ONLINE BANKING MANAGEMENT SYSTEM

A Project Submitted
In Partial Fulfillment of the Requirements
for the Degree of

MASTER OF COMPUTER APPICATIONS

By

SHASHANK MISHRA
(University Roll No.1900290140032)

Under the Supervision of
Ms. VIDUSHI
(Assistant Professor)
KIET Group of Institution, Delhi-NCR, Ghaziabad

to the

FACULTY OF COMPUTER APPLICATRIONS

DR. APJ ABDUL KALAM TECHNICAL UNIVERSITY LUCKNOW
(Formerly Uttar Pradesh Technical University, Lucknow)

June, 2022

DECLARATION

I hereby declare that the work presented in this report entitled “ONLINE BANKING MANEGMENT SYSTEM", was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute.

I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

I affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name : SHASHANK MISHRA
Roll. No. : 1900290140032
Branch : M.C.A. 6th Semester

(Candidate Signature)

CERTIFICATE

Certified that **Shashank Mishra** (**University Roll No.1900290140032),** have carried out the project work having "Online Banking Management System" for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Ms. Vidushi
Assistant Professor
Dept. of Computer Applications
KIET Group of Institutions, Delhi-NCR, Ghaziabad

External Examiner

Dr. Ajay Kumar Shrivastava
Professor & Head
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Date:

ABSTRACT

The Online Banking Management System is an application for maintaining a person's account in 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 work space 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 Java language. 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 analyzes 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 Java. 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.

ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to **Ms. Vidushi Mam** for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Ajay Kumar Shrivastava, Professor and Head, Department of Computer Applications**, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Shashank Mishra

Roll No. 1900290140032

TABLE OF CONTENTS

	Page No.
Declaration	02
Certificate	03
Abstract	04
Acknowledge	05
Table of content	
Chapter 1:	09-15
▪ Introduction	09-10
▪ Objective	11-15
Chapter 2: Literature Review	16-18
Chapter 3: Feasibility study	19-27
▪ Technical Feasibility & Operational Feasibility	19-20
▪ System requirement specification	21
▪ Features	22
▪ SDLC	23
▪ Introduction to front end Technology	24
▪ Introduction to backend Technology	25-27
Chapter 4: System Designee	28-41
▪ DFD	28-32
▪ Use-Case Diagram	33
▪ E-R DIAGRAM	34-36
▪ Data structures and database specification	37-41
Chapter 5:	42-47
▪ Snapshot	

Chapter 6: 48-77

- Coding

Chapter 7: 78-80

- Testing

Chapter 8:

Conclusion &Future Scope 81-82

- Conclusion 81
- Future Scope 82

Chapter 9: 83-84

- References

LIST OF TABLES

- **TABLE 1** SQL DATABASE TABLE……..26
- **TABLE 2** "ACCOUNT_INFO" TABLE. 37-39
- **TABLE 3** "BRANCH_INFO" TABLE. 39
- **TABLE 4** "DEPOSIT_INFO" TABLE…….39
- **TABLE 5** "FIXED_INFO" TABLE.....…..40
- **TABLE 6** "LOGIN_INFO" TABLE 40
- **TABLE 7** "WITHDRAWL_INFO" TABLE. 41
- **TABLE 8** "LOAN_INFO" TABLE. 41

LIST OF FIGURES

- **FIGURE 1** SOFTWARE DEVELOPMENT LIFE CYCLE……...23
- **FIGURE 2** SYTEM DATA FLOW DIAGRAM……29-32
- **FIGURE 3** USE CASE DIAGRAM……33
- **FIGURE 4** E-R DIAGRAM……....34

CHAPTER 1

INTRODUCTION

The “Online Banking 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 “Online Banking 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.

OBJECTIVE OF THE PROJECT

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 hard-hitting 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.

The main aim of designing and developing this Internet banking System Java 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.

Main Goal

Motto- 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.

1. **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.

2. **Saving Customer Time**: Client doesn't need to go to the bank to do small operation.

3. **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.

4. **Transferring Money:** Help client transferring money to/or another bank or country.

Methods

- We need to be able to generate an account number
- Account types: Savings or Current Account
- Maintain/update Balance
- Open/Close Account
- Withdraw/Deposit

Administrative Modules

Here in my project, there are two types of modules. This module is the main module which performs all the main operations in the system. The major operations in the system are:

Admin Module

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.

1- Admin login
2- Add/delete/update account

3- Withdrawal/deposit/statements transaction
4- Account Information
5- User details list
6- Active/Inactive account
7- View transaction histories

User Module

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.

1- User login, use PIN system
2- Creating/open new account registration
3- Funds transfer (local/international/domestic)
4- View statements transaction
5- User account details
6- Change Password and Pin

Banks terms:

1. All requests received from customers are logged for backend fulfillment and are effective from the time they are recorded at the branch.
2. Rules and regulations applicable to normal banking transactions in India will be applicable mutatis mutandis for the transactions executed through this site.
3. The BAMS Bank service cannot be claimed as a right. The bank may also convert this into a discretionary service anytime.
4. Dispute between the customer and the Bank in this service is subject to the jurisdiction of the courts in the Republic of India and governed by the laws prevailing in India.

The Bank reserves the right to modify the services offered or the Terms of service of BAMS Bank. The changes will be notified to the customers through a notification on the Site.

Customer's obligations

1. The customer has an obligation to maintain secrecy in regard to Username & Password registered with the Bank. The bank presupposes that login using valid Username and Password is a valid session initiated by none other than the customer.
2. Transaction executed through a valid session will be construed by RR to have emanated from the registered customer and will be binding on him/her.
3. The customer will not attempt or permit others to attempt accessing the BAMS Bank through any unlawful means.

Benefits of online banking

Many of us lead busy lives. Some of us are up before the crack of dawn, getting ourselves prepared so we can in turn get our families ready for the day. We rush to work, rush to get the kids to school, and at the end of the day we rush home only to brace ourselves for the next day. After a hectic day, the last thing you want to do is spend time waiting in line at the bank, or even the post office. That's where Online Banking comes in. Many of the benefits of doing our banking online are obvious:

1. You don't have to wait in line.
2. You don't have to plan your day around the bank's hours.
3. You can look at your balance whenever you want, not just when you get a statement.

There are some hidden benefits too. As a young bank customer, you're just learning how to manage your money and observe your spending patterns.
Online banking allows you to watch your money on a daily basis if you want to. By keeping close tabs on your funds, you'll always be aware of what's happening in your bank account.
For those experienced spenders, this option is far more appealing than the sudden discovery that you're broke!

It's also helpful to watch how much interest you're gathering on investments and savings or what service charges you have incurred.

Most available benefits

1. Online banking with key bank is fast, secure, convenient and free.
2. Quick, simple, authenticated access to accounts via the web application.
3. Simply scalable to grow with changing system requirement.
4. Global enterprise-wide access to information.
5. Improved data security, restricting unauthorized access.
6. Minimize Storage Space

CHAPTER 2

LITERATURE REVIEW

Dr. Geeta Sharma mentioned concerning role and edges concerning net banking in Indian banking sector. conjointly mentioned the services obtained through net banking area unit statements, on-line fund transfer, on-line payment services, online requests and intimations and maintaining demat account.

Anju Dagar mentioned concerning importance, advantages, numerous on-line services and issues pertaining in on-line banking. Ebubeogu Amarachukwu Felix developed package for banking management system victimization ASP.NET. This project performs the subsequent operations, gap associate degree account, deposits, withdraws, fund transfers and change the details. Mahmood sovereign mentioned concerning what's meant by E-banking, importance, overview, technologies, human involvement in banking sector, issues, strategic development and future trends in E- banking. Muhammad Abdus Sattar Titu and Md. Azizur Rahman explained adoption, major elements, major on-line banking services, client satisfaction and major problems encountered in on-line banking services in Bangladesh banks. Bahman Saeidipour et.al, analyzed the factors to adopt net banking. D.Amutha conducted a study from ninety respondents in Tuticorin district to understand the notice and satisfaction connected parameters in E-banking.

Internet banking provide convenience to bank customers, permitting them to use services from banks in distance and avoid hassles to travel to the bank branches as well as it generates substantial price

Savings to banks (Sullivan & Wang, 2014). Financial establishment in African nation cannot ignore technological info systems since they play a vital role in their operations [15]

Alternatively, existing banking companies produce virtual banks as separately capitalized subsidiary banks of a bank company (Furst et al.,2000). a 3rd route is investors purchase the present charter of a traditional bank, and so to recast the bank as a virtual bank underneath the present charter (Furst et al., 2000).[14]

Many skilled bank staff were offered early retirement and therefore the remaining staff baby-faced inflated workloads with shorter service hours (Ongkasuwan and Tantichattanon, 2002). This modification caused the bulk of the Thai banks to use net banking to reduce waiting time, errors and prices, and ultimately improve customers‟ satisfaction.

Internet revolution is international phenomenon and going by the present growth statistics, Bharat expects a spurt in the Internet penetration in coming back years particularly within the electronic commerce. It is a visible notion that electronic.

(Internet) banking and payments area unit likely to advance a

lot of or less in wheel with ecommerce. Researches indicate that net banking encompasses a vital impact on the business models of banks, securities commerce corporations, brokerage houses, insurance firms etc. Internet banking has conjointly attracted the 16 attentions of, regulators and lawmakers in the developing nations since the late 1990s. Internet banking may be a reason behind concern to majority of the offline banks WHO ought to be prepared for associate degree unprecedented competition from the non- traditional banking establishments that offer banking and money services over the web (Rajgopalan, 2001). though some of the standard banks have started offering their services on line, it is only associate degree extension of their offline services (Devi, 2001). Internet banking has currently started motivating customers to park their funds with the online banks, that encompasses a substantially impact on the deposit base of the brick-and-mortar banks. The use of technology in banking has direct relationship with the profit. Cetris paribus, investment in electronic banking increase the margin of profit of banks by reducing prices and increase in non-interest financial gain, which will increase the ROA and ROE (Sinkney, 1998). Cost-effectiveness in delivery of services directly implies relatively high client satisfaction and a subsequent modification in the revenue model for the banks. Adoption of the web mode of banking would end in inflated consumer awareness, attracts the entry of worldwide majors within the market and would result in the emergence of open standards within the banking industry (Treasury Management,2001). the combination of the banking services with e-commerce and emergence of e-cash would completely affect the potency lots of theanks (Scott, 1999) However, net banking is a mixed blessing within the variety of increased risk, the amount of confidence reposed by the shoppers and the problem of mixing it with the physical system (Hawke, 2001). Internet banking has brought about a new orientation to risks like Settlement risk, international technology transfer risk, crime or fraud risk, restrictive turning away risk, taxationturning away risk, and competition risk (Saunders, 1997). hypothesis. In India, some banks like HDFC and ICICI have introduced payment gateways running on secure systems having firewalls against hacking (Rao, 2001). Convenience, safety and value effectiveness area unit the jargons within the spectrum of on-line banking (Rose, 1998). Researchers on numerous occasions have raised several problems, which's should be addressed in context of net banking in Bharat. First, the availability of technology and infrastructure to support the new model of banking Second, the need for net banking itself – net Banking or associate degree economical system of instantaneous banking or convenient banking. Third, associate degree adequate mechanism to tackle the safety risk and operational risk aspects (Sharma, 2001). Fourth, a proper legal framework to require care of the rights and obligation of the consumers whereas most of those issues are somewhat addressed, a vital issue still remains – what existing and potential consumers feel concerning net banking and on the premise of this however 17 an applicable banking model will be developed in Indian context. There is

a need to live and analyze the consumer perception towards net banking, to search out what's wrong with traditional banks and supply a framework for the banks to strategically adopt the web therefore as to maximize worth for the shoppers.

CHAPTER 3

FEASIBILITY STUDY

In Software Engineering is a study to evaluate feasibility of proposed project or system. Feasibility study is one of stage among important four stages of Software Project Management Process. As name suggests feasibility study is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyse whether software product will be right in terms of development, implantation, contribution of project to the organization etc.

Types of Feasibility Study:

The feasibility study mainly concentrates on below five mentioned areas. Among these Economic Feasibility Study is most important part of the feasibility analysis and Legal Feasibility Study is less considered feasibility analysis.

1. Technical Feasibility –

In Technical Feasibility current resources both hardware software along with required technology are analysed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.

2. Operational Feasibility –

In Operational Feasibility degree of providing service to requirements is analysed along with how much easy product will be to operate and maintenance after deployment. Along with these other operational scopes are determining usability of product, determining suggested solution by

software development team is acceptable or not etc.

3. Economic Feasibility –

In Economic Feasibility study cost and benefit of the project is analysed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analysed whether project will be beneficial in terms of finance for organization or not.

SYSYTEM REQUIREMENTS

Hardware specifications:

Hardware is a set of physical components, which performs the functions of applying appropriate, predefined instructions. In other words, one can say that electronic and mechanical parts of computer constitute hardware.

This package is designed on a powerful programming language Visual Basic. It is a powerful Graphical User Interface. The backend is ACCESS, which is used to maintain database. It can run on almost all the popular microcomputers. The following are the minimum hardware specifications to run this package: -

A Machine with:-

- 4 GB RAM
- 1.2 GHz processor
- Intel i3/i5

Software Requirements:

The software is a set of procedures of coded information or a program which when fed into the computer hardware, enables the computer to perform the various tasks. Software is like a current inside the wire, which cannot be seen but its effect can be felt.

