# PROJECT REPORT

#### ON

**“WEBBLOG”**

**IN**

**TecHindustan Private Ltd. Mohali**

**A REPORT SUBMITTED FOR SOFTWARE DEVELOPMENT PROJECT (FOR MCA STUDENTS) IN PARTIAL FULFILMENT OF DEGREE OF**

**MASTER OF COMPUTER APPLICATION**

**SUBMITTED TO HIMACHAL PRADESH UNIVERSITY**



**GUIDED BY: SUBMITTED BY:**

Mr. Parvesh Sood Kritika Chauhan

Head of Technology Department MCA 6th Semester Univ.Roll No: 6510

**K.L.B.D.A.V. GIRLS COLLEGE 2013-2016**

**Acknowledgement**

My journey towards achieving my destination for the design and development of “WEBBLOG” has finally come to a fruitful culmination.

My efforts and wholehearted co-corporation of each and every one has ended on a successfulnote. During this journey, I faced numerous unforeseen problems and unknown challenges. However, at these junctures, a few enterprising people stepped in and guide me in a right direction.

I would like to extend my sincere acknowledgement to those who have supported and encouraged me during this tough journey. Many people met me during this endeavor and enriched me with their support and knowledge both personally and professionally that resulted in the project being better that it could possibly have been without them.

I express my sincere gratitude to my Internal Guide Mr. Parvesh Sood, who assisting me throughout this project. I thank him for providing me the reinforcement, confidence and most importantly the track for the project whenever I needed it.

I’ll also express my sincere gratitude to Our Project Leader Mr. Babban Singh of TecHindustan Pvt. Ltd., for their constant encouragement and support at all stages of this project. I also thank them for inducing professional attitude in me.

At last but not the least, I pay my due regards to my parents and Friends, because every time they encourage, and support me when I need or require.

Kritika Chauhan

**Internal Guide’s Profile**

**Name :**

**\_\_\_\_\_**

**University :**

**\_\_\_\_\_ \_**

**Designation :**

**\_\_\_\_\_**

**Current Experience :**

**\_\_\_\_\_**

**Past Experience (if any) :**

**\_\_\_\_\_**

**\_\_\_\_\_ \_**

**\_\_\_\_**

**\_\_\_\_\_ \_**

**\_\_\_\_**

**Qualification : \_\_\_\_\_ \_ Nos. of students guided : \_\_\_\_\_ \_\_**



**TABLE OF CONTENTS**

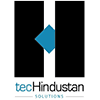
|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENTS** | **PAGE NO.** |
| **1.** | **INTRODUCTION TO COMPANY** | 1 |
| **2.** | **INTRODUCTION TO PROJECT** | 2,3 |
| 2.1 | **Weblog/webblog** | 3 |
| 2.2 | **Why we choose this project** | 3 |
| 2.3 | **How it is different from** | 3 |
| 2.3.1 | **Existing System**  **Drawbacks of existing** | 4 |
| 2.3.2 | **system** | 4 |
| 2.4.1 | **Benefits of Project** | 5 |
| 2.4.2 | **Application** | 5 |
| 2.4.3 | **scope of project** | 5 |
| 2.4.4 | **Organization of report** | 8,9 |
| **3.** | **SYSTEM MODULE** | 8 |
| 3.1 | **Module** | 8 |
| 3.2 | **Change account information** | 8 |
| 3.2.1 | **Search reference case** | 9 |
| 3.2.2 | **Other functionalities**  **Input requirement of system** | 9 |
| 3.2.3 | **output requirement of system** | 10 |
|  | *Maintenance* |  |

|  |  |  |
| --- | --- | --- |
| 3.3 |  | 15 |
| **4.** | **SYSTEM ANALYSIS** | 15-29 |
| **5.** | **PLATFORM** | 30-35 |
| **6.** | **DESIGN CONSIDERATIONS**  SOFTWARE REQUIREMENT | 36  37  38 |
|  | HARDWARE REQUIREMENT |  |
| **7.** | **FEASIBILITY STUDY** | 39 |
|  | INTRODUCTION | 40 |
|  | TYPES OF FEASIBILITY | 43 |
|  | SDLC | 44 |
|  | DFD | 46 |
|  | USE CASE DIAGRAM | 50 |
| **8.** | **SYSTEM DESIGN** | 52 |
|  | DATABASE | 54 |
|  | SNAPSHOTS OF PROJECT | 55 |



|  |  |  |
| --- | --- | --- |
| **9.** | **INTERFACE DESIGN, CODING** | 70 |
| **10.** | **SYSTEM IMPLEMENTATION** | 93 |
| **10.** | **SYSTEM TESTING** | 100 |
| **11.** | **CONTACT AND ABOUT US** | 117 |
| **13** | **CONCLUSION** | 118 |

1. **Introduction to Company-:-**



TecHindustan Solutions Pvt. Ltd is One of Asia's Best Full-spectrum It Services and Industry Solutions Providers with Global Delivery Capabilities. TecHindustan Solutions Pvt. Ltd Focuses On It Consulting and Web Services to Various Companies in Businesses of Financial Services, Telecommunications, Consumer Electronics, Manufacturing, Pharmaceuticals, Healthcare, Automobile, Media, Energy, Education Sectors. With the Best Track Record of Delivering Technology Leadership and Operation Excellence for Its Clients in Various Countries.

Headquartered in Mohali, Punjab, India, TecHindustan Looking Forward to Possesses Nationwide Branches, as Well as An International Footprint in the other States & Countries. Our Cross-globe Network of Talents, Flexible Engagement Models, and Professional Environment Enable Us to Provide Top-quality Cost- effective Services. **To attain optimum performance, we create responsive projects fulfilling the customer’s attributes. Our IT solutions are credible and trusted offering full customer satisfaction. We ensure our customers of a defined growth in their business.**

1. **Introduction**

**(2.1) Project Introduction:-**

Blogging has become such a mania that a new blog is being created every second of every minute of every hour of every day. A blog *is* your best bet for a voice among the online crowd. Blogs were usually the work of a single individual occasionally of a small group, and often covered a single subject. More recently, "multi-author blogs" (MABs) have developed, with posts written by large numbers of authors and professionally edited. MABs from [newspapers,](https://en.wikipedia.org/wiki/Newspaper) other media outlets, [universities,](https://en.wikipedia.org/wiki/University) [think tanks,](https://en.wikipedia.org/wiki/Think_tank) [advocacy groups,](https://en.wikipedia.org/wiki/Advocacy_group) and similar institutions account for an increasing quantity of blog traffic. The rise of [Twitter](https://en.wikipedia.org/wiki/Twitter) and other "[microblogging"](https://en.wikipedia.org/wiki/Microblogging) systems helps integrate MABs and single-author blogs into societal new streams. *Blog* can also be used as a verb, meaning to maintain or add content to a blog. A novel is a long, fictional narrative which describes intimate human experiences.

WEBLOG is a combination of both Blog as well as Novels. Blog contain the Information of various things related to Technology, Education, News, International, Business, Sports, Entertainment and ongoing college activities. The main aim of this project is to provide data to students in only one site. Students can gather the information from one site as well as give their feedback and create their own blog. Students can post their views and thought and analyze themselves. Besides all such core functionalities, the application also includes features like FAQ, request, feedback etc. so as to provide a satisfactory user experience.

##### (2.2) WHY WE CHOOSE THIS PROJECT:-

In recent past time Blogs are store in the paper files and difficult to search or modify any information, for expanding the Blogs infrastructure, Awareness of environmental issues or any other issues related to education, health, digital technology, and search for greater safety give to information to all persons in all age groups and a new role within the education system, I choose this project. As a result of these project initiatives phenomenal growth has taken place in all the activities of blogs and any user can share its information related to any topic to all users.

##### (2.3.1) EXISTING SYSTEM:-

Existing system is manual system. It requires a lot of file work to be done. It is a time consuming system. All customer information is maintained manually. Any searching requires so much effort manually.

There is no way of spreading the information so fast and in cheapest manner. In previous system all information does not get in one place. Here people can write whatever they want to write.

##### (2.3.2) DRAWBACKS OF EXISTING SYSTEM:-

* 1. **Data redundancy and formatting:** The various files are likely to have different formats and therefore lead to redundancy and inconsistency.
  2. **Maintaining registers is costly:** Traditionally documents have been stored in batches and they field in file cabinets and boxes. A numerical system is they assigned. Specifically a consumer number assigned to organize the files.
  3. **Error prone:** Existing systems are error prone, since manual work is required. More time is consumed and errors may propagate due to human mistakes.
  4. **Low security feature:** Due to maintenance's of records manually and shared and could view easily by anyone. Also these could be possible loss of data and confidential information due to some disaster in the form of fire, theft etc.
     1. **Benefits of Project**

This is a very simple design and implement. It has got following features:

* Data can be saved safely.
* No other person cannot view other person’s detail
* Greater efficiency
* User friendliness
* Minimum time required
* Free of cost

### Applications

WEBLOG enables the users to create innovative and attractive information with photos in just few simple steps. The user just needs to upload some images of his choice and can also upload the information or can select from the given category list. This website will provide a personalized environment that would contain the data in motion with images.

### Scope of the Project

* + - * To use the personal images in greeting our loved ones.
      * To use the music of our choice to greet in the language we want.
      * To provide a personalized mp4 video to the customers.
      * To satisfy the customers and provide them with the ordered video before time.

### Organization of Report:-

Based on the outline design of the system requirements in terms of inputs, output, Procedures, the technical issues raised during technical feasibility include:

* Does the necessary technology exist to do what is proposed?
* Does the proposed equipment have the technical capacity to hold the data required to use in the new system?
* Adequate responses provided by the proposed system?
* Is the system flexible enough to facilitate expansion?
* Is there any technical guarantee of accuracy, reliability, ease of access and data security?
* The system developer’s task is to view needed capabilities in light of currently available technology. Our site works hand in hand with high technology. A database has to be maintained in order to update and backup data whenever required. To create databases we use MySQL server.
* For the acceptance within the organization the following points are important and those are explained according to the topics
* Whether the system provides right information to the right place?
* In the current system which is the semi computerized system the information may be lost in the process of sending from one place to another. This is mainly due to human interaction in the process of the transferring information from one place to another. Whether the new system affects the current users in the system?

The new proposed system will affect the users in the following areas

* + - * Accuracy
      * Efficiency
      * Productivity

### SYSTEM MODULE:-

##### The modules involved in this project are:

* + 1. User
    2. Admin

##### User

In this module:

* User can signup
* User can login
* User can upload multiple images
* User can choose images from database
* User can add information
* User can add comments
* User can select any categories

##### Admin

In this module:

* Admin can block user id
* Admin can resume user id
* Admin can see all pages
* Admin can maintain all records of user
* Admin can maintain all site
* Admin can access and process all requests
* Admin can delete/update/select users
* Admin Provide all information related to any topic

##### Change Account Information:-

Update Account information like Name, Address, Email ID and their detail.

* 1. **Search Reference Case:-**

System will provide an interface for Administrator & Subordinate searching comprehensively the reference information from the blogs of application on the basis of various parameters, like Title of blogs e.g. health.

* 1. **Other Functionalities: -**

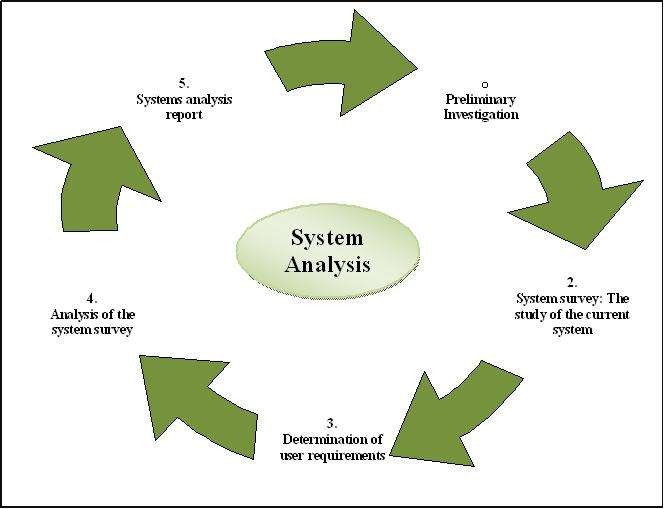
System will also provide other functionalities like:

* Feedback
  + Request
  + FAQ etc.
  1. **Inputs Requirements of the System:-**
* User Information
* Log in Information
* Comment Information
* Categories Information
* Recent Posts Information
* Registered Information
* Topic Information
* Feedback
* FAQ
* Request
  1. **Output Requirements of the System**
* Blog Information
* Novel Information
* Assigned information of users
* Categories information
* FAQ
* Request
  1. **Maintenance: -**

The system allows following Maintenance processes

* Manage Blogs
* Manage Blogs Categories
* Manage Recent posts
* Manage users
* Manage images
* Manage Comments
* Manage news
* Security Questions
* Manage Contact us

## SYSTEM ANALYSIS



##### DATA ANALYSIS

Before developing this project, we first analyse existing system of study. In existing system all greetings are given manually. As we know, now a day computer is used in every field. We can remove the manual work by using automatic system. We see it first that if it is feasible or not whether technically, economically, operationally. We test that whether it properly works or not. Its technical requirements are feasible or not. We analysed the system properly and then start designing it. After designing, we implement this project that whether this project works properly or not. After implementing the project, we check that whether there is any problem for the user while using this project. Prior to stating whether the system we have to develop is feasible or not we believe that we should emphasize on what is implied by the word “Analysis”. Analysis is the measure of how beneficial or practical the development of the system will be to the organization. It is a preliminary survey for the systems investigation. It aims to provide information to facilitate a later in-depth investigation.

**Types**

There are various measures of analysis that helps to decide whether a particular project is feasible or not.

These measures include –

* + - Operational Analysis
    - Technical Analysis
    - Economic Analysis

Each of these types will be explained in detail throughout the project report

##### Operational analysis

A proposed system is beneficial only if it can be turned into an information system that will meet the operational requirements of an organization. A system often fails if it does not fit within existing operations and if users resist the change. Important issues a systems developer must look into are:

* + - * Will the new system be used if implemented in an organization?
      * Are there any major barriers to implementation or is proposed system accepted without destructive resistance?

The whole purpose of computerizing it is to handle the work much more accurately and efficiently with less time consumption. There will be additional work to be completed, because now the website will have to maintain database of both their admins as well as their Customers. Compared to the semi- computerized system the chances of avoiding errors in a computerized system is much higher because the user need not stress himself unnecessarily resulting in recklessness. Unlike the semi-computerized system there would be backup data for all the information concerning the daily transactions occurred. Another important fact to be regarded is the security control, which is handled by the system. Since data regarding each Customer is confidential, security is a key issue. Information falling into the wrong hands could jeopardize the entire website organisation. Unlike in semi-computerized systems, the proposed

system offers adequate control to protect against fraud and embezzlement and guarantees the accuracy and Security of data and information. This is handled by the system providing individuals with separate login names and passwords. The new system is user-friendlier, which enables the end-user to complete his/her work efficiently and accurately with interest. After taking the above fact into consideration we can state the operating of the proposed system is feasible.

##### Economical Analysis

In making recommendations a study of the economics of the proposed system should be made. Even though finding out the costs of the proposed project is difficult we assume and estimate the costs and benefits as follows. According to the computerized system we propose, the costs can be broken down in two categories.

1. Costs associated with the development of the system.
2. Costs associated with operating the system.

##### System Security

System security is a vital aspect when it comes to developing a system. The system should ensure the **facility** of preventing unauthorized personnel from accessing the information and the data within the system. The system should provide total protection for each user’s information so that the integrity of data is sustained and also prevent hackers from hacking the system. The proposed system ensures the security and the integrity of data. This is done by providing a password login system for each authorized users. And for example the System Administrator has access to all kinds of information. By providing this facility information is properly managed and information is protected. For example the system administrator’s day to day tasks are lessened and easier because he doesn’t have to have a constant eye on the system and worry about hackers hacking the system.

The Analysis Phase is where the project lifecycle begins. The Analysis Phase is where you break down the deliverables in the high-level Project Charter into the more detailed business requirements. The Analysis Phase is also the part of the project where you identify the overall direction that the project will take through the creation of the project strategy documents.

Gathering requirements is the main attraction of the Analysis Phase. The process of gathering requirements is usually more than simply asking the users what they need and writing their answers down. Depending on the complexity of the application, the process for gathering requirements has a clearly defined process of its own. This process consists of a group of repeatable processes that utilize certain techniques to capture, document, communicate, and manage requirements. This formal process, which will be developed in more detail, consists of four basic steps.

1. **Elicitation** – I ask questions, you talk, I listen
2. **Validation** – I analyze, I ask follow-up questions
3. **Specification** – I document, I ask follow-up questions
4. **Verification** – We all agree

Although gathering requirements is the main focus during the Analysis Phase, there are other important activities during this phase as well. One is to create a Requirement Management Plan to define how the requirements will be documented, communicated, tracked and changed throughout the rest of the project lifecycle. This plan will specifically address establishing a baseline, a change control process, and a way to track the requirements through the rest of the lifecycle. Another important activity is to set the overall direction for work that does not take place until later. This is accomplished through a series of direction-setting strategy documents. For instance, once you have your requirements, you can start to set the overall direction for training in a Training Strategy document. The strategies are at a high-level and are later defined at a lower level before they are finally implemented toward the end of the project.

