

VISVESVARAYA TECHNOLOGICAL UNIVERSITY**“Jnana Sangama”, BELAGAVI – 590018****A MINI PROJECT REPORT****ON****“ONLINE BANK MANAGEMENT”**

Submitted in partial fulfillment of requirements for the *course*
DBMS Laboratory with Mini Project [17CSL58] of *Fifth Semester*
of *Bachelor of Engineering in Computer Science & Engineering*
during the academic year 2019-20.

Submitted By**KIRAN B A**
4MH17CS037**SUHAS Y**
4MH17CS101**Under the Guidance of**

Prof. Divya H N
Assistant Professor,
Dept. of CS&E,
MIT MYSORE – 571438.

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING****MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE****Belawadi, S.R. Patna (T), Mandya (D) – 571438.****2019 - 2020**

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE**Belawadi, S.R. Patna (T), Mandya (D) – 571438.****DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING****CERTIFICATE**

This is to certify that the mini project work entitled “**ONLINE BANK MANAGEMENT**” is a bonafide work carried out by **KIRAN B A (4MH17CS037)** and **SUHAS Y (4MH17CS101)** in partial fulfillment for the **DBMS Laboratory with Mini Project [17CSL58]** prescribed by the Visvesvaraya Technological University, Belgaum during the year 2019-2020 for the fifth semester B.E. in Computer Science and Engineering. The mini project report has been approved as it satisfies the academic requirements.

Signature of Guide

Prof. Divya H N
Assistant Professor,
Dept. of CS&E
MIT MYSORE

Signature of HOD

Dr. DEEPU R
Professor & HOD,
Dept. of CS&E
MIT MYSORE

Name of the Examiners

1.....

2.....

Signature with date

.....

.....

ACKNOWLEDGEMENT

We sincerely owe our gratitude to all the persons who helped and guided us in completing this mini project work.

We are thankful to **Dr. B.G. Naresh Kumar, Principal, Maharaja Institute of Technology Mysore**, for having supported us in our academic endeavors.

We are extremely thankful to **Dr. Deepu R, Professor & HOD, Department of Computer Science and Engineering**, for his valuable support and his timely inquiries into the progress of the work.

We are greatly indebted to our guide **Prof. Divya H N, Assistant Professor, Department of Computer Science and Engineering**, for his consistent co-operation and support.

We are obliged to all **teaching and non-teaching staff members of Department of Computer Science and Engineering**, for the valuable information provided by them in their respective field's .We are grateful for their co-operation during the period of our mini project.

Lastly we thank almighty, our parents and friends for their constant encouragement without which this mini project would not be possible.

KIRAN B A (4MH17CS037)

SUHAS Y (4MH17CS101)

ABSTRACT

This project is basically a medium for users to perform their bank transactions of any bank account. This Project will provide user friendly interface to perform their actions. It is a platform where it provides all types of service's , oriented to the bank .

It also offers the functionalities like:

- Insertion and deletion in the database can be done only by the admin.
- Regular users cannot make any changes in the database.

We have created pages for inserting values into all the tables. Then to display all the tables, to delete the details from the tables that is already created, the check the transactions of the user.

TABLE OF CONTENTS

<u>Chapter No</u>	<u>Chapter Name</u>	<u>Page No.</u>
01	Introduction	06
	1.1 Introduction to DBMS	06
	1.2 About SQL	07
	1.3 Project Description	08
02	Requirements Specification	09
	2.1 Software Requirements	09
	2.2 Hardware Requirements	09
03	Design	10
	3.1 ER Diagram	10
	3.2 Data Flow Diagram	11
	3.3 Use Case Diagram	12
	3.4 UML Diagram	13
04	Implementation	14
	4.1 Description of Tables	14
	4.2 Triggers and Stored Procedures	17
	4.3 Bank end implementation	17
	4.4 Front end implementation	20
	4.5 SQL Queries	23
05	Testing	32
	Conclusion and Future Enhancement	32
	Snapshots	32
	References	37

Chapter 1

INTRODUCTION

1.1 Introduction to DBMS

DBMS stands for Database Management System. We can break it like this DBMS = Database + Management System. Database is a collection of data and Management System is a set of programs to store and retrieve those data. Therefore DBMS is a collection of inter-related data and set of programs to store & access those data in an easy and effective manner.

Database systems are basically developed for large amount of data. When dealing with huge amount of data, there are two things that require optimization: **Storage of data** and **retrieval of data**.

Storage: According to the principles of database systems, the data is stored in such a way that it acquires lot less space as the redundant data (duplicate data) has been removed before storage. Let's take a layman example to understand this: In a banking system, suppose a customer is having two accounts, one is saving account and another is salary account. Let's say bank stores saving account data at one place (these places are called tables we will learn them later) and salary account data at another place, in that case if the customer information such as customer name, address etc. are stored at both places then this is just a wastage of storage (redundancy/ duplication of data), to organize the data in a better way the information should be stored at one place and both the accounts should be linked to that information somehow. The same thing we achieve in DBMS.

Fast Retrieval of data: Along with storing the data in an optimized and systematic manner, it is also important that we retrieve the data quickly when needed. Database systems ensure that the data is retrieved as quickly as possible.

1.2 About SQL

SQL (Structured Query Language) is a standardized programming language used for managing relational databases and performing various operations on the data in them. Initially created in the 1970s, SQL is regularly used by database administrators, as well as by developers writing data integration scripts and data analysts looking to set up and run analytical queries.

The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics applications. Queries and other SQL operations take the form of commands written as statements -- commonly used SQL statements include select, add, insert, update, delete, create, alter and truncate.

SQL is widely popular because it offers the following advantages:

- Allows user to access data in the RDBMS.
- Allows the user to describe the data.
- Allows user to create and drop database and table.
- Allows user to set permission on table, procedures and views.
- Allows user to create view, stored procedure, function in a database.
- Allows user to define data in a database and manipulate the data.
- Allows to embed within other language using SQL modules, libraries and pre-compilers.

1.3 Project Description

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.

- Our motto is to develop a Project for managing the entire bank process related to User accounts and their various transaction processes efficiently.
- In this Management system we will be providing users to integrate various bank accounts under a single platform.
- Access to privilege banking zone.
- Anywhere Banking with Higher Limits.

This project is basically a medium for users to perform their bank transactions of any bank account. This Project will provide user friendly interface to perform their actions. It is a platform where it provides all types of service's , oriented to the bank .

We have created pages for inserting values into all the tables. Then to display all the tables, to delete the details from the tables that is already created, the check the transactions of the user.

Chapter 2

REQUIREMENT SPECIFICATION

2.1 Software requirements:

Operating System and version: Windows 10 and so on.

Front end tool: Visual Studio Code.

Back end tool: MySQL , XAMPP.

Web Server: Apache Web Server

Programming Language: PHP, JS, HTML5 and CSS3.

