VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belgaum-590014

A TECHNICAL MINI PROJECT REPORT

ON

"QUIZ APPLICATION"

Submitted in the partial fulfillment for the requirement of 5th Semester

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

KATABATHUNI BOSE (1BC18CS009)
KAPIL RAJORIA (1BC18CS008)
NIDHI SHUKLA (1BC18CS018)
MUSKAAN PORWAL (1BC18CS016)
AMIT SINGH (1BC17CS002)

Under the Guidance of

Mrs. RAJITHA K R

Dept. of CSE,

BCET

BANGALORE COLLEGE OF ENGINEERING & TECHNOLOGY

Bangalore-560099

BANGALORE COLLEGE OF ENGINEERING & TECHNOLOGY

Department of Computer Science and Engineering

Bangalore- 560099

CERTIFICATE

This is to certify that the mini Project work entitled ***"QUIZ APPLICATION"*** is a bonafide work carried out by **KATABATHUNI BOSE, KAPIL RAJORIA, NIDHI SHUKLA, MUSKAAN PORWAL, AMIT SINGH** and in partial fulfilment for the requirement of **5th Semester**, **Bachelor of Engineering** in **Computer Science and Engineering** of **Visvesvaraya Technological University**, Belgaum during the year **2020-2021.** It is certified that all corrections / suggestions indicated for the internal assessment have been incorporated in the report. This report has been approved as it satisfies the academic requirements in respect of mini project work prescribed for **Bachelor of Engineering Degree**.

ACKNOWLEGMENT

It is indeed with a great pleasure and immense sense of gratitude that we acknowledge the help of the Management of Bangalore College of Engineering and Technology for providing such a healthy environment for the successful completion of project work. We are highly indebted to our Principal **Dr. CHANNANKAIAH, PhD**, Bangalore College Of Engineering And Technology, for the facilities provided to accomplish this main project.

We would like to thank our **Mrs. T M KAVITHA**, Head of the Department of Computer Science and Engineering, Bangalore College of Engineering And Technology, for this constructive criticism throughout our project.

We feel elated in manifesting our sense of gratitude tour internal project guide **Mrs. RAJITHA K R, Asst Professor, Department of Computer Science and Engineering**, Bangalore College of Engineering And Technology. He has been a constant source of inspiration for us and we are very deeply thankful to him for his support and valuable advice.

We extremely grateful to our Departmental staff members, Lab technicians and Non-teaching staff members for their extreme help throughout our project.

Finally we express our heartful thanks to all of our friends who helped us in successful completion of this project.

Project Associates:

KATABATHUNI BOSE (1BC18CS009)
KAPIL RAJORIA (1BC18CS008)
NIDHI SHUKLA (1BC18CS018)
MUSKAAN PORWAL (1BC18CS016)
AMIT SINGH (1BC17CS002)

ABSTRACT

Project details – Due to covid 19 colleges can no longer take offline exams. For theoretical questions of course human help is needed to check the answers but for objective type questions we can use this project. Using several languages we can create an application that will list out a series of questions for users and the users will be allowed to answer those questions. Only one of the answer will be correct. User can select one of the four options (using radio buttons) available per question. The application will display the final score and the right answers of the questions after submitting.

In the **history section** there will be the history of all the quiz taken by a single person (same login).

In the **ranking section** you can see the details of all the quiz takers and there scores with ranking and with the **login page** we can either register with a new account or login with an existing account.

Technical Details – or front-end we will be using the following languages: mainly PHP and a little HTML , CSS , Bootstrap , Javascript and Jquery. And for back-end we will be using Mysql.

Steps for achieving the goal:

1. Xampp software is used for mysql.
2. Using xampp we can make our quiz appear online on the **link**: http://localhost/Online-Quiz /
3. To create different quizzes we can edit our SQL code.
4. To make tables for our database we will use SQL language.
5. We can import our SQL code in a newly created database "exam" using Xampp (phpMyAdmin/localhost).
6. We will be making the web page mainly using PHP. With PHP code we will also use some HTML , CSS and javascript.

INDEX

1. INTRODUCTION

1.1 Introduction:

Online Quiz are an important method of evaluating the success potential of students. This research effort the individuals under consideration were students who would be enrolling in computer courses or Technologies Registrations. A prototype of a web-based placement Quiz system is described from the standpoint of the research effort, end user, and software development. Class Marker's secure, professional web-based Quiz maker is an easy-to-use, customizable online testing solution for business, training & educational assessment with Test & Quizzes graded instantly saving your hours of paperwork.

1.2 Objective:

The Main Objective of Online Quiz is User-friendly systems are not only needed for the creator, but also for participants. A responsive design is an approach where the web designer wants to reach an optimal web experience for a wide range of devices. Multiple choice is the options you can use with our online Quiz system. Having more than one option is necessary to check several types of knowledge. Auto quiz publish, log in with username and password. Automatically check answers and give the result to the students.

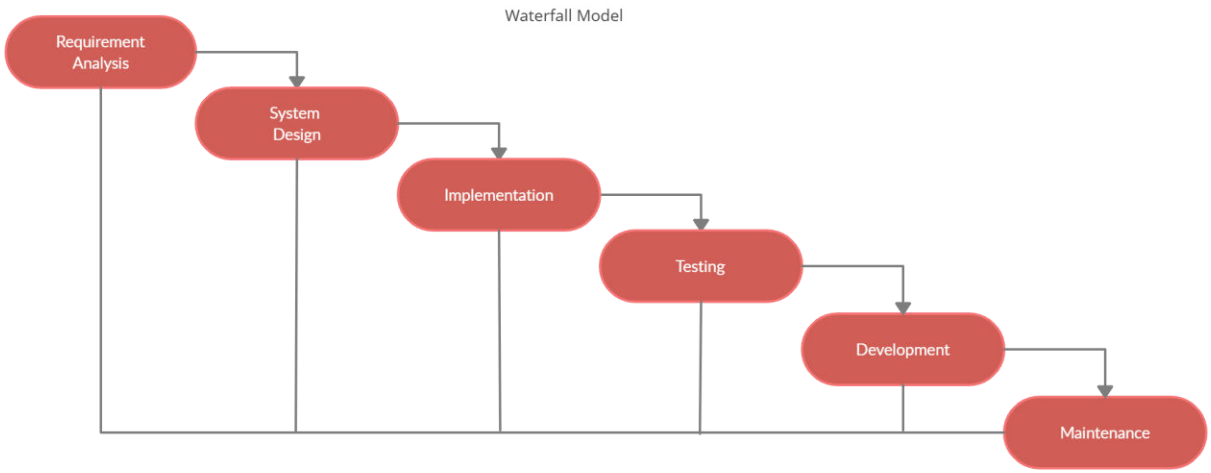
1.3 MODULES:

The entire project mainly consists of :

- **Methodology Development Model**

The sequential phases in Waterfall model are −

Requirement Gathering and analysis − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.



- **System Design** − The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

2. REQUIREMENT SPECIFICATION

To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These pre-requisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

2.1 HARDWARE REQUIREMENTS:

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

HARDWARE REQUIREMENTS FOR PRESENT PROJECT:

PROCESSOR : OS - Windows, Intel Quad Core, i5

RAM : 16 GB

HARD DISK : 1152 GB

2.2 SOFTWARE REQUIREMENTS:

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Language Used: Php

Database: Mysql

User Interface Design: Html, Java Script, Css, Bootstrap

Software: XAMPP Server

Technology Implemented**:** Apache Server

2.2.1 EXTERNAL INTERFACES:

- This interface will be actual interface through which the user will communication with the application and perform the desired tasks.

Admin login

ID:

Role: Admin wishes to login to the system

Precondition: Username and Password

Success end Condition: Main option of screen display

Failed end Condition: User has entered incorrect Username and Password or both Delete.

