





**Project On  
Self Publishing  
Portal  
(Online Blogging System)**

**By  
Kaustubh Vijay Dalal  
Seat No:1811014  
MCA(2018-2021)**

**CERTIFICATE FROM GUIDE:**

## **ACKNOWLEDGMENT:**

My gratification and elation on the success of this project would be incomplete without mentioning the names of all the people who helped me with it and without whose guidance and encouragement this wouldn't have been successful.

I extend my sincere thanks to Dr.Jayashree Patil, Faculty Member, MES's Institute of Management and Career Courses,Pune, for having spared his valuable time with me and for all the guidance given in executing the project as per requirements.

I hope that I can build upon the experience and knowledge that I have gained and make a valuable contribution towards this industry in coming future.

Place:

Kaustubh Dalal

Date:

MCA(III)

## Project Index

Sr. No.	Name of Topic	Page No.
1	<b>Chapter 1 : Introduction</b>	
	1.1 Existing System and Need for System	10
	1.2 Scope of Work	11
	1.3 Operating Environment – Hardware and Software	13
	1.4 Detail Description of Technology Used	14
2	<b>Chapter 2 : Proposed system</b>	
	2.1 Proposed System	20
	2.2 Objectives of System	22
	2.3 User Requirements	23
3	<b>Chapter 3 : Analysis &amp; Design</b>	
	3.1 Object Diagram	25
	3.2 Class Diagram	26
	3.3 Use Case Diagrams	27
	3.4 Activity Diagrams	30
	3.5 Sequence Diagrams	33
	3.6 Entity Relationship Diagram	36
	3.7 Module Hierarchy Diagram	37
	3.8 Component Diagram	38
	3.9 Deployment Diagram	39
	3.10 Module Specifications	40
	3.12 Web Site Map Diagram	42
	3.13 User Interface Design	43
	3.14 Data Dictionary	48
	3.15 Table specifications	51
	3.16 Test Procedures and Implementation	55
4	<b>CHAPTER 4 : USER MANUAL</b>	
	4.1 User Manual	72
	4.2 Operations Manual / Menu Explanation	76
	4.3 Program Specifications / Flow Charts	79
5	<b>Drawbacks and Limitations</b>	83
6	<b>Proposed Enhancements</b>	84
7	<b>Conclusions</b>	85
8	<b>Bibliography</b>	86

9	<b>ANNEXURES :</b>	
	<b>ANNEXURE 1 : USER INTERFACE SCREENS</b>	
	<b>ANNEXURE 2 : OUTPUT REPORTS WITH DATA</b>	
	<b>ANNEXURE 3 : SAMPLE PROGRAM CODE</b>	

**CHAPTER 1**  
**INTRODUCTION**



## **Introduction:**

The "Self Publishing Portal" has been developed to override the problems prevailing in the practicing manual system. Self publishing portal , as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization in better utilization of resources.

Various Institutions & organization, whether big or small, has challenges to overcome and managing the information of Comment, Posts, New Posts, Comment, Technology Posts. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals.also for those busy executives who are always on the go,or system comes with remote access feature.

These System will ultimately allow you to better manage resources.

## **1.1 Existing System and Need for 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.

The other existing software are publicly accessible to anyone, there is no security for confidential data. In the past, students have generally written exclusively for their teachers. It's hard for them to get excited about writing something that they know no one else will see, on a subject that seems to serve no real purpose outside the classroom. The platform also has its darker side with some limitations on the enjoyment of its features and functionality.

Following are some available existing systems for blog:

- 1) *WordPress*:
- 2) *Drupal*
- 3) *Serendipity*

it allows people to express and hone their creativity

## **1.2 Scope of Work:**

### **About System scope:**

- Provides the searching facilities based on various factors. Such as Posts, New Category, New Posts, Technology Posts
- this project also manage the Comment details online for New Blog details, Technology Posts details, Posts.
- It tracks all the information of Comment, Comment, New Posts etc • Manage the information of Comment • Shows the information and description of the Posts, New Category
- To increase efficiency of managing the Posts, Comment
- it deals with monitoring the information and transactions of new posts
- manage the information of post
- Editing ,adding and updating of records is improved which result in proper resource management of post data.

### **User's Scope:**

- before any step initially a user can see System and able to read posts see gallery images, but cant perform any action
- After Registration/Login user create an account and enter in system and now user has full control over he/her work.
- User can manage, edit he/her profile.
- now user can able to perform operations on system like can add post , can see others posts, can make comment, can give likes, can add new images, can save his/her work as draft.
- another user don't have right to perform any operation on another's posts.
- here user is it self playing role of admin means user has its own dashboard where he can do all this operations.
- User create new posts and there are 2 options either publish or save as draft.
- User can see comments on his/her posts.
- Post contains category ,tags
- User can choose posts category which is setted by super admin
- User can manage add delete edit it's own posts
- User can add comment or give like to other posts and also gives feedback to system
- User can search particular posts in a system

**Admin's scope:**

1.Admin has it's own functionality admin is here we can say super admin

2.Like user he can perform all activities which user performs

3.Rather than this admin can do some more like below

- Admin can login in system
- Admin can manage users
- Admin can delete specific user
- Admin can make any Posts as editors choice
- Admin can make comment on any blog
- Admin can reply to users feedback
- Admin can maintain stats of all blogs on system means how many blogs of which category is present which are deleted etc.
- Admin can add new posts categories
- Admin can have full control over system
- Admin can delete specific users post if its is not good
- Admin can search particular posts in a system
- Admin can manage comments.
- Admin can like or rate any Post

### 1.3 Operating Environment – Hardware and Software:

#### Software Requirements:

Name of Components	Specifications
Operating System	Windows 10,8,7
Language	PHP
Database	MySQL DB Server
Browser	Any Mozilla,Opera,Chrome,etc
Web Server	Xampp
Scripting Languages	Html,css,js,jquery,Bootstrap.etc

#### Hardware Requirements:

Name of Components	Specifications
Operating System	Windows 10,8,7
Processor	Intel I3 core
RAM	4GB
HardDisk	1 TB

## **1.4 Detail Description of Technology Used:**

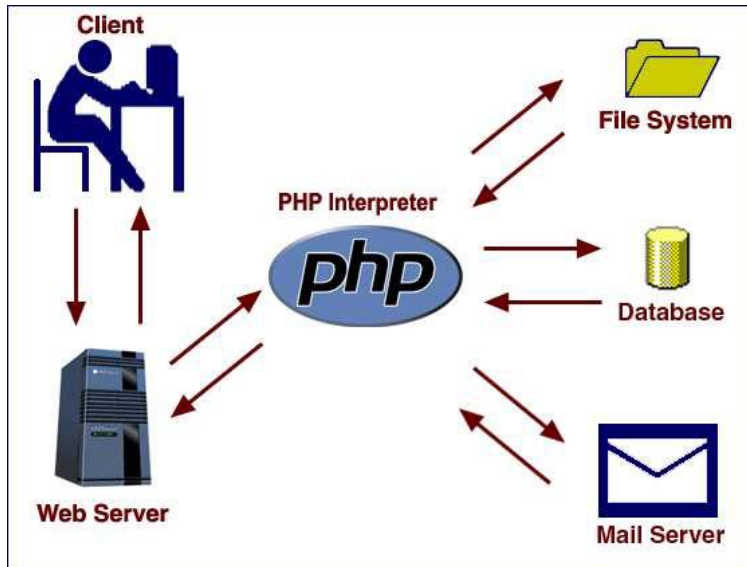
### **Platform:**

- a. **Front end:** HTML, CSS, JAVASCRIPT, JQUERY, Bootstrap.
- b. **Back end:** MYSQL.

### **1.4.1 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 HTML 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.



### 1.4.2 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

### 1.4.3 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

#### **1.4.4 XAMPP:-**

**XAMPP** is a free and open source cross-platform web server solution stack package developed by Apache Friends,[1] consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming languages.[2][3] 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.

#### **1.4.5 MYSQL:-:**

MySQL is an open-source 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 RDBMS. The SQL acronym stands for 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 open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL Perl/PHP/Python". Free-software open-source projects that require a full-featured database management system often use MySQL. Applications that use the MySQL database include: TYPO3, Joomla, WordPress, php BB, My BB, Drupal and othersoftware.

- 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.

#### **1.4.6 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 objectoriented, 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.

#### **1.4.7 Bootstrap:-:**

**Bootstrap** is a [free and open-source CSS framework](#) directed at responsive, [mobile-first front-end web development](#). It contains [CSS](#)- and (optionally) [JavaScript](#)-based

design templates for [typography](#), [forms](#), [buttons](#), [navigation](#), and other interface components.

**CHAPTER 2**  
**PROPOSED SYSTEM**

## **2.1 Proposed System:**

A proposed system is beneficial only if it can be turned into an information system that will meet the operational requirements of an organization. the proposed system offers adequate control to protect against fraud 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.

The aim of proposed system is to develop a system of improved facility.it can overcome all the limitations of the existig system.the system provides proper security and reduces manual work.

- Security of Data
- Ensure Data accuracy
- Good Efficiency
- User friendliness and interactive
- Minimum time required
- Better service

### **Modules of Self Publishing Portal: •**

- Post Management Modules: use for managing the Post details
- Comment Management Module: used for the managing the informations and details of the comment.
- New Catagory Module: used for managing the new category details
- New blog Module: used for managing new blog information.
- Login/Registration Module: Used for managing the login details
- User Modules: Used for managing the users of the system.

- View Module: Used to perform various activities on system.

## 2.2 Objective of System:

The main objective of the Project on Self publishing portal is to **manage the details of Posts, Comment, New Category, New Posts, Technology Posts**. It manages all the information about Blogs, Comment, Technology Blog, Blogs. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Blogs, Comment, Comment, New Category. It tracks all the details about the New Category, New Blog, Technology Blog.

The purpose of this project is to **automate the existing manual system** by the help of computerized equipments and full-fledged computer software. fulfilling user requirements, so that their valuable data/information can be stored for a longe

period with easy accessing and manipulation of the same .

This platform **use for educational purpose** it allows teachers and students to create their blog. By using this blogging platform teachers can easily forward important notice related to organization to their students. Students can post their suggestions and feedback for the organization. If someone is not able to express her/his view in front of people those also share the things through this blogging system.

This **system provides an easy way for teachers and student** to get connected to each other after leaving the particular organization.

This platform provides an environment for posting or sharing knowledge throughout the organization. This gives detailed information about functional and non-functional requirements of users or students i.e. whatever they want to share then they can easily write on their blog and it can be viewed by the member in the organization. The purpose meets the goal of sharing the information in formal or in secure way.

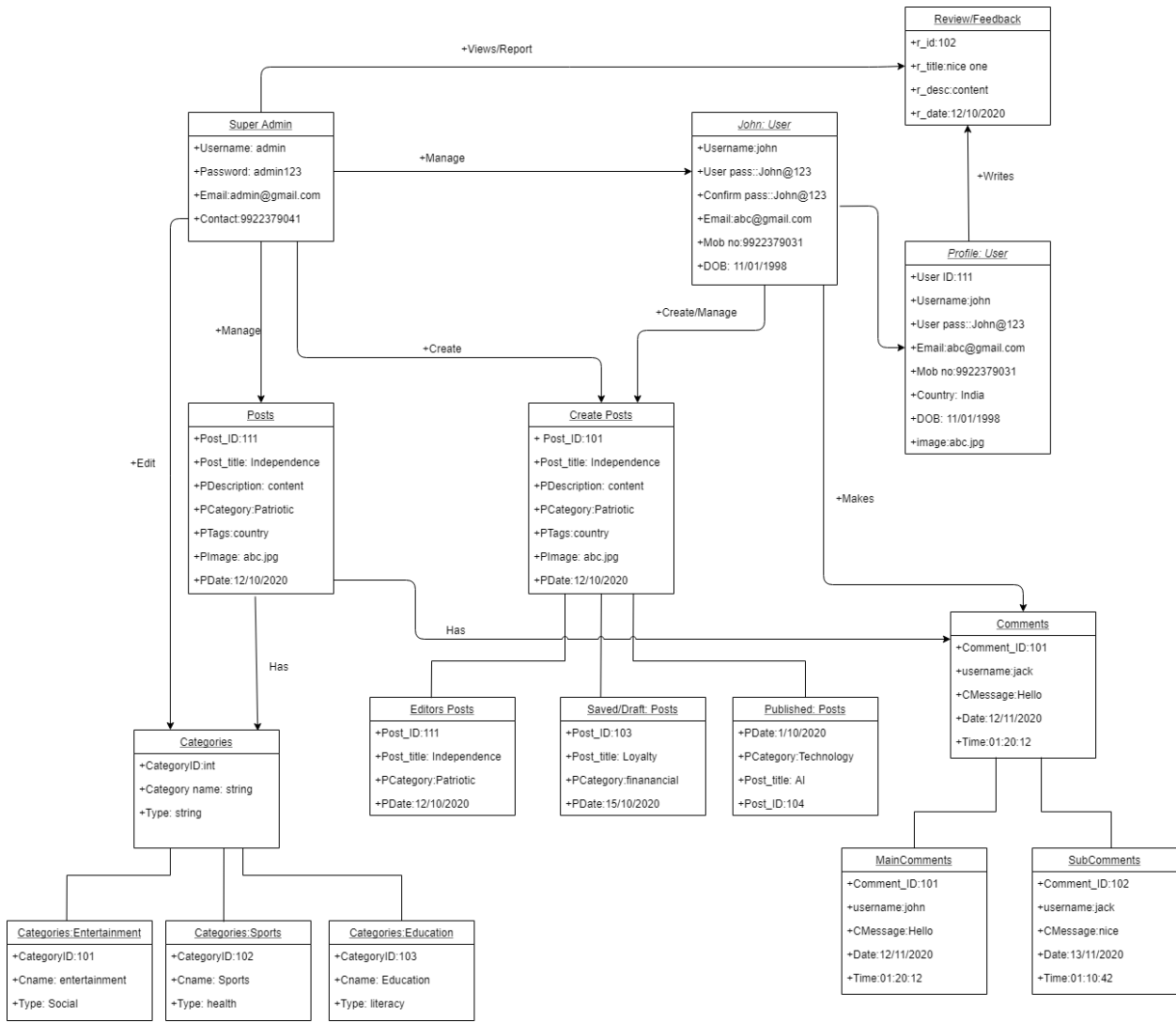
## 2.3 User Requirements:

- **Functional Requirements** should define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs
- **Data requirements** describe visible data and the user needs to manage or to process it.
- **user interface requirements** specify the interface between the site and the user. This includes (e.g. required screen formats, page or window layouts, content of any reports or menus, etc. specific layout which is familiar to the user.)
- **Security requirements** specify the factors that protect the software from accidental use or modification.
- User can need to share his/her thoughts with people in minimum time and resources.
- User can need a platform where they can publish his/her articles or any other content without doing any file work.
- User can need a simple user friendly interface to manage all their activities.
- User need a platform to connect with audience(bloggers)

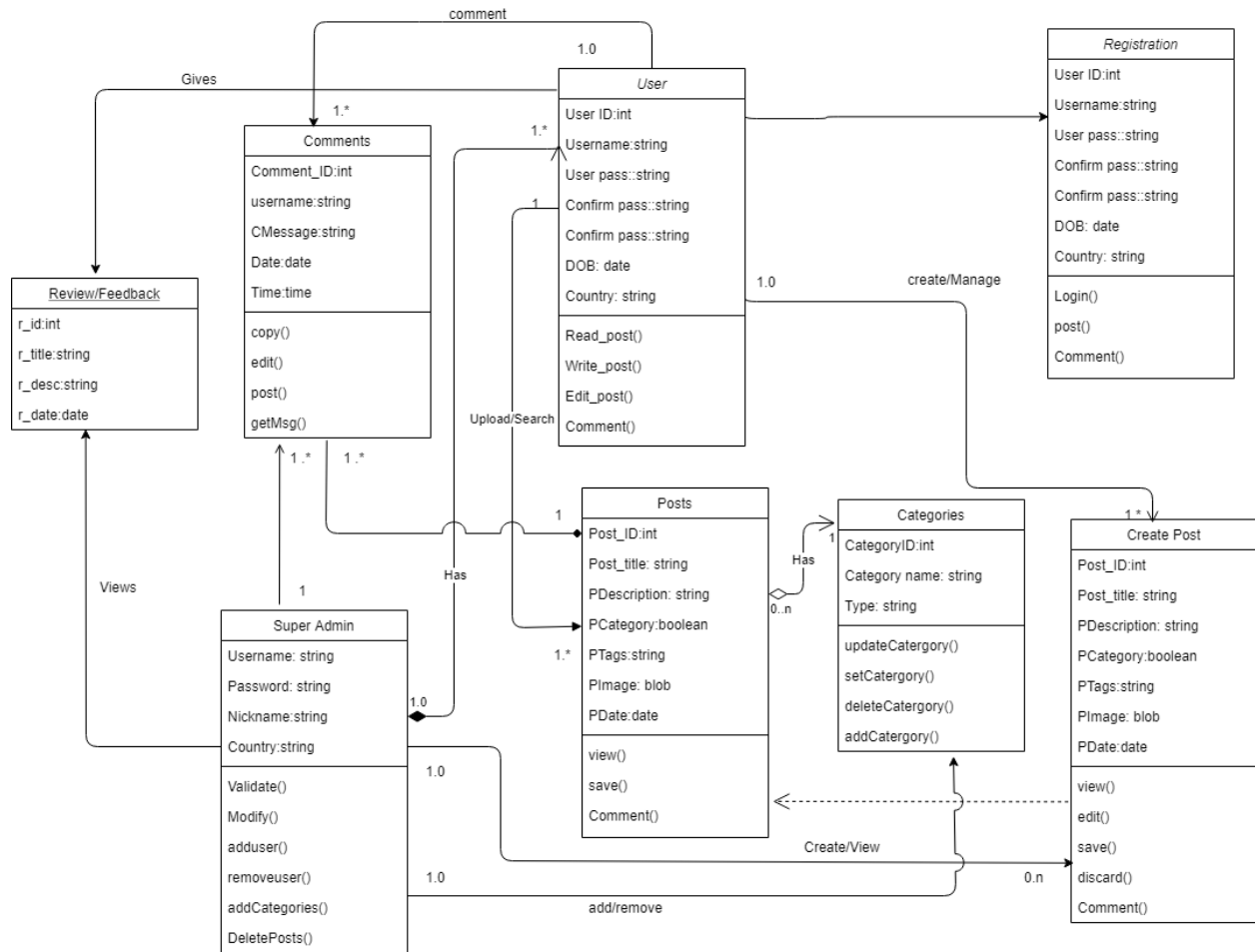
**CHAPTER 3**  
**ANALYSIS & DESIGN**



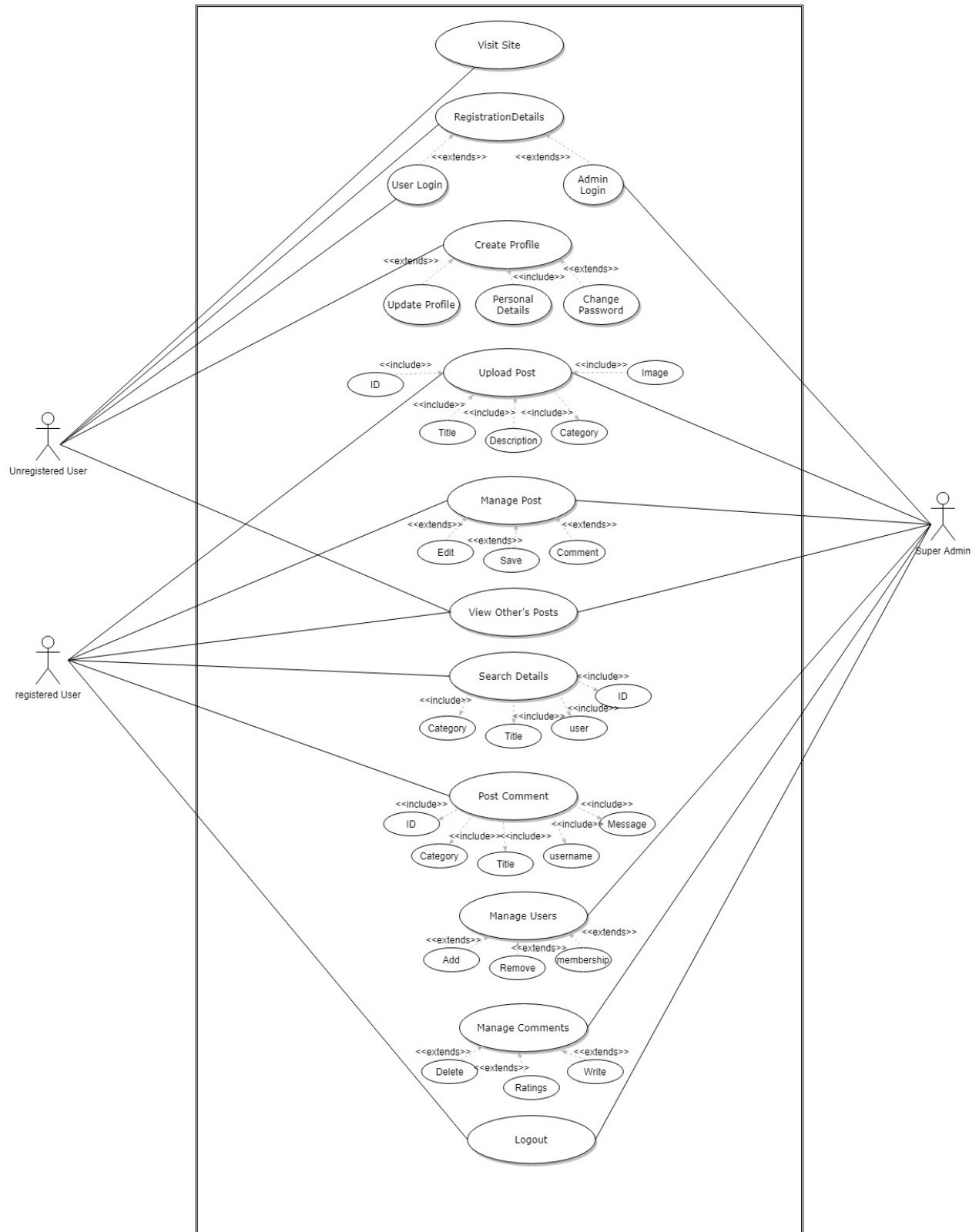
### 3.1 Object Diagram:



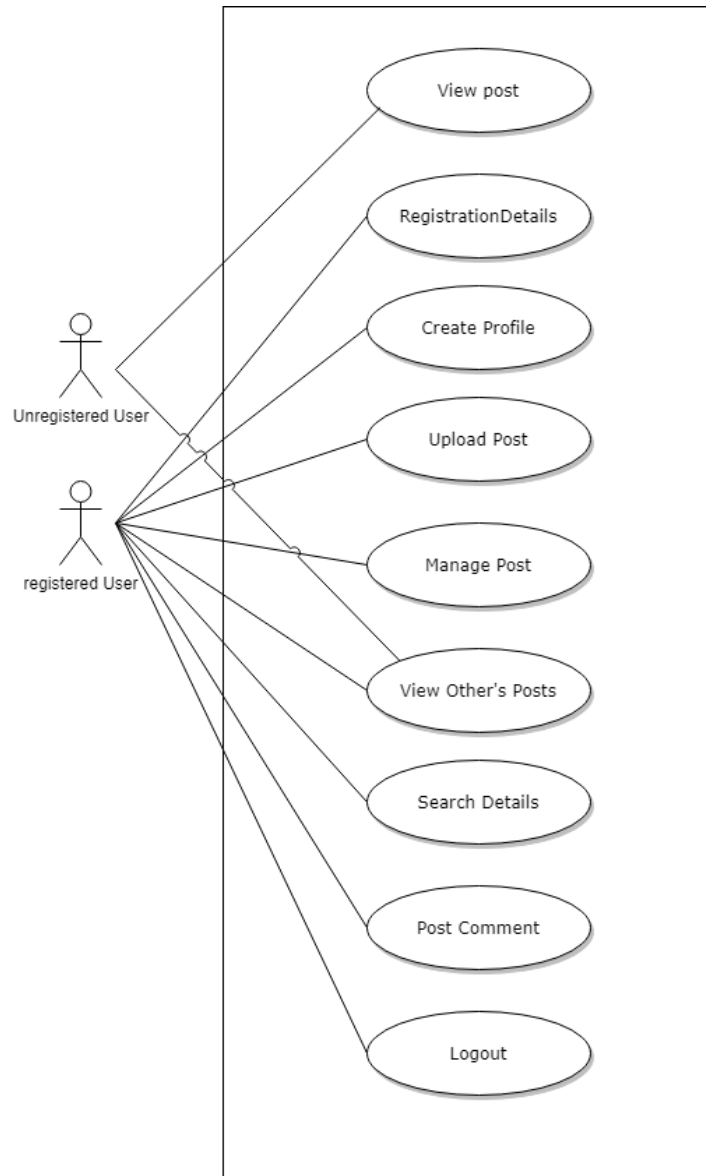
### 3.2 Class Diagram:



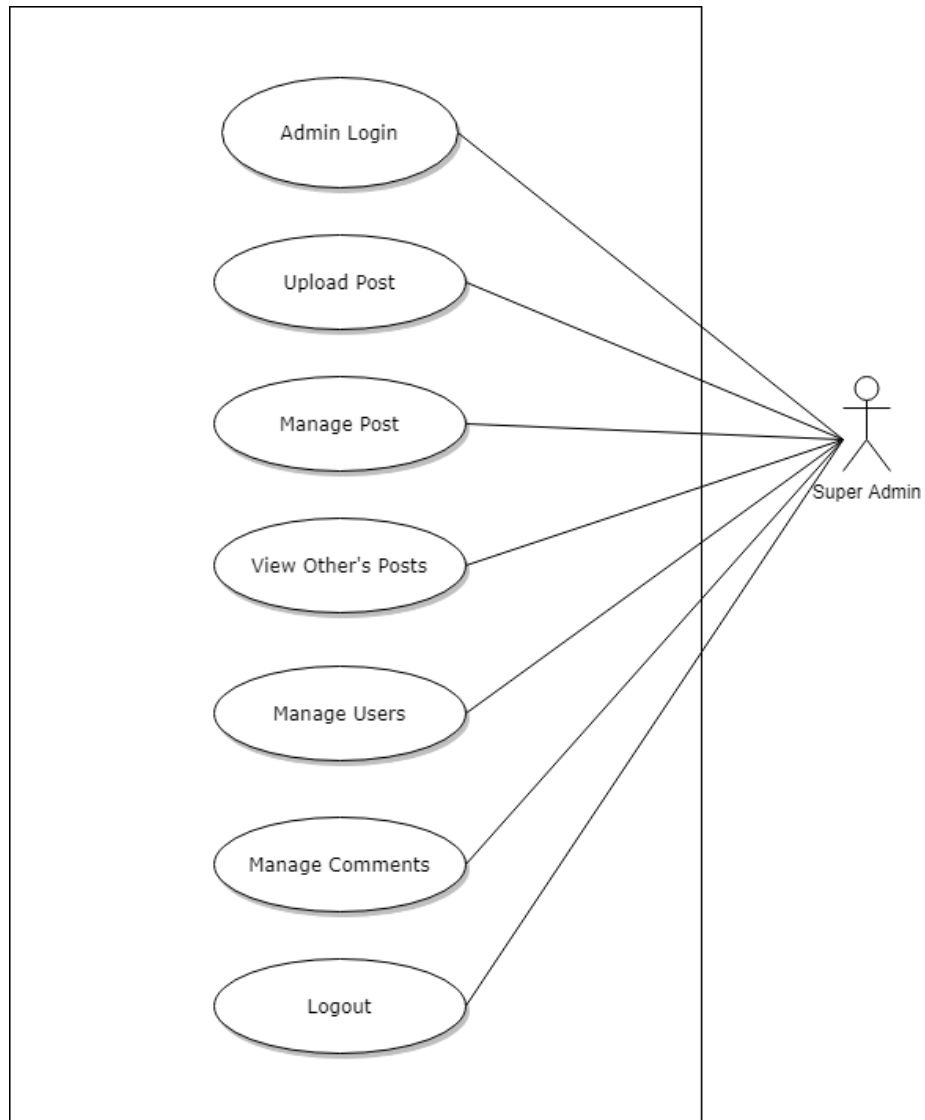
### 3.3 Use Case Diagram:



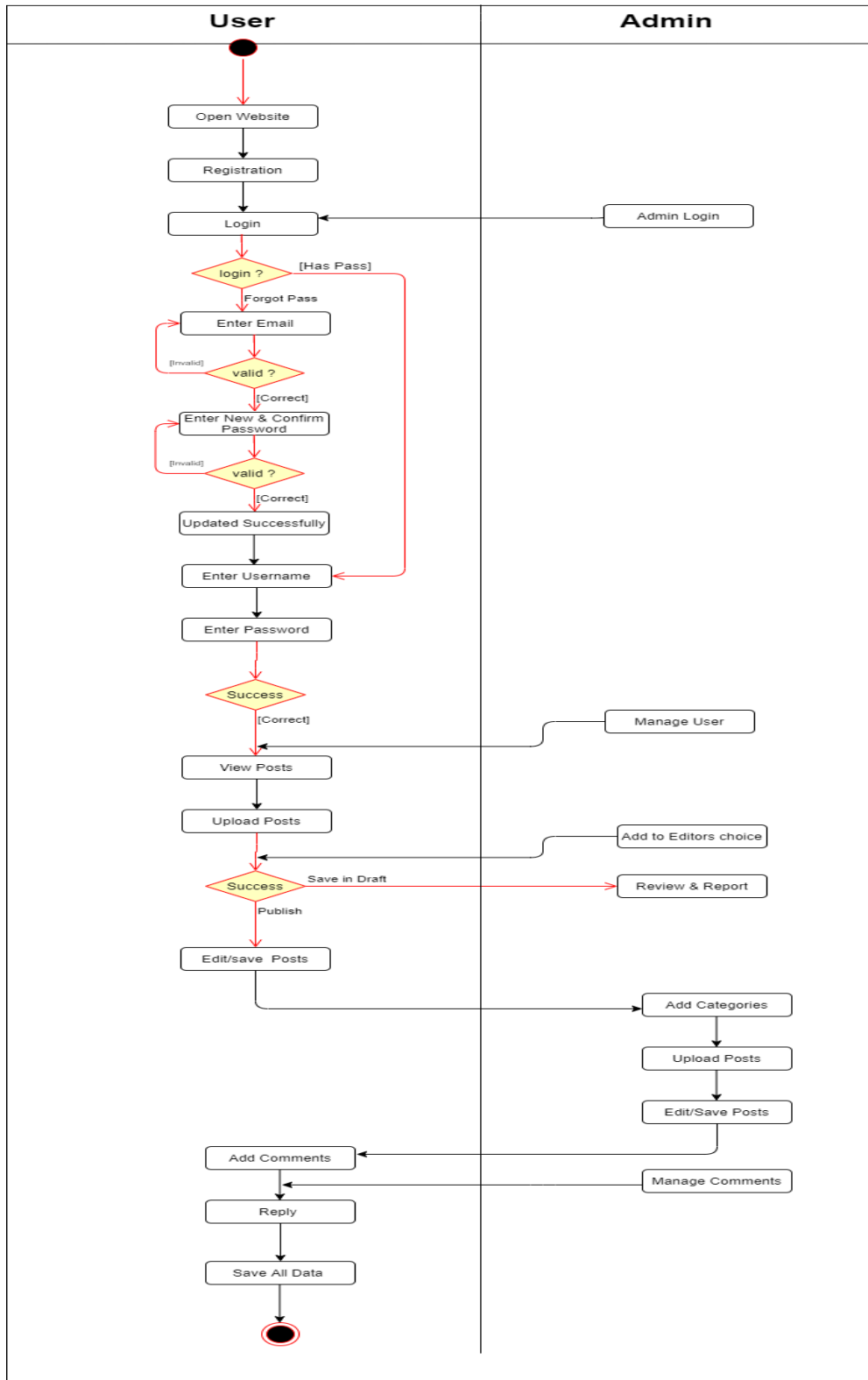
### 3.3.1 User Use Case Diagram:



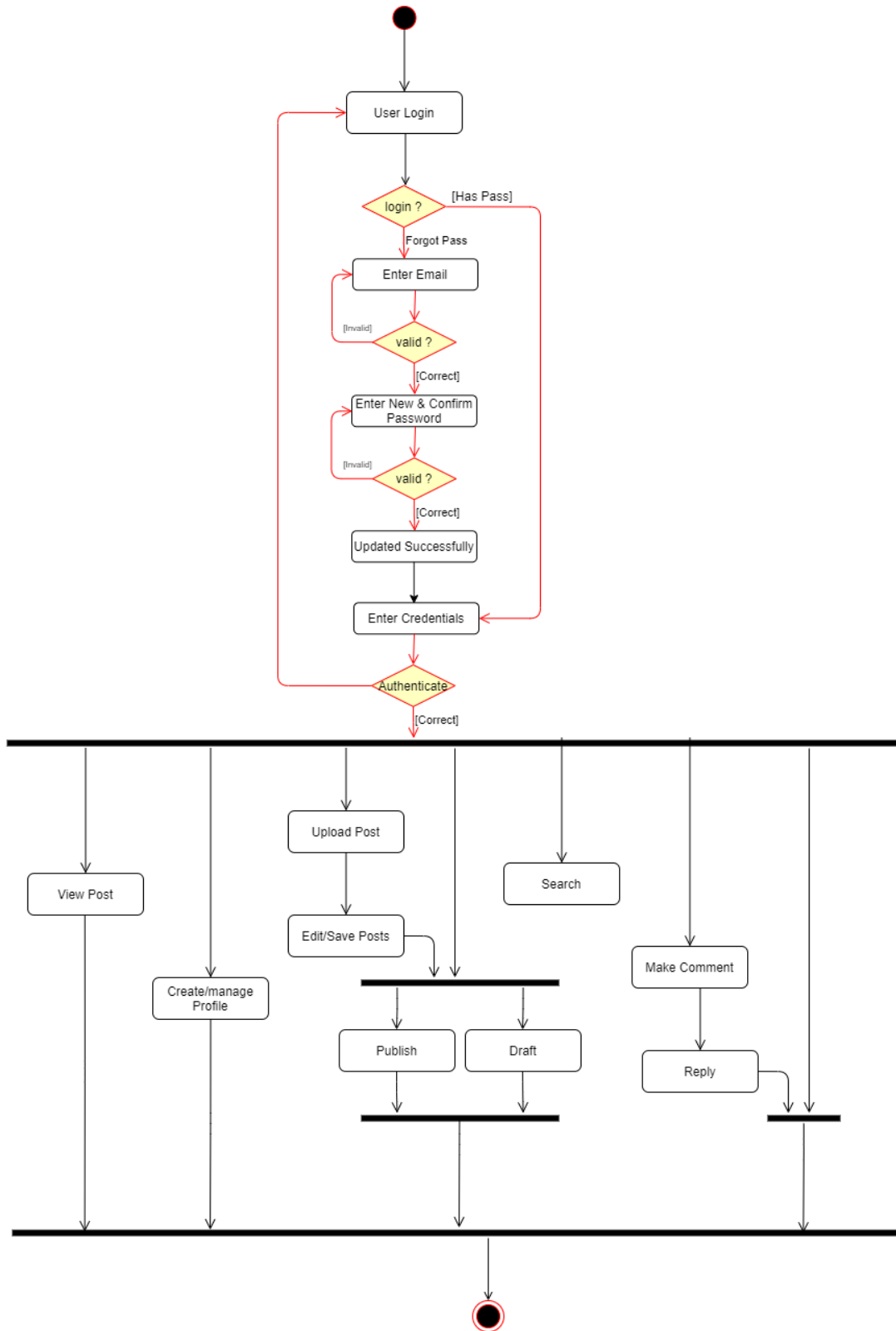
### 3.3.2 Admin Use Case Diagram:



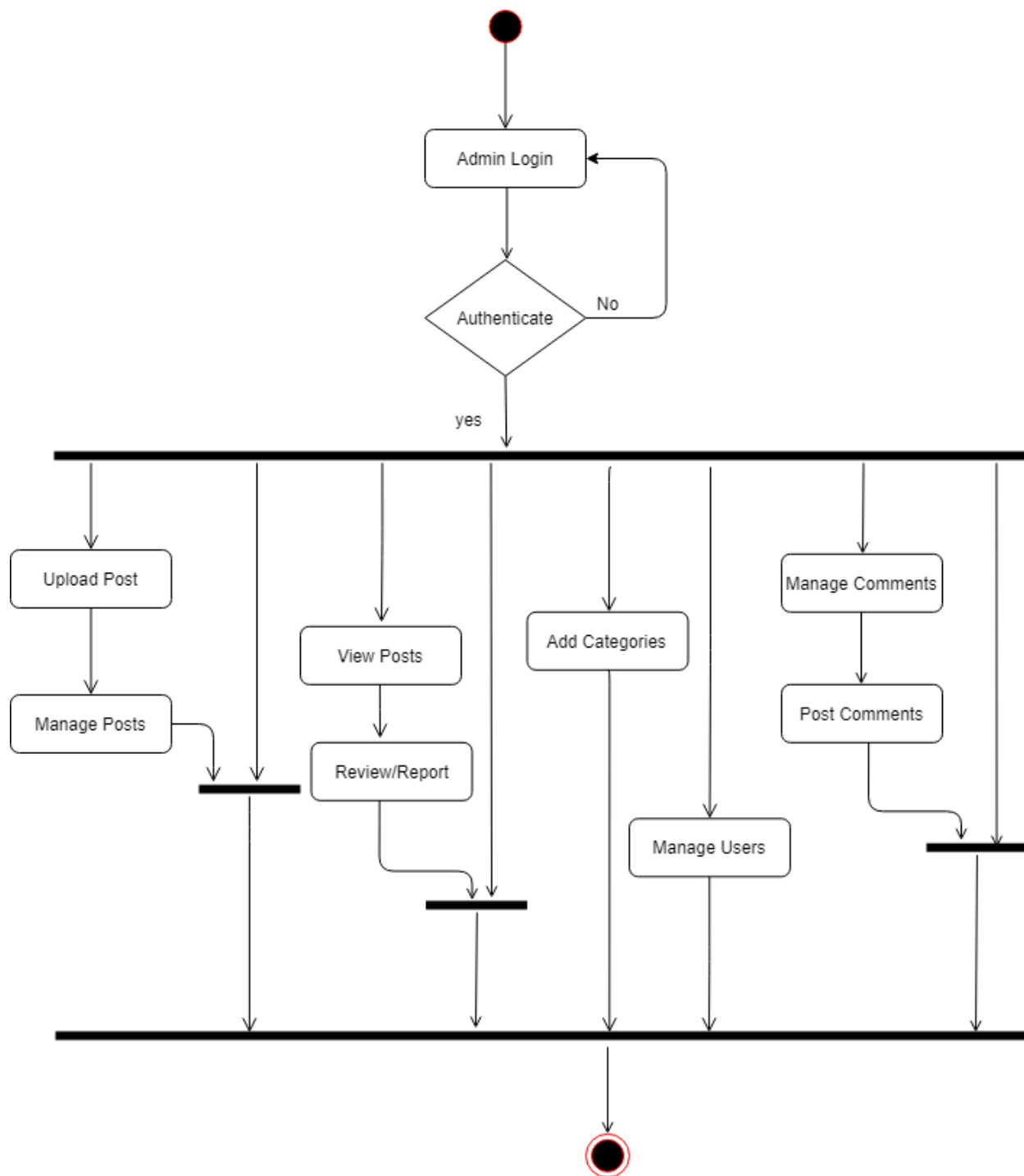
### 3.4 Activity Diagram:



### 3.4.1 User Activity Diagram:

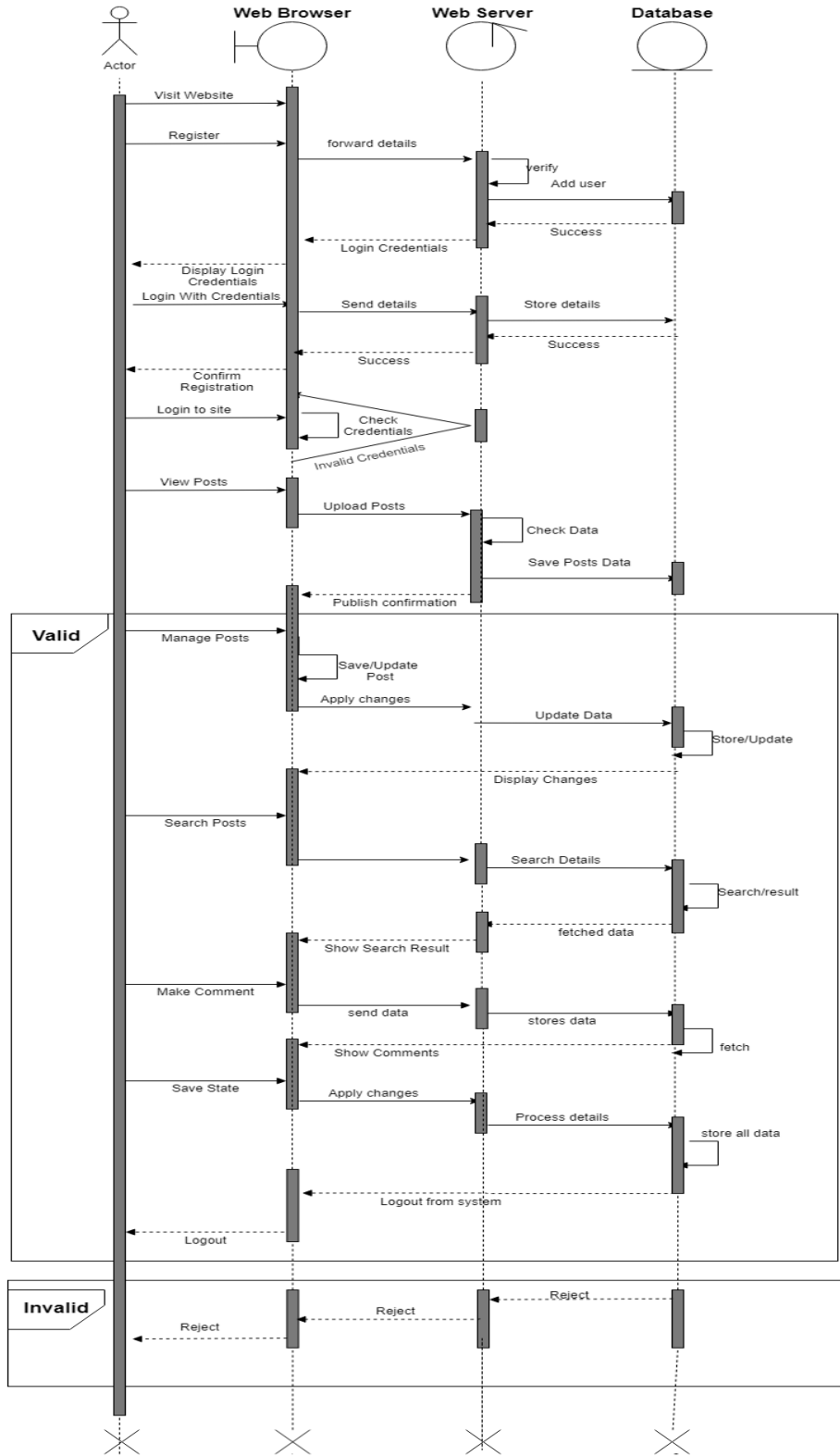


### 3.4.2 Admin Activity Diagram:

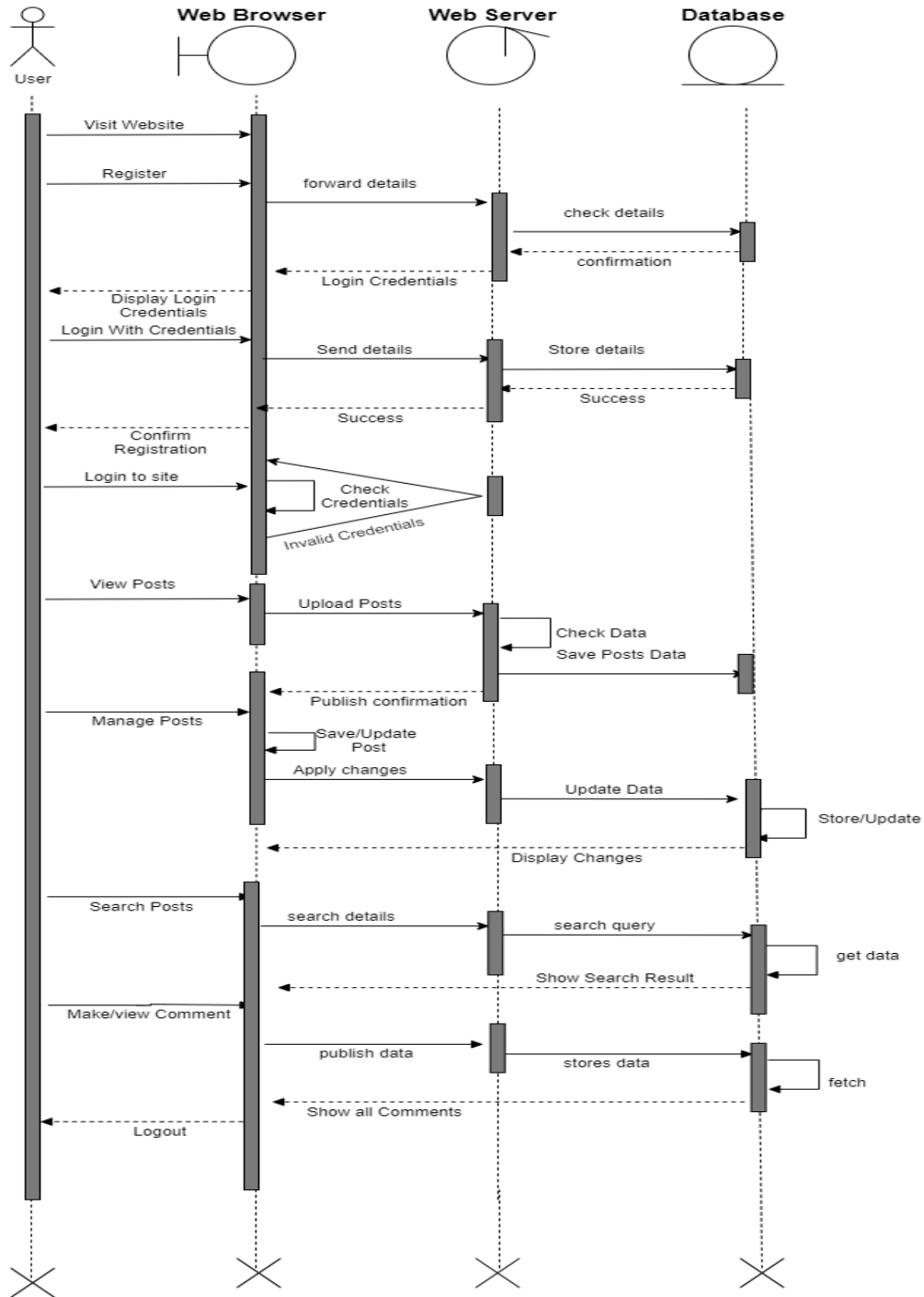




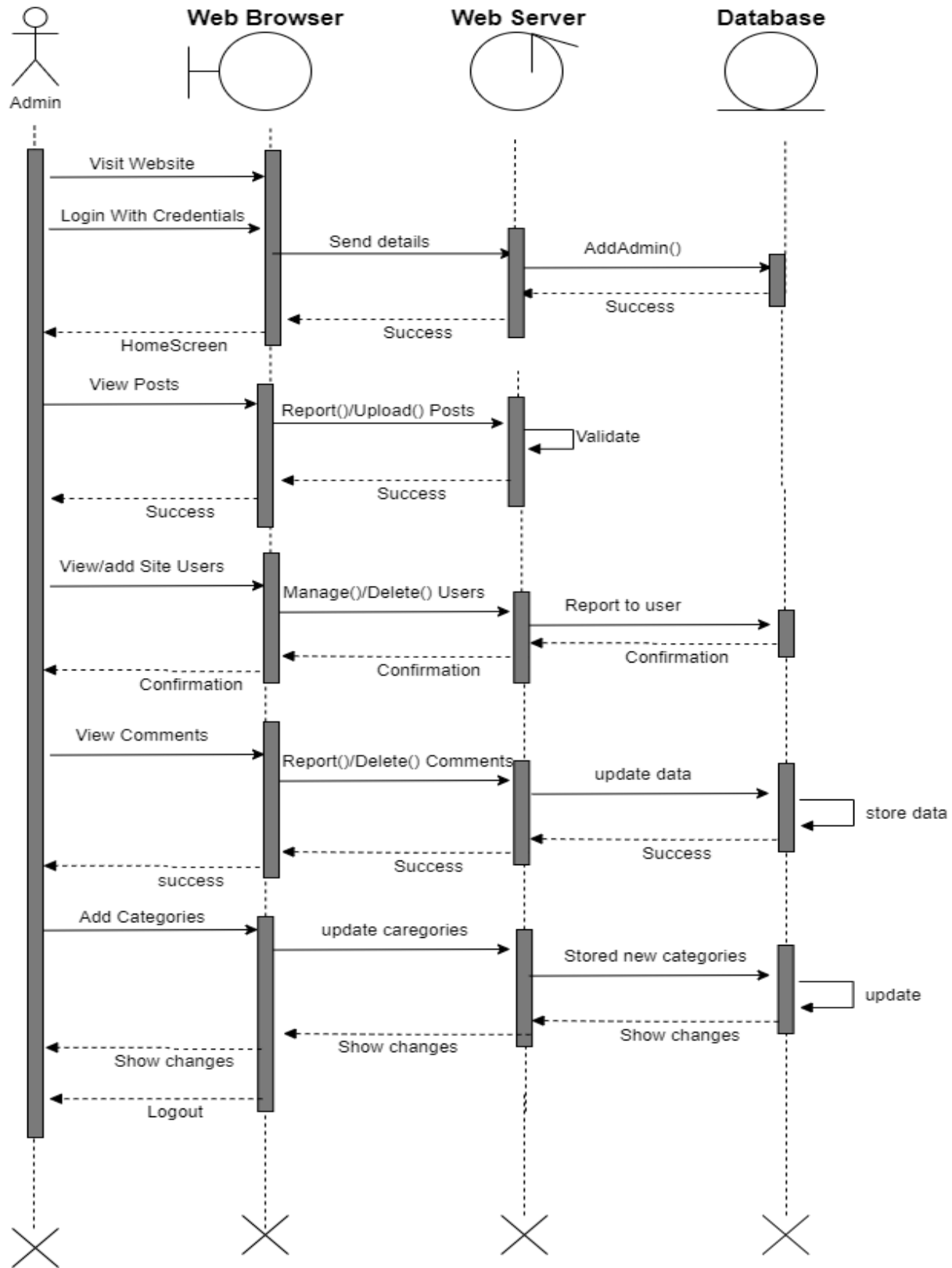
### 3.5 Sequence Diagram:



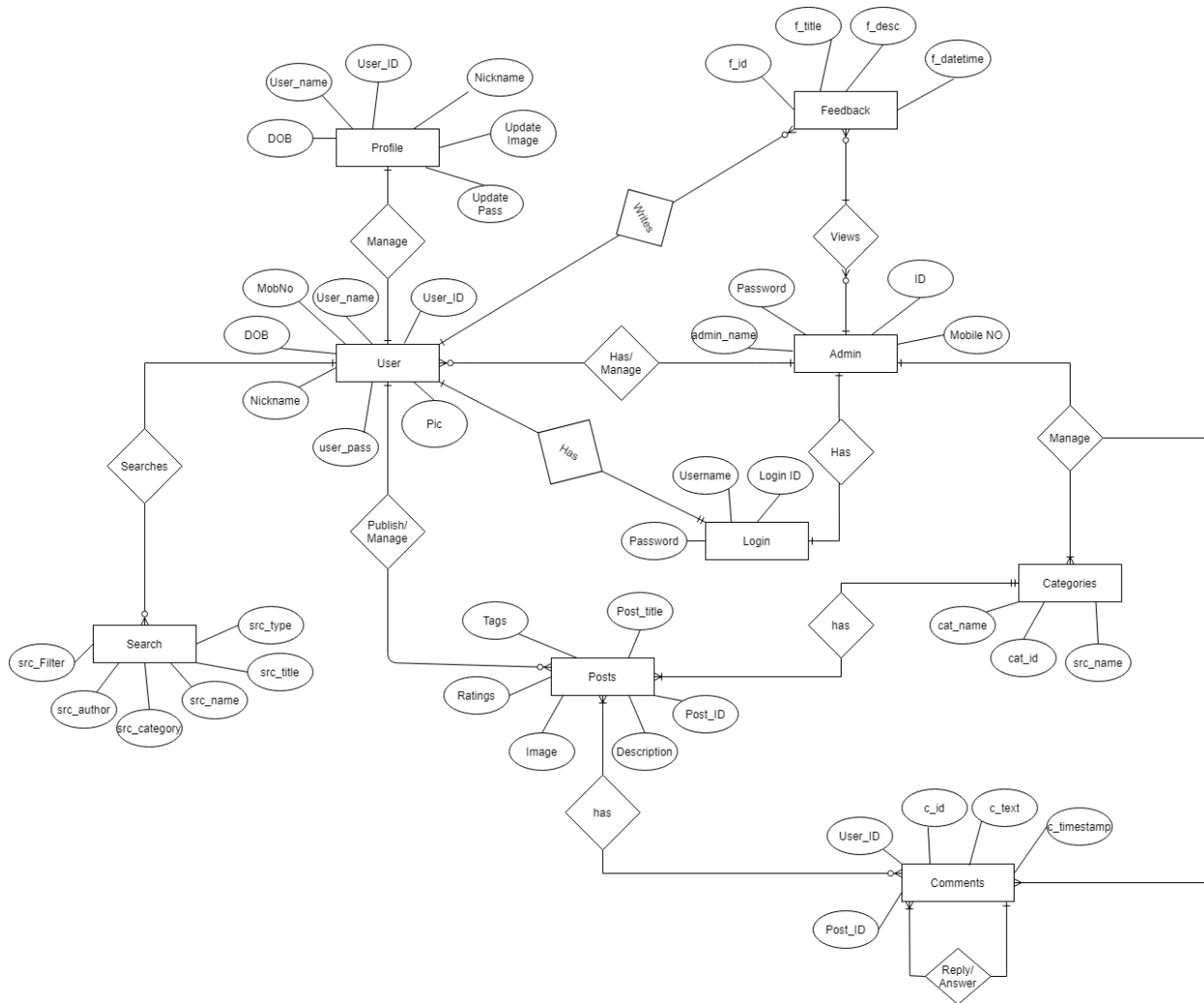
### 3.5.1 User Sequence Diagram:



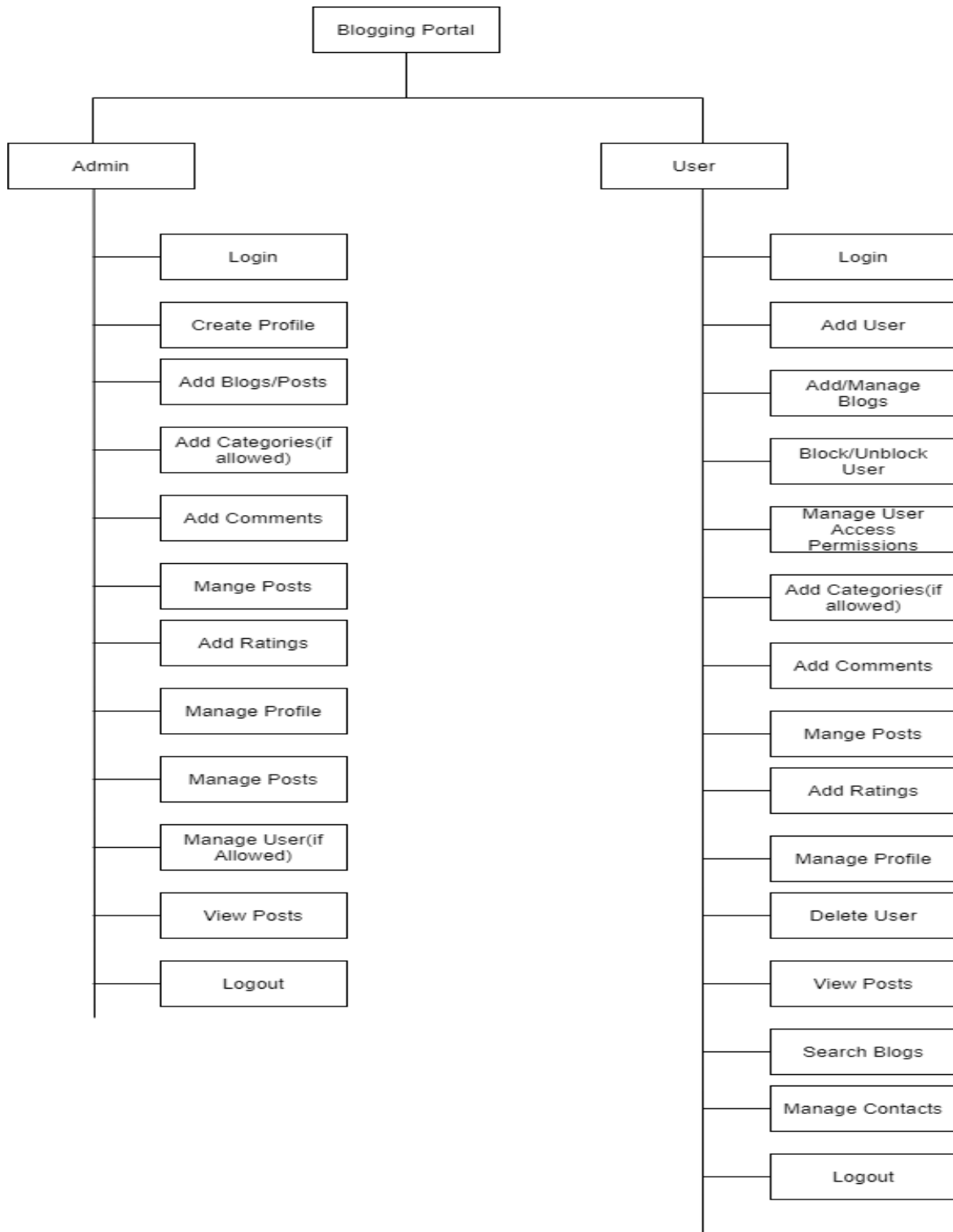
### 3.5.2 Admin Sequence Diagram:



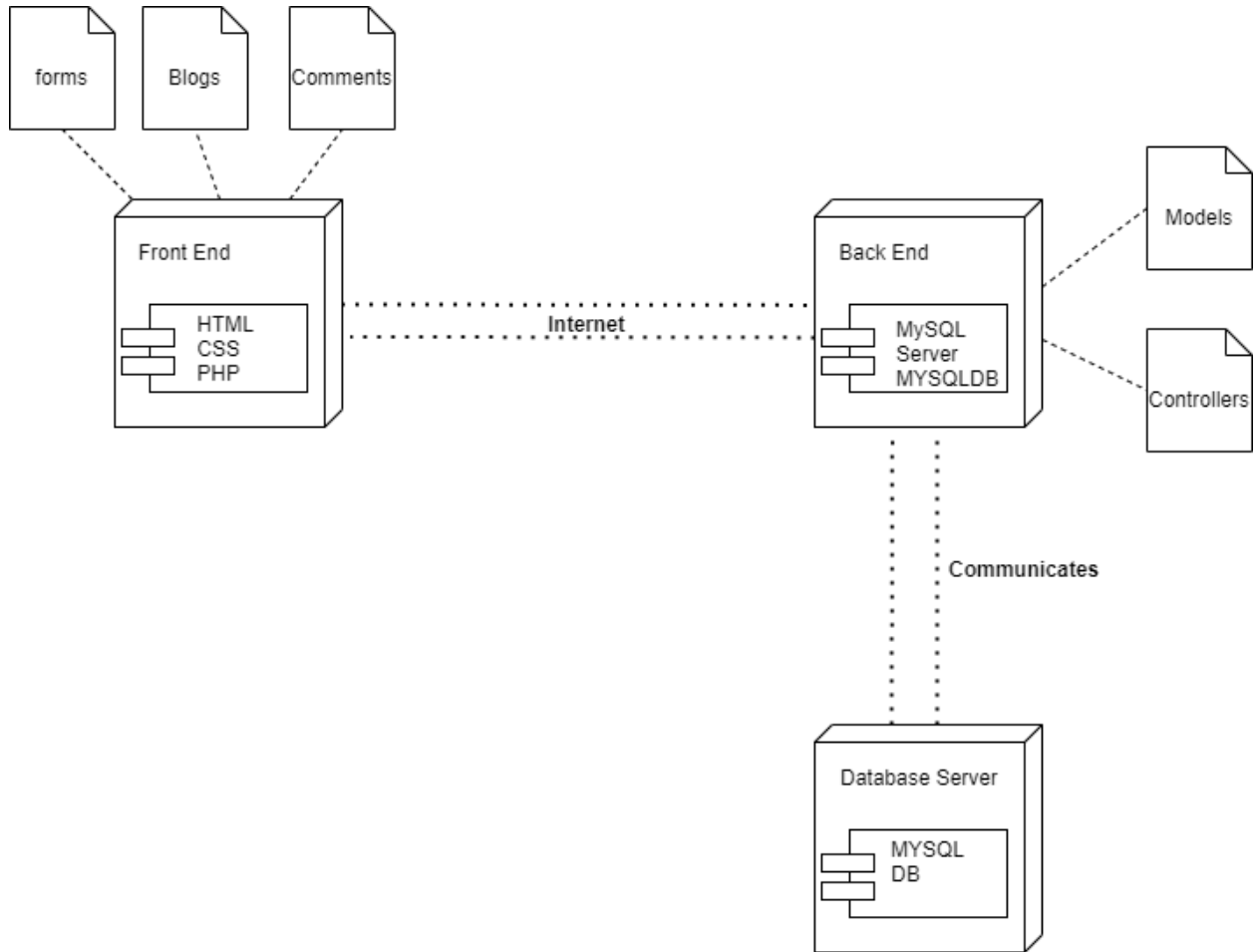
### 3.6 ER Diagram:



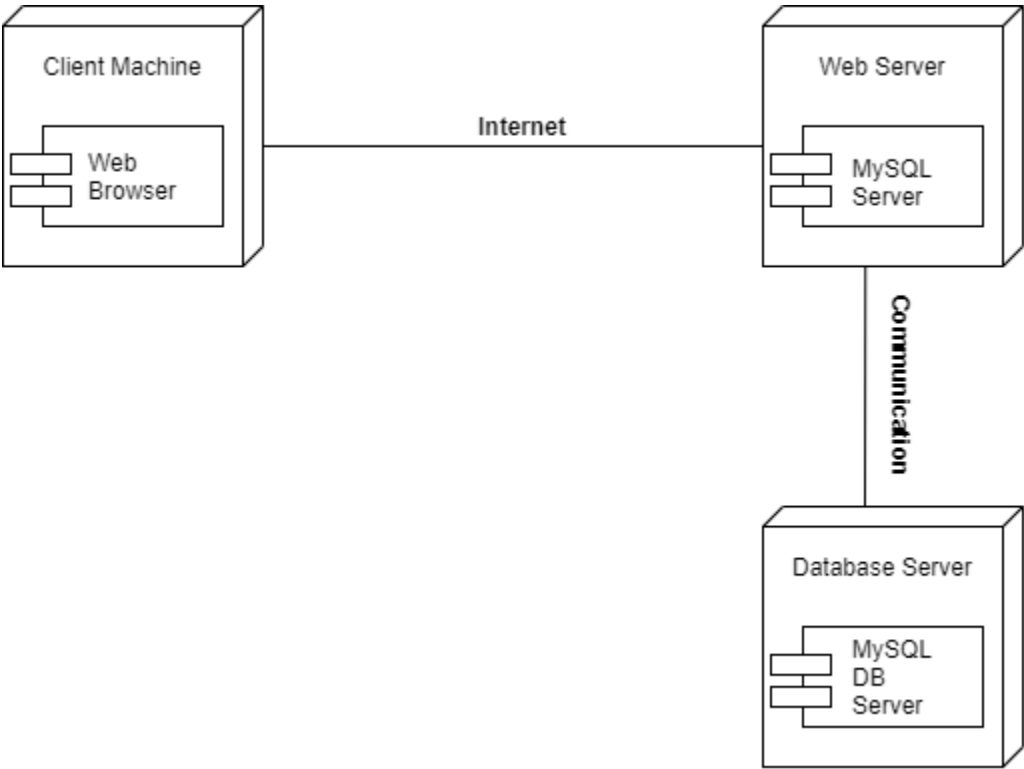
### 3.7 Module Heirarchy Diagram:



### 3.8 Component Diagram:



### 3.9 Deployment Diagram:



### **3.10 Module Specification:**

The Online Blogging System is divided into following modules:

#### **1) User Authentication:**

- Before authentication user is unknown to the system to interact with system functionality a user need to create his/her account.
- If user is already registered in the system then authentication will be done on the normal user login stage.
- For the authentication purpose, we implement the user authentication module, the user can able to login into the system by his/her login credentials.
- To check the user authentication we have used the username and password to authenticate with the data in the database.
- For e.g. Using the same credentials of User account, we can login to the system. Based on the user name returned from the database, the user login will be categorized as Normal user login, or Admin Head Login.

#### **2) Blog Module:**

- Blog Module is a main module of the blogging system. It contains :
  - 1.Single Blog
  - 2.Blog Comments
  - 3.Blog Views
  - 4.Blog Likes
  - 5.Blog Content
- A single blog contains the Blog Data(Date Created ,views ,Content ,Image ,Likes , etc) on the single blog page.
- Blogs has separated in categories
- Blogs can be saved as draft or publish by user or admin.
- Blogs either publish by user or an Admin.
- Blogs can be made as Editor's choice by the admin.

#### **3) Comments Module:**

- Each Post on the system has 0 or more comments made by the system users on the single post page.
- A comment contains sub comments made by different users on the system.
- A Comment has created date time with the comment writer's name with it display on the single post page.
- Each Comment has like and reply option to reply to different users in the system.

#### **4) User Access Permissions Module:**

admin can set permissions for users which are like user can edit post or not ,add categories or not, add comments or not ,publish posts or not. Permissions will be set from admin dashboard and user can see permission on his account. This module make system more secure.



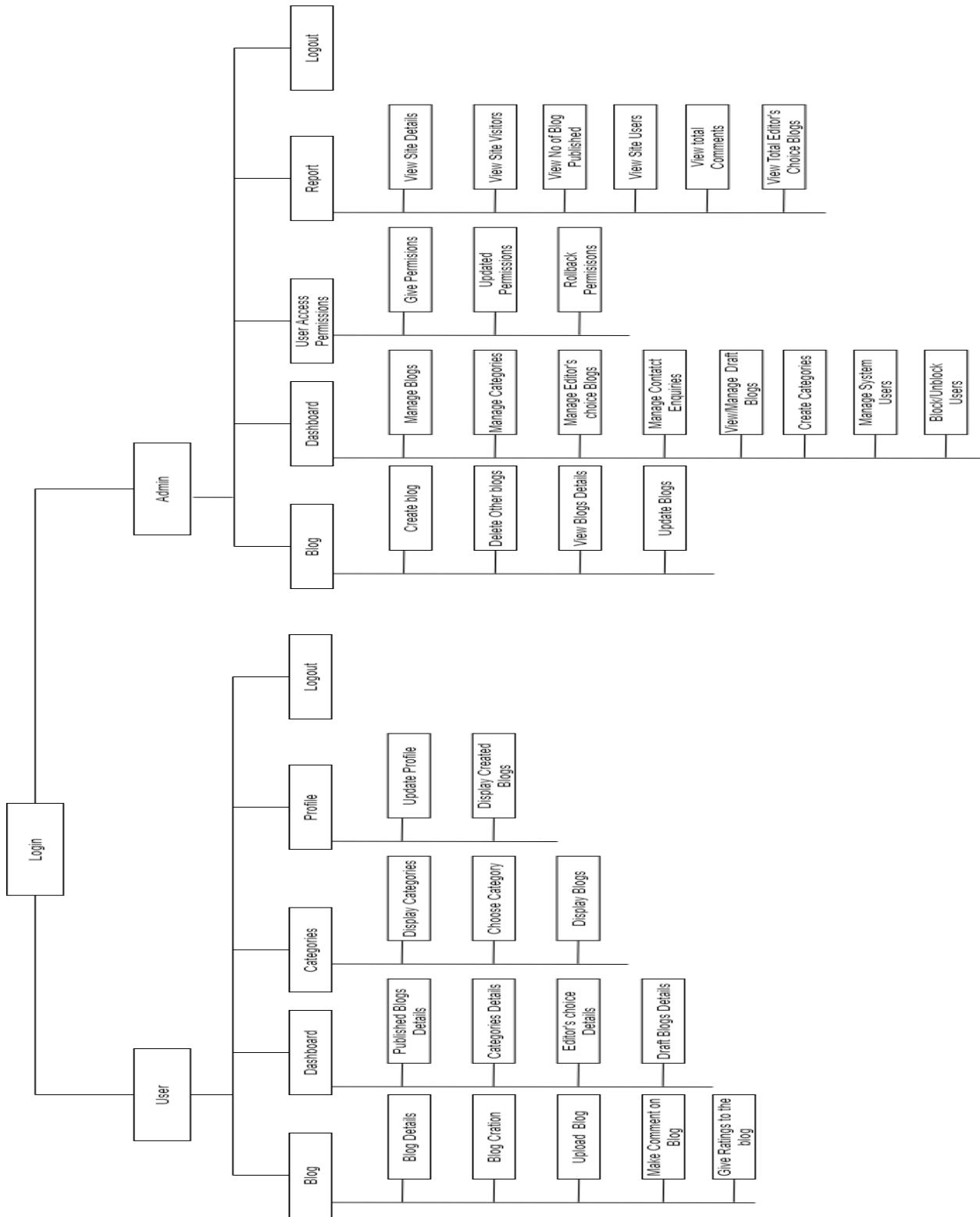
### **5) Search Module:**

- This Module contains the Search Functionality in the system
- A User can search blog by its name, categories, author name or date in the system.
- The Searched result will be displayed in the search result page on the front end.
- Search made by :
  1. Blog Title
  2. Categories
  3. Blog Author
  4. Blog Content

