**Exercise 1: Sum of Two Numbers**

**Task**: Write a JavaScript function sumTwoNumbers(a, b) that takes two numbers as arguments and returns their sum. Call the function with different sets of numbers and log the result.

**Exercise 2: Factorial Function**

**Task**: Write a function factorial(n) that takes a number n as an argument and returns its factorial. Use recursion to solve this.

**Exercise 3: Check if a String is a Palindrome**

**Task**: Write a function isPalindrome(str) that checks if a given string is a palindrome (reads the same forward and backward).

**Exercise 4: Find the Maximum Number in an Array**

**Task**: Write a function findMax(arr) that takes an array of numbers and returns the maximum value.

**Exercise 5: Convert Temperature from Celsius to Fahrenheit**

**Task**: Write a function convertToFahrenheit(celsius) that takes a temperature in Celsius and converts it to Fahrenheit using the formula: F = C \* 9/5 + 32.

**Exercise 6: Reverse a String**

**Task**: Write a function reverseString(str) that takes a string and returns it reversed.

**Exercise 7: Sum of Array Elements**

**Task**: Write a function sumArray(arr) that takes an array of numbers and returns the sum of all its elements.

**Exercise 8: Check Prime Number**

**Task**: Write a function isPrime(num) that takes a number and returns true if the number is prime, and false otherwise.

**Exercise 9: Count Vowels in a String**

**Task**: Write a function countVowels(str) that takes a string and returns the number of vowels in the string.

**Exercise 10: Remove Duplicates from an Array**

**Task**: Write a function removeDuplicates(arr) that removes duplicate values from an array and returns a new array.

**Exercise 11: Find the Longest Word in a Sentence**

**Task**: Write a function longestWord(sentence) that takes a string (sentence) and returns the longest word in it.

**Exercise 12: Calculate the Area of a Circle**

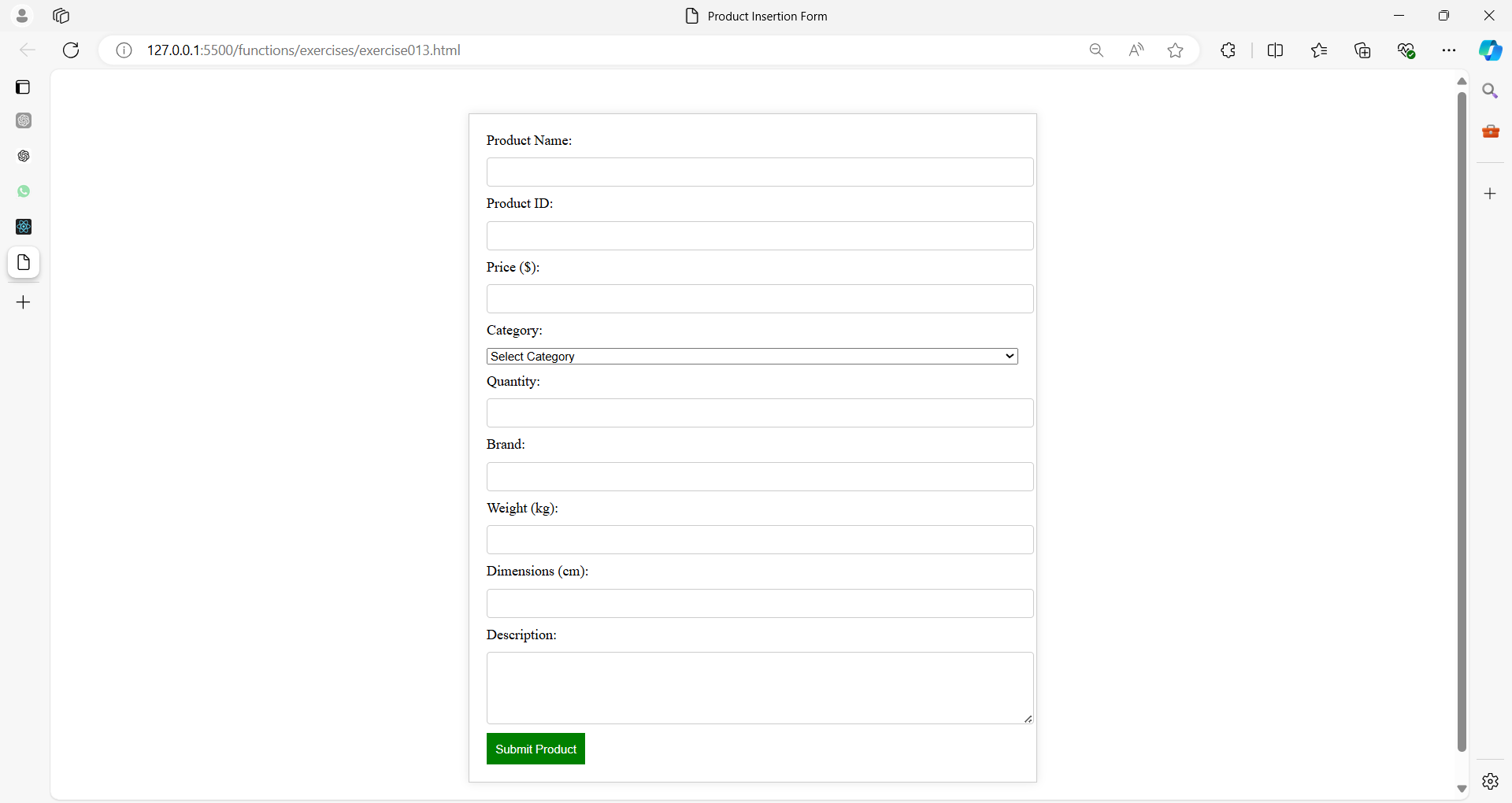
**Task**: Write a function areaOfCircle(radius) that takes the radius of a circle and returns its area using the formula: Area = π \* r^2.

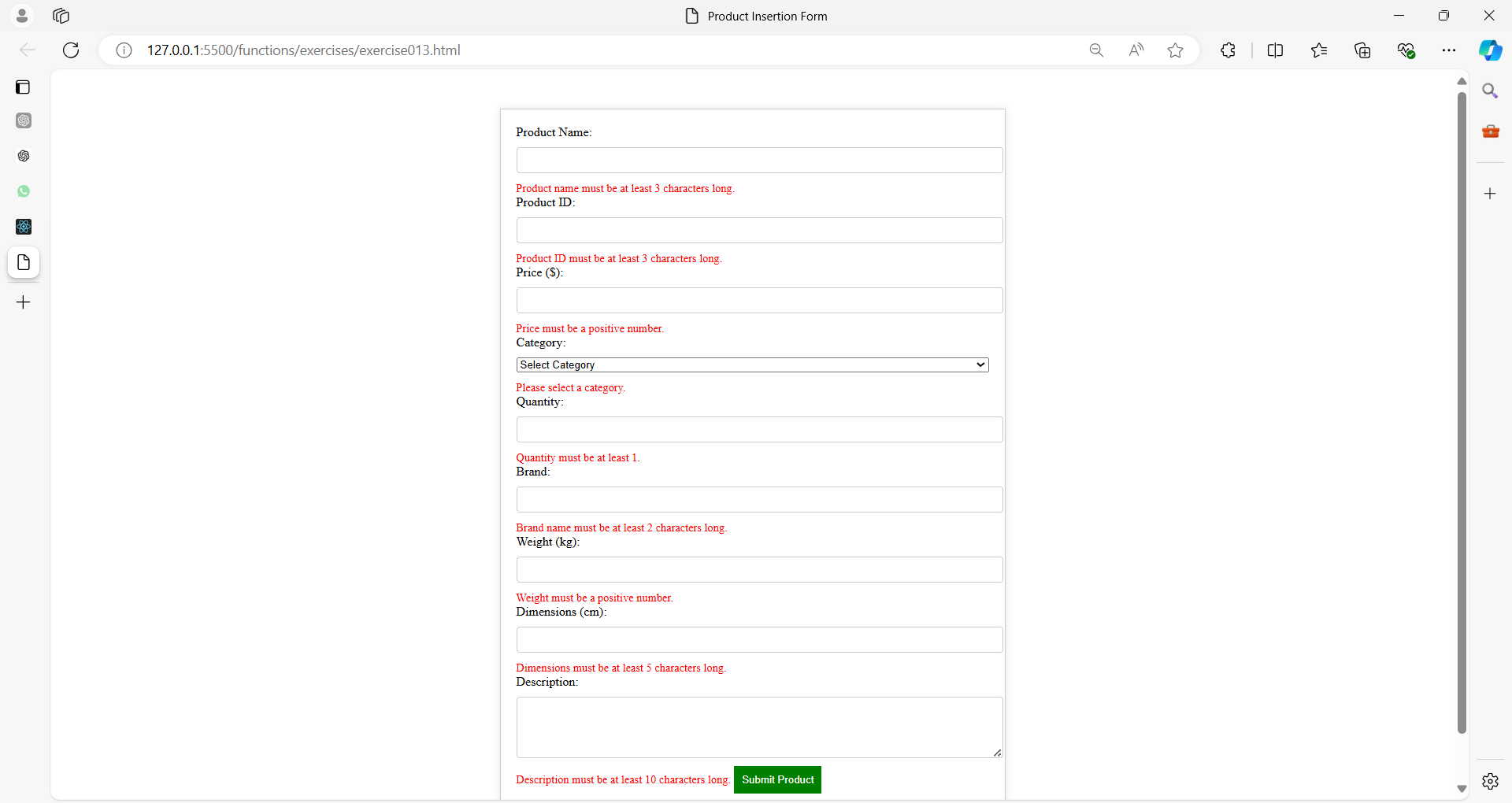
**Exercise 13: Product Insertion Form**

**Objective**: Create a product insertion form with various fields such as:

1. Product Name
2. Product ID
3. Price
4. Category
5. Quantity
6. Brand
7. Weight
8. Dimensions
9. Description

Use JavaScript to validate each field and display error messages in red when necessary.



