

BABU RAM DEGREE COLLEGE



H.N.B GARHWAL UNIVERSITY



**SESSION 2019-2020
SYNOPSIS
ON
Online Examination System
BCA 3RD YEAR**

SUBMITTED TO –
Mr. Yogesh Kashyap

SUBMITTED BY-
Mustafa Hasan

Acknowledgement

The Submission of this Project File give me an opportunity to convey my gratitude to all those people who come to my rescue when I needed help for completing this project.

First and foremost, I would also like to express my profound gratitude and sincere acknowledgement to Mr. Yogesh Kashyap.
She has been a source of perpetual inspiration to me.

My sincere acknowledgement to all the Staff Members (Department Of Computer Science) who were to my side whenever I needed them throughout the development of this project report.

Last but Not Least, My heartfelt thanks My Family member, who helped me each and Every situation even to completing, project also.

I have taken effort in this project. However, it would not have been possible without kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

I would like to express my special thanks of gratitude to my teachers, as well as our teacher **Mr. Yogesh Kashyap**. Sir Who gave me the golden opportunity to do this wonderful project on the topic “**Online Examination**” which thankful to them.
Their support and instruction for the project are priceless.

My thanks and appreciation also goes to my classmates who helped also goes to my classmates who helped me in developing the project and laboratory Assistant Who helped me out with this abilities.

MUSTAFA HASAN

Certificate

It is certified that “Major Project Report “entitled “Teacher” Submitted by *Mr. Yogesh Kashyap*.

For The Partial Fulfillment of Requirement for the award of the B.sc From Babu Ram Degree College Of Roorkee is a record of this own work carried out by under my supervision and guidance

I wish his all the success in his life

TABLE OF CONTENTS

| <u>S. N</u> | <u>ABSTRACT</u> |
|--------------------|------------------------|
|--------------------|------------------------|

| | | |
|-----------|--------------------------|-----------|
| 1. | INTRODUCTION | 6 |
| 2. | ANALYSIS | 9 |
| 2.1 | SYSTEM ANALYSIS | 10 |
| 2.2 | SYSTEM SPECIFICATIONS | 12 |
| 3. | DESIGN APPROACH | 13 |
| 3.1 | INTRODUCTION TO DESIGN | 14 |
| 3.2 | GANT CHART | 15 |
| 3.3 | PERT CHART | 16 |
| 3.4 | UML DIAGRAMS | 17 |
| 3.5 | DATA FLOW DIAGRAMS | 27 |
| 3.6 | E-R DIAGRAMS | 33 |
| 4. | PROJECT MODULES | 34 |
| 5. | FEASIBILITY STUDY | 40 |
| 6. | IMPLEMENTATION | 42 |
| 7. | TESTING | 43 |
| 7.1 | TEST CASES | 57 |
| 8. | CONCLUSION | 59 |

| | | |
|------------|----------------------------|----|
| 9. | FUTURE ENHANCEMENTS | 61 |
| 10. | BIBLIOGRAPHY | 63 |
| 11. | OUTPUT SCREENS | 65 |
| 12. | CODING PART | 76 |

Introduction

INTRODUCTION:

Online examinations are an important method of evaluating the success potential of students. This research effort the individuals under consideration were students who would be enrolling in computer courses or Technologies Registrations. A prototype of a web-based placement examination system is described from the standpoint of the research effort, end user, and software development.

An on-line educational system including exam processing and electronic journal features. An instructor builds a course based questions which on-line contain in identification of assignments. Which are compiled into an on-line exam syllabus?

Users enrolled in the platform may access the electronic details they provided and perform various functions with the on-line educational system in order to participate in the on-line examinations. Users can receive an on-line exam, having multimedia content, for the course, and they can electronically provide answers for the exam. And after Completion of their duration of exam they are provided the grade or marks secured in their examinations.

Online 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 project, 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 user or student. And the details of students who attempted Online Examination are maintained at administrator.

Analysis

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:-

- Intel core i5 Processor.
- 2 GB Ram
- 512 KB Cache Memory
- Hard disk 300 GB
- Microsoft Compatible 101 or more Key Board

Software Requirements: -

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

Design

INTRODUCTION:

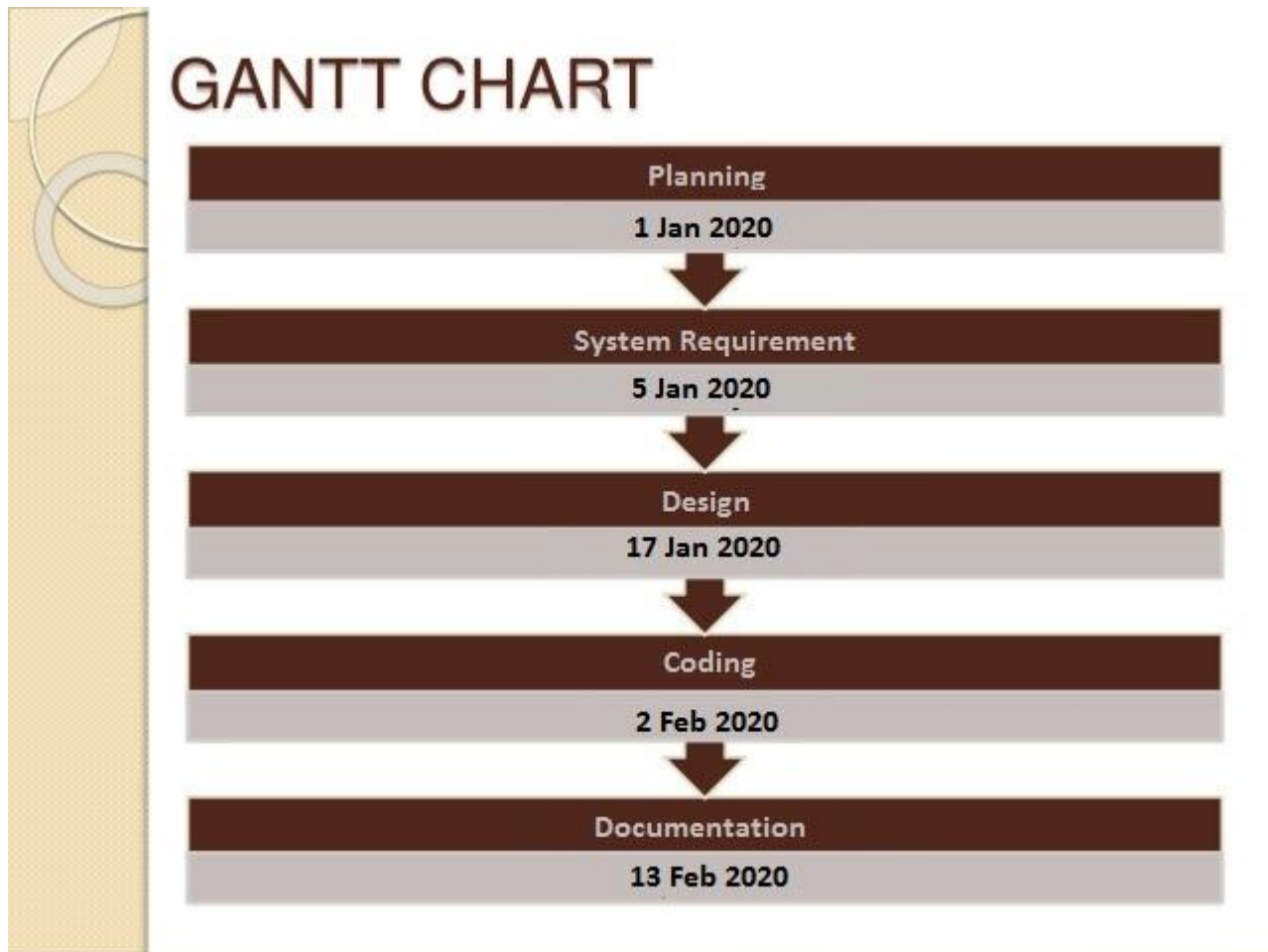
Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

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.

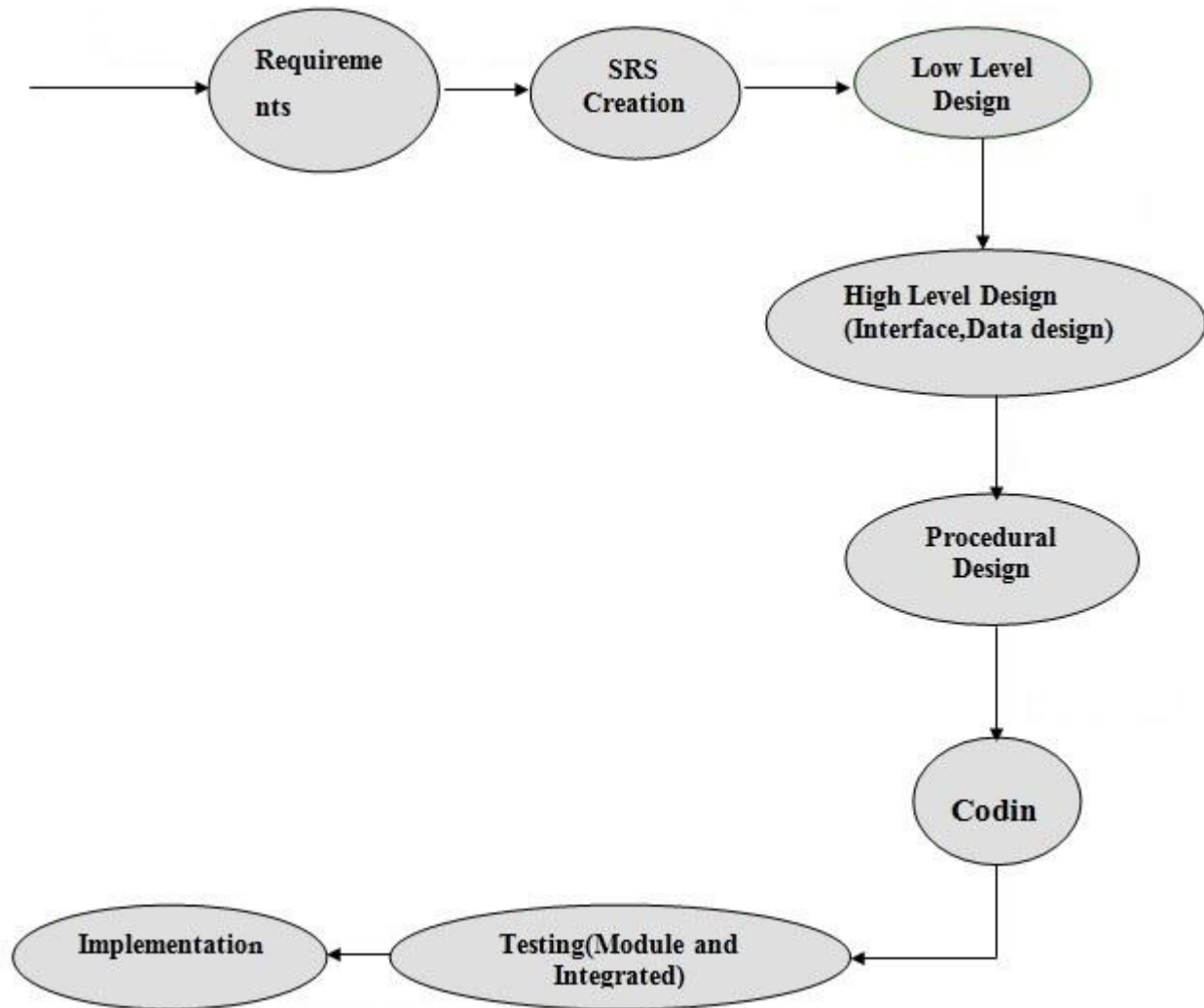
The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the 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.

Gant Chart



PERT Chart



UML Diagrams:

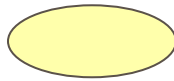
Actor:

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.

