



CSE370 : Database Systems
Project Report
Project Title
Brac University Blogger

Group No : 05 , CSE370 Lab Section : 10, Fall 2023		
ID	Name	Contribution
21201728	Md. Istiak Hossain	Frontend & Backend
21301217	Swapnil Saha	Frontend & Backend
21201775	Munira Akter Mysha	Frontend & Backend

Table of Contents

Section No	Content	Page No
1	Introduction	03
2	Project Features	03
3	ER/EER Diagram	04
4	Schema Diagram	04
5	Frontend Development	05-13
6	Backend Development	14-25
7	Conclusion	26
8	References	26

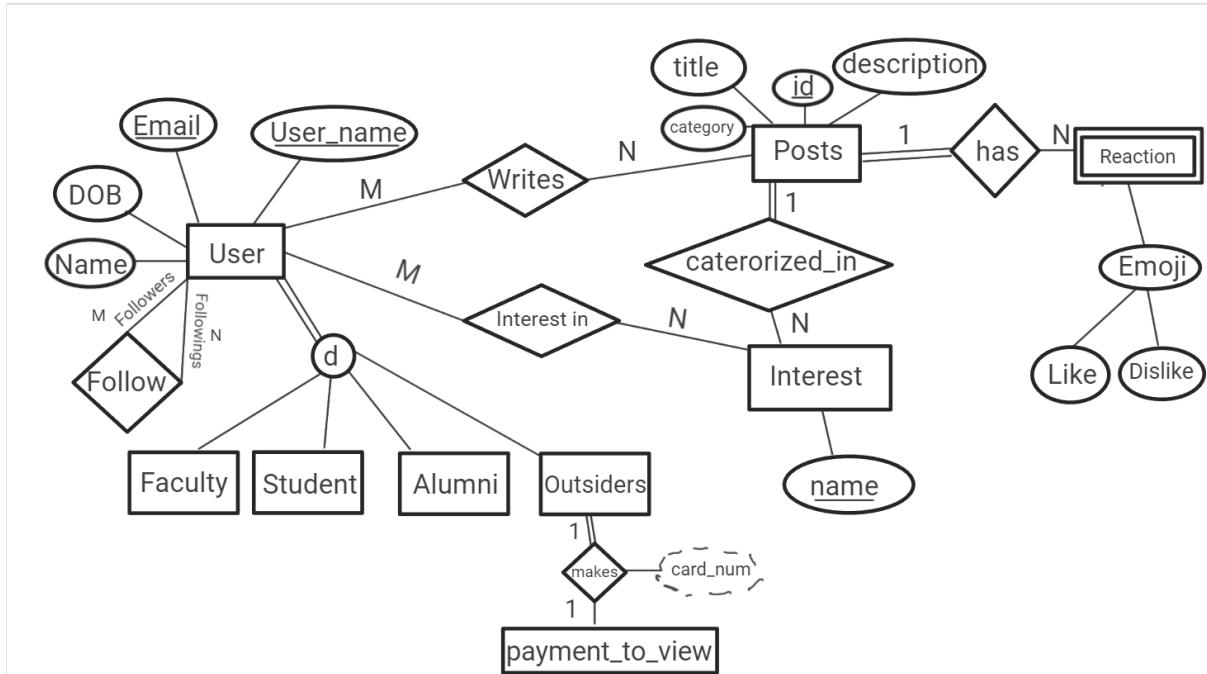
Introduction

Brac University Blogger is a lightweight micro blogging website where we are providing facilities for the members of the Brac University community to connect with each other and share personalized opinions and thoughts over different topics. This great opportunity would help all the Bracian members regardless of their position in the University to be active in the community lifetime to get benefited and actively communicate with their fellows from any part of the world. This platform is built on a very popular tech stack HTML5, CSS3, JavaScript and PHP.

