**Introduction to the Industry/ Training Institute**

# Alpha in K P Road, Anantnag

ALPHA Computer Training Institutes is one of the highly recognized training centres in this city and was established in 2008. It has been playing a pivotal role ever since its establishment by enabling the students to have easy access to rich learning programs. It has been constantly empowering the students by helping them identify the uniqueness of their courses and curriculum while at the same time making them understand as to how to use these courses to their best advantage. The establishment is run by a team of qualified and professional people, having enormous industry knowledge and skills. The locality around is well-connected with different parts of the city due to which accessing the institute is a convenient affair. This listing is also listed in Computer Training Institutes.

## 1.1 Services Offered:

ALPHA enjoys a good infrastructure. The facility is endowed with a conducive learning and knowledge sharing environment. Students have convenient access to essential learning tools and study materials. The faculty team is very approachable; always keen to help and support whenever a student needs that. Each of them is very passionate about teaching and mentoring. A lot of its previous students are doing great in their professional life, many of whom recognize this institute for that and feel lucky to have had enrolled here. This establishment has been successful in shaping the future of many of its students and continues to do so. It identifies the talent in each individual, analyses their needs and nurtures them accordingly so that they can obtain the needed excellence.

One can get various trainings related to compuer science and IT’s like:

PHP

HTML

JAVA

PYTHON

.NET

CLOUD COMPUTING

LINUX

NETWORKING

C++

**1.2 How an effective industrial training contributes to shape your carrier?**

Along with several other benefits, Industrial training provides the first-hand working experience to the students alongside their course. Industrial training is an attempt to make the candidates familiar with the real working environment. The curriculum of such training is designed with the motive to brush up the skills of the students such as personal attitude, communication skills, learning capacity, etc. This further gives a definite shape to your dreams.

With every day new inventions and enhancement in technologies, it is essential to keep track of it in order to not miss the train. Industrial training incorporates all the latest news in their curriculum to help keep the students updated.

**Build Confidence:** Industrial training provides students with a real-like working environment in order to make them familiar and confident with it. The faculties have a friendly approach towards their students in order to encourage them to ask as many questions which in turn will build their confidence towards life.

**Prioritize Lab Practical:**It is a fact that practical teachings are always better than the theoretical ones. Human mind grabs the visuals more rapidly than the words. We are completely aware of this fact and so the industrial training is more focused towards lab practical in order to visually enhance the knowledge of the candidates.

**Future Building tool:** Industrial training is designed with the aim to build up all the essentials required in the candidate for a brighter future. From skill-building to improvising knowledge among the student, the industrial training programs guarantee visible results.

**Conduction of extra classes:** Extra classes are conducted with the motive to leave no doubt in the student. This session is open for the students to come up with their queries so that our experienced trainers could help them solve it.

**Brush up skills and increase knowledge:** The ability of the industrial training to enhance skills and develop knowledge in the students is like a cherry on the top. It is this benefit of the industrial training that helps the students in the process of building a brighter career in their near future.

**1.3 Why should you choose to get the industrial training at Alpha**

If you are searching the best institute in Anantnag then ALPHA should definitely be your option. Established since 2008 is one of the top-notch names who provide the **best six months industrial training in Anantnag**. Our teams of well experienced and qualified trainers are easily approachable. We provide quality and cost-effective training in various courses such as PHP, Java, Cloud Computing, C/C++ etc. We do our best to provide the best learning experience to the candidates.

In a nutshell:

* Deliver best training in various courses such as PHP, Java, etc.
* Keep you updated about all the latest inventions and technologies.
* Conduct doubt clearing sessions to leave no space for any sort of doubt in the student.
* Faculties are the working professionals and experienced in their field.
* Make students familiar with the real working environment.
* Focus on Practical works more than memorizing theories

**PROJECT TITLE**

**MY FEEDBACK SYSTEM**

* 1. **INTRODUCTION:**

My Feedback System for college students have been developed which aims to rate and analyse the college faculty’s performance.

This type of Student Feedback system reduces, the strenuous work of physically examining the feedback pages of each and every student.

The system also reduces the burden of efforts and burden of keeping and maintaining the records on a manual base, it requires a lot of space and safety to keep up such records.

Also the students feedbacks can be tempered for wrong reasons in case of paper based feedbacks wherein the MFS will always ensure safety of feedbacks privacy.

Another important features of the MFS is that physical presence of neither the admin nor the student is required for the either giving the feedback nor for assessing the feedback.

Also further enhancement can be done and more features can be added for better retrieval of the feedback details.

My feedback system is the web based feedback collecting system from the students and provides the automatic generation of a feedback which is given by students. I have developed student feedback system to provide feedback in a quick and easy manner to the particular department. So we call it as Student Staff Feedback System which delivers via the student staff interface as online system which acting as a Service Provider.

By using this technology we can give feedback in online system as fast as compare to the existing paper feedback system. This project has three kinds of users :

\* Student .

\* Faculty.

\* Admin.

The student can give feedback in online system provided by college staff. The existing system carries more time to do a piece of work for this reason the online system feedback is implemented. First of staff can prepare questions & add, update these questions to the online system. After that it was viewed by the students and can give feedback about the lecturers.

Students will fill online feedback using a standard form. In this project security is also maintain that is the result of feedback is only visible to authentic user. This feedback report was checked by the Admin

.

**2.2 OBJECTIVES**

1) Decision making power is provided by this system.

2) Accurate result can be obtained.

3) This system makes Selection process more effective

4) To increase efficiency proposed system is depend on classification method.

5) Proposed system is used to reduce confusion at the time of processing feedback.

**2.3 SYSTEM IMPLEMENTATION**

In this project we implemented the system which works on the concept of online feedback from students which is the replacement of existing paper feedback system which is so complicated method for finding an average for particular subject feedback which is submitted by students. This project has three kinds of users Student, Faculty and Admin. In this system we have design main three forms student login form, admin login form, faculty login form. Student will login in their login form with proper id and password which is provided by admin at the time of registration. The student can give feedback in online system provided by college staff. First of staff can prepare questions and add, update these questions to the online system. After giving the feedback the page appeared which contains a message like you are feedback is submitted. Then after feedback is submitted there is a log out button after click on it student will successfully log out from their profile. In main page it contains some hyperlinks like contact us, frequently ask questions, and submit the queries. And contains three particular modules like student login, admin login, faculty login etc. Admin will fill the information of students as well as view their feedback, add and delete student and generate the average report of that feedback. After that log out button occurs and admin simply logout from the system. Student will login by using user id and password and after the login page will be open contains subjects, suggestion box, dashboard, submit feedback and after their occurs a log out button for logout successfully from the system.

**2.4 MODULES OF MY FEEDBACK SYSTEM**

The proposed system consists of three modules:

* + 1. **Student :**

Student can give the feedback about the lecturers on the scale of five. Students can give feedback about the lecturer based on interaction of lecturer in the class room with students. According to that, students can give feedback as per the given grades.

The student will have the authorities on this as following:

1. Add questions
2. Update questions
3. View questions
4. Giving feedback about the lecturers
   * 1. **Faculty:**

The feedback given by the students can be viewed by the faculty and improve their performance in teaching and other aspects.

The Faculty will have the authorities on this as following:

1) Add questions

1. Update questions
2. View questions
3. View report
   * 1. **Admin :**

These feedback reports were checked by the Admin ie Hod’s . He can view overall grades and view the grades obtained to the lecturers and give this report to the principal and he can give counseling to the college staff.

The Admin will have the authorities on this as following:

1) Add questions

1. Update questions
2. View questions
3. View report
4. Delete feedback

**2.5 HIGHLIGHTS:**

 Saves your time

 Manage the entire process in easy and quick way

 Enhance the staff

 Improve the issuing standards

**2.6 System Actors(Users)**

* Admin
* Faculty
* Student

### **2.7 Admin Features**

* Admin Login
* Admin can add new faculty
* Admin can View faculty list
* Admin can Update faculty data
* Admin can Delete faculty
* Admin can Manage Student
* Admin can Check feedback given by student
* Admin can check average feedback for faculty
* Admin can update the contact details dynamically
* Admin can update password

### **2.8 Faculty Features**

* Faculty can login
* Faculty can update their profile
* Faculty can update Password
* Faculty can Check feedback given by students
* Faculty can check average feedback given by students
* Logout

### **2.9 Student Features**

* Student can Register
* Student can Login
* Student can view profile
* Student can update profile
* Student can update Password
* Student can choose faculty and give feedback for every Questions

### **2.10 Software Requirements**

* XAMPP server
* **Language Used**
  1. **Front End :**HTML 5 ,BOOTSTRAP
  2. **Server Language :**PHP 5.5
  3. **Backe End :**MYSQL

**OPERATIONAL STRUCTURE**

The general methodology in developing a system is involved in different phases, which describe the systems life cycle model for developing software project. The concept includes not only forward motion but also have the possibility to return that is cycle back to an activity previously completed. This cycle back or feedback may occur as a result of the failure with the system to meet a performance objective or as a result of changes in redefinition of system activities. Like most systems, the life cycle of the computer based system also exhibits distinct phases.

Those are,

* **Requirement Analysis Phase**
* **Design Phase**
* **Development Phase**
* **Coding Phase**
* **Testing Phase**

**3.1 REQUIREMENT ANALYSIS PHASE:**

This phase includes the identification of the problem, in order to identify the problem; we have to know information about the problem, the purpose of the evaluation for problem to be known. We have to clearly know about the client’s requirements and the objectives of the project.

**3.2 SYSTEM ANALYSIS PHASE:**

Feasibility analysis involves the benefits of various approaches and the determination of the alternative approaches all through methods like questionnaires and interviews etc., different data about the project are collected and the data through out the project is represented in the graphical form.

**3.3 DESIGN PHASE:**

Software design is a process through which the requirements are translated into a representation of software(s/w). One of the software requirements have been analyzes and specifies, the s/w design involves three technical activities: design, code generation and testing. The design of the system is in modular form i.e., the s/w is logically partitioned into components that perform specific functions and sub functions. The design phase leads to modules that exhibit independent functional characteristics. It even leads to an interface that reduces the complexity of the connections between modules and with the external environment. The design phase main importance because in this activity, decisions ultimately affect the success of s/w implementation and maintenance.

**3.4 DEVELOPMENT PHASE:**

The development phase includes choosing of a suitable s/w to solve the particular problem given. The various facilities and the sophistication in the selected s/w give a better development of the problem.

**3.5 CODING PHASE:**

The coding phase is for translating the design of the system produced during the design phase into code in a programming language, which can be executed by a computer and which performs the computation specified by the user.

**3.6 TESTING PHASE:**

Testing is done in various ways such as testing the algorithm, programming code, simple data debugging is also one of the testing methodology.

**BACKGROUND INFORMATION**

## 4.1 What is an HTML File?

* HTML stands for **H**yper **T**ext **M**arkup **L**anguage
* Hyper text is ordinary text that has been dressed up with features such as formatting, images, multimedia and links to other documents.
* Markup is the process of taking ordinary text and adding extra symbols.
* An HTML file is a text file containing small **markup tags**
* The markup tags tell the Web browser **how to display** the page
* An HTML file must have an .**htm** or .**html** file extension
* An HTML file can be created using a **simple text editor**

## 4.2 AN EXAMPLE OF HTML:

If you are running Windows, start Notepad and type in the following text:

<html>

<head>

<title>Title of page</title>

</head>

<body>

This is my first homepage. <b>This text is bold</b>

</body>

</html>

Start your Internet browser. Select "Open" (or "Open Page") in the File menu of your browser. A dialog box will appear. Select "Browse" (or "Choose File") and locate the HTML file you just created - "mypage.htm" - select it and click "Open". Now you should see an address in the dialog box, for example "C:\MyDocuments\mypage.htm". Click OK, and the browser will display the page.