1. Operating System: - Windows 7/8/8.1/10…
2. Application Software: - Application software uses front end Visual Studio , Xampp and database access etc.

FEATURES

Public

- Login Page
- Announcement Page
- About us Page

Client-Side

- Dashboard Page *(display the account number and current balance)*
- List of Transactions History
- Deposit
- Withdraw
- Fund Transfer
- Manage System Credentials

Admin Side

- Dashboard Page
- List of All Transactions History
- Deposit for Client
- Withdraw for Client
- Fund Transfer for Client
- Manage System Credentials
- Manage System Settings/Info

SOFTWARE DEVELOPMENT LIFE CYCLE

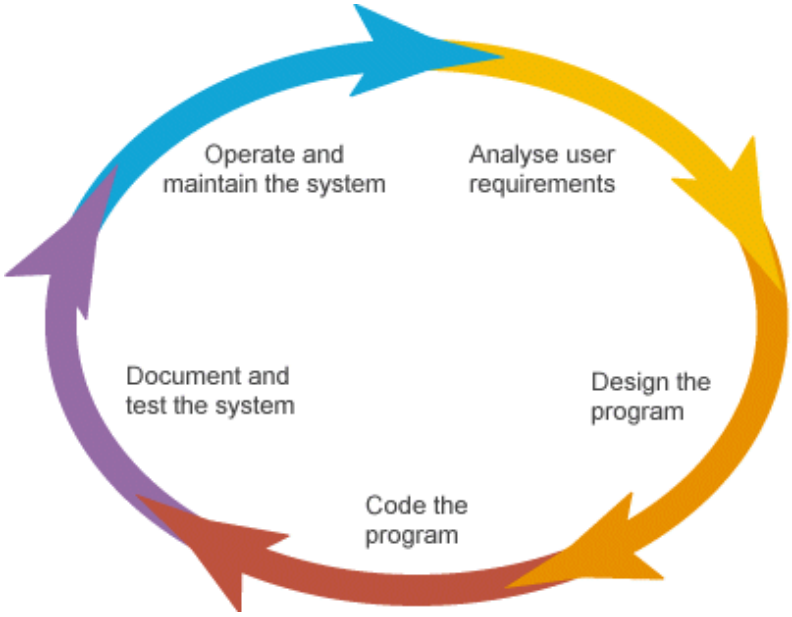
A system development life cycle is a logical process by which system analysts, software engineers, programmers, and end users build information systems and computer applications to solve business problems and needs.

The major phases involved in the MIS development process are referred to as system development life cycle. Each phase of the development process must have well defined objectives, and at the end of each phase, progress towards meeting the objectives must be evaluated.

The development process should not continue until the objectives of all prior phases have been met.

System development life cycle is a phased approach to analysis and design to ensure that systems are best developed.

The system development life cycle can be divided into seven phases as shown in fig



INTRODUCTION TO FRONT END TECHNOLOGY/TOOL

HTML:

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

CSS:

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files

JavaScript:

JavaScript, often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled and multi-paradigm. It has dynamic typing, prototype-based object-orientation and first-class functions.

PHP:

- PHP is an acronym for "PHP: Hypertext Pre-processor"
- PHP is a widely-used, open-source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

INTRODUCTION TO BACK-END TECHNOLOGY/TOOL

Introduction to SQL: -

SQL is a standard computer language for accessing and manipulating databases.

- SQL stands for **S**tructured **Q**uery **L**anguage.
- SQL allows you to access a database.
- SQL is an ANSI standard computer language.
- SQL can execute queries against a database.
- SQL can retrieve data from a database.
- SQL can insert new records in a database.
- SQL can delete records from a database.
- SQL can update records in a database.
- SQL is easy to learn.

- SQL is an ANSI (American National Standards Institute) standard computer language for accessing and manipulating database systems. SQL statements are used to retrieve and update data in a database. SQL works with database programs like MS Access, DB2, Informix, MS SQL Server, Oracle, Sybase, etc.

- Unfortunately, there are many different versions of the SQL language, but to be in compliance with the ANSI standard; they must support the same major keywords in a similar manner (such as SELECT, UPDATE, DELETE, INSERT, WHERE, and others).

- **SQL Database Tables: -**

A database most often contains one or more tables. Each table is identified by a name (e.g. "Customers" or "Orders"). Tables contain records (rows) with data.

Below is an example of a table called "Persons": -

Last Name	First Name	Address	City
Hansen	Ola	Timoteivn 10	Sandnes
Svendson	Tove	Borgvn 23	Sandnes
Pettersen	Kari	Storgt 20	Stavanger

- The table above contains three records (one for each person) and four columns (Last Name, First Name, Address, and City).

- **SQL Queries: -**

With SQL, we can query a database and have a result set returned.

- **A query like this: -**

```
SELECT Last Name FROM Persons
```

- **Gives a result set like this: -**

Last Name
Hansen
Svendson
Petersen

SQL Data Manipulation Language (DML)

- SQL (Structured Query Language) is syntax for executing queries. But the SQL language also includes syntax to update, insert, and delete records.
- These query and update commands together form the **Data Manipulation Language (DML) part of SQL: -**

  - **SELECT** - extracts data from a database table
  - **UPDATE** - updates data in a database table
  - **DELETE** - deletes data from a database table
  - **INSERT INTO** - inserts new data into a database table

SQL Data Definition Language (DDL)

- The Data Definition Language (DDL) part of SQL permits database tables to be created or deleted. We can also define indexes (keys), specify links between tables, and impose constraints between database tables.

- **The most important DDL statements in SQL are: -**

  - **CREATE TABLE** - creates a new database table
  - **ALTER TABLE** - alters (changes) a database table
  - **DROP TABLE** - deletes a database table
  - **CREATE INDEX** - creates an index (search key)
  - **DROP INDEX** - deletes an index MS SQL SERVER 2000

CHAPTER 4

DATA FLOW DIAGRAM

- **DATA FLOW DIAGRAM: -**

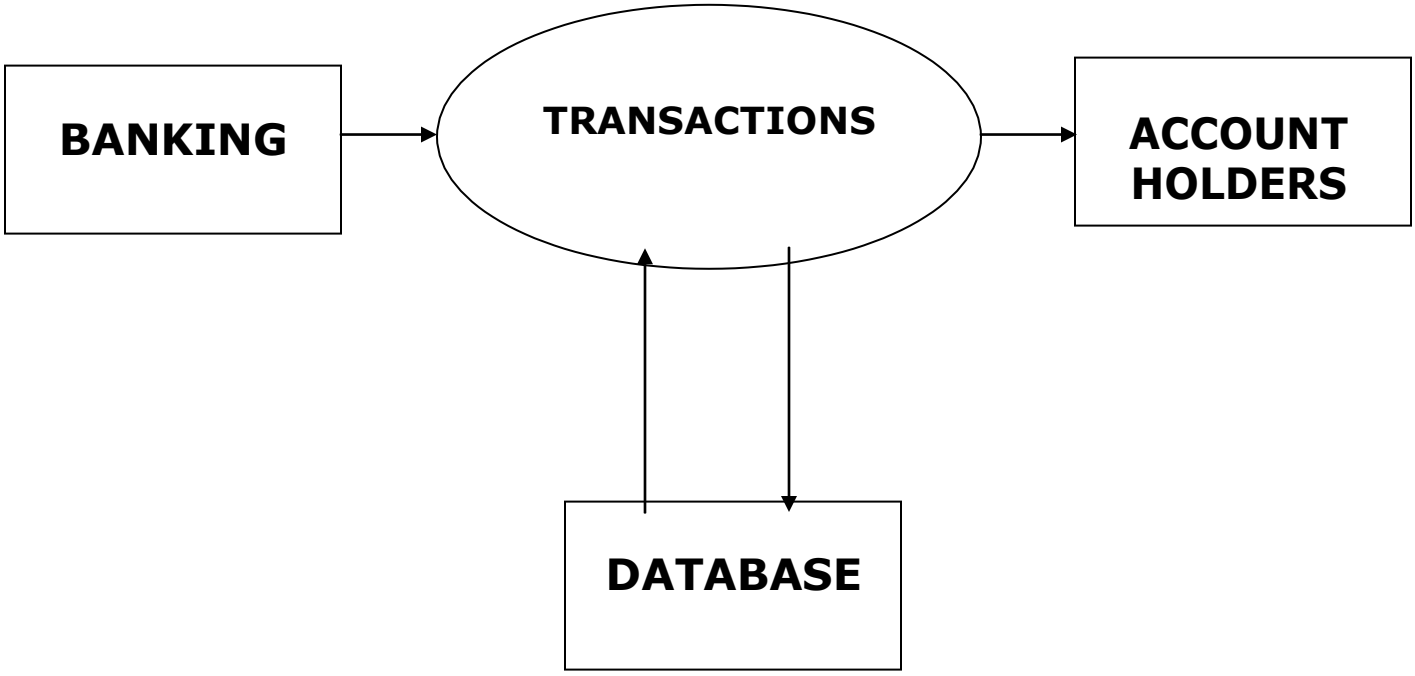
The data flow diagram is also known as “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design so it is the starting point of specification down to the lowest level of detail. A DFDs consists of a series if bubbles joined by lines. The bubbles represent data transformation and the lines represent the data flow in the system.

- **DFD SYMBOLS:**

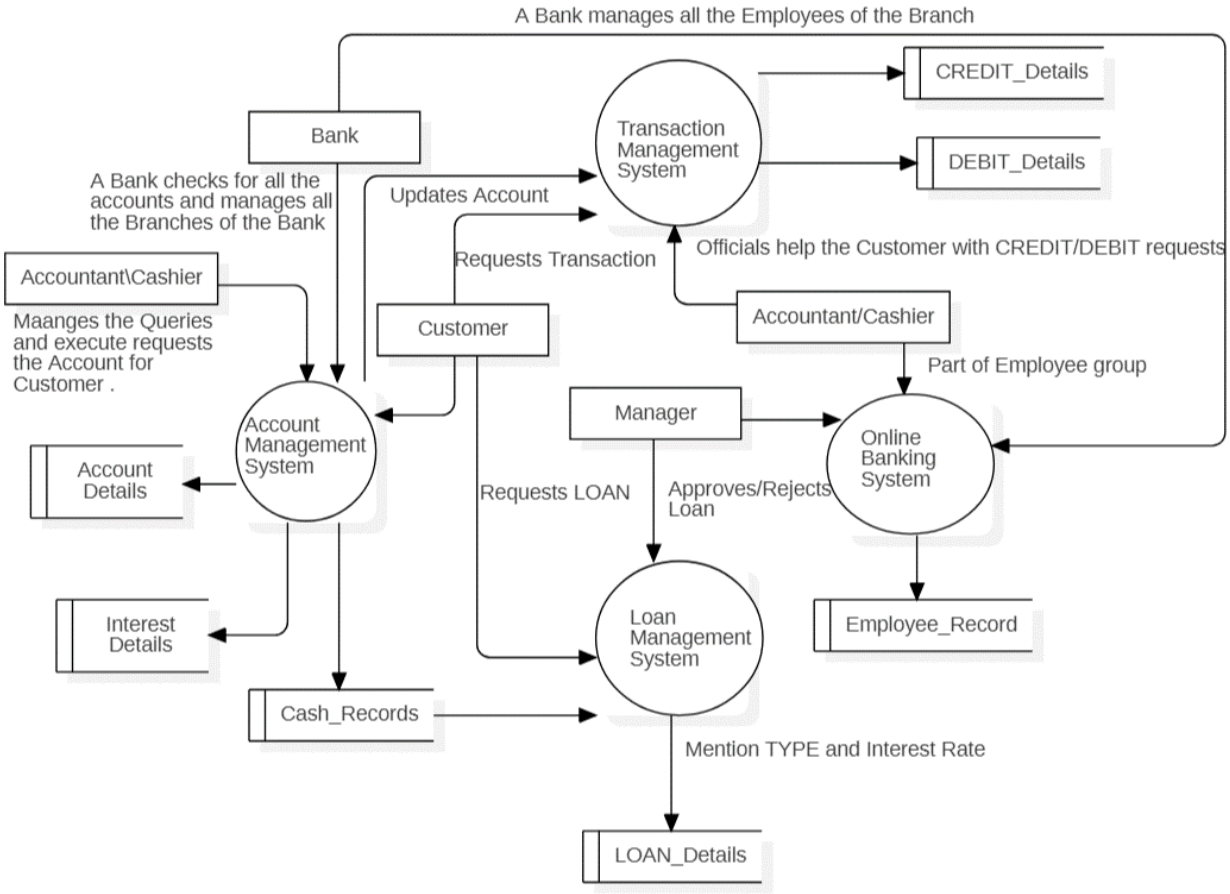
  - A system defined a source or destination of data.
  - An arrow identifies data flow, data in motion.
  - A circle represents the process that transforms incoming data flow to outgoing data flow.
  - An open rectangular is data store-data at rest or a temporary repository of data.

