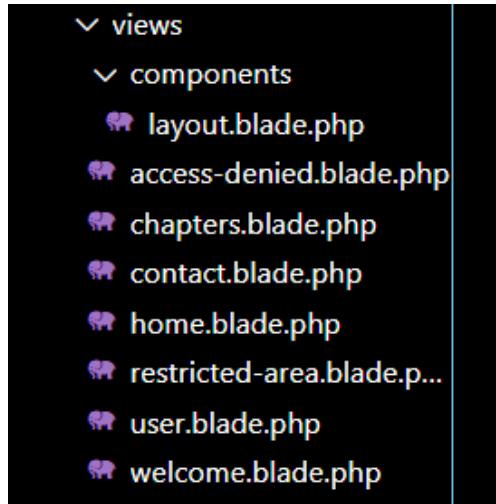


## DOCUMENTATION:

## BLADE TEMPLATE FILES:

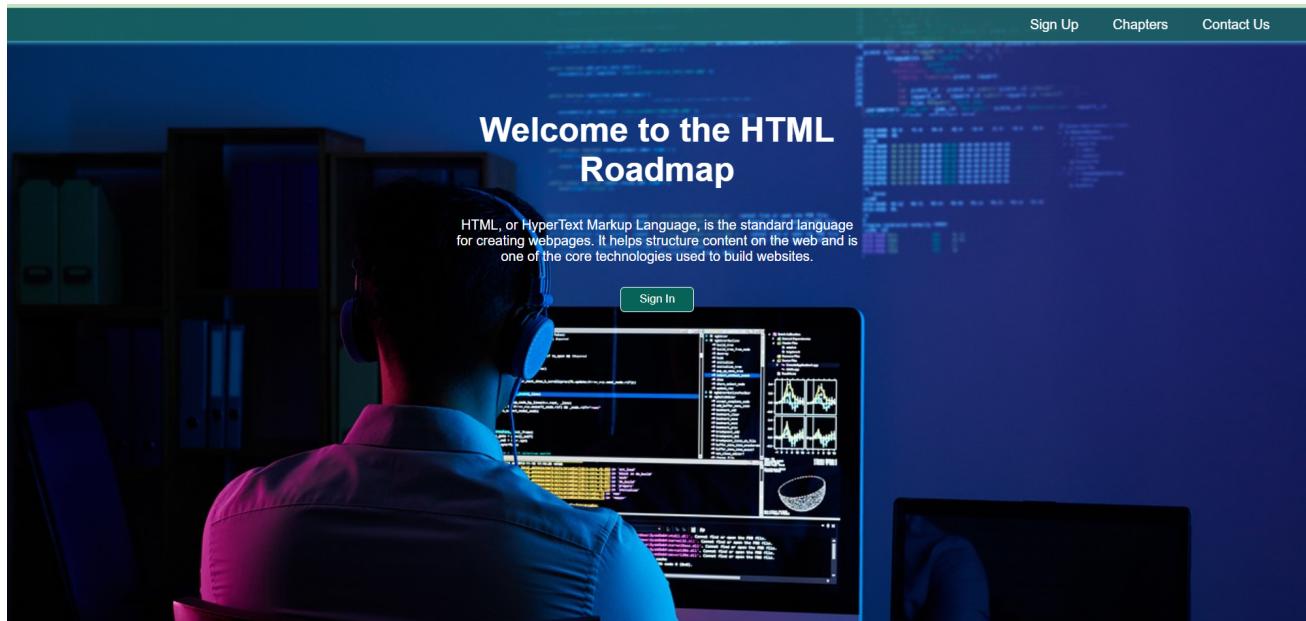


## Home Page (home.blade.php)

- Shows the main landing page for all users.
- Accessible to all users.

```

1  @extends('components.Layout')
2
3  @section('title', 'HTML Roadmap - Home')
4
5  @section('content')
6  <div class="container home-container">
7    <header>
8      <h1 class="header-roadmap-title">Welcome to the HTML Roadmap</h1>
9    </header>
10   <div class="introduction">
11     <p>
12       HTML, or HyperText Markup Language, is the standard language for creating webpages.
13       It helps structure content on the web and is one of the core technologies used to build websites.
14     </p>
15     <form action="{{ url('/welcome') }}" method="get">
16       <button type="submit" class="btn btn-primary sign-in-button">Sign In</button>
17     </form>
18   </div>
19   @endsection
20
21   @section("no_footer", true)
22
  
```



## Welcome Page (welcome.blade.php)

- Contains the form for submitting the user's age.
- Collects the age and submits it via POST to the /store-age route.