ID:

Precondition: Admin has successfully navigated to the search result

Success end Condition: Admin has successfully made the changes

3. ANALYSIS

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

The objective of the system analysis activity is to develop structured system specification for the proposed system. The structured system specification should describe what the proposed system would do; independent of the technology, which will be used to implement these requirements. The structured system specification will be used to implement these requirements.

The essential model may itself consist of multiple models, modeling different aspect of the system. The data flow diagrams may model the data and there relationships and the state transition diagram may model time dependent behavior of the system. The essential model thus consists of the following.

- Context diagram
- Leveled data flow diagrams
- Process specification for elementary bubbles
- Data dictionary for the flow and stores on the DFDs.

3.4 SOFTWARE SPECIFICATION

HTML:

HTML or **Hypertext Markup Language** is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.[4] Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

- **CASCADING STYLE SHEETS** (**CSS**):

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation

characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate css file, and reduce complexity and repetition in the structural content.

- **BOOTSTRAP:**

Bootstrap is a free and open-source front-end framework for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only.

- **MySQL:**

MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.

FEATURES OF MySQL:

Internals and portability:

- Written in C and C++.
- Tested with a broad range of different compilers.
- Works on many different platforms.
- Tested with Purify (a commercial memory leakage detector) as well as with Val grind, a GPL tool.
- Uses multi-layered server design with independent modules.

Security:

- A privilege and password system that is very flexible and secure, and that enables host-based verification.
- Password security by encryption of all password traffic when you connect to a server.

Scalability and Limits:

- Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.

- Support for up to 64 indexes per table (32 before MySQL 4.1.2). Each index may consist of 1 to 16 columns or parts of columns. The maximum index width is 767 bytes for `InnoDB` tables, or 1000 for `MyISAM`; before MySQL 4.1.2, the limit is 500 bytes. An index may use a prefix of a column for `CHAR`, `VARCHAR`, `BLOB`, or `TEXT` column types.

WHY TO USE MySQL:

- Leading open source RDBMS
- Ease of use – No frills
- Fast
- Robust
- Security
- Multiple OS support
- Free
- Technical support
- Support large database– up to 50 million rows, file size limit up to 8 Million TB

➢ **JAVASCRIPT:**

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it.

WHY TO USE JAVASCRIPT:

JavaScript is one of the 3 languages all web developers must learn:

1. HTML to define the content of web pages
2. CSS to specify the layout of web pages
3. JavaScript to specify the behavior of web pages

OTHER USES OF JAVASCRIPT:

- Delete HTML elements
- Create new HTML elements
- Copy HTML elements
- In HTML, JavaScript is a sequence of statements that can be executed by the web browser.

JAVASCRIPT PROPERTIES:

- Properties are the values associated with a JavaScript object.
- A JavaScript object is a collection of unordered properties.
- Properties can usually be changed, added, and deleted, but some are read only.

➢ **PHP:**

WHAT IS PHP?

- PHP is an acronym for "PHP Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP costs nothing, it is free to download and use

WHAT IS PHP FILE?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code
- PHP code are executed on the server, and the result is returned to the browser as plain HTML
- PHP files have extension ".php"

WHAT CAN PHP DO?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can restrict users to access some pages on your website
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies. You can also output any text, such as XHTML and XML.

WHY PHP?

- PHP runs on various platforms (Windows, Linux, UNIX, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases

PHP is free. Download it from the official PHP resource: www.php.net

- **APACHE:**

The Apache HTTP Server is web server software notable for playing a key role in the initial growth of the World Wide Web. In 2009 it became the first web server software to surpass the 100 million web site milestone. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Since April 1996 Apache has been the most popular HTTP server software in use. As of November 2010 Apache served over 59.36% of all websites and over 66.56% of the first one million busiest websites.

- **XAMPP:**

XAMPP is a small and light Apache distribution containing the most common web development technologies in a single package. Its contents, small size, and portability make it the ideal tool for students developing and testing applications in PHP and MySQL. XAMPP is available as a free download in two specific packages: full and lite. While the full package download provides a wide array of development tools, XAMPP Lite contains the necessary technologies that meet the Ontario Skills Competition standards. The light version is a small package containing Apache HTTP Server, PHP, MySQL, phpMyAdmin, Openssl, and SQLite.

4. DESIGN

4.1 SYSTEM DESIGN

System design involves transformation of the user implementation model into software design. The design specification of the proposed system consists of the following.

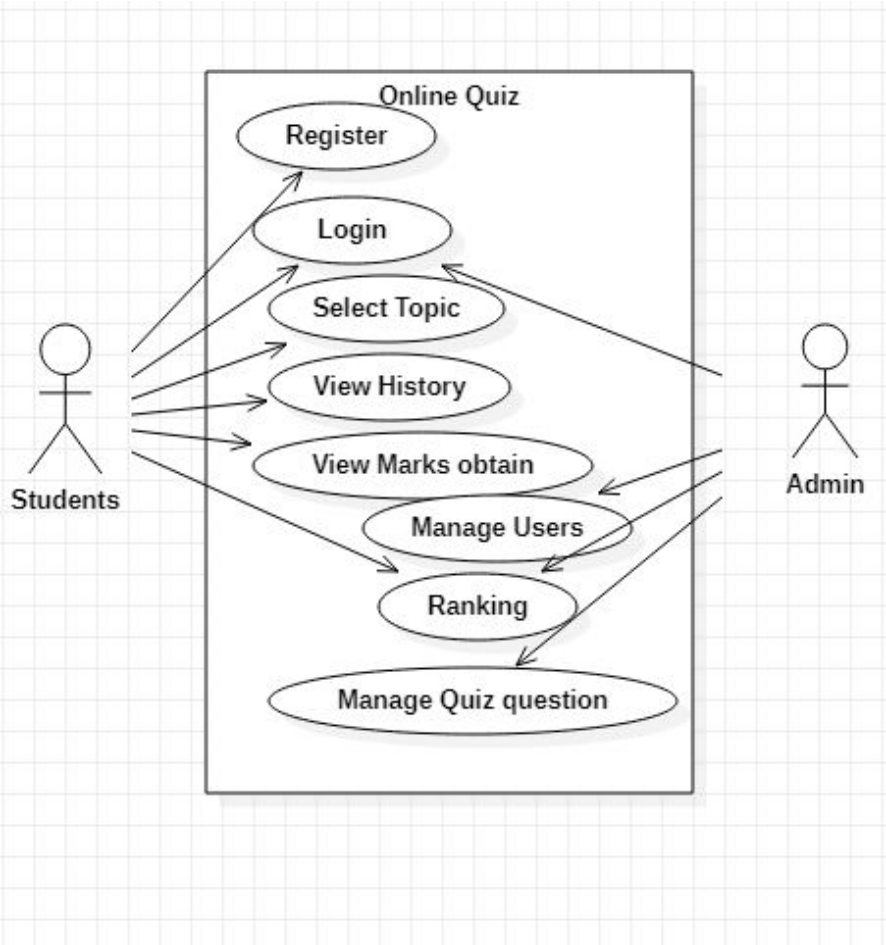
1. Use case diagram
2. Sequence diagram
3. Flowchart diagram
4. ER diagram

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

Use case diagrams are formally included in two modeling languages defined by the OMG: The Unified modeling language (UML) and the systems modeling language (SML).