SYSTEM DATA FLOW DIAGRAM

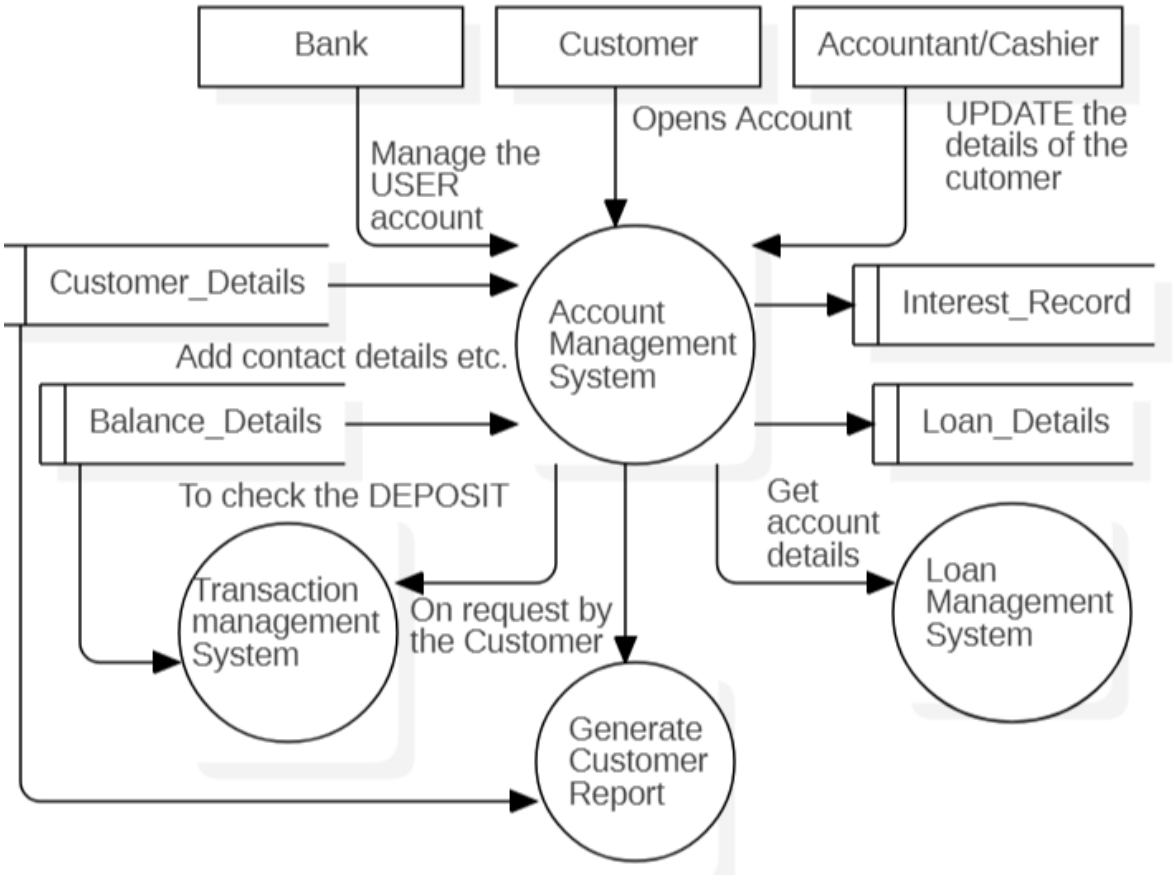
DFD 0 level:-



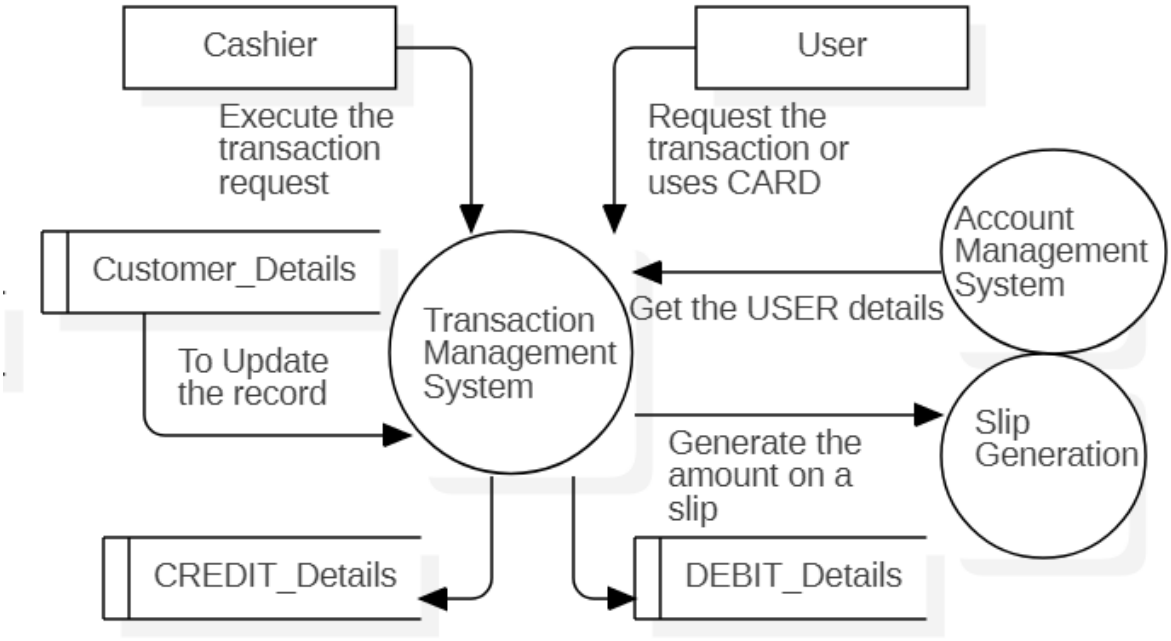
Level-1 DFD:


A Bank manages all the Employees of the Branch
Bank
CREDIT_Details
Transaction Management System
DEBIT_Details
A Bank checks for all the accounts and manages all the Branches of the Bank
Updates Account
Requests Transaction
Officials help the Customer with CREDIT/DEBIT requests
Accountant\Cashier
Maanges the Queries and execute requests the Account for Customer .
Customer
Accountant/Cashier
Part of Employee group
Manager
Account Management System
Online Banking System
Account Details
Requests LOAN
Approves/Rejects Loan
Interest Details
Loan Management System
Employee_Record
Cash_Records
Mention TYPE and Interest Rate
LOAN_Details

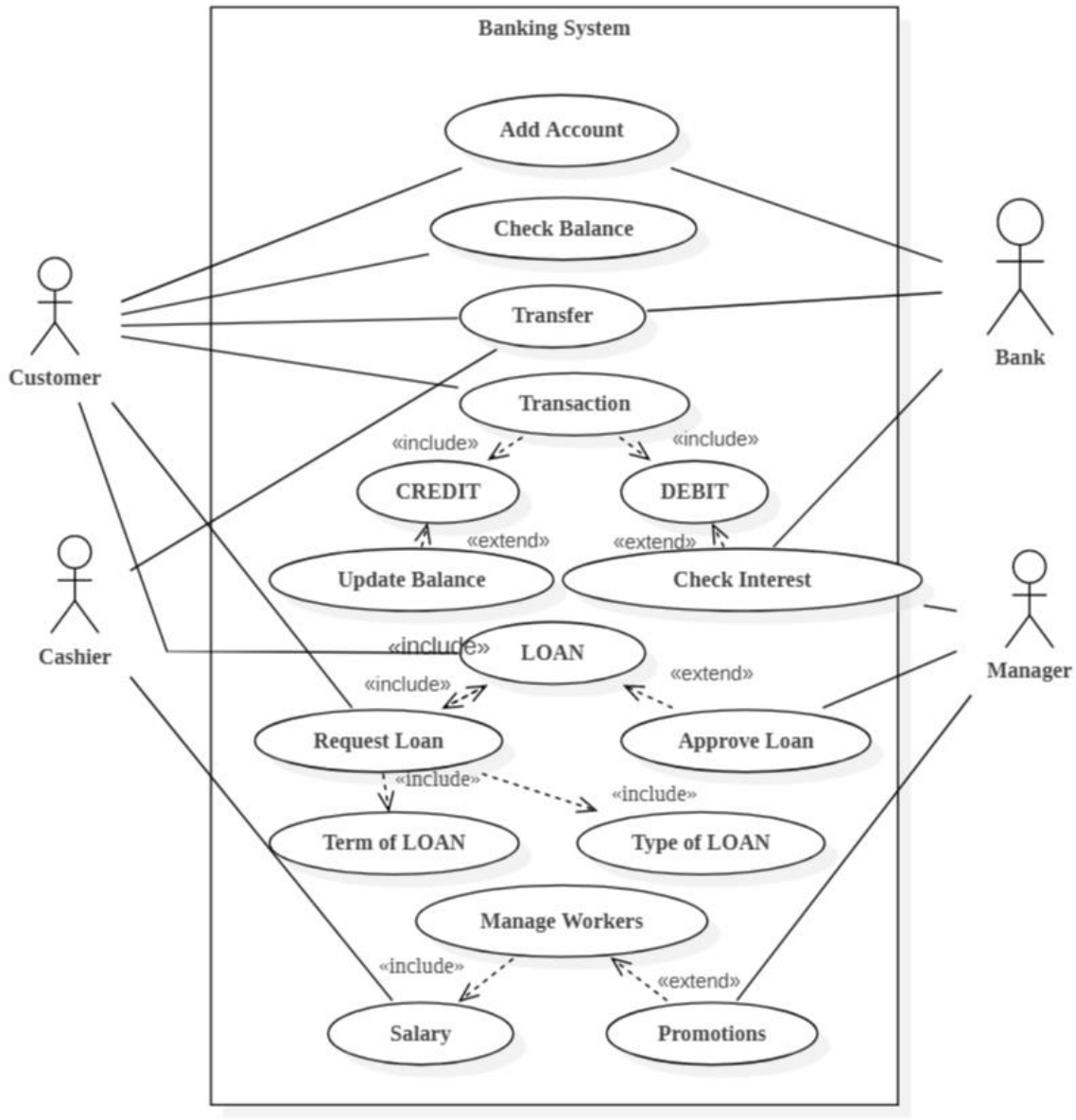
Level-2 DFD (ACCOUNT MAGEMENT SYSTEM):



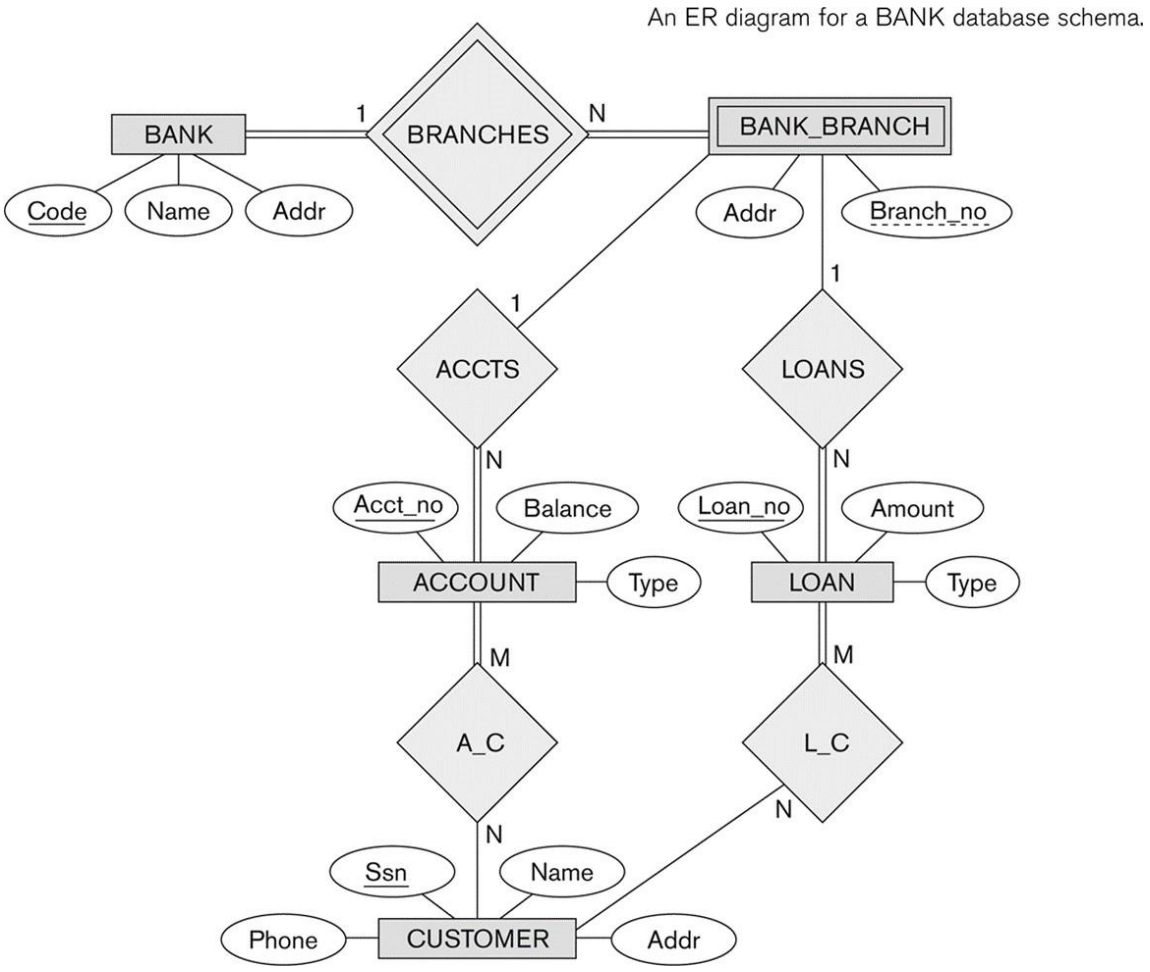
Level-2 DFD (TRANSITION MANGEMENT SYSTEM):



USE-CASE DIGRAM:



E-R DIAGRAM



ER-modeling is a data modeling technique used in software engineering to produce a conceptual data model of a information system. Diagrams created using this ER- modeling technique are called Entity-Relationship Diagrams, or ER diagrams or ERDs.

