**ONLINE SLAM BOOK**



Submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

In

Computer Science and Engineering

by

**K SHARAN KUMAR - 21K91A05C3**

**K MADHUJA - 21K91A05C2**

**K SAI MANISH - 21K91A05B0**

**G DATHA SAI - 21K91A0577**

**G ADITHYA - 21K91A0575**

**Under the guidance of**

Mrs K NAGA MAHA LAKSHMI

**ASSISTANT PROFESSOR**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

TKR COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

(ACCREDITED BY NBA AND NAAC WITH ’A’ GRADE)

**Medbowli, Meerpet, Balapur(M), Hyderabad-500097**

**DECLARATION BY THE CANDIDATE**

I, Mr.K.SHARAN KUMAR bearing Hall Ticket Number:21K91A05C3,Ms.K.MADHUJA bearing Hall Ticket Number:21K91A05C2,Mr.K.SAI MANISH bearing Hall Ticket:21K91A05B0,G DATHA SAI bearing Hall Ticket:21K91A0577,Mr.G ADITHYA bearing Hall Ticket:21K91A0575 , hereby declare that the main project report titled ONLINE SLAM BOOK under the guidance of Mrs.K.NAGA MAHA LAKSHMI, Assistant professor in Department of Computer Science and Engineering is submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering.

Signature of the candidate

Roll number

Place : Meerpet

Date :

**CERTIFICATE**

This is to certify that the main project report entitled ONLINE SLAM BOOK , being submitted by Mr./Ms. K SHARAN KUMAR ,K MADHUJA,K SAI MANISH,G DATHA SAI,G ADITHYA , ROLL.NO:21K91A05C3,21K91A05C2,21K91A05B0, 21K91A0577,21K91A0575 in partial fulfillment of requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering, to the TKR College of Engineering and Technology is a record of bonafide work carried out by him/her under my guidance and supervision.

Name and signature of the HOD

Name and Signature of the Guide

Place: Meerpet

Date :

**TABLE OF CONTENTS**

**ABSTRACT 5**

**ACKNOWLEDGEMENTS 6**

1. **INTRODUCTION 7**

Motivation

Problem definition

1. **REQUIREMENT ANALYSIS 8**

Functional Requirements

Non-Functional Requirements

1. **DESIGN 10**

DFDs and UML diagrams

1. **CODING 14**

Pseudo Code (I.e, sample code to create database, the code relevant connect Backend to frontend)

1. **IMPLEMENTATION and RESULTS 31**

Method of Implementation

Output Screens

**6 TESTING AND VALIDATION 35**

**7 CONCLUSION 39**

**REFERENCES 40**

ONLINE SLAM BOOK

**ABSTRACT**

An online slam book is a digital platform that modernizes the traditional slam book concept, allowing users to create, share, and view personal entries in an interactive and user-friendly environment. This system leverages web technologies to facilitate seamless interactions between users, providing a space for them to fill out and share entries that capture their thoughts, experiences, and personal reflections.

The primary objective of the online slam book is to offer a secure and accessible medium for users to express themselves and connect with others. Users can register and log in to the platform, create personalized slam book entries, and browse entries created by others. The system ensures data integrity and security through robust authentication and data management processes.

TEAM MEMBERS

K SHARAN KUMAR (LEAD) ------------------21K91A05C3

K MADHUJA ------------------------------------21K91A05C2

K SAI MANISH----------------------------------21K91A05B0

G DATHA SAI------------------------------------21K91A0577

G ADITHYA--------------------------------------21K91A0575

GUIDE:

(Mrs. K.Naga Maha Lakshmi)

**ACKNOWLEDGEMENTS**

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragement and guidance have crowned my efforts with success.

I am indebted to the Internal Guide, Mrs. K. Naga Maha Lakshmi, Assistant Professor, Dept. of Computer Science and Engineering, TKR College of Engineering and Technology, for his/her support and guidance throughout my Thesis/Dissertation.

I am also indebted to the Head of the Department, Dr. A. Suresh Rao, Professor, Computer Science and Engineering, TKR College of Engineering and Technology, for his support and guidance throughout my Thesis/Dissertation.

I extend my deep sense of gratitude to the Principal, Dr. D. V. Ravi Shankar, TKR College of Engineering and Technology, for permitting me to undertake this Thesis/Dissertation.

Finally, I express my thanks to one and all that have helped me in successfully completing this Thesis/Dissertation. Furthermore, I would like to thank my family and friends for their moral support and encouragement

NAME OF THE STUDENT

K SHARAN KUMAR 21K91A0573

K MADHUJA 21K91A05C2

K SAI MANISH 21K91A05B0

G DATHA SAI 21K91A0577

G ADITHYA 21K91A0575

Place: Meerpet

Date:

INTRODUCTION

The Online Slam Book is a web-based application that brings the cherished tradition of slam books into the digital age. Historically, slam books have been a popular means for friends to share personal thoughts, memories, and reflections in a fun and interactive way. This project aims to capture that essence by providing a platform where users can create, view, and manage their own slam book entries online.

The primary goal of this project is to offer a secure, user-friendly environment that facilitates the creation and sharing of personal entries. By leveraging modern web technologies, the Online Slam Book enhances the traditional experience with added convenience and accessibility, ensuring users can connect and share their thoughts from anywhere.

**Key Objectives:**

* **User Authentication**: Secure registration and login mechanisms to protect user data.
* **Entry Creation**: A simple interface for users to create and personalize their slambook entries.
* **Viewing Entries**: An easy-to-navigate system for browsing entries created by friends and other users.
* **Data Security**: Robust measures to ensure the privacy and security of user data.

The Online Slam Book is designed to preserve the nostalgic charm of traditional slam books while incorporating the benefits of digital technology. Whether users are reminiscing about the past or creating new memories, this platform provides a modern, interactive space for documenting and sharing personal experiences.