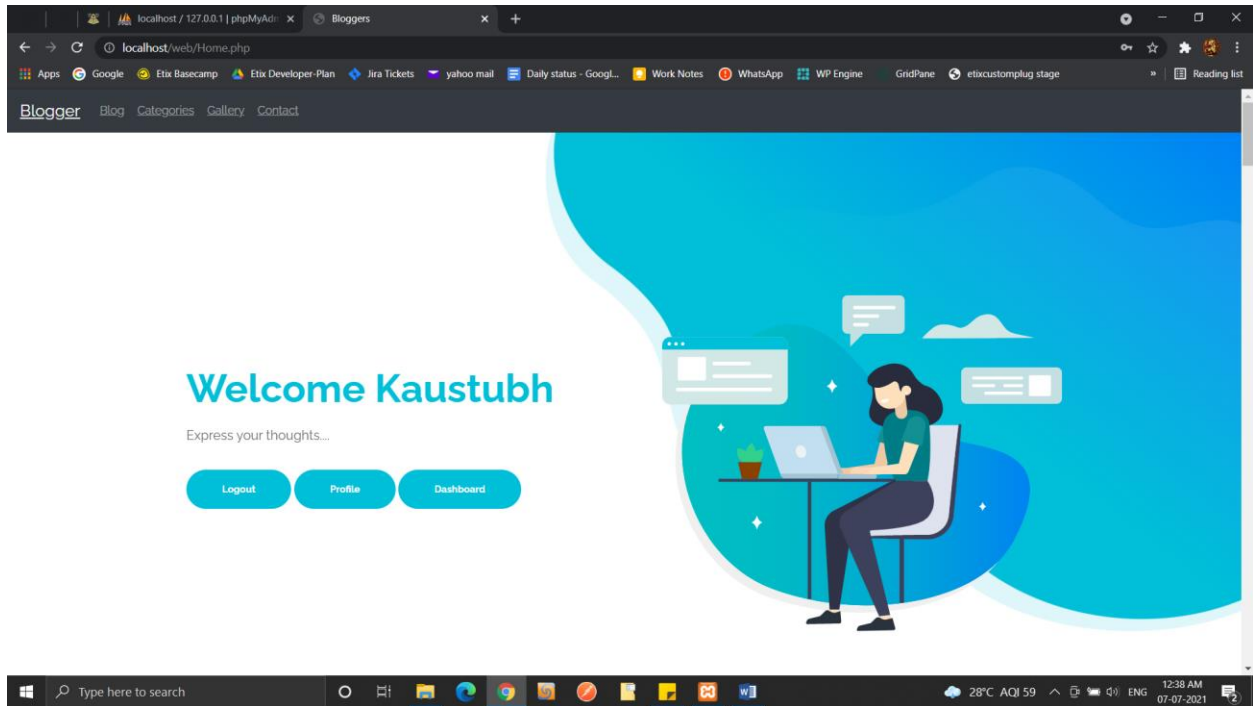
### **6) Blog Categories Module:**

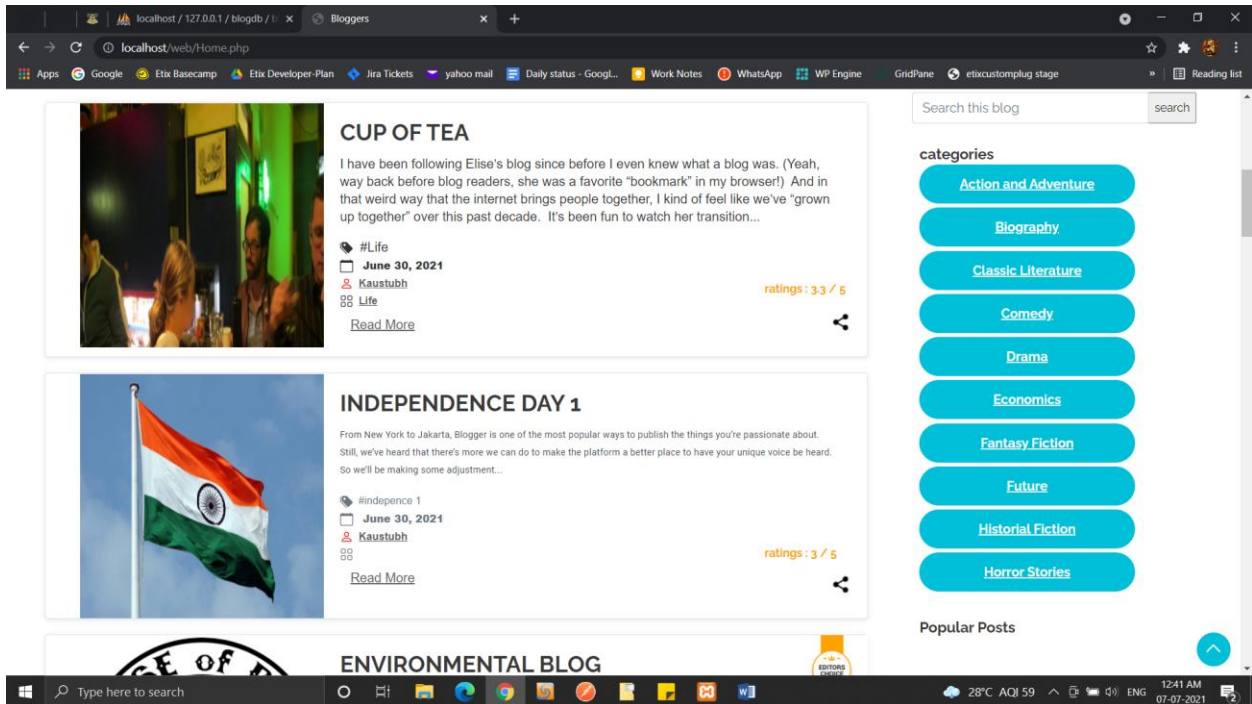
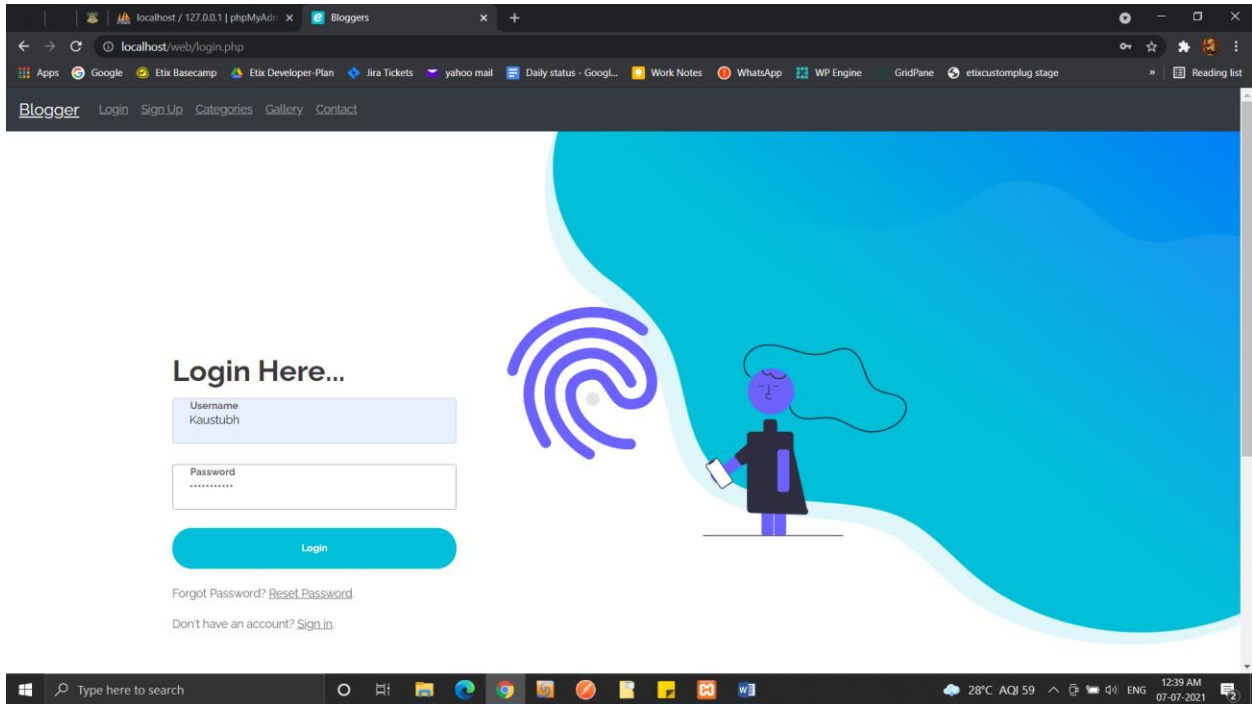
- This Module Categorised the blogs in the system in different sections by their genre.
- A user can choose particular blog by category or a user can assign a blog to the particular category in the system.
- An Admin can create new categories and those created categories are available for the normal user while publishing any blog in the system.
- An admin can delete or update the blog categories in the admin panel.

### 3.12 Web Site Map Path:



### 3.13 User Interface Design





localhost / 127.0.0.1 / blogdb / i / x Bloggers

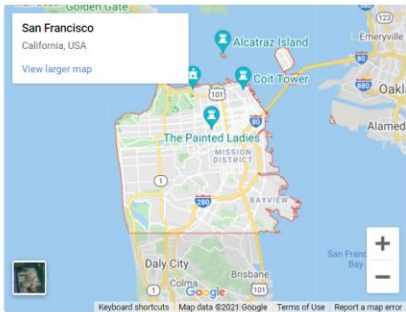
localhost/web/contact.php

Blogger Blog Categories Gallery Contact

## Contact Information

Don't hesitate to give us a call or send us a contact form message

22 Innovative Area, Delhi, CA 94043, India [+81 720 2212](tel:+817202212) [office@blogger.com](mailto:office@blogger.com)



Name

Email

Your message

I have read and agree with Evlo's stated [Privacy Policy](#) and [Terms Conditions](#)

message

Type here to search 28°C AQI 59 12:41 AM 07-07-2021

localhost / 127.0.0.1 / blogdb / i / x Blog Post - Start Bootstrap Tem

localhost/web/single\_blog.php?id=169

Rate This blog

★★★★★

Submit

Leave a Comment Kaustubh:

Write Comment

Add Comment

By Kaustubh on 2021-07-01 00:31:13

fdfs

Reply

By Kaustubh on 2021-07-01 00:32:21

fsfdaaa

Reply

By Kaustubh on 2021-06-22 23:28:41

hey

Reply

Type here to search 28°C AQI 59 12:42 AM 07-07-2021


localhost / 127.0.0.1 / blogdb / i / x Bloggers

localhost/web/search.php?search=blog

Apps Google Etix Basecamp Etix Developer-Plan Jira Tickets yahoo mail Daily status - Googl... Work Notes WhatsApp WP Engine GridPane etixcustomplug stage Reading list

Blogger Blog Categories Gallery Contact

### Search Result for "blog"



#### ENVIRONMENTAL BLOG

Since 1999, millions of people have expressed themselves on Blogger. From ...


Trees #Environment

30-06-2021

Kaustubh

Life

[Read More](#)



#### INDEPENDENCE DAY 1

From New York to Jakarta, Blogger is one of the most popular ways to publi...

independence 1

30-06-2021

Kaustubh

[Read More](#)

Type here to search

28°C AQI 59 12:42 AM 07-07-2021

Browser tabs: (396 unread) - k, (3) WhatsApp, Dashboard, Remote Access, Remote Access, localhost / 127, robin cook inva, ICW-37211 Alle, +

Address bar: localhost/web/adminpanel/admin\_report.php

Browser extensions: Apps, Google, Etix Basecamp, Etix Developer-Plan, etixcustomplug stage, etixcustomplug live, Box Office, Jira Tickets, yahoo mail, WhatsApp, WP Engine, GridPane, Reading list

Home icon

### Overview

<p>6 Site Users</p> <p>Line chart showing user activity over time.</p>	<p>9 Total Site Blogs</p> <p>Line chart showing total site blogs over time.</p>	<p>6 This Month Blogs</p> <p>Line chart showing this month's blogs over time.</p>	<p>2 Editor's choice blogs</p> <p>Bar chart showing editor's choice blogs.</p>
<p>55 Total Site Visitors</p> <p>Bar chart showing total site visitors.</p>	<p>28 Total comments</p> <p>Bar chart showing total comments.</p>	<p>2 Total Enquiries</p> <p>Bar chart showing total enquiries.</p>	

Windows taskbar: Type here to search, File Explorer, Edge, Chrome, VS Code, Remote Desktop, PDF, Word, PowerPoint, Outlook, Mail, Photos, Music, Video, Settings, Network, Volume, ENG, 03:58 PM, 11-06-2021

### 3.14 Data Dictionary:

Sr no	Column Name	Data Type	Size	Description
1.	AdminId	Integer	5	Stores adminId of admin in admin table
2.	Name	Varchar	40	Stores name of admin in admin table
3.	Password	Varchar	30	Stores password of admin in admin table
4.	Contact	Integer	10	Stores contact of admin in admin table
5.	UserId	Integer	5	Stores UserId of User in User table
6.	Name	Varchar	40	Stores name of User in User table
7.	Password	Varchar	30	Stores password of User in User table
8.	Email	Varchar	35	Stores email of User in User table
9.	Contact	Integer	10	Stores contact of User in User table
10.	Country	Varchar	30	Stores name of user's country in user table
11.	FirstName	Varchar	40	Stores fname of user in user table
12.	LastName	Varchar	40	Stores lname of user in user table
13.	Email	Varchar	30	Stores email of user in user table
14.	Password	Varchar	15	Stores password in user table
15.	PostID	Integer	5	PK stores ID of User
16.	Title	Varchar	200	Stores title of post in posts table
17.	Tags	Varchar	40	Stores tags of post in posts table



18.	Content	Longtext		Stores Content of post in posts table
19.	Image	Varbinary	max	Stores Image of post in posts table
20.	Posted	Datetime		Stores Datetime of post in posts table
21.	Date	Date		Stores date of post in posts table
22.	Author	Varchar	50	Stores Authorm of post in posts table
23.	Category	varchar	50	Stores Category of post in posts table
24.	CategoryID	Integer	5	PK Stores id of category in category table
25.	Name	Varchar	40	Stores name of category in category table
26.	CommentID	Integer	5	PK Stores id of comments in comment table
27.	Description	Varhchar	200	Stores the content of comment in comment table
28.	Date	Date		Stores date posted comment
29.	Time	DateTime		Stores time of comment in comment table
30.	UserID	Integer	5	FK takes user id form user table
31.	SubCommentID	Integer	5	FK takes subcomemnt id form subcomemnt table
32.	SubCommentID	Integer	5	PK Stores ID of Subcomments in subcoment table
33.	Description	Varchar	200	Stores the message of subcomment in Subcomment table

34.	Date	Date		Stores date of subcomment in Subcomment table
35.	Time	DateTime		Stores time of subcomment in Subcomment table
36.	UserID	Integer	5	FK takes user id form user table
37.	CommentId	Integer	5	FK Takes id of comments in comment table
38.	FeedbackID	Integer	5	PK Stores id of Feedback in feedback table
39.	Description	Varchar	200	Stores the Feedback Content in feedback table
40.	Date	Date		Stores date of feedback in feedback table
41.	UserID	Integer	5	FK takes user id form user table
42.	Id	Integer	5	PK stores id of editors choice posts
43.	PostID	Integer	5	FK Takes ID of Posts
44.	Date	Date		Stores date of Posts in editor table
45.	UserID	Integer	5	FK takes user id form user table

### 3.15 Table Specification:

#### 1. User Table

Column Name	Data type	Size	Constraint
UserId	Integer	5	Primary Key (AI)
FirstName	Varchar	40	Not Null
LastName	Varchar	40	Not Null
Email	Varhar	30	Not Null
Password	Varchar	12	Not Null
ConfirmPassword	Varchar	12	Not Null
Contact	Integer	10	Not Null
City	Varchar	40	Not Null
Address	Varchar	100	Not Null

#### 2. Blog Record Table

Column Name	Data type	Size	Constraint
BlogId	Integer	5	Primary Key(AI)
Title	Varchar	200	Not Null
Tags	Varchar	40	Not Null
Content	longtext		Not Null
Image	Varchar	100	Not Null
Status	Varchar	10	Not Null
Posted	Datetime		Not Null
Date	date		Not Null
Author	Varchar	50	Not Null
Category	Varchar	50	Not Null

UserId	Integer	5	Forein Key
Month	Datetime		Not Null

### 3. Categories Table

Column Name	Data type	Size	Constraint
CategoryId	Integer	5	Primary Key(AI)
Name	Varchar	40	Not Null
Image	Varchar	100	Not Null

### 4. Comments Table

Column Name	Data type	Size	Constraint
CommentId	Integer	15	Primary Key(AI)
Parent CommentId	Integer	15	Not Null
Comment Sender name	Varchar	40	Not Null
Description	Varchar	200	Not Null
Date	date		Not Null
Time	datetime		Not Null
UserID	int	5	Foreign Key
BlogId	int	5	Foreign Key

### 5.Contact Table

Column Name	Data type	Size	Constraint
CID	Integer	5	Primary Key
Email	Varchar	30	Not Null

Description	Varchar	20	Not Null
Date	Date/Time		Not Null
UserId	Integer	5	Foreign Key

### 6.Editor's Choice Table

Column Name	Data type	Size	Constraint
Id	Integer	5	Primary Key
PostId	Integer	5	Foreign Key
Date	Date/Time		Not Null
UserId	Integer	5	Foreign Key

### 7.User Permissions Table

Column Name	Data type	Size	Constraint
pId	Integer	5	Primary Key
Options	Varchar	20	Not Null
UserId	Integer	5	Foreign Key
Allow insert	Int	10	Not Null
Allow view	Int	10	Not Null
Allow Edit	Int	10	Not Null
Allow Delete	Int	10	Not Null

### 8.Ratings Table

<b>Column Name</b>	<b>Data type</b>	<b>Size</b>	<b>Constraint</b>
RId	Integer	5	Primary Key
rate	Float		Not Null
Blogid	Int	10	Foreign Key
UserId	Integer	5	Foreign Key

### 9.Counter Table

<b>Column Name</b>	<b>Data type</b>	<b>Size</b>	<b>Constraint</b>
Cntid	Integer	5	Primary Key
Counts	Int	10	Not Null

### **3.16 Test Procedures and Implementation:**

Software testing is a critical element of software quality assurance & represents the ultimate review of specification, design and code generation.

It is the process of executing a program with a primary objective of finding errors.

Testing gives the guarantee that the software does not fail and runs according to its specification and in the way the end user expects.

This can be done by various software testing techniques which provide a systematic guidance for designing tests that exercise the internal logic of software components, and exercise the input and output domains of the program to uncover errors in programming functions, behavior and performance.

Testing is the exposure of system to trial input to see whether it produces correct output.

Testing is the process of detecting presence of faults. Once the source code has been generated, software must be tested to uncover as many errors as possible before delivery to your customer. Our goal is to design a series of test cases that have likelihood of finding errors. That's where Software testing Techniques enter into the picture. A set of test cases designed to exercise both internal logic and external requirements is designed and documented, expected results are defined and actual results are recorded.

#### **Testing Objectives:-**

The testing objectives are summarized in the following three steps

1. Testing is the process of executing a program with the intent of finding a bug.
2. A good case is one that has a high probability of finding an as yet undiscovered error.
3. A successful test is the one that uncover yet an undiscovered error.

#### **Unit testing:**

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code usually at the functional level. In an object-oriented environment, this is usually at class-level and the minimal unit tests include the constructors and destructors. These type of tests are usually written by developers as they work on code (white-box style), to ensure that the specific function is working as expected.

One function might have multiple tests, to catch corner cases or other branches in the code. Unit testing alone cannot verify the functionality of a piece of software, but rather is used to assure that the building blocks of the software work independently of each other.

#### **Integration Testing**

Integration Testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an interactive way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be localized more quickly and fixed.

Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively user groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a software.

### **System Testing :**

System Testing tests a completely integrated system to verify that it meets its requirements.

The testing phase is an important part of software development, It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

### **Acceptance Testing :**

Acceptance testing is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behaviour of the system; the internal logic of the program is not emphasized.

Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once.

The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

Acceptance testing is performed along with the client to show that to see that all requirements are satisfied whatever may be the attributes its working well provided all the attributes are valid. If not it displays corresponding messages for getting valid attributes.

### **Alpha Testing:**

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

### **Beta Testing:**

Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Versions of the software, known beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the products have few faults or bugs.

Sometimes, beta versions are made available to the open public to increase the feedback filled to a maximal number of future users.



**Usability Testing:**

Usability testing is needed to check if the user interface is easy to use and understand. It is connected mainly with the use of the application.

**Security Testing:**

Security testing is essential for software that processes confidential data to prevent system intrusion by hackers.

**White Box Testing :**

This is the unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors.

We tested stepwise every piece of code, taking care that every statement in the code is executed at least once; the white box testing is also called glass box Testing.

**Black Box Testing**

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details as statement level.

Output for a given set of input combinations are forwarded other module.

## Test Cases:

TEST CASE ID	SCENARIO TO TEST	STEPS TO PERFORM	EXPECTED RESULT	ACTUAL RESULT	PASS/FAIL
TC1	Log-in into website as User/ Admin	1.Open the log in page of the Websiye. 2.Enter the valid user name. 3.Enter valid password. 4.Click on Log in button.	Website should except valid user name and valid password entered by user and should redirect user to respected dashboard.	Log in into Website is successful	Pass
TCL1.1	Log-in into Website	1.Open the log in page of the application. 2.Enter in-valid user name. 3.Enter valid password. 4.Click on Log in button.	Website should not accept invalid user name. Application should throw message "Invalid Credentials".	Log in denied with appropriate message	Pass
TC 2	If user is new to the system	1.Click on New Registration request on the dashboard. 2.User Registration tab will be opened . 3.Click on Register.	1.Validation errors should be displayed with respective fields of User Registration Page 2.Form should be redirected to next tabs.	Registration accepted with appropriate message	Pass

TC 2.1	If user is new to the system	1.Click on Registration request on the dashboard. 2.User Registration tab will be opened . 3.Fill all the required details.	Redirect user to User Dashboard	Redirected user to Homepage	Pass
TC 2.2	If user is new to the system	1.Click on Registration request on the dashboard. 2. Registration tab will be opened and fill all the required details,now click Register. 3. Registration tab will get open ,now click on Register.	1.Validation errors should be displayed with respective fields of User Registration. 2.Form should not be redirected to next tabs.	1.Validation errors displayed with respective fields of Registration page. 2.Form not redirected to next tabs.	Pass
TC2.3	If user is new to the system and if registering in the system	1.Click on Registration request on the dashboard. 2.User Registration tab will be opened and fill all the required	Form will get submitted.	Form Submitted	Pass

		<p>details,now click next.</p> <p>3.Travel Details tab will get opened .</p> <p>4.Select Domestic from Drop down Travel Type.</p> <p>5) Fill all the required details.</p>			
TC 2.4	If user is new to the system and if user is registering on site	<p>1.Click on Registration request on the dashboard.</p> <p>2.User Registration tab will be opened and fill all the required details,now click Register.</p> <p>3.Blog Details tab will get opened . displayed.</p>	<p>1.Validation errors should be displayed with respective fields of Registration Details.</p> <p>2.Form should not be redirected to next</p>	<p>1.Validation errors displayed with respected fields of Registration Details.</p> <p>2.Form not redirected to next</p>	Pass
TC 2.5	If user is Existing user	<p>1.Click on Registration request on the dashboard.</p> <p>2.User Registration tab will be opened .</p> <p>With filled in</p>	Form will be directed to Login Page.	Form redirected to Login Page	Pass

		information of User which can be edited by user.			
TC 3.1	If user Forgot Password	1.Click on Reset Password request on the Login Page. 2.Password Recovery page will be open . 3.With filled in information of User which can be edited by user.	Form will be directed to Reset Password Page.	Form redirected to Reset Password Page	Pass
TC 3.2	If user Forgot Password And enter correct mail	1.Click on Reset Password request on the Login Page. 2.Password Recovery page will be open . 3.With filled in information of User which can be	Form will be directed to Password Recovery Page.	Form will be directed to Password Recovery Page.	Pass

		edited by user.			
TC 3.3	If user Forgot Password And enter incorrect mail	1.Click on Reset Password request on the Login Page. 2.Password Recovery page will be open . 3.With filled in information of User which can be edited by user.	Form will not be directed to Password Recovery Page.	Form will not be directed to Password Recovery Page.	Pass
TC 4	If user got forgot password link on email	1.Click on Reset Password token in email 2.Password Reset page will be open . 3.With filled in information of User which can be edited by user.	Form will be directed to Login Page on Successfully resetting new password.	Form will be directed to Login Page on Successfully resetting new password.	Pass

TC 4.1	If user got forgot password link on email	<ol style="list-style-type: none"> <li>1.Click on Reset Password token in email</li> <li>2.Password Reset page will be open .</li> <li>3.User enter Password with invalid validations</li> </ol>	<ol style="list-style-type: none"> <li>1.Validation errors should be displayed with respective fields of Password Reset.</li> <li>2.Form should not be redirected to Login Page.</li> </ol>	<ol style="list-style-type: none"> <li>1.Validation errors should be displayed with respective fields of Password Reset.</li> <li>2.Form should not be redirected to Login Page.</li> </ol>	Pass
TC 5	If user Login in system	<ol style="list-style-type: none"> <li>1.Click on User Profile</li> <li>2.User Profile page will be open .</li> <li>3.User Can able to update his/her information/</li> </ol>	<ol style="list-style-type: none"> <li>User Data Updation form will be display.</li> <li>2.Form should be redirected to Same Page after updating the data.</li> </ol>	<ol style="list-style-type: none"> <li>User Data Updation form will be display.</li> <li>2.Form should not be redirected to Same Page after updating the data.</li> </ol>	Pass

TC 5.1	If user Login in system and update profile with invalid validations	<ol style="list-style-type: none"> <li>1.Click on User Profile</li> <li>2.User Profile page will be open .</li> <li>3.User Can able to update his/her information</li> </ol>	<ol style="list-style-type: none"> <li>1.User Data Updation form will be display.</li> <li>2. Validation errors should be displayed with respective fields.</li> <li>3.Form should not be redirected to Same Page after updating the data.</li> </ol>	<ol style="list-style-type: none"> <li>1.User Data Updation form will be display.</li> <li>2. Validation errors should be displayed with respective fields.</li> <li>3.Form should not be redirected to Same Page after updating the data.</li> </ol>	Pass
TC 6	If user Login in system and Click on Dashboard	<ol style="list-style-type: none"> <li>1.Click on Dashboard</li> <li>2.Respective user Dashboard will be open .</li> <li>3.User Can able to Mange all activities from the dashboard</li> </ol>	<ol style="list-style-type: none"> <li>1. Respective user Dashboard will be display.</li> <li>2. All Dashboard Tabs will be display.</li> </ol>	<ol style="list-style-type: none"> <li>1. Respective user Dashboard will be display.</li> <li>2. All Dashboard Tabs will be display.</li> </ol>	Pass
TC 6.1	If user Login in system and Click on Categories	<ol style="list-style-type: none"> <li>1.Click on Dashboard</li> <li>2.Respective user Dashboard will be open .</li> <li>3.Click on Categories Tab</li> </ol>	<ol style="list-style-type: none"> <li>1. Present Blog Categories Will be display/</li> <li>2. User can only view Categories if allowed.</li> </ol>	<ol style="list-style-type: none"> <li>1. Present Blog Categories Will be display/</li> <li>2. User can only view Categories if allowed.</li> </ol>	Pass



TC 6.1.1	If user Login in system and Click on new category	<ol style="list-style-type: none"> <li>1.Click on Categories</li> <li>2.All Blog categories will be display .</li> <li>3.Click on newCategories option</li> </ol>	<ol style="list-style-type: none"> <li>1. New Category form will be display</li> <li>2. User can Create new Categories if allowed.</li> <li>3.After new category creation user will redirected to the same page</li> </ol>	<ol style="list-style-type: none"> <li>1. New Category form will be display</li> <li>2. User can Create new Categories if allowed.</li> <li>3.After new category creation user will redirected to the same page</li> </ol>	Pass
TC 6.2	If user Login in system and Click on Blogs	<ol style="list-style-type: none"> <li>1.Click on Blogs</li> <li>2.All Blog created by respective will be display .</li> <li>3.Click on new Blog option</li> </ol>	<ol style="list-style-type: none"> <li>1. New Blog form will be display</li> <li>2. User can Create new Blogs if allowed.</li> <li>3.After new Blog creation user will redirected to the Homepage</li> </ol>	<ol style="list-style-type: none"> <li>1. New Blog form will be display</li> <li>2. User can Create new Blogs if allowed.</li> <li>3.After new Blog creation user will redirected to the Homepage</li> </ol>	Pass

TC 6.2.1	If user Login in system and Click on Blogs	<ol style="list-style-type: none"> <li>1. Click on Blogs</li> <li>2. All Blog created by respective will be display .</li> <li>3. Click on new Blog option and enter invalid data</li> </ol>	<ol style="list-style-type: none"> <li>1. New Blog form will be display</li> <li>2. User can Create new Blogs if allowed.</li> <li>3. Validation errors should be displayed with respective fields.</li> <li>4. After new Blog creation user will redirected to the Homepage</li> </ol>	<ol style="list-style-type: none"> <li>1. New Blog form will be display</li> <li>2. User can Create new Blogs if allowed.</li> <li>3. Validation errors should be displayed with respective fields.</li> <li>4. After new Blog creation user will redirected to the Homepage</li> </ol>	Pass
TC 6.2.2	If user Login in system and Click on Blogs and delete blog	<ol style="list-style-type: none"> <li>1. Click on Blogs</li> <li>2. All Blog created by respective will be display and Delete and update option will be display.</li> <li>3. Click on Delete Blog</li> </ol>	<ol style="list-style-type: none"> <li>1. On Click of delete option blog will be deleted from the system</li> <li>2. User can Delete any Blogs created by him/her if allowed.</li> </ol>	<ol style="list-style-type: none"> <li>1. On Click of delete option blog will be deleted from the system</li> <li>2. User can Delete any Blogs created by him/her if allowed.</li> </ol>	Pass

