

Online Student Registration System

Submitted By
Devdutt (23BCS13788)
Piyush Bansal (23BCS13844)
Deepanshu (23BCS10107)

In Partial Fulfillment For The Award Of The Degree

Of

(Bachelor Of Engineering)

In

CHANDIGARH UNIVERSITY

November – 2025

Acknowledgement

Apart from the efforts of myself, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project. I would like to show my greatest appreciation to **Zeeshan Ahmed sir** . I can't say thank you enough for his tremendous support and help. I feel motivated and encouraged every. Without his encouragement and guidance this project would not have materialized.

The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the project. I am grateful for their constant support and help.

BONAFIDE CERTIFICATE

Certified that this project report **Online Student Registration System**
is the bonafide work of Devdutt, Piyush, Deepanshu.
who carried out the project work under my supervision.

SIGNATURE

SUPERVISOR

Department of computer application

INDEX

- 1. INTRODUCTION**
- 2. ANALYSIS**
 - SYSTEM ANALYSIS
 - SYSTEM SPECIFICATIONS
- 3. DESIGN APPROACH**
 - ✦ INTRODUCTION TO DESIGN
 - ✦ UML DIAGRAMS
 - ✦ DATA FLOW DIAGRAMS
 - ✦ E-R DIAGRAMS
- 4. PROJECT MODULE**
- 5. TESTING**
- 6. CONCLUSION**
- 7. FUTURE ENHANCEMENTS**
- 8. BIBILIOGRAPHY**
- 9. SCREENSHOT OF OUTPUT**
- 10. SOURCE CODE**

INTRODUCTION:

on-line examinations contents providers to focus on creating effective assessment questions and focusing on exam's feedback delivery to students. In the paper we present techniques that are pertinent to the elements of assessment process: answers submission, computerized grading, and feedback after submission.

As the modern organizations are automated and computers are working as per the instructions, it becomes essential for the coordination of human beings, commodity and computers in a modern organization.

The administrators ,instructor,Students who are attending for online examination can communicate with the system through this projects, thus facilitating effective implementation and monitoring of various activities of Online Examinations like conducting Exams as per scheduled basis and delivering result to that particular use or student.And the details of students who attempted Online Examination are maintained at administrator.

SYSTEM ANALYSIS:

1. Existing System

Existing system is a manual one in which users are maintaining books to store the information like Student Details, Instructor Details, Schedule Details and feedbacks about students who attempted exam as per schedule.. It is very difficult to maintain historical data.

DISADVANTAGES:

The following drawbacks of existing system emphasize the need for computerization:

1. A lot of copies of question papers have to be made
2. A lot of correction work hence delay in giving the results
3. A lot of tabulation work for each subject results

2. Proposed System

This application is used to conduct online examination. The students can sit at individual terminals and login to write the exam in the given duration. . The questions have to be given to the students. This application will perform correction, display the result immediately and also store it in database. This application provides the administrator with a facility to add new exams. This application provides the Instructor add questions to the exam, modify questions in the exam in a particular exam. This application takes care of authentication of the administrator, Instructor as well as the student.

3. Objective of the System

The objective of the Online Examination Tool is to provide better information for the users of this system for better results for their maintenance in student examination schedule details and grading details.

System Specifications

Hardware Requirements:-

- Pentium-IV(Processor).
- 256 MB Ram
- 512 KB Cache Memory
- Hard disk 10 GB
- Microsoft Compatible 101 or more Key Board

Software Requirements: -

- **Operating System :** Windows
- **Web-Technology:** PHP
- **Front-End:** HTML,CSS,JAVASCRIPT
- **Back-End:** MySQL

DESIGN

INTRODUCTION:

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

UML Diagrams:

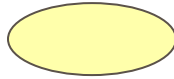
Actor:

A coherent set of roles that users of use cases play when interacting with the use `cases.



Use case:

A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.



UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design:

They are as follows:

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram
- State chat Diagram

USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor.

Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.

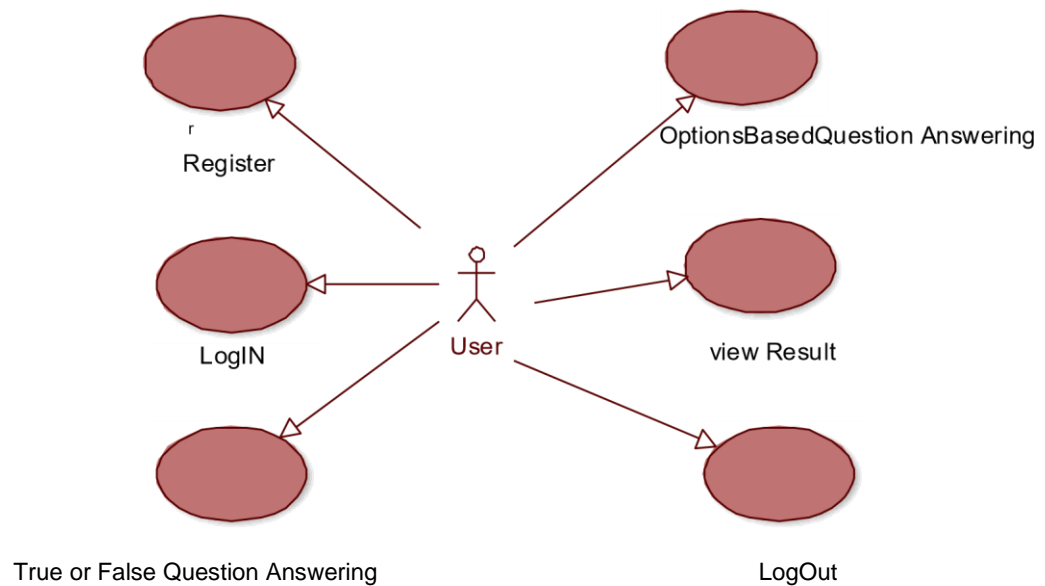
Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.

USECASE DIAGRAM:

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object.

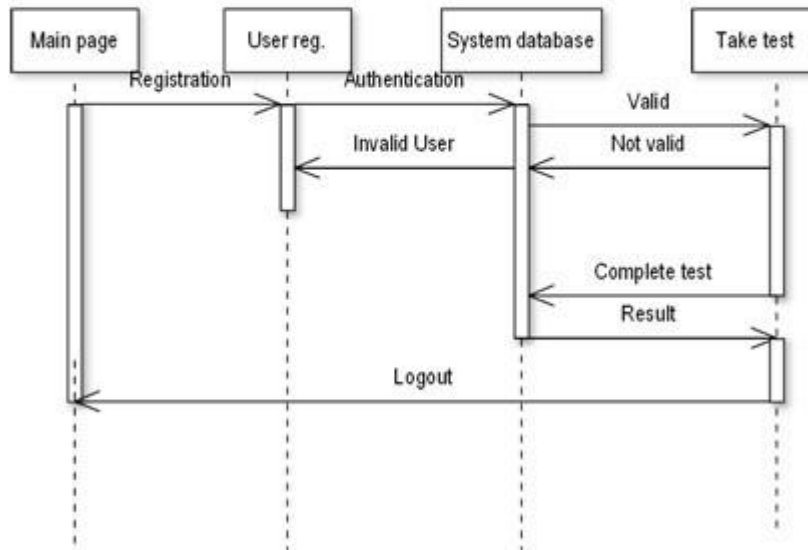
Primary Actor – Sender, Secondary ActorReceiver.

**SEQUENCE DIAGRAM:**

Sequence diagram and collaboration diagram are called **INTERACTION DIAGRAMS**. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis

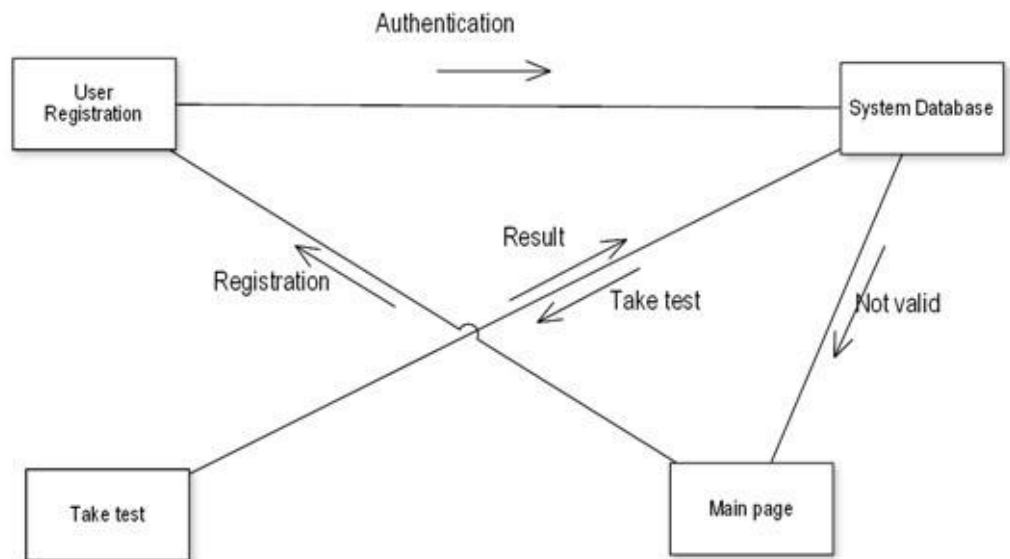
Sequence Diagram



COLLABORATION DIAGRAM:

A collaboration diagram is an introduction diagram that emphasizes the structural organization of the objects that send and receive messages. Graphically a collaboration diagram is a collection of vertices and arcs.

