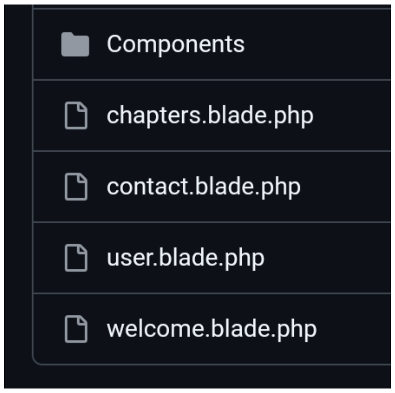
RIALYN CAMPOSANO BSIT-3C WEB DEVELOPMENT

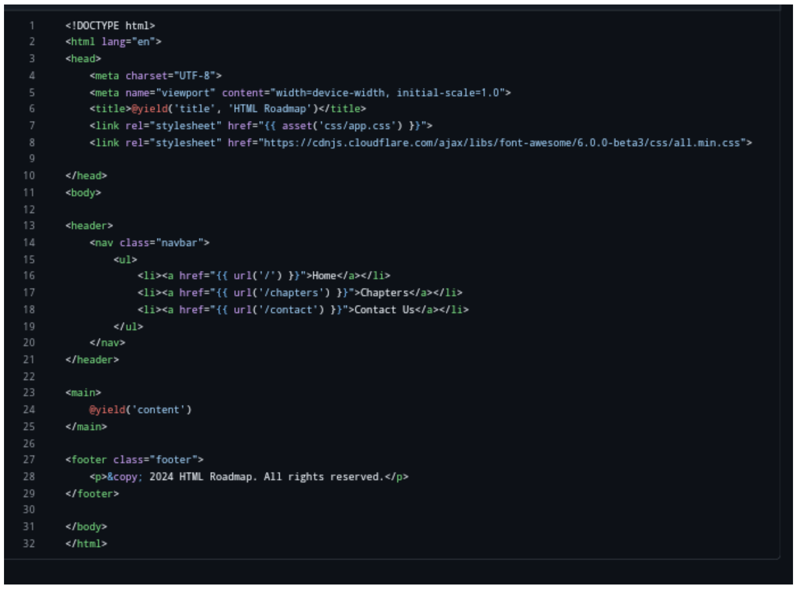
DOCUMENTATION

Folder Structure

```
/Resources
    /Views
        /components
            - Layout.blade.php
        - welcome.blade.php
        - chapters.blade.php
        - contact.blade.php
        - user.blade.php
/routes
    - web.php
```



Layout.blade.php



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>@yield('title', 'HTML Roadmap')</title>
    <link rel="stylesheet" href="{{ asset('css/app.css') }}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">

</head>
<body>

<header>
    <nav class="navbar">
        <ul>
            <li><a href="{{ url('/') }}">Home</a></li>
            <li><a href="{{ url('/chapters') }}">Chapters</a></li>
            <li><a href="{{ url('/contact') }}">Contact Us</a></li>
        </ul>
    </nav>
</header>

<main>
    @yield('content')
</main>

<footer class="footer">
    <p>&copy; 2024 HTML Roadmap. All rights reserved.</p>
</footer>

</body>
</html>
```

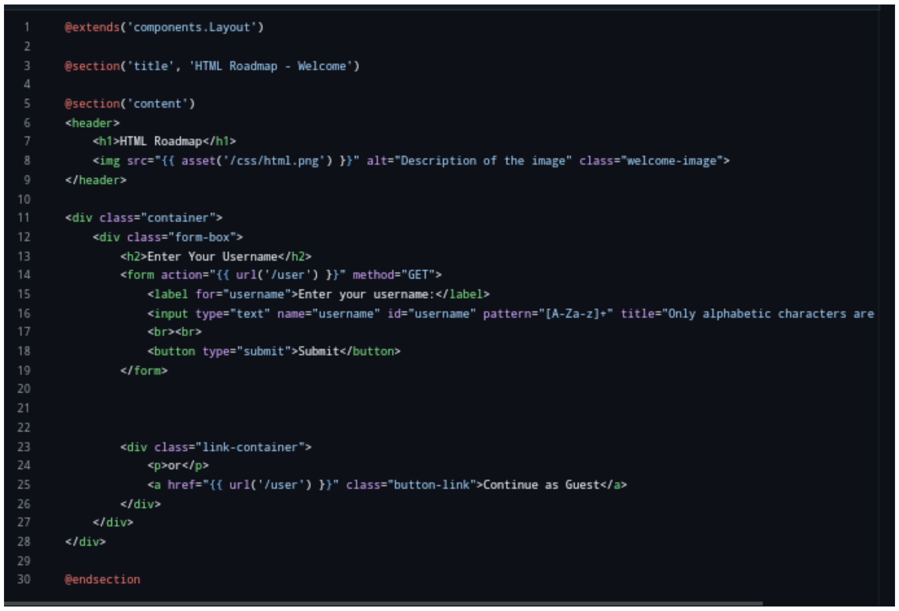
This layout file provides a consistent structure for all pages. It includes a header with navigation links, a main section where page-specific content will be inserted, and a footer with copyright information. The use of @yield allows for dynamic title

• The @yield('title', 'HTML Roadmap') directive allows each view to set a specific title. If no title is provided, it defaults to "HTML Roadmap."

• {{ asset('css/app.css') }} links to a local CSS file for custom styles.

• The <main> tag contains the @yield('content') directive, which will be replaced by the content from each specific view. This makes the layout dynamic, allowing different views to insert their unique content.

welcome.blade.php