**REQUIREMENT ANALYSIS**

**FUNCTIONAL REQUIREMENTS:**

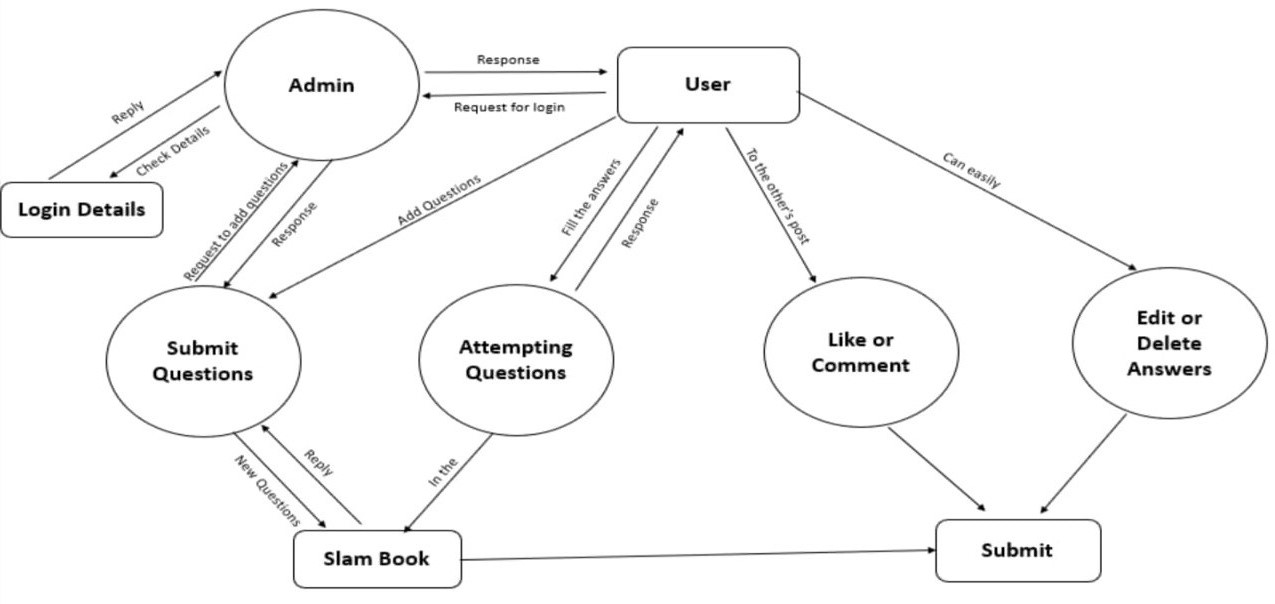
* **HARDWARE REQUIREMENTS**:
* Processor: intel i3
* Speed:1.1GHZ
* RAM:8GB
* Hard disk:500GB
* Keyboard: standard windows keyboard
* Mouse:2 or 3 button mouse
* **SOFTWARE REQUIREMENTS**:
* Operating system: windows(any version)
* Server side Technologies : PHP
* Client side Technologies : HTML, CSS, JavaScript
* Database: MYSQL

**NON FUNCTIONAL REQUIREMENTS:**

1. **Data Integrity** **:** Various Error checking and Error detection methods have been applied in order to maintain integrity of data.
2. **Performance :** Performance of the system has been tuned for stable, fast, efficient and reliable experience.
3. **Optimisation :** The system is optimized for devices of all screen size like Mobile, Tablet and Computer in order to maintain similar user experience across all devices.

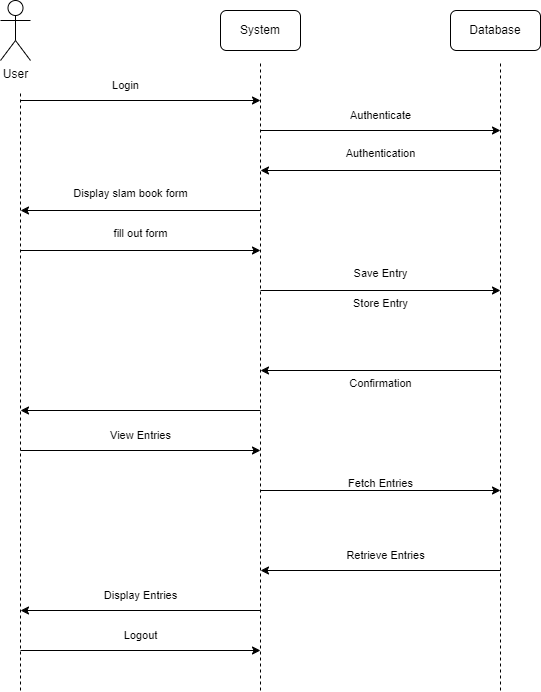
**DESIGN**

**DATA FLOW DIAGRAM**

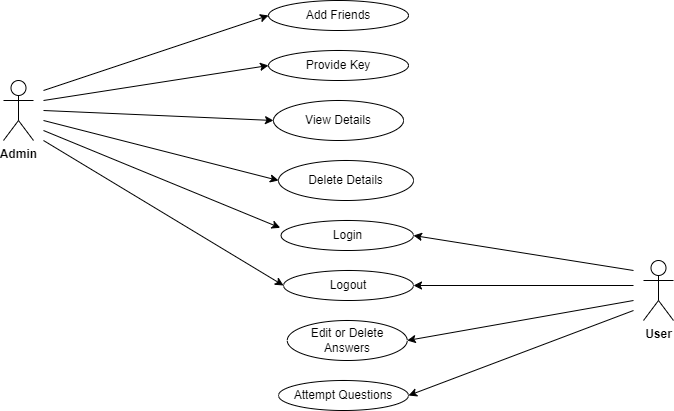
****

**UML DIAGRAMS:**

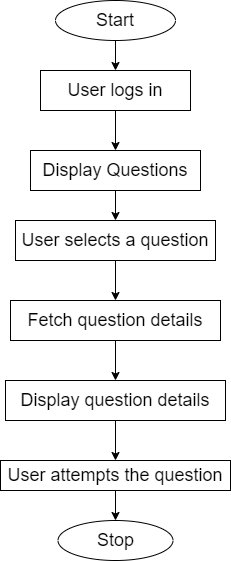
**SEQUENCE DIAGRAM:**

****

**USE CASE DIAGRAM:**

****

**ACTIVITY DIAGRAM:**

****

**CODING**

**INDEX.PHP:**

<?php

    session\_start();

    require\_once 'config/dbcon.php';

    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Admin Login \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

    if (!empty($\_POST['login']))

    {

        $uname = $\_POST['uname'];

        $passwd = $\_POST['passwd'];

        $sqlselect = "SELECT \* FROM admin WHERE uname = '$uname' and passwd = '$passwd' ";

        $sql = mysqli\_query($conn,$sqlselect);

        $rows = mysqli\_fetch\_assoc($sql);

