**CHAPTER 1**

**INTRODUCTION**

1. **INTRODUCTION**

As the name specifies “Faculty Management System” is an application developed for maintaining the faculty records. Faculty Management System can be used by education institutes to maintain the records of Faculties easily. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solvedusing this project. And in this project we have provided different functionalities for the faculties like adding publications, workshop details, viewing their publications, workshops and also faculties can apply leaves.

* 1. **PROBLEM DESCRIPTION**
     1. **Problem statement**

Faculty Management System is software which is helpful for Faculties as well as the college authorities. In the current system all the activities are done manually. It is very time consuming and costly. Our Faculty Management System deals with the various activities related to the Faculties.

There are mainly 2 modules in this software

* Admin module
* Faculty Module

In the Software we can register new Faculties. Administrator has the power to add new Faculty and can view and delete a Faculty. A Faculty has different functionalities like adding and viewing publications and workshop details. And faculty has the power to change his password.

* + 1. **Objective and Scope of the Project**

The objective of **Faculty Management System** is to allow the administrator of any organization to add and find out the personal details of a Faculty and allows the Faculty to keep up to date his profile .It’ll also facilitate keeping all the records of Faculties, such as their id, name, mailing address, phone number, DOBetc…So all the information about the Faculty will be available in a few seconds. Overall, it’ll make Faculty Management System an easier job for the administrator and the organization.

**CHAPTER 2**

**LITERATURE SURVEY**

The system has come up with much functionality for educational institutions to track the faculty progress and managing the faculty details.The first step in system development life cycle is the identification of need of change to improve or enhance an existing system. An initial investigation on existing system was carried out. The present system of faculty is completely manual. Many problems were identified during the initial study of the existing system. The Existing system of “Faculty management system” is to be computerized in order to overcome the problems which affect the existing system. Computerizing the existing system with the help of web programming language.

The Paper basically provides the platform for the faculty so that they can have ease of their work as well as reducing the paper work will help to save the environment. This application stated also provides standalone application platform to provide the various notices for the faculties.

This application provides a huge help to both each and every field. What happen exactly in the earlier system uses the man to man working strategy as well as the notices or the information needs to be use of paper. This leads to didn’t ensure that the information will exactly provide to each person in the system.

Making the efficient login for faculties in college. Different login can provide a various privileges to everyone. From the above literature survey we can easily concluded that the application is very useful for the faculties. It reduces a lot of paper work and a huge multitasking making use of single application. This application make a very useful development in every field is used effectively.

1. **EXISTING SYSTEM**

The “Faculty management system” of the organization is build an application program to reduce the manual work for managing the faculty. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming.

This application provides a huge help to both each and every field. What happen exactly in the earlier system uses the man to man working strategy as well as the notices or the information needs to be use of paper. This leads to didn’t ensure that the information will exactly provide to each person in the system.

1. **PROPOSED SYSTEM**

* Administrator has the power to add, view and remove the faculty.
* Maintaining the Data of Faculty details.
* Faculty can add publications and view it.
* Faculty can add
* conducted and attended workshops details and can view it.
* Faculty can change his password.
* Recommended minimum software

**CHAPTER 3**

**SYSTEM REQUIREMENTS**

* + 1. **HARDWARE REQUIREMENT**
* Processor: Intel Core I3 processor and above
* RAM: 4GB Ram and above
* Hard disk: 40GB and above
  + 1. **SOFTWARE REQUIREMENTS**
    - Operating System:windows7 and later version
    - Browsers: Mozilla,opera,chrome etc.
    - Frontend: HTML,CSS, Bootstrap and J-Query.
    - Scripting: Java script, PHP
    - Tools used: Notepad++
    - Server: wamp server
    - Database : MySQL

**CHAPTER 4**

**SYSTEM REQUIREMENT SPECIFICATION**

1. **INTRODUCTION**

A requirement is a feature or a constraint that the system must satisfy to be accepted by the client. This software design document provides a complete description of Faculty Management System, as a final design project of software, including all issues. This document will guarantee that the design will correctly implement all the functionalities identified in the SRS document; it will be easily understandable, efficient and suitable to change.

1. **PURPOSE**

To design and develop this application which provides flexibility of maintaining faculty information properly. This document aims to represent a software system that is used as a medium for communicating software design information, namely for “FUNCTIONALITIES OF FACULTIES” application that implements data analysis. It will give overall information about the desired project and necessary steps in order to accomplish the task, such as system overview, design considerations.

1. **SCOPE**

The project which is going to be presented in this document is called “FACULTY MANAGEMENT SYSTEM”. This application is planned to be used by person belonging to particular institution who join as a faculty.

1. **SPECIFIC REQUIREMENTS**
   * + 1. **FUNCTIONAL REQUIREMENTS**

The functional requirement of the system describes the functionality that the system should provide. It is organized by the process and features encapsulated in the software application.

1. **ADMINISTRATION LOGIN**

Administrator logs in to the system with his/her credentials then application has to validate for authentication.

* Input: Login with his/her username and password.
* Process: Validate and authenticate username and password. If valid he/she proceeds further otherwise, error message has to be displayed.
* Output: Administrator logged in successfully.
* Error handling: If valid username and password not entered then alert message will display as invalid username or password.

1. **FACULTY REGISTRATION**

In this module contains all the faculty information

* Input : faculty details like id and password
* Process: information stored in database.
* Output: it generatesfaculty id and password.
* Error handling:If the faculty details are not filled properly appropriate alert message will display for each field.

1. **FACULTY LOGIN**

Faculty logs in to the system with his/her credentials then application has to validate for authentication.

* Input: Login with his/her id and password.
* Process: Validate and authenticate id and password. If valid he/she proceeds further otherwise, error message has to be displayed.
* Output: Faculty logged in successfully.
* Error handling: If valid username and password not entered then alert message will display as invalid username or password.

1. **WORKSHOP**

This functionality has a detail of workshop conducted and attended by the faculty.

* Input: Date,title, venue, number of days, resource person name.
* Process: Authenticates and process.
* Output: Information successful.
* Error handling: If all the details are not filled properly appropriate alert message will be displayed.

1. **ADD PUBLICATION**

This functionality has a list of course.

* Input: Article titles, publication id, date of publish, Description.
* Process: add and view the information of publication. .
* Output: Information successful.
* Error handling:If all the details of publications are not filled properlyappropriate alert message will display for each field.
  + - 1. **NON-FUNCTIONAL REQUIREMENTS**

Non Functional requirements define the needs in terms of performance, logical database requirements, design constraints, reliability, availability, security, maintainability, and portability.

* + - 1. **performance requirements**

The software has to be highly interactive and time efficient so that the information can be processed with the minimum number of clicks. The server to process the request and response faster without causing much overload on the server. There will be one time DB connection. So it will save time.

* + - 1. **Safety and Security Requirements**

This software requires User authentication. That is, the user needs to provide valid credentials, without which the user will not be able to login into the application.

* + - 1. **Software Quality Attributes**
* **Availability:** The Faculty Management system must be available 24X7 for the authorized user to perform all functionalities only if network is available.
* **Maintainability:** The Faculty Management system should be built with modularity approach which ensures the flexibility of software so that it should be able to extend with new modules or functionalities without affecting existing system.
* **Portability:** The Faculty Management system should be designed and developed in such a way that it should be able to run on any platform or any devices, because it is a web application that we can access on any platform.

**CHAPTER 5**

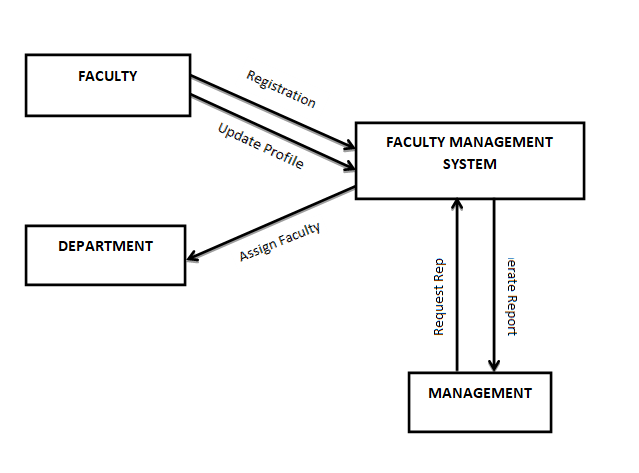
**SYSTEM DESIGN**

Systems design is the process of defining the [architecture](https://en.wikipedia.org/wiki/Systems_architecture), modules, interfaces, and [data](https://en.wikipedia.org/wiki/Data) for a [system](https://en.wikipedia.org/wiki/System) to satisfy specified [requirements](https://en.wikipedia.org/wiki/Requirement). Systems design could be seen as the application of [systems theory](https://en.wikipedia.org/wiki/Systems_theory) to [product development](https://en.wikipedia.org/wiki/Product_development)**.**

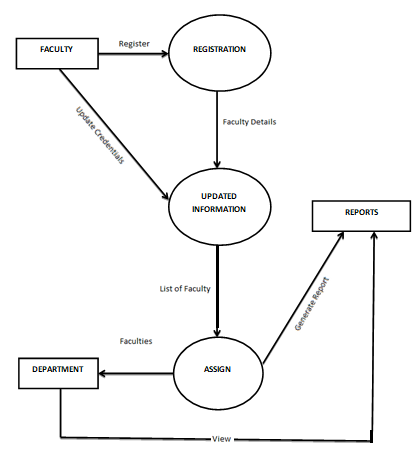
In our project there are two modules in the system design:

* **ADMIN MODULE:** Administrator log in to the system with his/her credentials then application has to validate for authentication. Admin will add the new faculty information, remove the faculty and view all the details of faculties.
* **FACULTY MODULE:** Faculty module contains all the faculty information. Faculty can add and view the publication, workshop attended and conducted by the faculty and faculty can apply the leave.