Collaboration Diagram



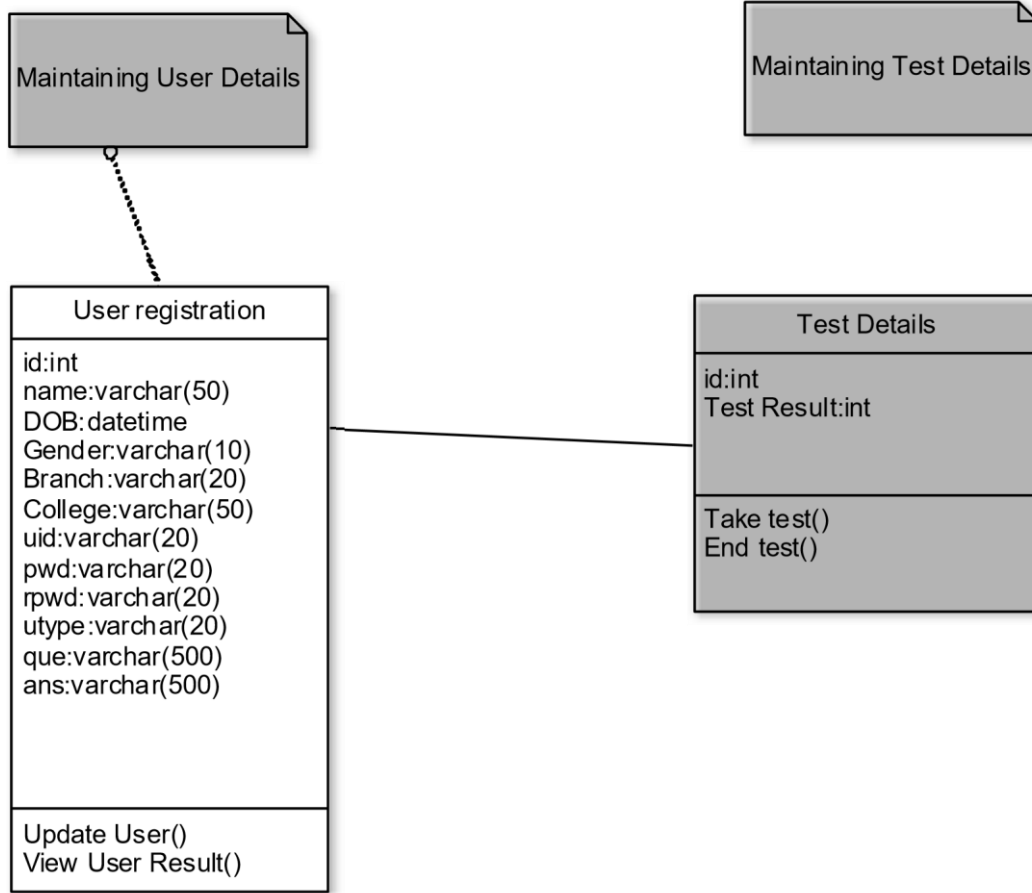
CLASS DIAGRAM:

Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relationships. There is most common diagram in modeling the object oriented systems and are used to give the static view of a system. It shows the dependency between the classes that can be used in our system.

The interactions between the modules or classes of our projects are shown below. Each block contains Class Name, Variables and Methods.

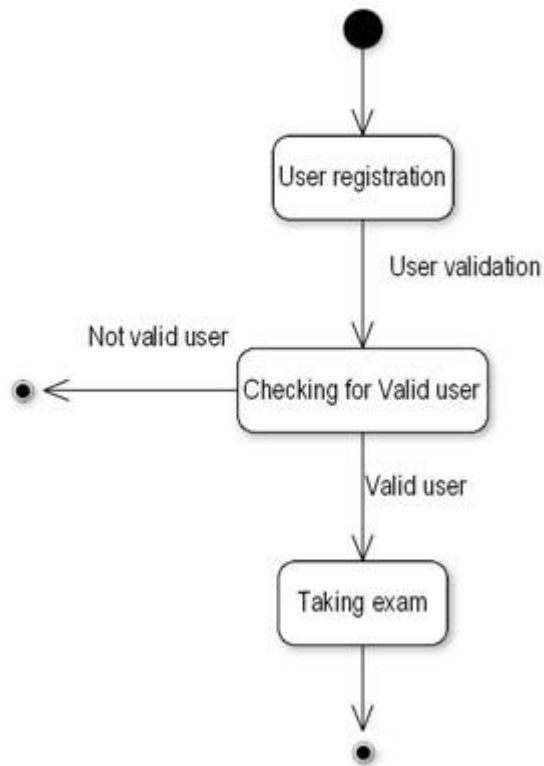
CLASS:

A description of set of objects that share the same attributes, operations, relationships, and semantics



State Chart Diagram

Statechart Diagram



DATA FLOW DIAGRAMS:

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e. the first data flow model

represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst perform an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embody the applications.

A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

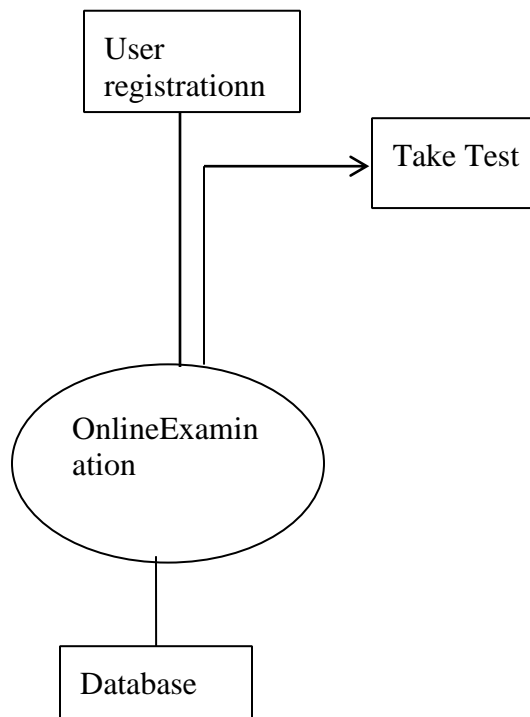
RULES FOR DFD:

- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions
- Reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with Rounded circles.
- A process is required for all the data transformation and Transfers. Therefore, never connect a data store to a data Source or the destinations or another data store with just a Data flow arrow.

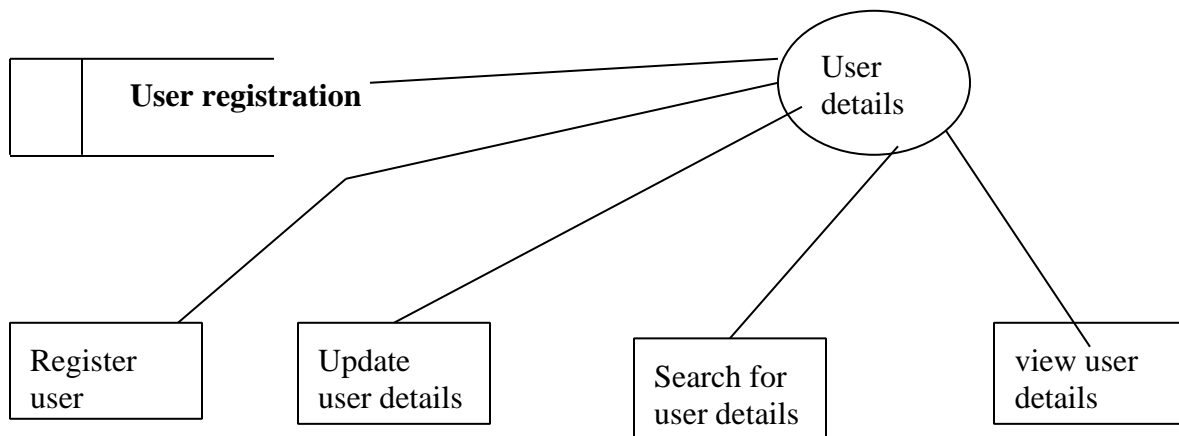
- Do not indicate hardware and ignore control information.
- Make sure the names of the processes accurately convey everything the process is done.
- There must not be unnamed process.
- Indicate external sources and destinations of the data, with Squares.
- Number each occurrence of repeated external entities.
- Identify all data flows for each process step, except simple Record retrievals.
- Label data flow on each arrow.
- Use details flow on each arrow.
- Use the details flow arrow to indicate data movements.

DATAFLOW DIAGRAMS:

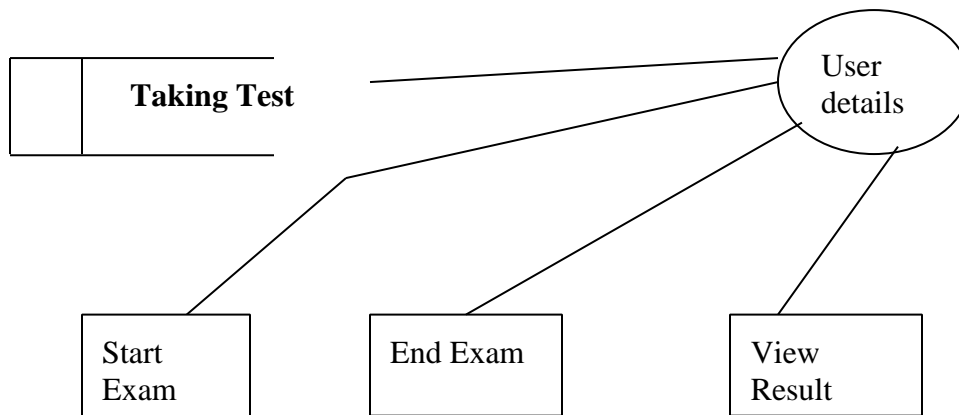
Database:



user registration



Taking Test



E-R Diagrams:

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represents data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design For the database designer, the utility of the ER model is:

- it maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- it is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in a specific database management software.