## PLATFORM

#### TECHNOLOGIES TO BE USED:-

**Platform:**

* 1. **Front end:** HTML, CSS, JAVASCRIPT, JQUERY, JSON, Ajax.
  2. **Back end:** MYSQL 2005 as the back end.
  3. **PHP:-**

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general purpose scripting language that is especially suited for web development and can be embedded into HTM to produce dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document rather than calling an external file to process data. The code is interpreted by a Web server with a PHP processor module which generates the resulting Web page.

PHP is a general purpose server side scripting language originally designed for web app to development to produce dynamic web pages. It is one of the first developed server side scripting language to be embedded into an HTML sources document rather than calling an external file to process data. The code is interpreted by a web server with the PHP processor, module which generated the resulting web page. It also has evolved to include a command line interpreter capably and can be deployed on most web servers and also a standalone share on almost every operating system and platform free of charge. A competitor to Microsoft active server pages server side script engine and similar language. PHP is installed on more than 20 million web sites and 1 million web server. Software that uses PHP includes Mediawiki, Joomla, and Word process, Cocrete5, MyBB and Drupal.

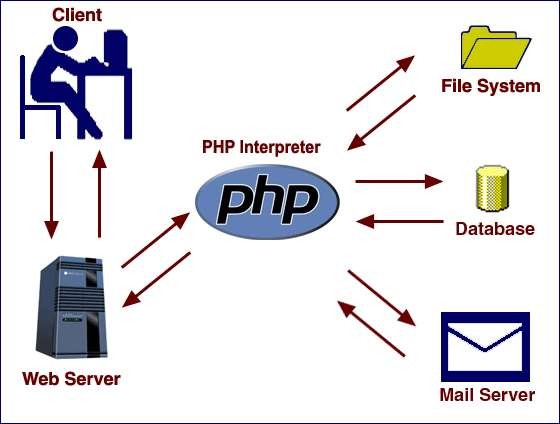
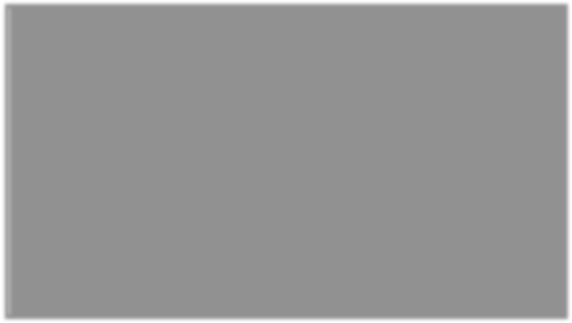


Fig 2.1 PHP interpreter

##### Advantages of PHP

* + PHP is accessible.
  + It is available for free.
  + It is available with documentation in many languages.
  + There are many support groups and team support PHP.
  + There is wealth of online information regarding PHP.
  + It is quick to develop in PHP.
  + PHP is loosely typed which makes basic script much faster to develop with less Attention to design.
  + It runs on many different operating systems.

##### What is the scope of PHP?

If you are planning to build your career in Web technology then PHP is the best programming scripting language to learn and is also a good career option. PHP

Is basically a scripting language used for web development. The websites created by PHP are dynamic and attractive. So Because of this reason it is demanded the

Most in web technology.PHP is an open source so is used freely without any cost And so greatly in demand. Scope in PHP really high as PHP is a language knows in The world of technology since many years. So it has gained the maximum popularity In this era.

##### HTML:

HTML is a language for describing web pages.

* + HTML stands for Hyper Text Markup Language
  + HTML is a markup language
  + A markup language is a set of markup tags
  + The tags describe document content
  + HTML documents contain HTML tags and plain text
  + HTML documents are also called web pages

**HTML Tags:**

HTML markup tags are usually called HTML tags.

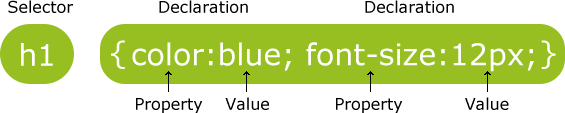
* + HTML tags are keywords (tag names) surrounded by angle brackets like <html>
  + HTML tags normally come in pairs like <b> and </b>
  + The first tag in a pair is the start tag, the second tag is the end tag
  + The end tag is written like the start tag, with a forward slash before the tag name
  + Start and end tags are also called opening tags and closing tags

##### <Tag name>content</Tag name>

* + 1. **CSS:**
  + CSS stands for Cascading Style Sheets
  + Styles define how to display HTML elements
  + Styles were added to HTML 4.0 to solve a problem
  + External Style Sheets can save a lot of work
  + External Style Sheets are stored in CSS files

**CSS Syntax:**

A CSS rule has two main parts: a selector, and one or more declarations:



* + The selector is normally the HTML element you want to style.
  + Each declaration consists of a property and a value.
  + The property is the style attribute you want to change. Each property has a value.
    1. **XAMPP:-**

**XAMPP** is a [free and open source](https://en.wikipedia.org/wiki/Free_software) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) [solution stack](https://en.wikipedia.org/wiki/Solution_stack) package developed by Apache Friends,[[1]](https://en.wikipedia.org/wiki/XAMPP#cite_note-kaiseidlerinterview-1) consisting mainly of the [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [Maria](https://en.wikipedia.org/wiki/MariaDB) [DB](https://en.wikipedia.org/wiki/MariaDB) [database,](https://en.wikipedia.org/wiki/Database) and [interpreters](https://en.wikipedia.org/wiki/Interpreter_(computing)) for scripts written in the [PHP](https://en.wikipedia.org/wiki/PHP) and [Per](https://en.wikipedia.org/wiki/Perl)l [programming](https://en.wikipedia.org/wiki/Programming_language)

[languages.](https://en.wikipedia.org/wiki/Programming_language)[[2][3]](https://en.wikipedia.org/wiki/XAMPP#cite_note-x_mariadb-2) XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (Maria DB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

|  |  |
| --- | --- |
| **Acronym Letter** | **Acronym Meaning** |
| X | XAMPP or an [ideographic](https://en.wikipedia.org/wiki/Ideogram) letter X, meaning [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) |
| A | [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server) or its expanded form, Apache HTTP Server[[5]](https://en.wikipedia.org/wiki/XAMPP#cite_note-abbreviations.com-5) |

|  |  |
| --- | --- |
| M | [MariaDB(](https://en.wikipedia.org/wiki/MariaDB)formerly: [MySQL](https://en.wikipedia.org/wiki/MySQL)) |
| P | [PHP](https://en.wikipedia.org/wiki/PHP) |
| P | [Perl](https://en.wikipedia.org/wiki/Perl) |

#### MY SQL:-:

MySQL 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); in July 2013, it was the world's second most widely used RDBMS, and the most widely used open-source [client–server model](https://en.wikipedia.org/wiki/Client%E2%80%93server_model) RDBMS. The [SQL](https://en.wikipedia.org/wiki/SQL) acronym stands for [Structured](https://en.wikipedia.org/wiki/Structured_Query_Language) [Query Language.](https://en.wikipedia.org/wiki/Structured_Query_Language)

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used [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))". [Free-software](https://en.wikipedia.org/wiki/Free_software) open-source projects that require a full-featured database management system often use MySQL. Applications that use the MySQL database include: [TYPO3,](https://en.wikipedia.org/wiki/TYPO3) [Joomla](https://en.wikipedia.org/wiki/Joomla), [WordPress](https://en.wikipedia.org/wiki/WordPress), [php BB](https://en.wikipedia.org/wiki/PhpBB), [My BB,](https://en.wikipedia.org/wiki/MyBB) [Drupal](https://en.wikipedia.org/wiki/Drupal) and other software.

* + MySQL is a database server.
  + MySQL is ideal for both small and large applications.
  + MySQL supports standard SQL.
  + MySQL compiles on a number of platforms.
  + MySQL is free to download and use.

The data in MySQL is stored in database objects called tables.

A table is a collection of related data entries and it consists of columns and rows. Databases are useful when storing information categorically. A company may have a database with the following tables: "Employees", "Products", "Customers" and "Orders".

**JAVASCRIPT:** JavaScript is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions. It is a multi-paradigm language, supporting object- oriented, imperative, and functional programming styles. JavaScript was formalized in the ECMA Script language standard and is primarily used in the form of client-side JavaScript,

implemented as part of a Web browser in order to provide enhanced user interfaces and dynamic websites. This enables programmatic access to computational objects within a host environment. JavaScript's use in applications outside Web pages — for example in PDF documents, site-specific browsers, and desktop widgets — is also significant. Newer and faster JavaScript VMs and frameworks built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. JavaScript uses syntax influenced by that of C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the self and Scheme programming languages.

* + - JavaScript was basically used for client-side validation.
    - JavaScript is compatible with all versions of Microsoft Internet
    - Explorer and Netscape Navigator.

**JSON:** JavaScript object notation is a lightly weight data-interchange format. It is easy for humans to read and write. It is for machines to parse and generate. It is based on a subset of JavaScript. Json is text format that is complete language independent but uses conventions that are familiar to programmers of the C family of languages, including C, C++, C#, Java, JavaScript, Perl, Python and many languages. These properties make Json an ideal data-interchange language.

**AJAX:** Asynchronous JavaScript and XML. Ajax is a technique for creating fast and dynamic web pages. Ajax allows web pages to be updated asynchronously by exchanging small amounts of data with server behind the scenes. This means that it is possible to update parts of web page, without reloading the whole page.

### Scripting Languages:-

* + - 1. HTML, XML, XSL, XHTML
      2. DHTML – CSS, JavaScript, DOM
      3. AJAX
      4. Bootstrap
      5. AngularJS

### DESIGN CONSIDERATIONS:-

* + - * 1. **Software Requirements:-**

JDK

Java SE Development Kit (JDK) 7

Web Server

◦ Apache 2.4.9x

Web Scripting Language

◦ PHP 5.5x

Database

◦ MySQL 5.6x

Environment

◦ NetBeans IDE 8.0.2x

Supported Operating Systems

◦ Windows 2000/XP/2003/Vista / Windows 7/ Windows 8/ XP64 / Vista64 / Windows 7 64 /

### Hardware Requirements:-

* **Minimum Hardware Configurations**
  + Processor: 800MHz Intel Pentium III or equivalent
  + Memory: 512 MB
  + Disk space: 750 MB of free disk space
  + Screen resolution is 1024x768 pixels.

#### Recommended Hardware Configurations

* + Processor: Intel Core i5 or equivalent
  + Memory: 2 GB (32-bit), 4 GB (64-bit)
  + Disk space: 1.5 GB of free disk space
  + Screen resolution is 1024x768 pixels

### FEASIBILTY STUDY:-

##### (7.1) INTRODUCTION:-

Feasibility is the measure of how beneficial or practical the development of the system will be to the organization. It is a preliminary survey for the systems investigation. It aims to provide information to facilitate a later in-depth investigation.

The report produced at the end of the feasibility study contains suggestions and reasoned arguments to help management decide whether to commit further resources to the proposed project.

Within the scheduled duration we were assigned to study both the positive and negative aspects of the current manual system, in which we have come up with a number of drawbacks that prevent the progress of the clinic if it is continued to function manually.

Having gone through all measures of feasibility we report to the management to figure out if the objectives of the new system are met.

##### For e.g.: –

* Is the system within the budget allowed for it?
* Will the organization’s needs, be met by the new proposed system as originally envisaged?

##### (7.2) TYPES OF FEASIBILITY:

There are various measures of feasibility that helps to decide whether a particular project is feasible or not. These measures include:

* + Technical Feasibility
  + Operational Feasibility
  + Economical Feasibility

Each of these types will be explained in detail throughout the project report.

##### (7.2.1) TECHNICAL FEASIBILITY

Based on the outline design of system requirements in terms of inputs, outputs, files, procedures and staff, the technical issues raised during technical feasibility include:

* Does the necessary technology exist to do what is proposed?
* Does the proposed equipment have the technical capacity to hold the data required to use in the new system?
* Adequate responses provided by the proposed system?
* Is the system flexible enough to facilitate expansion?
* Is there any technical guarantee of accuracy, reliability, ease of access and data security?

##### (7.2.2) OPERATIONAL FEASIBILITY:-

A system often fails if it does not fit within existing operations and if users resist the change.

Important issues a systems developer must look into are:

* Will the new system be used if implemented in an organization?
* Are there major barriers to implementation or is proposed system accepted without destructive resistance?

If we are considering the performance and response time for each task, it is very much faster since there is less paper work to be completed. When entering data into the system to relieve the user from additional work and typing incorrect data, the system provides options such as combo boxes, check boxes, option buttons and etc. if the users type in incorrect data they would be informed immediately about the error by the error detection control.

Another important fact to be regarded is the security control, which is handled by the system. Since data regarding each user is confidential, security is a key issue. Here, in this system, data regarding users is stored in database which can only be accessed by the authorized administrator.

The new system is more user-friendly, which enables the end-user to complete his/her work efficiently and accurately with interest. After taking the above fact into consideration we can state the operating of the proposed system within the organization is feasible.

##### (7.2.3) ECONOMICAL FEASIBILITY:-

The proposed system must be justifiable in terms of cost and benefit, to ensure that the investment in a new/changed system provide a reasonable return.

Even though finding out the costs of the proposed project is difficult we assume and estimate the costs and benefits as follows.

According to the computerized system we propose, the costs can be broken down to two categories:

1. Costs associated with the development of the system.
2. Costs associated with operating the system.

##### (7.3) S/W DEVELOPMENT LIFE CYCLE:

Every software development consists several phases, have certain predefined works and at the end of each phase document is prepared. This phase is based on certain Software Development Model.

##### Software Development Model :

Software engineering is a discipline that integrates process, methods, and tools for the development of computer software. To solve actual problems in an industry setting, software engineer or a team of software engineers must incorporate a development strategy that encompasses, methods, and tools. This strategy is often referred to as a process model or a software-engineering paradigm.

A number of different process models for the software engineering have been proposed, each exhibiting strengths and weaknesses, but all having a series of generic phases in common. Some of the commonly used software process models are:

* The linear sequential model
* The prototyping model
* The RAD model
* The incremental model
* The spiral model

A particular process model for software engineering is chosen on the nature of the project and the application at hand, the methods and the tools to be used, and the controls are required.

All software development can be characterized as a problem-solving loop in which four distinct stages are encountered:

* + Status Quo
  + Problem Definition
  + Technical Development
  + Solution Integration

##### Software Requirement Analysis:

The requirements gathering process is intensified and focused specifically on software. To understand the nature of the program(s) to be built, the software engineer (“analyst”) must understand the information domain for the software, as well as required function, behavior, performance, and interfacing. Requirements for the both the system and the software are documented and reviewed with the customer.

##### Design:

Software design is actually a multi-step process that focuses on four distinct attributes of a program: data structures, software architecture, interface representations, and procedural (algorithm) detail. The design process translates requirement into a representation of the software that can be assessed for quality before code generation begins.

##### Code Generation:

The design must be translated in to a machine-readable form. The testing process focuses on the logical internals of the software, assuring that all statements have been tested and on the functional externals that is, conducting tests to uncover errors and ensure that defined inputs will produce actual results that agree with required results.

##### Testing:

Once code has been generated, program testing begins. The testing process focuses on the logical internals of the software, assuring that all statements have been tested and on the

functional externals that is, conducting tests to uncover errors and ensure that defined inputs will produce actual results that agree with required results.

##### Maintenance

Software will undoubtedly undergo change after it is delivered to the customer (A possible exception is embedded software).Change will occur because errors have been encountered, because the software must be adapted to accommodate change in its external environment (e.g. A change required because of a new operating system or peripheral device), or because the customer requires functional or performance enhancements.