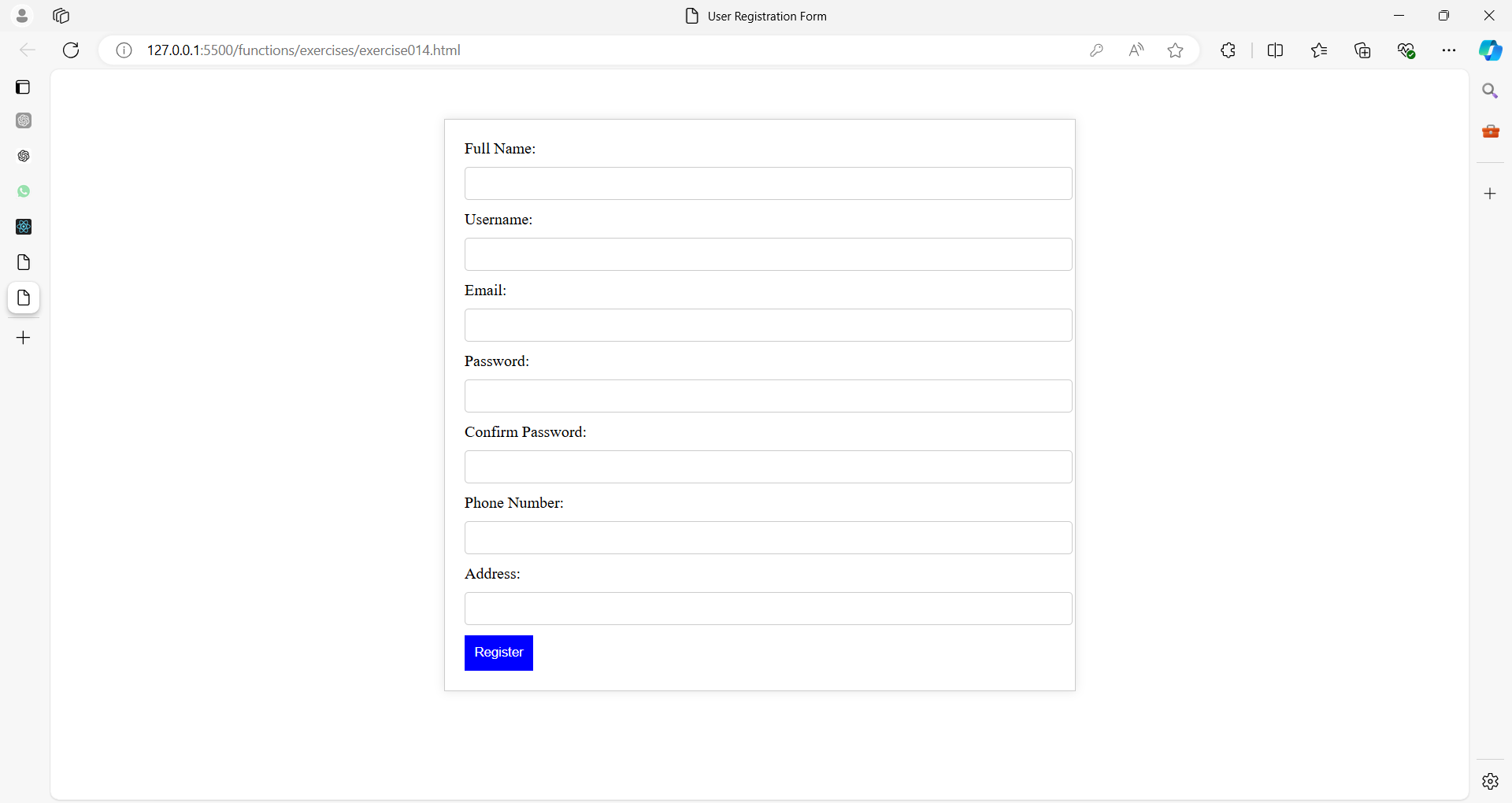


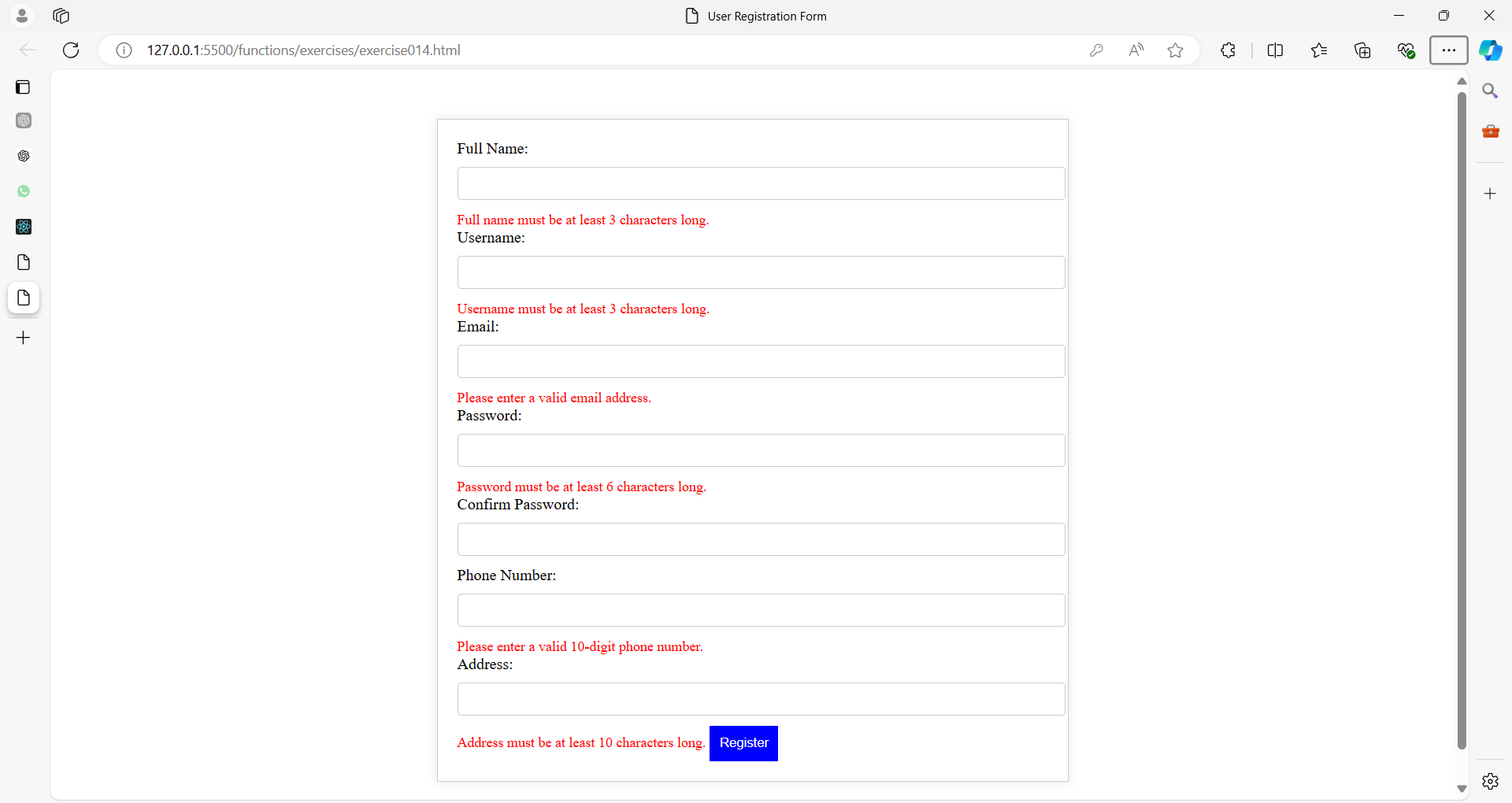
**Exercise 14: User Registration Form**

**Objective**: Create a user registration form with the following fields:

1. Full Name
2. Username
3. Email
4. Password
5. Confirm Password
6. Phone Number
7. Address

Validate each field using JavaScript and display error messages in red.