Connectivity and Cardinality

The basic types of connectivity for relations are: one-to-one, one-to-many, and many-to-many. A *one-to-one* (1:1) relationship is when at most one instance of a entity A is associated with one instance of entity B. For example, "employees in the company are each assigned their own office. For each employee there exists a unique office and for each office there exists a unique employee.

A *one-to-many* (1:N) relationships is when for one instance of entity A, there are zero, one, or many instances of entity B, but for one instance of entity B, there is only one instance of entity A. An example of a 1:N relationships is

a department has many employees

each employee is assigned to one department

A *many-to-many* (M:N) relationship, sometimes called non-specific, is when for one instance of entity A, there are zero, one, or many instances of entity B and for one instance of entity B there are zero, one, or many instances of entity A. The connectivity of a relationship describes the mapping of associated

ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used, among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- ✦ **entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
- ✦ **relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
- ✦ **attributes**, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.
- ✦ **cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.

- ✦ **existence** is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional

ONLINE EXAMINATION SYSTEM

MODULES:

1:ADMIN MODULE

2.INSTRUCTOR MODULE

3.STUDENT MODULE

1.ADMIN MODULE:

- 1.:REGISTER
- 2.LOGIN
- 3.CHANGE PASSWORD&FORGOTPASSWORD
- 4.STUDENT -MODIFYING DETAILS
- 5.DEPARTMENTS-ENTERING/MODIFYING DETAILS
- 6.INSTRUCTOR DETAILS-MODIFYING DETAILS
- 1.REGISTER:To be authenticated first have to be registered.
- 2.LOGIN:The Registered User Can be Allowed to view inner details for which he Permitted
- 3.CHANGE PASSWORD&FORGOTPASSWORD:User has rights to modify his login details& also be informed through mails if he is unable to login.

- 4.STUDENT -MODIFYING DETAILS:User can be modified to change status of each User.
- 5.DEPARTMENTS-ENTERING/MODIFYING DETAILS:New departments adding and old departmentd deletions are spend by this user.
- 6.INSTRUCTOR DETAILS-MODIFYING DETAILS:According to staff he can add or delete Instructors for specific platforms.

2.INSTRUCTOR MODULE:

- 1.REGISTER
- 2. LOGIN
- 3CHANGE PASSWORD&FORGOT PASSWORD
- 4.ADD QUESTIONS-DEPARTMENTS VERIFING.
- 5.UPDATE QUESTIONS -DEPARTMENTS VERIFING
- 6.CREATE EXAMS
- 7.UPDATE EXAMS
- 8.VIEW EXAM DETAILS- VIEW NO OF REGISTERED STUDENTS
- VIEW NO OF ATTENDED STUDENTS
- 9.EVALUATE QUESTION:MULTIPLE CHOICE
- TUE/FALSE
- 1.REGISTER:To be authenticated first have to be registered.
- 2.LOGIN:The Registered User Can be Allowed to view inner details for which he ○ Permitted
- 3.CHANGE PASSWORD&FORGOTPASSWORD:User has rights to modify his logging details& also be informed through mails if he is unable to login
- 4.ADD QUESTIONS-DEPARTMENTS VERIFING:According to flow of questions & Technology he can add questions into the database.
- 5.UPDATE QUESTIONS -DEPARTMENTS VERIFING:If any corrections in data of questions he can modify them
- 6.CREATE EXAMS:He will be prepared schedule for exams periodically.
- 7.UPDATE EXAMS:He has rights to modify exam schedule.
- 8.VIEW EXAM DETAILS- VIEW NO OF REGISTERED STUDENTS, VIEW NO OF ATTENDED STUDENTS:Can view at attended students who has registered.
- 9.EVALUATE QUESTION:MULTIPLE CHOICE

- TUE/FALSE:Evaluation of marks based on his initiations when ○ adding questions

3. STUDENT DETAILS:

- 1.REGISTER
- 2.LOGIN
- 3.TAKE EXAM- MULTIPLE CHOICE
- TRUE/FALSE
- 4. SEE EXAM RESULTS
- 5.LOGOUT

- 1.REGISTER:To be authenticated first have to be registered
- 2.LOGIN:The Registered User Can be allowed to view inner details for which he ○ Permitted
- 3.TAKE EXAM- MULTIPLE CHOICE, TRUE/FALSE:The registred student allowed to start the exam
- 4. SEE EXAM RESULTS:After Completion of exam he can view at his result.
- 5.LOGOUT:After the process of examination he turned to Logout page

OVERVIEW OF TECHNOLOGIES USED

PHP

PHP: Hypertext Preprocessor, is a widely used, general-purpose scripting language that was originally designed for web development, to produce dynamic web pages. It can be embedded into HTML and generally runs on a web server, which needs to be configured to process PHP code and create web page content from it. It can be deployed on most web servers and on almost every operating system and platform free of charge.

PHP was originally created by Rasmus Lerdorf in 1995 and has been in continuous development ever since. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software released under the PHP License, which is incompatible with the GNU General Public License (GPL) because of restrictions on the use of the term PHP

PHP has evolved to include a command line interface capability and can also be used in standalone graphical applications.

USAGE

PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create

dynamic web page content. It can also be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. Since PHP 4, the PHP parser compiles input to produce byte code for processing by the Zend Engine, giving improved performance over its interpreter predecessor

Originally designed to create dynamic web pages, PHP now focuses mainly on server-side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft's Active Server Pages, Sun Microsystems' JavaServer Pages and mod_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote rapid application development (RAD). Some of these include CakePHP, Symfony, CodeIgniter, and Zend Framework, offering features similar to other web application frameworks.

About HTML

HTML, which stands for **Hyper Text Markup Language**, is the predominant markup language for web pages. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists etc as well as for links, quotes, and other items. It allows images and objects to be embedded and can be used to create interactive forms. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content. It can include or can load scripts in languages such as JavaScript which affect the behavior of HTML processors like Web browsers; and Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both HTML and CSS standards, encourages the use of CSS over explicit presentational markup.

Hyper Text Markup Language(HTML) is the encoding scheme used to create and format a web document. A user need not be an expert programmer to make use of HTML for creating hypertext documents that can be put on the internet.

Most graphical e-mail clients allow the use of a subset of HTML (often ill-defined) to provide formatting and semantic markup not available with plain text. This may include typographic information like coloured headings, emphasized and quoted text, inline images and diagrams. Many such clients include both a GUI editor for composing HTML e-mail messages and a rendering engine for displaying them. Use of HTML in e-mail is controversial because of compatibility issues, because it can help disguise phishing attacks, because it can confuse spam filters and because the message size is larger than plain text.

NAMING CONVENTIONS

The most common filename extension for files containing HTML is .html. A common abbreviation of this is .htm, which originated because some early operating systems and file systems, such as DOS and FAT, limited file extensions to three letters.

HTML APPLICATION

26/54

An HTML Application is a Microsoft Windows application that uses HTML and Dynamic HTML in a browser to provide the application's graphical interface. A regular HTML file is confined to the security model of the web browser, communicating only to web servers and manipulating only webpage objects and site cookies. An HTA runs as a fully trusted application and therefore has more privileges, like creation/editing/removal of files and Windows Registry entries. Because they operate outside the browser's security model, HTAs cannot be executed via HTTP, but must be downloaded (just like an EXE file) and executed from local file system

ABOUT JAVASCRIPT

JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.

PROTOTYPE-BASED

JavaScript uses prototypes instead of classes for inheritance. It is possible to simulate many class-based features with prototypes in JavaScript.

Functions double as object constructors along with their typical role. Prefixing a function call with new creates a new object and calls that function with its local this keyword bound to that object for that invocation. The constructor's prototype property determines the object used for the new object's internal prototype. JavaScript's built-in constructors, such as Array, also have prototypes that can be modified.

Unlike many object-oriented languages, there is no distinction between a function definition and a method definition. Rather, the distinction occurs during function calling; a function can be called as a method. When a function is called as a method of an object, the function's local this keyword is bound to that object for that invocation.

USAGE

The primary use of JavaScript is to write functions that are embedded in or included from HTML pages and interact with the Document Object Model (DOM) of the page.

Because JavaScript code can run locally in a user's browser (rather than on a remote server) it can respond to user actions quickly, making an application feel more responsive. Furthermore, JavaScript code can detect user actions which HTML alone cannot, such as individual keystrokes. Applications such as Gmail take advantage of this: much of the user-interface logic is written in JavaScript, and JavaScript dispatches requests for information (such as the content of an e-mail message) to the server. The wider trend of Ajax programming similarly exploits this strength.

A JavaScript engine (also known as *JavaScript interpreter* or *JavaScript implementation*) is an interpreter that interprets JavaScript source code and executes the script accordingly. The first JavaScript engine was created by Brendan Eich at Netscape Communications Corporation, for the Netscape Navigator web browser.

