_____

SYSTEM DESIGN

DOCUMENT

for
Educational Blog Website

Prepared By-

GROUP- 15

NAME	ROLL NUMBER	BATCH
AKHILESH NAGA	B200800CS	A
ADITYA KINKAR	B200848CS	A
FAROOQUI NABEEL AHMED ASHFAK AHMED	B200804CS	A
PRACHI MISHRA	B200788CS	A

Instructor: **Dr. Prabhu M**

Course   : **CS3002D Database Management System**

Date   : **30 October 2022**

Table of Contents

1. INTRODUCTION

This Blog system is designed to solve the large communication barrier between faculty and students. We lack a platform where all students and faculties may come and share the knowledge and content related to their field, and also get to know about all fields they are interested in. Blog has the capacity to simultaneously strike a professional and a warm relationship between the creator and the reader. We have decided to implement a blogging mechanism for NITC. This Educational Blog system is a web application intended to convey students of NITC regarding all the information teachers want to share related to their subject out of their books course .We usually find that teachers want to share some videos, research papers, news, links, some articles and many more stuffs which are not to be shared in our eduserver portal. Hence this blog website will give them functionality to share this stuffs also students will be allowed to comment and provide their views on the post. Only teachers will be having access to administration of site so they can control the comments and activities on the blog post. Additionally, it is seen that college event organizing groups have to use different social media to promote their events. So we will be giving a special blog posting facilities for such groups to promote their college events they will be organizing.

The design of database is based on the information posted by the blogger, and served to reader in reverse chronological order. All bloggers data is also separated on basis of blog category. Special permission are grated to bloggers who have ability to view states related to post, allow and disallow comments to be shown etc. The data is stored in SQL. Access to data will be through a web interface running on web server and using PHP.

The driving philosophy behind the database design was to have an efficient, normalized database that would be easy to maintain and expand, as well as allow easy data entry and access. Easy deletion and update of data is also taken under consideration.

2. PURPOSE

The Document is a Database Design Document for the "Educational Blog System" web app for Institutes. The Database Design Document maps the logical data model to the target database management system with consideration to the system's performance requirements. This document can be used for:

- Knowing the basis for application's database design
- Provide expected data volumes, functional/non-functional usages of tables.
- Ensure database transaction meets or exceed performance requirements.

The Database Design converts logical or conceptual data constructs to physical storage constructs (e.g., tables, files) of the target Database Management System (DBMS).

2.1 DOCUMENT OBJECTIVES:

The Database Design Document has the following objectives:

- To serve as basis for implementing database and related software units.
- It provides basic idea of design to acquirer and help in designing and developing software required for the application.
- Describing design of database that can be accessed by users or computer developers.

2.2 INTENDED AUDIENCE:

This Document is intended for several audiences, including system designer and developers.

1. The database designer can use this for designing database according to needs of project.

2. The developer would be benefited by knowing the requirements of client .

3. Architects, whose overall architecture design must meet the requirements specified in this document.

4. Quality Assurance personnel, whose test cases must validate the requirements specified in this document.

2.3 DOCUMENT CONVENTIONS:

| USER | Someone who interacts with the web application. |
Admin	Who manages and organizes the blog website.
Blog	A web application consisting of posts typically displayed in reverse chronological
DBA	Database Administration.

3. SOFTWARE AND HARDWARE SPECIFICATIONS

3.1 Software Selection :

Following reviews of the data that would be entered into the database, and the requirements of retrieving the data several criteria were identified that need to be met by the database software. The minimum requirements for the software were:

1. There must be restrictions on data values entered in the columns of relation. By restricting type and range of data values entered we can control data we get and hence reduce error.
2. Multiple user access to the database simultaneously must be allowed. Since multiple bloggers are allowed to post simultaneously, many users will be posting at same time. Further many users will be retrieving data from the database at same time.
3. Allow data entry from the internet. Most of the data will be entered into the forms from the internet.
4. Allow triggers on the data tables. Triggers will allow predetermined actions to be taken when information is entered, edited, or deleted from a data table.
5. Should support the relational database model, and at least basic version of SQL.
6. Allow running of stored SQL scripts. There are many processes that can be automated with stored scripts, to facility management, updates, editing, and querying of the database. This is especially important if users are accessing the data from the internet. Being able to call and run a stored script is far easier and more efficient than trying to code all the information into a web form.
7. Allow joins within a query. The joins are different ways of selecting items from one or more tables, in either a query or a view.

