#### EDiEdiGlobe Project

#### Web Development Project

Title : Online Examination System(OES)

Created By : JAYARAJ V

Problem Statement :

In an online examination system examine get their user id and password with his/her admit card. This id is already saved in the examination server. When examine login to the server he/she get his/her profile already register. On the certain time examine gets the message to start the examination. All answers given by examine are saved into the server with his/her profile information. Online examination system also allows to correct the answer if the examine needed to change any answer in the examination time duration, however, after the time duration any change will not allow. This also makes c checking the answer easy and error proof as computers are more accurate than man and provide fast results too. Php is a web base language so we can create an online examination system.

In the world of internet, all task has been done through internet, so we have decide why Exam has not conducted through internet. For convert current exam system into digital exam system, we have build this small Online Exam system project. If this system has build in professional level then it will automate our existing examination system into digitize exam system. In this system it will required less force for execute system and it will be more accurate and less time consuming and at the same time we can conduct more person exam at the same time and it will publish result in a very short time. Below you can find benefits of Online Examination System. If this system has been implemented then examination will not limited in to four wall of class room, but student can part into exam from any place.

Scope of the Project :

1.User Roles and Authentication:

* Define different user roles such as students, instructors, and administrators.
* Implement secure authentication mechanisms to ensure authorized access to the system.

2.Exam Management:

* Allow instructors to create, edit, and delete exams.
* Support various question types including multiple-choice, true/false, fill-in-the-blank, and essay questions.
* Enable instructors to set exam parameters such as duration, start/end times, and allowed resources.
* Provide the ability to organize exams into categories or courses.

3.Exam Taking:

* Allow students to access scheduled exams within the designated time frame.
* Provide an intuitive interface for answering exam questions and navigating between them.
* Implement features to prevent cheating such as time limits, randomizing question order, and disabling copying/printing.

4.Results and Feedback:

* Calculate and store exam results securely.
* Provide immediate feedback to students upon completing exams, including scores and correct/incorrect answers.
* Enable instructors to review individual student performances and overall class statistics.
* Generate detailed reports on exam results for analysis and record-keeping purposes.

5.User Management:

* Allow administrators to manage user accounts including registration, profile updates, and password resets.
* Implement role-based access control to restrict functionalities based on user roles.
* Ensure data privacy and compliance with relevant regulations such as GDPR or CCPA.

6.System Administration:

* Provide administrative tools for managing system settings, configurations, and maintenance tasks.
* Monitor system performance, logs, and security incidents.
* Implement backup and recovery mechanisms to prevent data loss.

7.Security and Integrity:

* Implement robust security measures to protect against unauthorized access, data breaches, and tampering.
* Encrypt sensitive data such as user credentials and exam results.
* Regularly update software components and apply security patches to mitigate vulnerabilities.

8.Scalability and Performance:

* Design the system to handle a large number of concurrent users and exams.
* Optimize performance to ensure fast response times and minimal downtime during peak usage periods.
* Plan for scalability by utilizing scalable infrastructure and cloud services if necessary.

9.User Experience (UX):

* Design an intuitive and user-friendly interface for all stakeholders.
* Ensure accessibility for users with disabilities by adhering to accessibility standards such as WCAG.

10.Support and Training:

* Provide user support channels such as FAQs, help documentation, and customer service.
* Conduct training sessions for instructors and administrators on how to effectively use the system.

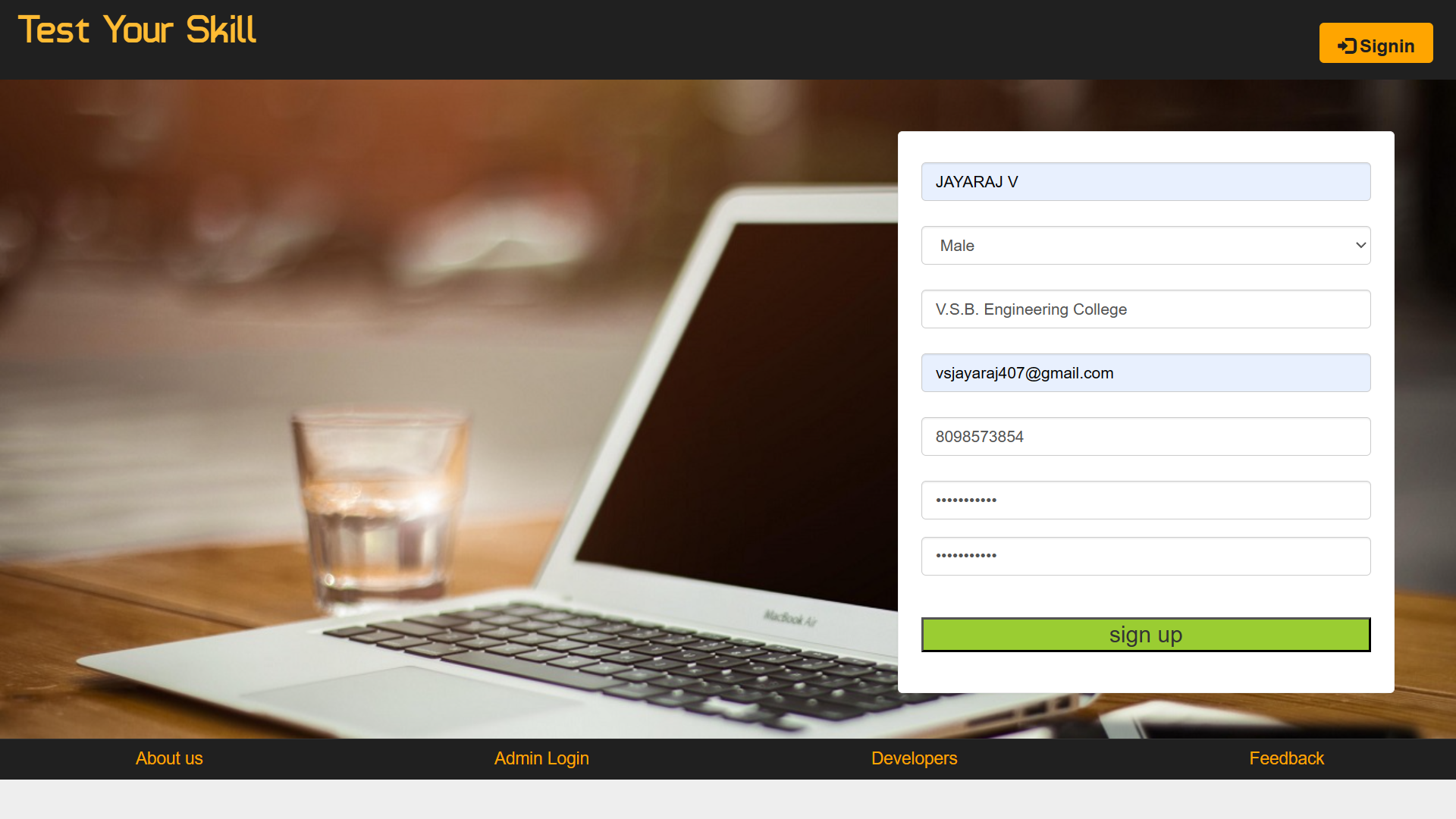
11.Compliance and Legal Considerations:

* Ensure compliance with relevant educational standards and regulations governing online assessments.
* Address legal aspects such as copyright issues, intellectual property rights, and data protection laws.