TC 6.2.3	If user Login in system and Click on Blogs and update blog	<ol style="list-style-type: none"> <li>1.Click on Blogs</li> <li>2.All Blog created by respective will be display and Delete and update option will be display.</li> <li>3.Click on Update Blog</li> </ol>	<ol style="list-style-type: none"> <li>1. Update Blog form will be display.</li> <li>2. User can update any Blogs created by him/her if allowed.</li> <li>3. Validation errors should be displayed with respective fields.</li> </ol>	<ol style="list-style-type: none"> <li>1. Update Blog form will be display.</li> <li>2. User can update any Blogs created by him/her if allowed.</li> <li>3. Validation errors should be displayed with respective fields.</li> </ol>	Pass
TC 6.3	If user Login in system and Click on Draft	<ol style="list-style-type: none"> <li>1.Click on Draft</li> <li>2.All Blog Drafted by respective will be display and Delete and update option will be display.</li> <li>3.Click on Delete Blog</li> </ol>	<ol style="list-style-type: none"> <li>1. On Click of delete option blog will be deleted from the system</li> <li>2. User can Delete any Blogs created by him/her if allowed.</li> </ol>	<ol style="list-style-type: none"> <li>1. On Click of delete option blog will be deleted from the system</li> <li>2. User can Delete any Blogs created by him/her if allowed.</li> </ol>	Pass
TC 6.3.1	If user Login in system and Click on Draft	<ol style="list-style-type: none"> <li>1.Click on Draft</li> <li>2.All Blog Drafted by respective will be display and Delete and update option will be display.</li> <li>3.Click on Update Blog</li> </ol>	<ol style="list-style-type: none"> <li>1. Update Blog form will be display.</li> <li>2. User can update any Blogs created by him/her if allowed.</li> <li>3. Validation errors should be displayed with</li> </ol>	<ol style="list-style-type: none"> <li>1. Update Blog form will be display.</li> <li>2. User can update any Blogs created by him/her if allowed.</li> <li>3. Validation errors should be displayed with</li> </ol>	Pass

			respective fields.	respective fields.	
TC 6.4	If user Login in system and Click on Editors choice	1.Click on Editors choice 2.All Blog which are in editor choice option will be display. 3.Click on Add Blog in editor choice	1.A Blog will be added as editors choice, 2.user will be redirected to the same page after adding blog in editor choice	1.A Blog will be added as editors choice, 2.user will be redirected to the same page after adding blog in editor choice	Pass
TC 6.4.1	If user Login in system and Click on Editors choice	1.Click on Editors choice 2.All Blog which are in editor choice option will be display. 3.Click on Delete blog in editor choice.	1.A Blog will be remove from editors choice, 2.user will be redirected to the same page after deleting blog from editor choice	1.A Blog will be remove from editors choice, 2.user will be redirected to the same page after deleting blog from editor choice	Pass
TC 6.5	If user Login in system and Click on Manage users if allowed	1.Click on Manage users 2.All users which are in system will be display. 3.Click on Delete user if allowed	1.A User will be remove from system, 2.user will be redirected to the same page after deleting user from system	1.A User will be remove from system, 2.user will be redirected to the same page after deleting user from system	Pass

TC 6.5.1	If user Login in system and Click on Manage users if allowed	1.Click on Manage users 2.All users which are in system will be display. 3.Click on Update user if allowed	1.A User will be update from system, 2.user will be redirected to the same page after update user from system	1.A User will be update from system, 2.user will be redirected to the same page after update user from system	Pass
TC 6.5.1.1	If user Login in system and update user if allowed	1.Click on Manage users to block or unblock user. 2.set user access permissions to respective user	1.A User will be updated either block or unblock, 2. User Accessibilities will be set for respective user	1.A User will be updated either block or unblock, 2. User Accessibilities will be set for respective user	Pass
TC 6.6	If Admin Login in system and see reports	Click on reports to see site statistics.	1.Admin will be see all site statistics in chart format, 2. After seeing all data admin can redirect to dashboard	1.Admin will be see all site statistics in chart format, 2. After seeing all data admin can redirect to dashboard	Pass

TC 7	If user Login in system and Rate the blog	1.Click on ratings to give ratings for respective blog.	1.A Blog ratings will be updated on single blog page, 2. User will be redirected to the homepage.	1.A Blog ratings will be updated on single blog page, 2. User will be redirected to the homepage.	Pass
------	---	---	--	--	------

TC 8	If user Login in system and Comment on blog	1.Click on Add Comment to Add Comment for respective blog.	1.A Blog Comment will be updated on single blog page, 2. User will be redirected to the homepage.	1.A Blog Comment will be updated on single blog page, 2. User will be redirected to the homepage.	Pass
------	---	--	--	--	------

**CHAPTER 4**  
**USER MANUAL**

## 4.1 User Manual:

For any system to be successful it is important that the intended user find the system easy to operate. The purpose of the user manual is to make user acquainted with the system and help user understand the system and operate it conveniently. The User Manual is prepared reflexively because it is an item that must accompany every system.

The manual contain several screenshots that describes how to use the entire system. This Manual helps user to navigate efficiently through the system and help user to solve issues wherever they occur.

Information about the system.

The system contains following users:

- 1.User
- 2.User As Admin Facilities
- 3.Super Admin

The System has following features:

1. User/Super Admin Dashboard
2. Creation of new Blog/Post
3. Block/Unblock User
4. Commenting System
5. Rating System
6. Updating various Blogs, Categories, User Data.
7. Editing/Updating User Access Permission
8. Displaying All Types of Posts ,Comments , etc on the homepage and on the single blog page.

Log in into the system using Login credentials. You are redirected to dashboard after successful credential verification of Username and Password.

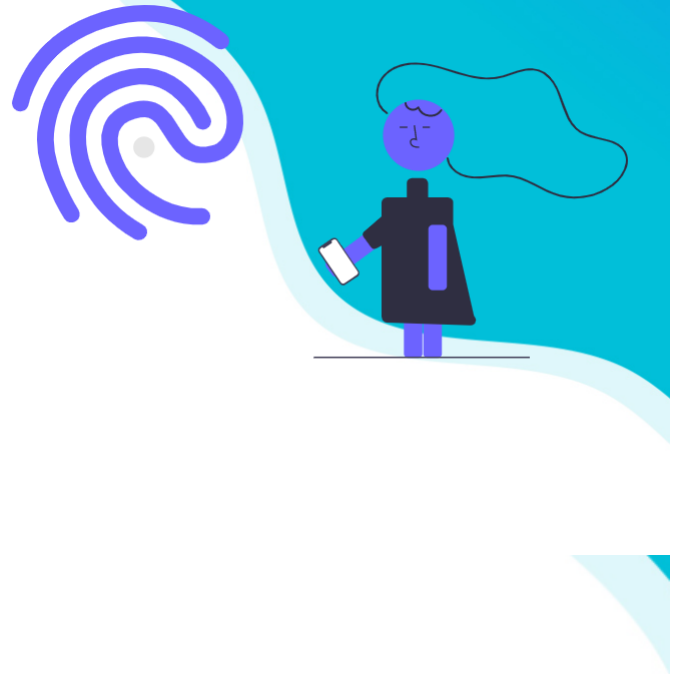


## Login Here...

Login

Forgot Password? [Reset Password](#).

Don't have an account? [Sign in](#).



# Welcome Kaustubh

Express your thoughts...

Logout

Profile

Dashboard

Dashboard

This tab denotes the dashboard, which contains all the Blog related activities made by the user.

Profile

If the login person is a user then this tab denotes the Profile Section of the user.

 Blogs


This tab in the dashboard is used to view currently published/Available Blogs in the system

 Draft

This tab in the dashboard is used to view blogs/posts which are saved as draft.

 Categories

This tab in the dashboard is used to view currently available blog categories in the system. In this Admin can create new categories which are available for the users.

 Editor's Choice

This tab in the dashboard is used to view the blogs which are made as editor's choice by the Admin and in this Admin can make any blog as editor's choice blog.

 Manage Users

This tab in the dashboard is used to view the users In the system and Admin can manage an individual user by using this tab in the dashboard.

+ ADD NEW BLOG

This Button in the Blogs tab used to create a new blog in the system by user or admin.

+ ADD NEW CATEGORY

This Button in the category tab used to create a new category in the system by user or admin.

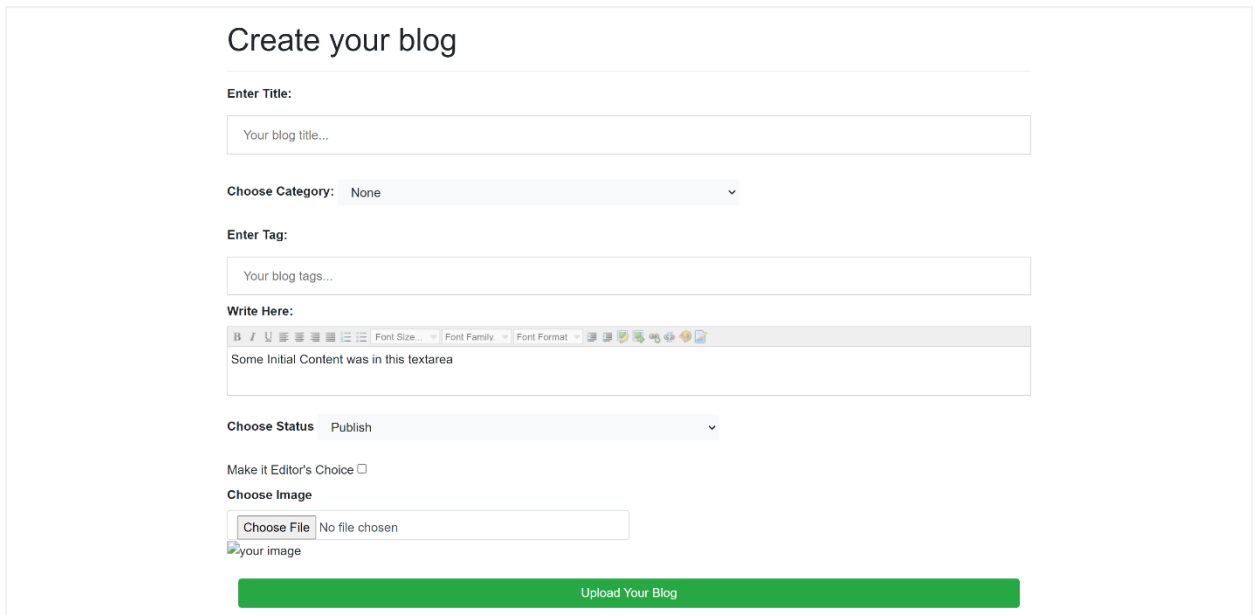
+ ADD NEW USER

This Button in the Manage User tab used to create a new User in the system by the admin.

### Utilities:

This “UPLOAD Your Blog” button is used to upload Blogs/Posts by certain category and status as publish in the system. etc.

Blog in the system is create by the following window as follows:



















The screenshot shows a web form titled "Create your blog". It contains the following fields and elements:

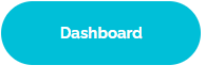








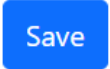


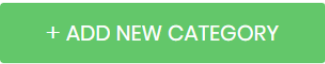
- Enter Title:** A text input field with the placeholder "Your blog title...".
- Choose Category:** A dropdown menu currently set to "None".
- Enter Tag:** A text input field with the placeholder "Your blog tags...".
- Write Here:** A rich text editor with a toolbar (bold, italic, underline, list, link, font size, font family, font format) and the text "Some Initial Content was in this textarea".
- Choose Status:** A dropdown menu currently set to "Publish".
- Make it Editor's Choice:** A checkbox that is currently unchecked.
- Choose Image:** A file selection area with a "Choose File" button, the text "No file chosen", and a small image icon labeled "your image".
- Upload Your Blog:** A prominent green button at the bottom of the form.

## 4.2 Operations Manual:

There are various symbols and buttons on the web application. Here are their descriptions :

Symbol	Meaning
	Denotes to navigate on to the homepage.
	Denotes the author of the blog on the homepage .
	Denotes the Published Date of the blog in the system
	This Symbol shows the sharing option of the blog on various social media platform.
	Denotes the Tag of the perticular published blog in the system
	This rating symbol is used to rate the blog in the system.
	Denotes the Categories of blog on the homepage.
	Used to display (Location)
	Used to display(Email) of the system
	Used to display(Call) in the system
	This Denotes the Blog is made as Editor's Choice by the Admin.
 Blogs	This Symbol used to show all blogs available in the system.
 Categories	This Symbol used to show all categories available in the system.
 Draft	This Symbol used to show all Draft blogs available in the system.
 Editor's Choice	This Symbol used to show all blogs Which are made as Editor's Choice in the system.

 Manage Users	This Symbol used to show all the users available in the system.
--	---

Buttons	Meaning
	Go to the dashboard of the user/Admin
	Go to the Profile of System User
	Logout the login user from the system
	Used to Add Comment on the blog.
	Used to Add Blog in Editor's choice.
	Used to Cancel any action perform in profile section
	Used to Save any data changes in the system.
	Used to Update any data changes in the system.
	Used to Delete any data changes in the system.
<input type="checkbox"/> Unlocked 	This Button is used to block or unblock the user from the system.
<input type="button" value="Choose File"/> No file chose	Pick a file from the system / Browse file from the system.
<a href="#">Read More</a>	Navigate the user to the single blog page.
	This used to toggle user access permissions in the system.
	Adds new blog in the system
	Adds new category in the system

<p>+ ADD NEW USER</p>	<p>Adds new user in the system by Admin</p>
-----------------------	---

## 4.3 Program Specifications:

### 1) User Registration Details

Module	User Registration Module
Program Name	User Registration Details
Purpose	Creates a new user in system with valid data and information on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The details of the users are stored in the Membership_user table.

### 2) Add Blog Details

Module	User/Admin Blog Creation Module
Program Name	Create New Blog
Purpose	Creates a new blog in system with valid data and images to publish on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	The details of the Blogs are stored in the Blog_records table.

### 3) Update Profile Details

Module	Update User Profile
Program Name	Update the system user profile data
Purpose	Updated a new User in system with valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Profile updation details are stored in the Membership_user table.

#### 4) Add Categories

Module	New Category Creation for Blogs
Program Name	Create the new blog category in the system
Purpose	Created a new new blog category in system with image and valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Categories details are stored in the blog_categories table.

#### 5) Update Blogs

Module	Blog Updation Module
Program Name	Update the blog in the system
Purpose	Updates a existing craeted blog in system with image and valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Updated blog details are stored in the blog_records table.

#### 6) Add User

Module	Add User Module
Program Name	Add New user
Purpose	Add a new user in system with image and valid data on the site by admin.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Newly added user details are stored in the membership_user table.



### 7) Add Comment:

Module	Comment Management Module
Program Name	Add a Comment on blog
Purpose	Adds a new comment or reply on existing comment in system valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Newly added comment details are stored in the Comments table.

### 8) Add Ratings:

Module	Ratings Management Module
Program Name	Add a rating on blog
Purpose	Adds a new Rating on existing blog in system valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Newly added ratings details are stored in the Ratings table.

### 9) Editor's Choice Blogs:

Module	Editor's choice Management Module
Program Name	Make Editor choice blogs
Purpose	Adds a new or existing blog as Editor's choice section in system valid data on the site.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Update Editor's choice blog details are stored in the blog_records table.

## 10) Adds Users Permissions:

Module	User Access Permission Management Module
Program Name	Add user Permissions
Purpose	Adds a permission to any user where he/she can able to acces some part of system or not.
Input Details	The required fields should not be blank and the user should provide valid data for each field.
Output	Added Permissions details are stored in user_permissions table.

## **5. Drawbacks & Limitations:**

1. Forget Password Functionality is not available on mail (Gmail).
2. Site is not fully Responsive on all module of the system.
3. The system doesn't have grammatical corrections and word fill functionalities in the blogs writing section.
4. User doesn't have fully access of the system.
5. The system doesn't have user and admin chat functionality.
6. Currently no any testimonial section is available for users in the system.
7. Currently System doesn't have report section for admin to view user's statistics.
8. System doesn't contain any group membership module to create and manage a community by the Admin.

## **6. Proposed Enhancements:**

1. Add pin blog in the user account to read it later.
2. Add Follow a author/user in the system and user will get the notification if he/she publish any article/blog
3. Add comment notification module: whenever any user add comment on another blog the blog author will get notification about the comment.
4. User will login by his/her google account and save his/her writing on the drive and can access his/her content which is store on cloud.
5. Add Cloud platform for user to make system more useful to the users.
6. Will add new features for security of Data
7. Ensure Data Accuracy.
8. Good Efficiency
9. Try to make User friendliness and interactive
10. Minimum time required
11. Better service

## **7.Conclusion:**

All the requirements stated by the user have been addressed in this web application. The application includes the following :

### **System User:**

1. Login to the system using login credentials.
2. Create New Blog, Categories
3. Save, update, edit Blogs in the system.
4. User can manage his/her profile.
5. User can manage add delete edit it's own posts.
6. User can search particular posts in a system.

### **Super Admin:**

1. Admin has it's own functionality admin is here we can say super admin
2. Like user he can perform all activities which user performs
3. Rather than this admin can do some more like below
  - Admin can login in system
  - Admin can manage users
  - Admin can delete specific user
  - Admin can make any Posts as editor's choice
  - Admin can make comment on any blog
  - Admin can have full control over system
  - Admin can delete specific users post if its is not good
  - Admin can search particular posts in a system

The application is tested successfully.

As we know, that any project, even on completion, requires constant improvement and changes which gives way for release of new version.

This application is very user-friendly.

For developing this application, technologies such as HTML, CSS, PHP, Bootstrap are used which are in great demand in IT market currently for developing Web Applications.

## **8.Bibliography:**

### **Books:**

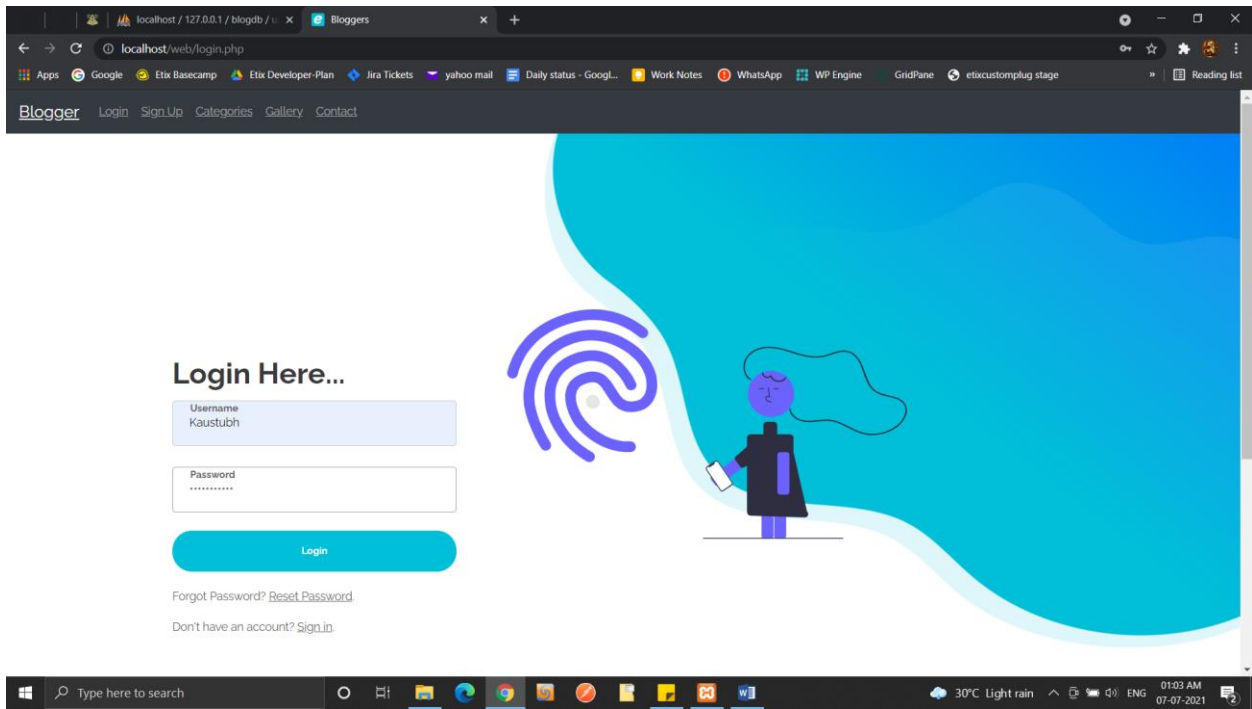
PHP & MySQL Web Development – by Luke Welling & Laura Thompson

### **Websites:**

1. PHP mother site @ <http://php.net/>.
2. PHP Manual @ <http://php.net/manual/en/>.
3. PHP Language Reference @ <http://php.net/manual/en/langref.php>.

**ANNEXURE 1**  
**USER INTERFACE SCREENS**

# 1. Login Page





## 2. Signup Page

[Blogger](#) [Login](#) [Sign Up](#) [Categories](#) [Gallery](#) [Contact](#)

### Register Here...

Please fill in this form to create an account.

**Username**  
Kaustubh

**Email**  
kaustubhd5@gmail.com

**Password**  
.....

**Repeat Password**  
.....


**Your Address**  
new york street

**Enter City**  
Pune

**Mobile No**  
9922379041

**Register**

Already have an account? [Sign in](#)



**About Blogger**  
We're passionate about offering some of the best business growth services for startups

**Important Links**

- Our business partners [abc.com](#)
- Read our [Terms & Conditions](#) [Privacy Policy](#)

**Social Media**  
[f](#) [t](#) [G+](#) [i](#) [in](#)

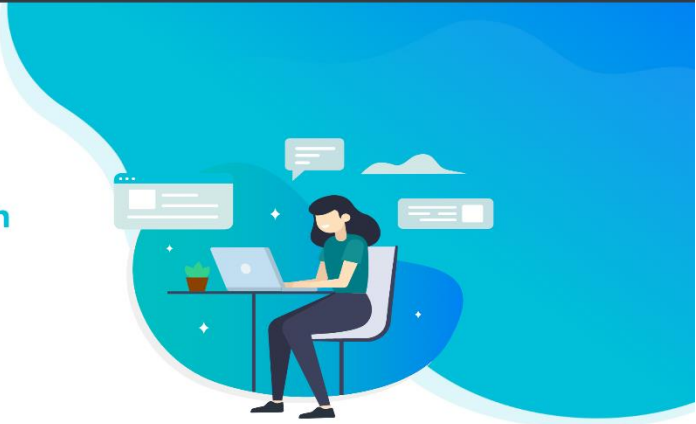
# 3.Homepage:

[Blogger](#) [Blog](#) [Categories](#) [Gallery](#) [Contact](#)

## Welcome Kaustubh

Express your thoughts...

[Logout](#) [Profile](#) [Dashboard](#)



Search this blog

**categories**

- Classic Literature
- Mythology
- Comedy
- Economics
- Drama
- Poetry
- Biography
- Society
- Society

**Popular Posts** 0 results

### This month blogs

Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea

Thanks

- cricket
- May 29, 2021
- admin
- Programming

ratings: 0 / 5

[Read More](#)

### Creative Minds

A creative mind is the one that sees information – particularly from the world around it – and relates it to their experiences and existing knowledge in order to create. Create what exactly? Solutions to problems, expressions for feelings and situations, objects that inspire and motivate, innovations, and nearly anything...