2.2 Hardware requirements:

Processor : Any

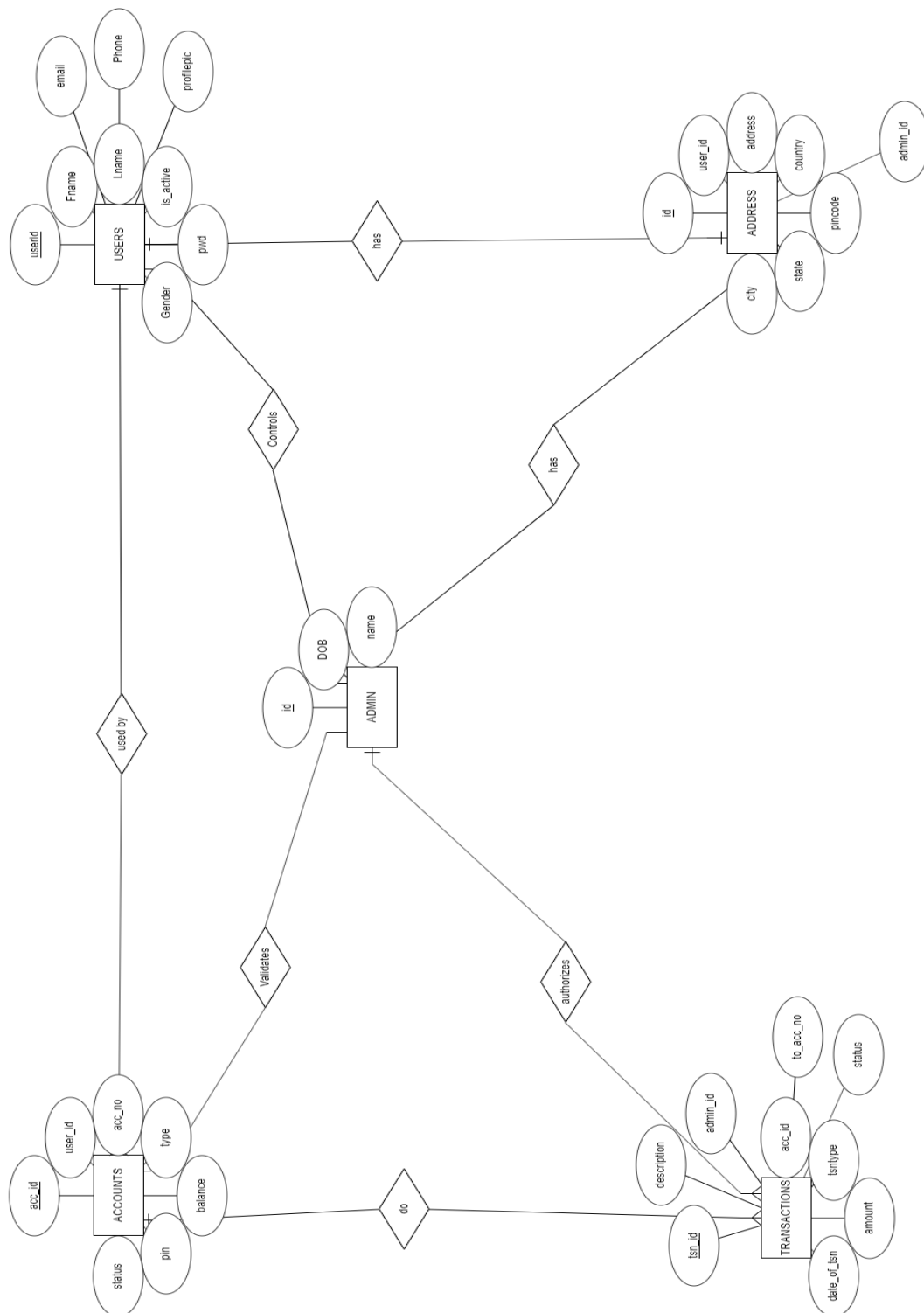
RAM : 1 GB

Hard disk : Suitable Capacity

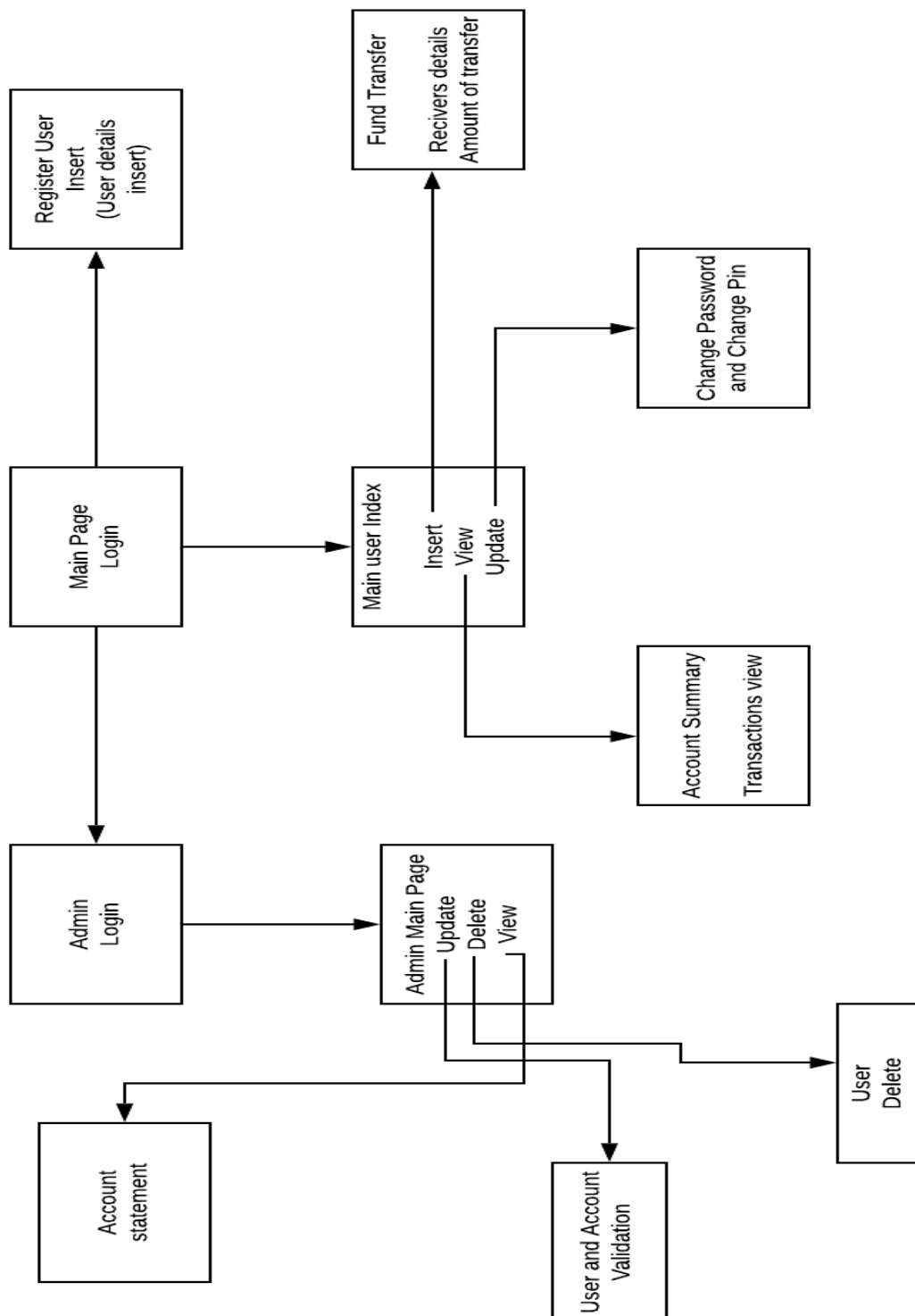
Chapter 3

SYSTEM ANALYSIS AND DESIGN

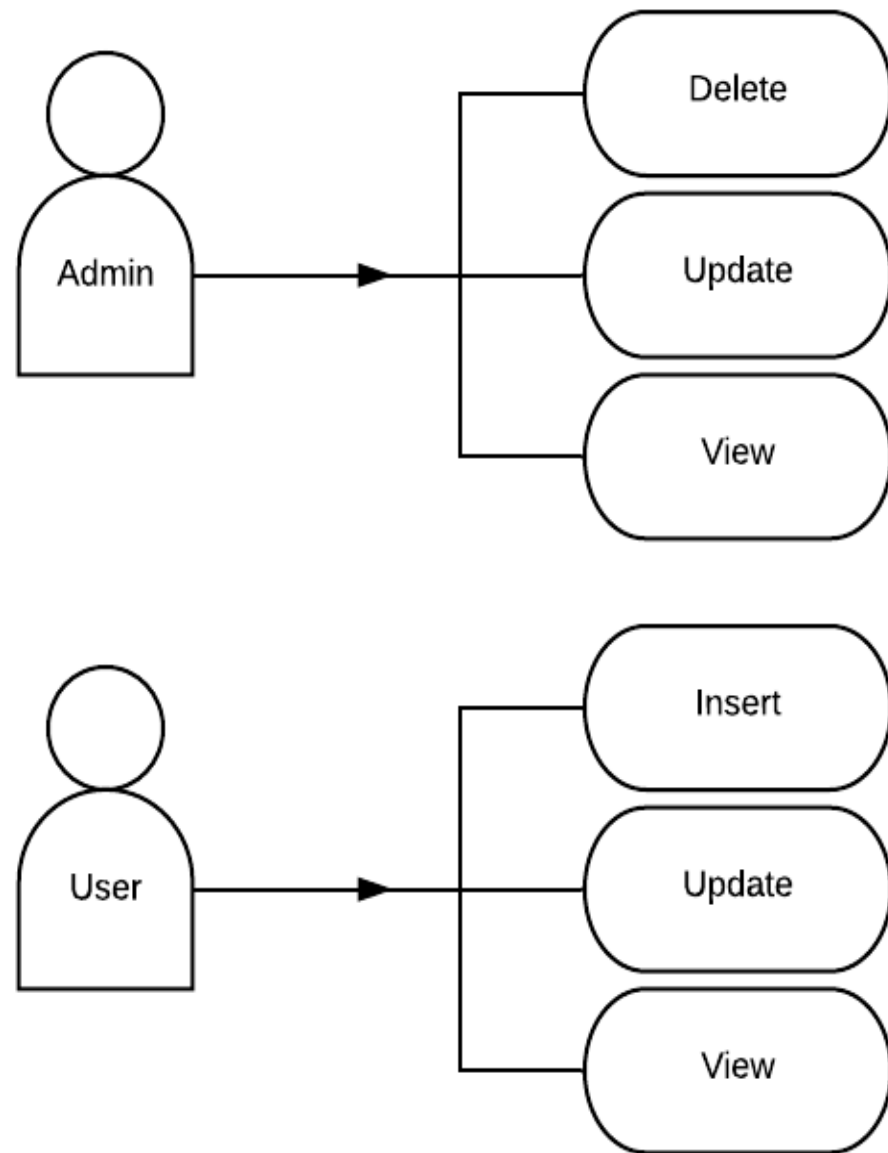
3.1 E R Diagram



3.2 Data Flow Diagram



3.3 Use Case Diagram



Chapter 4

IMPLEMENTATION

4.1 Description of tables:



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(10)			No	None		AUTO_INCREMENT
2	user_id	int(10)			No	None		
3	acc_no	varchar(20)	latin1_swedish_ci		No	None		
4	type	varchar(10)	latin1_swedish_ci		No	None		
5	balance	double			No	1000		
6	pin	int(10)			No	None		
7	status	varchar(10)	latin1_swedish_ci		No	None		
8	bdate	varchar(100)	latin1_swedish_ci		No	None		
9	adm_id 	int(11)			No	1		

Fig.4.1.1 : Accounts Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(10)			No	None		AUTO_INCREMENT
2	user_id	int(10)			No	None		
3	address	varchar(200)	latin1_swedish_ci		No	None		
4	city	varchar(40)	latin1_swedish_ci		No	None		
5	state	varchar(40)	latin1_swedish_ci		No	None		
6	zipcode	int(10)			No	None		
7	country	varchar(40)	latin1_swedish_ci		No	None		
8	adm_id 	int(11)			No	1		

Fig.4.1.2 : Address Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(10)			No	None		AUTO_INCREMENT
2	tx_no	varchar(20)	latin1_swedish_ci		No	None		
3	tx_type	varchar(10)	latin1_swedish_ci		No	None		
4	amount	double			No	None		
5	date	varchar(100)	latin1_swedish_ci		No	None		
6	description	varchar(100)	latin1_swedish_ci		No	None		
7	to_accno	varchar(20)	latin1_swedish_ci		No	None		
8	status	varchar(10)	latin1_swedish_ci		No	None		
9	tdate	varchar(40)	latin1_swedish_ci		No	None		
10	comments	varchar(200)	latin1_swedish_ci		No	None		
11	do_verify	varchar(7)	latin1_swedish_ci		No	false		
12	user_id	int(11)			No	None		
13	adm_id 	int(11)			No	1		

Fig.4.1.3 : Fund Transfer Table