1. **DATA FLOW DIAGRAM**



**Figure 5.1.1: Context level DFD**

****

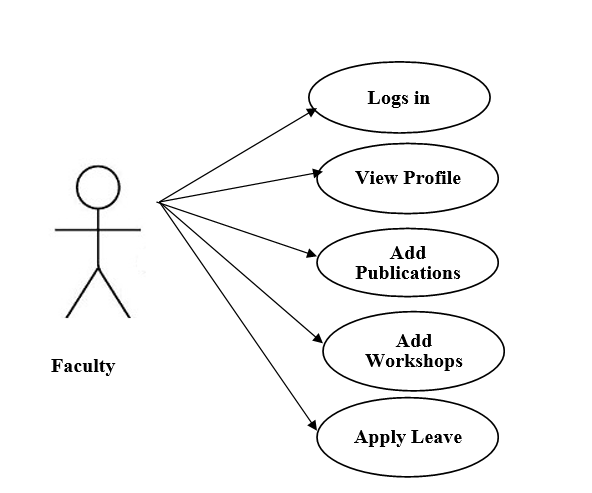
**Figure5.1.2: Detailed level DFD**

# C:\Users\Priya\AppData\Local\Packages\Microsoft.Office.Desktop_8wekyb3d8bbwe\AC\INetCache\Content.Word\usecase1.pngUSE CASE DIAGRAM

**Figure 5.2.1**Use Case Diagram for Admin

**Module**: Admin Management Module.

**Actors**: Administrator.

**Use case:** Add and remove faculty.

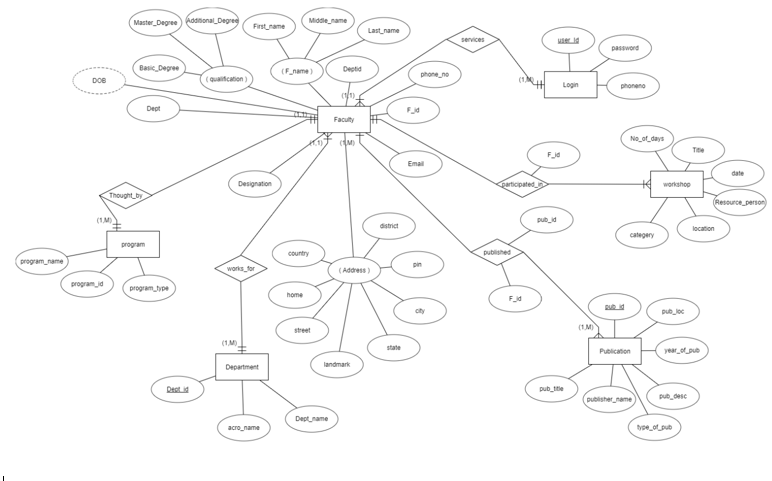
**Figure 5.2.2**Use Case Diagram for Faculty

**Module**: Faculty Management Module

**Actors**: Faculty.

**Use cases**: Add, view publication and maintains the record of workshop attended and conducted.

# ER DIAGRAM



**Figure 5.3.1**ER Diagram for faculty management system

1. **ARCHITECTURAL DIAGRAM**

**FACULTY**

**ADMIN**

**Presentation Tier**

**GUI**

**WORKSHOP CONDUCTED AND ATTENDED**

**VIEW PUBLICATION**

**ADD PUBLICATION**

**FACULTY LOGIN**

**REMOVE FACULTY**

**VIEW FACULTY**

**ADD FACULTY**

**ADMIN LOGIN**

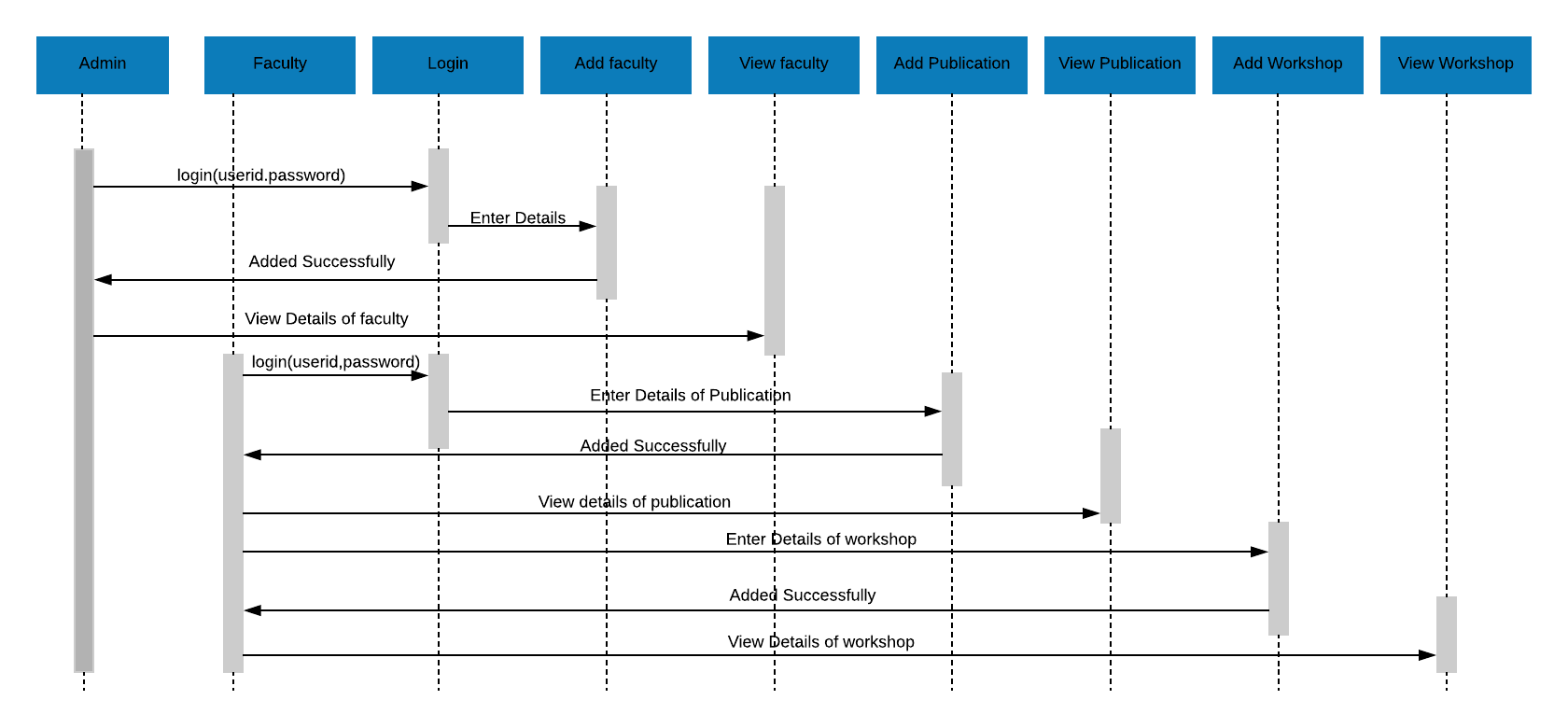
**Logic Tier**

**Data Tier**

**DATABASE**

**Figure 5.4.1**Architectural Diagram for faculty management system

1. **SEQUENCE DIAGRAM**

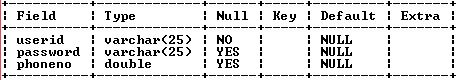
****

**Figure5.5.1:sequence diagram of Faculty Management System**

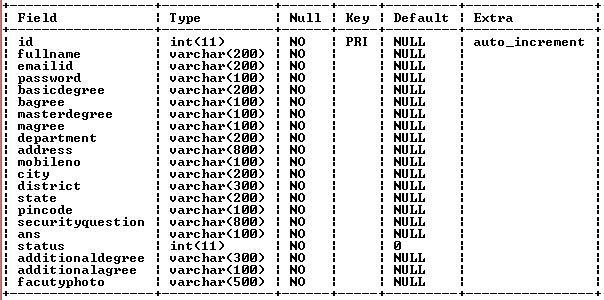
1. **DATABASE DESIGN:**

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the database structures used to store the data. In the relational model, these are the tables and views.In an object database are entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the database structures, but also the forms and queries used as part of the overall database application within the database management system(DBMS).

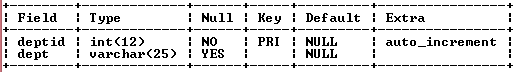
**LOGIN TABLE:**

****

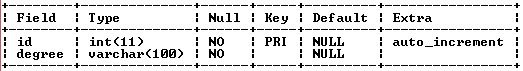
**FACULTY TABLE:**

****

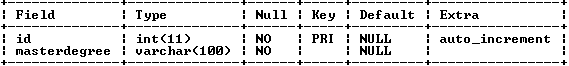
**DEPARTMENT TABLE:**

****

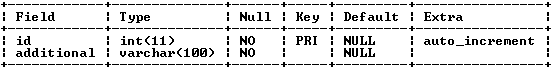
**DEGREE TABLE:**

****

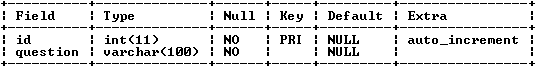
**MASTER DEGREE TABLE:**

****

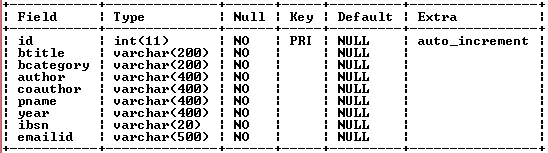
**ADDITIONAL DEGREE TABLE:**

****

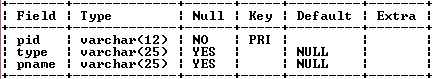
**SECURITY QUESTION TABLE:**

****

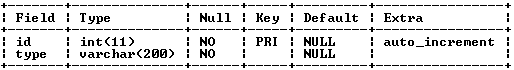
**BOOKPUBLICATION:**

****

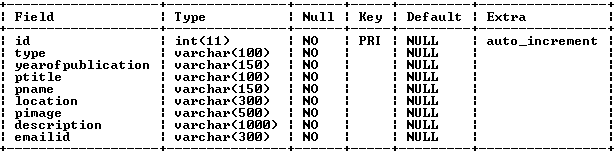
**PROGRAM:**

****

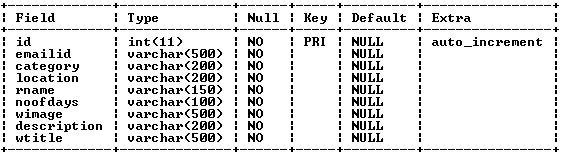
**PUBLICATION TYPE:**

****

**PUBLICATION:**

****

**WORKSHOP:**

****

**CHAPTER 6**

**IMPLEMENTATION**

1. **INTRODUCTION**

Implementation is the stage where the theoretical design is turned into a working system. Once the design is complete, most of the major decisions about the system have been made. The goal of the coding phase is to translate the design of the system into code in a given programming language. For a given design, the aim is to implement the design in the best possible manner.

The coding phase affects both testing and maintenance profoundly. Since the testing and maintenance costs of software are much higher than the coding cost, the goal of the coding should be to reduce the testing and maintenance effort. Hence, during coding, the focus should be on developing the programs that are easy to read and understand, neither simply on developing the program that are easy to write.

**The implementation stage consists of:**

* Making the necessary changes for the system as desired by the user.
* Detection and Correction of errors.
* Testing the developed program with the simple data.

