Submitted by
Elaine Mae A.Bertiz
BSIT 3-C

September 1, 2024

IT ELEC 1

WEB DEVELOPMENT

Submitted to
Prof.John Paul Azore

Part 1: Defining Basic Routes

1. *Create a simple route that returns a view for the homepage. The view should display a welcome message.*
2. *Create additional routes that:*
   - *Return a view for an "About Us" page.*
   - *Redirect from /home to / (the homepage).*
   - *Display a "Contact Us" form.*

Part 2: Using Route Parameters

1. *Define a route with a required parameter:*
   - *Create a route that accepts a username parameter and displays a welcome message that includes the username.*
   - *Example: /user/johndoe should return a view with "Welcome, johndoe!".*
2. *Define a route with an optional parameter:*
   - *Modify the previous route to make the username optional. If no username is provided, display a generic welcome message.*
   - *Example: /user should return a view with "Welcome, Guest!".*
3. *Apply regular expression constraints to the route parameters:*
   - *Ensure that the username only accepts alphabetic characters (a-z, A-Z).*

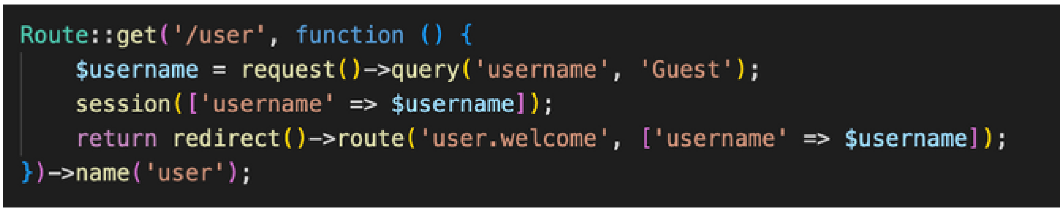
Part 3: Documentation (Individual)

1. *Take screenshots of your application runtime along with its code.*
2. *Provide a detailed explanation of how your code works, highlighting the purpose of each key section.*

DOCUMENTATION

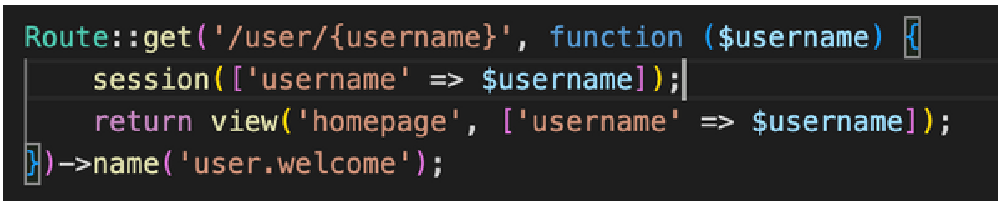
ROUTES

User Route(Query Parameters): This route handles `/user` and accepts a `username` query parameter (e.g., `/user?username=johndoe`). It stores the username in the session and redirects to the `user.welcome` route with the username as a parameter for a personalized welcome page.



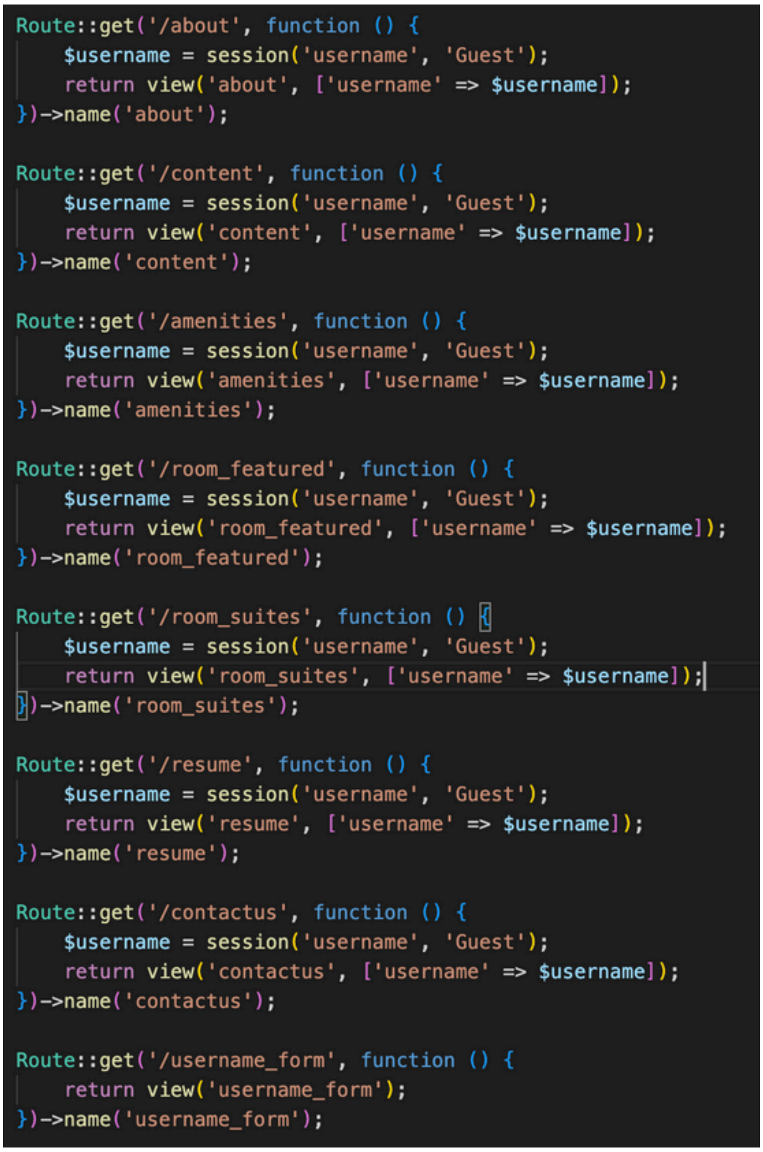
```
Route::get('/user', function () {
    $username = request()->query('username', 'Guest');
    session(['username' => $username]);
    return redirect()->route('user.welcome', ['username' => $username]);
})->name('user');
```

User Route Parameters: This route handles the /user/{username} URL. It accepts a username as a URL parameter (e.g., /user/johndoe). The route stores the username in the session and then returns the homepage view with the username passed as data. This provides a personalized welcome page.