#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	Adm_id 	int(11)			No	None		AUTO_INCREMENT
2	user_id	int(11)			No	None		
3	name	varchar(20)	latin1_swedish_ci		No	None		
4	dob	date			Yes	NULL		

Fig.4.1.4 : Admin Table



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(10)			No	None		AUTO_INCREMENT
2	fname	varchar(100)	latin1_swedish_ci		No	None		
3	lname	varchar(40)	latin1_swedish_ci		No	None		
4	pwd	varchar(100)	latin1_swedish_ci		No	None		
5	email	varchar(100)	latin1_swedish_ci		No	None		
6	phone	varchar(20)	latin1_swedish_ci		No	None		
7	gender	varchar(10)	latin1_swedish_ci		No	None		
8	is_active	varchar(10)	latin1_swedish_ci		No	None		
9	utype	varchar(10)	latin1_swedish_ci		No	None		
10	pics	varchar(200)	latin1_swedish_ci		No	None		
11	bdate	varchar(100)	latin1_swedish_ci		No	None		
12	adm_id 	int(11)			No	1		

Fig.4.1.5 : Users Table

4.2 Triggers and stored procedure:

Triggers: If amount to be transacted exceeds 1,00,000 notify the admin and halt the transaction.

```
DELIMITER $$
```

```
CREATE TRIGGER before_transaction_update
  AFTER INSERT ON tbl_transaction
  FOR EACH ROW
  BEGIN
    IF NEW.amount > 100000 THEN
      update tbl_accounts set tbl_accounts.doVerify='true';
    END IF;
  END$$
```

```
DELIMITER ;
```

Stored Procedure: To obtain the transaction history of a user for a month.

```
CREATE PROCEDURE `statement` (
  IN acc_no bigint
)
BEGIN
  SELECT * FROM tbl_transaction WHERE to_accno = acc_no
  ORDER BY id DESC LIMIT 20;
END
```

4.3 Back end implementation:

4.3.1. Database functions

```
<?php
```

```
require_once 'config.php';
```

```
$dbConn = mysql_connect ($dbHost, $dbUser, $dbPass) or die ('MySQL connect failed. ' . mysql_error());
mysql_select_db($dbName) or die('Cannot select database. ' . mysql_error());
```

```
function dbQuery($sql)
{
    $result = mysql_query($sql) or die(mysql_error());
    return $result;
}
```

```
function dbAffectedRows()
{