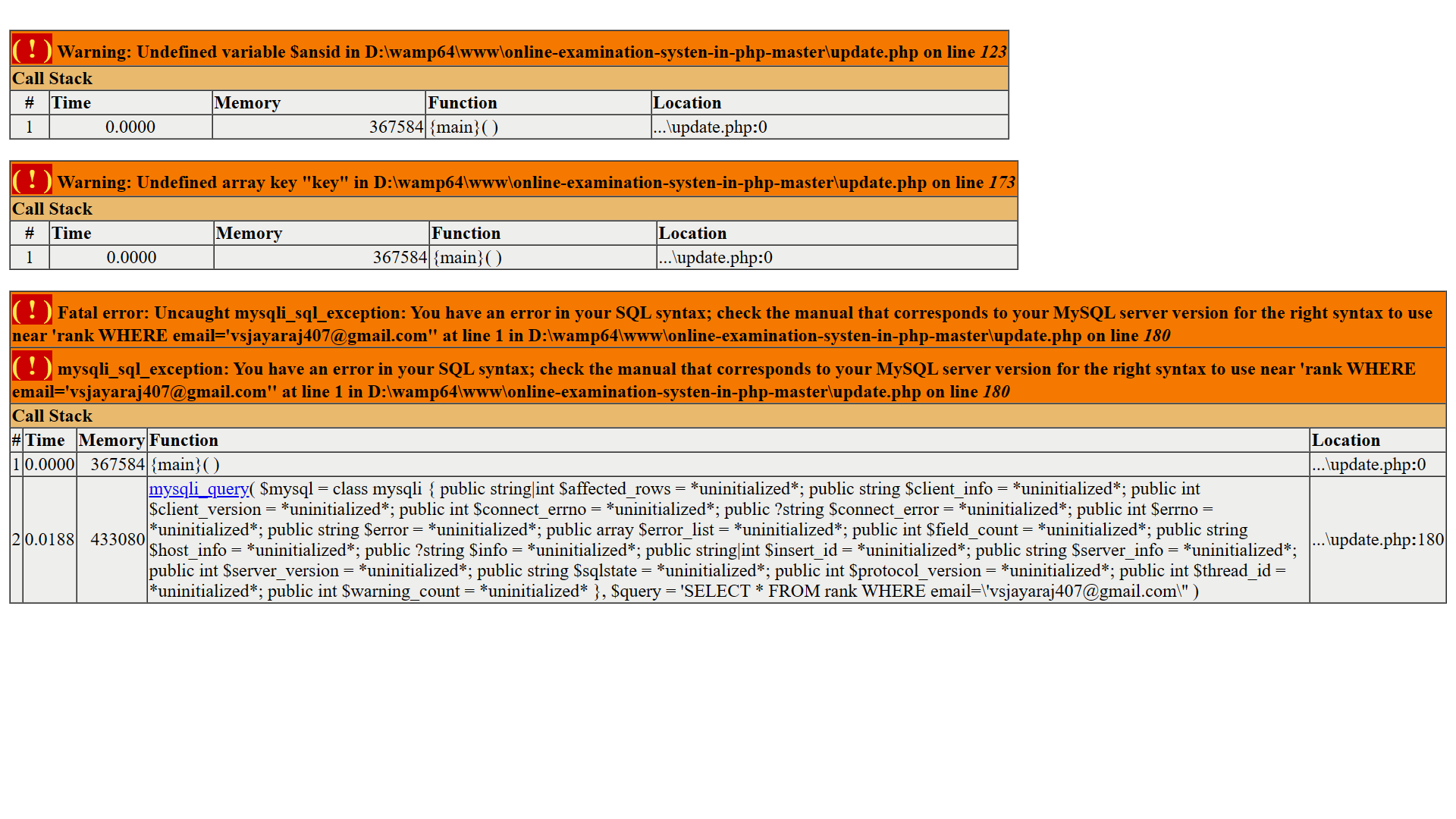
12.Integration and Interoperability:

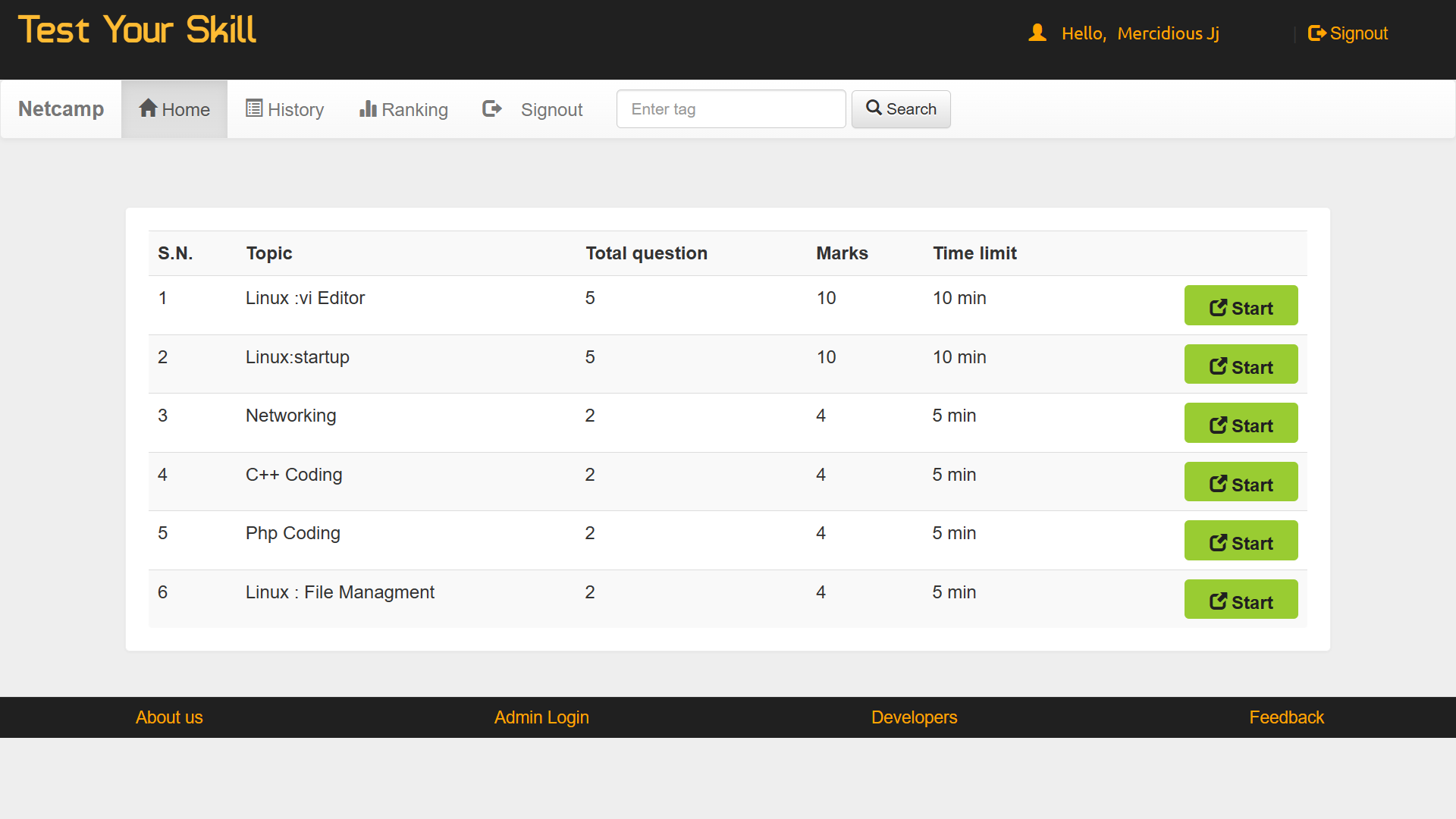
* Integrate with existing learning management systems (LMS), student information systems (SIS), or other educational platforms if required.
* Provide APIs or interoperability standards for seamless integration with third-party applications or services.