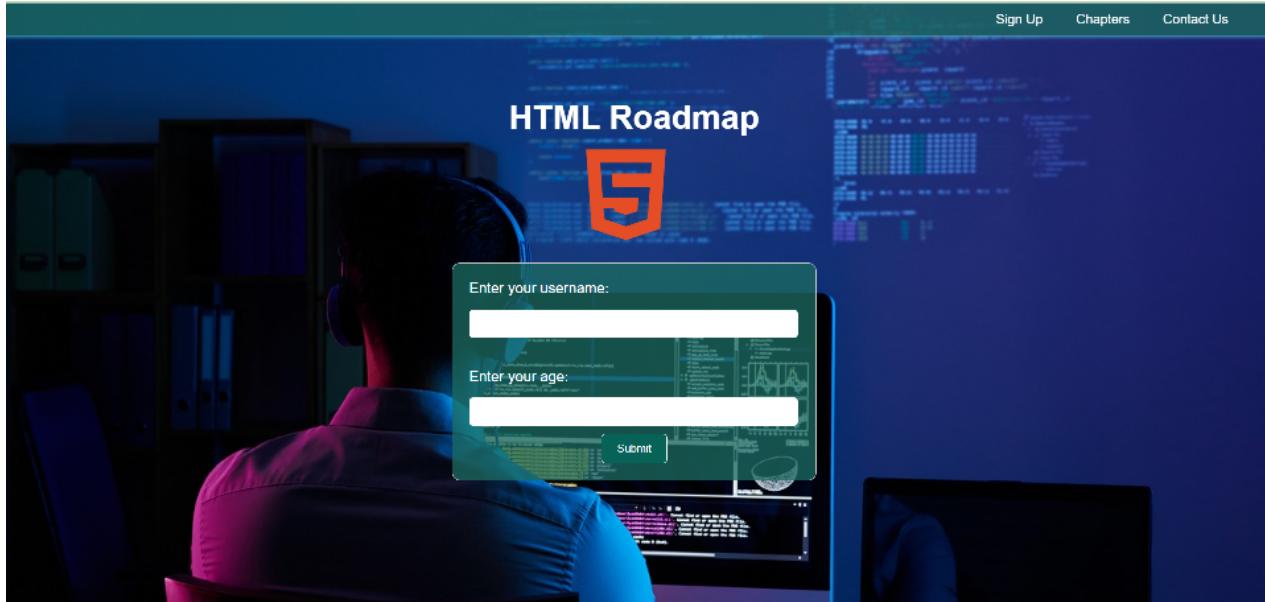
```
@extends('components.Layout')

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

@section('content')
<header>
  <h1 class="header-roadmap-title">HTML Roadmap</h1>
  
</header>

<div class="container">
  <div class="form-box">
    <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 allowed" required>
    </form>
    <br><br>
    <form action="{{ route('store.age') }}" method="POST">
      @csrf
      <label for="age">Enter your age:</label>
      <input type="text" name="age" id="age" required min="1" max="100">
    <div class="submit-home">
      <button type="submit">Submit</button>
    </div>
    </form>
  </div>
</div>

@section('no_footer', true)
@endsection
```



## Restricted Area Page (restricted-area.blade.php)

- Only users aged 21 and above can access this page.
- This is a restricted area, which is protected by CheckAge

```
2     @extends('Components.layout')
3
4     @php
5         $title = 'Access Denied';
6     @endphp
7
8     @push('styles')
9         <link href="{{ asset('css/access-denied.css') }}" rel="stylesheet">
10    @endpush
11
12
13    @section('body-class', 'font-sans antialiased dark:bg-black dark:text-white/50')
14
15    @section('main-class', 'main-content')
16
17    @section('content')
18        <div class="blur-background"></div>
19
20        <div class="center-container">
21            <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100" width="100" height="100" class="circle-x-icon">
22                <!-- Outer Circle -->
23                <circle cx="50" cy="50" r="48" class="outer-circle" />
24
25                <!-- X shape -->
26                <line x1="30" y1="30" x2="70" y2="70" class="cross-line" />
27                <line x1="70" y1="30" x2="30" y2="70" class="cross-line" />
28            </svg>
29
30            <h1>Restricted Area</h1>
31            <p>Welcome to the RESTRICTED AREA.</p>
32        </div>
33    @endsection
34
```