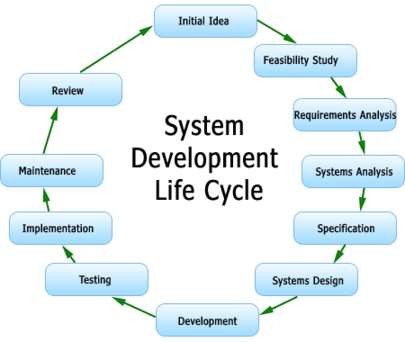


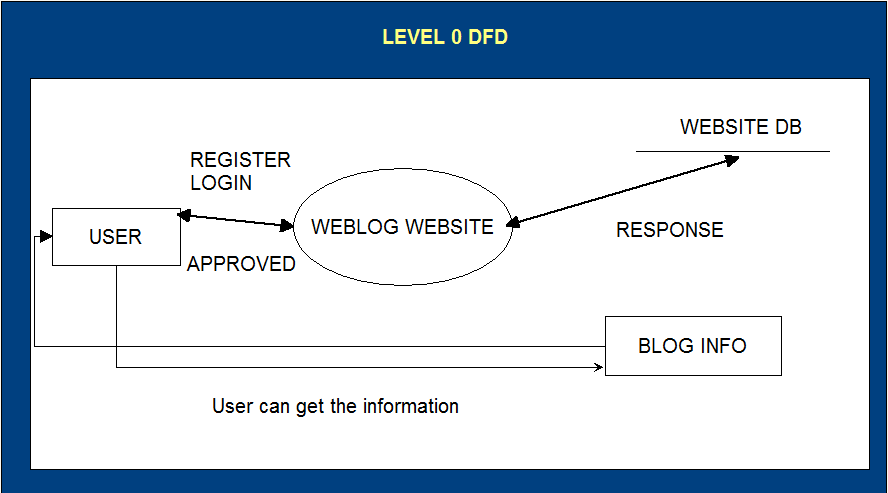
Fig. shows SDLC phases

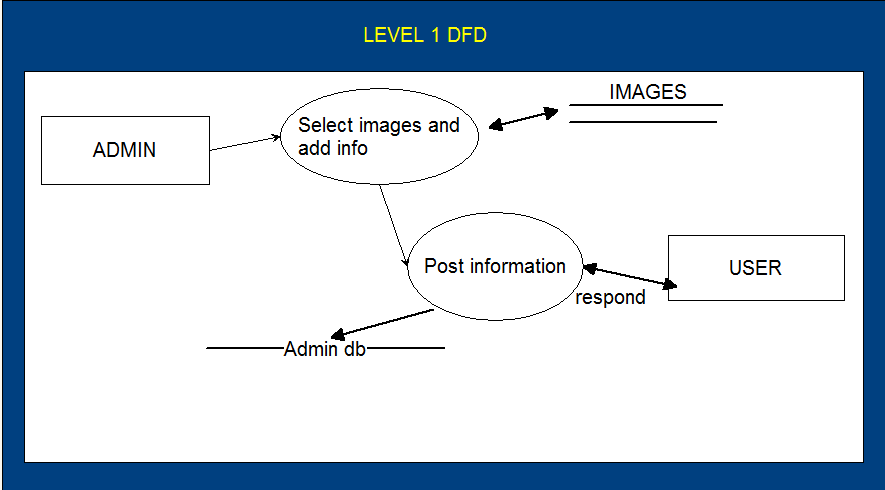
## (7.4) DATA FLOW DIAGRAM (DFD):-

Data flow diagrams model the flow of data into, through, and out of an information system:

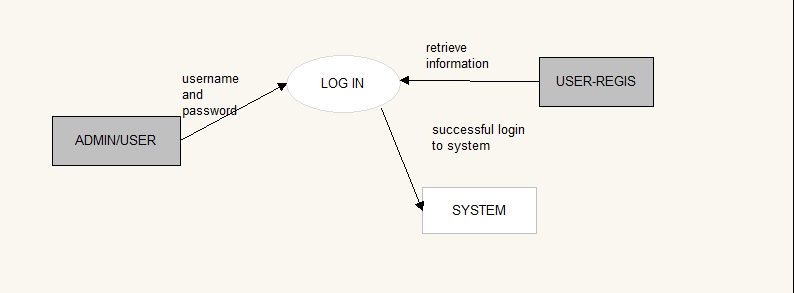
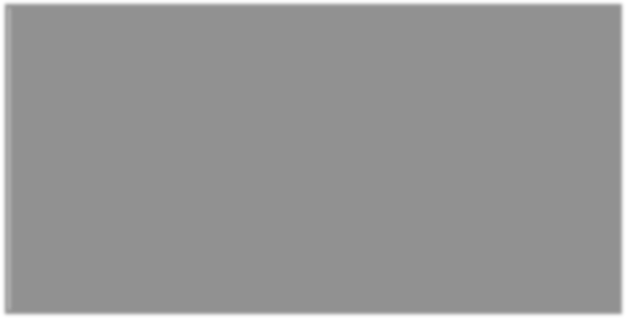
* Show the processes that change or transform data
* Show the movement of data between processes
* Represent a system as a network of processes which transform data flowing between them

#### CONTEXT DATA FLOW DIAGRAM

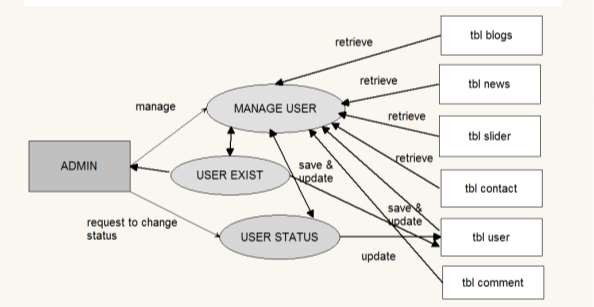
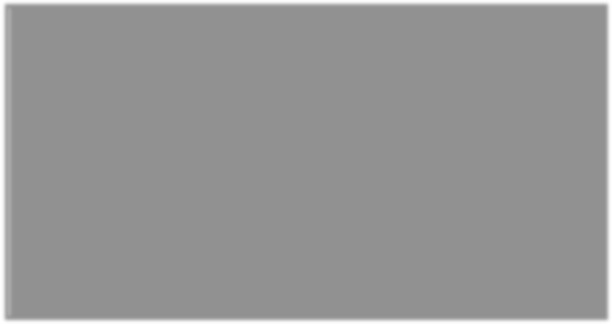




**LOGIN**



**MANAGE USER**



##### System Requirements

The only link to an external system is the link to the MySQL Database to verify the membership of a Customer and to store all the data provided by the customer. The Admin believes that a member is much more likely to be an effective reviewer and has imposed a membership requirement for a Reviewer. The Database fields of interest to the Admin are member’s name, membership (ID) number, contact numbers and e-mail addresses. The details of new members signing up for the events can be added in the list of the previous members and the amendments can be made to the list accordingly.

The requests of each customer can be inserted in the database so that the admin can process it for further use.

##### Clients, Customers, User

These are the ones who visits this particular website and thus has full access to the site and all the functionalities prevailing in it. The user can choose media and upload multiple files. Their session expires as the user selects a song for the background view. The user can select any festival or event according to his requirement.

##### Functional and Data Requirements

The requirements gathering process is intensified and focused specifically on software. To understand the nature of the program(s) to be built, the software engineer (“analyst”) must understand the information domain for the software, as well as required function, behavior, performance, and interfacing. Requirements for the both the system and the software are documented and reviewed with the customer.

##### Non-Functional Requirements

The software developed here assumes the use of PHP for connection between the Front End and the database. The speed of the User’s connection will depend on how fast they approach the site. The Admin will run the users requests and will have an access to database.

* **Reliability**- The system is reliable i.e. it’s well trusted.
* **Security**- The system is well secured, i.e. admin and user both have their unique ID and password to login into the system.
* **Maintainability**- Our project is well maintained with all required features involved.
* **Portability**- The system is portable i.e. it can work on any other system of the admin.
* The only requirement is the Internet connection.

##### Use Case Diagram

Use cases are used during the analysis phase of a project to identify system functionality. They separate the system into actors and use cases. Actors represent roles that are played by users of the system. Users may be humans, other computers, or even other software system.

##### Customer Use Case

**Use Case:** Upload Data

##### Brief Description:

The user approaches the site and retrieve data of his choice.

##### Diagram:



Retrieve Data



User

**Use Case:** Choose Media

##### Brief Description:

The customer submits his chosen data to the database for the implementation purpose.

##### Diagram:



Choose Media



User

**Admin Use Case:**

The admin has the following set of use cases:

* + - * Log In
      * View Requests
      * Process Requests
      * Access Database



Access Db

Access

Comments



Admin

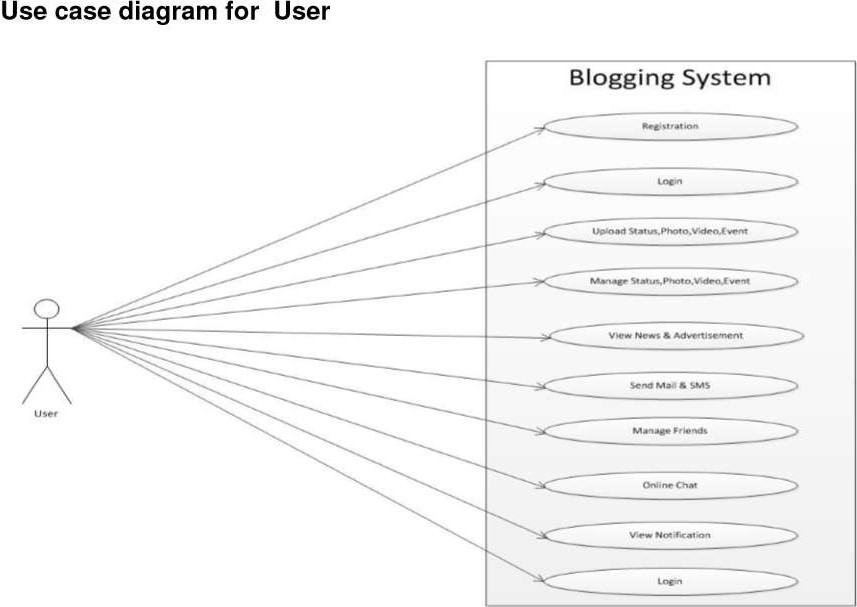


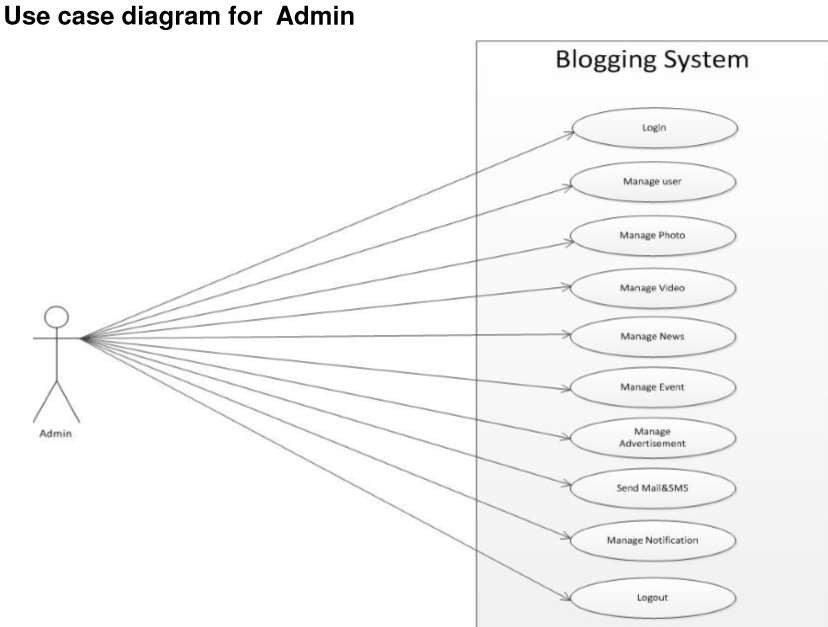
Process

View

Request

Log In





##### Object Oriented Analysis

Object–Oriented Analysis (OOA) is the procedure of identifying software engineering requirements and developing software specifications in terms of a software system’s object model, which comprises of interacting objects. In the system analysis or object-oriented analysis phase of software development, the system requirements are determined, the classes are identified and the relationships among classes are identified. The three analysis techniques that are used in conjunction with each other for object-oriented analysis are object modelling, dynamic modelling, and functional modelling.

**Object Modelling**

Object modelling develops the static structure of the software system in terms of objects. It identifies the objects, the classes into which the objects can be grouped into and the relationships between the objects. It also identifies the main attributes and operations that characterize each class.

The process of object modelling can be visualized in the following steps:

* + - * Identify objects and group into classes
      * Identify the relationships among classes
      * Create user object model diagram
      * Define user object attributes
      * Define the operations that should be performed on the classes

**Dynamic Modelling**

After the static behavior of the system is analyzed, its behavior with respect to time and external changes needs to be examined. This is the purpose of dynamic modelling.

Dynamic Modelling can be defined as “a way of describing how an individual object responds to events, either internal events triggered by other objects, or external events triggered by the outside world”.

The process of dynamic modelling can be visualized in the following steps:

* Identify states of each object
* Identify events and analyse the applicability of actions
* Construct dynamic model diagram, comprising of state transition diagrams
* Express each state in terms of object attributes
* Validate the state–transition diagrams drawn

**Functional Modelling**

Functional Modelling is the final component of object-oriented analysis. The functional model shows the processes that are performed within an object and how the data changes

as it moves between methods. It specifies the meaning of the operations of object modelling and the actions of dynamic modelling. The functional model corresponds to the data flow diagram of traditional structured analysis.

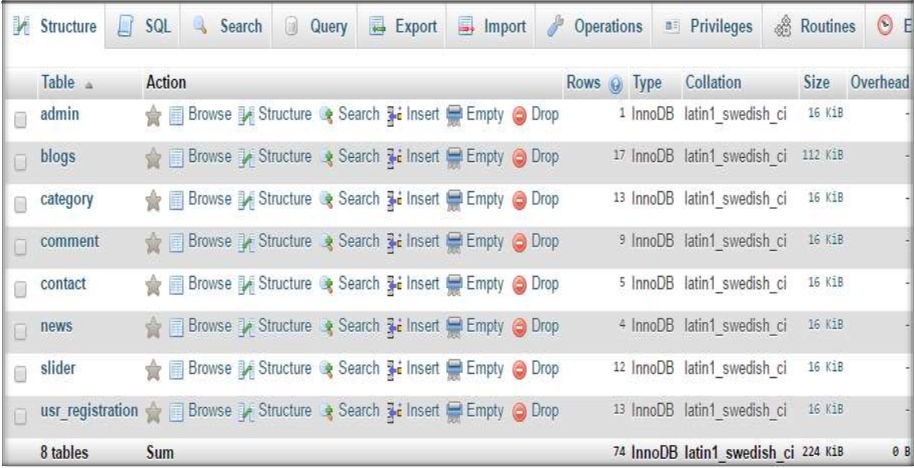
The process of functional modelling can be visualized in the following steps:

* Identify all the inputs and outputs
* Construct data flow diagrams showing functional dependencies
* State the purpose of each function
* Identify constraints
* Specify optimization criteria

## System Design

### Database

Data base is used to store the relevant information of the individuals. A database is a collection of rows and columns in which rows indicates the tuple and column indicates the domain of table. Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters. Need to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. 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 the logical design of the relation of the base data structures used to store the data. In the relational model these are the classes and named relationships. However, the term database design could also be used to apply to overall process of designing, not just the base data structure, but also the forms and queries used as part of the overall database application within the database management system (DBMS).

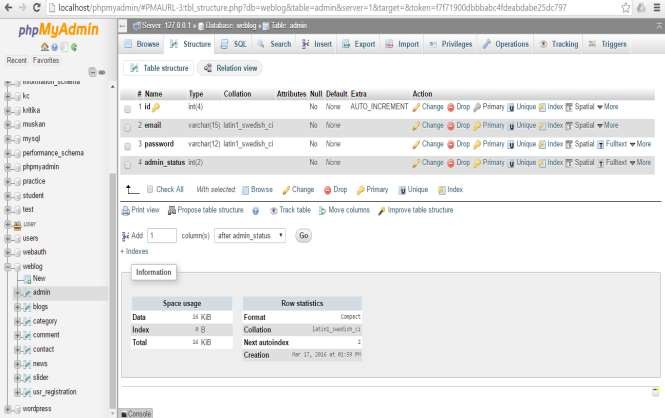
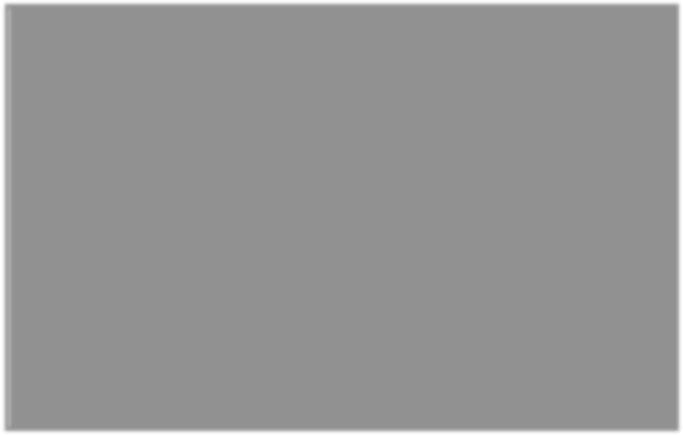


#### Fig. shows database of weblog

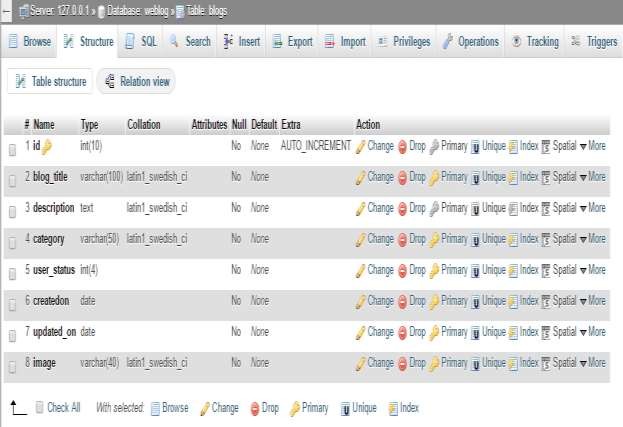
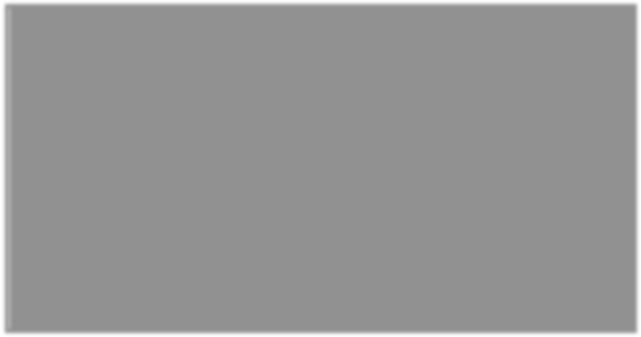
It includes eight tables namely

* Admin
* Blogs
* Category
* Comment
* Contact
* News
* Slider
* User registration.