1. Login system secured by password :

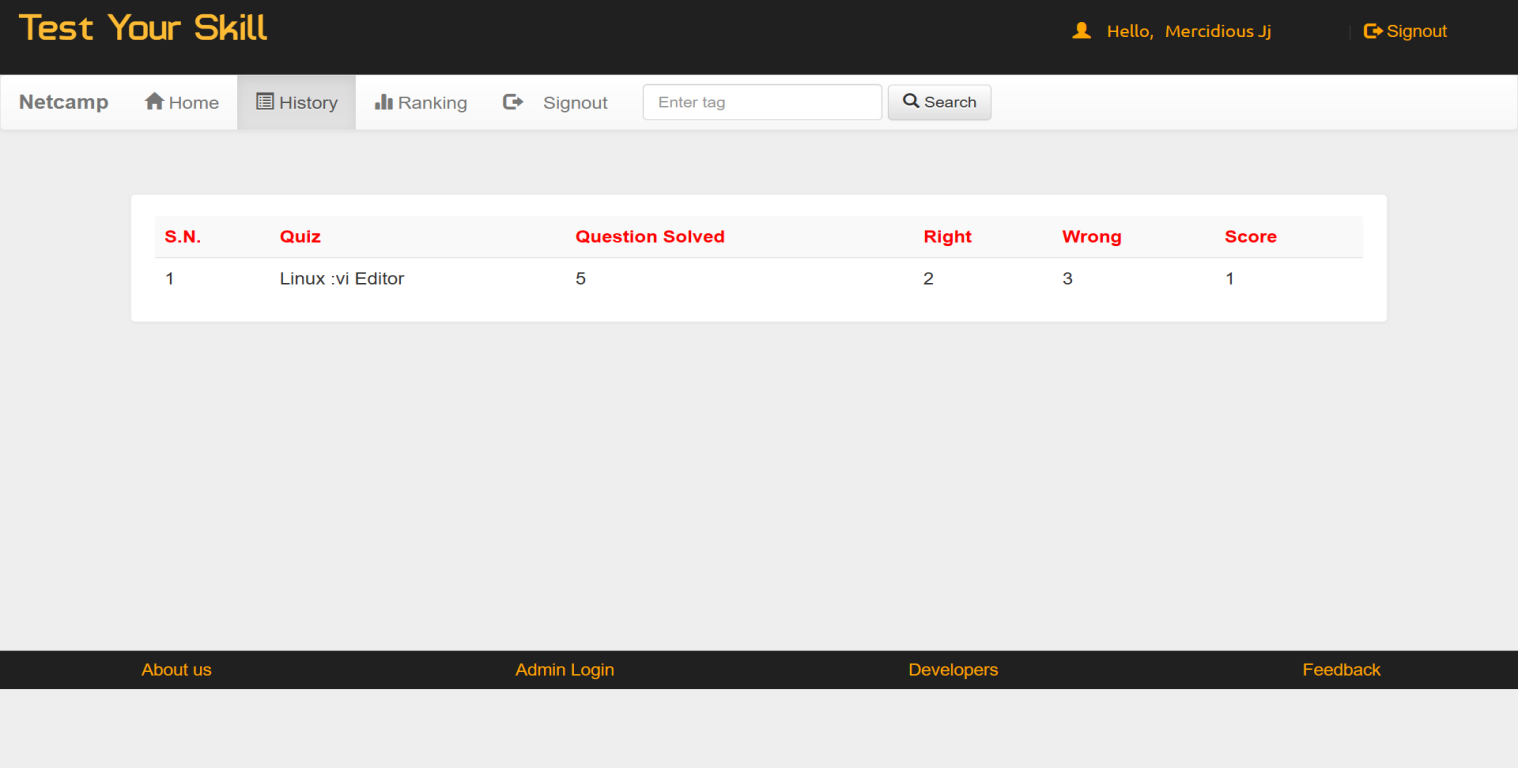


1. Ability to save answer :

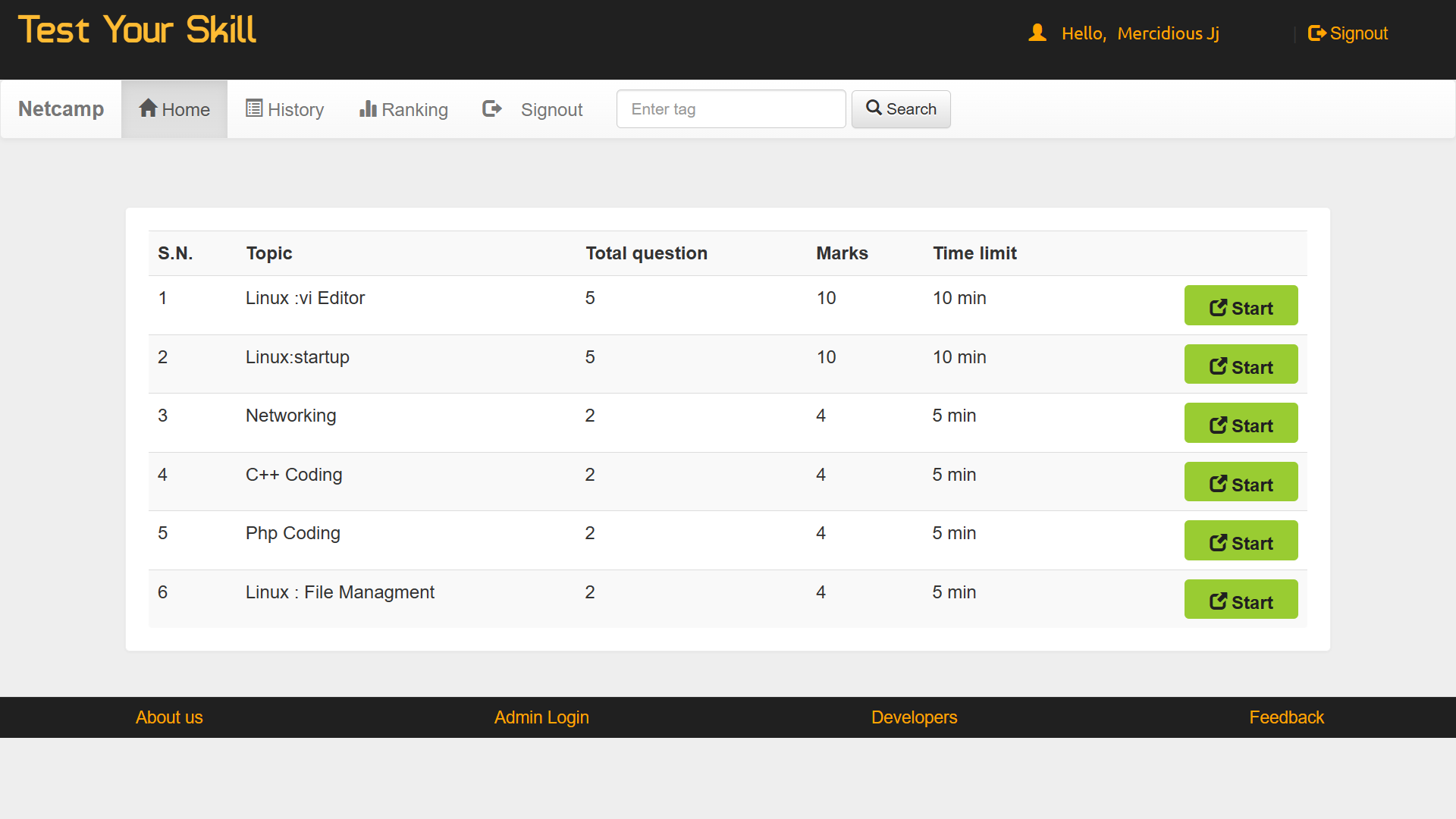


3.Ability to save the answers along with questions :

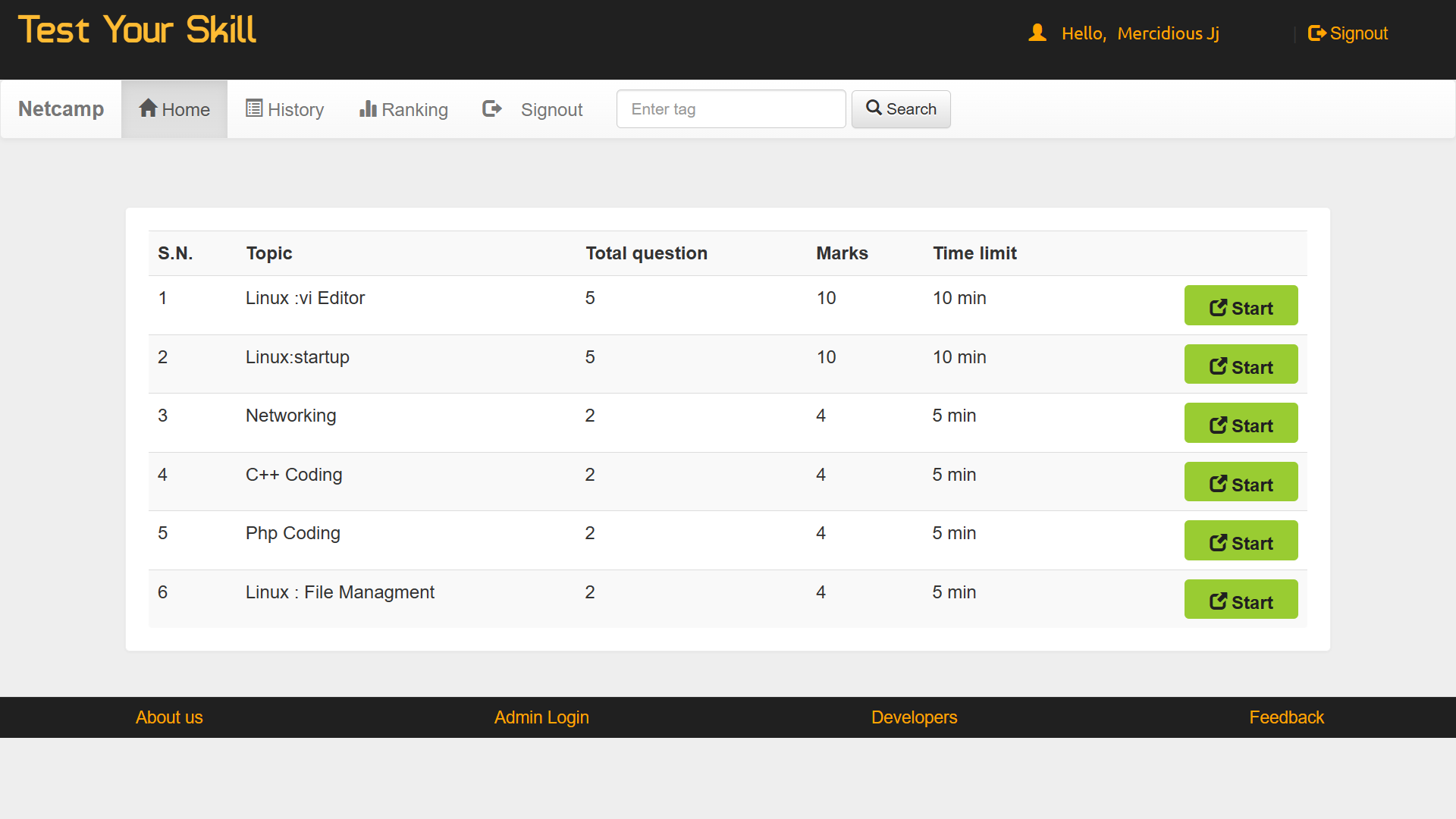
4.Answer checking system :



