PRN: 21070126039   Name: Jainil Patel   Batch: AI/ML A2

Full Stack Developments - LAB 4

Interactive Web App: Password Protection & Input Validation

Problem:

Design a web-based application that enhances user interactivity by implementing a password-protected activation mechanism for input fields. The application prompts users to enter a password to activate textboxes, subsequently validating the format of mobile numbers and email addresses entered by the user.

- The application should present users with a user-friendly interface containing input fields for mobile number, email address.
- Upon accessing the application, users are prompted to enter a password to activate the input fields. The password serves as a security measure to control access to the form.
- After successfully activating the textboxes by entering the correct password, users can input their mobile number and email address.
- The application must validate the format of the mobile number entered by users, ensuring it consists of 10 digits. Similarly, the application must validate the format of the email address entered by users, ensuring it follows the standard email format (e.g., user@example.com).
- If the format of the mobile number or email address entered by the user is incorrect, the application should display an alert with guidance on the correct format.

```
Use // Regular expression to validate mobile number (10 digits)
var mobileRegex = /^\d{10}$/;
// Check mobile number format
if (!mobileRegex.test(mobileNumber)) {
    alert("Mobile number is not correct.");
    return;
}
// Regular expression to validate email
var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
// Check email format
if (!emailRegex.test(email)) {
    alert("Email is not correct.\nEmail must be in the format: user@example.com");
    return;
```

Finally, when the user clicks the submit button, a confirmation box should appear, allowing users to confirm their submission. Upon confirmation, the application can proceed with further processing or display a success message accordingly.

This problem statement aims to develop a user-friendly and interactive web application that enforces security through password protection while ensuring data integrity by validating user input formats.

HTML Tags used:

1. **<html>: Defines the root element of an HTML document.**

```
<html>
    <!-- HTML content goes here -->
</html>
```

2. **<head>: Contains meta-information about the HTML document, such as title, links to stylesheets, and scripts.**

```
<head>
    <title>My Page Title</title>
    <!-- Other meta-information goes here -->
</head>
```

3. **<title>: Sets the title of the HTML document displayed in the browser's title bar or tab.**

```
<title>My Page Title</title>
```

4. **<style>: Defines CSS styles that apply to elements within the document.**

```
<style>
    body {
        font-family: Arial, sans-serif;
    }
</style>
```

5. **<body>: Contains the content of the HTML document that is displayed in the browser window.**

```
<body>
    <h1>Hello, World!</h1>
    <p>This is a paragraph.</p>
</body>
```

6. **<script>: Defines client-side JavaScript code within the HTML document, allowing for dynamic behavior and interaction.**

```
<script>
    .
    .
    .
</script>
```

7. **<form>: Defines an HTML form for user input. It can contain input elements like text fields, checkboxes, radio buttons, submit buttons, and more.**

```
<form id="validationForm">
    .
    .
    .
</form>
```

8. **<input type="button”> : Defines a clickable button that can trigger JavaScript functions**

```
<input type="button" value="Activate Textboxes" onclick="activateTextboxes()" >
```

9. **<input type="submit”>: Defines a submit button for submitting form data to a server.**

```
<input type="submit" value="Submit" onclick="submitForm(event)">
```