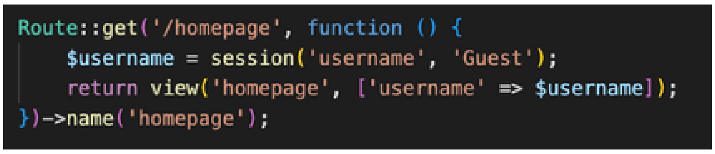


```
Route::get('/user/{username}', function ($username) {
    session(['username' => $username]);
    return view('homepage', ['username' => $username]);
})->name('user.welcome');
```

User Route Parameters: This route handles the /user/{username} URL. It accepts a username as a URL parameter (e.g., /user/johndoe). The route stores the username in the session and then returns the homepage view with the username passed as data. This provides a personalized welcome page.

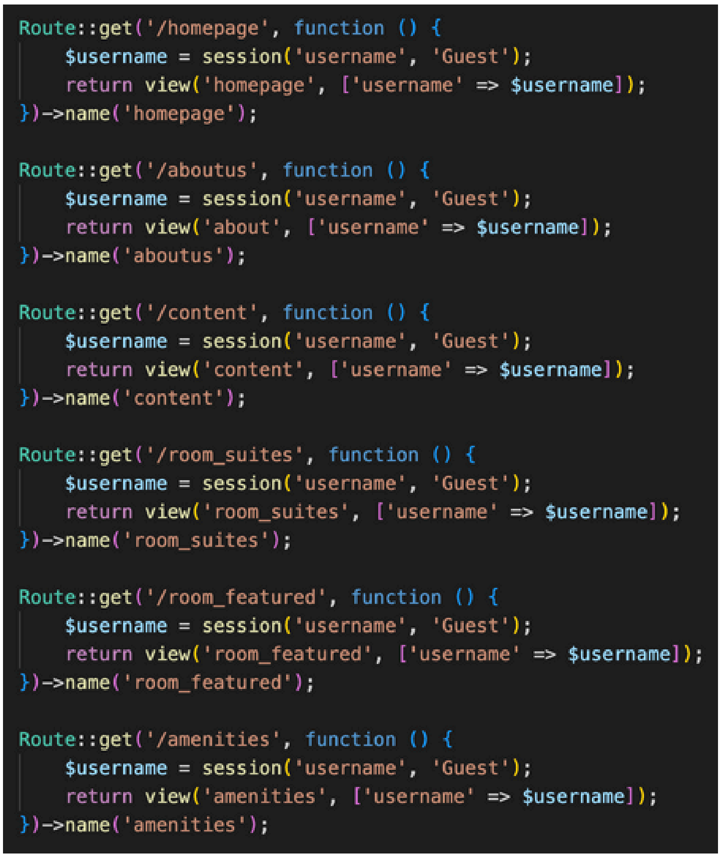


```php
Route::get('/about', function () {
    $username = session('username', 'Guest');
    return view('about', ['username' => $username]);
})->name('about');

Route::get('/content', function () {
    $username = session('username', 'Guest');
    return view('content', ['username' => $username]);
})->name('content');

Route::get('/amenities', function () {
    $username = session('username', 'Guest');
    return view('amenities', ['username' => $username]);
})->name('amenities');

Route::get('/room_featured', function () {
    $username = session('username', 'Guest');
    return view('room_featured', ['username' => $username]);
})->name('room_featured');

Route::get('/room_suites', function () {
    $username = session('username', 'Guest');
    return view('room_suites', ['username' => $username]);
})->name('room_suites');

Route::get('/resume', function () {
    $username = session('username', 'Guest');
    return view('resume', ['username' => $username]);
})->name('resume');

Route::get('/contactus', function () {
    $username = session('username', 'Guest');
    return view('contactus', ['username' => $username]);
})->name('contactus');

Route::get('/username_form', function () {
    return view('username_form');
})->name('username_form');
```



```php
Route::get('/homepage', function () {
    $username = session('username', 'Guest');
    return view('homepage', ['username' => $username]);
})->name('homepage');
```