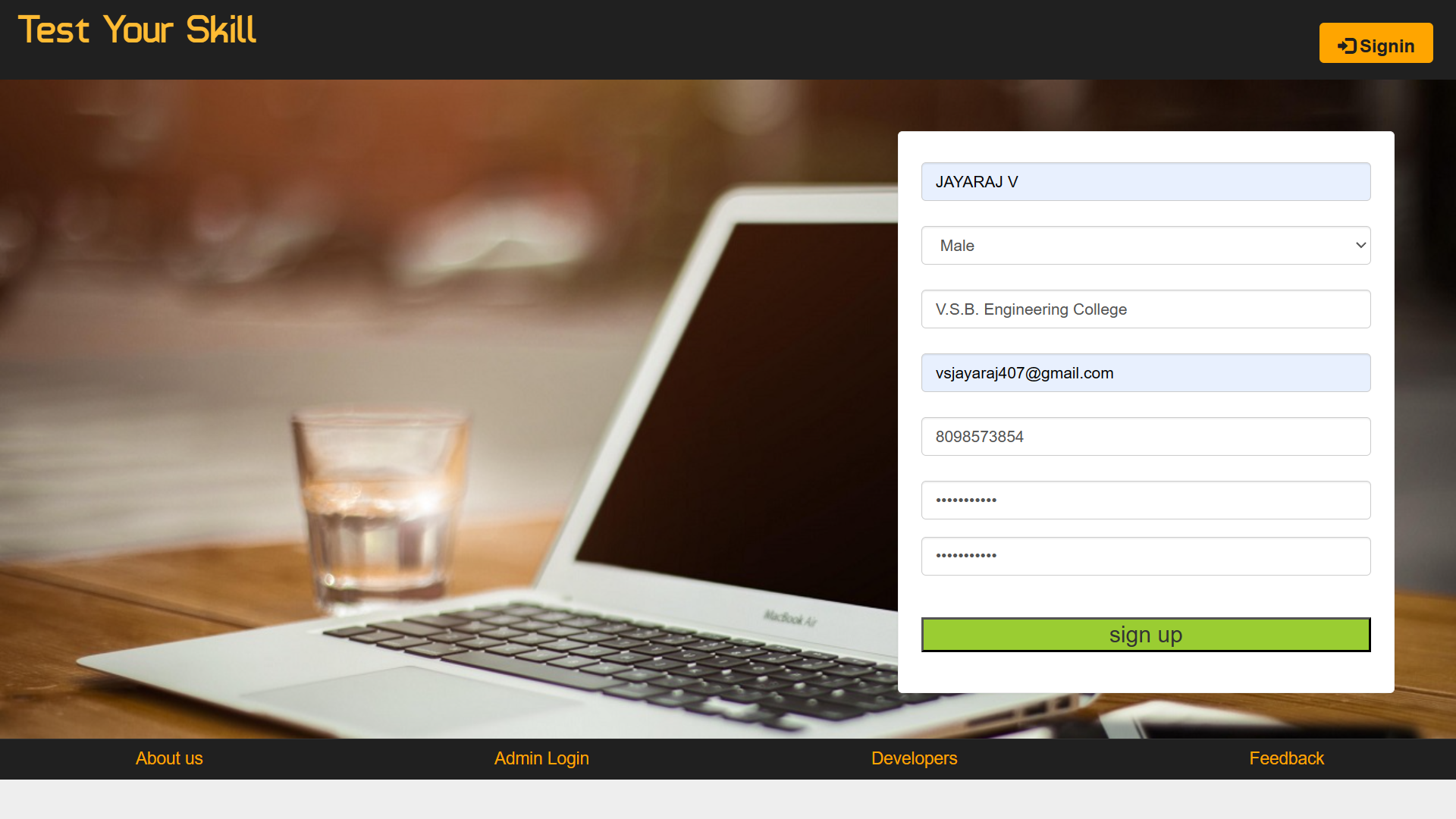
1. Update Profile Option :



1. Log out option :

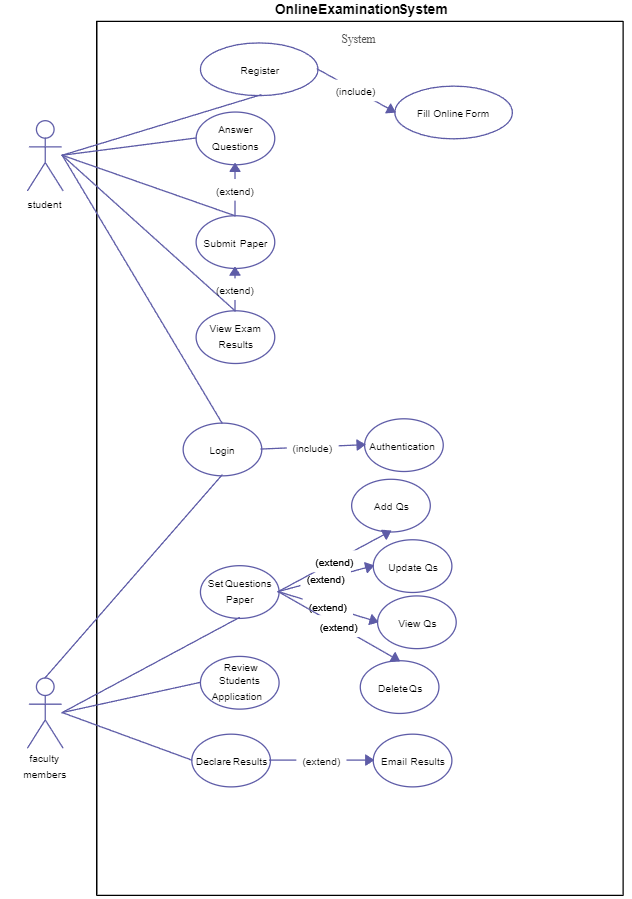


7.Admin login :



