# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", BELAGAVI – 590018

A MINI PROJECT REPORT ON

## "DEPARTMENT QUIZ MANAGEMENT SYSTEM FOR MITM"

Submitted in partial fulfillment of requirements for the course
DBMS Laboratory with Mini Project [18CSL58] of Fifth Semester
of Bachelor of Engineering in Computer Science & Engineering
during the academic year 2021-22.

Submitted By

NITHIN U SHANKAR
4MH19CS066

PAVAN KUMAR K
4MH19CS069

Under the Guidance of

Prof. HEMANTH S R
Assistant Professor,
Dept. of CS&E.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE
Belawadi, S.R. Patna ( T ), Mandya ( D ) – 571477.

2021 – 2022

# MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna ( T ), Mandya ( D ) – 571477.

## DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

## CERTIFICATE

This is to certify that the mini project work entitled "DEPARTMENT QUIZ MANAGEMENT SYSTEM" is a bonafide work carried out by NITHIN U SHANAKR [4MH19CS066] and PAVAN KUMAR K [4MH19CS069] in partial fulfillment for the DBMS Laboratory with Mini Project (18CSL58) prescribed by the Visvesvaraya Technological University, Belagavi during the year 2021-2022 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<br>(Prof.HEMANTH S R)<br>Assistant Professor, Dept. of CS&E<br>MIT Mysore | Signature of HOD<br>(Dr.Shivamurthy R C)<br>Professor & Head, Dept. of CS&E<br>MIT Mysore |

| Name of the Examiners | Signature with date |
|---|---|
| 1.....<br>2..... | ….....<br>….. |

# ACKNOWLEDGEMENT

The completion of Project brings with and sense of satisfaction, but it is never completed without thanking the persons who are all responsible for its successful completion. First and foremost, we wish to express our deep sincere feelings of gratitude to our Institution, **MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE,** for providing us an opportunity to do our education.

We extend our deepest sense of sincere gratitude to **Dr.NARESH KUMAR**, **Principal, MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE** , for having permitted us to carry out the project work on **"DEPARTMENT MANAGEMENT QUIZ FOR MITM"**

We express our heartfelt sincere gratitude to **DR SHIVMURTHY** Head of the Department ,Computer Science and Engineering, **MAHARAJA INSTITUE OF TECHNOLOGY MYSORE**, for his valuable suggestions and support.

We express our special in-depth, heartfelt, sincere gratitude to **HEMANTH S R,** Assistant Professor, Department of Computer Science and Engineering, **MAHARAJA INSTITUE OF TECHNOLOGY MYSORE,** for their constant support in completing the project.

Finally, we would like to thank all the Teaching, Technical faculty and supporting staff members of Department of Computer Science and Engineering, **MAHARAJA INSTITUE OF TECHNOLOGY MYSORE**, for their support.

**NITHIN U SHANKAR [4MH19CS066]**

**PAVAN KUMAR K [4MH19CS069]**

# ABSTRACT

**DEPARTMENT QUIZ MANAGEMENT FOR MITM** is a web-based examination system where examinations are given online. either through the internet or intranet using computer system. The main goal of this online examination system is to effectively evaluate the student thoroughly through a totally automated system that not only reduce the required time but also obtain fast and accurate results. The existing system is in manual way the student and staff should present at the college/school for attending quiz. The Manual quiz needs pen,paper etc. The Evaluation process takes lots of time for the more number of students.

**DEPARTMENT QUIZ MANAGEMNET FOR MITM** is an online test simulator is to take online examination, test in an efficient manner and no time wasting for manually checking of the test paper. The main objective of this web based online examination system is to efficiently evaluate the student thoroughly through a fully automated system that not only saves lot of time but also gives fast and accurate results. For students they give papers according to their convenience from any location by using internet and time and there is no need of using extra thing like paper, pen etc.

# ~~~~~ TABLE OF CONTENTS ~~~~~

**CHAPTER 01**

# INTRODUCTION

## 1.1 AIM OF THE PROJECT

Designing the database which enables the Students & Teachers to register for the system. Students are allowed to take the online test and see their progress. Also, to enable the Teachers to add, delete, update thetest Questions and also to keep track of the students progress.

## 1.2 OVERVIEW OF THE PROJECT

Proposed System: In this project, we are designing a database that can be used to insert, retrieve, update and perform additional operations.

General Description of Inputs & Outputs: The system will be getting input from the Users from various Pages. The output also given bythe system depending on input given by the User.

Continuous updating of database: Database is updated from time to time after each effect on the proposal. So data is more accurate and perfect as all the updation is done simultaneously. The database will be updated by the Teacher in Timely manner.

Instantaneous retrieval of data: As the database is update from timeto time the data can updated any time by Teachers and Students can take exam any time.

The Main Two modules in the System are:-

- Student Module.
- Staff Module.

## 1.3 OUTCOME OF THE PROJECT

From this project student will be able to attend Any quiz given by the department/Staffs anywhere by online mode .There will be no need of physical Comtact.. Staff can add the quiz anytime they want and instruct students to Take the quiz and Once the quiz is made the project will automatically evalute so the time of evalution is also Reduced or even vanished.

## 1.4 SOFTWARE REQUIREMENTS

| | | |
|---|---|---|
| Programming language | : | PHP, MYSQL |
| Operating system | : | ANY OS (Recommended : Windows10) |
| Application required | : | XAMPP |
| Coding language | : | HTML,CSS,JAVASCRPIT |

## CHAPTER 02

# DESIGN

## 2.1 SCHEMA DIAGRAM

The design of the database is called a schema. This tells us about the structural view of the database. It gives us an overall description of the database. A database schema defines how the data is organized using the schema diagram.

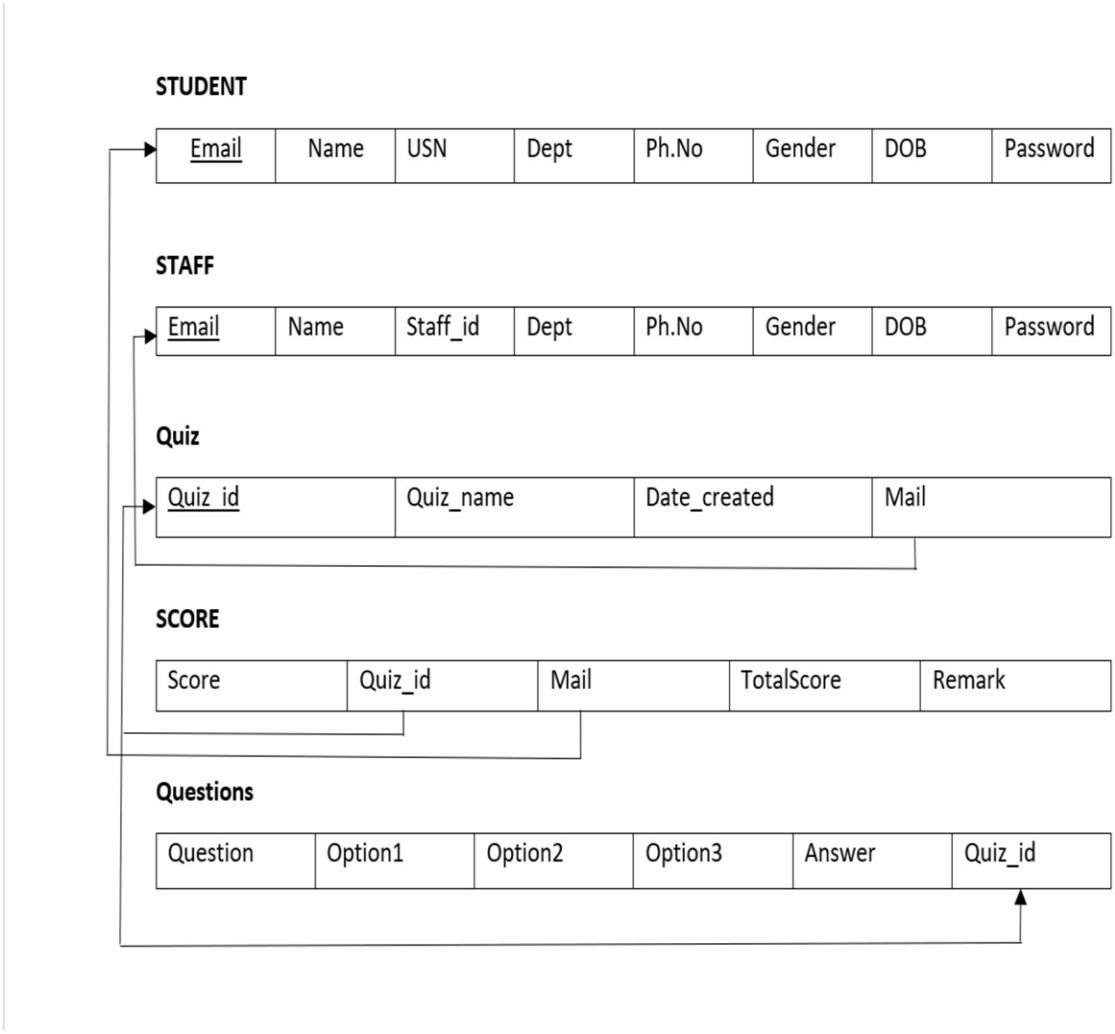


**Fig 2.1**:Schema Diagram contains the description of Data base that is being used for Department Quiz Management System for MITM

## 2.2 E-R DIAGRAM

ER-Diagram is a pictorial representation of data that describes how data is communicated and related to each other. Any object, such as entities, attributes of and identity, sets of relationship, and other attributes of relationship, can be characterized with the help of the ER diagram.