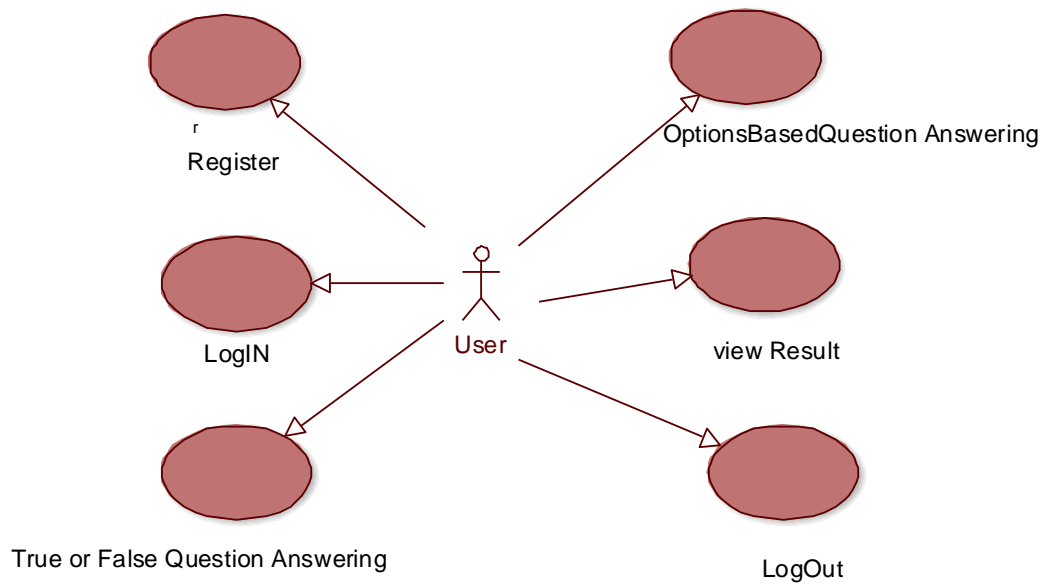
Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- 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 Actor Receiver.

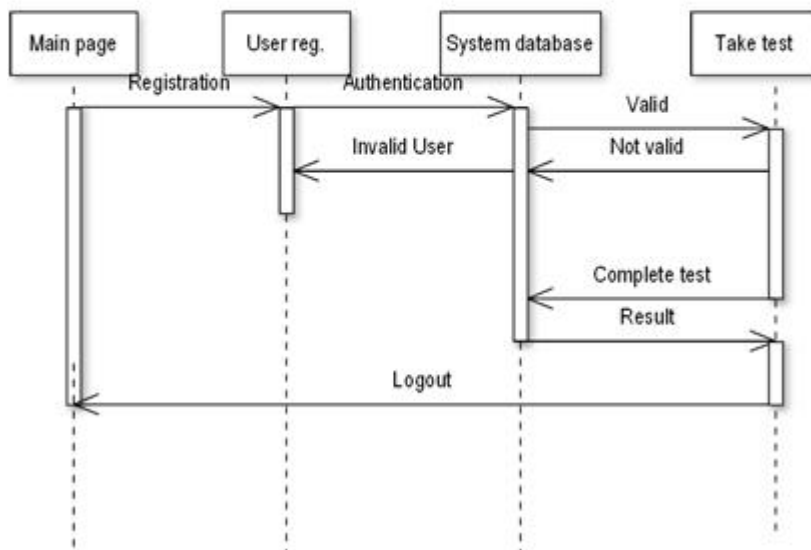


SEQUENCE DIAGRAM:

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

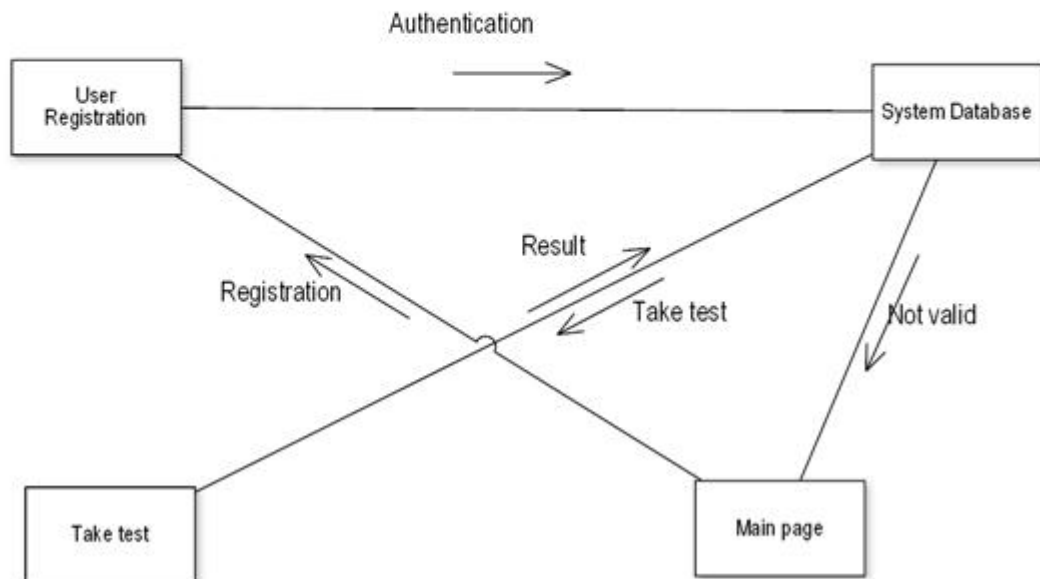
Sequence Diagram



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



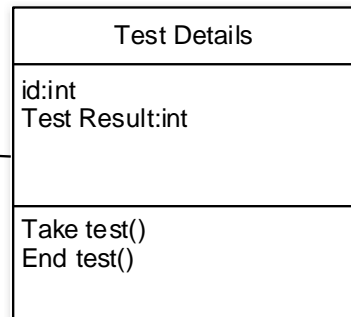
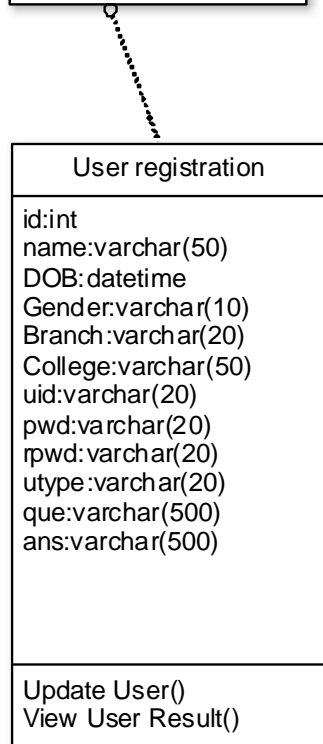
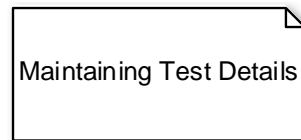
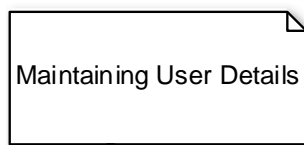
CLASS DIAGRAM:

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 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.

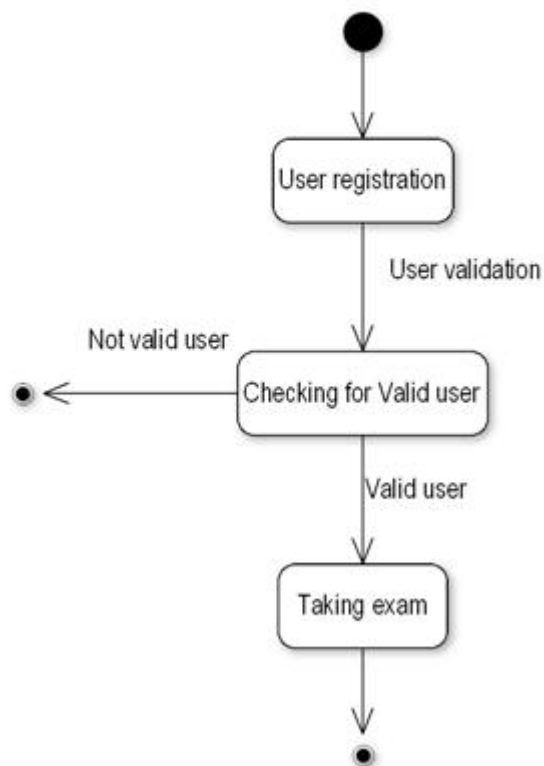
CLASS:

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 performs 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 processes that embody the applications.

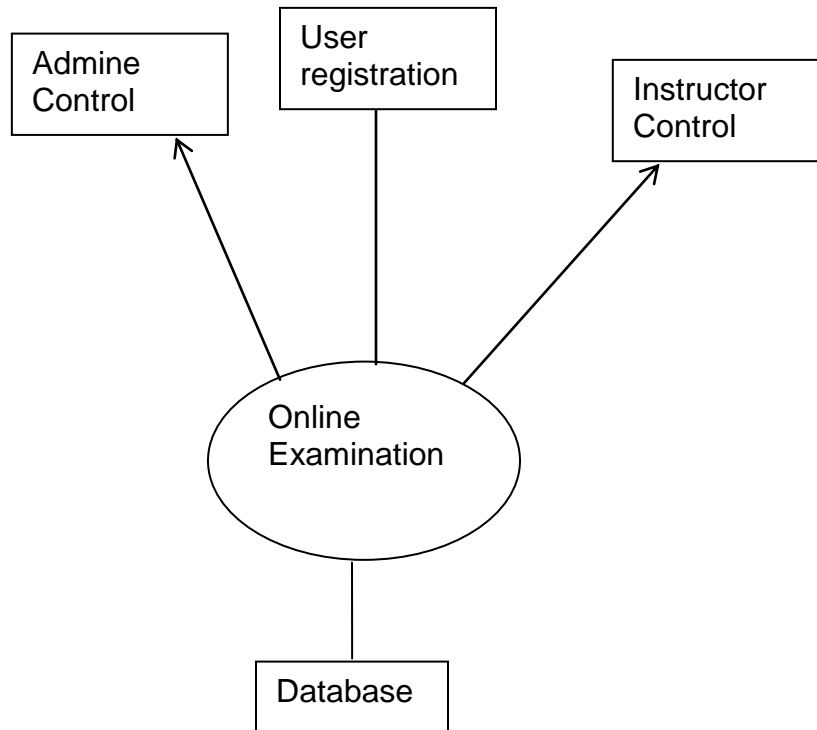
A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

RULES FOR DFD:

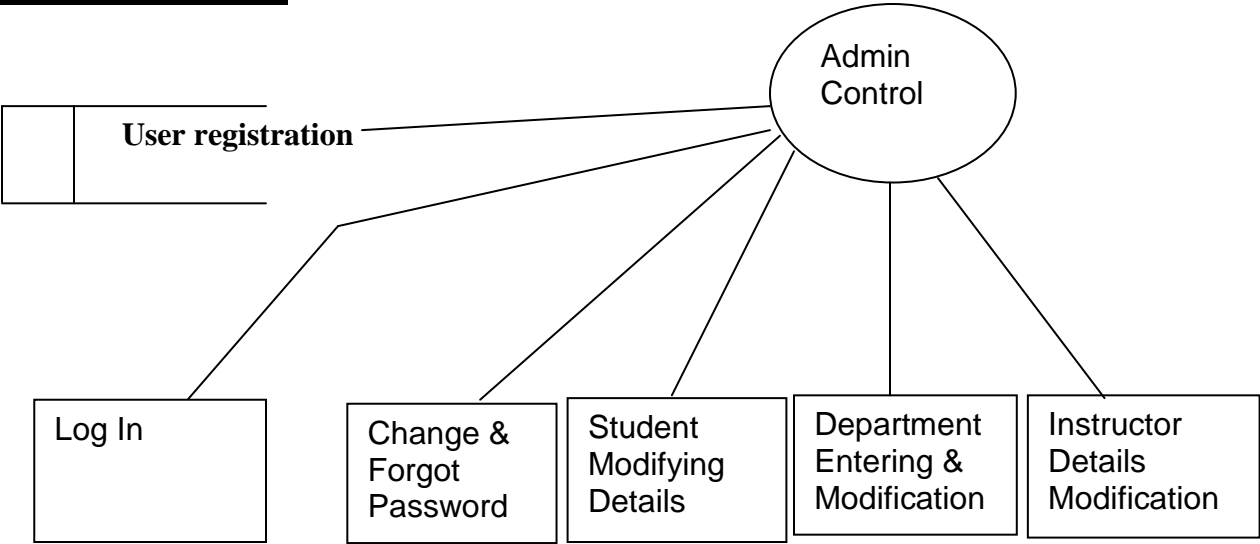
- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions
- Reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with Rounded circles.
- A process is required for 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:

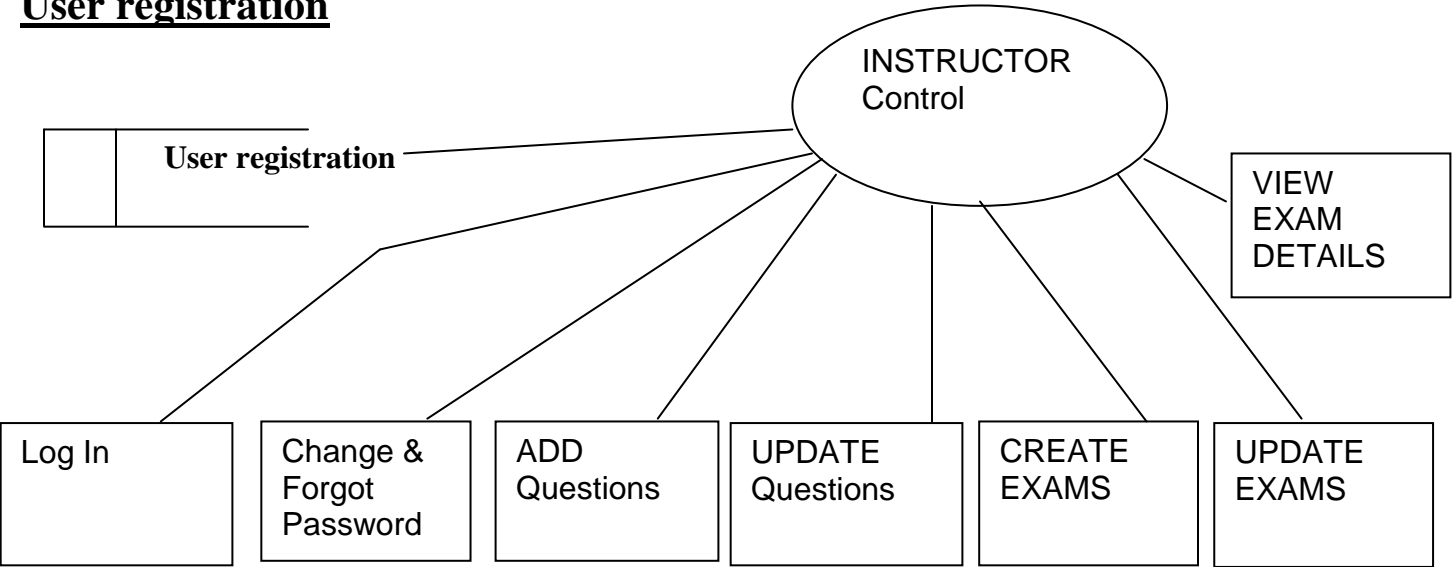
Database:



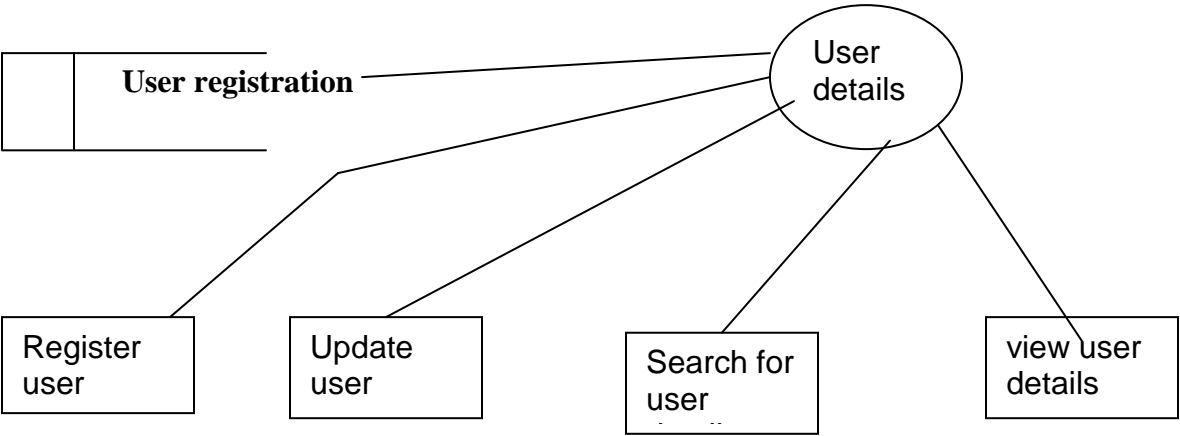
User registration



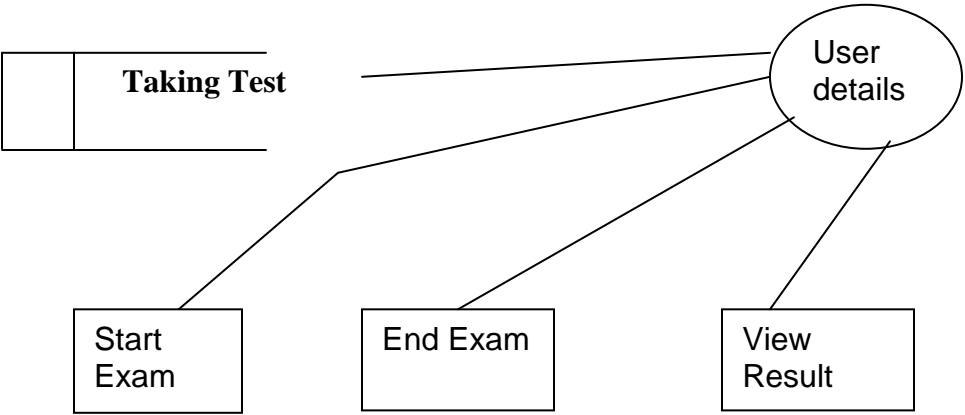
User registration



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) relationship 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 relationship 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 academic 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.

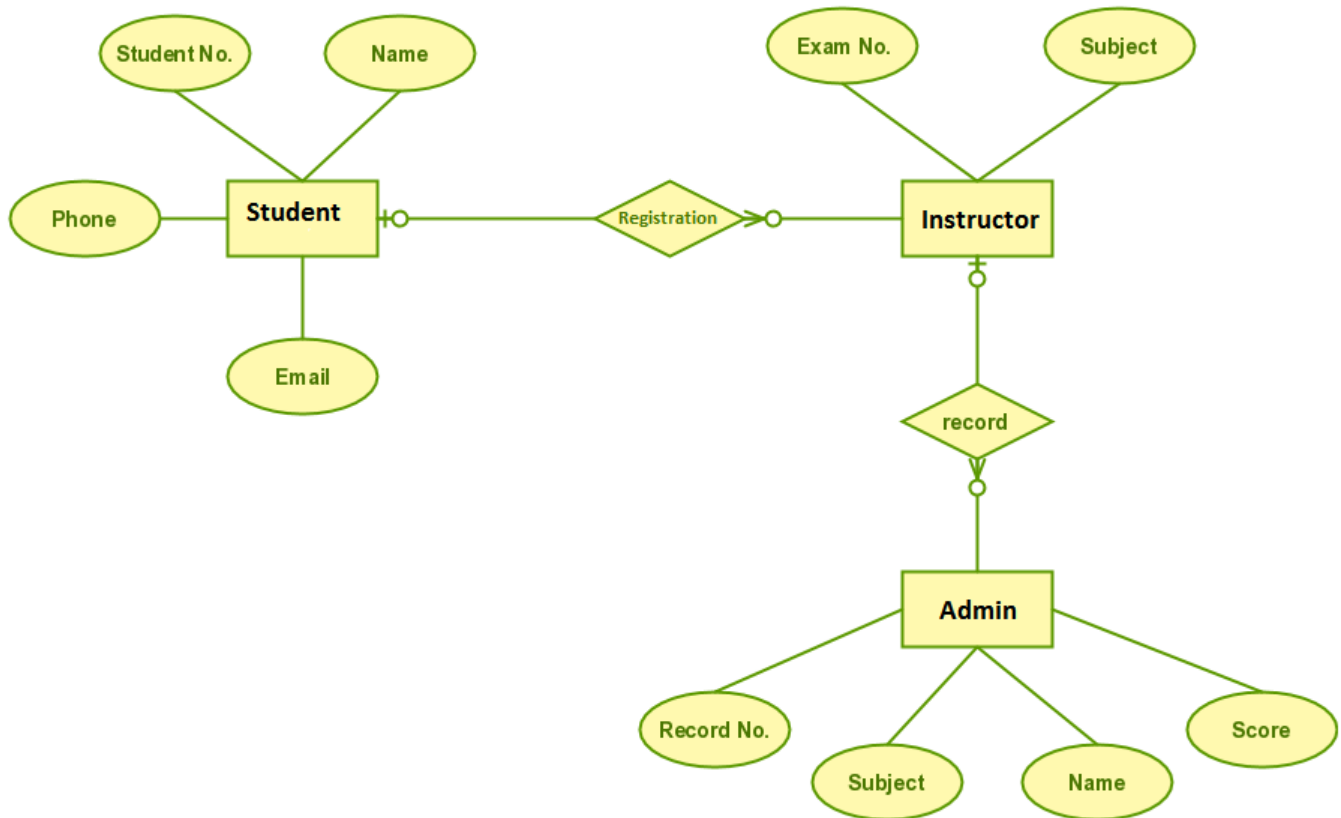
All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

-
- **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

ER- Diagram



MODULES

ONLINE EXAMINATION SYSTEM

MODULES:

- 1.ADMIN MODULE
- 2.INSTRUCTOR MODULE
- 3.STUDENT MODULE

1.ADMIN MODULE:

- 1.REGISTER
- 2.LOG IN
- 3.CHANGE PASSWORD&FORGOTPASSWORD
- 4.STUDENT -MODIFING 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 -MODIFING DETAILS:** User can be modified to change status of each User.

5.**DEPARTMENTS-ENTERING/MODIFYING DETAILS:** New departments adding and old department 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
- 3.CHANGE 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
a. 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 have
registered

9.EVALUATE QUESTION: MULTIPLE CHOICES TUE/FALSE: Evaluation of marks based on his initiations when adding questions.

3. STUDENT Module:

- 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 registered 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.

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 | |

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

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities. The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since, processing speed is very high and the work is reduced in the maintenance point of view management convince that the project is operationally feasible.

2. Economical Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a system and compare them with costs, decisions is made to design and implement the system. This part of feasibility study gives the top management the economic justification for the new system. This is an important input to the management the management, because very often the top management does not like to get confounded by the various technicalities that bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases. In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving lot of time. And paper work and extra staff recruitment and a specific exam hall.

3. Operational Feasibility

In this project is use to conduct exam of the student for providing them a certificate according to the performance of the student in exam. Mainly purpose to give a computerize environment to the student to make it easy and suitable. It help to keep details in safe and secured.

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.

TESTING:

The testing phase is an important part of software development. It is the computerized system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. Software testing is carried out in three steps:

The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether the objectives have been met. Errors are noted down and corrected immediately. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.

The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole. The individual modules are clipped under this major module and tested again and verified the results. This is due to poor interfacing, which may results in data being lost across an interface. A module can have inadvertent, adverse effect on any other or on the global data structures, causing serious problems.

The final step involves validation and testing which determines which the software functions as the user expected. Here also some modifications were. In the completion of the project it is satisfied fully by the end user.

Maintenance and environment:

As the number of computer based systems, griever libraries of computer software began to expand. In house developed projects produced tones of thousand soft program source statements. Software products purchased from the outside added hundreds of thousands of new statements. A dark cloud appeared on the horizon. All of these programs, all of those source statements-had to be corrected when false were detected, modified as user requirements changed, or adapted to new hardware that was purchased. These activities were collectively called software Maintenance.

The maintenance phase focuses on change that is associated with error correction, adaptations required as the software's environment evolves, and changes due to enhancements brought about by changing customer requirements. Four types of changes are encountered during the maintenance phase.

Correction
Adaptation
Enhancement
Prevention

Correction:

Even with the best quality assurance activities is lightly that the customer will uncover defects in the software. Corrective maintenance changes the software to correct defects.

Maintenance is a set of software Engineering activities that occur after software has been delivered to the customer and put into operation. Software configuration management is a set of tracking and control activities that began when a software project begins and terminates only when the software is taken out of the operation.

We may define maintenance by describing four activities that are undertaken after a program is released for use:

Corrective Maintenance

Adaptive Maintenance

Perfective Maintenance or Enhancement

Preventive Maintenance or reengineering

Only about 20 percent of all maintenance work are spent "fixing mistakes". The remaining 80 percent are spent adapting existing systems to changes in their external environment, making enhancements requested by users, and reengineering an application for use.

ADAPTATION:

Over time, the original environment (E>G., CPU, operating system, business rules, external product characteristics) for which the software was developed is likely to change. Adaptive maintenance results in modification to the software to accommodate change to its external environment.

ENHANCEMENT:

As software is used, the customer/user will recognize additional functions that will provide benefit. Perceptive maintenance extends the software beyond its original function requirements.

PREVENTION:

Computer software deteriorates due to change, and because of this, preventive maintenance, often called software re engineering, must be conducted to enable the software to serve the needs of its end users. In essence, preventive maintenance makes changes to computer programs so that they can be more easily corrected, adapted, and enhanced. Software configuration management (SCM) is an umbrella activity that is applied throughout the software process. SCM activities are developed to

SOFTWARE METHODOLOGY

The software methodology followed in this project includes the object-oriented methodology and the application system development methodologies. The description of these methodologies is given below.