## 4.3 Example Explained:

The first tag in your HTML document is <html>. This tag tells your browser that this is the start of an HTML document. The last tag in your document is </html>. This tag tells your browser that this is the end of the HTML document.

The text between the <head> tag and the </head> tag is header information. Header information is not displayed in the browser window.

The text between the <title> tags is the title of your document. The title is displayed in your browser's caption.

The text between the <body> tags is the text that will be displayed in your browser.

The text between the <b> and </b> tags will be displayed in a bold font.

## 4.4 HTML Tags:

* HTML tags are used to mark-up HTML **elements**
* HTML tags are surrounded by the **two characters < and >**
* The surrounding characters are called **angle brackets**
* HTML tags normally **come in pairs** like <b> and </b>
* The first tag in a pair is the **start tag,** the second tag is the **end tag**
* The text between the start and end tags is the **element content**
* HTML tags are **not case sensitive,** <b> means the same as <B>

## 4.5 HTML Elements:

This is an HTML element:

<b>This text is bold</b>.

The HTML element starts with a **start tag**: <b>  
The **content** of the HTML element is: This text is bold  
The HTML element ends with an **end tag**: </b>.

The purpose of the <b> tag is to define an HTML element that should be displayed as bold.

## 4.6 HTML Tag Attributes:

* HTML tags can have attributes.
* Attributes provide additional information to an HTML element.
* Attributes always come in name/value pairs like this: name="value".
* <body>: This tag defines the body of an HTML document.<body bgcolor="yellow"> has additional information about the background color.
* <table>: This tag defines an HTML table. (You will learn more about HTML tables later)
* <table border="1"> has additional information about the border around the table
* Attributes are always specified in the start tag of an HTML element.

**4.7 HTML Advantages:**

* HTML is the primary format used on the World Wide Web.
* HTML can display Web pages with a wide range of colors, shapes, and objects.
* HTML is so flexible, many browsers and Web applications have added their own functionality to the base HTML protocol.
* HTML ensures consistency in style across elements that have the same meaning.

**DATABASE MANAGEMENT SYSTEM**

**5.1 DATABASE:**

A database is a collection of data, with some internal meaning design built populated with data for a specific purpose.

**5.2 CHARACTERISTICS:**

1. Represents complex relationship between data.
2. Keeps control on data redundancy.
3. Enforces data access authorization.
4. Has automatic intelligent backup and recovery procedure for data.
5. Keeps a centralized data dictionary for the storage of information pertaining to data and manipulation.

**5.3 DATA MODELS:**

The structure of the database is the conceptual model, collection of tools for describing data relationship, semantics and consistency constrains.

**5.3.1 Object Based models**

Object-based logical model are used in describing at the conceptual and view levels there are two of object-based models:

**i.** The entity relationship model: this model is model based on a Real world, which consists of a collection of objects called entities and relationship among these.

**ii.** The object Oriented model: The Object oriented model is based on the collection of objects. Methods are grouping into classes a class may be viewed as a type definition-for-objects.

**5.3.2** **Records based Models:**

Record based models are so named because the data is structured in fixed format records. The three most accepted record based data models are relation, network, and hierarchical models.

**i**. **Relation Model**: The Relational represents the data and relational ship among data by a collection of tables, where each table has a number of columns with unique names.

**ii. Network model**: Data in network model are represented by collection of records and relationships among data are represented by links, which can be viewed as pointers. The records in the database are organized as collection of arbitrary graphs.

**iii. Hierarchical Model**: This model is similar to network model in the sense that data and relationships among data represented by records and links, respectively. It differs from the network model in that the records are organized as collection of trees rather than arbitrary graphs.

**5.4 RELATIONAL DATABASE MODEL:**

A Relational database consists of collection of the tables, each of which is assigned unique name. Each table contains number of columns also with unique names. Relational is an association among several entities. The row in a table represents a relation ship among a set of values.

**5.4.1 TABLES:**

In relational system data is organized and presented in tables. Tables are simply a collection of related rows and columns.

**5.4.2 ROWS:**

Rows are the horizontal components of a table. They are unnamed and unordered. This means that there is no way to specify a particular table rows by its position in a table are accessed by data values only.

**5.4.3 COLUMNS:**

Columns are the vertical components of table. Unlike rows, columns are named and ordered. Every table columns has a unique name within the table and add a data attributes for it.

**5.4.4 FIELDS:**

Fields refer to the specific data values stored in a table for a particular intersection of row and column. A field is the smallest unit of data in SQL. A null field denotes absence of data values for that particular field.

**5.5 ADVANTAGES OF RDMS**:

**1. Redundancy can be reduced**: In a non-database system, each application has its own private file. This often leads to considerable redundancy in stored data with result wastage in storage space.

**2. Inconsistency can be avoided**: There will be some occasions on which two entries do not agree. This is called inconsistency. It can be avoided in a database by propagation updates.

**3. Data can be shared**: It means that not only the existing application can share the data in a database but also new applications can be developed.

**4. Standards can be enforced**: With central control of data, certain standards like Industrial, National, and international can be developed.

**5. Security Restrictions can be applied**: Having complete control of data, we can ensure that the only means of accessing data is through proper channel.

**6. Integrity can be maintained**: The problem of integrity is the problem of ensuring data in the database is accurate. It can be ensured that by definition validation procedures whenever updating operation is to be carried out.

**7. Conflicting requirements can be balanced**: Database can be structured to provide an over-all service.

**MYSQL**

### **6.1 What is MySQL?**

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB.

* **MySQL is a database management system**.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

* **MySQL is a relational database management system**.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MySQL” stands for “Structured Query Language.” SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard.

* **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs.

* **The MySQL Database Server is very fast, reliable, and easy to use**.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years.

* **MySQL Server works in client/server or embedded systems.**

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different back ends. We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

### **6.2 The Main Features of MySQL:**

* Written in C and C++.
* Tested with a broad range of different compilers.
* Works on many different platforms.
* The MySQL Server design is multi-layered with independent modules.
* Fully multi-threaded using kernel threads. It can easily use multiple CPUs if they are available.
* Relatively easy to add other storage engines. This is useful if you want to provide an SQL interface for an in-house database.
* A very fast thread-based memory allocation system.
* In-memory hash tables, which are used as temporary tables.
* The MySQL code is tested with Purify (a commercial memory leakage detector).
* The server is available as a separate program for use in a client/server networked environment. It is also available as a library that can be embedded (linked) into standalone applications. Such applications can be used in isolation or in environments where no network is available.

**6.3 Distinguishing features:**

The following features are implemented by MySQL but not by some other RDBMSes:

* Multiple storage engines, allowing you to choose the one which is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in. In MySQL 5.1, storage engines can be dynamically loaded at Run time.
* Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, BDB, EXAMPLE)
* Partner-developed storage engines (InnoDB, solidDB, NitroEDB, BrightHouse)
* Community-developed storage engines (memcached, httpd, PBXT)

**6.4 DATATYPES:**

* Many data types: signed/unsigned integers 1, 2, 3, 4, and 8 bytes long, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM.

**6.5 STATEMENTS AND FUNCTIONS:**

* Full operator and function support in the SELECT list and WHERE clause of queries.
* For example:
* mysql> **SELECT CONCAT(first\_name, ' ', last\_name)**
* -> **FROM citizen**
* -> **WHERE income/dependents > 10000 AND age > 30;**
* Full support for SQL GROUP BY and ORDER BY clauses. Support for group functions ([COUNT()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_count), [COUNT(DISTINCT ...)](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_count), [AVG()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_avg), [STD()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_std), [SUM()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_sum), [MAX()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_max), [MIN()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_min), and [GROUP\_CONCAT()](http://dev.mysql.com/doc/refman/5.0/en/group-by-functions.html#function_group-concat)).
* Support for LEFT OUTER JOIN and RIGHT OUTER JOIN with both standard SQL and ODBC syntax.
* DELETE, INSERT, REPLACE, and UPDATE return the number of rows that were changed (affected). It is possible to return the number of rows matched instead by setting a flag when connecting to the server.
* The MySQL-specific SHOW statement can be used to retrieve information about databases, storage engines, tables, and indexes.
* The EXPLAIN statement can be used to determine how the optimizer resolves a query.

**6.6 SECURITY:**

* A privilege and password system that is very flexible and secure, and that allows host-based verification.
* Passwords are secure because all password traffic is encrypted when you connect to a server.

**6.7 SCALABILITY AND LIMITS:**

* Handles large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 60,000 tables and about 5,000,000,000 rows.

**6.8 CONNECTIVITY:**

* Clients can connect to MySQL Server using several protocols:
  + Clients can connect using TCP/IP sockets on any platform.
  + On UNIX systems, clients can connect using UNIX domain socket files.
* MySQL client programs can be written in many languages. A client library written in C is available for clients written in C or C++, or for any language that provides C bindings.
* APIs for C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, and Tcl are available, allowing MySQL clients to be written in many languages.
* The Connector/ODBC (MyODBC) interface provides MySQL support for client programs that use ODBC (Open Database Connectivity) connections. The Connector/J interface provides MySQL support for Java client programs that use JDBC connections. Clients can be run on Windows or UNIX. Connector/J source is available.

MySQL Connector/NET enables developers to easily create .NET applications that require secure, high-performance data connectivity with MySQL aware tools. Developers can build applications using their choice of .NET languages.

**6.9 WHY MYSQL?**

* The MySQL® database has become the world's most popular open source database because of its consistent fast performance, high reliability and ease of use. It's used on every continent (even Antarctica!) by individual Web developers as well as many of the world's largest and fastest-growing organizations to save time and money powering their high-volume Web sites, business-critical systems and packaged software -- including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, and Zappos.com.
* Not only is MySQL the world's most popular open source database, it's also become the database of choice for a new generation of applications built on the LAMP stack (Linux, Apache, MySQL, PHP / Perl / Python.) MySQL runs on more than 20 platforms including Linux, Windows, OS/X, HP-UX, AIX, Netware, giving you the kind of flexibility that puts you in control.
* Whether you're new to database technology or an experienced developer or DBA, MySQL offers a comprehensive range of certified software, support, training and consulting to make you successful.

## 

## PHP

## 7.1 What is PHP?

* PHP stands for **P**HP: **H**ypertext **P**reprocessor
* PHP is a server-side scripting language, like ASP
* PHP scripts are executed on the server
* PHP supports many databases (MySQL, Informix, Oracle, Sybase, Solid, Generic ODBC, etc.)
* PHP is an open source software (OSS)
* PHP is free to download and use
* PHP is a reflective programming language originally designed for producing dynamic web pages.

## 7.2 What is a PHP File?

* PHP files may contain text, HTML tags and scripts
* PHP files are returned to the browser as plain HTML
* PHP files have a file extension of ".php", ".php3", or ".phtml"

## 7.3 Why PHP?

* PHP runs on different platforms (Windows, Linux, Unix, etc.)
* PHP is compatible with almost all servers used today (Apache, IIS, etc.)
* PHP is FREE to download from the official PHP resource: [www.php.net](http://www.php.net/)
* PHP is easy to learn and runs efficiently on the server side.