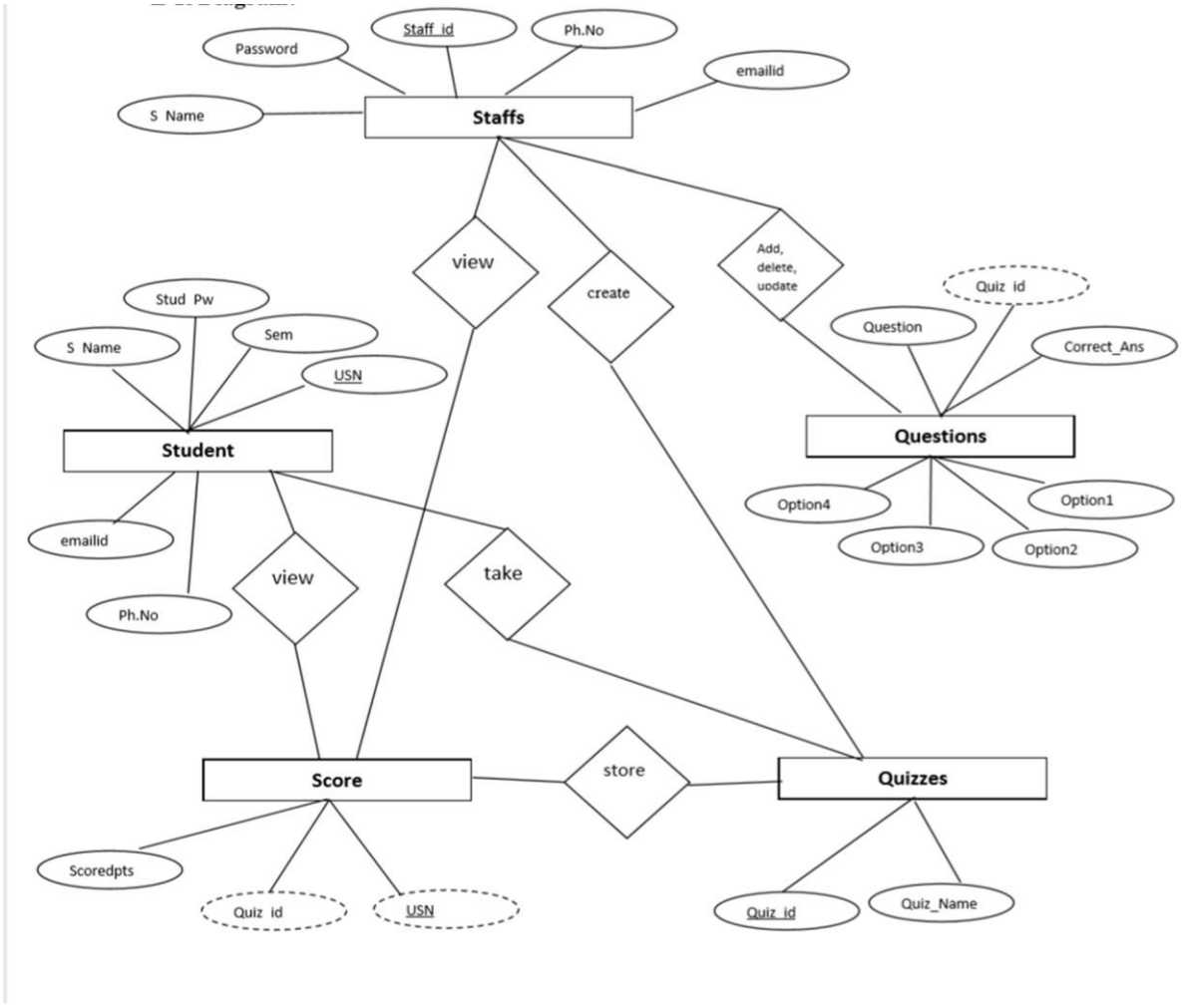


**Fig 2.2 :** Entity Relation diagram for Department Quiz Management System for MITM contains the relations that are required to connect on table to another table or the relations that are having between the table.

## 2.3 SEQUENCE DIAGRAM

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

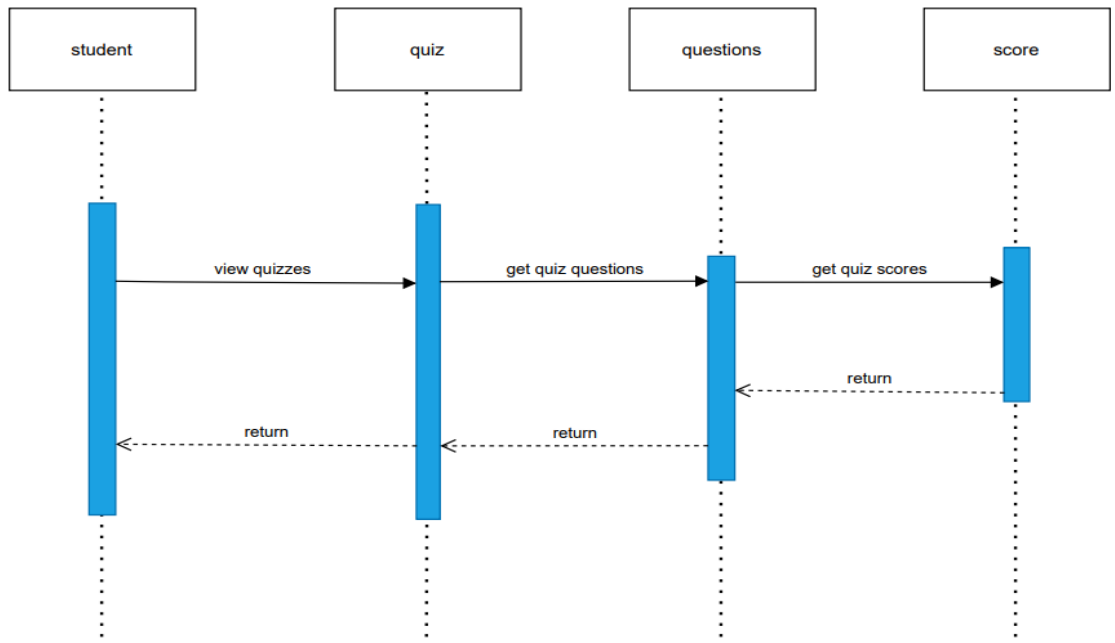


**Fig 2.3** : Sequence Diagram for Department Quiz Management System for MITM contains the description of the way in which the user can interact with the System.