Application System Development – A Life cycle Approach

Although there are a growing number of applications (such as decision support systems) that should be developed using an experimental process strategy such as prototyping, a significant amount of new development work continue to involve major operational applications of broad scope. The application systems are large highly structured. User task comprehension and developer task proficiency is usually high. These factors suggest a linear or iterative assurance strategy. The most common method for this stage class of problems is a system development life cycle modal in which each stage of development is well defined and has straightforward requirements for deliverables, feedback and sign off. The system development life cycle is described in detail since it continues to be an appropriate methodology for a significant part of new development work.

The basic idea of the system development life cycle is that there is a well-defined process by which an application is conceived and developed and implemented. The life cycle gives structure to a creative process. In order to manage and control the development effort, it is necessary to know what should have been done, what has been done, and what has yet to be accomplished. The phrases in the system development life

cycle provide a basis for management and control because they define segments of the flow of work, which can be identified for managerial purposes and specifies the documents or other deliverables to be produced in each phase.

The phases in the life cycle for information system development are described differently by different writers, but the differences are primarily in the amount of necessity and manner of categorization. There is a general agreement on the flow of development steps and the necessity for control procedures at each stage.

The information system development cycle for an application consists of three major stages.

- 1) Definition.
- 2) Development.
- 3) Installation and operation.

The first stage of the process, which defines the information requirements for a feasible cost effective system. The requirements are then translated into a physical system of forms, procedures, programs etc., by the system design, computer programming and procedure development. The resulting system is test and put into operation. No system is perfect so there is always a need for maintenance changes. To complete the cycle, there should be a post audit of the system to evaluate how well it performs and how well it meets the cost and performance specifications. The stages of definition, development and installation and operation can therefore be divided into smaller steps or phrases as follows.

Definition

Proposed definition : preparation of request for proposed applications.

Feasibility assessment : evaluation of feasibility and cost benefit of proposed system.

Information requirement analysis : determination of information needed.

Design

Conceptual design : User-oriented design of application development.

Physical system design : Detailed design of flows and processes in applications processing system and preparation of program specification.

Development

Program development : coding and testing of computer programs.

Procedure development : design of procedures and preparation of user instructions.

Installation and operation

Conversion : final system test and conversion.

Operation and maintenance : Month to month operation and maintenance

Post audit : Evaluation of development process, application system and results of use at the completion of the each phase, formal approval sign-off is required from the users as well as from the manager of the project development.

Testing

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

Testing Principles

1. All tests should be traceable to end user requirements
2. Tests should be planned long before testing begins
3. Testing should begin on a small scale and progress towards testing in large
4. Exhaustive testing is not possible
5. To be most effective testing should be conducted by a independent third party

The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

White box testing.

Black box testing.

White-box testing:

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides thorough test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

Testing strategies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that all small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

Testing fundamentals:

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

Testing Information flow:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

Unit testing:

Unit testing is essential for the verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed design description as a guide, important paths are tested to uncover errors within the boundary of the modules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

Integration testing :

Integration testing focuses on unit tested modules and build the program structure that is dictated by the design phase.

System testing:

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and its original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire system is working properly or not will be tested here, and specified path ODBC

connection will correct or not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

Acceptance Testing:

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

Tools to special importance during acceptance testing include:

Test coverage Analyzer – records the control paths followed for each test case.

Timing Analyzer – also called a profiler, reports the time spent in various regions of the code are areas to concentrate on to improve system performance.

Coding standards – static analyzers and standard checkers are used to inspect code for deviations from standards and guidelines.

Test Cases:

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structure to assure their validity.

The test case specification for system testing has to be submitted for review before system testing commences.

Conclusion

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

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 is 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 online exam.

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

Bibliography

BIBLIOGRAPHY

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

Books Referred:

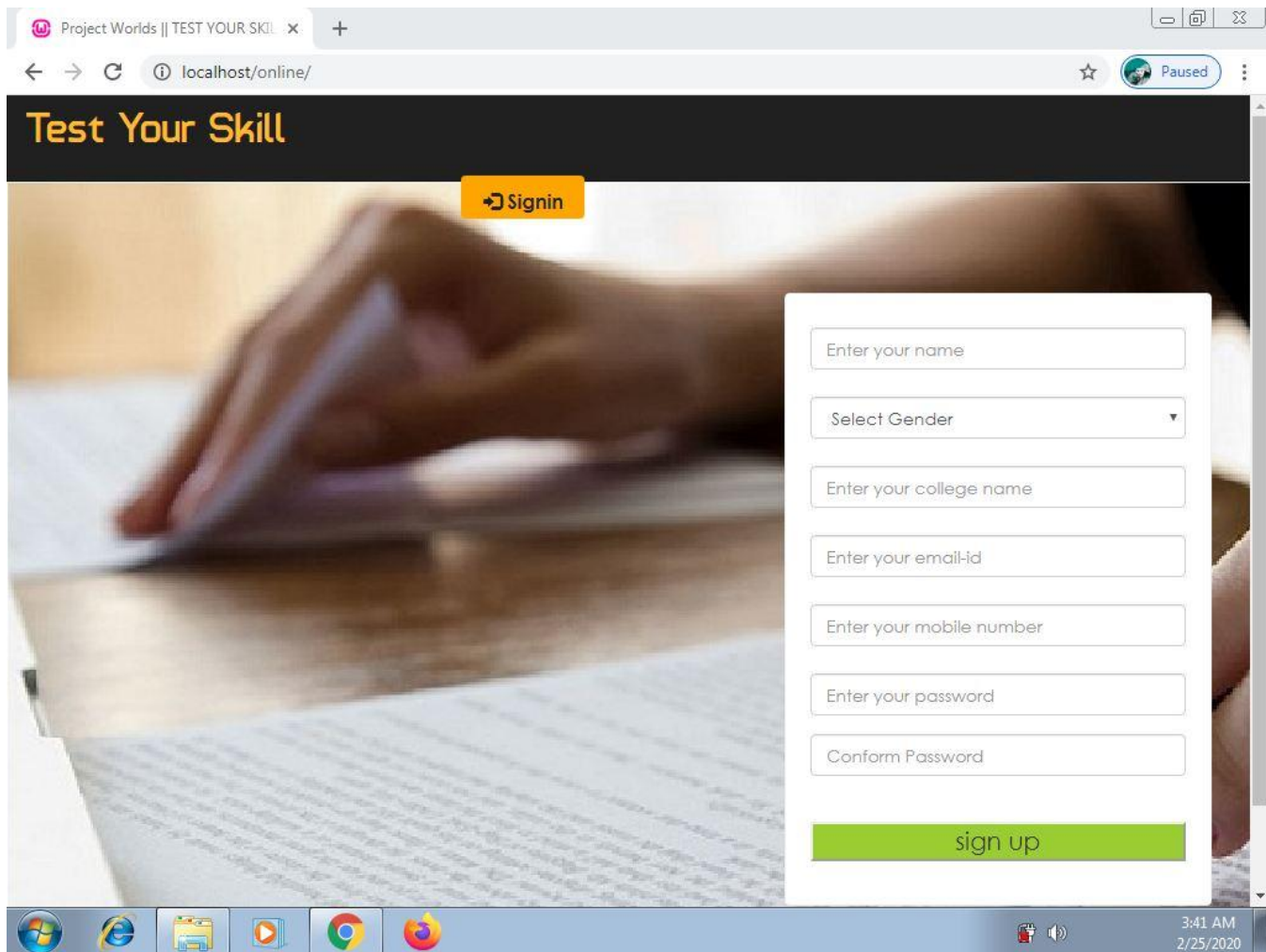
- | | | |
|------|-------------------------------|---------------------------|
| I. | BEGINNING PHP 5 | --- DAVE MERCER |
| II. | BLACK BOOK HTML | --- WILEY DREAMTECH |
| III. | PHP AND MYSQL WEB DEVELOPMENT | --- LUKEWELLING, LAURA |
| IV. | MICROSOFT SQL SERVER-2000 | --- RANKIN, PAUL & JENSEN |
| V. | SQL SERVER-2000 | --- DUSAN PETKOVIC |
| VI. | PHP IN A NUTSHELL | --- PAUL HUDSON |

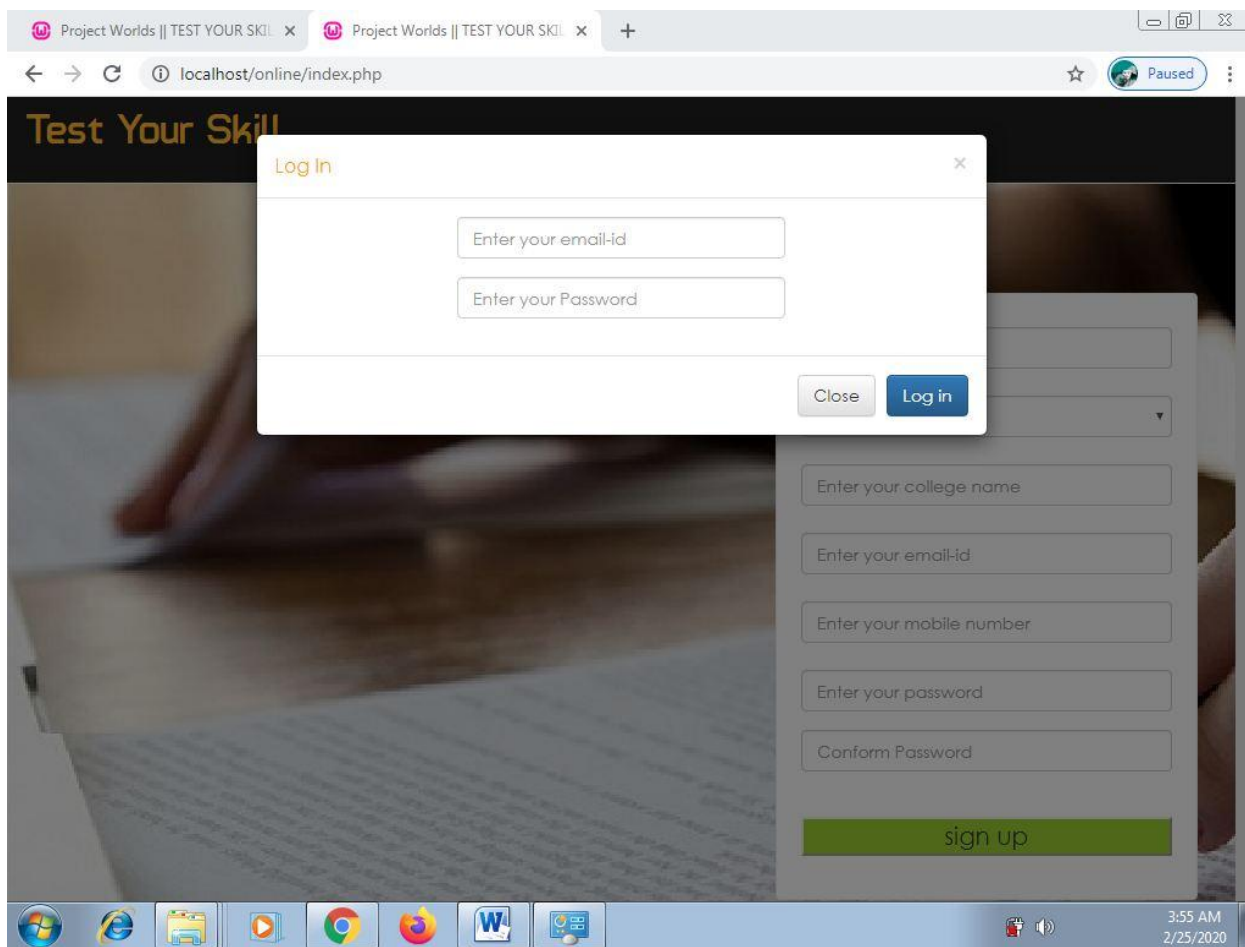
✓ Websites Referred:

<http://www.google.com>
<http://www.wikipidea.com>



Output screens





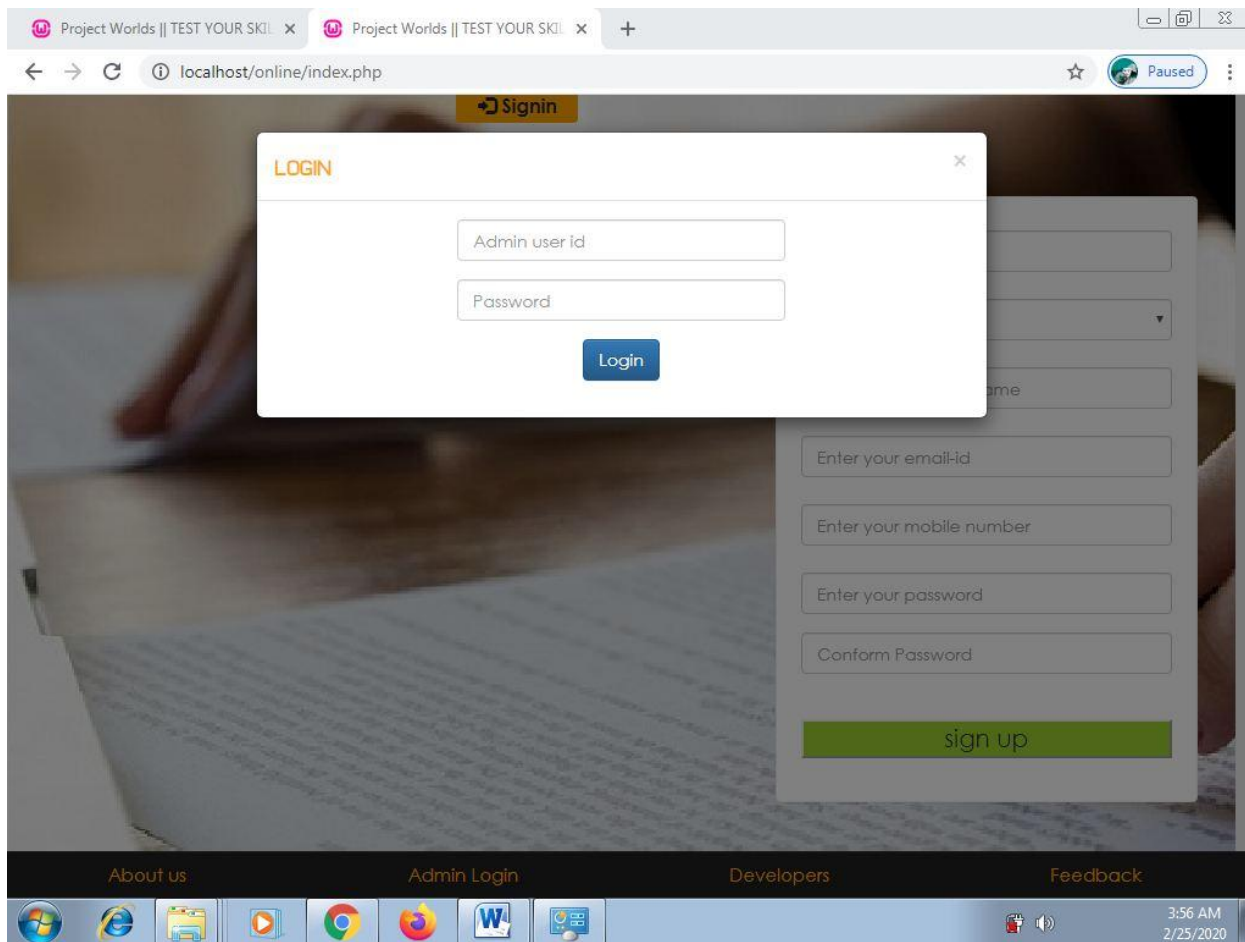
Question 2 ::
which is a variable of php??










- ☐ int a
- ☒ \$a
- ☐ long int a
- ☐ int a\$

Submit

Result

| | |
|-----------------|---|
| Total Questions | 2 |
| right Answer ✓ | 2 |
| Wrong Answer ✗ | 0 |
| Score ★ | 4 |



| S.N. | Name | Gender | College | Email | Mobile | |
|------|----------------|--------|--------------------------|-----------------------|------------|---|
| 1 | Avantika | F | KNIT sultanpur | avantika420@gmail.com | 7785068889 |  |
| 2 | Mark Zukarburg | M | Stanford | ceo@facebook.com | 987654321 |  |
| 3 | Komal | F | KNIT sultanpur | komalpd2011@gmail.com | 7785068889 |  |
| 4 | Tom Cruze | M | Hollywood | mi5@hollywood.com | 7785068889 |  |
| 5 | Netcamp | M | KNIT sultanpur | netcamp@gmail.com | 987654321 |  |
| 6 | Nikunj | M | XYZ | nik1@gmail.com | 987 |  |
| 7 | Sunny | M | KNIT sultanpur | sunnygkp10@gmail.com | 7785068889 |  |
| 8 | User | M | cim1 | user@user.com | 11 |  |
| 9 | Vikash | M | KNIT sultanpur@gmail.com | vikash@gmail.com | 7785068889 |  |

Enter Quiz Details

Enter Quiz title

Enter total number of questions

Enter marks on right answer

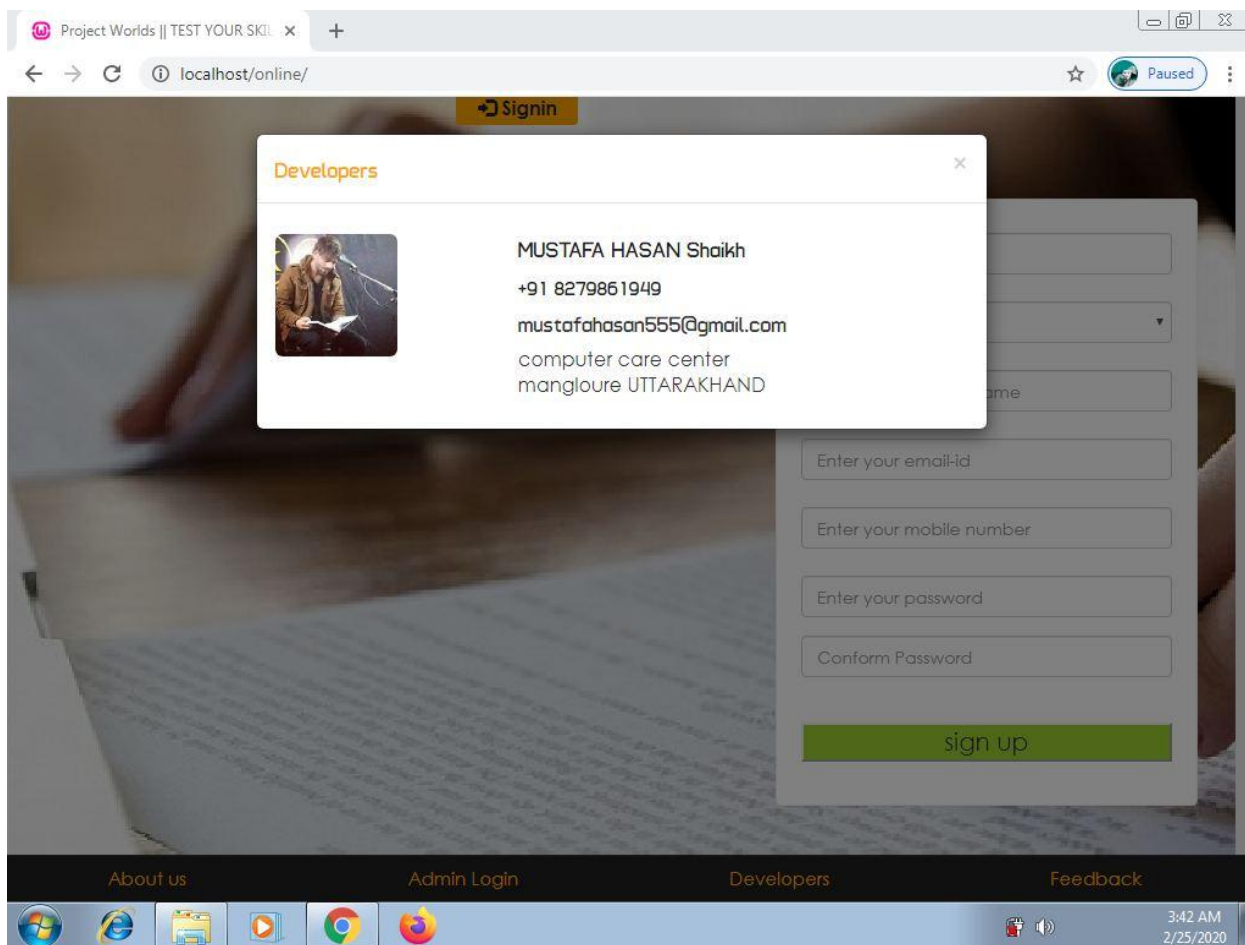
Enter minus marks on wrong answer without sign

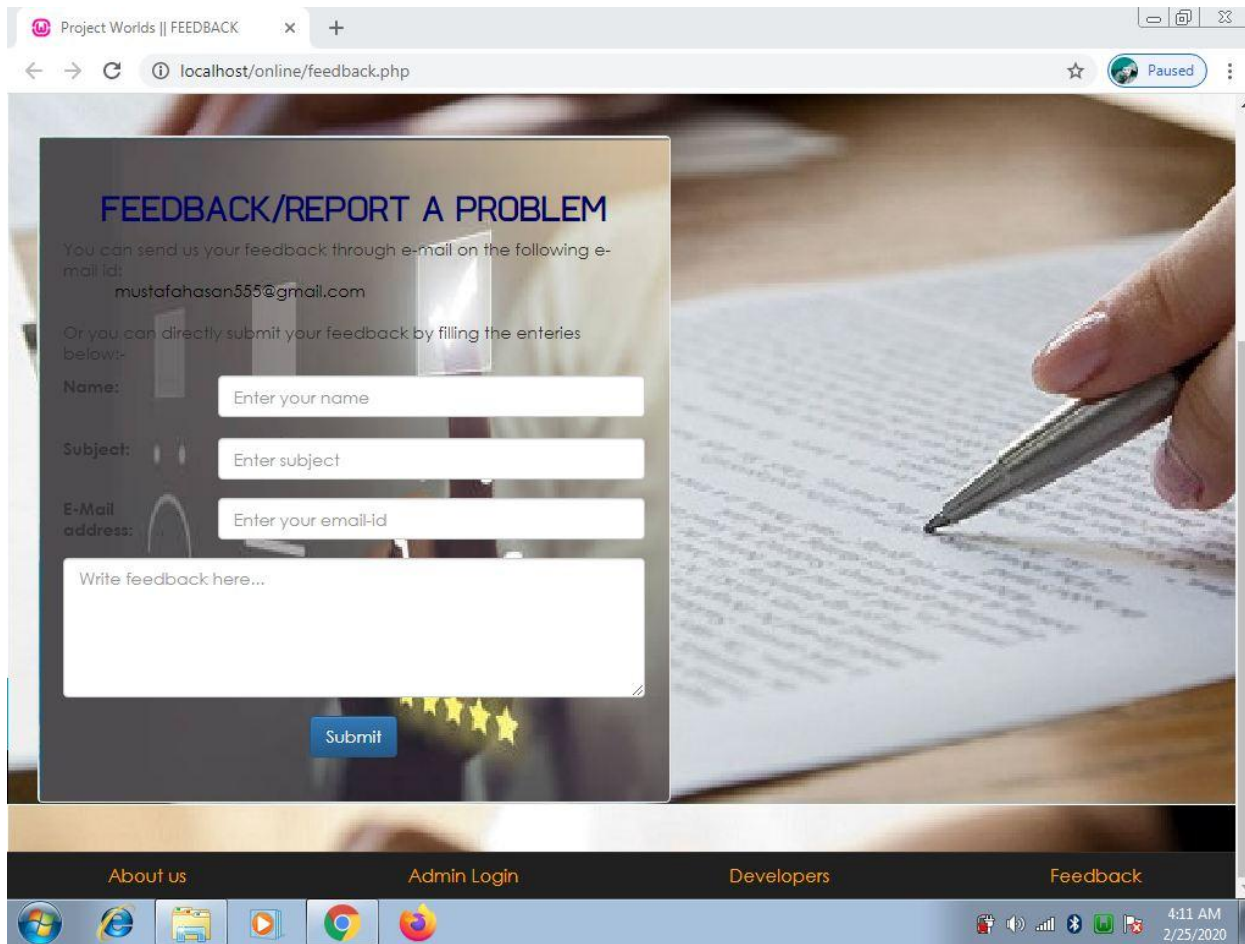
Enter time limit for test in minute

Enter #tag which is used for searching

Write description here...

| S.N. | Topic | Total question | Marks | Time limit | |
|------|------------------------|----------------|-------|------------|------------------------|
| 1 | Linux :vi Editor | 5 | 10 | 10 min | Remove |
| 2 | Linux:startup | 5 | 10 | 10 min | Remove |
| 3 | Networking | 2 | 4 | 5 min | Remove |
| 4 | C++ Coding | 2 | 4 | 5 min | Remove |
| 5 | Php Coding | 2 | 4 | 5 min | Remove |
| 6 | Linux : File Managment | 2 | 4 | 5 min | Remove |





PHP Code

Account.php

```
<!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>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';
```

