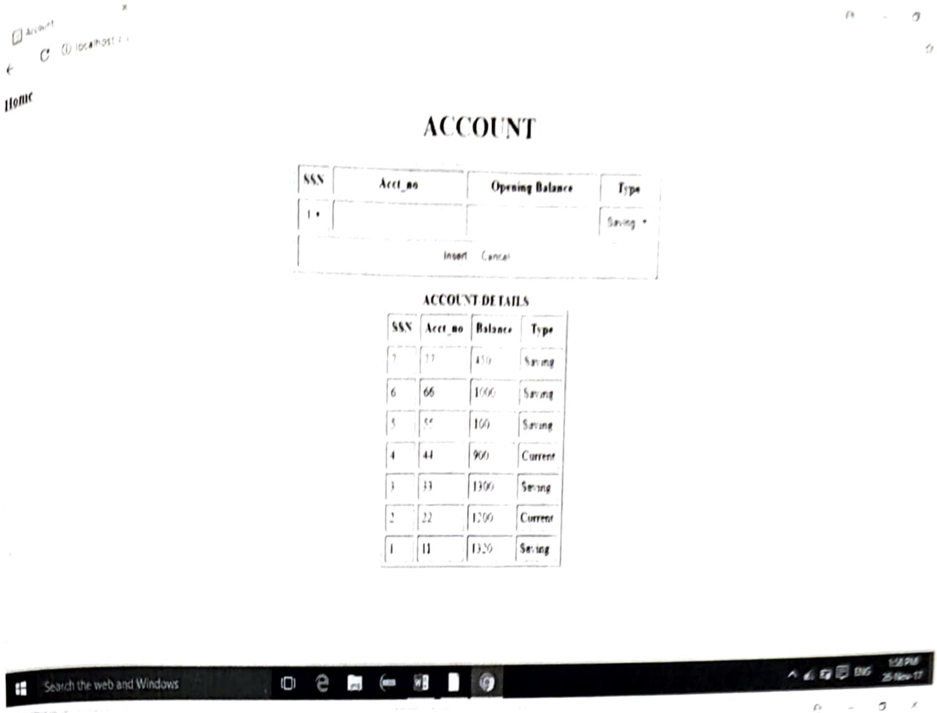


Customer

Ssn	Name	Address	Phone
9			

CUSTOMER-DETAILS

Ssn	Name	Address	Phone
8	abcz	gadhinglaj	[illegible]
7	vinayak	Borgal	[illegible]
6	mustaque	Sankeshwar	7387158131
5	akshay	Ankali	8083810656
4	somnath	kurnur	7075601566
3	sandeep	Ankali	[illegible]
2	sachin	Ankali	8798921213
1	yuvraj	Bhadgaon	[illegible]



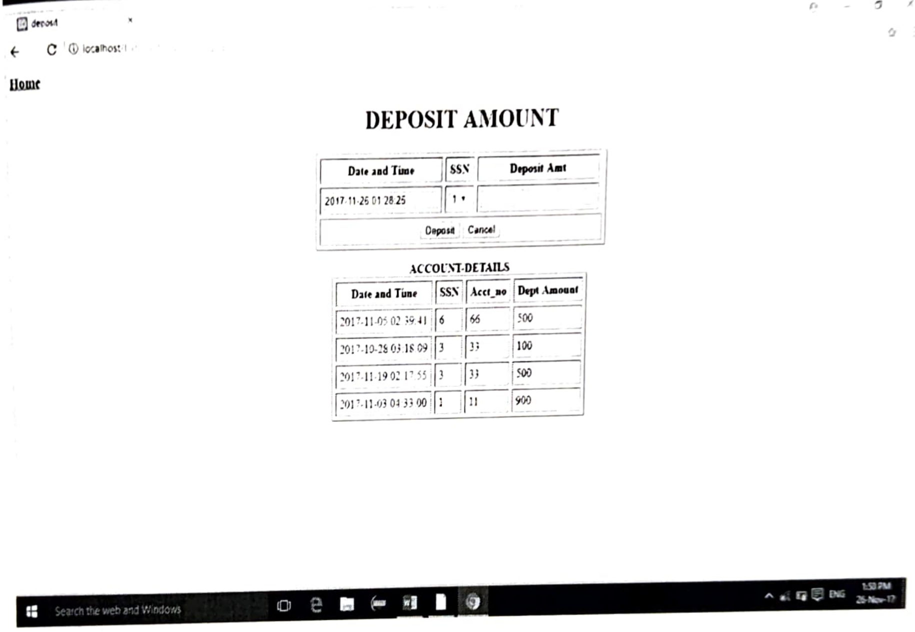
ACCOUNT

SSN	Acct_no	Opening Balance	Type
1 ▾			Saving ▾

Insert Cancel

ACCOUNT DETAILS

SSN	Acct_no	Balance	Type
7	77	450	Saving
6	66	1000	Saving
5	55	100	Saving
4	44	900	Current
3	33	1300	Saving
2	22	1200	Current
1	11	1320	Saving