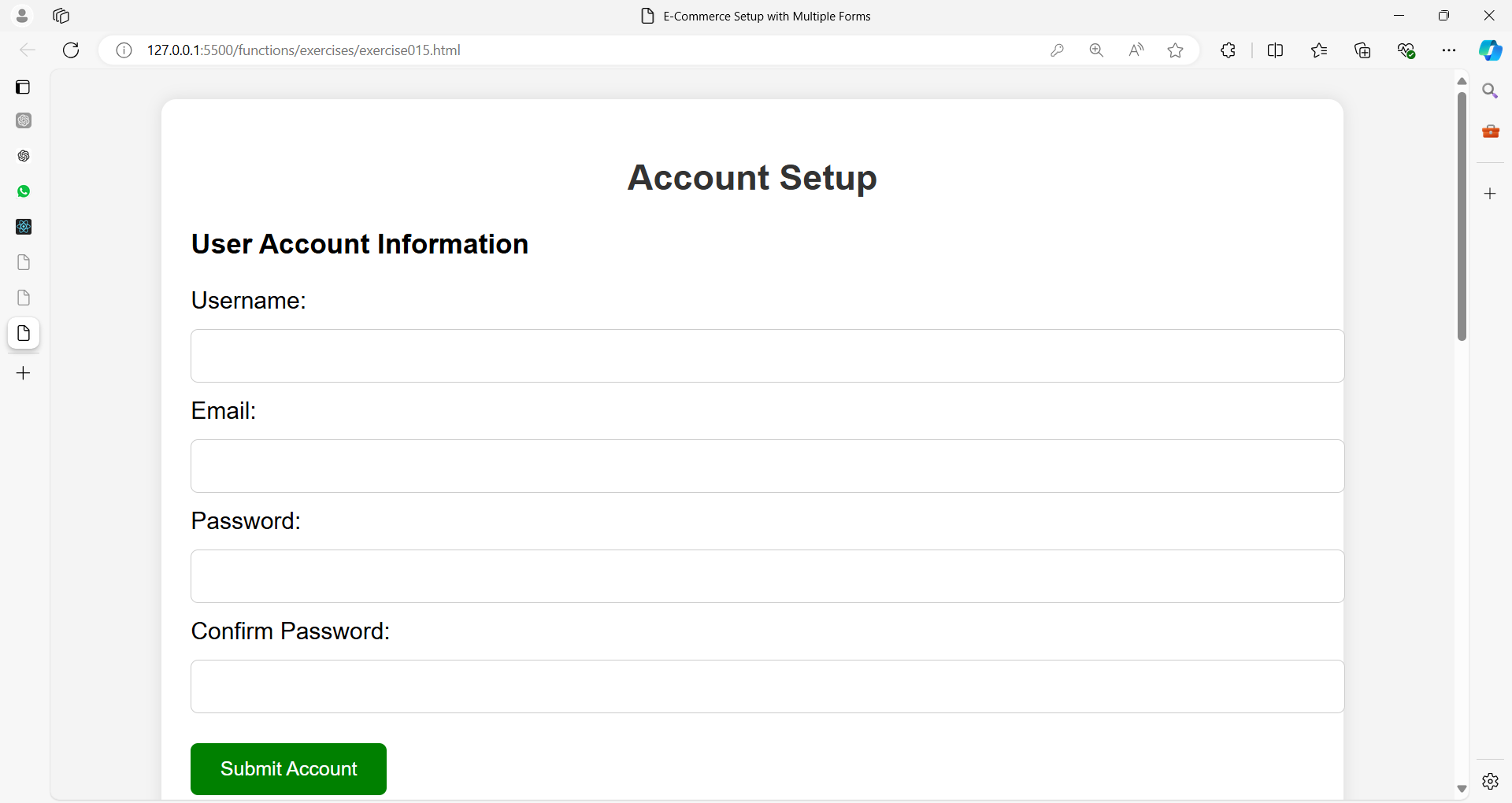


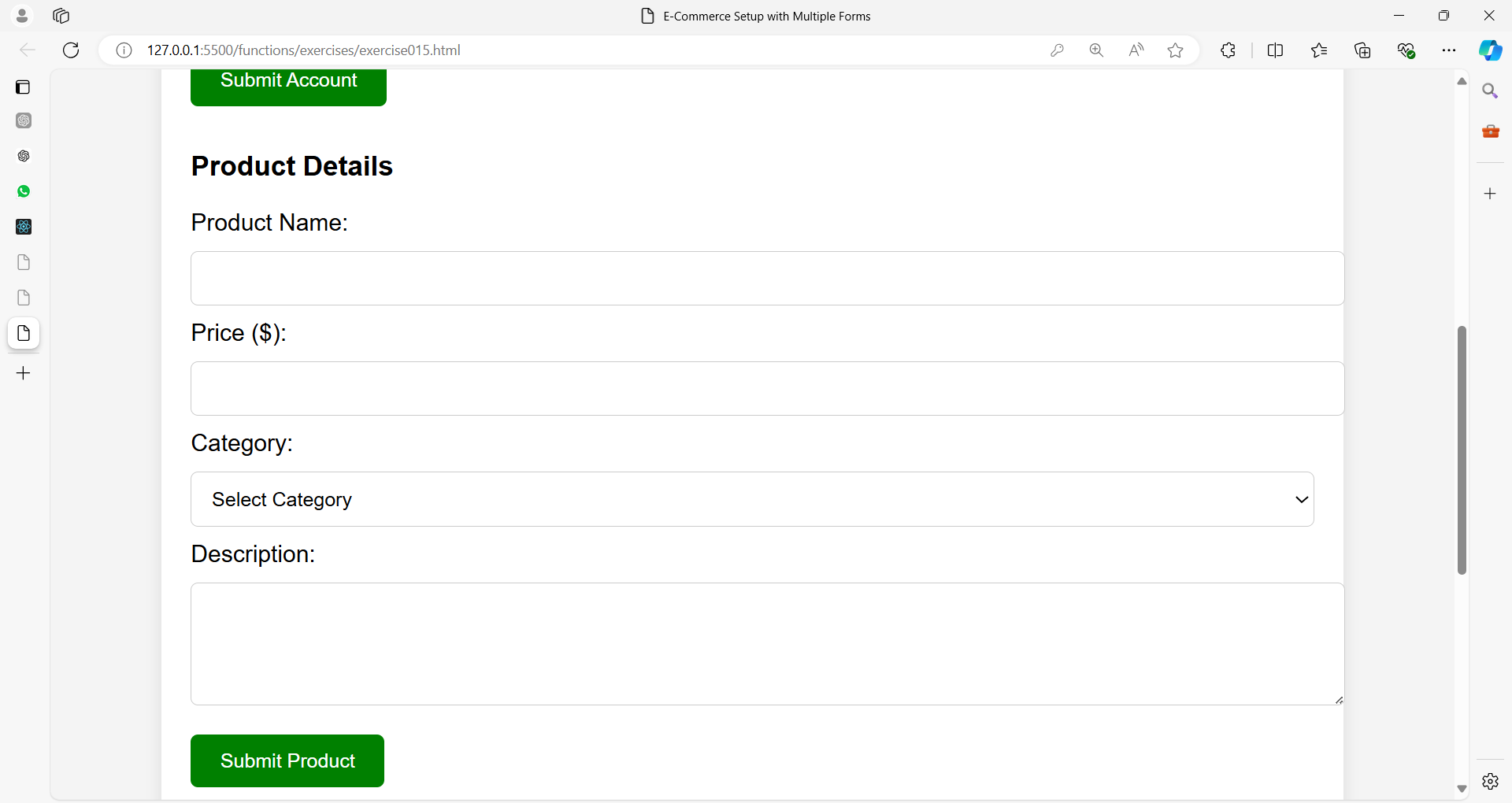
**Exercise 15: E-commerce Setup with Multiple Forms**

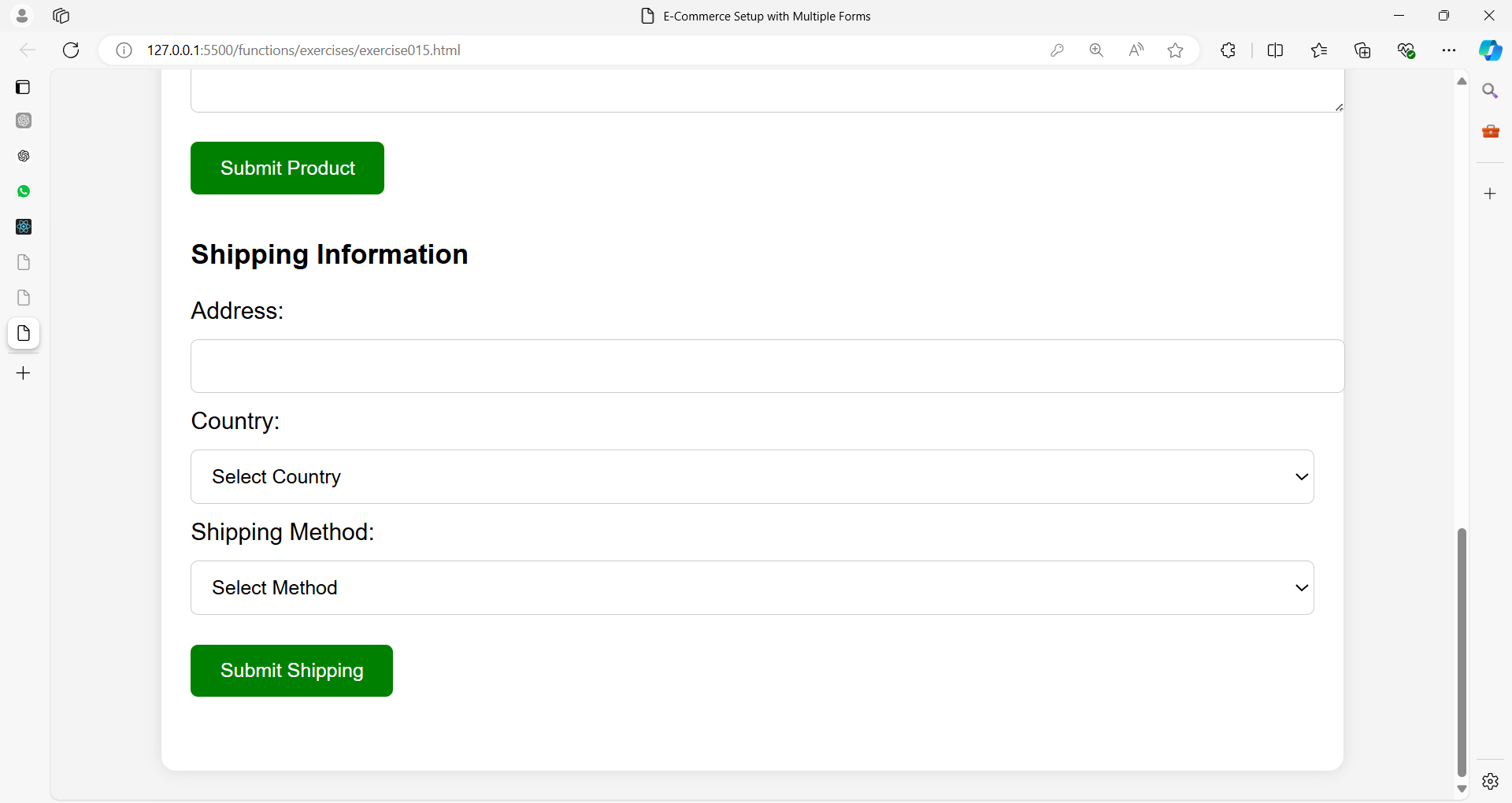
The system consists of three forms:

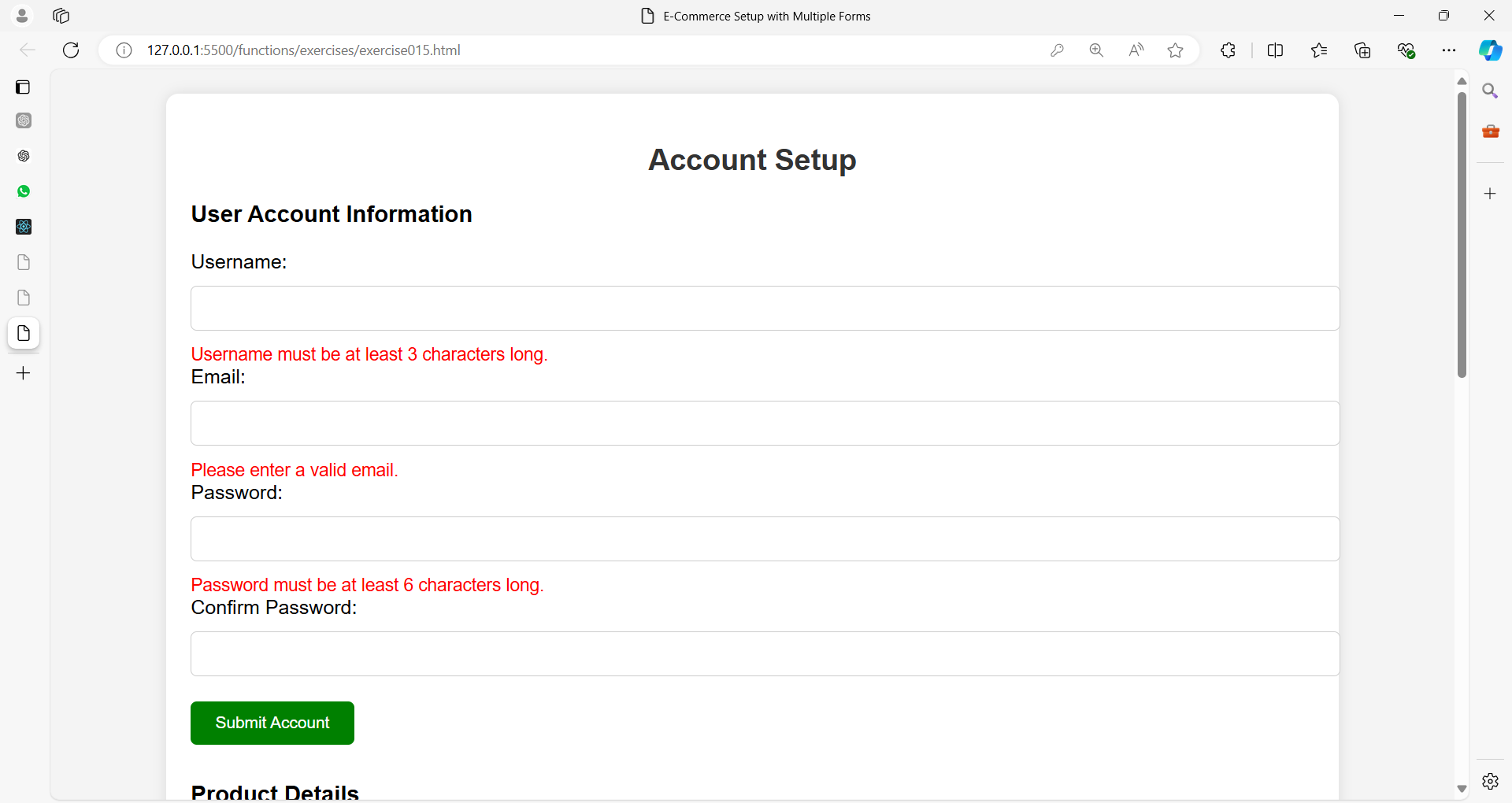
1. **User Account Form**: Collects basic user details like name, email, and password.
2. **Product Details Form**: Gathers information on the product, including name, price, category, and description.
3. **Shipping Information Form**: Collects address, country, and shipping method information.