[illegible]

```

$result = mysqli_query($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" aria-
hidden="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>'. $time.'&nbsp;min</td>
<td><b><a
href="update.php?q=quize&step=25&eid='.$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;'.$sn.'&nbsp;::<br />'.$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="form-horizontal">
<br />';

while($row=mysqli_fetch_array($q) )
{
$option=$row['option'];
$optionid=$row['optionid'];
echo '<input type="radio" name="ans" value="'.$optionid.'">'.$option.'<br /><br />';
}
echo '<br /><button type="submit" class="btn btn-primary"><span class="glyphicon glyphicon-lock" aria-
hidden="true"></span>&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>'.$qa.'</td></tr>
<tr style="color:#99cc32"><td>right Answer&nbsp;<span class="glyphicon glyphicon-ok-circle" aria-
hidden="true"></span></td><td>'.$r.'</td></tr>';

```

```

                                <tr style="color:red"><td>Wrong Answer&nbsp;<span
class="glyphicon glyphicon-remove-circle" aria-hidden="true"></span></td><td>'.Sw.'</td></tr>
                                <tr style="color:#66CCFF"><td>Score&nbsp;<span
class="glyphicon glyphicon-star" aria-hidden="true"></span></td><td>'.Ss.'</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" aria-
hidden="true"></span></td><td>'.Ss.'</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>'.Sc.'</td><td>'.Ss.'</td><td>'.Sqa.'</td><td>'.Sr.'</td><td>'.Sw.'</td><td>'.Ss.'</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">Mustafa hasan</a>
        <h4 style="color:#202020; font-family:'typo' ;font-size:16px"
class="title1">+91 8279861949</h4>
        <h4 style="font-family:'typo'
">Mustafahasan555@gmail.com</h4>
        <h4 style="font-family:'typo' ">Manglore
        </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>

Admin.php

```
<?php
include_once 'dbConnection.php';
$ref=@$_GET['q'];
$email = $_POST['uname'];
$password = $_POST['password'];

$email = stripslashes($email);
$email = addslashes($email);
$password = stripslashes($password);
$password = addslashes($password);
$result = mysqli_query($con,"SELECT email FROM admin WHERE email = '$email' and password = '$password'") or
die('Error');
$count=mysqli_num_rows($result);
if($count==1){
session_start();
if(isset($_SESSION['email'])){
session_unset();}
$_SESSION["name"] = 'Admin';
$_SESSION["key"] ='sunny7785068889';
$_SESSION["email"] = $email;
header("location:dash.php?q=0");
}
else header("location:$ref?w=Warning : Access denied");
?>
```

Dash.php

```
<!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>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
{
$na

```

```
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;&Hello,</span> <a href="account.php" class="log  
log1">'. $name.</a>&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;&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>
        <span class="icon-bar"></span>
```

[illegible]

```

                                <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" aria-
hidden="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>
                                <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>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>';}

?>

<!--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" aria-
hidden="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><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" aria-
hidden="true"></span></b></a></td>';
                                                echo '<td><a title="Delete Feedback"
href="update.php?fdid='.$id.'"><b><span class="glyphicon glyphicon-trash" aria-hidden="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" aria-hidden="true"></span></b></a><h2 style="text-align:center; margin-top:-15px;font-family: "Ubuntu", sans-
serif;"><b>'.$subject.'</b></h1>';
echo '<div class="mCustomScrollbar" data-mcs-theme="dark" style="margin-left:10px;margin-right:10px; max-
height:450px; line-height:35px;padding:5px;"><span style="line-height:35px;padding:5px;">-
&nbsp;<b>DATE:</b>&nbsp;<span>'.$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=" '.$i.'.1"></label>
  <div class="col-md-12">
    <input id=" '.$i.'.1" name=" '.$i.'.1" 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=" '.$i.'.2"></label>

```

```

<div class="col-md-12">
<input id="'.$.2" name="'.$.2" 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="'.$.3"></label>
<div class="col-md-12">
<input id="'.$.3" name="'.$.3" 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="'.$.4"></label>
<div class="col-md-12">
<input id="'.$.4" name="'.$.4" placeholder="Enter option d" class="form-control input-md" type="text">

</div>
</div>
<br />
<b>Correct answer</b><br />
<select id="ans'$.'" name="ans'$.'" placeholder="Choose correct answer " class="form-control input-md" >
<option value="a">Select answer for question '$.'.</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.'</td><td>'.$sahi*$total.'</td><td>'.$time.'&nbsp;min</td>
    <td><b><a href="update.php?q=rmquiz&eid='.$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>

```

Dbconnection.php

```

<?php
//all the variables defined here are accessible in all the files that include this one
$con= new mysqli('localhost','root','','project')or die("Could not connect to mysql".mysqli_error($con));

?>

```

Feed.php

```

<?php
include_once 'dbConnection.php';
$ref=@$_GET['q'];
$name = $_POST['name'];
$email = $_POST['email'];
$subject = $_POST['subject'];
$id=uniqid();
$date=date("Y-m-d");
$time=date("h:i:sa");
$feedback = $_POST['feedback'];
$q=mysqli_query($con,"INSERT INTO feedback VALUES ('$id', '$name', '$email', '$subject', '$feedback', '$date',
'$time')")or die ("Error");
header("location:$ref?q=Thank you for your valuable feedback");

?>

```

Feedback.php

```
<!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" />
<title>Project Worlds || FEEDBACK </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>

<body>

<!--header start-->
<div class="row header">
<div class="col-lg-6">
<span class="logo">Test Your Skill</span></div>
<div class="col-md-2">
</div>
<div class="col-md-4">
<?php
include_once 'dbConnection.php';
session_start();
if(!isset($_SESSION['email']))){
echo '<a href="#" class="pull-right sub1 btn title3" data-toggle="modal" data-target="#myModal"><span class="glyphicon
glyphicon-log-in" aria-hidden="true"></span>&nbsp;Signin</a>&nbsp;';}
else
{
echo '<a href="logout.php?q=feedback.php" class="pull-right sub1 btn title3"><span class="glyphicon glyphicon-log-out"
aria-hidden="true"></span>&nbsp;Signout</a>&nbsp;';}
?>

<a href="index.php" class="pull-right btn sub1 title3"><span class="glyphicon glyphicon-home" aria-
hidden="true"></span>&nbsp;Home</a>&nbsp;
</div></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-->

<!--header end-->

<div class="bg1">
<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6 panel" style="background-image:url(image/bg1.jpg); min-height:430px;">
<h2 align="center" style="font-family:typo'; color:#000066">FEEDBACK/REPORT A PROBLEM</h2>
<div style="font-size:14px">
<?php if(@$_GET['q'])echo '<span style="font-size:18px;"><span class="glyphicon glyphicon-ok" aria-
hidden="true"></span>&nbsp;<span>'.@$_GET['q'].'</span>';

```

```

else
{echo'
You can send us your feedback through e-mail on the following e-mail id:<br />
<div class="row">
<div class="col-md-1"></div>
<div class="col-md-10">
<a href="mailto:chiraggoel.53784@gmail.com" style="color:#000000">mustafahasan555@gmail.com</a><br /><br />
</div><div class="col-md-1"></div></div>
<p>Or you can directly submit your feedback by filling the enteries below:-</p>
<form role="form" method="post" action="feed.php?q=feedback.php">
<div class="row">
<div class="col-md-3"><b>Name:</b><br /><br /><br /><b>Subject:</b></div>
<div class="col-md-9">
<!-- Text input-->
<div class="form-group">
<input id="name" name="name" placeholder="Enter your name" class="form-control input-md" type="text"><br />
<input id="name" name="subject" placeholder="Enter subject" class="form-control input-md" type="text">
</div>
</div>
</div>
</div><!--End of row-->

<div class="row">
<div class="col-md-3"><b>E-Mail address:</b></div>
<div class="col-md-9">
<!-- Text input-->
<div class="form-group">
<input id="email" name="email" placeholder="Enter your email-id" class="form-control input-md" type="email">
</div>
</div>
</div><!--End of row-->

<div class="form-group">
<textarea rows="5" cols="8" name="feedback" class="form-control" placeholder="Write feedback here..."></textarea>
</div>
<div class="form-group" align="center">
<input type="submit" name="submit" value="Submit" class="btn btn-primary" />
</div>
</form>';}>
</div><!--col-md-6 end-->
<div class="col-md-3"></div></div>
</div></div>
</div><!--container end-->

<!--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 class="col-md-5">
  <a href="http://yugeshverma.blogspot.in"
style="color:#202020; font-family:'typo' ; font-size:18px" title="Find on Facebook">Mustafa Hasan Shaikh</a>
  <h4 style="color:#202020; font-family:'typo' ;font-size:16px"
class="title1">+91 8279861949 </h4>

">mustafahasan555@gmail.com</h4>

Mangloure UTTARAKHAND</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>

```

Index.php

```

<!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>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 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>

alt="Sunny Prakash Tiwari" class="img-rounded">

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

<div class="col-md-5">
<a href="http://yugeshverma.blogspot.in"
style="color:#202020; font-family:'typo' ; font-size:18px" title="Find on Facebook">MUSTAFA HASAN Shaikh</a>
<h4 style="color:#202020; font-family:'typo' ;font-size:16px"
class="title1">+91 8279861949</h4>

">mustafahasan555@gmail.com</h4>

mangloure UTTARAKHAND</div></div>

</div>

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

<!--Modal for admin 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>

```

Login.php

```

<?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");

?>

```

Loguot.php

```

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

```

Sign.php

```

<?php
include_once 'dbConnection.php';
ob_start();
$name = $_POST['name'];
$name= ucwords(strtolower($name));
$gender = $_POST['gender'];
$email = $_POST['email'];
$college = $_POST['college'];
$mob = $_POST['mob'];
$password = $_POST['password'];
$name = stripslashes($name);
$name = addslashes($name);
$name = ucwords(strtolower($name));
$gender = stripslashes($gender);
$gender = addslashes($gender);
$email = stripslashes($email);
$email = addslashes($email);
$college = stripslashes($college);
$college = addslashes($college);
$mob = stripslashes($mob);
$mob = addslashes($mob);

$password = stripslashes($password);
$password = addslashes($password);
$password = md5($password);

```

```

$q3=mysqli_query($con,"INSERT INTO user VALUES ('$name' , '$gender' , '$college','$email' , '$mob' , '$password')");
if($q3)
{
    session_start();
    $_SESSION["email"] = $email;
    $_SESSION["name"] = $name;

    header("location:account.php?q=1");
}
else
{
    header("location:index.php?q7=Email Already Registered!!!");
}
ob_end_flush();
?>

```

Update.php

```

<?php
include_once 'dbConnection.php';
session_start();
$email=$_SESSION['email'];
//delete feedback
if(isset($_SESSION['key'])){
if(@$_GET['fdid'] && $_SESSION['key']=='sunny7785068889') {
$fid=@$_GET['fdid'];
$result = mysqli_query($con,"DELETE FROM feedback WHERE id='$fid' ") or die('Error');
header("location:dash.php?q=3");
}
}

//delete user
if(isset($_SESSION['key'])){
if(@$_GET['demail'] && $_SESSION['key']=='sunny7785068889') {
$demail=@$_GET['demail'];
$r1 = mysqli_query($con,"DELETE FROM rank WHERE email='$demail' ") or die('Error');
$r2 = mysqli_query($con,"DELETE FROM history WHERE email='$demail' ") or die('Error');
$result = mysqli_query($con,"DELETE FROM user WHERE email='$demail' ") or die('Error');
header("location:dash.php?q=1");
}
}

//remove quiz
if(isset($_SESSION['key'])){
if(@$_GET['q']== 'rmquiz' && $_SESSION['key']=='sunny7785068889') {
$eid=@$_GET['eid'];
$result = mysqli_query($con,"SELECT * FROM questions WHERE eid='$eid' ") or die('Error');
while($row = mysqli_fetch_array($result)) {
                                $qid = $row['qid'];
$r1 = mysqli_query($con,"DELETE FROM options WHERE qid='$qid'") or die('Error');
$r2 = mysqli_query($con,"DELETE FROM answer WHERE qid='$qid' ") or die('Error');
}
$r3 = mysqli_query($con,"DELETE FROM questions WHERE eid='$eid' ") or die('Error');
}
}

```