A web browser is by far the most common host environment for JavaScript. Web browsers typically use the public API to create "host objects" responsible for reflecting the DOM into JavaScript.

ABOUT MySQL***MySQL Introduction***

There are a large number of database management systems currently available, some commercial and some free.

Some of them : Oracle, Microsoft Access, Mysql and PostgreSQL.

These database systems are powerful, feature-rich software, capable of organizing and searching millions of records at very high speeds.

Understanding Databases, Records, and Primary Keys

Every Database is composed of one or more tables.

These Tables, which structure data into rows and columns, Impose organization on the data.

The records in a table(below) are not arranged in any particular order.

To make it easy to identify a specific record,therefore, it becomes necessary

standing Relationships and Foreign Keys(RDBMS)

You already know that a single database can hold multiple tables.

In a Relational database management system(RDBMS), these tables can be linked to each other by one or more common fields, called **foreign keys**.

Database administrator is the super user of database, he has unrestricted rights and privileges to access database, grant permission to other database users.

Database user is the person who uses the database in a restricted privileges, provided by database administrator.

DATABASE TABLES:**USER REG TABLE**

NAME	NULL/NOTNULL	TYPE	KEY
ID	NOTNULL	INT	PRIMARYKEY
NAME	NULL	VARCHAR(50)	
DOB	NULL	DATETIME	
GENDER	NULL	VARCHAR(10)	

BRANCH	NULL	VARCHAR(20)
COLLEGE	NULL	VARCHAR(50)
UID	NULL	VARCHAR(50)
PWD	NULL	
RPWD	NULL	VARCHAR(20)
UTYPE	NULL	VARCHAR(20)
QUE	NULL	
ANS	NULL	VARCHAR(20)
		VARCHAR(500)
		VARCHAR(500)

True/False Based Question Table

NAME	NULL/NOTNULL	TYPE	KEY
ID	NOTNULL	INT	PRIMARYKEY
QUE	NULL	VARCHAR(500)	
AW	NULL	VARCHAR(500)	

True/False Based Answer Table

NAME	NULL/NOTNULL	TYPE	KEY
ID	NOTNULL	INT	FOREIGNKEY
AW	NULL	VARCHAR(500)	

Options Based Question Table

NAME	NULL/NOTNULL	TYPE	KEY
QID	NOTNULL	INT	PRIMARYKEY
QN	NULL	VARCHAR(500)	
OPTIONS1	NULL	VARCHAR(100)	
OPTIONS2	NULL	VARCHAR(100)	
ANSWER	NULL	VARCHAR(100)	

Options Based Answers

NAME	NULL/NOTNULL	TYPE	KEY
QID	NOTNULL	INT	FOREIGNKEY
ANSWER	NULL	VARCHAR(10)	

All Student Marks

NAME	NULL/NOTNULL	TYPE	KEY
------	--------------	------	-----

ID	NULL	INT
MARKS	NULL	INT

Exam Schedule

NAME	NULL/NOTNULL	TYPE	KEY
ENAME	NULL	VARCHAR(30)	
EDATE	NULL	DATETIME	

Testing

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding.

System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3. A successful test is one that uncovers an undiscovered error

CONCLUSION:

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- Automation of the entire system improves the efficiency
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

FUTURE ENHANCEMENTS:

This application avoids the manual work and the problems concern with it. It is an easy way to obtain the information regarding the different scheduled examinations information that are Currently issued.

Well I and my team members have worked hard in order to present an improved website better than the existing one's regarding the information about the various activities. Still ,we found out that the project can be done in a better way. Primarily, when we request information about a particular schedules it just shows the exam date and platform. So, after getting the information we can get access to the onlineexam.

The enhancement that we can add the searching option. We can directly search to the particular student details from this site.

BIBLIOGRAPHY

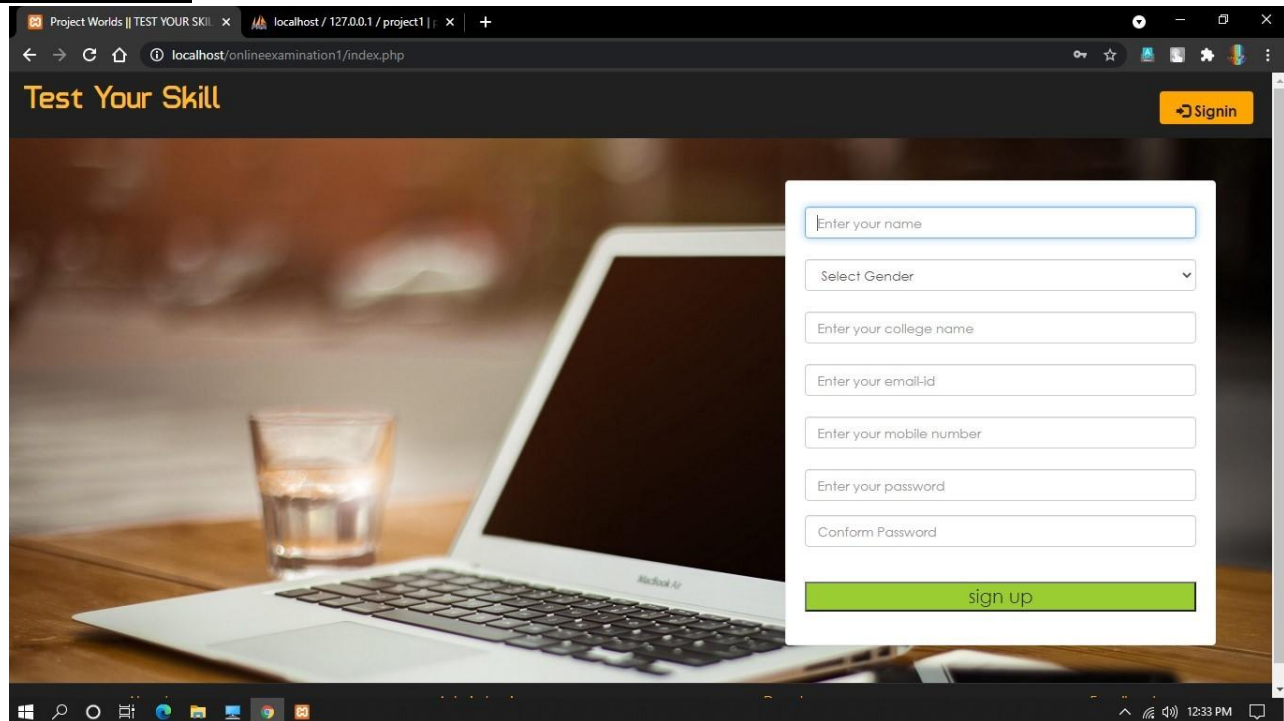
The following books were referred during the analysis and execution phase of the project

Books Referred:

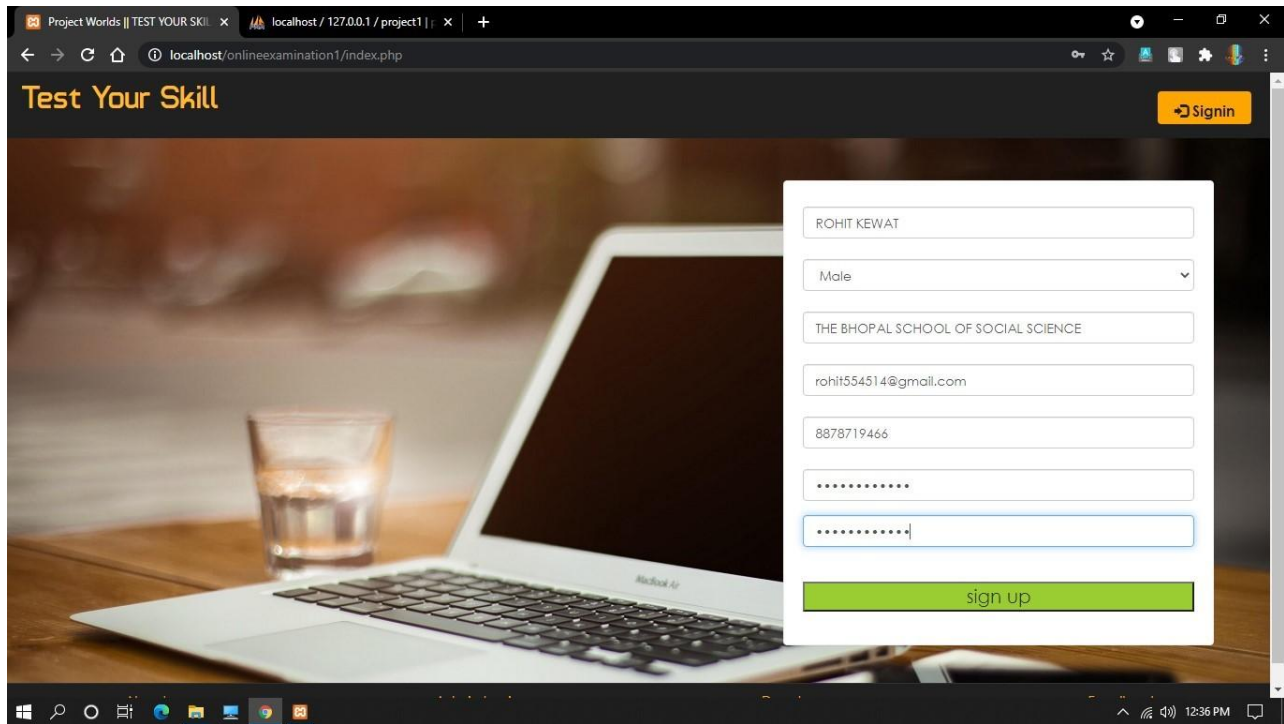
- BEGINNING PHP 5 ---DAVE MERCER
- BLACK BOOK HTML ---WILEY DREAMTECH
- PHP AND MYSQL WEB DEVELOPMENT --- LUKEWELLING,LAURA
- MICROSOFT SQL SERVER-2000 ---RANKIN, PAUL & JENSEN
- SQL SERVER-2000 ---DUSAN PETKOVIC