3.2 HARDWARE ARCHITECTURE :
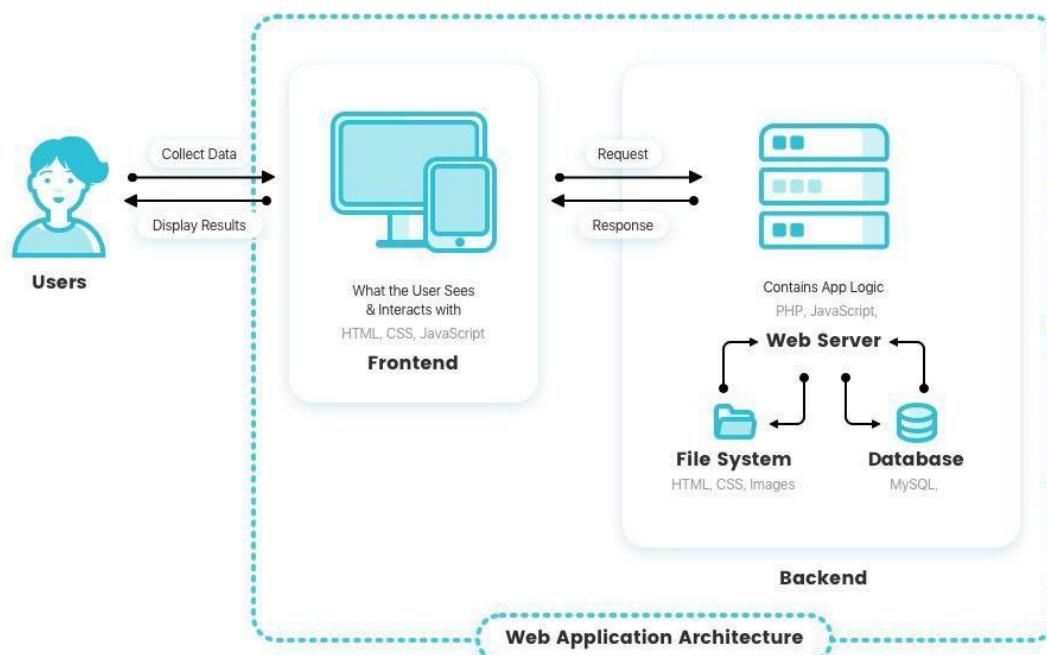
3.3 WEBSITE ARCHITECTURE :
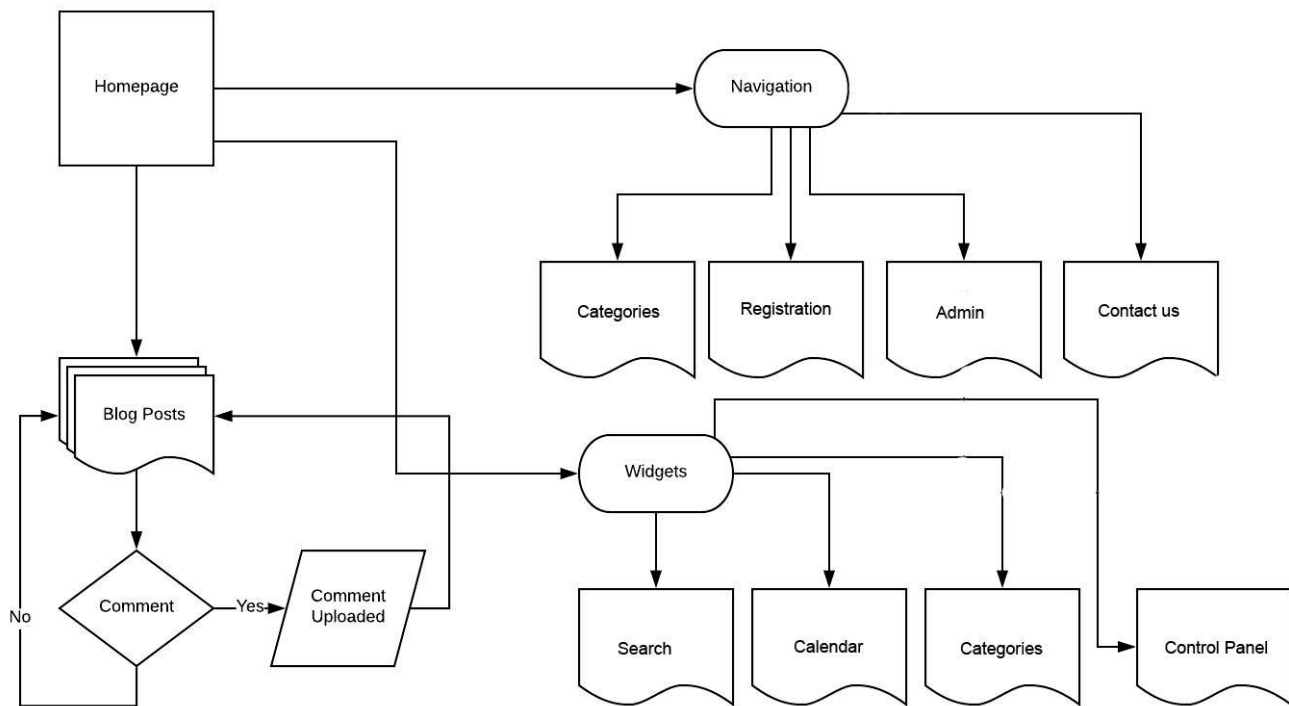
3.4 HARDWARE INTERFACE :