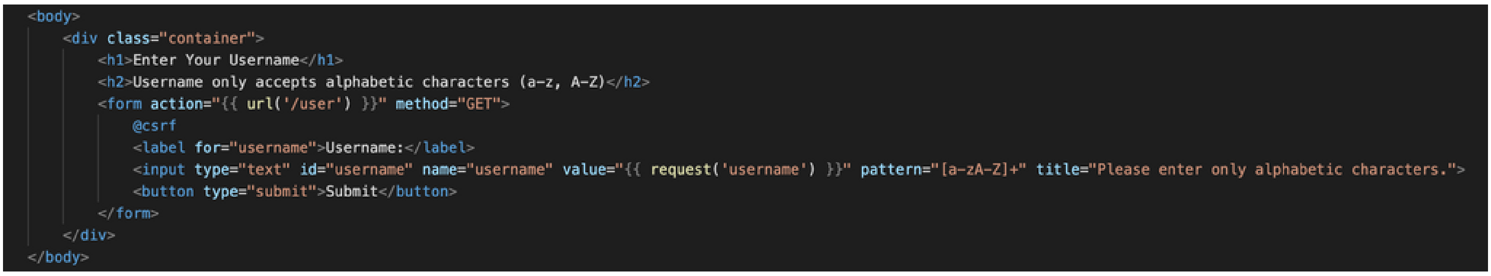
This route handles the /homepage URL. It gets the username from the session and shows the homepage with the username included.



```php
Route::get('/homepage', function () {
    $username = session('username', 'Guest');
    return view('homepage', ['username' => $username]);
})->name('homepage');

Route::get('/aboutus', function () {
    $username = session('username', 'Guest');
    return view('about', ['username' => $username]);
})->name('aboutus');

Route::get('/content', function () {
    $username = session('username', 'Guest');
    return view('content', ['username' => $username]);
})->name('content');

Route::get('/room_suites', function () {
    $username = session('username', 'Guest');
    return view('room_suites', ['username' => $username]);
})->name('room_suites');

Route::get('/room_featured', function () {
    $username = session('username', 'Guest');
    return view('room_featured', ['username' => $username]);
})->name('room_featured');

Route::get('/amenities', function () {
    $username = session('username', 'Guest');
    return view('amenities', ['username' => $username]);
})->name('amenities');
```

These routes manage different pages in a Laravel app. They check the session for a username, defaulting to 'Guest' if none is set, and then show the corresponding page with the username.

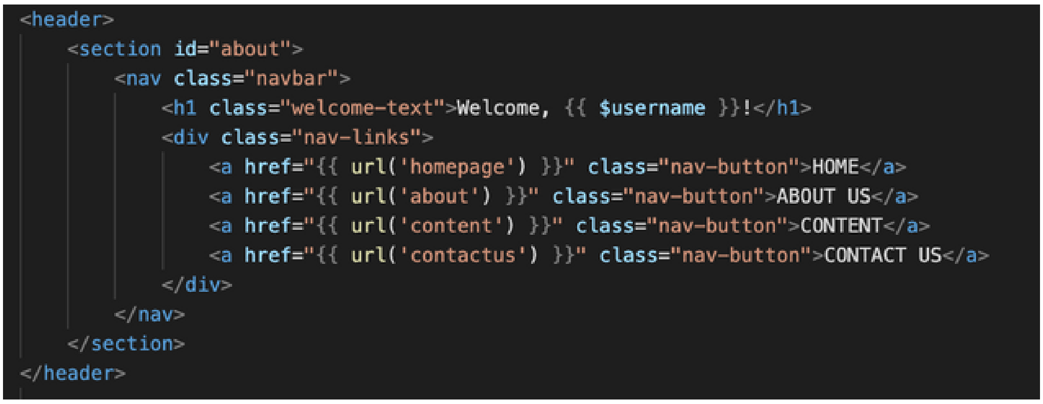
USERNAME_FORM



```html
<body>
    <div class="container">
        <h1>Enter Your Username</h1>
        <h2>Username only accepts alphabetic characters (a-z, A-Z)</h2>
        <form action="{{ url('/user') }}" method="GET">
            @csrf
            <label for="username">Username:</label>
            <input type="text" id="username" name="username" value="{{ request('username') }}" pattern="[a-zA-Z]+" title="Please enter only alphabetic characters.">
            <button type="submit">Submit</button>
        </form>
    </div>
</body>
```

Username_form allow users to type in their username. It has a heading telling users to enter their username and a note saying it should only include letters (a-z, A-Z). The form will send the username to the `/user` URL when submitted. It includes a security token to keep the form safe. The input field only accepts letters and will show a message if invalid characters are used. If the user has already entered a username, it will appear in the field. There’s also a button to submit the form.