Code:

```
<html>
    <head>
        <title>Lab 4</title>


        <style>
            body {
                font-family: Arial, sans-serif;
                display: flex;

            }

            label {
                font-weight: bold;
            }

        </style>
    </head>
    <body>

        <form id="validationForm">
```

```
            <h1>Form Validation Activation</h1>

            <label for = "mobile"> Mobile Number :</label>
            <input type="text" name="mobile" id="mobile" placeholder="Enter your mobile number"
disabled>

            <br>
            <br>

            <label for = "email"> Email :</label>
            <input type="email" name="email" id="email" placeholder="Enter your email address"
disabled>

            <br>
            <br>
            <br>
            <br>

            <input type="button" value="Activate Textboxes" onclick="activateTextboxes()" >



            <input type="submit" value="Submit" onclick="submitForm(event)">

        </form>

        <script>

            function activateTextboxes() {

                // Prompt for password
                var password = prompt('Enter password:');
                if (password !== '1234') {
                    alert('Incorrect password!');
                    return;
                }

                // Enable textboxes

                document.getElementById('mobile').disabled = false;
                document.getElementById('email').disabled = false;
            }

            function submitForm(event) {
                event.preventDefault(); // Prevent form submission

                var mobileNumber = document.getElementById('mobile').value;
                var email = document.getElementById('email').value;

                // Check for empty fields
                if (mobileNumber.trim() === '' || email.trim() === '') {
                    alert('Mobile number and email are required fields.');
                    return;
                }

                var mobileRegex = /^\d{10}$/;
                var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

                if (!mobileRegex.test(mobileNumber)) {
                    alert('Mobile number is not correct.');
                    return;
                }


                // Check for valid email
                if (!emailRegex.test(email)) {
                    alert('Email is not correct.\nEmail must be in the format: 
user@example.com');
                    return;
                }

                // If everything is fine, you can submit the form
                alert('Form submitted successfully!');
                document.getElementById('validationForm').submit();
            }

        </script>
    </body>
</html>
```

CSS Code:

```
<style>
            body {
                font-family: Arial, sans-serif;
                display: flex;

            }

            label {
                font-weight: bold;
            }

        </style>
```

Script Code:

```
<script>
            function activateTextboxes() {
                // Prompt for password
                var password = prompt('Enter password:');
                if (password !== '1234') {
                    alert('Incorrect password!');
                    return;
                }

                // Enable textboxes

                document.getElementById('mobile').disabled = false;
                document.getElementById('email').disabled = false;
            }

            function submitForm(event) {
                event.preventDefault(); // Prevent form submission

                var mobileNumber = document.getElementById('mobile').value;
                var email = document.getElementById('email').value;

                // Check for empty fields
                if (mobileNumber.trim() === '' || email.trim() === '') {
                    alert('Mobile number and email are required fields.');
                    return;
                }

                var mobileRegex = /^\d{10}$/;
                var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

                if (!mobileRegex.test(mobileNumber)) {
                    alert('Mobile number is not correct.');
                    return;
                }
                // Check for valid email
                if (!emailRegex.test(email)) {
                    alert('Email is not correct.\nEmail must be in the format:
user@example.com');
                    return;
                }
                // If everything is fine, you can submit the form
                alert('Form submitted successfully!');
                document.getElementById('validationForm').submit();
            }

</script>
```

activateTextboxes():