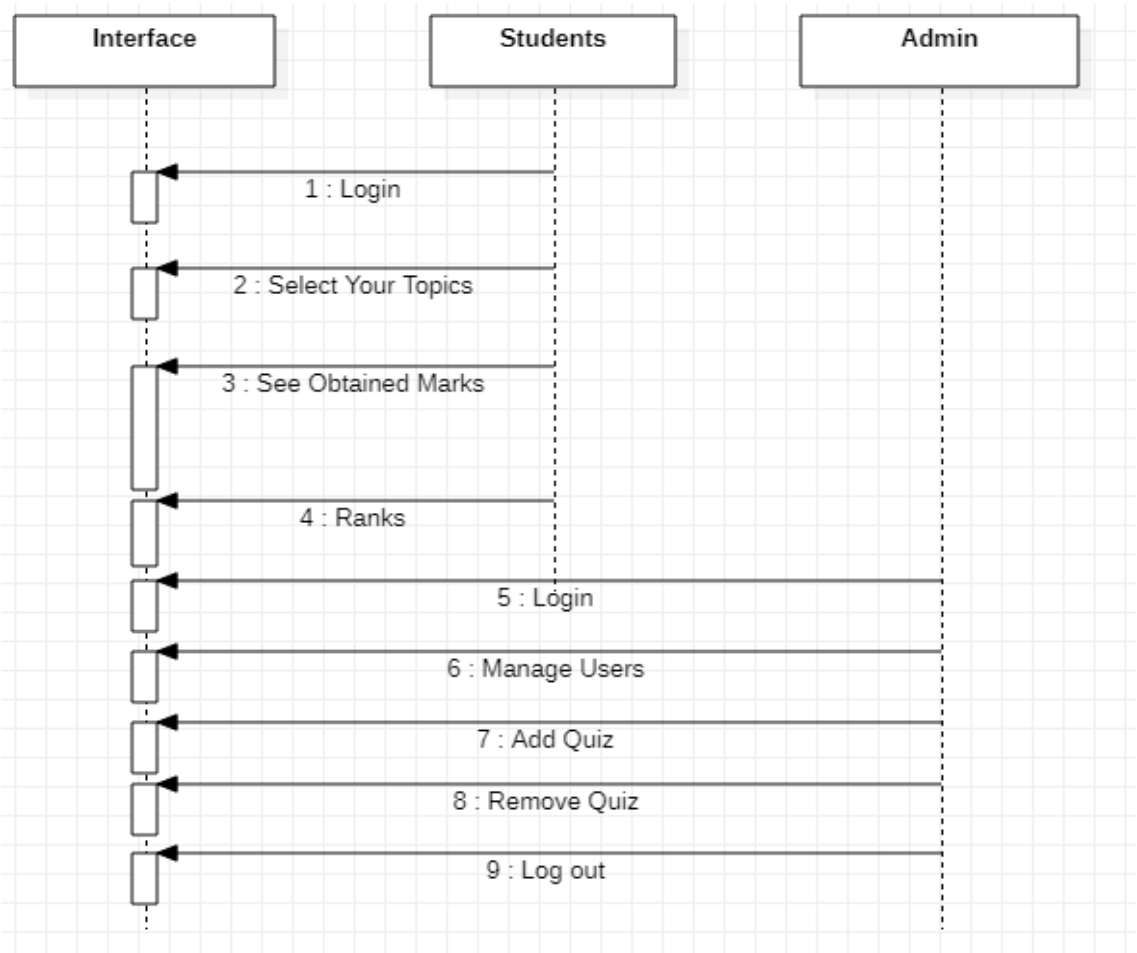
Use case diagram:



SEQUENCE DIAGRAM:

A **Sequence Diagram** is an interaction diagram that emphasis the time ordering of messages; a collaboration diagram is an interaction diagram that emphasizes the structural organization of the objects that send and receive messages. Sequence diagrams and collaboration diagrams are isomorphic, meaning that you can take one and transform it into the other.

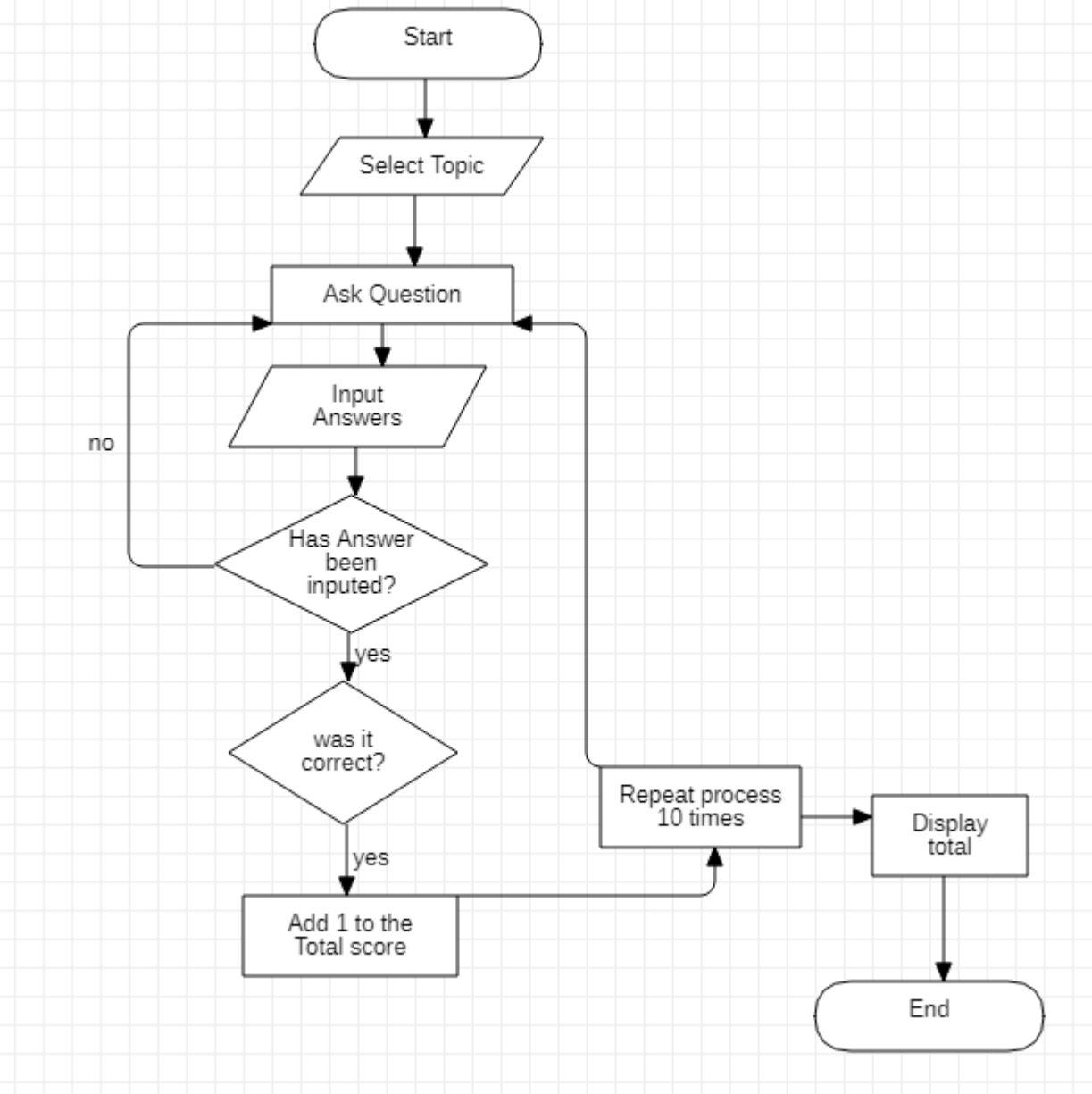
Sequence diagram:



FLOWCHART DIAGRAM:

A flowchart is simply a graphical representation of steps. It shows steps in sequential order and is widely used in presenting the flow of algorithms, workflow or processes. Typically, a flowchart shows the steps as boxes of various kinds, and their order by connecting them with arrows.