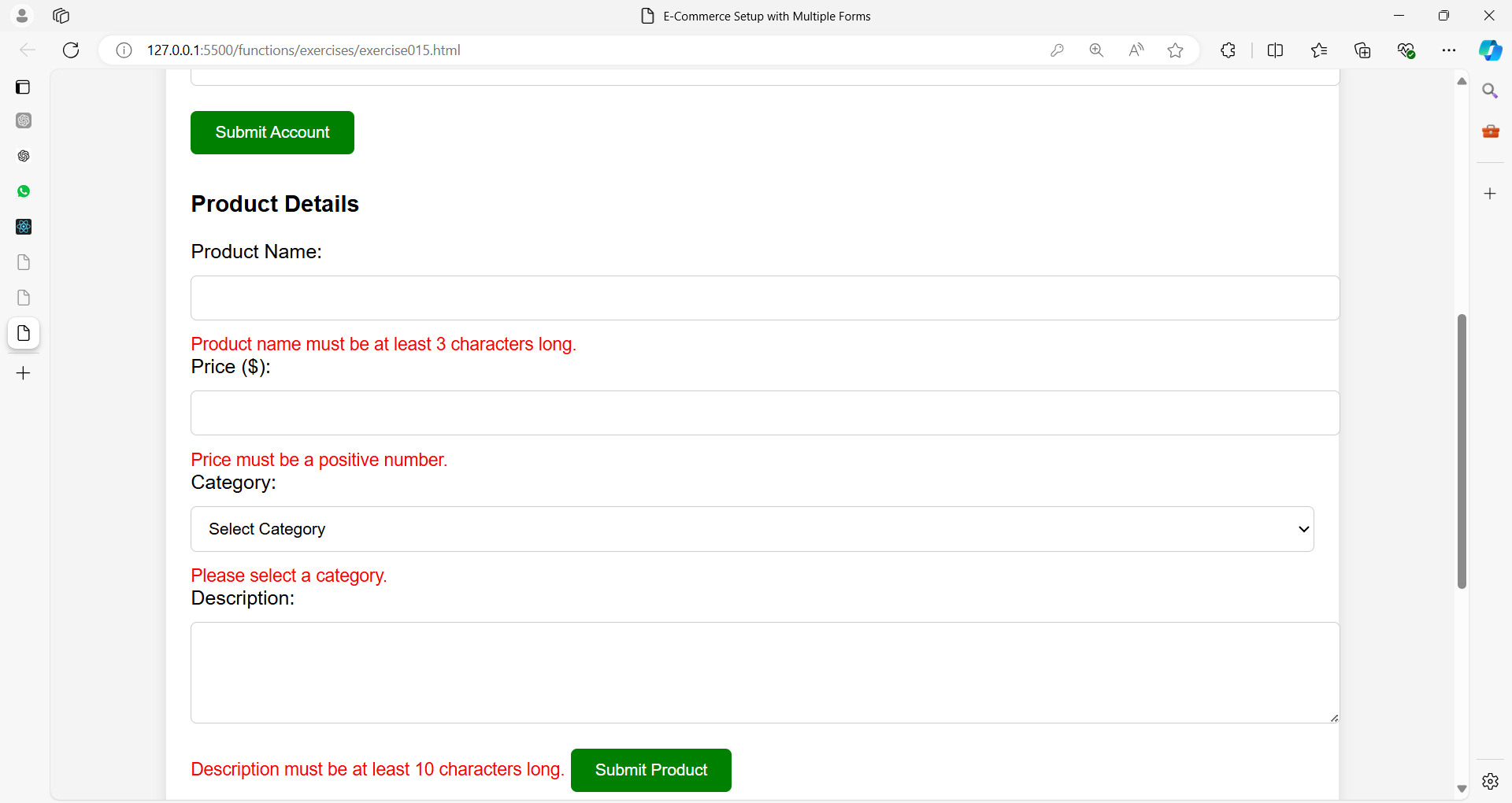
Each form has its own validation, and all forms are validated separately.