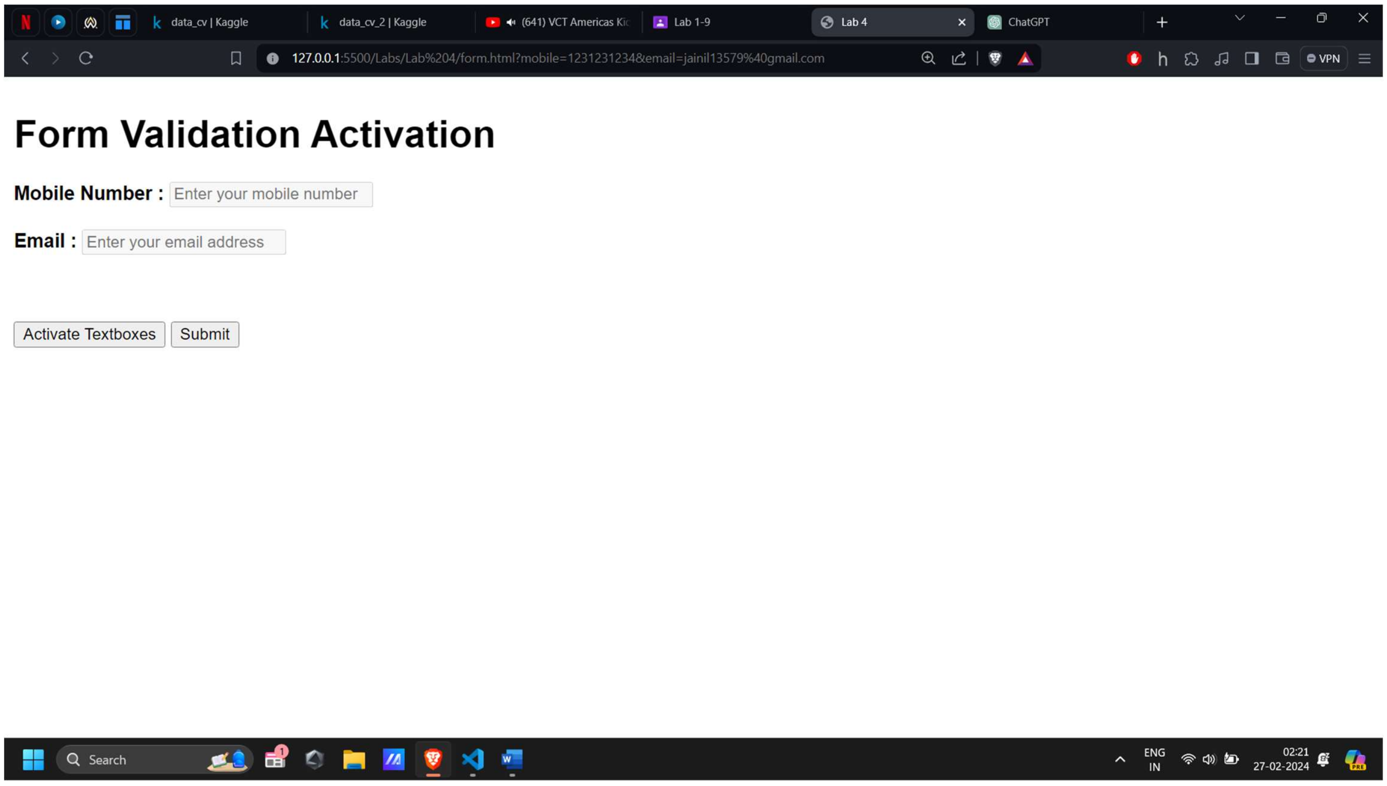
- This function is triggered when the user clicks the "Activate Textboxes" button.
- It prompts the user to enter a password using the prompt() function.
- If the entered password is not '1234', it displays an alert indicating "Incorrect password!" and returns, preventing further action.
- If the password is correct, it enables the textboxes for mobile number and email address by setting their disabled attribute to false using document.getElementById().

submitForm(event):

- This function is triggered when the user submits the form.
- It prevents the default form submission behavior using event.preventDefault() to allow custom validation before submission.
- It retrieves the values entered by the user in the mobile number and email fields.
- It checks if either of the fields is empty, displaying an alert if they are.
- It defines regular expressions to validate the format of the mobile number and email address.
- It tests the user-input values against the defined regular expressions.
- If the mobile number format is incorrect, it displays an alert stating, "Mobile number is not correct."
- If the email format is incorrect, it displays an alert stating "Email is not correct. Email must be in the format: user@example.com".
- If both mobile number and email format are correct, it displays an alert "Form submitted successfully!" and submits the form using document.getElementById('validationForm').submit().

This JavaScript code enhances the functionality of the web application by enforcing password protection, enabling input fields based on correct password entry, and validating user input before form submission to ensure data integrity and format consistency.