**7.4 Features of PHP:**

* Robust support for object oriented programming
* The php data objects extension, which defines a lightweight and consistent interface for accessing databases.
* Better support for mysql through a completely rewritten extension.
* Error handling through exception

**7.5 Sample PHP Code:**

Copy the code paste it on you account, and save it as info.php4.

## <?php

## Phpinfo();

## ?>

## It will output information about the current state of PHP, which information of PHP compilation options and extensions, the PHP version, server information and environment, the PHP environment, OS version information, paths, master and local values of configuration options, HTTP headers, and the PHP license.

## 7.6 Comments in PHP:

In PHP, we use // to make a single-line comment or /\* and \*/ to make a large comment block.

<html>

<body>

<?php

//This is a comment

/\*

This is

a comment

block

\*/

?>

</body>

## </html>

## 7.7 PHP as Server-side scripting:

## Server-side scripting is a web server technology in which a user's request is fulfilled by running a script directly on the web server to generate dynamic HTML pages. It is usually used to provide interactive web sites that interface to databases or other data stores. This is different from client-side scripting where scripts are run by the viewing web browser, usually in JavaScript.

## The primary advantage to server-side scripting is the ability to highly customize the response based on the user's requirements, access rights, or queries into data stores.

## 7.8 Basic PHP Syntax:

A PHP scripting block always starts with **<?php** and ends with **?>**. A PHP scripting block can be placed anywhere in the document.

A PHP file normally contains HTML tags, just like an HTML file, and some PHP scripting code.

Below, we have an example of a simple PHP script which sends the text "Hello World" to the browser:

<html>

<body>

<?php

echo "Hello World";

?>

</body>

</html>

Each code line in PHP must end with a semicolon. The semicolon is a separator and is used to distinguish one set of instructions from another.

There are two basic statements to output text with PHP: **echo** and **print**. In the example above we have used the echo statement to output the text "Hello World".

## 7.9 Variables in PHP:

* Variables are used for storing a values, like text strings, numbers or arrays.
* When a variable is set it can be used over and over again in your script
* All variables in PHP start with a $ sign symbol.
* The correct way of setting a variable in PHP: $var\_name=value

**7.10 DATATYPES:**

## PHP stores whole numbers in a platform-dependent range. This range is typically that of 32-bit signed integers. Integer variables can be assigned using decimal (positive and negative), octal and hexadecimal notations. Real numbers are also stored in a platform-specific range.

PHP has a native Boolean type, named "boolean", similar to the native Boolean types in Java and C++. Using the Boolean type conversion rules, non-zero values are interpreted as true and zero as false, as in Perl.

There are eight data types in PHP:

1. Integer
2. Double
3. Boolean
4. String
5. Object
6. Array
7. Null
8. Resource

The null data type represents a variable that has no value. The only value in the null data type is NULL.

### 

### **7.11 Libraries:**

PHP includes a large number of free and open source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing FTP servers, many database servers, embedded SQL libraries such as embedded MySQL and SQLite, LDAP servers, and others. Many functions familiar to C programmers such as those in the ‘stdio’ family are available in the standard PHP build.

### **7.12 Extension:**

PHP allows developers to write extensions in ‘C’ to add functionality to the PHP language. These can then be compiled into PHP or loaded dynamically at runtime.

### **7.13 Debuggers and profilers:**

Debuggers and profilers allow developers to analyze running PHP code for potential and noted software bugs and bottlenecks. Examples of such software for PHP include APD and Xdebug.

**7.14 The Advantages of PHP:**

* PHP is one of the most popular server side scripting languages running today. It is used for creating dynamic webpages that interact with the user offering customized information.
* PHP offers many advantages; it is fast, stable, secure, easy to use and open source (free).

**REQUIREMENTS SPECIFICATION**

**8.1 SOFTWARE REQUIREMENTS:**

* Operating system : WINDOWS XP/ 7/8/10/RED HAT/LINUX
* Connectivity : PHP
* Data base : MYSQL
* Web browser : MOZILLA or IE
* Front end : HTML
* Server : XAMPP

**8.2 HARDWARE REQUIREMENTS:**

* Processor : PENTIUM-IV & above.
* RAM : 256MB & above.
* Hard Disk : 20GB & above.
* Monitor

**CODE USED**

**9.1 PHP & HTML:**

* 1. **Index.php**

<?php

session\_start();

require('dbconfig.php'); ?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<meta name="description" content="">

<meta name="author" content="">

<title>My feedback System</title>

<!-- Bootstrap Core CSS -->

<link href="css/bootstrap.min.css" rel="stylesheet">

<!-- Custom CSS -->

<link href="css/modern-business.css" rel="stylesheet">

<!-- Custom Fonts -->

<link href="font-awesome/css/font-awesome.min.css" rel="stylesheet" type="text/css">

</head>

<body>

<!-- Navigation -->

<nav class="navbar navbar-default navbar-fixed-top" role="navigation" style="background:#0000FF">

<div class="container" >

<!-- Brand and toggle get grouped for better mobile display -->

<div class="navbar-header"><button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="index.php" style="color:#FFFFFF">MY FEEDBACK SYSTEM</a>

</div>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

<ul class="nav navbar-nav navbar-right">

<li style="color:#FFFFFF">

<a style="color:#FFFFFF" href="index.php"><i class="fa fa-home fa-fw"></i>HOME</a>

</li>

<li style="color:#FFFFFF">

<a style="color:#FFFFFF" href="index.php?info=about"><i class="fa fa-home fa-fw"></i>ABOUT</a>

</li>

<li><a style="color:#FFFFFF" href="index.php?info=registration"><i class="fa fa-sign-out fa-fw"></i>REGISTRATION</a></li>

<li class="dropdown">

<a style="color:#FFFFFF" href="#" class="dropdown-toggle" data-toggle="dropdown" href="#"><i class="fa fa-sign-in fa-fw"></i>LOGIN

<span class="caret"></span></a>

<ul class="dropdown-menu">

<li><a href="index.php?info=login">STUDENT</a></li>

<li><a href="index.php?info=faculty\_login">FACULTY</a></li>

<li><a href="admin">ADMIN</a></li>

</ul>

</li>

<li>

<a style="color:#FFFFFF" href="index.php?info=contact"><i class="fa fa-phone fa-fw"></i>CONTACT</a>

</li>

</ul>

</div>

<!-- /.navbar-collapse -->

</div>

<!-- /.container -->

</nav>

<?php

@$info=$\_GET['info'];

if($info!="")

{

if($info=="about")

{

include('about.php');

}

else if($info=="contact")

{

include('contact.php');

}

else if($info=="login")

{

include('login.php');

}

else if($info=="faculty\_login")

{

include('faculty\_login.php');

}

else if($info=="registration")

{

include('registration.php');

}

}

else

{

?>

<!-- slider start -->

<header id="myCarousel" class="carousel slide">

<!-- Indicators -->

<ol class="carousel-indicators">

<li data-target="#myCarousel" data-slide-to="0" class="active"></li>

<li data-target="#myCarousel" data-slide-to="1"></li>

<li data-target="#myCarousel" data-slide-to="2"></li>

</ol>

<!-- Wrapper for slides -->

<div class="carousel-inner">

<div class="item active">

<div class="fill" style="background-image:url('images/1.jpg');"></div>

<div class="carousel-caption">

</div>

</div>

<div class="item">

<div class="fill" style="background-image:url('images/2.jpg');"></div>

<div class="carousel-caption">

c

</div>

</div>

<div class="item">

<div class="fill" style="background-image:url('images/3.jpg');"></div>

<div class="carousel-caption">

</div>

</div>

<!-- Controls -->

<a class="left carousel-control" href="#myCarousel" data-slide="prev">

<span class="icon-prev"></span>

</a>

<a class="right carousel-control" href="#myCarousel" data-slide="next">

<span class="icon-next"></span>

</a>

</header>

<!-- slider -->

<!-- Page Content -->

<div class="container">

<div class="row">

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

<div class="col-sm-10" style="margin-top:60px;margin-bottom:80px">

<h2>About My feedback System</h2>

The Feedback System In PHP is a simple mini project for giving feedback to any company services or any system performance. The project contains an admin side and the user side. The admin can be view the feedback from others and evaluate it. The purpose of giving feedback is to improve the situation or the person’s/organization’s performance.

The users are the feedback providers. A user can write their feedback with their point of view and submit it. Users’ names, email, and phone are optional, they can submit without writing those things. This project is a very simple project that makes a convenient way for any organization to evaluate their works, services, and facilities.

</div>

<?php } ?>

</div>

</div>

<!-- /.container -->

<div class="navbar-fixed-bottom nav navbar-inverse text-center" style="padding:15px;height:40px; background:#000000">

<span style="color:#FFFFFF">Developed By ....... <a href="http://www.cyberarju.com">ARJUMAND NISSAR</a> </span>

</div>

<!-- jQuery -->

<script src="css/jquery.js"></script>

<!-- Bootstrap Core JavaScript -->

<script src="css/bootstrap.min.js"></script>

<!-- Script to Activate the Carousel -->

<script>

$('.carousel').carousel({

interval: 5000 //changes the speed

})

</script>

</body>

</html>

* 1. **Admin**

<?php

session\_start();

if(!isset($\_SESSION['user']))

{

header('location:index.php');

}

include('../dbconfig.php');

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<meta name="description" content="">

<meta name="author" content="">

<title>Admin Dashboard</title>

<!-- Bootstrap Core CSS -->

<link href="../css/bootstrap.min.css" rel="stylesheet">

<!-- MetisMenu CSS -->

<link href="../css/metisMenu.min.css" rel="stylesheet">

<!-- Custom CSS -->

<link href="../css/sb-admin-2.css" rel="stylesheet">

<!-- Custom Fonts -->

<link href="../css/font-awesome.min.css" rel="stylesheet" type="text/css">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->

<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>

<script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body>

<div id="wrapper">

<!-- Navigation -->

<nav class="navbar navbar-default navbar-static-top" role="navigation" style="margin-bottom: 0">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="dashboard.php">My Feedback System</a>

</div>

<!-- /.navbar-header -->

<ul class="nav navbar-top-links navbar-right">

<!-- /.dropdown -->

<li class="dropdown">

<a class="dropdown-toggle" data-toggle="dropdown" href="#">

<i class="fa fa-user fa-fw"></i> <i class="fa fa-caret-down"></i>

</a>

<ul class="dropdown-menu dropdown-user">

<li><a href="dashboard.php?info=update\_password"><i class="fa fa-gear fa-fw"></i>Change Password</a>

</li>

<li class="divider"></li>

<li><a href="logout.php"><i class="fa fa-sign-out fa-fw"></i> Logout</a>

</li>

</ul>

<!-- /.dropdown-user -->

</li>

<!-- /.dropdown -->

</ul>

<!-- /.navbar-top-links -->

<div class="navbar-default sidebar" role="navigation">

<div class="sidebar-nav navbar-collapse">

<ul class="nav" id="side-menu">

<li>

<a href="dashboard.php"><i class="fa fa-dashboard fa-fw"></i> Dashboard</a>

</li>

<li>

<a href="#"><i class="fa fa-user fa-fw"></i>Faculty<span class="fa arrow"></span></a>

<ul class="nav nav-second-level">

<li>

<a href="dashboard.php?info=add\_faculty"><i class="fa fa-plus fa-fw"></i> AddFaculty</a>

</li>

<li>

<a href="dashboard.php?info=show\_faculty"><i class="fa fa-eye"></i> Manage faculty</a>

</li>

</ul>

<!-- /.nav-second-level -->

</li>

<li>

<a href="#"><i class="fa fa-user fa-fw"></i>Student<span class="fa arrow"></span></a>

