Contents

[Table of figures 1](#_Toc163343839)

[My Project 2](#_Toc163343840)

[Login form 2](#_Toc163343841)

[Title and Description: 2](#_Toc163343842)

[Purpose 2](#_Toc163343843)

[Components and Fields: 2](#_Toc163343844)

[Styling: 4](#_Toc163343845)

[Validation Rules: 5](#_Toc163343846)

[User Interactions: 6](#_Toc163343847)

[PHP Script Documentation: 6](#_Toc163343848)

[Components: 7](#_Toc163343849)

[Database Connection: 7](#_Toc163343850)

[Form Submission Handling: 7](#_Toc163343851)

[Password Hashing: 7](#_Toc163343852)

[Database Insertion: 7](#_Toc163343853)

[Error Handling: 8](#_Toc163343854)

[Error Handling: 8](#_Toc163343855)

[Visual Design: 8](#_Toc163343856)

[Integration and Security: 8](#_Toc163343857)

[Accessibility Considerations: 8](#_Toc163343858)

[Testing Guidelines: 9](#_Toc163343859)

[Version History and Updates: 9](#_Toc163343860)

# Table of figures

[Figure 1.loginform 2](#_Toc163343748)

[Figure 2.Username 3](#_Toc163343749)

[Figure 3.Password 3](#_Toc163343750)

[Figure 4.remember me 4](#_Toc163343751)

[Figure 5.forgot password 4](#_Toc163343752)

[Figure 6.login 4](#_Toc163343753)

[Figure 7.register account 4](#_Toc163343754)

[Figure 8sytle.css 5](#_Toc163343755)

[Figure 9.Validation rules 5](#_Toc163343756)

[Figure 10.interactions 6](#_Toc163343757)

[Figure 11.loginp.php 7](#_Toc163343758)

# My Project

Login form  
Documentation is essential for maintaining clarity and ensuring effective communication among team members during development.

### Title and Description:

Modern login form

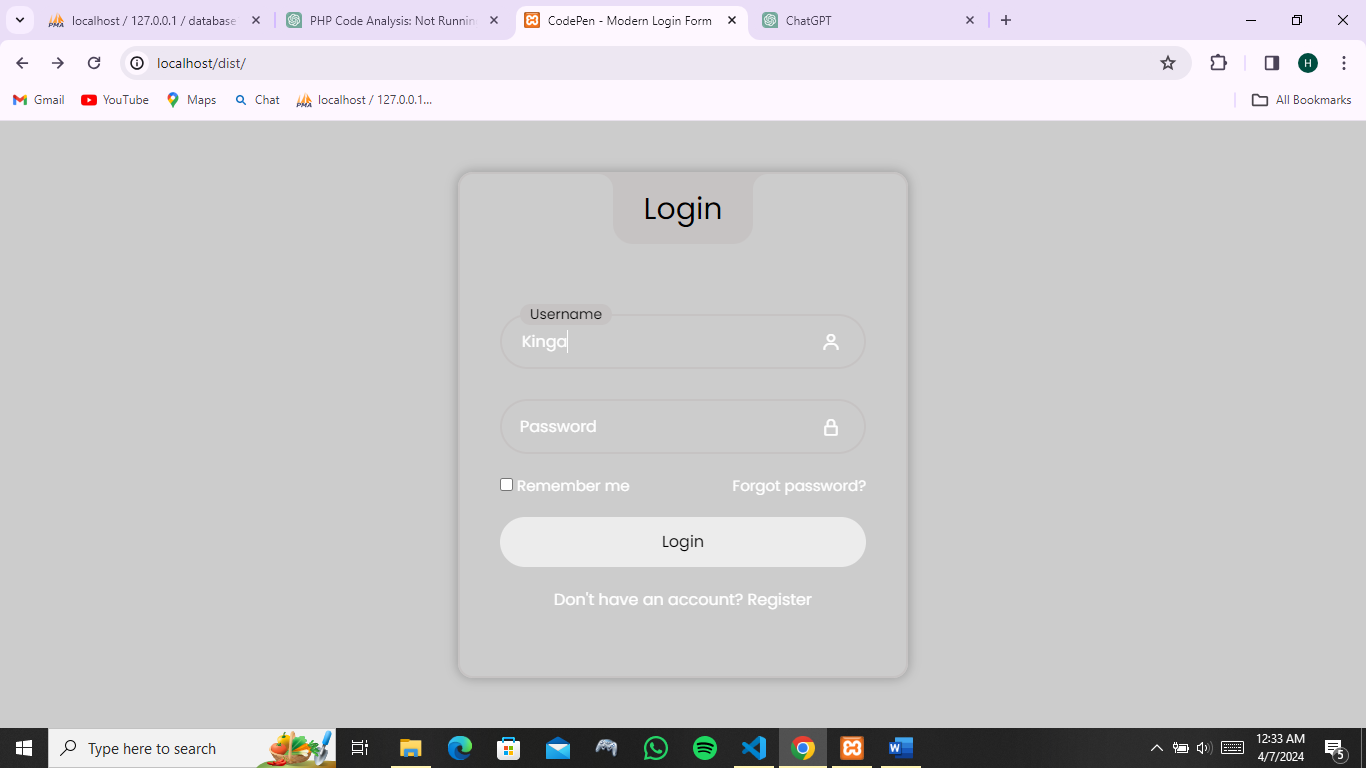


Figure 1.loginform

### Purpose

The form is designed to capture login credentials from users.

### Components and Fields:

**Username Input Field:**

Type: Text

ID: user

Name: user

Class: input-field

Required: Yes

Description: Allows users to input their username.

