Karl Christian O. Carlos
BSIT - 3C

Documentation for Web Development 2

Layout File
- Serves as the template for different views to reduce code redundancy and makes the code consistent in each view. Using layouts can help maintainability because you can reuse the navbar in every view for code cleanliness.

```
Layout.blade.php  M  ×
C: > Users > karlc > Documents > jhona > resources > views > Components > Layout.blade.php
28    <body>
29        <!-- Navbar -->
30        <div class="navbar-container">
31            <nav class="navbar navbar-expand-lg navbar-light bg-light">
32                <div class="container">
33                    <a class="navbar-brand" href="{{ url('/') }}">Mac and Gab</a>
34                    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">
35                        <span class="navbar-toggler-icon"></span>
36                    </button>
37                    <div class="collapse navbar-collapse" id="navbarNav">
38                        <ul class="navbar-nav ms-auto">
39                            <li class="nav-item">
40                                <a class="nav-link" href="{{ route('home') }}">Home</a>
41                            </li>
42                            <li class="nav-item">
43                                <a class="nav-link" href="{{ route('about') }}">About</a>
44                            </li>
45                            <li class="nav-item">
46                                <a class="nav-link" href="{{ route('contact') }}">Contact</a>
47                            </li>
48                        </ul>
49                    </div>
50                </div>
51            </nav>
52        </div>
53
54        <!-- Main Content -->
55        <div class="container">
56            <div class="row">
57                @yield('content')
58            </div>
59        </div>
60        @push('scripts')
61    <script>
62        $(document).ready(function() {
63            @if(session('order_status'))
64                $('#orderModal').modal('show');
65            @endif
66        });
67    </script>
68    @endpush
69    <!-- jQuery -->
70    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
71    <!-- Bootstrap JS -->
72    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
```

@extends & @section to use the layout inside the components folder the file called layout.blade

```php
contact.blade.php ⬤M ✕

resources > views > 🐘 contact.blade.php
   1    @extends(view: 'Components.Layout')
   2
   3    @section(section: 'title', content: 'Contact Us')
   4
   5    @section(section: 'content')
   6        <section id="contact">
   7            <div class="container">
   8                <h2 class="text-left">Contact Us</h2>
   9                <div class="row">
  10                    <div class="col-md-6">
  11                        <form action="{{ route(name: 'contact.submit') }}" method="POST">
  12                            @csrf
  13                            <div class="mb-4">
  14                                <label for="username">Name</label>
  15                                <input type="text" class="form-control" id="username" name="username" required>
  16                            </div>
  17                            <div class="mb-4">
  18                                <label for="email" class="form-label">Email</label>
  19                                <input type="email" class="form-control" id="email" name="email" required>
  20                            </div>
  21                            <div class="mb-4">
  22                                <label for="phone" class="form-label">Phone</label>
  23                                <input type="tel" class="form-control" id="phone" name="phone">
  24                            </div>
  25                            <div class="mb-4">
  26                                <label for="message" class="form-label">Message</label>
  27                                <textarea class="form-control" id="message" rows="4" name="message" required></textarea>
  28                            </div>
  29                            <button type="submit" class="btn btn-primary">Submit</button>
  30                        </form>
  31                    </div>
  32
  33                </div>
  34            </div>
  35        </section>
  36    @endsection
  37
```

Routes

```php
🐘 web.php  ✕

routes > 🐘 web.php
 ⬤  1    <?php
   2
   3    use Illuminate\Support\Facades\Route;
   4    use Illuminate\Http\Request;
   5    use App\Http\Controllers\ContactController;
   6    use App\Http\Controllers\OrderController;
   7    use App\Http\Controllers\HomeController;
   8
   9
  10    Route::get(uri: '/home', action: function (): Factory|View {
  11        return view(view: 'home');
  12    })->name(name: 'home');
  13
  14    Route::get(uri: '/about', action: function (): Factory|View {
  15        return view(view: 'about');
  16    })->name(name: 'about');
  17
  18    Route::get(uri: '/contact', action: function (): Factory|View {
  19        return view(view: 'contact');
  20    })->name(name: 'contact');
  21
  22
  23    Route::get(uri: '/', action: [HomeController::class, 'index'])->name(name: 'home');
  24    // Route for adding items to the order
  25    Route::post(uri: '/order/add', action: [OrderController::class, 'add'])->name(name: 'order.add');
  26
  27    Route::post(uri: '/contact/submit', action: [ContactController::class, 'submit'])->name(name: 'contact.submit');
  28
  29    Route::post(uri: '/order/place', action: [OrderController::class, 'placeOrder'])->name(name: 'order.place');
  30
  31    Route::post(uri: '/order/cancel', action: [OrderController::class, 'cancel'])->name(name: 'order.cancel');
  32
```

The routing setup and how it serves the views

The routes /home, /about, /contact, serve as the link to the home, about, contact views.
OrderController uses the methods /add, /place, /cancel to do their respective functions.
ContactController uses the method /submit to send user data that is inputted in the forms