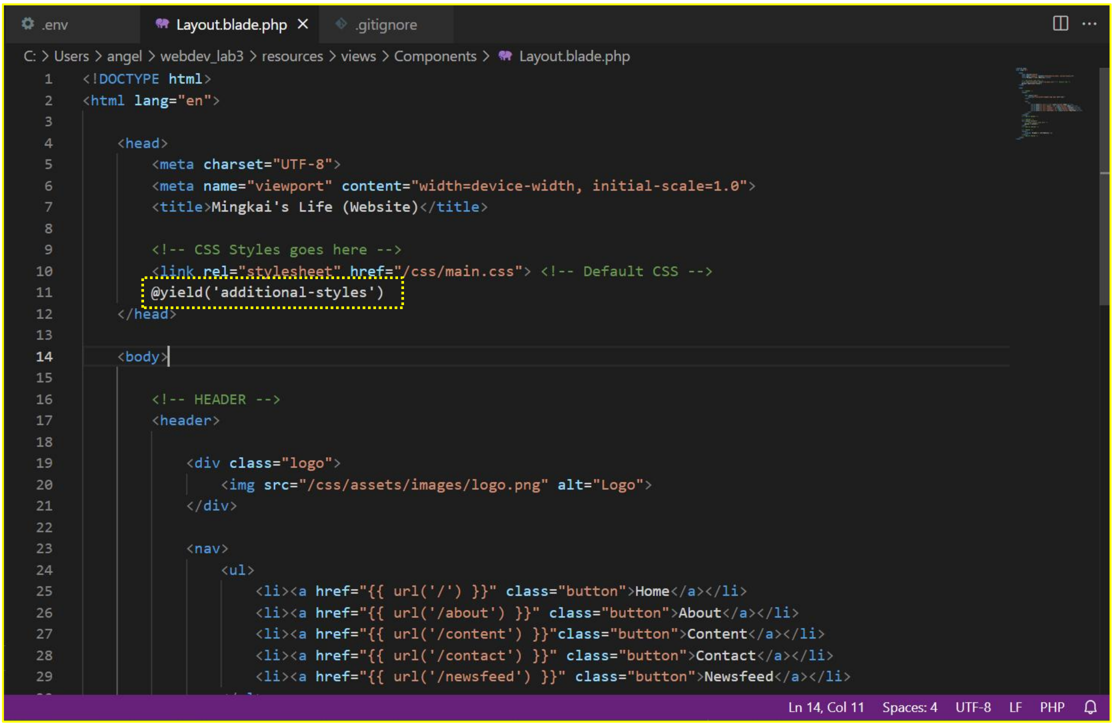
ABAD, ANGELA B.
BSIT 3C

WEB DEVELOPMENT

Part 1: Layout File (Layout.blade.php)

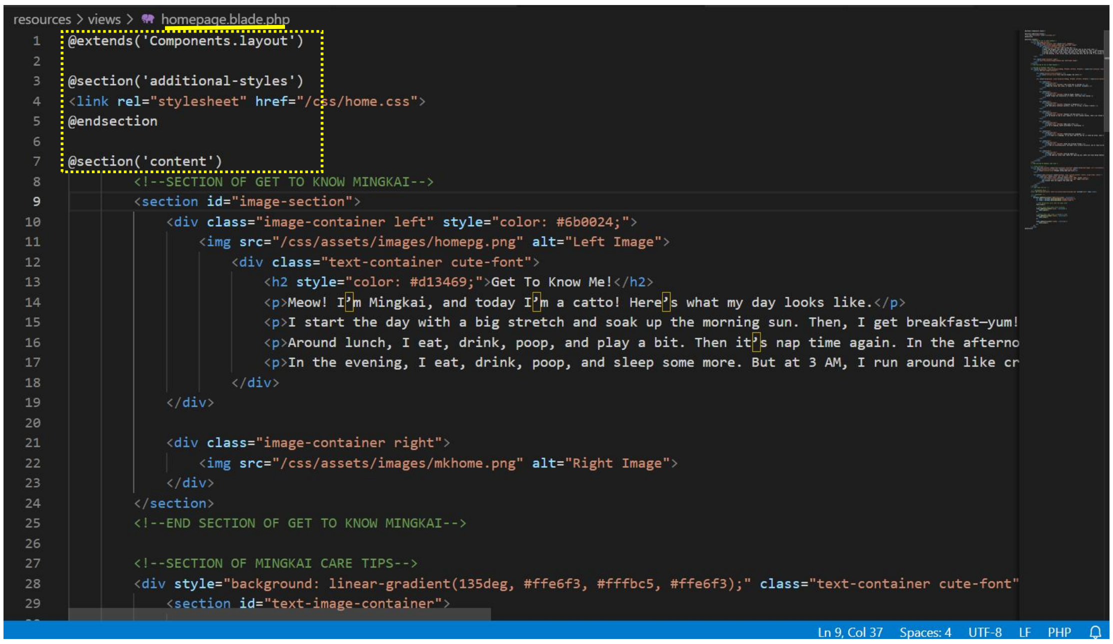
The layout file is a reusable template that defines the overall structure of the pages, such as the header, footer, and body layout. Views extend this layout to maintain consistency across pages. By using @yield, specific content sections can be filled in by child views.