        //print\_r($rows);

        if (($rows['uname'] == $uname) AND $rows['passwd'] == $passwd)

        {

            $\_SESSION['uname'] = $uname;

            header("Location: admin/");

        }

        else

        {

            echo "<script type='text/javascript'>alert('Invalid Login Credentials');</script>";

        }

    }

    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Admin Login End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Friend's Login \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

    if (!isset($\_POST['flogin'])) {

        $sqlselect = "SELECT \* FROM fname";

        $sql = mysqli\_query($conn,$sqlselect);

    } else

    {

        $fname = $\_POST['fname'];

        $lkey = $\_POST['lkey'];

        $sqlselect = "SELECT \* FROM fname WHERE fname = '$fname' AND lkey = '$lkey' ";

        $sql = mysqli\_query($conn,$sqlselect);

        $rows = mysqli\_fetch\_assoc($sql);

        if (($rows['fname'] == $fname) AND ($rows['lkey'] == $lkey)) {

            $\_SESSION['fname'] = $fname;

            header("Location: friend/");

        }else

        {

            echo "<script type='text/javascript'>alert('Invalid Key Entered!');</script>";

        }

    }

    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Friend's Login End \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

?>

<!DOCTYPE html>

<html>

<head>

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">

    <title>My SlamBook</title>

    <link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.min.css">

    <link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.css">

    <link rel="stylesheet" type="text/css" href="assets/style.css">

</head>

<body>

    <div class="container">

        <div class="text-capitalize text-center" >

            <h1><strong>Welcome To My SlamBook</strong></h1>

        </div>

        <div class="row">

            <div class="col-md-6" style="padding-top:50px">

                <h3 class="text-center">Friend's Login</h3>

                <br /><br /><br />

                <form action="" method="POST" class="form-group">

                    <div class="form-group">

                        <label>Friend's Name</label>

                        <select class="form-control" name="fname">

                            <option>Select Your Name</option>

                            <?php

                                while ($rows = mysqli\_fetch\_assoc($sql)) {

                                    arsort($rows);

                            ?>

                            <option><?php echo $rows['fname']; ?></option>

                            <?php } ?>

                        </select>

                    </div>

                    <div class="form-group">

                        <label >Your Key</label>

                        <input type="number" class="form-control" name="lkey" placeholder="Unique Key">

                    </div>

                    <div class="form-group">

                        <input type="submit" name="flogin" value="Proceed" class="btn btn-success">

                        <!--button type="button" class="btn btn-success" name="login">Login</button-->

                    </div>

                </form>

            </div>

            <div class="col-md-6" style="padding-top:50px">

                <h3 class="text-center">Admin Login</h3>

                <br /><br /><br />

                <form action="" method="POST" class="form-group">

                    <div class="form-group">

                        <label class="col-sm-2 control-label">Username</label>

                        <input type="text" class="form-control" name="uname" placeholder="Username" value="admin" readonly>

                    </div>

                    <div class="form-group">

                        <label class="col-sm-2 control-label">Password</label>

                        <input type="password" class="form-control" name="passwd" placeholder="Username">

                    </div>

                    <div class="form-group">

                        <input type="submit" name="login" value="login" class="btn btn-success">

                        <!--button type="button" class="btn btn-success" name="login">Login</button-->

                    </div>

                </form>

            </div>

        </div>

    </div>

</body>

</html>

**FRIEND**

**INDEX.PHP:**

<?php

    session\_start();

    require\_once '../config/dbcon.php';

    require\_once 'pic.php';

    if ($\_SESSION['fname']) {

        $fname = $\_SESSION['fname'];

        /\*\*\*\*\*\*\* To accept mobile number \*\*\*\*\*\*\*\*\*/

        /\*

        $sqlselect = "SELECT \* FROM fname WHERE fname = '$fname' ";

        $sql = mysqli\_query($conn,$sqlselect);

        $rows = mysqli\_fetch\_assoc($sql);

        \*/

        $finfo\_select = "SELECT \* FROM finfo WHERE fname = '$fname' ";

        $finfo\_sql = mysqli\_query($conn,$finfo\_select);

        $finfo\_rows = mysqli\_fetch\_assoc($finfo\_sql);

        /\*

        $profilepic = $\_SESSION['fname'].'.jpeg';

        if (!file\_exists("/upload/$profilepic")) {

        }

\*/

        require\_once 'finfo\_update\_query.php';

    }

    else

    {

        header("Location: ../");

    }

?>

<!DOCTYPE html>

<html>

<head>

    <title>Hello <?php echo $\_SESSION['fname']; ?> </title>

    <?php require\_once 'bootstrap\_links.php'; ?>

    <!-- <?php require\_once 'navbar.php'; ?> -->

    <?php require\_once 'pic.php'; ?>

    <?php require\_once 'profile\_pic.php'; ?>

</head>

<body>

    <?php require\_once 'sample.php'; ?>

    <div class="container">

        <div class="card">

        <div class="row">

            <div class="col-md-4" >

                <div class="card">

                    <img src="upload/<?php echo $\_SESSION['fname'].'.jpeg'; ?>" class="img-responsive" alt="<?php echo $\_SESSION['fname']; ?>" style="max-height:400px">

                    <div class="card-block">

                        <h4 class="card-title"><b><?php echo $\_SESSION['fname']; ?></b></h4>

                        <!-- The data encoding type, enctype, MUST be specified as below -->

                        <form enctype="multipart/form-data" action=" " method="POST">              <input name="userfile" type="file" />

                        <input type="submit" value="Send File" name="submit" class="btn btn-primary" />

                        </form>

                    </div>

                </div>

            </div>

            <br />

            <div class="col-md-8">

                    <?php require\_once 'finfo\_form.php'; ?>

                    <br /><br /><br />

            </div>

        </div>

        </div>

    </div>

</body>

</html>

**PROFILE\_PIC.PHP:**

<?php

$user=$\_SESSION['user'];

$db=new mysqli('localhost','root','!@#$','ws');

if($db->connect\_errno){

echo $db->connect\_error;}

$pull="select \* from userImage  where user='$user'";

$allowedExts = array("jpg", "jpeg", "gif", "png","JPG");

$extension = @end(explode(".", $\_FILES["file"]["name"]));

if(isset($\_POST['pupload'])){

if ((($\_FILES["file"]["type"] == "image/gif")

|| ($\_FILES["file"]["type"] == "image/jpeg")

|| ($\_FILES["file"]["type"] == "image/JPG")

|| ($\_FILES["file"]["type"] == "image/png")

|| ($\_FILES["file"]["type"] == "image/pjpeg"))