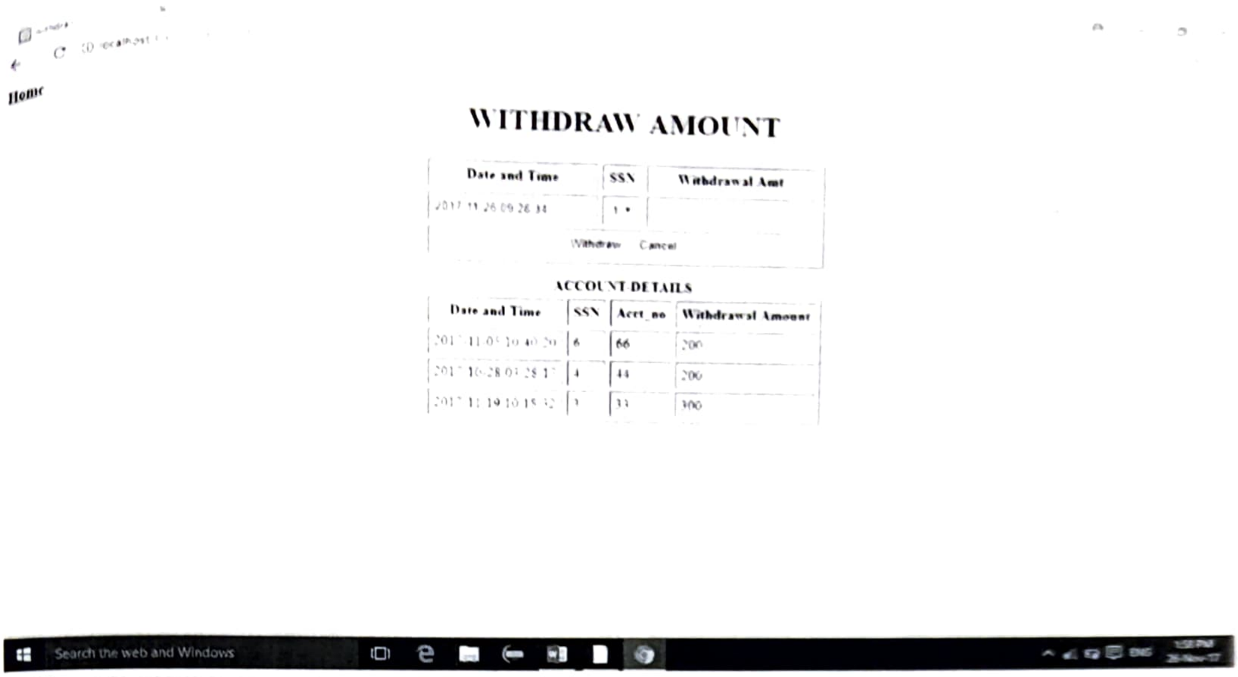
DEPOSIT AMOUNT

Date and Time	SSN	Deposit Amt
2017-11-26 01 28:25	1 ▾	

Deposit Cancel

ACCOUNT-DETAILS

Date and Time	SSN	Acct_no	Dept Amount
2017-11-05 02 39:41	6	66	500
2017-10-28 03:18 09	3	33	100
2017-11-19 02 17:55	3	33	500
2017-11-03 04 33 00	1	11	900