<ul class="nav nav-second-level">

<li>

<a href="dashboard.php?info=display\_student"><i class="fa fa-eye"></i> Manage Student</a>

</li>

</ul>

</li>

<!-- feedback-->

<li>

<a href="#"><i class="fa fa-user fa-book"></i>Feedback<span class="fa arrow"></span></a>

<ul class="nav nav-second-level">

<li><a href="dashboard.php?info=feedback"><i class="fa fa-eye"></i> feedback</a></li>

<li><a href="dashboard.php?info=feedback\_average"><iclass="fa fa-eye"></i> feedback Average</a></li>

</ul>

</li>

<!--feedback end-->

<li>

<a href="dashboard.php?info=contact"><i class="fa fa-eye"></i> Contact us</a>

</li>

</ul>

</div>

<!-- /.sidebar-collapse -->

</div>

<!-- /.navbar-static-side -->

</nav>

<div id="page-wrapper">

<div class="row">

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

<?php

@$id=$\_GET['id'];

@$info=$\_GET['info'];

if($info!="")

{

if($info=="add\_faculty")

{

include('add\_faculty.php');

}

elseif($info=="show\_faculty")

{

include('show\_faculty.php');

}

elseif($info=="edit\_faculty")

{

include('edit\_faculty.php');

}

elseif($info=="display\_student")

{

include('display\_student.php');

}

elseif($info=="contact")

{

include('contact.php');

}

elseif($info=="feedback")

{

include('feedback.php');

}

elseif($info=="feedback\_average")

{

include('feedback\_average.php');

}

else if($info=="update\_password")

{

include('update\_password.php');

}

}

else

{

include('dashboard\_home.php');

}

?>

</div>

<!-- /.col-lg-12 -->

</div>

<!-- /.row -->

</div>

<!-- /#page-wrapper -->

</div>

<!-- /#wrapper -->

<!-- jQuery -->

<script src="../css/jquery.min.js"></script>

<!-- Bootstrap Core JavaScript -->

<script src="../css/bootstrap.min.js"></script>

<!-- Metis Menu Plugin JavaScript -->

<script src="../css/metisMenu.min.js"></script>

<!-- Custom Theme JavaScript -->

<script src="../css/sb-admin-2.js"></script>

</body>

</html>

* 1. **User**

<?php

session\_start();

include('../dbconfig.php');

$user= $\_SESSION['user'];

if($user=="")

{header('location:../index.php');}

$sql=mysqli\_query($conn,"select \* from user where email='$user' ");

$users=mysqli\_fetch\_assoc($sql);

//print\_r($users);

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<!-- The above 3 meta tags \*must\* come first in the head; any other head content must come \*after\* these tags -->

<meta name="description" content="">

<meta name="author" content="">

<link rel="icon" href="../../favicon.ico">

<title>My feedback System</title>

<!-- Bootstrap core CSS -->

<link href="../css/bootstrap.min.css" rel="stylesheet">

<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->

<link href="../css/ie10-viewport-bug-workaround.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="../css/dashboard.css" rel="stylesheet">

<!-- Just for debugging purposes. Don't actually copy these 2 lines! -->

<!--[if lt IE 9]><script src="../../assets/js/ie8-responsive-file-warning.js"></script><![endif]-->

<script src="../js/ie-emulation-modes-warning.js"></script>

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>

<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

<![endif]-->

</head>

<body>

<nav class="navbar navbar-inverse navbar-fixed-top" style="background:#428bca">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">

<span class="sr-only">Toggle navigation</span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" style="color:#FFFFFF" href="#">Hello <?php echo $users['name'];?></a>

</div>

<div id="navbar" class="navbar-collapse collapse">

<ul class="nav navbar-nav navbar-right">

<li><a href="logout.php" style="color:#FFFFFF">Logout</a></li>

</ul>

</div>

</div>

</nav>

<div class="container-fluid">

<div class="row">

<div class="col-sm-3 col-md-2 sidebar">

<ul class="nav nav-sidebar">

<li class="active"><a href="index.php">Dashboard <span class="sr-only">(current)</span></a></li>

<!-- find users' image if image not found then show dummy image -->

<!-- check users profile image -->

<?php

$q=mysqli\_query($conn,"select image from user where email='".$\_SESSION['user']."'");

$row=mysqli\_fetch\_assoc($q);

if($row['image']=="")

{

?>

<li><a href="#"><img style="border-radius:50px" src="../images/person.jpg" width="100" height="100" alt="not found"/></a></li>

<?php

}

else

{

?>

<li><a href="#"><img style="border-radius:50px" src="../images/<?php echo $\_SESSION['user'];?>/<?php echo $row['image'];?>" width="100" height="100" alt="not found"/></a></li>

<?php

}

?>

<li><a href="index.php?page=update\_password"><span class="glyphicon glyphicon-user"></span> Update Password</a></li>

<li><a href="index.php?page=update\_profile"><span class="glyphicon glyphicon-asterisk"></span> Update Profile</a></li>

<li><a href="index.php?page=feedback"><span class="glyphicon glyphicon-thumbs-down"></span> Feedback</a></li>

</ul>

</div>

<div class="col-sm-9 col-sm-offset-3 col-md-10 col-md-offset-2 main">

<!-- container-->

<?php

@$page= $\_GET['page'];

if($page!="")

{

if($page=="update\_password")

{

include('update\_password.php');

}

if($page=="update\_profile")

{

include('update\_profile.php');

}

if($page=="feedback")

{

include('give\_feedback.php');

}

}

else

{

?>

<h1 class="page-header">Dashboard</h1>

<div class="row placeholders">

<div class="col-xs-6 col-sm-3 placeholder">

<img src="data:image/gif;base64,R0lGODlhAQABAIAAAHd3dwAAACH5BAAAAAAALAAAAAABAAEAAAICRAEAOw==" width="150" height="150" class="img-responsive" alt="Generic placeholder thumbnail">

<h4>Label 1</h4>

<span class="text-muted"></span>

</div>

<div class="col-xs-6 col-sm-3 placeholder">

<img src="data:image/gif;base64,R0lGODlhAQABAIAAAHd3dwAAACH5BAAAAAAALAAAAAABAAEAAAICRAEAOw==" width="150" height="150" class="img-responsive" alt="Generic placeholder thumbnail">

<h4>Label 2</h4>

<span class="text-muted"></span>

</div>

<div class="col-xs-6 col-sm-3 placeholder">

<img src="data:image/gif;base64,R0lGODlhAQABAIAAAHd3dwAAACH5BAAAAAAALAAAAAABAAEAAAICRAEAOw==" width="150" height="150" class="img-responsive" alt="Generic placeholder thumbnail">

<h4>Label 3</h4>

<span class="text-muted"></span>

</div>

<div class="col-xs-6 col-sm-3 placeholder">

<img src="data:image/gif;base64,R0lGODlhAQABAIAAAHd3dwAAACH5BAAAAAAALAAAAAABAAEAAAICRAEAOw==" width="150" height="150" class="img-responsive" alt="Generic placeholder thumbnail">

<h4>Label 4</h4>

<span class="text-muted"></span>

</div>

</div>

<?php } ?>

</div>

</div>

</div>

<!-- Bootstrap core JavaScript

================================================== -->

<!-- Placed at the end of the document so the pages load faster -->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>

<script>window.jQuery || document.write('<script src="../js/vendor/jquery.min.js"><\/script>')</script>

<script src="../js/bootstrap.min.js"></script>

<!-- Just to make our placeholder images work. Don't actually copy the next line! -->

<script src="../js/vendor/holder.min.js"></script>

<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->

<script src="../js/ie10-viewport-bug-workaround.js"></script>

</body>

</html>

**9.2 My SQL**

**Database**

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET time\_zone = "+00:00";

--

-- **Database: `feedback\_system`**

--

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

--

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

--

CREATE TABLE IF NOT EXISTS `admin` (

`admin\_id` int(11) NOT NULL AUTO\_INCREMENT,

`user` varchar(50) NOT NULL,

`pass` varchar(50) NOT NULL,

PRIMARY KEY (`admin\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=2 ;

--

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

--

INSERT INTO `admin` (`admin\_id`, `user`, `pass`) VALUES

(1, 'admin@gmail.com', 'admin');

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

--

-- **Table structure for table `contact`**

--

CREATE TABLE IF NOT EXISTS `contact` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` varchar(50) NOT NULL,

`email` varchar(100) NOT NULL,

`mobile` bigint(20) NOT NULL,

`message` text NOT NULL,

`Date` datetime NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=7 ;

--

-- **Dumping data for table `contact`**

--

INSERT INTO `contact` (`id`, `name`, `email`, `mobile`, `message`, `Date`) VALUES

(5, 'Aasim', 'aasimparray@gmail.com', 9596338761, 'ddd', '2016-06-29 17:53:28'),

(6, 'Aasim', 'aasimparray@gmail.com', 9596338761, 'ddd', '2016-06-29 17:53:43');

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

--

-- **Table structure for table `faculty`**

--

CREATE TABLE IF NOT EXISTS `faculty` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`user\_alias` varchar(30) NOT NULL,

`Name` varchar(30) NOT NULL,

`designation` varchar(100) NOT NULL,

`programme` varchar(50) NOT NULL,

`semester` varchar(10) NOT NULL,

`email` varchar(255) NOT NULL,

`password` varchar(75) NOT NULL,

`mobile` bigint(20) NOT NULL,

`date` datetime NOT NULL,

`status` int(11) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `email` (`email`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=12 ;

--

-- **Dumping data for table `faculty`**

--

INSERT INTO `faculty` (`id`, `user\_alias`, `Name`, `designation`, `programme`, `semester`, `email`, `password`, `mobile`, `date`, `status`) VALUES

(7, 'ish30', 'ish', 'Associate Professior', 'B.Tech', 'ii', 'ish@gmail.com', 'ish', 9015501897, '2016-07-13 14:30:53', 0),

(8, 'arju9015', 'arju', 'Developer', 'B.tech', 'ii', 'arju@gmail.com', 'arju', 9015501897, '2016-07-13 14:37:35', 0),

(11, 'saak9015', 'saaku', 'aaaa', 'B.tec', 'i', 'saaku@gmail.com', 'saaku', 901550189, '2016-07-13 14:40:35', 1);

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

--

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

--

CREATE TABLE IF NOT EXISTS `feedback` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`faculty\_id` varchar(50) NOT NULL,

`Teacher provided the course outline having weekly content plan w` enum('5','4','3','2','1') NOT NULL,

`Course objectives,learning outcomes and grading criteria are cle` enum('5','4','3','2','1') NOT NULL,

`Course integrates throretical course concepts with the real worl` enum('5','4','3','2','1') NOT NULL,

`Teacher is punctual,arrives on time and leaves on time` enum('5','4','3','2','1') NOT NULL,

`Teacher is good at stimulating the interest in the course conten` enum('5','4','3','2','1') NOT NULL,

`Teacher is good at explaining the subject matter` enum('5','4','3','2','1') NOT NULL,

`Teacher's presentation was clear,loud ad easy to understand` enum('5','4','3','2','1') NOT NULL,

`Teacher is good at using innovative teaching methods/ways` enum('5','4','3','2','1') NOT NULL,

`Teacher is available and helpful during counseling hours` enum('5','4','3','2','1') NOT NULL,

`Teacher has competed the whole course as per course outline` enum('5','4','3','2','1') NOT NULL,

`Teacher was always fair and impartial:` enum('5','4','3','2','1') NOT NULL,

`Assessments conducted are clearly connected to maximize learinin` enum('5','4','3','2','1') NOT NULL,