● Since the application runs over the internet, the system must always be connected to the internet.
● Backup storage for retrieval of data in case of unexpected failures.
● Laptop or mobiles to access the site are required.

4. NON-FUNCTIONAL REQUIREMENTS

4.1 PERFORMANCE REQUIREMENTS :

- A server is required to run this web application.
- The request response time will depend on network speed and latency of the server.
- So, a SQL server capable of handling approx 1000 concurrent requests is required.
- The site load time depends on speed of internet connection and proximity from the server
- The load time may even depend on type of hardware on which site is running

4.2 SAFETY AND SECURITY REQUIREMENTS :

- Unique Login Id and password (may or may not be unique) for every user.
- Data backup to be done periodically.
- Admin panel having feature of posting and manipulating the blogs is accessible by admins only.
- Only teachers and selected student representing a college club is selected as admin of blog website.
- User's id and password are secured and no visible to even admins.
- Users have option to save their login credentials.
- All Users have unique id and passwords.
- Password will be hidden while typing.
- Sign out and change password features are given to users

4.3 SOFTWARE QUALITY ATTRIBUTES :

1. Reliability

- New posts in database are reflected without any delay.
- Any registration by the user is reflected in database almost instantly.
- Posts are shown in reverse chronological order, hence newest posts will be shown first.
- Application is made to protect inside information from users, only few selected people have access to handle all features of blog website.

2. Usability

- In case of invalid situations an error message pops up on the screen, directing the user so as to resolve the issue.
- This application works in every browser.
- This application is responsive, so works great with every size of screen.

3. Correctness

● To provide only reliable content in the post only few people are allowed to handle admin panel.
● Admin have full control to manipulate comments and posts. So in case they don't like the comments on their posts they can delete it.

4. Maintainability

● Backup of database will be kept.
● SQL server would be having database maintenance routine scheduled monthly.
● Periodic testing and debugging will be done based on users and admins feedbacks.
● In case of any error, re-initialization of program will be done.

5. DESIGN DECISIONS

5.1 KEY FACTORS INFLUENCING DESIGN :

Some key features that will influence our database design are-
1. Only registered user will be able to post.
2. User who post get a feature of control panel where he will be given states related to posts and have special features to maintain comments and other activities related to post.
3. Blog posts are separated on basis of the categories.
4. Blogs are displayed in reverse chronological order.
5. Login and registration feature are different for people who posts and for one who just reads blogs.

5.2 BEHAVIOR :

Since blogs are divided in categories, view by categories is provided. To implement this we will be using view feature of SQL. Also search by name is also included, for this user have to include blog post name at search bar and this runs the sql query at backend of server. Also it will not be necessary to write full blog post name just some words will be sufficient to search. For this feature to work string comparison will be used and searching is done on the database.

For all queries to be performed related to database, user won't have to write any query. Some proper user interface will be provided where they can run the related query and retrieve result in better looking interface.