```
@extends('components.Layout')

@section('title', 'HTML Roadmap - Welcome')

@section('content')
<header>
    <h1>HTML Roadmap</h1>
    <img src="{{ asset('/css/html.png') }}" alt="Description of the image" class="welcome-image">
</header>

<div class="container">
    <div class="form-box">
        <h2>Enter Your Username</h2>
        <form action="{{ url('/user') }}" method="GET">
            <label for="username">Enter your username:</label>
            <input type="text" name="username" id="username" pattern="[A-Za-z]+" title="Only alphabetic characters are
            <br><br>
            <button type="submit">Submit</button>
        </form>



        <div class="link-container">
            <p>or</p>
            <a href="{{ url('/user') }}" class="button-link">Continue as Guest</a>
        </div>
    </div>
</div>

@endsection
```

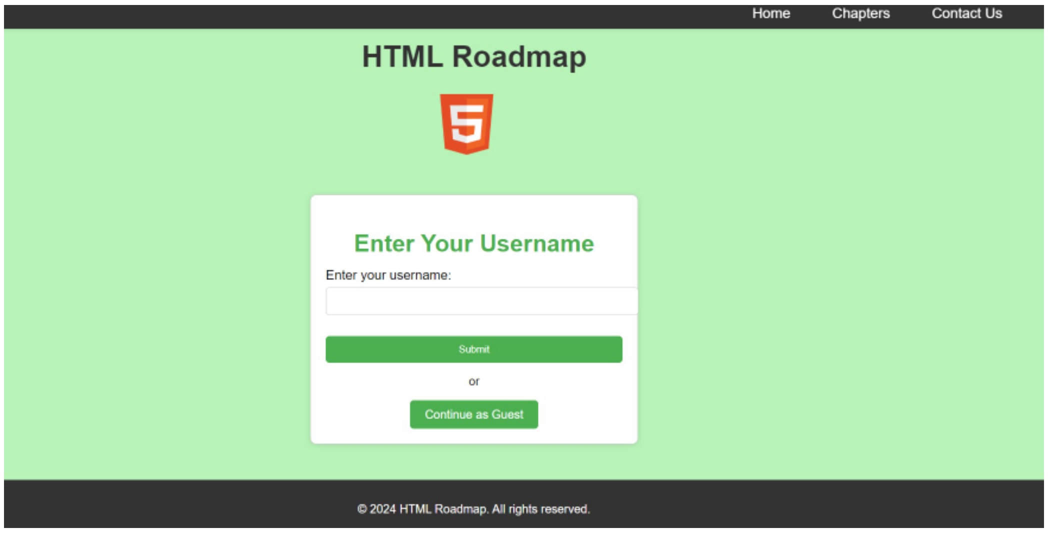
The welcome page serves as the landing page of the HTML Roadmap project. It features a title, a welcoming header, and a form that allows users to enter their username. Users can also choose to continue as a guest.

• @extends('components.Layout') - this directive indicates that the welcome page inherits from a layout file, ensuring a consistent structure across all pages.

