In Laravel, the `extend`, `section`, and `yield` directives are used for creating and extending layouts, allowing you to define reusable templates for your web pages.

Here's how each of these directives works with examples:

1. `@extends` Directive:

The `@extends` directive is used to specify a layout that a view should inherit or extend. This directive is placed at the top of a Blade view file (child view) to indicate which layout should be used as its parent.

Example:

Suppose you have a layout file named `layouts.app.blade.php`:

```html

<!DOCTYPE html>

<html>

<head>

<title>@yield('title')</title>

</head>

<body>

@yield('content')

</body>

</html>

```

In a child view file, you can extend this layout using `@extends`:

```html

@extends('layouts.app')

@section('title', 'My Page Title')

@section('content')

<div>

<h1>Welcome to my page!</h1>

<p>This is the content of the page.</p>

</div>

@endsection

```

In this example, the `@extends` directive specifies that the child view extends the `layouts.app` layout. The `@section` directives define the content that should replace the `@yield` placeholders in the layout.

2. `@section` Directive:

The `@section` directive is used to define content that will be placed within a named section of the layout. Sections are placeholders where you can inject content from child views.

Example:

In the child view example above, we used `@section('title')` and `@section('content')` to define the title and content sections in the layout.

3. `@yield` Directive:

The `@yield` directive is used in the layout file to define where the content from child views should be placed. It acts as a placeholder that displays the content of a named section defined in child views.

Example:

In the `layouts.app.blade.php` layout file:

html

<!DOCTYPE html>

<html>

<head>

<title>@yield('title')</title>

</head>

<body>

@yield('content')

</body>

</html>

In the child view, we used `@section('title')` and `@section('content')` to specify the content for these sections. The `@yield('title')` and `@yield('content')` placeholders in the layout will be replaced with the content defined in the child view.

When you load the child view, Laravel combines the layout and child view, replacing the `@yield` placeholders with the corresponding content from the child view. This allows you to create consistent layouts for your web pages while customizing the content of each page as needed.

In summary, `@extends`, `@section`, and `@yield` are essential Blade directives in Laravel for creating flexible and reusable layouts for your web application's views. They help you structure your templates and maintain a consistent design across your application.