## 2.4 DATAFLOW DIAGRAM

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both. It shows how data enters and leaves the system, what changes the information, and where data is stored. The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

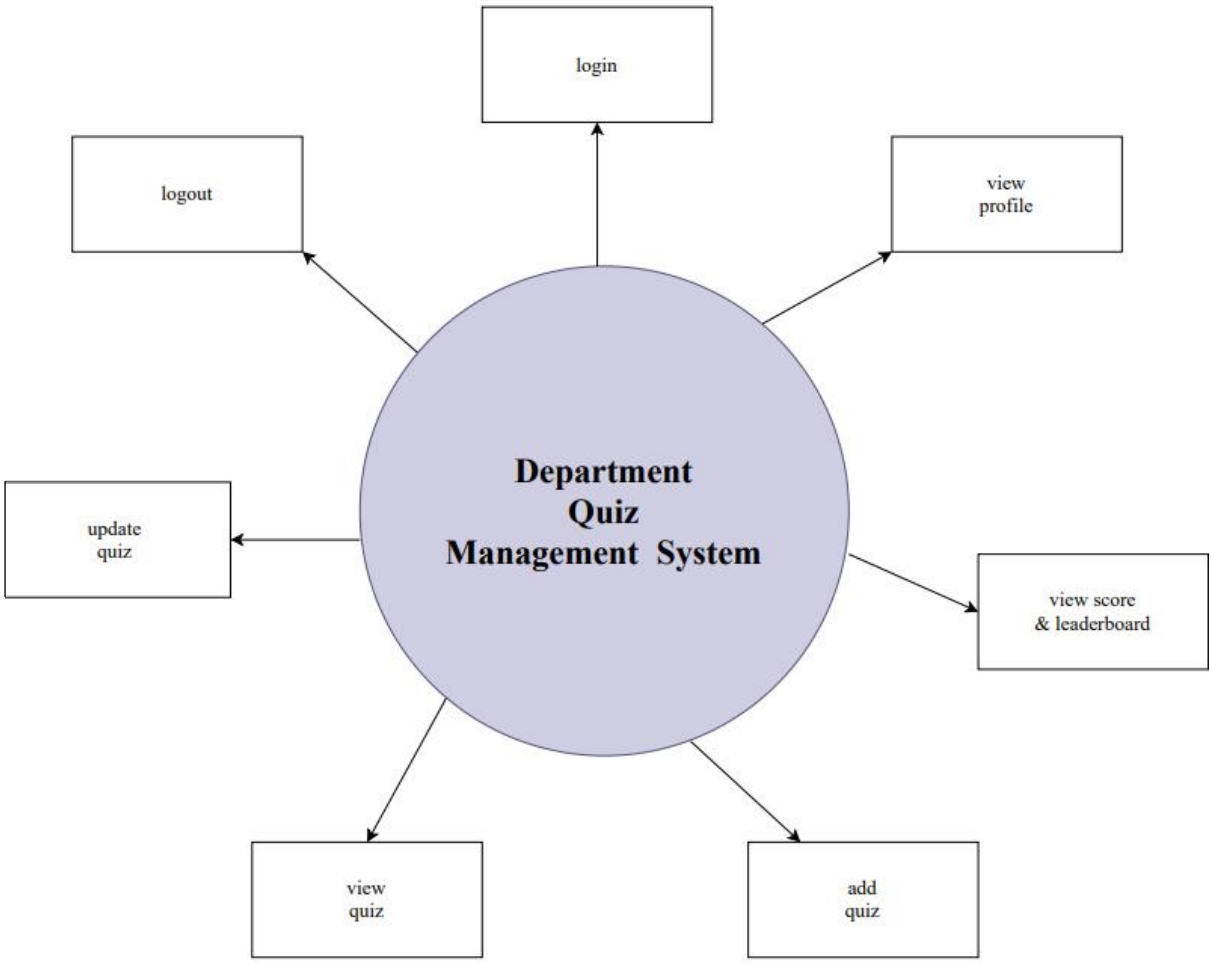


**Fig 2.4** Data Flow Diagram contains the information about the flow in which the Department Quiz Management System for MITM works.

## 2.5 USECASE DIAGRAM

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system

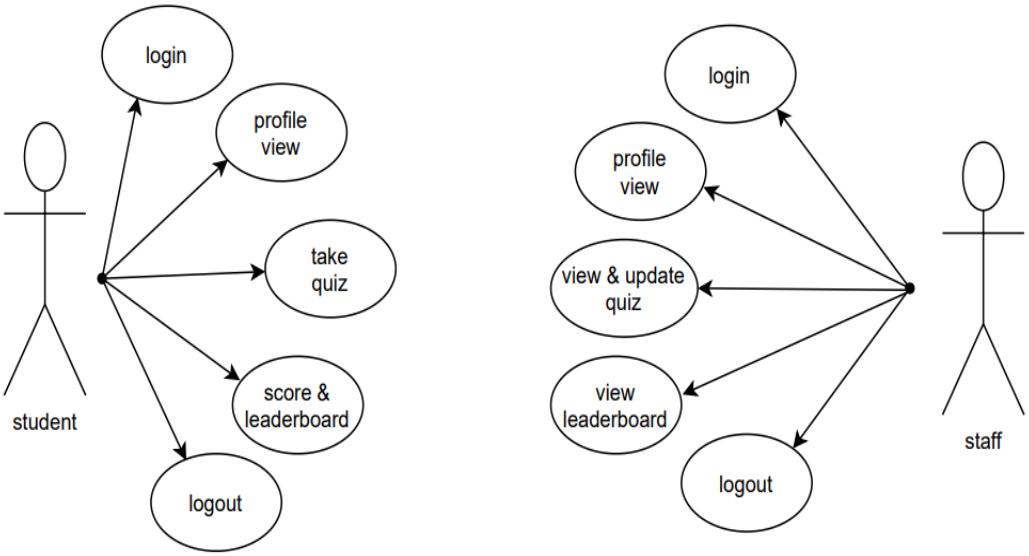


**Fig 2.5** :Use Case Diagram contains the working procedure that take place when it comes to Staff and the Student.

## CHAPTER 03

# IMPLEMENTATION

## 3.1 DESCRPTIONS OF TABLES:-

### Table 3.1.1: Student

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| usn | varchar(10) | NO | UNI | *NULL* | |
| name | varchar(20) | NO | | *NULL* | |
| mail | varchar(30) | NO | PRI | *NULL* | |
| phno | varchar(10) | NO | UNI | *NULL* | |
| gender | varchar(1) | NO | | *NULL* | |
| DOB | varchar(10) | NO | | *NULL* | |
| pw | varchar(200) | NO | | *NULL* | |
| dept | varchar(3) | YES | MUL | *NULL* | |

The above table contains the necessary information of the student that is required when they singup and all the information is displayed in Student profile.

### Table 3.1.2: Staff

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| staffid | varchar(10) | NO | UNI | *NULL* | |
| name | varchar(20) | NO | | *NULL* | |
| mail | varchar(30) | NO | PRI | *NULL* | |
| phno | varchar(10) | NO | | *NULL* | |
| gender | varchar(1) | NO | | *NULL* | |
| DOB | varchar(10) | NO | | *NULL* | |
| pw | varchar(200) | NO | | *NULL* | |
| dept | varchar(3) | YES | | *NULL* | |

The above table contains the necessary information of the staff that is required when they singup and all the information is displayed in Staff profile.

### Table 3.1.3: Score

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| slno | int(11) | NO | PRI | *NULL* | auto_increment |
| score | int(11) | NO | | *NULL* | |
| quizid | int(11) | NO | MUL | *NULL* | |
| mail | varchar(30) | YES | MUL | *NULL* | |
| totalscore | int(11) | YES | | *NULL* | |
| remark | varchar(20) | YES | | *NULL* | |

The above table contains the description of the Score that the Student as scored after attending the quiz and will be updated according to the Student Rank and slno is auto_incremented.

### Table 3.1.4: Department

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| dept_id | int(11) | NO | PRI | *NULL* | |
| dept_name | varchar(3) | YES | | *NULL* | |

The above table contains the description of the Department.It contains the department name and department id.

### Table 3.1.5: Questions

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| qs | varchar(200) | NO | PRI | *NULL* | |
| op1 | varchar(30) | NO | | *NULL* | |
| op2 | varchar(30) | NO | | *NULL* | |
| op3 | varchar(30) | NO | | *NULL* | |
| answer | varchar(30) | NO | | *NULL* | |
| quizid | int(11) | NO | MUL | *NULL* | |

This table contains the description of the questions and their option and the correct answer that are given by the Staff.

**Table 3.1.6: Quiz**

| Field | Type | Null | Key | Default | Extra |
|---|---|---|---|---|---|
| quizid | int(11) | NO | PRI | *NULL* | auto_increment |
| quizname | varchar(20) | NO | | *NULL* | |
| date_created | timestamp | NO | | current_timestamp() | |
| mail | varchar(30) | YES | MUL | *NULL* | |

The above table contains the complete description of the total Quiz that the Staff has given . date will be updated according to the current time of the device.

## 3.2 Constraints On Tables.

1. In table student, Field usn and phno is unique Key , mail is primary key and dept is mulivalued key.
2. In table staff, Field staffed is Unique key and mani is primary key.
3. In table score,Field Slno is Primary Key and quizid, mail is Multivalued key.
4. In table dept , Filed dept_id is primary key.
5. In Table Questions Field qs is primary key and quizid is multivalued key.
6. In Table quiz Field quizid is primary key and mail is multivalued key.