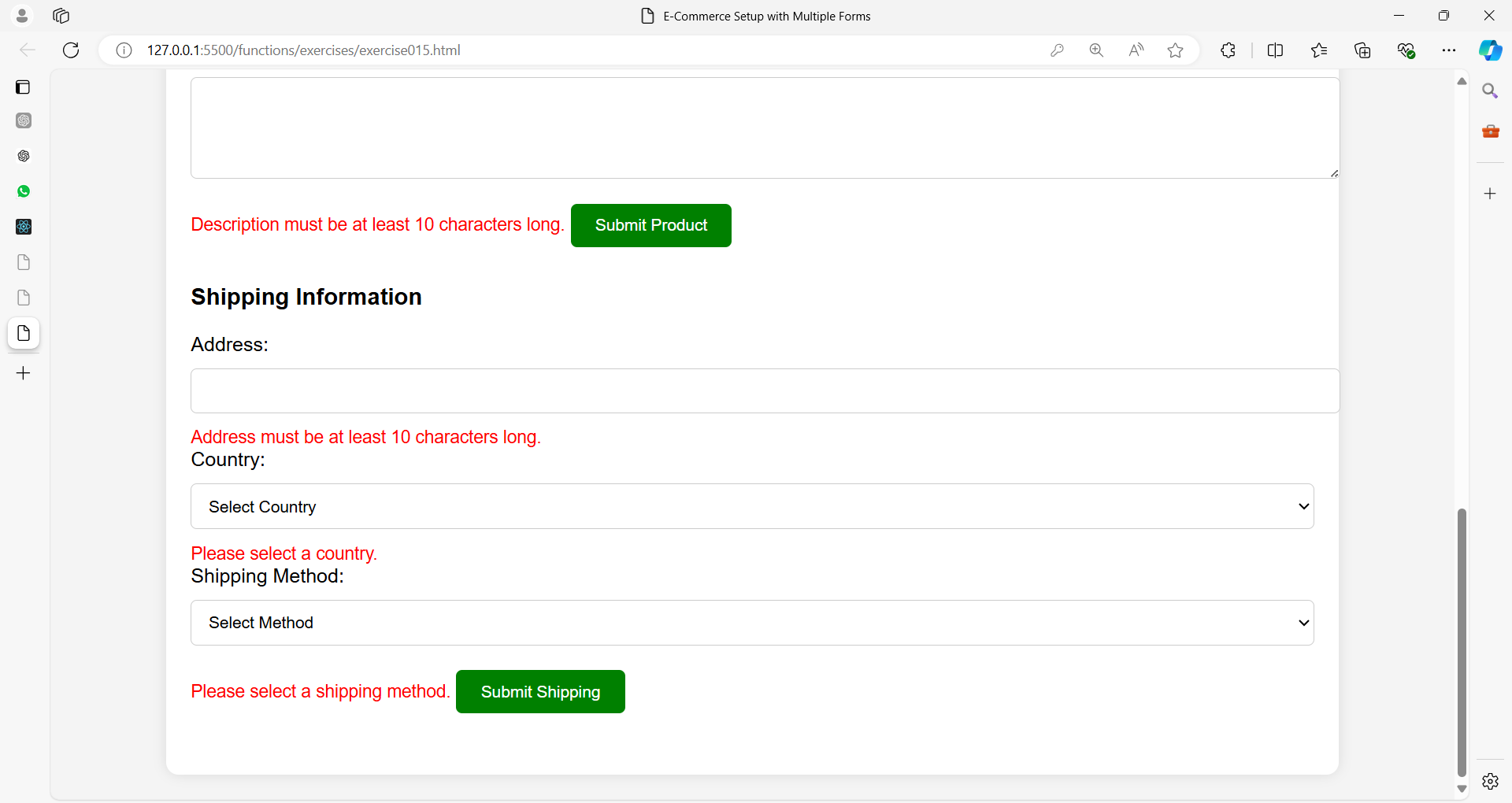








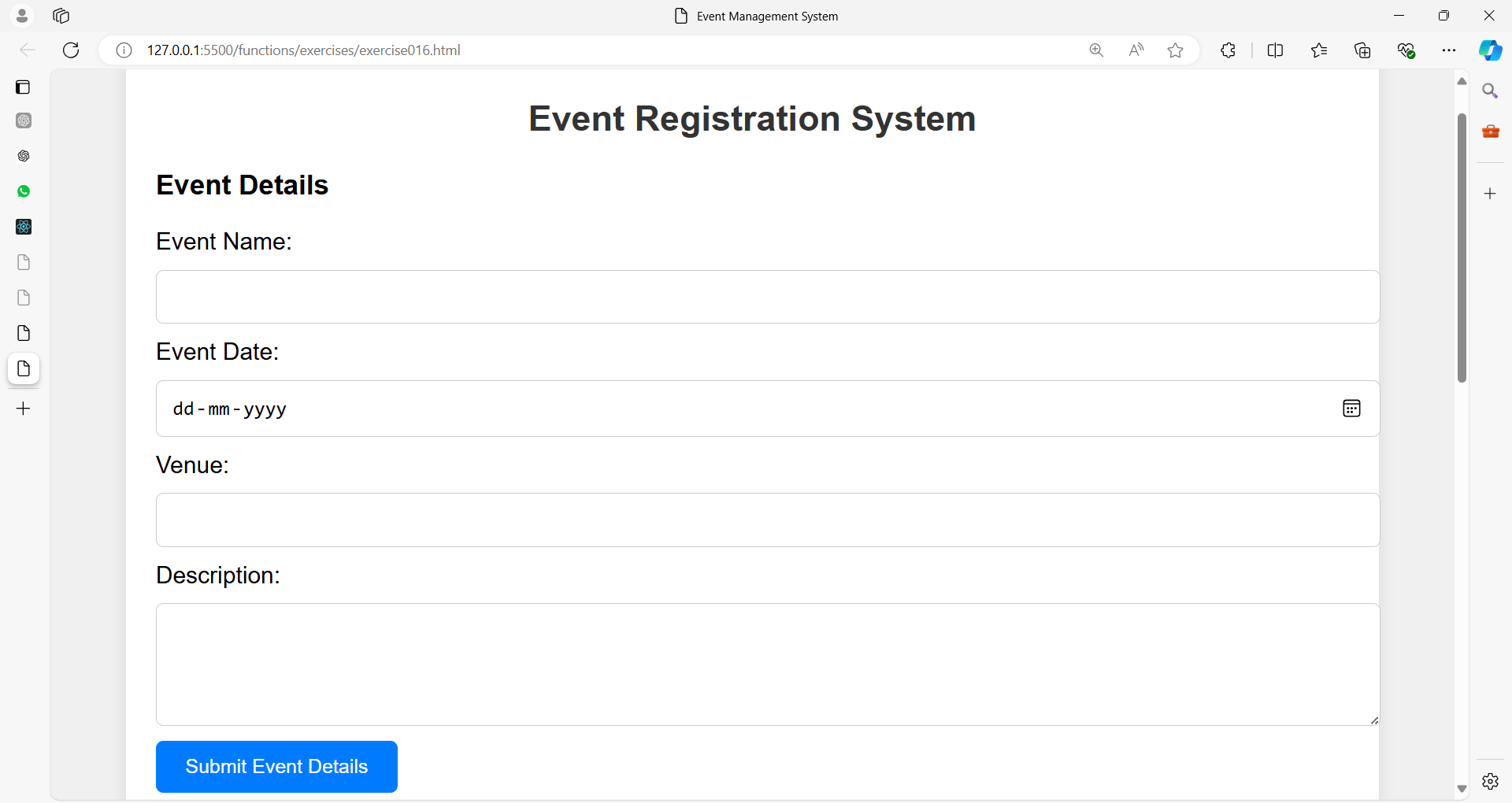


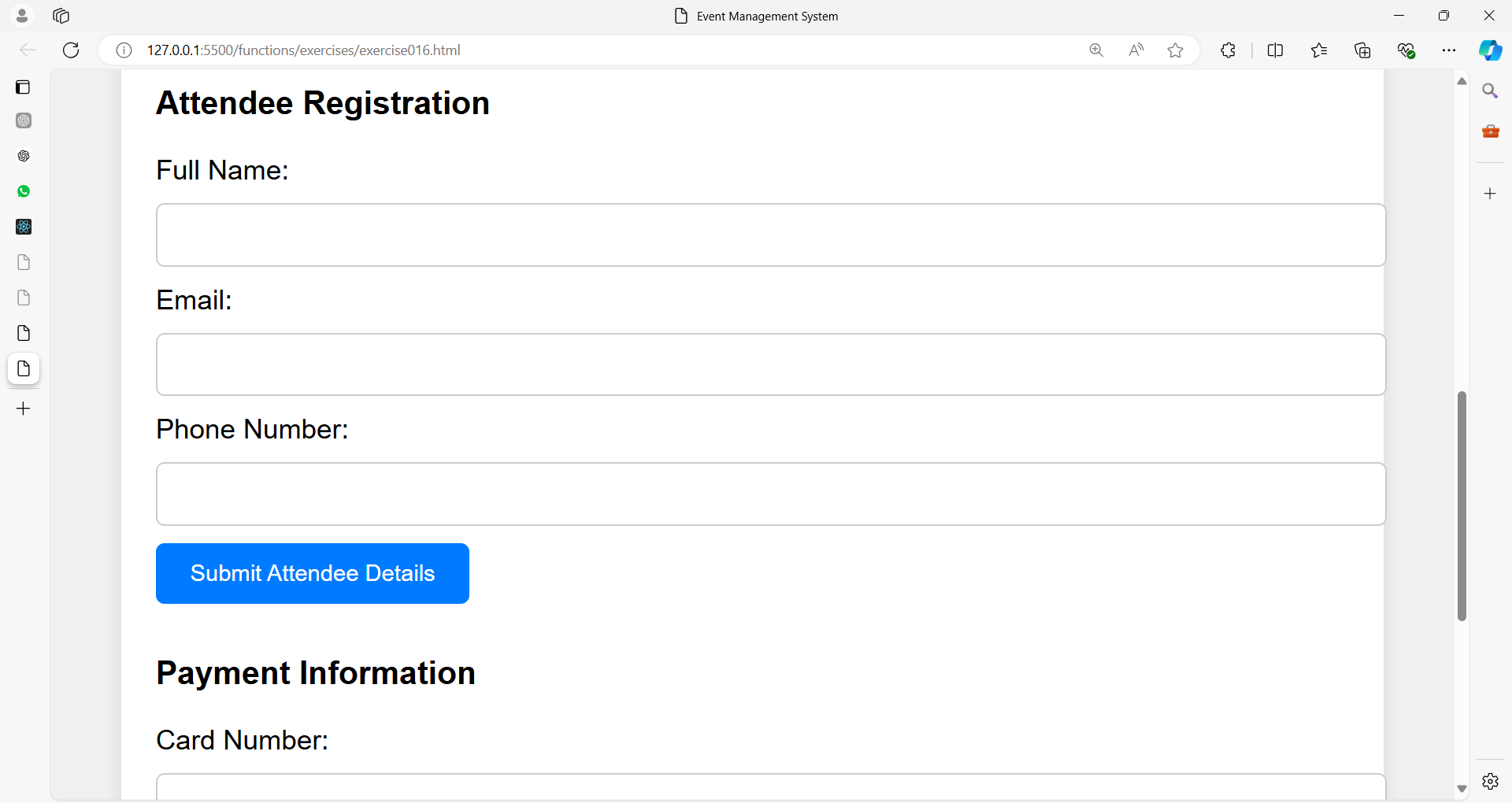


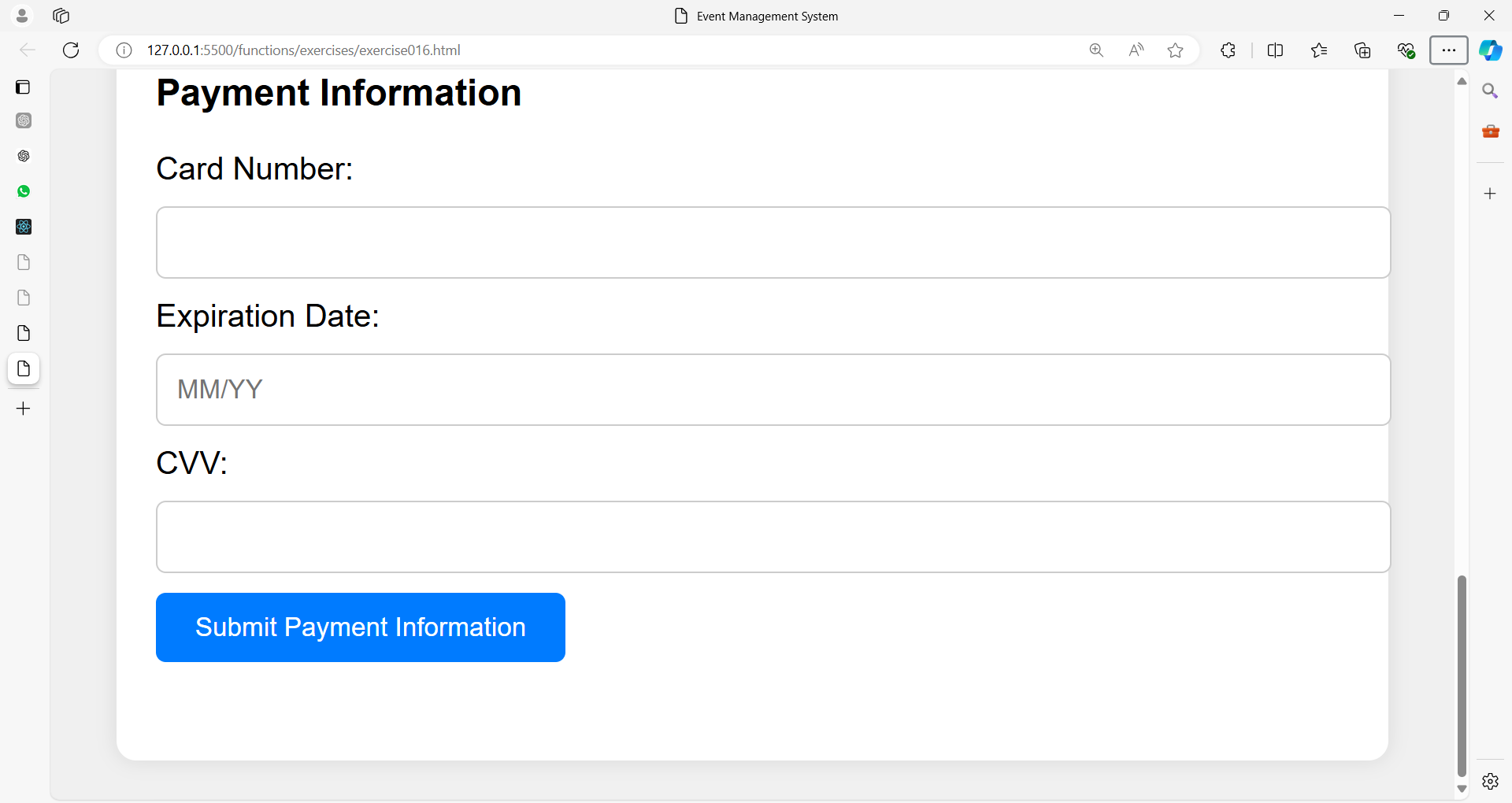
**Exercise 16: Event Registration System with Multiple Forms**

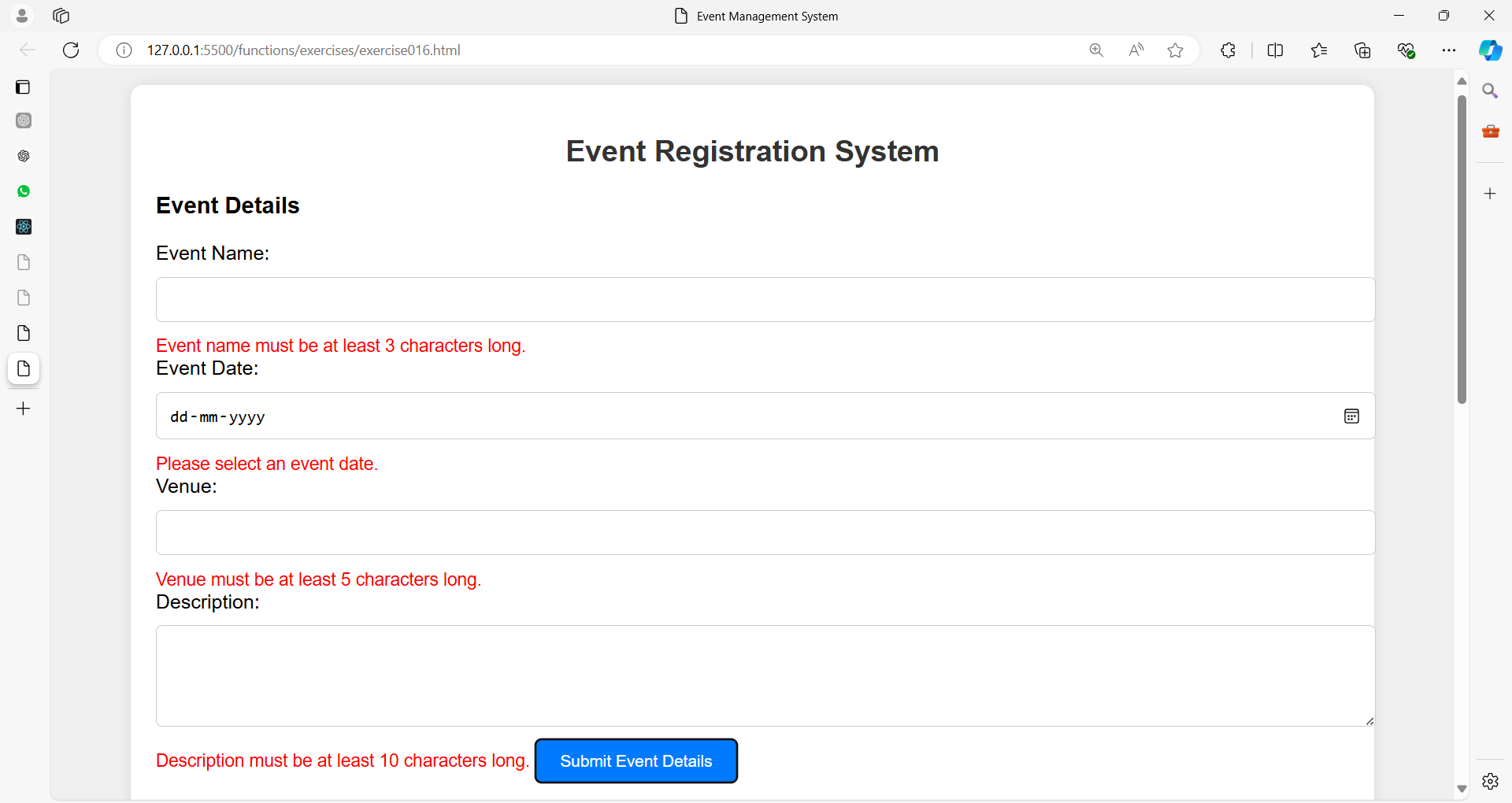
The system consists of:

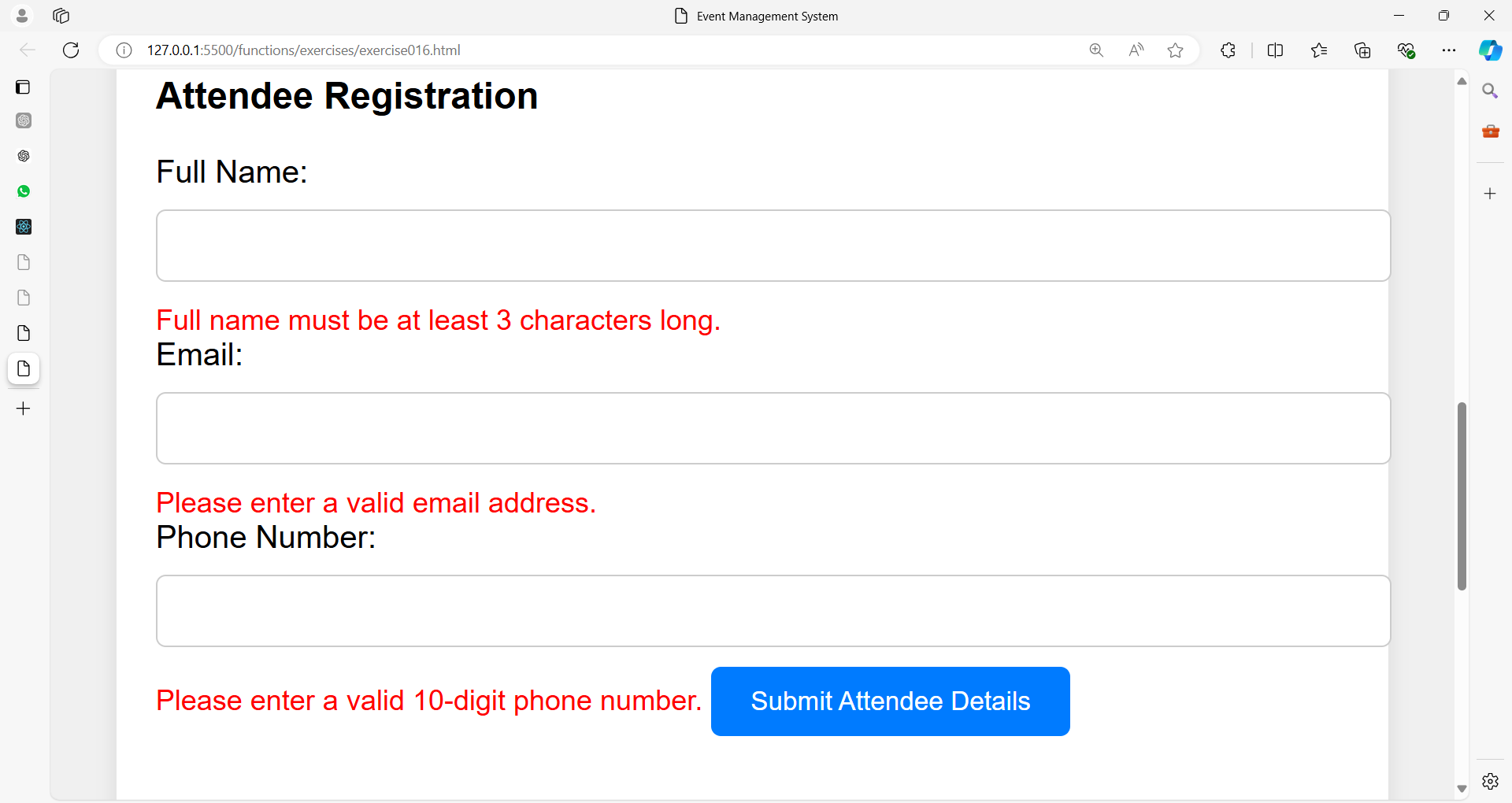
1. **Event Details Form**: Gathers event information, such as event name, date, venue, and description.
2. **Attendee Registration Form**: Collects attendee's personal information, like full name, email, and phone number.
3. **Payment Information Form**: Collects credit card details, including card number, expiration date, and CVV.

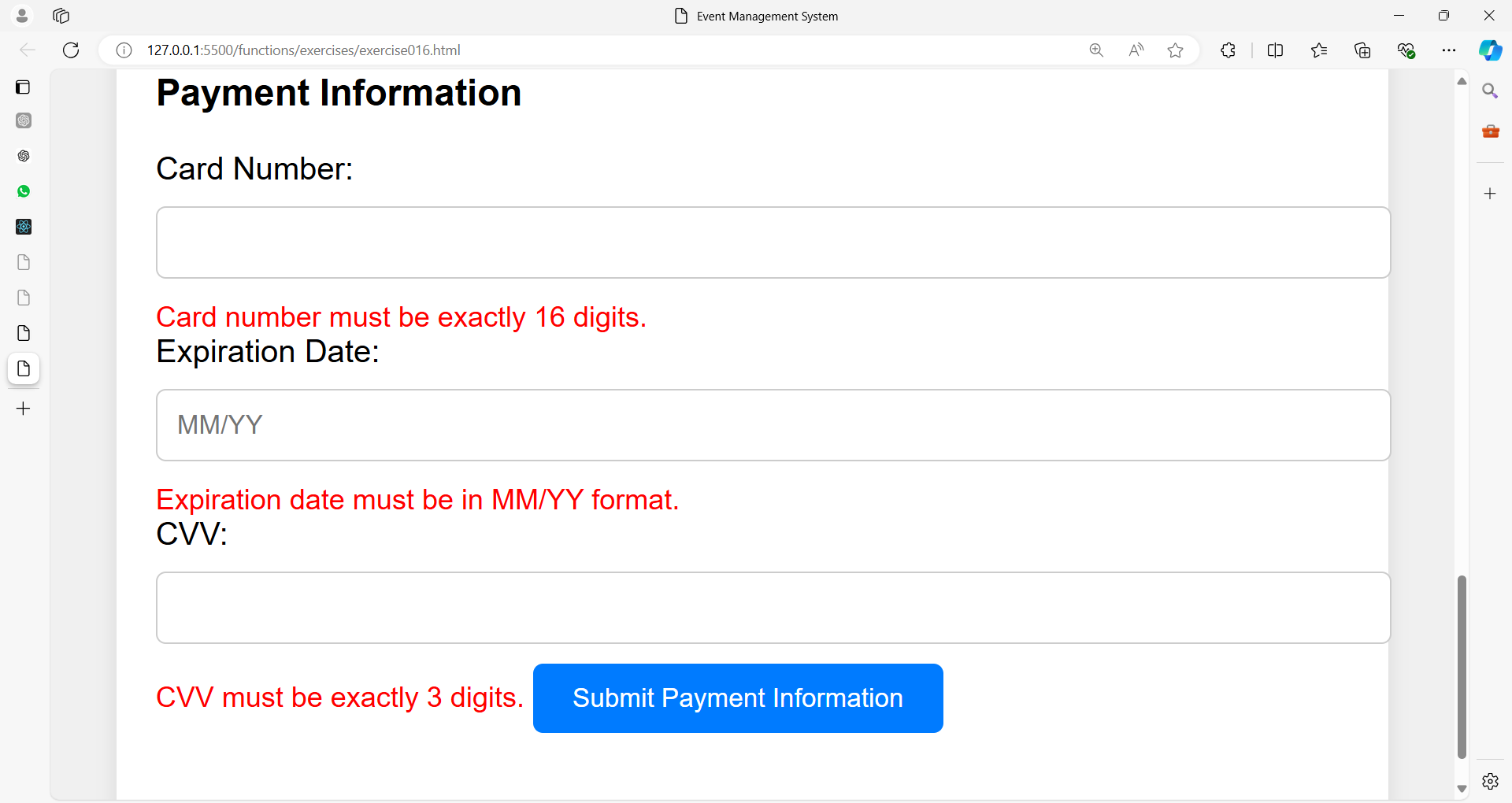












**Exercise 17: Multi-Step Job Application Form**

**Problem Statement:**

You are tasked with creating a **multi-step job application form**.

The form should be broken down into three steps:

1. **Personal Information**
   * Full Name
   * Email
   * Phone Number
2. **Educational Background**
   * Highest Degree
   * University/College
   * Graduation Year
3. **Work Experience**
   * Company Name
   * Job Title
   * Years of Experience

**Requirements:**

1. **Form Layout**:
   * Use the <fieldset> and <legend> tags to group related form controls.
   * Each section should be displayed as a separate form that appears one after the other.
   * Include "Next" and "Previous" buttons to navigate between the sections.
   * The forms should be hidden and revealed as the user progresses through the steps.
2. **Validation**:
   * Perform validation for each form before allowing the user to proceed to the next section:
     + **Full Name** must be at least 3 characters long.
     + **Email** must follow a valid email format.
     + **Phone Number** must be exactly 10 digits long.
     + **Highest Degree** and **Company Name** should be at least 2 characters long.
     + **Graduation Year** should be between 1900 and the current year.
     + **Years of Experience** should be a number between 0 and 50.
3. **JavaScript**:
   * Write JavaScript code to handle the validation of each form.
   * Use error messages displayed in red if the input is invalid.
   * Implement the logic to show/hide forms based on user navigation (Next/Previous).
4. **Styling**:
   * Style the form elements using CSS. The form should be visually appealing and responsive.
   * Error messages should be displayed in red under each invalid input field.
   * Ensure that the form is laid out clearly and is easy to use.

**Additional Requirements:**

* Ensure that all three forms are submitted as a single entity after validation.
* Display a success message using alert() or on the page itself when the form is submitted successfully.