Output:

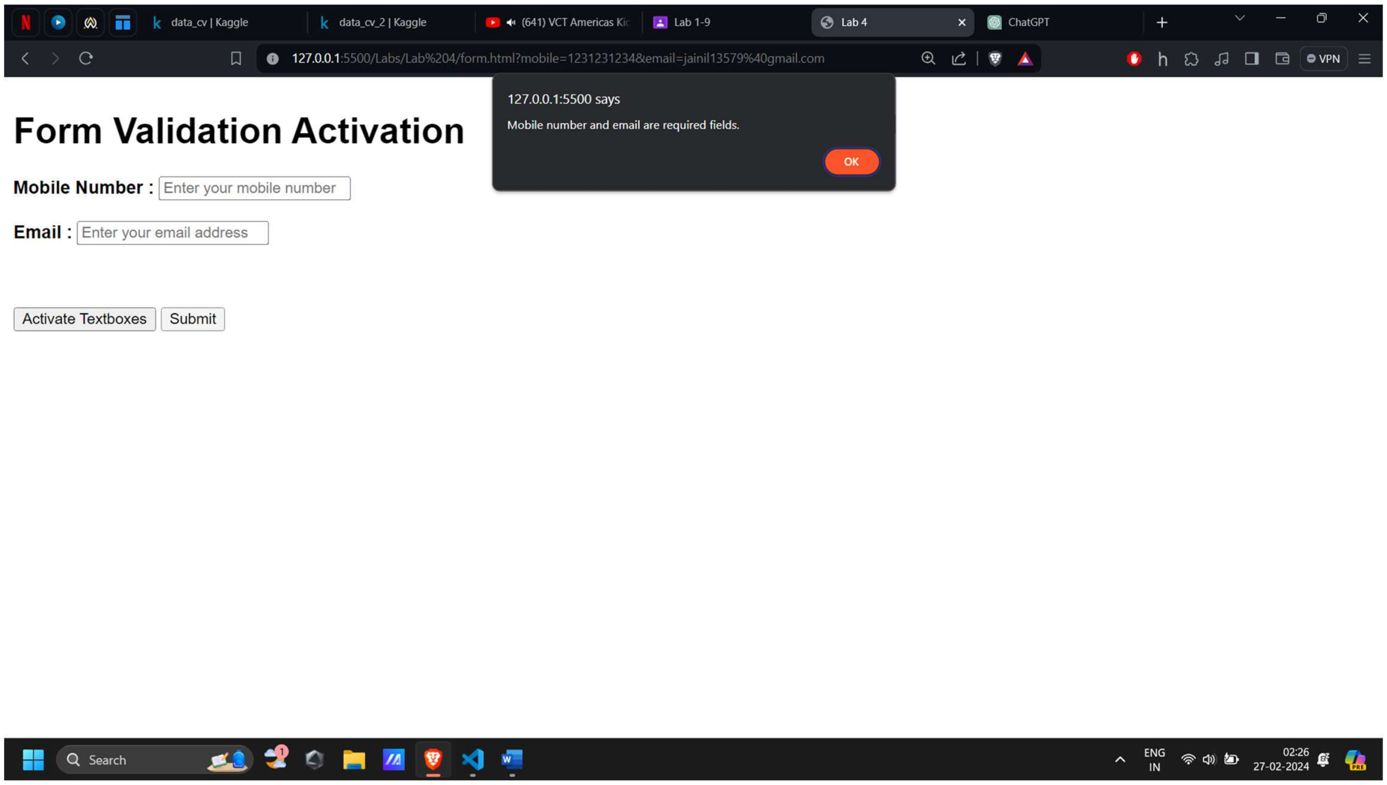


127.0.0.1:5500/Labs/Lab%204/form.html?mobile=1231231234&email=jainil13579%40gmail.com

Form Validation Activation

Mobile Number : Enter your mobile number

Email : Enter your email address



127.0.0.1:5500/Labs/Lab%204/form.html?mobile=1231231234&email=jainil13579%40gmail.com