```

```
        global $dbConn;
        return mysql_affected_rows($dbConn);
    }

    function dbFetchArray($result, $resultType = MYSQL_NUM) {
        return mysql_fetch_array($result, $resultType);
    }

    function dbFetchAssoc($result)
    {
        return mysql_fetch_assoc($result);
    }

    function dbFetchRow($result)
    {
        return mysql_fetch_row($result);
    }

    function dbFreeResult($result)
    {
        return mysql_free_result($result);
    }

    function dbNumRows($result)
    {
        return mysql_num_rows($result);
    }

    function dbSelect($dbName)
    {
        return mysql_select_db($dbName);
    }

    function dbInsertId()
    {
        return mysql_insert_id();
    }
?>
```

4.3.2. Configuration Functions

```
<?php
ini_set('display_errors', 'on');
//ob_start("ob_gzhandler");
//error_reporting(E_ALL);
```

```
// start the session
session_start();

// database connection config
$dbHost = 'localhost';
$dbUser = 'root';
$dbPass = '';
$dbName = 'db_hlbank';

//Project data
$site_title = 'Online Banking';
$email_id = 'suhaskiran@gmail.com';

// setting up the web root and server root for
// this shopping cart application
$thisFile = str_replace("\\', '", __FILE__);
$docRoot = $_SERVER['DOCUMENT_ROOT'];

$webRoot = str_replace(array($docRoot, 'library/config.php'), "", $thisFile);
$srvRoot = str_replace('library/config.php', "", $thisFile);

define('WEB_ROOT', $webRoot);
define('SRV_ROOT', $srvRoot);

// these are the directories where we will store all
// category and product images
define('USER_IMAGE_DIR', 'images/thumbnails/');

// some size limitation for the category
// and product images

// all category image width must not
// exceed 75 pixels
define('MAX_USER_IMAGE_WIDTH', 180);

// do we need to limit the product image width?
// setting this value to 'true' is recommended
define('LIMIT_USER_WIDTH', true);

// the width for product thumbnail
define('THUMBNAIL_WIDTH', 180);

if (!get_magic_quotes_gpc()) {
    if (isset($_POST)) {
        foreach ($_POST as $key => $value) {
            $_POST[$key] = trim(addslashes($value));
        }
    }
}
```

```

    }

    if (isset($_GET)) {
        foreach ($_GET as $key => $value) {
            $_GET[$key] = trim(addslashes($value));
        }
    }
}

require_once 'database.php';
require_once 'common.php';

?>

```

4.4 Front end implementation:

4.4.1.Admin Login Page

```

<?php
require_once '../library/config.php';
require_once '../library/functions.php';

$errorMessage = '&nbsp;';

if (isset($_POST['txtUserName'])) {
    $result = doLogin();

    if ($result != "") {
        $errorMessage = $result;
    }
}

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title><?php echo $site_title; ?></title>

<link href="<?php echo WEB_ROOT;?>css/admin.css" rel="stylesheet" type="text/css">
<link href="<?php echo WEB_ROOT;?>css/styles.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="<?php echo WEB_ROOT; ?>/admin/style1.css">

<link href="<?php echo WEB_ROOT; ?>library/spry/textfieldvalidation/SpryValidationTextField.css"
rel="stylesheet" type="text/css" />

```

```
<script src="<?php echo WEB_ROOT; ?>library/spry/textfieldvalidation/SpryValidationTextField.js"
type="text/javascript"></script>

<link href="<?php echo WEB_ROOT; ?>library/spry/passwordvalidation/SpryValidationPassword.css"
rel="stylesheet" type="text/css" />
<script src="<?php echo WEB_ROOT; ?>library/spry/passwordvalidation/SpryValidationPassword.js"
type="text/javascript"></script>
<script>
<!--
var sprytextfield1 = new Spry.Widget.ValidationTextField("sprytextfield1", "integer", { validateOn:["blur",
"change"]});
var sprypassword1 = new Spry.Widget.ValidationPassword("sprypassword1", { minChars:6, maxChars: 12,
validateOn:["blur", "change"]});
//-->
</script>
</head>

<body>
  <div class="loginbox">
    
    <h1>Admin Login Here</h1>
    <form action="#" method="post" enctype="multipart/form-data" id="acclogin">
      <div class="errorMessage" align="center"><?php echo $errorMessage; ?></div>
      <p>Admin User ID</p>
      <span id="sprytextfield1" style="text-align:left;">
        <input name="txtUserName" type="text" class="box" id="txtUserName" size="20" maxlength="20">
      <br/>
      <span class="textfieldRequiredMsg">Admin ID is required.</span>
      <span class="textfieldInvalidFormatMsg">Invalid Admin ID.</span>
    </span>
    <p>Password</p>
    <span id="sprypassword1" style="text-align:left;">
      <input name="txtPassword" type="password" class="box" id="txtPassword" size="20">
      <span class="passwordRequiredMsg">Password is required.</span>
      <span class="passwordMinCharsMsg">You must specify at least 6 characters.</span>
      <span class="passwordMaxCharsMsg">You must specify at max 10 characters.</span>
    <input type="submit" name="" value="Login">
    <a href="<?php echo WEB_ROOT; ?>login.php">!...Not an Admin...!</a>
  </form>

</div>

</div>
<p>&nbsp;</p>
</body>
</head>
</html>
```

4.4.2. User Login Page

```
<?php
require_once './library/config.php';
require_once './library/functions.php';

$errorMessage = '&nbsp;';

if (isset($_POST['accno']) && isset($_POST['pass'])) {
    $result = doLogin();

    if ($result != "") {
        $errorMessage = $result;
    }
}

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title><?php echo $site_title; ?></title>

<link href="<?php echo WEB_ROOT;?>css/admin.css" rel="stylesheet" type="text/css">
<link href="<?php echo WEB_ROOT;?>css/styles.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="<?php echo WEB_ROOT; ?>style.css">

<link href="<?php echo WEB_ROOT; ?>library/spry/textfieldvalidation/SpryValidationTextField.css"
rel="stylesheet" type="text/css" />
<script src="<?php echo WEB_ROOT; ?>library/spry/textfieldvalidation/SpryValidationTextField.js"
type="text/javascript"></script>

<link href="<?php echo WEB_ROOT; ?>library/spry/passwordvalidation/SpryValidationPassword.css"
rel="stylesheet" type="text/css" />
<script src="<?php echo WEB_ROOT; ?>library/spry/passwordvalidation/SpryValidationPassword.js"
type="text/javascript"></script>
<script>
<!--
var sprytextfield1 = new Spry.Widget.ValidationTextField("sprytextfield1", "integer", { validateOn:["blur",
"change"]});
var sprypassword1 = new Spry.Widget.ValidationPassword("sprypassword1", { minChars:6, maxChars: 12,
validateOn:["blur", "change"]});
//-->
</script>
</head>
```

```

<body>
<div class="loginbox">

<h1>Login Here</h1>
<form action="#" method="post" enctype="multipart/form-data" id="acclogin">
<div class="errorMessage" align="center"><?php echo $errorMessage; ?></div>
<p>Account Number</p>
<span id="sprytextfield1" style="text-align:left;">
<input type="text" name="accno" placeholder="Enter Username" id="accno" tabindex="10" size="30"
maxlength="30"/>
<br/>
<span class="textfieldRequiredMsg">Account Number is required.</span>
<span class="textfieldInvalidFormatMsg">Invalid Account Number.</span>
</span>
<p>Password</p>
<span id="sprypassword1" style="text-align:left;">
<input type="password" name="pass" placeholder="Enter Password"tabindex="20" size="30"/>
<span class="passwordRequiredMsg">Password is required.</span>
<span class="passwordMinCharsMsg">You must specify at least 6 characters.</span>
<span class="passwordMaxCharsMsg">You must specify at max 10 characters.</span>
<input type="submit" name="" value="Login">
<a href="<?php echo WEB_ROOT; ?>register.php">Don't have an account?</a><br>
<a href="<?php echo WEB_ROOT; ?>admin/login.php">Admin Login..!</a>
</form>

</div>
<p>&nbsp;</p>
</body>
</head>
</html>