&& ($\_FILES["file"]["size"] < 200000)

&& in\_array($extension, $allowedExts))

  {

  if ($\_FILES["file"]["error"] > 0)

    {

    echo "Return Code: " . $\_FILES["file"]["error"] . "<br>";

    }

  else

    {

    echo '<div class="plus">';

    echo "Uploaded Successully";

    echo '</div>';

    echo"<br/><b><u>Image Details</u></b><br/>";

    echo "Name: " . $\_FILES["file"]["name"] . "<br/>";

    echo "Type: " . $\_FILES["file"]["type"] . "<br/>";

    echo "Size: " . ceil(($\_FILES["file"]["size"] / 1024)) . " KB";

    if (file\_exists("upload/" . $\_FILES["file"]["name"]))

      {

      unlink("upload/" . $\_FILES["file"]["name"]);

      }

    else

      {

          $pic=$\_FILES["file"]["name"];

            $conv=explode(".",$pic);

            $ext=$conv['1'];

      move\_uploaded\_file($\_FILES["file"]["tmp\_name"],

      "upload/". $user.".".$ext);

      echo "Stored in as: " . "upload/" . $user.".".$ext;

      $url=$user.".".$ext;

      $query="update userImage set url='$url', lastUpload=now() where user='$user'";

      if($upl=$db->query($query)){

          echo "<br/>Saved to Database successfully";

              }

      }

    }

  }

else

  {

  echo "File Size Limit Crossed 200 KB Use Picture Size less than 200 KB";

  }

}

?>

<form action="" method="post" enctype="multipart/form-data">

  <?php

    $res=$db->query($pull);

    $pics=$res->fetch\_assoc();

    echo '<div class="imgLow">';

    echo "<img src='upload/$pics[url]' alt='profile picture' width='80' height='64'   class='doubleborder'/></div>";

    ?>

    <input type="file" name="file" />

    <input type="submit" name="pupload" class="button" value="Upload"/>

</form>

<!--div class="card">

    <img src="../images/bk2.jpg" class="img-responsive" alt="<?php echo $\_SESSION['fname']; ?>">

    <div class="card-block">

      <h4 class="card-title"><b><?php echo $\_SESSION['fname']; ?></b></h4>

      <a href="#" class="btn btn-primary">Upload</a>

    </div>

</div-->

**ADMIN:**

**INDEX.PHP:**

<?php

  session\_start();

  require\_once '../config/dbcon.php';

  if ($\_SESSION['uname']) {

  }

  else

  {

    header("Location: ../ ");

  }

?>

<!DOCTYPE html>

<html>

<head>

  <title> Hey <?php echo $\_SESSION['uname']; ?></title>

  <?php require\_once 'links.php'; ?>

  <link rel="stylesheet" href="admin.css">

</head>

<body>

  <div class="container">

    <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Added \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

      <?php require\_once 'navbar.php'; ?>

    <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Ended \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

    <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* Add Friend's Card start \*\*\*\*\*\*\*\*\*-->

      <div class="row">

        <div class="col s12 m6">

          <div class="card">

            <div class="card-image">

              <img src="../images/friends1.jpg">

              <span class="card-title" style="color:ghostwhite">Add New Friend</span>

            </div>

            <div class="card-content">

              <p>The Greatest Gift Of Life Is Friendship,And I Have Recieved It.</p>

            </div>

            <div class="card-action">

              <a href="addfriend.php">Click Here To Add New Friend</a>

            </div>

          </div>

        </div>

      </div>

      <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* Add Friend's Card End \*\*\*\*\*\*\*\*\*\*\*\*\*-->

      <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Friend's Card start \*\*\*\*\*\*\*\*\*-->

        <div class="col s12 m6">

          <div class="card">

            <div class="card-image">

              <img src="../images/friends.jpg">

              <span class="card-title" style="color:deeppink"><b>View Friend's</b></span>

            </div>

            <div class="card-content">

              <p>Love is the only force capable of transforming an enemy into friend.</p>

            </div>

            <div class="card-action">

              <a href="viewfriend.php">Click Here To View Friend's</a>

            </div>

          </div>

        </div>

      <!-- </div> -->

      <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Friend's Card End \*\*\*\*\*\*\*\*\*-->

  </div>

</body>

</html>

**ADMIN:**

**ADMIN.CSS:**

/\* Apply styles to body \*/

body {

    font-family: Arial, sans-serif;

    background: url('../images/background.jpg') no-repeat center center fixed;

    background-size: cover;

    margin: 0;

    padding: 0;

    color: #333;

}

/\* Style the container \*/

.container {

    max-width: 800px;

    margin: 120px auto; /\* Adjust margin for spacing \*/

    padding: 20px;

    background-color: rgba(255, 255, 255, 0.8); /\* Semi-transparent background \*/

    box-shadow: 0 0 10px rgba(0,0,0,0.1);

    border-radius: 5px;

}

/\* Style the navbar \*/

nav {

    width: 100%;

    background-color: #333; /\* Dark background color \*/

    color: white;

    padding: 10px 0;

    position: fixed;

    top: 0;

    z-index: 1000;

}

nav ul {

    list-style-type: none;

    margin: 0;

    padding: 0;

    text-align: center;

}

nav ul li {

    display: inline-block;

    margin-right: 20px; /\* Adjust spacing between navbar items \*/

}

nav ul li a {

    color: white;

    text-decoration: none;

    padding: 10px 15px;

}

nav ul li a:hover {

    background-color: #555; /\* Darken background on hover \*/

    border-radius: 5px;

}

/\* Style the "hello admin" text \*/

.page-title {

    text-align: center;

    font-size: 2em;

    margin-top: 40px; /\* Adjust spacing from navbar \*/

}

/\* Style the card image and titles \*/

.card-image {

    position: relative;

    overflow: hidden;

    height: 400px; /\* Adjust height as needed \*/

}

.card-image img {

    width: 100%;

    height: auto;

}

.card-title {

    position: absolute;

    bottom: 0;

    left: 0;

    width: 100%;

    background-color: rgba(0,0,0,0.5);

    padding: 10px;

    color: white;

    font-size: 1.5em;

    text-align: center;

}

/\* Style card content and action \*/

.card-content {

    padding: 15px;