.env
Layout.blade.php
.gitignore
C: > Users > angel > webdev_lab3 > resources > views > Components > Layout.blade.php
Ln 14, Col 11   Spaces: 4   UTF-8   LF   PHP

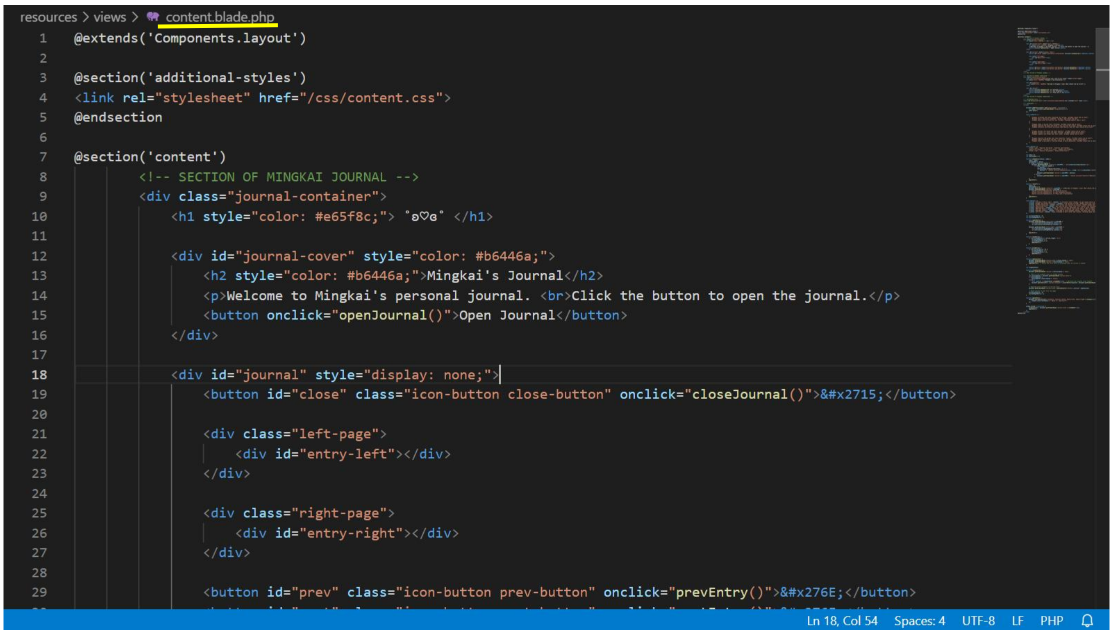
```
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Mingkai's Life (Website)</title>

        <!-- CSS Styles goes here -->
        <link rel="stylesheet" href="/css/main.css"> <!-- Default CSS -->
        @yield('additional-styles')
    </head>

    <body>

        <!-- HEADER -->
        <header>

            <div class="logo">
                <img src="/css/assets/images/logo.png" alt="Logo">
            </div>

            <nav>
                <ul>
                    <li><a href="{{ url('/') }}" class="button">Home</a></li>
                    <li><a href="{{ url('/about') }}" class="button">About</a></li>
                    <li><a href="{{ url('/content') }}"class="button">Content</a></li>
                    <li><a href="{{ url('/contact') }}" class="button">Contact</a></li>
                    <li><a href="{{ url('/newsfeed') }}" class="button">Newsfeed</a></li>
```

Part 2: Views

Each view file (home.blade.php, about.blade.php, etc.) extends the layout using @extends('Components.Layout'). Page-specific content is inserted using @section, allowing for unique content within the shared layout structure.


resources > views > homepage.blade.php
Ln 9, Col 37   Spaces: 4   UTF-8   LF   PHP