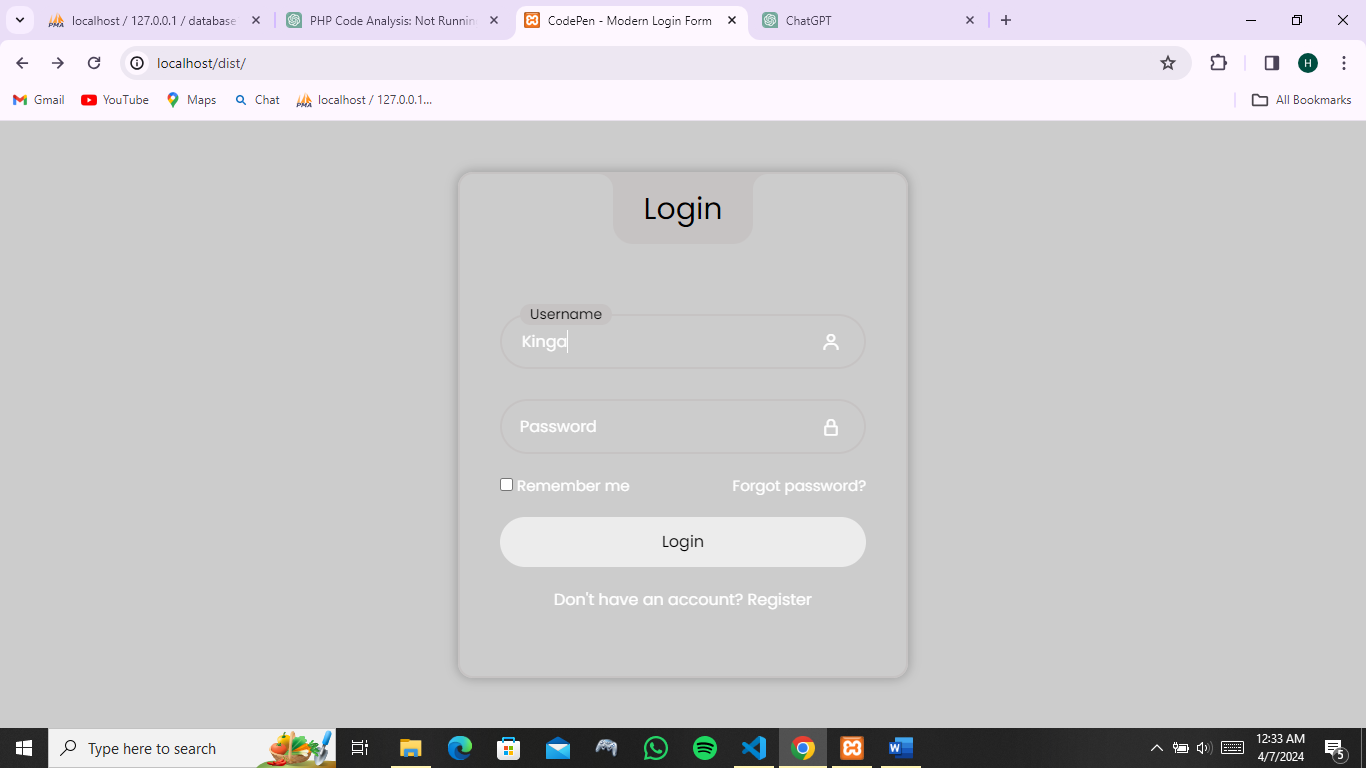


Figure 2.Username

**Password Input Field:**

Type: Password

ID: pass

Name: pass

Class: input-field

Required: Yes

Description: Allows users to input their password.