**Example of Inputs:**

**Step 1**: Personal Information

Full Name: John Doe

Email: johndoe@example.com

Phone Number: 1234567890

**Step 2**: Educational Background

Highest Degree: Bachelor of Science

University/College: XYZ University

Graduation Year: 2020

**Step 3**: Work Experience

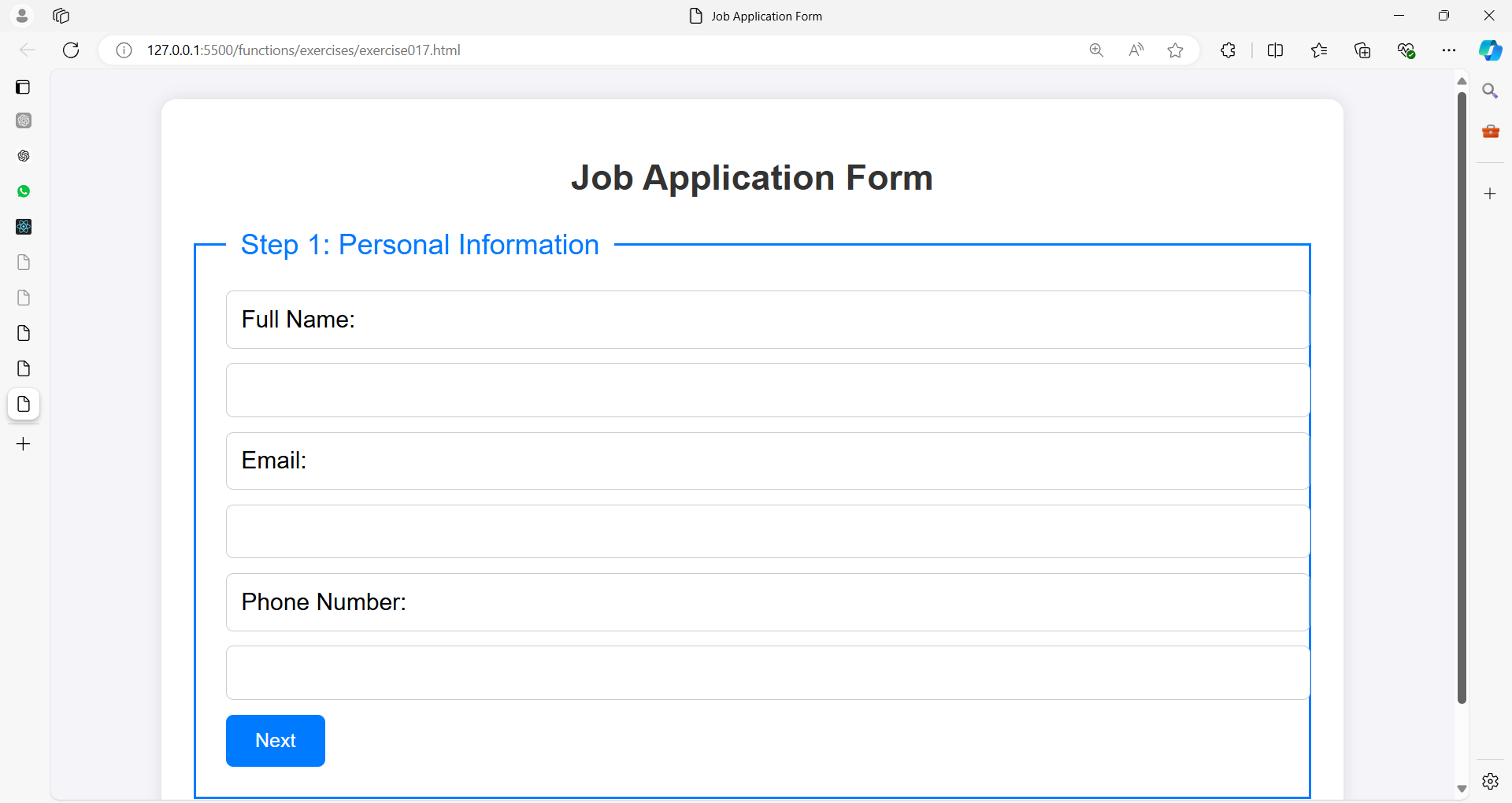
Company Name: ABC Corp

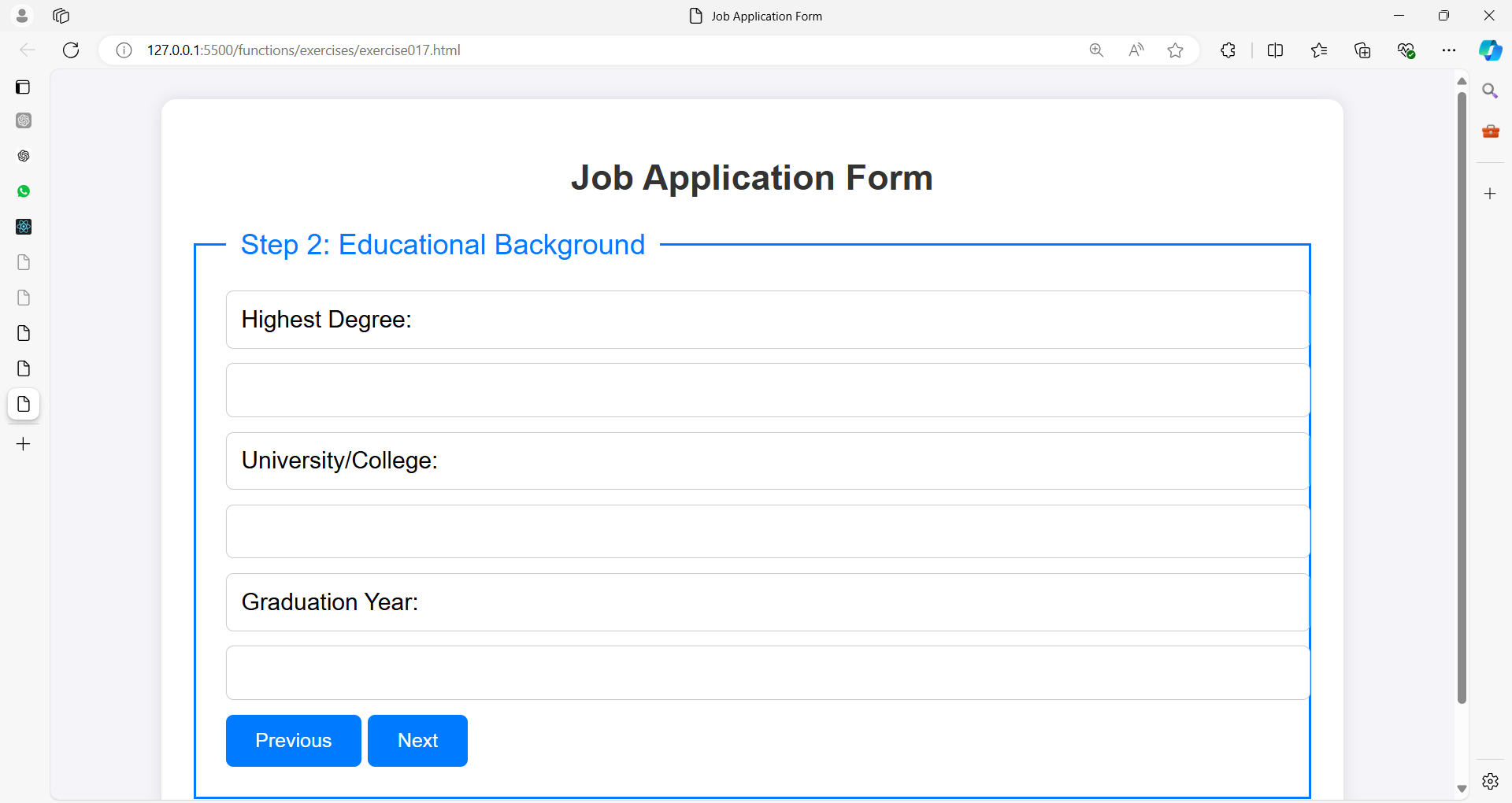
Job Title: Software Engineer

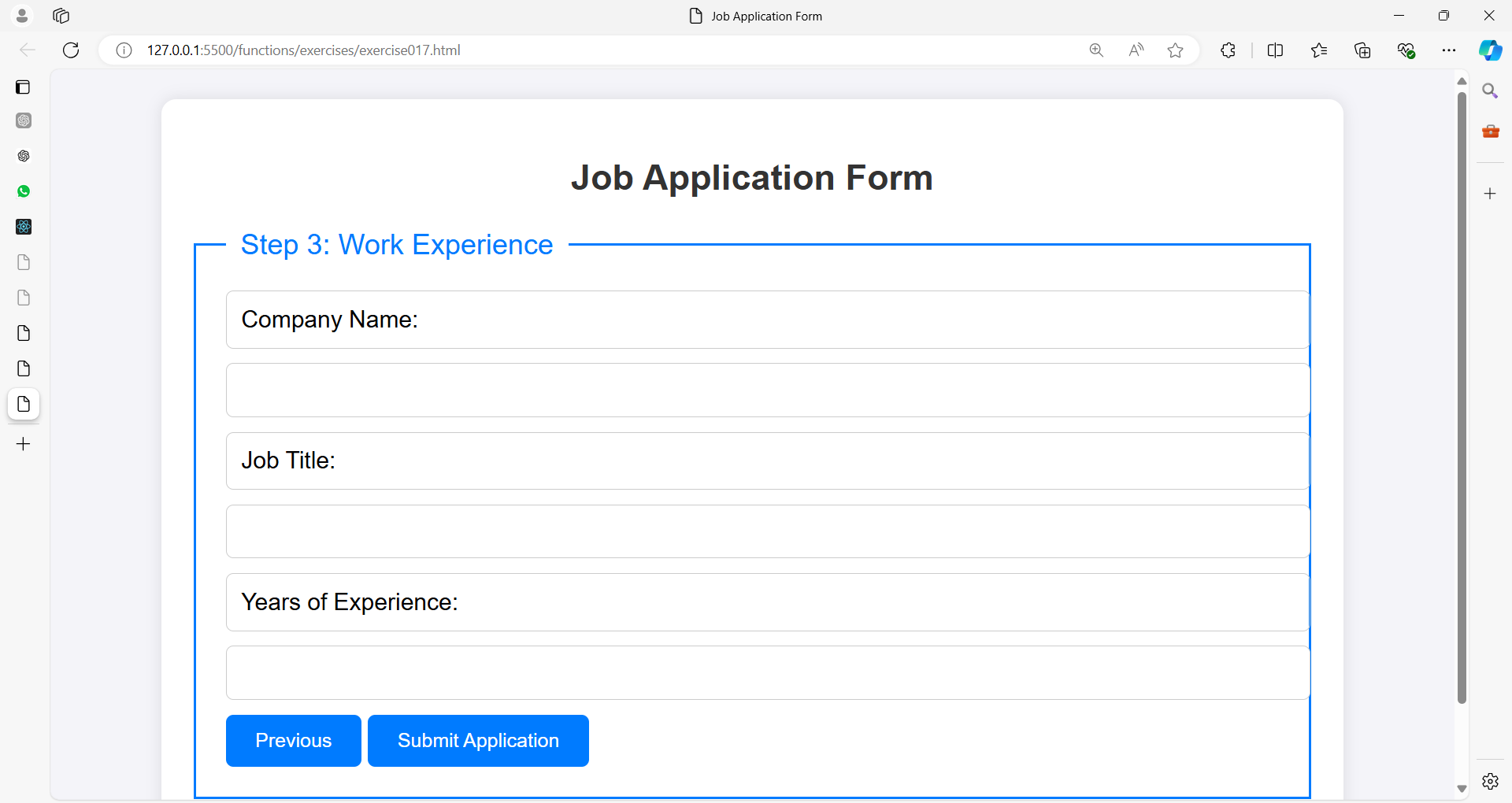
Years of Experience: 5

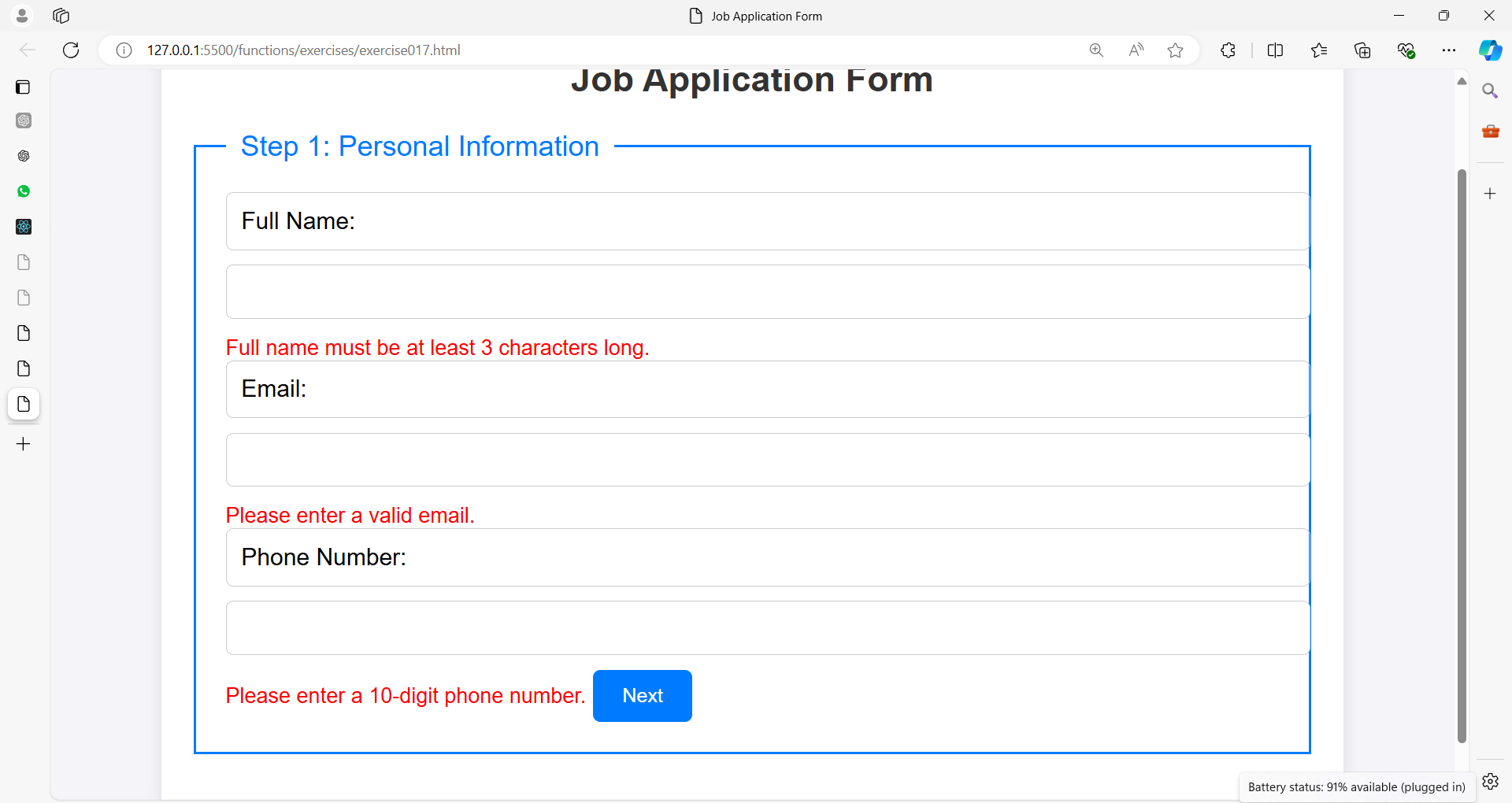
**Deliverables:**

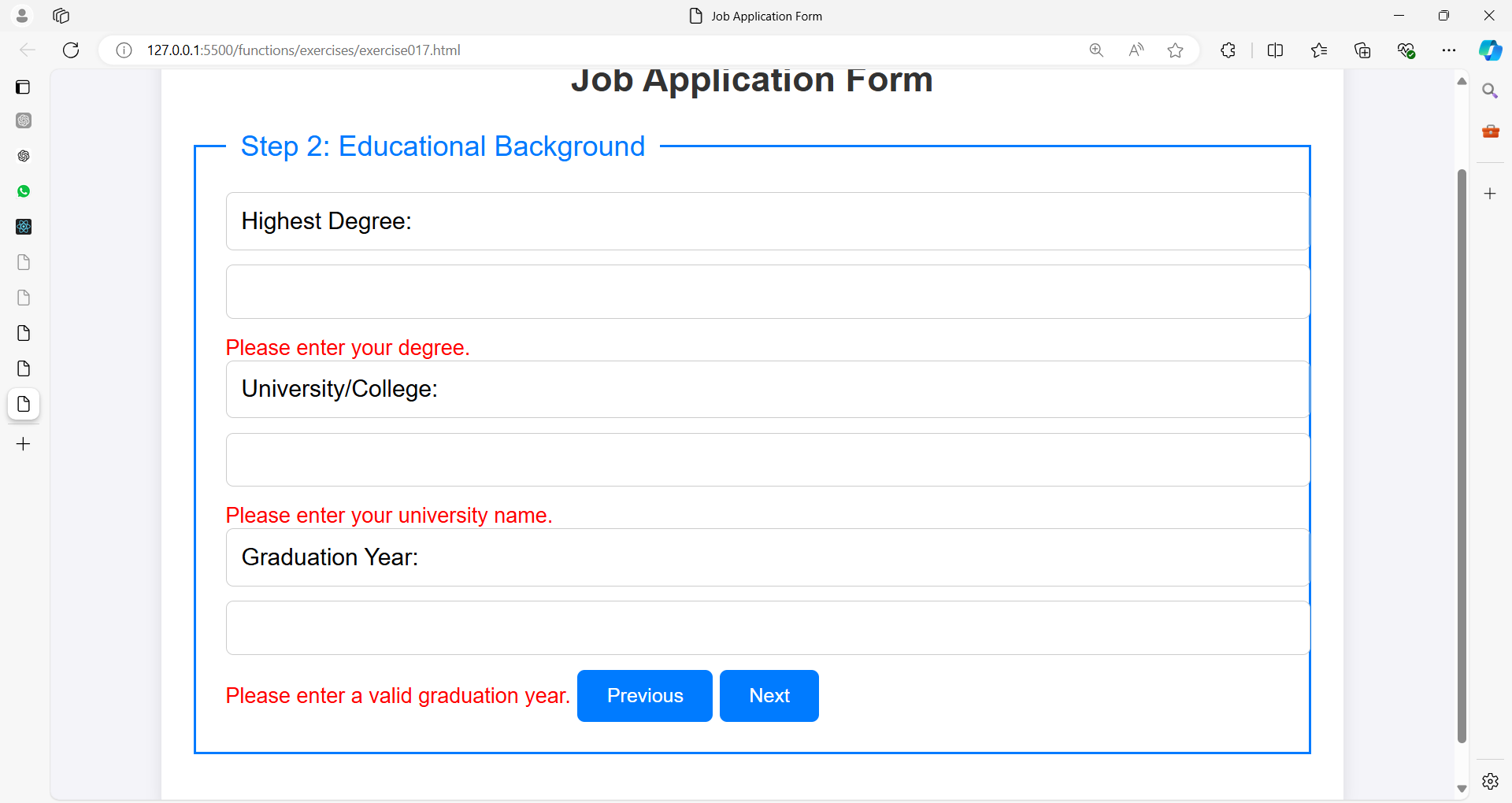
1. HTML code for the job application form.
2. CSS code to style the form and error messages.
3. JavaScript code for validation and form navigation.

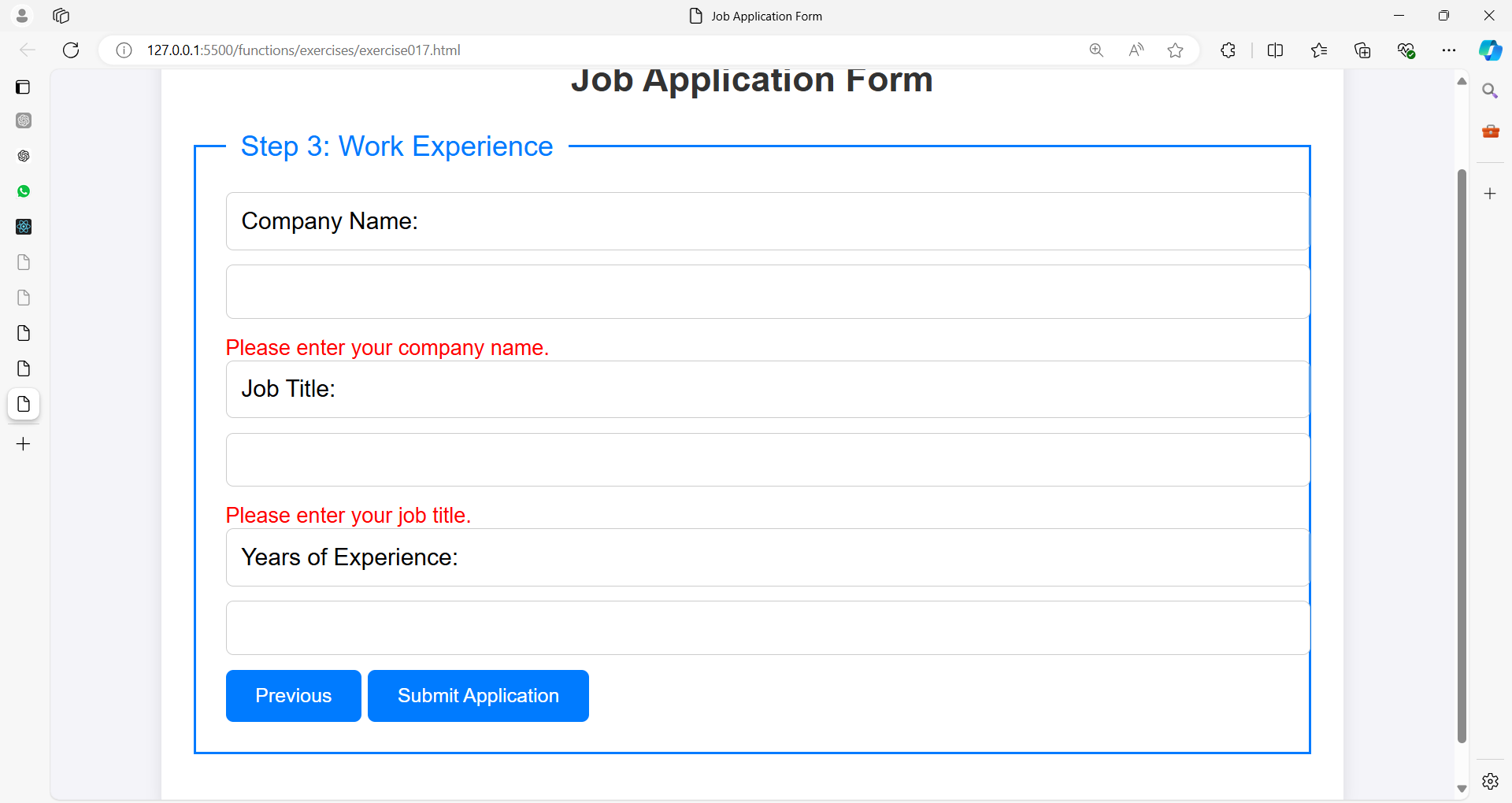












**Exercise 18: Multi-Step Product Order Form**

**Problem Statement:**

Create a **multi-step product order form** that allows users to place an order for a product. The form should be divided into three steps:

1. **Product Selection**
   * Product Name
   * Category (Dropdown: Electronics, Furniture, Clothing, etc.)
   * Quantity
2. **Shipping Information**
   * Full Name
   * Address
   * City
   * Postal Code
   * Country (Dropdown with at least 5 countries)
3. **Payment Information**
   * Cardholder's Name
   * Credit Card Number
   * Expiration Date
   * CVV

**Requirements:**

1. **Form Layout**:
   * Use the <fieldset> and <legend> tags to clearly group form fields.
   * Display one form section at a time (Product Selection, Shipping Information, Payment Information).
   * Implement "Next" and "Previous" buttons to navigate between the steps.
   * Each step should be validated before moving to the next section.
2. **Validation**:
   * Implement validation checks in JavaScript:
     + **Product Name** must be at least 3 characters.
     + **Quantity** must be a number greater than 0.
     + **Full Name** must be at least 3 characters.
     + **Postal Code** must be a 5-digit number.
     + **Credit Card Number** must be 16 digits long.
     + **Expiration Date** must be a valid future date.
     + **CVV** must be exactly 3 digits.
3. **JavaScript**:
   * Write JavaScript to validate each form step independently.
   * Display error messages in red if validation fails.
   * Use JavaScript to show or hide each form step based on user interaction.
4. **CSS**:
   * Style the form fields, buttons, and error messages.
   * The form should be user-friendly and visually appealing.
   * Error messages should appear below the relevant input fields in red.
   * Ensure the form layout is responsive.
5. **Submission**:
   * Once the final step is completed and all fields are valid, display a message indicating successful order submission.
   * Use either an alert() or display the success message on the page.

**Example Inputs:**

**Step 1**: Product Selection

Product Name: Laptop

Category: Electronics

Quantity: 2

**Step 2**: Shipping Information

Full Name: Jane Doe

Address: 123 Main St

City: New York

Postal Code: 12345

Country: United States

**Step 3**: Payment Information

Cardholder's Name: Jane Doe

Credit Card Number: 1234567812345678

Expiration Date: 12/25

CVV: 123

**Deliverables:**

1. HTML code for the multi-step product order form.
2. CSS to style the form and error messages.
3. JavaScript for form validation and step navigation.

**Bonus Challenge:**

* Allow users to see a summary of their order (product and shipping details) before submitting the payment.
* Include a "Confirm Order" button after showing the summary.