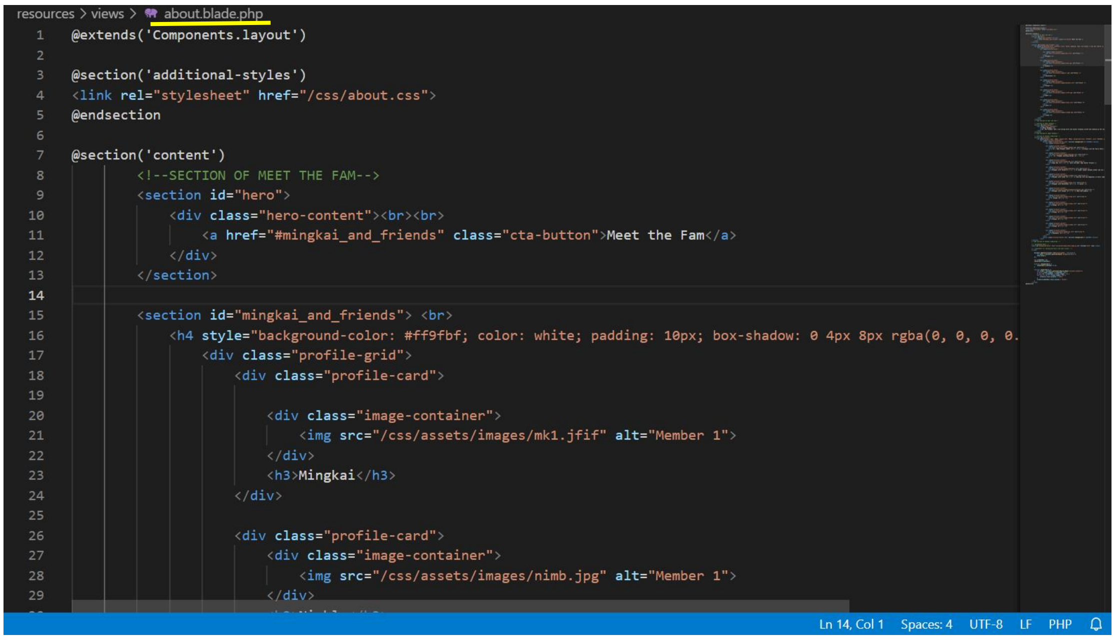
```
@extends('Components.layout')

@section('additional-styles')
<link rel="stylesheet" href="/css/home.css">
@endsection

@section('content')
        <!--SECTION OF GET TO KNOW MINGKAI-->
        <section id="image-section">
            <div class="image-container left" style="color: #6b0024;">
                <img src="/css/assets/images/homepg.png" alt="Left Image">
                    <div class="text-container cute-font">
                        <h2 style="color: #d13469;">Get To Know Me!</h2>
                        <p>Meow! I’m Mingkai, and today I’m a catto! Here’s what my day looks like.</p>
                        <p>I start the day with a big stretch and soak up the morning sun. Then, I get breakfast—yum!
                        <p>Around lunch, I eat, drink, poop, and play a bit. Then it’s nap time again. In the afterno
                        <p>In the evening, I eat, drink, poop, and sleep some more. But at 3 AM, I run around like cr
                    </div>
            </div>

            <div class="image-container right">
                <img src="/css/assets/images/mkhome.png" alt="Right Image">
            </div>
        </section>
        <!--END SECTION OF GET TO KNOW MINGKAI-->

        <!--SECTION OF MINGKAI CARE TIPS-->
        <div style="background: linear-gradient(135deg, #ffe6f3, #fffbc5, #ffe6f3);" class="text-container cute-font"
            <section id="text-image-container">
```



resources > views > content.blade.php

```
@extends('Components.layout')

@section('additional-styles')
<link rel="stylesheet" href="/css/content.css">
@endsection

@section('content')
        <!-- SECTION OF MINGKAI JOURNAL -->
        <div class="journal-container">
            <h1 style="color: #e65f8c;"> ˚ʚ♡ɞ˚ </h1>

            <div id="journal-cover" style="color: #b6446a;">
                <h2 style="color: #b6446a;">Mingkai's Journal</h2>
                <p>Welcome to Mingkai's personal journal. <br>Click the button to open the journal.</p>
                <button onclick="openJournal()">Open Journal</button>
            </div>

            <div id="journal" style="display: none;">
                <button id="close" class="icon-button close-button" onclick="closeJournal()">&#x2715;</button>

                <div class="left-page">
                    <div id="entry-left"></div>
                </div>

                <div class="right-page">
                    <div id="entry-right"></div>
                </div>

                <button id="prev" class="icon-button prev-button" onclick="prevEntry()">&#x276E;</button>
```