So you can say that Entity Relationship Diagrams illustrate the logical structure of databases.

Dr. Peter Chen is the originator of the Entity-Relationship Model. His original paper about ER-modeling is one of the most cited papers in the computer software field.
Currently the ER model serves as the foundation of many system analysis and design methodologies, computer-aided software engineering (CASE) tools, and repository systems.

The original notation for ER-Diagrams uses rectangles to represent entities, and diamonds to represent relationships.

There are three basic elements in ER-Diagrams:

- Entities are the "things" for which we want to store information. An entity is a person, place, thing or event.
- Attributes are the data we want to collect for an entity.
- Relationships describe the relations between the entities.

ERDs show entities in a database and relationships between tables within that database. It is essential to have ER-Diagrams if you want to create a good database design. The diagrams help focus on how the database actually works.

Entity (Instance)

An instance of a physical object in the real world. Entity Class
: Group of objects of the same type.

E.g. Entity Class “Student”, Entities “John”, “Trish” etc

Attributes

Properties of Entities that describe their characteristics. Types:

Simple

: Attribute that is not divisible, e.g. age. Composite

: Attribute composed of several simple attributes,

e.g. address (house number, street, district)

Multiple

: Attribute with a set of possible values for the same entity,

e.g. Phone (home, mobile etc.) or email

Key

: Uniquely Ids the Entity e.g. PPSN, Chassis No.

Each simple attribute associated with a VS that may be assigned to that attribute for each individual entity,

e.g. age = integer

DATA STRUCTURES AND DATABASE SPECIFICATIONS

"ACCOUNT_INFO" Table: -

Field Name	Type	Constraints
Account_No	Int	Primary Key
Branch_No	Varchar(7)	References Branch_Info(Branch_No)
Branch_Name	VARCHAR(50)	Not Null
Account_H_Type	Varchar(15)	Not Null
No_Account_H	Varchar(5)	Not Null
Saluation_F	Varchar(5)	Not Null
Name_P_O_F	Varchar(50)	Not Null
Fa_Name_F	Varchar(50)	Not Null
Gender_F	Varchar(6)	Not Null
DOB_F	DateTime	Not Null
Age_F	Varchar(3)	Check(Age_F>=0 and Age_F<100)
Occupation_F	Varchar(15)	Not Null
Photo_F	Image	Not Null
Sign_F	Image	Not Null
Address_F	Varchar(100)	Not Null
Ph_No_F	Varchar(11)	Not Null
Mob_No_F	Varchar(14)	Not Null
Saluation_S	Varchar(5)	
Name_P_O_S	Varchar(50)	
Fa_Name_S	Varchar(50)	
Gender_S	Varchar(6)	

DOB_S	DateTime	
Age_S	Varchar(3)	Check(Age_S>=0 and Age_S<100)
Occupation_S	Varchar(15)	
Photo_S	Image	
Sign_S	Image	
Address_S	Varchar(100)	
Ph_No_S	Varchar(11)	
Mob_No_S	Varchar(14)	
Saluation_T	Varchar(5)	
Name_P_O_T	Varchar(50)	
Fa_Name_T	Varchar(50)	
Gender_T	Varchar(6)	
DOB_T	DateTime	
Age_T	Varchar(3)	Check(Age_T>=0 and Age_T<100)
Occupation_T	Varchar(15)	
Photo_T	Image	
Sign_T	Image	
Address_T	Varchar(100)	
Ph_No_T	Varchar(11)	
Mob_No_T	Varchar(14)	
Account_Type	Varchar(25)	Not Null
Witness_Name	Varchar(50)	Not Null
Witness_Sign	Image	Not Null
Nominee_Rel	Varchar(10)	Not Null

Nominee_Name	Varchar(50)	Not Null
Nominee_Sign	Image	Not Null
Opening_Bal	Varchar(10)	
T_Date	DateTime	Not Null

"Branch_Info" Table: -

Field Name	Type	Constraints
Branch_No	Varchar(7)	Primary Key
Branch_Name	Varchar(50)	

"Deposit_Info" Table: -

Field Name	Type	Constraints
Account_No	Int	References Account_Info(Account_No)
Branch_No	Varchar(7)	
Depositor_Name	Varchar(150)	
Account_H_Type	Varchar(15)	
Deposit_Amt	Varchar(10)	
Deposit_Date	DateTime	

"Fixed_Info" Table: -

Field Name	Data type	Constraints

Account_No	Int	References Account_Info(Account_No)
Branch_No	Varchar(7)	
Depositor_Name	Varchar(50)	
Account_H_Type	Varchar(15)	
Time_Span	Varchar(5)	References Rate_Of_Interest_Info(Time_Span)
ROI	Varchar(5)	
Start_Date	DateTime	
Mature_Date	Varchar(15)	
Deposit_Amt	Varchar(10)	
Mature_Amt	Varchar(10)	

"Login_Info" Table: -

Field Name	Data type	Description
UserName	Varchar(20)	Primary Key
UserPassWord	Varchar(15)	

Field Name	Data type	Description
Time_Span	Varchar(5)	Primary Key
ROI	Varchar(5)	

"Withdrawl_Info" Table: -

Field Name	Type	Constraints
Account_No	Int	References Account_Info(Account_No)
Branch_No	Varchar(7)	
Withdrawee_Name	Varchar(150)	
Account_H_Type	Varchar(15)	
Withdrawl_Amt	Varchar(10)	
Withdrawl_Date	DateTime	

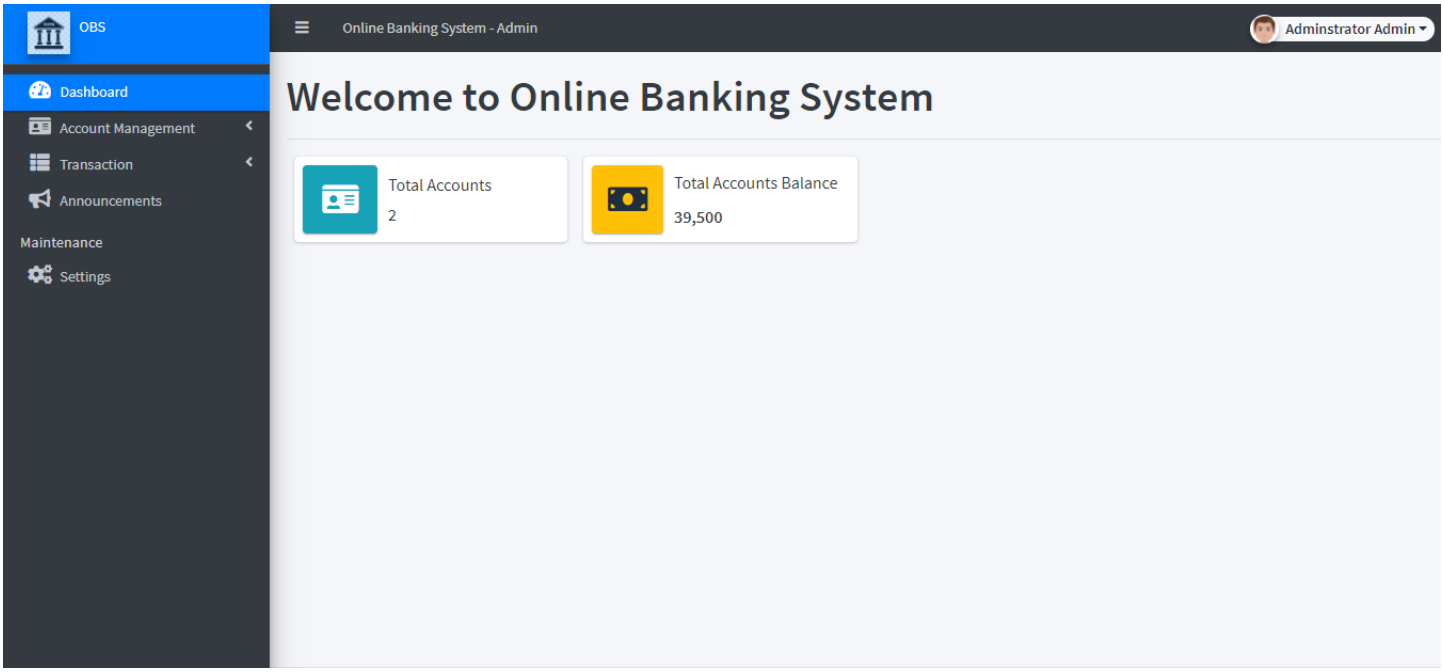
"Loan_Info" Table: -

Field Name	Type	Constraints
Account_No	Int	References Account_Info(Account_No)
Branch_No	Varchar(7)	
Acc_Holder_Name	Varchar(50)	
Account_Type	Varchar(15)	
Account_Sub_Type	Varchar(15)	
Time_Span	Varchar(5)	References Rate_Of_Interest_Info(Time_Span)
ROI	Varchar(5)	
Issue_Date	DateTime	
Due_Date	Varchar(15)	
Loan_Sanctioned	Varchar(10)	
No_Installments	Varchar(5)	
EMI	Varchar(10)	
Total Loan Rate	Varchar(10)	

