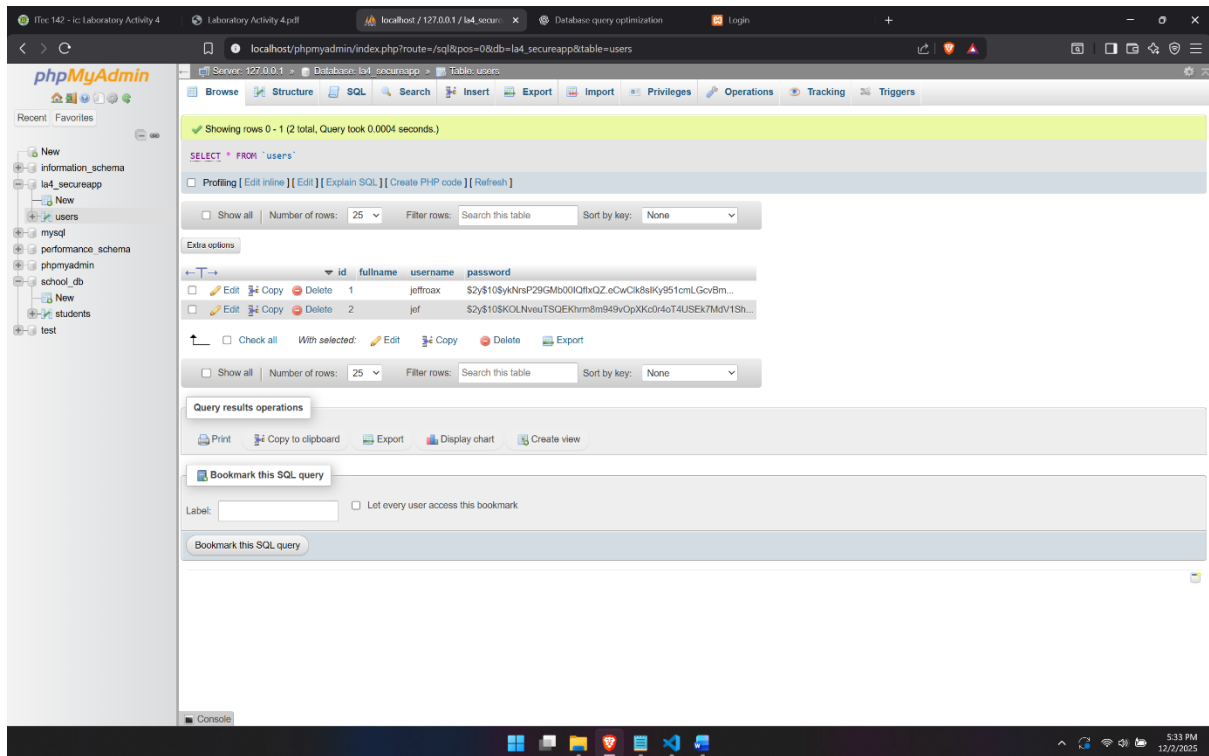
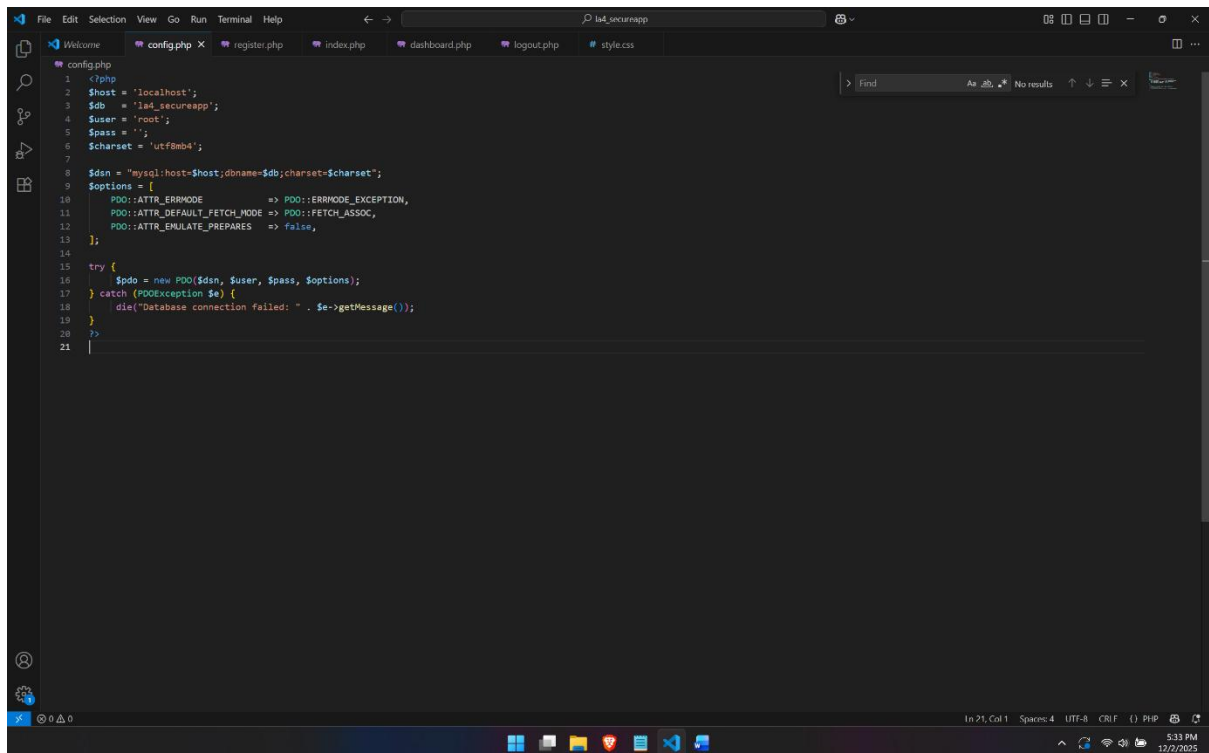