}

.card-action {

    background-color: #f0f0f0;

    padding: 10px;

    text-align: center;

}

.card-action a {

    color: #007bff;

    text-decoration: none;

}

.card-action a:hover {

    text-decoration: underline;

    color: #0056b3;

}

**VIEWFRIEND.PHP:**

<?php

    session\_start();

    require\_once '../config/dbcon.php';

    if ($\_SESSION['uname']) {

        $sqlselect = "SELECT \* FROM fname";

        $sql = mysqli\_query($conn,$sqlselect);

        //$row=mysqli\_fetch\_array($sql,MYSQLI\_ASSOC);

        //print\_r($row);

        //$rows = mysqli\_fetch\_assoc($sql);

        $id = 1;

    }

    else

    {

        header("Location: ../ ");

    }

?>

<!DOCTYPE html>

<html>

<head>

    <title>Hey Admin : View Friend's</title>

    <?php require\_once 'links.php'; ?>

    <link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.css">

<link rel="stylesheet" type="text/css" href="assets/style.css">

</head>

<body>

    <div class="container">

        <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Added \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

            <?php require\_once 'navbar.php'; ?>

        <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Ended \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

        <div class="col s12 m7">

            <!--h2 class="header">Your Friend's List</h2-->

            <div class="card horizontal">

                <div class="card-stacked">

                    <div class="card-content">

                        <h4 class="header">Your Friend's List</h4>

                    </div>

                    <div class="card-action">

                        <!--a href="#">This is a link</a-->

                    </div>

                </div>

            </div>

        </div>

        <table class="responsive-table centered highlight">

            <thead>

                <tr>

                    <th data-field='id'>Id</th>

                    <th data-field='name'>Name</th>

                    <th data-field='mobile'>Mobile</th>

                    <th data-field='key'>Key</th>

                    <th data-field='view'>View</th>

                    <th data-field='delete'>Delete</th>

                </tr>

            </thead>

            <tbody>

            <?php

                //array\_multisort(while($rows = mysqli\_fetch\_assoc($sql){}));

                while($rows = mysqli\_fetch\_assoc($sql)) {

            ?>

                <tr>

                    <td><?php echo $id; ?></td>

                    <td><?php echo $rows['fname']; ?></td>

                    <td><?php echo $rows['mobile']; ?></td>

                    <td><?php echo $rows['lkey']; ?></td>

                    <?php $\_SESSION['fname'] = $rows['fname'];  ?>

                    <!--td><a href="view.php?fname=<?php echo $rows['fname']; ?> & type=view ">view</a></td-->

                    <td><a href="view.php?fname=<?php echo $rows['fname'];?>">view</a></td>

                    <td><a href="del.php?id=<?php echo $rows['id'];?>">Delete</a></td>

                </tr>

            <?php

                $id = $id +1;

            } ?>

            </tbody>

        </table>

    </div>

</body>

</html>

**ADDFRIEND.PHP:**

<?php

    session\_start();

    require\_once '../config/dbcon.php';

    if ($\_SESSION['uname']) {

        if (isset($\_POST['submit'])) {

            $fname = $\_POST['fname'];

            $mobile = $\_POST['mobile'];

            $lkey = rand(11111 , 9999999999);

            $sqlinsert = "INSERT INTO fname (fname, mobile, lkey)

                                    VALUES ('$fname', '$mobile', '$lkey')";

            $finfo\_insert = "INSERT INTO finfo (fname, mobile)

                                    VALUES ('$fname', '$mobile')";

            $sql = mysqli\_query($conn,$sqlinsert);

            $finfo\_sql = mysqli\_query($conn,$finfo\_insert);

            if ($sql) {

                echo "<script type='text/javascript'>alert('Friend Added Sucessfully.');</script>";

            }

            else{

                echo "<script type='text/javascript'>alert('There is a problem to Adding friend.');</script>";

            }

        }

    }

    else

    {

        header("Location: ../ ");

    }

?>

<!DOCTYPE html>

<html>

<head>

    <title>Hey Admin : Add Friend</title>

    <?php require\_once 'links.php'; ?>

    <link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="assets/bootstrap/bootstrap.css">

<link rel="stylesheet" type="text/css" href="assets/style.css">

</head>

<body>

    <div class="container">

        <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Added \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

            <?php require\_once 'navbar.php'; ?>

        <!--\*\*\*\*\*\*\*\*\*\*\*\*\*\* Nav Bar Ended \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

        <div class="row">

            <form class="col s12" action="" method="POST">

            <br /> <br /><br /><br /><br />

              <div class="row">

                <div class="input-field col s5">

                  <input id="icon\_prefix" type="text" name="fname" class="validate" placeholder="Full Name">

                </div>

                <div class="input-field col s5">

                  <input id="icon\_telephone" type="tel" name="mobile" class="validate" placeholder="Mobile No.">

