

# LAYOUTS

[Layouts](#)

[Txantilo bat luzatzea](#)

[Partials](#)

## Layouts

Layouts, gure bisten eraikuntza elkartzeko interegarriak izan daitezke.

Alde batetik **@include** erabili dezakegu, banatuta dagoen kodea erabili ahal izateko:

```
@include('footer')
//Hala, view-en zuzenean gordetako elementu bat sartuko dugu, php edo luzapena duena blade.php
```

Orrri baten barruan nahi beste @include sartu daitezke.

### Helper asset()

Asset helperrak gure fitxategiaren url absolutua ematen digu, modu honetan saiestuko dugu fitxategiaren ruta aldatzea segun eta zein URL-a dugun.

```
<Link href="{{ asset('css/style.css') }}" rel="stylesheet">
```

Baina benetan erabilgarriena **Layouts**-ak erabiltzea izango litzateke. Gure txantiloia hainbat fitxategietan banandu beharrean, txantiloia bat sortu dezakegu gure lekuaren estruktura guztia duena. Adibidez **layout.blade.php** izena jarri diezaiokegu eta barruan gure diseinuaren kode guztia sartu. Normalean hauek layout azpikarpetan gordetzen dira.

Txantiloia barruan **@yield** direktiba erabiliz atalak (*secciones*) adierazi ditzakegu, argumentu bezala atalaren izena pasatuz, eta gero txantiloia bakoitza atalari dagokion kodea sartuko genuke:

```
<main role="main" class="container">
    @yield('content')
</main>
```

Layouts-ak edozein izen izan ditzakete, baina beti *.blade.php* luzapena izan behar dute.

**@yield** direktiban hainbat modutan erabili daitezke, adibidez gure webgunearen titulua personalizatzeko.

```
<title>@yield('title') - Nireweb.net</title>
```

**@section** erabiliko dugu atalak zehazteko eta **@yield** direktiba, atal horiek gure txantiloian (gure layout-ean) kokatzeko.

Beraz, atalek **@section** direktiba erabiliko dute, argumentu moduan dagokion izena pasatuko diogu:

```
@section('title') Usuario {{ $id }} @endsection
```

```
@section('content')
    <!-- Contenido de la sección -->
@endsection
```

Aurreko adibidean, title-ren kasuan, lerro bakarra denez, izenburua parametro moduan pasatu dezakegu:

```
@section('title', "Usuario {{ $id }}")
```

## Txantiloia bat luzatzea

Banakako txantiloia bakoitzean, header edo footerra sartu beharrean, Laraveli esango diogu ikuspegia **layout.blade.php**-z luzatu behar duela.

Ez da beharrezkoa fitxategiaren luzapena jartzea, ezta ibilbide osoa jartzea ere ez, Laravelek, lehenespenez, **resources/view** bilatuko baitu fitxategia.

```
@extends('layout')
```

▪ view karpeta barruan beste karpeta berri baten egotekotan, adibidez **resources/view/txantiloiak**, horrela idatzi beharko genuke:

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

Behin hau eginda, atalak sortuko ditugu:

```
@section('content')
    <!-- Contenido de la sección -->
@endsection
```

## Partials

Layouts erabiltzeaz gain, **partials** izeneko artxiboak ere badaude; artxibo horiek kode-segmentu txikiak dira, eta normalean

views  
alumnos

blade nomen kode segimena erabiltzeko, eta normalizazio sistemaren zati jakin batzuetan erabiltzen dira, menuak, footerrak etab.

Ondoko adibidea jarraituz, alumnos.blade.php txatiloain deiak hurrengo kodean agertzen den moduan egin beharko genituzke:

```
<body>
    @include('partials.menu')
    @yield('content')
    @include('partials.footer')
</body>
```