**USECASE DIAGRAM:**

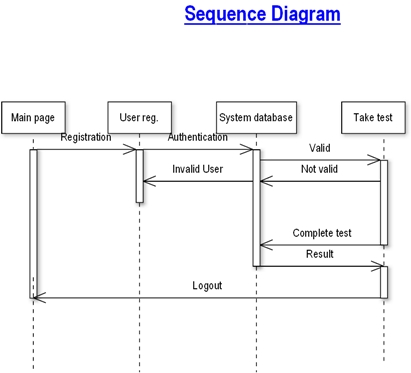
A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors.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

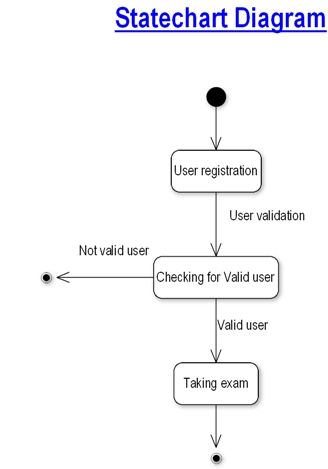


**CLASS DIAGRAM:**

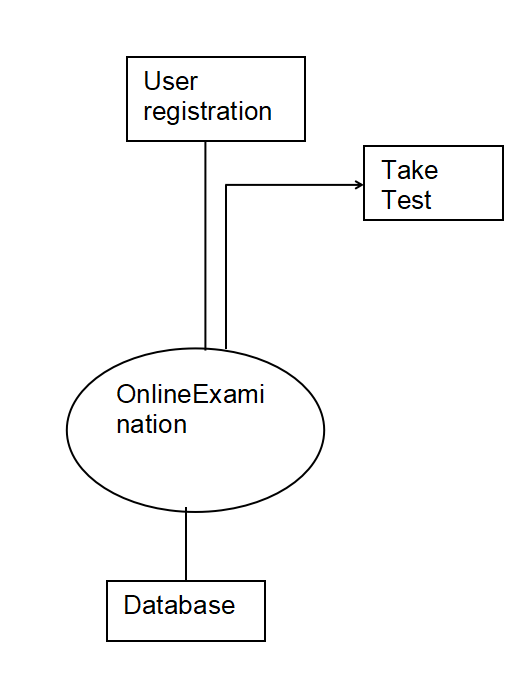
Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relating ships. There is most common diagram in modeling the object oriented systems and are used to give the static view of a system. It shows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



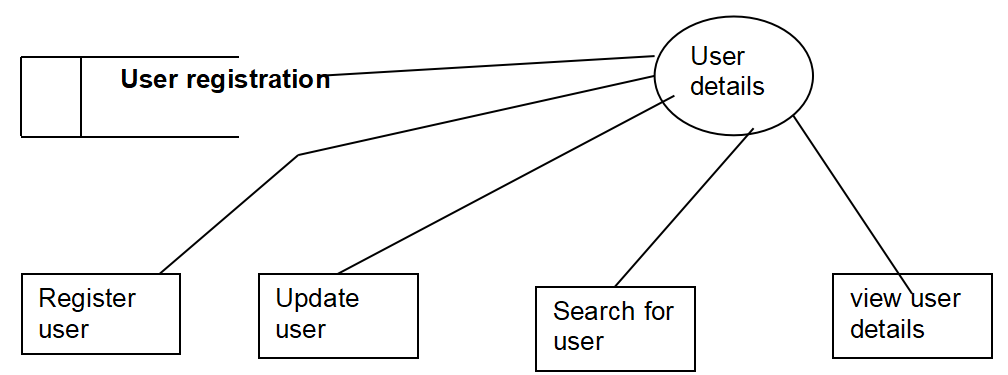
State chart diagram :



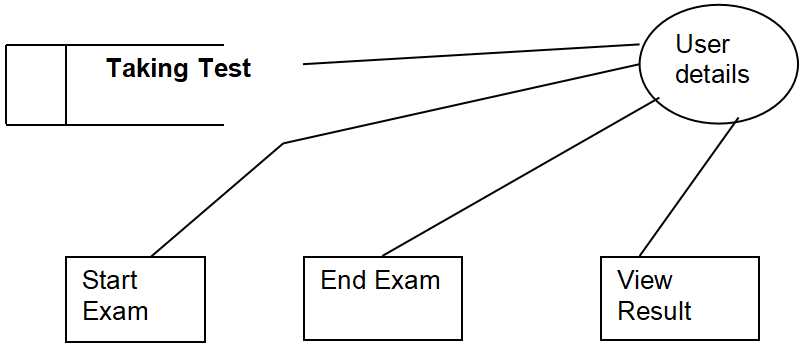
Data Flow Diagram :



User registration :



Taking test :



Overview of technology used

PHP :

PHP: Hypertext Preprocessor, is a widely used, general-purpose [scripting language](http://en.wikipedia.org/wiki/Scripting_language" \o "Scripting language) that was originally designed for [web development](http://en.wikipedia.org/wiki/Web_development" \o "Web development), to produce [dynamic web pages](http://en.wikipedia.org/wiki/Dynamic_web_page" \o "Dynamic web page). It can be embedded into [HTML](http://en.wikipedia.org/wiki/HTML" \o "HTML) and generally runs on a [web server](http://en.wikipedia.org/wiki/Web_server" \o "Web server), which needs to be configured to process PHP code and create [web page](http://en.wikipedia.org/wiki/Web_page" \o "Web page) content from it. It can be deployed on most web servers and on almost every [operating system](http://en.wikipedia.org/wiki/Operating_system" \o "Operating system) and [platform](http://en.wikipedia.org/wiki/Platform_(computing)" \o "Platform (computing)) free of charge.

PHP was originally created by [Rasmus Lerdorf](http://en.wikipedia.org/wiki/Rasmus_Lerdorf" \o "Rasmus Lerdorf) in [1995](http://en.wikipedia.org/wiki/1995" \o "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](http://en.wikipedia.org/wiki/De_facto_standard" \o "De facto standard) for PHP as there is no [formal specification](http://en.wikipedia.org/wiki/Formal_specification" \o "Formal specification).PHP is [free software](http://en.wikipedia.org/wiki/Free_software" \o "Free software) released under the [PHP License](http://en.wikipedia.org/wiki/PHP_License" \o "PHP License), which is incompatible with the [GNU General Public License](http://en.wikipedia.org/wiki/GNU_General_Public_License" \o "GNU General Public License) (GPL) because of restrictions on the use of the term PHP

PHP has evolved to include a [command line interface](http://en.wikipedia.org/wiki/Command_line_interface" \o "Command line interface) capability and can also be used in [standalone](http://en.wikipedia.org/wiki/Standalone_software" \o "Standalone software) [graphical applications](http://en.wikipedia.org/wiki/Graphical_user_interface" \o "Graphical user interface).