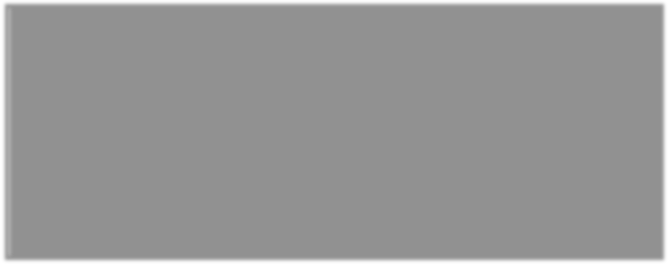
**ADMIN TABLE**



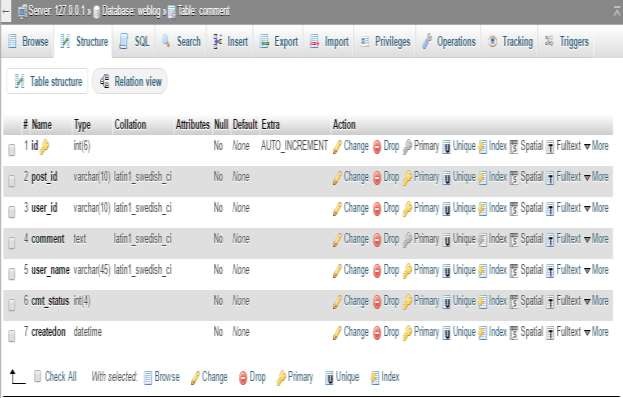
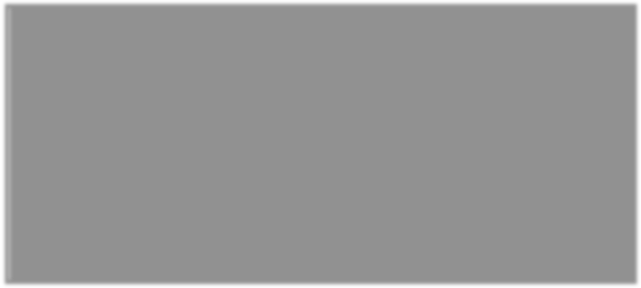
**BLOGS TABLE**



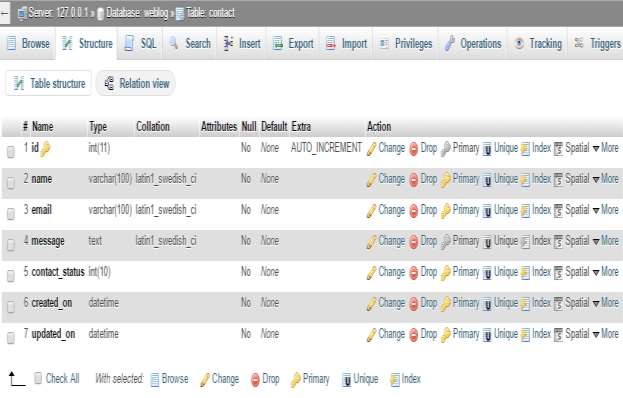
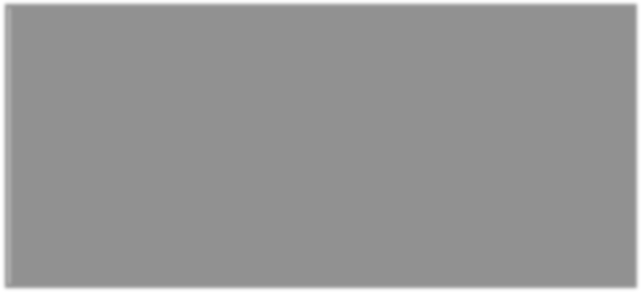
**CATEGORY TABLE**



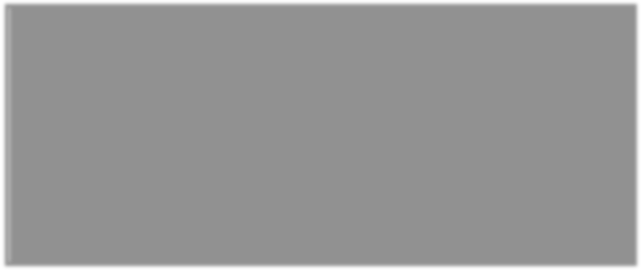
**COMMENT TABLE**



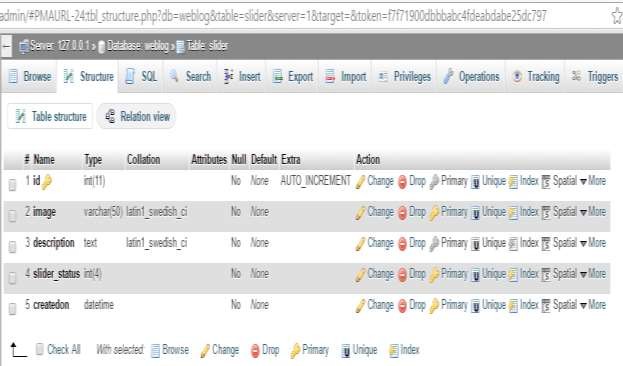
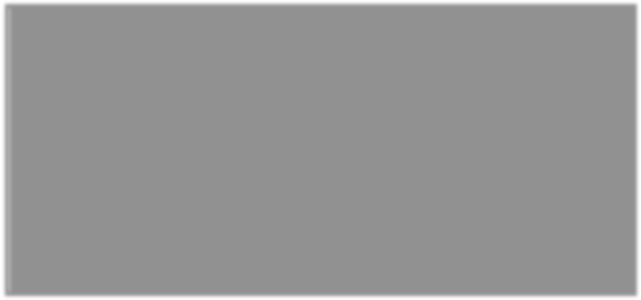
**CONTACT TABLE**



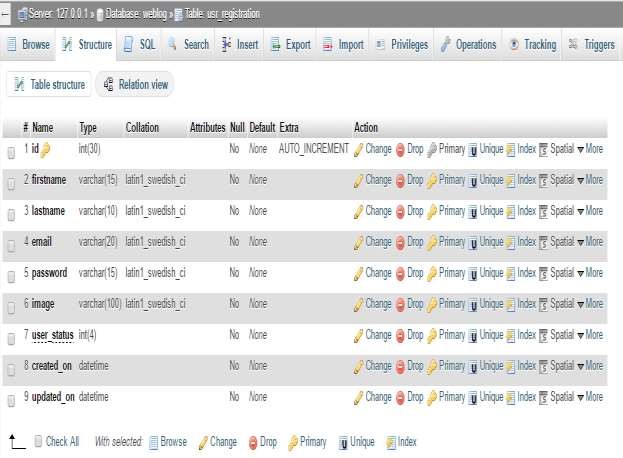
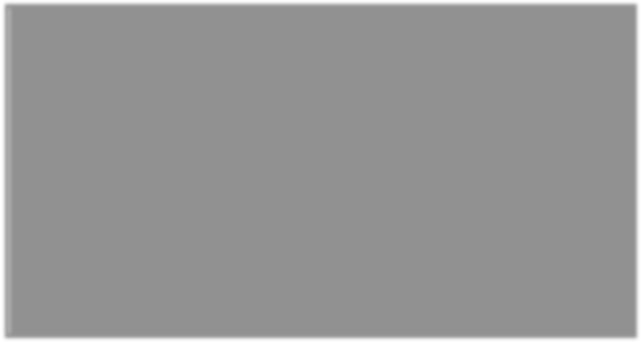
**NEWS TABLE**



**SLIDER TABLE**



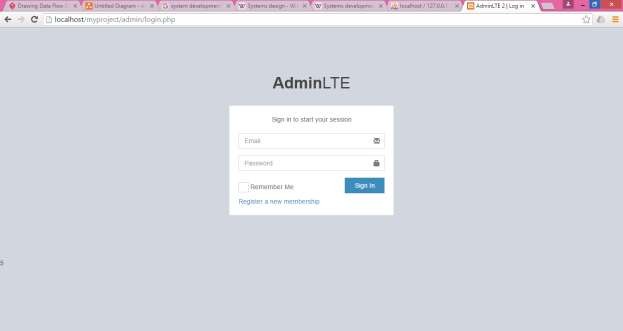
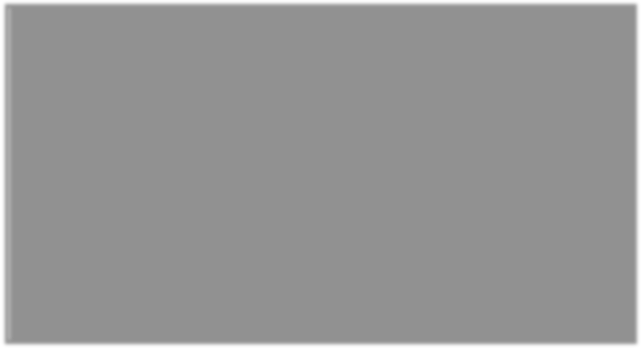
**USER-REGISTRATION TABLE**



* + - 1. **INPUT/OUTPUT**

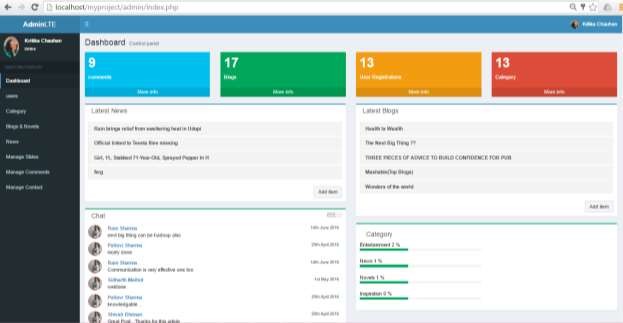
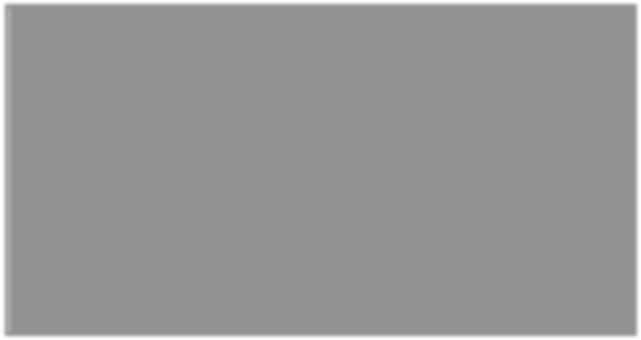
**(INTERFACE DESIGN)**

### ADMIN LOGIN PAGE



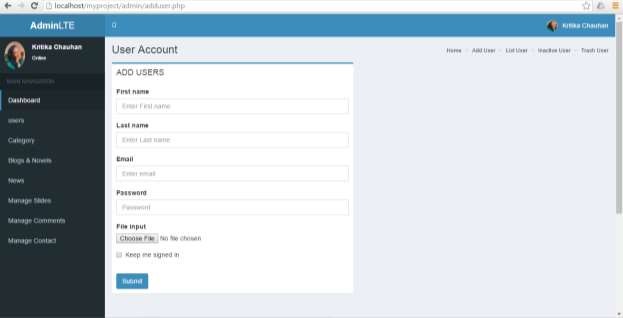
#### In this page admin LOGIN with valid username and password

**ADMIN HOME PAGE**



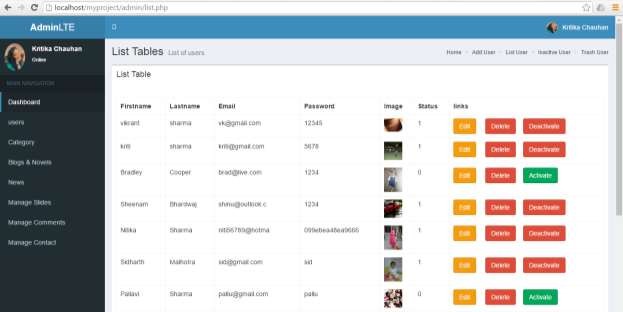
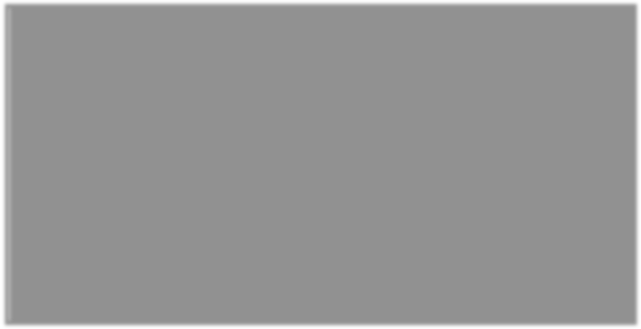
**In this admin page and user will redirect to this page after log in.**

**ADD USER**



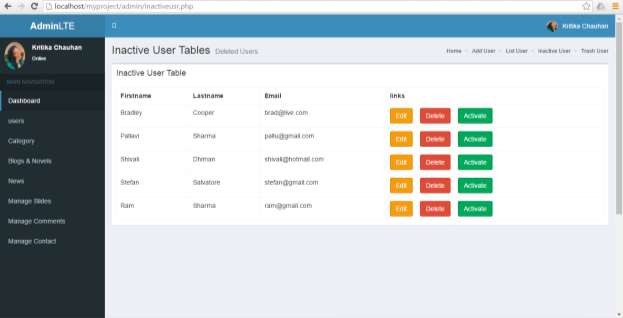
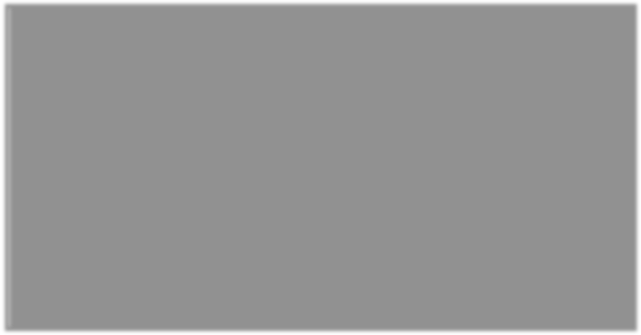
**In this admin can add user and fill the information**

**LIST USER**



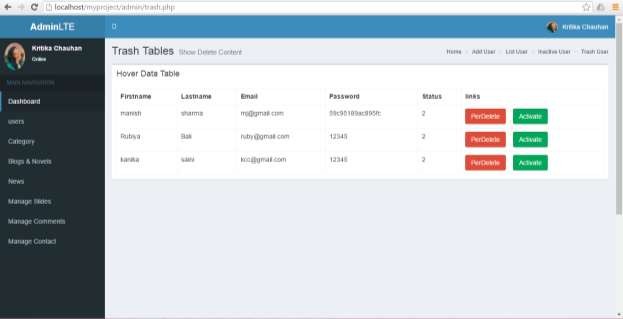
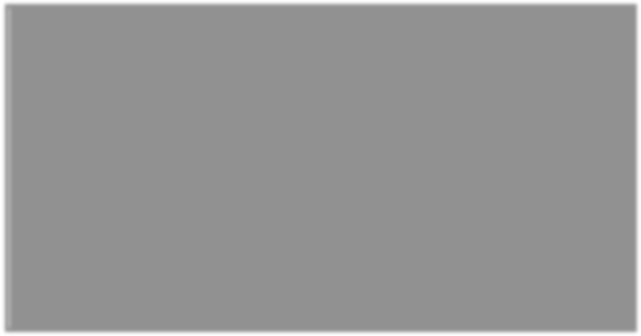
**This will show the list of user that are registered by user**

**INACTIVE USER PAGE**



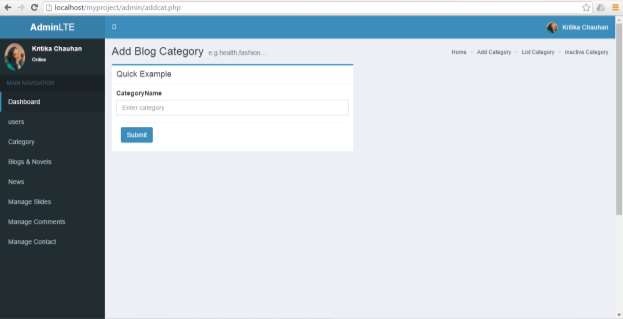
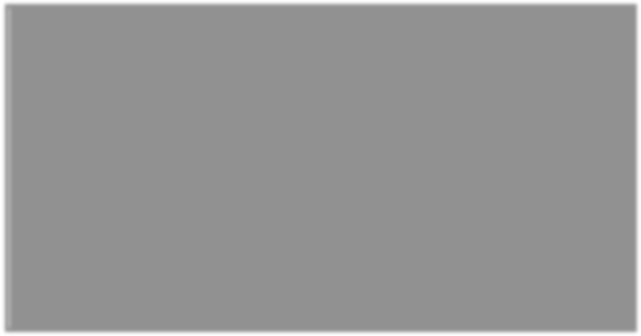
**In this page all deleted/deactivate user are shown**