Output screenshot

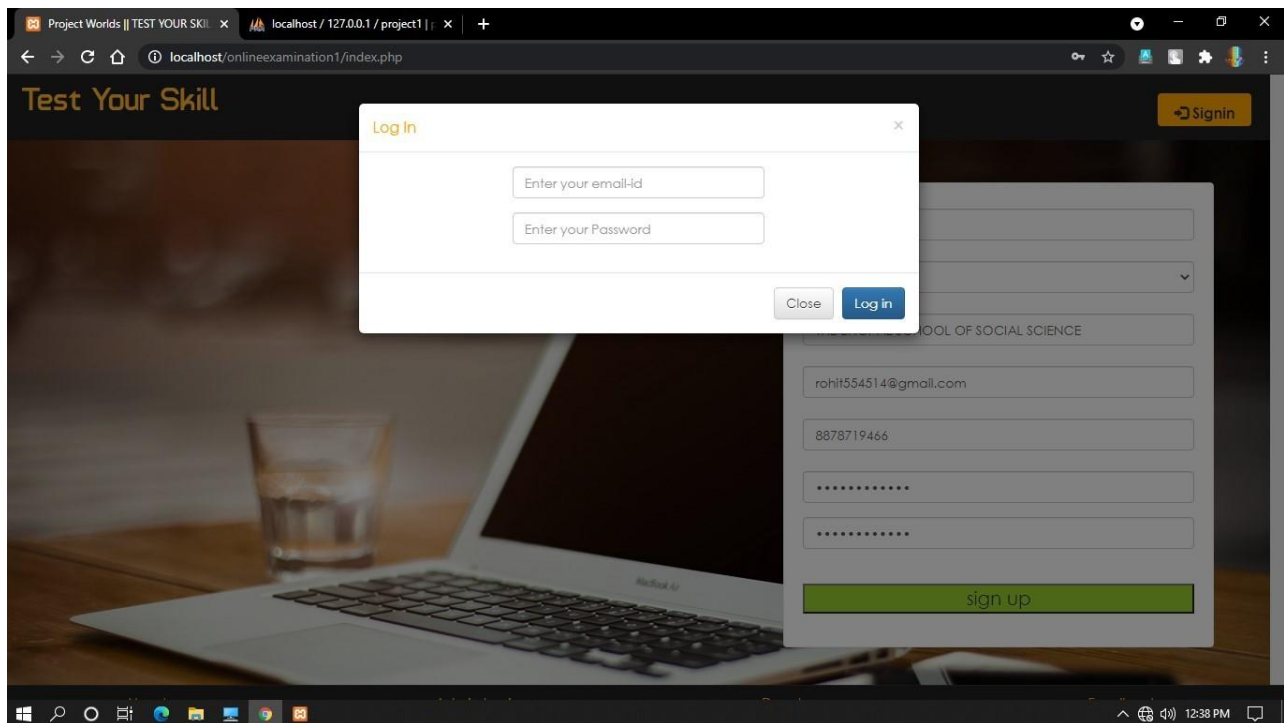
Home page :



Registration by student :



Login by student :



Subject selected by student :

The screenshot shows a web application titled "Test Your Skill" with a user profile "Hello, Rohit Kewat" and a "Signout" button. The page features a navigation bar with "Netcamp", "Home", "History", "Ranking", "Signout", and a search bar. The main content area displays a table of quiz questions with the following data:

S.N.	Topic	Total question	Marks	Time limit	Action
1	Linux :vi Editor ✓	5	10	10 min	Restart
2	Linux:startup	5	10	10 min	Start
3	Networking	2	4	5 min	Start
4	C++ Coding	2	4	5 min	Start
5	Php Coding ✓	2	4	5 min	Restart
6	Linux : File Management	2	4	5 min	Start

The footer contains links for "About us", "Admin Login", "Developers", and "Feedback".

Exam given by student :

The screenshot shows the same "Test Your Skill" page, but with a quiz question displayed. The question is: "Question 1 :: what is command for print in php??"

The options provided are:

- ☐ echo
- ☐ print
- ☐ printf
- ☐ cout

There is a "Submit" button at the bottom of the question area.

Result of student :

Test Your Skill Hello, Rohit Kewat Signout

Netcamp Home History Ranking Signout Enter tag Search

Result

Total Questions	2
right Answer	2
Wrong Answer	0
Score	4
Overall Score	2

About us Admin Login Developers Feedback

Feedback form :

Test Your Skill Home Signout

FEEDBACK/REPORT A PROBLEM

You can send us your feedback through e-mail on the following e-mail id:
serbermz2020@gmail

Or you can directly submit your feedback by filling the enteries below:-

Name:

Subject:

E-Mail address:

Write feedback here...

Submit

Source code :

Admin :

```
<!DOCTYPE html >
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Project Worlds || TEST YOUR SKILL </title>
<link rel="stylesheet" href="css/bootstrap.min.css"/>
<link rel="stylesheet" href="css/bootstrap-theme.min.css"/>
<link rel="stylesheet" href="css/main.css">
<link rel="stylesheet" href="css/font.css">
<script src="js/jquery.js" type="text/javascript"></script>


<script src="js/bootstrap.min.js" type="text/javascript"></script>
<link href='http://fonts.googleapis.com/css?family=Roboto:400,700,300' rel='stylesheet' type='text/css'> <!--
alert message-->
<?php if(@$_GET['w'])
{echo'<script>alert('".$_GET['w']."'");</script>';}
?>
<!--alert message end-->

</head> <?php
include_once 'dbConnection.php';
?>
<body>
<div class="header">
<div class="row">
<div class="col-lg-6">
<span class="logo">Test Your Skill</span></div>
<div class="col-md-4 col-md-offset-2">
<?php
include_once 'dbConnection.php';
session_start();
if(!isset($_SESSION['email']))){
header("location:index.php");
}
else
{
$name = $_SESSION['name'];
$email=$_SESSION['email'];

include_once 'dbConnection.php'; echo '<span class="pull-right top title1"><span class="log1"><span
class="glyphicon glyphicon-user" aria-hidden="true"></span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~Hello,</span>
<a href="account.php?q=1" class="log
log1">' . $name.'</a>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~<a href="logout.php?q=account.php" class="log"><span class="glyphicon glyphicon-log-out"
aria-hidden="true"></span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~Signout</button></a></span>';
}}?>
</div>
</div></div>
<div class="bg">

<!--navigation menu-->
<nav class="navbar navbar-default title1">
<div class="container-fluid">
<!-- Brand and toggle get grouped for better mobile display -->
```

```

<div class="navbar-header">
  <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
  aria-expanded="false">
    <span class="sr-only">Toggle navigation</span>
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
  </button>
  <a class="navbar-brand" href="#"><b>Netcamp</b></a>
</div>

<!-- Collect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
  <ul class="nav navbar-nav">
    <li <?php if(@$_GET['q']==1) echo'class="active"'; ?> <a href="account.php?q=1"><span class="glyphicon glyphicon-home"
    aria-hidden="true"></span>&nbsp;Home<span class="sr-only">(current)</span></a></li>
    <li <?php if(@$_GET['q']==2) echo'class="active"'; ?><a href="account.php?q=2"><span class="glyphicon glyphicon-list-alt"
    aria-hidden="true"></span>&nbsp;History</a></li>
    <li <?php if(@$_GET['q']==3) echo'class="active"'; ?><a href="account.php?q=3"><span class="glyphicon
    glyphicon-stats" aria-hidden="true"></span>&nbsp;Ranking</a></li>
    <li class="pull-right"> <a href="logout.php?q=account.php"><span class="glyphicon glyphicon-log-out"
    ariahidden="true"></span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;Signout</a></li>
  </ul>
  <form class="navbar-form navbar-left" role="search">
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Enter tag ">
    </div>
    <button type="submit" class="btn btn-default"><span class="glyphicon glyphicon-search"
    ariahidden="true"></span>&nbsp;Search</button>
  </form>
</div><!-- /.navbar-collapse -->
</div><!-- /.container-fluid -->
</nav><!-- navigation menu closed -->
<div class="container"><!-- container start -->
<div class="row">
<div class="col-md-12">

<!-- home start -->
<?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><b>S.N.</b></td><td><b>Topic</b></td><td><b>Total question</b></td><td><b>Marks</b></td><td><b>Time
limit</b></td><td></td></tr>';
$c=1;
while($row = mysqli_fetch_array($result)) {
    $title = $row['title'];
    $total = $row['total'];
    $sahi = $row['sahi'];
    $time = $row['time'];
    $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>'. $c++.'</td><td>'. $title.'</td><td>'. $total.'</td><td>'. $sahi*$total.'</td><td>'. $time.'&nbsp;min</td>
        <td><b><a href="account.php?q=quiz&step=2&eid='. $eid.'&n=1&t='. $total.'" class="pull-right btn sub1"