Name:	Jef Martin D. Ortega	Date:	12/02/2025
Course:	ITEC142 - Web Systems and Technologies 2		

La4_secureapp database and users table created

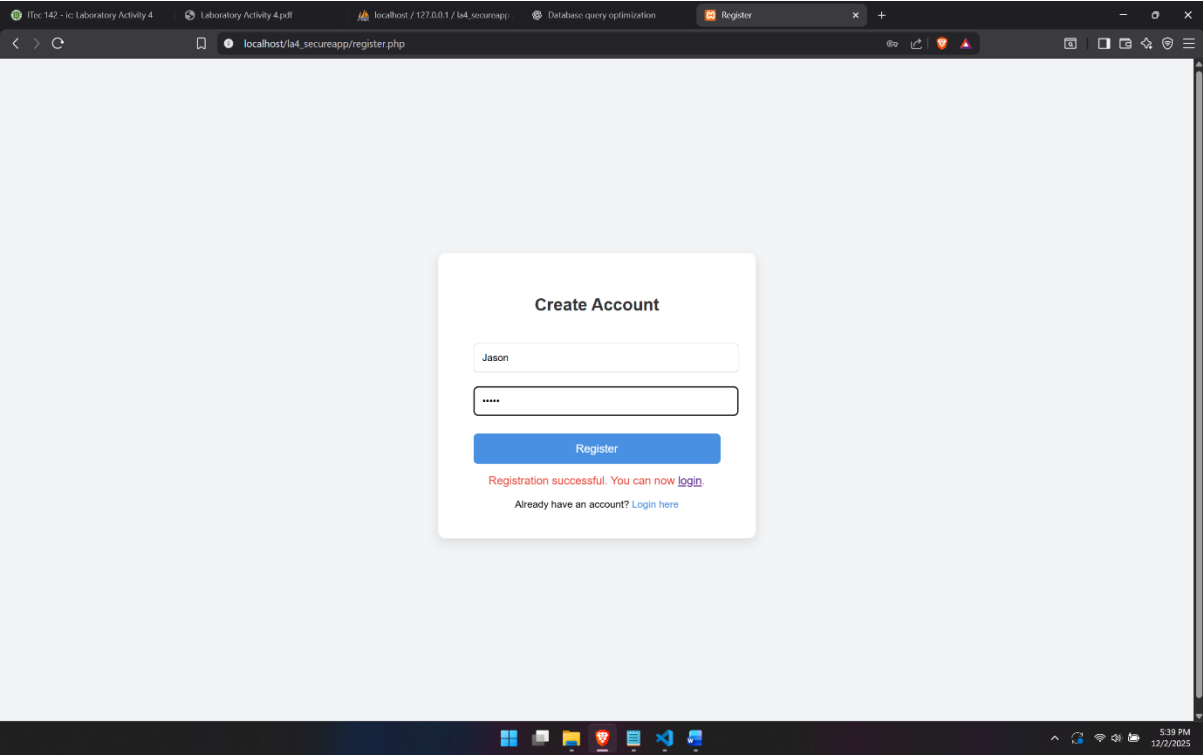


