|  |  |
| --- | --- |
| Index | |
| 1 | Introduction |
| 2 | Analysis |
|  | 2.1 System Analysis |
|  | 2.2 Proposed System |
| 3 | System Specifications |
|  | 3.1 Hardware Requirements: |
|  | 3.2 Software Requirements: |
| 4 | Design |
| 5 | Module |
|  | 5.1.1 Admin Module: |
|  | 5.1.2 Instructor Module: |
|  | 5.1.3 Student Module: |
| 6 | Tools And Technologies |
|  | 6.1 Overview Of Technologies |
| 7 | Data Dictionary |
| 8 | Feasibility Study |
| 9 | Implementation |
| 10 | Testing |
| 11 | Screenshots |
| 12 | Future Enhancements: |
| 13 | Bibliography |

1. **INTRODUCTION**

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

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

The administrators, instructor, Students who are attending for online examination can communicate with the system through this 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 use or student. And the details of students who attempted Online Examination are maintained at administrator.

1. **ANALYSIS**
   1. SYSTEM ANALYSIS

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

* + 1. 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.

1. SYSTEM SPECIFICATIONS
   1. Hardware Requirements:

* Pentium-IV(Processor).
* 256 MB Ram
* 512 KB Cache Memory
* Hard disk 10 GB
* Microsoft Compatible 101 or more Key Board
  1. Software Requirements:
* Operating System : Windows
* Web-Technology: PHP
* Front-End: HTML, CSS, JAVASCRIPT
* Back-End: MySQL
* Web Server: Apache SERVER.

4.DESIGN

4.1 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 analysed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

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

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

**UML Diagrams:**

UML stands for Unified Modelling 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 behaviour 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:

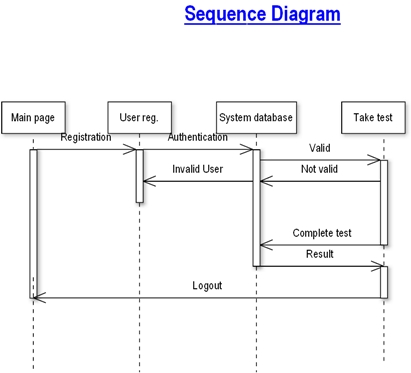
A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioural diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors.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



**SEQUENCE DIAGRAM:**

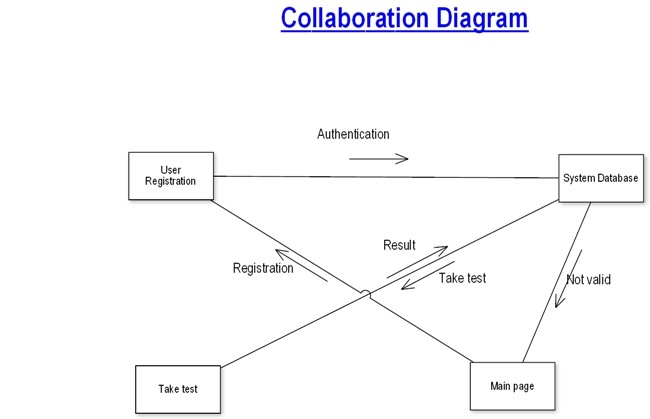
Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

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.



**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.



**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 modelling 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.

The following diagram is an example of an Order System of an application. It describes a particular aspect of the entire application.

* First of all, Order and Customer are identified as the two elements of the system. They have a one-to-many relationship because a customer can have multiple orders.
* Order class is an abstract class and it has two concrete classes (inheritance relationship) Special-order and Normal Order.
* The two inherited classes have all the properties as the Order class. In addition, they have additional functions like dispatch () and receive ().

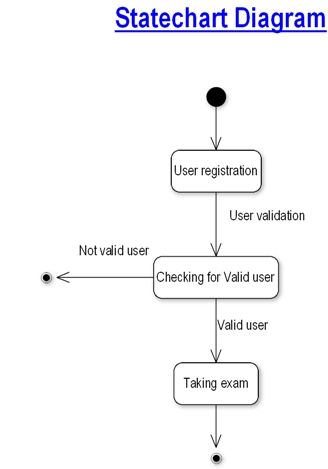


**STATE CHART DIAGRAM:**

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioural diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as State machines and State-chart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behaviour of a class in response to time and changing external stimuli. We can say that each and every class has a state but we don’t model every class using State diagrams. We prefer to model the states with three or more states.

Uses of state chart diagram –

* We use it to state the events responsible for change in state (we do not show what processes cause those events).
* We use it to model the dynamic behaviour of the system.
* To understand the reaction of objects/classes to internal or external stimuli.



**DATA FLOW DIAGRAMS:**

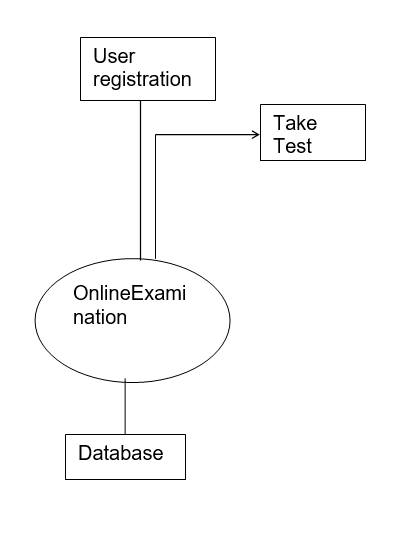
The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labelled 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.