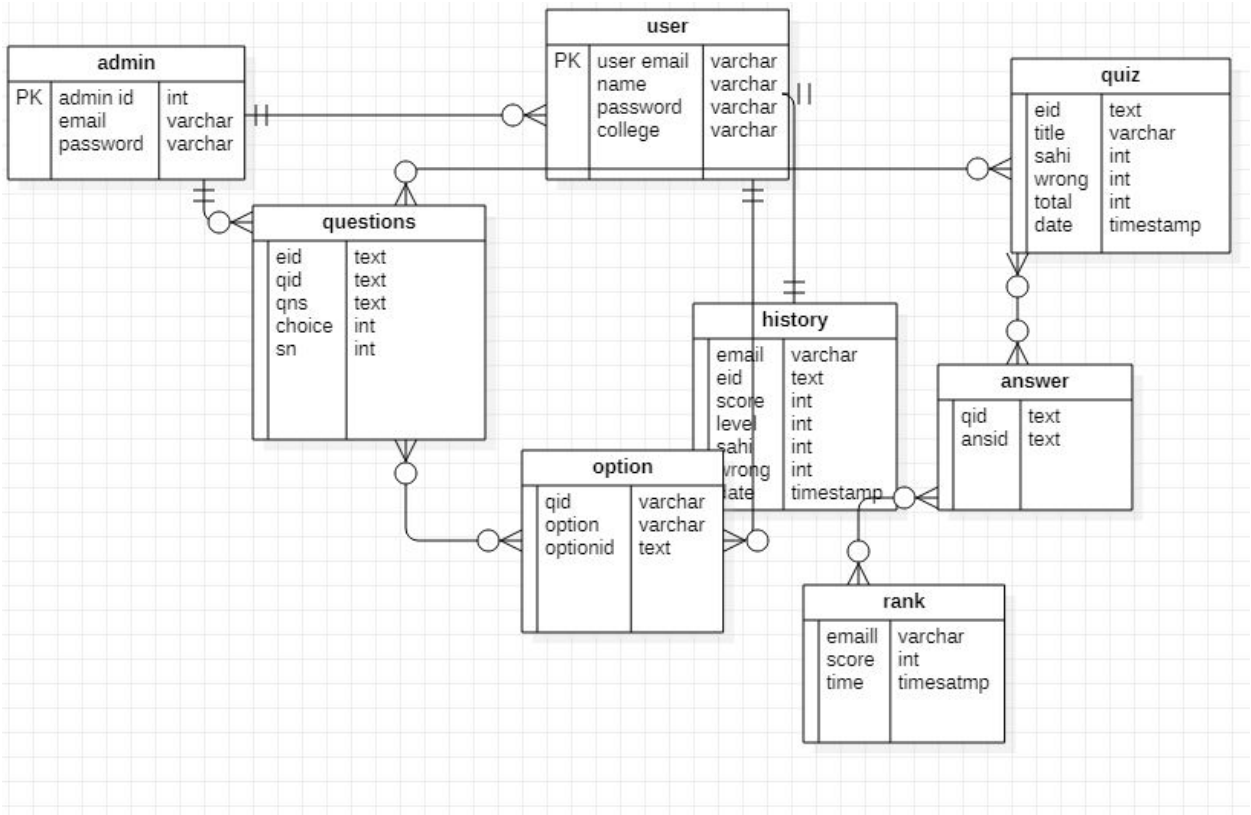
Flowchart diagram:



ER DIAGRAM:

An ER diagram shows the relationship among entity sets, In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

ER diagram:



5. SYSTEM IMPLEMENTATION

5.1 Introduction:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new.

This activity includes programming, testing and integration of modules into a progressively more complete system. Implementation is the process of collect all the required parts and assembles them into a major product.

5.2 Sample code:

Welcome.php

```
<?php
   include_once 'database.php';
   session_start();
   if(!(isset($_SESSION['email'])))
   {
      header("location:login.php");
   }
   else
   {
      $name = $_SESSION['name'];
      $email = $_SESSION['email'];
      include_once 'database.php';
   }
?>


<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Welcome | Online Quiz System</title>
         <div class="col-md-12">
            <?php if(@$_GET['q']==1)
```

```
{
    $result = mysqli_query($con,"SELECT * FROM quiz ORDER BY date DESC") or die('Error');
    echo  '<div class="panel"><div class="table-responsive"><table class="table table-striped title1">

<tr><td><center><b>S.N.</b></center></td><td><center><b>Topic</b></center></td><td><center><b>Total question</b></center></td><td><center><b>Marks</center></b></td><td><center><b>Action</b></center></td></tr>';
    $c=1;
    while($row = mysqli_fetch_array($result)) {
      $title = $row['title'];
      $total = $row['total'];
      $sahi = $row['sahi'];
      $eid = $row['eid'];
    $q12=mysqli_query($con,"SELECT score FROM history WHERE eid='$eid' AND email='$email'" )or die('Error98');
    $rowcount=mysqli_num_rows($q12);
    if($rowcount == 0){
      echo '<tr><td><center>'.$c++.'</center></td><td><center>'.$title.'</center></td><td><center>'.$total.'</center></td><td><center>'.$sahi*$total.'</center></td><td><center><b><a href="welcome.php?q=quiz&step=2&eid='.$eid.'&n=1&t='.$total.'" class="btn sub1" style="color:black;margin:0px;background:#1de9b6"><span class="glyphicon glyphicon-new-window" aria-hidden="true"></span> <span class="title1"><b>Start</b></span></a></b></center></td></tr>';
    }
    else
    {
```

```
                echo '<tr
style="color:#99cc32"><td><center>'.$c++.'</center></td><td><center>'.$title.' <span
title="This quiz is already solve by you" class="glyphicon glyphicon-ok" aria-
hidden="true"></span></center></td><td><center>'.$total.'</center></td><td><center>'.$sahi*
$total.'</center></td><td><center><b><a
href="update.php?q=quizre&step=25&eid='.$eid.'&n=1&t='.$total.'" class="pull-right btn sub1"
style="color:black;margin:0px;background:red"><span class="glyphicon glyphicon-repeat" aria-
hidden="true"></span> <span
class="title1"><b>Restart</b></span></a></b></center></td></tr>';

                }

                }

                $c=0;

                echo '</table></div></div>';

            }?>

            <?php

                if(@$_GET['q']== 'quiz' && @$_GET['step']== 2)

                {

                    $eid=@$_GET['eid'];

                    $sn=@$_GET['n'];

                    $total=@$_GET['t'];

                    $q=mysqli_query($con,"SELECT * FROM questions WHERE eid='$eid' AND
sn='$sn' " );

                    echo '<div class="panel" style="margin:5%">';

                    while($row=mysqli_fetch_array($q) )

                    {

                        $qns=$row['qns'];

                        $qid=$row['qid'];

                        echo '<b>Question  '.$sn.' ::<br /><br />'.$qns.'</b><br /><br />';
```

```
                }

                $q=mysqli_query($con,"SELECT * FROM options WHERE qid='$qid' " );

                echo '<form
action="update.php?q=quiz&step=2&eid='.$eid.'&n='.$sn.'&t='.$total.'&qid='.$qid.'"
method="POST"  class="form-horizontal">

                <br />';


                while($row=mysqli_fetch_array($q) )

                {

                   $option=$row['option'];

                   $optionid=$row['optionid'];

                   echo'<input type="radio" name="ans"
value="'.$optionid.'"> '.$option.'<br /><br />';

                }

                echo'<br /><button type="submit" class="btn btn-primary"><span
class="glyphicon glyphicon-lock" aria-
hidden="true"></span> Submit</button></form></div>';

            }


            if(@$_GET['q']== 'result' && @$_GET['eid'])

            {

               $eid=@$_GET['eid'];

               $q=mysqli_query($con,"SELECT * FROM history WHERE eid='$eid' AND
email='$email' " )or die('Error157');

               echo  '<div class="panel">

               <center><h1 class="title" style="color:#660033">Result</h1><center><br
/><table class="table table-striped title1" style="font-size:20px;font-weight:1000;">';
```

```
                while($row=mysqli_fetch_array($q) )

                {

                   $s=$row['score'];

                   $w=$row['wrong'];

                   $r=$row['sahi'];

                   $qa=$row['level'];

                   echo '<tr style="color:#66CCFF"><td>Total
Questions</td><td>'.$qa.'</td></tr>

                      <tr style="color:#99cc32"><td>right Answer <span class="glyphicon
glyphicon-ok-circle" aria-hidden="true"></span></td><td>'.$r.'</td></tr>

                      <tr style="color:red"><td>Wrong Answer <span class="glyphicon
glyphicon-remove-circle" aria-hidden="true"></span></td><td>'.$w.'</td></tr>

                      <tr style="color:#66CCFF"><td>Score <span class="glyphicon
glyphicon-star" aria-hidden="true"></span></td><td>'.$s.'</td></tr>';

                }

                $q=mysqli_query($con,"SELECT * FROM rank WHERE  email='$email' " )or
die('Error157');

                while($row=mysqli_fetch_array($q) )

                {

                   $s=$row['score'];

                   echo '<tr style="color:#990000"><td>Overall Score <span
class="glyphicon glyphicon-stats" aria-hidden="true"></span></td><td>'.$s.'</td></tr>';

                }

                echo '</table></div>';

             }

          ?>
```

```
<?php
    if(@$_GET['q']== 2)
    {
        $q=mysqli_query($con,"SELECT * FROM history WHERE email='$email' ORDER BY date DESC " )or die('Error197');
        echo  '<div class="panel title">
        <table class="table table-striped title1" >
        <tr style="color:black;"><td><center><b>S.N.</b></center></td><td><center><b>Quiz</b></center></td><td><center><b>Question Solved</b></center></td><td><center><b>Right</b></center></td><td><center><b>Wrong<b></center></td><td><center><b>Score</b></center></td>';
        $c=0;
        while($row=mysqli_fetch_array($q) )
        {
        $eid=$row['eid'];
        $s=$row['score'];
        $w=$row['wrong'];
        $r=$row['sahi'];
        $qa=$row['level'];
        $q23=mysqli_query($con,"SELECT title FROM quiz WHERE  eid='$eid' " )or die('Error208');

        while($row=mysqli_fetch_array($q23) )
        {  $title=$row['title'];  }
        $c++;
        echo '<tr><td><center>'.$c.'</center></td><td><center>'.$title.'</center></td><td><center>'.$qa.'</c
```

```
enter></td><td><center>'.$r.'</center></td><td><center>'.$w.'</center></td><td><center>'.$s.'
</center></td></tr>';

                }

                echo'</table></div>';

            }


            if(@$_GET['q']== 3)

            {

                $q=mysqli_query($con,"SELECT * FROM rank ORDER BY score DESC " )or
die('Error223');

                echo  '<div class="panel title"><div class="table-responsive">

                <table class="table table-striped title1" >

                <tr
style="color:red"><td><center><b>Rank</b></center></td><td><center><b>Name</b></cente
r></td><td><center><b>Email</b></center></td><td><center><b>Score</b></center></td></t
r>';

                $c=0;


                while($row=mysqli_fetch_array($q) )

                {

                    $e=$row['email'];

                    $s=$row['score'];

                    $q12=mysqli_query($con,"SELECT * FROM user WHERE email='$e' " )or
die('Error231');

                    while($row=mysqli_fetch_array($q12) )

                    {

                        $name=$row['name'];
```

```
                    }
                    $c++;
                    echo '<tr><td
style="color:black"><center><b>'.$c.'</b></center></td><td><center>'.$name.'</center></td><
td><center>'.$e.'</center></td><td><center>'.$s.'</center></td></tr>';
                }
                echo '</table></div></div>';
            }
        ?>
</body>
</html>
```