Code editor showing config.php

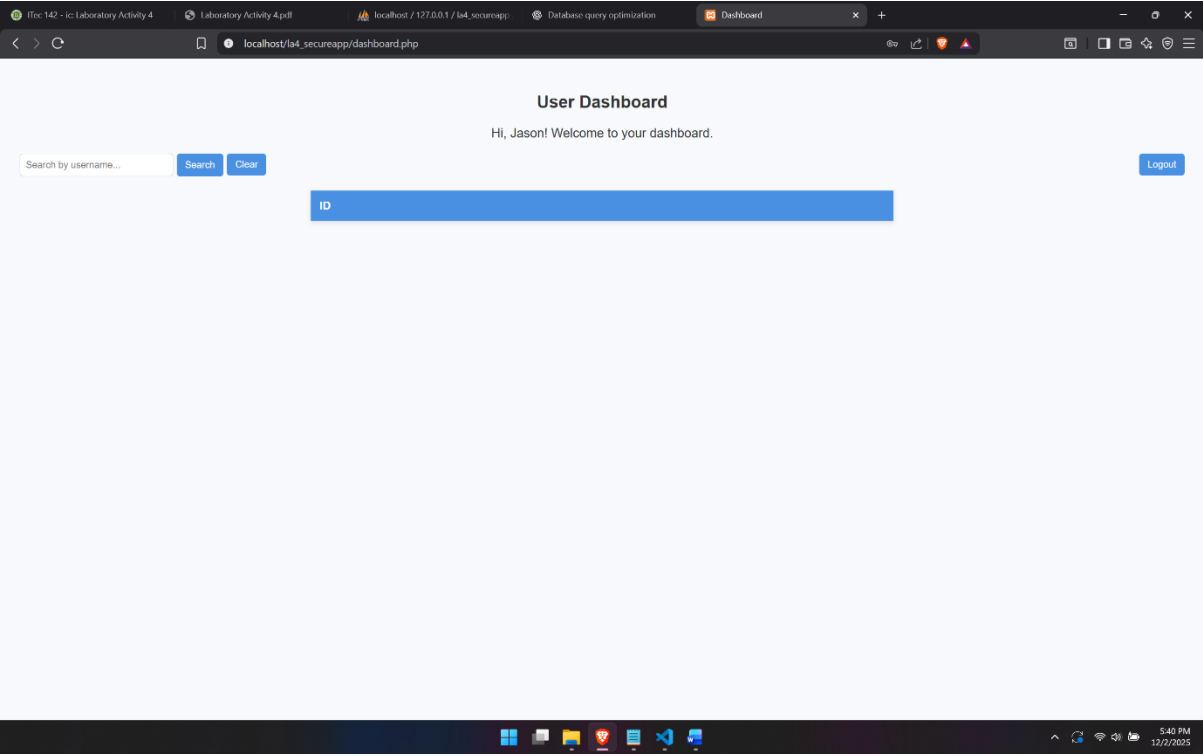


Name:	Jef Martin D. Ortega	Date:	12/02/2025
Course:	ITEC142 - Web Systems and Technologies 2		