#creative

- April 09, 2021
- Kaustubh
- education

ratings: 0 / 5

[Read More](#)

### Historical Blog

Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea

#history

- April 04, 2021
- Kaustubh
- Historical Fiction

ratings: 3.3 / 5

[Read More](#)

### Blog 1

Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea Some Initial Content was in this textarea

- cricket
- April 04, 2021
- admin
- education

ratings: 4 / 5

[Read More](#)

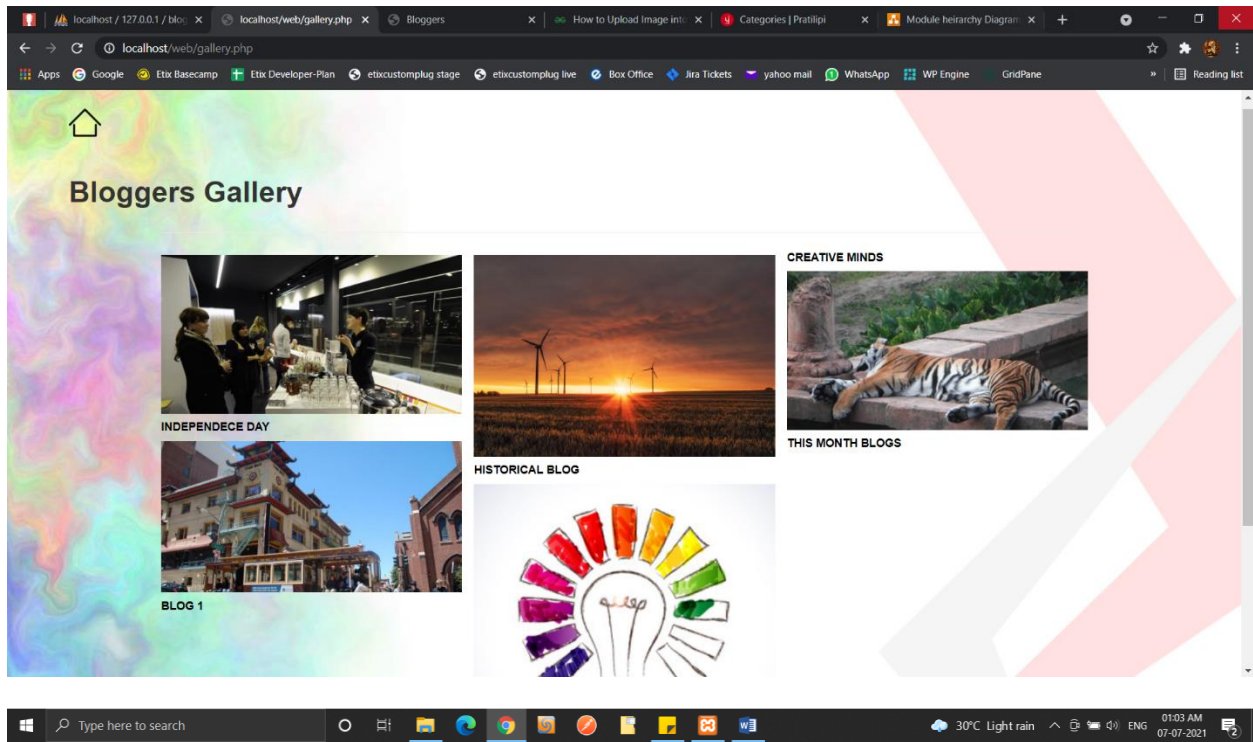
### Independence Day

- patriot
- March 20, 2021
- Brett
- Programming

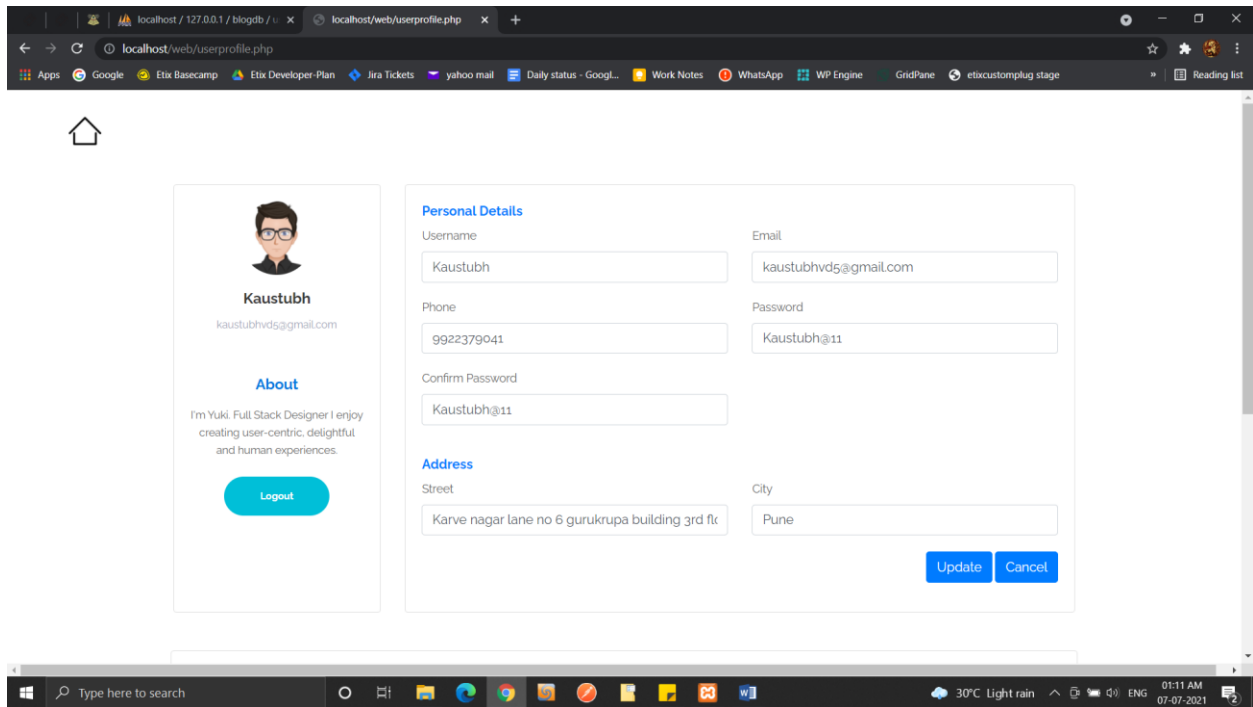
ratings: 2.98 / 5

[Read More](#)

## 4. Gallery Page



## 5. User Profile Page:



The screenshot displays a web browser window with the URL `localhost/web/userprofile.php`. The page features a user profile card on the left and a form for updating personal details on the right. The profile card includes a profile picture of a man with glasses, the name 'Kaustubh', the email 'kaustubhvd5@gmail.com', an 'About' section with a bio, and a 'Logout' button. The form is titled 'Personal Details' and contains fields for Username, Email, Phone, Password, Confirm Password, Address (Street and City), and 'Update' and 'Cancel' buttons.

**Personal Details**

Username:

Email:

Phone:

Password:

Confirm Password:

**Address**

Street:


City:

## 6. Search Page:

localhost / 127.0.0.1 / blogdb / ... x Bloggers

localhost/web/search.php?search=blog


### Search Result for "blog"



#### ENVIRONMENTAL BLOG

Since 1999, millions of people have expressed themselves on Blogger. From ...

Trees #Environment  
30/06/2021  
Kautubh  
Life  
[Read More](#)



#### INDEPENDENCE DAY 1

From New York to Jakarta, Blogger is one of the most popular ways to publi ...

independence 1  
30/06/2021  
Kautubh  
[Read More](#)

Type here to search

30°C AQI 58 01:14 AM 07-07-2021

## 7. Blogs by categories page:

The screenshot displays a web browser window with the following elements:

- Browser Tab:** Bloggers
- Address Bar:** localhost/web/category\_blog.php?CName=Life
- Page Title:** Life Blogs
- Blog Entry 1:**
  - Image:** A circular logo for 'HOUSE OF ROCK' featuring a rooster and the text 'corpus christi. tx'.
  - Title:** ENVIRONMENTAL BLOG
  - Text:** Since 1999, millions of people have expressed themselves on Blogger. From detailed posts about almost every apple variety you could ever imagine to a blog dedicated to the art of blogging itself, the ability to easily share, publish and express oneself on the web is at the ...
  - Metadata:** Trees #Environment, 30/06/2021, Kautubh, Life
  - Action:** [Read More](#)
- Blog Entry 2:**
  - Image:** A photograph of a person in a dimly lit room.
  - Title:** CUP OF TEA
  - Text:** I have been following Elise's blog since before I even knew what a blog was. (Yeah, way back before blog readers, she was a favorite "bookmark" in my browser!) And in that weird way that the internet brings people together, I kind of feel like we've "grown up together" over this past decade. It's been fun to watch her transition...
  - Metadata:** Life, 30/06/2021, Kautubh, Life
- Windows Taskbar:** Search bar, taskbar icons, system tray showing 30°C AQI 58, and date/time 01:12 AM 07-07-2021.

## 8. Contact us page:

localhost / 127.0.0.1 / blogdb / ... x Bloggers

localhost/web/contact.php

Blogger Blog Categories Gallery Contact

### Contact Information

Don't hesitate to give us a call or send us a contact form message

22 Innovative Area, Delhi, CA 94043, India [+81 720 2212](tel:+817202212) [office@blogger.com](mailto:office@blogger.com)

San Francisco  
California, USA  
View larger map

Name

Email

Your message

I have read and agree with Evoto's stated [Privacy Policy](#) and [Terms Conditions](#)

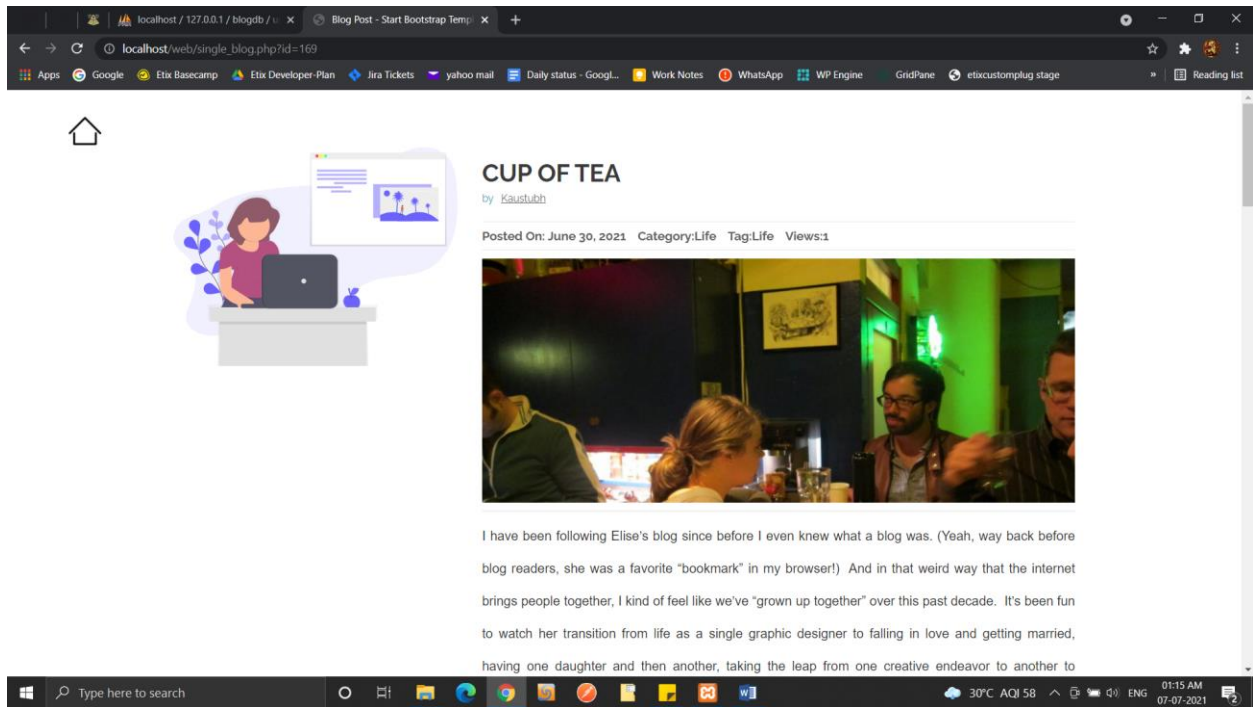
message

Type here to search

30°C AQI 58

01:14 AM  
07-07-2021

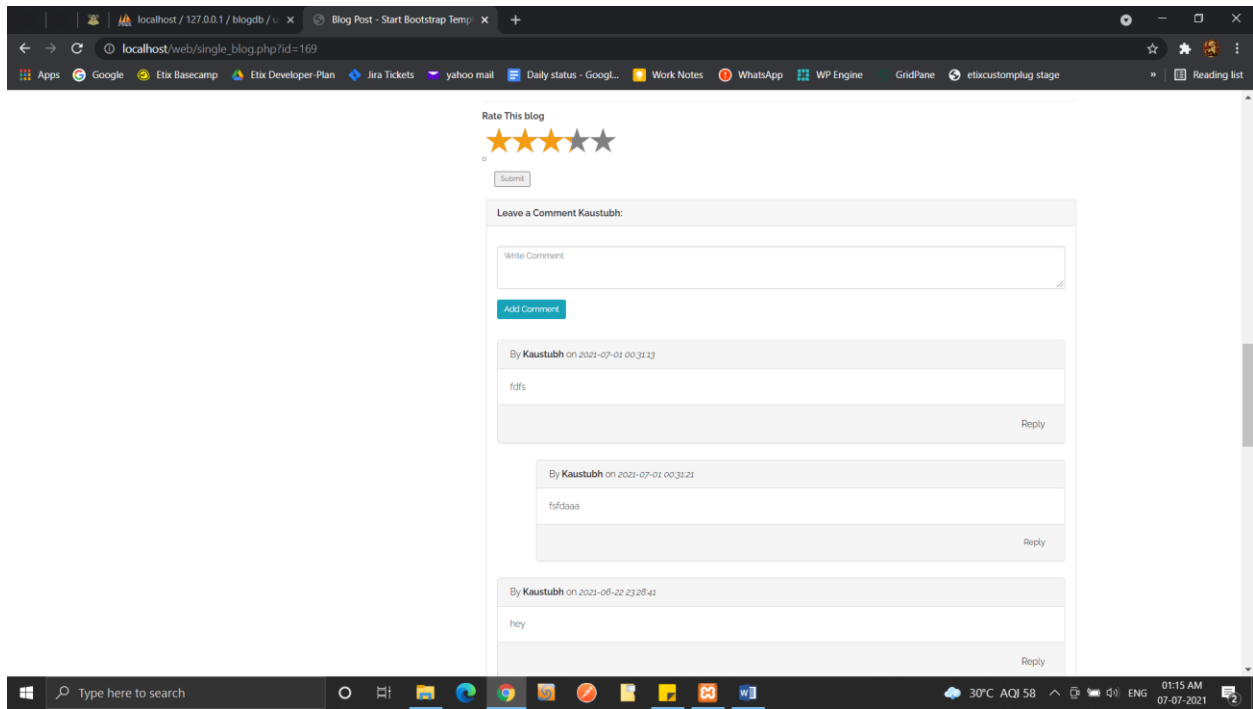
## 9.Single blog page:



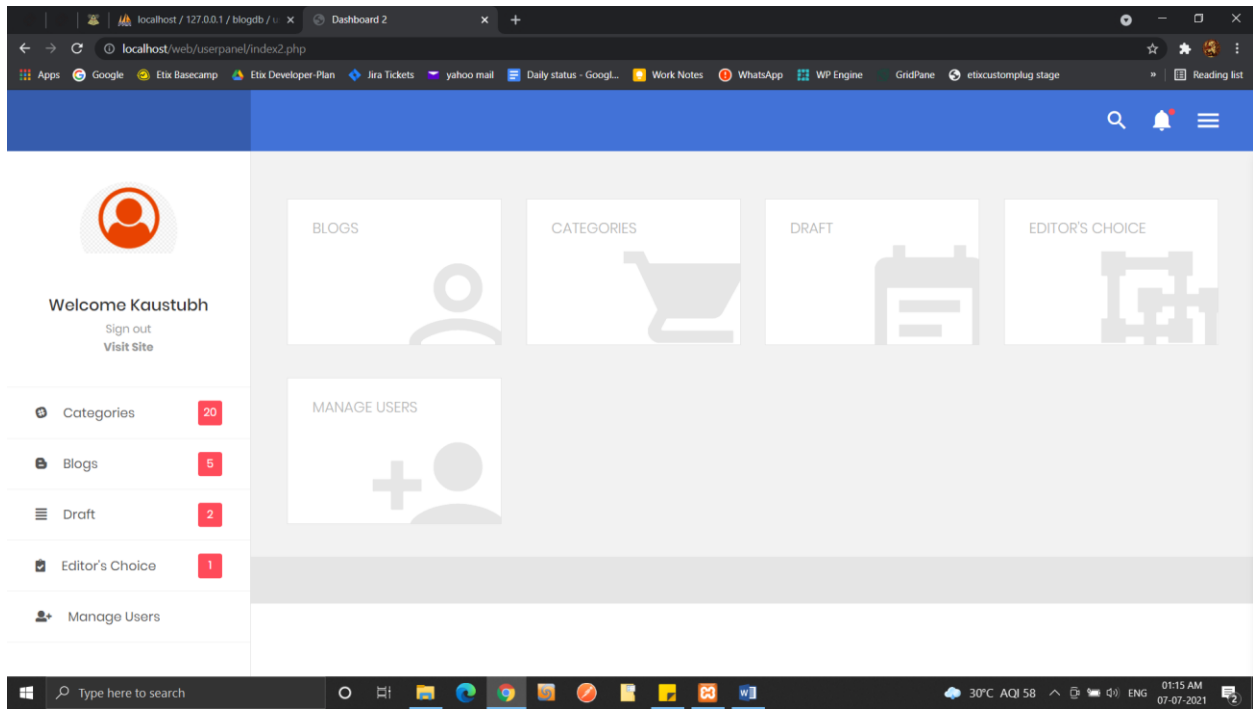
The screenshot shows a web browser window displaying a single blog post. The browser's address bar shows the URL `localhost/web/single_blog.php?id=169`. The page features a home icon in the top left corner. The main content area includes an illustration of a person working at a laptop, the title "CUP OF TEA" by Kaustubh, and a photograph of people at a bar. The text of the post begins with "I have been following Elise's blog since before I even knew what a blog was. (Yeah, way back before blog readers, she was a favorite 'bookmark' in my browser!) And in that weird way that the internet brings people together, I kind of feel like we've 'grown up together' over this past decade. It's been fun to watch her transition from life as a single graphic designer to falling in love and getting married, having one daughter and then another, taking the leap from one creative endeavor to another to



## 10. Comments and rating of blog:



# 11 User Dashboard:



## 12 . Categories page:

The screenshot shows a web dashboard for managing categories. The browser address bar indicates the URL is `localhost/web/userpanel/admin_categories.php`. The dashboard has a blue header with a search icon, a notification bell, and a menu icon. Below the header, there is a breadcrumb trail: "You are here: Home / Dashboard". A green button labeled "+ ADD NEW CATEGORY" is located in the top right corner of the main content area.

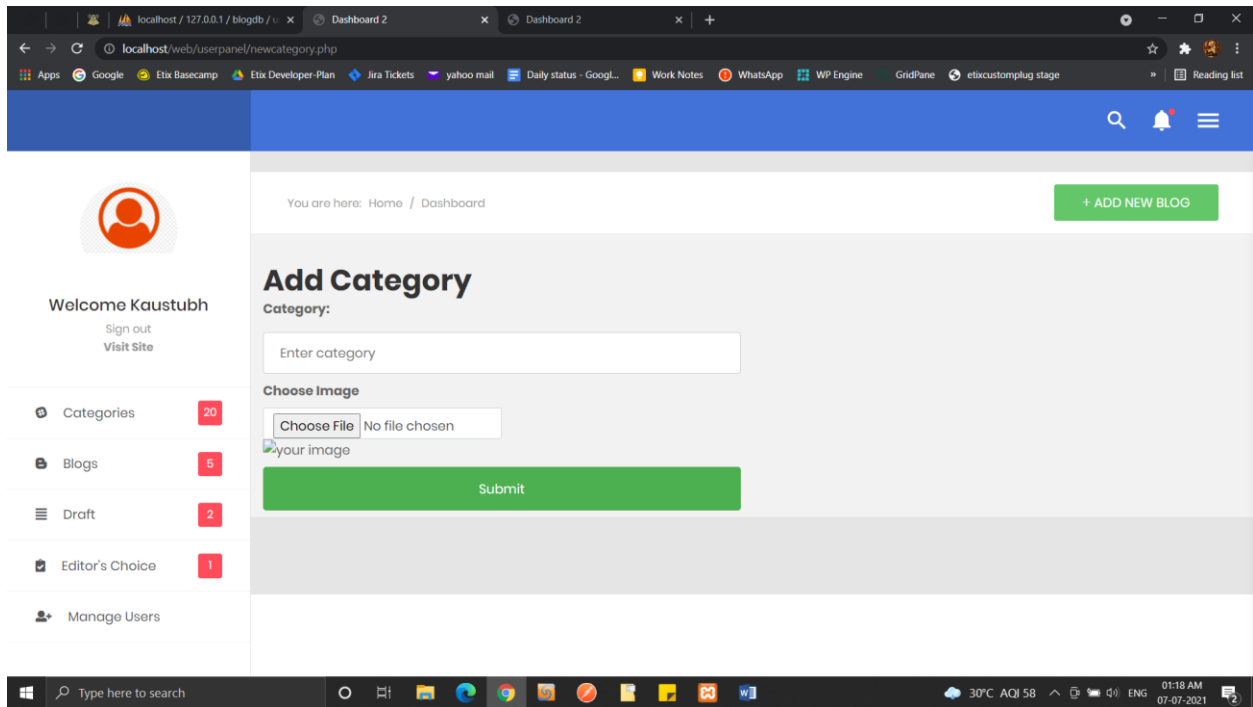
The main content area features a table with the following data:

	Id	category
<input type="checkbox"/>	141	Classic Literature
<input type="checkbox"/>	142	Historical Fiction
<input type="checkbox"/>	143	Mythology
<input type="checkbox"/>	144	Poetry
<input type="checkbox"/>	145	Science Fiction
<input type="checkbox"/>	146	Biography
<input type="checkbox"/>	147	Society
<input type="checkbox"/>	148	Comedy
<input type="checkbox"/>	149	Life
<input type="checkbox"/>	150	Drama

The left sidebar contains a user profile for "Kaustubh" with a "Sign out" and "Visit Site" link. Below the profile are several menu items with red notification badges: "Categories" (20), "Blogs" (5), "Draft" (2), "Editor's Choice" (1), and "Manage Users".

The Windows taskbar at the bottom shows the system tray with a search bar, task view, and system icons including weather (30°C), network, and time (01:18 AM, 07-07-2021).

## 13 Add New Category Page:



# 14 users blogs page:

The screenshot shows a web dashboard for managing blogs. The browser address bar indicates the URL is localhost/web/userpanel/adminblogs.php. The dashboard has a blue header with a search icon, a notification bell, and a menu icon. Below the header, there is a breadcrumb trail: "You are here: Home / Dashboard" and a green button labeled "+ ADD NEW BLOG".

The main content area features a table with the following columns: Id, Title, category, Blog Status, Author, Delete Blog, Image, and Update Blog. The table contains three rows of data:

Id	Title	category	Blog Status	Author	Delete Blog	Image	Update Blog
<input type="checkbox"/>	ENVIRONMENTAL BLOG	Life	Publish	Kaustubh	<a href="#">Delete</a>		<a href="#">Update</a>
<input type="checkbox"/>	INDEPENDENCE DAY 1		Publish	Kaustubh	<a href="#">Delete</a>		<a href="#">Update</a>
<input type="checkbox"/>	CUP OF TEA	Life	Publish	Kaustubh	<a href="#">Delete</a>		<a href="#">Update</a>

On the left sidebar, there is a user profile for "Kaustubh" with a "Sign out" and "Visit Site" link. Below the profile are several menu items with red notification badges: "Categories" (20), "Blogs" (5), "Draft" (2), "Editor's Choice" (1), and "Manage Users".

The Windows taskbar at the bottom shows the search bar, several application icons, and system tray information including temperature (30°C), AQI (58), and the date/time (01:18 AM, 07-07-2021).

## 15 Create Blogs Page:



## 16 Saved Draft page:

The screenshot displays a web browser window with the URL `localhost/web/userpanel/admin_draft.php`. The dashboard header includes a search icon, a notification bell, and a menu icon. Below the header, the breadcrumb path is "You are here: Home / Dashboard" and a green button labeled "+ ADD NEW BLOG" is visible.

	Id	Title	Tag	Category	Author	DateTime	Image	Remove blog	Update blog
<input type="checkbox"/>	168	INDEPENDENCE DAY 1	indepence 1		Kaustubh	2021-06-30 00:20:36		Delete	Update
<input type="checkbox"/>	169	CUP OF TEA	Life	Life	Kaustubh	2021-06-30 00:23:25		Delete	Update

The left sidebar contains the following navigation items:

- Categories: 20
- Blogs: 5
- Draft: 4
- Editor's Choice: 1
- Manage Users

The user profile section on the left shows a profile picture, the name "Welcome Kaustubh", and links for "Sign out" and "Visit Site".

The Windows taskbar at the bottom shows the search bar, system tray with weather (30°C, AQI 58), and the date/time (01:23 AM, 07-07-2021).

## 17. Editor's Choice Page:

The screenshot shows a web browser window displaying a dashboard. The browser's address bar shows the URL `localhost/web/userpanel/admin_editchoice.php`. The dashboard has a blue header with a search icon, a notification bell, and a menu icon. Below the header, there is a navigation breadcrumb "You are here: Home / Dashboard" and a green button labeled "+ ADD NEW BLOG".

On the left side, there is a user profile section for "Kaustubh" with a "Sign out" and "Visit Site" link. Below this is a sidebar menu with the following items:

- Categories: 20
- Blogs: 5
- Draft: 4
- Editor's Choice: 4
- Manage Users

The main content area features a dropdown menu labeled "Open this select blog" with an "Add" button below it. Below the dropdown is a table of selected blogs:

Id	Editor's choice blogs	Remove from editor choice
<input type="checkbox"/>	167 ENVIRONMENTAL BLOG	Remove
<input type="checkbox"/>	168 INDEPENDENCE DAY 1	Remove
<input type="checkbox"/>	169 CUP OF TEA	Remove
<input type="checkbox"/>	171 THE EVERYGIRL	Remove

The Windows taskbar at the bottom shows the system tray with a search bar, task view, and system icons including temperature (30°C), AQI (58), and time (01:23 AM, 07-07-2021).



## 18. Manage Users Page:

The screenshot displays the 'Manage Users' page of a web application. The browser address bar shows the URL `localhost/web/userpanel/manageusers.php`. The page features a blue header with search, notification, and menu icons. Below the header, a breadcrumb trail reads 'You are here: Home / Dashboard', and a green '+ ADD NEW USER' button is positioned in the top right corner.