                </div>

                <div class="input-field col s2">

                  <input type="submit" name="submit" class="validate" value="Submit">

                </div>

              </div>

            </form>

        </div>

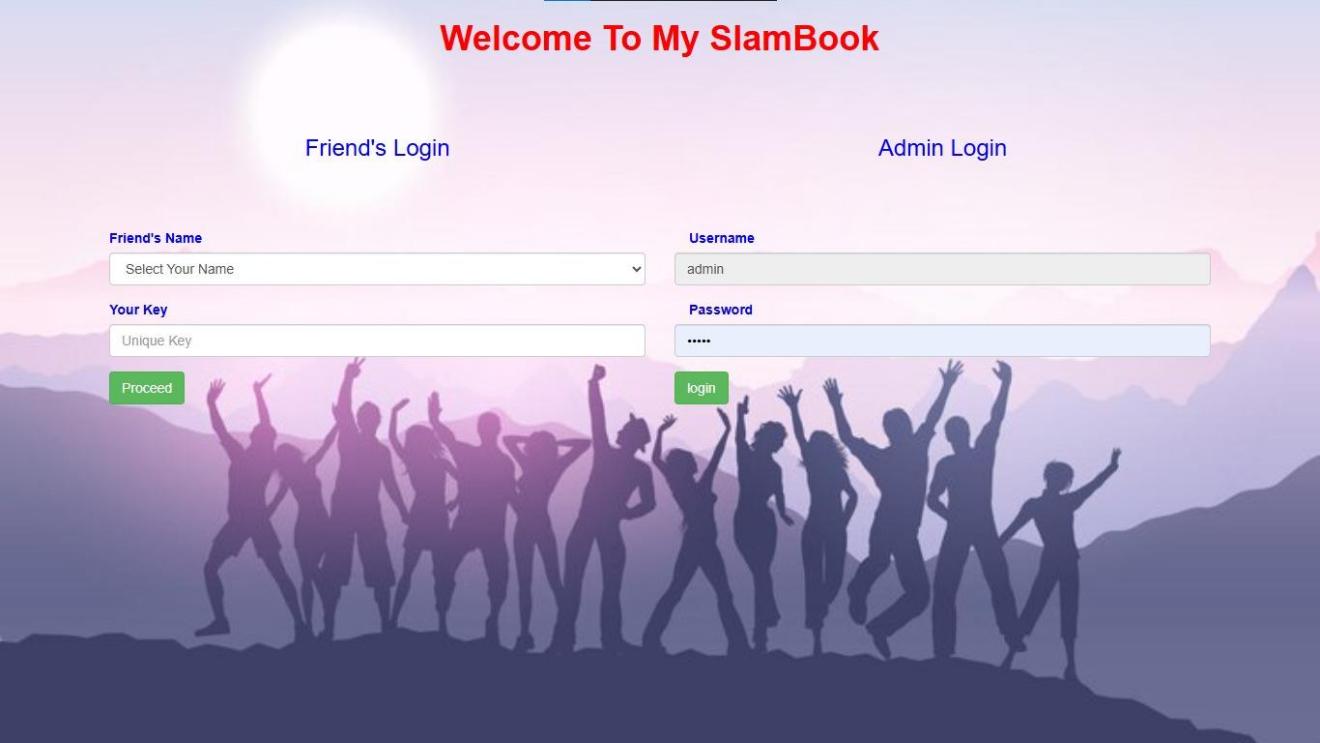
    </div>

</body>

</html>

**IMPLEMENTATION**

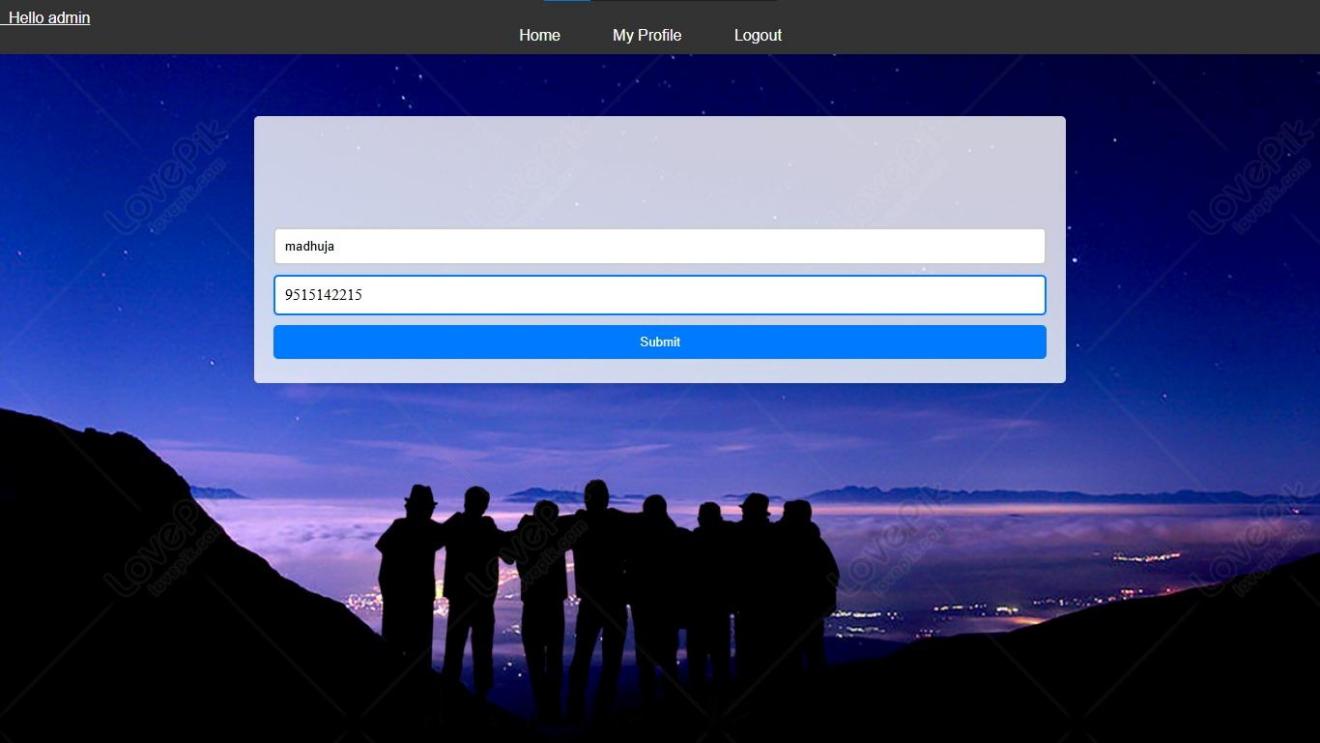
**LOGIN PAGE:**

****

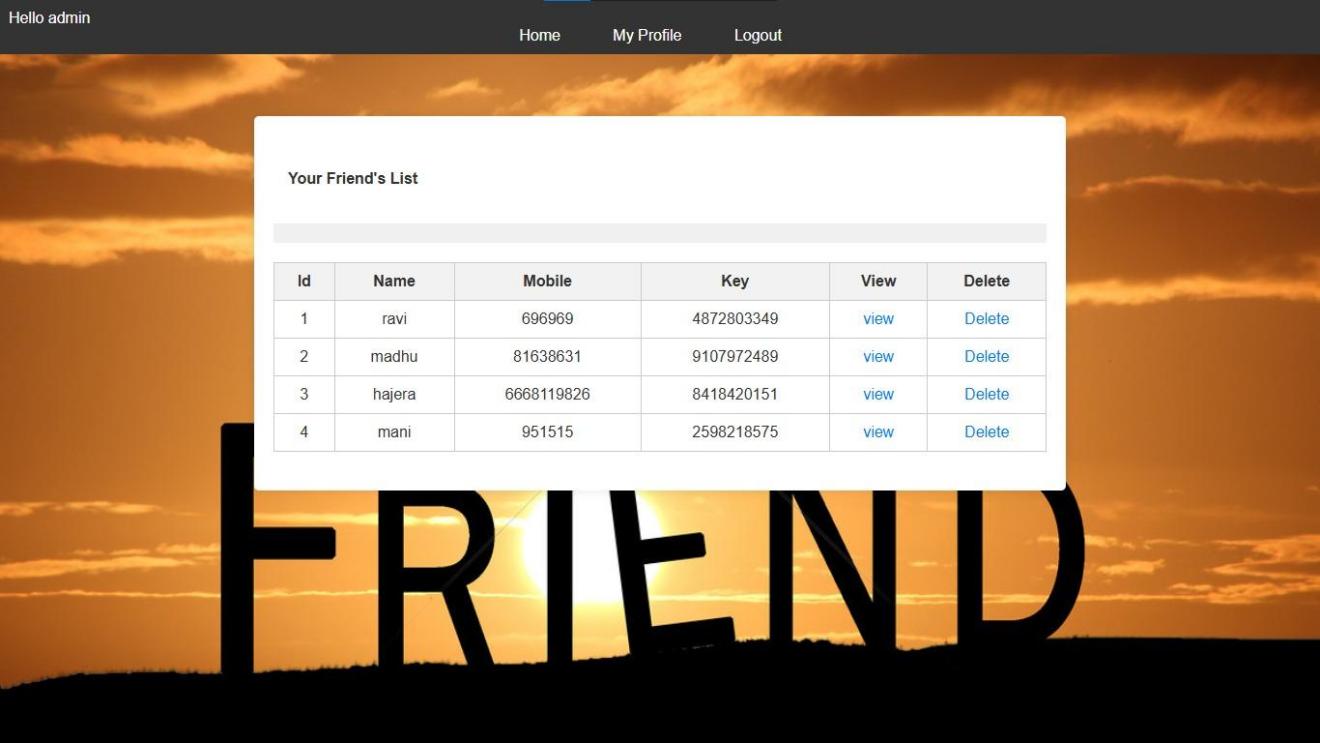
**ADMIN HOME PAGE:**

****

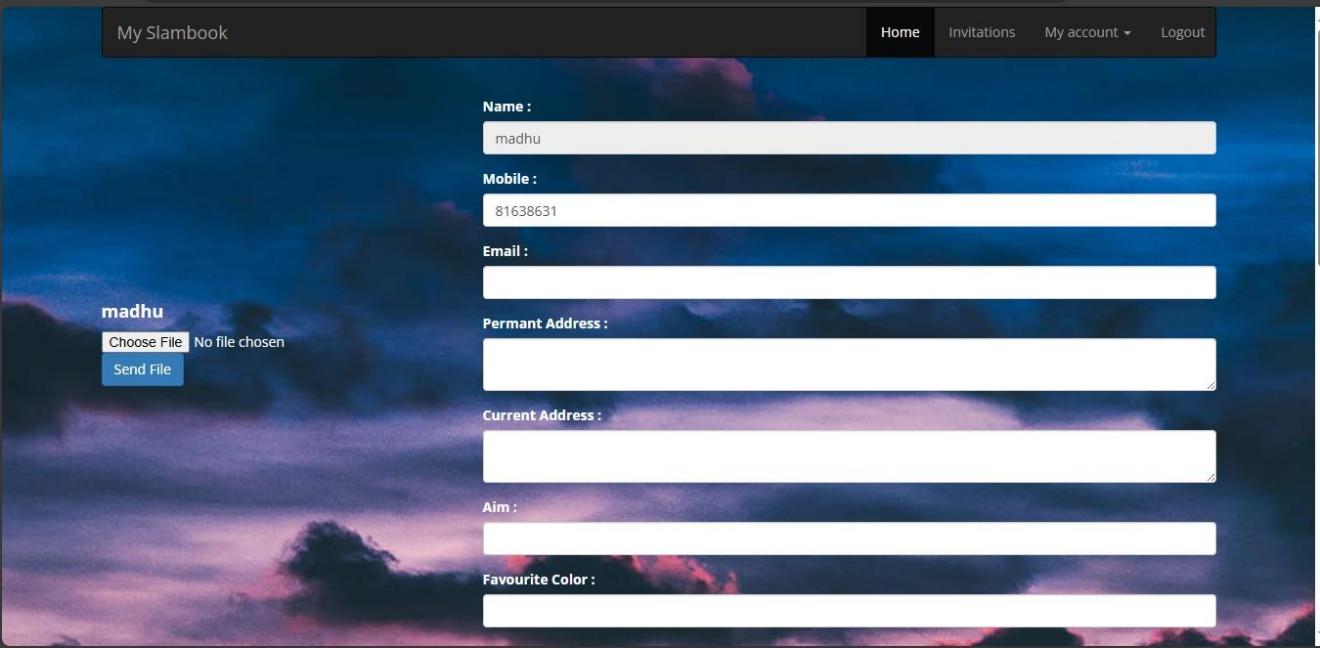