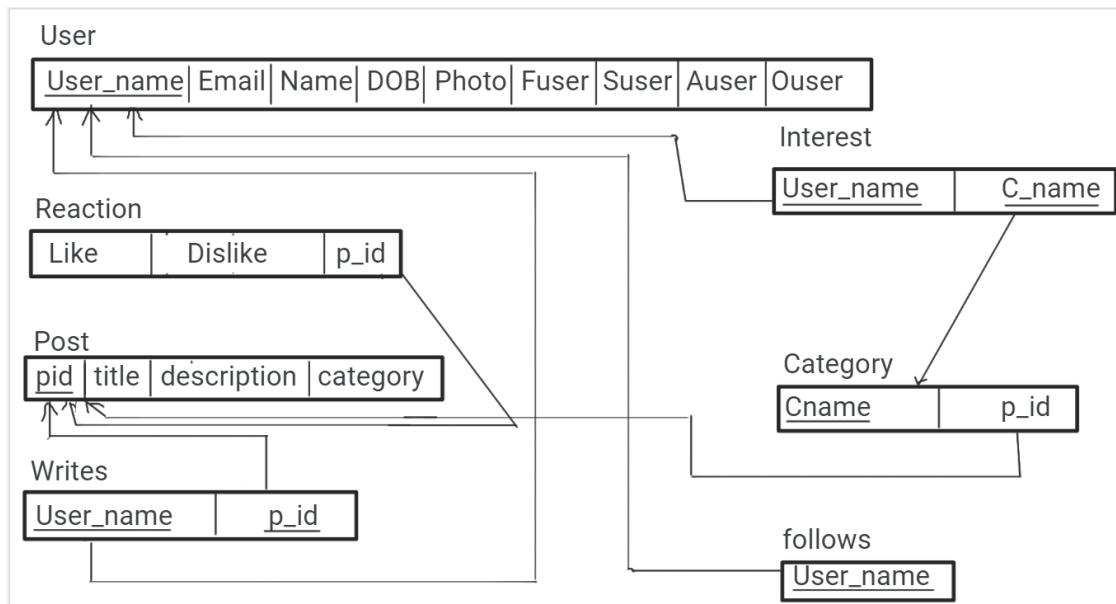
Project Features

University students, faculty, and alumni are granted free access to the platform, while outsiders will be required to use a payment system. The blogs on the platform will be categorized in two ways, with a recursive relationship between followers and followings. The homepage will display all blogs, showcasing those of users the individual follows. Users can also indicate their interests on their homepage. Additionally, users have the ability to write posts, which can give & receive ratings such as comments and reactions.

ER/EER Diagram



Schema Diagram



Frontend Development

Briefly discuss about Frontend Development and add Screenshots by mentioning Individual Contributions

Contribution of ID : 21301217, Name : Swapnil Saha

Registration, HomePage with css, Follow Page, Create Post, Sign Out button

Date Of Birth

Student Faculty Alumni Outsider

Profile Picture

Already Registered?? [Login Here](#)

Registration form for the new come users.

Follow/Unfollow Users

Username to Follow/Unfollow:

[Follow](#)

[Unfollow](#)

[Home](#)

The window whom do you want to follow or unfollow.

Explore
Interest
Follow
Sign Out

Swapnil Saha
Username: @SwapnilSaha
Hola!

Description: Hello World!
Category: Others
2023-12-11 13:16:20
[Like \(0\)](#) [Dislike \(0\)](#)

Swapnil Saha
Username: @SwapnilSaha
Exam

Description: How is Exam??
Category: Education
2023-12-11 09:19:33
[Like \(1\)](#) [Dislike \(0\)](#)

What's in your mind?

[Post](#)

Calendar

		Day	Month	January	February	March	April
BRAC UNIVERSITY Inspiring Excellence		Sunday	1			30	
		Monday	2			1	
Tuesday	3			2			
Wednesday	4	Wednesday		1	1		
Thursday	5			2	2		
Friday	6			3	3		
Saturday	7			4	4	1	
Sunday	8	Sunday		5	5	2	
Monday	9			6	6	3	
Tuesday	10			7	7	4	
Wednesday	11			8	8	5	
Thursday	12			9	9	6	
Friday	13			10	10	7	
Saturday	14			11	11	8	
Sunday	15	Sunday		12	12	9	
Monday	16			13	13	10	
Tuesday	17			14	14	11	
Wednesday	18			15	15	12	

Homepage newsfeed, where user can show his own post and also the post whom he followed. He can rate the blog by clicking like or dislike button. Also put the academic calendar

Brac University Blogger

Welcome!