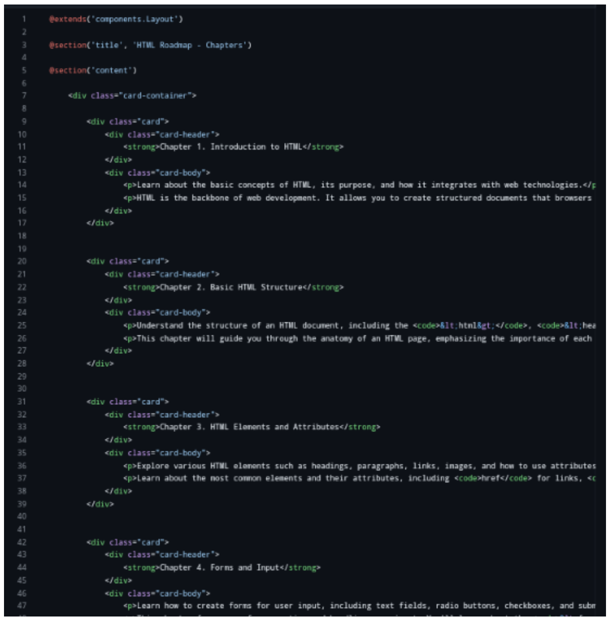
• @section('title', 'HTML Roadmap - Welcome') - defines a section named title, which will set the page's title in the layout. If not specified, it defaults to "HTML Roadmap".

• @section('content') - marks the beginning of the content section that will be injected into the layout file at the @yield('content') placeholder.

• <form action="{{ url('/user') }}" method="GET"> - creates a form that submits user input to the /user route using the GET method. The url() helper generates the appropriate URL.



chapters.blade.php

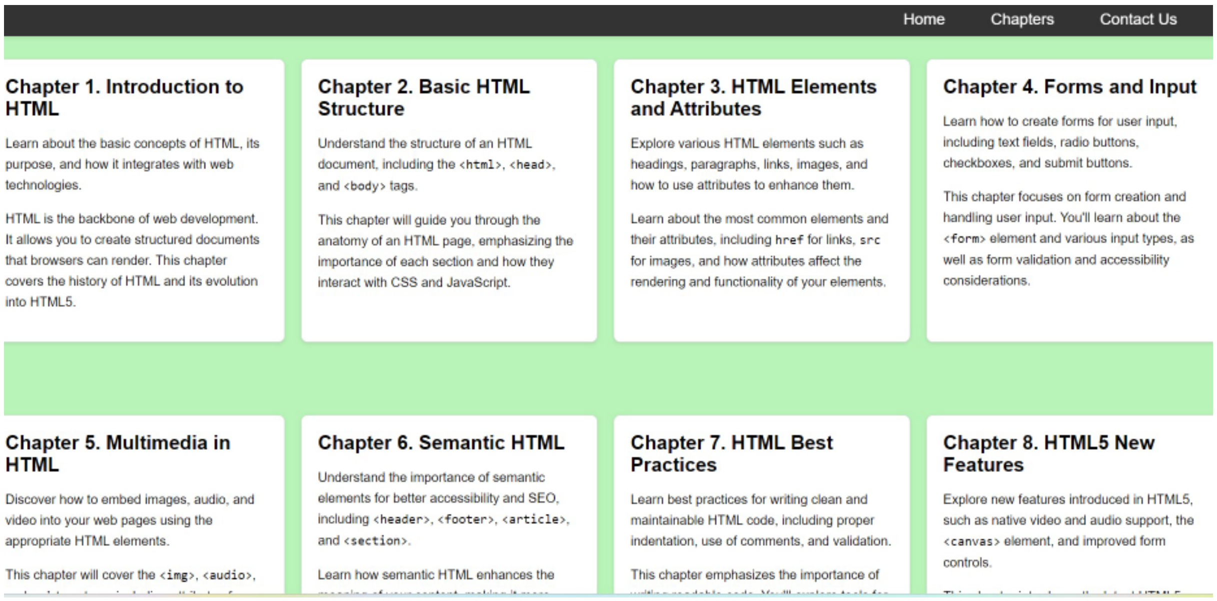


```
@extends('components.Layout')

@section('title', 'HTML Roadmap - Chapters')

@section('content')

    <div class="card-container">

        <div class="card">
            <div class="card-header">
                <strong>Chapter 1. Introduction to HTML</strong>
            </div>
            <div class="card-body">
                <p>Learn about the basic concepts of HTML, its purpose, and how it integrates with web technologies.</p
                <p>HTML is the backbone of web development. It allows you to create structured documents that browsers
            </div>
        </div>