Index.php

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>| Online Quiz System |</title>
    <link rel="stylesheet" type="text/css" href="css/index.css" />
    <link rel="shortcut icon" type="image/png" href="image/logo.png" />
    <style type="text/css">
      body {
        width: 100%;
        background: url(image/book.png) ;
        background-position: center center;
```

```
            background-repeat: no-repeat;
            background-attachment: fixed;
            background-size: cover;
        }
    </style>
</head>
<body>
    <center>
        <div class="intro">
            <h1> online quiz system </h1>
            <a href="login.php" class="btn"> login </a>  
            <a href="register.php" class="btn"> register </a>
            <h2> Good  Luck. </h2>
        </div>
    </center>
</body>
</html>
```

User–login.php

```
<?php
        require('database.php');
        session_start();
        if(isset($_SESSION["email"]))
        {
                session_destroy();
```

```
}

$ref=@$_GET['q'];
if(isset($_POST['submit']))
{
        $email = $_POST['email'];
        $pass = $_POST['password'];
        $email = stripslashes($email);
        $email = addslashes($email);
        $pass = stripslashes($pass);
        $pass = addslashes($pass);
        $email = mysqli_real_escape_string($con,$email);
        $pass = mysqli_real_escape_string($con,$pass);
        $str = "SELECT * FROM user WHERE email='$email' and password='$pass'";
        $result = mysqli_query($con,$str);
        if((mysqli_num_rows($result))!=1)
        {
                echo "<center><h3><script>alert('Sorry.. Wrong Username (or)
Password');</script></h3></center>";
                header("refresh:0;url=login.php");
        }
        else
        {
                $_SESSION['logged']=$email;
                $row=mysqli_fetch_array($result);
```

```
                    $_SESSION['name']=$row[1];
                    $_SESSION['id']=$row[0];
                    $_SESSION['email']=$row[2];
                    $_SESSION['password']=$row[3];
                    header('location: welcome.php?q=1');
                }
        }
?>
<!DOCTYPE html>
<html>
        <head>
                <meta charset="UTF-8">
                <meta name="viewport" content="width=device-width, initial-scale=1.0">
                <meta http-equiv="X-UA-Compatible" content="ie=edge">
                <title>Login | Online Quiz System</title>
                <link rel="stylesheet" href="scripts/bootstrap/bootstrap.min.css">
                <link rel="stylesheet" href="scripts/ionicons/css/ionicons.min.css">
                <link rel="stylesheet" href="css/form.css">
    <style type="text/css">
        body{
            width: 100%;
            background: url(image/book.png) ;
            background-position: center center;
            background-repeat: no-repeat;
            background-attachment: fixed;
```

```
            background-size: cover;
          }
        </style>
         </head>
         <body>
                <section class="login first grey">
                        <div class="container">
                                <div class="box-wrapper">
                                        <div class="box box-border">
                                                <div class="box-body">
                                                <center> <h5 style="font-family: Noto Sans;">Login to </h5><h4 style="font-family: Noto Sans;">Online Quiz System</h4></center><br>
                                                        <form method="post" action="login.php" enctype="multipart/form-data">
                                                                <div class="form-group">
                                                                        <label>Enter Your Email Id:</label>
                                                                        <input type="email" name="email" class="form-control">
                                                                </div>
                                                                <div class="form-group">
                                                                        <label class="fw">Enter Your Password:
                                                                                <a href="javascript:void(0)" class="pull-right">Forgot Password?</a>
                                                                        </label>
```

```
                                                    <input type="password"
name="password" class="form-control">
                                                </div>
                                                <div class="form-group text-right">
                                                    <button class="btn btn-
primary btn-block" name="submit">Login</button>
                                                </div>
                                                <div class="form-group text-
center">
                                                    <span class="text-
muted">Don't have an account?</span> <a href="register.php">Register</a> Here..
                                                </div>
                                            </form>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </section>
                        <script src="js/jquery.js"></script>
                        <script src="scripts/bootstrap/bootstrap.min.js"></script>
                    </body>
</html>
```

User-Register.php

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

```
body{
    width: 100%;
    background: url(image/book.png) ;
    background-position: center center;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: cover;
  }
</style>
 </head>
 <body>
        <section class="login first grey">
            <div class="container">
                <div class="box-wrapper">
                    <div class="box box-border">
                        <div class="box-body">
                            <center> <h5 style="font-family: Noto Sans;">Register to </h5><h4 style="font-family: Noto Sans;">Online Quiz System</h4></center><br>
                            <form method="post" action="register.php" enctype="multipart/form-data">
                <div class="form-group">
                                    <label>Enter Your Username:</label>
                                    <input type="text" name="name" class="form-control" required />
                                </div>
```