5.3 DBMS PLATFORM :

1. For database we will use - phpMyAdmin version 10.4.25.
   All version more than this will work fine. However if version is older, then it may give some different output than expected.
2. Server version:
   10.4.25
3. PHP Version: 8.1.10
   Some more updates versions of this are available now. All will work fine with this.

5.4 BACKUP AND MAINTENANCE :

- Backup of database will be kept.
- Periodic testing and debugging will be done based on users and admins feedbacks.
- SQL server would be having database maintenance routine scheduled monthly.
- In case of any error, re-initialization of program will be done.

5.5 DESIGN AND IMPLEMENTATION CONSTRAINTS :

1. SQL server must be running all day long.

2. Only specific users will have ability to post in blog site.

3. Admin have control on comments and other features of posts. However accessing database for them also is not be allowed.

4. Implementation of the database using a centralized database system.

5. Users must be enabled to access the website and login with their respective username and password credentials only.

6. The challenges in developing the product will involve scaling it to the required number of stakeholders. So providing fast request response time will be a necessity.

5.6 SECURITY :

Using proper login system ensures that no other user will be able to make changes in other people post. By providing features of controlling comment section of post, blogger can control content being shown on the site.

Also by providing regular backups we keep your data safe, so in case of some disparity data can be restored and blog site will work properly.