Enter User Name:

Enter Full Name:

Enter Password:

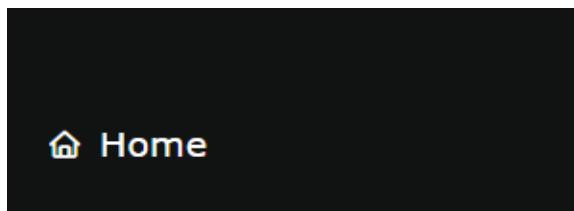
Login

Not registered yet?? [Register Here](#)

Login page search by Username, fullname..

Contribution of ID : 21201728, Name : Md. Istiak Hossain

Home Page Side bar with css, Payment gateway



Profile

Search

Write Post

Sign Out



Md. Istiak Hossain

user: @Md. Istiak

Hossain

Homepage side bar . It will come after user login successfully.

Payment ID:

Password:

Amount:

Submit Payment

Payment option has been added for the outsiders.

		Calendar		
Wednesday	4	Last Day of Submitting Classes	1	1
Thursday	5		2	2
Friday	6		3	3
Saturday	7		4	4
Sunday	8	Publication of Results, Fall 2022	5	5
Monday	9		6	Mid-Term Examinations
Tuesday	10		7	
Wednesday	11		8	8 Shab-e-Barat*
Thursday	12		9	9 Mid-Term Examinations
Friday	13		10	10
Saturday	14		11	11
Sunday	15	Payment of Tuition Fees Spring 2023 (15 Jan - 10 Feb)	12	12 Mid-Term Examinations
Monday	16		13	13 Mid-Term Examinations
Tuesday	17	Registration, Spring 2023	14	14
Wednesday	18		15	15 Mid-Term Examinations (PwC)
Thursday	19	FIEST DAY	16	16 Vacation RD
Friday	20		17	17 Birthday of Father of the Nation
Saturday	21	Courses of Spring 2023 Begin	18	18
Sunday	22		19	19
Monday	23		20	20
Tuesday	24	21 International Mother Language Day	21	18
Wednesday	25		22	19 Shab-e-Qadr*
Thursday	26		23	20
Friday	27		24	24 Undergraduate Admissions Test for Semester 2023 (Remaining)
Saturday	28	RD Orientation	25	25
Sunday	29		26	26 Independence Day
Monday	30		27	23 Eid-ul-Fitr*
				24

A calendar option has been added.

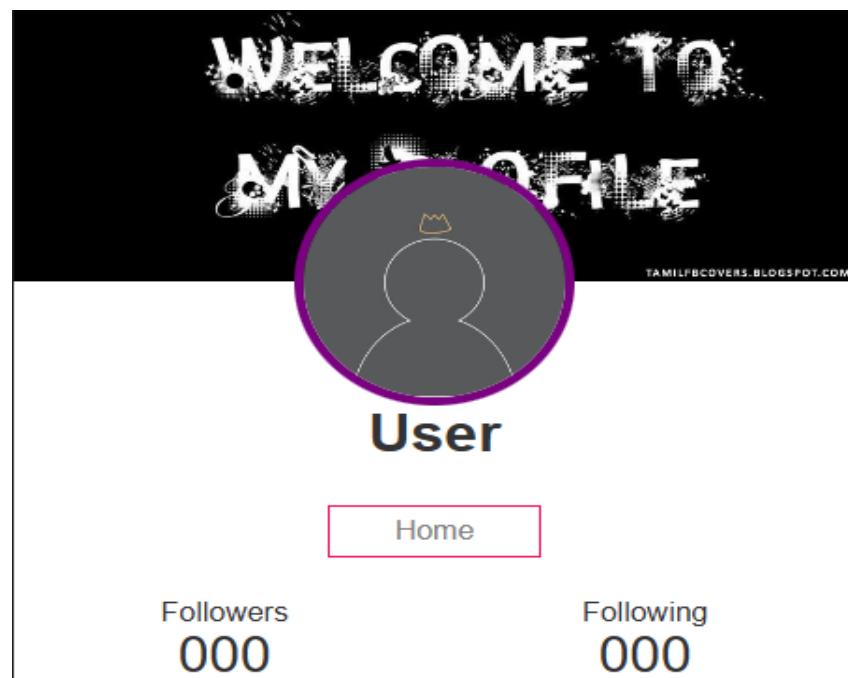
What's in your mind?