The main content area contains a table of users with the following columns: Id, Name, Email, Address, City, Mobile no, Delete Record, and Update Record. The table lists four users:

	Id	Name	Email	Address	City	Mobile no	Delete Record	Update Record
<input type="checkbox"/>	72	David	david@gmail.com	new york strret	Verlin	8822229429	Delete	Update
<input type="checkbox"/>	75	kurt	kurt@gmail.com	1722 Linden St	Chicago	6622993361	Delete	Update
<input type="checkbox"/>	76	Brett	brett@gmail.com	1722 Linden St	New York	5598320423	Delete	Update
<input type="checkbox"/>	78	admin	admin@gmail.com	1722 Linden St	New York	8273837384	Delete	Update

On the left side, a sidebar displays the user profile for 'Kaustubh' with a 'Sign out' and 'Visit Site' link. Below the profile, a list of navigation items is shown with red notification badges: Categories (9), Blogs (7), Draft (2), Editor's Choice (5), Manage Users (5), and Contacts (2).

## 19. Add user page:

The screenshot displays the 'Add User' page of a web application. The interface includes a blue header with search, notification, and menu icons. A left sidebar shows a user profile for 'Kaustubh' and a list of menu items with counts: Categories (9), Blogs (7), Draft (2), Editor's Choice (5), Manage Users (5), Contacts (2), and Report. The main content area is a form for adding a new user, with fields for Username, Email, Password, Repeat Password, Address, City, and Phone. A green 'Add User' button is at the bottom.

### Add User

**Username**  
Enter Username

**Email**  
Enter Email

**Password**  
Enter Password

**Repeat Password**  
Repeat Password

**Address**  
Enter Address

**City**  
Enter City

**Phone**  
Enter Phone number

Add User

## 20. Contacts Data Page:

The screenshot shows a web browser window displaying a dashboard. The browser's address bar shows the URL `localhost/web/adminpanel/contactinfo.php`. The dashboard has a blue header with a search icon, a notification bell, and a menu icon. Below the header, there is a breadcrumb trail: "You are here: Home / Dashboard".

The main content area features a table with the following data:

	Id	Name	Email	Message	Delete Message
<input type="checkbox"/>	2	Kaustubh Dalal	kaustubhd5@gmail.co	hii its awsome	<a href="#">Delete</a>

On the left side of the dashboard, there is a sidebar with a user profile section and a list of menu items:

- Welcome Admin  
Sign out  
Visit Site
- Categories: 20
- Blogs: 5
- Draft: 4
- Editor's Choice: 6
- Manage Users: 7
- Contacts: 1

The Windows taskbar at the bottom shows the search bar, task view, and various application icons. The system tray on the right indicates a temperature of 30°C, an AQI of 58, and the date/time: 01:24 AM, 07-07-2021.

# Database Tables Screenshots:

## 1.Database Tables:

The screenshot displays the phpMyAdmin interface for a database named 'blogdb'. The left sidebar shows a tree view of databases and tables. The main area shows a table structure view with the following data:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> blog_categories	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	20	InnoDB	utf8mb4_general_ci	6.0 KiB	-
<input type="checkbox"/> blog_records	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	5	InnoDB	utf8mb4_general_ci	48.0 KiB	-
<input type="checkbox"/> comments	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	35	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> contact	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> editors_choice	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	6	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> membership_users	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	7	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> member_permissions	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	72	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> ratings	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	47	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<input type="checkbox"/> user_counter	<a href="#">Browse</a> <a href="#">Structure</a> <a href="#">Search</a> <a href="#">Insert</a> <a href="#">Empty</a> <a href="#">Drop</a>	1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
<b>9 tables</b>	<b>Sum</b>	<b>194</b>	<b>InnoDB</b>	<b>utf8mb4_general_ci</b>	<b>6.2 KiB</b>	<b>0 B</b>

Below the table list, there is a 'Create table' form with the following fields:

- Name:
- Number of columns:
- Go button

## 2.Users Data Table:

The screenshot shows the phpMyAdmin interface for a database named 'blogdb'. The 'membership\_users' table is selected, and the 'Structure' tab is active. The table contains 5 rows of data. The SQL query shown is 'SELECT \* FROM `membership\_users`'. Below the query, there is an 'UPDATE' statement: 'UPDATE `membership\_users` SET `CPassword` = \*f637cbcaa905a08d4b9a03165ecc1f\* WHERE `membership\_users`.`ID` = 78;'. The table data is displayed in a grid with columns: ID, Username, Email, Password, CPassword, Address, City, Phone, and Is\_ApprovedBanned. The rows are as follows:

ID	Username	Email	Password	CPassword	Address	City	Phone	Is_ApprovedBanned
72	David	david@gmail.com	f637cbcaa905a08d4b9a03165ecc1f	f637cbcaa905a08d4b9a03165ecc1f	new york strret	Verlin	8822229429	0
75	kurt	kurt@gmail.com	f637cbcaa905a08d4b9a03165ecc1f	f637cbcaa905a08d4b9a03165ecc1f	1722 Linden St	Chicago	6622993361	0
76	Brett	brett@gmail.com	f637cbcaa905a08d4b9a03165ecc1f	f637cbcaa905a08d4b9a03165ecc1f	1722 Linden St	New York	5598320423	0
77	Kaustubh	kaustubhd5@gmail.com	f637cbcaa905a08d4b9a03165ecc1f	f637cbcaa905a08d4b9a03165ecc1f	Karve nagar lane no 6 gurukrupa building 3rd floor...	Pune	9922379041	0
78	admin	admin@gmail.com	f637cbcaa905a08d4b9a03165ecc1f	f637cbcaa905a08d4b9a03165ecc1f	1722 Linden St	New York	8273837384	0

### 3.Blogs Record Table:

The screenshot shows the phpMyAdmin interface for the 'blog\_records' table. The table contains 7 rows of data, each representing a blog entry. The columns include 'bid', 'btitle', 'bcategory', 'btag', 'bcontent', 'bstatus', 'bimage', 'DateTime', 'user', 'userid', 'is\_editor\_choice', 'views', and 'Month'.

bid	btitle	bcategory	btag	bcontent	bstatus	bimage	DateTime	user	userid	is_editor_choice	views	Month
167	ENVIRONMENTAL BLOG	Life	Trees #Environment	<span style="color: rgb(55, 71, 79); font-family: ...	Publish	download-1.png	2021-06-30 00:18:39	Kaustubh	85	1	0	May
168	INDEPENDENCE DAY 1		independence 1	<span style="color: rgb(55, 71, 79); font-family: ...	Draft	Independence-Day.jpg	2021-06-30 00:20:36	Kaustubh	85	1	6	May
169	CUP OF TEA	Life	Life	<font color="#323232" size="3" face="arial">i have...	Draft	download22.jpg	2021-06-30 00:23:25	Kaustubh	85	1	1	May
170	CUPCAKES 1	Drama	dramaticLife 1	I had always heard people talk about this blog, bu...	Draft	download.jpg	2021-06-30 00:30:54	Brett	81	0	0	May
171	THE EVERYGIRL	Tragedy	SimpleLiving	I go back and forth between loving and liking this...	Draft	korean-dramas-the-everygirl-1-474x316.jpg	2021-06-30 00:33:29	Brett	81	1	1	May

## 4.Blogs Categories Page:

The screenshot displays the phpMyAdmin interface for a database named 'blogdb'. The 'blog\_categories' table is selected, showing 20 rows of data. The table structure is as follows:

cid	CName	cat_image	uid
141	Classic Literature	[BLOB - 136.6 KiB]	0
142	Historical Fiction	[BLOB - 453.5 KiB]	0
143	Mythology	[BLOB - 138.1 KiB]	0
144	Poetry	[BLOB - 37.3 KiB]	0
145	Science Fiction	[BLOB - 189.3 KiB]	0
146	Biography	[BLOB - 806.1 KiB]	0
147	Society	[BLOB - 6.7 KiB]	0
148	Comedy	[BLOB - 218.0 KiB]	0
149	Life	[BLOB - 182.0 KiB]	0
150	Drama	[BLOB - 683.1 KiB]	0
151	Economics	[BLOB - 429.6 KiB]	0
152	Fantasy Fiction	[BLOB - 276.8 KiB]	0
153	Horror Stories	[BLOB - 49.0 KiB]	0
155	Tragedy	[BLOB - 8.8 KiB]	0
156	Suspence Stories	[BLOB - 6.9 KiB]	0
157	Romance Stories	[BLOB - 8.0 KiB]	0

The interface includes a sidebar with a database tree, a top navigation bar with options like 'Browse', 'Structure', and 'SQL', and a bottom status bar showing system information like '30°C Light rain' and '01:25 AM 07-07-2021'.

## 5. User Access Permission Table:

The screenshot displays the phpMyAdmin interface for the 'blogdb' database, specifically the 'member\_permissions' table. The table contains 72 rows, with the first 24 rows visible. The columns are: pld, mem\_id, options, allow\_insert, allow\_view, allow\_edit, and allow\_delete. The data shows various permissions for different users (mem\_id) across different categories (options).

pld	mem_id	options	allow_insert	allow_view	allow_edit	allow_delete
	82	Blogs	1	1	1	1
	83	Categories	1	0	0	1
	84	manage users	1	1	1	1
	85	manage contacts	1	1	1	1
	86	Blogs	1	1	1	1
	87	Categories	1	1	1	1
	88	manage users	1	1	0	1
	89	manage contacts	1	1	1	1
	90	Blogs	1	1	1	0
	91	Categories	1	1	1	1
	92	manage users	1	0	1	1
	93	manage contacts	0	1	0	0
	94	Blogs	1	1	1	0
	95	Categories	1	1	1	1
	96	manage users	1	0	1	1
	97	manage contacts	0	1	0	0



## 6.Ratings Table:

The screenshot shows the phpMyAdmin interface for a MySQL database. The 'ratings' table is selected, and the data is displayed in a table format. The table has four columns: 'rid', 'rate', 'blog\_id', and 'cur\_user\_id'. The data is sorted by 'rid' in ascending order. The table contains 24 rows of data.

Showing rows 0 - 24 (47 total, Query took 0.0004 seconds)

```
SELECT * FROM `ratings`
```

Options:  Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Number of rows: 25 | Filter rows: Search this table | Sort by key: None

rid	rate	blog_id	cur_user_id
8	4.7	71	64
9	1.7	71	55
10	2.5	71	55
11	2	71	55
12	1.7	71	55
13	4.9	71	55
14	1.6	68	55
15	0.3	68	55
16	4.8	71	55
17	1.4	70	0
18	3.5	71	55
19	2.5	71	65
20	4.5	68	55
21	4.5	68	55
22	4.4	68	55
23	4.4	68	55

## 7. Comments Table:

The screenshot shows the phpMyAdmin interface for the 'blogdb' database, specifically the 'comments' table. The table structure and data are as follows:

comment_id	parent_comment_id	comment_desc	comment_sender_name	date	blogid	bloguserid
30	0	nice post	john	2021-03-20 11:13:24	0	0
31	30	thanks	john	2021-03-20 11:13:36	0	0
32	31	yes john	Kaustubh	2021-03-20 11:17:57	0	0
33	0	hiiii	Kaustubh	2021-03-20 22:40:56	0	0
34	0	dasdsdad	Kaustubh	2021-04-07 00:49:05	0	0
35	0	hi	Kaustubh	2021-04-14 14:16:40	0	0
36	0	hey	Kaustubh	2021-04-14 14:18:20	0	0
37	0	yep	Kaustubh	2021-04-14 14:19:22	0	0
38	0	nbvrvv	Kaustubh	2021-04-14 14:20:42	0	0
39	0	ddasd	Kaustubh	2021-04-14 14:21:49	0	0
40	0	dfsdfdd	Kaustubh	2021-04-14 14:27:36	0	0
41	0	dstdfdis	Kaustubh	2021-04-14 14:28:07	0	0
42	0	nnrvnbv	Kaustubh	2021-04-14 14:29:02	0	0
43	0	nbvrvv	Kaustubh	2021-04-14 14:32:53	0	0
44	0	nbvrvv	Kaustubh	2021-04-14 14:32:54	0	0
45	0	nbvrvv	Kaustubh	2021-04-14 14:32:54	0	0

## 7. Comments Table:

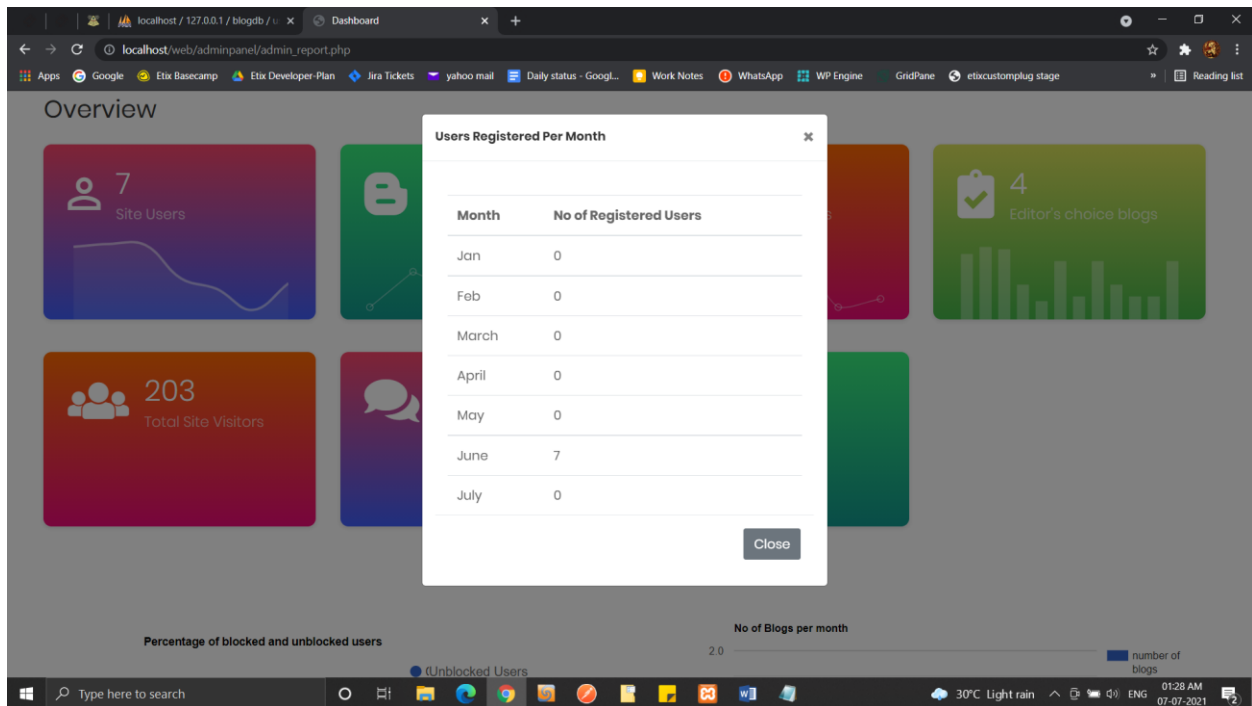
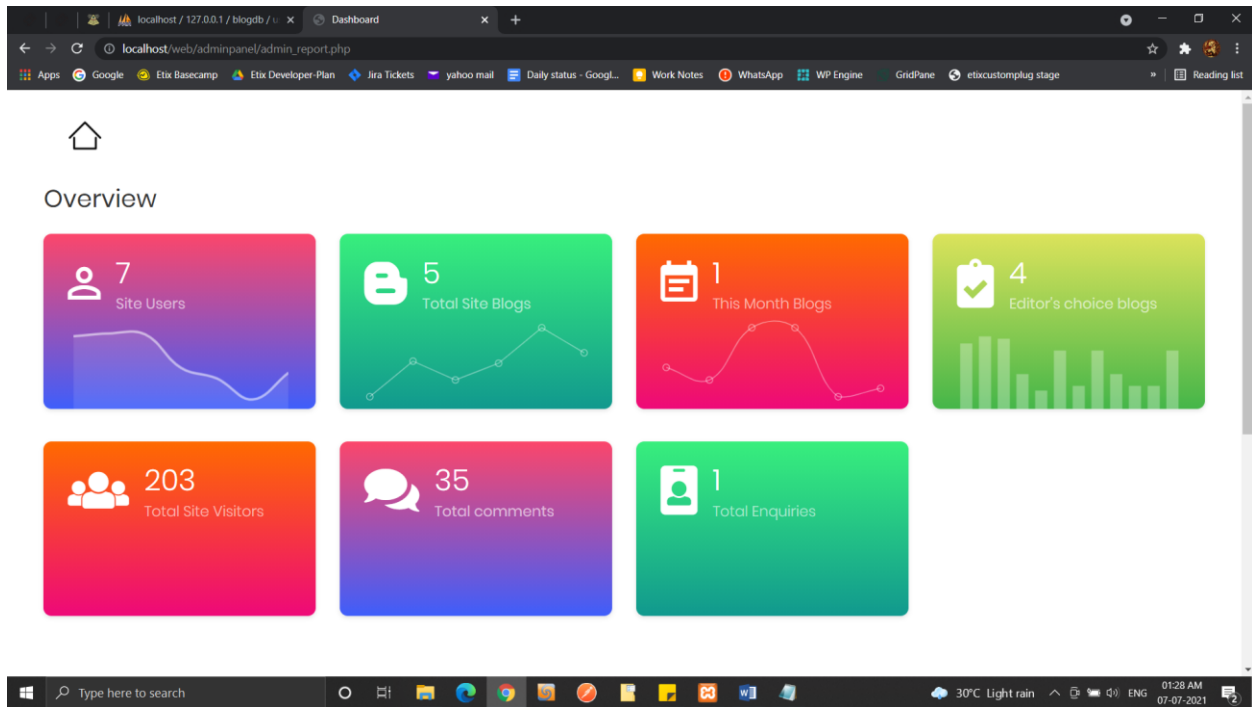
The screenshot displays the phpMyAdmin web interface. The browser address bar shows the URL: localhost/phpmyadmin/sql.php?server=1&db=blogdb&table=user\_counter&pos=0. The interface includes a top navigation bar with tabs for Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, and Triggers. A left sidebar shows a tree view of databases, with 'user\_counter' selected under the 'blogdb' database. The main content area shows a SQL query: `SELECT * FROM `user_counter``. Below the query, a status bar indicates 'Showing rows 0 - 0 (1 total, Query took 0.0005 seconds)'. The table structure is displayed as follows:

id	counts
1	203

Below the table, there are options for 'Query results operations' including Print, Copy to clipboard, Export, Display chart, and Create view. The Windows taskbar at the bottom shows the date and time as 01:28 AM on 07-07-2021.

**ANNEXURE 2**  
**OUTPUT REPORTS WITH DATA**

# Site Statistics Report:



localhost / 127.0.0.1 / blogdb / ... Dashboard

localhost/web/adminpanel/admin\_report.php

Overview

7 Site Users

203 Total Site Visitors

4 Editor's choice blogs

**Blogs Published Per Month**

Month	No of Blogs Per Month
Jan	0
Feb	0
March	0
April	0
May	0
June	5
July	0

Close

Percentage of blocked and unblocked users

No of Blogs per month

2.0

number of blogs

Type here to search

30°C Light rain

01:29 AM 07-07-2021

localhost / 127.0.0.1 / blogdb / ... Dashboard

localhost/web/adminpanel/admin\_report.php

Overview

7 Site Users

203 Total Site Visitors

4 Editor's choice blogs

**Blogs Made as Editor's Choice Per Month**

Month	No of Blogs Per Month
Jan	0
Feb	0
March	0
April	0
May	0
June	4
July	0

Close

Percentage of blocked and unblocked users

No of Blogs per month

2.0

number of blogs

Type here to search

30°C Light rain

01:29 AM 07-07-2021

localhost / 127.0.0.1 / blogdb / ... Dashboard

localhost/web/adminpanel/admin\_report.php

Overview

7 Site Users

203 Total Site Visitors

4 Editor's choice blogs

### Comments Made Per Month

Month	No of Comments Per Month
Jan	0
Feb	0
March	4
April	14
May	0
June	15
July	2

Close

Percentage of blocked and unblocked users

No of Blogs per month

2.0

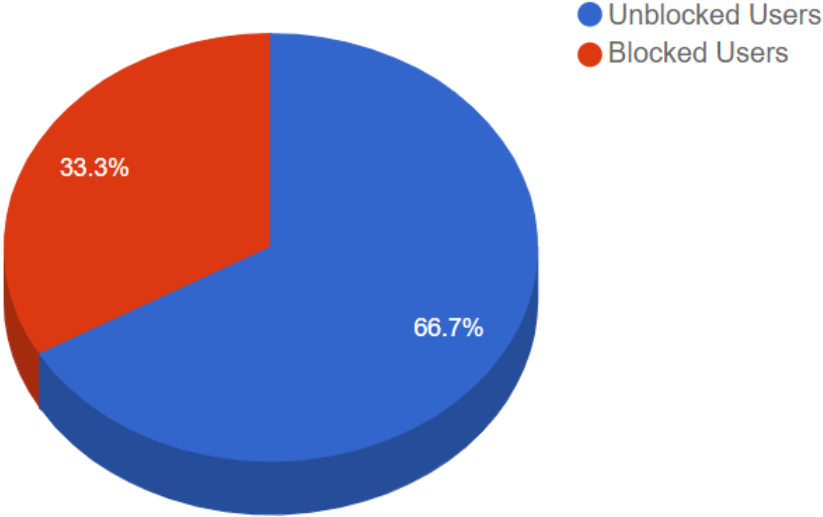
number of blogs

Type here to search

30°C Light rain 01:29 AM 07-07-2021

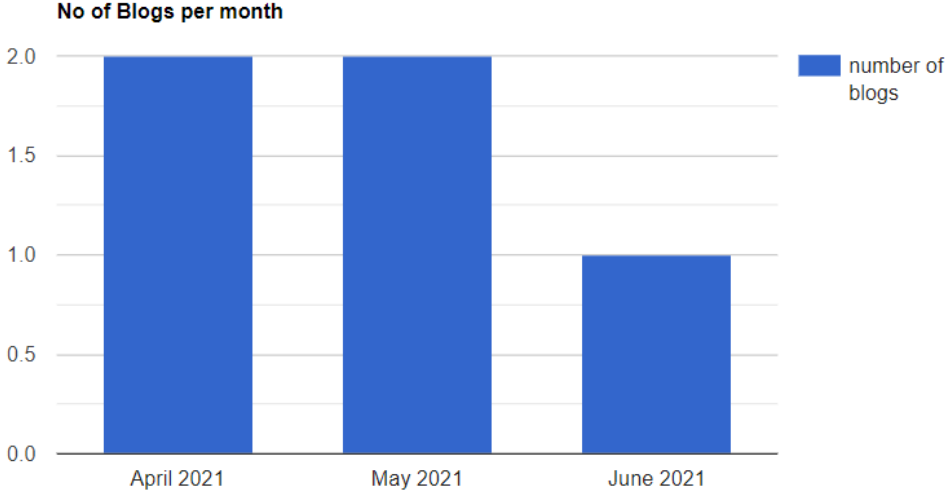
# Percentage of Blocked and Unblocked Users

Percentage of blocked and unblocked users





# No of Blog Publish in per month



# System Users Report:

The screenshot shows a web application interface for managing users. The browser address bar indicates the URL is localhost/web/userpanel/manageusers.php. The page features a sidebar with navigation links and a main content area with a table of users.