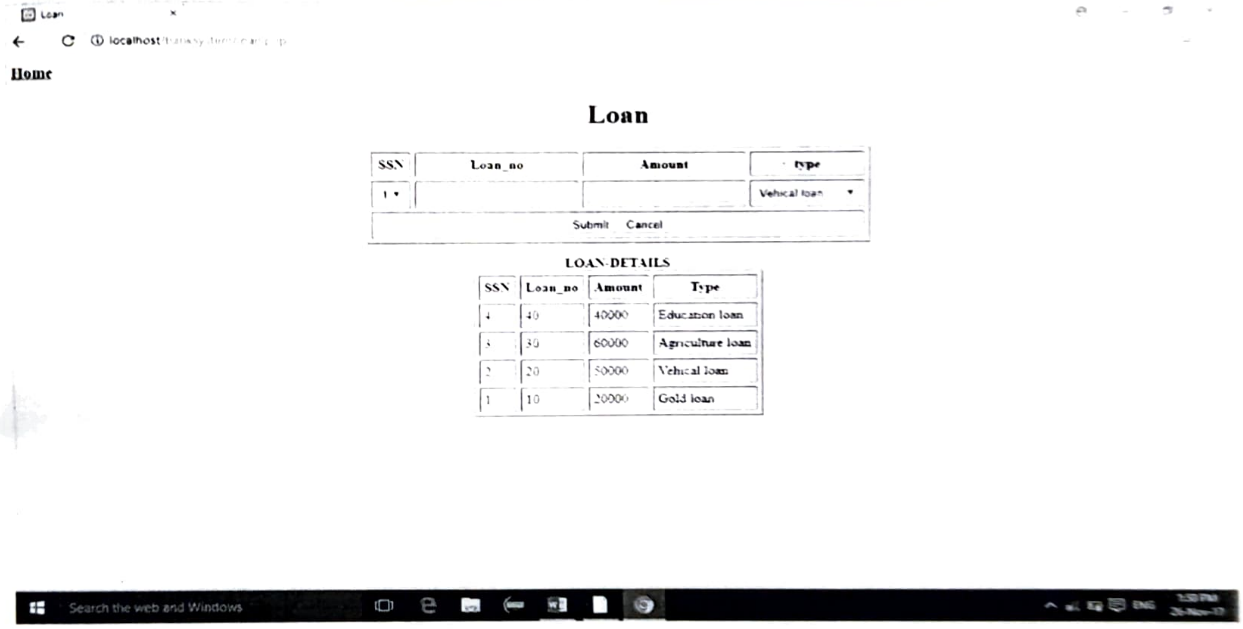


WITHDRAW AMOUNT

Date and Time	SSN	Withdrawal Amt
2017-11-26 09:26:34	1 ▾	
Withdraw Cancel		

ACCOUNT-DETAILS

Date and Time	SSN	Acct_no	Withdrawal Amount
2017-11-05 10:40:20	6	66	200
2017-10-28 03:28:17	4	44	200
2017-11-19 10:15:32	3	33	300