**ADMIN CAN ADD NEW FRIEND:**

****

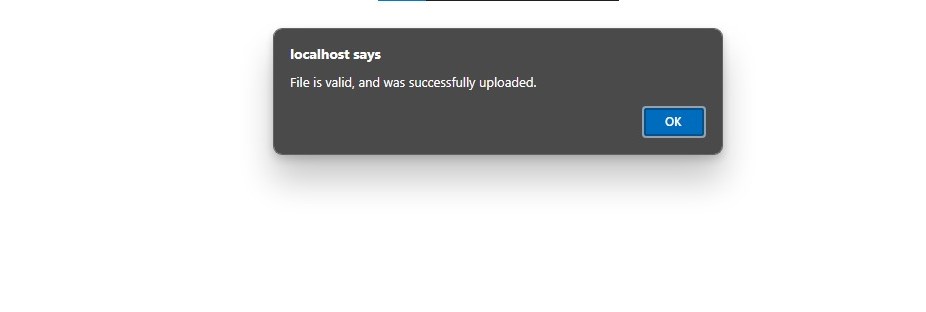
**USER CAN SEE ALL FRIENDS:**



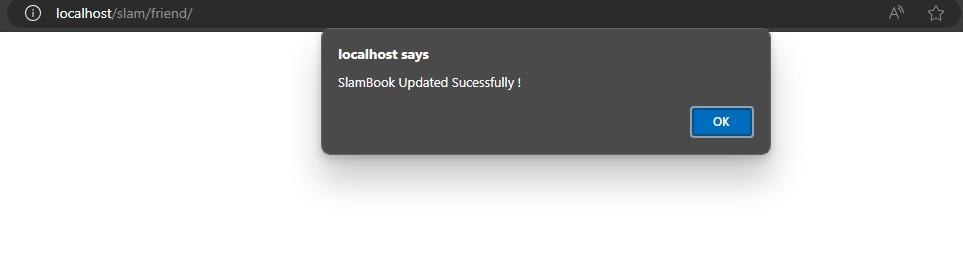
**AFTER USER LOGIN :**



**WHEN USER UPLOADS A PIC:**



**WHEN USER UPLOADS DATA :**



**TESTING AND VALIDATION**

Testing and validation are crucial to ensure that your online slam book functions correctly and meets user expectations. Here is a detailed testing and validation plan for your online slam book.