Ln 18, Col 54   Spaces: 4   UTF-8   LF   PHP



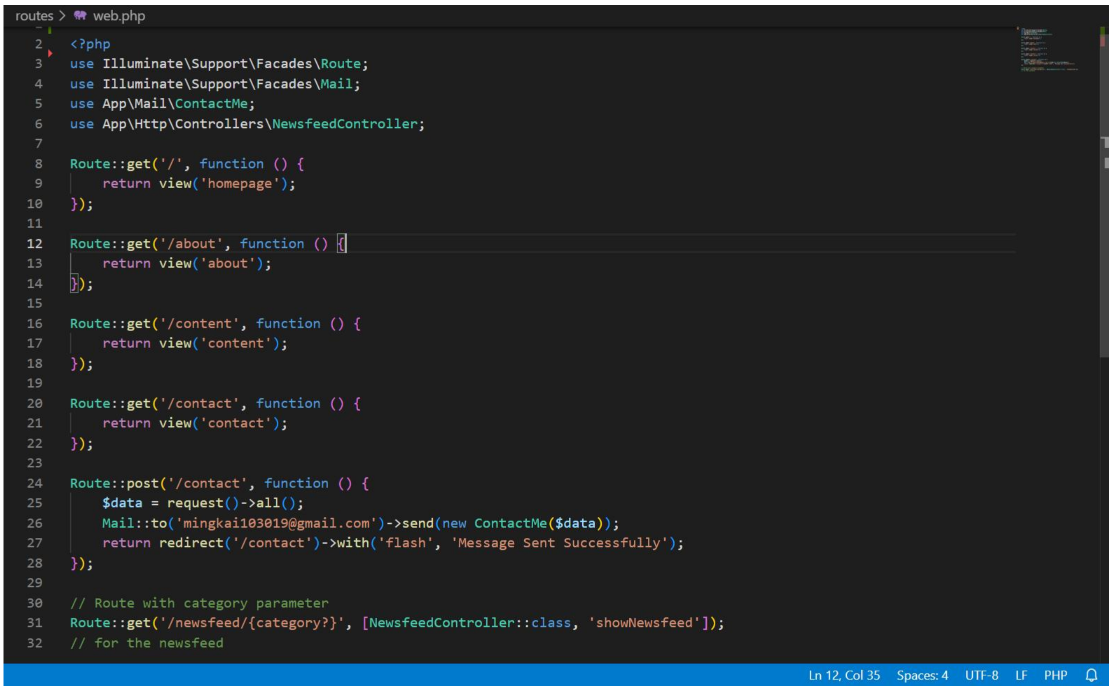
resources > views > about.blade.php

```
@extends('Components.layout')

@section('additional-styles')
<link rel="stylesheet" href="/css/about.css">
@endsection

@section('content')
        <!--SECTION OF MEET THE FAM-->
        <section id="hero">
            <div class="hero-content"><br><br>
                <a href="#mingkai_and_friends" class="cta-button">Meet the Fam</a>
            </div>
        </section>

        <section id="mingkai_and_friends"> <br>
            <h4 style="background-color: #ff9fbf; color: white; padding: 10px; box-shadow: 0 4px 8px rgba(0, 0, 0, 0.
                <div class="profile-grid">
                    <div class="profile-card">

                        <div class="image-container">
                            <img src="/css/assets/images/mk1.jfif" alt="Member 1">
                        </div>
                        <h3>Mingkai</h3>
                    </div>

                    <div class="profile-card">
                        <div class="image-container">
                            <img src="/css/assets/images/nimb.jpg" alt="Member 1">
                        </div>
```