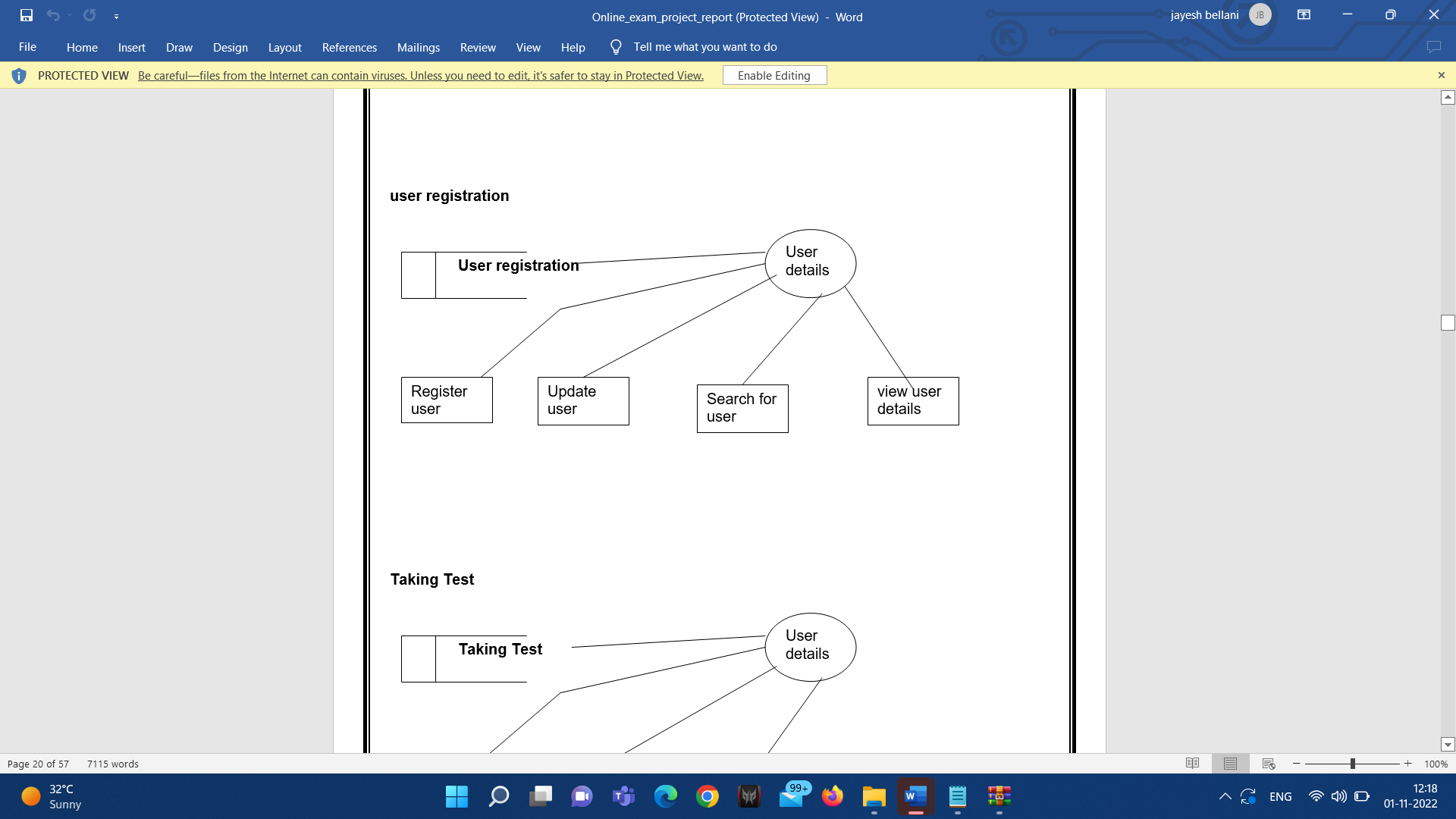
The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst performs an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding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

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

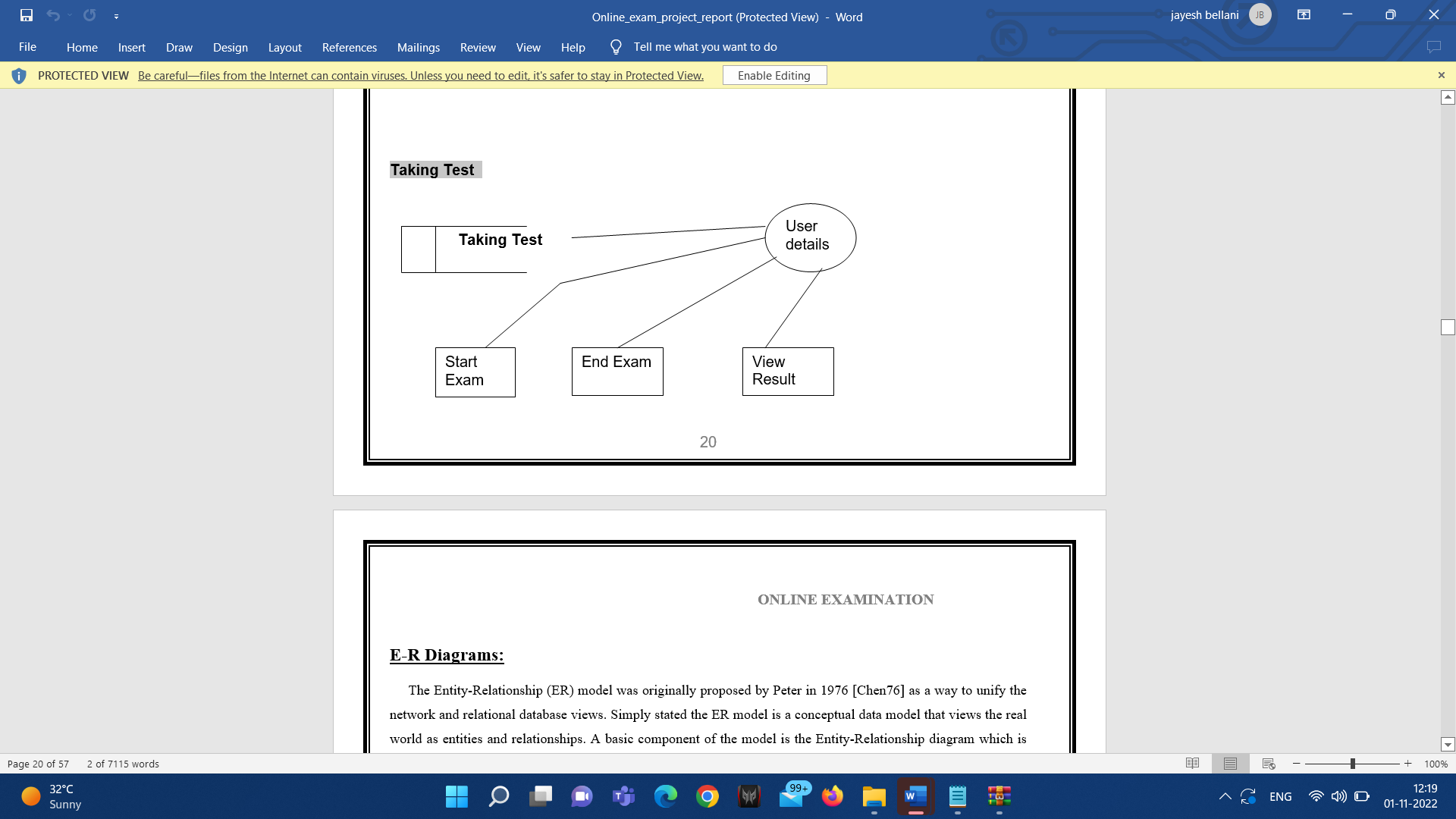
DATABASE:



USER REGISTRATION



TAKING TEST



5. MODULES

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

**5.1.1 ADMIN MODULE:**

1. REGISTER
2. LOGIN
3. CHANGE PASSWORD&FORGOTPASSWORD
4. STUDENT -MODIFING DETAILS
5. DEPARTMENTS-ENTERING/MODIFYING DETAILS
6. INSTRUCTOR DETAILS-MODIFYING DETAILS

**5.1.2 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 TRUE/FALSE

**5.1.3 STUDENT MODULE:**

1. REGISTER
2. LOGIN
3. TAKE EXAM- MULTIPLE CHOICE TRUE/FALSE
4. SEE EXAM RESULTS
5. LOGOUT
6. TOOLS AND TECHNOLOGIES
   1. OVERVIEW OF TECHNOLOGIES

PHP

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

PHP was originally created by [Rasmus Lerdorf](http://en.wikipedia.org/wiki/Rasmus_Lerdorf) in [1995](http://en.wikipedia.org/wiki/1995) and has been in continuous development ever since.

USAGE

PHP is a general-purpose scripting language that is especially suited for [web development](http://en.wikipedia.org/wiki/Web_development). PHP generally runs on a [web server](http://en.wikipedia.org/wiki/Web_server). Any PHP code in a requested file is [executed](http://en.wikipedia.org/wiki/Execution_(computing)) by the PHP runtime, usually to create [dynamic web page](http://en.wikipedia.org/wiki/Dynamic_web_page) content. It can also be used for [command-line](http://en.wikipedia.org/wiki/Command-line) scripting and [client-side](http://en.wikipedia.org/wiki/Client-side) [GUI](http://en.wikipedia.org/wiki/Graphical_user_interface) applications.

It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.PHP primarily acts as a [filter](http://en.wikipedia.org/wiki/Filter_(software)), taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML.

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

HTML

HTML, which stands for Hyper Text Markup Language, is the predominant [markup language](http://en.wikipedia.org/wiki/Markup_language) for [web pages](http://en.wikipedia.org/wiki/Web_page). It provides a means to create [structured documents](http://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](http://en.wikipedia.org/wiki/Semantic) for text such as headings, paragraphs, lists etc as well as for links, quotes, and other items. It allows [images and objects](http://en.wikipedia.org/wiki/HTML_element#Images_and_objects) to be embedded and can be used to create [interactive forms](http://en.wikipedia.org/wiki/HTML_element#Forms). It is written in the form of [HTML elements](http://en.wikipedia.org/wiki/HTML_element) consisting of "tags" surrounded by [angle brackets](http://en.wikipedia.org/wiki/Brackets#Angle_brackets_or_chevrons_.3C_.3E) within the web page content. It can include or can load [scripts](http://en.wikipedia.org/wiki/Scripting_language) in languages such as [JavaScript](http://en.wikipedia.org/wiki/JavaScript) which affect the behavior of HTML processors like [Web browsers](http://en.wikipedia.org/wiki/Web_browser); and [Cascading Style Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) to define the appearance and layout of text and other material.

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

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

An HTML Application is a [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows) application that uses HTML and Dynamic HTML in a browser to provide the application's graphical interface. A regular HTML file is confined to the security model of the web browser, communicating only to web servers and manipulating only webpage objects and [site cookies](http://en.wikipedia.org/wiki/HTTP_cookie). An HTA runs as a fully trusted application and therefore has more privileges, like creation/editing/removal of files and [Windows Registry](http://en.wikipedia.org/wiki/Windows_Registry) entries.

JAVASCRIPT

JavaScript is an [object-oriented](http://en.wikipedia.org/wiki/Object-oriented) [scripting language](http://en.wikipedia.org/wiki/Scripting_language) used to enable [programmatic](http://en.wikipedia.org/wiki/Computer_programming) access to objects within both the [client application](http://en.wikipedia.org/wiki/Client_(computing)) and other [applications](http://en.wikipedia.org/wiki/Application_software). It is primarily used in the form of [client-side JavaScript](http://en.wikipedia.org/wiki/Client-side_JavaScript), implemented as an integrated component of the [web browser](http://en.wikipedia.org/wiki/Web_browser), allowing the development of enhanced [user interfaces](http://en.wikipedia.org/wiki/User_interface) and dynamic [websites](http://en.wikipedia.org/wiki/Website).

JavaScript is a [dialect](http://en.wikipedia.org/wiki/Programming_language_dialect) of the [ECMAScript](http://en.wikipedia.org/wiki/ECMAScript) standard and is characterized as a [dynamic](http://en.wikipedia.org/wiki/Dynamic_language), [weakly typed](http://en.wikipedia.org/wiki/Weak_typing), [prototype-based](http://en.wikipedia.org/wiki/Prototype-based_programming) language with [first-class functions](http://en.wikipedia.org/wiki/First-class_function). JavaScript was influenced by many languages and was designed to look like [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), but to be easier for non-programmers to work with.

JavaScript uses [prototypes](http://en.wikipedia.org/wiki/Prototype-based_programming) instead of [classes](http://en.wikipedia.org/wiki/Class_(computer_science)) for [inheritance](http://en.wikipedia.org/wiki/Inheritance_(computer_science)). It is possible to simulate many class-based features with prototypes in JavaScript.Functions double as object constructors along with their typical role. Prefixing a function call with new creates a new object and calls that function with its local this keyword bound to that object for that invocation. The constructor's prototype property determines the object used for the new object's internal prototype.