## 3.3 Backend Implementation.

```
-- phpMyAdmin SQL Dump
-- version 4.1.14
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Nov 20, 2019 at 04:29 PM
-- Server version: 5.6.17
-- PHP Version: 5.5.12
```

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101
SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101
SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101
SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
--
-- Database: `project`
--
DELIMITER $$
--
-- Procedures
--
CREATE DEFINER=`root`@`localhost` PROCEDURE `leaderboard`()
   NO SQL
select q.quizname,s.score,s.totalscore,st.name,s.mail from score s,student st,quiz q where s.mail=st.mail and q.quizid=s.quizid order by score DESC$$

DELIMITER ;
-- --------------------------------------------------------
--
-- Table structure for table `dept`
--
CREATE TABLE IF NOT EXISTS `dept` (
                                        `dept_id` int(11) NOT NULL,
                                          `dept_name` varchar(3) DEFAULT NULL,
                                          PRIMARY KEY (`dept_id`)
                                          )
 ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `dept`
--
INSERT INTO `dept` (`dept_id`, `dept_name`) VALUES
(1, 'CSE'),
(2, 'ISE'),
(3, 'ECE'),
(4, 'CIV'),
```

```
(5, 'MEC');

-- --------------------------------------------------------

--

-- Table structure for table `questions`

--

CREATE TABLE IF NOT EXISTS `questions` (

  `qs` varchar(200) NOT NULL,

  `op1` varchar(30) NOT NULL,

  `op2` varchar(30) NOT NULL,

  `op3` varchar(30) NOT NULL,

  `answer` varchar(30) NOT NULL,

  `quizid` int(11) NOT NULL,

  UNIQUE KEY `qs` (`qs`),

  KEY `quizid` (`quizid`),

  KEY `quizid_2` (`quizid`),

  KEY `quizid_3` (`quizid`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `questions`

--

INSERT INTO `questions` (`qs`, `op1`, `op2`, `op3`, `answer`, `quizid`) VALUES

('/* Assume that integers take 4 bytes.<br>  #include<iostream> <br>  using namespace std; <br> class Test  { <br>  static int i;<br>  int j;<br> }; <br>  int Test::i; <br>  int main() { ', '1', '2', '3', '4', 5),

('DBMS FUll form', 'Database multiple system','DataBase Mutual system','None Of the Above','DataBase Management System',01);

-- --------------------------------------------------------

--

-- Table structure for table `quiz`

--

CREATE TABLE IF NOT EXISTS `quiz` (

  `quizid` int(11) NOT NULL AUTO_INCREMENT,

  `quizname` varchar(20) NOT NULL,

  `date_created` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,

  `mail` varchar(30) DEFAULT NULL,

  PRIMARY KEY (`quizid`),

  KEY `mail` (`mail`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;

--

-- Dumping data for table `quiz`

--

INSERT INTO `quiz` (`quizid`, `quizname`, `date_created`, `mail`) VALUES
```

```
(1, 'DBMS', '2022-01-30 10:36:29', 'nithinushankar@gmail.com');
--
-- Triggers `quiz`
--
DROP TRIGGER IF EXISTS `ondeleteqs`;
DELIMITER //
CREATE TRIGGER `ondeleteqs` AFTER DELETE ON `quiz`
 FOR EACH ROW delete from questions where questions.quizid=old.quizid
//
DELIMITER ;
-- --------------------------------------------------------
--
-- Table structure for table `score`
--
CREATE TABLE IF NOT EXISTS `score` (
  `slno` int(11) NOT NULL AUTO_INCREMENT,
  `score` int(11) NOT NULL,
  `quizid` int(11) NOT NULL,
  `mail` varchar(30) DEFAULT NULL,
  `totalscore` int(11) DEFAULT NULL,
  `remark` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`slno`),
  KEY `quizid` (`quizid`),
  KEY `mail` (`mail`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
--
-- Dumping data for table `score`
INSERT INTO `score` (`slno`, `score`, `quizid`, `mail`, `totalscore`, `remark`) VALUES(01, 1, 1,
'nithinushankar@gmail.com', 1, 'good');
--
-- Triggers `score`
--
DROP TRIGGER IF EXISTS `remarks`;
DELIMITER //
CREATE TRIGGER `remarks` BEFORE INSERT ON `score`
 FOR EACH ROW set NEW.remark = if(NEW.score = 0, 'bad', 'good')
//
DELIMITER ;
-- --------------------------------------------------------
--
-- Table structure for table `staff`
```

```
--
CREATE TABLE IF NOT EXISTS `staff` (
  `staffid` varchar(10) NOT NULL,
  `name` varchar(20) NOT NULL,
  `mail` varchar(30) NOT NULL,
  `phno` varchar(10) NOT NULL,
  `gender` varchar(1) NOT NULL,
  `DOB` varchar(10) NOT NULL,
  `pw` varchar(200) NOT NULL,
  `dept` varchar(3) DEFAULT NULL,
  PRIMARY KEY (`mail`),
  UNIQUE KEY `mail` (`mail`,`phno`),
  UNIQUE KEY `staffid` (`staffid`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `staff`
--

INSERT INTO `staff` (`staffid`, `name`, `mail`, `phno`, `gender`, `DOB`, `pw`, `dept`) VALUES('01', 'Nithin',
'nithinushankar@gmail.com', '9380732272', 'M', '1990-05-04', 'Nithin', 'CSE'),
-- --------------------------------------------------------
--
-- Table structure for table `student`
--
CREATE TABLE IF NOT EXISTS `student` (
  `usn` varchar(10) NOT NULL,
  `name` varchar(20) NOT NULL,
  `mail` varchar(30) NOT NULL,
  `phno` varchar(10) NOT NULL,
  `gender` varchar(1) NOT NULL,
  `DOB` varchar(10) NOT NULL,
  `pw` varchar(200) NOT NULL,
  `dept` varchar(3) DEFAULT NULL,
  PRIMARY KEY (`mail`),
  UNIQUE KEY `mail` (`mail`),
  UNIQUE KEY `phno` (`phno`),
  UNIQUE KEY `usn` (`usn`),
  KEY `dept` (`dept`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Dumping data for table `student`
```

```
--
INSERT INTO `student` (`usn`, `name`, `mail`, `phno`, `gender`, `DOB`, `pw`, `dept`) VALUES('4MH19CS066', 'NITHIN U SHNAKAR', 'nithin@gmail.com', '8073362805', 'M', '2002-03-05', 'Nithinn', 'CSE');
--
-- Constraints for dumped tables
--
--
-- Constraints for table `quiz`
--
ALTER TABLE `quiz`
 ADD CONSTRAINT `quiz_ibfk_1` FOREIGN KEY (`mail`) REFERENCES `staff` (`mail`) ON DELETE CASCADE;
--
-- Constraints for table `score`
--
ALTER TABLE `score`
 ADD CONSTRAINT `score_ibfk_1` FOREIGN KEY (`quizid`) REFERENCES `quiz` (`quizid`) ON DELETE CASCADE,
 ADD CONSTRAINT `score_ibfk_2` FOREIGN KEY (`mail`) REFERENCES `student` (`mail`) ON DELETE CASCADE ON UPDATE CASCADE;
/*!40101
SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101
 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101
 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## 3.4 Frontend Implementation:-

### 3.4.1Login:-

```
<?php
      if (isset($_POST['login'])) {
     if (isset($_POST['usertype']) && isset($_POST['username']) && isset($_POST['pass']))
  {
            require_once 'sql.php';
           $conn = mysqli_connect($host, $user, $ps, $project);if (!$conn) {
             echo "<script>alert(\"Database error retry after some time !\")</script>";
  }
           $type = mysqli_real_escape_string($conn, $_POST['usertype']);
           $username = mysqli_real_escape_string($conn, $_POST['username']);
```

```
        $password = mysqli_real_escape_string($conn, $_POST['pass']);
        $sql = "select * from " . $type . " where mail='{$username}'";
        $res =   mysqli_query($conn, $sql);
        if ($res == true)
    {
          global $dbmail, $dbpw;
          while ($row = mysqli_fetch_array($res)) {
            $dbpw = $row['pw'];
            $dbmail = $row['mail'];
            $_SESSION["name"] = $row['name'];
            $_SESSION["type"] = $type;
            $_SESSION["username"] = $dbmail;
      }
          if ($dbpw === $password) {
            if ($type === 'student') {
              header("location:homestud.php");
            } elseif ($type === 'staff') {
              header("Location: homestaff.php");
            }
          } elseif ($dbpw !== $password && $dbmail === $username) {
            echo "<script>alert('password is wrong');</script>";
          } elseif ($dbpw !== $password && $dbmail !== $username) {
            echo "<script>alert('username name not found sing up');</script>";
          }
        }
      }
    }
    ?>
```