        <div class="card">
            <div class="card-header">
                <strong>Chapter 2. Basic HTML Structure</strong>
            </div>
            <div class="card-body">
                <p>Understand the structure of an HTML document, including the <code>&lt;html&gt;</code>, <code>&lt;hea
                <p>This chapter will guide you through the anatomy of an HTML page, emphasizing the importance of each
            </div>
        </div>


        <div class="card">
            <div class="card-header">
                <strong>Chapter 3. HTML Elements and Attributes</strong>
            </div>
            <div class="card-body">
                <p>Explore various HTML elements such as headings, paragraphs, links, images, and how to use attributes
                <p>Learn about the most common elements and their attributes, including <code>href</code> for links, <c
            </div>
        </div>


        <div class="card">
            <div class="card-header">
                <strong>Chapter 4. Forms and Input</strong>
            </div>
            <div class="card-body">
                <p>Learn how to create forms for user input, including text fields, radio buttons, checkboxes, and subm
```

• @extends('components.Layout') - Inherits from the layout file, ensuring a unified structure and styling across the application.

• @section('title', 'HTML Roadmap - Chapters') - Sets the title of the page specifically for the chapters section, enhancing SEO and user experience.

• @section('content') - Marks the beginning of the content section that will replace the @yield('content') in the layout.



Chapter 1. Introduction to HTML

Learn about the basic concepts of HTML, its purpose, and how it integrates with web technologies.

HTML is the backbone of web development. It allows you to create structured documents that browsers can render. This chapter covers the history of HTML and its evolution into HTML5.

Chapter 2. Basic HTML Structure

Understand the structure of an HTML document, including the `<html>`, `<head>`, and `<body>` tags.

This chapter will guide you through the anatomy of an HTML page, emphasizing the importance of each section and how they interact with CSS and JavaScript.

Chapter 3. HTML Elements and Attributes

Explore various HTML elements such as headings, paragraphs, links, images, and how to use attributes to enhance them.

Learn about the most common elements and their attributes, including `href` for links, `src` for images, and how attributes affect the rendering and functionality of your elements.

Chapter 4. Forms and Input

Learn how to create forms for user input, including text fields, radio buttons, checkboxes, and submit buttons.

This chapter focuses on form creation and handling user input. You'll learn about the `<form>` element and various input types, as well as form validation and accessibility considerations.

Chapter 5. Multimedia in HTML

Discover how to embed images, audio, and video into your web pages using the appropriate HTML elements.

This chapter will cover the `<img>`, `<audio>`,

Chapter 6. Semantic HTML

Understand the importance of semantic elements for better accessibility and SEO, including `<header>`, `<footer>`, `<article>`, and `<section>`.

Learn how semantic HTML enhances the

Chapter 7. HTML Best Practices

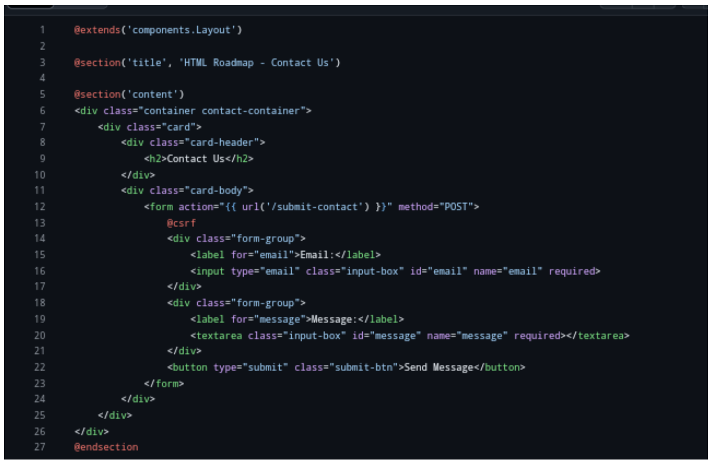
Learn best practices for writing clean and maintainable HTML code, including proper indentation, use of comments, and validation.

This chapter emphasizes the importance of

Chapter 8. HTML5 New Features

Explore new features introduced in HTML5, such as native video and audio support, the `<canvas>` element, and improved form controls.

contact.blade.php



```
@extends('components.Layout')

@section('title', 'HTML Roadmap - Contact Us')