1. **THE TECHNOLOGIES USED FOR IMPLEMENTATION**

* **CSS/CSS3:** CSS is cascading style sheet which is used to give designer look to HTML5 using the external file.
* **JavaScript:** Java script is used for client side scripting which can help in using validation on the website and many other functions.
* **JQuery:** It is a lightweight, "write less, and do more". It also simplifies a lot of the complicated things from JavaScript.
* **WampServer:**WAMP also includes [MySQL](https://techterms.com/definition/mysql) and [PHP](https://techterms.com/definition/php), which are two of the most common technologies used for creating [dynamicwebsites](https://techterms.com/definition/dynamicwebsite). MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server.

While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

* **PHP:**HypertextPreprocessor (or simply PHP) is a [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting) language designed for [Web development](https://en.wikipedia.org/wiki/Web_development), but also used as a [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). It was originally created by [RasmusLerdorf](https://en.wikipedia.org/wiki/Rasmus_Lerdorf) in 1994.
* PHP code may be embedded into [HTML](https://en.wikipedia.org/wiki/HTML) code, or it can be used in combination with various [web template systems](https://en.wikipedia.org/wiki/Web_template_system), web content management systems, and [web frameworks](https://en.wikipedia.org/wiki/Web_framework). PHP code is usually processed by a PHP [interpreter](https://en.wikipedia.org/wiki/Interpreter_(computing)) implemented as a [module](https://en.wikipedia.org/wiki/Plugin_(computing))in the web server or as a [Common Gateway Interface](https://en.wikipedia.org/wiki/Common_Gateway_Interface) (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a [command-line interface](https://en.wikipedia.org/wiki/Command-line_interface) (CLI) and can be used to implement [standalone](https://en.wikipedia.org/wiki/Computer_software)[graphical applications](https://en.wikipedia.org/wiki/Graphical_user_interface).
* **MYSQL:**"My S-Q-L” is an [open-source](https://en.wikipedia.org/wiki/Open-source)[relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS). Its name is a combination of "My", the name of co-founder [Michael Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)'s daughter,[]](https://en.wikipedia.org/wiki/MySQL#cite_note-7) and "[SQL](https://en.wikipedia.org/wiki/SQL)", the abbreviation for [Structured Query Language](https://en.wikipedia.org/wiki/Structured_Query_Language). The MySQL development project has made its [source code](https://en.wikipedia.org/wiki/Source_code) available under the terms of the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License), as well as under a variety of [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) agreements. MySQL was owned and sponsored by a single [for-profit](https://en.wikipedia.org/wiki/Business) firm, the [Swedish](https://en.wikipedia.org/wiki/Sweden) company [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), now owned by [Oracle Corporation](https://en.wikipedia.org/wiki/Oracle_Corporation).[[8]](https://en.wikipedia.org/wiki/MySQL#cite_note-sunacquire-8) For proprietary use, several paid editions are available, and offer additional functionality.

MySQL is a central component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) open-source web application software stack (and other "[AMP](https://en.wikipedia.org/wiki/List_of_AMP_packages)" stacks). LAMP is an acronym for "[Linux](https://en.wikipedia.org/wiki/Linux), [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), MySQL, [Perl](https://en.wikipedia.org/wiki/Perl)/[PHP](https://en.wikipedia.org/wiki/PHP)/[Python](https://en.wikipedia.org/wiki/Python_(programming_language))". Applications that use the MySQL database include: [TYPO3](https://en.wikipedia.org/wiki/TYPO3), [MODx](https://en.wikipedia.org/wiki/MODx), [Joomla](https://en.wikipedia.org/wiki/Joomla), [WordPress](https://en.wikipedia.org/wiki/WordPress), [Simple Machines Forum](https://en.wikipedia.org/wiki/Simple_Machines_Forum), [phpBB](https://en.wikipedia.org/wiki/PhpBB), [MyBB](https://en.wikipedia.org/wiki/MyBB), and [Drupal](https://en.wikipedia.org/wiki/Drupal). MySQL is also used in many high-profile, large-scale [websites](https://en.wikipedia.org/wiki/Website), including [Google](https://en.wikipedia.org/wiki/Google) (though not for searches), [Facebook](https://en.wikipedia.org/wiki/Facebook), [Twitter](https://en.wikipedia.org/wiki/Twitter), [Flickr](https://en.wikipedia.org/wiki/Flickr), and [YouTube](https://en.wikipedia.org/wiki/YouTube).

* **HTML: HypertextMarkup Language** (**HTML**) is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for creating [web pages](https://en.wikipedia.org/wiki/Web_page) and [web applications](https://en.wikipedia.org/wiki/Web_application). With [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [JavaScript](https://en.wikipedia.org/wiki/JavaScript), it forms a triad of cornerstone technologies for the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web).

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a web page [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](https://en.wikipedia.org/wiki/HTML_element#Images_and_objects) and other objects such as [interactive forms](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. HTML provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, [links](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by *tags*, written using [angle brackets](https://en.wikipedia.org/wiki/Bracket#Angle_brackets).

Tags such as <img/> and <input/> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a [scripting language](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript), which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

1. **BRIEF DESCRIPTION ABOUT THE MODULES:**

* [update**FACULTY:**

The faculty will add the publications and workshops details and also can view it.

The faculty has the power to change the password.

* **ADMINISTRATOR:**

The admin has the power to add, view and remove faculty details.

1. **SOURCE CODE:**

* **home page code:**

<!doctype html>

<!--[if lt IE 7]><html class="no-js lt-ie9 lt-ie8 lt-ie7" lang=""><![endif]-->

<!--[if IE 7]><html class="no-js lt-ie9 lt-ie8" lang=""><![endif]-->

<!--[if IE 8]><html class="no-js lt-ie9" lang=""><![endif]-->

<!--[if gt IE 8]><!--><html class="no-js" lang=""><!--<![endif]-->

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">

<title></title>

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

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

<link rel="apple-touch-icon" href="apple-touch-icon.png">

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

<!--<link rel="stylesheet" href="assets/css/bootstrap-theme.min.css">-->

<!--For Plugins external css-->

<link rel="stylesheet" href="assets/css/plugins.css" />

<link rel="stylesheet" href="assets/css/raleway-webfont.css" />

<!--Theme custom css -->

<link rel="stylesheet" href="assets/css/style.css">

<!--Theme Responsive css-->

<link rel="stylesheet" href="assets/css/responsive.css" />

<script src="assets/js/vendor/modernizr-2.8.3-respond-1.4.2.min.js"></script>

<style>

body

{

min-width: 1000px;

height: 560px;

margin: 0;

}

\* {box-sizing: border-box;}

body {font-family: Verdana, sans-serif;}

.mySlides {display: none;}

img {vertical-align: middle;}

/\* Slideshow container \*/

.slideshow-container {

max-width: 1000px;

position: relative;

margin: auto;

}

/\* Caption text \*/

.text {

color: #f2f2f2;

font-size: 15px;

padding: 8px 12px;

position: absolute;

bottom: 8px;

width: 100%;

text-align: center;

}

/\* Number text (1/3 etc) \*/

.numbertext {

color: #f2f2f2;

font-size: 12px;

padding: 8px 12px;

position: absolute;

top: 0;

}

/\* The dots/bullets/indicators \*/

.dot {

height: 15px;

width: 15px;

margin: 0 2px;

background-color: #bbb;

border-radius: 50%;

display: inline-block;

transition: background-color 0.6s ease;

}

.active {

background-color: #717171;

}

/\* Fading animation \*/

.fade {

-webkit-animation-name: fade;

-webkit-animation-duration: 1.5s;

animation-name: fade;

animation-duration: 1.5s;

}

@-webkit-keyframes fade {

from {opacity: .4}

to {opacity: 1}

}

@keyframes fade {

from {opacity: .4}

to {opacity: 1}

}

/\* On smaller screens, decrease text size \*/

@media only screen and (max-width: 300px) {

.text {font-size: 11px}

}

</style>

</head>

<body>

<!--[if lt IE 8]>

<p class="browserupgrade">You are using an <strong>outdated</strong> browser. Please <a href="http://browsehappy.com/">upgrade your browser</a> to improve your experience.</p>

<![endif]-->

<div class='preloader'><div class='loaded'>&nbsp;</div></div>

<nav class="navbarnavbar-default navbar-fixed-top">

<div class="container">

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

<div class="navbar-header">

<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">

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

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

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

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

</button>

<h3 style="float:left;color:white;">Faculty Management System</h3>

</div>

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

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

<ul class="navnavbar-navnavbar-right">

<li class="active"><a href="index.html">Home</a></li>

<li><a href="about\_us.html">About</a></li>

<li><a href="adminlogin.php">Admin Login</a></li>

<li><a href="facultylogin.php">Faculty Login</a></li>

</ul>

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

</div><!-- /.container-fluid -->

</nav>

<!--Home page style-->

<!--slide-->

<body style="background-color:#DCDCDC;margin-top:95px;">

<div class="slideshow-container">

<div class="mySlides fade">

<div class="text" style="background-color:black;">WELCOME TO KLE TECHNOLOGICAL UNIVERCITY</div>

<div class="numbertext"></div>

<imgsrc="FMS/new2.jpg" style="height:500px;width:100%">

</div>

<div class="mySlides fade">

<div class="numbertext"></div>

<imgsrc="FMS/new5.jpg" style="height:500px;width:100%">

<div class="text" style="background-color:black;">WELCOME TO KLE TECHNOLOGICAL UNIVERCITY</div>

</div>

<div class="mySlides fade">

<div class="text">WELCOME TO KLE TECHNOLOGICAL UNIVERCITY</div>

<div class="numbertext"></div>

<imgsrc="FMS/new6.jpg" style="height:500px;width:100%">

<div class="text" style="background-color:black;">WELCOME TO KLE TECHNOLOGICAL UNIVERCITY</div>

</div>

</div>

<br>

<div style="text-align:center">

<span class="dot"></span>

<span class="dot"></span>

<span class="dot"></span>

</div>

<!-- Sections --><script>

varslideIndex = 0;

showSlides();

functionshowSlides() {

vari;

var slides = document.getElementsByClassName("mySlides");

var dots = document.getElementsByClassName("dot");

for (i = 0; i<slides.length; i++) {

slides[i].style.display = "none";

}

slideIndex++;

if (slideIndex>slides.length) {slideIndex = 1}

for (i = 0; i<dots.length; i++) {

dots[i].className = dots[i].className.replace(" active", "");

}

slides[slideIndex-1].style.display = "block";

dots[slideIndex-1].className += " active";

setTimeout(showSlides, 2000); // Change image every 2 seconds

}

</script><scriptsrc="assets/js/vendor/jquery-1.11.2.min.js"></script>

<scriptsrc="assets/js/vendor/bootstrap.min.js"></script>

<scriptsrc="assets/js/jquery.easypiechart.min.js"></script>

<scriptsrc="assets/js/plugins.js"></script>

<scriptsrc="assets/js/modernizr.js"></script><scriptsrc="assets/js/main.js"></script>