```

4.5 SQL Queries:

4.5.1. Sql queries in functions.php

```

<?php
require_once('mail.php');

/*
    Check if a session user id exist or not. If not set redirect
    to login page. If the user session id exist and there's found
    $_GET['logout'] in the query string logout the user
*/
function checkUser()
{
    // if the session id is not set, redirect to login page
    if (!isset($_SESSION['hlbank_user'])) {

```

```
        header('Location: ' . WEB_ROOT . 'login.php');
        exit;
    }
    // the user want to logout
    if (isset($_GET['logout'])) {
        doLogout();
    }
}

function next_tx_no() {
    $sql = "SELECT tx_no FROM tbl_transaction ORDER BY id DESC LIMIT 1";
    $result = dbQuery($sql);
    extract(dbFetchAssoc($result));
    $tx_num = (int)substr($tx_no, 2);
    $next_id = $tx_num+1; // increment by One
    return "TX".$next_id;
}

function str_number($str) {
    $number = "";
    $number = str_replace('$', "", $str);
    $number = str_replace(',', "", $number);
    return doubleval($number);
}

function doPinValidation() {
    $errorMessage = "";

    $pin = $_SESSION['hlbank_tmp']['pin'];
    $ipPin = $_POST['accpin'];

    if($pin == $ipPin) {
        $_SESSION['hlbank_user'] = $_SESSION['hlbank_tmp'];
        unset($_SESSION['hlbank_tmp']);
        header('Location: index.php');
        exit;
    }
    else {
        $errorMessage = 'Invalid pin numbers, please try again.';
    }
    return $errorMessage;
}

/*

*/

function doLogin()
```



```
{

    $errorMessage = "";

    $accno  = (int)$_POST['accno'];
    $pwd    = $_POST['pass'];

    $sql = "SELECT u.fname, u.lname, u.email, u.is_active, u.pics, u.phone,
                a.acc_no, a.user_id, a.pin, a.type, a.status,
                ad.address, ad.city, ad.state, ad.zipcode
            FROM tbl_users u, tbl_accounts a, tbl_address ad
            WHERE a.acc_no = $accno AND u.pwd = PASSWORD('$pwd')
            AND u.id = a.user_id AND ad.user_id = u.id AND u.is_active != 'FALSE'";

    $result = dbQuery($sql);

    if (dbNumRows($result) == 1) {
        $row = dbFetchAssoc($result);
        $_SESSION['hlbank_tmp'] = $row;
        $_SESSION['hlbank_user_name'] = strtoupper( $row['fname'].' '.$row['lname']);
        header('Location: pin.php');
        exit;
    }
    else {
        $errorMessage = 'Not valid account number or password or Account is not Active';
    }
    return $errorMessage;
}

/*
    Logout a user
*/
function doLogout()
{
    if (isset($_SESSION['hlbank_user'])) {
        unset($_SESSION['hlbank_user']);
        //session_unregister('hlbank_user');
    }
    header('Location: login.php');
    exit;
}

function doRegister()
{
    $fname = $_POST['firstname'];
    $lname = $_POST['lastname'];
    $pwd   = $_POST['password'];
}
```

```

$email = $_POST['email'];
$phone = $_POST['phone'];
$dob = $_POST['dob'];

$gender = $_POST['gender'];
$add = $_POST['address'];
$city = $_POST['city'];
$state = $_POST['state'];
$zip = (int)$_POST['zipcode'];

// $accno = (int)$_POST['accno'];
$type = $_POST['acctype'];
$pin = (int)$_POST['pin'];

$errorMessage = "";

$sql = "SELECT fname FROM tbl_users WHERE fname = '$fname'";
$result = dbQuery($sql);
if (dbNumRows($result) == 1) {
    $errorMessage = 'Username is already exist, please try another name.';
    return $errorMessage;
}

//first check if account number is already register or not...
$accno = rand(9999999999, 99999999999);
$accno = strlen($accno) != 10 ? substr($accno, 0, 10) : $accno;
/*
$sql = "SELECT acc_no FROM tbl_accounts WHERE acc_no = $accno";
$result = dbQuery($sql);
if (dbNumRows($result) == 1) {
    $errorMessage = 'Account number is already register.';
    return $errorMessage;
}
*/

$images = uploadProductImage('pic', SRV_ROOT . 'images/thumbnails/');
$thumbnail = $images['thumbnail'];
$insert_id = 0;
$sql = "INSERT INTO tbl_users (fname, lname, pwd, email, phone, gender, is_active, utype, pics, bdate)
VALUES ('$fname', '$lname', PASSWORD('$pwd'), '$email', '$phone', '$gender',
'FALSE', 'USER', '$thumbnail', NOW())";
dbQuery($sql);
$insert_id = dbInsertId();

//now create a user address.
$sql = "INSERT INTO tbl_address (user_id, address, city, state, zipcode, country)
VALUES ($insert_id, '$add', '$city', '$state', $zip, 'USA')";

```

```
dbQuery($sql);

//and now create a account table entry...
$sql = "INSERT INTO tbl_accounts (user_id, acc_no, type,pin, status, bdate)
        VALUES ($insert_id, $accno, '$type',$pin, 'INACTIVE', NOW())";
dbQuery($sql);

//now send email
//email it now.
$subject = "Account Registration";
$to = $email;
$msg_body = "Dear Customer,<br/><br/>
This is to inform you that your Account # $accno is register successfully with Hong Leong Bank and
currently in Inactive state. We will soon contact you once it get activate.<br/><br/>In case you need any further
clarification for the same, please do get in touch with your Home Branch.<br/><br/>
Regards,<br/>Hong Leong Bank";

$mail_data = array('to' => $to, 'sub' => $subject, 'msg' => 'register', 'body' => $msg_body);
send_email($mail_data);

header('Location: aregister.php');
exit;

}

/*
Create a thumbnail of $srcFile and save it to $destFile.
The thumbnail will be $width pixels.
*/
function createThumbnail($srcFile, $destFile, $width, $quality = 75)
{
    $thumbnail = "";

    if (file_exists($srcFile) && isset($destFile))
    {
        $size    = getimagesize($srcFile);
        $w       = number_format($width, 0, ',', '');
        $h       = number_format(($size[1] / $size[0]) * $width, 0, ',', '');

        $thumbnail = copyImage($srcFile, $destFile, $w, $h, $quality);
    }

    // return the thumbnail file name on sucess or blank on fail
    return basename($thumbnail);
}

/*
```

Copy an image to a destination file. The destination
image size will be \$w X \$h pixels