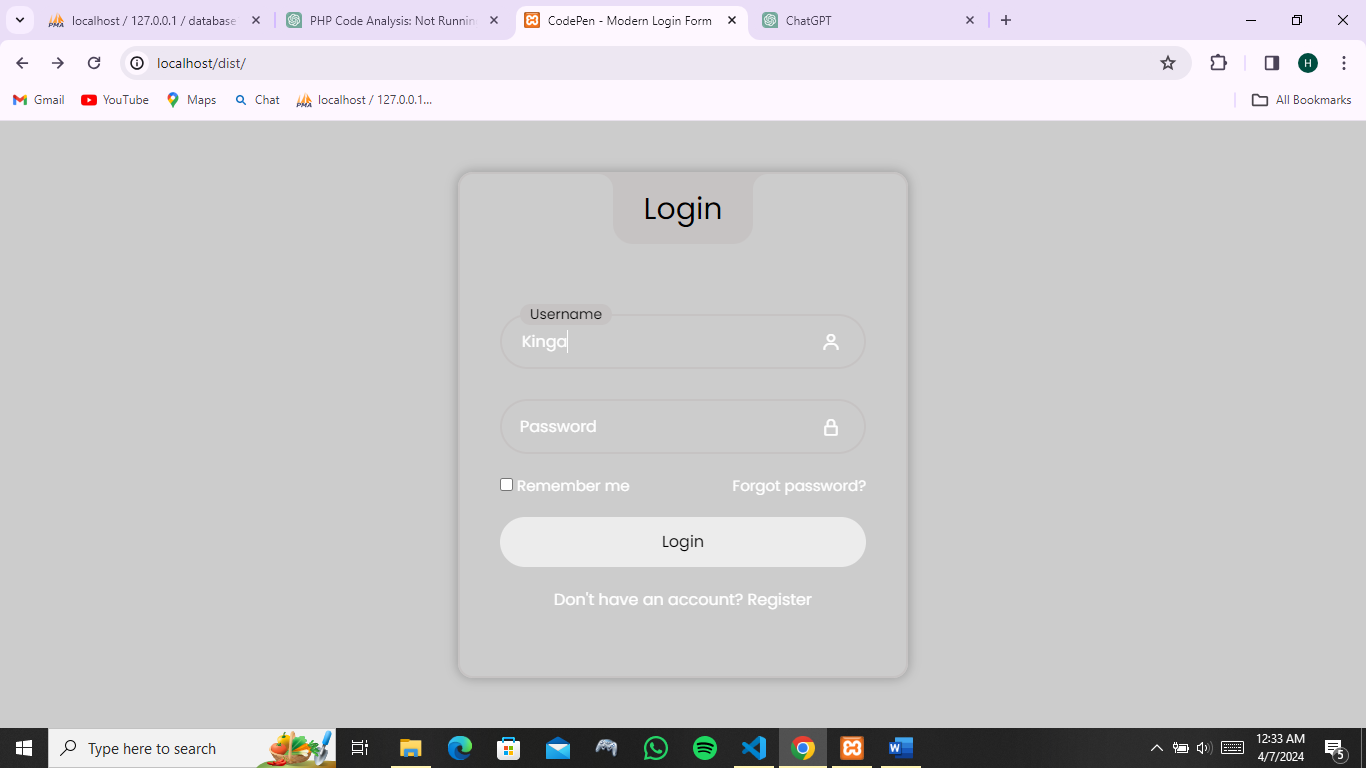


Figure 3.Password

**Remember Me Checkbox:**

Type: Checkbox

ID: remember

Description: Allows users to indicate whether they want to be remembered.

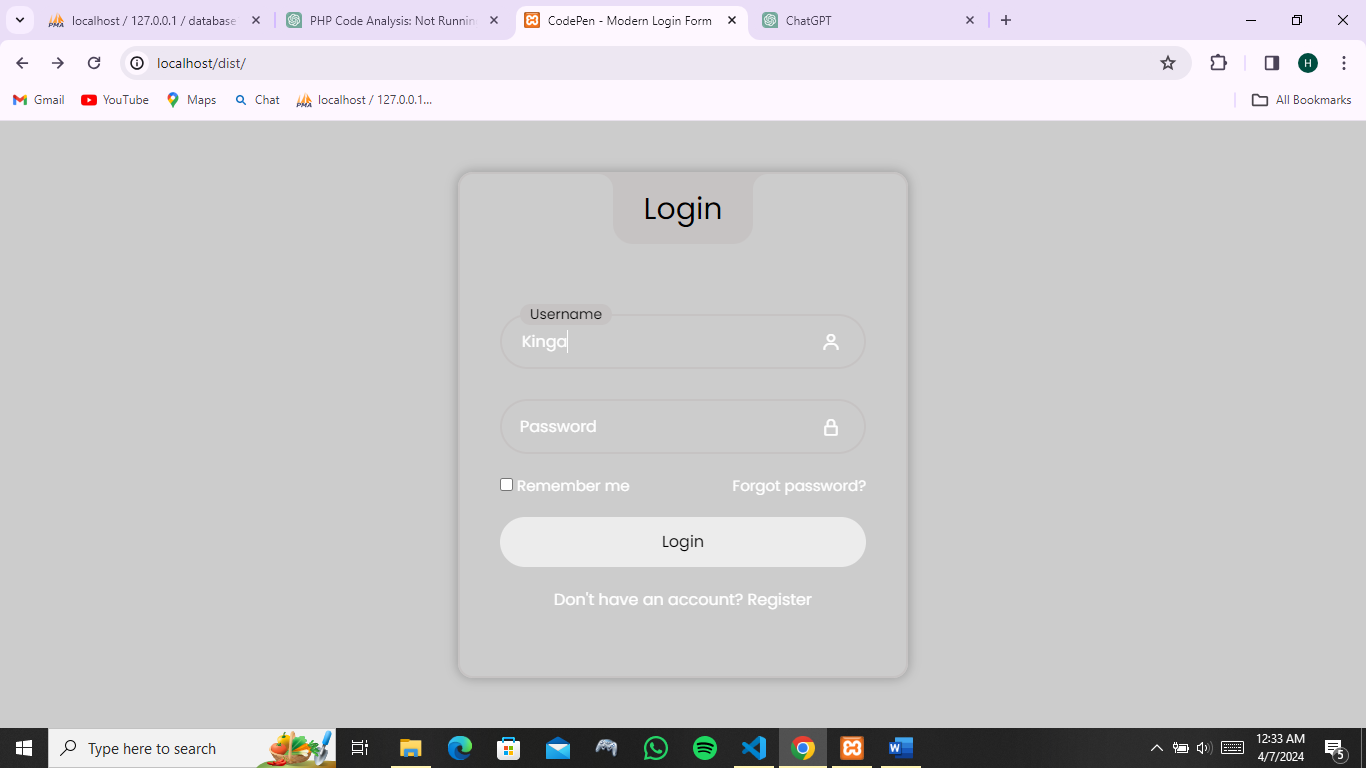


Figure 4.remember me

**Forgot Password Link:**

Description: Provides a link for users to reset their password if forgotten.