Post

A Quick Post button has been added.

Contribution of ID : 21201775, Name : Munira Akter Mysha

Profile Page, Friend's User Page, Search page, Login Page, Profile Picture Bar, Update Image Bar



Profile of the people who have an account.



user: **@SwapnilSaha**

[Home](#)

Followers

2

Following

0

Profile of the people who have an account and are friends together.

Search

Enter Username:

[submit](#) [Home](#)

Search button url where user can look up for his friends and watch their profile.

LOGIN

User name

Password

[Login](#)

[Sign Up](#)

Login page for users who are already registered. Outsiders can also sign up here for a new account.

Profile Picture

[Choose File](#)

No file chosen

[Sign Up](#)

[Login](#)

Attaching an image of the user while creating an account.

Profile Picture

[Choose File](#)

No file chosen



[Update](#)

[Home](#)

Updating profile picture of the user.

Backend Development

Briefly discuss about Backend Development and add Screenshots by mentioning Individual Contributions

Tables_in_bracuniversityblogger
follows
post
reactions
users

```
MariaDB [bracuniversityblogger]> describe post;
+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default          | Extra           |
+-----+-----+-----+-----+-----+
| id         | int(100)   | NO   | PRI  | NULL            | auto_increment |
| user_id    | int(11)    | NO   | MUL  | NULL            |
| title      | varchar(200)| NO  |       | NULL            |
| description | varchar(300)| NO  |       | NULL            |
| category   | varchar('50')| NO  |       | NULL            |
| timestamp_column | timestamp | NO  |       | current_timestamp() |
+-----+-----+-----+-----+-----+
6 rows in set (0.020 sec)

MariaDB [bracuniversityblogger]> describe follows;
+-----+-----+-----+-----+-----+
| Field      | Type     | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| follower_id | int(11) | NO   | PRI  | NULL    |
| following_id | int(11) | NO   | PRI  | NULL    |
+-----+-----+-----+-----+
2 rows in set (0.009 sec)

MariaDB [bracuniversityblogger]> describe users;
+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default          | Extra           |
+-----+-----+-----+-----+-----+
| id         | int(11)    | NO   | PRI  | NULL            | auto_increment |
| fullname   | varchar(30) | NO   |       | NULL            |
| username   | varchar(30) | NO   | UNI  | NULL            |
| iden       | enum('S','F','A','O') | NO  |       | NULL            |
| DateOfBirth | date      | YES  |       | NULL            |
| email      | varchar(250) | NO  |       | NULL            |
| password   | varchar(250) | NO  |       | NULL            |
| pp         | varchar(255) | NO  |       | default.png    |
+-----+-----+-----+-----+-----+
8 rows in set (0.008 sec)

MariaDB [bracuniversityblogger]> ■
```

```

MariaDB [(none)]> use bracuniversityblogger;
Database changed
MariaDB [bracuniversityblogger]> describe reactions;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| id | int(11) | NO | PRI | NULL | auto_increment |
| blog_id | int(11) | YES | MUL | NULL | |
| user_id | int(11) | YES | MUL | NULL | |
| reaction_type | enum('like','dislike') | NO | | NULL | |
+-----+-----+-----+-----+-----+
4 rows in set (0.019 sec)

```

Reaction table: CREATE TABLE reactions (

```

id INT PRIMARY KEY AUTO_INCREMENT,
blog_id INT,
user_id INT,
reaction_type ENUM('like', 'dislike') NOT NULL,
FOREIGN KEY (blog_id) REFERENCES blogs(id),
FOREIGN KEY (user_id) REFERENCES users(id)
);

```

Follows table: CREATE TABLE follows (

```

follower_id INT,
following_id INT,
PRIMARY KEY (follower_id, following_id),
FOREIGN KEY (follower_id) REFERENCES users(id),
FOREIGN KEY (following_id) REFERENCES users(id)
);