Successful Registration message

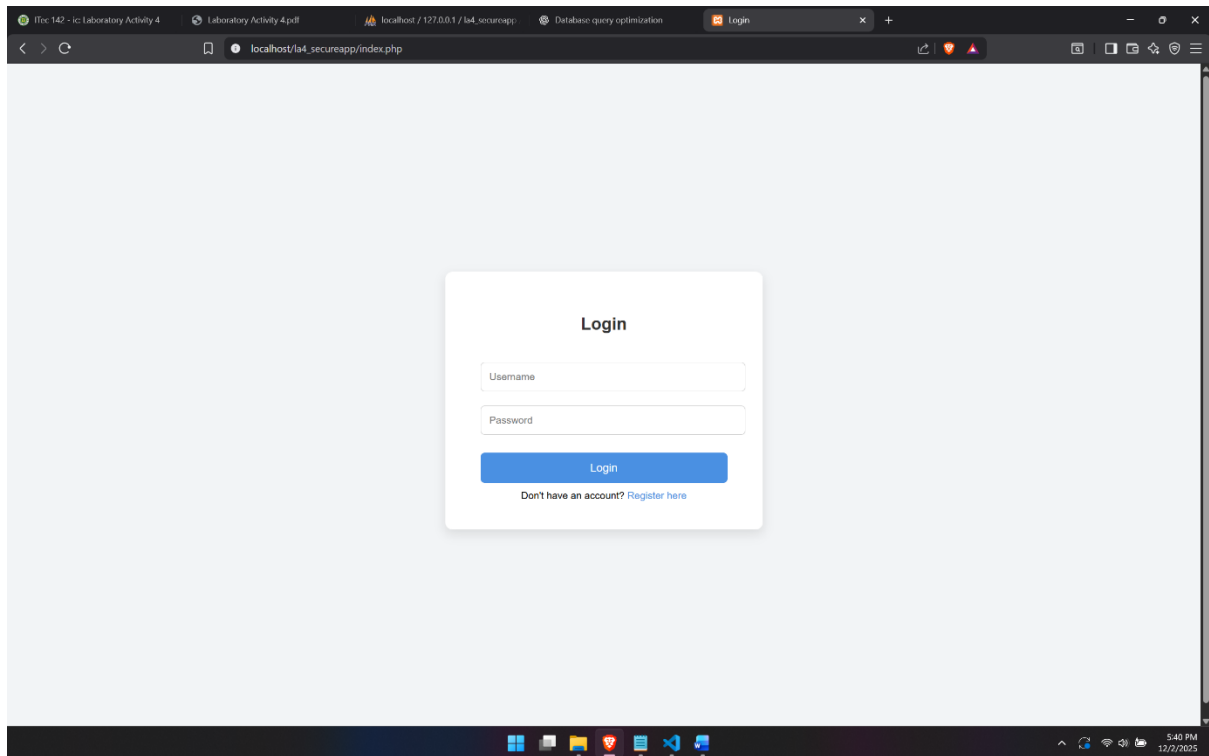


Dashboard logged-in view



Name:	Jef Martin D. Ortega	Date:	12/02/2025
Course:	ITEC142 - Web Systems and Technologies 2		

Logout successful and returned to login



Password_hash

```
<?php
session_start();
require 'config.php';

$message = '';
if (isset($_POST['register'])) {
    $username = trim($_POST['username']);
    $password = password_hash($_POST['password'], PASSWORD_DEFAULT);

    $stmt = $pdo->prepare("INSERT INTO users (username, password) VALUES (?, ?)");
    try {
        $stmt->execute([$username, $password]);
        $message = "Registration successful. You can now <a href='index.php'>login</a>.";
    } catch (PDOException $e) {
        $message = "Username already exists!";
    }
}
```

Name:	Jef Martin D. Ortega	Date:	12/02/2025
Course:	ITEC142 - Web Systems and Technologies 2		