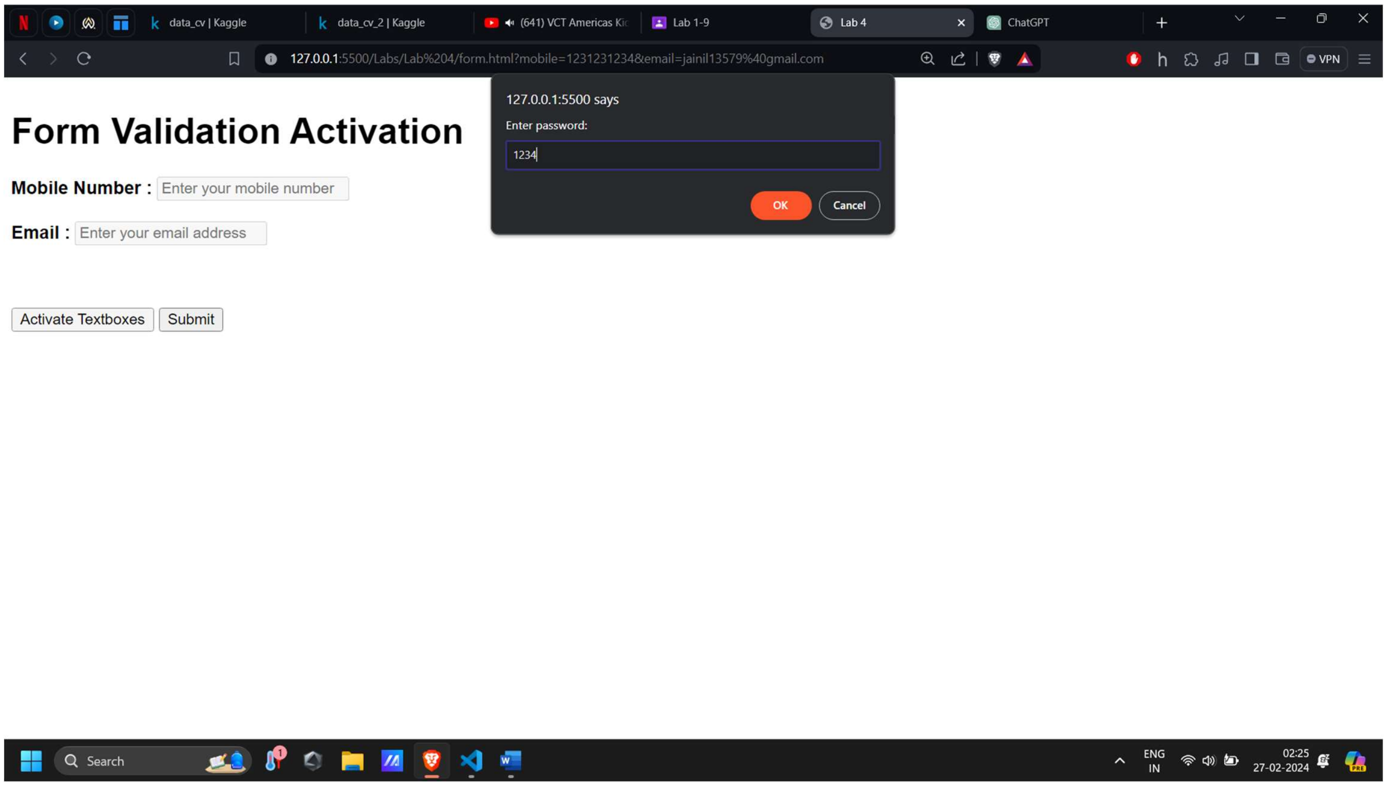
127.0.0.1:5500 says

Mobile number and email are required fields.