Testing Plan

1. Unit Testing:
   * Objective: Verify that individual components of the system work as expected.
   * Scope: Test functions, methods, and classes.
   * Tools: JUnit, NUnit, Mocha, Jasmine.
   * Examples:
     + Test user registration and login functions.
     + Test profile creation and editing methods.
     + Test the functionality of sending and accepting friend requests.
2. Integration Testing:
   * Objective: Ensure that different modules and components work together correctly.
   * Scope: Test interactions between integrated units.
   * Tools: Postman, Selenium, Apache JMeter.
   * Examples:
     + Test the interaction between user registration and login modules.
     + Test how profile information is retrieved and displayed.
     + Verify the integration of chat and messaging features.
3. System Testing:
   * Objective: Validate the complete and integrated software product.
   * Scope: Test the entire system as a whole.
   * Tools: Selenium, TestComplete, QTP.
   * Examples:
     + Test the overall user workflow from registration to creating slam book entries.
     + Test the end-to-end process of adding and managing friends.
     + Validate privacy settings and their impact on entry visibility.
4. User Acceptance Testing (UAT):
   * Objective: Ensure the system meets the requirements and expectations of end-users.
   * Scope: Test with real users in a staging environment.
   * Tools: UserTesting, TestRail.
   * Examples:
     + Collect feedback from users on the ease of use of the new interface.
     + Validate that users can successfully create and share slam book entries.
     + Ensure users can navigate and use new features like media integration and real-time chat.
5. Performance Testing:
   * Objective: Assess the speed, responsiveness, and stability of the system under load.
   * Scope: Test system performance under various conditions.
   * Tools: Apache JMeter, LoadRunner, Gatling.
   * Examples:
     + Test the system's behavior with a large number of concurrent users.
     + Measure page load times and response times for different actions.
     + Identify bottlenecks and optimize performance.
6. Security Testing:
   * Objective: Identify vulnerabilities and ensure the system is secure.
   * Scope: Test for security flaws and potential exploits.
   * Tools: OWASP ZAP, Burp Suite, Nessus.
   * Examples:
     + Test for SQL injection, XSS, and other common vulnerabilities.
     + Validate the effectiveness of two-factor authentication.
     + Ensure data encryption and secure storage of user information.

### Validation Plan

1. Requirement Validation:
   * Objective: Ensure that the system meets all defined requirements.
   * Scope: Review requirement specifications and compare them with the implemented system.
   * Methods: Requirement traceability matrix, reviews, inspections.
   * Examples:
     + Validate that all user stories and use cases are addressed.
     + Ensure that all mandatory features are implemented.
2. Functional Validation:
   * Objective: Confirm that the system functions correctly.
   * Scope: Validate all functional aspects of the system.
   * Methods: Functional testing, UAT.
   * Examples:
     + Verify that all buttons and links perform their intended actions.
     + Confirm that user inputs are correctly processed and stored.
3. Usability Validation:
   * Objective: Ensure the system is user-friendly and meets usability standards.
   * Scope: Validate the user interface and user experience.
   * Methods: Usability testing, user feedback.
   * Examples:
     + Conduct usability testing sessions with a diverse group of users.
     + Collect and analyze user feedback to identify areas for improvement.
4. Performance Validation:
   * Objective: Ensure the system performs well under expected load conditions.
   * Scope: Validate the system's performance metrics.
   * Methods: Performance testing, load testing.
   * Examples:
     + Validate that the system can handle peak loads without degradation.
     + Ensure that response times are within acceptable limits.
5. Security Validation:
   * Objective: Ensure the system is secure and meets security requirements.
   * Scope: Validate the system's security measures.
   * Methods: Security testing, vulnerability assessments.
   * Examples:
     + Confirm that all identified vulnerabilities have been addressed.
     + Validate that user data is protected and privacy settings are effective.

### Documentation and Reporting

1. Test Plan Document:
   * Outline the overall testing strategy, objectives, scope, resources, and schedule.
2. Test Cases and Scripts:
   * Detailed test cases for each testing phase, including expected results and acceptance criteria.
3. Test Execution Reports:
   * Reports detailing the results of each testing phase, including passed, failed, and blocked test cases.
4. Bug Reports:
   * Detailed descriptions of identified issues, including steps to reproduce, severity, and potential fixes.
5. Validation Reports:
   * Summarize the results of validation activities, highlighting areas of compliance and non-compliance.

**CONCLUSION**

The Online Slambook successfully bridges the gap between nostalgic traditions and modern technology by transforming the classic slam book experience into a digital format. This web-based application provides users with a secure, user-friendly platform to create, share, and view personal entries, preserving the essence of slam books while enhancing accessibility and convenience.

Throughout the project, key objectives such as user authentication, entry creation, and data security have been effectively implemented. Users can confidently document their thoughts, memories, and reflections, knowing their data is protected and easily accessible from anywhere.

The Online Slambook not only captures the charm and fun of traditional slam books but also fosters a sense of community by allowing users to connect and share their personal experiences. Future enhancements, such as social media integration, advanced search functionality, and mobile app development, will further expand the platform's capabilities and improve user experience.

In conclusion, the Online Slambook is a successful initiative that modernizes a beloved tradition, offering a secure, interactive, and enjoyable space for users to document and share their life's journey. This project stands as a testament to the enduring appeal of slam books and their ability to adapt and thrive in the digital age.

**REFERENCES**

* PHP and CSS languages.
* Object-Oriented Analysis and Design(OOAD).
* “Database System Concepts” by Abraham Silberchatz,Henry F.Korth,andS.Sudarshan.
* SQL Tutorials.