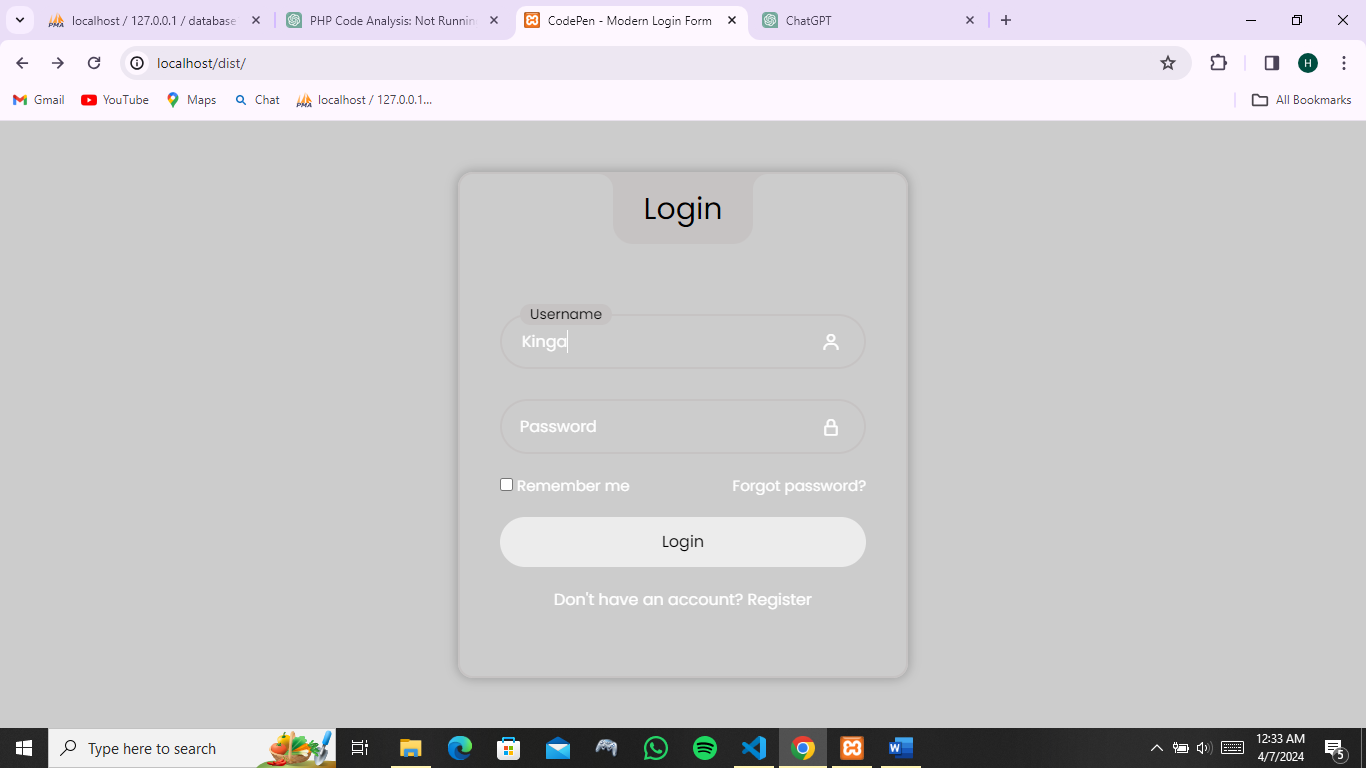


Figure 5.forgot password

**Login Button:**

Type: Submit

Value: Login

Description: Submits the form data to the server for login authentication.

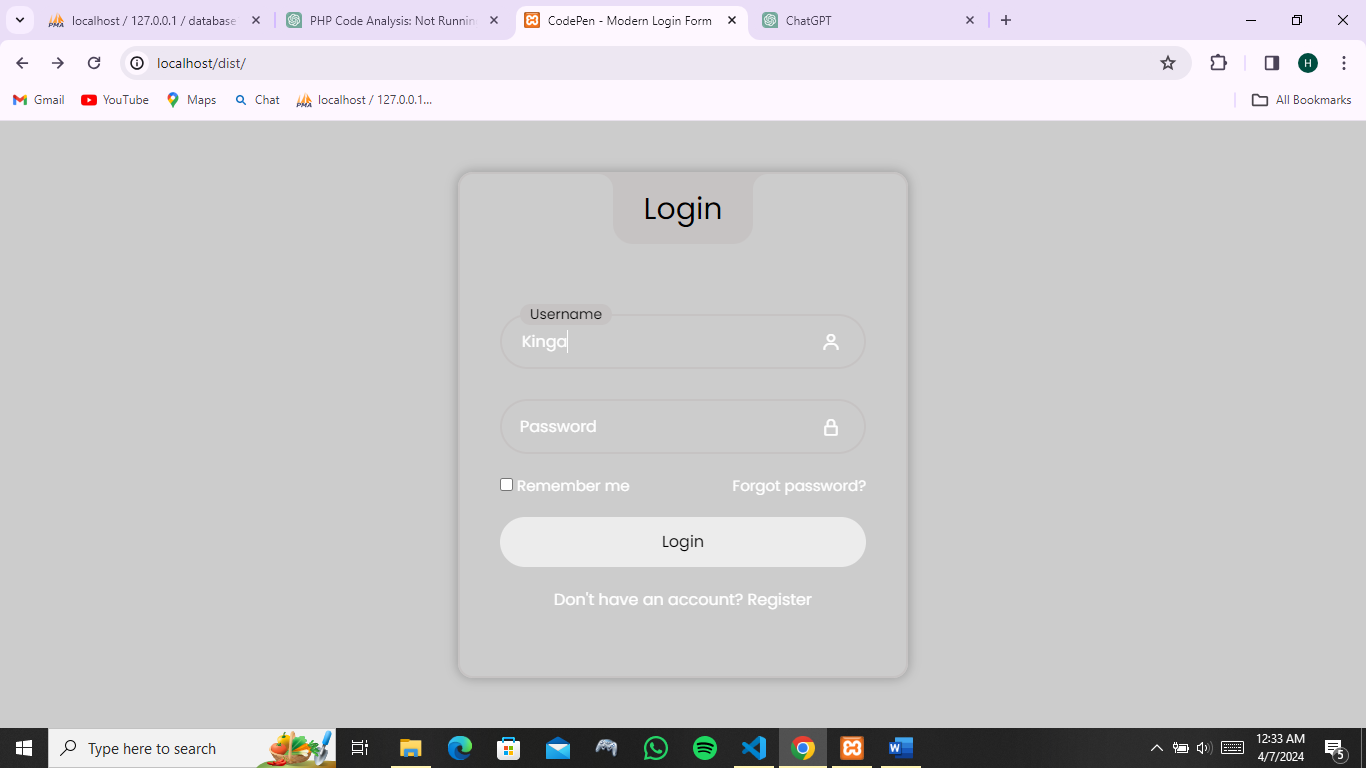


Figure 6.login

**Register Link:**

Description: Provides a link for users to register for a new account.