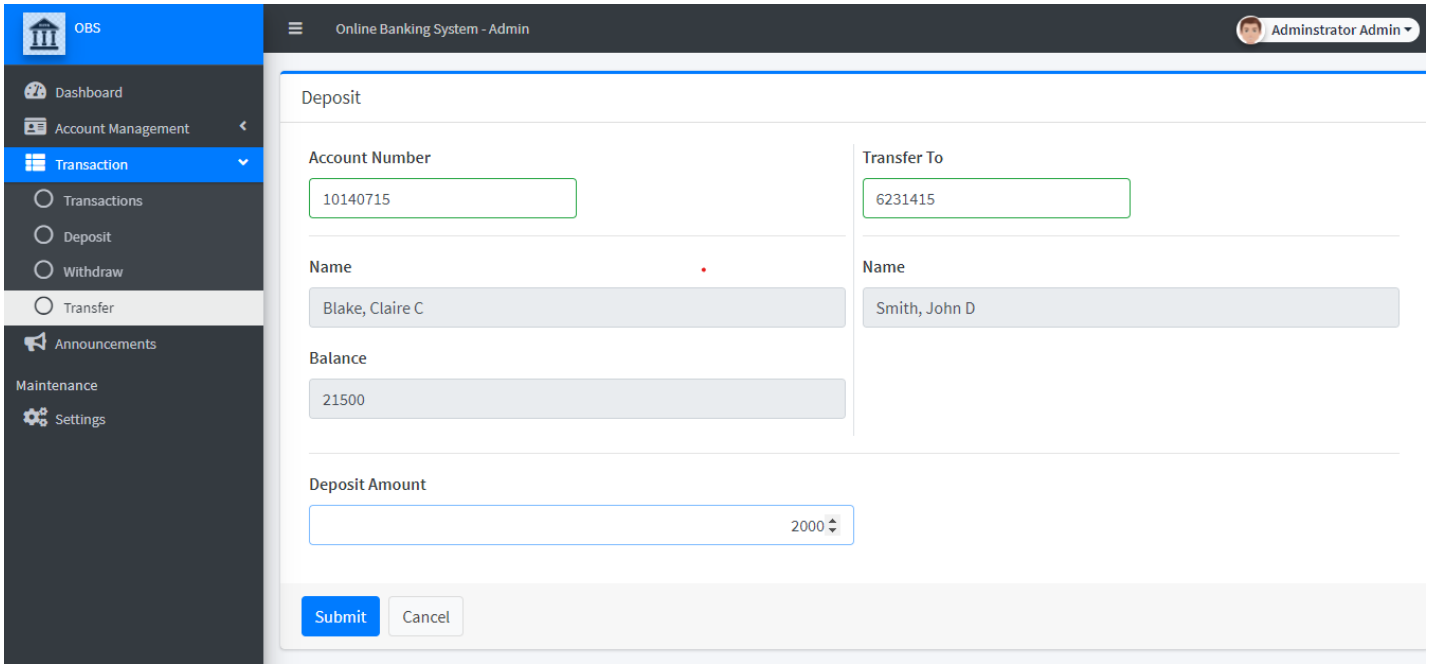
CHAPTER 5

SNAPSHOTS/DESIGN

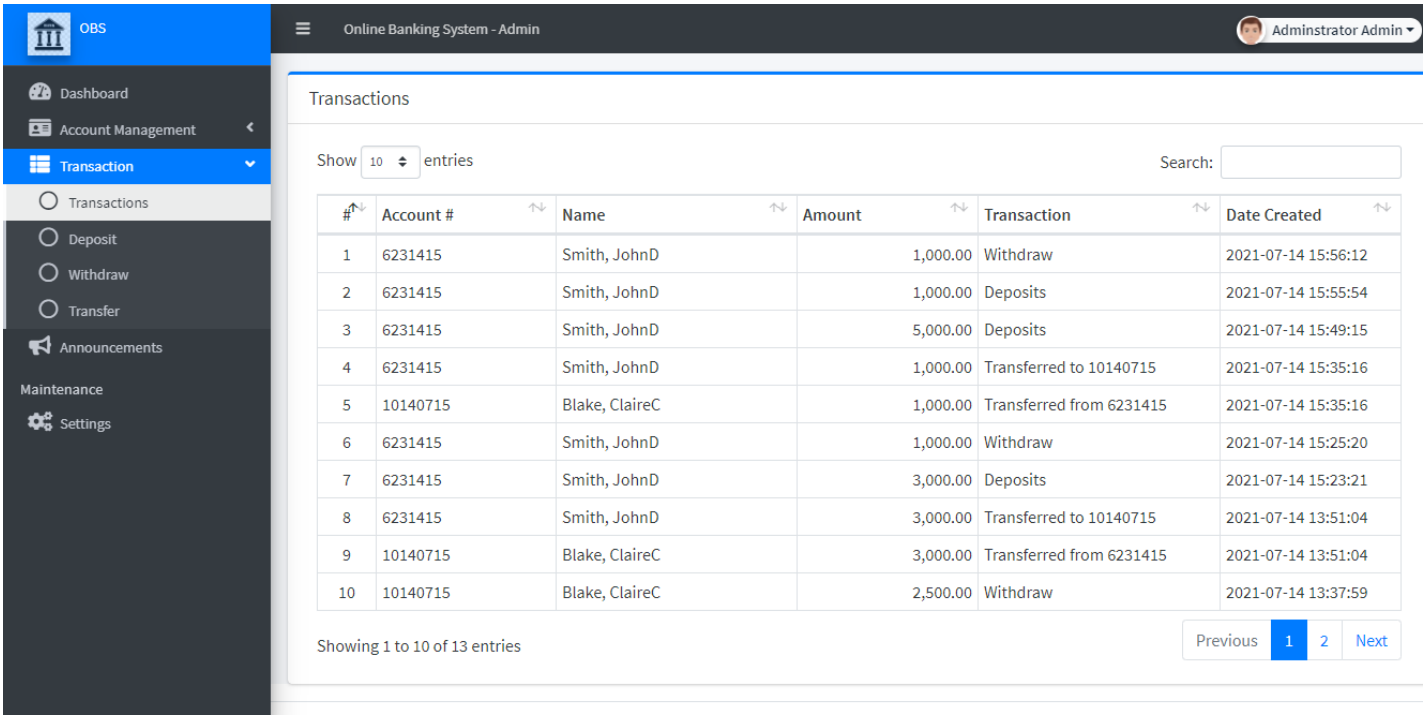
ADMIN HOME PAGE:



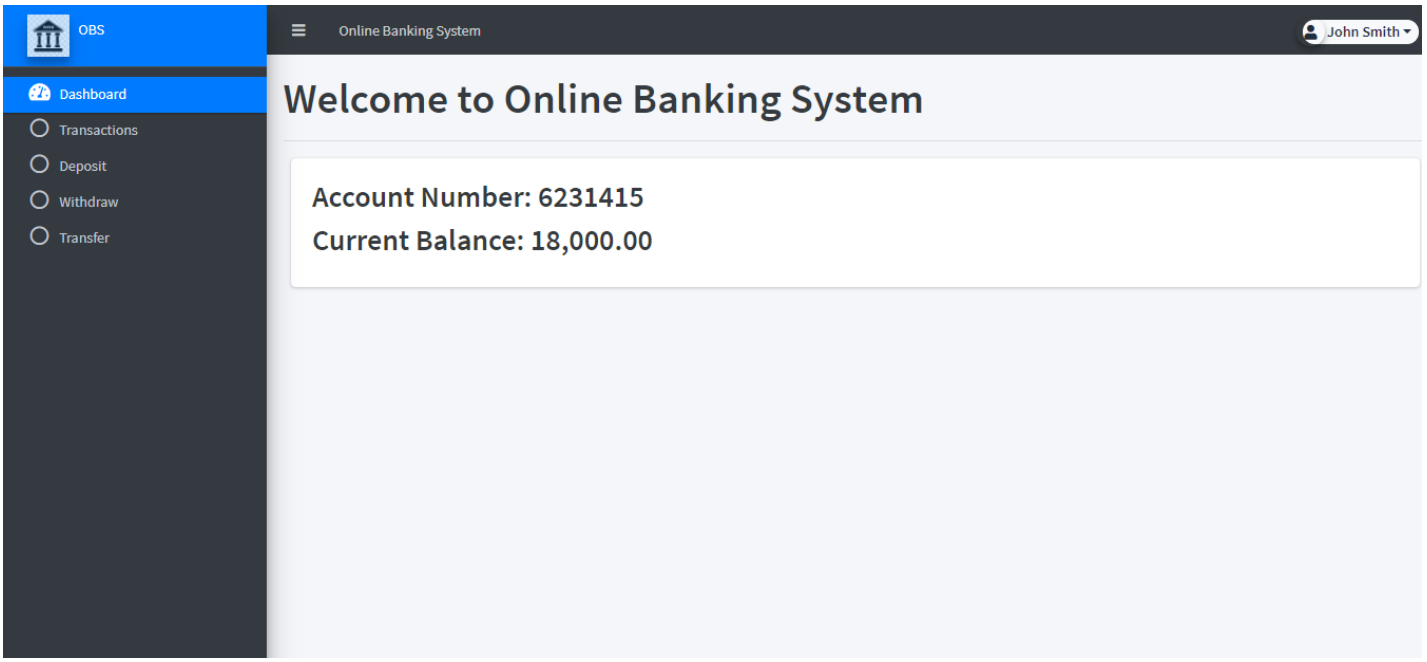
TRANSFER FUND PAGE:



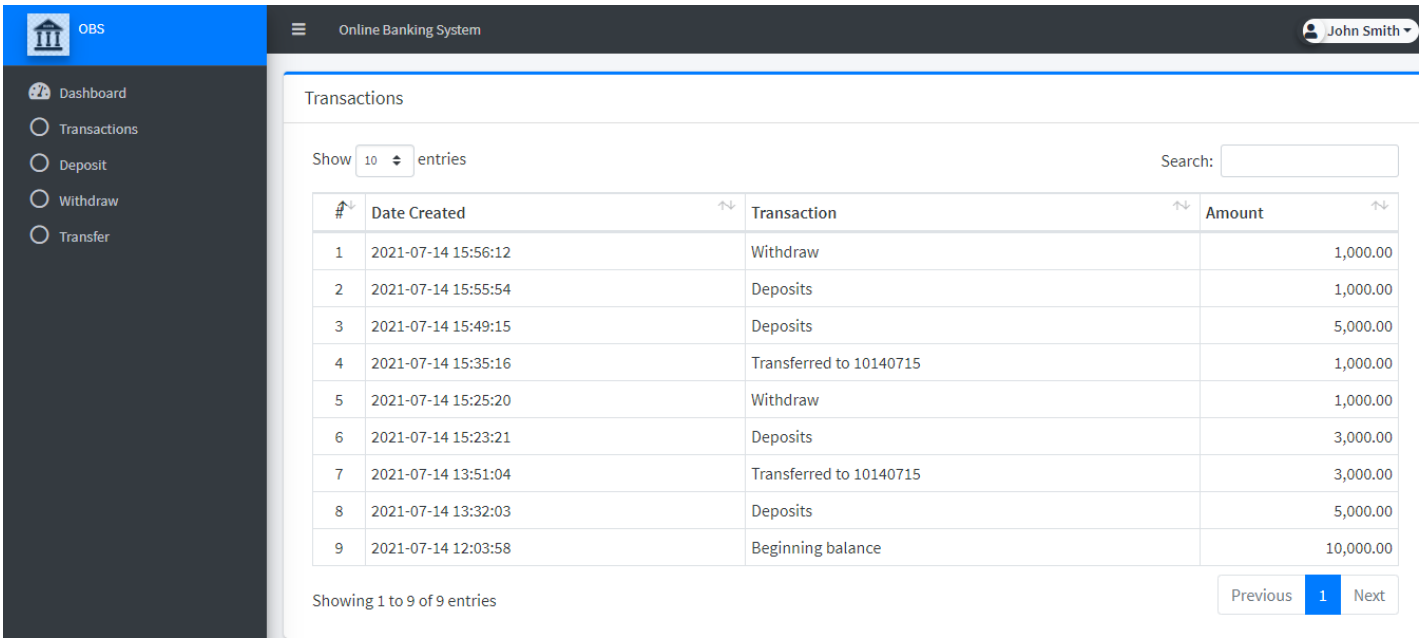
ADMIN TRANSTION LIST PAGE:



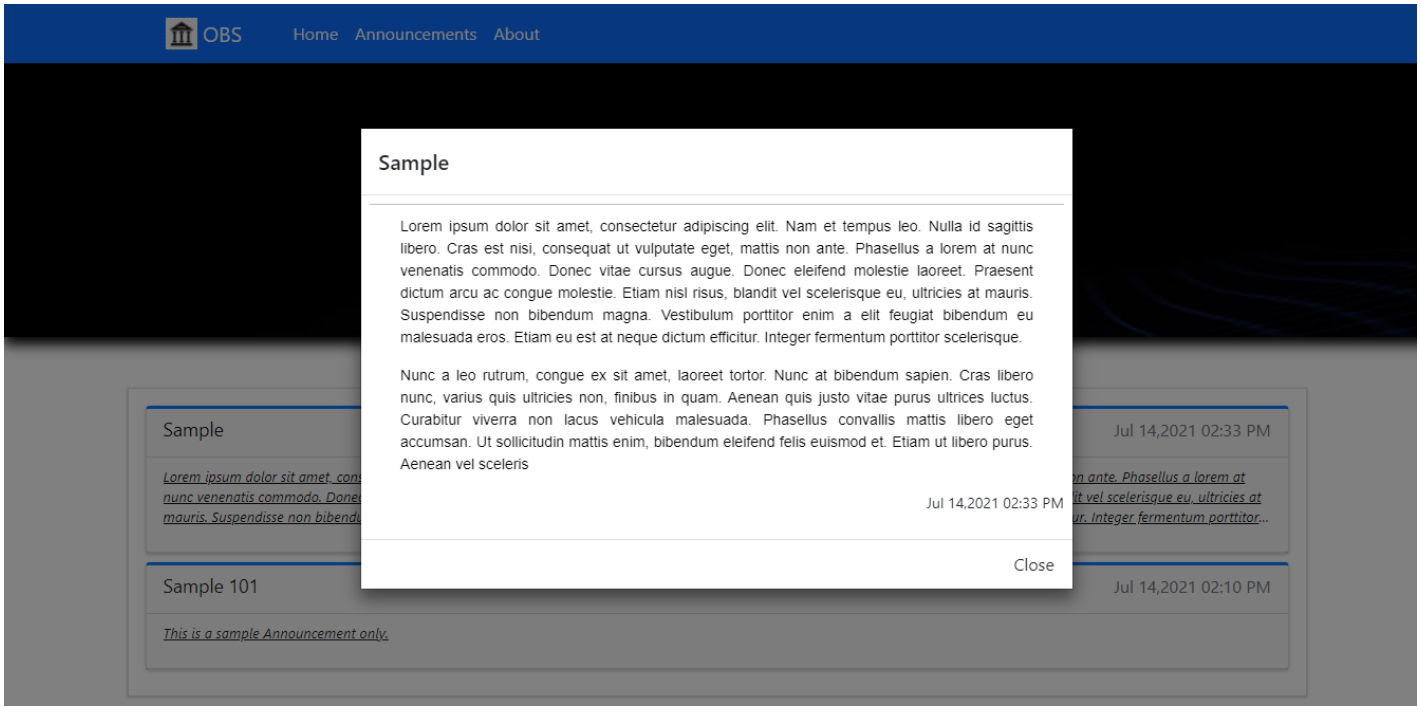
CLIENT DASHBOARD:



CLIENT TRANSICTION LIST:



PUBLIC ANNOUNCMENT PAGE:



CHAPTER 6

CODING

Index.php:

```
<?php if($_settings->chk_flashdata('success')): ?>
<script>
alert_toast("<?php echo $_settings->flashdata('success') ?>",'success')
</script>
<?php endif;?>
<style>
#selectAll{
top:0
}
</style>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title">List of Individual</h3>
<!-- <div class="card-tools">
<a href="?page=individual/manage_individual" class="btn btn-flat btn-primary"><span class="fas fa-plus"></span>  Create New</a>
</div> -->
</div>
<div class="card-body">
<div class="container-fluid">
<div class="row" style="display:none" id="selected_opt">
<div class="w-100 d-flex">
<div class="col-2">
<label for="" class="controllabel"> With Selected:</label>
</div>
<div class="col-3">
<select id="w_selected" class="custom-select select" >
<option value="no_show">Mark as No Show</option>
<option value="delete">Delete</option>
<option value="done">Mark as Done</option>
</select>
</div>
```

```
<div class="col">
<button class="btn btn-primary" type="button" id="selected_go">Go</button>
</div>
</div>
</div>
<table class="table table-bordered table-stripped" id="indi-list">
<colgroup>
<col width="5%">
<col width="5%">
<col width="10%">
<col width="20%">
<col width="15%">
<col width="25%">
<col width="10%">
<col width="10%">
</colgroup>
<thead>
<tr>
<td class="text-center"><div class="form-check">
<input type="checkbox" class="form-check-input" id="selectAll">
</div></td>
<th class="text-center">#</th>
<th>Code</th>
<th>Name</th>
<th>Schedule</th>
<th>Location</th>
<th>Status</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<?php
$i = 1;
$qry = $conn->query("SELECT i.*,s.date_sched,s.status,l.`location` from `individuals` i inner join `schedules` s on i.id = s.individual_id inner join `location` l on l.id = s.location_id order by unix_timestamp(s.date_sched) desc ");
while($row = $qry->fetch_assoc()):
?>

<tr>
<td class="text-center">
```

```
<div class="form-check">
<input type="checkbox" class="form-check-input invCheck" value="<?php echo $row['id'] ?>">
</div>
</td>
<td class="text-center"><?php echo $i++; ?></td>
<td><?php echo $row['code'] ?></td>
<td><?php echo $row['name'] ?></td>
<td><?php echo $row['date_sched'] ?></td>
<td class=""><p class='m-0'><?php echo $row['location'] ?></p></td>
<td class="text-center">
<?php
switch($row['status']){
case(0):
echo '<span class="badge badge-primary">Scheduled</span>';
break;
case(1):
echo '<span class="badge badge-success">Done</span>';
break;
case(2):
echo '<span class="badge badge-danger">No Show</span>';
break;
default:
echo '<span class="badge badge-secondary">NA</span>';
}
?>
</td>
<td align="center">
<button type="button" class="btn btn-flat btn-default btn-sm dropdown-toggle dropdown-icon" data-toggle="dropdown">
Action
<span class="sr-only">Toggle Dropdown</span>
</button>
<div class="dropdown-menu" role="menu">
<a class="dropdown-item view_data" href="javascript:void(0)" data-id="<?php echo $row['code'] ?>">View</a>
</div>
</td>
</tr>
<?php endwhile; ?>
</tbody>
</table>
```

```
</div>
</div>
</div>
<script>
var indiList;
$(document).ready(function(){
$('.view_data').click(function(){
uni_modal("Indiviual Details","individual/view_details.php?code="+$(this).attr('data-id'))
})
$('#selectAll').change(function(){
// if($(this).is(":checked") == true){
//         $('.invCheck').prop("checked",true)
// }else{
//         $('.invCheck').prop("checked",false)
// }
var _this = $(this)
count = indiList.api().rows().data().length
for($i = 0 ; $i < count; $i++){
var node = indiList.api().row($i).node()
console.log($(node).find('.invCheck'))
if(_this.is(":checked") == true){
$(node).find('.invCheck').prop("checked",true)
$('#selected_opt').show('slow')
}else{
$(node).find('.invCheck').prop("checked",false)
$('#selected_opt').hide('slow')
}
}
})

})
$(function(){
indiList = $('#indi-list').dataTable({
columnDefs:[{
targets:[0,7],
orderable:false
}],
order:[[1,'asc']],
});
// console.log(indiList)
$(indiList.fnGetNodes()).find('.invCheck').change(function(){
```

```
if($(this).is(":checked")==true){
if($('#selected_opt').is(':visible') == false){
$('#selected_opt').show('slow')
}

}else{
if($(indiList.fnGetNodes()).find('.invCheck:checked').length <= 0){
if($('#selected_opt').is(':visible') == true){
$('#selected_opt').hide('slow')
}
}
}
if($(indiList.fnGetNodes()).find('.invCheck:checked').length                                                    ==
$(indiList.fnGetNodes()).find('.invCheck').length){
$('#selectAll').prop('checked',true)
}else if($(indiList.fnGetNodes()).find('.invCheck:checked').length <= 0){
$('#selectAll').prop('checked',false)
}else{
$('#selectAll').prop('checked',false)
}
})

$('#selected_go').click(function(){
start_loader();
var ids = [];
$(indiList.fnGetNodes()).find('.invCheck:checked').each(function(){
ids.push($(this).val())
})
var _action = $('#w_selected').val()
$.ajax({
url:_base_url_+'classes/Master.php?f=multiple_action',
method:"POST",
data:{ids:ids,_action:_action},
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured",'error');
end_loader();
},
success:function(resp){
if(typeof resp =='object' && resp.status == 'success'){
```

```
location.reload();
}else if(resp.status == 'failed' && !!resp.msg){
alert_toast(resp.msg,'error');
end_loader()
}else{
alert_toast("An error occured",'error');
end_loader();
console.log(resp)
}
}
})
})
})
</script>
```

About.php:

```
<!-- Header-->
<header class="bg-dark py-5" id="main-header">
<div class="container px-4 px-lg-5 my-5">
<div class="text-center text-white">
<h1 class="display-4 fw-bolder">About <?php echo $_settings->info('name') ?></h1>
</div>
</div>
</header>
<section class="py-5">
<div class="container">
<div class="card rounded-0">
<div class="card-body">
<?php include "about.html" ?>
</div>
</div>
</div>
</section>
```

Home.php:

```
<!-- Header-->
<header class="bg-dark py-5" id="main-header">
<div class="container px-4 px-lg-5 my-5">
<div class="text-center text-white">
<h1 class="display-4 fw-bolder">Welcome <?php echo $_settings->info('name') ?></h1>
</div>
</div>
</header>
<!-- Section-->
<?php
$sched_arr = array();
$max = 0;
?>
<section class="py-5">
<div class="container d-flex justify-content-center">
<div class="card col-md-6 p-0">
<div class="card-header">
<div class="card-title text-center w-100">Login</div>
</div>
<div class="card-body">
<form action="" id="login-client">
<div class="form-group">
<label for="email" class='control-label'>Email</label>
<input type="text" class="form-control" name="email" required>
</div>
<div class="form-group">
<label for="password" class='control-label'>Password</label>
<input type="password" class="form-control" name="password" required>
</div>
<div class="form-group d-flex justify-content-end">
<button class="btn btn-sm btn-primary btn-flat">Login</button>
</div>
</form>
</div>
</div>
</div>
</section>
<script>
</script>
```

Mange Account.php:

```
<?php
if(isset($_GET['id']) && $_GET['id'] > 0){
$qry = $conn->query("SELECT * from `accounts` where id = '{$_GET['id']}' ");
if($qry->num_rows > 0){
foreach($qry->fetch_assoc() as $k => $v){
$$k=$v;
}
}
}
?>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title"><?php echo isset($_GET['id']) ? 'Update Account' : "Create New Account"; ?></h3>
</div>
<div class="card-body">
<div class="container-fluid">
<form id="account-form">
<input type="hidden" name="id" value='<?php echo isset($id)? $id : " ?>'>
<div class="form-group">
<label class="control-label">Account Number</label>
<input type="text" class="form-control col-sm-6" name="account_number" value="<?php echo isset($account_number)? $account_number : " ?>" required>
</div>
<hr>
<div class="row">
<div class="form-group col-sm-4">
<label class="control-label">First Name</label>
<input type="text" class="form-control" name="firstname" value="<?php echo isset($firstname)? $firstname : " ?>" required>
</div>
<div class="form-group col-sm-4">
<label class="control-label">Middle Name</label>
<input type="text" class="form-control" name="middlename" value="<?php echo isset($middlename)? $middlename : " ?>" placeholder="(optional)" required>
</div>
<div class="form-group col-sm-4">
<label class="control-label">Last Name</label>
<input type="text" class="form-control" name="lastname" value="<?php echo isset($lastname)? $lastname : " ?>" required>
```

```
</div>
</div>
<hr>
<div class="form-group">
<label class="control-label">Email</label>
<input type="text" class="form-control col-sm-6" name="email" value="<?php echo isset($email)? $email : " ?>" required>
</div>

<div class="form-group">
<label class="control-label">Password</label>
<div class="input-group m-0 p-0  col-sm-6">
<input type="text" class="form-control" name="generated_password" value="<?php echo isset($generated_password)? $generated_password : " ?>" <?php echo (!isset($id)) ? "required" : " ?>>
<div class="input-group-append">
<button class="btn btn-outline-primary" type="button" id="generate_pass">Generate</button>
</div>
</div>
</div>
<?php if(!isset($id)): ?>
<div class="form-group">
<label class="control-label">PIN</label>
<input type="text" class="form-control col-sm-6" name="pin" value="<?php echo isset($pin)? $pin : " ?>" required>
</div>
<div class="form-group">
<label class="control-label">Beginning Balance</label>
<input type="number" step='any' min = "0" class="form-control col-sm-6 text-right" name="balance" value="0" required>
</div>
<?php endif; ?>
</form>
</div>
</div>
<div class="card-footer">
<div class="d-flex w-100">
<button form="account-form" class="btn btn-primary mr-2">Save</button>
<a href="./?page=accounts" class="btn btn-default">Cancel</a>
</div>
</div>
</div>
```

```
<script>
$(function(){
$('#generate_pass').click(function(){
var randomstring = Math.random().toString(36).slice(-8);
$('[name="generated_password"]').val(randomstring)
})
$('[name="account_number"]').on('input',function(){
if($('._checks').length > 0)
$('._checks').remove()
$('button[form="account-form"]').attr('disabled',true)
$(this).removeClass('border-danger')
$(this).removeClass('border-success')
var checks = $('<small class="_checks">')
checks.text("Checking availablity")
$('[name="account_number"]').after(checks)
$.ajax({
url:_base_url_+'classes/Master.php?f=check_account',
method:'POST',
data:{id:$('[name="id"]').val(),account_number: $(this).val()},
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'available'){
checks.addClass('text-success')
checks.text('Available')
$('[name="account_number"]').addClass('border-success')
$('button[form="account-form"]').attr('disabled',false)
}else if(resp.status == 'taken'){
checks.addClass('text-danger')
checks.text('Account already exist')
$('[name="account_number"]').addClass('border-danger')
$('button[form="account-form"]').attr('disabled',true)
}else{
alert_toast('An error occured',"error")
$('[name="account_number"]').addClass('border-danger')
console.log(resp)
}
```

```
end_loader()
}
})
})
$('#account-form').submit(function(e){
e.preventDefault()
start_loader()
if($('.err_msg').length > 0)
$('.err_msg').remove()
$.ajax({
url:_base_url_+'classes/Master.php?f=save_account',
method:'POST',
data:$(this).serialize(),
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
location.href="./?page=accounts"
}else if(!!resp.msg){
var msg = $('<div class="err_msg"><div class="alert alert-danger">'+resp.msg+'</div></div>')
$('#account-form').prepend(msg)
msg.show('slow')
}else{
alert_toast('An error occured',"error")
console.log(resp)
}
end_loader()
}
})
})
})
</script>
```

Transactions.php:

```
<?php if($_settings->chk_flashdata('success')): ?>
<script>
alert_toast("<?php echo $_settings->flashdata('success') ?>",'success')
</script>
<?php endif;?>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title">Transactions</h3>
<!-- <div class="card-tools">
<a href="?page=accounts/manage_account" class="btn btn-flat btn-primary"><span class="fas fa-plus"></span>  Create New</a>
</div> -->
</div>
<div class="card-body">
<div class="container-fluid">
<table class="table table-bordered table-stripped" id="indi-list">
<colgroup>
<col width="5%">
<col width="15%">
<col width="20%">
<col width="20%">
<col width="15%">
<col width="15%">
<col width="10%">
</colgroup>
<thead>
<tr>
<th class="text-center">#</th>
<th>Account #</th>
<th>Name</th>
<th>Amount</th>
<th>Transaction</th>
<th>Date Created</th>
</tr>
</thead>
<tbody>
<?php
$i = 1;
$qry = $conn->query("SELECT t.*,concat(a.lastname,', ',a.firstname, a.middlename) as
```

```
`name`,a.account_number from `transactions` t inner join `accounts` a on a.id = t.account_id order by unix_timestamp(t.date_created) desc ");
while($row = $qry->fetch_assoc()):
?>


<tr>
<td class="text-center"><?php echo $i++; ?></td>
<td><?php echo $row['account_number'] ?></td>
<td><?php echo $row['name'] ?></td>
<td class='text-right'><?php echo number_format($row['amount'],2) ?></td>
<td><?php echo $row['remarks'] ?></td>
<td><?php echo $row['date_created'] ?></td>
</tr>
<?php endwhile; ?>
</tbody>
</table>
</div>
</div>
</div>
<script>
var indiList;
$(document).ready(function(){
// $('.view_data').click(function(){
//         uni_modal("Indiviual Details","accounts/view_details.php?code="+$(this).attr('data-id'))
// })
})
$(function(){
$('#indi-list').dataTable()
})
</script>
```