Prepare ()

```

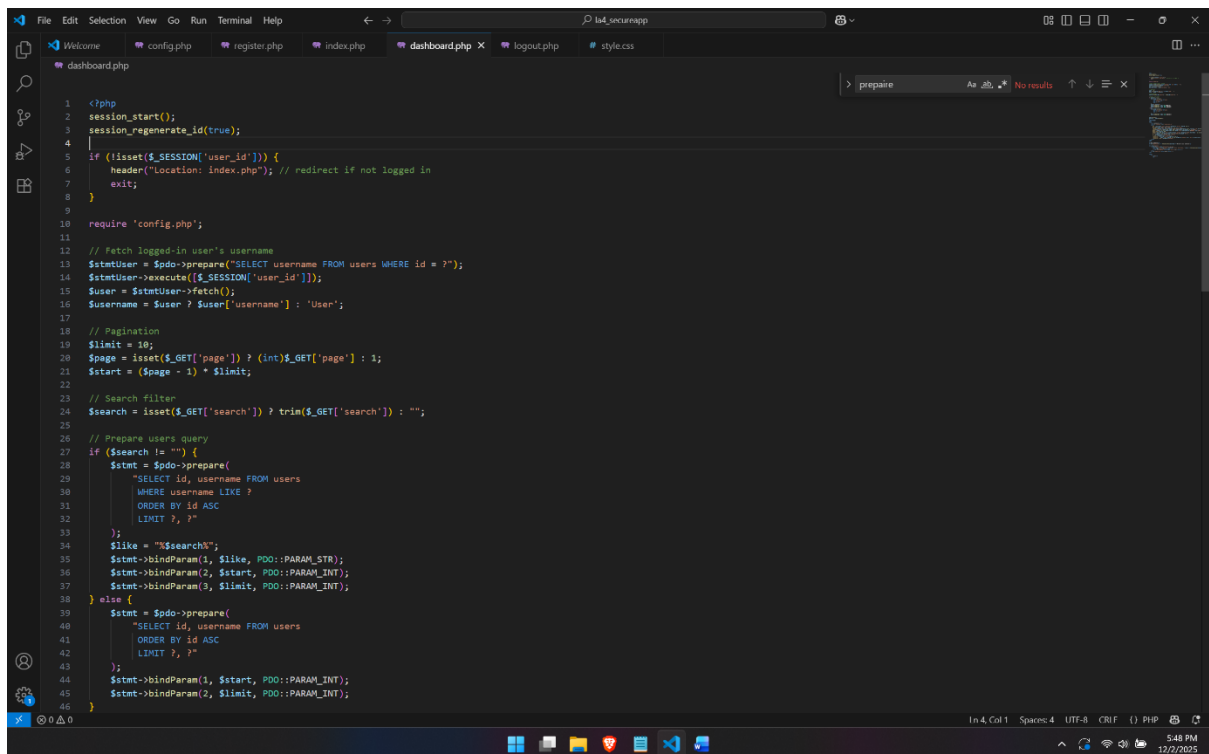
$message = '';
if (isset($_POST['login'])) {
    $username = trim($_POST['username']);
    $password = $_POST['password'];

    $stmt = $pdo->prepare("SELECT * FROM users WHERE username = ?");
    $stmt->execute([$username]);
    $user = $stmt->fetch();

    if ($user && password_verify($password, $user['password'])) {
        session_start();
        session_regenerate_id(true); // Generate new session ID and delete the old one
        $_SESSION['user_id'] = $user['id'];
        header("Location: dashboard.php");
        exit;
    } else {
        $message = "Invalid username or password.";
    }
}
?>

```

Session_start() and session_regenerate_id()



```

1  <?php
2  session_start();
3  session_regenerate_id(true);
4  }
5  if (isset($_SESSION['user_id'])) {
6      header("Location: index.php"); // redirect if not logged in
7      exit;
8  }
9
10 require 'config.php';
11
12 // Fetch logged-in user's username
13 $stmtUser = $pdo->prepare("SELECT username FROM users WHERE id = ?");
14 $stmtUser->execute([$_SESSION['user_id']]);
15 $user = $stmtUser->fetch();
16 $username = $user ? $user['username'] : 'User';
17
18 // Pagination
19 $limit = 10;
20 $page = isset($_GET['page']) ? (int)$_GET['page'] : 1;
21 $start = ($page - 1) * $limit;
22
23 // Search filter
24 $search = isset($_GET['search']) ? trim($_GET['search']) : '';
25
26 // Prepare users query
27 if ($search != '') {
28     $stmt = $pdo->prepare(
29         "SELECT id, username FROM users
30         WHERE username LIKE ?
31         ORDER BY id ASC
32         LIMIT ?, ?"
33     );
34     $like = "%$search%";
35     $stmt->bindParam(1, $like, PDO::PARAM_STR);
36     $stmt->bindParam(2, $start, PDO::PARAM_INT);
37     $stmt->bindParam(3, $limit, PDO::PARAM_INT);
38 } else {
39     $stmt = $pdo->prepare(
40         "SELECT id, username FROM users
41         ORDER BY id ASC
42         LIMIT ?, ?"
43     );
44     $stmt->bindParam(1, $start, PDO::PARAM_INT);
45     $stmt->bindParam(2, $limit, PDO::PARAM_INT);
46 }

```

Config.php

Name:	Jef Martin D. Ortega	Date:	12/02/2025
Course:	ITEC142 - Web Systems and Technologies 2		

Database:

Users table stores unique usernames and hashed passwords.

Register:

Collects username and password, hashes it, and saves securely.

Login:

Verifies credentials, starts session, regenerates session ID, and redirects to dashboard.

Session Security:

Only logged-in users access dashboard; session ID prevents fixation.

Dashboard:

Greets user, shows users list with search and pagination, includes logout button.

Logout:

Destroys session and redirects to login.

Security:

Uses password hashing, prepared statements, input validation, and session checks.