## 3.4.2Add Question By staff:-

```
<?php
session_start();
require_once 'sql.php';
        $conn = mysqli_connect($host, $user, $ps, $project);if (!$conn) {
  $qname = $_SESSION['qname'];
  $sql = "select quizid from quiz where quizname='{$qname}'";
  $res =   mysqli_query($conn, $sql);
  if ($res == true) {
    global $qid;
    while ($row = mysqli_fetch_array($res)) {
```

```
            $qid = $row['quizid'];
        }
    }
    if (isset($_POST['submit'])) {
        $qs = $_POST["qs"];
        $op1 = $_POST["op1"];
        $op2 = $_POST["op2"];
        $op3 = $_POST["op3"];
        $ans = $_POST["ans"];

$sql="insertintoquestions(qs,op1,op2,op3,answer,quizid)values('$qs','$op1','$op2','$op3','$ans','$
qid');";
        $res =   mysqli_query($conn, $sql);
        if ($res == true) {
            echo '<script>history.pushState({}, "", "");</script>';
        } elseif ($res != true) {
            echo '<script>alert("Question already exsits");</script>';
        }
    }
    if (isset($_POST['submit1'])) {
        $qs = $_POST["qs"];
        $op1 = $_POST["op1"];
        $op2 = $_POST["op2"];
        $op3 = $_POST["op3"];
        $ans = $_POST["ans"];
$sql="insertintoquestions(qs,op1,op2,op3,answer,quizid)values('$qs','$op1','$op2','$op3','$ans','$
qid');";
        if ($res == true) {
            header("Location: homestaff.php");
        } elseif ($res != true) {
            echo '<script>alert("Question already exsits");</script>';
        }
    }
}
?>
```

## 3.4.3 View Question:-

```
<?php
        if(isset($_GET["qid"])){
        $qid=$_GET["qid"];
            $sql ="select * from questions where quizid='{$qid}'";
```

```
        $res=mysqli_query($conn,$sql);
        if($res)
        {
           $count=mysqli_num_rows($res);
           if(mysqli_num_rows($res)==0)
           {
              echo "No questions found under this quiz please come later";
              echo "<form method=\"POST\">";
           echo "<input id=\"btn\" type=\"submit\" name=\"submit\" value=\"Add
Questions\"><br><br><br>";
           }else{
           $i=1;
           $j=0;
           echo "<form method=\"POST\">";
           echo "<input id=\"btn\" type=\"submit\" name=\"submit\" value=\"Add
Questions\"><br><br><br>";
           echo "</form><br><br>";
           while ($row = mysqli_fetch_assoc($res)) {
              echo $i.". ".$row["qs"]."<br>";
              echo"<inputtype=\"radio\"value=\"".$j."\"name=\"ans".$i.$j."\">".$row["op1"]."<br>"
echo"<inputtype=\"radio\"value=\"".($j+1)."\"name=\"ans".$i.$j."\">".$row["op2"]."<br>";
echo"<inputtype=\"radio\"value=\"".($j+2)."\"name=\"ans".$i.$j."\">".$row["op3"]."<br>";
echo"<inputtype=\"radio\"value=\"".($j+3)."\"name=\"ans".$i.$j."\">".$row["answer"]."<br><br>";
              $i++;
           }
           echo "</form><br><br>";
        }
        }
        else
        {
           echo "error".mysqli_error($conn).".";
        }
        if(isset($_POST["submit"])){
           echo "<script>window.location.replace(\"addq.php?qid=".$qid."\")</script>";
        }
    } ?>
<?php
"}</script>";
?>
```

### 3.4.4 Take Quiz By Student

```
<?php
 if(isset($_GET["qid"])){
    $qid=$_GET["qid"];
    $sql ="select * from questions where quizid='{$qid}'";
    $res=mysqli_query($conn,$sql);
            $answers_index_arr = [];
    if($res)
    {
            $count=mysqli_num_rows($res);
            if(mysqli_num_rows($res)==0)
            {
                    echo "No questions found under this quiz please come later";
            }
                    else{
                    $i=1;
                    $j=0;
            echo "<form method=\"POST\">";
            while ($row = mysqli_fetch_assoc($res)) {
                            $q_options_arr =
[$row["op1"],$row["op2"],$row["op3"],$row["answer"]];
                            shuffle($q_options_arr);
                            $answer_index=
array_search($row["answer"],$q_options_arr);
                            array_push($answers_index_arr,$answer_index);
                    echo $i.". ".$row["qs"]."<br>";
                    echo "<input type=\"radio\" value=\"".$j."\"
name=\"ans".$i.$j."\">".$q_options_arr[0]."<br>";
                    echo "<input type=\"radio\" value=\"".($j+1)."\"
name=\"ans".$i.$j."\">".$q_options_arr[1]."<br>";
                    echo "<input type=\"radio\"
value=\"".($j+2)."\"name=\"ans".$i.$j."\">".$q_options_arr[2]."<br>";
                    echo "<input type=\"radio\"value=\"".($j+3)."\"
name=\"ans".$i.$j."\">".$q_options_arr[3]."<br><br>";
                    $i++;


            }


            echo "<input id=\"btn\" type=\"submit\" name=\"submit\"
value=\"submit\"><br><br><br>";
                    foreach($answers_index_arr as $value){
             echo "<input type='hidden' name='answers_index_arr[]' value='".$value."' />";
```

```
                        }
                    echo "</form><br><br>";
            }
            }
            else
            {
                echo "error".mysqli_error($conn).".";
            }
            if(isset($_POST["submit"])){
                $score=0;
             for($i=1;$i<=$count;$i++)
                {
                    if(isset($_POST["ans".$i.$j]) &&
($_POST["ans".$i.$j]==$_POST['answers_index_arr'][($i-1)])){
                        $score++;
                    }
                }
                echo "<script>alert(\"u scored ".$score." out of ".$count."\");</script>";
                $sql ="insert into score(score,mail,quizid,totalscore)
values('$score','$dbmail','$qid','$count');";
                $res=mysqli_query($conn,$sql);
                if($res)
                {
                    echo '<script>history.pushState({}, "", "");</script>';
                    echo "<script>window.location.replace(\"homestud.php\");</script>";
                }
                        else{
                    echo "<script>alert(\"error occured updating score in
database".mysqli_error($conn)."\");</script>";
                } }   } ?
```

## Chapter 04

# 4.SNAPSHOTS AND DISCUSSION

## 4.1 SNAPSHOTS

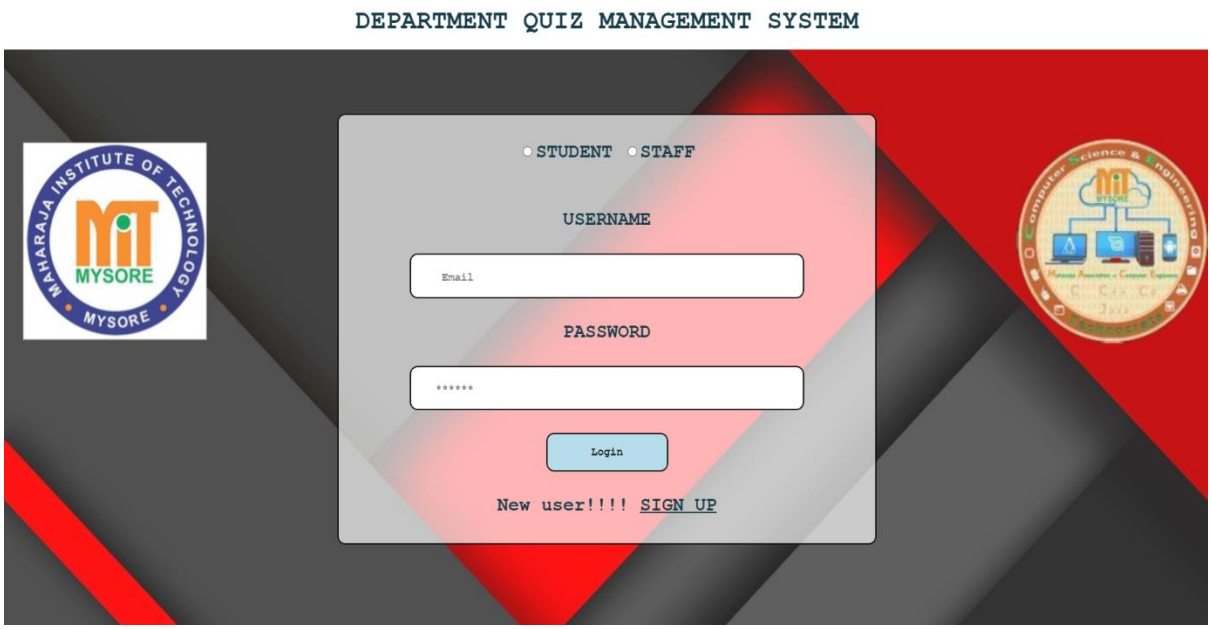


**Fig 4.1.1:** LOGIN PAGE FOR BOTH STUDENT AND TEACHER

Staff and Student can login to the system using the Credentials i.e mail and password.