*/

```
function copyImage($srcFile, $destFile, $w, $h, $quality = 75)
{
    $tmpSrc    = pathinfo(strtolower($srcFile));
    $tmpDest   = pathinfo(strtolower($destFile));
    $size      = getimagesize($srcFile);

    if ($tmpDest['extension'] == "gif" || $tmpDest['extension'] == "jpg")
    {
        $destFile = substr_replace($destFile, 'jpg', -3);
        $dest     = imagecreatetruecolor($w, $h);
        imageantialias($dest, TRUE);
    } elseif ($tmpDest['extension'] == "png") {
        $dest = imagecreatetruecolor($w, $h);
        imageantialias($dest, TRUE);
    } else {
        return false;
    }

    switch($size[2])
    {
        case 1:    //GIF
            $src = imagecreatefromgif($srcFile);
            break;
        case 2:    //JPEG
            $src = imagecreatefromjpeg($srcFile);
            break;
        case 3:    //PNG
            $src = imagecreatefrompng($srcFile);
            break;
        default:
            return false;
            break;
    }

    imagecopyresampled($dest, $src, 0, 0, 0, 0, $w, $h, $size[0], $size[1]);

    switch($size[2])
    {
        case 1:
        case 2:
            imagejpeg($dest, $destFile, $quality);
            break;
        case 3:
            imagepng($dest, $destFile);
    }
}
```

```

    }
    return $destFile;

}

/*
    Create the paging links
*/
function getPagingNav($sql, $pageNum, $rowsPerPage, $queryString = "")
{
    $result = mysql_query($sql) or die('Error, query failed. ' . mysql_error());
    $row = mysql_fetch_array($result, MYSQL_ASSOC);
    $numrows = $row['numrows'];

    // how many pages we have when using paging?
    $maxPage = ceil($numrows/$rowsPerPage);

    $self = $_SERVER['PHP_SELF'];

    // creating 'previous' and 'next' link
    // plus 'first page' and 'last page' link

    // print 'previous' link only if we're not
    // on page one
    if ($pageNum > 1)
    {
        $page = $pageNum - 1;
        $prev = " <a href=\"\$self?page=$page{$queryString}\">[Prev]</a> ";

        $first = " <a href=\"\$self?page=1{$queryString}\">[First Page]</a> ";
    }
    else
    {
        $prev = ' [Prev] '; // we're on page one, don't enable 'previous' link
        $first = ' [First Page] '; // nor 'first page' link
    }

    // print 'next' link only if we're not
    // on the last page
    if ($pageNum < $maxPage)
    {
        $page = $pageNum + 1;
        $next = " <a href=\"\$self?page=$page{$queryString}\">[Next]</a> ";

        $last = " <a href=\"\$self?page=$maxPage{$queryString}\">[Last Page]</a> ";
    }
    else

```

```

    {
        $next = ' [Next] ';    // we're on the last page, don't enable 'next' link
        $last = ' [Last Page] '; // nor 'last page' link
    }

    // return the page navigation link
    return $first . $prev . " Showing page <strong>$pageNum</strong> of <strong>$maxPage</strong> pages
    " . $next . $last;
}

/*
    Upload an image and return the uploaded image name
*/
function uploadProductImage($inputName, $uploadDir)
{
    $image = $_FILES[$inputName];
    $imagePath = "";
    $thumbnailPath = "";

    // if a file is given
    if (trim($image['tmp_name']) != "") {
        $ext = substr(strrchr($image['name'], "."), 1); // $extensions[$image['type']];

        // generate a random new file name to avoid name conflict
        $imagePath = md5(rand() * time()) . ".$ext";

        list($width, $height, $type, $attr) = getimagesize($image['tmp_name']);

        // make sure the image width does not exceed the
        // maximum allowed width

        if (LIMIT_USER_WIDTH && $width > MAX_USER_IMAGE_WIDTH) {
            $result = createThumbnail($image['tmp_name'], $uploadDir . $imagePath,
MAX_USER_IMAGE_WIDTH);
            $imagePath = $result;
        } else {
            $result = move_uploaded_file($image['tmp_name'], $uploadDir . $imagePath);
        }

        if ($result) {
            // create thumbnail
            $thumbnailPath = md5(rand() * time()) . ".$ext";
            $result = createThumbnail($uploadDir . $imagePath, $uploadDir . $thumbnailPath,
THUMBNAIL_WIDTH);

            // create thumbnail failed, delete the image

```

```
        if (!$result) {
            unlink($uploadDir . $imagePath);
            $imagePath = $thumbnailPath = "";
        } else {
            $thumbnailPath = $result;
        }
    } else {
        // the product cannot be upload / resized
        $imagePath = $thumbnailPath = "";
    }
}

return array('image' => $imagePath, 'thumbnail' => $thumbnailPath);
}

?>
```

Chapter 5

TESTING

5.1 Conclusion:

It has been a great pleasure for us to work on this project. We hereby like to state that this application provides very good platform to everyone who wants a online platform for online transaction , better platform for bank related services . The admin has the privilege to activate the account and user and also delete the user from the database i.e., insertion, deletion, updation, of users details and etc.

5.2 Future enhancement:

- To provide much more services on this platform.
- Provide Security for the transaction's and the account details.
- Much more reliable services for the users.