**USAGE of PHP**

PHP is a general-purpose scripting language that is especially suited for [web development](http://en.wikipedia.org/wiki/Web_development" \o "Web development). PHP generally runs on a [web server](http://en.wikipedia.org/wiki/Web_server" \o "Web server). Any PHP code in a requested file is [executed](http://en.wikipedia.org/wiki/Execution_(computing)" \o "Execution (computing)) by the PHP runtime, usually to create [dynamic web page](http://en.wikipedia.org/wiki/Dynamic_web_page" \o "Dynamic web page) content. It can also be used for [command-line](http://en.wikipedia.org/wiki/Command-line" \o "Command-line) scripting and [client-side](http://en.wikipedia.org/wiki/Client-side" \o "Client-side) [GUI](http://en.wikipedia.org/wiki/Graphical_user_interface" \o "Graphical user interface) applications. PHP can be deployed on most [web servers](http://en.wikipedia.org/wiki/Web_server" \o "Web server), many [operating systems](http://en.wikipedia.org/wiki/Operating_system" \o "Operating system) and [platforms](http://en.wikipedia.org/wiki/Platform_(computing)" \o "Platform (computing)), and can be used with many [relational database management systems](http://en.wikipedia.org/wiki/Relational_database_management_system" \o "Relational database management system). 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](http://en.wikipedia.org/wiki/Filter_(software)" \o "Filter (software)), 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](http://en.wikipedia.org/wiki/Parser" \o "Parser) [compiles](http://en.wikipedia.org/wiki/Compiler" \o "Compiler) input to produce [byte code](http://en.wikipedia.org/wiki/Bytecode" \o "Bytecode) for processing by the [Zend Engine](http://en.wikipedia.org/wiki/Zend_Engine" \o "Zend Engine), giving improved performance over its [interpreter](http://en.wikipedia.org/wiki/Interpreter_(computing)" \o "Interpreter (computing)) predecessor.

Originally designed to create dynamic web pages, PHP now focuses mainly on [server-side scripting](http://en.wikipedia.org/wiki/Server-side_scripting" \o "Server-side scripting), and it is similar to other server-side scripting languages that provide dynamic content from a web server to a [client](http://en.wikipedia.org/wiki/Client_(computing)" \o "Client (computing)), such as [Microsoft](http://en.wikipedia.org/wiki/Microsoft" \o "Microsoft)'s [Active Server Pages](http://en.wikipedia.org/wiki/Active_Server_Pages" \o "Active Server Pages), [Sun Microsystems](http://en.wikipedia.org/wiki/Sun_Microsystems" \o "Sun Microsystems)' [JavaServer Pages](http://en.wikipedia.org/wiki/JavaServer_Pages" \o "JavaServer Pages) and [mod\_perl](http://en.wikipedia.org/wiki/Mod_perl" \o "Mod perl). PHP has also attracted the development of many [frameworks](http://en.wikipedia.org/wiki/Software_framework" \o "Software framework) that provide building blocks and a design structure to promote [rapid application development](http://en.wikipedia.org/wiki/Rapid_application_development" \o "Rapid application development) (RAD). Some of these include [CakePHP](http://en.wikipedia.org/wiki/CakePHP" \o "CakePHP), [Symfony](http://en.wikipedia.org/wiki/Symfony" \o "Symfony), [CodeIgniter](http://en.wikipedia.org/wiki/CodeIgniter" \o "CodeIgniter), and [Zend Framework](http://en.wikipedia.org/wiki/Zend_Framework" \o "Zend Framework), offering features similar to other [web application frameworks](http://en.wikipedia.org/wiki/List_of_web_application_frameworks" \o "List of web application frameworks).

**HTML :**

**HTML**, which stands for **Hyper Text Markup Language**, is the predominant [markup language](http://en.wikipedia.org/wiki/Markup_language" \o "Markup language) for [web pages](http://en.wikipedia.org/wiki/Web_page" \o "Web page). It provides a means to create [structured documents](http://en.wikipedia.org/wiki/Structured_document" \o "Structured document) by denoting structural [semantics](http://en.wikipedia.org/wiki/Semantic" \o "Semantic) for text such as headings, paragraphs, lists etc as well as for links, quotes, and other items. It allows [images and objects](http://en.wikipedia.org/wiki/HTML_element" \l "Images_and_objects" \o "HTML element) to be embedded and can be used to create [interactive forms](http://en.wikipedia.org/wiki/HTML_element" \l "Forms" \o "HTML element). It is written in the form of [HTML elements](http://en.wikipedia.org/wiki/HTML_element" \o "HTML element) consisting of "tags" surrounded by [angle brackets](http://en.wikipedia.org/wiki/Brackets" \l "Angle_brackets_or_chevrons_.3C_.3E" \o "Brackets) within the web page content. It can include or can load [scripts](http://en.wikipedia.org/wiki/Scripting_language" \o "Scripting language) in languages such as [JavaScript](http://en.wikipedia.org/wiki/JavaScript" \o "JavaScript) which affect the behavior of HTML processors like [Web browsers](http://en.wikipedia.org/wiki/Web_browser" \o "Web browser); and [Cascading Style Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets" \o "Cascading Style Sheets) (CSS) to define the appearance and layout of text and other material. The [W3C](http://en.wikipedia.org/wiki/W3C" \o "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](http://en.wikipedia.org/wiki/E-mail" \o "E-mail) clients allow the use of a subset of HTML (often ill-defined) to provide formatting and [semantic](http://en.wikipedia.org/wiki/Semantic_web" \o "Semantic web) markup not available with [plain text](http://en.wikipedia.org/wiki/Plain_text" \o "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](http://en.wikipedia.org/wiki/GUI" \o "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](http://en.wikipedia.org/wiki/Phishing" \o "Phishing) attacks, because it can confuse [spam](http://en.wikipedia.org/wiki/E-Mail_spam" \o "E-Mail spam) filters and because the message size is larger than plain text.