```

$r4 = mysqli_query($con,"DELETE FROM quiz WHERE eid='$eid' ") or die('Error');
$r4 = mysqli_query($con,"DELETE FROM history WHERE eid='$eid' ") or die('Error');

header("location:dash.php?q=5");
}
}

//add quiz
if(isset($_SESSION['key'])){
if(@$_GET['q']== 'addquiz' && $_SESSION['key']=='sunny7785068889') {
$name = $_POST['name'];
$name= ucwords(strtolower($name));
$total = $_POST['total'];
$sahi = $_POST['right'];
$wrong = $_POST['wrong'];
$time = $_POST['time'];
>tag = $_POST['tag'];
$desc = $_POST['desc'];
$id=uniqid();
$q3=mysqli_query($con,"INSERT INTO quiz VALUES ('$id','$name' , '$sahi' , '$wrong','$total','$time' , '$desc','$tag',
NOW())");

header("location:dash.php?q=4&step=2&eid=$id&n=$total");
}
}

//add question
if(isset($_SESSION['key'])){
if(@$_GET['q']== 'addqns' && $_SESSION['key']=='sunny7785068889') {
$n=@$_GET['n'];
$eid=@$_GET['eid'];
$sch=@$_GET['ch'];

for($i=1;$i<=$n;$i++)
{
$qid=uniqid();
$qns=$_POST['qns'.$i];
$q3=mysqli_query($con,"INSERT INTO questions VALUES ('$eid','$qid','$qns' , '$sch' , '$i')");
$soaid=uniqid();
$sobid=uniqid();
$socid=uniqid();
$sodid=uniqid();
$a=$_POST[$i.'1'];
$b=$_POST[$i.'2'];
$c=$_POST[$i.'3'];
$d=$_POST[$i.'4'];
$qa=mysqli_query($con,"INSERT INTO options VALUES ('$qid','$a','$soaid')") or die('Error61');
$qb=mysqli_query($con,"INSERT INTO options VALUES ('$qid','$b','$sobid')") or die('Error62');
$qc=mysqli_query($con,"INSERT INTO options VALUES ('$qid','$c','$socid')") or die('Error63');
$qd=mysqli_query($con,"INSERT INTO options VALUES ('$qid','$d','$sodid')") or die('Error64');
$e=$_POST['ans'.$i];
switch($e)
{

```

```

case 'a':
    $ansid=$oaid;
    break;
case 'b':
    $ansid=$obid;
    break;
case 'c':
    $ansid=$ocid;
    break;
case 'd':
    $ansid=$odid;
    break;
default:
    $ansid=$oaid;
}

```

```

$qans=mysqli_query($con,"INSERT INTO answer VALUES ('$qid','$ansid')");

```

```

}
header("location:dash.php?q=0");
}
}

```

```

//quiz start

```

```

if(@$_GET['q']== 'quiz' && @$_GET['step']== 2) {
    $eid=@$_GET['eid'];
    $sn=@$_GET['n'];
    $total=@$_GET['t'];
    $ans=$_POST['ans'];
    $qid=@$_GET['qid'];
    $q=mysqli_query($con,"SELECT * FROM answer WHERE qid='$qid' ");
    while($row=mysqli_fetch_array($q) )
    {
        $ansid=$row['ansid'];
    }
    if($ans == $ansid)
    {
        $q=mysqli_query($con,"SELECT * FROM quiz WHERE eid='$eid' ");
        while($row=mysqli_fetch_array($q) )
        {
            $sahi=$row['sahi'];
        }
        if($sn == 1)
        {
            $q=mysqli_query($con,"INSERT INTO history VALUES('$email','$eid' ,0,0,0,0,NOW())")or die('Error');
        }
        $q=mysqli_query($con,"SELECT * FROM history WHERE eid='$eid' AND email='$email' ")or die('Error115');

        while($row=mysqli_fetch_array($q) )
        {
            $s=$row['score'];
            $r=$row['sahi'];

```

```

}
$r++;
$s=$s+$sahi;
$q=mysqli_query($con,"UPDATE `history` SET `score`=$s,`level`=$sn,`sahi`=$r, date= NOW() WHERE email = '$email'
AND eid = '$eid'" )or die('Error124');

}
else
{
$q=mysqli_query($con,"SELECT * FROM quiz WHERE eid='$eid' " )or die('Error129');

while($row=mysqli_fetch_array($q) )
{
$wrong=$row['wrong'];
}
if($sn == 1)
{
$q=mysqli_query($con,"INSERT INTO history VALUES('$email','$eid','0','0','0','0',NOW() )" )or die('Error137');
}
$q=mysqli_query($con,"SELECT * FROM history WHERE eid='$eid' AND email='$email' " )or die('Error139');
while($row=mysqli_fetch_array($q) )
{
$s=$row['score'];
$w=$row['wrong'];
}
$w++;
$s=$s-$wrong;
$q=mysqli_query($con,"UPDATE `history` SET `score`=$s,`level`=$sn,`wrong`=$w, date=NOW() WHERE email =
'$email' AND eid = '$eid'" )or die('Error147');
}
if($sn != $total)
{
$sn++;
header("location:account.php?q=quiz&step=2&eid=$eid&n=$sn&t=$total")or die('Error152');
}
else if( $_SESSION['key']!= 'sunny7785068889')
{
$q=mysqli_query($con,"SELECT score FROM history WHERE eid='$eid' AND email='$email'" )or die('Error156');
while($row=mysqli_fetch_array($q) )
{
$s=$row['score'];
}
$q=mysqli_query($con,"SELECT * FROM rank WHERE email='$email'" )or die('Error161');
$rowcount=mysqli_num_rows($q);
if($rowcount == 0)
{
$q2=mysqli_query($con,"INSERT INTO rank VALUES('$email','$s',NOW() )" )or die('Error165');
}
else
{
while($row=mysqli_fetch_array($q) )
{
$sun=$row['score'];

```

```

}
$sun=$s+$sun;
$q=mysqli_query($con,"UPDATE `rank` SET `score`=$sun ,time=NOW() WHERE email= '$email'" )or die('Error174');
}
header("location:account.php?q=result&eid=$eid");
}
else
{
header("location:account.php?q=result&eid=$eid");
}
}

//restart quiz
if( @$_GET['q']== 'quize' && @$_GET['step']== 25 ) {
$eid=@$_GET['eid'];
$n=@$_GET['n'];
$t=@$_GET['t'];
$q=mysqli_query($con,"SELECT score FROM history WHERE eid='$eid' AND email='$email'" )or die('Error156');
while($row=mysqli_fetch_array($q) )
{
$s=$row['score'];
}
$q=mysqli_query($con,"DELETE FROM `history` WHERE eid='$eid' AND email='$email' " )or die('Error184');
$q=mysqli_query($con,"SELECT * FROM rank WHERE email='$email'" )or die('Error161');
while($row=mysqli_fetch_array($q) )
{
$sun=$row['score'];
}
$sun=$sun-$s;
$q=mysqli_query($con,"UPDATE `rank` SET `score`=$sun ,time=NOW() WHERE email= '$email'" )or die('Error174');
header("location:account.php?q=quiz&step=2&eid=$eid&n=1&t=$t");
}

?>

```

SQL Coding

Project.sql

```

-- phpMyAdmin SQL Dump
-- version 4.6.4
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Nov 06, 2017 at 11:50 AM
-- Server version: 5.7.14
-- PHP Version: 5.6.25

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

```

```
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
```

```
--
-- Database: `project`
--
```

```
-- -----
```

```
--
-- Table structure for table `admin`
--
```

```
CREATE TABLE `admin` (
  `admin_id` int(11) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(500) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Dumping data for table `admin`
--
```

```
INSERT INTO `admin` (`admin_id`, `email`, `password`) VALUES
(1, 'sunnygkp10@gmail.com', '123456'),
(2, 'admin@admin.com', 'admin');
```

```
-- -----
```

```
--
-- Table structure for table `answer`
--
```

```
CREATE TABLE `answer` (
  `qid` text NOT NULL,
  `ansid` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Dumping data for table `answer`
--
```

```
INSERT INTO `answer` (`qid`, `ansid`) VALUES
('55892169bf6a7', '55892169d2efc'),
('5589216a3646e', '5589216a48722'),
('558922117fcef', '5589221195248'),
('55892211e44d5', '55892211f1fa7'),
('558922894c453', '558922895ea0a'),
('558922899ccaa', '55892289aa7cf'),
('558923538f48d', '558923539a46c');
```

```
('55892353f05c4', '55892354051be'),
('558973f4389ac', '558973f462e61'),
('558973f4c46f2', '558973f4d4abe'),
('558973f51600d', '558973f526fc5'),
('558973f55d269', '558973f57af07'),
('558973f5abb1a', '558973f5e764a'),
('5589751a63091', '5589751a81bf4'),
('5589751ad32b8', '5589751adbdbd'),
('5589751b304ef', '5589751b3b04d'),
('5589751b749c9', '5589751b9a98c');
```

```
--
-- Table structure for table `feedback`
--
```

```
CREATE TABLE `feedback` (
  `id` text NOT NULL,
  `name` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `subject` varchar(500) NOT NULL,
  `feedback` varchar(500) NOT NULL,
  `date` date NOT NULL,
  `time` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Dumping data for table `feedback`
--
```

```
INSERT INTO `feedback` (`id`, `name`, `email`, `subject`, `feedback`, `date`, `time`) VALUES
('55846be776610', 'testing', 'sunnygkp10@gmail.com', 'testing', 'testing stART', '2015-06-19', '09:22:15pm'),
('5584ddd0da0ab', 'netcamp', 'sunnygkp10@gmail.com', 'feedback', 'mLBLB', '2015-06-20', '05:28:16am'),
('558510a8a1234', 'sunnygkp10', 'sunnygkp10@gmail.com', 'dl;dsnlfn', 'fmdsfld fdj', '2015-06-20', '09:05:12am'),
('5585509097ae2', 'sunny', 'sunnygkp10@gmail.com', 'kcsnck', 'l.mdsavn', '2015-06-20', '01:37:52pm'),
('5586ee27af2c9', 'vikas', 'vikas@gmail.com', 'trial feedback', 'trial feedback', '2015-06-21', '07:02:31pm'),
('5589858b6c43b', 'nik', 'nik1@gmail.com', 'good', 'good site', '2015-06-23', '06:12:59pm');
```

```
--
-- Table structure for table `history`
--
```

```
CREATE TABLE `history` (
  `email` varchar(50) NOT NULL,
  `eid` text NOT NULL,
  `score` int(11) NOT NULL,
  `level` int(11) NOT NULL,
  `sahi` int(11) NOT NULL,
  `wrong` int(11) NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `history`
--

INSERT INTO `history` (`email`, `eid`, `score`, `level`, `sahi`, `wrong`, `date`) VALUES
('sunnygkp10@gmail.com', '558921841f1ec', 4, 2, 2, 0, '2015-06-23 09:31:26'),
('sunnygkp10@gmail.com', '558920ff906b8', 4, 2, 2, 0, '2015-06-23 13:32:09'),
('avantika420@gmail.com', '558921841f1ec', 4, 2, 2, 0, '2015-06-23 14:33:04'),
('avantika420@gmail.com', '5589222f16b93', 4, 2, 2, 0, '2015-06-23 14:49:39'),
('sunnygkp10@gmail.com', '5589741f9ed52', 4, 5, 3, 2, '2015-06-23 15:07:16'),
('mi5@hollywood.com', '5589222f16b93', 4, 2, 2, 0, '2015-06-23 15:12:56'),
('nik1@gmail.com', '558921841f1ec', 1, 2, 1, 1, '2015-06-23 16:11:50'),
('sunnygkp10@gmail.com', '5589222f16b93', 1, 2, 1, 1, '2015-06-24 03:22:38');

```

```

--
-- Table structure for table `options`
--

```

```

CREATE TABLE `options` (
  `qid` varchar(50) NOT NULL,
  `option` varchar(5000) NOT NULL,
  `optionid` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