</body></html>

* **Addfaculty code:**

<!DOCTYPE html>

<!--[if lt IE 7 ]> <html lang="en" class="ie6"> <![endif]-->

<!--[if IE 7 ]> <html lang="en" class="ie7"> <![endif]-->

<!--[if IE 8 ]> <html lang="en" class="ie8"> <![endif]-->

<!--[if IE 9 ]> <html lang="en" class="ie9"> <![endif]-->

<!--[if (gt IE 9)|!(IE)]><!-->

<html lang="en">

<!--<![endif]-->

<head>

<link rel="stylesheet" href="http://necolas.github.io/normalize.css/2.1.3/normalize.css">

<link rel="stylesheet" href="css/jquery.idealforms.css">

<meta charset=utf-8 />

<title>jQuery Ideal Forms 3 Example</title>

<link rel="apple-touch-icon" href="apple-touch-icon.png">

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

<!--<link rel="stylesheet" href="assets/css/bootstrap-theme.min.css">-->

<!--For Plugins external css-->

<link rel="stylesheet" href="assets/css/plugins.css" />

<link rel="stylesheet" href="assets/css/raleway-webfont.css" />

<!--Theme custom css -->

<link rel="stylesheet" href="assets/css/style.css">

<!--Theme Responsive css-->

<link rel="stylesheet" href="assets/css/responsive.css" />

<script src="assets/js/vendor/modernizr-2.8.3-respond-1.4.2.min.js"></script>

<style>

body {

max-width: 980px;

margin: 2em auto;

font: normal 15px/1.5 Arial, sans-serif;

color: #353535;

overflow-y: scroll;

}

.content {

margin: 100px 30px;

}

.field.buttons button {

margin-right: .5em;

}

#invalid {

display: none;

float: left;

width: 290px;

margin-left: 120px;

margin-top: .5em;

color: #CC2A18;

font-size: 130%;

font-weight: bold;

}

.idealforms.adaptive #invalid {

margin-left: 0 !important;

}

.idealforms.adaptive .field.buttons label {

height: 0;

}

</style>

<link href="http://www.jqueryscript.net/css/jquerysctipttop.css" rel="stylesheet" type="text/css">

</head>

<body>

<nav class="navbar navbar-default navbar-fixed-top">

<div class="container">

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

<div class="navbar-header">

<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">

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

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

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

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

</button>

<h3 style="float:left;color:white;">Faculty Management System</h3>

</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 class="active"><a href="addfaculty.php">Add Faculty</a></li>

<li><a href="viewfaculty.php">View Faculty</a></li>

<li><a href="index.html">Log Out</a></li>

</ul>

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

</div><!-- /.container-fluid -->

</nav>

<div class="content">

<div class="idealsteps-container">

<nav class="idealsteps-nav"></nav>

<form action="facultyadd.php" autocomplete="off" class="idealforms" method="post" enctype="multipart/form-data">

<div class="idealsteps-wrap">

<!-- Step 1 -->

<section class="idealsteps-step">

<div class="field">

<label class="main">First Name</label>

<input name="fname" type="text" data-idealforms-ajax="ajax.php" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Middle Name</label>

<input name="mname" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Last Name</label>

<input name="lname" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Email ID</label>

<input name="emailid" type="email" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Picture:</label>

<input id="picture" name="file1" type="file" />

<span class="error"></span> </div>

<div class="field">

<label class="main">Department</label>

<select name="department[]">

<?php

include 'connfile.php';

$select=$con->prepare("select \*from department");

$select->execute();

$row=$select->rowCount();

if($row>0)

{

while($fetch=$select->fetch(PDO::FETCH\_ASSOC))

{

?>

<option value="<?php echo $fetch['dept'];?>"><?php echo $fetch['dept'];?></option>

<?php

}

}

?>

</select>

<span class="error"></span> </div>

<div class="field buttons">

<label class="main">&nbsp;</label>

<button type="button" class="next">Next &raquo;</button>

</div>

</section>

<!-- Step 2 -->

<section class="idealsteps-step">

<div class="field">

<label class="main">Basic Degree</label>

<select name="basic[]">

<?php

include 'connfile.php';

$select=$con->prepare("select \*from degree\_table");

$select->execute();

$row=$select->rowCount();

if($row>0)

{

while($fetch=$select->fetch(PDO::FETCH\_ASSOC))

{

?>

<option value="<?php echo $fetch['degree'];?>"><?php echo $fetch['degree'];?></option>

<?php

}

}

?>

</select>

<span class="error"></span> </div>

<div class="field">

<label class="main">Aggregate</label>

<input name="bagree" type="number" min="0" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Master Degree</label>

<select name="master[]">

<?php

include 'connfile.php';

$select=$con->prepare("select \*from masterdegree\_table");

$select->execute();

$row=$select->rowCount();

if($row>0)

{

while($fetch=$select->fetch(PDO::FETCH\_ASSOC))

{

?>

<option value="<?php echo $fetch['masterdegree'];?>"><?php echo $fetch['masterdegree'];?></option>

<?php

}

}

?>

</select>

<span class="error"></span> </div>

<div class="field">

<label class="main">Aggregate</label>

<input name="magree" type="number" min="0" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Additional Degree</label>

<select name="adegree[]">

<?php

include 'connfile.php';

$select=$con->prepare("select \*from additionaldegree");

$select->execute();

$row=$select->rowCount();

if($row>0)

{

while($fetch=$select->fetch(PDO::FETCH\_ASSOC))

{

?>

<option value="<?php echo $fetch['additional'];?>"><?php echo $fetch['additional'];?></option>

<?php

}

}

?>

</select>

<span class="error"></span> </div>

<div class="field">

<label class="main">Aggregate</label>

<input name="aagree" type="number" min="0" required />

<span class="error"></span> </div>

<div class="field buttons">

<label class="main">&nbsp;</label>

<button type="button" class="next">Next &raquo;</button>

</div>

</section>

<!-- Step 3 -->

<section class="idealsteps-step">

<div class="field">

<label class="main">Address</label>

<input name="address" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Phone Number</label>

<input name="phoneno" type="number" min="0" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">City</label>

<input name="city" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">District</label>

<input name="district" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">State</label>

<input name="state" type="text" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Pincode</label>

<input name="pincode" type="number" min="0" required />

<span class="error"></span> </div>

<div class="field">

<label class="main">Security Question</label>

<select name="security[]">

<?php

include 'connfile.php';

$select=$con->prepare("select \*from securityquestions");

$select->execute();

$row=$select->rowCount();

if($row>0)

{

while($fetch=$select->fetch(PDO::FETCH\_ASSOC))

{

?>

<option value="<?php echo $fetch['question'];?>"><?php echo $fetch['question'];?></option>

<?php

}

}

?>

</select>

<span class="error"></span> </div>

<div class="field">

<label class="main">Answer</label>

<input name="ans" type="text" required />

<span class="error"></span> </div>

<div class="field buttons">

<label class="main">&nbsp;</label>

<button type="button" class="prev">&laquo; Prev</button>

<button type="submit" class="submit" name="submit">Submit</button>

</div>

</section>

</div>

<span id="invalid"></span>

</form>

</div>

</div>

<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>

<script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.10.3/jquery-ui.min.js"></script>

<script src="js/out/jquery.idealforms.js"></script>

<!--<script src="js/out/jquery.idealforms.min.js"></script>-->

<script>

$('form.idealforms').idealforms({

silentLoad: false,

rules: {

'fname': 'required first name',

'mname': 'required middle name',

'lname': 'required last name',

'emailid': 'required email',

'department': 'select:default',

'basic': 'select:default',

'master': 'select:default',

'security': 'select:default',

'picture': 'required extension:jpg:png',

'bagree': 'please enter basic degree aggregate',

'magree': 'please enter master degree aggregate',

'aagree': 'please enter aaditional degree aggregate',

'address': 'required address',

'ans':'required answer',

'city': 'required city',

'district': 'required district',

'state': 'required state',

'pincode': 'required pincode',

'adegree': 'select:default',

},

errors: {

'fname': {

ajaxError: 'Username not available'

}

},

}

);

$('form.idealforms').find('input, select, textarea').on('change keyup', function() {

$('#invalid').hide();

});

$('form.idealforms').idealforms('addRules', {

'comments': 'required minmax:50:200'

});

$('.prev').click(function(){

$('.prev').show();

$('form.idealforms').idealforms('prevStep');

});

$('.next').click(function(){

$('.next').show();

$('form.idealforms').idealforms('nextStep');

});

</script>

</body>

</html>

**CHAPTER 7**

**TESTING AND RESULT**

**TWO TYPES OF TESTING:**

1. White box testing (also known as clear box testing, glass box testing and transparent-box testing and structural testing)
2. Black box testing: Treats the software as a “black box”, examining functionality without any knowledge of internal implementation. It is also known as Release testing, functional testing. This method of test can be applied to software testing: unit, integration, system and acceptance.
3. **UNIT TESTING:**

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code, usually at the function level. IN an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors. Unit testing alone cannot verify the functionality of a piece of software, but rather is used to assure that the building blocks the software uses work independently of each other. Depending on the organizations expectations for software development, unit testing might include static code analysis and other software verification practices.

1. **SYSTEM TESTING:**

System testing tests a completely integrated system to verify that it meets its requirements. The concern of this testing is to check the behavior of the whole system as defined by the scope of the project. The main concern of system testing is to verify the system against the specified requirements. While carrying out the test, the one who tests is not concerned with the internal of the system, but checks if the system behaves as per expectations.

1. **TEST REPORT:**

* **PERFORMANCE:** The application will be developed with a intension of providing the system that is highly interactive and takes less time and maintains the faculty information properly.
* **SCALABILITY:** Application will be flexible enough to include the resources; any changes with dataset will be reflected in functionality.
* **AVAILABILITY:** Once the system is deployed the system will be available to the users if you are connected.
* **RELIABILITY:** Reliability ensures that system will not fail to perform its intended functions. Any sudden changes will be handled in programming concept.
* **USABILITY:** This application is very easy for usage. Environment is built user friendly. No complications are seen in the system.

1. **TEST CASES:**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of the software implementation.