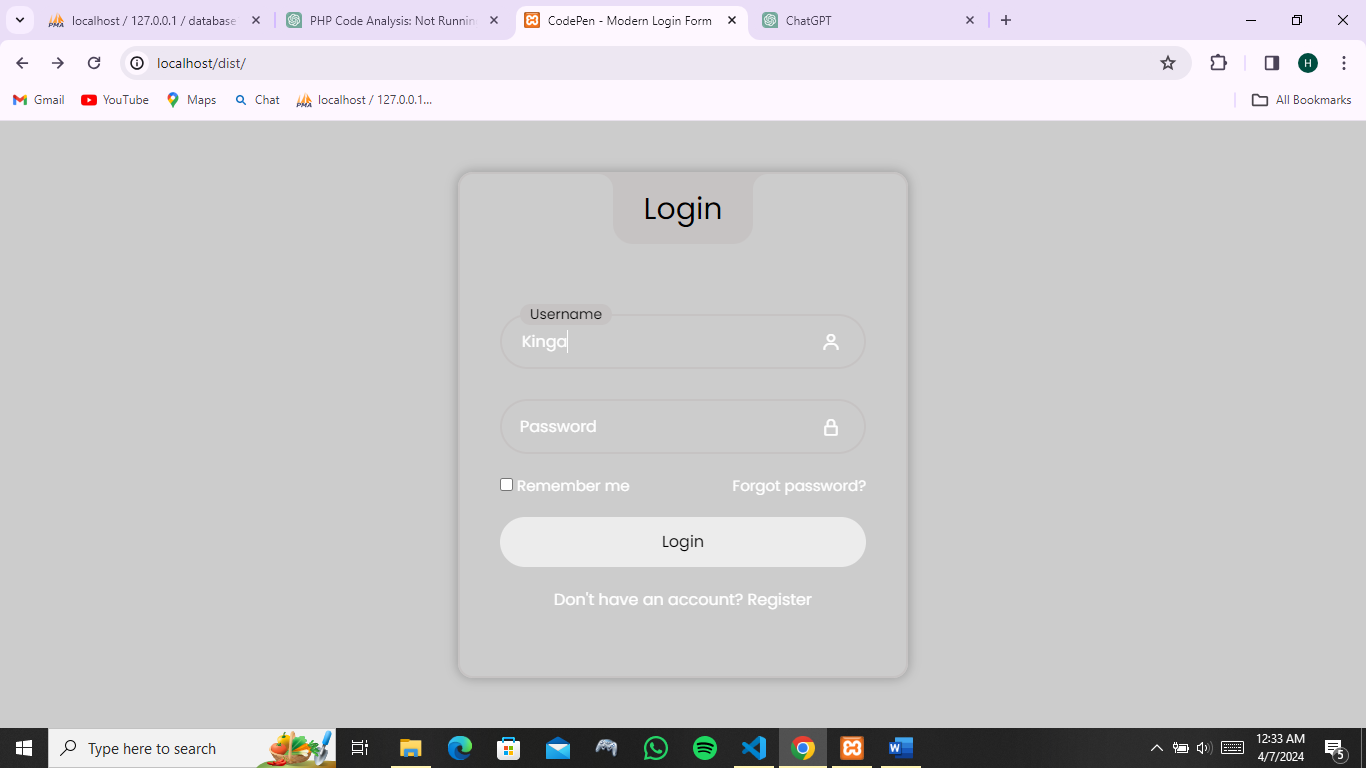


Figure 7.register account

### Styling:

The form utilizes CSS classes for styling purposes. These classes are not detailed in the documentation but can be found in the associated stylesheet (style.css).