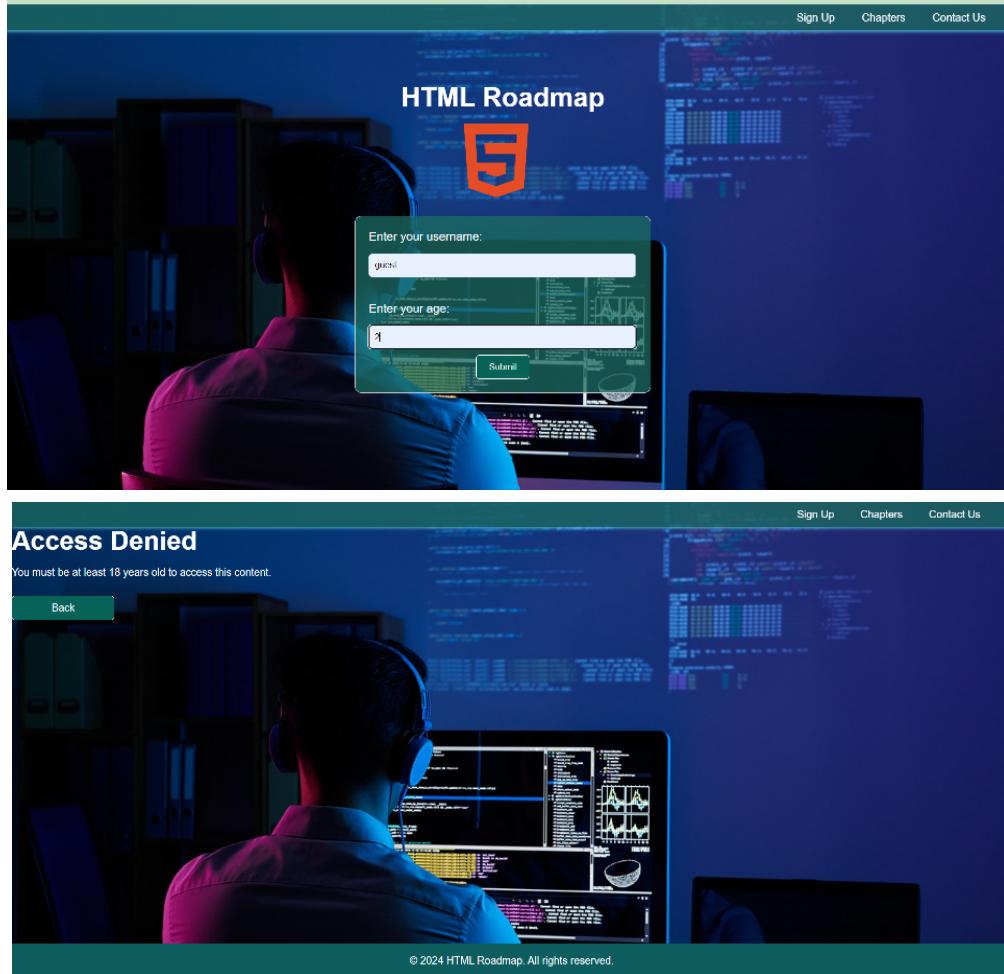
(restricted-2.1.blade.php)

```
1     @extends('Components.layout')
2
3     @section('title', 'Restricted Area - 21+ Only')
4
5     @section('content')
6        <div class="container">
7            <h1>Welcome to the Restricted Area!</h1>
8            <p>You have access to content for users aged 21 and older.</p>
9        </div>
10    @endsection
```