HOMEPAGE NAVIGATION



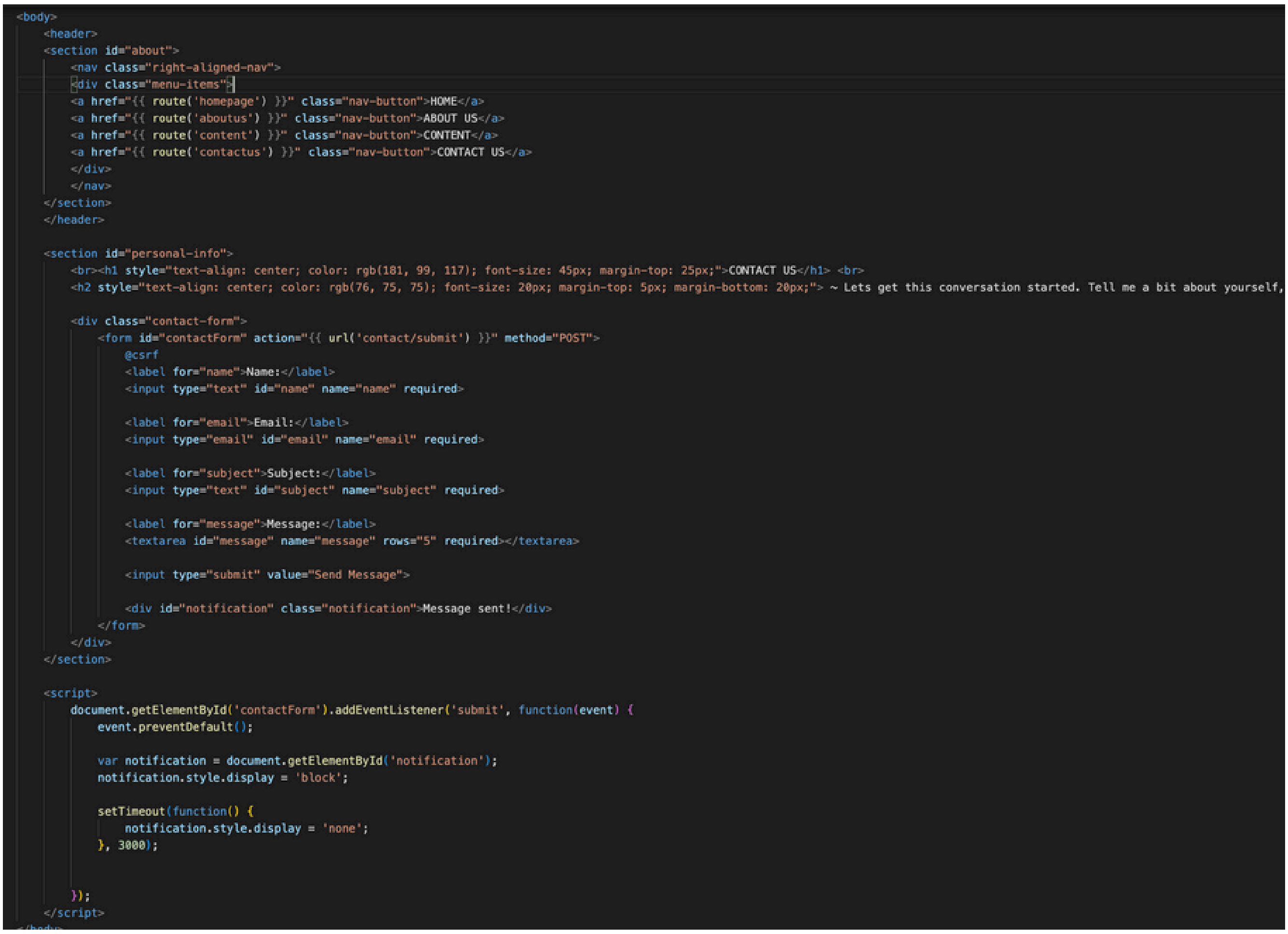
```
<header>
    <section id="about">
        <nav class="navbar">
            <h1 class="welcome-text">Welcome, {{ $username }}!</h1>
            <div class="nav-links">
                <a href="{{ url('homepage') }}" class="nav-button">HOME</a>
                <a href="{{ url('about') }}" class="nav-button">ABOUT US</a>
                <a href="{{ url('content') }}" class="nav-button">CONTENT</a>
                <a href="{{ url('contactus') }}" class="nav-button">CONTACT US</a>
            </div>
        </nav>
    </section>
</header>
```

Added the Welcome Text

The line `<h1 class="welcome-text">Welcome, {{ $username }}!</h1>` displays a heading that greets the user by name, with the actual username being inserted into the message when the page is rendered.

SOURCE CODE FOR THE CONTACT US FORM

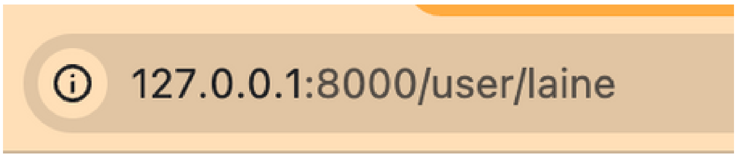
This code creates a webpage with a navigation bar and a contact form. The navigation bar includes links to different pages using Laravel routes. The contact form collects user information and displays a "Message sent!" notification when the form is submitted, which disappears after three seconds.



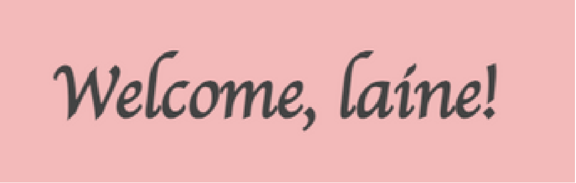
```
<body>
    <header>
    <section id="about">
        <nav class="right-aligned-nav">
        <div class="menu-items">
        <a href="{{ route('homepage') }}" class="nav-button">HOME</a>
        <a href="{{ route('aboutus') }}" class="nav-button">ABOUT US</a>
        <a href="{{ route('content') }}" class="nav-button">CONTENT</a>
        <a href="{{ route('contactus') }}" class="nav-button">CONTACT US</a>
        </div>
        </nav>
    </section>
    </header>

    <section id="personal-info">
        <br><h1 style="text-align: center; color: rgb(181, 99, 117); font-size: 45px; margin-top: 25px;">CONTACT US</h1> <br>
        <h2 style="text-align: center; color: rgb(76, 75, 75); font-size: 20px; margin-top: 5px; margin-bottom: 20px;"> ~ Lets get this conversation started. Tell me a bit about yourself,

        <div class="contact-form">
            <form id="contactForm" action="{{ url('contact/submit') }}" method="POST">
                @csrf
                <label for="name">Name:</label>
                <input type="text" id="name" name="name" required>

                <label for="email">Email:</label>
                <input type="email" id="email" name="email" required>

                <label for="subject">Subject:</label>
                <input type="text" id="subject" name="subject" required>

                <label for="message">Message:</label>
                <textarea id="message" name="message" rows="5" required></textarea>

                <input type="submit" value="Send Message">

                <div id="notification" class="notification">Message sent!</div>
            </form>
        </div>
    </section>

    <script>
        document.getElementById('contactForm').addEventListener('submit', function(event) {
            event.preventDefault();

            var notification = document.getElementById('notification');
            notification.style.display = 'block';

            setTimeout(function() {
                notification.style.display = 'none';
            }, 3000);


        });
    </script>
</body>
```

RESULTS

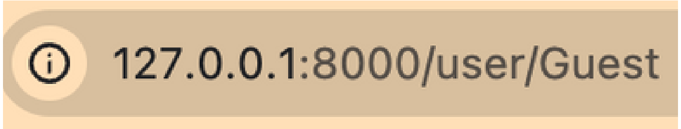
It displays a welcome message that includes the username. The URL will be modified to `user/[username]`, and if the user doesn't input a name, it will automatically be saved as 'guest'.