Form Validation Activation

Mobile Number : Enter your mobile number

Email : Enter your email address



127.0.0.1:5500/Labs/Lab%204/form.html?mobile=1231231234&email=jainil13579%40gmail.com

127.0.0.1:5500 says

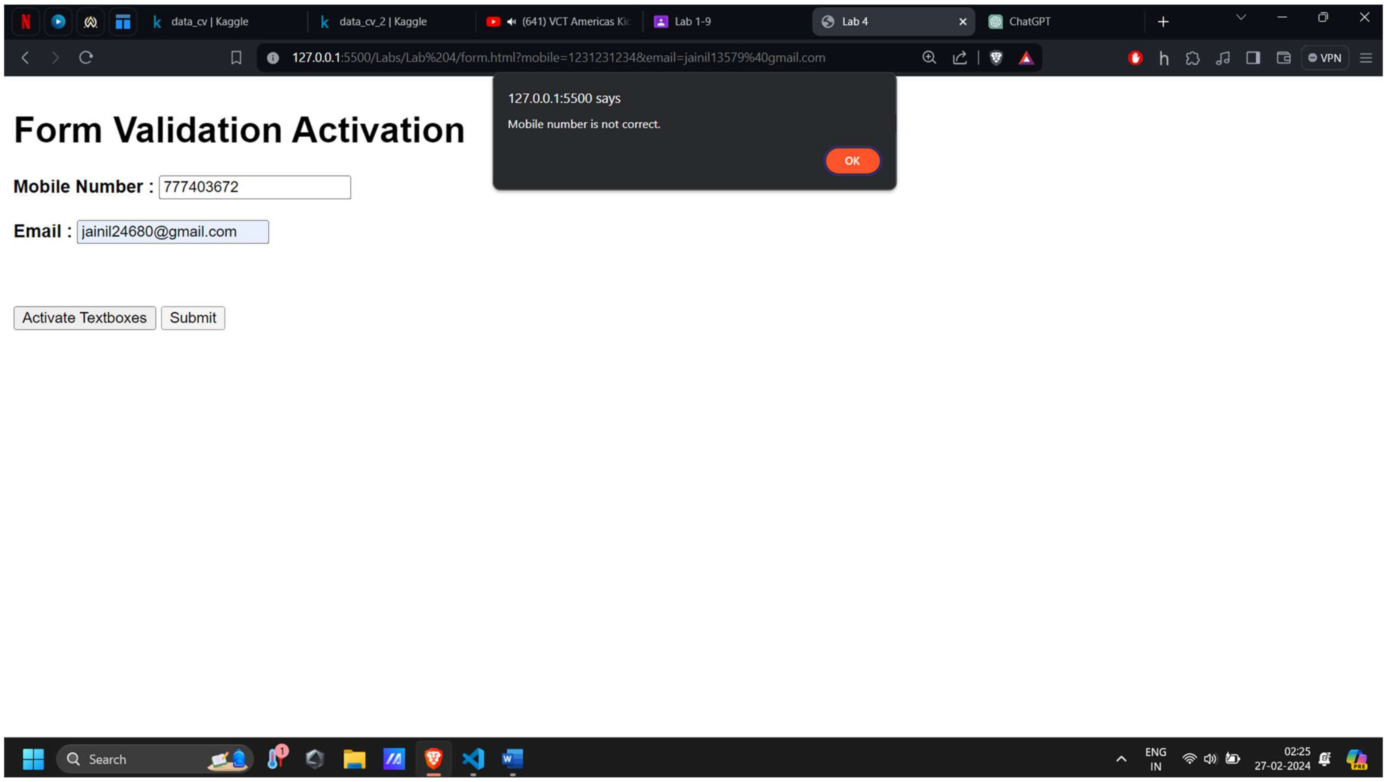
Enter password:

1234

Form Validation Activation

Mobile Number : Enter your mobile number

Email : Enter your email address



127.0.0.1:5500/Labs/Lab%204/form.html?mobile=1231231234&email=jainil13579%40gmail.com