@section('content')
<div class="container contact-container">
    <div class="card">
        <div class="card-header">
            <h2>Contact Us</h2>
        </div>
        <div class="card-body">
            <form action="{{ url('/submit-contact') }}" method="POST">
                @csrf
                <div class="form-group">
                    <label for="email">Email:</label>
                    <input type="email" class="input-box" id="email" name="email" required>
                </div>
                <div class="form-group">
                    <label for="message">Message:</label>
                    <textarea class="input-box" id="message" name="message" required></textarea>
                </div>
                <button type="submit" class="submit-btn">Send Message</button>
            </form>
        </div>
    </div>
</div>
@endsection
```

This view allows users to submit their contact information through a form. It includes CSRF protection for security.

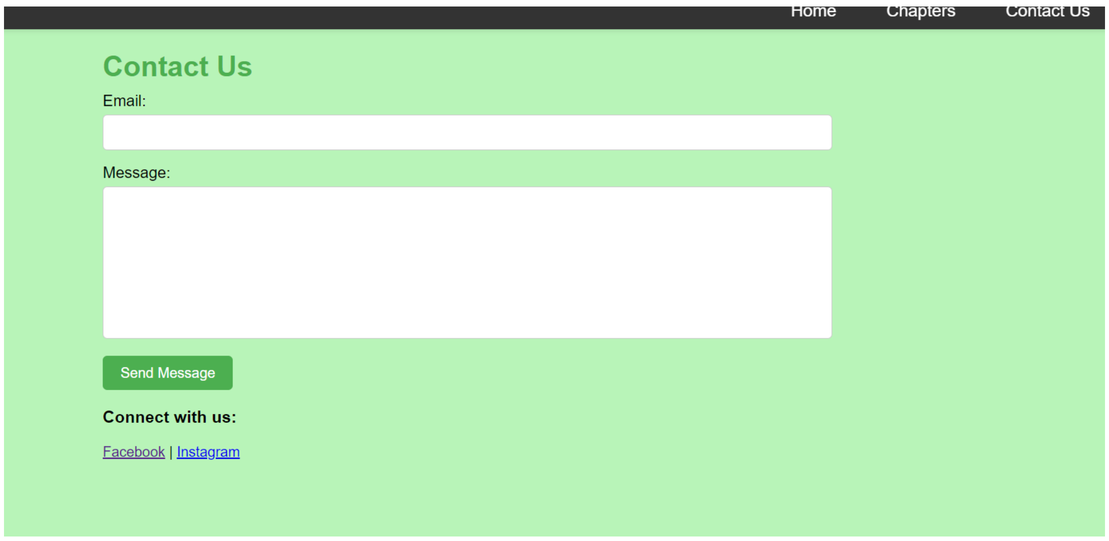
• @extends('components.Layout') - Inherits the base layout, ensuring a consistent structure across all pages.

• @section('title', 'HTML Roadmap - Contact Us') - Sets the page title specifically for the contact section, improving SEO and user navigation.

• @section('content') - Begins the content section that will be injected into the layout's @yield('content').

• <form action="{{ url('/submit-contact') }}" method="POST"> - Defines a form for submitting contact information to the /submit-contact route using the POST method.

• @csrf - Includes a CSRF token for security, protecting against cross-site request forgery attacks.



Contact Us

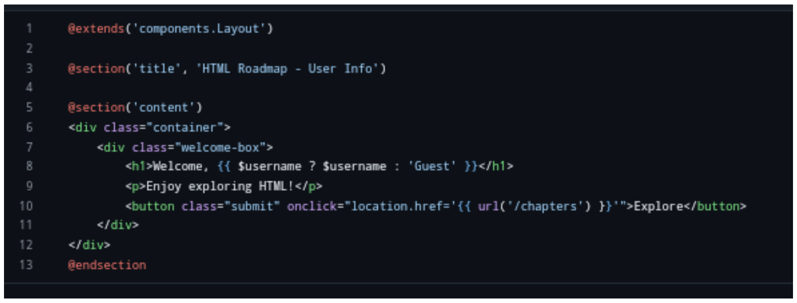
Email:

Message:

Connect with us:

Facebook | Instagram

user.blade.php



```
@extends('components.Layout')

@section('title', 'HTML Roadmap - User Info')