**TRASH PAGE**



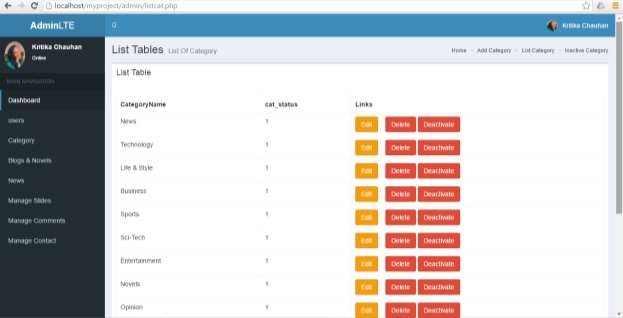
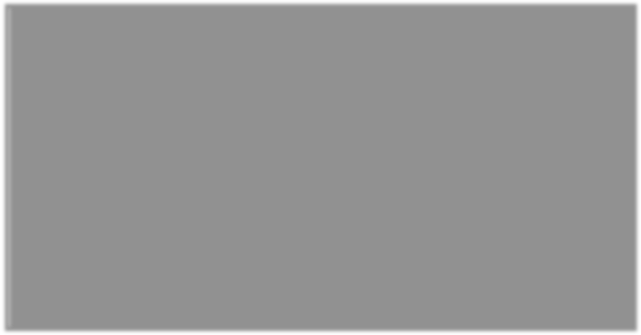
**In this Page it will show the deleted user and admin can activate its status to move to list user page**

**CATEGORY PAGE**



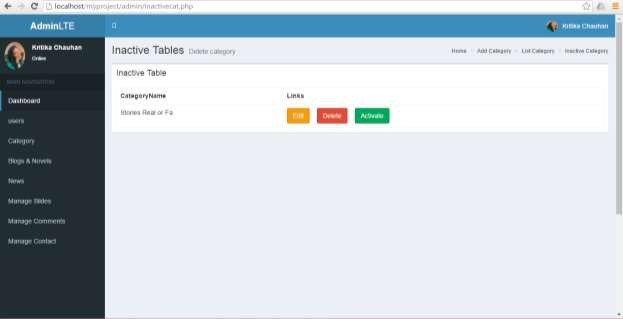
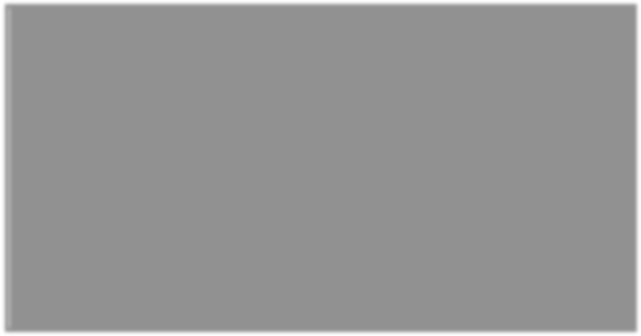
**In this page admin can add various categories and shown in add blog page**

**LIST CATEGORY PAGE**



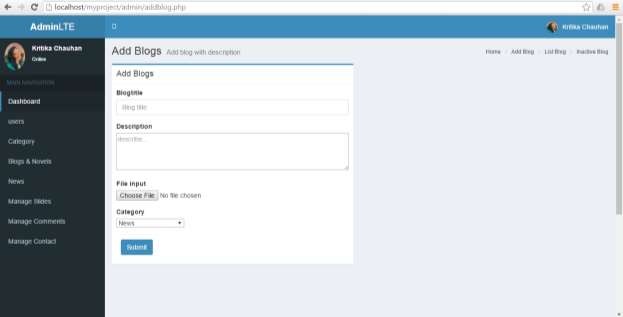
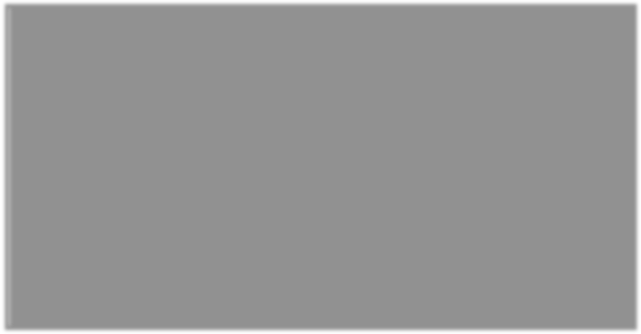
**In this page list of all the categories are shown**

**INACTIVE CATEGORY PAGE**



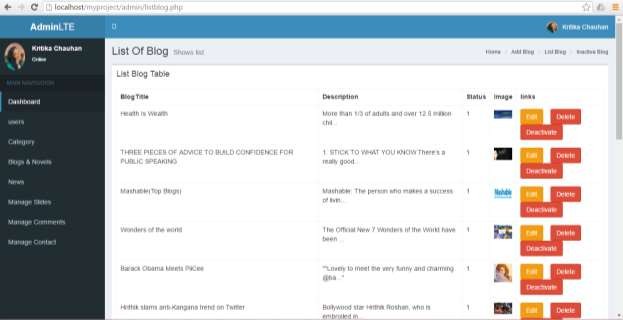
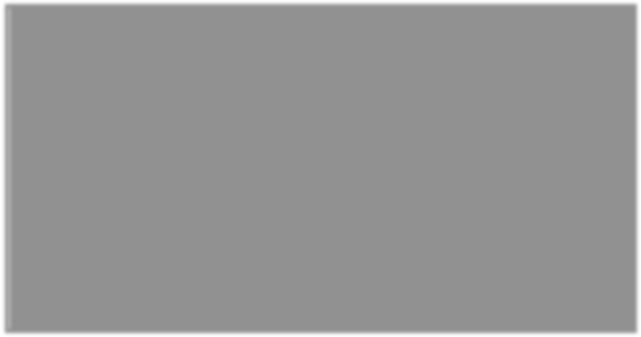
**In this page deleted category are shown**

**ADD BLOG PAGE**



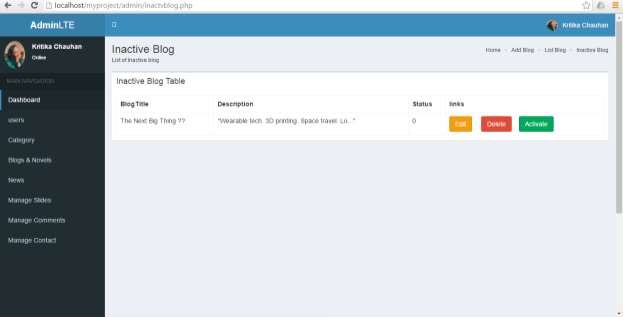
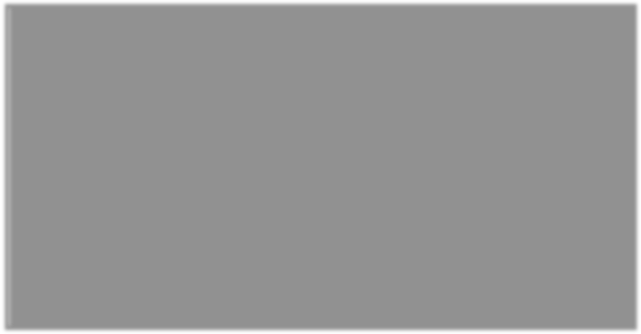
**In this admin can add blog and choose images from folder and select any category related to blog**

**LIST BLOG PAGE**



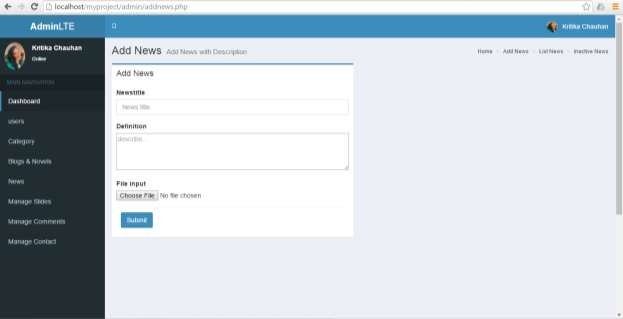
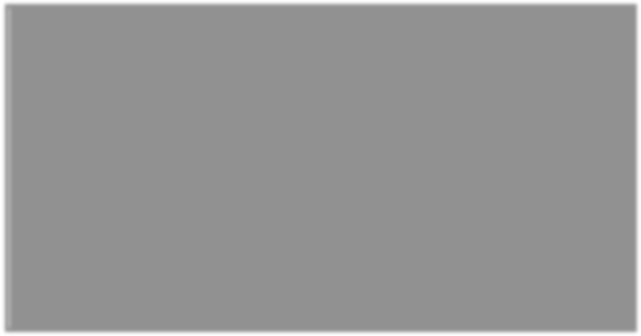
**In this page list of all the BLOGS are shown**

**INACTIVE BLOG**



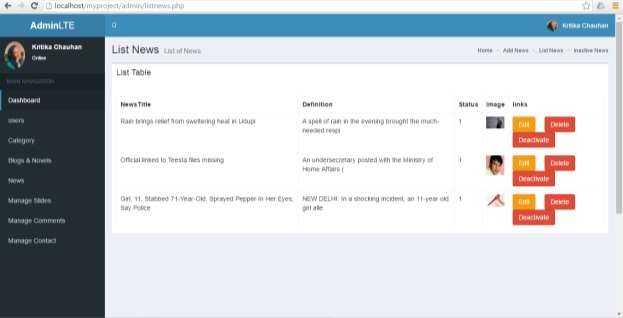
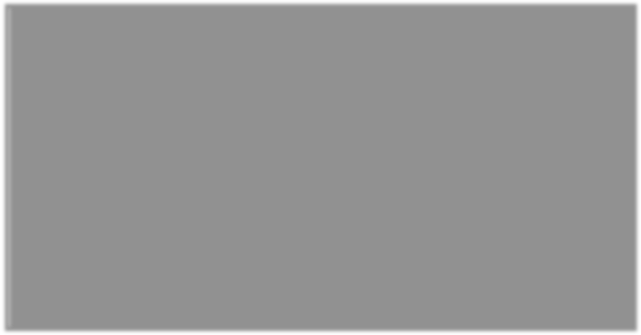
**In this page deleted category are shown**

**ADD NEWS PAGE**



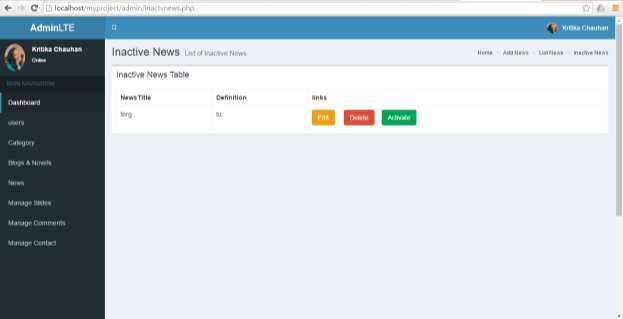
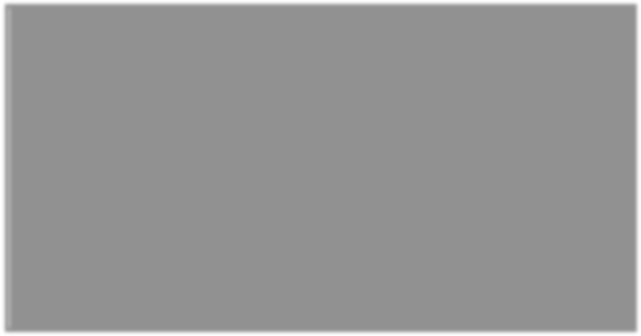
**In this admin can add NEWS and choose images from folder**

**LIST NEWS**



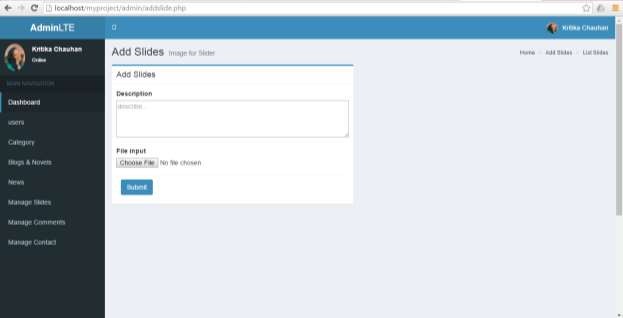
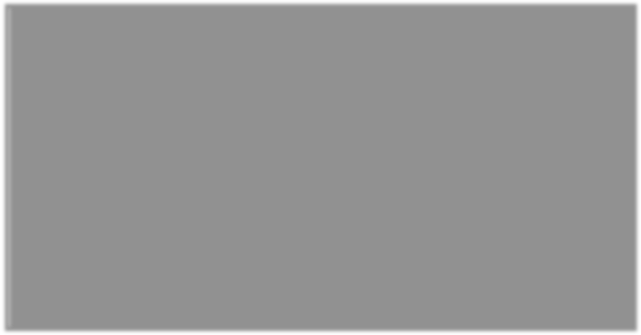
**In this page list of all NEWS are shown**

**INACTIVE PAGE OF NEWS SECTION**



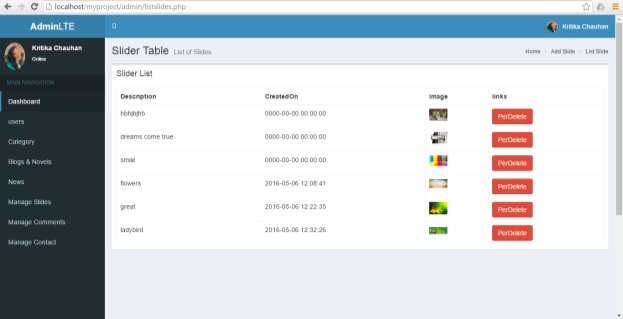
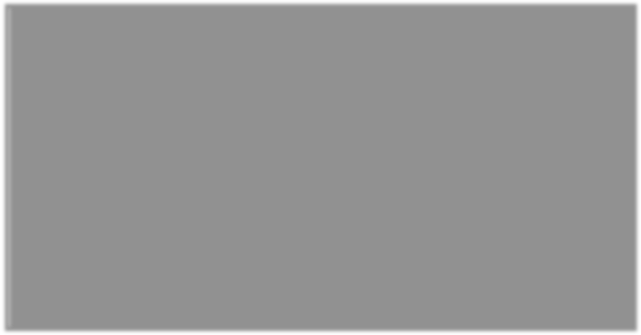
**In this page deleted NEWS are shown**

**ADD SLIDE PAGE**



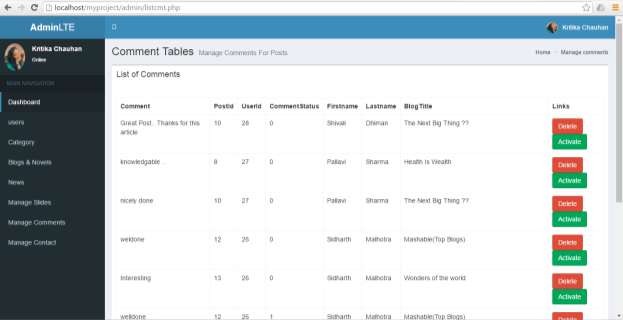
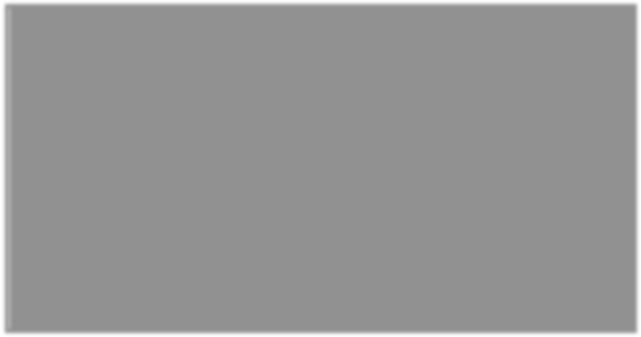
**In this page user can add images in the slider shown in top of page**