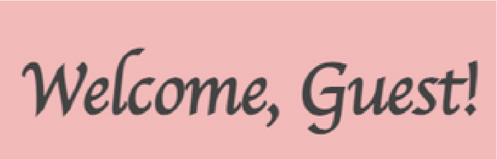
127.0.0.1:8000/user/laine



Welcome, laine!

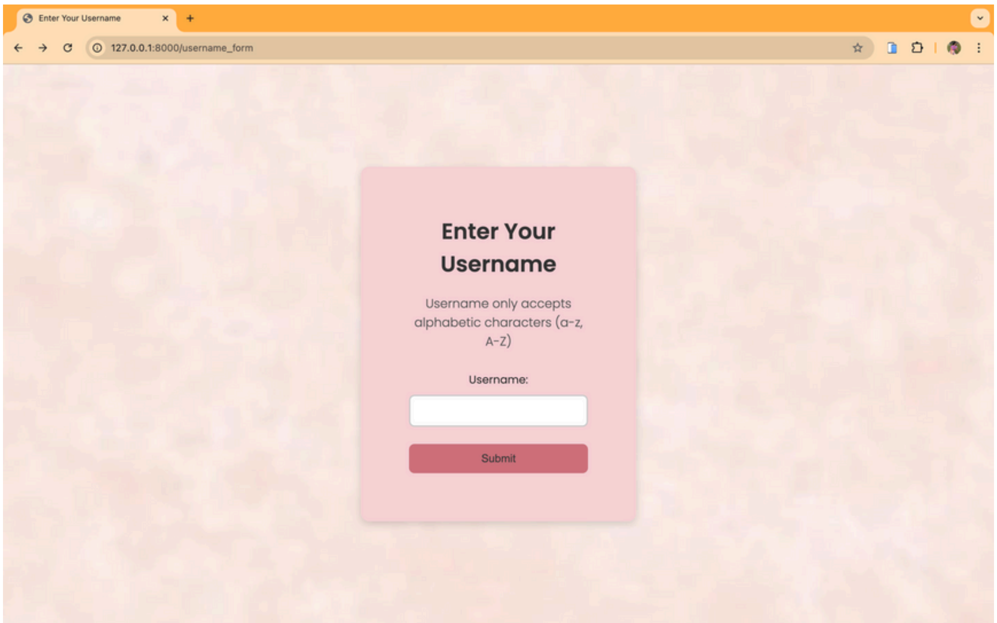


127.0.0.1:8000/user/Guest



Welcome, Guest!

USERNAME FORM



Enter Your Username

127.0.0.1:8000/username_form

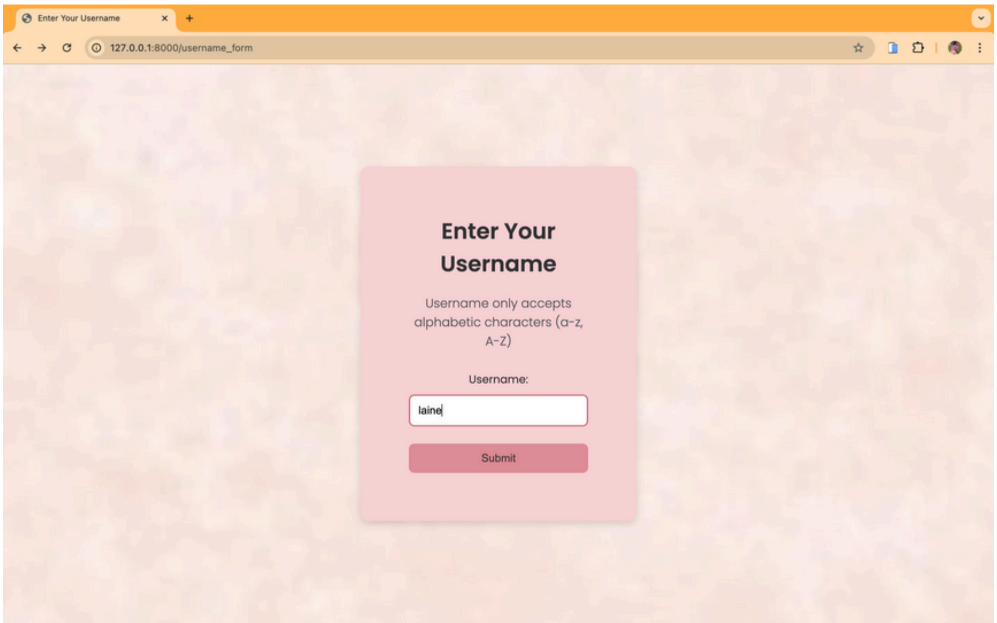
Enter Your Username

Username only accepts alphabetic characters (a-z, A-Z)

Username:

Submit

Here, the system will ask the user to enter a username



Enter Your Username

127.0.0.1:8000/username_form

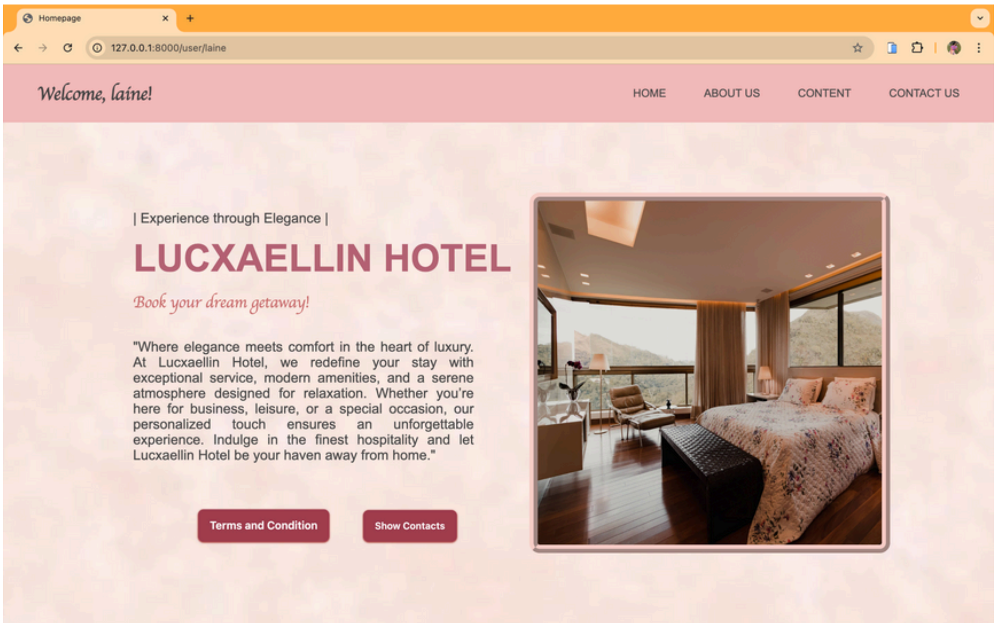
Enter Your Username

Username only accepts alphabetic characters (a-z, A-Z)

Username: laine

Submit

If the user input her username and once it clicked submit,



Homepage

127.0.0.1:8000/user/laine

Welcome, laine!

HOME ABOUT US CONTENT CONTACT US

| Experience through Elegance |

LUCXAELLIN HOTEL

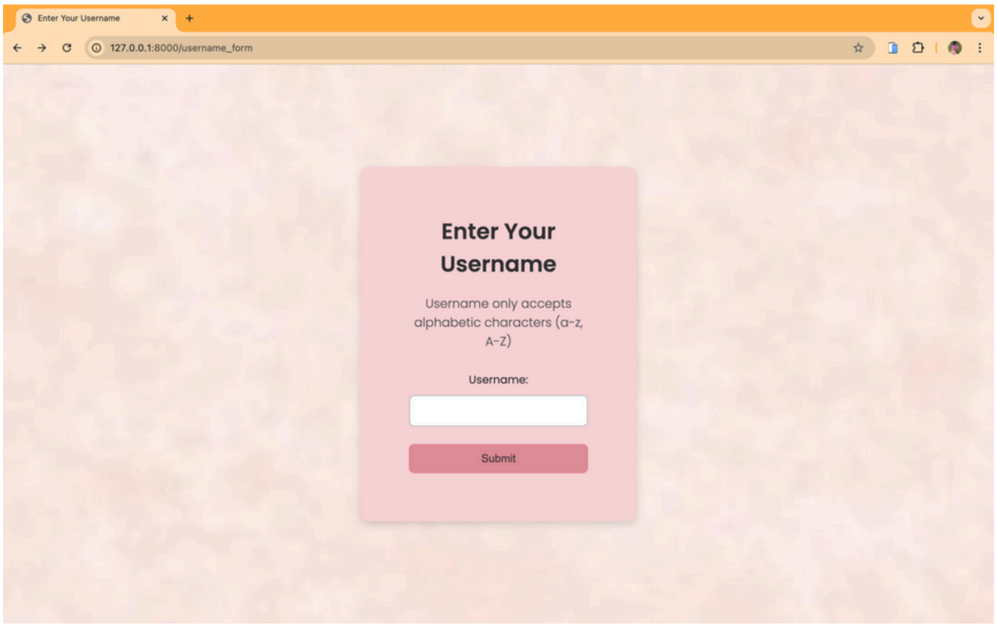
Book your dream getaway!

"Where elegance meets comfort in the heart of luxury. At Lucxaellin Hotel, we redefine your stay with exceptional service, modern amenities, and a serene atmosphere designed for relaxation. Whether you're here for business, leisure, or a special occasion, our personalized touch ensures an unforgettable experience. Indulge in the finest hospitality and let Lucxaellin Hotel be your haven away from home."

Terms and Condition

Show Contacts

The system will automatically direct the user to the homepage, with her username display on the left side of the page. The url on the otherhand, will also save the user username, making it /user/laine.

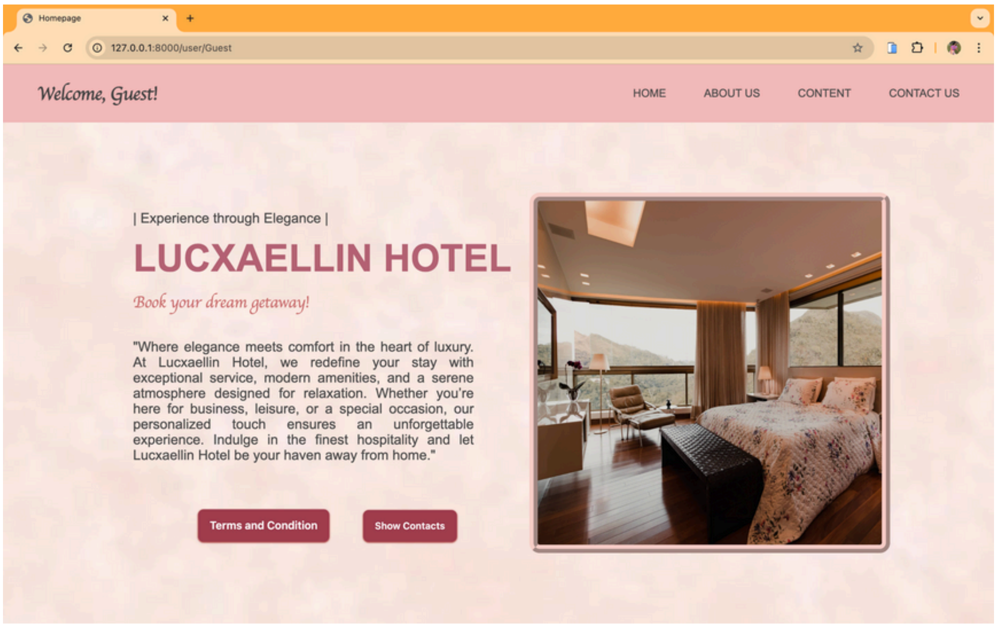


Enter Your Username

Username only accepts alphabetic characters (a-z, A-Z)