A [JavaScript engine](http://en.wikipedia.org/wiki/JavaScript_engine) (also known as JavaScript interpreter or JavaScript implementation) is an [interpreter](http://en.wikipedia.org/wiki/Interpreter_(computing)) that interprets JavaScript [source code](http://en.wikipedia.org/wiki/Source_code) and executes the [script](http://en.wikipedia.org/wiki/Computer_program) accordingly. The first JavaScript engine was created by [Brendan Eich](http://en.wikipedia.org/wiki/Brendan_Eich) at Netscape Communications Corporation, for the [Netscape Navigator](http://en.wikipedia.org/wiki/Netscape_Navigator) [web browser](http://en.wikipedia.org/wiki/Web_browser).

A web browser is by far the most common host environment for JavaScript. Web browsers typically use the public [API](http://en.wikipedia.org/wiki/Application_programming_interface) to create "host objects" responsible for reflecting the [DOM](http://en.wikipedia.org/wiki/Document_Object_Model) into JavaScript.

**MySQL**

MySQL tutorial provides basic and advanced concepts of MySQL. Our MySQL tutorial is designed for beginners and professionals.

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company.

Our MySQL tutorial includes all topics of MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

1. DATA DICTIONARY

USER REG TABLE

|  |  |  |  |
| --- | --- | --- | --- |
| NAME | NULL/NOTNULL | TYPE | KEY |
| ID | NOTNULL | INT | PRIMARYKEY |
| NAME | NULL | VARCHAR(50) |  |
| DOB | NULL | DATETIME |  |
| GENDER | NULL | VARCHAR(10) |  |
| BRANCH | NULL | VARCHAR(20) |  |
| COLLEGE | NULL | VARCHAR(50) |  |
| UID | NULL | VARCHAR(50) |  |
| PWD | NULL | VARCHAR(20) |  |
| RPWD | NULL | VARCHAR(20) |  |
| UTYPE | NULL | VARCHAR(20) |  |
| QUE | NULL | VARCHAR(500) |  |
| ANS | NULL | VARCHAR(500) |  |

TRUE/FALSE BASED QUESTION TABLE

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

TRUE/FALSE BASED ANSWER TABLE

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

Options Based Question Table

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

OPTIONS BASED ANSWERS

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

ALL STUDENT MARKS

|  |  |  |  |
| --- | --- | --- | --- |
| NAME | NULL/NOTNULL | TYPE | KEY |
| ID | NULL | INT |  |
| MARKS | NULL | INT |  |

EXAM SCHEDULE

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

1. FEASIBILITY STUDY

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

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

1. Technical Feasibility

2. Economic Feasibility

3. Operational Feasibility.

1. Technical Feasibility

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

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

2. Economic Feasibility

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

3. Operational Feasibility

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

1. 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 posterized 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 result 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, grieve 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’s 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.

**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.

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 continues 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 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

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.

1. TESTING

Testing is a process of executing a program with the indent 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:**

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 through 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 with in 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:

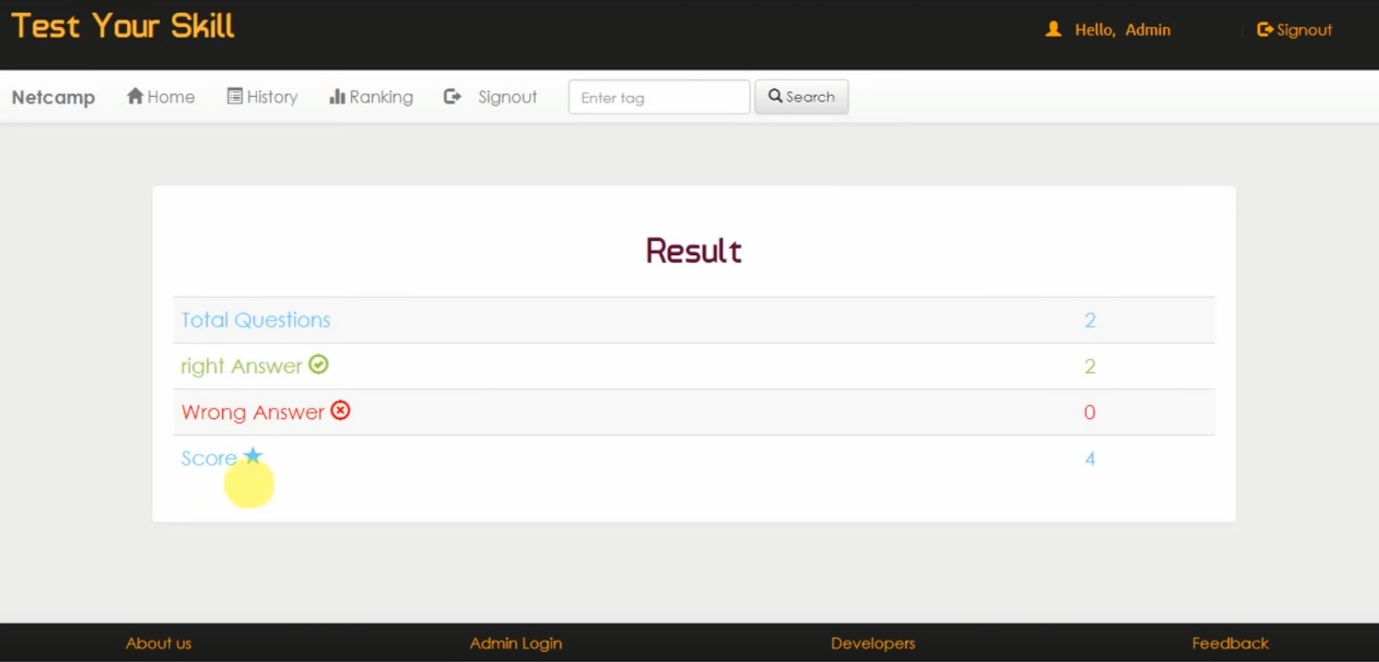
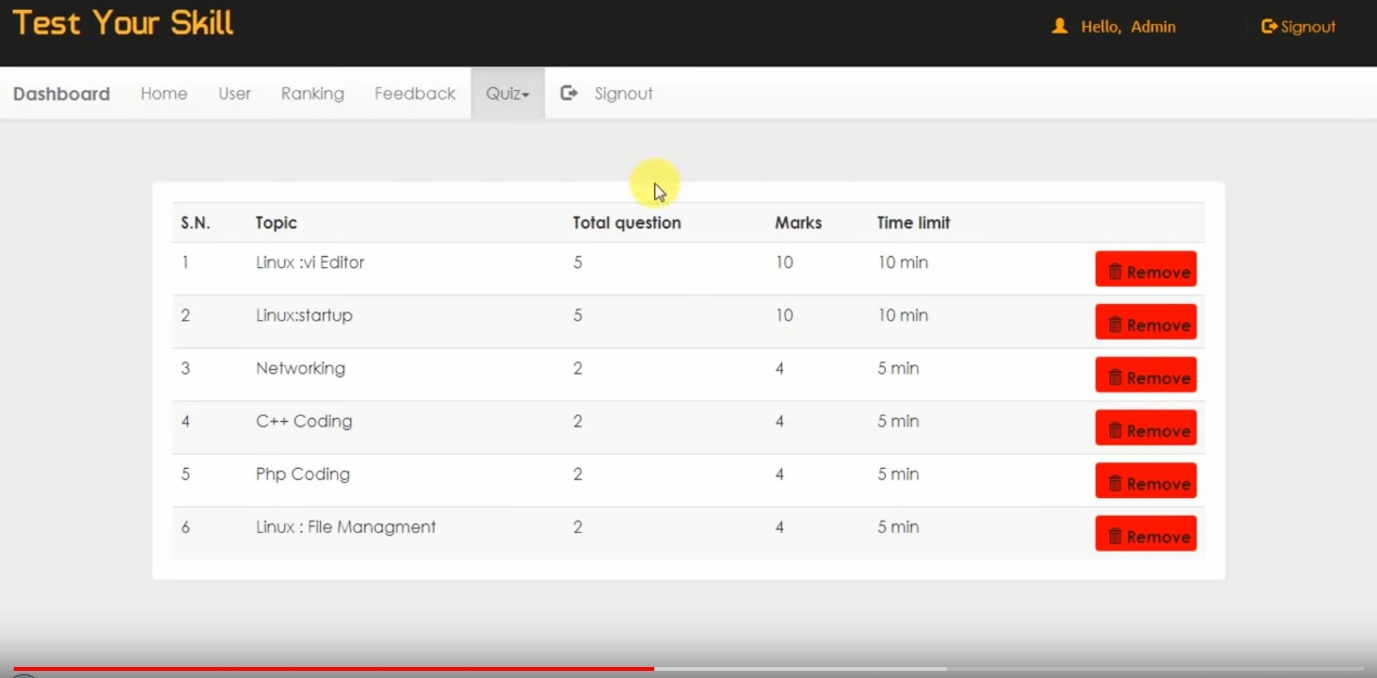
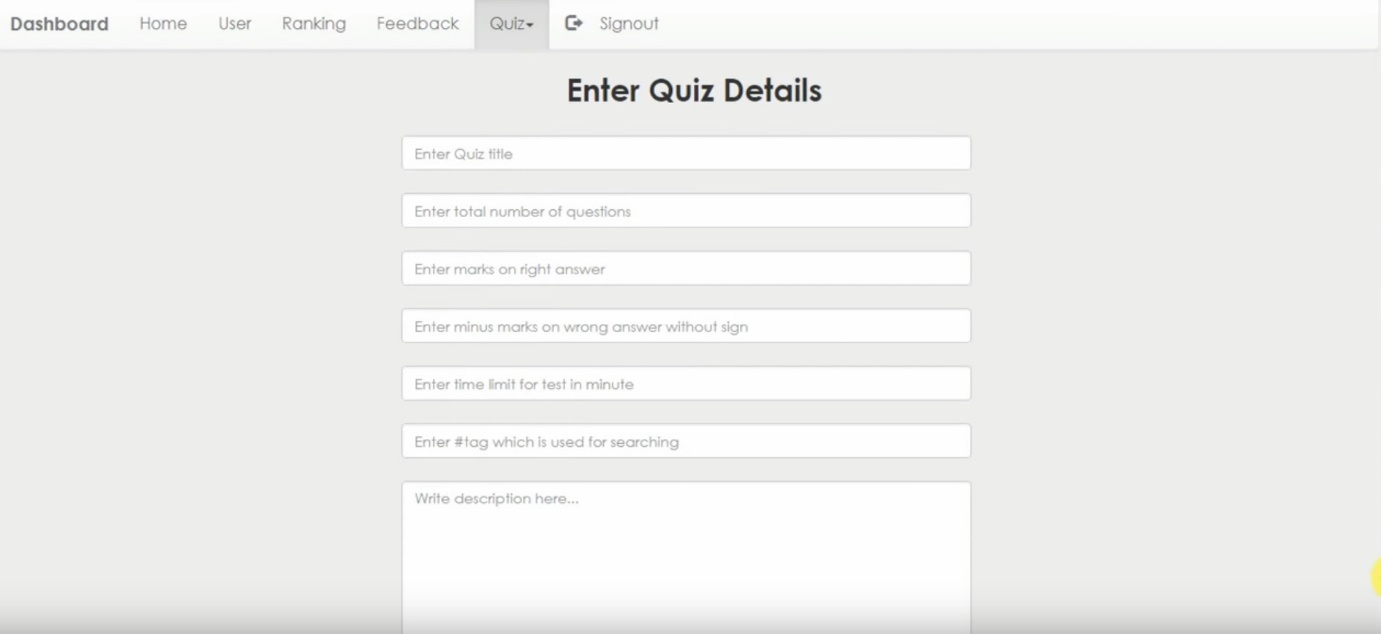
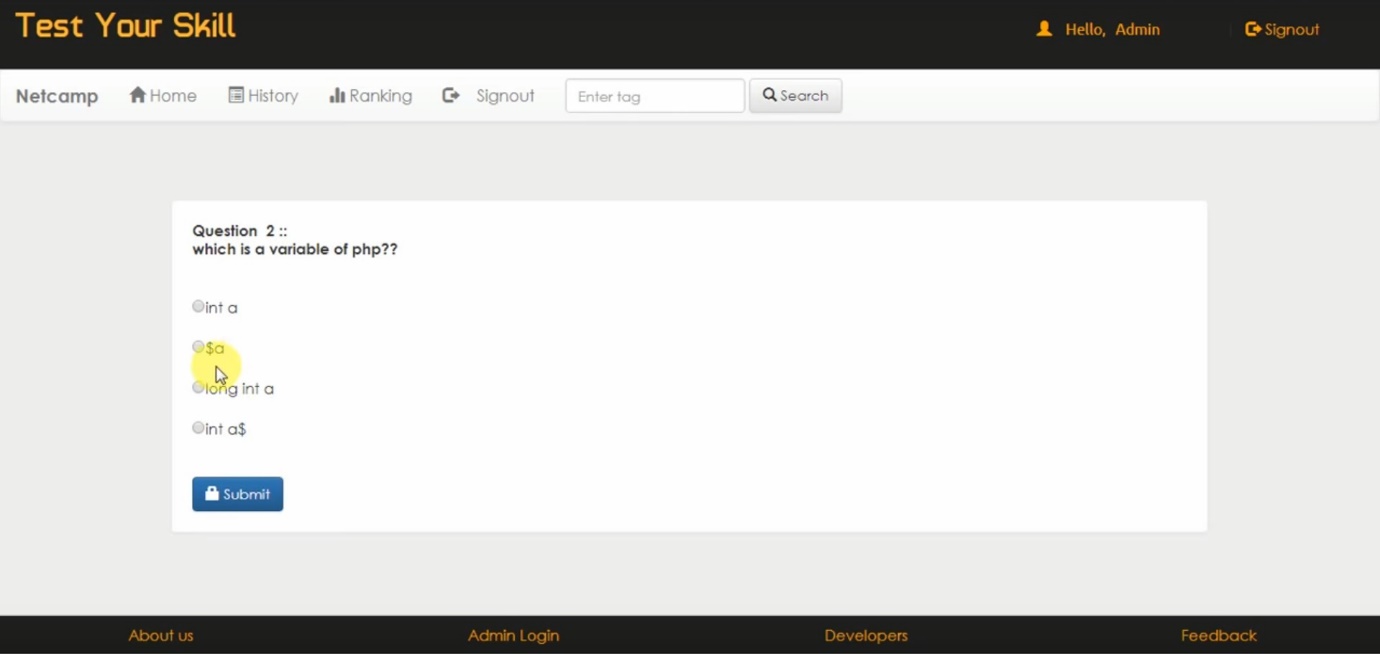
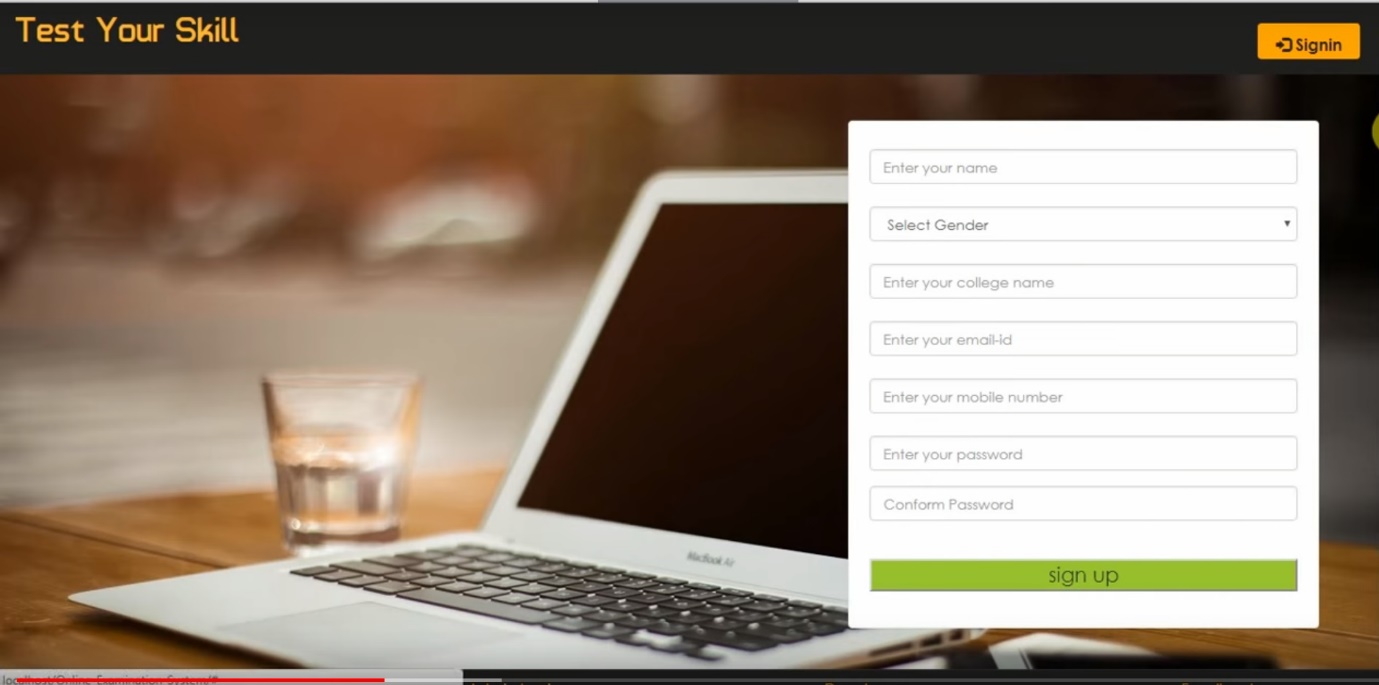
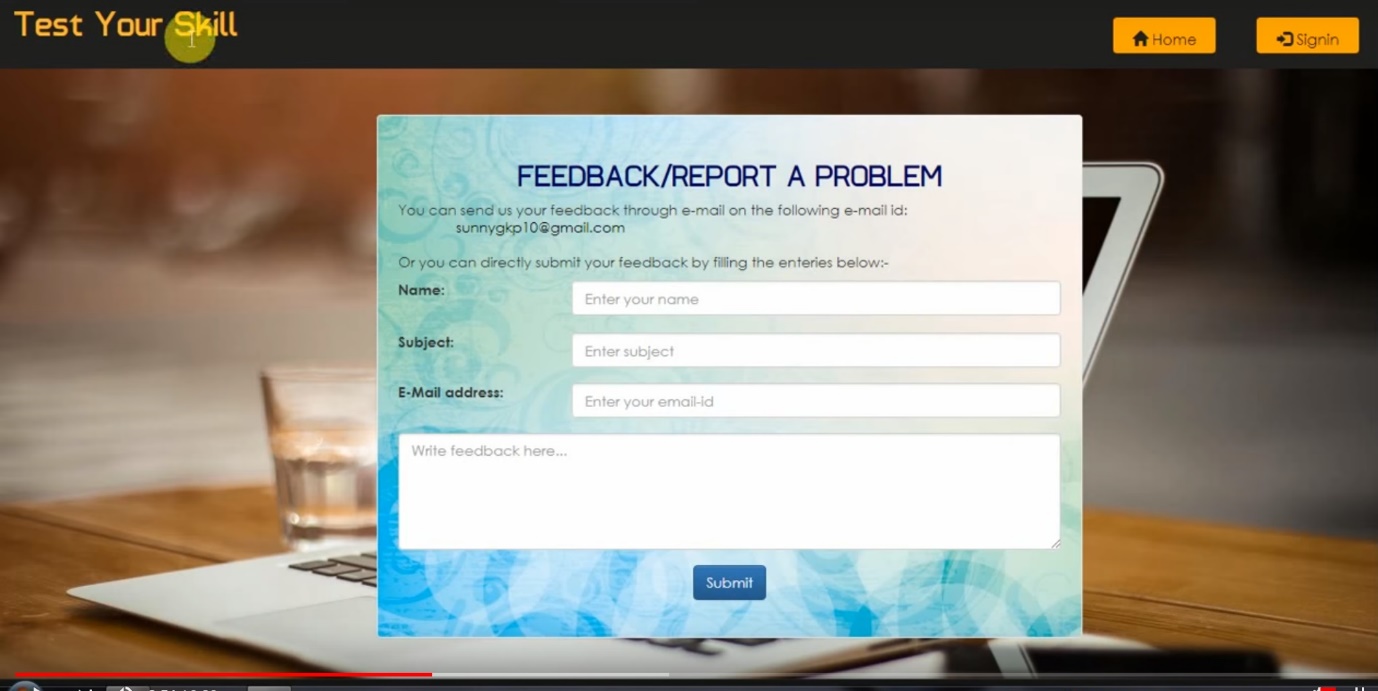
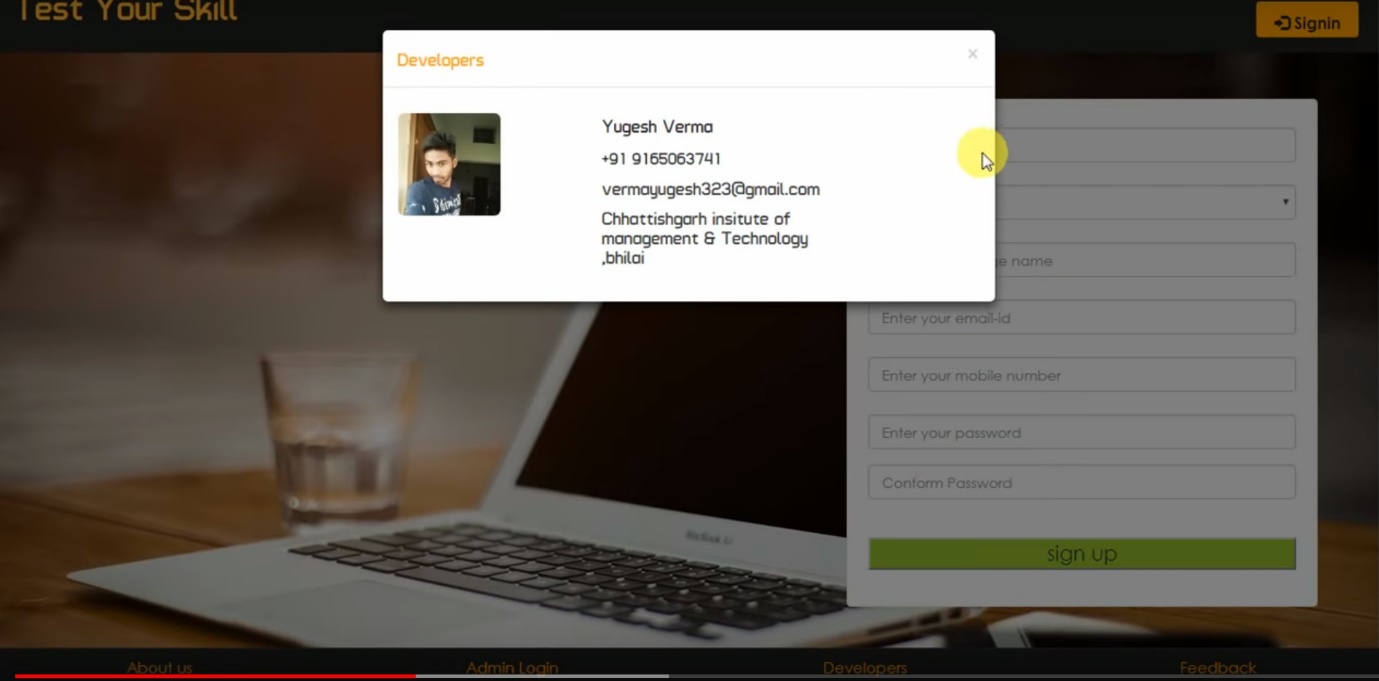
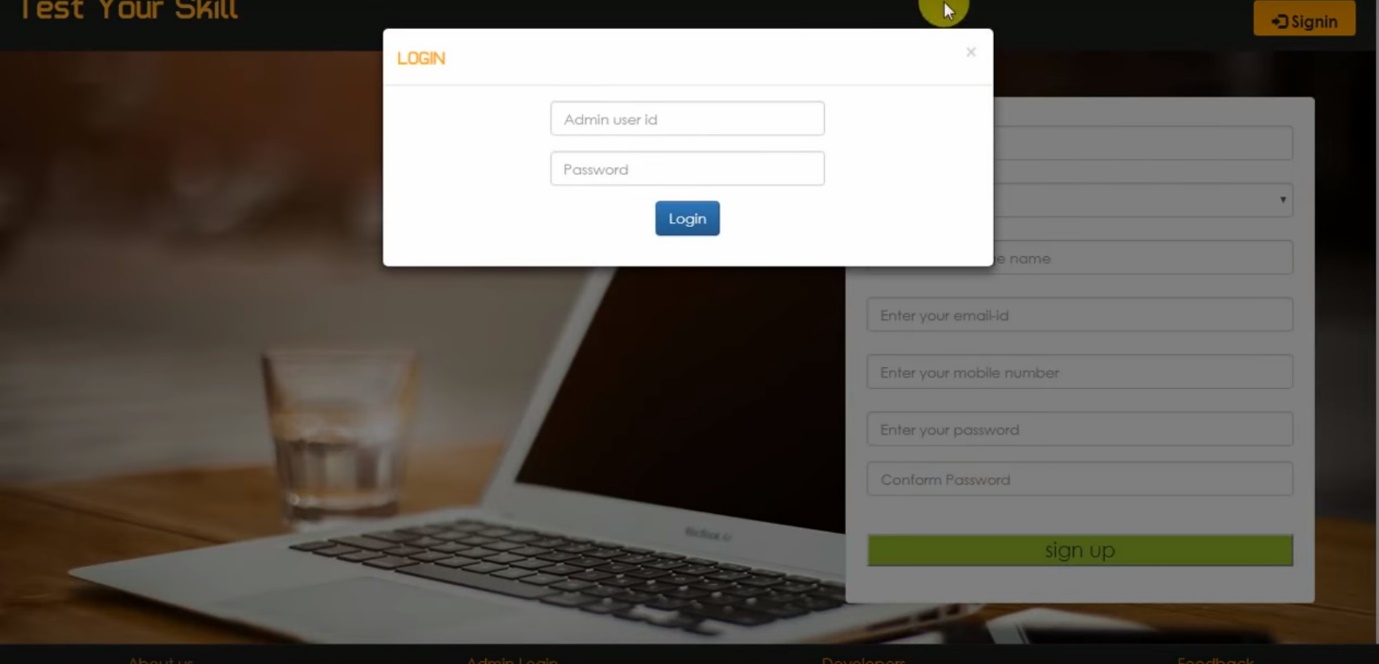