## Access Denied Page (access-denied.blade.php)

- Shows a message for users under 18 who are denied access.
- This page explains that access is restricted based on the user's age.

```
1  @extends('Components.layout')
2
3  @php
4      $title = 'Restricted Area';
5  @endphp
6
7  @push('styles')
8      <link href="{{ asset('css/access-denied.css') }}" rel="stylesheet">
9  @endpush
10
11 @section('body-class', 'font-sans antialiased dark:bg-black dark:text-white/50')
12
13 @section('main-class', 'main-content')
14
15 @section('content')
16     <div class="blur-background"></div>
17
18     <div class="center-container">
19         <h1>Access Denied</h1>
20         <p>You must be at least 18 years old to access this content.</p>
21
22         <!-- Redirect Button -->
23         <a href="{{ url('/')}}" class="redirect-button">Back</a>
24     </div>
25 @endsection
```



The image shows a screenshot of the HTML Roadmap website. At the top, there is a navigation bar with links for 'Sign Up', 'Chapters', and 'Contact Us'. The main content area features a large image of a person wearing headphones and working on a computer, with a 'HTML Roadmap' logo and a large orange '5' icon. A modal window is overlaid on the image, prompting the user to 'Enter your username:' with a text input field containing 'guest' and 'Enter your age:' with a text input field containing '1'. Below this, the word 'Submit' is visible. The bottom portion of the screenshot shows the 'Access Denied' page with the message 'You must be at least 18 years old to access this content.' and a 'Back' button. The footer of the website includes the text '© 2024 HTML Roadmap. All rights reserved.'

## Chapters Page (chapters.blade.php):

- Only users aged 18-20 can access this after passing the CheckAge middleware

The image displays three screenshots of a web application's 'Chapters' page, likely a Laravel application given the file name. The top screenshot shows the 'HTML Basics' section. The middle screenshot shows the 'HTML Basic Elements' section. The bottom screenshot shows a full view of the 'Chapters' page with a sidebar and a main content area.

**Chapters Page (chapters.blade.php):**

- Only users aged 18-20 can access this after passing the CheckAge middleware

**HTML Basics**

Learn the basics of HTML, including structure, elements, and how to create a simple web page. For more information, visit the [HTML Introduction](#).

- Understand the role of HTML in web development. Read more [here](#).
- Learn about tags, elements, and document structure. [Details here](#).
- Explore the various types of HTML documents and their basic components.
- Discover how to set up your first HTML document and use an HTML editor effectively.

1. HTML Introduction

2. HTML Basics

3. HTML Layout

4. HTML Editors

5. HTML Comments

**HTML Basic Elements**

Explore the fundamental HTML elements used to build the structure of a web page, including headings and paragraphs. Read more about [HTML Elements](#).

- Learn how to use headings to structure content. Find out more [here](#).
- Understand the importance of paragraphs for readability. [More details](#).
- Get familiar with the use of lists (ordered and unordered) in HTML documents.
- Discover how to format text using spans and divs for layout control.

1. HTML Headings

2. HTML Paragraphs

**Chapters**

HTML Basics

Learn the basics of HTML, including structure, elements, and how to create a simple web page. For more information, visit the [HTML Introduction](#).

- Understand the role of HTML in web development. Read more [here](#).
- Learn about tags, elements, and document structure. [Details here](#).
- Explore the various types of HTML documents and their basic components.
- Discover how to set up your first HTML document and use an HTML editor effectively.