Username:

But, when the user don't want to input a username and just clicked submit,



Welcome, Guest!

| Experience through Elegance |

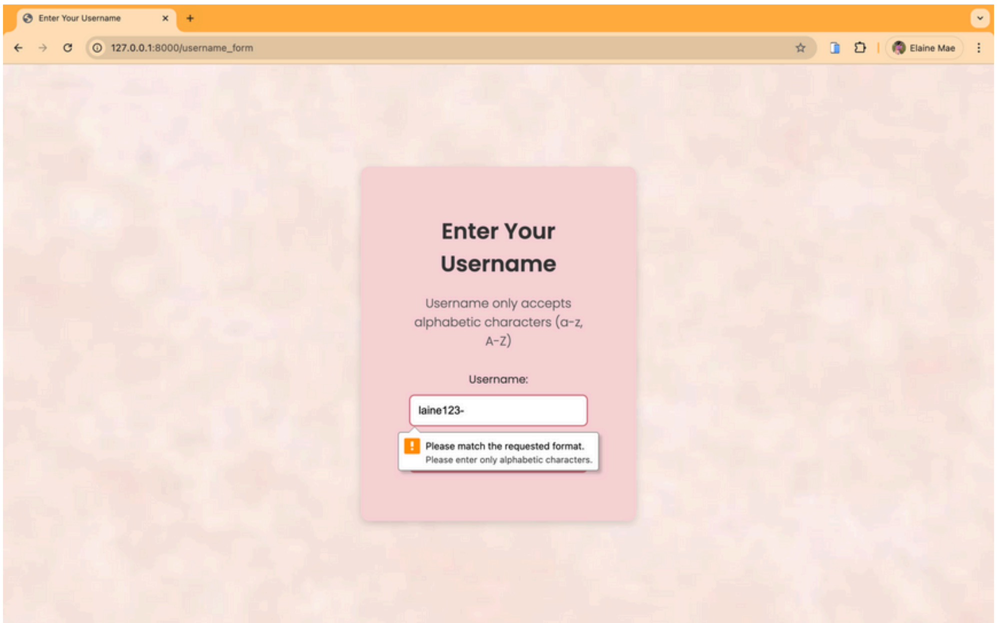
LUCXAELLIN HOTEL

Book your dream getaway!

"Where elegance meets comfort in the heart of luxury. At Lucxaellin Hotel, we redefine your stay with exceptional service, modern amenities, and a serene atmosphere designed for relaxation. Whether you're here for business, leisure, or a special occasion, our personalized touch ensures an unforgettable experience. Indulge in the finest hospitality and let Lucxaellin Hotel be your haven away from home."

The system will stoll automatically direct the user to the homepage, but with the username saved as Guest and the url is also /user/Guest.

DATA VALIDATION FOR USERNAME



Enter Your Username

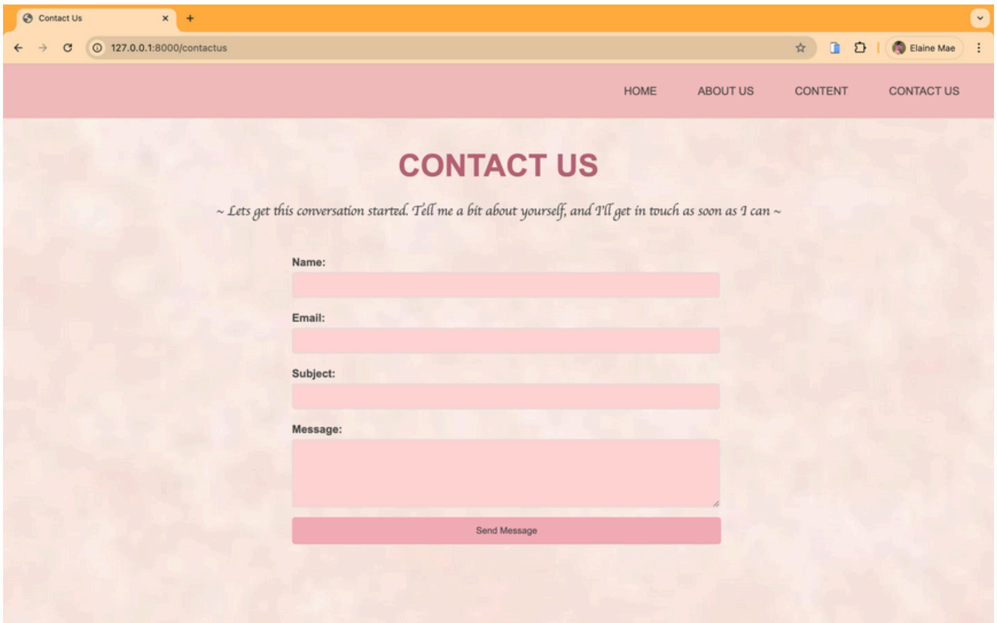
Username only accepts alphabetic characters (a-z, A-Z)

Username: laine123-

! Please match the requested format.
Please enter only alphabetic characters.