```

```

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

}>
<!--<span id="countdown" class="timer"></span>
<script> var seconds = 40; function
secondPassed() { var minutes =
Math.round((seconds - 30)/60); var
remainingSeconds = seconds % 60;
if (remainingSeconds < 10) {
remainingSeconds = "0" + remainingSeconds;
}
document.getElementById('countdown').innerHTML = minutes + ":" + remainingSeconds;
if (seconds == 0) {
clearInterval(countdownTimer);
document.getElementById('countdown').innerHTML = "Buzz Buzz";
} else {
seconds--;
}
}
var countdownTimer = setInterval('secondPassed()', 1000);
</script>-->

<!--home closed-->

<!--quiz start-->
<?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 &nbsp;<span>'. $sn. '&nbsp;<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="formhorizontal">
<br />';

while($row=mysqli_fetch_array($q) )

```



```

{
    $option=$row['option']; $optionid=$row['optionid'];
    echo<input type="radio" name="ans" value="".$optionid."">'. $option.'<br /><br />';
}
echo<br /><button type="submit" class="btn btn-primary"><span class="glyphicon glyphicon-lock"
ariahidden="true"></span>&nbsp;Submit</button></form></div>';
//header("location:dash.php?q=4&step=2&eid=$id&n=$total");
}
//result display
if(@$_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&nbsp;<span class="glyphicon glyphicon-ok-circle"
        ariahidden="true"></span></td><td>'. $r.'</td></tr>
        <tr style="color:red"><td>Wrong Answer&nbsp;<span class="glyphicon glyphicon-remove-circle"
        ariahidden="true"></span></td><td>'. $w.'</td></tr>
        <tr style="color:#66CCFF"><td>Score&nbsp;<span class="glyphicon glyphicon-star"
        ariahidden="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&nbsp;<span class="glyphicon glyphicon-stats"
        ariahidden="true"></span></td><td>'. $s.'</td></tr>';
    }
    echo '</table></div>';

}
?>
<!--quiz end-->
<?php
//history start
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:red"><td><b>S.N.</b></td><td><b>Quiz</b></td><td><b>Question
    Solved</b></td><td><b>Right</b></td><td><b>Wrong<b></td><td><b>Score</b></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>'. $c.'</td><td>'. $title.'</td><td>'. $qa.'</td><td>'. $r.'</td><td>'. $w.'</td><td>'. $s.'</td></tr>';
}
echo '</table></div>';
}

//ranking start
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><b>Rank</b></td><td><b>Name</b></td><td><b>Gender</b></td><td><b>College</b></td><td><b>Score</b></td></tr>';
$c=0;
while($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'];
$gender=$row['gender'];
$college=$row['college'];
}
$c++; echo
'<tr><td
style="color:#99cc32"><b>'. $c.'</b></td><td>'. $name.'</td><td>'. $gender.'</td><td>'. $college.'</td><td>'. $s.'</td><td>';
}
echo '</table></div></div>';}
?>

</div></div></div></div>
<!--Footer start-->
<div class="row footer">
<div class="col-md-3 box">
<a href="http://www.projectworlds.in/online-examination" target="_blank">About us</a> </div>
<div class="col-md-3 box">
<a href="#" data-toggle="modal" data-target="#login">Admin Login</a></div> <div
class="col-md-3 box">
<a href="#" data-toggle="modal" data-target="#developers">Developers</a> </div>
<div class="col-md-3 box">
<a href="feedback.php" target="_blank">Feedback</a></div></div>
<!-- Modal For Developers-->
<div class="modal fade title1" id="developers">

```

```

<div class="modal-dialog">
  <div class="modal-content">
    <div class="modal-header">
      <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
class="sr-only">Close</span></button>
      <h4 class="modal-title" style="font-family:'typo' "><span style="color:orange">Developers</span></h4>
    </div>

    <div class="modal-body">
      <p>

        <div class="row">
          <div class="col-md-4">
            
          </div>
          <div class="col-md-5">
            <a href="http://yugeshverma.blogspot.in" style="color:#202020; font-family:'typo' ; font-size:18px" title="Find on
Facebook">Yugesh Verma</a>
            <h4 style="color:#202020; font-family:'typo' ;font-size:16px" class="title1">+91 9165063741</h4>
            <h4 style="font-family:'typo' ">vermayugesh323@gmail.com</h4>
            <h4 style="font-family:'typo' ">Chhattishgarh insitute of management & Technology ,bhilai</h4></div></div>

          </div>

        </div><!-- /.modal-content -->
      </div><!-- /.modal-dialog -->
    </div><!-- /.modal -->

<!--Modal for admin login-->
  <div class="modal fade" id="login">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
class="sr-only">Close</span></button>
          <h4 class="modal-title"><span style="color:orange;font-family:'typo' ">LOGIN</span></h4>
        </div>
        <div class="modal-body title1">
          <div class="row">
            <div class="col-md-3"></div>
            <div class="col-md-6">
              <form role="form" method="post" action="admin.php?q=index.php">
                <div class="form-group">
                  <input type="text" name="uname" maxlength="20" placeholder="Admin user id" class="form-control"/>
                </div>
                <div class="form-group">
                  <input type="password" name="password" maxlength="15" placeholder="Password" class="form-control"/>
                </div>
                <div class="form-group" align="center">
                  <input type="submit" name="login" value="Login" class="btn btn-primary" /> </div>
              </form>
            </div><div class="col-md-3"></div></div>

            </div>
            <!--<div class="modal-footer">
              <button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
            </div-->
          </div><!-- /.modal-content -->

```

```

</div><!-- /.modal-dialog -->
</div><!-- /.modal -->
<!-- footer end -->

```

```

</body> </html>

```

Source code :

Login :

```

<?php
session_start();
if(isset($_SESSION["email"])){ session_destroy();
}
include_once 'dbConnection.php';
$ref=@$_GET['q'];
$email = $_POST['email'];
$password = $_POST['password'];

$email = stripslashes($email);
$email = addslashes($email);
$password = stripslashes($password);
$password = addslashes($password);
$password=md5($password);
$result = mysqli_query($con,"SELECT name FROM user WHERE email = '$email' and password = '$password'" ) or die('Error');
$count=mysqli_num_rows($result);
if($count==1){
while($row = mysqli_fetch_array($result)) {
    $name = $row['name'];
}
$_SESSION["name"] = $name; $_SESSION["email"]
= $email; header("location:account.php?q=1");
}
else
header("location:$ref?w=Wrong Username or Password");

?>

```

Source code :

Dashboard :

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Project Worlds || DASHBOARD </title>
<link rel="stylesheet" href="css/bootstrap.min.css"/>
<link rel="stylesheet" href="css/bootstrap-theme.min.css"/>
<link rel="stylesheet" href="css/main.css">
<link rel="stylesheet" href="css/font.css">

```

```
<script src="js/jquery.js" type="text/javascript"></script>

<script src="js/bootstrap.min.js" type="text/javascript"></script>
    <link href='http://fonts.googleapis.com/css?family=Roboto:400,700,300' rel='stylesheet' type='text/css'>

<script>
$(function () {
    $(document).on( 'scroll', function(){
        console.log('scroll top : ' + $(window).scrollTop());
        if($(window).scrollTop()>=$(("logo").height())
            {
                $(".navbar").addClass("navbar-fixed-top");
            }

            if($(window).scrollTop()<$(("logo").height())
            {
                $(".navbar").removeClass("navbar-fixed-top");
            }
        });
});</script>
</head>

<body style="background:#eee;">
<div class="header">
<div class="row">
<div class="col-lg-6">
<span class="logo">Test Your Skill</span></div>
<?php
include_once 'dbConnection.php';
session_start();
$email=$_SESSION['email'];
if(!isset($_SESSION['email']))){
header("location:index.php");
}
else
{
$name = $_SESSION['name'];;;

include_once 'dbConnection.php'; echo '<span class="pull-right top title1" ><span class="log1"><span
class="glyphicon glyphicon-user" ariahidden="true"></span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&Hello,</span>
<a href="account.php" class="log
log1">' . $name.'</a>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&<a href="logout.php?q=account.php" class="log"><span class="glyphicon glyphicon-log-out"
aria-hidden="true"></span>&nbsp;&nbsp;&nbsp;&nbsp;&Signout</button></a></span>; }?'>

</div></div>
<!-- admin start-->

<!--navigation menu-->
<nav class="navbar navbar-default title1">
<div class="container-fluid">
<!-- Brand and toggle get grouped for better mobile display -->
<div class="navbar-header">
<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1"
aria-expanded="false">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
```



```

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

}

```

```

//ranking start if(@$_GET['q']==
2)
{
$q=mysqli_query($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><b>Rank</b></td><td><b>Name</b></td><td><b>Gender</b></td><td><b>College</b></td><td><b>Scor
e</b></td></tr>';
$c=0;
while($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'];
$gender=$row['gender'];
$college=$row['college'];
}
$c++; echo
'<tr><td
style="color:#99cc32"><b>'. $c. '</b></td><td>'. $name. '</td><td>'. $gender. '</td><td>'. $college. '</td><td>'. $s. '</td><td>';
}
echo '</table></div></div>';}

?>

```

```

<!--home closed-->

```

```

<!--users start-->

```

```

<?php if(@$_GET['q']==1) {

```

```

$result = mysqli_query($con,"SELECT * FROM user") or die('Error');

```

```

echo '<div class="panel"><div class="table-responsive"><table class="table table-striped title1">

```