Software testing can be stated as the process of validating and verifying that a computer program meets the requirements that guided its design and development, application works as expected, product can be implemented with the same characteristics and satisfies the needs of customers.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sl. No** | **Case** | **Input** | **Expected Output** | **Actual Output** | **References** |
| 1. | Admin Login | Username and Password | Administrator Page | Administrator Page | Fig11.2 |
| 2. | Faculty Login | Faculty id and Password | Profile Page of Faculty | Profile Page of Faculty | Fig11.3 |
| 3. | Add Publications | Publications Credentials | Publication details adding | Publication details adding | Fig11.6 |
| 4. | Add Workshops | Workshop  credentials | Workshop details adding | Workshop details adding | Fig11.8 |
| 5. | View Publications |  | View page of publications | View page of publications | Fig11.7 |
| 6. | View Workshops |  | View page of workshops | View page of workshops | Fig11.9 |
| 7. | Add Book Publications | Book Publications Credentials | BookPublication details adding | BookPublication details adding | Fig11.10 |
| 8. | View Book Publications |  | View page of Book publications | View page of Book publications | Fig11.11 |

**CHAPTER 8**

**CONCLUSION**

System allows adding the faculty records and maintaining the faculty records. And this software gives the functionality for the faculties like adding publications details, adding workshops details and by using this software faculty can apply leave.

This software is completely designed for the Faculty Management and performing and maintaining their functionalities. The administrator has the power of adding, viewing and removing the faculty records,

**CHAPTER 9**

**FUTURE ENHANCEMENT**

* The given records are maintained and can be handled by the administrator.
* The faculty functionality will be get more modified to use the software in better way.
* They can make enhancements if they are willing to make.
* You can host this application and bring necessary changes.
* Auto generation of Email Notification can be implemented based on triggering concept.
* Android App can be developed further.

**CHAPTER 10**

**BIBILIOGRAPHY**

**BOOKS:**

* Ian Sommerville “Software Engineering” Pearson8th edition...
* John Allsopp “The Truth about HTML” 2012-13 edition...
* Charles Wyke-Smith“Stylin’ with CSS” Third edition
* “Web Programming”, by ‘Chris Bates’ Wiley Dreamtech India, 2nd Edition.
* Ian Somerville “Software Engineering”, 9th Edition, Pearson Publication,
* Jacobson, “Object-Oriented Software Engineering: A Use Case Driven
* RamezElmasri and Shamkant B. Navathe, “Fundamentals of Database”, 7th

**WEBSITE:**

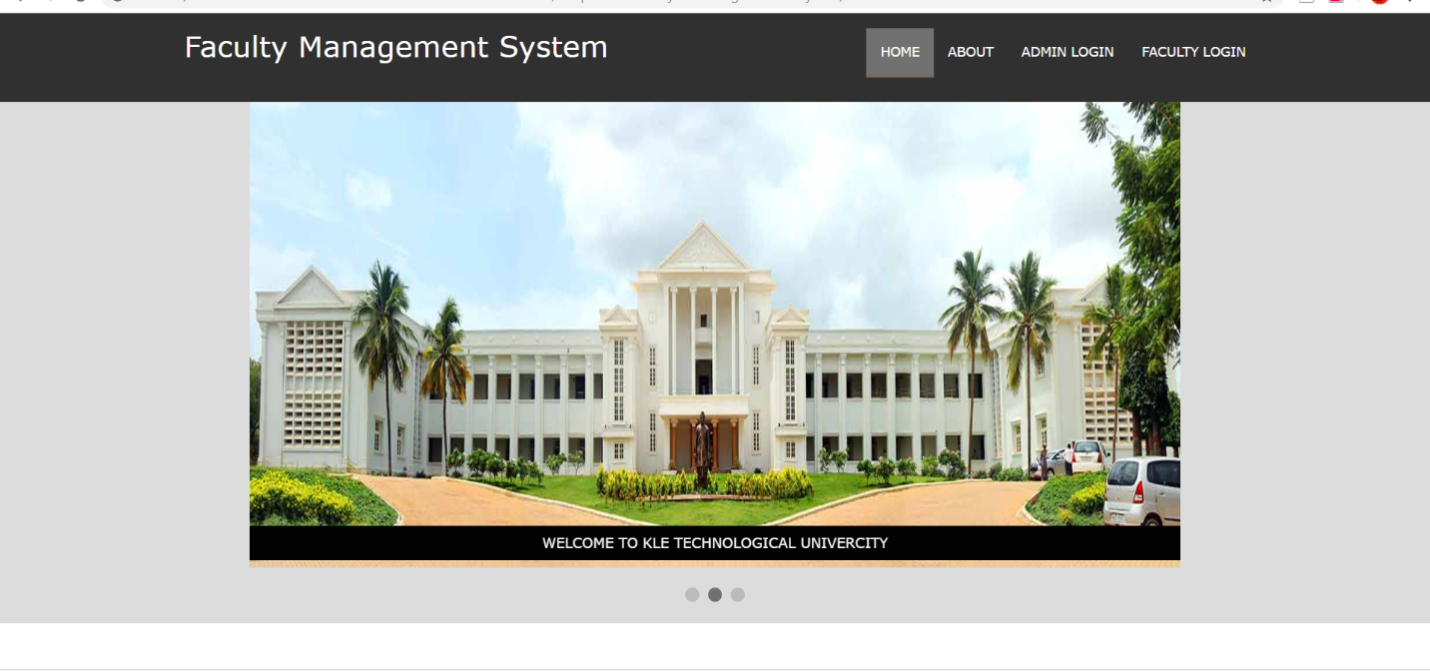
* http://en.wikipedia.org/wiki/html
* www.tutorialspoint.com/html
* <https://blog.commlabindia.com/elearning-development/characteristics-of-lms>
* <http://ijettjournal.org/archive/ijett-v34p263>
* <http://ijrise.org/asset/archive/17ICEMTE-COMP1.pdf>
* <https://www.researchgate.net/publication/325253983_Online_Car_Rental_System_using_Web-> Based\_and\_SMS\_Technology

**CHAPTER 11**

**USER MANUAL**

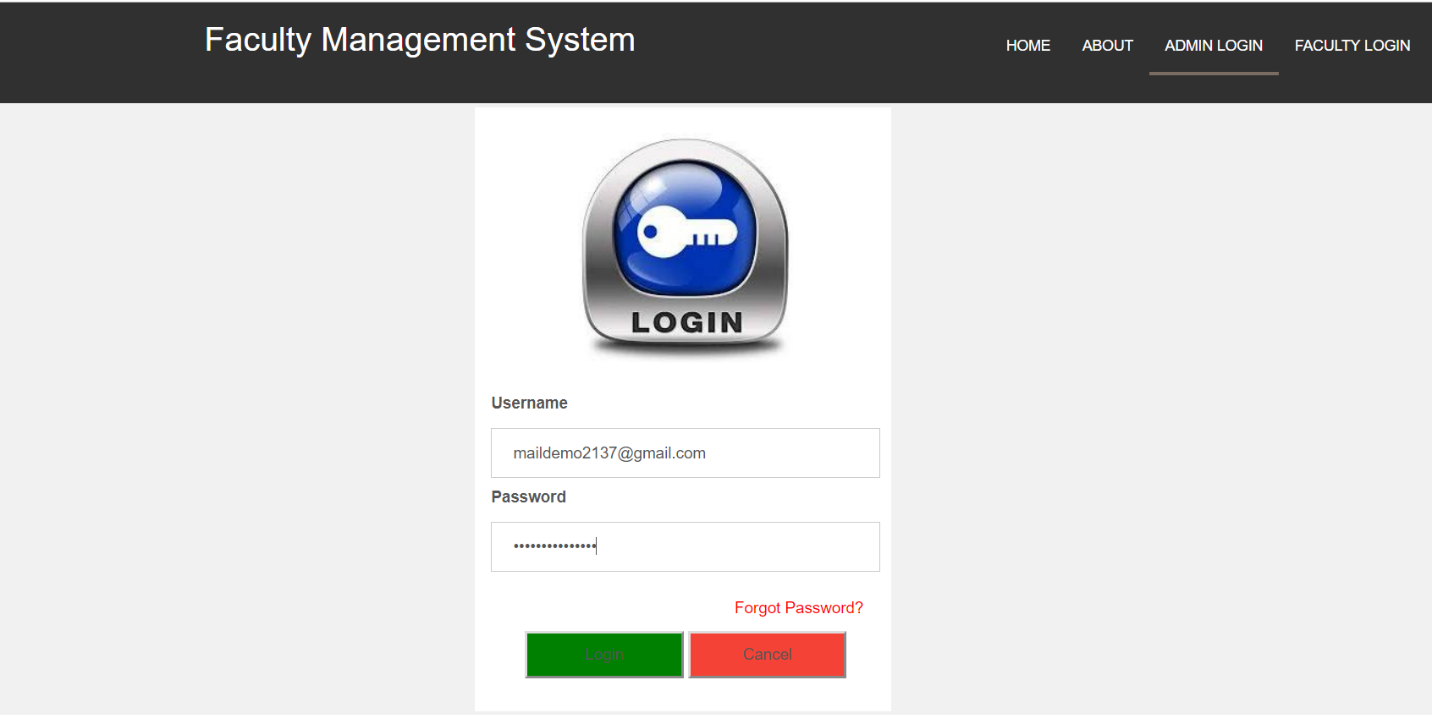
A user manual is a technical communication document intended to give assistance to people using a particular system. Faculty Management System is basically a system which will be used by the Faculty. This system will be used by faculty and admin for academic purpose.