```

Post: CREATE TABLE post (

```

id INT AUTO_INCREMENT PRIMARY KEY, user_id INT,
title VARCHAR(255) NOT NULL,
description TEXT,
category VARCHAR(50),
timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
ALTER TABLE `blogs` ADD INDEX(`id`);
ALTER TABLE `blogs` ADD CONSTRAINT `u_id` FOREIGN KEY (`id`) REFERENCES
`users`(`id`) ON DELETE RESTRICT ON UPDATE RESTRICT;

```

Users: CREATE TABLE users (

```

id INT AUTO_INCREMENT PRIMARY KEY,
fullname VARCHAR(255) NOT NULL,
username VARCHAR(255) NOT NULL,
iden ENUM('S', 'F', 'A', 'O') NOT NULL,
dob DATE NULL,
email VARCHAR(255) NOT NULL,
password VARCHAR(255) NOT NULL
);

```

Contribution of ID : 21301217, Name : Swapnil Saha

Follow and Following:

```
$checkUserQue = "SELECT id FROM users WHERE username = '$follower'";
$results = $conn->query($checkUserQue);
$followerId = $results->fetch_assoc()['id'];

if ($result->num_rows == 1) [
    $checkFollowQuery = "SELECT * FROM follows WHERE follower_id = $followerId AND following_id =
        (SELECT id FROM users WHERE username = '$following')";
    $resultFollow = $conn->query($checkFollowQuery);

    if (isset($_POST["follow"])) {
        if ($resultFollow->num_rows == 0) {
            $followQuery = "INSERT INTO follows (follower_id, following_id) VALUES
                ($followerId, (SELECT id FROM users WHERE username = '$following'))";
            $conn->query($followQuery);
            echo "You are now following $following.";
        } else {
            echo "You are already following $following.";
        }
    } elseif (isset($_POST["unfollow"])) {
        if ($resultFollow->num_rows > 0) {
            $unfollowQuery = "DELETE FROM follows WHERE
                follower_id = $followerId AND following_id = (SELECT id FROM users WHERE username = '$following')";
            $conn->query($unfollowQuery);
            echo "You have unfollowed $following.";
        } else {
            echo "You are not following $following.";
        }
    }
] else {
    echo "Invalid usernames.";
}
```

For the following and follower option connect the follow table with the username and id.

Follow and Following count:

```
}
$mysqli = new mysqli("localhost", "root", "", "bracuniversityblogger");

function getFollowerCount($userId, $mysqli) {
    $query = "SELECT COUNT(*) AS count FROM follows WHERE following_id = ?";
    $stmt = $mysqli->prepare($query);
    $stmt->bind_param("i", $userId);
    $stmt->execute();
    $result = $stmt->get_result();
    $row = $result->fetch_assoc();
    return $row['count'];
}

function getFollowingCount($userId, $mysqli) {
    $query = "SELECT COUNT(*) AS count FROM follows WHERE follower_id = ?";
    $stmt = $mysqli->prepare($query);
    $stmt->bind_param("i", $userId);
    $stmt->execute();
    $result = $stmt->get_result();
    $row = $result->fetch_assoc();
    return $row['count'];
}

$username = $_SESSION["username"];
$queryUser = "SELECT id FROM users WHERE username = '$username'";
$resultUser = $mysqli->query($queryUser);
$userRow = $resultUser->fetch_assoc();

if ($userRow) {
    $userId = $userRow['id'];
    $followerCount = getFollowerCount($userId, $mysqli);
    $followingCount = getFollowingCount($userId, $mysqli);
}
?>
```

In the follow table it is totally connected with users table. Shown you the count in the users profile.

Registration insert:

```
$sql = "INSERT INTO users (fullname, username, iden, DateOfBirth, email, password) VALUES ( ?, ?, ?, ?, ?, ? )";
$stmt = mysqli_stmt_init($conn);
$preparestmt = mysqli_stmt_prepare($stmt,$sql);
if ($preparestmt) {
    mysqli_stmt_bind_param($stmt,"sssdss",$fullName, $username, $iden, $dateofbirth, $email, $passwordHash);
    mysqli_stmt_execute($stmt);
    echo "<div class='alert alert-success'>You are registered successfully.</div>";
} else{
    die("Something went wrong");
}
```

For registration we have insert the values in the database users.