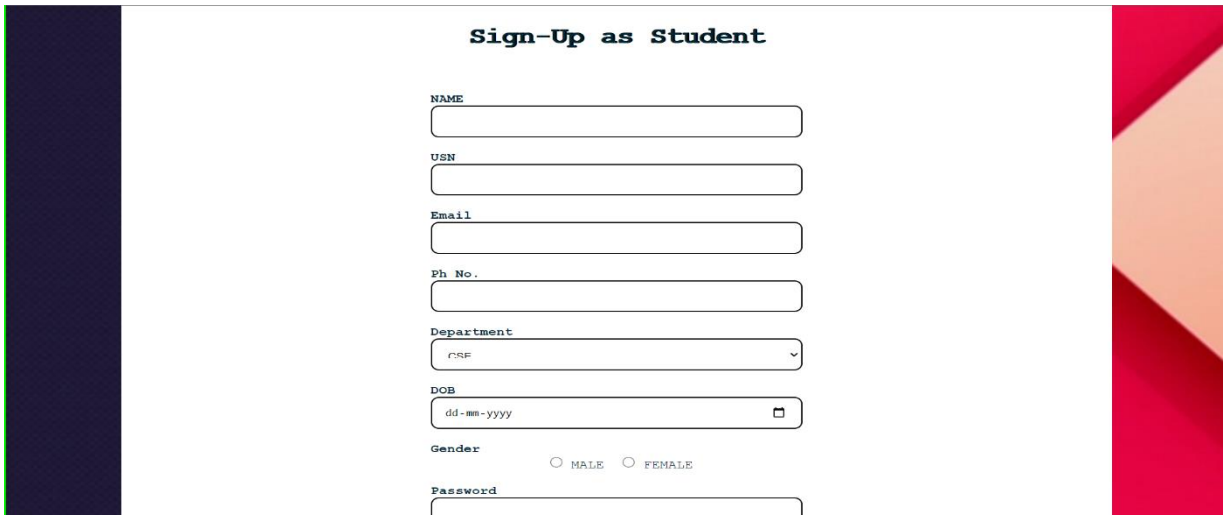


**Fig 4.1.2:** SIGN-UP FOR STUDENT

The New Student can sign up by entering the details Name,USN,Email,Ph.no,Department DOB,Gender,Password,Confirm Password then submit.By providing these information student can register.


**Fig 4.1.3:** SIGN-UP PAGE FOR STAFF

The New Staff can sign up by entering the details Name,Staff Id,Email,Ph.no,Department DOB,Gender,Password,Confirm Password then submit.By providing these information staff can register.

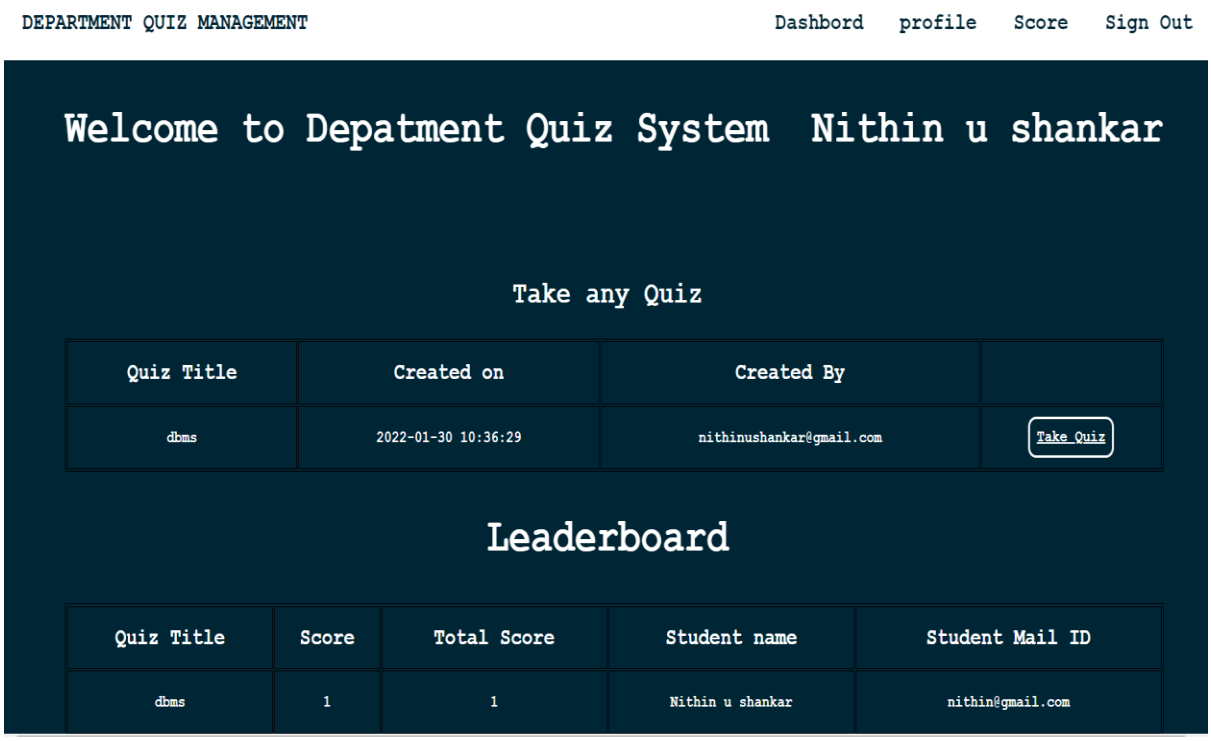


**Fig 4.1.4 :** STUDENT DASHBOARD

Here the Student Can take the Quiz and see the Score and leadbooard for Each Quiz.

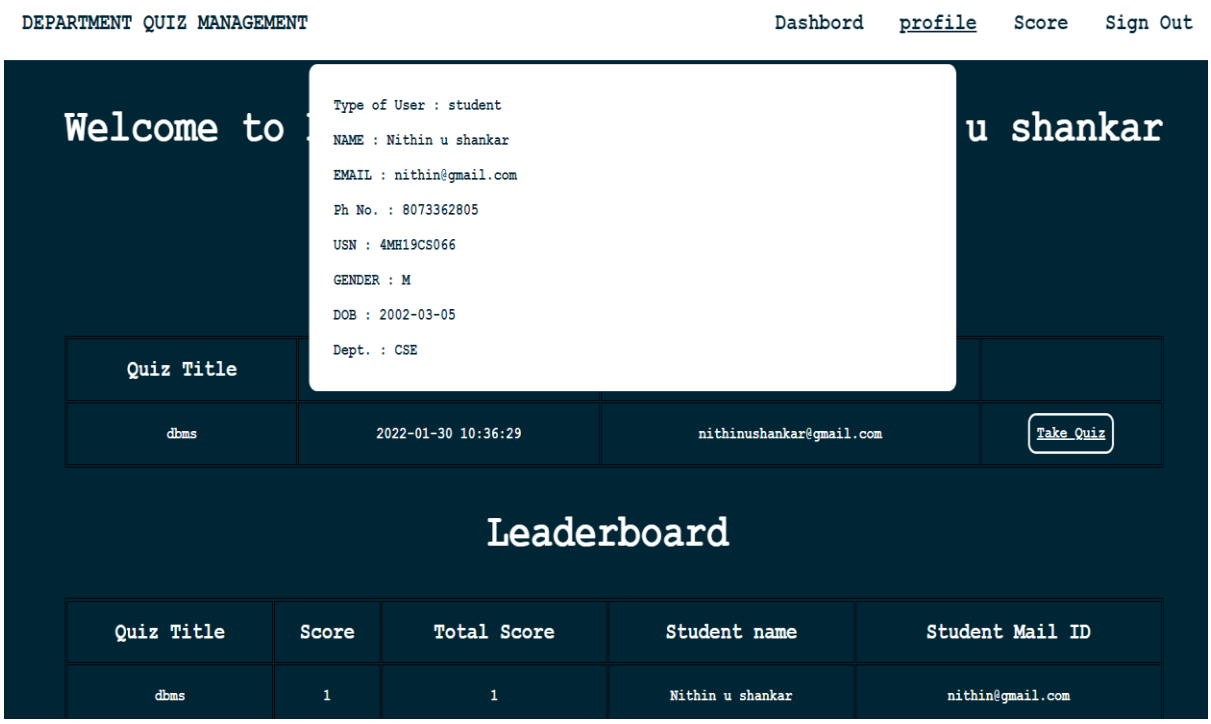


**Fig 4.1.5** :STUDENT PROFILE DISPLAY

In this page all the student details given during the signup is shown.