**LIST OF SLIDES**



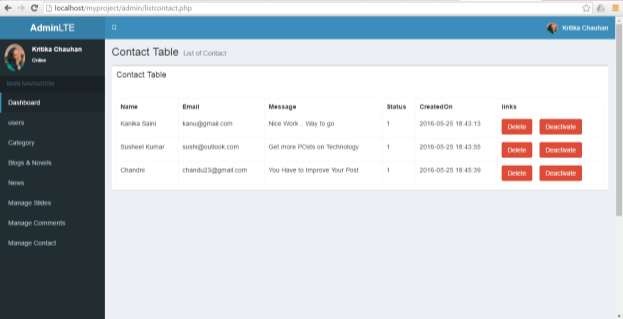
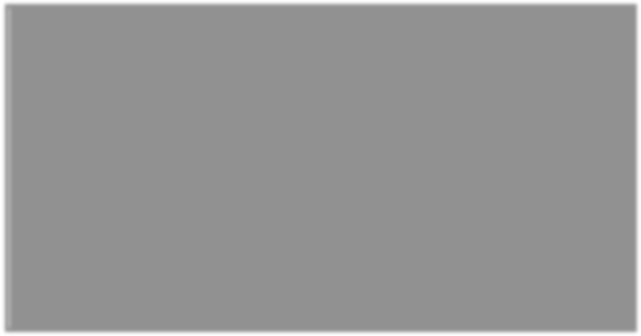
**It will show the slides description**

**COMMENT PAGE**



**In this page, it will manage the list of comments and join with user that are registered**

**CONTACT PAGE**



**This page shows list of contact to admin**

1. **SCREEN DESIGN (FRONT END)**

<?php include'include/connect.php'; include'include/header.php'; if($\_POST)

{

$email = $\_POST ['email'];

$password = $\_POST ['password'];

if($email != '' && $password != '')

{

$query = mysql\_query("SELECT \* FROM usr\_registration WHERE email='".$email."' AND password='".$password."'") or die(mysql\_error());

if(mysql\_num\_rows($query) > 0)

{

$row = mysql\_fetch\_array($query);

$data = array('email'=>$row['email'],'id'=>$row['id'],'name'=>$row['firstname'].' '.$row['lastname']);

$\_SESSION['myproject'] = $data; header("location: index.php");

}

else

{

}

}

header("location: index.php?status=error");

}

?>

<div class ="wrapper">

<div class= "container">

<div id="login" class="animate form">

<form method="post" action="" >

<h1 class = "head">Log in</h1>

<p>

<label for="username" class="uname" data-icon="u" > Your email or username </label>

<input id="username" name="email" required="required" type="text" placeholder="myusername or [mymail@mail.com](mailto:mymail@mail.com)"/>

</p>

<p>

<label for="password" class="youpasswd" data- icon="p"> Your password </label>

<input id="password" name="password" required="required" type="password" placeholder="eg. X8df!90EO" />

</p>

<p class="keeplogin">

<input type="checkbox" name="loginkeeping" id="loginkeeping" value="loginkeeping" />

<label

for="loginkeeping">Keep me logged in</label>

<p class="login button">

</p>

<input type="submit" value="submit" name="submit"/>

</p>

<p class="change\_link">

Not a member yet ?

<a href="usr\_regis.php" class="to\_register">Join us</a>

</p>

</form>

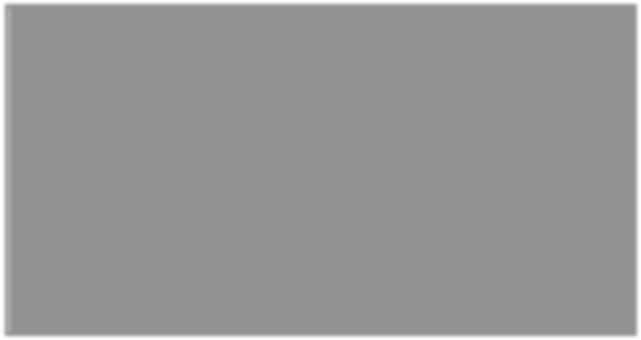
</div><!-- login ended-->

</div><!-- container ended-->

</div><!-- wrapper ended-->

<?php include'include/footer.php';?>

**LOGIN PAGE**



**USER\_REGISTRATION**

**CODING**

<?php include'include/connect.php'; ?>

<?php include'include/header.php';?>

<?php if(isset($\_POST['submit']))

{

$firstname=$\_POST['firstname'];

$lastname=$\_POST['lastname'];

$email=$\_POST['email'];

$password=$\_POST['password'];

$cpassword = $\_POST['cpassword'];

//echo $cpassword;

$error=0; if(empty($firstname))

{

$error=1;

$msg[]="plz enter your firstname";

}

if(empty($lastname))

{

$error=1;

$msg[]="plz enter your lastname";

}

/\*if(empty($dob))

{

$error=1;

$msg[]="plz enter your dob";

}

if(empty($city))

{

$error=1;

$msg[]="plz enter your city";

}

if(empty($address))

{

$error=1;

$msg[]="plz enter your address";

}

if(empty($phoneno))

{

}\*/

$error=1;

$msg[]="plz enter your phonenumber";

if(empty($email))

{

$error=1;

$msg[]="plz enter your email";

}

if(empty($password))

{

$error=1;

$msg[]="plz enter your password";

}

elseif($password != $cpassword)

{

$error=1;

$msg[]="password and confirm password not match!!";

}

if($error == 1)

{

foreach($msg as $val)

{

echo "<li>".$val."</li>";

}

}

$filename = $\_FILES['photo']['name'];

$size = $\_FILES['photo']['size'];

$type = $\_FILES['photo']['type'];

$tempname = $\_FILES['photo']['tmp\_name'];

// create random number

$random = rand('999','9999'); //4565

$nameexp = explode(".", $filename); // explode name by extension

$filename = $firstname."-".$random.".".$nameexp[1]; // create new name name14525.jpg

//checking formats

if($nameexp[1] == "jpeg" || $nameexp[1] == "jpg" || $nameexp[1] == "png"

|| $nameexp[1] == "JPG" || $nameexp[1] == "JPEG" || $nameexp[1] == "PNG" )

{

// check if file is uploaded

if(move\_uploaded\_file($\_FILES['photo']['tmp\_name'],'../uploads/'.$filename

))

{

$ins=mysql\_query("INSERT INTO

usr\_registration(firstname,lastname,email,password,created\_on,updated\_on,ima ge) VALUES

('".$firstname."','".$lastname."','".$email."','".$password."',NOW(),NOW(),'".$fi lename."') ") or die(mysql\_error());

}

else

{

}

echo "try again";

}

}

?>

<div class = "wrapper">

<div class="container">

<div id="register" class="animate form">

<form action="" autocomplete="on" method="post" enctype="multipart/form-data">

<h1 class ="head"> Sign up </h1>

<p>

<label for="usernamesignup" class="uname" data- icon="u">First Name</label>

<input id="usernamesignup" name="firstname" required="required" type="text" placeholder="mysuperusername690" />

</p>

<p>

<label for="usernamesignup" class="uname" data- icon="u">Last Name</label>

<input id="usernamesignup" name="lastname" required="required" type="text" placeholder="lastname" />

</p>

> Your email</label>

<p>

<label for="emailsignup" class="youmail" data-icon="e"

<input id="emailsignup" name="email"

required="required" type="email" placeholder="[mysupermail@mail.com](mailto:mysupermail@mail.com)"/>

</p>

<p>

<label for="passwordsignup" class="youpasswd" data- icon="p">Your password </label>

<input id="passwordsignup" name="password" required="required" type="password" placeholder="eg. X8df!90EO"/>

</p>

<p>

<label for="passwordsignup\_confirm" class="youpasswd" data-icon="p">Please confirm your password </label>

<input id="passwordsignup\_confirm" name="cpassword" required="required" type="password" placeholder="eg. X8df!90EO"/>

</p>

<p>

> Profile Pic</label>

<label for="emailsignup" class="youmail" data-icon="e"

<input id="emailsignup" name="photo" type="file"/>

</p>

<p class="signin button"> value="Sign up" name="submit"/>

<p class="change\_link"> Already a member ?

</p>

<input type="submit"

<a href="login.php" class="to\_register"> Go and log in </a></p>

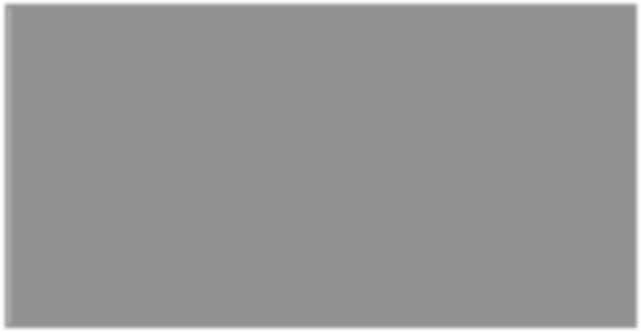
</form>

</div>

</div><!--container ended-->

</div><!--wrapper ended-->

<?php include 'include/footer.php';?>



**INDEX PAGE**

**CODING**

<?php include("include/connect.php"); include("include/header.php"); include("include/slider.php");

//$id=$\_GET[id]; if(isset($\_GET['p'])){

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

}else{

$page = 1;

}

$cur\_page = $page;

$page -= 1;

$per\_page = 2;

$previous\_btn = true;

$next\_btn = true;

$first\_btn = true;

$last\_btn = true;

$start = $page \* $per\_page;

$pagename = 'index.php';

?>

<div class="clear"></div>

<div id="right">

<?php

$pop=mysql\_query("SELECT \* FROM blogs LIMIT $start, $per\_page" ) or die (mysql\_error());

$count=mysql\_query("SELECT \* FROM blogs ") or die (mysql\_error());

if(mysql\_num\_rows($pop) > 0)

{$count1 = 1;

while($row=mysql\_fetch\_array($pop)) {

?>

<div class="content"><span class="headline\_one">

<a href="descrip.php?id=<?php echo $row['id'];?>"><?php echo

$row['blog\_title'];?></a>

</span><br />

<br />

<div class ="line"></div>

<span class="number"><?php echo $count1;?> </span>|<span class="date">On <?php echo date('jS F Y', strtotime($row['createdon']));?></span><br/>

<p class ="para"><?php echo substr($row['description'], 0, 800);?></p>

<?php if($row['image'] != ""){?>

<img src="uploads/<?php echo $row['image'];?>" style="float:left;width:100%;border-radius:20px;">

<?php }?>

<div class="comment\_box"><a href="descrip.php">Comments </a></div>

</div>

<?php $count1++;} ?>

<?php

$count = mysql\_num\_rows($count);

$no\_of\_paginations = ceil($count / $per\_page);

/\* ---------------Calculating the starting and endign values for the loop \*/

if ($cur\_page >= 7) {

$start\_loop = $cur\_page - 3;

if ($no\_of\_paginations > $cur\_page + 3)

$end\_loop = $cur\_page + 3;

else if ($cur\_page <= $no\_of\_paginations && $cur\_page

> $no\_of\_paginations - 6) {

$start\_loop = $no\_of\_paginations - 6;

$end\_loop = $no\_of\_paginations;

} else {

$end\_loop = $no\_of\_paginations;

}

} else {

$start\_loop = 1;

if ($no\_of\_paginations > 7)

$end\_loop = 7;

else

}

$end\_loop = $no\_of\_paginations;

/\* --------------------------------------------------------------------------

--------------------------------- \*/

@$msg .= "<div class='pagination'>";

// FOR ENABLING THE FIRST BUTTON

if ($first\_btn && $cur\_page > 1) {

$msg .= "<a href='$pagename?p=1' class='first'>First</a>";

} else if ($first\_btn) {

$msg .= "<div class='radiodesable'>First</div>";

}

</a>";

// FOR ENABLING THE PREVIOUS BUTTON

if ($previous\_btn && $cur\_page > 1) {

$pre = $cur\_page - 1;

$msg .= "<a href='$pagename?p=$pre' class='pre'> <<

} else if ($previous\_btn) {

$msg .= "<div class='radiodesable'><<</div>";

}

for ($i = $start\_loop; $i <= $end\_loop; $i++) {

if ($cur\_page == $i)

$msg .= "<div class='radiodesable'>$i</div>";

else

class='page456'>$i</a>";

}

$msg .= "<a href='$pagename?p=$i'

</a>";

// TO ENABLE THE NEXT BUTTON

if ($next\_btn && $cur\_page < $no\_of\_paginations) {

$nex = $cur\_page + 1;

$msg .= "<a href='$pagename?p=$nex' class='next'> >>

} else if ($next\_btn) {

$msg .= "<div class='radiodesable'>>></div>";

}

// TO ENABLE THE END BUTTON

if ($last\_btn && $cur\_page < $no\_of\_paginations) {

$msg .= "<a href='$pagename?p=$no\_of\_paginations' class='last'> Last </a>";

} else if ($last\_btn) {

$msg .= "<div class='radiodesable'>Last</div>";

}

//$goto = "<input type='text' class='goto' size='1' /><input type='button' id='go\_btn' class='go\_button' value='Go'/>";

$total\_string = "<div class='pagi\_goto'><span

a='$no\_of\_paginations'>Page <b>" . $cur\_page . "</b> of

<b>$no\_of\_paginations</b></span>";

$msg = $msg . $total\_string . "</div>"; // Content for

pagination

?>

</div>

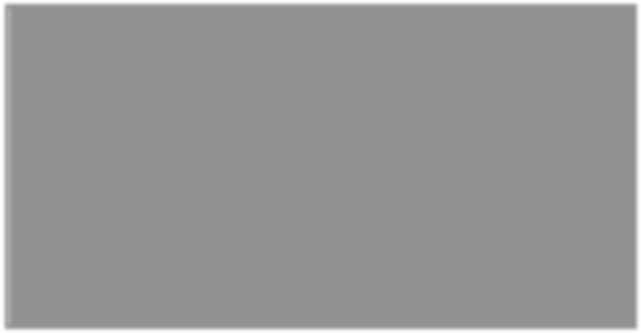
</div>

echo $msg;

}

<?php include("include/sidebar.php");?>

<?php include("include/footer.php"); ?>



**SIDEBAR PAGE**

**CODING**

<div id="left">

<form id="search-form" name="search" method="get" action="search.php">

<input id="search-input" name="s" type="text" placeholder="search here...">

</form>

<span class="headline\_three">Categories</span><br />

<ul class="category">

<?php $query =mysql\_query("SELECT \* FROM category")or die(mysql\_error()); ?>

<?php while($row =mysql\_fetch\_array($query)) {?>

<?php $id =$row['id'];?>

<?php $query1 =mysql\_query("SELECT id FROM blogs WHERE category='".$id."'")or die(mysql\_error());

$count =mysql\_num\_rows($query1);

// print\_r($count);?>

<li>

<a href="category.php?id=<?php echo $row['id'];?>">

<?php echo $row['category\_name']." (".$count.")";?>

</a>

</li>

<?php }?>

</ul>

<?php $query=mysql\_query("SELECT \* FROM Blogs ORDER BY RAND() LIMIT 0,5") or die(mysql\_error());?>

<span class="headline\_three">Recently Added</span><br />

<span class="small\_caps">Posts You can read</span>

<ul class ="added">

<?php while($row=mysql\_fetch\_array($query)) {

?>

<li>

<div class ="image">

<a href ="descrip.php?id=<?php echo $row['id'];?>">

<img src ="uploads/<?php echo $row ['image'];?>" alt="<?php echo $row['blog\_title'];?>"/>

<h3 ><?php echo $row['blog\_title'];?></h3>

</div>

</a>

</li>

<?php }?>

</ul>

</div>

**CATEGORY SECTION**

CODING

<?php include("include/connect.php"); include("include/header.php"); include("include/slider.php");

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

?>

<?php if(isset($\_GET['p'])){

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

}else{

$page = 1;

}

$cur\_page = $page;

$page -= 1;

$per\_page = 2;

$previous\_btn = true;

$next\_btn = true;

$first\_btn = true;

$last\_btn = true;

$start = $page \* $per\_page;

$pagename = 'category.php';

?>

<div class="clear"></div>

<div id="right">

<?php

$pop=mysql\_query("SELECT \* FROM blogs WHERE category ='".$id."' LIMIT $start, $per\_page" ) or die (mysql\_error());

$count = mysql\_query("SELECT \* FROM blogs WHERE category

='".$id."'") or die(mysql\_error()); if(mysql\_num\_rows($pop) > 0){

$count1 = 1; while($row=mysql\_fetch\_array($pop)){

?>

<div class="content"><span class="headline\_one">

<a href="descrip.php?id=<?php echo $row['id'];?>"><?php echo

$row['blog\_title'];?></a>

</span><br />

<br />

<div class ="line"></div>

<span class="number"><?php echo $count1;?> |</span> <span class="date">On <?php echo date('jS F Y', strtotime($row['createdon']));?><br

/>

<span class="small\_caps"> </span>

<p class ="para"><?php echo substr($row['description'], 0, 800);?></p>

<?php if($row['image'] != ""){?>

<img src="uploads/<?php echo $row['image'];?>" style="float:left;width:100%;border-radius:20px;">

<?php }?>

<div class="comment\_box"><a href="">Comments </a></div>

</div>

<?php $count1++;} ?>

<?php

$count = mysql\_num\_rows($count);

$no\_of\_paginations = ceil($count / $per\_page);

/\* ---------------Calculating the starting and endign values for the loop \*/

if ($cur\_page >= 7) {

$start\_loop = $cur\_page - 3;

if ($no\_of\_paginations > $cur\_page + 3)