Ln 14, Col 1   Spaces: 4   UTF-8   LF   PHP

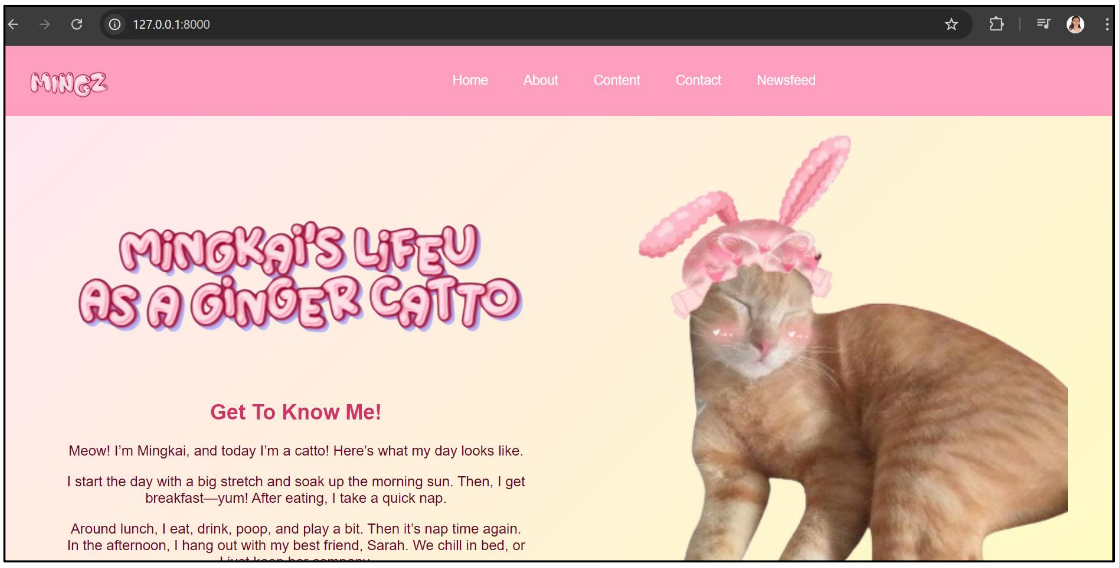
Part 3: Routes

In routes/web.php, each route points to a corresponding view. For example, Route::get('/home', ...) returns the home.blade.php view. This setup maps URLs to specific pages, making navigation possible.


routes > web.php

```php
<?php
use Illuminate\Support\Facades\Route;
use Illuminate\Support\Facades\Mail;
use App\Mail\ContactMe;
use App\Http\Controllers\NewsfeedController;

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

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

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

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

Route::post('/contact', function () {
    $data = request()->all();
    Mail::to('mingkai103019@gmail.com')->send(new ContactMe($data));
    return redirect('/contact')->with('flash', 'Message Sent Successfully');
});

// Route with category parameter
Route::get('/newsfeed/{category?}', [NewsfeedController::class, 'showNewsfeed']);
// for the newsfeed
```

