CRUD based Blog web application (BlogScript)

Project Report

Version 3.0

SUMMER INTERNSHIP REPORT EVALUATION

BCASI23501

BACHELOR OF COMPUTER APPLICATION

PROJECT GUIDE: **SUBMITTED BY:**

MR. NITIN KUMAR SAINI   PANKAJ KUMAR DAS
& MS. VARTIKA GUPTA   RAHU KUMAR SENUHAR
                                                              BANTY MISHRA

December 2025

COLLEGE OF COMPUTING SCIENCE & INFORMATION TECHNOLOGY

TEERTHANKER MAHAVEER UNIVERSITY, MORADABAD

DECLARATION

We hereby declare that this Project Report title **CRUD based blog web application (BlogScript)** submitted by us and approved by our project guide, Faculty of Engineering and Computing Sciences . Teerthanker Mahaveer University, Moradabad is a Bonafide work undertaken by us and it is not submitted to any other university or institution for the award of any degree diploma / certificate or published any time before .

Project ID:		
Title	**Name**	**Signature**
Student Name :	Pankaj Kumar Das (TCA2301489)	
Student Name :	Rahul Kumar Senuhar(TCA2301558)	
Student Name :	Banty Mishra (TCA2301154)	
Project Guide :	Mr. Nitin Kumar Saini	
Project Guide :	Ms. Vartika Gupta	

Table Of Contents

Appendix Details:

A. Data Flow Diagram (DFD)

B. Entity Relationship Diagram (ERD)

C. Use Case Diagram (UCD)

D. Data Dictionary (DD)

E. Screenshots

1. Project Title

CRUD-based blog web application it is designed to simplify the process of creating, managing, and sharing blog posts online. The term CRUD stands for Create, Read, Update, and Delete, which represent the four essential operations used to manage data in any web application.

In BlogScript, users can easily create new blog posts with titles, content, and images through an intuitive editor. Once published, readers can view and read the posts in a clean, responsive interface optimized for both desktop and mobile devices. Authors have the flexibility to edit or update existing articles whenever changes are needed, ensuring that content remains relevant and accurate. If a post is no longer required, it can be deleted permanently from the system

2. Project Domain

Web development in PHP & MySQL , using this technology create a user-friendly blog application where a user can create , read , update delete and post their scripted blogs .

3. Problem Statement

Creators often face difficulties in finding a minimal platform where they can quickly publish their ideas without distractions. They require a system where content ownership is secure, and posts can be managed anytime.

BlogScript provides a solution by offering a **personal blogging space** where users can manage their profiles, write blogs, and update or delete them whenever needed. It focuses on **ease of use, content control, and secure authentication** for creators.

4. Project Description

BlogScript is a CRUD-based web application designed for blogging enthusiasts who want a simple and user-friendly platform to share their ideas. The platform allows users to **sign up, log in, create, update, and delete blog posts** securely. It also provides a password reset feature to ensure account safety. With a clean UI and structured database, BlogScript focuses on making content creation smooth and accessible.

4.1 Scope of the Work

The scope of this project involves building a **full-stack web application** using PHP and MySQL as backend technologies and HTML, CSS, and JavaScript for frontend development. The system includes:

- User registration and login system with password hashing

- Post creation, editing, and deletion

- Secure password reset via email verification

4.2   Project Modules

1.   **User Authentication Module**

- **Sign Up**: Allows new users to create an account with username, email, and password.

- **Login**: Provides secure access to the platform using email and password.

- **Email Verification**: Confirms the user's email before account activation.

- **Password Reset / Update**: Enables users to reset or update their password securely via email.

2.   **Blog Post Management Module**

- **Create Post**: Users can write and publish blogs.

- **Read Post**: Blogs are displayed on the homepage or dashboard.

- **Update Post**: Users can edit/update their previously created posts.

- **Delete Post**: Users can remove their posts from the platform.

3.   **Email Integration Module**

- Sends verification emails during sign-up.

- Delivers password reset link to registered users.