```

<tr><td><b>S.N.</b></td><td><b>Name</b></td><td><b>Gender</b></td><td><b>College</b></td><td><b>Email</b></td><td>
<b>Mobile</b></td><td></td></tr>';

```

```

$c=1;

```

```

while($row = mysqli_fetch_array($result)) {

```

```

    $name = $row['name'];

```

```

    $mob = $row['mob'];

```

```

        $gender = $row['gender'];
        $email = $row['email'];
        $college = $row['college'];

        echo
'<tr><td>'. $c++. '</td><td>'. $name. '</td><td>'. $gender. '</td><td>'. $college. '</td><td>'. $email. '</td><td>'. $mob. '</td>
<td><a title="Delete User" href="update.php?demail='. $email. '"><b><span class="glyphicon glyphicon-trash"
ariahidden="true"></span></b></a></td></tr>';
    }
    $c=0;
    echo '</table></div></div>';

}??
<!--user end-->

<!--feedback start-->
<?php if(@$_GET['q']==3) {
$result = mysqli_query($con,"SELECT * FROM `feedback` ORDER BY `feedback`.`date` DESC") or die('Error'); echo
'<div class="panel"><div class="table-responsive"><table class="table table-striped title1">
<tr><td><b>S.N.</b></td><td><b>Subject</b></td><td><b>Email</b></td><td><b>Date</b></td><td><b>Time</b></td><td><b>
By</b></td></tr></td></td></td></td></tr>';
    $c=1;
    while($row = mysqli_fetch_array($result)) {
        $date = $row['date'];
        $date= date("d-m-Y",strtotime($date));
        $time = $row['time'];
        $subject = $row['subject'];
        $name = $row['name'];
        $email = $row['email'];        $id =
        $row['id'];        echo '<tr><td>'. $c++. '</td>';
        echo '<td><a title="Click to open feedback"
href="dash.php?q=3&fid='. $id. '">'. $subject. '</a></td><td>'. $email. '</td><td>'. $date. '</td><td>'. $time. '</td><td>'. $name. '</td>
<td><a title="Open Feedback" href="dash.php?q=3&fid='. $id. '"><b><span class="glyphicon glyphicon-folder-open"
ariahidden="true"></span></b></a></td>';
        echo '<td><a title="Delete Feedback" href="update.php?fid='. $id. '"><b><span class="glyphicon glyphicon-trash"
ariahidden="true"></span></b></a></td>

        </tr>';
    }
    echo '</table></div></div>';
}
??
<!--feedback closed-->

<!--feedback reading portion start-->
<?php if(@$_GET['fid']) { echo
'<br />'; $id=@$_GET['fid'];
$result = mysqli_query($con,"SELECT * FROM feedback WHERE id='$id' ") or die('Error'); while($row
= mysqli_fetch_array($result)) {
    $name = $row['name'];
    $subject = $row['subject'];
    $date = $row['date'];
    $date= date("d-m-Y",strtotime($date));
    $time = $row['time'];
    $feedback = $row['feedback'];

```



```

echo '<div class="panel"<a title="Back to Archive" href="update.php?q1=2"><b><span class="glyphicon glyphicon-level-up"
ariahidden="true"></span></b></a><h2 style="text-align:center; margin-top:-15px;font-family: "Ubuntu",
sans-serif;"><b>'. $subject.'</b></h1>';
echo '<div class="mCustomScrollbar" data-mcs-theme="dark" style="margin-left:10px;margin-right:10px; max-height:450px;
lineheight:35px;padding:5px;"><span style="line-height:35px;padding:5px;">-&nbsp;<b>DATE:</b>&nbsp;<span>'. $date.'</span>
<span style="line-height:35px;padding:5px;">&nbsp;<b>Time:</b>&nbsp;<span>'. $time.'</span><span
style="line-height:35px;padding:5px;">&nbsp;<b>By:</b>&nbsp;<span>'. $name.'</span><br />'. $feedback.'</div></div>');
}?'>

```

```

<!--Feedback reading portion closed-->

```

```

<!--add quiz start-->

```

```

<?php

```

```

if(@$_GET['q']==4 && !(@$_GET['step'])) { echo
'

```

```

<div class="row">

```

```

<span class="title1" style="margin-left:40%;font-size:30px;"><b>Enter Quiz Details</b></span><br /><br />

```

```

<div class="col-md-3"></div><div class="col-md-6"> <form class="form-horizontal title1" name="form"
action="update.php?q=addquiz" method="POST">
<fieldset>

```

```

<!-- Text input-->

```

```

<div class="form-group">

```

```

<label class="col-md-12 control-label" for="name"></label>

```

```

<div class="col-md-12">

```

```

<input id="name" name="name" placeholder="Enter Quiz title" class="form-control input-md" type="text">

```

```

</div>

```

```

</div>

```

```

<!-- Text input-->

```

```

<div class="form-group">

```

```

<label class="col-md-12 control-label" for="total"></label>

```

```

<div class="col-md-12">

```

```

<input id="total" name="total" placeholder="Enter total number of questions" class="form-control input-md" type="number">

```

```

</div>

```

```

</div>

```

```

<!-- Text input-->

```

```

<div class="form-group">

```

```

<label class="col-md-12 control-label" for="right"></label>

```

```

<div class="col-md-12">

```

```

<input id="right" name="right" placeholder="Enter marks on right answer" class="form-control input-md" min="0"
type="number">

```

```

</div>

```

```

</div>

```

```

<!-- Text input-->

```

```

<div class="form-group">

```

```

<label class="col-md-12 control-label" for="wrong"></label>

```

```

<div class="col-md-12">

```

```
<input id="wrong" name="wrong" placeholder="Enter minus marks on wrong answer without sign" class="form-control input-md"
min="0" type="number">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="time"></label>
```

```
<div class="col-md-12">
```

```
<input id="time" name="time" placeholder="Enter time limit for test in minute" class="form-control input-md" min="1"
type="number">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="tag"></label>
```

```
<div class="col-md-12">
```

```
<input id="tag" name="tag" placeholder="Enter #tag which is used for searching" class="form-control input-md" type="text">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="desc"></label>
```

```
<div class="col-md-12">
```

```
<textarea rows="8" cols="8" name="desc" class="form-control" placeholder="Write description here..."></textarea>
```

```
</div>
```

```
</div>
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for=""></label>
```

```
<div class="col-md-12">
```

```
<input type="submit" style="margin-left:45%" class="btn btn-primary" value="Submit" class="btn btn-primary"/>
```

```
</div>
```

```
</div>
```

```
</fieldset>
```

```
</form></div>;
```

```
}
```

```
?>
```

```
<!--add quiz end-->
```

```
<!--add quiz step2 start-->
```

```
<?php
```

```
if(@$_GET['q']==4 && (@$_GET['step'])==2 ) { echo
```

```
,
```

```
<div class="row">
```

```

<span class="title1" style="margin-left:40%;font-size:30px;"><b>Enter Question Details</b></span><br /><br />
<div class="col-md-3"></div><div class="col-md-6"><form class="form-horizontal title1" name="form"
action="update.php?q=addqns&n='.$_GET['n'].'&eid='.$_GET['eid'].'&ch=4 " method="POST"> <fieldset>
';

for($i=1;$i<=$_GET['n'];$i++)
{
echo '<b>Question number&nbsp;'.$i.'&nbsp;:</b><br /><!-- Text input-->
<div class="form-group">
<label class="col-md-12 control-label" for="qns'.$i.'"></label>
<div class="col-md-12">
<textarea rows="3" cols="5" name="qns'.$i.'" class="form-control" placeholder="Write question number '.$i.'" here..."></textarea>
</div>
</div>
<!-- Text input-->
<div class="form-group">
<label class="col-md-12 control-label" for="a'.$i.'"></label>
<div class="col-md-12">
<input id="a'.$i.'" name="a'.$i.'" placeholder="Enter option a" class="form-control input-md" type="text">

</div>
</div>
<!-- Text input-->
<div class="form-group">
<label class="col-md-12 control-label" for="b'.$i.'"></label>
<div class="col-md-12">
<input id="b'.$i.'" name="b'.$i.'" placeholder="Enter option b" class="form-control input-md" type="text">

</div>
</div>
<!-- Text input-->
<div class="form-group">
<label class="col-md-12 control-label" for="c'.$i.'"></label>
<div class="col-md-12">
<input id="c'.$i.'" name="c'.$i.'" placeholder="Enter option c" class="form-control input-md" type="text">

</div>
</div>
<!-- Text input-->
<div class="form-group">
<label class="col-md-12 control-label" for="d'.$i.'"></label>
<div class="col-md-12">
<input id="d'.$i.'" name="d'.$i.'" placeholder="Enter option d" class="form-control input-md" type="text">

</div>
</div>
<br />
<b>Correct answer</b><br />
<select id="ans'.$i.'" name="ans'.$i.'" placeholder="Choose correct answer " class="form-control input-md" >
<option value="a">Select answer for question '.$i.'</option>
<option value="a">option a</option>
<option value="b">option b</option>
<option value="c">option c</option>
<option value="d">option d</option> </select><br /><br />
}
echo '<div class="form-group">

```

```

<label class="col-md-12 control-label" for=""></label>
<div class="col-md-12">
  <input type="submit" style="margin-left:45%" class="btn btn-primary" value="Submit" class="btn btn-primary"/>
</div>
</div>