**User Profile:**

- Avatar:
- Name: **Welcome Kaustubh**
- Actions: [Sign out](#), [Visit Site](#)

**Navigation Sidebar:**

- Categories: 9
- Blogs: 7
- Draft: 2
- Editor's Choice: 5
- Manage Users: 5
- Contacts: 2

**Main Content Area:**

You are here: Home / Dashboard [+ ADD NEW USER](#)

	<b>Id</b>	<b>Name</b>	<b>Email</b>	<b>Address</b>	<b>City</b>	<b>Mobile no</b>	<b>Delete Record</b>	<b>Update Record</b>
<input type="checkbox"/>	72	David	david@gmail.com	new york strret	Verlin	8822229429	<a href="#">Delete</a>	<a href="#">Update</a>
<input type="checkbox"/>	75	kurt	kurt@gmail.com	1722 Linden St	Chicago	6622993361	<a href="#">Delete</a>	<a href="#">Update</a>
<input type="checkbox"/>	76	Brett	brett@gmail.com	1722 Linden St	New York	5598320423	<a href="#">Delete</a>	<a href="#">Update</a>
<input type="checkbox"/>	78	admin	admin@gmail.com	1722 Linden St	New York	8273837384	<a href="#">Delete</a>	<a href="#">Update</a>

**ANNEXURE 3**  
**SAMPLE PROGRAM CODE**

## 1.CreateBlogs.php

```
<?php
ob_start();
session_start();
//include("../header.php");
include_once '../conn.php';

//$userinfo=$_SESSION['login_user'];
//$userid=$_SESSION['UserId'];

error_reporting(0);

include_once 'conn.php';
if(isset($_POST['upload']))
{

$title=$_POST['btitle'];
$cat=$_POST['category'];
$tag=$_POST['tag'];
$edit=$_POST['area2'];
$stat=$_POST['bstatus'];
$user_id=$_POST['authorid'];
$editblog=$_POST['editorblog'];
if(isset($_POST['editorblog']))
```

```

{
    $editblog=1;
}
else{
    $editblog=0;
}

echo $editblog;

$user=$_SESSION['admin_user'];

$file=addslashes(file_get_contents($_FILES["imagefile"]["tmp_name"]));

    $sql = "INSERT into `blog_records`
(bttitle,bcategory,btag,bcontent,bstatus,is_editor_choice,bimage,user,user_id)
VALUES ('$title','$cat','$tag','$edit','$stat','$editblog','$file','$user','$user_id)";

if (mysqli_query($conn, $sql)) {
    header("location:../Home.php");
    echo "blog record created successfully !";
} else {
    echo "Error: " . $sql . "
" . mysqli_error($conn);
}

mysqli_close($conn);
}

```

?>

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<!doctype html>
```

```
<html lang="en">
```

```
<head>
```

```
<!-- Required meta tags -->
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<!-- Bootstrap CSS -->
```

```
<link rel="stylesheet"
```

```
href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
```

```
integrity="sha384-
```

```
Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
```

```
crossorigin="anonymous">
```

```
<title>Hello, world!</title>
```

```
</head>
```

```
<!-- Optional JavaScript -->
```

```
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
```

```
<script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa4849bIE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>
```

```
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
```

```
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script>
```

```
<script src="https://cdn.tiny.cloud/1/no-api-key/tinymce/5/tinymce.min.js" referrerpolicy="origin"></script>
```

```
<script>
```

```
  tinymce.init({  
    selector: 'textarea#editor',  
    menubar: false  
  });
```

```
</script>
```

```
</body>
```

```
</html>
```

```
<body style="background-image: url('../images/back6.jpg');background-repeat: no-repeat;background-size: 1580px 1024px;">
```

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

```
</head>
```

```
<body>
```

```

<?php

include_once 'conn.php';

$sql = 'SELECT * FROM blog_categories';

$result = mysqli_query($conn, $sql);

?>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
<script>

var loadFile = function(event) {

var reader = new FileReader();

reader.onload = function(){

var output = document.getElementById('output');

output.src = reader.result;

};

reader.readAsDataURL(event.target.files[0]);

};

</script>

<script type="text/javascript" src="http://js.nicedit.com/nicEdit-latest.js"></script> <script
type="text/javascript">

//

bkLib.onDomLoaded(function() { nicEditors.allTextAreas() });

//]]&gt;

&lt;/script&gt;
</pre>
</div>
<div data-bbox="850 917 890 935" data-label="Page-Footer">
<p>128</p>
</div>
```



```

<form action="" method="post" name="blog_form" style="margin-top: 5%;"
enctype="multipart/form-data" >

  <div class="col-8 container">

    <h1>Create your blog</h1>

    <hr>

    <input type="hidden" placeholder="Your blog title..." name="authorid" id="authorid"
value="<?php echo $userid ?>" >

    <label for="title"><b>Enter Title:</b></label>

    <input type="text" placeholder="Your blog title..." name="btitle" id="btitle" >

<br><br>

    <label for="category"><b>Choose Category:</b></label>

    <select class="col-6 btn btn-light dropdown-toggle3" aria-label="Default select example"
name="category" id="category">

<option selected>None</option>

    <?php if (mysqli_num_rows($result) > 0) {
        while($row = mysqli_fetch_assoc($result)) {?>
<option value="<?php echo $row["CName"];?>"><?php echo $row["CName"];?></option>
    <?php
        }
    } else {
        echo "0 results";
    }
    mysqli_close($conn);

```

?>

</select>

<br><br>

<label for="tag"><b>Enter Tag:</b></label>

<input type="text" placeholder="Your blog tags..." name="tag" id="tag" >

<!--<div class="container mt-4 mb-4 ml-0 mr-0">

<label for="tag"><b>Write Here:</b></label>

<div class="row ">

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

<div class="form-group">

<textarea id="editor" name="editor"></textarea>

</div>

<!--<button type="submit" class="btn btn-primary">Submit</button>-

</div>

</div>

</div>-->

<br>

<label for="tag"><b>Write Here:</b></label>

<textarea name="area2" id="area2" style="width: 100%;">

Some Initial Content was in this textarea

```

</textarea>

<br>

<label class="form-label" for="stat"><b>Choose Status</b></label>

<select class="col-6 btn btn-light dropdown-toggle" aria-label="Default select example"
name="bstatus" id="bstatus">

  <option selected>Publish</option>

  <option value="Draft">Draft</option>

</select><br><br>

<label class="form-label" for="customFile">Make it Editor's Choice</label>

<input type="checkbox" name="editorblog" id="editorblog" value="1" />

<br>

<label class="form-label" for="customFile"><b>Choose Image</b></label>

<input type="file" class="form-control col-6 " accept="image/*" name="imagefile"
id="imagefile" onchange="loadFile(event)"/>

<img id="output" alt="your image" height="200" width="200" />

<br>

<br>

<div class="col-12 ">

  <input type="submit" class="registerbtn col-12 btn btn-success" value="Upload Your Blog"
name="upload">

  </div>

</form>

</body>

</html>

```

## 2.Home.php(View Blogs)

```
<?php
error_reporting(0);

include_once 'conn.php';

session_start();

$userdata=$_SESSION['login_user'];

$userid=$_SESSION['UserId'];

$usersql = "SELECT * FROM membership_users where Username='$userdata'";

$userresult = mysqli_query($conn, $usersql);

if (mysqli_num_rows($userresult) > 0) {
    while($userrow = mysqli_fetch_assoc($userresult)) {?>
<tr style="visibility: hidden;">
    <td ><?php $userrow["ID"];
    $userid=$userrow["ID"];
    ?></td>
<?php $_SESSION['UserId'] = $userid;?>
</tr>
<?php
    }
    } else {
        echo "0 results";
    }
    //mysqli_close($conn);

?>
```

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <!-- SEO Meta Tags -->

  <meta name="description" content="Create a stylish landing page for your business startup
and get leads for the offered services with this HTML landing page template.">

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

  <!-- OG Meta Tags to improve the way the post looks when you share the page on LinkedIn,
Facebook, Google+ -->

  <meta property="og:site_name" content="" /> <!-- website name -->

  <meta property="og:site" content="" /> <!-- website link -->

  <meta property="og:title" content="" /> <!-- title shown in the actual shared post -->

  <meta property="og:description" content="" /> <!-- description shown in the actual shared
post -->

  <meta property="og:image" content="" /> <!-- image link, make sure it's jpg -->

  <meta property="og:url" content="" /> <!-- where do you want your post to link to -->

  <meta property="og:type" content="article" />

  <!-- Website Title -->

  <title>Bloggers</title>

  <!-- Styles -->
```

```
<link
href="https://fonts.googleapis.com/css?family=Raleway:400,400i,600,700,700i&subset=latin-ext" rel="stylesheet">
```

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

```
<link href="css/fontawesome-all.css" rel="stylesheet">
```

```
<link href="css/swiper.css" rel="stylesheet">
```

```
<link href="css/magnific-popup.css" rel="stylesheet">
```

```
<link href="css/styles.css" rel="stylesheet">
```

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/bootstrap.min.css">
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
```

```
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.6/umd/popper.min.js"></script>
```

```
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js"></script>
```

```
<!-- Favicon -->
```

```
<link rel="icon" href="images/favicon.png">
```

```
</head>
```

```
<body data-spy="scroll" data-target=".fixed-top">
```

```
<!-- Preloader -->
```

```
<div class="spinner-wrapper">
```

```
<div class="spinner">
```

```
<div class="bounce1"></div>
```

```
<div class="bounce2"></div>
```

```
<div class="bounce3"></div>
```

```

        </div>

</div>

<!-- end of preloader -->

<?php

include('header.php');?>

<!-- Header -->

<header id="header" class="header">

    <div class="header-content">

        <div class="container">

            <div class="row">

                <div class="col-lg-6">

                    <div class="text-container">

                        <h1><span class="turquoise"><?php echo "Welcome
".$_SESSION['login_user']; ?></span> </h1>

                        <p class="p-large">Express your thoughts.... </p>

                        <a class="btn-solid-lg page-scroll" href="logout.php">Logout</a>

                        <a class="btn-solid-lg page-scroll" href="userprofile.php">Profile</a>

                        <a class="btn-solid-lg page-scroll"
href="userpanel/index2.php">Dashboard</a>

                    </div> <!-- end of text-container -->

                </div> <!-- end of col -->

                <div class="col-lg-6">

                    <div class="image-container">

```

```
        
```

```
    </div> <!-- end of image-container -->
```

```
</div> <!-- end of col -->
```

```
</div> <!-- end of row -->
```

```
</div> <!-- end of container -->
```

```
</div> <!-- end of header-content -->
```

```
</header> <!-- end of header -->
```

```
<!-- end of header -->
```

```
<!-- <h3 style="position: absolute;left:73%;top:125%;">Social Links
```

```
<div id="buttons">
```

```
<div class="facebook button">
```

```
<i class="icon">
```

```
<i class="fa fa-facebook"></i>
```

```
</i>
```

```
<div class="slide">
```

```
<p>
```

```
Facebook
```

```
</p>
```

```
</div>
```

```
<iframe
```

```
src="https://www.facebook.com/plugins/like.php?href=https%3A%2F%2Fwww.facebook.com%2Fenablingwp%2F&width=62&layout=button&action=like&size=large&show_faces=false&share=false&height=65&appId" width="62" height="65" style="border:none;overflow:hidden" scrolling="no" frameborder="0" allowTransparency="true"></iframe>
```

```
</div>
```



```
<div class="google button">
  <i class="icon">
    <i class="fa fa-google-plus"></i>
  </i>
```

```
<div class="slide">
  <p>
    Google+
  </p>
</div>
```

<!-- Place this tag where you want the +1 button to render. -

```
<div class="g-follow" data-annotation="bubble" data-
href="https://plus.google.com/+Enablingwp/" data-rel="publisher"></div>
```

<!-- Place this tag after the last +1 button tag. -

```
<script type="text/javascript">
(function() {
  var po = document.createElement('script'); po.type = 'text/javascript'; po.async = true;
  po.src = 'https://apis.google.com/js/platform.js';
  var s = document.getElementsByTagName('script')[0]; s.parentNode.insertBefore(po, s);
})();
</script>
</div>
```

```
<div class="youtube button">
  <i class="icon">
    <i class="fa fa-youtube"></i>
```

```
</i>
```

```
<div class="slide">
```

```
<p>
```

```
  YouTube
```

```
</p>
```

```
</div>
```

```
<div class="g-ytsubscribe" data-channel="techpingo" data-layout="default" data-count="default" data-onytevent="onYtEvent"></div>
```

```
<script src="https://apis.google.com/js/platform.js"></script>
```

```
<script>
```

```
function onYtEvent(payload) {
```

```
  if (payload.eventType == 'subscribe') {
```

```
    // Add code to handle subscribe event.
```

```
  } else if (payload.eventType == 'unsubscribe') {
```

```
    // Add code to handle unsubscribe event.
```

```
  }
```

```
  if (window.console) { // for debugging only
```

```
    window.console.log('YT event: ', payload);
```

```
  }
```

```
}
```

```
</script>
```

```
</div>
```

```
<div class="twitter button">
```

```

    <i class="icon">
      <i class="fa fa-twitter"></i>
    </i>
    <div class="slide">
      <p>
        Twitter
      </p>
    </div>

    <a href="https://twitter.com/iamrahuldc" class="twitter-follow-button" data-show-
count="false" data-via="mariuCSS">
      Tweet
    </a>
    <script>
      !function(d,s,id){
        var js,fjs=d.getElementsByTagName(s)[0],p=/^http:/.test(d.location)?'http':'https';
        if(!d.getElementById(id)){
          js=d.createElement(s);
          js.id=id;

          js.src=p+'://platform.twitter.com/widgets.js';fjs.parentNode.insertBefore(js,fjs); }(document,
'script', 'twitter-wjs');
    </script>
  </div>

  <center></h3>

```

-->

```
<h5 style="position: absolute;left:74%;top:110%;">categories<br>
```

```
<?php
```

```
include_once 'conn.php';
```

```
$sql3 = 'SELECT * FROM blog_categories';
```

```
$sql_cat_count = 'SELECT count(*) as total1 from blog_categories ';
```

```
$result3 = mysqli_query($conn, $sql3);
```

```
$result_cat_count = mysqli_query($conn, $sql_cat_count);
```

```
$dat=mysqli_fetch_assoc($result_cat_count);
```

```
if ((mysqli_num_rows($result3) > 0) && (mysqli_num_rows($result_cat_count) > 0)) {
```

```
while(($row3 = mysqli_fetch_assoc($result3)) && ($row_cat_count = $dat)) {?>
```

```
    <div class="btn-solid-lg page-scroll" style=" text-align: center; font-size: 16px  
!important;width: 265px; margin-bottom: 5px;">
```

```
        <span class="inbox-num"><!--<?php echo $dat['total1']; ?>--></span>
```

```
        <div class="cathov"> <?php echo " <a  
href='category_blog.php?CName=$row3[CName] ' style='color:white;'>"?> <?php echo  
$row3["CName"];?> </a></div>
```

```
    </div>
```

```
<?php
```

```

    }
} else {#B31217;
    echo "0 results";
}

?>
<br><br>
<b>Popular Posts</b>

<?php
    include_once 'conn.php';
    $sql4 = "SELECT * FROM blog_records where views>=10 ";
    $result4 = mysqli_query($conn, $sql4);

    if (mysqli_num_rows($result4) > 0) {
        while($row4 = mysqli_fetch_assoc($result4)) {?>

            <div class="card" style="max-width: 300px !important;">
<div class="card-body">
    <?php echo $row4["btitle"];?> <br>
    <?php echo " <a href='single_blog.php?id=$row4[bid] ' >Read More</a>";?>
</div>
</div>

<?php

```

```
    }
    } else {#B31217;
        echo "0 results";
    }
?> </h3>
```

```
<form action="search.php" method="get">
    <div class="col-lg-3" style="margin-left: 1100px;">
        <div class="input-group">
            <input type="text" class="form-control" name="search" id="search" placeholder="Search
this blog">
            <div class="input-group-append">
                <input type="submit" value="search" />
                </button>
            </div>
        </div>
    </div>
</form>
```

```
<?php
function custom_echo($x, $length)
{
    if(strlen($x)<=$length)
    {
        echo $x;
```

```

}
else
{
    $y=substr($x,0,$length) . '...';
    echo $y;
}
}

?>

<?php
    $sql1 = 'SELECT * FROM blog_records where bstatus="Publish" ORDER BY bid DESC limit
5';

    $result1 = mysqli_query($conn, $sql1);

    if (mysqli_num_rows($result1) > 0) {
        while($row1 = mysqli_fetch_assoc($result1)) {?>

            <div class="card col-8 mb-3 box ml-5 " style="margin-top: -40px;">

                <!-- <a href="single.php?post=<?php echo $row1["bid"];?>" class="abc">-->

                <?php $blogs_id=$row1["bid"];?>

            <div class="row no-gutters">

                <div class="col-lg-4">

                    <td >

```

```
<?php echo '';?>
```

```
</td>
```

```
</div>
```

```
<div class="col-md-8">
```

```
<div class="card-body">
```

```
<h3 class="card-title"><?php echo $row1["btitle"];?></h3>
```

```
<p class="card-text"><small class="text-muted" style="text-align: left
!important;">
```

```
<?php
```

```
if($row1['is_editor_choice']==1)
```

```
{?>
```

```

```

```
<?php }?>
```

```
<?php custom_echo($row1["bcontent"], 400); ?>
```

```
<div class="card-text" style="font-size: 16px; font-weight: 800;"><svg
xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-
tags-fill" viewBox="0 0 16 16">
```

```
<path d="M2 2a1 1 0 0 1 1-1h4. 0 0 0 0 3z"/>
```

```
<path d="M1.293 7.70717 7a1 1 0 0 0 1.414 0l.043-.043-7.457-7.457z"/>
```

```
</svg>&nbsp;&nbsp;<?php echo $row1["btag"];?></div>
```



```
<div class="card-text" style="font-size: 13px; font-weight: 600;"><svg
xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor" class="bi bi-
calendar" viewBox="0 0 16 16">
```

```
<path d="M3.5 0a.5. 1 0 0 0 1 1h12a1 1 0 0 0 1-1V4H1z"/>
```

```
</svg> <b> &nbsp;
```

```
<?php
```

```
$date=$row1["DateTime"];
```

```
echo date('F d, Y',strtotime($date)); ?>
```

```
<!--<?php echo $row1["DateTime"];?--></b></div>
```

```
<b><div class="card-text" style="color:red !important;font-size: 13px; font-weight:
600;"><svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor"
class="bi bi-person" viewBox="0 0 16 16">
```

```
<path d="M8 8a3 3 0 1 332-.678.678-.83 1.418-.832 1.664h10z"/>
```

```
</svg></a>&nbsp; <?php echo " <a href='user_account.php?uid=$row1[userid] ' >"?><?php
echo $row1["user"];?></a></div></b>
```

```
<b><div class="card-text" style="font-size: 13px; font-weight: 600;">
```

```
<svg xmlns="http://www.w3.org/2000/svg" width="16" height="16" fill="currentColor"
class="bi bi-grid" viewBox="0 0 16 16">
```

```
<path d="M1 2.5A1. 1.5 0 0 1 9 13.5v-3zm1.5-.5a.5.5 0 0 0-.5.5v3a.5.5 0 0 0 .5.5h3a.5.5 0 0 0
.5-.5v-3a.5.5 0 0 0-.5-.5h-3z"/>
```

```
</svg>&nbsp; <?php echo " <a href='category_blog.php?CName=$row1[bcategory] ' >"?>
<?php echo $row1["bcategory"];?></a></div></b>
```

```
<div class="btn btn-link">
<?php echo " <a href='single_blog.php?id=$row1[bid] ' >Read More</a>";?>
```

```
<?php
```

```
    error_reporting(E_ALL);

    $ratesql= mysqli_query($conn,"SELECT avg(rate) as myrate FROM ratings
where blog_id='$blogs_id'");
```

```
    $raterow = mysqli_fetch_assoc($ratesql);

    $aa=round($raterow['myrate'], 2);

    echo " <p style='position:absolute;color:orange;left:83%
!important;top:70%;font-weight:700; '> ratings : " . $aa . ' / 5' ?>
```

```
    <div class="rateyo" id= "rating"
    data-rateyo-rating="<?php echo $aa . ' / 5';?>"
    >
    </div>
```

```
<?php ?>
```

```
    <div class="share-button">

    <div class="share-button__back">

    <a class="share__link" href="www.twitter.com" title="twitter">

    <svg class="share__icon share__icon--twitter" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px"
y="0px" width="49.652px" height="49.652px" viewBox="0 0 49.652 49.652" style="enable-
background:new 0 0 49.652 49.652;" xml:space="preserve">
```

```

<g>
  <g>
    <path
d="M24.826,0C11.137,0,0,11.137,0,24.826c0,13.688,11.137,24.826,24.826,24.826c13.688,0,24
5.746,16.255-16.259,16.,2.261-0.444,3.282-0.899
C37.987,17.334,37.018,18.341,35.901,19.144z"></path>
  </g>
</g>
</svg>
</a>
<a class="share__link " href="www.facebook.com" title="facebook">
  <svg class="share__icon share__icon--facebook" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px"
y="0px" width="49.652px" height="49.652px" viewBox="0 0 49.652 49.652" style="enable-
background:new 0 0 49.652 49.652;" xml:space="preserve">
  <g>
    <g>
      <path d="M24.826,.939c0,0-2.695,0-3.219,0c-0.524,0-1.269,0.262-
1.269,1.386v2.99h4.56L31,25.7z  "></path>
    </g>
  </g>
</svg>
</a>
<a class="share__link" href="www.googleplus.com" title="google plus">
  <svg class="share__icon share__icon--google"
version="1.1"xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" width="49.652px"
height="49.652px" viewBox="0 0 49.652 49.652" style="enable-background:new 0 0 49.652
49.652;" xml:space="preserve">

```

```
<g>
<g>
<g>
<path d="M21.5,28.94c-0.161-0.107-0.326-0.223-0.499-0.34c-0.503-0.154-1.037-0.234-1.
735,21.5,28.94z"></path>
<path d="M19.719,22.352c0.002,0,0.002,0,0.002,0c0.601,0,1.108-0.237,1.501-0.687c0.616-
0. 0.705,3.012 352z"></path>
<path
d="M24.826,0C11.137,0,0,11.137,0,24.826c0,13.688,11.137,24.826,24.826,24.826c13.688,0,24.
826-11.138,24.826-24.826 C49.652,11.137,38.516,0,24.826,0z
c0.319,0,0.58,0.26,0.58,0.58V24.253z"></path>
</g>
</g>
</g>
</svg>
</a>
<a class="share__link" href="www.dribbble.com" title="dribbble">
<svg class="share__icon share__icon--dribbble" version="1.1"
xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px"
y="0px" viewBox="0 0 96 96" style="enable-background:new 0 0 96 96;"
xml:space="preserve">
<g>
<path d="M49.297,47.914c0.172-0.054,0.343-0.106,0.515-0.157c-0.414-0.928-0.8731.612-
8.381-4.253-11.694C62.53,36.825,59.238,40.612,52.611,43.378z"></path>
</g>
</svg>
</a>
</div>
```

```

<div class="share-button__front">
  <p class="share-button__text"> </p>
</div>
</div>
      </div>
      </div>
      </div>
      </div>
      </div>
<br><br>
      <?php
      }
      } else {
        echo "0 results";
      }
      //mysqli_close($conn);

?>

<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>
<!------- Include the above in your HEAD tag ----->

```

```
<!DOCTYPE html><html lang='en' class=''>

<head><script src='//production-assets.codepen.io/assets/editor/live/console_runner-
079c09a0e3b9ff743e39ee2d5637b9216b3545af0de366d4b9aad9dc87e26bfd.js'></script><script
src='//production-assets.codepen.io/assets/editor/live/events_runner-
73716630c22bbc8cff4bd0f07b135f00a0bdc5d14629260c3ec49e5606f98fdd.js'></script><script
src='//production-assets.codepen.io/assets/editor/live/css_live_reload_init-
2c0dc5167d60a5af3ee189d570b1835129687ea2a61bee3513dee3a50c115a77.js'></script><meta
charset='UTF-8'><meta name="robots" content="noindex"><link rel="shortcut icon"
type="image/x-icon" href="//production-assets.codepen.io/assets/favicon/favicon-
8ea04875e70c4b0bb41da869e81236e54394d63638a1ef12fa558a4a835f1164.ico" /><link
rel="mask-icon" type="" href="//production-assets.codepen.io/assets/favicon/logo-pin-
f2d2b6d2c61838f7e76325261b7195c27224080bc099486ddd6dccb469b8e8e6.svg" color="#111"
/><link rel="canonical"
href="https://codepen.io/Eyjjafjallajokull/pen/KddvMe?depth=everything&order=popularity&pa
ge=34&q=thumbnail&show_forks=false" />
```

```
<link rel='stylesheet prefetch'
href='//cdnjs.cloudflare.com/ajax/libs/flickity/1.1.0/flickity.min.css'><link rel='stylesheet
prefetch'
href='//fonts.googleapis.com/css?family=Roboto:400,400italic,500,500italic,300italic,300'>
```

```
<style class="cp-pen-styles">.flickity-prev-next-button.previous {
  left: -30px;
}
.flickity-prev-next-button.next {
  right: -30px;
}
.flickity-prev-next-button .arrow {
  fill: #F5F7FA;
}
.flickity-prev-next-button {
  background-color: #434A54;
```

```

width: 38px;
height: 38px;
}
.flickity-prev-next-button:hover {
background-color: #656D78;
}
</style></head><body>
<html lang="en">

<head>
<meta charset="UTF-8" />
<title>Infinite Flickity Slider Test</title>
</head>
<body>
<div class="site-content cf">

<h2 class="course-heading" style="font-family: Roboto;font-size: 20px; color:
black;">Editor's Choice</h2>

<div class="courses cf" id="showcase">

<?php
$editsql = 'SELECT * FROM blog_records where is_editor_choice=1';
Sedit_result = mysqli_query($conn, $editsql);

if (mysqli_num_rows($Sedit_result) > 0) {
while($Sedit_row = mysqli_fetch_assoc($Sedit_result)) {?>
<div class="course-item slide">

```

```

<div class="course-summary card">
    <div class="course-thumbnail">
        <?php echo '';?>
    </div>
    <div class="course-info">
        <h1 class="course-topic webdesign"><?php echo $edit_row["btitle"];?></h1>
        <h3 class="course-title"><?php echo $edit_row["bcategory"];?></h3>
        <h4 class="course-title"><?php custom_echo($edit_row["bcontent"], 100); ?></h4>
        <time class="course-date"><?php echo $edit_row["btag"];?></time><br>
        <time class="course-date"> <?php echo $edit_row["user"];?><?php echo " <a
href='user_account.php?uid=$row1[userid] ' >See</a>";?></time><br>
        <span class="course-duration"><?php echo $edit_row["DateTime"];?></span><br>
        <div class="btn btn-link">
            <?php echo " <a href='single_blog.php?id=$edit_row[bid] ' >Read
More</a>";?>
        </div>
    </div>
</div>
</div>
</div>
</div>
    <?php
    }
} else {
    echo "0 results";
}

```



```
}
```

```
//mysqli_close($conn);
```

```
?>
```