$end\_loop = $cur\_page + 3;

else if ($cur\_page <= $no\_of\_paginations && $cur\_page

> $no\_of\_paginations - 6) {

$start\_loop = $no\_of\_paginations - 6;

$end\_loop = $no\_of\_paginations;

} else {

$end\_loop = $no\_of\_paginations;

}

} else {

$start\_loop = 1;

if ($no\_of\_paginations > 7)

$end\_loop = 7;

else

}

$end\_loop = $no\_of\_paginations;

/\* --------------------------------------------------------------------------

--------------------------------- \*/

@$msg .= "<div class='pagination'>";

// FOR ENABLING THE FIRST BUTTON

if ($first\_btn && $cur\_page > 1) {

$msg .= "<a href='$pagename?p=1' class='first'>First</a>";

} else if ($first\_btn) {

$msg .= "<div class='radiodesable'>First</div>";

}

</a>";

// FOR ENABLING THE PREVIOUS BUTTON

if ($previous\_btn && $cur\_page > 1) {

$pre = $cur\_page - 1;

$msg .= "<a href='$pagename?p=$pre' class='pre'> <<

} else if ($previous\_btn) {

$msg .= "<div class='radiodesable'><<</div>";

}

for ($i = $start\_loop; $i <= $end\_loop; $i++) {

if ($cur\_page == $i)

$msg .= "<div class='radiodesable'>$i</div>";

else

class='page456'>$i</a>";

}

$msg .= "<a href='$pagename?p=$i&id=$id'

</a>";

// TO ENABLE THE NEXT BUTTON

if ($next\_btn && $cur\_page < $no\_of\_paginations) {

$nex = $cur\_page + 1;

$msg .= "<a href='$pagename?p=$nex' class='next'> >>

} else if ($next\_btn) {

$msg .= "<div class='radiodesable'>>></div>";

}

// TO ENABLE THE END BUTTON

if ($last\_btn && $cur\_page < $no\_of\_paginations) {

$msg .= "<a href='$pagename?p=$no\_of\_paginations' class='last'> Last </a>";

} else if ($last\_btn) {

$msg .= "<div class='radiodesable'>Last</div>";

}

//$goto = "<input type='text' class='goto' size='1' /><input type='button' id='go\_btn' class='go\_button' value='Go'/>";

$total\_string = "<div class='pagi\_goto'><span

a='$no\_of\_paginations'>Page <b>" . $cur\_page . "</b> of

<b>$no\_of\_paginations</b></span>";

$msg = $msg . $total\_string . "</div>"; // Content for

pagination

?>

</div>

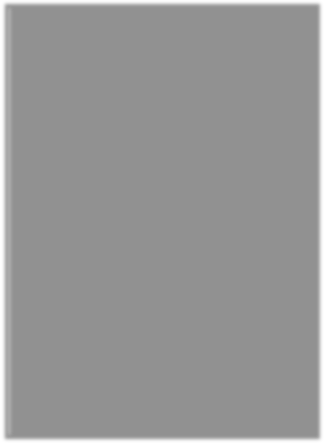
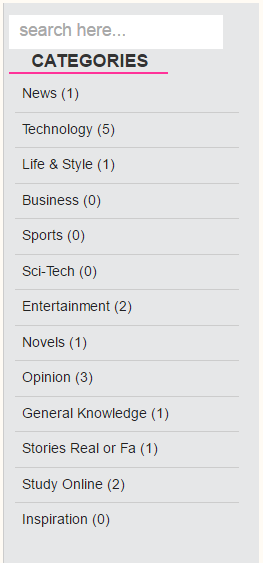
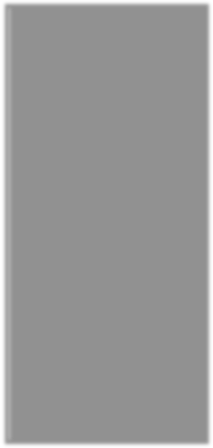
</div>

echo $msg;

}

<?php include("include/sidebar.php");?>

<?php include("include/footer.php"); ?>



## NEWS SECTION

#### CODING

<?php include("include/connect.php"); include("include/header.php"); include("include/slider.php");

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

?>

<?php if(isset($\_GET['p'])){

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

}else{

$page = 1;

}

$cur\_page = $page;

$page -= 1;

$per\_page = 2;

$previous\_btn = true;

$next\_btn = true;

$first\_btn = true;

$last\_btn = true;

$start = $page \* $per\_page;

$pagename = 'news.php';

?>

<div class="clear"></div>

<div id="right">

<?php

$pop=mysql\_query("SELECT \* FROM news LIMIT $start, $per\_page" ) or die (mysql\_error());

$count = mysql\_query("SELECT \* FROM news") or die(mysql\_error()); if(mysql\_num\_rows($pop) > 0)

{

$count1 = 1; while($row=mysql\_fetch\_array($pop)){

?>

<div class="content"><span class="headline\_one">

<a href="newsdescription.php?id=<?php echo $row['id'];?>"><?php echo

$row['news\_title'];?></a>

</span><br />

<br />

<div class ="line"></div>

<span class="number"><?php echo $count1;?> |</span> <span class="date">On <?php echo date('jS F Y', strtotime($row['created\_on']));?></span>

<span class="small\_caps"></span>

<p class ="para"><?php echo substr($row['definition'], 0, 800);?></p>

<?php if($row['image'] != ""){?>

<img src="uploads/<?php echo $row['image'];?>" style="float:left;width:100%;border-radius:20px;">

<?php }?>

</div>

<?php $count1++;} ?>

<?php

$count = mysql\_num\_rows($count);

$no\_of\_paginations = ceil($count / $per\_page);

/\* ---------------Calculating the starting and endign values for the loop \*/

if ($cur\_page >= 7) {

$start\_loop = $cur\_page - 3;

if ($no\_of\_paginations > $cur\_page + 3)

$end\_loop = $cur\_page + 3;

else if ($cur\_page <= $no\_of\_paginations && $cur\_page

> $no\_of\_paginations - 6) {

$start\_loop = $no\_of\_paginations - 6;

$end\_loop = $no\_of\_paginations;

} else {

$end\_loop = $no\_of\_paginations;

}

} else {

$start\_loop = 1;

if ($no\_of\_paginations > 7)

$end\_loop = 7;

else

}

$end\_loop = $no\_of\_paginations;

/\* --------------------------------------------------------------------------

--------------------------------- \*/

@$msg .= "<div class='pagination'>";

// FOR ENABLING THE FIRST BUTTON

if ($first\_btn && $cur\_page > 1) {

$msg .= "<a href='$pagename?p=1' class='first'>First</a>";

} else if ($first\_btn) {

$msg .= "<div class='radiodesable'>First</div>";

}

</a>";

// FOR ENABLING THE PREVIOUS BUTTON

if ($previous\_btn && $cur\_page > 1) {

$pre = $cur\_page - 1;

$msg .= "<a href='$pagename?p=$pre' class='pre'> <<

} else if ($previous\_btn) {

$msg .= "<div class='radiodesable'><<</div>";

}

for ($i = $start\_loop; $i <= $end\_loop; $i++) {

if ($cur\_page == $i)

$msg .= "<div class='radiodesable'>$i</div>";

else

class='page456'>$i</a>";

}

$msg .= "<a href='$pagename?p=$i'

</a>";

// TO ENABLE THE NEXT BUTTON

if ($next\_btn && $cur\_page < $no\_of\_paginations) {

$nex = $cur\_page + 1;

$msg .= "<a href='$pagename?p=$nex' class='next'> >>

} else if ($next\_btn) {

$msg .= "<div class='radiodesable'>>></div>";

}

// TO ENABLE THE END BUTTON

if ($last\_btn && $cur\_page < $no\_of\_paginations) {

$msg .= "<a href='$pagename?p=$no\_of\_paginations' class='last'> Last </a>";

} else if ($last\_btn) {

$msg .= "<div class='radiodesable'>Last</div>";

}

//$goto = "<input type='text' class='goto' size='1' /><input type='button' id='go\_btn' class='go\_button' value='Go'/>";

$total\_string = "<div class='pagi\_goto'><span

a='$no\_of\_paginations'>Page <b>" . $cur\_page . "</b> of

<b>$no\_of\_paginations</b></span>";

$msg = $msg . $total\_string . "</div>"; // Content for

pagination

?>

</div>

</div>

echo $msg;

}

<?php include("include/sidebar.php");?>

<?php include("include/footer.php"); ?>

**Link to the NEWS section : newsdescription.php**

<?php include("include/connect.php"); include("include/header.php");

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

?>

<style>

#cmnt\_data{width:100%; float:left;}

.success{color:green;}

.error{color:red;}

</style>

<head>

<script src="js/jquery-1.6.3.min.js"></script>

<!-- Include all compiled plugins (below), or include individual files as needed -

->

</head>

<div class="clear"></div>

<div id="right">

<?php

$query = mysql\_query("SELECT \* FROM news where id ='".$id."'") or die(mysql\_error());

$count = 1; while($row=mysql\_fetch\_array($query)){

?>

<div class="content"><span class="headline\_one">

<?php echo $row['news\_title'];?></span><br /><br />

<span class="number">.<?php echo $count;?> |</span> <span class="date">On <?php echo date('jS F Y', strtotime($row['created\_on']));?><br

/>

<span class="small\_caps">Information</span>

<p class ="para"><?php echo $row['definition'];?></p>

<img src="uploads/<?php echo $row['image'];?>" style="float:left;width:100%;border-radius:20px;">

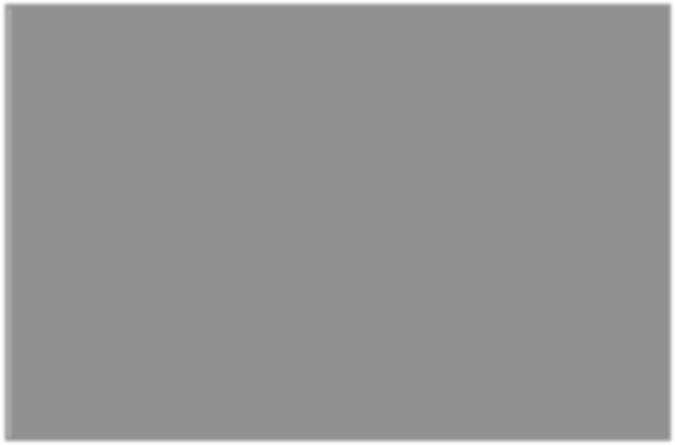
</div>

<?php $count++;} ?>

</div>

<?php include("include/sidebar.php");?>

<?php include("include/footer.php"); ?>



**CONTACT US PAGE**

**CODING**

<?php include'include/connect.php'; ?>

<?php include'include/header.php';

?>

<?php if(isset($\_POST['submit']))

{

$name=$\_POST['name'];

$email=$\_POST['email'];

$message=$\_POST['message'];

$error=0; if(empty($name))

{

$error=1;

$msg[]="Please enter your name";

}

if(empty($email))

{

$error=1;

$msg[]="Please enter your email";

}

if(empty($message))

{

}

else{

$error=1;

$msg[]="Please enter your message";

$ins=mysql\_query("INSERT INTO contact(name,email,message,contact\_status,created\_on,updated\_on) VALUES ('".$name."','".$email."','".$message."','1',NOW(),NOW())") or die(mysql\_error());

}

}

?>

<div class = "wrapper">

<div class="container">

<div id="register" class="animate form">

<form action="" autocomplete="on" method="post" enctype="multipart/form-data">

<h1 class ="head"> Contact Us </h1>

<p>

<label for="usernamesignup" class="uname" data- icon="u">Name</label><br>

<input id="usernamesignup" name="name" type="text"

placeholder="mysuperusername690" />

</p>

<p>

<label for="emailsignup" class="youmail" data-icon="e"

> Your email</label><br>

<input id="emailsignup" name="email" required="required" type="email" placeholder="[mysupermail@mail.com](mailto:mysupermail@mail.com)"/>

</p>

<p>

<label for="emailsignup" class="youmail" data-icon="e" > Message</label><br>

<textarea name="message"

rows="7" cols="30">

</textarea><br></p>

<p class="signin button"> value="Submit" name="submit"/>

</form>

</div>

<div class ="content">

</p>

<input type="submit"

<span class="headline\_one">Contact Info:</span>

<p id ="contact">D185, Phase 8B, Industrial Area,

Mohali, Punjab<br>

Email: [kritikachauhan19@gmail.com](mailto:kritikachauhan19@gmail.com)<br>

Contact No: +91 9056808418, +91 8628090521</p>

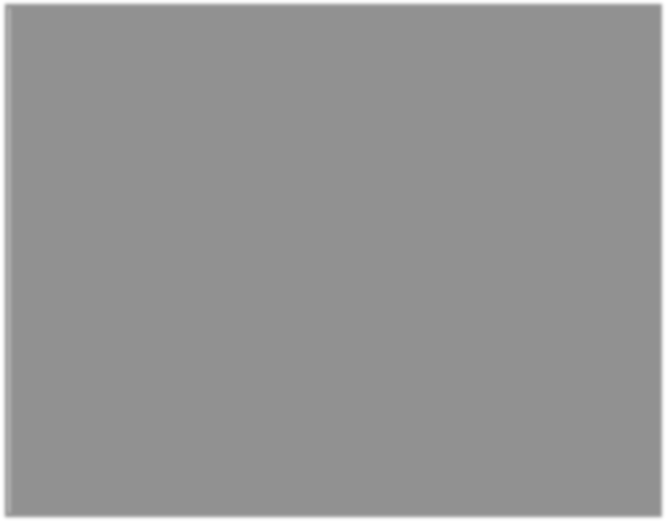
</div>

</div><!--container ended-->

</div><!--wrapper ended-->

<?php include'include/recntpost.php';?>

<?php include 'include/footer.php';?>



**ABOUT US PAGE**

**CODING**

<?php include("include/connect.php"); include("include/header.php"); include("include/recntpost.php");?>

<div id ="right">

<img src="uploads/blog.jpg" style="width:100%;border-radius:40px;"/>

<div class="content"><span class="headline\_one">About Us</span>

<p class ="para">WEBLOG is a combination of both Blog as well as Novels. Blog contain the Information of various things related to Technology, Education, News, International, Business, Sports, Entertainment and ongoing college activities. The main aim of this project is to provide data to students in only one site. Students can gather the information from one site as well as give their feedback and create their own blog. Students can post their views and thought and analyze themselves.</p>

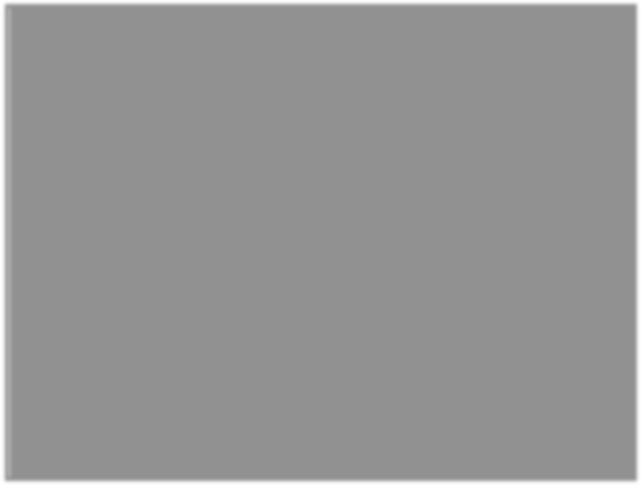
<p class="para"> The WEBLOG not only provide information related to many fields like Edu blogs, health, digital technology, latest news, trends and can give feedback and share his/her views regarding any topic. The WEBLOG have high quality content survive. The young generation are now moving from static content to dynamic content like flash and updating information like twitter. The WEBLOG provide the latest information, you just have to visit the specific blog of your interest and view the main page. The main page of website gives you latest information about the site content. The blogging is more taking attention to tech news like iPhone, iPad, laptops, iPod, mobiles and other technology news.

In the Novel section you can see the article of any blogger by clicking on author name. Weblogs are discussing about medical treatment and how to remain healthy and smart, so that mean you can write on other information also. People always looking for new techniques and words. WEBLOG gives visitors a new way of reading and not boring content. </p>

</div>

</div>

<?php include("include/footer.php"); ?>



1. **System Implementation**
   1. **Implementation**

Implementation is the stage in the project where the theoretical design is turned into the working system and is giving confidence to the new system for the users i.e. will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of method to achieve the changeover, an evaluation, of change over methods. A part from planning major task of preparing the implementation is education of users. The more complex system is implemented, the more involved will be the system analysis and design effort required just for implementation. An implementation coordinating committee based on policies of individual organization has been appointed. The implementation process begins with preparing a plan for the implementation for the system. According to this plan, the activities are to be carried out, discussions may regarding the equipment has to be acquired to implement the new system Implementation is the final and important phase. The most critical stage is in achieving a successful new system and in giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it found to working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain types of transaction while using the new system.

The major elements of implementation plan are test plan, training plan, equipment installation plan, and a conversion plan.

There are three types of implementation:

* + - Implementation of a computer system to replace a manual system.
    - Implementation of a new computer system to replace an existing system.
    - Implementation of a modified application to replace an existing one, using the same computer.
    - Successful implementation may not guarantee improvement in the organization using the new system, but improper installation will prevent it.