If the user does not follow the displayed format and does not meet the requirements, the system will notify the user with 'Please match the requested format' and 'Please enter only alphabetic characters.'

CONTACT US



CONTACT US

~ Lets get this conversation started. Tell me a bit about yourself, and I'll get in touch as soon as I can ~

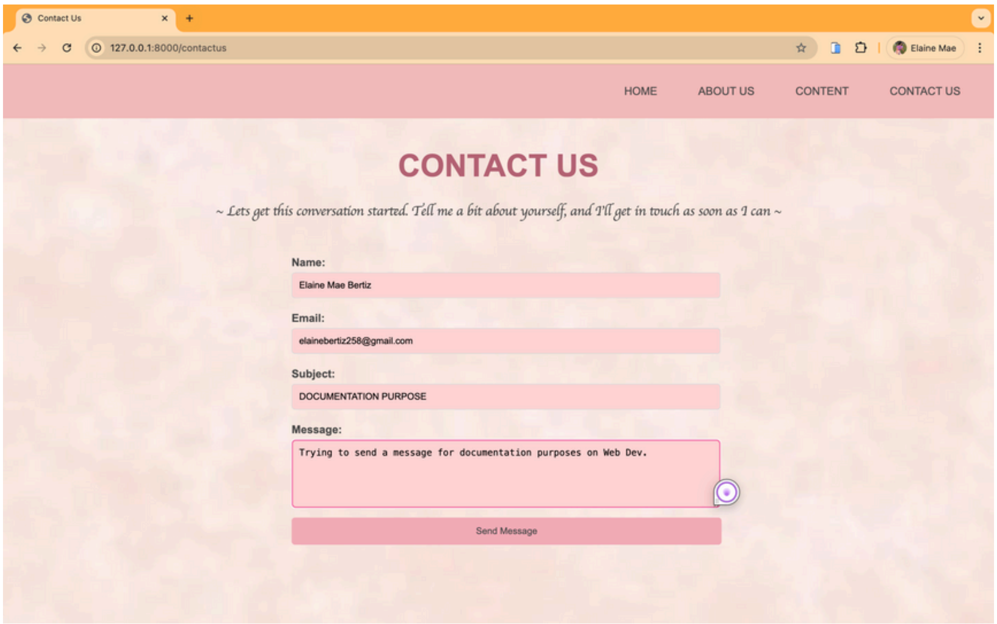
Name:

Email:

Subject:

Message:

'Contact Us' is added as a navigation item. When you click it, you will be directed to the Contact Form. The Contact Form consists of fields for Name, Email, Subject, and Message for the user to fill out.



CONTACT US

~ Lets get this conversation started. Tell me a bit about yourself, and I'll get in touch as soon as I can ~

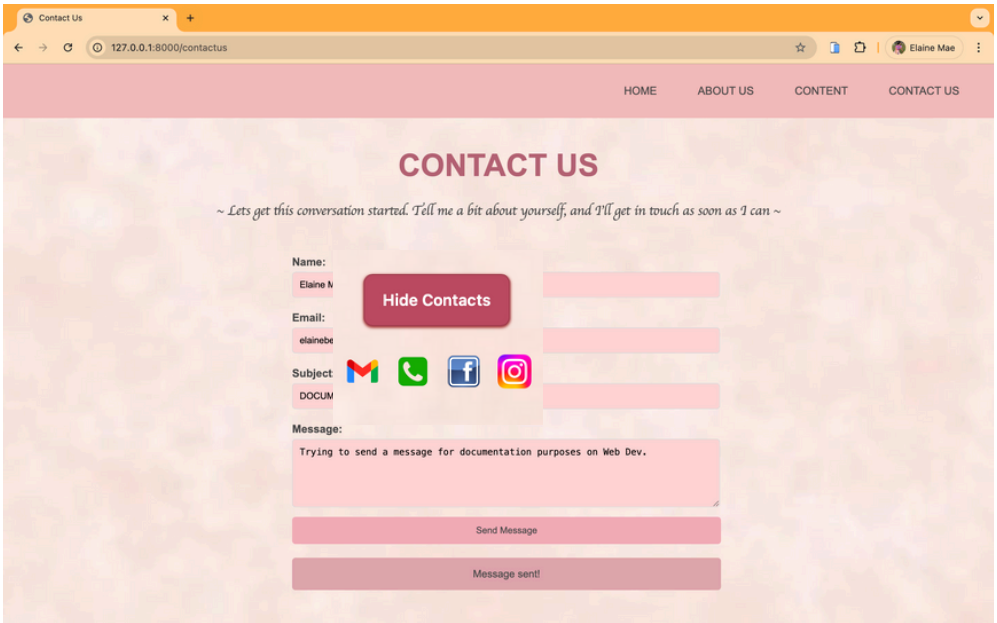
Name: Elaine Mae Bertiz

Email: elainebertiz258@gmail.com

Subject: DOCUMENTATION PURPOSE

Message: Trying to send a message for documentation purposes on Web Dev.

Once the user has filled out the form, they can send the message. All text boxes must be filled in, or the message will not be sent.



CONTACT US

~ Lets get this conversation started. Tell me a bit about yourself, and I'll get in touch as soon as I can ~

Name: Elaine M

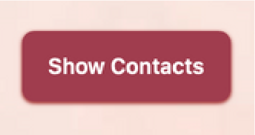
Email: elainebe

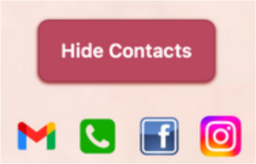
Subject: DOCUM

Message: Trying to send a message for documentation purposes on Web Dev.

Message sent!

Once the user clicks submit, the message will be sent automatically, and a notification saying 'Message sent!' will appear to inform the user that the message has been sent.





I also added a button that allows users to access the social media account and use it as a contact method.