Users.php:

```
<?php
require_once('../config.php');
Class Users extends DBConnection {
private $settings;
public function __construct(){
```

```
global $_settings;
$this->settings = $_settings;
parent::__construct();
}
public function __destruct(){
parent::__destruct();
}
public function save_users(){
extract($_POST);
$data = '';
foreach($_POST as $k => $v){
if(!in_array($k,array('id','password'))){
if(!empty($data)) $data .=" , ";
$data .= " {$k} = '{$v}' ";
}
}
if(!empty($password) && !empty($id)){
$password = md5($password);
if(!empty($data)) $data .=" , ";
$data .= " `password` = '{$password}' ";
}

if(isset($_FILES['img']) && $_FILES['img']['tmp_name'] != ''){
$fname = 'uploads/'.strtotime(date('y-m-d H:i')).'_'.$_FILES['img']['name'];
$move = move_uploaded_file($_FILES['img']['tmp_name'],'../'. $fname);
if($move){
$data .=" , avatar = '{$fname}' ";
if(isset($_SESSION['userdata']['avatar']) && is_file('../'.$_SESSION['userdata']['avatar']))
unlink('../'.$_SESSION['userdata']['avatar']);
}
}
if(empty($id)){
$qry = $this->conn->query("INSERT INTO users set {$data}");
if($qry){
$this->settings->set_flashdata('success','User Details successfully saved.');
foreach($_POST as $k => $v){
if($k != 'id'){
if(!empty($data)) $data .=" , ";
$this->settings->set_userdata($k,$v);
}
}
```

```
return 1;
}else{
return 2;
}

}else{
$qry = $this->conn->query("UPDATE users set $data where id = {$id}");
if($qry){
$this->settings->set_flashdata('success','User Details successfully updated.');
foreach($_POST as $k => $v){
if($k != 'id'){
if(!empty($data)) $data .=" , ";
$this->settings->set_userdata($k,$v);
}
}
if(isset($fname) && isset($move))
$this->settings->set_userdata('avatar',$fname);

return 1;
}else{
return "UPDATE users set $data where id = {$id}";
}

}
}
public function delete_users(){
extract($_POST);
$qry = $this->conn->query("DELETE FROM users where id = $id");
if($qry){
$this->settings->set_flashdata('success','User Details successfully deleted.');
return 1;
}else{
return false;
}
}
public function save_client(){
extract($_POST);
$data = "";
foreach($_POST as $k => $v){
if(!in_array($k, array('id','password'))){
if(!empty($data)) $data .= ", ";
```

```
$data .= " `{$k}` = '{$v}' ";
}
}
$data .= ", `password` = '".md5($password)."' ";
$data .= ", `generated_password` = '' ";

$sql = "UPDATE accounts set {$data} where id = $id";
$save = $this->conn->query($sql);

if($save){
$this->settings->set_flashdata('success','User Details successfully updated.');
foreach($_POST as $k => $v){
if(!in_array($k,array('id','password'))){
if(!empty($data)) $data .=" , ";
$this->settings->set_userdata($k,$v);
}
}
return 1;
}else{
$resp['error'] = $sql;
return json_encode($resp);
}

}

}

$users = new users();
$action = !isset($_GET['f']) ? 'none' : strtolower($_GET['f']);
switch ($action) {
case 'save':
echo $users->save_users();
break;
case 'save_client':
echo $users->save_client();
break;
echo $users->delete_users();
break;
default:
// echo $sysset->index();
break;
```

```
}
```

Deposit.php:

```
<?php
if(isset($_GET['id']) && $_GET['id'] > 0){
    $qry = $conn->query("SELECT * from `accounts` where id = '{$_GET['id']}' ");
    if($qry->num_rows > 0){
        foreach($qry->fetch_assoc() as $k => $v){
            $$k=$v;
        }
}
}
?>
<?php if($_settings->chk_flashdata('success')): ?>
<script>
        alert_toast("<?php echo $_settings->flashdata('success') ?>",'success')
</script>
<?php endif;?>
<div class="card card-outline card-primary">
    <div class="card-header">
    <h3 class="card-title">Deposit</h3>
    </div>
    <div class="card-body">
        <div class="container-fluid">
            <form id="account-form">
                <div class="form-group">
                    <label class="control-label">Account Number</label>
                    <input type="text" class="form-control col-sm-6" name="account_number" value="<?php echo $_settings->userdata('account_number') ?>" readonly autocomplete="off">
                    <input type="hidden" value="<?php echo $_settings->userdata('id') ?>" name="account_id" >
                    <input type="hidden" value="<?php echo $_settings->userdata('balance') ?>" name="current" >
                </div>
                <div class="form-group">
                    <h4><b>Current Balance: <?php echo number_format($_settings->userdata('balance',2)) ?></b></h4>
                </div>
                <hr>
```

```
            <div class="form-group">
                <label class="control-label">Deposit Amount</label>
                <input type="number" step='any' min = "0" class="form-control col-sm-6 text-right" name="balance" value="0" required>
            </div>
        </form>
    </div>
  </div>
  <div class="card-footer">
    <div class="d-flex w-100">
      <button form="account-form" class="btn btn-primary mr-2">Submit</button>
      <a href="./?page=transaction" class="btn btn-default">Cancel</a>
    </div>
  </div>
</div>
<script>
  $(function(){
    $('#account-form').submit(function(e){
      e.preventDefault()
      start_loader()
      if($('.err_msg').length > 0)
        $('.err_msg').remove()
      $.ajax({
        url:_base_url_+'classes/Master.php?f=deposit',
        method:'POST',
        data:$(this).serialize(),
        dataType:'json',
        error:err=>{
          console.log(err)
          alert_toast("An error occured","error")
          end_loader()
        },
        success:function(resp){
          if(resp.status == 'success'){
            location.reload();
          }else if(!!resp.msg){
            var msg = $('<div class="err_msg"><div class="alert alert-danger">'+resp.msg+'</div></div>')
            $('#account-form').prepend(msg)
            msg.show('slow')
          }else{
            alert_toast('An error occured',"error")
```

```
                console.log(resp)
            }
            end_loader()
        }
    })
})
})
</script>
```

Transfer.php:

```
<?php
if(isset($_GET['id']) && $_GET['id'] > 0){
$qry = $conn->query("SELECT * from `accounts` where id = '{$_GET['id']}' ");
if($qry->num_rows > 0){
foreach($qry->fetch_assoc() as $k => $v){
$$k=$v;
}
}
}
?>
<?php if($_settings->chk_flashdata('success')): ?>
<script>
alert_toast("<?php echo $_settings->flashdata('success') ?>",'success')
</script>
<?php endif;?>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title">Deposit</h3>
</div>
<div class="card-body">
<div class="container-fluid">
<form id="account-form">
<input type="hidden" name="id" value='<?php echo isset($id)? $id : " ?>'>
<div class="row">
<div class="col-md-6 border-right">
<div class="form-group">
<label class="control-label">Account Number</label>
```

```
<input type="text" class="form-control col-sm-6" name="account_number" value="<?php echo $_settings->userdata('account_number') ?>" readonly autocomplete="off">
<input type="hidden" value="<?php echo $_settings->userdata('id') ?>" name="account_id" >
<input type="hidden" value="<?php echo $_settings->userdata('balance') ?>" name="current" >
</div>
<div class="form-group">
<h4><b>Current Balance: <?php echo number_format($_settings->userdata('balance',2)) ?></b></h4>
</div>
</div>
<div class="col-md-6">
<div class="form-group">
<label class="control-label">Transfer To</label>
<input type="text" class="form-control col-sm-6" name="transfer_number" value="<?php echo isset($transfer_number)? $transfer_number : " ?>" required autocomplete="off">
</div>
<hr>
<div class="form-group">
<input type="hidden" name="transfer_id" value="">
<label class="control-label">Name</label>
<input type="text" class="form-control" id="transfer_name"  name="transfer_name" readonly>
</div>
</div>
</div>

<hr>
<div class="form-group">
<label class="control-label">Deposit Amount</label>
<input type="number" step='any' min = "0" class="form-control col-sm-6 text-right" name="balance" value="0" required>
</div>
</form>
</div>
</div>
<div class="card-footer">
<div class="d-flex w-100">
<button form="account-form" class="btn btn-primary mr-2">Submit</button>
<a href="./?page=transaction" class="btn btn-default">Cancel</a>
</div>
</div>
</div>
<script>
```

```
$(function(){
$('#generate_pass').click(function(){
var randomstring = Math.random().toString(36).slice(-8);
$('[name="generated_password"]').val(randomstring)
})
$('[name="account_number"]').on('input',function(){
if($('._checks').length > 0)
$('._checks').remove()
$('[name="account_id"]').val('')
$('#name').val('')
$('#balance').val('')
$(this).removeClass('border-danger')
$(this).removeClass('border-success')
if($(this).val() == '')
return false;
$('button[form="account-form"]').attr('disabled',true)
var checks = $('<small class="_checks">')
checks.text("Checking availablity")
$('[name="account_number"]').after(checks)
$.ajax({
url:_base_url_+'classes/Master.php?f=get_account',
method:'POST',
data:{account_number: $(this).val()},
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
checks.hide('slow').remove()
$('[name="account_number"]').addClass('border-success')
$('button[form="account-form"]').attr('disabled',false)
$('[name="account_id"]').val(resp.data.id)
$('#name').val(resp.data.name)
$('#balance').val(resp.data.balance)
}else if(resp.status == 'not_exist'){
checks.addClass('text-danger')
checks.text('Account doesn\'t exist')
$('[name="account_number"]').addClass('border-danger')
```

```
$('button[form="account-form"]').attr('disabled',true)
}else{
alert_toast('An error occured',"error")
$('[name="account_number"]').addClass('border-danger')
console.log(resp)
}
end_loader()
}
})
})
$('[name="transfer_number"]').on('input',function(){
if($('._checks2').length > 0)
$('._checks2').remove()
$('[name="transfer_id"]').val('')
$('#transfer_name').val('')
$(this).removeClass('border-danger')
$(this).removeClass('border-success')
if($(this).val() == '')
return false;
$('button[form="account-form"]').attr('disabled',true)
var checks = $('<small class="_checks2">')
checks.text("Checking availablity")
$('[name="transfer_number"]').after(checks)
$.ajax({
url:_base_url_+'classes/Master.php?f=get_account',
method:'POST',
data:{account_number: $(this).val()},
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
checks.hide('slow').remove()
$('[name="transfer_number"]').addClass('border-success')
$('button[form="account-form"]').attr('disabled',false)
$('[name="transfer_id"]').val(resp.data.id)
$('#transfer_name').val(resp.data.name)
}else if(resp.status == 'not_exist'){
```

```
checks.addClass('text-danger')
checks.text('Account doesn\'t exist')
$('[name="transfer_number"]').addClass('border-danger')
$('button[form="account-form"]').attr('disabled',true)
}else{
alert_toast('An error occured',"error")
$('[name="transfer_number"]').addClass('border-danger')
console.log(resp)
}
end_loader()
}
})
})
$('#account-form').submit(function(e){
e.preventDefault()
if(parseFloat($('[name="current"]').val()) < parseFloat($('[name="balance"]').val())){
alert_toast("Amount is greater than client's balance",'warning')
return false;
}
start_loader()
if($('.err_msg').length > 0)
$('.err_msg').remove()
$.ajax({
url:_base_url_+'classes/Master.php?f=transfer',
method:'POST',
data:$(this).serialize(),
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
location.reload();
}else if(!!resp.msg){
var msg = $('<div class="err_msg"><div class="alert alert-danger">'+resp.msg+'</div></div>')
$('#account-form').prepend(msg)
msg.show('slow')
}else{
alert_toast('An error occured',"error")
```

```
console.log(resp)
}
end_loader()
}
})
})
})
</script>
```

Withdraw.php:

```
<?php
if(isset($_GET['id']) && $_GET['id'] > 0){
$qry = $conn->query("SELECT * from `accounts` where id = '{$_GET['id']}' ");
if($qry->num_rows > 0){
foreach($qry->fetch_assoc() as $k => $v){
$$k=$v;
}
}
}
?>
<?php if($_settings->chk_flashdata('success')): ?>
<script>
alert_toast("<?php echo $_settings->flashdata('success') ?>",'success')
</script>
<?php endif;?>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title">Widthdraw</h3>
</div>
<div class="card-body">
<div class="container-fluid">
<form id="account-form">
<div class="form-group">
<label class="control-label">Account Number</label>
<input type="text" class="form-control col-sm-6" name="account_number" value="<?php echo $_settings-
```

```
>userdata('account_number') ?>" readonly autocomplete="off">
<input type="hidden" value="<?php echo $_settings->userdata('id') ?>" name="account_id" >
<input type="hidden" value="<?php echo $_settings->userdata('balance') ?>" name="current" >
</div>
<div class="form-group">
<h4><b>Current Balance: <?php echo number_format($_settings->userdata('balance',2)) ?></b></h4>
</div>
<hr>
<div class="form-group">
<label class="control-label">Deposit Amount</label>
<input type="number" step='any' min = "0" class="form-control col-sm-6 text-right" name="balance"
value="0" required>
</div>
</form>
</div>
</div>
<div class="card-footer">
<div class="d-flex w-100">
<button form="account-form" class="btn btn-primary mr-2">Submit</button>
<a href="./?page=transaction" class="btn btn-default">Cancel</a>
</div>
</div>
</div>
<script>
$(function(){

$('#account-form').submit(function(e){
e.preventDefault()
if(parseFloat($('[name="current"]').val()) < parseFloat($('[name="balance"]').val())){
alert_toast("Amount is greater than your current balance",'warning')
return false;
}
start_loader()
if($('.err_msg').length > 0)
$('.err_msg').remove()
$.ajax({
url:_base_url_+'classes/Master.php?f=withdraw',
method:'POST',
data:$(this).serialize(),
dataType:'json',
error:err=>{
```

```
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
location.reload();
}else if(!!resp.msg){
var msg = $('<div class="err_msg"><div class="alert alert-danger">'+resp.msg+'</div></div>')
$('#account-form').prepend(msg)
msg.show('slow')
}else{
alert_toast('An error occured',"error")
console.log(resp)
}
end_loader()
}
})
})
})
</script>
```