**HTML APPLICATION**

An HTML Application is a [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows" \o "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](http://en.wikipedia.org/wiki/HTTP_cookie" \o "HTTP cookie). An HTA runs as a fully trusted application and therefore has more privileges, like creation/editing/removal of files and [Windows Registry](http://en.wikipedia.org/wiki/Windows_Registry" \o "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](http://en.wikipedia.org/wiki/EXE" \o "EXE)) and executed from local file system.

**JAVASCRIPT :**

**JavaScript** is an [object-oriented](http://en.wikipedia.org/wiki/Object-oriented" \o "Object-oriented) [scripting language](http://en.wikipedia.org/wiki/Scripting_language" \o "Scripting language) used to enable [programmatic](http://en.wikipedia.org/wiki/Computer_programming" \o "Computer programming) access to objects within both the [client application](http://en.wikipedia.org/wiki/Client_(computing)" \o "Client (computing)) and other [applications](http://en.wikipedia.org/wiki/Application_software" \o "Application software). It is primarily used in the form of [client-side JavaScript](http://en.wikipedia.org/wiki/Client-side_JavaScript" \o "Client-side JavaScript), implemented as an integrated component of the [web browser](http://en.wikipedia.org/wiki/Web_browser" \o "Web browser), allowing the development of enhanced [user interfaces](http://en.wikipedia.org/wiki/User_interface" \o "User interface) and dynamic [websites](http://en.wikipedia.org/wiki/Website" \o "Website). JavaScript is a [dialect](http://en.wikipedia.org/wiki/Programming_language_dialect" \o "Programming language dialect) of the [ECMAScript](http://en.wikipedia.org/wiki/ECMAScript" \o "ECMAScript) standard and is characterized as a [dynamic](http://en.wikipedia.org/wiki/Dynamic_language" \o "Dynamic language), [weakly typed](http://en.wikipedia.org/wiki/Weak_typing" \o "Weak typing), [prototype-based](http://en.wikipedia.org/wiki/Prototype-based_programming" \o "Prototype-based programming) language with [first-class functions](http://en.wikipedia.org/wiki/First-class_function" \o "First-class function). JavaScript was influenced by many languages and was designed to look like [Java](http://en.wikipedia.org/wiki/Java_(programming_language)" \o "Java (programming language)), but to be easier for non-programmers to work with.

**PROTOTYPE-BASED :**

JavaScript uses [prototypes](http://en.wikipedia.org/wiki/Prototype-based_programming" \o "Prototype-based programming) instead of [classes](http://en.wikipedia.org/wiki/Class_(computer_science)" \o "Class (computer science)) for [inheritance](http://en.wikipedia.org/wiki/Inheritance_(computer_science)" \o "Inheritance (computer science)). 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](http://en.wikipedia.org/wiki/Method_(computer_science)" \o "Method (computer science)) 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](http://en.wikipedia.org/wiki/HTML" \o "HTML) pages and interact with the [Document Object Model](http://en.wikipedia.org/wiki/Document_Object_Model" \o "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](http://en.wikipedia.org/wiki/Gmail" \o "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](http://en.wikipedia.org/wiki/AJAX" \o "AJAX) similarly exploits this strength.

A [JavaScript engine](http://en.wikipedia.org/wiki/JavaScript_engine" \o "JavaScript engine) (also known as *JavaScript interpreter* or *JavaScript implementation*) is an [interpreter](http://en.wikipedia.org/wiki/Interpreter_(computing)" \o "Interpreter (computing)) that interprets JavaScript [source code](http://en.wikipedia.org/wiki/Source_code" \o "Source code) and executes the [script](http://en.wikipedia.org/wiki/Computer_program" \o "Computer program) accordingly. The first JavaScript engine was created by [Brendan Eich](http://en.wikipedia.org/wiki/Brendan_Eich" \o "Brendan Eich) at Netscape Communications Corporation, for the [Netscape Navigator](http://en.wikipedia.org/wiki/Netscape_Navigator" \o "Netscape Navigator) [web browser](http://en.wikipedia.org/wiki/Web_browser" \o "Web browser). A web browser is by far the most common host environment for JavaScript. Web browsers typically use the public [API](http://en.wikipedia.org/wiki/Application_programming_interface" \o "Application programming interface) to create "host objects" responsible for reflecting the [DOM](http://en.wikipedia.org/wiki/Document_Object_Model" \o "Document Object Model) into JavaScript.

**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**.

### What is Database administrator(DBA) ?

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

### What is Database user(DBU) ?

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

### Download MySQL Database

If you have installed PHP’s WAMP or XAMPP server, then mysql database already exists. if you don’t have then download mysql database from here[http://www.mysql.com](http://www.phptpoint.com/mysql/)

**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 | VARCHAR(20) |  |
| RPWD | NULL | VARCHAR(20) |  |
| UTYPE | NULL | VARCHAR(20) |  |
| QUE | NULL | VARCHAR(500) |  |
| ANS | NULL | 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 |  |

**1**

**FEASIBILITY STUDY:**

Feasibility study is conducted once the problem is clearly understood. Feasibility study is a high level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points.

1. Technical Feasibility

2. Economical Feasibility

3. Operational Feasibility.

1. Technical Feasibility :

The project entitles "Courier Service System” is technically feasibility because of the below mentioned feature. The project was developed in Java which Graphical User Interface.

It provides the high level of reliability, availability and compatibility. All these make Java an appropriate language for this project. Thus the existing software Java is a powerful language.

1. Economical Feasibility :

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and manpower utilization are expected to go up by 80-90% approximately. The costs incurred of not creating the system are set to be great, because precious time can be wanted by manually.

1. Operational Feasibility :

In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per their requirements and necessaries.

**Implementation:**

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification.

It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods a part from planning. Two major tasks of preparing the implementation are education and training of the users and testing of the system.

The more complex the system being implemented, the more involved will be the systems analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

Source code for Index Page

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

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

<title>Online Examination System </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 aria-hidden="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-12control-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="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">

     <img src="image/JJ\_PIC.jpg" width=100 height=120 alt="JJ" class="img-rounded">

     </div>

     <div class="col-md-5">

    <h4 style="font-family:'typo' ">JAYARAJ V</h4>

    <h4 style="color:#202020; font-family:'typo' ;font-size:16px" class="title1">+91 8098573854</h4>

    <a href="mailto:https://mail.google.com/mail/u/0/?tab=rm&ogbl#inbox" title="Mail Me"><h4 style="font-family:'typo' " targt="blank">jayaraj.veluchamy@gmail.com</h4></a>

    <h4 style="font-family:'typo' ">V.S.B. Engineering College, Karur</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="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>