**ScreenShot 1: HOME PAGE:**



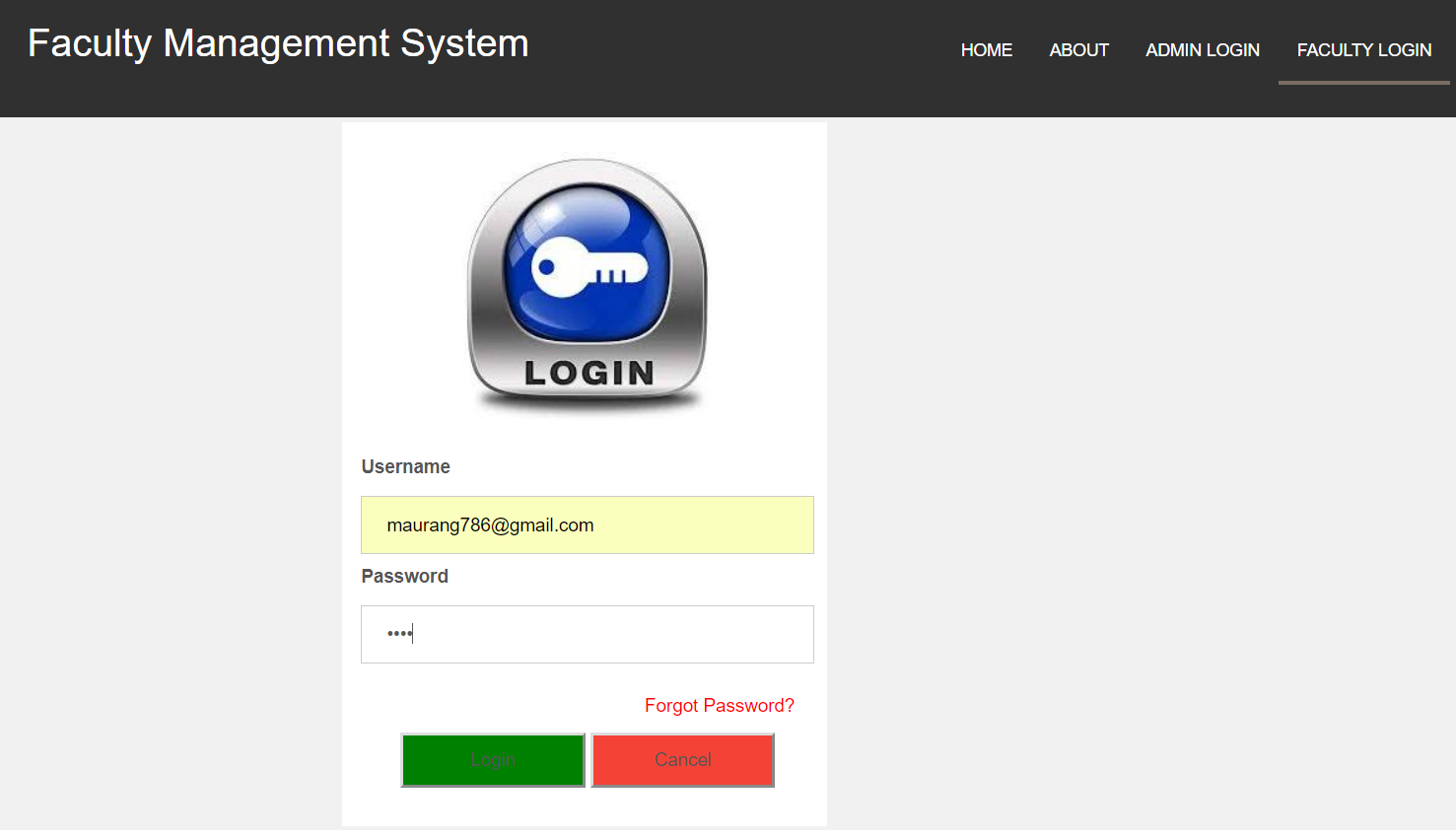
It is a home page from here admin and faculty can go to their respective login page

**ScreenShot 2: ADMIN LOGIN:**



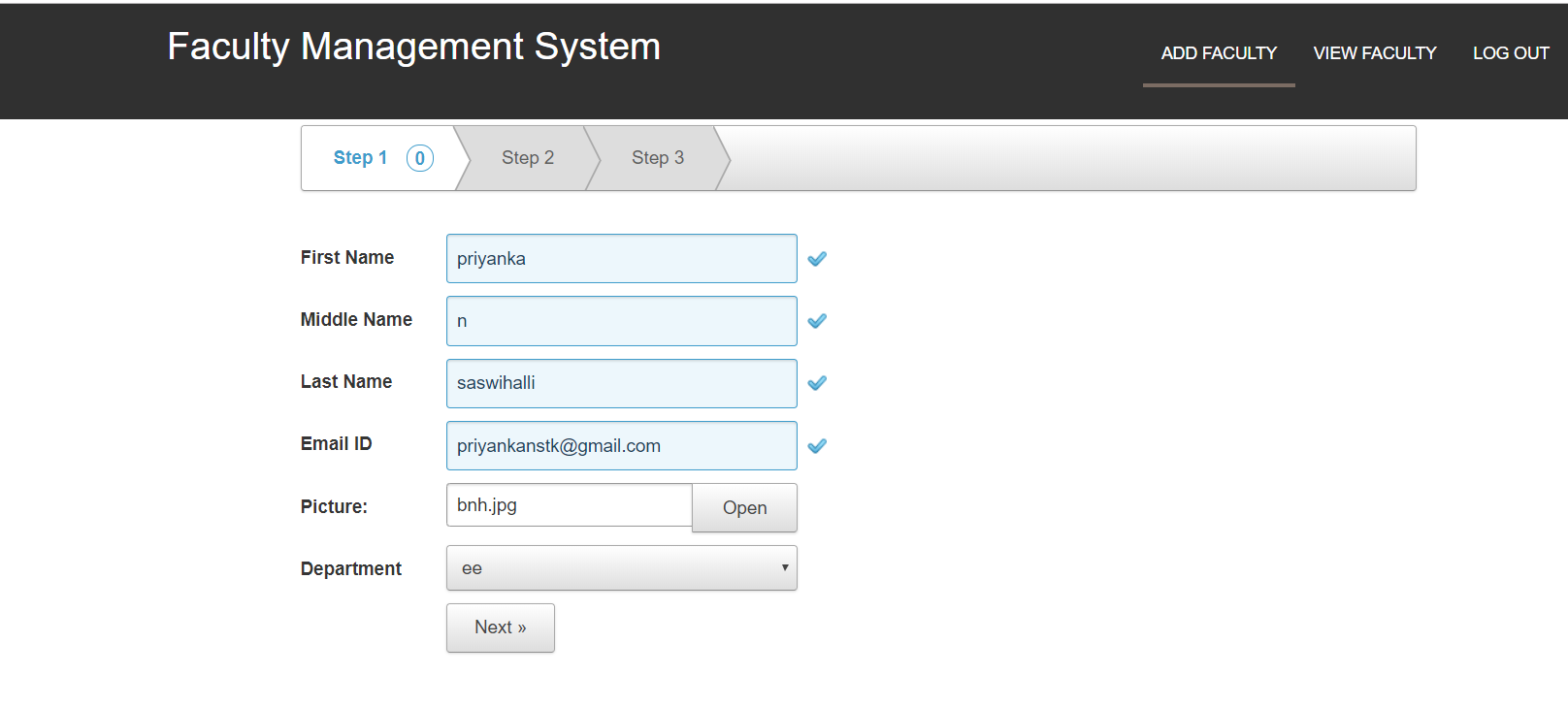
Admin can login with his/her valid username and password.

**ScreenShot 3: FACULTY LOGIN:**



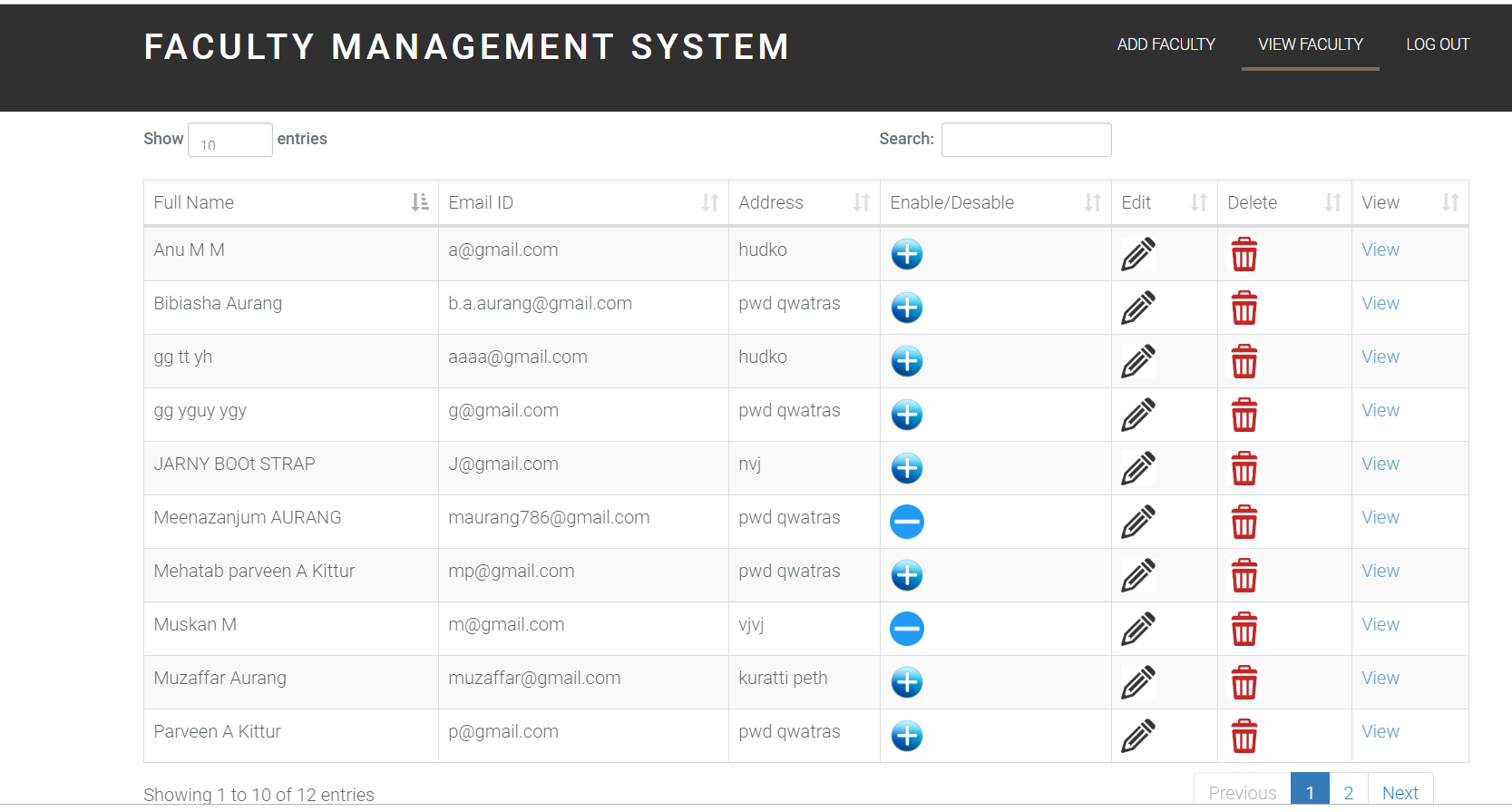
faculty can login with his/her valid emailid and password.