```
                                        <div class="form-group">
                                            <label>Enter Your Email
Id:</label>
                                            <input type="email"
name="email" class="form-control" required />
                                        </div>
                                        <div class="form-group">
                                            <label>Enter Your
Password:</label>
                                            <input type="password"
name="password" class="form-control" required />
                                    </div>
                                        <div class="form-group">
                                            <label>Enter Your College
Name:</label>
                                            <input type="text"
name="college" class="form-control" required />
                                        </div>
                                    <div class="form-group text-right">
                                            <button class="btn btn-
primary btn-block" name="submit">Register</button>
                                        </div>
                                        <div class="form-group text-
center">
                                            <span class="text-
muted">Already have an account! </span> <a href="login.php">Login </a> Here..
                                        </div>
                                    </form>
```

```
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </section>
                        <script src="js/jquery.js"></script>
                        <script src="scripts/bootstrap/bootstrap.min.js"></script>
                    </body>
</html>
</body>
<!-- end: BODY -->
</html>
```

Admin.php

```
<?php
   include_once 'database.php';
   session_start();
   if(isset($_SESSION["email"]))
        {
                <input type="email" name="email" class="form-control">
                                                        </div>
                                                        <div class="form-group">
                                                            <label class="fw">Enter
Your Password:
                                                                <a
href="javascript:void(0)" class="pull-right">Forgot Password?</a>
```

```
                                                    </label>
                                                    <input type="password"
name="password" class="form-control">
                                                </div>
                                                <div class="form-group text-right">
                                                    <button class="btn btn-
primary btn-block" name="submit">Login</button>
                                                </div>