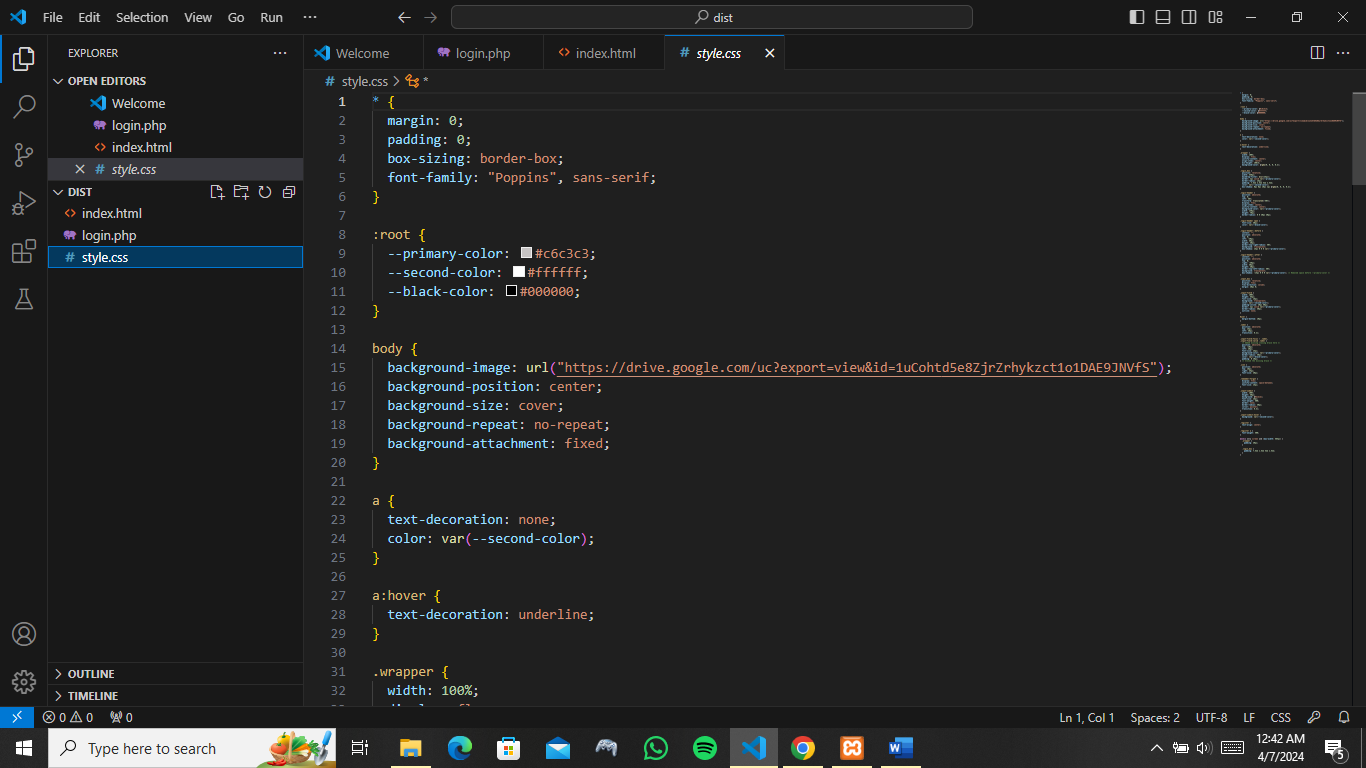


Figure 8sytle.css

### Validation Rules:

Document the validation rules applied to each field.

The required attribute ensures that users must fill in both the username and password fields before submitting

Username/Email: Required, must be a valid email format.

Password: Required, minimum length of 8 characters, must include at least one uppercase letter, one lowercase letter, one number, and one special character.

Error messages that will be displayed if the user input doesn't meet these rules.

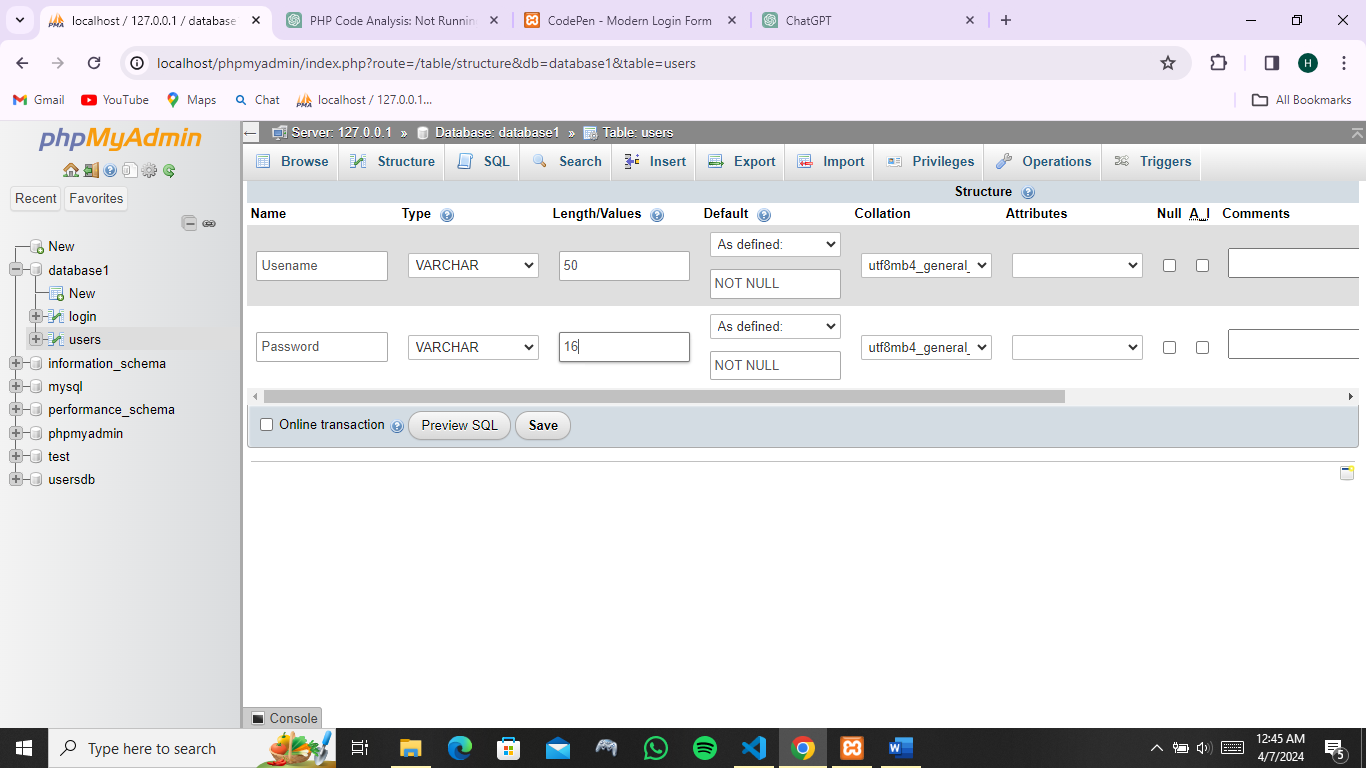


Figure 9.Validation rules

# User Interactions:

Entering username/email and password: Users are required to fill in the required fields correctly

Clicking the login button to submit the form: The login button submits the filled in credentials to the msqldatabase

Behavior when incorrect credentials are entered: An error message is displayed

Actions triggered by clicking "Remember Me" or "Forgot Password" options: Your password is saved for easier login and when you can’t remember the password