It has been observed that even the best system cannot show good result if the analysts managing the implementation do not attend to every important detail. This is an area where the systems analysts need to work with utmost care.

##### Implementation Phase

1. Training personnel
2. Conversion Procedures
3. Post-implementation review

##### Training Of Personal Involved With System

Even well designed system can succeed or fail because of the way they are operated and used. Therefore, the quality of training received by the personal involved with the system in various capacities helps or hinders and may even prevent the successful implementation of management information system.

**System Operators Training** Running of the system successfully depend on the personnel working in the Computer Centre. They are Responsible for providing the necessary support. Their training must ensure that they are able to handle all possible operations, both routine and extra-ordinary in nature. If the system calls for the installation of new equipment, such as new computer system, special terminals or different data entry machines, the operators training should include such fundamentals as how to turn the equipment on and use it, how to power off and a knowledge of what constitutes normal operations. The operators should also be trained on different type of malfunctioning, how to recognize them and what steps should also be taken whenever they arise.

##### User Training

User may be trained on use equipment, particularly in the case where, e.g. a microcomputer is in use and individual involved is both operator and user. In such cases, user must be given training on how to operate and user. In such cases, user must be given training on how to operator the system also. Questions that may be trivial to the analyst, such as how to turn on a terminal, how to insert a diskette into a micro-computer or when it is safe to

turn off equipment without danger of data loss are significant problems to new users who are not familiar. In most of the cases user training deals with the operation of the system itself, with proper attention given to data handling techniques. It is imperative that users be properly trained in methods of entering transaction, editing data, formulating inquiries, deleting and inserting of records. No training is complete without familiarizing users with simple systems maintenance activities. Weakness in any aspect of training may lead of awkward situation that creates user frustration and error.

##### Conversion Methods

A conversion is the process of changing from the old system to the new one. It must be properly planned and executed. Four methods are common in use. They are Parallel Systems, Direct Conversion, Pilot System and Phase In method. Each method should be considered in the light of the opportunities that it offers and problems that it may create. In general, system conversion should be accomplished in shortest possible time. Long conversion periods create problems for all persons involved including both analysts and users.

##### Parallel System

The most secure method of converting from an old to new system is to run both systems in parallel. This method is safest one because it ensures that in case of any problem in using new system, the organization can still fall back to the old system without the loss of time and money.

The Disadvantages Of Parallel Systems Approach Are:

1. It doubles operating costs.
2. The new system may not get fair trial.

##### Direct Conversion

This method converts from the old system to new system abruptly, sometimes over a weekend or even overnight. The old system is used until a planned conversion day, when it is replaced by the new system.

##### Pilot System

Pilot approach is often preferred in the case of the new system which involves new techniques or some drastic changes in organization performance. In this method, a working version of the system is implemented in one part of the organization, such as a single work area or department.

##### Phase –In- Method

This method is used when it is not possible to install a new system throughout an organization all at once. The conversion of files, training of personnel or arrival of equipment may force the staging of the implementation over a period of time, ranging from weeks to months.

##### Post Implementation Review

After the system is implemented and conversion is complete, a review should be conducted to determine whether the system is meeting expectations and where improvements are needed. A post implementation review measures the systems performance against predefined requirement. It determines how well the system continues to meet the performance specifications.

## 11. System Testing

### System Testing

System testing is a critical element of quality assurance and represents the ultimate review of analysis, design and coding. Test case design focuses on a set of techniques for the creation of test because that meet overall testing objective. When a system is developed it is hoped that it performs properly. The main purpose of testing an information system is to find the errors and correct them. The scope of system testing should include both manual and computerized operations. System testing is comprehensive evaluation of the programs, manual procedures, computer operations and controls.

System testing is the process of checking whether the developed system is working according to the objective and requirement. All testing is to be conducted in accordance to the test conditions specified earlier. This will ensure that the test coverage meets the requirements and that testing is done in a systematic manner.

##### TEST CHARACTERS :

1. A good test has a high probability of finding an error.
2. A good test is not redundant.
3. A good test should be “best of breed”.

**Table 11.1 Levels Of Testing**

|  |  |
| --- | --- |
| Client Needs | Acceptance Testing |
| Requirements | System Testing |
| Design | Integration Testing |
| Code | Unit Testing |

##### BLACK BOX TESTING:

The method of Black Box Testing is used by the software engineer to derive the required results of the test cases:

1. Black Box Testing alludes to test that are conducted at the software interface.
2. A Black Box Test examines some fundamental aspect of a system with little regard for the internal logic structure of the software.
3. A limited number of important logical paths can be selected and exercised. Black box testing was performed to find errors in the following categories:-
   * Incorrect or missing functions
   * Graphics error.
   * Errors in data in binary format.
   * Error in data in integer format.
   * File error.
   * Pointer error
   * Variable error

##### WHITE BOX TESTING:

White Box Testing is sometimes called Glass Box Testing. Using White Box Testing methods the software engineer can derive the following test cases:

1. Guarantee that all independent paths within a module have been exercised at least once.
2. Exercise all logical decisions on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.

In White Box Testing efforts were made to handle the following:-

* Number of input parameters equal to number of arguments.

•Parameters and arguments attributes match.

* Number of arguments transmitted is called modules equal to attributes of parameters.

•Unit system of argument transmitted is called modules equal unit system of parameter.

* Number of attributes and order of arguments to build in functions correct.

•Any references to parameters not associated to build in functions correct.

•Input only arguments altered.

•Global variable definition consistent across module.

•Files attributes correct.

##### Unit Testing

Code testing was carried out to see the correctness of the logic involved and the correctness of the modules. Tests were conducted based upon sample and live data as well. All the modules are checked separately for assuming correctness and accuracy in all the calculations.

##### Specification Testing

It examines the specification stating about what program should do and how it performs under various conditions. This testing strategy is better strategy since it focuses on the way the software is expected to work.

##### Integration Testing

The next level testing that was performed is often referred to as integration testing. During this phase many unit tested modules were combined into subsystems, which were then tested. The goal here was to see if modules can be integrated properly. Here the emphasis was on testing interfaces between different constituent modules of system.

##### Functionality Testing

Here the entire software system was tested. The reference document for this process is the requirements document, and the goal was to see if software solution meets its requirements. This level of testing is essentially a validation exercise, and in many situation it is the only validation activity.

##### Stress Testing

Proxy server developed for the specified purpose was testing under heavy load, i.e. a large no. of clients were made to sit in lab and were asked to send requests for logging in and then were asked to request for text on internet. System responded to request as desired.

##### Acceptance Testing

Acceptance was performed in the real environment with realistic data of the client to demonstrate if the software developed is working satisfactorily. Here the main focus was on the external behavior of the system; the internal logic of the program was not emphasized.

##### Test Data and Test Cases

The primary objective of test case design is to derive a set of tests that have the highest likelihood of uncovering errors in software. The test case specification is the major activity in the testing process. Careful selection of test cases that satisfy the criterion on approach specified is essential for proper testing. Various characteristics of test cases that are required for portal are:

* A good test has a high probability of finding an error.
* A good test is not redundant.
* A good test should be “Best of Breed”.
* A good test should be neither too simple not too complex.

##### Overview of Testing

1. **Testing:** Testing involves executing the program (or part of it) using sample data and inferring from the output whether the software performs correctly or not. This can be done either during module development (unit testing) or when several modules are combined (system testing).
2. **Direct Testing:** Defect testing is testing for situation where the program does not meet its functional specification. Performance testing tests a system's performance or reliability under realistic loads. This may go some way to ensuring that the program meets its non- functional requirements.
   1. **Debugging**

Debugging is a cycle of detection, location, repair and test. Debugging is a hypothesis testing process. When a bug is detected, the tester must form a hypothesis about the cause and location of the bug. Further examination of the execution of the program (possible including many returns of it) will usually take place to confirm the hypothesis. If the hypothesis is demonstrated to be incorrect, a new hypothesis must be formed. Debugging tools that show the state of the program are useful for this, but inserting print statements is often the only approach. Experienced debuggers use their knowledge of common and/or obscure bugs to facilitate the hypothesis testing process. After fixing a bug, the system must be reset to ensure that the fix has worked and that no other bugs have been introduced. This is called regression testing. In principle, all tests should be performed again but this is often too expensive to do.

##### Test Planning

Testing needs to be planned, to be cost and time effective. Planning is setting out standards for tests. Test plans set out the context in which individual engineers can place their own work. Typical test plan contains:

**Interface Testing:** Usually done at integration stage when modules or sub-systems are combined. Objective is to detect errors or invalid assumptions about interfaces between modules. Reason these are not shown in unit testing is that test case may perpetuate same incorrect assumptions made by module designer. Particularly important when OO development has been used.

##### Four Types Of Interface

1. Parameter: data (or occasionally function references) passed from one unit to another.
2. Shared memory: block of memory shared between units (e.g. global variable) .One places data there and the other retrieves it.
3. Procedural: Object-Oriented or abstract data type form of interface, encapsulating several procedures.
4. Message passing: one sub-system requests a service by passing a message. Client-server interface also used by some OO architectures.

##### Typical levels of testing:

Acceptance testing - whole system with real data (involve customer, user, etc) .Alpha testing is acceptance testing with a single client (common for bespoke systems). Beta testing involves distributing system to potential customers to use and provide feedback. **In, this project, Beta testing has been followed. This exposes system to situations and errors that might not be anticipated by us.**

## Maintenance

Once the website is launched, it enters the maintenance phase. All systems need maintenance. Maintenance is required because there are often some residual errors remaining in the system that must be removed as they are discovered. Maintenance involves understanding the effects of the change, making the changes to both the code and

the documents, testing the new parts and retesting the old parts that were not changed. Maintenance is mainly of two types:

1. Corrective Maintenance
2. Adaptive Maintenance

##### Corrective Maintenance

Almost all software that is developed has residual errors or bugs in them. Many of these surfaces only after the system have been in operation, sometimes for a long time. These errors once discovered need to be removed, leading to the software to be changed. This is called Corrective Maintenance.

##### Adaptive Maintenance

Even without bugs, software frequently undergoes change. The software often must be upgraded and enhanced to include more features and provide more services.

**Perfective Maintenance** - This includes modifications and updates done in order to keep the software usable over long period of time. It includes new features, new user requirements for refining the software and improve its reliability and performance.

**Preventive Maintenance** - This includes modifications and updating to prevent future problems of the software. It aims to attend problems, which are not significant at this moment but may cause serious issues in future.

**Cost of Maintenance**

Reports suggest that the cost of maintenance is high. A study on estimating software maintenance found that the cost of maintenance is as high as 67% of the cost of entire software process cycle.

On an average, the cost of software maintenance is more than 50% of all SDLC phases. There are various factors, which trigger maintenance cost go high, such as:

##### Real-world factors affecting Maintenance Cost

* + The standard age of any software is considered up to 10 to 15 years.
  + Older software’s, which were meant to work on slow machines with less memory and storage capacity cannot keep themselves challenging against newly coming enhanced software’s on modern hardware.
  + As technology advances, it becomes costly to maintain old software.
  + Most maintenance engineers are newbie and use trial and error method to rectify problem.
  + Often, changes made can easily hurt the original structure of the software, making it hard for any subsequent changes.
  + Changes are often left undocumented which may cause more conflicts in future.

##### Software-end factors affecting Maintenance Cost

* + Structure of Software Program
  + Programming Language
  + Dependence on external environment
  + Staff reliability and availability

**Maintenance Activities**

IEEE provides a framework for sequential maintenance process activities. It can be used in iterative manner and can be extended so that customized items and processes can be included.

These activities go hand-in-hand with each of the following phase:

* + **Identification & Tracing** - It involves activities pertaining to identification of requirement of modification or maintenance. It is generated by user or system may itself report via logs or error messages. Here, the maintenance type is classified also.
  + **Analysis** - The modification is analyzed for its impact on the system including safety and security implications. If probable impact is severe, alternative solution is looked for. A set of required modifications is then materialized into requirement specifications. The cost of modification/maintenance is analyzed and estimation is concluded.
  + **Design** - New modules, which need to be replaced or modified, are designed against requirement specifications set in the previous stage. Test cases are created for validation and verification.
  + **Implementation** - The new modules are coded with the help of structured design created in the design step.Every programmer is expected to do unit testing in parallel.
  + **System Testing** - Integration testing is done among newly created modules. Integration testing is also carried out between new modules and the system. Finally the system is tested as a whole, following regressive testing procedures.
  + **Acceptance Testing** - After testing the system internally, it is tested for acceptance with the help of users. If at this state, user complaints some issues they are addressed or noted to address in next iteration.
  + **Delivery** - After acceptance test, the system is deployed all over the organization either by small update package or fresh installation of the system. The final testing takes place at client end after the software is delivered.

Training facility is provided if required, in addition to the hard copy of user manual.

* + **Maintenance management** - Configuration management is an essential part of system maintenance. It is aided with version control tools to control versions, semi- version or patch management.

**Software Re-engineering**

When we need to update the software to keep it to the current market, without impacting its functionality, it is called software re-engineering. It is a thorough process where the design of software is changed and programs are re-written.

Legacy software cannot keep tuning with the latest technology available in the market. As the hardware become obsolete, updating of software becomes a headache. Even if software grows old with time, its functionality does not.

For example, initially Unix was developed in assembly language. When language C came into existence, Unix was re-engineered in C, because working in assembly language was difficult.

Other than this, sometimes programmers notice that few parts of software need more maintenance than others and they also need re-engineering.

##### Re-Engineering Process

* + **Decide** what to re-engineer. Is it whole software or a part of it?
  + **Perform** Reverse Engineering, in order to obtain specifications of existing software.
  + **Restructure Program** if required. For example, changing function-oriented programs into object-oriented programs.
  + **Re-structure data** as required.
  + **Apply Forward engineering** concepts in order to get re-engineered software. There are few important terms used in Software re-engineering

##### Reverse Engineering

It is a process to achieve system specification by thoroughly analyzing, understanding the existing system. This process can be seen as reverse SDLC model, i.e. we try to get higher abstraction level by analyzing lower abstraction levels.

An existing system is previously implemented design, about which we know nothing. Designers then do reverse engineering by looking at the code and try to get the design. With design in hand, they try to conclude the specifications. Thus, going in reverse from code to system specification.

##### Program Restructuring

It is a process to re-structure and re-construct the existing software. It is all about re- arranging the source code, either in same programming language or from one programming language to a different one. Restructuring can have either source code- restructuring and data-restructuring or both.

Re-structuring does not impact the functionality of the software but enhance reliability and maintainability. Program components, which cause errors very frequently can be changed, or updated with re-structuring.

The dependability of software on obsolete hardware platform can be removed via re- structuring.

##### Forward Engineering

Forward engineering is a process of obtaining desired software from the specifications in hand which were brought down by means of reverse engineering. It assumes that there was some software engineering already done in the past.

Forward engineering is same as software engineering process with only one difference – it is carried out always after reverse engineering.

**Component reusability**

A component is a part of software program code, which executes an independent task in the system. It can be a small module or sub-system itself.

##### Example

The login procedures used on the web can be considered as components, printing system in software can be seen as a component of the software.

Components have high cohesion of functionality and lower rate of coupling, i.e. they work independently and can perform tasks without depending on other modules.

In OOP, the objects are designed are very specific to their concern and have fewer chances to be used in some other software.

In modular programming, the modules are coded to perform specific tasks which can be used across number of other software programs.

There is a whole new vertical, which is based on re-use of software component, and is known as Component Based Software Engineering (CBSE).

Re-use can be done at various levels

* + **Application level** - Where an entire application is used as sub-system of new software.
  + **Component level** - Where sub-system of an application is used.
  + **Modules level** - Where functional modules are re-used.

Software components provide interfaces, which can be used to establish communication among different components.

##### Reuse Process

Two kinds of method can be adopted: either by keeping requirements same and adjusting components or by keeping components same and modifying requirements.

* + **Requirement Specification** - The functional and non-functional requirements are specified, which a software product must comply to, with the help of existing system, user input or both.
  + **Design** - This is also a standard SDLC process step, where requirements are defined in terms of software parlance. Basic architecture of system as a whole and its sub- systems are created.
  + **Specify Components** - By studying the software design, the designers segregate the entire system into smaller components or sub-systems. One complete software design turns into a collection of a huge set of components working together.
  + **Search Suitable Components** - The software component repository is referred by designers to search for the matching component, on the basis of functionality and intended software requirements..
  + **Incorporate Components** - All matched components are packed together to shape them as complete software.

## 13. How to Contact us & about us

##### Contact Us

If user want to contact us he/she can contact us through the contact us option in the main menu. Our E-mail id and contact no. is provided. User can contact us through any of the above links.

##### About Us

If user want to know anything about this website then he/she can know about us through the about us option in the main menu. Our website’s brief information is provided there.

## 14 Conclusion

While developing the system a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources – that would generate a proper system for cases.

While making the system, an eye has been kept on making it as user-friendly. As such one may hope that the system will be acceptable to any user and will adequately meet his/her needs. As in case of any system development process where there are a number of short comings, there have been some shortcomings in the development of this system also.