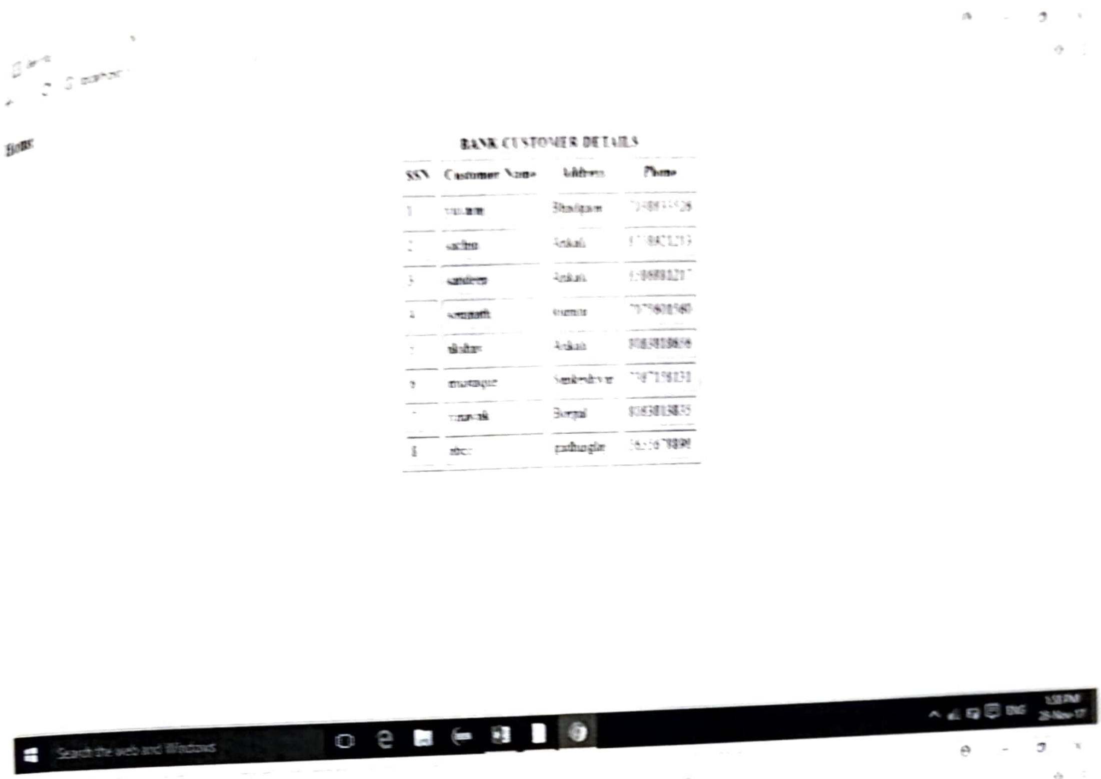


Loan

SSN	Loan_no	Amount	type
1 ▾			Vehical loan ▾
Submit Cancel			

LOAN-DETAILS

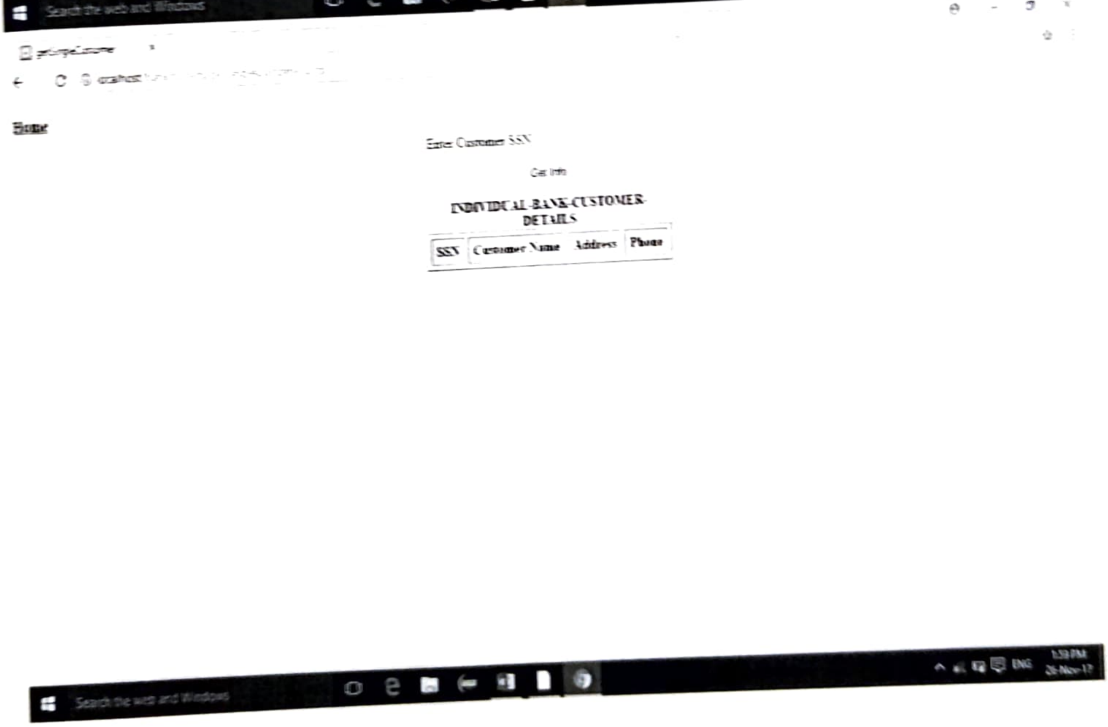
SSN	Loan_no	Amount	Type
4	40	40000	Education loan
3	30	60000	Agriculture loan
2	20	50000	Vehical loan
1	10	20000	Gold loan



Home

BANK CUSTOMER DETAILS

SSN	Customer Name	Address	Phone
1	[illegible]	Bhadgaon	[illegible]
2	[illegible]	Ankali	[illegible]
3	[illegible]	Ankali	[illegible]
4	[illegible]	[illegible]	[illegible]
5	[illegible]	Ankali	[illegible]
6	[illegible]	Sankeshwar	[illegible]
7	[illegible]	Borgal	[illegible]
8	[illegible]	gadhinglaj	[illegible]