1. HTML Introduction

2. HTML Basics

3. HTML Layout

4. HTML Editors

5. HTML Comments

Sign Up Chapters Contact Us

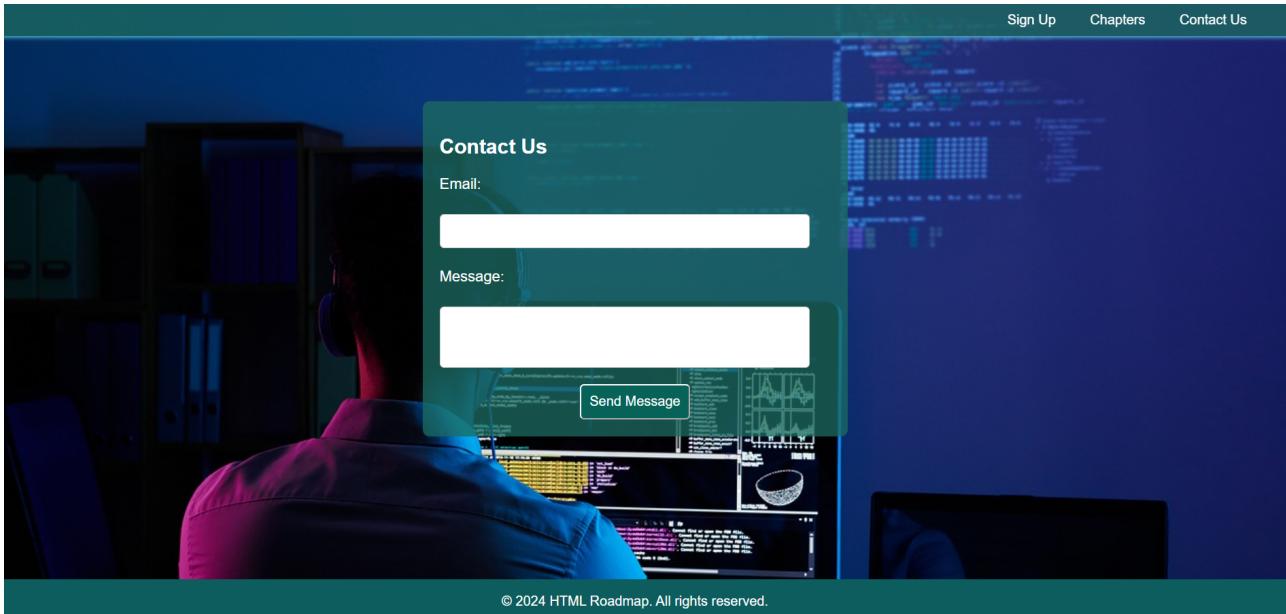
Sign Up Chapters Contact Us

Sign Up Chapters Contact Us

## Contact Page (contact.blade.php):

- This is also a protected page.
- Protected by CheckAge and accessible based on age.

```
1  @extends('components.layout')
2
3  @section('title', 'HTML Roadmap - Contact Us')
4
5  @section('content')
6  <div class="container contact-container">
7      <div class="card">
8          <div class="card-header">
9              <h2>Contact Us</h2>
10         </div>
11         <div class="card-body">
12             <form action="{{ url('/submit-contact') }}" method="POST">
13                 @csrf
14                 <div class="form-group">
15                     <label for="email">Email:</label>
16                     <input type="email" class="input-box" id="email" name="email" required>
17                 </div>
18                 <br>
19                 <div class="form-group">
20                     <label for="message">Message:</label>
21                     <textarea class="input-box" id="message" name="message" required></textarea>
22                 </div>
23                 <div class="submit-home">
24                     <button type="submit" class="submit-btn">Send Message</button>
25                 </div>
26             </form>
27         </div>
28     </div>
29 </div>
30 @endsection
```



## ROUTING CONFIGURATION:

### routes/web.php

```
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use Illuminate\Http\Request;
5  use App\Http\Middleware\CheckAge;
6
7  // Unrestricted Home Route
8  Route::get('/', function () {
9      return view('home');
10 })->name('home');
11
12 // Welcome route for age submission
13 Route::get('/welcome', function () {
14     return view('welcome');
15 })->name('welcome');
16
17 // Restricted area route (21+)
18 Route::get('/restricted-area', function () {
19     return view('restricted-area');
20 })->name('restricted.area');
21
22 // Access Denied route for underage users
23 Route::get('/access-denied', function () {
24     return view('access-denied');
25 })->name('access.denied');
26
27 // Route to handle age submission and redirect based on age
28 Route::post('/store-age', function (Request $request) {
29     $request->validate([
30         'age' => 'required|integer|min:1|max:120',
31     ]);
32
33     // Store the user's age in session
34     session(['age' => $request->input('age')]);
35     $age = $request->input('age');
36
37     // Redirect based on age
38     if ($age < 18) {
39         return redirect()->route('access.denied');
40     } elseif ($age >= 21) {
41         return redirect()->route('restricted.area');
42     } else {
43         return redirect()->route('chapters');
44     }
45 })->name('store.age');
46
47 // Protected Routes Group - Only accessible if CheckAge allows
48 Route::middleware([CheckAge::class])->group(function () {
49     Route::get('/chapters', function () {
50         return view('chapters');
51     })->name('chapters');
52
53     Route::get('/contact', function () {
54         return view('contact');
55     })->name('contact');
56 });
57
```