                                            </form>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </section>
                        <script src="js/jquery.js"></script>
                        <script src="scripts/bootstrap/bootstrap.min.js"></script>
                    </body>
</html>
```

6. TESTING

6.1 INTRODUCTION TO SYSTEM TESTING:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the

functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TESTING METHODS:

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing: User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

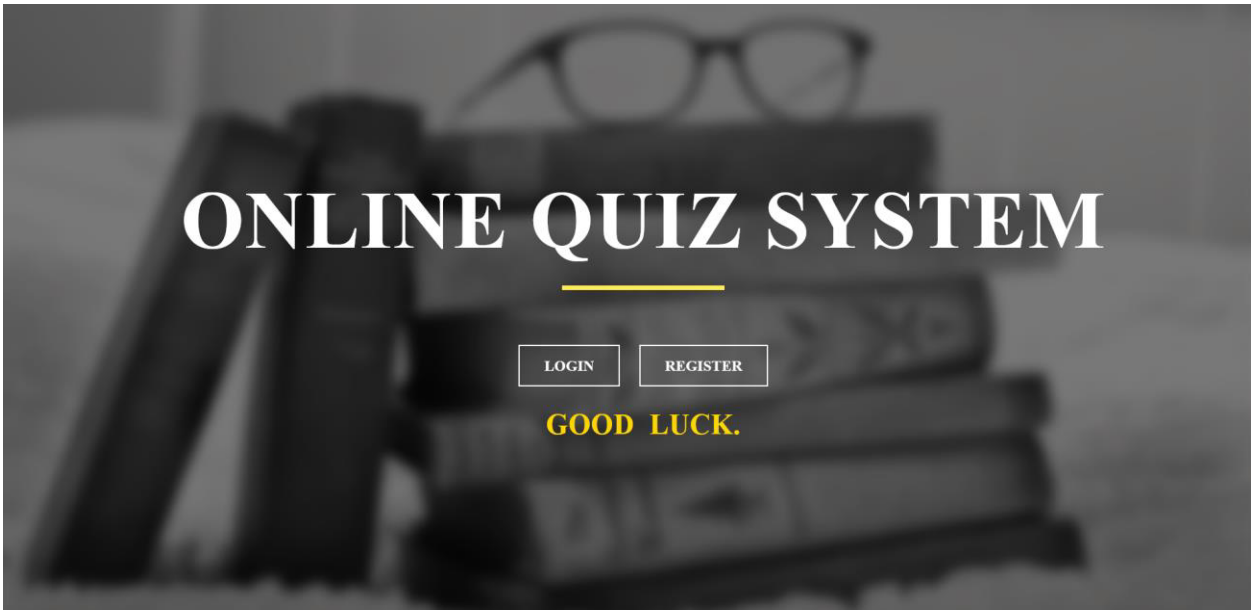
Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Test Generation:

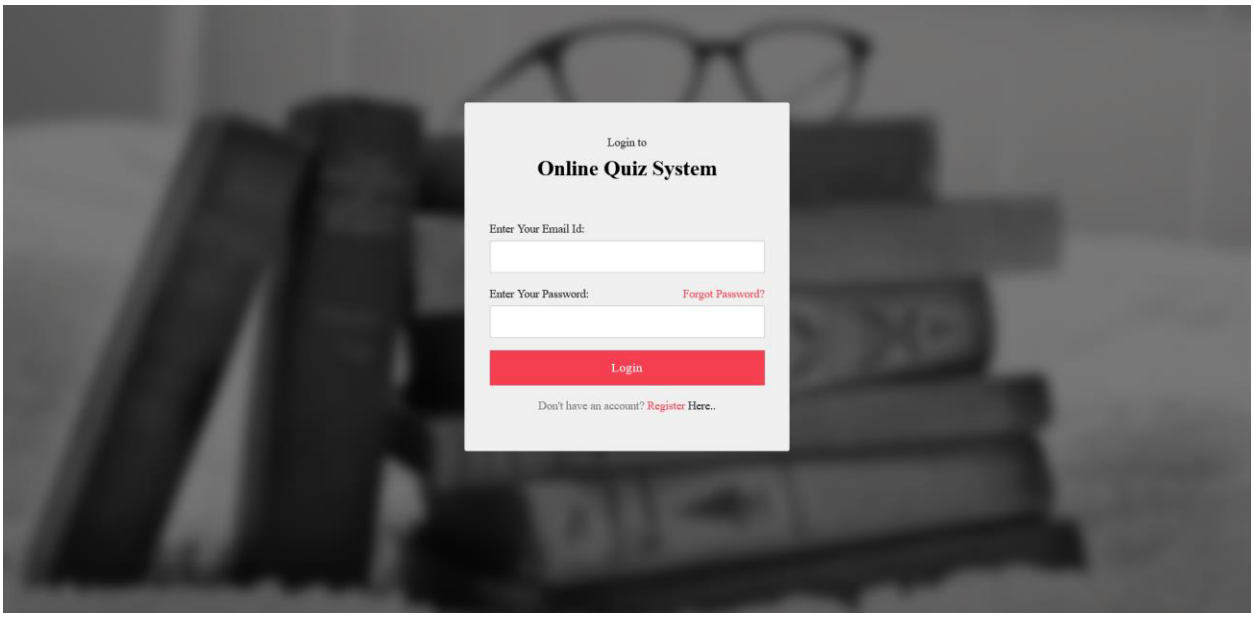
This activity generates a set of test data, which can be used to test the new system before accepting it. In the test generation phase all the parts are come which are to be tested to ensure that system does not produce any error. If there are some errors then we remove them and further it goes for accepting.

7. SAMPLE SCREENSHOTS

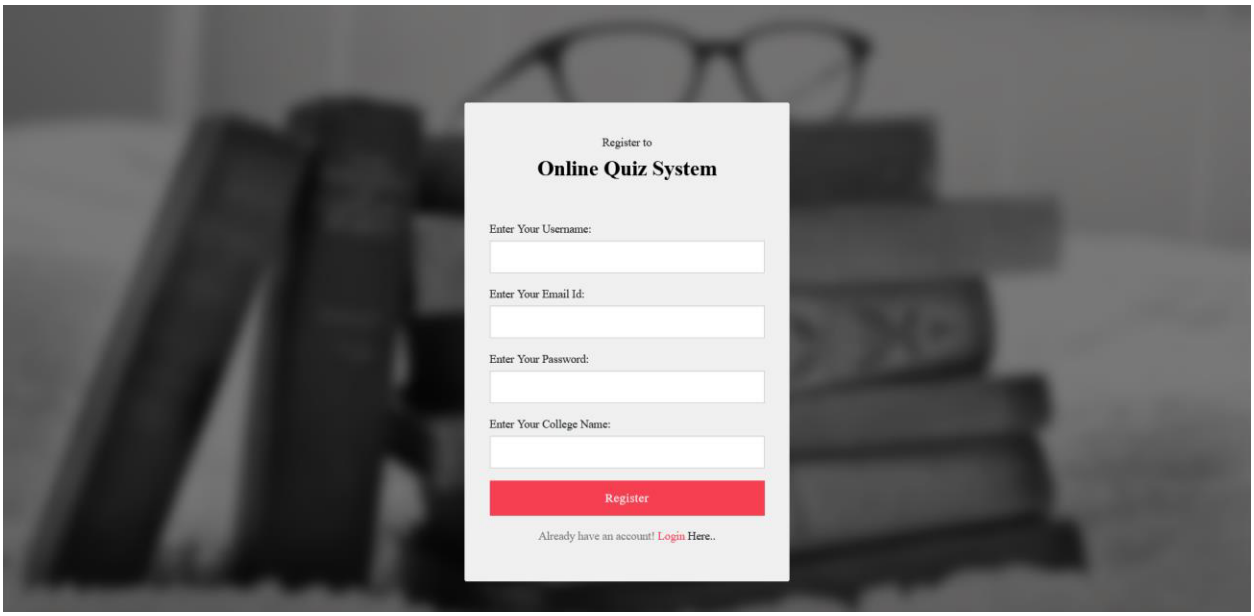
LOGIN PAGE:



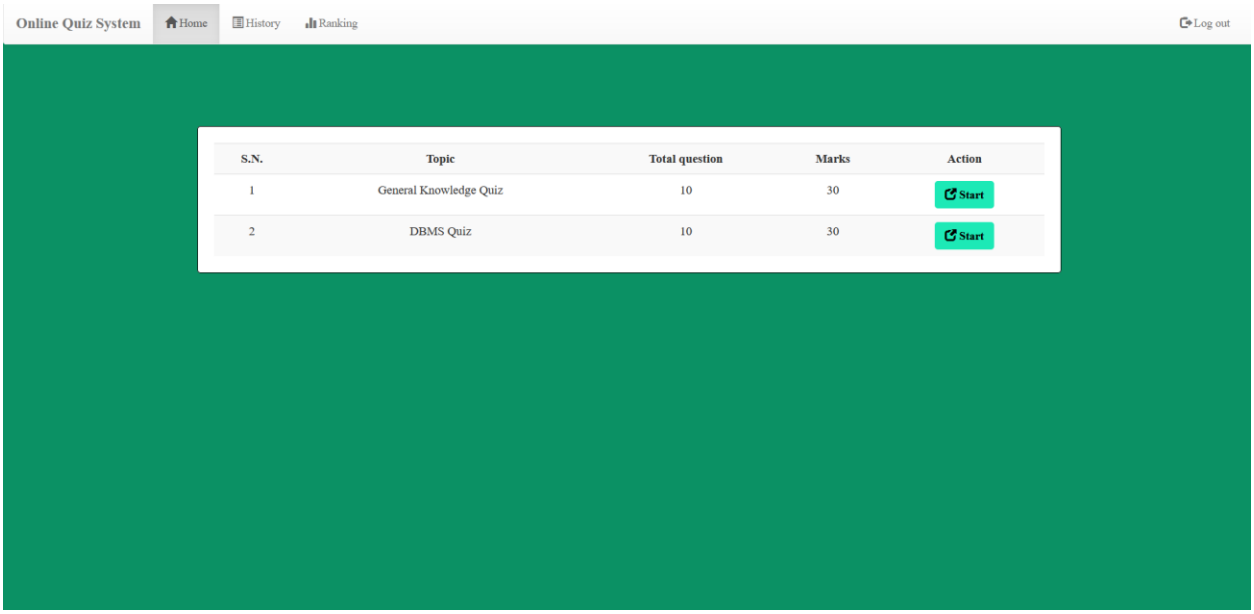
ADMIN LOGIN:



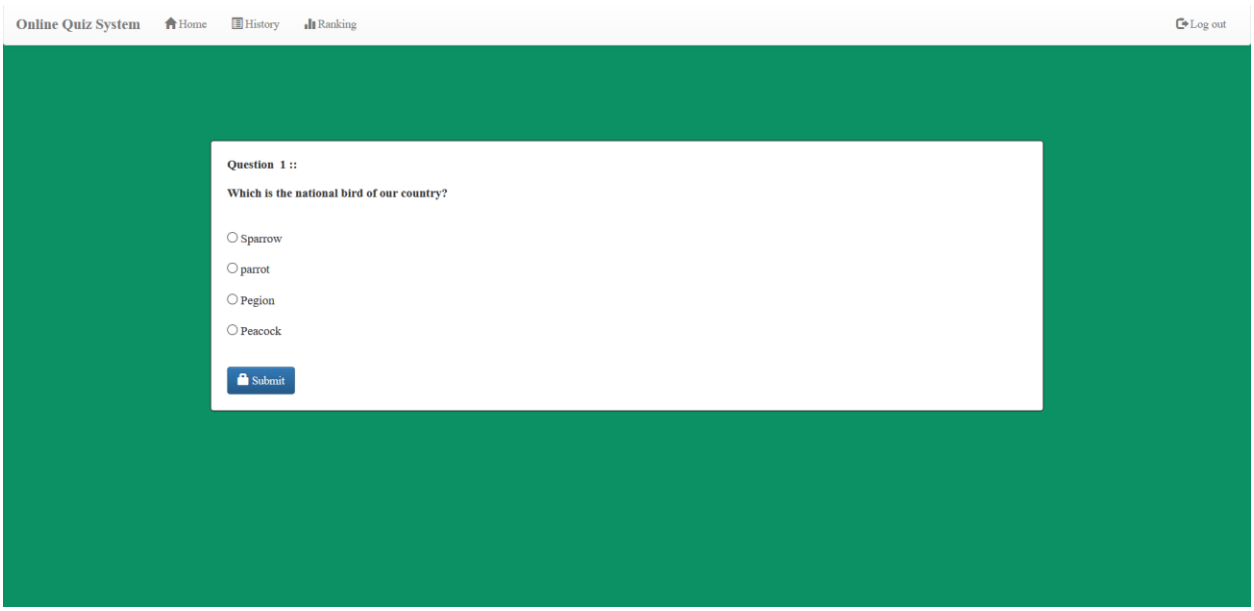
REGISTER LOGIN:



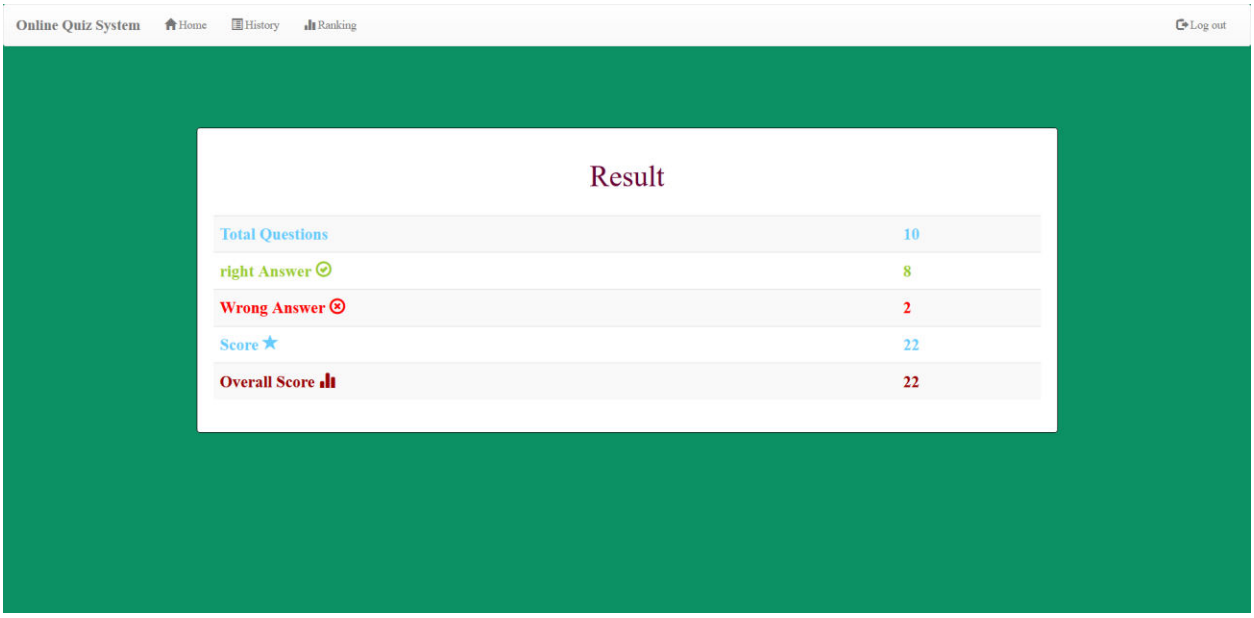
HOMEPAGE:



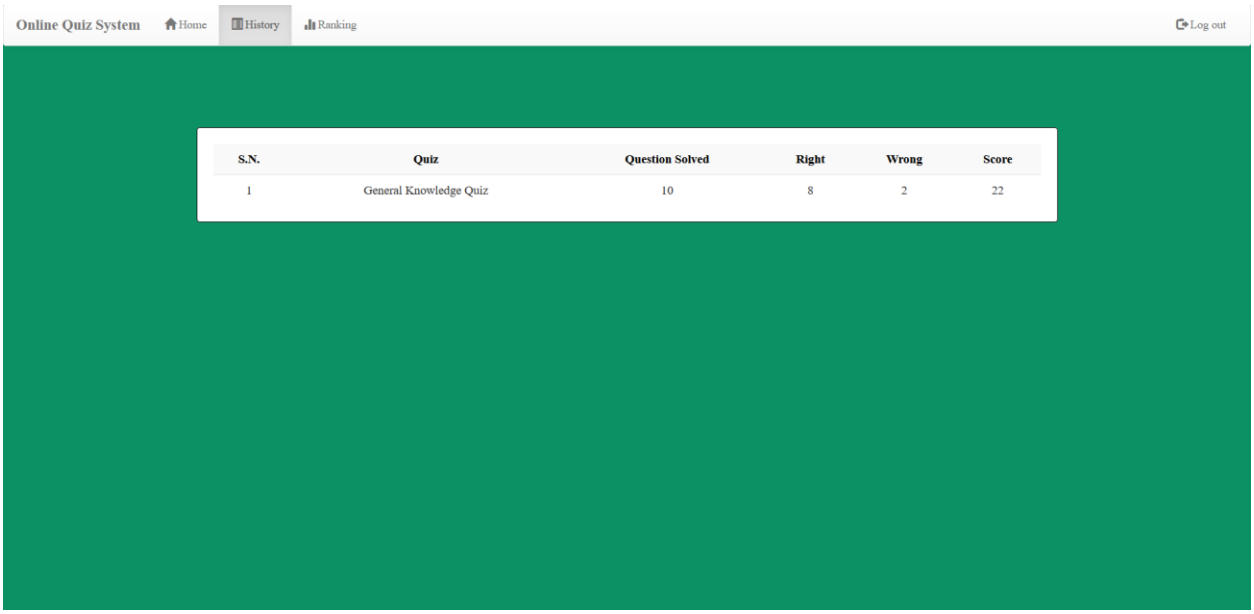
QUIZ:



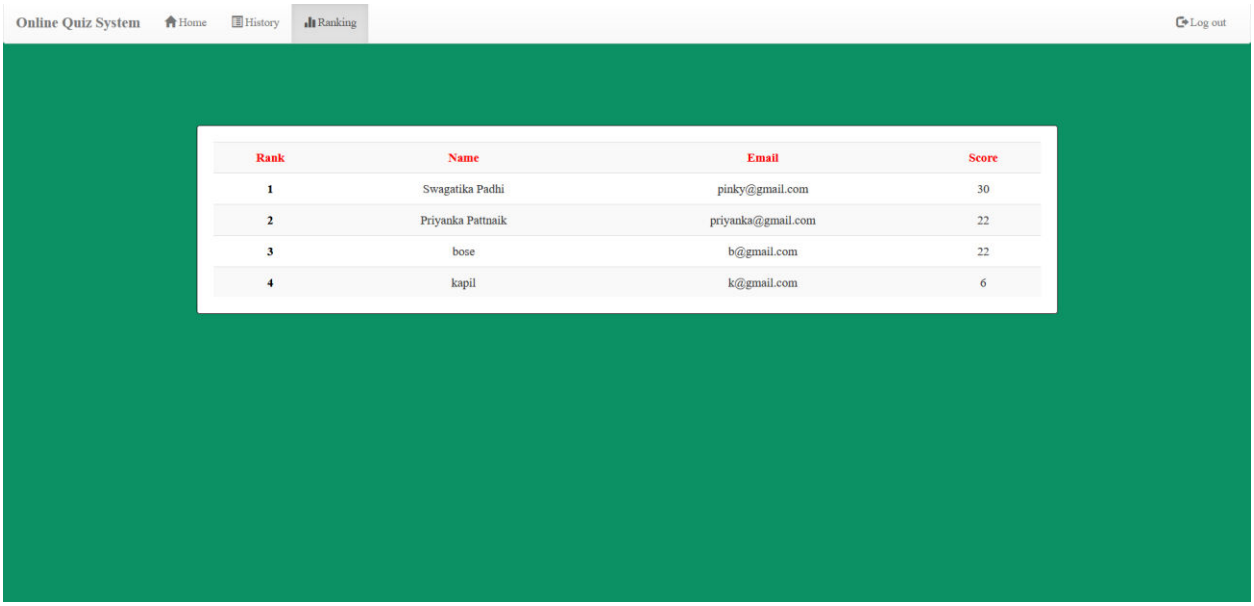
RESULT:



HISTORY:



RANKING:



8. CONCLUSION:

When we are enter the details of the patients electronically in the "Quiz Application", data will be secured. Using this application we can retrieve user's history with a single click. Thus processing information will be faster. It guarantees accurate maintenance of User details. It easily reduces the human effort and increases accuracy speed.

Quiz Application lead to a better organization structure since the information management of the users are well structured. Quiz Application can be used by Schools, Colleges and many Institutions to maintain the records easily. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project. Our project Quiz Application was developed by all five of us. We, a team of four persons took a step by step approach in order to reach our goal.

9. REFERENCES:

BOOKS:

1) **The Joy of PHP:** A Beginner's Guide to Programming Interactive Web Applications with PHP and MySQL by Alan Forbes.
2) **MySQL Explained:** Your Step By Step Guide to Database Design 2nd Edition by Mr. Andrew Comeau.
3) **HTML & CSS:** Design and Build Web Sites by Jon Duckett.
4) **JavaScript & JQuery:** Interactive Front-End Web Development by Jon Duckett.

WEB URL:

1) https://www.w3schools.com/
2) https://www.geeksforgeeks.org/
3) https://www.udemy.com/course/the-web-developer-bootcamp/
4) https://www.freecodecamp.org/news/tag/web-development/
5) https://www.edureka.co/blog/mysql-tutorial/