5.3 Snapshots:

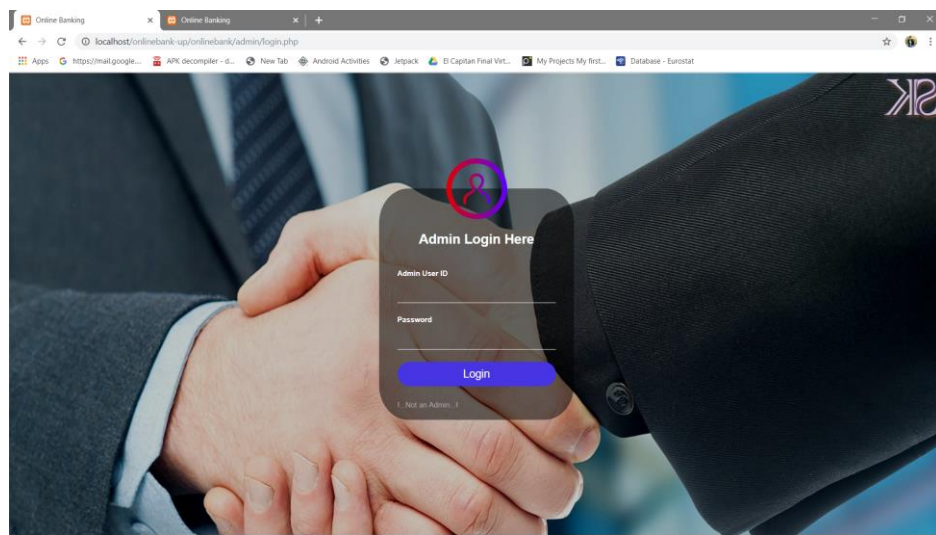


Fig.5.3.1 : Admin Login

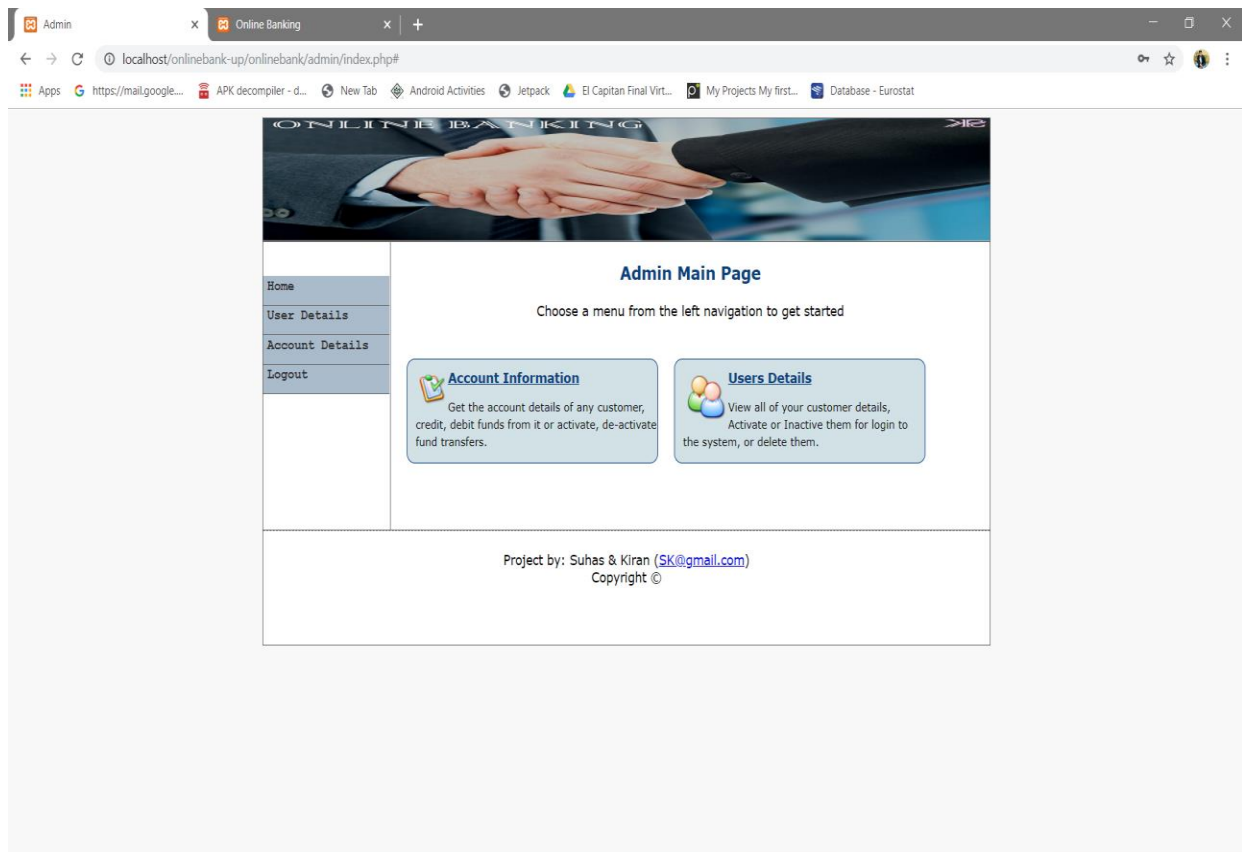


Fig.5.3.2 : Admin Main Page

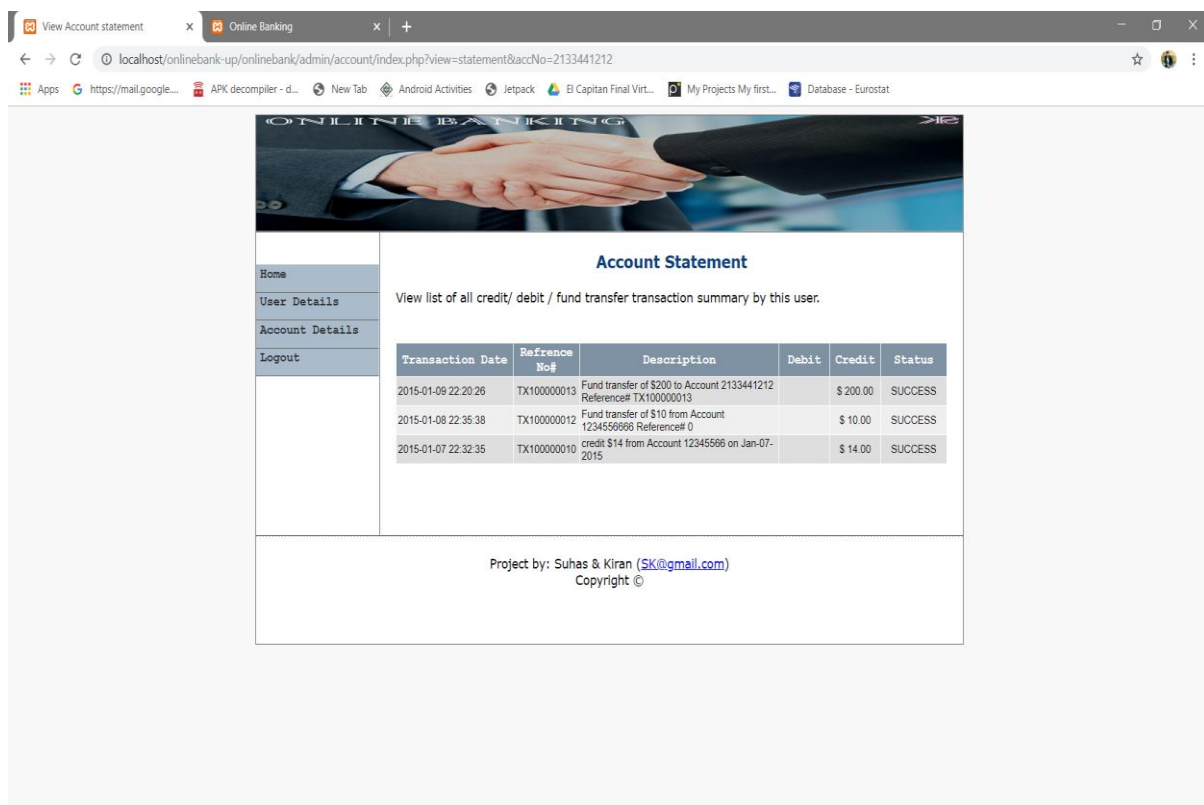


Fig.5.3.3 : User's account transaction through Admin

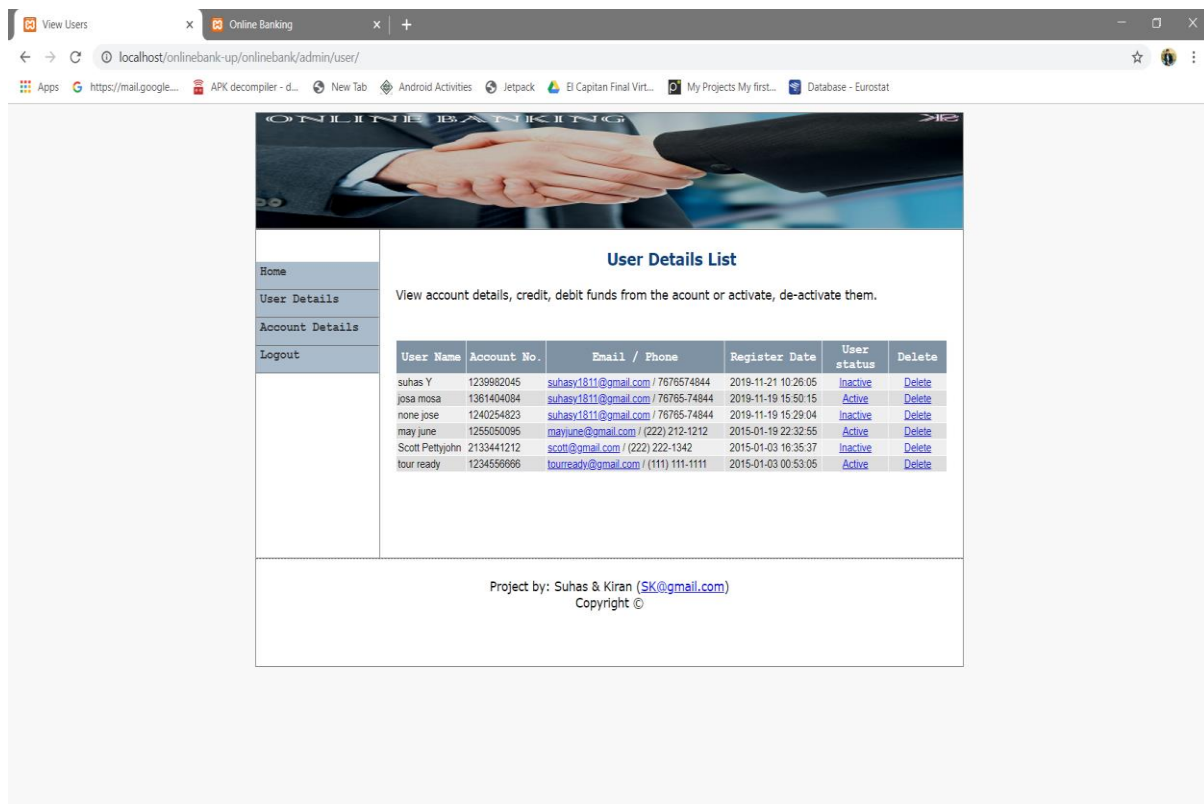


Fig.5.3.4 : Users Details list in Admin portal

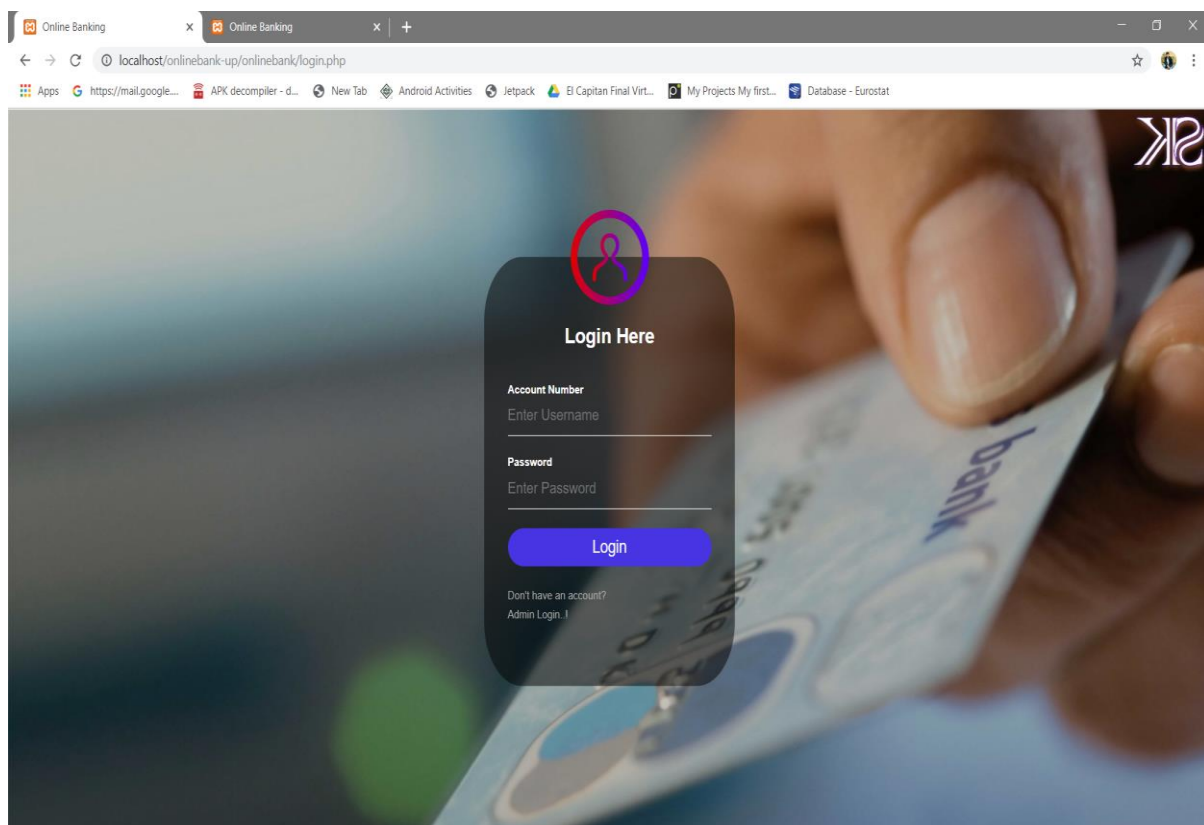


Fig.5.3.5 : Users Login Page

Register Account:

Please register your account with us to take the benefits of our Online Banking facilities.

Personal Information

First Name

Last Name

Password

Confirm Password

Email ID

Phone Number is XXXXX-XXXXX

Date of Birth

Profile Pics No file chosen

Gender

Address Information

Address

Fig.5.3.6 : User's Registration Page

Welcome, JOSIA MOSA

Account Statements | **Account Details** | Fund Transfer

User Account Details

If you feel that you have a weaker strength password, then please change it. We recommend to change your password in every 45 days to make it secure.

User Account Details	
User Fullname	JOSIA MOSA
Email ID	suhasy1811@gmail.com
Phone Number	76765-74844
Address	778
City, State	mysore, karnataka
Zip Code	570019
Account Number	1361404084
Account Balance	2,000 \$
Account PIN Code	1811

Security Settings

Change Password
Change PIN
Sign Out

Project by: Suhas & Kiran (SK@gmail.com)
Copyright ©

Fig.5.3.7 : Users Account Details

The screenshot shows a web browser window with the URL `localhost/onlinebank-up/onlinebank/view/?v=Summary`. The page features a header with the text "ONLINE BANKING" and a background image of two hands shaking. On the left, there is a sidebar menu with sections: "Types of Scholarships" (listing 5 items), "Account Details" (with links to Account Summary, Account Details, Account Statement, and Fund Transfers), and "Security