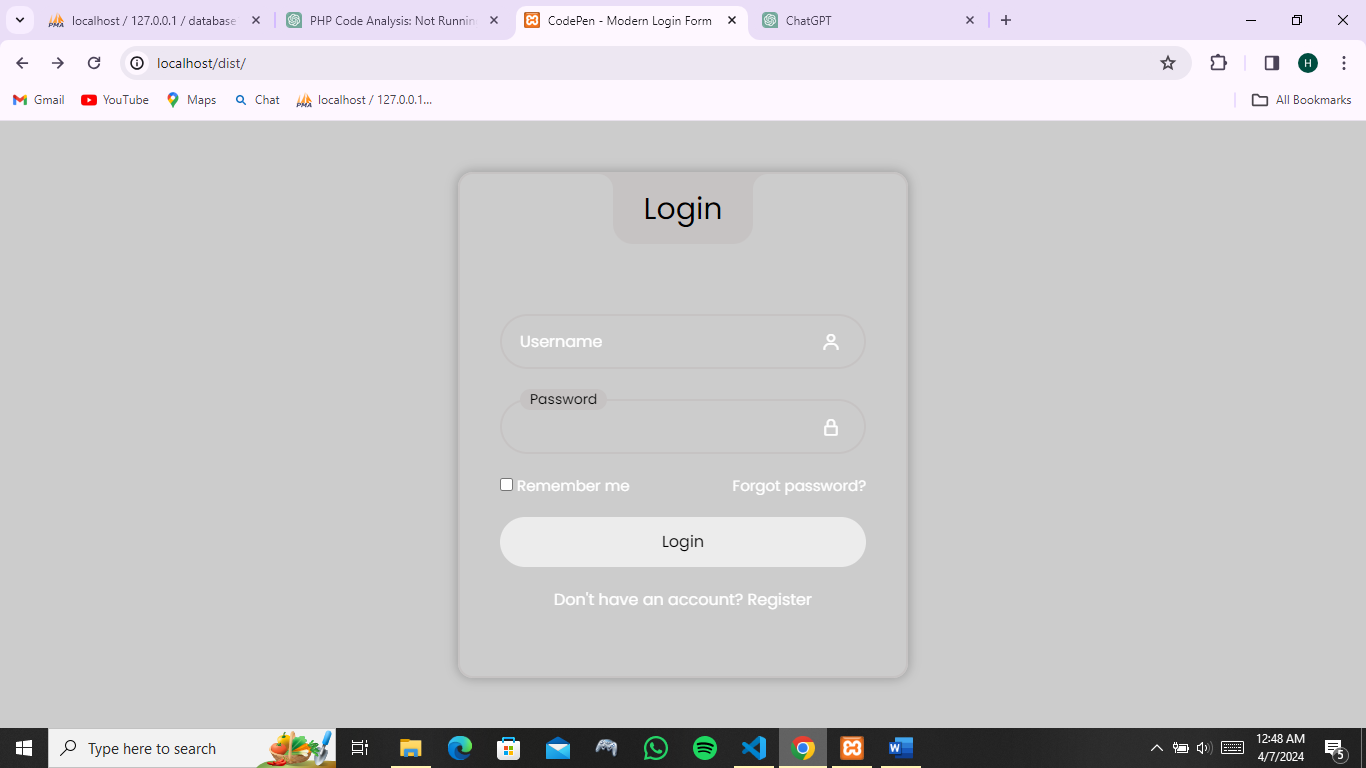


Figure 10.interactions

# PHP Script Documentation:

**Purpose:**

The PHP script login.php is responsible for processing the login form data and inserting user details into a MySQL database.