- Sends notification emails (optional future scope).
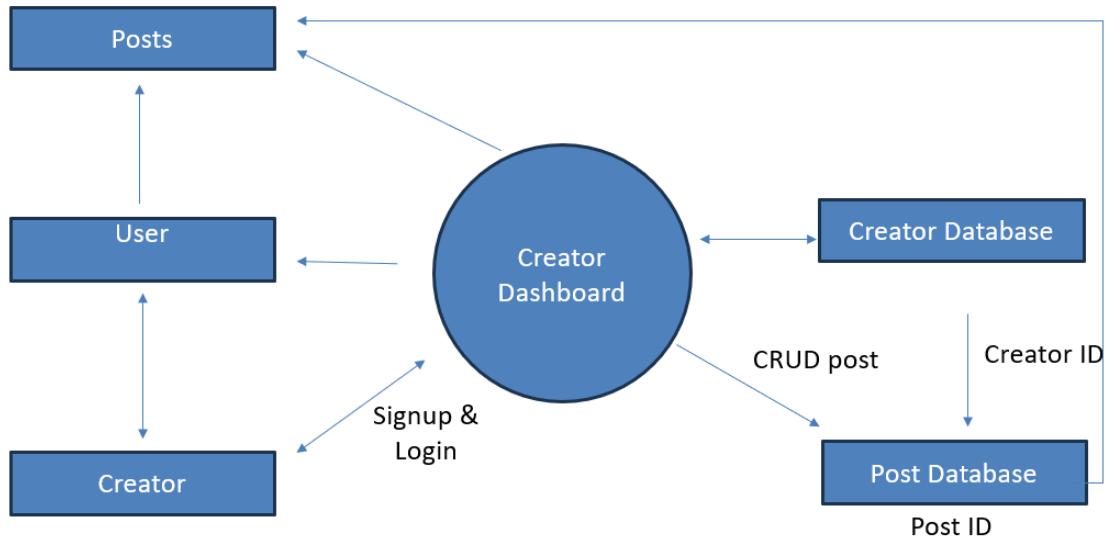
4.   **Database Management Module**

- Database schema includes users and posts tables.

- Handles CRUD operations for both users and posts.

- Ensures relational integrity between users and their posts.

5.   **Security Module**

- Passwords are hashed using **PHP's password_hash()** and **password_verify()**

- SQL Injection prevention using **prepared statements**.

- Secure session management and validation.

4.3 Context Diagram
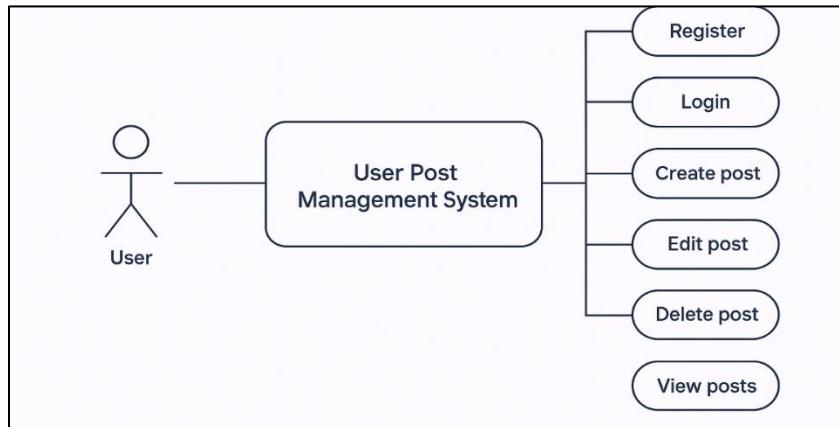
5. Implementation Methodology

Step 1: Requirement Analysis

- Identified core features: user authentication, blog post management, and email verification.

- Finalized technology stack: **PHP, MySQL, HTML, CSS, JavaScript, PHPMailer**.

Step 2: System Design

- Designed the **database schema** with users and posts tables.

- Designed responsive frontend layouts for signup, login, dashboard, and post pages.

Step 3: Backend Development

- Implemented **User Authentication** (signup, login, password hashing).

- Added **Email Verification** using **PHPMailer**.

- Developed CRUD operations for **posts** (create, read, update, delete).

- Implemented **password reset functionality** with email link.

Step 4: Frontend Development

- Built responsive UI using **HTML, CSS (custom + media queries), and JavaScript**.

- Integrated forms for login, signup, post creation, and update.

Step 5: Integration

- Integrated database queries with prepared statements for security.

- Linked PHPMailer for sending verification and reset emails.

Step 6: Testing

- Conducted **unit testing** for individual modules.

- Performed **integration testing** for login, signup, and post management.

- Tested the platform on different devices for **responsive design**.

Step 7: Deployment & Future Scope

- Deployed the project on **local XAMPP server** for testing.

- hosted on a **live server** with domain and SSL for real-world use.

- Future scope includes adding **admin dashboard, comments, and SEO optimization**.

6. Technologies to be used

6.1 Software Platform

a) Frontend Technologies

- HTML5 - For creating the structure and layout of web pages.

- CSS3 - For styling, responsiveness, and UI design.