`What I liked about the course` text NOT NULL,

`Why I disliked about the course` text NOT NULL,

`date` date NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=19 ;

--

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

--

INSERT INTO `feedback` (`id`, `student\_id`, `faculty\_id`, `Teacher provided the course outline having weekly content plan w`, `Course objectives,learning outcomes and grading criteria are cle`, `Course integrates throretical course concepts with the real worl`, `Teacher is punctual,arrives on time and leaves on time`, `Teacher is good at stimulating the interest in the course conten`, `Teacher is good at explaining the subject matter`, `Teacher's presentation was clear,loud ad easy to understand`, `Teacher is good at using innovative teaching methods/ways`, `Teacher is available and helpful during counseling hours`, `Teacher has competed the whole course as per course outline`, `Teacher was always fair and impartial:`, `Assessments conducted are clearly connected to maximize learinin`, `What I liked about the course`, `Why I disliked about the course`, `date`) VALUES

(16, 'abc@gmail.com', 'arju@gmail.com', '5', '5', '5', '5', '5', '5', '5', '5', '5', '5', '5', '5', '\r\nddddddddddddd', 'aa', '2016-07-15'),

(17, 'sangmail.com', 'arju@gmail.com', '5', '3', '1', '5', '5', '3', '3', '3', '3', '5', '5', '5', '\r\n', '\r\n', '2016-07-15'),

(18, 'war@yahoo.com', 'arju@gmail.com', '5', '5', '5', '2', '1', '5', '5', '4', '5', '5', '5', '5', '\r\ndfdfdfdfdfd', '\r\n', '2016-07-17');

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

--

-- **Table structure for table `files`**

--

CREATE TABLE IF NOT EXISTS `files` (

`notice\_id` int(11) NOT NULL AUTO\_INCREMENT,

`attachment` varchar(255) NOT NULL,

`subject` varchar(100) NOT NULL,

`Description` text NOT NULL,

`Date` datetime NOT NULL,

PRIMARY KEY (`notice\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=9 ;

--

-- **Dumping data for table `files`**

--

INSERT INTO `files` (`notice\_id`, `attachment`, `subject`, `Description`, `Date`) VALUES

(8, 'AteekCV\_java (1).docx', 'aaaaa', 'dfdfdfd', '2016-07-03 12:39:35');

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

--

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

--

CREATE TABLE IF NOT EXISTS `user` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` char(40) NOT NULL,

`email` varchar(40) NOT NULL,

`pass` varchar(40) NOT NULL,

`mobile` bigint(11) NOT NULL,

`programme` varchar(20) NOT NULL,

`semester` varchar(10) NOT NULL,

`gender` varchar(40) NOT NULL,

`hobbies` varchar(40) NOT NULL,

`image` varchar(50) NOT NULL,

`dob` date NOT NULL,

`regid` varchar(20) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `email` (`email`),

FULLTEXT KEY `name` (`name`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO\_INCREMENT=15 ;

--

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

--

INSERT INTO `user` (`id`, `name`, `email`, `pass`, `mobile`, `programme`, `semester`, `gender`, `hobbies`, `image`, `dob`, `regid`) VALUES

(10, 'sanj', 'sanj2@gmail.com', '98d34c1758b15b5a359b69c2b08c5767', 9015501897, 'B.tech', '3rd', 'm', 'reading,playing', 'Jellyfish.jpg', '1961-09-15', '2147483647'),

(12, 'ravi', 'rav@gmail.com', '63dd3e154ca6d948fc380fa576343ba6', 9015501897, 'M.Tech', 'ii', 'm', 'reading', 'Desert.jpg', '1965-10-15', '2016-07-13 15:52:01'),

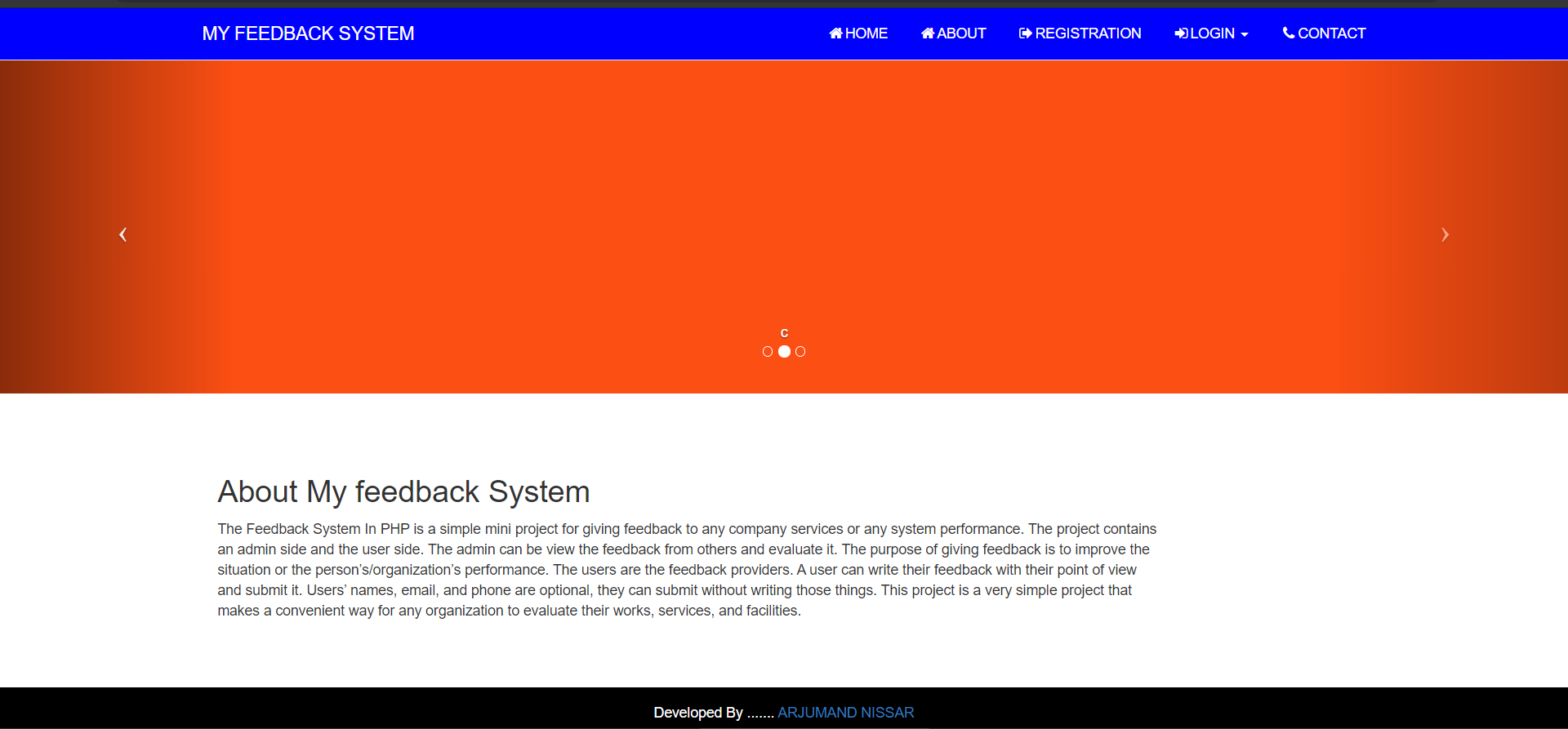
(13, 'warda', 'warda@yahoo.com', '827ccb0eea8a706c4c34a16891f84e7b', 32457895212, 'BCA', 'ii', 'f', 'reading', 'Koala - Copy.jpg', '1965-10-06', '2016-07-17 15:39:08'),

(14, 'test', 'test@gmail.com', '098f6bcd4621d373cade4e832627b4f6', 989898989, 'MCA', 'i', 'm', 'reading,singin', 'Chrysanthemum.jpg', '1963-08-12', '2017-02-10 16:04:10');