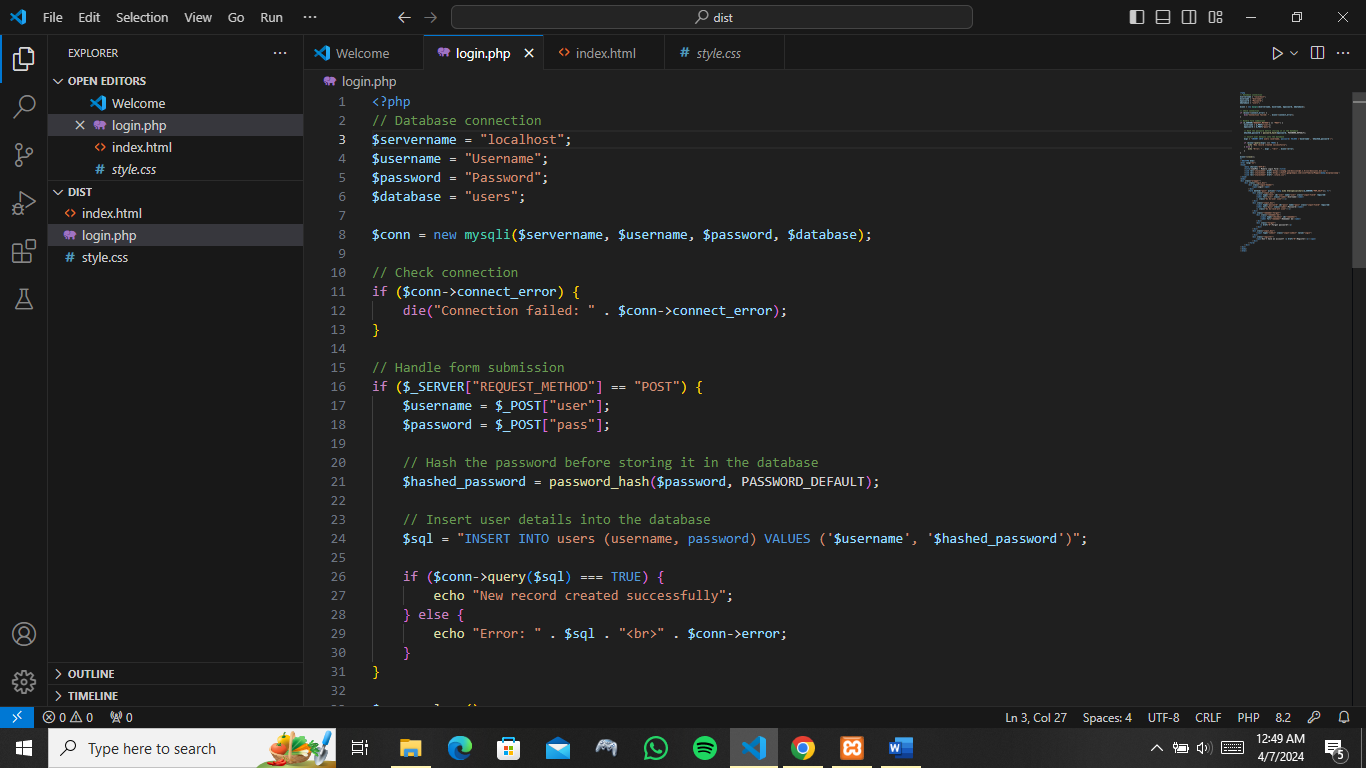


Figure 11.loginp.php

## Components:

### Database Connection:

Establishes a connection to the MySQL database using provided credentials ($servername, $username, $password, $database).

### Form Submission Handling:

Checks if the form has been submitted using $\_SERVER["REQUEST\_METHOD"] == "POST".

Retrieves the username and password from the POST data.

### Password Hashing:

Hashes the password using PHP's password\_hash function for secure storage in the database.

### Database Insertion:

Constructs an SQL query to insert the username and hashed password into a table named users in the database.

### Error Handling:

Checks if the database query was executed successfully. If successful, echoes a success message. Otherwise, echoes an error message with details about the SQL query and the error encountered.

This documentation provides an overview of the purpose and components of the HTML form and PHP script. It can serve as a reference for developers working with the provided code.

# Error Handling:

Document how errors are handled in the login form. This includes:

Displaying error messages for invalid input.

Handling server-side validation errors.

Handling authentication failures (e.g., incorrect username/password).

# Visual Design:

If applicable, include screenshots or wireframes illustrating the visual design of the login form. Highlight important elements such as input fields, buttons, and error messages.

# Integration and Security:

If the login form interacts with external systems (e.g., authentication server), document the integration details.

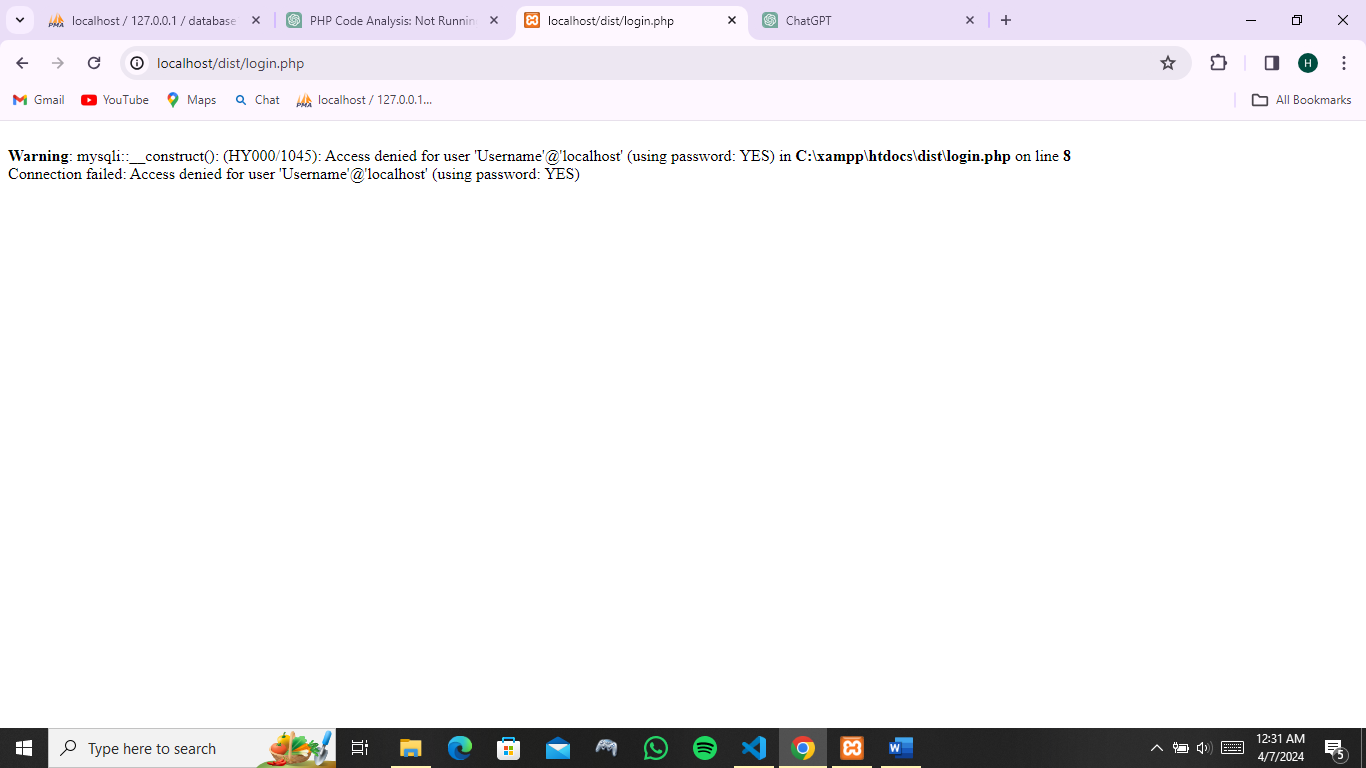
Highlight any security measures implemented, such as encryption of passwords or protection against brute-force attacks.

# Accessibility Considerations:

Ensure the login form complies with accessibility standards (e.g., WCAG) to accommodate users with disabilities. Document any specific accessibility features implemented.

# Testing Guidelines:

Provide guidelines for testing the login form, including test cases for various scenarios (valid credentials, invalid credentials, error handling, etc.).



# Version History and Updates:

Maintain a version history of the document to track changes and updates made to the login form over time.