Part 4: Rendered Web Pages

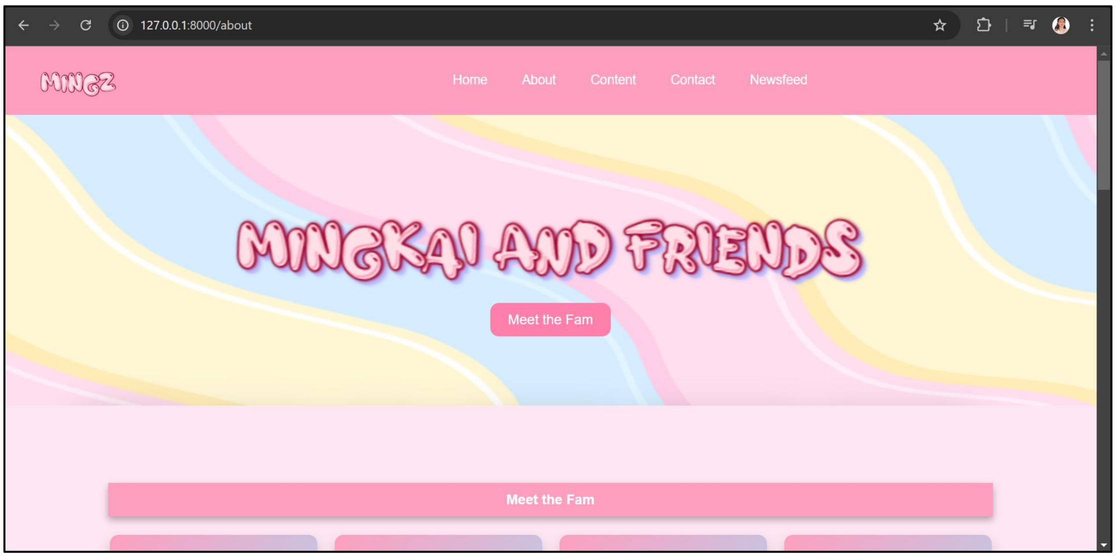


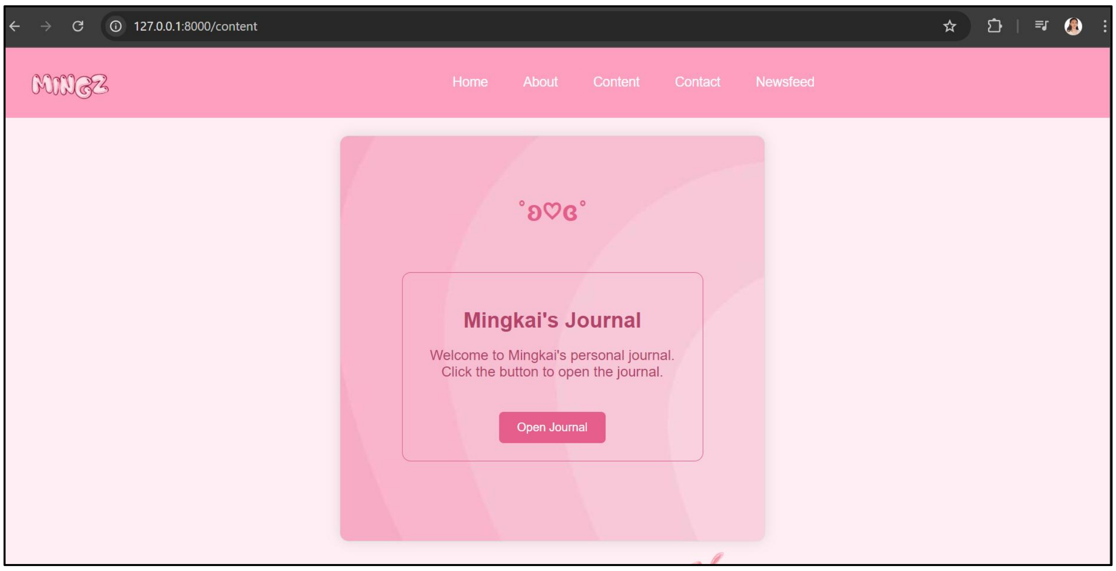
Get To Know Me!

Meow! I'm Mingkai, and today I'm a catto! Here's what my day looks like.

I start the day with a big stretch and soak up the morning sun. Then, I get breakfast—yum! After eating, I take a quick nap.

Around lunch, I eat, drink, poop, and play a bit. Then it's nap time again. In the afternoon, I hang out with my best friend, Sarah. We chill in bed, or





Mingkai's Journal

Welcome to Mingkai's personal journal. Click the button to open the journal.

//NOTE: There are other views aside from ***content.blade.php, homepage.blade.php*** and ***about.blade.php***. But I did not include them in the documentation since only 3 blade files are required.

✧ **Difference between @yield** and **{{$slot}}:**

@yield

- **How we used it**: In our Layout.blade.php file, we placed @yield to define areas where specific content from different pages would be inserted, like @yield('content'). This allowed us to keep the overall layout consistent while each view, like the homepage or about page, used @section('content') to fill that space with its unique content.

{{$slot}}

- **How it works**: We did not used this but, {{$slot}} operates similarly yet it is typically used within components. While @yield defines sections in the layout, {{$slot}} is used inside reusable components, acting as a placeholder where we can pass different content. For instance, in a card or button component, we would use {{$slot}} to insert specific details, much like how @yield lets us insert content into the layout.

Both @yield and {{$slot}} allow us to inject dynamic content, but while @yield is for layouts, {{$slot}} is for more modular components

PROBLEM ENCOUTERED:

Each time I pull changes from GitHub, I encounter the need to manually modify the `` `.env` `` file to suit the configuration of my local environment. Specifically, the paths and settings in the `` `.env` `` file—particularly those related to the database—are set according to my groupmates machine. For instance, the database path is often set to their local file structure, such as ***`C:/Users/winOSx/cd/webdev_lab3/database/database.sqlite`***, which does not match my environment. On my laptop, the correct path is ***`C:/Users/angel/webdev_lab3/database/database.sqlite`.***

This constant adjustment of the `` `.env` `` file after each pull becomes a tedious and time-consuming process. The file must be edited to reflect the absolute paths, database settings, and other environment-specific variables unique to my system. This process interrupts the flow of development, making it harder to maintain consistency across different environments. **As a result, the configuration is not standardized between team members, further complicating the issue.**It creates a challenge in collaborative projects where environments differ.