Post displayed in homepage:

```
q1 = "SELECT post.* , users.fullname , users.username
      ON post
      IN users ON post.user_id = users.id
      WHERE post.user_id = $ui OR post.user_id IN (
          SELECT following_id FROM follows WHERE follower_id = $ui
      )
      ORDER BY post.timestamp_column DESC";
$result = $conn->query($sql);
if($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        $blogId = $row['id'];
        $likeCount = mysqli_fetch_assoc(mysqli_query($conn, "SELECT COUNT(*) AS count FROM reactions WHERE blog_id = $blogId AND reaction_type = 'like'"))['count'];
        $dislikeCount = mysqli_fetch_assoc(mysqli_query($conn, "SELECT COUNT(*) AS count FROM reactions WHERE blog_id = $blogId AND reaction_type = 'dislike'"))['count'];
        echo "<div class='post2'>";
        echo "<h2>{$row['fullname']}
```

In the post, we will gonna see the post which user will follow and which user will post. For that i had created the post table where post id is unique and there will be content inserted. If a user will publish the post the users unique id will also be stored in the post table. That's how we can generate user id also. We can also rate the post(like, dislike). Also the post will be order by timestamp desc.

Create post:

```
<?php

if (isset($_POST["publish"])) {
    $title = $_POST["title"];
    $description = $_POST["description"];
    $category = $_POST["category"];
    $errors = array();
    require_once "database.php";

    $follower = $_SESSION["username"];
    $checkUserQue = "SELECT id FROM users WHERE username = '$follower'";
    $results = $conn->query($checkUserQue);
    $userID = $results->fetch_assoc()['id'];

    $sql = "INSERT INTO post (user_id, title, description, category) VALUES (?, ?, ?, ?)";
    $stmt = mysqli_stmt_init($conn);
    $prepareStmt = mysqli_stmt_prepare($stmt, $sql);
    if ($prepareStmt) {
        mysqli_stmt_bind_param($stmt, "ssss", $userID, $title, $description, $category);
        mysqli_stmt_execute($stmt);
        echo "<div class='alert alert-success'>You have added successfully.</div>";
    } else {
        die("Something went wrong");
    }
}
```

To create a post, insert title, description and category also. As we created the login session so, the user's id will also be stored in the post table. From that, we can access username and fullname which helps us in displaying the post. The timestamp will be auto generated.

Handling the like-dislike post:

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $blogId = $_POST['blog_id'];

    $userId = $uid; // Replace with the actual user ID (assuming you have a logged-in user)
    $existingReaction = mysqli_fetch_assoc(mysqli_query($conn, "SELECT id, reaction_type FROM reactions WHERE blog_id = $blogId AND user_id = $userId"));
    $reactionType = $_POST['reaction'];
    if ($existingReaction) {
        $existingReactionId = $existingReaction['id'];
        mysqli_query($conn, "DELETE FROM reactions WHERE id = $existingReactionId");
    } else {
        mysqli_query($conn, "INSERT INTO reactions (blog_id, user_id, reaction_type) VALUES ($blogId, $userId, '$reactionType')");
    }
    header("Location: index.php");
}
```

In the reactions table, we will have post id will be the foreign key of post table id(auto increment). It will also stored the users id which will help to get the notification information. Also it will help to show us the like and dislike count.

Contribution of ID : 21201728, Name : Md. Istiak Hossain

Database:

```
MariaDB [(none)]> use bracuniversityblogger;
Database changed
MariaDB [bracuniversityblogger]> show tables;
+-----+
| Tables_in_bracuniversityblogger |
+-----+
| follows
| payment
| post
| users
+-----+
4 rows in set (0.001 sec)
```

Here I was responsible for creating the payment table in the database. Payment table contains the payment data of the outsiders. Other users do not need to access the payment table and its options.

Payment Table:

```
MariaDB [bracuniversityblogger]> describe payment;
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| payment_id | int(11)    | NO   | PRI | NULL    | auto_increment |
| amount      | decimal(10,2)| YES  |     | NULL    |                |
| password    | varchar(255) | NO   |     | NULL    |                |
| user_id     | int(11)    | YES  | MUL | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.027 sec)
```

Here I have created the payment table which has payment_id, amount, password, user_id attributes. They have type int, decimal, varchar, int respectively.

Payment.php:

```
$username = "root";
$password = "";
$dbname = "bracuniversityblogger";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Get the user's ID based on 'iden' of '0' from the users table
$username = $_SESSION['username']; // Assuming username is used for identification
$sql = "SELECT id FROM users WHERE iden = '0' AND username = '$username'";
$result = mysqli_query($conn, $sql);

if ($result && mysqli_num_rows($result) > 0) {
    $row = mysqli_fetch_assoc($result);
    $userId = $row['id']; // User's ID with 'iden' of '0'

    // Get payment details from the form
    $paymentId = $_POST['payment_id'];
    $password = $_POST['password'];
    $amount = $_POST['amount'];

    // Insert payment details into the payment table
    $insertSql = "INSERT INTO payment (user_id, payment_id, password, amount) VALUES ('$userId', '$paymentId', '$password', '$amount')";
    if (mysqli_query($conn, $insertSql)) {
        // Redirect to all_post.php after successful payment
        header("Location: all_post.php");
        exit();
    } else {
        // Failed to insert payment details
        echo "<p>Failed to insert payment details.</p>";
    }
} else {
    // User with 'iden' of '0' not found
    echo "<p>User with 'iden' of '0' not found.</p>";
}
} else {
    // Payment failed
```

Here the code implements the payment method for those users who have iden='0' which means that he is an outsider. So after login, he will get a payment cart page. If he is not an outsider, then he will directly be directed to the home page.

All_post.php:

```
<div style="padding: 10px 10px;" class="article-container flex-column">
<?php
$servername = "localhost";
$username = "root"; // Replace with your actual database username
$password = ""; // Replace with your actual database password
$dbname = "bracuniversityblogger"; // Replace with your actual database name

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Fetch posts from the 'post' table with associated usernames
$sql = "SELECT post.*, users.username
        FROM post
        INNER JOIN users ON post.user_id = users.id";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {

        echo "<div class='post2'>";
        echo "Username: @".<h7>{$row['username']}</h3>";
        echo "<h3>{$row['title']}
```

Here it is the all_post.php page where it shows all the posts of the users regardless of their identity. All the posts are sorted on the basis of time uploaded in the blog

Index.php:

```
<?php
$follower = $_SESSION["username"];
require_once "database.php";

$checkUserQue = "SELECT id FROM users WHERE username = '$follower'";
$results = $conn->query($checkUserQue);
$ui = $results->fetch_assoc()['id'];

$sql = "SELECT post.*, users.fullname , users.username
FROM post
JOIN users ON post.user_id = users.id
WHERE post.user_id = $ui OR post.user_id IN (
    SELECT following_id FROM follows WHERE follower_id = $ui
)
ORDER BY post.timestamp_column DESC";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        echo "<div class='post2'>";
        echo "<h3>{$row['fullname']}</h3>.".<br>";
        echo "Username: @".<h3>{$row['username']}</h3>";
        echo "<h3>{$row['title']}</h3>.".<br>";
        echo "Description: ."."<p>{$row['description']}</p>";
        echo "Category: ."."<p>{$row['category']}</p>";
        echo "<small>{$row['timestamp_column']}</small>";
        echo "</div>";
        echo "<br>";

    }
} else {
    echo "No posts found.";
}

?>
```

This is the index.php page which contains the post of the followers of a user which he has interest in. all the posts of the selected users that one user has interest in, shows in the wall after hitting the button interest on the top, in the options div

Contribution of ID : 21201775, Name : Munira Akter Mysha

LOGIN:

```
<?php
if (isset($_POST["login"])) {
    $username = $_POST["username"];
    $fullname = $_POST["fullname"];
    $password = $_POST["password"];
    require_once "database.php";
    $sql = "SELECT * FROM users WHERE username = '$username' && fullname = '$fullname'";
    $result = mysqli_query($conn, $sql);
    $total = mysqli_num_rows($result);
    $user = mysqli_fetch_array($result, MYSQLI_ASSOC);
    if ($user) {
        if (password_verify($password, $user["password"])) {
            session_start();
            $_SESSION['username'] = $username;
            $_SESSION['fullname'] = $fullname;
            $_SESSION["user"] = "yes";
            header("Location: index.php");
            die();
        } else{
            echo "<div class='alert alert-danger'>Password does not match</div>";
        }
    } else{
        echo "<div class='alert alert-danger'>User Name and FullName does not match</div>";
    }
}
```

Here a session will be started so that only the user can see their data. It will help to keep user's data secured.

LOGOUT:

```
<?php
session_start();
session_destroy();
header("Location: login.php");
?>
```

The created session will be destroyed here and the login page will be shown again.

SEARCH:

```
<?php

$servername = "localhost";
$username = "root";
$password = "";
$dbname = "bracuniversityblogger";

$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
if (isset($_POST["submit"])) {
    $search_username = $_POST["search"];
$search_username = mysqli_real_escape_string($conn, $search_username);
$sql = "SELECT * FROM users WHERE username = '$search_username'";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    $user_profile = $result->fetch_assoc();
    session_start();
    $_SESSION['username'] = $search_username;
    $_SESSION["user1"] = "yes";
    header("Location: friend-profile.php?user_id=" . $user_profile['user_id']);
    die();
    exit();
} else {
    echo "User not found";
}

$conn->close();
}
?>
```

Here username is the primary key that was taken from the users table and it is used to search a friend who already has an account.

IMAGE EDIT:

```
MariaDB [auth_db]> describe users;
+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| id    | int(11) | NO   | PRI | NULL    | auto_increment |
| fname | varchar(255) | NO  |     | NULL    | |
| username | varchar(255) | NO  |     | NULL    | |
| password | varchar(255) | NO  |     | NULL    | |
| pp    | varchar(255) | NO  |     | default-pp.png | |
+-----+-----+-----+-----+-----+
5 rows in set (0.008 sec)

$ sql = "UPDATE users
        SET fname=?, username=?, pp=?
        WHERE id=?";
$stmt = $conn->prepare($sql);
$stmt->execute([$fname, $uname, $new_img_name, $id]);
$_SESSION['fname'] = $fname;
header("Location: ../edit.php?success=Your account has been updated successfully");
exit;
} else {
    $em = "You can't upload files of this type";
    header("Location: ../edit.php?error=$em&$data");
    exit;
}
} else {
    $em = "unknown error occurred!";
    header("Location: ../edit.php?error=$em&$data");
    exit;
}

} else {
    $sql = "UPDATE users
        SET fname=?, username=?
        WHERE id=?";
$stmt = $conn->prepare($sql);
$stmt->execute([$fname, $uname, $id]);

header("Location: ../edit.php?success=Your account has been updated successfully");
exit;
```

Here users can choose an image file while creating an account or they can also update profile picture after creating a profile.

Conclusion

Building a blogging database for a university offers a thorough introduction to database structure, SQL operations, logins, and organizing files. It includes learning how to handle file uploads, create an effective relational structure, and implement security login for users. In the future, there may be ideas to improve blogging functionality, improve search, design for mobile, include data analysis, explore possibilities for group blogging, and think about advertising. The framework may be developed into a flexible and full of features application by ongoing iteration and user input. It may also be able to integrate with outside resources and act as a basis for future, more complicated projects.

References

- <https://www.w3schools.com/php/default.asp>
- <https://www.phptutorial.net/>
- <https://www.javatpoint.com/php-tutorial>
- <https://www.geeksforgeeks.org/php-tutorial/>
- <https://www.frontendmentor.io/>
- https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics
- [Tutorial Republic - Online Web Development Tutorials](#)
- https://www.w3schools.com/bootstrap/bootstrap_get_started.asp
- [Smashing Magazine — For Web Designers And Developers](#)
- <https://stylestage.dev/>