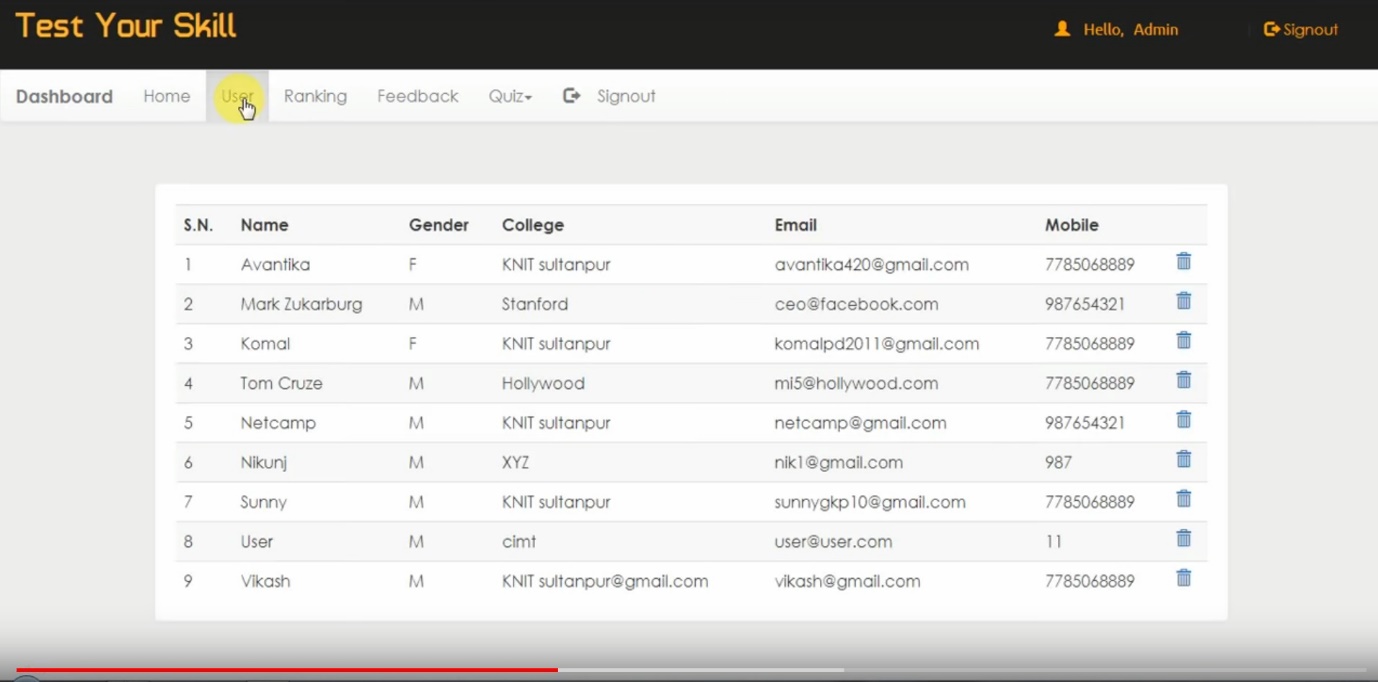
Test coverage Analyzer – records the control paths followed for each test case.

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.

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 with in 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.

1. SCREENSHOTS



1. FUTURE ENHANCEMENTS:

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

Currently issued.

Well I and my team members have worked hard in order to present an improved website better than the existing one’s regarding the information about the various activities. Still ,we found out that the project can be done in a better way. Primarily, when we request information about a particular schedule 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.

1. BIBLIOGRAPHY

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

Books Referred:

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