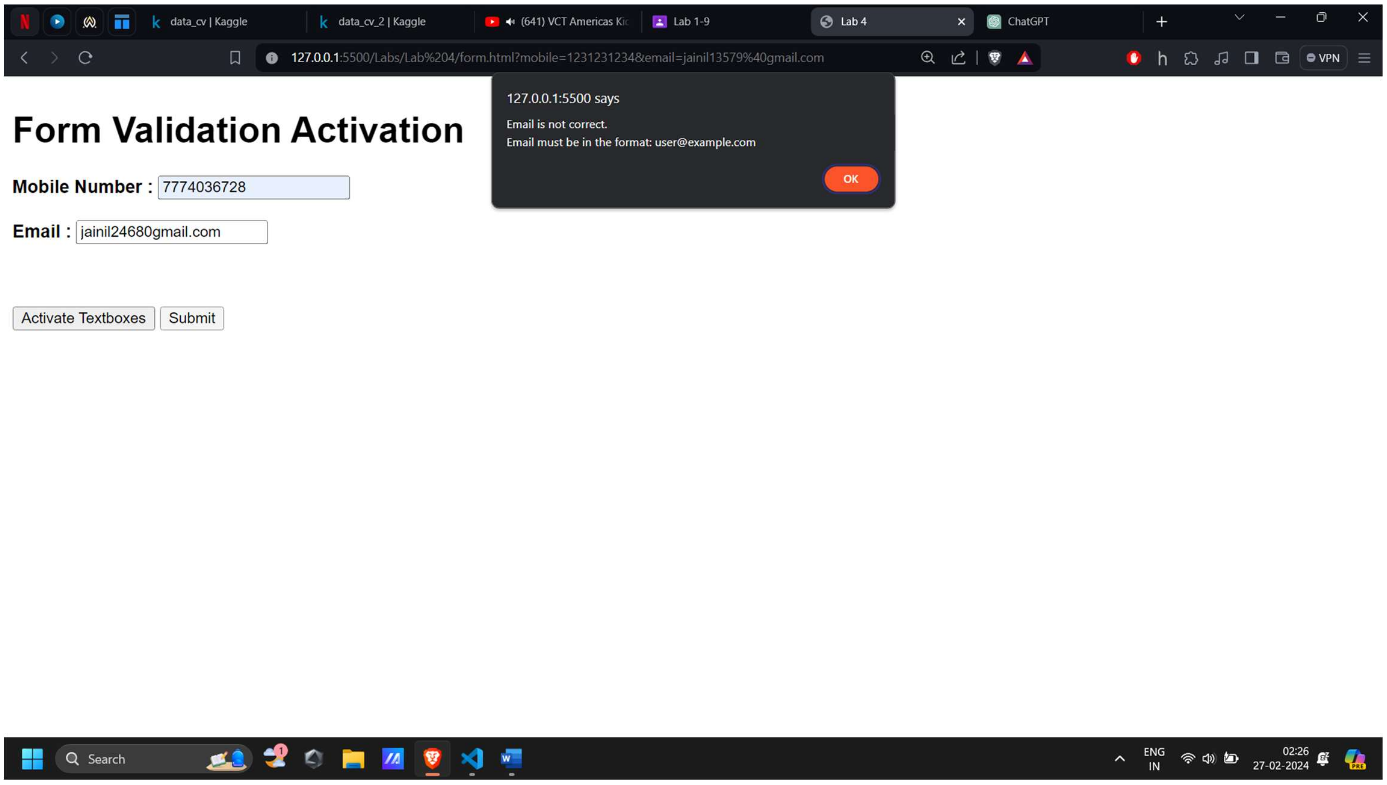
127.0.0.1:5500 says

Mobile number is not correct.