Announcement.php

```
<!-- Header-->
<header class="bg-dark py-5" id="main-header">
<div class="container px-4 px-lg-5 my-5">
<div class="text-center text-white">
<h1 class="display-4 fw-bolder">Announcements</h1>
</div>
</div>
</header>
<section class="py-5">
<div class="container">
<div class="card rounded-0">
<div class="card-body">
<?php
$qry = $conn->query("SELECT * FROM `announcements` order by unix_timestamp(date_created) desc");
```

```
while($row = $qry->fetch_assoc()):
$row['announcement'] = strip_tags(stripslashes(html_entity_decode($row['announcement'])))

?>
<a class="card text-dark card-outline card-primary mb-2 view_data" href="javascript:void(0)" data-id='<?php echo $row['id'] ?>' data-title='<?php echo $row['title'] ?>' >
<div class="card-header">
<h5 class="card-title"><?php echo $row['title'] ?></h5>
<span class="float-right text-muted"><?php echo date("M d,Y h:i A",strtotime($row['date_created'])) ?></span>
</div>
<div class="card-header">
<p class="truncate"><?php echo $row['announcement'] ?></p>
</div>
</a>
<?php endwhile; ?>
</div>
</div>
</div>
</section>
<script>
$(function(){
$('.view_data').click(function(){
uni_modal($(this).attr('data-title'),'./view_accouncement.php?id='+$(this).attr('data-id'),'mid-large')
})
})
</script>
```

Mange Announcement.php

```
<?php
if(isset($_GET['id']) && $_GET['id'] > 0){
$qry = $conn->query("SELECT * from `announcements` where id = '{$_GET['id']}' ");
if($qry->num_rows > 0){
foreach($qry->fetch_assoc() as $k => $v){
$$k=$v;
```

```
}
}
}
?>
<div class="card card-outline card-primary">
<div class="card-header">
<h3 class="card-title"><?php echo isset($_GET['id']) ? 'Update Announcement' : "Create New Announcement";
?></h3>
</div>
<div class="card-body">
<div class="container-fluid">
<form id="announcements-form">
<input type="hidden" name="id" value='<?php echo isset($id)? $id : " ?>'>
<div class="form-group">
<label class="control-label">Title</label>
<input type="text" class="form-control col-sm-8" name="title" value="<?php echo isset($title)? $title : " ?>"
required>
</div>
<div class="form-group">
<label class="control-label">Announcement</label>
<textarea type="text" class="form-control summernote" name="announcement" required><?php echo
isset($announcement)? stripslashes(html_entity_decode($announcement)) : " ?></textarea>
</div>
</form>
</div>
</div>
<div class="card-footer">
<div class="d-flex w-100">
<button form="announcements-form" class="btn btn-primary mr-2">Save</button>
<a href="./?page=announcements" class="btn btn-default">Cancel</a>
</div>
</div>
</div>
<script>
$(function(){
$('.summernote').summernote({
height: 200,
toolbar: [
[ 'style', [ 'style' ] ],
[ 'font', [ 'bold', 'italic', 'underline', 'strikethrough', 'superscript', 'subscript', 'clear'] ],
[ 'fontname', [ 'fontname' ] ],
```

```
[ 'fontsize', [ 'fontsize' ] ],
[ 'color', [ 'color' ] ],
[ 'para', [ 'ol', 'ul', 'paragraph', 'height' ] ],
[ 'table', [ 'table' ] ],
[ 'view', [ 'undo', 'redo', 'fullscreen', 'codeview', 'help' ] ]
]
})

$('#announcements-form').submit(function(e){
e.preventDefault()
start_loader()
if($('.err_msg').length > 0)
$('.err_msg').remove()
$.ajax({
url:_base_url_+'classes/Master.php?f=save_announcement',
method:'POST',
data:$(this).serialize(),
dataType:'json',
error:err=>{
console.log(err)
alert_toast("An error occured","error")
end_loader()
},
success:function(resp){
if(resp.status == 'success'){
location.href="./?page=announcements"
}else if(!!resp.msg){
var msg = $('<div class="err_msg"><div class="alert alert-danger">'+resp.msg+'</div></div>')
$('#announcements-form').prepend(msg)
msg.show('slow')
}else{
alert_toast('An error occured',"error")
console.log(resp)
}
end_loader()
}
})
})
})
</script>
```

View Announcement.php:

```
<?php
require_once('config.php');
if(isset($_GET['id']) && $_GET['id'] > 0){
    $qry = $conn->query("SELECT * from `announcements` where id = '{$_GET['id']}' ");
    if($qry->num_rows > 0){
        foreach($qry->fetch_assoc() as $k => $v){
            $$k=$v;
        }
}
}
?>
<style>
#uni_modal .modal-content>.modal-footer{
    display:none;
}
#uni_modal .modal-body{
    padding:0 !important;
}
</style>
<div class="container-fluid p-2">
    <div><?php echo stripslashes(html_entity_decode($announcement)) ?></div>
    <div class="d-flex w-100 justify-content-end">
        <small class="mb-3"><?php echo date("M d,Y h:i A",strtotime($date_created)) ?></small>
    </div>
</div>
<div class="modal-footer p-0">
<button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
</div>
```

CHAPTER 7

TESTING AND DEBUGGING

The implementation phase of software development is concerned with translating design specification into source code. The preliminary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified, and so that debugging, testing and modifications are eased. This goal can be achieved by making the source code as clear and straight forward as possible. Simplicity, clarity and elegance are the hallmark of good programs, obscurity, cleverness, and complexity are indications of inadequate design and misdirected thinking.

Source code clarity is enhanced by structured coding techniques, by good coding style, by, appropriate supporting documents, by good internal comments, and by feature provided in modern programming languages.

The implementation team should be provided with a well-defined set of software requirement, an architectural design specification, and a detailed design description. Each team member must understand the objectives of implementation.

TERMS IN TESTING FUNDAMENTAL

1. **Error**

   The term error is used in two ways. It refers to the difference between the actual output of software and the correct output, in this interpretation, error is essential a measure of the difference between actual and ideal. Error is also to used to refer to human action that result in software containing a defect or fault.

2. **Fault**

   Fault is a condition that causes to fail in performing its required function. A fault is a

basic reason for software malfunction and is synonymous with the commonly used term Bug.

3. **Failure**

   Failure is the inability of a system or component to perform a required function according to its specifications. A software failure occurs if the behavior of the software is the different from the specified behavior. Failure may be caused due to functional or performance reasons.

Types of Testing:

- **Unit Testing**

  The term unit testing comprises the sets of tests performed by an individual programmer prior to integration of the unit into a larger system.

  A program unit is usually small enough that the programmer who developed it can test it in great detail, and certainly in greater detail than will be possible when the unit is integrated into an evolving software product. In the unit testing the programs are tested separately, independent of each other. Since the check is done at the program level, it is also called program teasing.

- **Module Testing**

  A module and encapsulates related component. So can be tested without other system module.

- **Subsystem Testing**

  Subsystem testing may be independently design and implemented common problems are sub-system interface mistake in this checking we concentrate on it.

There are four categories of tests that a programmer will typically perform on a program unit.

1) Functional Test
2) Performance test
3) Stress test

- **Functional Test**

  Functional test cases involve exercising the code with Nominal input values for which expected results are known; as well as boundary values (minimum values, maximum values and values on and just outside the functional boundaries) and special values.

- **Performance Test**

  Performance testing determines the amount of execution time spent in various parts of the unit, program throughput, response time, and device utilization by the program unit. A certain amount of avoid expending too much effort on fine-tuning of a program unit that contributes little to the over all performance of the entire system. Performance testing is most productive at the subsystem and system levels.

- **Stress Test**

  Stress test are those designed to intentionally break the unit. A great deal can be learned about the strengths and limitations of a program by examining the manner in which a program unit breaks.

DEBUGGING

Defect testing is intended to find areas where the program does not confirm to its specifications. Tests are designed to reveal the presence of defect in the system. When defect have been found in the program. There must be discovered and removed. This is called “Debugging”.

CHAPTER 8

CONCLUSION & FUTURE SCOPE

This project is developed to nurture the needs of a user in a banking sector by embedding all the tasks of transactions taking place in a bank. Future version of this project will still be much enhanced than the current version. Writing and depositing checks are perhaps the most fundamental ways to move money in and out of a checking account, but advancements in technology have added ATM and debit card transactions. All banks have rules about how long it takes to access your deposits, how many debit card transactions you're allowed in a day, and how much cash you can withdraw from an ATM. Access to the balance in your checking account can also be limited by businesses that place holds on your funds.

Banks are providing internet banking services also so that the customers can be attracted. By asking the bank employs we came to know that maximum numbers of internet bank account holders are youth and business man. Online banking is an innovative tool that is fast becoming a necessity. It is a successful strategic weapon for banks to remain profitable in a volatile and competitive marketplace of today. If proper training should be given to customer by the bank employs to open an account will be beneficial secondly the website should be made friendlier from where the customers can directly make and access their accounts.

Thus, the Bank Management System it is developed and executed successfully

FUTURE SCOPE

The “Banking Online System is a big and ambitious project. I am thankful for being provided this great opportunity to work on it. As already mentioned, this project has gone through extensive research work. On the basis of the research work, we have successfully designed and implemented banking online System. To know what the future of online banking looks like, it’s probably worth looking at the present – online banking isn’t new. When you think of online banking, you probably think about a computer (either a desktop or laptop), a three or four step security process and then an interface that lets you view the balance of your various bank accounts and credit cards, whilst permitting you to

transfer money and pay bills. And you’re not wrong either. The most valuable future looks are following below:

1. More branches of the bank, maybe it will be international, that means more ATM machines outside.
2. Customer issues development based on their needs, so the help desk will be aware of their needs and easy to use.
3. Developing a mobile App for banking system that help users to do the obtained his operations without go to the bank only he needs to sign in using his A/C NO. And password and then use your own PIN. Finally the system will update automatically.

REFERENCES

1. Iztok Fajfar “Start Programming Using HTML,CSS and JavaScript” issue 21 October 2015 Pub. Location New York.
2. Preston Zhang “Practical Guide to Oracle SQL,T-SQL and MySQL” issue 21 November 2017 Pub. Location Boca Raton
3. Mitsunori Ogihara “Fundamentals of Java Programming” issue 25 January 2016 Pub.
    a. Springe,Cham
4. Richard Wiener “Fundamentals of OOP and Data Structures in Java” issue on June 2012 Pub. Cambridge University Press.
5. Dr. Geeta Sharma, “Study of Internet Banking Scenario in India”, International Journal of Emerging Research in Management &Technology, ISSN: 2278- 9359, Volume 5, Issue 5, 2016, pp.43-48.
6. Anju Dagar, “Online Banking : Benefits and Related Issues”, International Journal of Commerce, Business and Management (IJCBM), ISSN: 2319–2828,Vol. 3, No. 5, 2014, pp.715-719.
7. Ebubeogu Amarachukwu Felix, “ Bank Customers Management System”, International Journal of Scientific & Technology Research Volume 4, Issue 08, 2015, pp.326- 343.
8. Mahmood Shah, “E-Banking Management: Issues, Solutions, and Strategies”, 2009. [9]Muhammad Abdus Sattar Titu and Md. Azizur Rahman, “ Online Banking System-Its Application in Some Selected Private Commercial Banks in Bangladesh”, IOSR Journal of Business and Management (IOSR-JBM) e-ISSN: 2278- 487X, Volume 9, Issue 4,2013, pp.37-44.
9. Bahman Saeidipour , Hojat Ranjbar and Saeed Ranjbar, “Adoption of Internet banking”, IOSR Journal of Business and Management (IOSR-JBM) e-ISSN: 2278-487X, Volume 11, Issue 2 2013,
10. D.Amutha, “A Study of Consumer Awareness towards e- Banking”, Int J Econ Manag Sci, ISSN: 2162-6359, Volume 5, issue 4, 2016,

11. Sakkthivel, A. M. (2006). Impact of Demographics On The Consumption Of Different Services Online India. Journal of Internet Banking and Commerce, December 2006, Vol. 11, No.3.
12. Pikkarainen, T., Pikkarainen, K., Karjaluoto, H., and Pahnila, S. (2004). Consumer acceptance of online banking: An extension of the technology acceptance model. Internet Research, 14(3), 224-235.
13. Baldock, R. (1997). The Virtual Bank: Four Marketing Scenarios for the Future. Journal of Financial Service Marketing, 1(3), 260-268.
14. Chan, L.M. (2001). Is Online Banking Safe, the Star (February), 42