Settings" (with links to Change Password, Change PIN, and Sign Out). The main content area is titled "Welcome, JOSIA MOSA" and includes tabs for "Account Statements", "Account Details", and "Fund Transfer". The "Fund Transfer" tab is active, displaying a "Funds Transfer" section with a description: "Funds transfer is a process of transferring funds from your account to other account in same Bank. Please make sure that you have enough funds available in your account to transfer. Also don't forget to validate receiver's account number." Below this is a "Transfer Funds" form with fields for: Receiver's Bank Name, Receiver's Name, Receiver's Account Number, SWIFT/ABA Routing Number, Sender's Account Number (pre-filled with 1361404084), Amount to Transfer (in RS), Fund Transfer Option (a dropdown menu), Date of Transfer, and Transfer Description. A "Fund Transfers" button is at the bottom of the form. The footer of the page states "Project by: Suhas & Kiran (SK@gmail.com) Copyright ©".

Fig.5.3.8 : Funds transfer Portal

The screenshot shows the same web browser window, but the "Account Statements" tab is active. The main content area displays "Welcome, JOSIA MOSA" and "Account Statements". Below this, it says "View list of all credit/ debit / fund transfer transaction summary by this user." and shows a table with the following structure:

Transaction Date	Reference No#	Description	Debit	Credit
You have no transaction history yet, seems that you haven't done any transaction yet.				

Below the table, it states "Available Credit Balance: 2,000.00". The sidebar menu and footer are identical to the previous screenshot.

Fig.5.3.9 : Users Account Statement Portal

REFERENCES

1. Wikipedia (<https://en.wikipedia.org/wiki/XAMPP>).
2. Fundamentals of DataBase Systems -Ramez Navathe.
3. Stack Overflow