@section('content')
<div class="container">
    <div class="welcome-box">
        <h1>Welcome, {{ $username ? $username : 'Guest' }}</h1>
        <p>Enjoy exploring HTML!</p>
        <button class="submit" onclick="location.href='{{ url('/chapters') }}'">Explore</button>
    </div>
</div>
@endsection
```

This view displays a welcome message to the user based on the provided username or defaults to "Guest." It provides a button to navigate to the chapters.

• **@extends('components.Layout') -** Inherits the layout, ensuring the user page maintains a consistent design with the overall application.

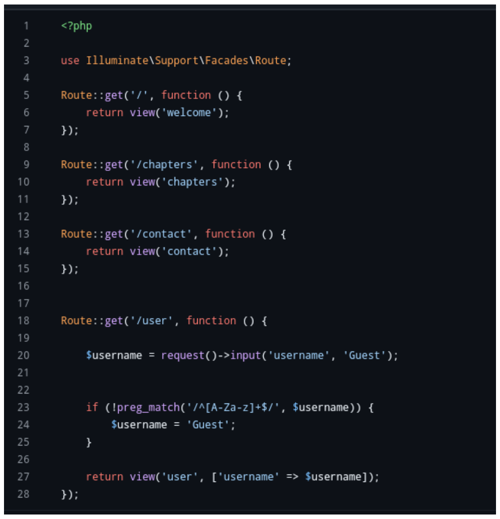
• **@section('title', 'HTML Roadmap - User Info') -** Sets the specific title for the user information page, enhancing SEO and user experience.

• **@section('content') -** Begins the content section to be displayed in the layout's @yield('content').

• **<h1>Welcome, {{ $username ? $username : 'Guest' }}</h1> -** Displays a personalized welcome message. If a username is provided, it shows that; otherwise, it defaults to "Guest."

• **<button class="submit" onclick="location.href='{{ url('/chapters') }}'">Explore</button> -** A button that directs users to the chapters page, facilitating easy navigation and encouraging exploration.

routes/web.php



```php
<?php

use Illuminate\Support\Facades\Route;

Route::get('/', function () {
    return view('welcome');
});

Route::get('/chapters', function () {
    return view('chapters');
});

Route::get('/contact', function () {
    return view('contact');
});


Route::get('/user', function () {

    $username = request()->input('username', 'Guest');


    if (!preg_match('/^[A-Za-z]+$/', $username)) {
        $username = 'Guest';
    }

    return view('user', ['username' => $username]);
});
```

• **use Illuminate\Support\Facades\Route;** - Imports the Route facade, allowing the definition of routes in the application.

• **Route::get('/', function () { return view('welcome'); });** - Defines a route for the homepage (`/`), returning the `welcome` view when accessed.

• **Route::get('/chapters', function () { return view('chapters'); });** - Sets up a route for the chapters page (`/chapters`), returning the `chapters` view.

• **Route::get('/contact', function () { return view('contact'); });** - Establishes a route for the contact page (`/contact`), which returns the `contact` view.

• **Route::get('/user', function () { ... });** - Defines a route for the user page (`/user`):

• **$username = request()->input('username', 'Guest');** - Retrieves the username from the request. If not provided, it defaults to "Guest."

• **if (!preg_match('/^[A-Za-z]+$/', $username)) { $username = 'Guest'; }** - Validates the username to ensure it only contains alphabetic characters. If it doesn't match, it sets the username to "Guest."

• **return view('user', ['username' => $username]);** - Returns the `user` view with the validated username.

Difference Between {{$slot}} and @yield

{{$slot}}: Used primarily in Blade components, allowing for the insertion of dynamic content. It represents the content provided to the component when it is instantiated

@yield: Used in layout files to define sections that can be filled by child views. It creates placeholders for content that will be defined in extending views

Challenges Faced and Resolutions

1. Validation of User Input:

Challenge: Ensuring the user input (username) adhered to specific validation rules, accepting only alphabetic characters.

Resolution: A regular expression (`preg_match('/^[A-Za-z]+$/', $username)`) was implemented in the user route to validate the username input. If the input did not match, the system defaulted the username to "Guest."

2. Maintaining Consistent Layout Across Pages:

Challenge: Achieving a uniform layout structure (header, navigation, and footer) across multiple pages without duplicating code.

Resolution: Created a central layout file (`Layout.blade.php`) and utilized Blade's `@extends` and `@yield` directives to extend this layout in individual views. This ensured all pages had a consistent design while allowing page-specific content to be dynamically inserted.

3. Inserting Dynamic Content:

Challenge: Incorporating page-specific content into a fixed layout in a modular way.

Resolution: Utilized Blade's `@section` and `@yield` features to insert unique content for each page while retaining a shared layout.