- JavaScript (jQuery) - For client-side validation, interactivity, and DOM manipulation.

b) Backend Technologies

- PHP (Core PHP) - For server-side scripting, handling requests, implementing business logic, and integrating with the database.

- PHPMailer - For sending verification and password reset emails using SMTP.

c) Database

- MySQL - To store and manage user accounts, authentication data, and blog posts.

- Users table (for user details, passwords, verification codes)

- posts table (for blog posts created by users)

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	ID 🔑	int			No	None		AUTO_INCREMENT	🖉 Change	⊖ Drop	More
☐	2	USER_NAME	varchar(16)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More
☐	3	EMAIL	varchar(28)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More
☐	4	PASSWORD	varchar(60)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More
☐	5	VERIFICATION_CODE	int			Yes	NULL			🖉 Change	⊖ Drop	More
☐	6	VERIFICATION_STATUS	varchar(7)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
☐	1	id 🔑	int			No	None		AUTO_INCREMENT	🖉 Change	⊖ Drop	More
☐	2	title	varchar(40)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More
☐	3	content	varchar(2346)	utf8mb3_general_ci		Yes	NULL			🖉 Change	⊖ Drop	More
☐	4	created_at	date			Yes	curdate()		DEFAULT_GENERATED	🖉 Change	⊖ Drop	More
☐	5	user_id	int			Yes	NULL			🖉 Change	⊖ Drop	More

d) Server Environment

- XAMPP (Apache, MySQL, PHP, Perl) - Used as the local development environment.

- Apache Server - To host and run the PHP application locally.

e) Development Tools

- VS Code - Code editor for writing and debugging code.

- phpMyAdmin - For managing MySQL databases.

- Git/GitHub (optional) - For version control and project collaboration.

f) Other Supporting Tools

- Google Fonts (Poppins) - For better typography.

- Browser Developer Tools - For testing responsiveness and debugging frontend issues.

g) Hosting Server

- Infinity free - For free hosting and testing server with SSL certificate and sub domain.

6.2 Hardware Platform

RAM, Hard Disk , OS, Editor etc

7. Advantages of this Project

- Simple and user-friendly blogging platform.

- Secure login, signup, and password reset with hashing.

- Email verification to avoid fake accounts.

- Full blog management with **CRUD operations**.

- Responsive design for mobile and desktop

- Practical for real-world blogging needs.

- Customizable for different users.

- Enhances full-stack development skills.

8. Assumptions, if any

The project assumes that users have basic internet access and fundamental knowledge of using web applications. It is also assumed that users will provide valid email addresses for account verification and password recovery. The system is designed for small to medium-scale usage, where server resources and network connectivity remain stable.

9. Future Scope and further enhancement of the Project

- **Comments & Likes System** – For better engagement between readers and authors

- **Rich Text Editor** – To allow users to format text, add images, and embed media.

- **SEO Optimization** – For making blogs search engine–friendly.

- **User Profile Customization** – With profile pictures, bio, and social links.

- **Notification System** – Email or in-app alerts for new posts, likes, and comments.

- **Multi-language Support** – To make the platform usable for wider audiences.

- **Mobile App Integration** – Extending BlogScript as an Android/iOS application.

10. Project Repository Location

Repository: https://github.com/Pankajdas0025/ApexPlanet_Internship_Project

Live : https://blogscriptapp.free.nf/

S#	Project Artifacts	Location	Verified by Project Guide	Verified By Lab In-Charge
1.	Project Synopsis Report	Pankajdas0025/ApexPlanet_Internship_Project/Report/Synopsis		
2.	Project Progress updates	Pankajdas0025/ApexPlanet_Internship_Project/Report/Update		
3.	Project Requirement Specification	Pankajdas0025/ApexPlanet_Internship_Project/Report		
4.	Project Report	Pankajdas0025/ApexPlanet_Internship_Project/Report		
5.	Test Repository	Pankajdas0025/BlogScript-website		
6.	Project Source Code	https://github.com/Pankajdas0025/ApexPlanet_Internship_Project		

11. Definition , Acronyms, and Abbreviations

- **CRUD**: Create, Read, Update, Delete – basic operations performed on database records.
- **PHP**: Hypertext Preprocessor – a server-side scripting language used for web development.
- **MySQL**: An open-source relational database management system.
- **SMTP**: Simple Mail Transfer Protocol – used for sending emails.
- **UI**: User Interface – the visual part of the application.
- **UX**: User Experience – overall experience of users while interacting with the system.
- **ERD**: Entity Relationship Diagram – represents database structure.
- **DFD**: Data Flow Diagram – shows flow of data within the system.