**ScreenShot 4: ADD FACULTY:**



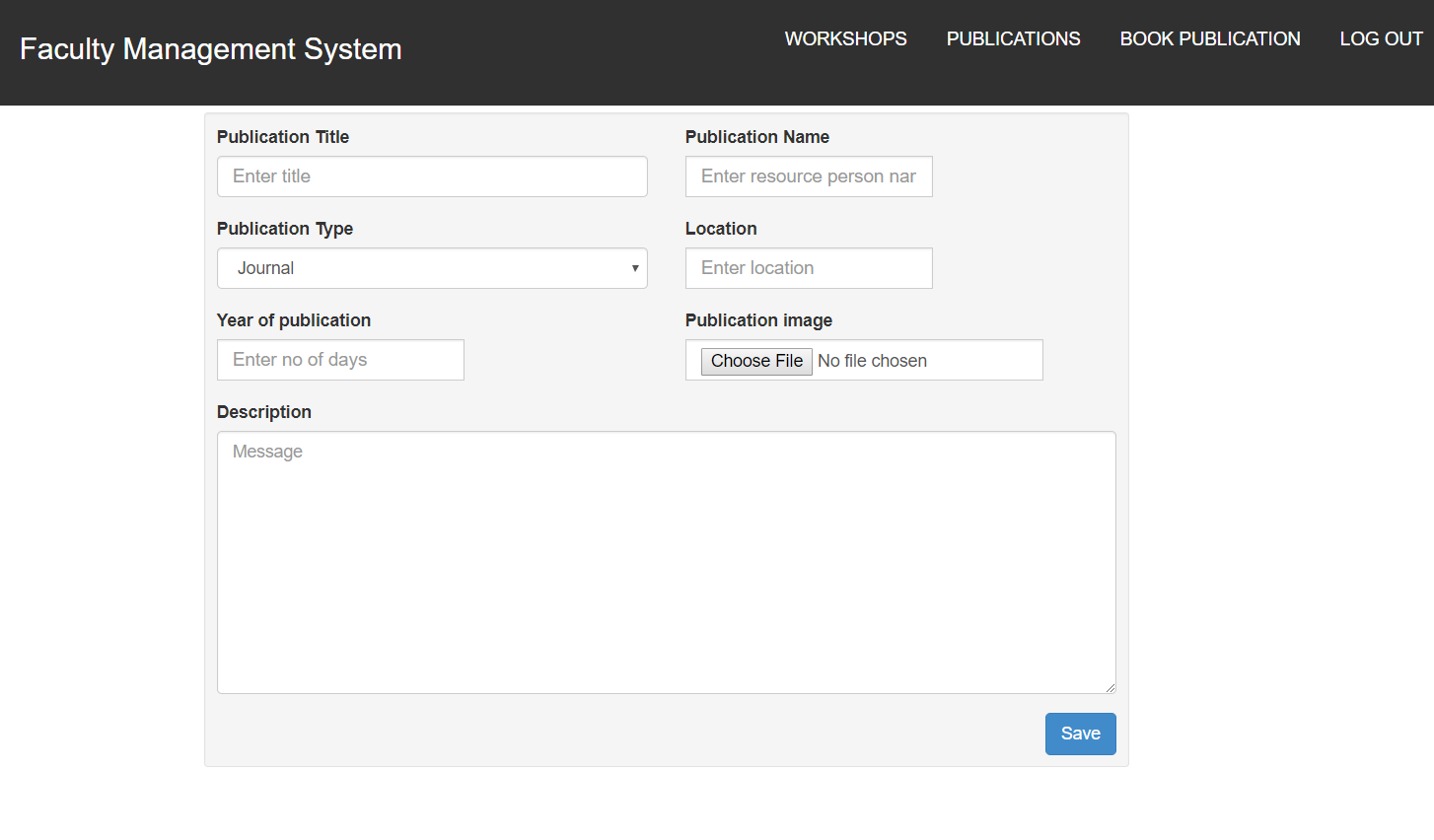
Admin can add the details of faculty like personal details, Qualification details and professional carrier details.

**ScreenShot 5: VIEW FACULTY:**



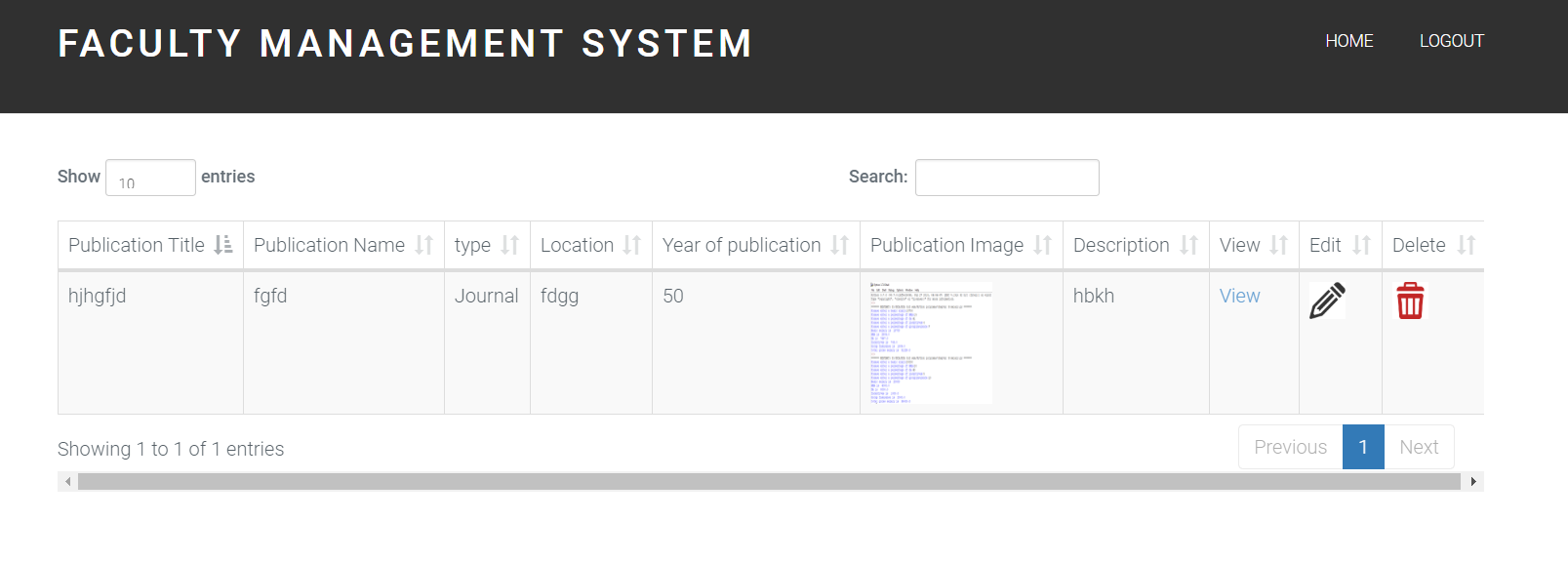
Faculty Details are displayed . Like their name, id, qualification and departmentid etc...

**ScreenShot 6: ADD PUBLICATION:**



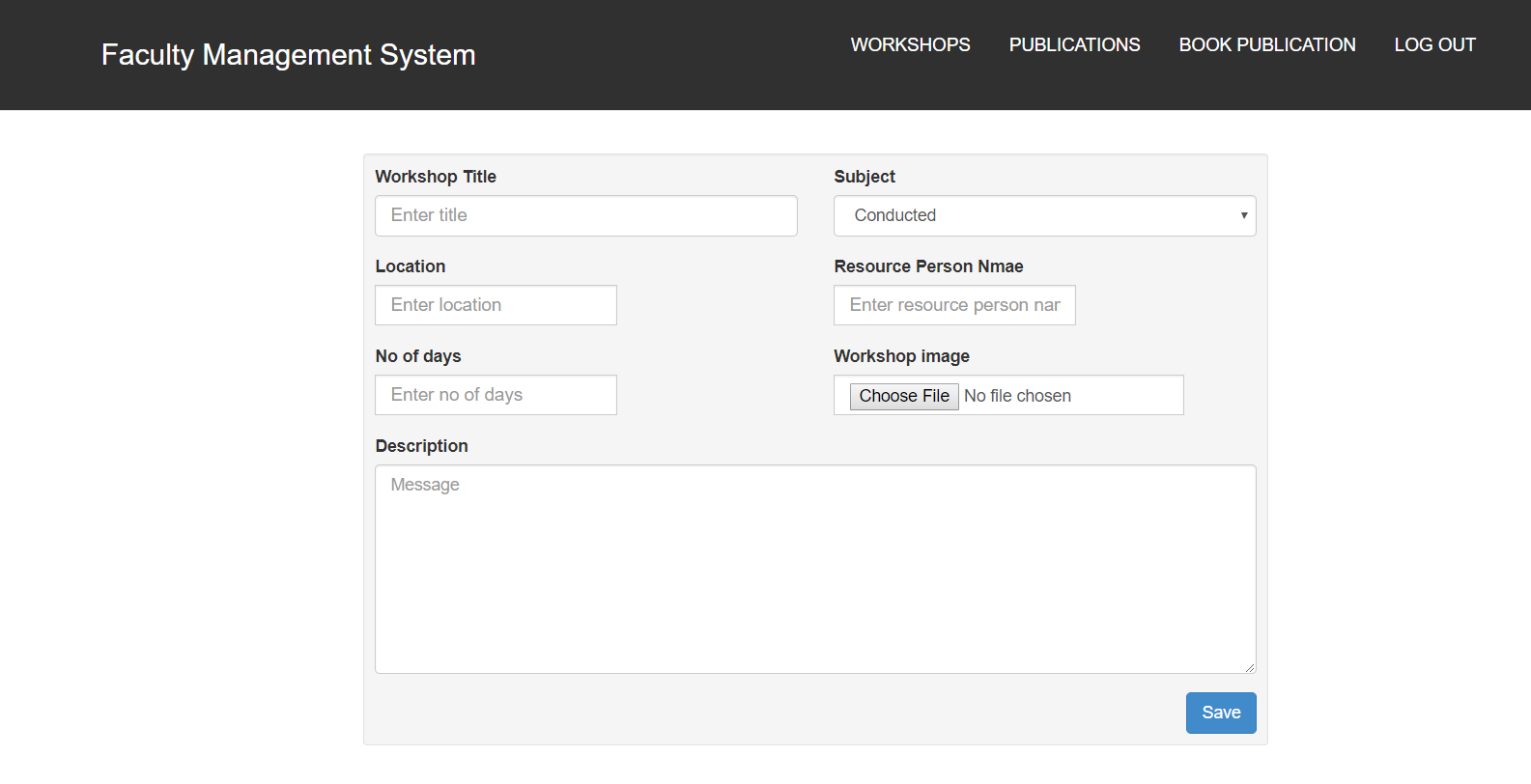
Faculty can add the details of publication like publication id, type of publication, year of publication etc…