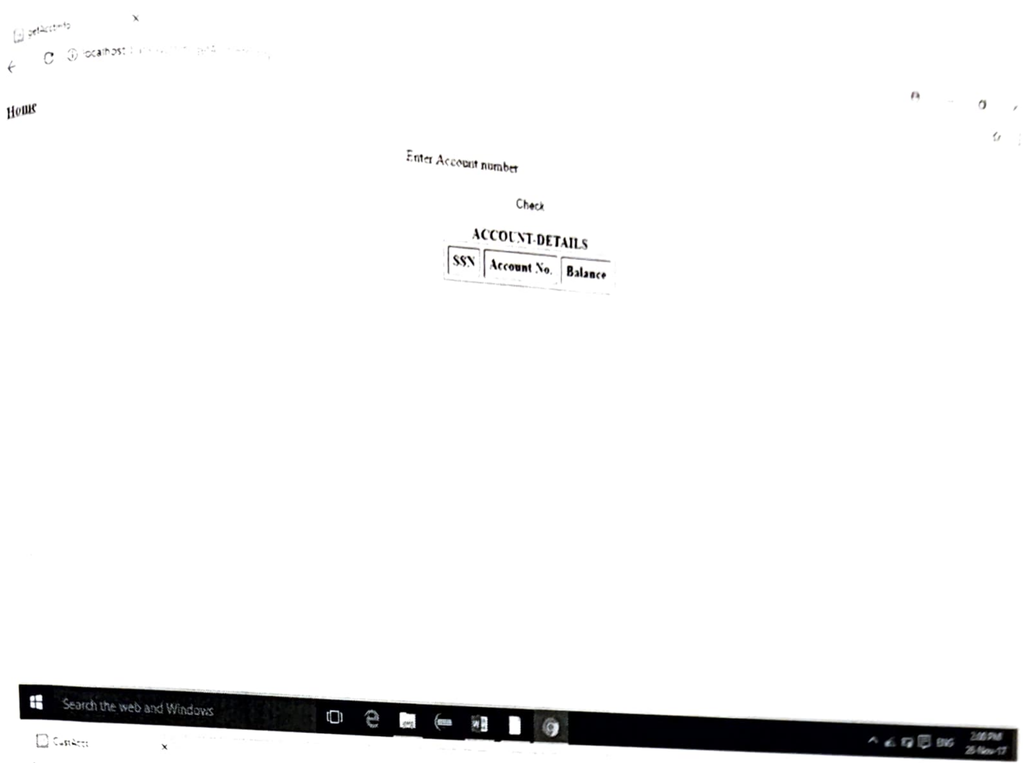


Home

Enter Customer SSN

INDIVIDUAL-BANK-CUSTOMER-DETAILS

SSN	Customer Name	Address	Phone



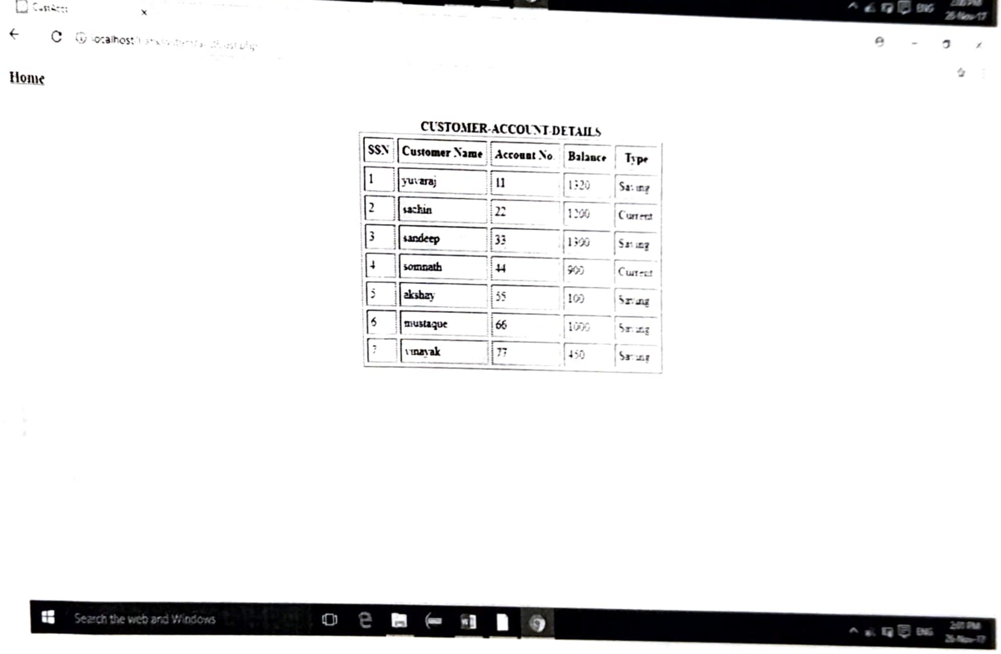
Home

Enter Account number

Check

ACCOUNT-DETAILS

SSN	Account No.	Balance



Home

CUSTOMER-ACCOUNT-DETAILS

SSN	Customer Name	Account No.	Balance	Type
1	yuvaraj	11	1320	Saving
2	sachin	22	1200	Current
3	sandeep	33	1300	Saving
4	somnath	44	900	Current
5	akshay	55	100	Saving
6	mustaque	66	1000	Saving
7	vinayak	77	450	Saving