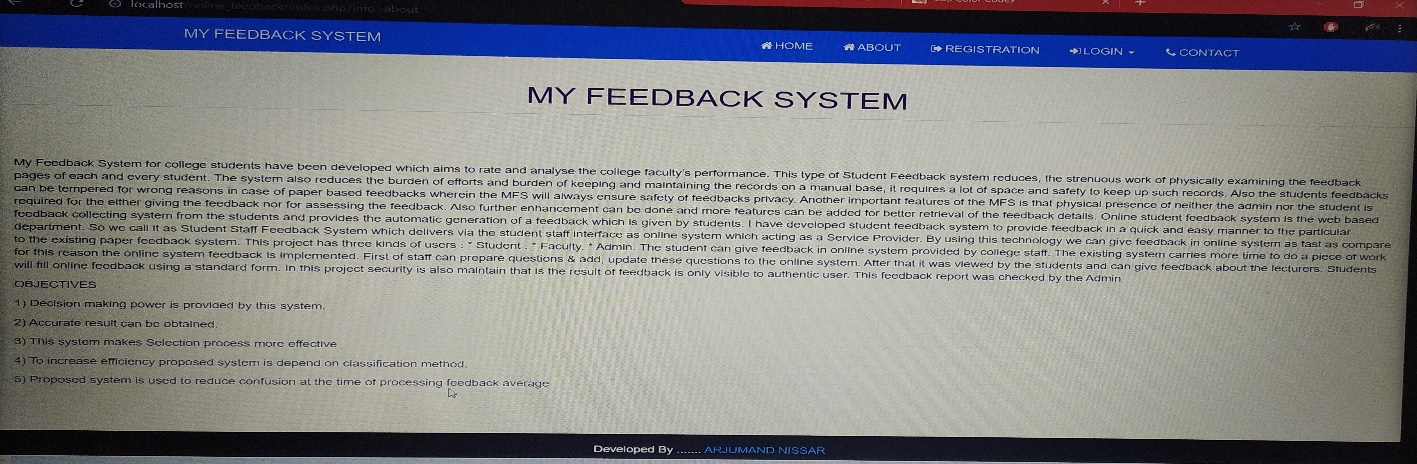
**PROJECT LAYOUT**

**9.1 SCREENSHOTS**

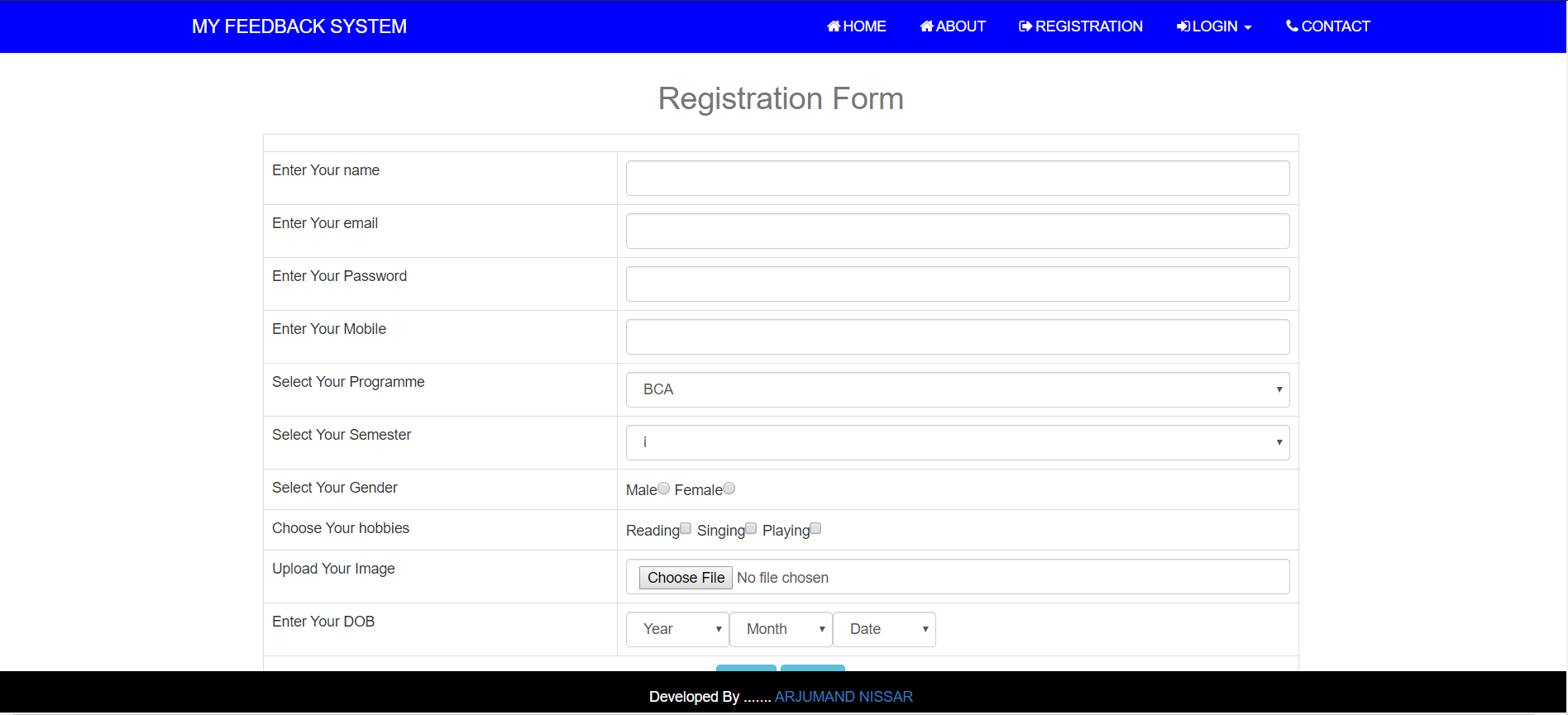
* 1. **HOME**



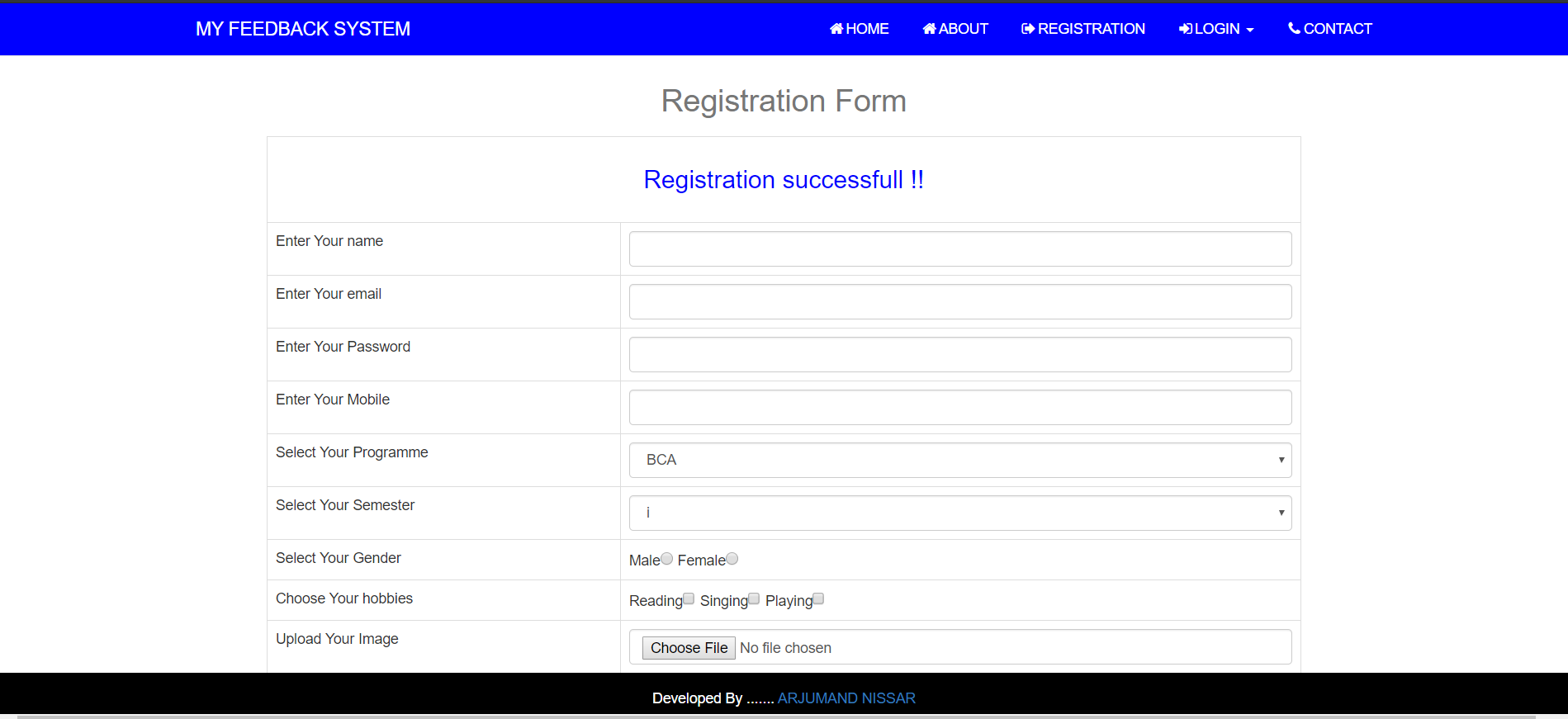
* 1. **ABOUT**



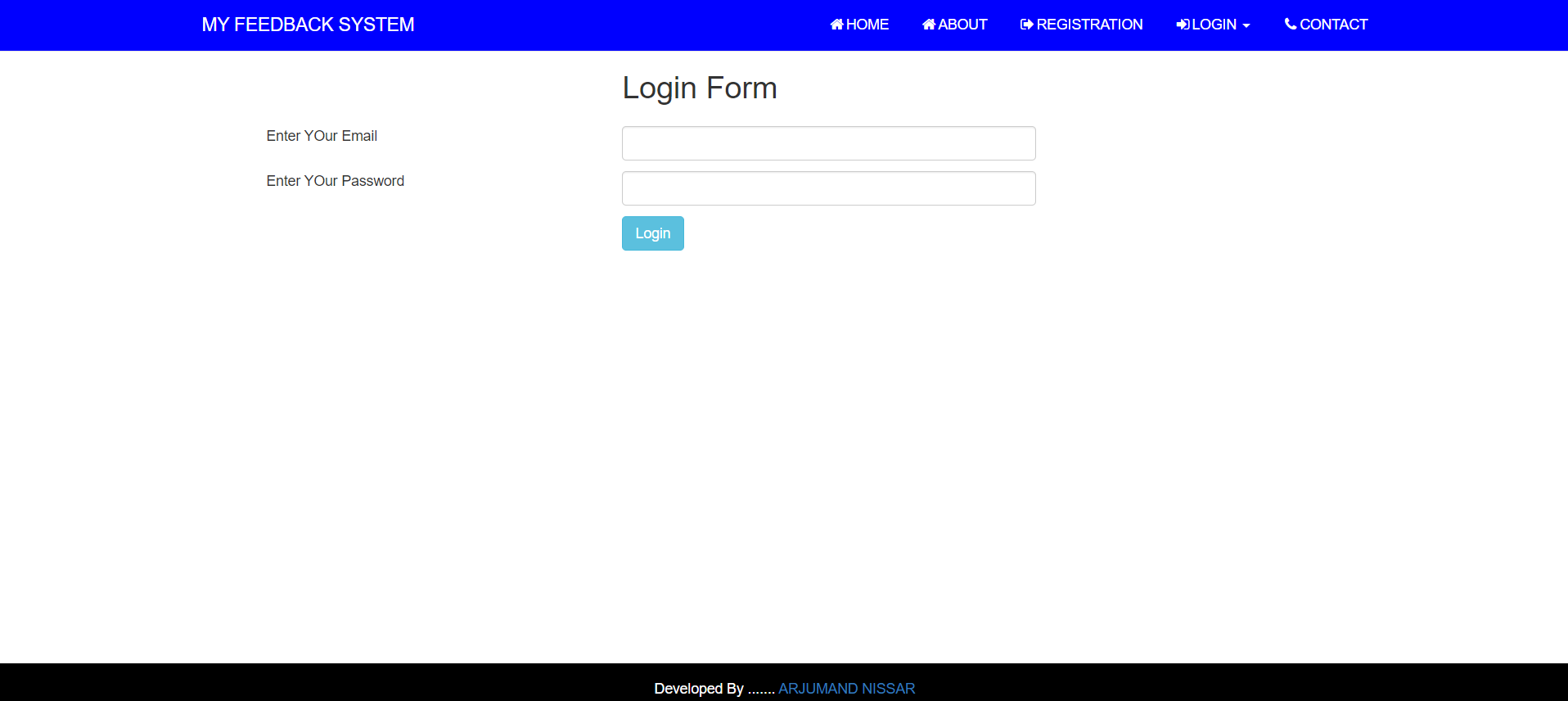
* 1. **REGISTRATION**



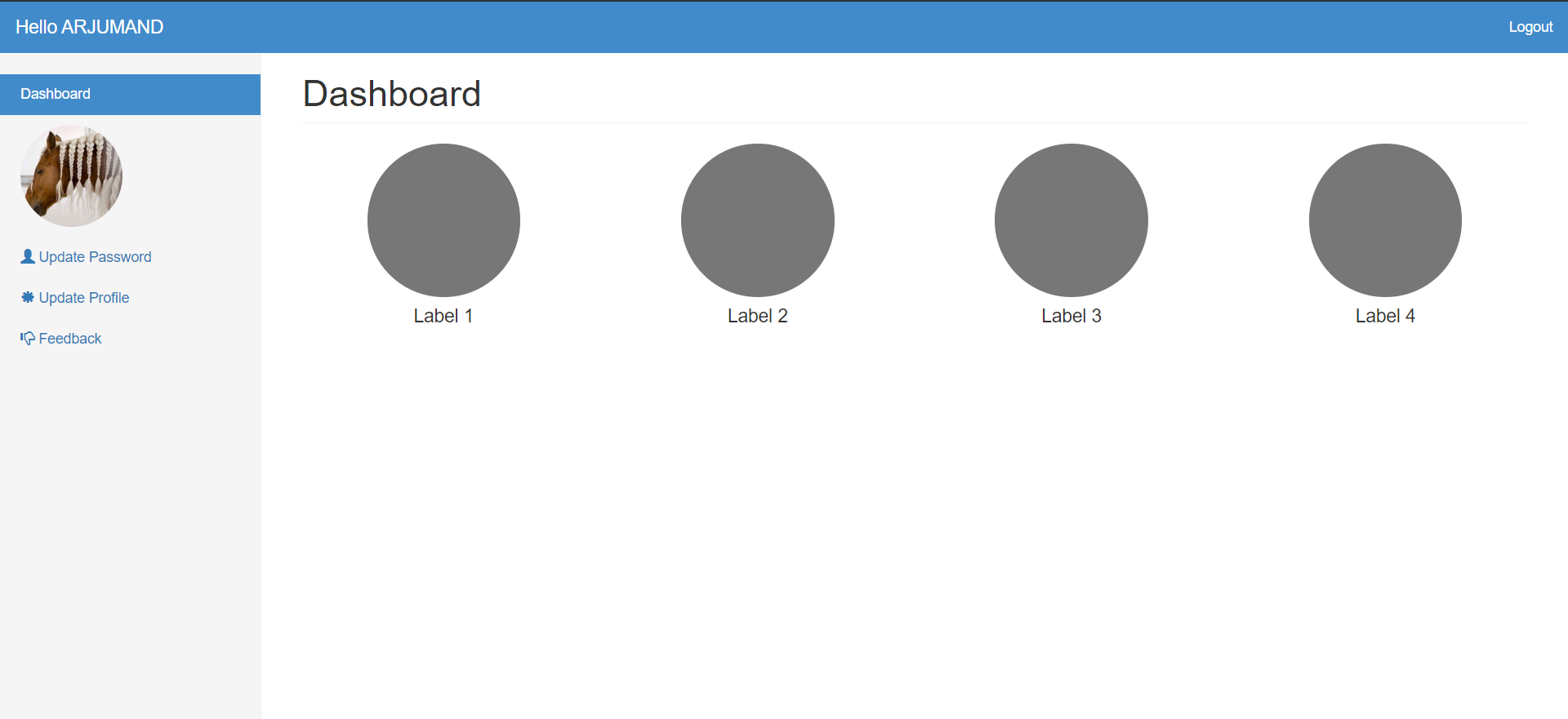
* 1. **AFTER** **REGISTRATION**



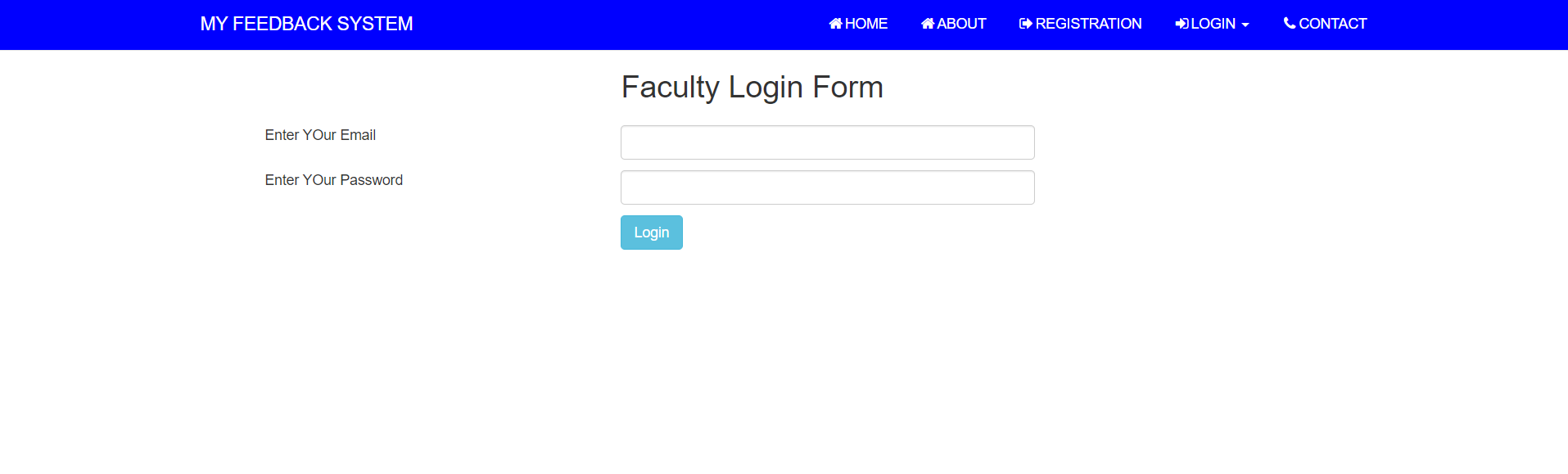
* 1. **STUDENT LOGIN PAGE**



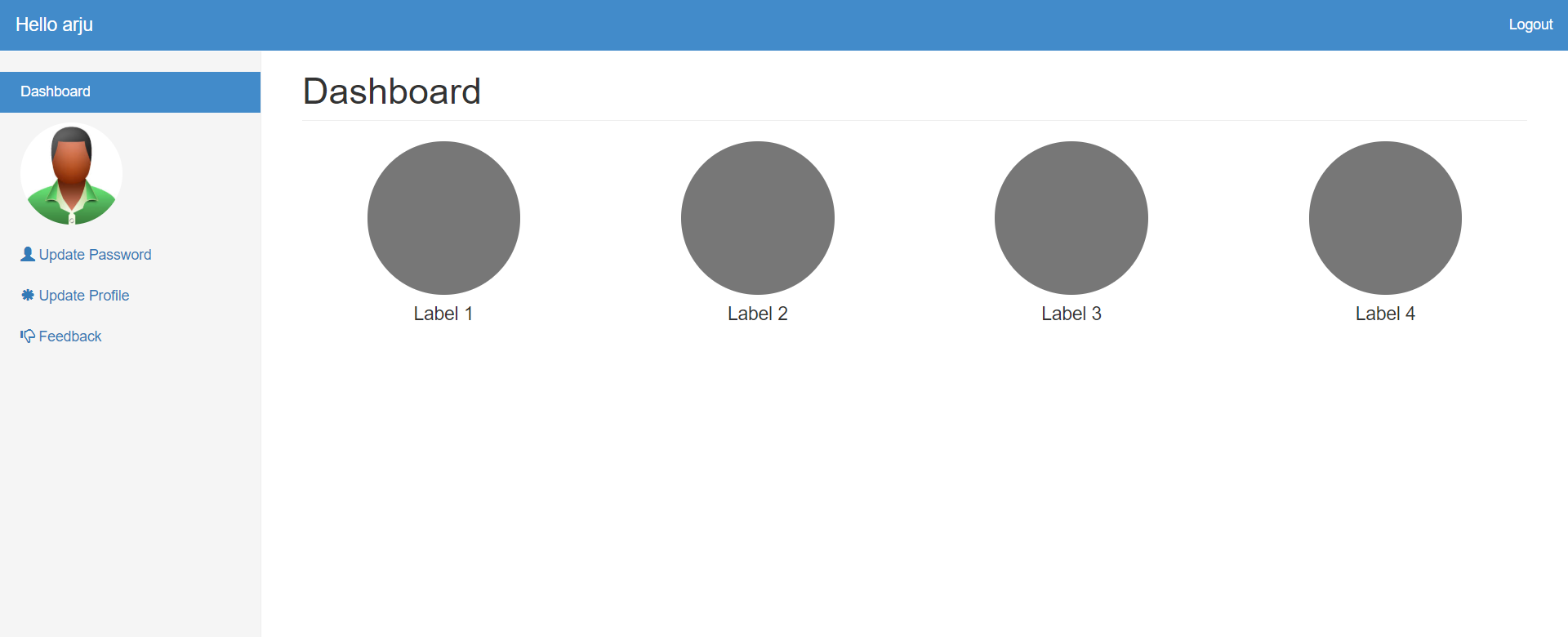
* 1. **AFTER LOGIN**



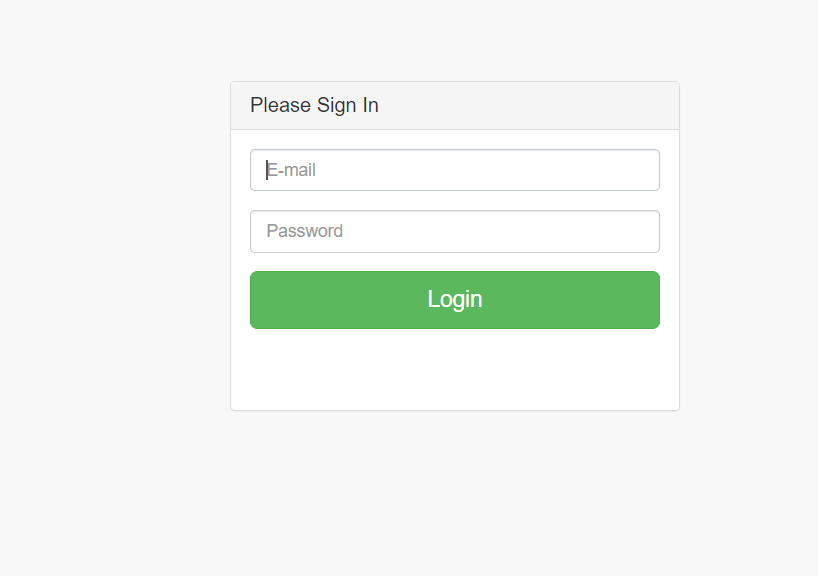
* 1. **FACULTY LOGIN PAGE**



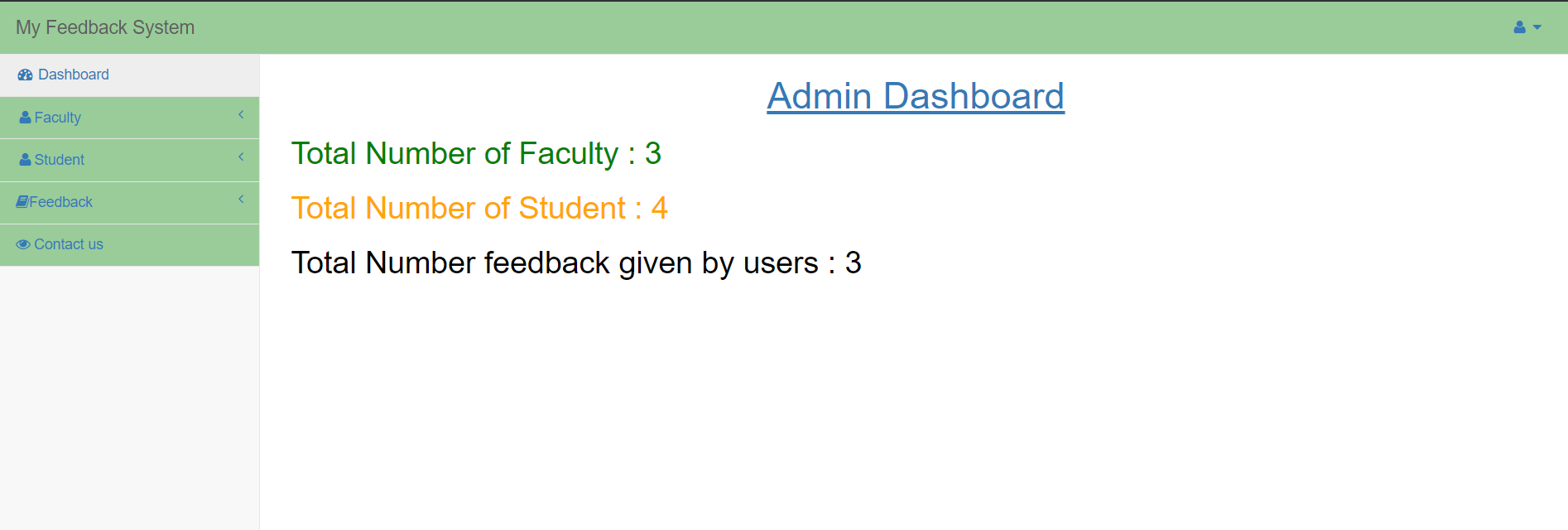
* 1. **AFTER LOGIN**



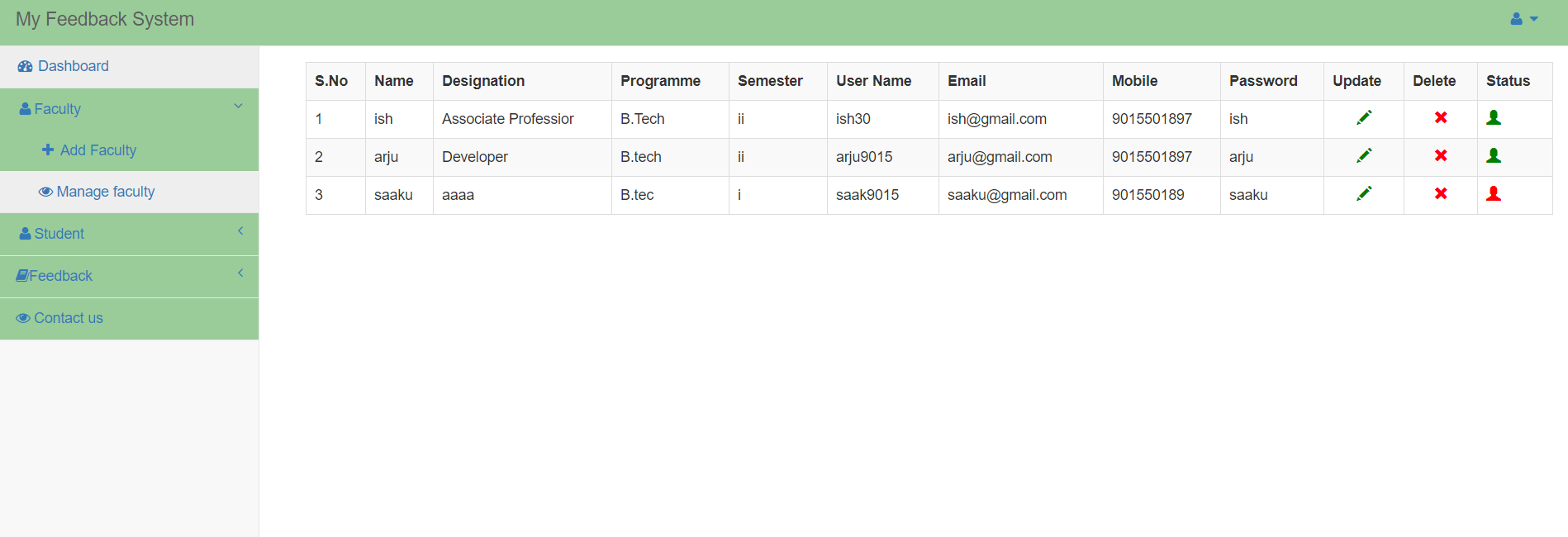
* 1. **ADMIN LOGIN PAGE**



* 1. **AFTER LOGIN**



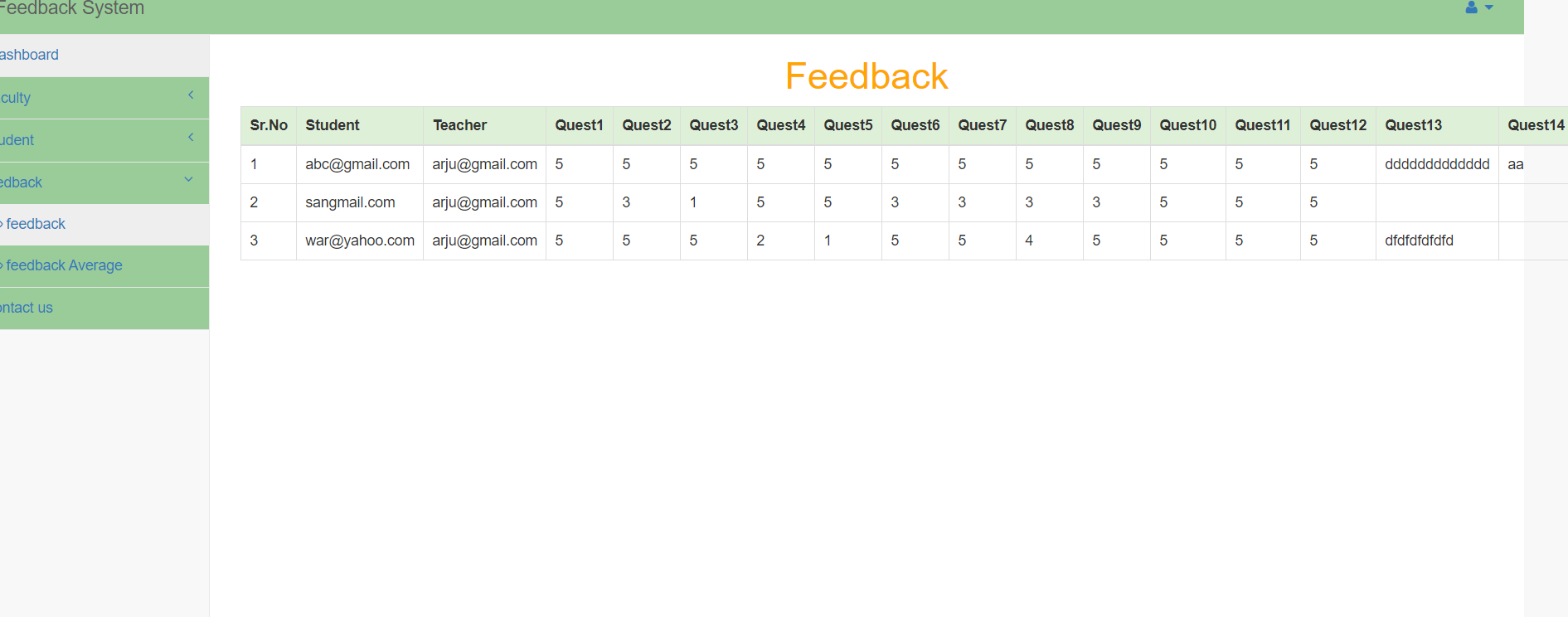
* 1. **MANAGE FACULTY**