- 1. Home Route (/): line 8-10** This is an open route that all users can access without any restrictions
- 2. Age Submission Route (/store-age): line 28-45** Collects the user's age, validates it, and stores it in the session.
  - Redirects the user based on their age:
    - Under 18: Redirects to the Access Denied page.
    - 21 and above: Redirects to the Restricted Area.
    - 18-20: Redirects to the Chapters page.
- 3. Protected Routes (Middleware): line 48-55** These routes are restricted and protected by the CheckAge middleware.
  - Only users with the correct age can access the Chapters and Contact pages.
- 4. Access Denied Route (/access-denied): line 23-25** This page is shown to users who are under 18 and denied access

## Registration and Use of Global & Route-Specific Middleware

### (Kernel.php)

```
1  <?php
2
3  namespace App\Http;
4
5  use Illuminate\Foundation\Http\Kernel as HttpKernel;
6
7  class Kernel extends HttpKernel
8  {
9      protected $routeMiddleware = [
10          // ...
11          'CheckAge' => \App\Http\Middleware\CheckAge::class,
12      ];
13     protected $middlewareGroups = [
14         'web' => [
15             // ...
16             \App\Http\Middleware\LogRequests::class,
17         ],
18         // ...
19     ];
20
21 }
22 }
```

#### Global Middleware (LogRequests): line 13-17

- This middleware logs every request made to the application, helping track user activity and diagnose issues
- The LogRequests middleware is added under the 'web' group. This ensures that the middleware is globally applied to all routes under the 'web' middleware group, logging every incoming request made to the application

#### Route-Specific Middleware (CheckAge): line 9-12

- The CheckAge middleware is registered in the routeMiddleware array. This allows it to be applied to specific routes by using the 'CheckAge' key. This middleware controls access to specific routes based on the user's age
- If the age is less than 18, the user is redirected to the access-denied page.

## Middleware Parameters

- Our CheckAge middleware does not accept parameters like 18 and 21. It simply retrieves the age from the session and uses conditional logic to determine the redirection logic.

The CheckAge middleware can be modified to accept parameters like minimum and maximum age: **line 17-33 and line 39**

```
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7
8  class CheckAge
9  {
10     /**
11      * Handle an incoming request.
12      *
13      * @param \Illuminate\Http\Request $request
14      * @param \Closure $next
15      * @return mixed
16      */
17     public function handle(Request $request, Closure $next)
18     {
19         $age = session('age'); // Retrieve age from session
20
21         // If age is not set, redirect to welcome to input age
22         if (is_null($age)) {
23             return redirect('/welcome');
24         }
25
26         // Redirect based on age
27         if ($age < 18) {
28             return redirect('/access-denied');
29         } elseif ($age >= 21) {
30             // Check if user has already visited the restricted area
31             if (!session('visited_restricted_area')) {
32                 // Set session variable to prevent future redirects to restricted-area
33                 session(['visited_restricted_area' => true]);
34                 return redirect('/restricted-area');
35             }
36         }
37
38         // If age is 18-20, allow access to the requested page
39         return $next($request);
40     }
41 }
```

Apply the middleware with parameters in routes:

```
Route::get('/restricted', function () {
    return view('restricted');
})->middleware('CheckAge:18,21');
```

## Terminable Middleware

- special type of middleware that can perform additional actions after the response has been sent to the client

**Purpose:** Useful for logging, cleanup, or other actions that need to occur after the request lifecycle is complete

### Example:

The LogRequests middleware can implement a terminate method to log response details after the response is sent: **line 9-23**

```
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Closure;
6  use Illuminate\Http\Request;
7  use Symfony\Component\HttpFoundation\Response;
8  use Illuminate\Support\Facades\Log;
9
10 class LogRequests
11 {
12     /**
13      * Handle an incoming request.
14      *
15      * @param \Illuminate\Http\Request $request
16      * @param \Closure $next
17      * @return mixed
18      */
19     public function handle(Request $request, Closure $next)
20     {
21         $logData = '[' . now() . ']' . $request->method() . ' ' . $request->fullUrl();
22         Log::channel('custom')->info($logData);
23         return $next($request);
24     }
25 }
```

This logs the HTTP response status, providing insights into the request lifecycle