12. Conclusion

The BlogScript project successfully demonstrates the design and implementation of a CRUD-based blog web application that addresses the essential needs of modern content creators. By providing a clean and intuitive interface, the system allows users to register securely, authenticate their accounts, and manage blog posts efficiently through create, read, update, and delete operations. The inclusion of email verification and password reset functionality strengthens account security and enhances user trust in the platform.

From a technical perspective, the project effectively integrates frontend technologies such as HTML, CSS, and JavaScript with backend technologies like Core PHP and MySQL, ensuring smooth interaction between the user interface and the database. The use of prepared statements and password hashing techniques helps protect sensitive user data and prevents common security vulnerabilities. Additionally, the modular architecture of BlogScript improves code maintainability and allows future developers to extend features with minimal effort.

Overall, BlogScript serves as a practical real-world application that reflects strong understanding of full-stack web development concepts. It not only fulfils the academic objectives of the internship but also provides a scalable foundation for future enhancements such as comments, admin controls, SEO optimization, and mobile integration. The project enhances problem-solving skills, reinforces database design knowledge, and prepares the developer for professional web development challenges.
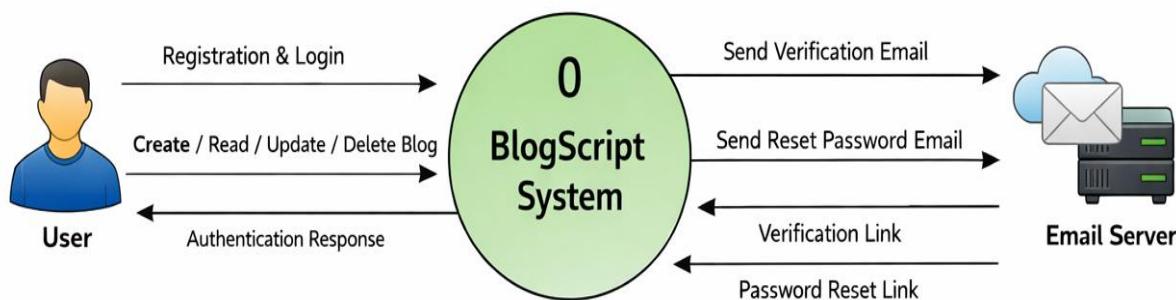
13. References

a) PHP Documentation Team. *PHP Manual*. Available at:

https://www.php.net/docs.php

b) Oracle Corporation. *MySQL Documentation*. Available at:

https://dev.mysql.com/doc/

c) Geeksforgeeks. *Web Development Tutorials (HTML, CSS, JavaScript, PHP)*. Available at:

https://www.geeksforgeeks.org

d) W3C. *HTML5 Specification*. Available at:

https://www.w3.org/TR/html5/

e) Mozilla Developer Network (MDN). *Web Docs for HTML, CSS, and JavaScript*. Available at:

https://developer.mozilla.org

f) GitHub. *GitHub Documentation*. Available at:

https://docs.github.com

14. Appendices
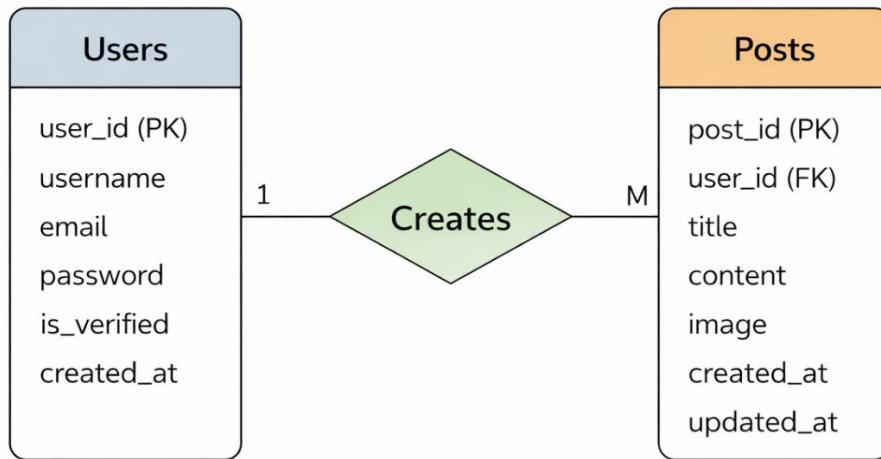
A. Data Flow Diagram (DFD)

Shows how data moves between users, the application, and the database during login, blog creation, and management processes.
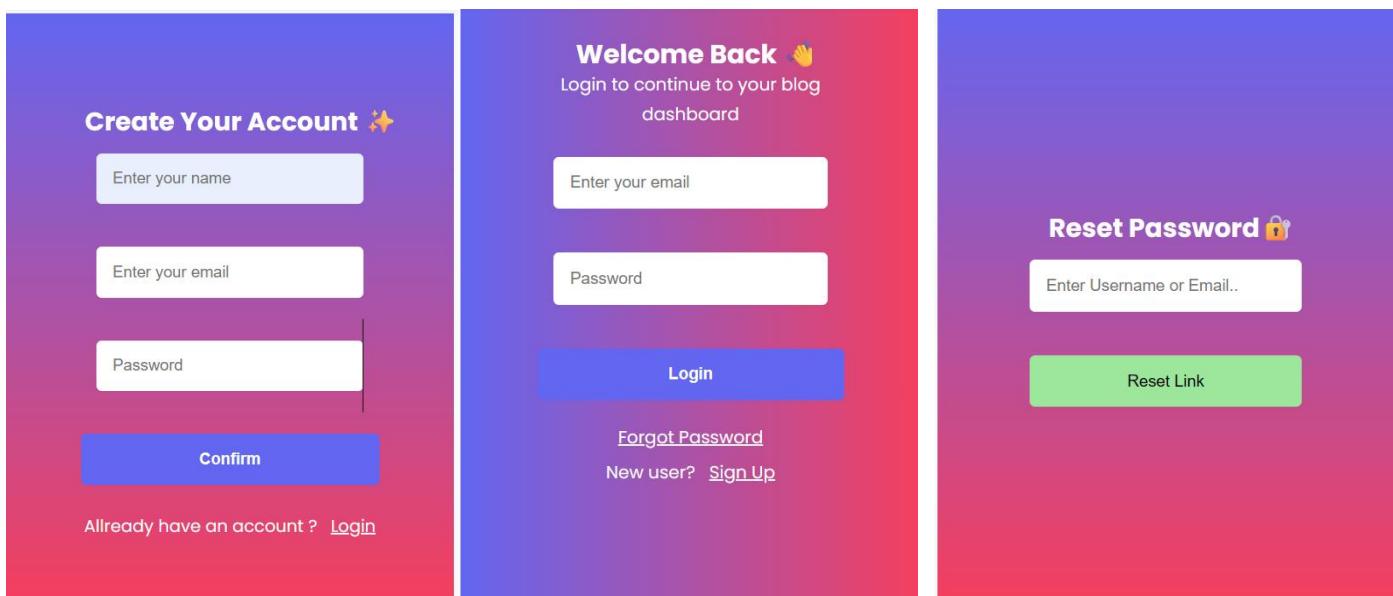
B. Entity Relationship Diagram (ERD)

Illustrates the relationship between users and posts tables, where one user can create multiple blog posts.

C. Use Case Diagram (UCD)

Depicts user interactions such as registration, login, create blog, edit blog, delete blog, and logout.

D. Data Dictionary (DD)

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
1	ID 🔑	int			No	None		AUTO_INCREMENT	Change	Drop	More
2	USER_NAME	varchar(16)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More
3	EMAIL	varchar(28)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More
4	PASSWORD	varchar(60)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More
5	VERIFICATION_CODE	int			Yes	NULL			Change	Drop	More
6	VERIFICATION_STATUS	varchar(7)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
1	id 🔑	int			No	None		AUTO_INCREMENT	Change	Drop	More
2	title	varchar(40)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More
3	content	varchar(2346)	utf8mb3_general_ci		Yes	NULL			Change	Drop	More
4	created_at	date			Yes	curdate()		DEFAULT_GENERATED	Change	Drop	More
5	user_id	int			Yes	NULL			Change	Drop	More

Attachments

CERTIFICATE
OF INTERNSHIP

VaultofCodes

THIS CERTIFICATE IS AWARDED TO

Rahul Kumar Senuhar

EMAIL ID: rahulsenuharofficial@gmail.com

UNIVERSITY STUDENT ID: TCA2301558
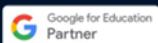
AICTE STUDENT ID: STU682632e6a16e71747333862

HAS SUCCESSFULLY COMPLETED A

2 Month Internship in Web Development

at **VaultofCodes.in**

Starting from 08/06/2025

Kartik Ahlawat
Founder

Varshita Singh
HR Manager

Google for Education Partner

GET IT ON Google Play

TO VERIFY THE AUTHENTICITY OF THIS CERTIFICATE AND ACCESS THE STUDENT'S DETAILED PERFORMANCE REPORT AND PROJECT STATISTICS, SCAN THE QR CODE PROVIDED.