**ScreenShot 7: VIEW PUBLICATION:**



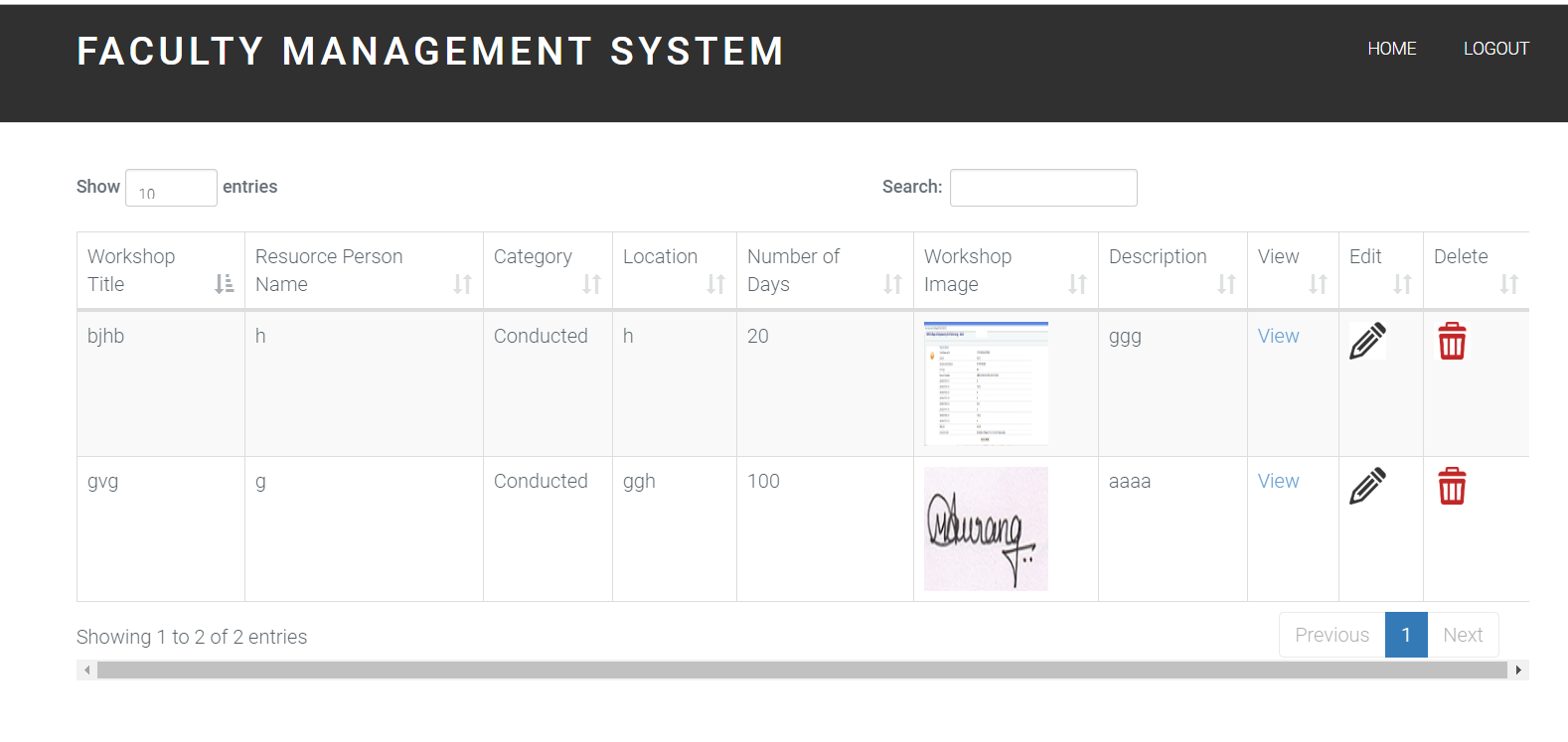
Publication details are viewed like publication id, type of publication, year of publication etc…

**ScreenShot 8: WORKSHOP ATTENDED AND CONDUCTED:**



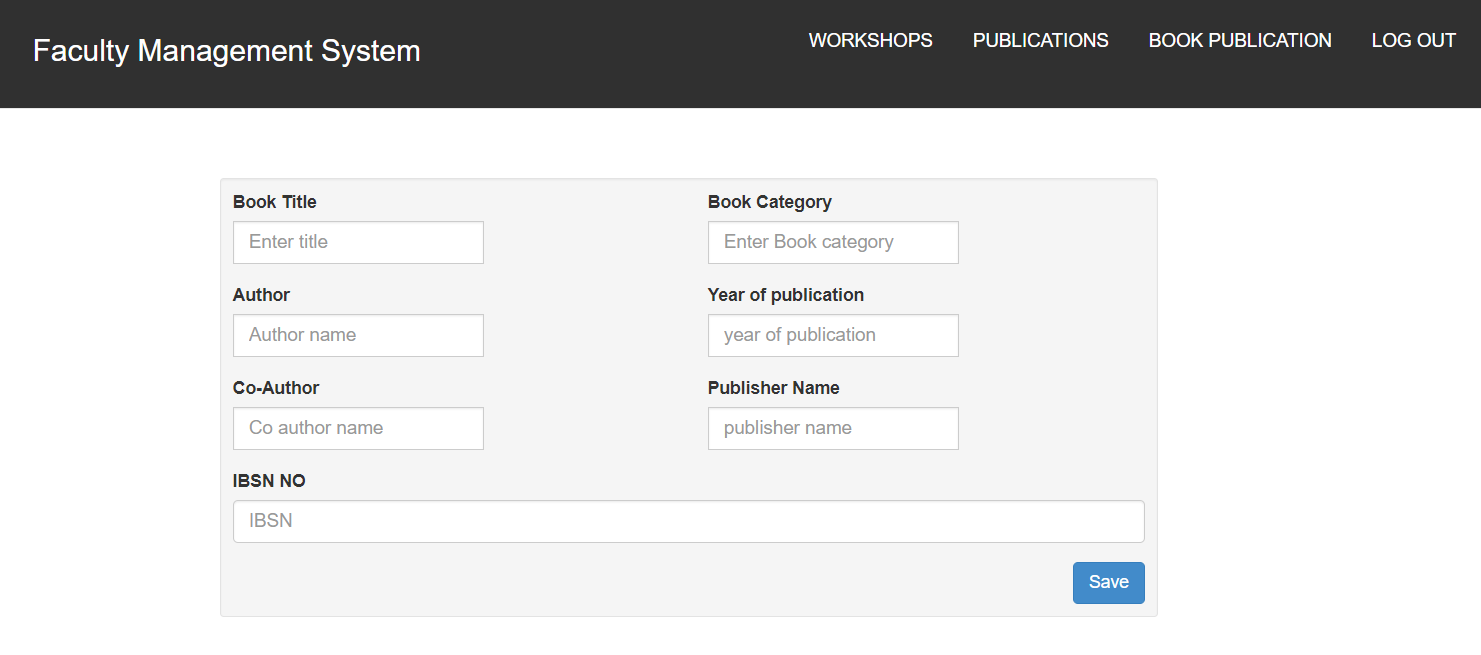
Workshop attended details can add like title of the workshop, attended date, location and number of days attended etc…

**ScreenShot 9: WORKSHOP DETAILS:**



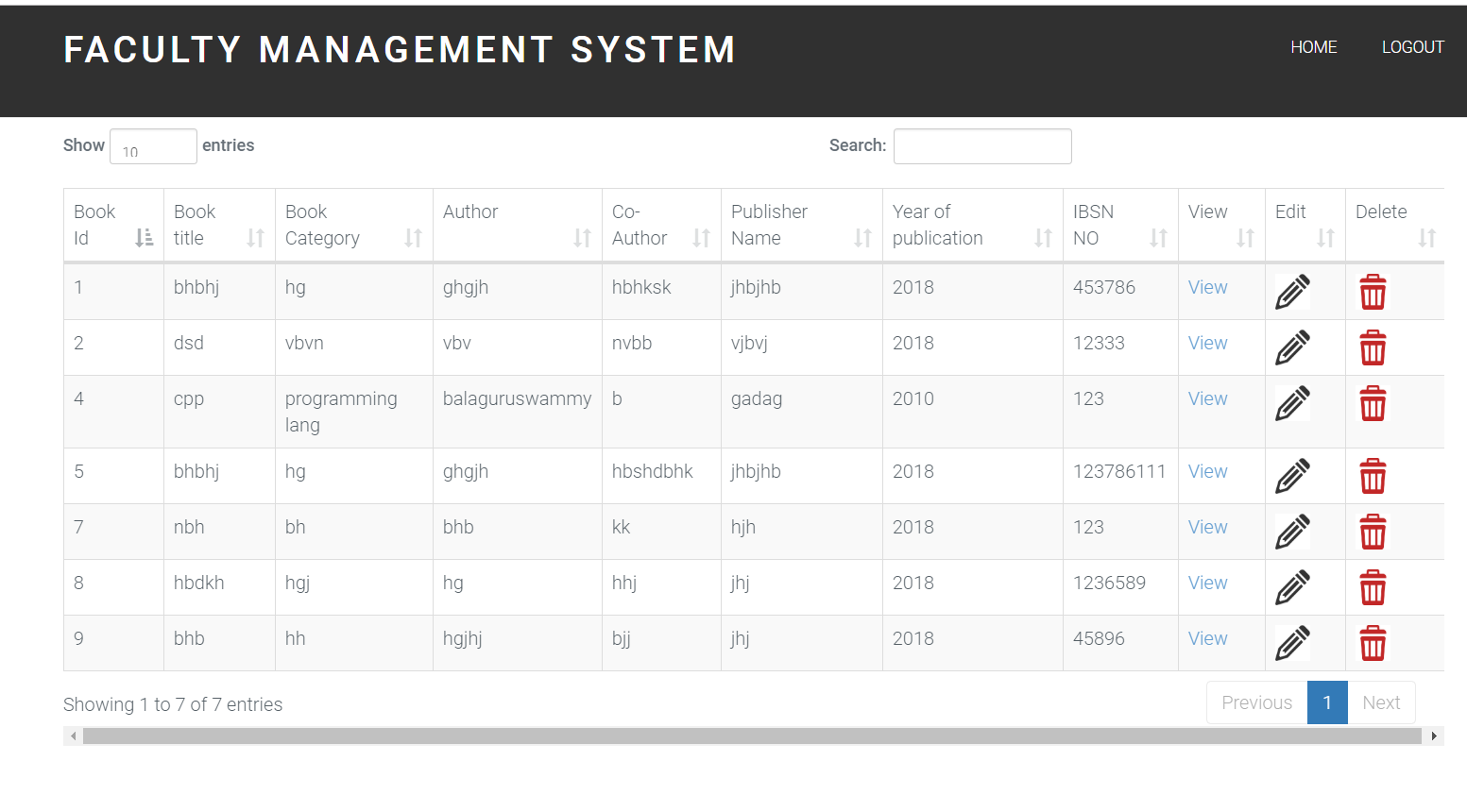
Workshop attended details are view like title of the workshop, attended date, location and number of days attended etc…

**ScreenShot 10: ADD BOOK PUBLICATION:**



Faculty can add the details of book publication like book title ,book category, author ,year of publication etc…

**ScreenShot 11: VIEW BOOK PUBLICATION:**



Book Publication details are viewed like book title ,book category, author ,year of publication etc…