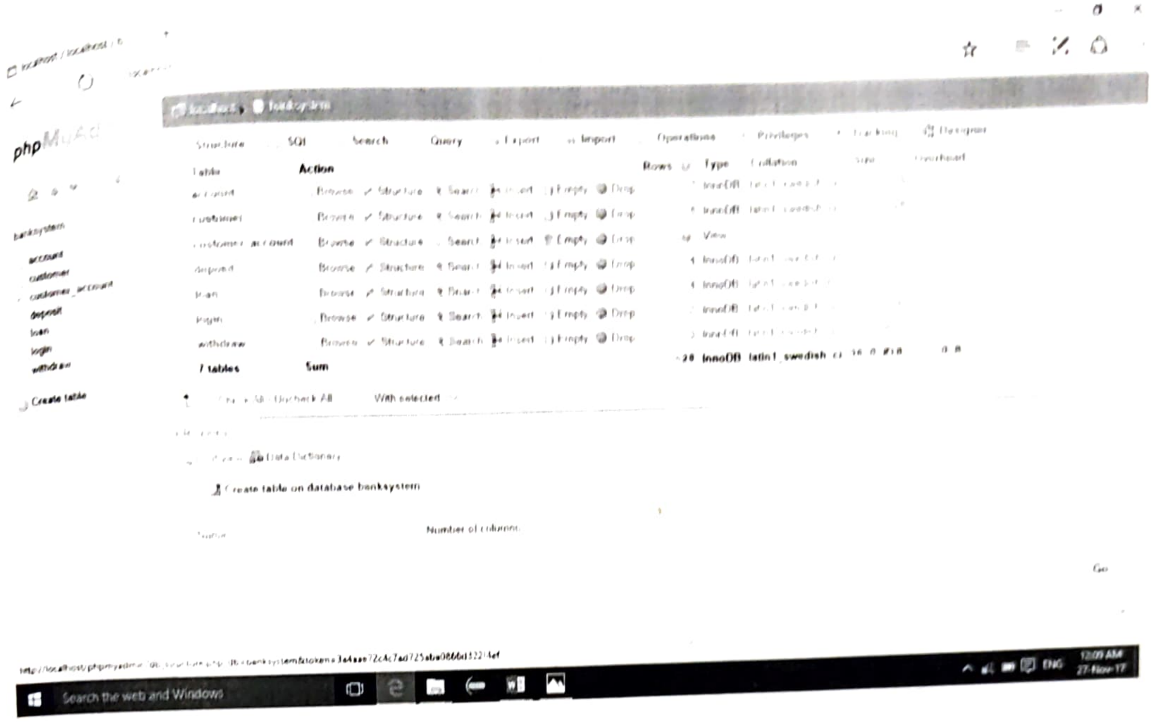


Table	Action	Rows	Type	Collation	Size	Overhead
account	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	[illegible]		
customer	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci		
customer_account	Browse Structure Search Insert Empty Drop	[illegible]	View			
deposit	Browse Structure Search Insert Empty Drop	4	InnoDB	[illegible]		
loan	Browse Structure Search Insert Empty Drop	4	InnoDB	[illegible]		
login	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	[illegible]		
withdraw	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	[illegible]		
7 tables	Sum	20	InnoDB	latin1_swedish_ci	[illegible]	0 B

6.Conclusion

Initially the requirement of this project is to perform all the banking transaction computerized so the complexity and paper work will be reduced. By using this application we will bring the technology to the village it will help in improve the country through village improvement.

The project is mainly based on the idea of developing a banking system which would replace the existing costlier banking system. All the operations that are carried out in the bank manually would be performed automatically and easily by the Banking Solution. The concept of centralized banking is taken into account in this solution. A well interfaced GUI would be used for connecting to the main database server for updating and retrieving the data of the customers.

Computerized accounting many has benefits such as accuracy in issuance of bank statements and fast processing of financial statements as well as easing the highly cumbersome auditing procedure. As electronic commerce is now regarded as the panacea for the survival of any modern day business.. The introduction of this computerized banking system will improved banking activities in a very convenient way. This will as resulted in most banks making huge profits. Data processing and analysis at the bank are faster, accurate and timely which meets management need for decision-making.

7.Technology and Languages Used

7.1 APACHE WEB SERVER:

Apache HTTP Server, colloquially called Apache, is free and open-source crossplatform web server software, released under the terms of Apache License 2.0. ... Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled.

Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation.

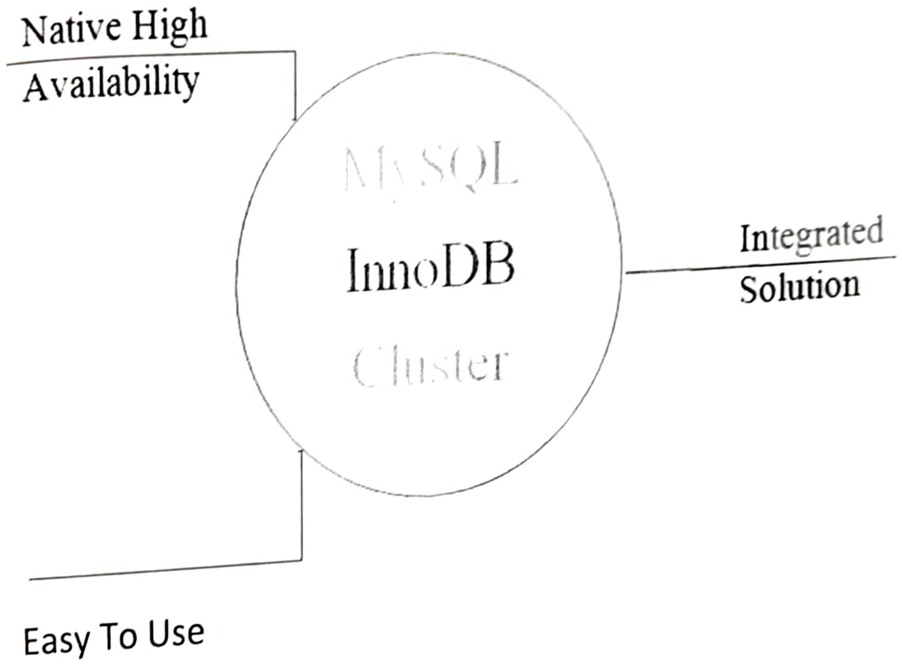
The Apache HTTP Server is cross-platform; as of 1 June 2017 92% of Apache HTTPS Server copies run on Linux distributions. Version 2.0 improved support for non-Unix operating systems such as Windows and OS/2.Old versions of Apache were ported to run on OpenVMS and NetWare.

Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth of the World Wide Web, quickly overtaking NCSA HTTPd as the dominant HTTP server, and has remained most popular since April 1996. In 2009, it became the first web server software to serve more than 100 million websites. As of July 2016 was estimated to serve 46% of all active websites and 43% of the top million websites.

Instead of implementing a single architecture, Apache provides a variety of MultiProcessing Modules (MPMs), which allow Apache to run in a process-based, hybrid (process and thread) or event-hybrid mode, to better match the demands of each particular infrastructure. This implies that the choice of correct MPM and the correct configuration is important. Where compromises in performance need to be made, the design of Apache is to reduce latency and increase throughput, relative to simply handling more requests, thus ensuring consistent and reliable processing of requests within reasonable time-frames.

For delivery of static pages, Apache 2.2 series was considered significantly slower than nginx and varnish. To address this issue, the Apache developers created the Event MPM, which mixes the use of several processes and several threads per process in an asynchronous event-based loop. This architecture, and the way it was implemented in the Apache 2.4 series, provides for performance equivalent or slightly better than event-based web servers, as is cited by Jim Jagielski and other independent sources. However, some independent, but significantly outdated, benchmarks show that it still is half as fast as nginx.

7.2 MySQL:



MySQL is an open-source relational database management system(RDBMS). The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements.

MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a central component of the LAMP open-source web application software stack. LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python".

Applications that use the MySQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MySQL is also used in many high-profile, large-scale websites, including Google, Facebook, Twitter, Flickr, and YouTube.

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home brewed lexical analyzer. MySQL works on many system platforms,including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, macOS, Microsoft, Windows, NetBSD, Novell Netware, OpenBSD, Open Solaris, OS/2 Wrap, QNX, Oracle Solaris, Symbian, Sun OS, SCO OpenServer, SCO UnixWare, Sanos, Tru64. A port of MySQL to OpenVMS also exists.

MySQL has received positive reviews, and reviewers noticed it "performs extremely well in the average case" and that the "developer interfaces are there, and the documentation is very, very good". It has also been tested to be a "fast, stable and true multi-user, multi-threaded sql database server"

MySQL can be built and installed manually from source code, but it is more commonly installed from a binary package unless special customizations are required. On most Linux distributions, the package management system can download and install MySQL with minimal effort, though further configuration is often required to adjust security and optimization settings.

Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well. It is still most commonly used in small to medium scale single-server deployments, either as a component in a LAMPhttps://en.wikipedia.org/wiki/LAMP_(software_bundle)based web application or as a standalone database server. Much of MySQL's appeal originates in its relative simplicity and ease of use, which is enabled by an ecosystem of open source tools such as phpMyAdmin. In the medium range, MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory.

7.3 PHP:

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHPcode in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere.

PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML or HTML5 markup, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a

PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge

The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP, although non-PHP text is still subject to control structuresdescribed in PHP code. The most common delimiters are

<?php to open and ?> to close PHP sections. The shortened form <? also exists. This short delimiter makes script files less portable, since support for them can be disabled in the local PHP configuration and it is therefore discouraged.

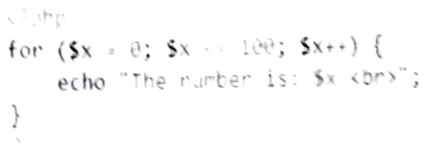
The first form of delimiters, <?php and ?>, in XHTML and other XML documents, creates correctly formed XML processing instructions. This means that the resulting mixture of PHP code and other markup in the server-side file is itself well-formed XML.

Variables are prefixed with a dollar symbol, and a type does not need to be specified in advance. PHP 5 introduced type hinting that allows functions to force their parameters to be objects of a specific class, arrays, interfaces or callback functions. However, before PHP 7.0, type hints could not be used with scalar types such as integer or string.[53]

Unlike function and class names, variable names are case sensitive. Both double-quoted ("") and heredoc strings provide the ability to interpolate a variable's value into the string.[96]PHP treats newlines as whitespace in the manner of a free-form language, and statements are terminated by a semicolon. PHP has three types of comment syntax: /*   */ marks block and inline comments; // as well as # are used for one-line comments. The echo statement is one of several facilities PHP provides to output text, e.g., to a web browser.

In terms of keywords and language syntax, PHP is similar to the C style syntax. if conditions, for and while loops, and function returns are similar in syntax to languages such as C, C++, C#, Java and Perl.

The following is an example of PHP for loop:



```
<?php
for ($x = 0; $x <= 100; $x++) {
    echo "The number is: $x <br>";
}
?>
```

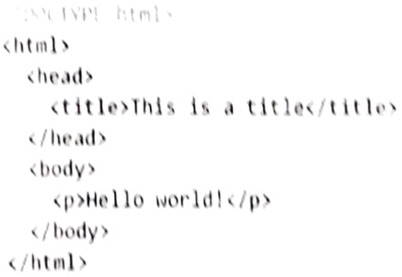
.4 HTML:

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as forms, may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img>and<input >introduce content into the page directly. Others such as<p> ... </p>surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript which affect the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML.

The following is an example of the classic "Hello, World!" program, a common test employed for comparing programming languages, scripting languages and markup languages.



```
<!DOCTYPE html>
<html>
  <head>
    <title>This is a title</title>
  </head>
  <body>
    <p>Hello world!</p>
  </body>
</html>
```

7.5 CASCADING STYLE SHEET:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

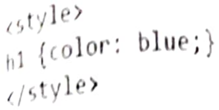
CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice, and on Braille-based tactile devices. It can also display the web page differently depending on the screen size or viewing device. Readers can also specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author specified.

Changes to the graphic design of a document (or hundreds of documents) can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing markup in the documents.

The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities are calculated and assigned to rules, so that the results are predictable.

The following example shows the style element that gives red colour to fonts:



```
<style>
h1 {color: blue;}
</style>
```

8.References

❖ BOOKS REFERRED

- Programming the World Wide Web – Robert W. Sebesta, 4th Edition, Pearson Education, 2008.
- PHP and MySQL Web Development by Luke Welling & Laura Thompson, Sam Publishing, ISBN 0672-32525-X, First print 2003, Pages: 912.
- PHP – Writing PHP scripts & using Variables R. Berdan.
- HTML the definite guide – OREILLY, By Chuck Musciano& Bill Kennedy; ISBN 1-56592-492-4, Third Edition, August 1998, Pages: 576.

❖ WEB LINK

- http://php-guide-lines.blogspot.in/
- http://www.codingforums.com/
- http://www.php.happycodings.com/
- http://www.programmingfacts.com/
- http://911-need-code-help.blogspot.in/
- http://en.wikipedia.org/wiki/
- http://www.dreamincode.net
- http://www.a1vbcode.co