**Fig 4.1.6 :** TAKING QUIZ BY STUDENT

In this page user is allowed to take the quiz questions and answer the quiz and After finishing the quiz Just click Submit button.

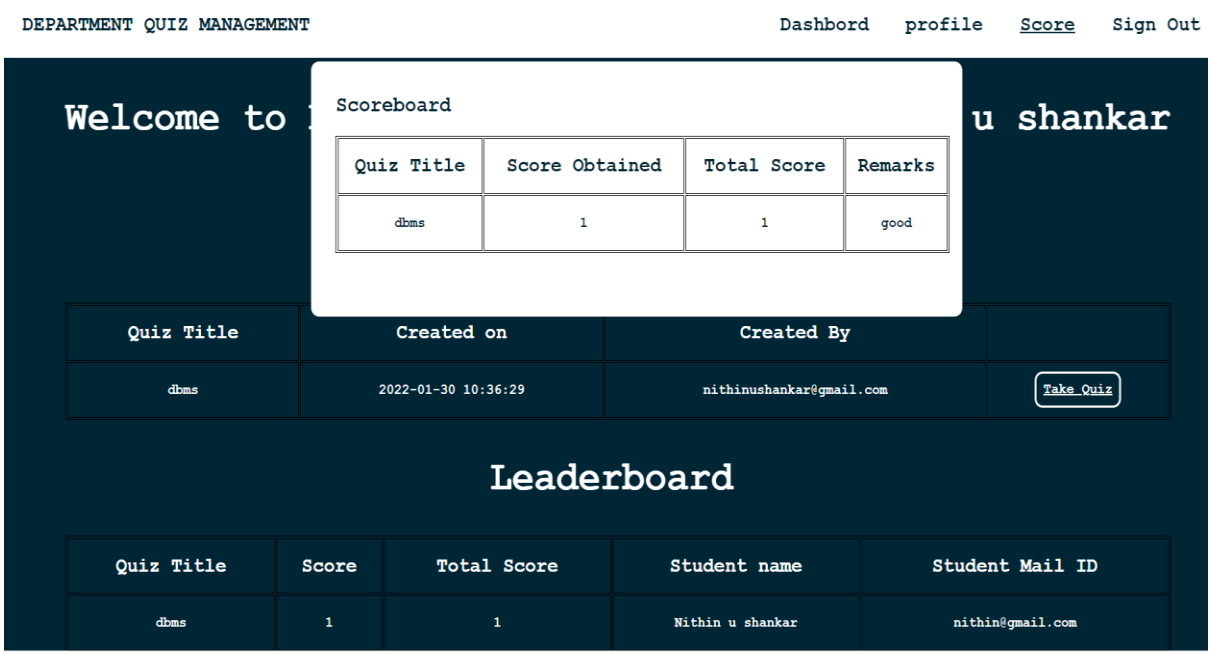


**Fig 4.1.7 :** SCORECARD DISPLAY.

In this page the Student can view the result of the Quiz that he had attended and remarks for the scores.

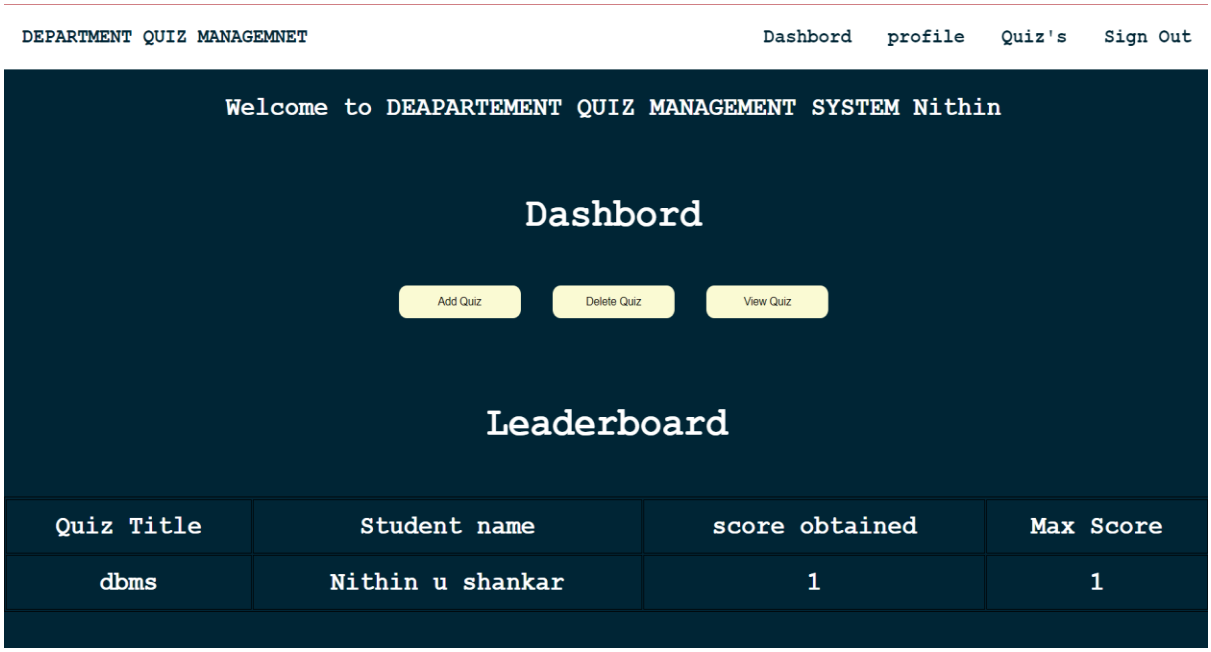


**Fig 4.1.8 :** STAFF DASHBOARD

Here the Staff Can Add quiz , Delete quiz and View Quiz and leadboard of the syudent who have attended the quiz.

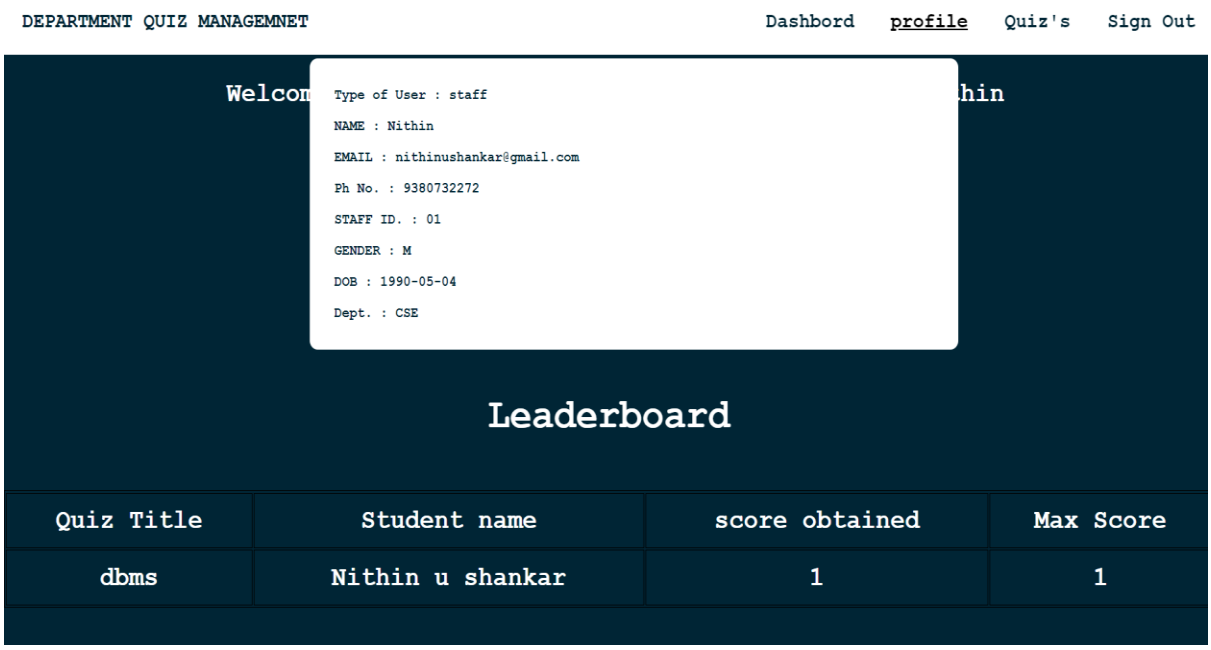


**Fig 4.1.9 :** STAFF PROFILE DISPLAY

In this page all the staff details given during the signup is shown.