**xi) MANAGE STUDENT**



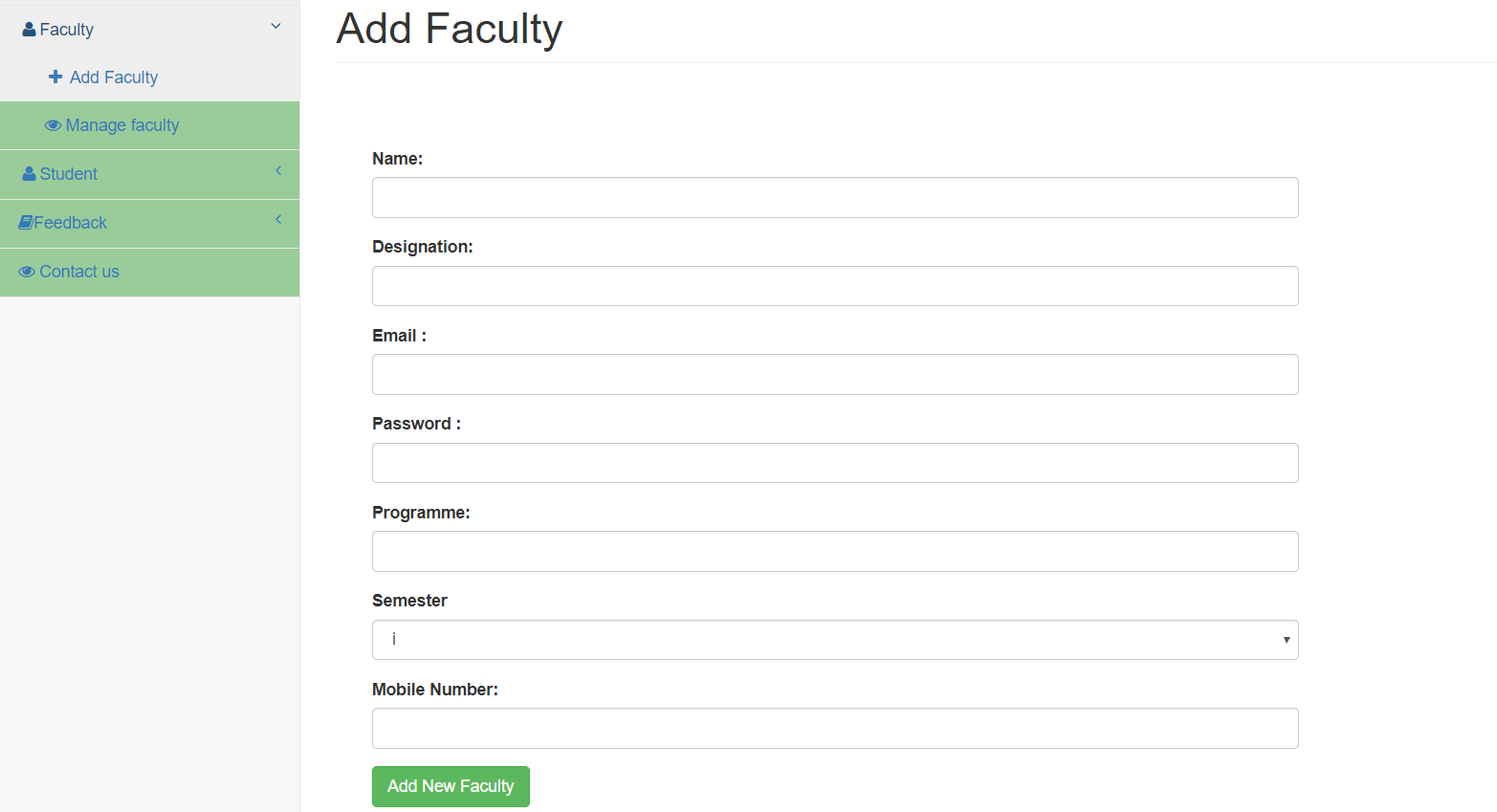
**xii) FEEDBACK**



**xii) CONTACT US**



**xiii) ADD FACULTY**



**CONCLUSIONS AND ENHANCEMENTS**

**Conclusion:**

The Project “MY FEEDBACK SYSTEM” is designed in order reduce the burden of maintaining bulk of records of all the students feedback details of who study in an Educational Institution. Inserting, retrieving and updating the feedback details of a student are easy when it is compared to the manual feedback and storing. Maintaining the project is also easy which can is easily understandable. Maintaining the details in the database is manageable.

By using this system we can give feedback in online system as fast as compare to the existing paper feedback system.

In this project security is also maintained that is the result of feedback is only visible to authentic user.

**Future Enhancements**:

Due to the lack of time, the design part is not done so attractive. Further enhancements can be made in designing the screens. Some more forms can also be added so as to better retrieve the feedback details. Various other options can also be added for the better usability of project.

**REFERENCES**

* 1. [www.w3schools.com/php/php\_mysql\_intro.asp](http://www.w3schools.com/php/php_mysql_intro.asp)
  2. DATABASE MANAGEMENT SYSTEMS” Raghu Rama Krishnan Gehrke, McGraw-Hill Publications 3rd edition.
  3. PHP PROGRAMMING “Vikram Vaswani”
  4. PHP reference manual obtained from [www.php.net](http://www.php.net)
  5. HTML and PHP reference from www.w2schools.com
  6. <https://getbootstrap.com/docs/4.2/components/carousel/>
  7. https://www.geeksforgeeks.org/