</fieldset>
</form></div>;

}
?><!--add quiz step 2 end-->

<!--remove quiz-->
<?php if(@$_GET['q']==5) {

$result = mysqli_query($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><b>S.N.</b></td><td><b>Topic</b></td><td><b>Total question</b></td><td><b>Marks</b></td><td><b>Time
limit</b></td><td></td></tr>;
$c=1;
while($row = mysqli_fetch_array($result)) {
    $title = $row['title'];
    $total = $row['total'];
    $sahi = $row['sahi'];
    $time = $row['time'];
    $eid = $row['eid']; echo
'<tr><td>'. $c++. '</td><td>'.
$title.</td><td>'. $total.</t
d><td>'. $sahi* $total.</td>
<td>'. $time.<td>
>
    <td><b><a href="update.php?q=rmquiz&eid='.$eid.'" class="pull-right btn sub1"
style="margin:0px;background:red"><span class="glyphicon glyphicon-trash" aria-hidden="true"></span>&nbsp;<span
class="title1"><b>Remove</b></span></a></b></td></tr>;
}
$c=0;
echo '</table></div></div>;

}
?>

</div><!--container closed-->
</div></div>
</body>
</html>

```

Source code :**Index :**

<!DOCTYPE html>

```

<html >
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Project Worlds || TEST YOUR SKILL </title>
<link rel="stylesheet" href="css/bootstrap.min.css"/>
<link rel="stylesheet" href="css/bootstrap-theme.min.css"/>
<link rel="stylesheet" href="css/main.css">
<link rel="stylesheet" href="css/font.css">
<script src="js/jquery.js" type="text/javascript"></script>

<script src="js/bootstrap.min.js" type="text/javascript"></script>
<link href='http://fonts.googleapis.com/css?family=Roboto:400,700,300' rel='stylesheet' type='text/css'> <?php
if(@$_GET['w'])
{echo'<script>alert('".$_GET['w']."'");</script>';}
?> <script>
function validateForm() {var y = document.forms["form"]["name"].value;      var letters = /^[A-Za-z]+$;/if (y == null || y == "")
{alert("Name must be filled out.");return false;}var z=document.forms["form"]["college"].value;if (z == null || z == "")
{alert("college must be filled out.");return false;}var x = document.forms["form"]["email"].value;var atpos = x.indexOf("@"); var
dotpos = x.lastIndexOf(".");if (atpos<1 || dotpos<atpos+2 || dotpos+2>=x.length) {alert("Not a valid e-mail address.");return
false;}var a = document.forms["form"]["password"].value;if(a == null || a == ""){alert("Password must be filled out");return
false;}if(a.length<5 || a.length>25){alert("Passwords must be 5 to 25 characters long.");return false;} var b =
document.forms["form"]["cpassword"].value;if (a!=b){alert("Passwords must match.");return false;}} </script>

</head>

<body>
<div class="header">
<div class="row">
<div class="col-lg-6">
<span class="logo">Test Your Skill</span></div>
<div class="col-md-2 col-md-offset-4">
<a href="#" class="pull-right btn sub1" data-toggle="modal" data-target="#myModal"><span class="glyphicon glyphicon-log-in"
aria-hidden="true"></span>&nbsp;<span class="title1"><b>Signin</b></span></a></div>
<!-- sign in modal start-->
<div class="modal fade" id="myModal">
<div class="modal-dialog">
<div class="modal-content title1">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
ariahidden="true">&times;</span></button>
<h4 class="modal-title title1"><span style="color:orange">Log In</span></h4>
</div>
<div class="modal-body">
<form class="form-horizontal" action="login.php?q=index.php" method="POST"> <fieldset>

<!-- Text input-->
<div class="form-group">
<label class="col-md-3 control-label" for="email"></label>
<div class="col-md-6">
<input id="email" name="email" placeholder="Enter your email-id" class="form-control input-md" type="email">

</div>

```

```
</div>
```

```
<!-- Password input-->
```

```
<div class="form-group">
```

```
<label class="col-md-3 control-label" for="password"></label>
```

```
<div class="col-md-6">
```

```
<input id="password" name="password" placeholder="Enter your Password" class="form-control input-md" type="password">
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="modal-footer">
```

```
<button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
```

```
<button type="submit" class="btn btn-primary">Log in</button>
```

```
</fieldset>
```

```
</form>
```

```
</div>
```

```
</div><!-- /.modal-content -->
```

```
</div><!-- /.modal-dialog -->
```

```
</div><!-- /.modal -->
```

```
<!-- sign in modal closed-->
```

```
</div><!--header row closed-->
```

```
</div>
```

```
<div class="bg1">
```

```
<div class="row">
```

```
<div class="col-md-7"></div>
```

```
<div class="col-md-4 panel">
```

```
<!-- sign in form begins -->
```

```
<form class="form-horizontal" name="form" action="sign.php?q=account.php" onSubmit="return validateForm()"
method="POST">
```

```
<fieldset>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="name"></label>
```

```
<div class="col-md-12">
```

```
<input id="name" name="name" placeholder="Enter your name" class="form-control input-md" type="text">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="gender"></label>
```

```
<div class="col-md-12">
```

```
<select id="gender" name="gender" placeholder="Enter your gender" class="form-control input-md" >
```

```
<option value="Male">Select Gender</option>
```

```
<option value="M">Male</option>
```

```
<option value="F">Female</option> </select>
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="name"></label>
```

```
<div class="col-md-12">
```

```
<input id="college" name="college" placeholder="Enter your college name" class="form-control input-md" type="text">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label title1" for="email"></label>
```

```
<div class="col-md-12">
```

```
<input id="email" name="email" placeholder="Enter your email-id" class="form-control input-md" type="email">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="mob"></label>
```

```
<div class="col-md-12">
```

```
<input id="mob" name="mob" placeholder="Enter your mobile number" class="form-control input-md" type="number">
```

```
</div>
```

```
</div>
```

```
<!-- Text input-->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="password"></label>
```

```
<div class="col-md-12">
```

```
<input id="password" name="password" placeholder="Enter your password" class="form-control input-md" type="password">
```

```
</div>
```

```
</div>
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for="cpassword"></label>
```

```
<div class="col-md-12">
```

```
<input id="cpassword" name="cpassword" placeholder="Conform Password" class="form-control input-md" type="password">
```

```
</div>
```

```
</div>
```

```
<?php if(@$_GET['q7'])
```

```
{ echo'<p style="color:red;font-size:15px;">'.@$_GET['q7'];}>
```

```
<!-- Button -->
```

```
<div class="form-group">
```

```
<label class="col-md-12 control-label" for=""></label>
```

```
<div class="col-md-12">
```

```
<input type="submit" class="sub" value="sign up" class="btn btn-primary"/>
```

```
</div>
```

```
</div>
```

```

</fieldset>
</form>
</div><!--col-md-6 end-->
</div></div>
</div><!--container end-->

<!--Footer start-->
<div class="row footer">
<div class="col-md-3 box">
<a href="http://www.projectworlds/online-examination" target="_blank">About us</a> </div>
<div class="col-md-3 box">
<a href="#" data-toggle="modal" data-target="#login">Admin Login</a></div> <div
class="col-md-3 box">
<a href="#" data-toggle="modal" data-target="#developers">Developers</a> </div>
<div class="col-md-3 box">
<a href="feedback.php" target="_blank">Feedback</a></div></div>
<!-- Modal For Developers-->
<div class="modal fade title1" id="developers">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
class="sronly">Close</span></button>
<h4 class="modal-title" style="font-family:'typo' "><span style="color:orange">Developers</span></h4>
</div>

<div class="modal-body">
<p>

<div class="row">
<div class="col-md-4">

</div>
<div class="col-md-5">
<a href="https://www.facebook.com/donnieboiii" style="color:#202020; font-family:'typo' ; font-size:18px"
title="Find on Facebook">Lyndon Rife Bermoy</a>
<h4 style="color:#202020; font-family:'typo' ;font-size:16px" class="title1">+639079373999</h4>
<h4 style="font-family:'typo' ">serbermz2020@gmail.com</h4>
<h4 style="font-family:'typo' ">Philippine Science High School - Caraga Region Campus</h4></div></div>
</p>
</div>

</div><!-- /.modal-content -->
</div><!-- /.modal-dialog -->
</div><!-- /.modal -->

<!--Modal for admin login-->
<div class="modal fade" id="login">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">&times;</span><span
class="sronly">Close</span></button>
<h4 class="modal-title"><span style="color:orange;font-family:'typo' ">LOGIN</span></h4>
</div>
<div class="modal-body title1">

```



```

<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
<form role="form" method="post" action="admin.php?q=index.php">
<div class="form-group">
<input type="text" name="uname" maxlength="20" placeholder="Admin user id" class="form-control"/>
</div>
<div class="form-group">
<input type="password" name="password" maxlength="15" placeholder="Password" class="form-control"/> </div>
<div class="form-group" align="center">
<input type="submit" name="login" value="Login" class="btn btn-primary" />
</div>
</form>
</div><div class="col-md-3"></div></div>
</div>
<!--<div class="modal-footer">
<button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
</div-->
</div><!-- /.modal-content -->
</div><!-- /.modal-dialog -->
</div><!-- /.modal -->
<!--footer end-->

</body>
</html>

```

Source code :

Logout :

```

<?php session_start();
if(isset($_SESSION['email'])){
session_destroy();} $ref=
@$_GET['q'];
header("location:$ref");
?>

```