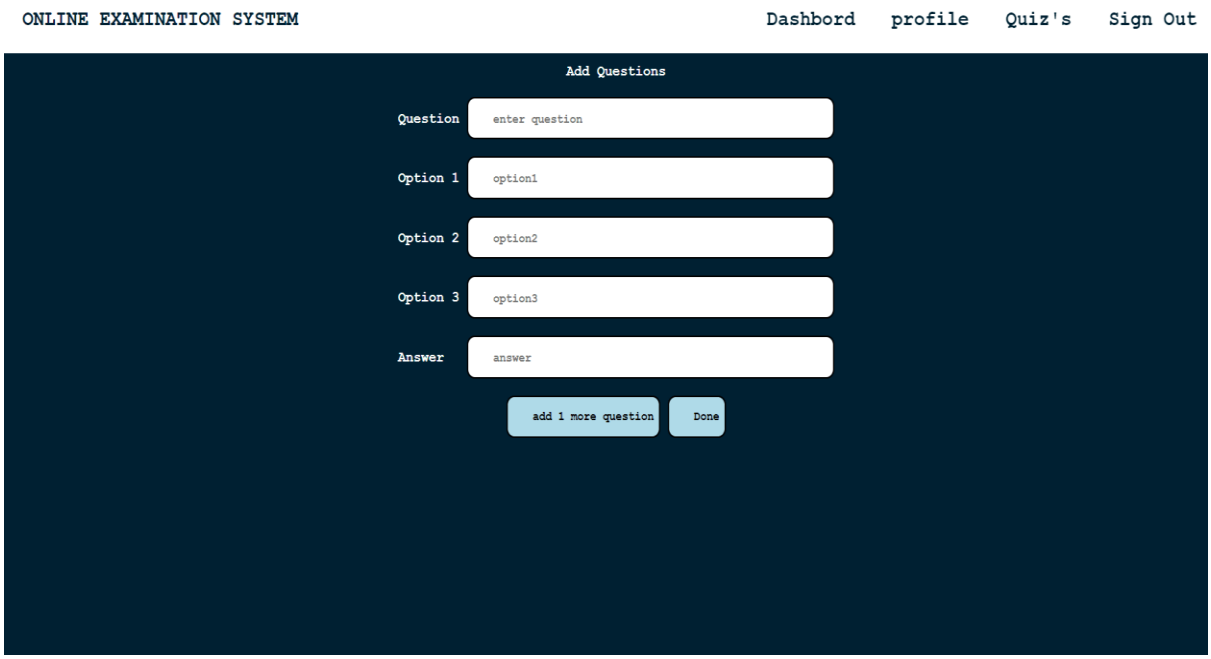


**Fig 4.1.10:** ADDING QUESTION BY STAFF.

In this page Staff can add the question to a specified Quiz and the Options and the Correct answer.

## 4.2 DISCUSSION

Department Quiz management system is that which enables the Students & Teachers to register for the system. Students are allowed to take the online test and see their progress. Also, it enables the Teachers to add, delete, update the test questions and also to keep track of the student progress.

- Student.
- Staff.
- Score.
- Department.
- Questions.
- Quiz.

### 4.2.1. Student:

Student table has Fields like usn with Unique key, name field , mail field as Primary key , phno[Phone Number] Field as a unique key ,gender Field ,DOB[Date of the birth] Field, pw[Pasword]Field , dept[Department] Field as multivalued key. Student has to log in to the system and can then view all the quizzes. Student can see the quiz list and attend the quizzes. After attending the quizzes students will get instant result and the same will be stored in the database.

### 4.2.2. Staff:

Staff Table has Fields like staffid with unique key, name field , mail field as Primary key ,phno[Phone Number]Field as a unique key ,gender Field ,dob[Date of Birth] Field, pw[Password] Field , dept[Department] Field . Staff also has to log in to the system first, then they can add/remove quizzes. They can see the progress and they can also update the existing quiz.

### 4.2.3.Score:

Score Table has Fields like slno[Sl Number] with Primary key, score field , quizid field as Multivalued key , mail Field as a Multivalued key ,totalscore Field ,remark Field. Staff can see the scoreboard of the quiz which is added by him/her. And student can see the score of the quiz which he/she is attended.

### 4.2.4.Department:-

dept] Departement Table has fields like dept_id[Department Id] Field as Primary key ,dept_name[Department Name] Field. In this table we have Stored all the Department Info and all Department Id Information.

### 4.2.5.Questions:-

Questions Table Contains Fields Like qs[Questions] field as Primary Value,op1[Option 1 ] Field, op2[Option 2 ] Field, op3[Option 3 ] Field, ans[Answer ] Field,and quizid Filed as Multivalued key. Here we store all the questions,options and answers updated by the staff.

### 4.2.6. QUIZ :-

Quiz Table Contains Fields like quizid field as Primary key and also autoincrement,quizname Field, date_created Field with Current_timestamp,and a mail Field. Here it stores all the Information Realted to a quiz

## 4.3 TESTING

| Test Case Id | Test Case | Expected Output | Actual output | Status |
|---|---|---|---|---|
| TC1 | Login of both student and staff with wrong email address. | Username name not found sign up. | As expected | PASS |

| | | | | |
|---|---|---|---|---|
| TC2 | Login of both student and staff with wrong Password . | Password is wrong. | As expected | PASS |
| TC3 | Login of both student and staff with wrong Email and correct password. | Username name not found sign up. | As expected | PASS |
| TC4 | Login of both student and staff with correct Email and wrong password. | Password is wrong. | As expected | PASS |
| TC5 | Login with correct Email and correct password for student. | Redirect to Student Dashboard. | As expected | PASS |
| TC6 | Login with correct Email and correct password for staff. | Redirect to Staff Dashboard. | As expected | PASS |
| TC7 | When student attending quiz, if they do not attempt any question. | Answer should be considered as Wrong answer. | As expected | PASS |
| TC8 | When Staff adding question if answer or any other option is empty | Please fill all the fileds. | As expected | PASS |
| TC9 | Registration Not entering Username. | Diplay username is required. | Error | FAIL |
| TC10 | Press Logout button. | Return to Home page. | Access Denied | FAIL |

**Tc10**: we must correct the link that is given the code correctly.

**CHAPTER 05**

# CONCLUSION AND SCOPE OF ENHANCEMENT

## 5.1 CONCLUSION

The Departement quiz management system provides better functionality for an examination to be more efficient and reduce manual paperwork in order to automate all possible tasks. For implementing this system, PHP, HTML, CSS, JavaScript and MySql are used.

The system comprises of following features:

- MANAGEMENT QUIZ
- AUTOMATED GRADING
- ADDING/DELETING QUIZZES AND QUESTIONS

## 5.2 SCOPE OF ENHANCEMENT

There are also few features which can be integrated with the system to make it more flexible.

Below list shows the future points to be considered:

1.Implementing the timer for the quiz.

2.Sending mails on sign up and when student takes the quiz.

3.Supporting all type of questions including MCQ's.

4.Including Programming Questiond where user can compile or interpret on site only.

**Chapter 06**

# 6.Bilblography

## 6.1 TEXTBOOK

Database System Models, Languages, Ramez Elmasri and Sham Kant B. Navathe, 7th Edition, 2017 Pearson.

Fundamentals of Web Development, Randy Connolly and Ricardo Hoar ,First Impression, 2016 Pearson.

## 6.2WEBSITES

1.www.stackoverflow.com

2.www.youtube.com

3.https://www.php.net

4.https://www.w3schools.com