Form Validation Activation

Mobile Number : 777403672

Email : jainil24680@gmail.com



127.0.0.1:5500/Labs/Lab%204/form.html?mobile=1231231234&email=jainil13579%40gmail.com

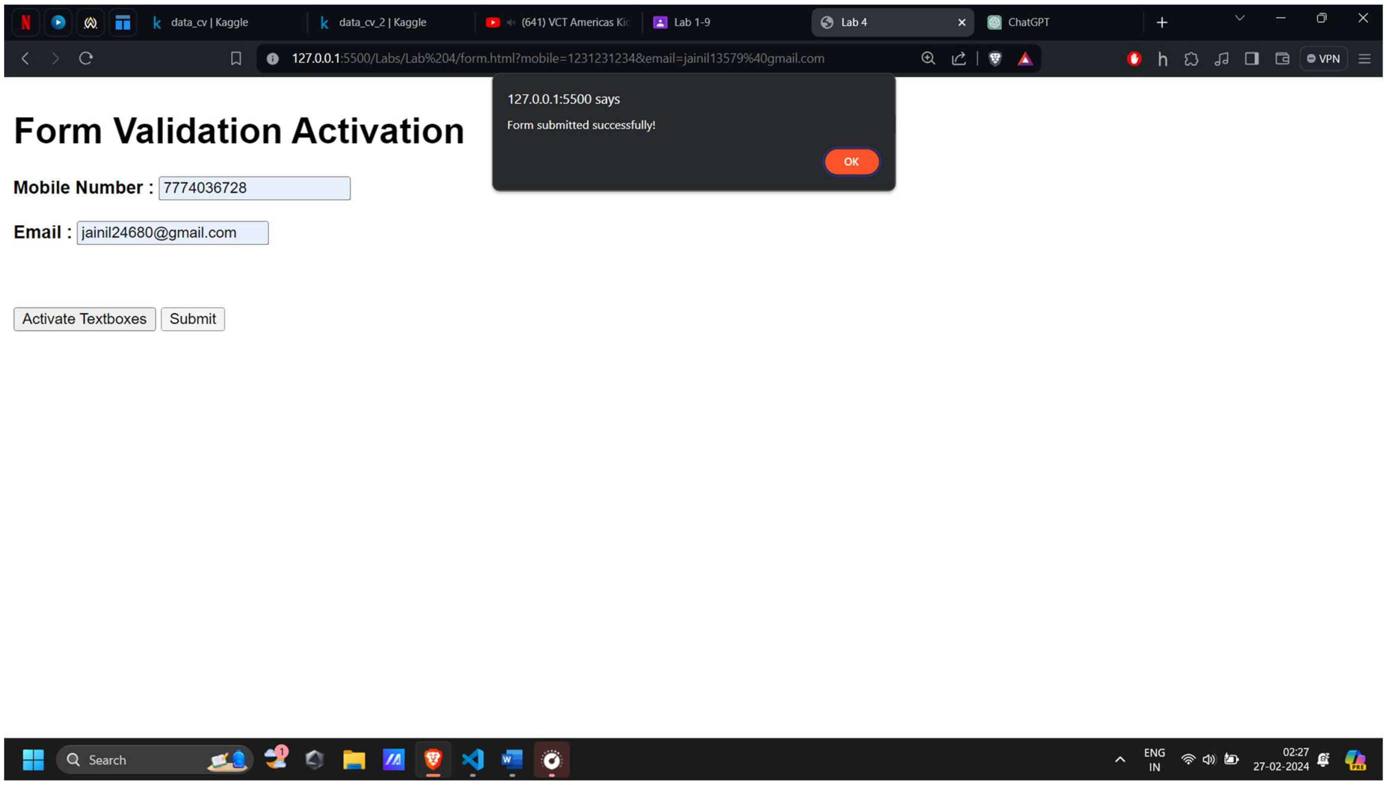
127.0.0.1:5500 says

Email is not correct.
Email must be in the format: user@example.com

Form Validation Activation

Mobile Number : 7774036728

Email : jainil24680gmail.com



127.0.0.1:5500 says

Form submitted successfully!

Form Validation Activation

Mobile Number : 7774036728

Email : jainil24680@gmail.com

Activate Textboxes | Submit