6.Database Design

6.1 ENTITY – RELATIONSHIP MODEL :
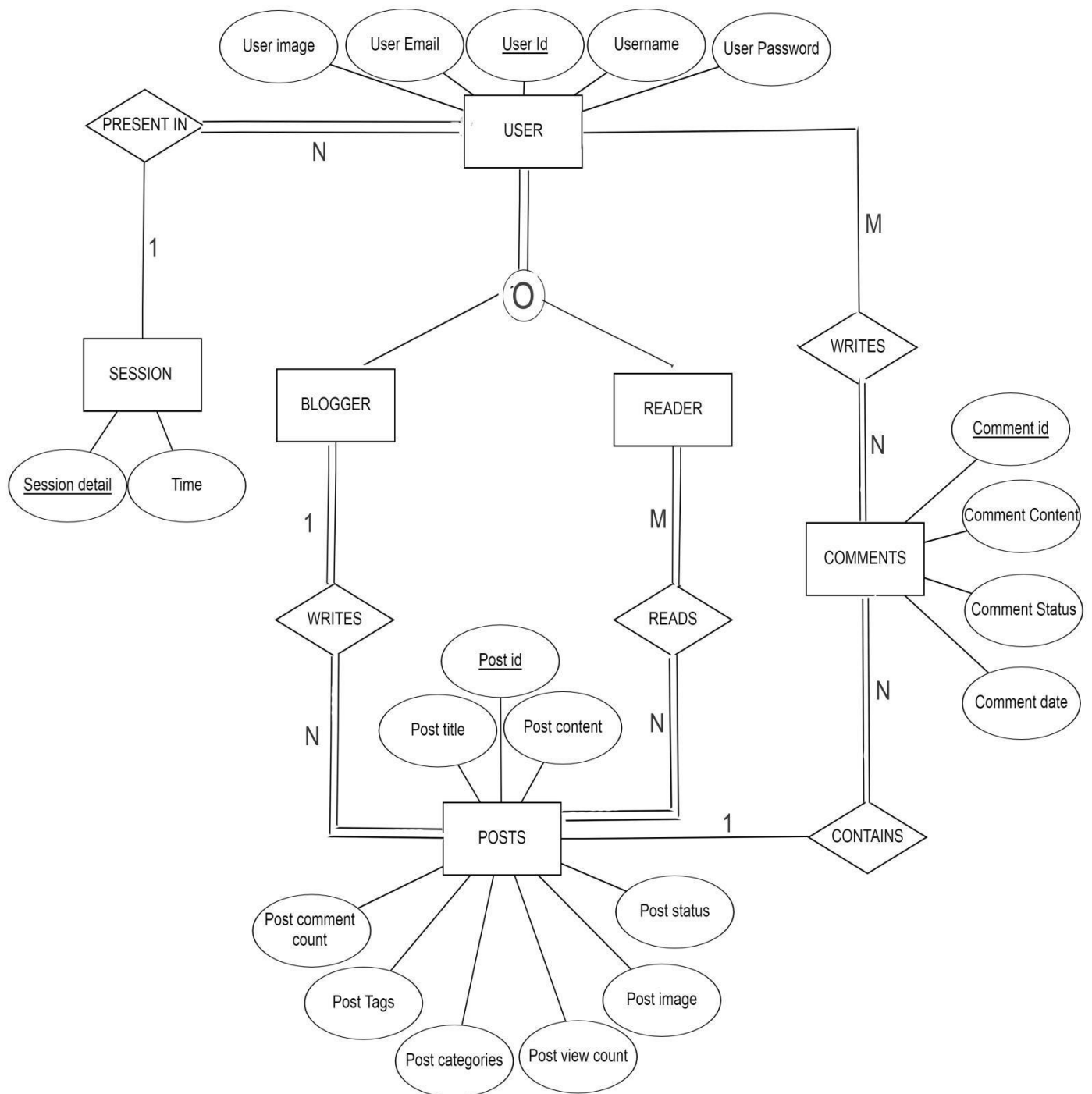
6.2 COMMON TABLES :

1. USERS

COLUMN NAME	TYPE	DESCRIPTION
User id	INT	Primary key, Unique id Given to all users after registration.
Username	VARCHAR(255)	Unique , not NULL Used while login also.
User password	VARCHAR(255)	Not Null Used while login also.
User email	VARCHAR(255)	Unique , Not Null Refers to registered email address of user.
User image	TEXT	Image of user that will be displayed on screen.
User role	VARCHAR(255)	Not Null Role can be either Blogger or reader only.

2. USERS-ONLINE

COLUMN NAME	TYPE	DESCRIPTION
Session	VARCHAR(255)	Primary key ,UNQIUE Uniquely generated session id for a particular times.
Id	INT	Foreign key, Not NULL Refers to user's table user Id.
Time	INT	Unique, Not Null Refers to time session started

3. **CATEGORIES**

COLUMN NAME	TYPE	DESCRIPTION
Cat_id	INT(3)	Primary key ,UNQIUE Refers to id given to categories of post.
Cat_title	VARCHAR(255)	Not NULL Specify title of category.

4. **POSTS**

COLUMN NAME	TYPE	DESCRIPTION
Post_id	INT(3)	Primary key ,UNQIUE Uniquely defines every post.
Post_cat_id	INT(3)	Foreign key, Not NULL Specify category the post belongs to. Referenced to Categories relation.
Post_title	VARCHAR(255)	Unique, Not Null Specify title of post.
Post_author_id	INT	Foreign key, Not Null Referenced from Users, tells id of author.
Post_author	VARCHAR(255)	Not Null Displayed in blogs as it's Author
Post_date	DATE	Auto generated Tells time post was Written.
Post_image	TEXT	Tells image to be displayed for the post.
Post_content	TEXT	Not Null Main content of post
Post_comment_count	INT	Auto generated, Not Null Count comments on post.
Post_status	VARCHAR(255)	Not Null, Default draft Approved, default, reject
Post_views_count	INT	Auto generated, Not Null Counts views on post.

5. COMMENTS

COLUMN NAME	TYPE	DESCRIPTION
Comment_id	INT(3)	Primary key ,UNQIUE Refers to id given to categories of post.
Comment_post_id	VARCHAR(255)	Not NULL Specify title of category.
Comment_author_id	INT	Foreign key, Not Null Refers comment writer id. His name will be displayed below comment.
Comment_content	TEXT	Not Null Content of comment.
Comment_status	VARCHAR(255)	Not Null
Comment_date	DATE	

6.3 DATA TYPES :

CHAR(n)	fixed-length character string
DATE	calendar date (year, month, day)
INTEGER	signed four-byte integer (-2147483648 to +2147483647)
TEXT	variable-length character string
TIME	time of day
TIMESTAMP	date and time
VARCHAR(n)	variable-length character string

6.4 RELATIONAL SCHEMA :

USERS_ONLINE

Session
Time
User id   (FK)

USERS

User id
Username
User_password
User email
User image
User_role

Comments

Comment_id
Comment_author_id   (FK)
comment_post_id   (FK)
Comment_content
Comment_status

POSTS

Post_id
Post_Cat_id   (FK)
Post_view_count
Post_title
Post_author_id   (FK)
Post_author
Post_date
Post_image
Post_content
Post_comment_count
Post_status

CATEGORIES

Cat_id
Cat_title