--
-- Dumping data for table `options`
--

```

```

INSERT INTO `options` (`qid`, `option`, `optionid`) VALUES
('55892169bf6a7', 'usermod', '55892169d2efc'),
('55892169bf6a7', 'useradd', '55892169d2f05'),
('55892169bf6a7', 'useralter', '55892169d2f09'),
('55892169bf6a7', 'groupmod', '55892169d2f0c'),
('5589216a3646e', '751', '5589216a48713'),
('5589216a3646e', '752', '5589216a4871a'),
('5589216a3646e', '754', '5589216a4871f'),
('5589216a3646e', '755', '5589216a48722'),
('558922117fce', 'echo', '5589221195248'),
('558922117fce', 'print', '558922119525a'),
('558922117fce', 'printf', '5589221195265'),
('558922117fce', 'cout', '5589221195270'),
('55892211e44d5', 'int a', '55892211f1f97'),
('55892211e44d5', '$a', '55892211f1fa7'),
('55892211e44d5', 'long int a', '55892211f1fb4'),
('55892211e44d5', 'int a$', '55892211f1fbd'),
('558922894c453', 'cin>>a;', '558922895ea0a'),
('558922894c453', 'cin<<a;', '558922895ea26'),
('558922894c453', 'cout>>a;', '558922895ea34'),
('558922894c453', 'cout<a;', '558922895ea41'),

```

('558922899ccaa', 'cout', '55892289aa7cf'),
 ('558922899ccaa', 'cin', '55892289aa7df'),
 ('558922899ccaa', 'print', '55892289aa7eb'),
 ('558922899ccaa', 'printf', '55892289aa7f5'),
 ('558923538f48d', '255.0.0.0', '558923539a46c'),
 ('558923538f48d', '255.255.255.0', '558923539a480'),
 ('558923538f48d', '255.255.0.0', '558923539a48b'),
 ('558923538f48d', 'none of these', '558923539a495'),
 ('55892353f05c4', '192.168.1.100', '5589235405192'),
 ('55892353f05c4', '172.168.16.2', '55892354051a3'),
 ('55892353f05c4', '10.0.0.0.1', '55892354051b4'),
 ('55892353f05c4', '11.11.11.11', '55892354051be'),
 ('558973f4389ac', 'containing root file-system required during bootup', '558973f462e44'),
 ('558973f4389ac', 'Contains only scripts to be executed during bootup', '558973f462e56'),
 ('558973f4389ac', 'Contains root-file system and drivers required to be preloaded during bootup', '558973f462e61'),
 ('558973f4389ac', 'None of the above', '558973f462e6b'),
 ('558973f4c46f2', 'Kernel', '558973f4d4abe'),
 ('558973f4c46f2', 'Shell', '558973f4d4acf'),
 ('558973f4c46f2', 'Commands', '558973f4d4ad9'),
 ('558973f4c46f2', 'Script', '558973f4d4ae3'),
 ('558973f51600d', 'Boot Loading', '558973f526f9d'),
 ('558973f51600d', 'Boot Record', '558973f526fb9'),
 ('558973f51600d', 'Boot Strapping', '558973f526fc5'),
 ('558973f51600d', 'Booting', '558973f526fce'),
 ('558973f55d269', 'Quick boot', '558973f57aef1'),
 ('558973f55d269', 'Cold boot', '558973f57af07'),
 ('558973f55d269', 'Hot boot', '558973f57af17'),
 ('558973f55d269', 'Fast boot', '558973f57af27'),
 ('558973f5abb1a', 'bash', '558973f5e7623'),
 ('558973f5abb1a', 'Csh', '558973f5e7636'),
 ('558973f5abb1a', 'ksh', '558973f5e7640'),
 ('558973f5abb1a', 'sh', '558973f5e764a'),
 ('5589751a63091', 'q', '5589751a81bd6'),
 ('5589751a63091', 'wq', '5589751a81be8'),
 ('5589751a63091', 'both (a) and (b)', '5589751a81bf4'),
 ('5589751a63091', 'none of the mentioned', '5589751a81bfd'),
 ('5589751ad32b8', 'moves screen down one page', '5589751adbdbd'),
 ('5589751ad32b8', 'moves screen up one page', '5589751adbdc'),
 ('5589751ad32b8', 'moves screen up one line', '5589751adbdd8'),
 ('5589751ad32b8', 'moves screen down one line', '5589751adbde2'),
 ('5589751b304ef', 'yy', '5589751b3b04d'),
 ('5589751b304ef', 'yw', '5589751b3b05e'),
 ('5589751b304ef', 'yc', '5589751b3b069'),
 ('5589751b304ef', 'none of the mentioned', '5589751b3b073'),
 ('5589751b749c9', 'X', '5589751b9a98c'),
 ('5589751b749c9', 'x', '5589751b9a9a5'),
 ('5589751b749c9', 'D', '5589751b9a9b7'),
 ('5589751b749c9', 'd', '5589751b9a9c9'),
 ('5589751bd02ec', 'autoindentation is not possible in vi editor', '5589751bdadaa');

--


```
-- Table structure for table `questions`
```

```
--
```

```
CREATE TABLE `questions` (  
  `eid` text NOT NULL,  
  `qid` text NOT NULL,  
  `qns` text NOT NULL,  
  `choice` int(10) NOT NULL,  
  `sn` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
```

```
-- Dumping data for table `questions`
```

```
--
```

```
INSERT INTO `questions` (`eid`, `qid`, `qns`, `choice`, `sn`) VALUES  
(  
'558920ff906b8', '55892169bf6a7', 'what is command for changing user information??', 4, 1),  
(  
'558920ff906b8', '5589216a3646e', 'what is permission for view only for other??', 4, 2),  
(  
'558921841f1ec', '558922117fce', 'what is command for print in php??', 4, 1),  
(  
'558921841f1ec', '55892211e44d5', 'which is a variable of php??', 4, 2),  
(  
'5589222f16b93', '558922894c453', 'what is correct statement in c++??', 4, 1),  
(  
'5589222f16b93', '558922899ccaa', 'which command is use for print the output in c++?', 4, 2),  
(  
'558922ec03021', '558923538f48d', 'what is correct mask for A class IP??', 4, 1),  
(  
'558922ec03021', '55892353f05c4', 'which is not a private IP??', 4, 2),  
(  
'55897338a6659', '558973f4389ac', 'On Linux, initrd is a file', 4, 1),  
(  
'55897338a6659', '558973f4c46f2', 'Which is loaded into memory when system is booted?', 4, 2),  
(  
'55897338a6659', '558973f51600d', 'The process of starting up a computer is known as', 4, 3),  
(  
'55897338a6659', '558973f55d269', 'Bootstrapping is also known as', 4, 4),  
(  
'55897338a6659', '558973f5abb1a', 'The shell used for Single user mode shell is:', 4, 5),  
(  
'5589741f9ed52', '5589751a63091', 'Which command is used to close the vi editor?', 4, 1),  
(  
'5589741f9ed52', '5589751ad32b8', 'In vi editor, the key combination CTRL+f', 4, 2),  
(  
'5589741f9ed52', '5589751b304ef', 'Which vi editor command copies the current line of the file?', 4, 3),  
(  
'5589741f9ed52', '5589751b749c9', 'Which command is used to delete the character before the cursor location in vi editor?',  
4, 4),  
(  
'5589741f9ed52', '5589751bd02ec', 'Which one of the following statement is true?', 4, 5);
```

```
-- -----
```

```
--
```

```
-- Table structure for table `quiz`
```

```
--
```

```
CREATE TABLE `quiz` (  
  `eid` text NOT NULL,  
  `title` varchar(100) NOT NULL,  
  `sahi` int(11) NOT NULL,  
  `wrong` int(11) NOT NULL,  
  `total` int(11) NOT NULL,  
  `time` bigint(20) NOT NULL,  
  `intro` text NOT NULL,  
  `tag` varchar(100) NOT NULL,  
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
--
-- Dumping data for table `quiz`
--

INSERT INTO `quiz` (`eid`, `title`, `sahi`, `wrong`, `total`, `time`, `intro`, `tag`, `date`) VALUES
('558920ff906b8', 'Linux : File Managment', 2, 1, 2, 5, '', 'linux', '2015-06-23 09:03:59'),
('558921841f1ec', 'Php Coding', 2, 1, 2, 5, '', 'PHP', '2015-06-23 09:06:12'),
('5589222f16b93', 'C++ Coding', 2, 1, 2, 5, '', 'c++', '2015-06-23 09:09:03'),
('558922ec03021', 'Networking', 2, 1, 2, 5, '', 'networking', '2015-06-23 09:12:12'),
('55897338a6659', 'Linux:startup', 2, 1, 5, 10, '', 'linux', '2015-06-23 14:54:48'),
('5589741f9ed52', 'Linux :vi Editor', 2, 1, 5, 10, '', 'linux', '2015-06-23 14:58:39');

-----

--
-- Table structure for table `rank`
--

CREATE TABLE `rank` (
  `email` varchar(50) NOT NULL,
  `score` int(11) NOT NULL,
  `time` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `rank`
--

INSERT INTO `rank` (`email`, `score`, `time`) VALUES
('sunnygkp10@gmail.com', 9, '2015-06-24 03:22:38'),
('avantika420@gmail.com', 8, '2015-06-23 14:49:39'),
('mi5@hollywood.com', 4, '2015-06-23 15:12:56'),
('nik1@gmail.com', 1, '2015-06-23 16:11:50');

-----

--
-- Table structure for table `user`
--

CREATE TABLE `user` (
  `name` varchar(50) NOT NULL,
  `gender` varchar(5) NOT NULL,
  `college` varchar(100) NOT NULL,
  `email` varchar(50) NOT NULL,
  `mob` bigint(20) NOT NULL,
  `password` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Dumping data for table `user`
--
```

```

INSERT INTO `user` (`name`,`gender`,`college`,`email`,`mob`,`password`) VALUES
('Avantika', 'F', 'KNIT sultanpur', 'avantika420@gmail.com', 7785068889, 'e10adc3949ba59abbe56e057f20f883e'),
('Mark Zukarburg', 'M', 'Stanford', 'ceo@facebook.com', 987654321, 'e10adc3949ba59abbe56e057f20f883e'),
('Komal', 'F', 'KNIT sultanpur', 'komalpd2011@gmail.com', 7785068889, 'e10adc3949ba59abbe56e057f20f883e'),
('Tom Cruze', 'M', 'Hollywood', 'mi5@hollywood.com', 7785068889, 'e10adc3949ba59abbe56e057f20f883e'),
('Netcamp', 'M', 'KNIT sultanpur', 'netcamp@gmail.com', 987654321, 'e10adc3949ba59abbe56e057f20f883e'),
('Nikunj', 'M', 'XYZ', 'nik1@gmail.com', 987, '202cb962ac59075b964b07152d234b70'),
('Sunny', 'M', 'KNIT sultanpur', 'sunnygkp10@gmail.com', 7785068889, 'e10adc3949ba59abbe56e057f20f883e'),
('User', 'M', 'cimt', 'user@user.com', 11, 'e10adc3949ba59abbe56e057f20f883e'),
('Vikash', 'M', 'KNIT sultanpur@gmail.com', 'vikash@gmail.com', 7785068889, 'e10adc3949ba59abbe56e057f20f883e');

```

```

--
-- Indexes for dumped tables
--

--
-- Indexes for table `admin`
--
ALTER TABLE `admin`
  ADD PRIMARY KEY (`admin_id`);

```

```

--
-- Indexes for table `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`email`);

```

```

--
-- AUTO_INCREMENT for dumped tables
--

```

```

--
-- AUTO_INCREMENT for table `admin`
--
ALTER TABLE `admin`
  MODIFY `admin_id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

LICENSE

The MIT License (MIT)

Copyright (c) 2015 sunnygkp10

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

README.md

online-examination-system-in-php

Online Examination System Today Online Examination System has become a fast growing examination method because of its speed and accuracy. It is also needed less manpower to execute the examination. Almost all organizations now-a-days, are conducting their objective exams by online examination system, it saves students time in examinations. Organizations can also easily check the performance of the student that they give in an examination. As a result of this, organizations are releasing results in less time. It also helps the environment by saving paper. According to today's requirement, online examination project in php is very useful to learn it. What is an online examination system? 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 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 PHP. Administrator of Online Examination has multiple features such as Add, Delete, Update Topics and Question. To Login as Admin put inside your browser "http://www.applicationname/admin" The user will automatically get the updated version by logging using the user ID and Password provided at the time of registration. No need of reprinting, appearance, vigilance and the job is done. Online examination system features Login system must be present and secured by password. Ability to save the answer given by the candidate along with the question. Answer checking system should be available. Could Update Profile Log out after the over. Admin Panel Project objective: Online examination system is a non removable examination pattern of today's life. We need more time saving and more accurate examination system as the number of applicants is increasing day by day. For all IT students and professionals, it is very important to have some basic understanding about the online examination system. On this site you will get source code with the running project. It will help you to understand the concept of the project. Here you find project in php free download. Brief overview of the technology: Front end: HTML, CSS, JavaScript HTML: HTML is used to create and save web document. E.g. Notepad/Notepad++ CSS : (Cascading Style Sheets) Create attractive Layout JavaScript: it is a programming language, commonly use with web browsers. Back end: PHP, MySQL PHP: Hypertext Preprocessor (PHP) is a technology that allows software developers to create dynamically generated web pages, in HTML, XML, or other document types, as per client request. PHP is open source software. MySQL: MySQL is a database, widely used for accessing querying, updating, and managing data in databases. Software Requirement(any one) WAMP Server XAMPP Server MAMP Server LAMP Server

Installation Steps 1. Download zip file and Unzip file on your local server. 2. Put this file inside "c:/wamp/www/" . 3. Database Configuration Open phpmyadmin Create Database named quiz_new. Import database quiz_new.sql from downloaded folder(inside database) 4. Open Your browser put inside "http://localhost/Online_exam_New/" 5. To Login as admin put "http://localhost/Online_exam_New/admin" Admin Login Details Login Id: admin@a