

Routing



Maximilian Schwarzmüller

AUTHOR

@maxedapps www.mschwarzmueller.com



Overview



Route creation in Laravel

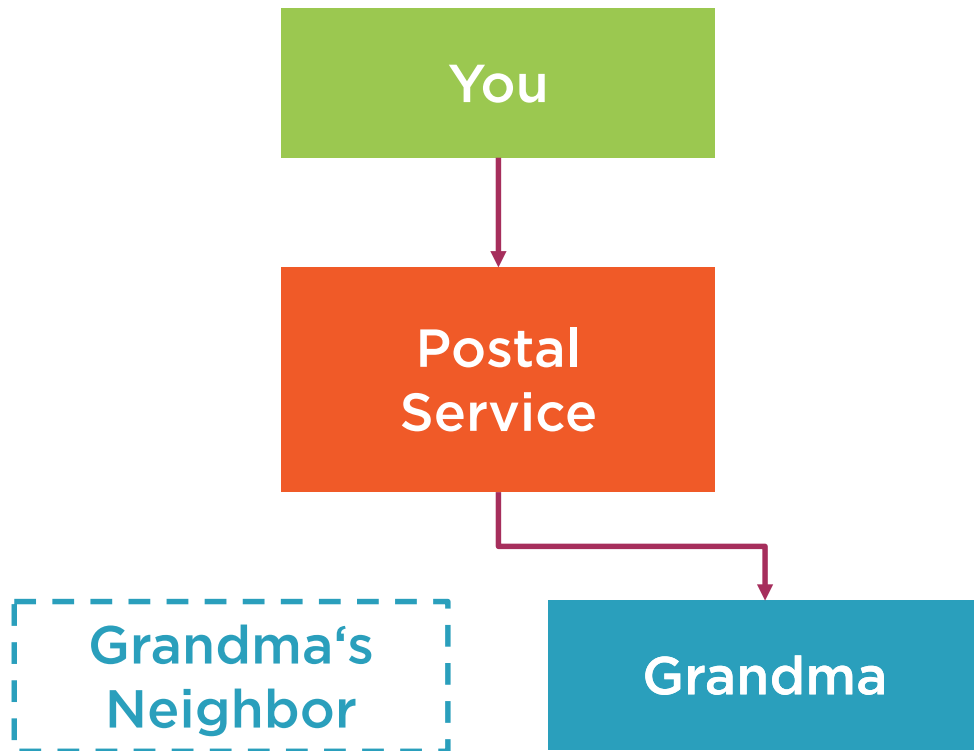
Mapping of controller actions

Managing route groups and middleware

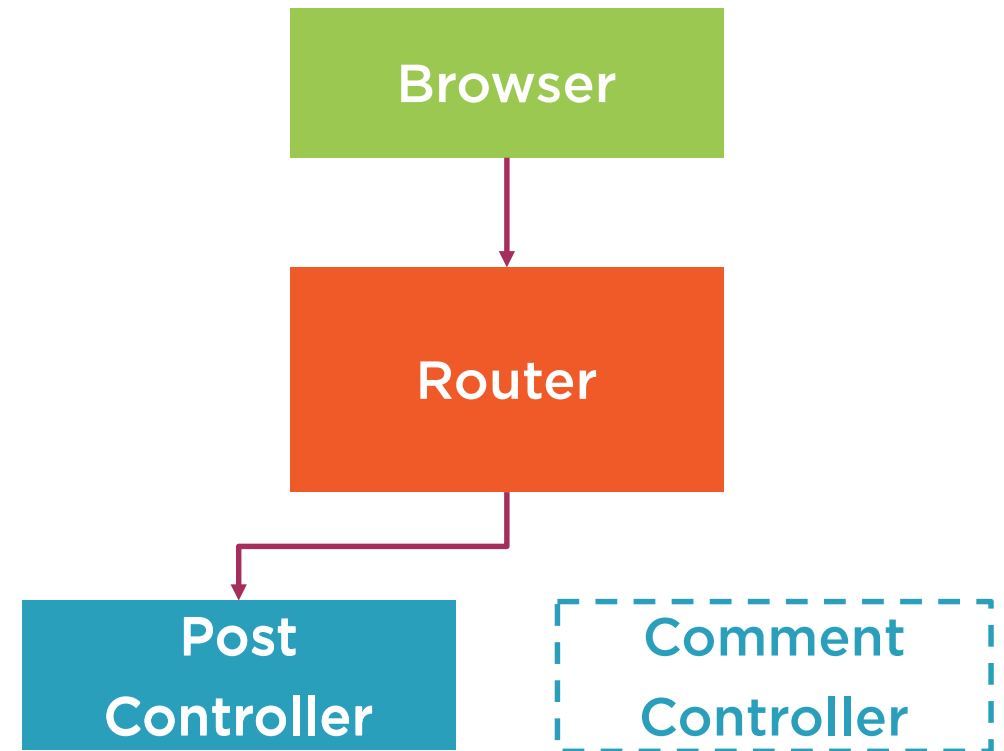


Routing Overview

Sending a Letter



Sending an HTTP Request



Creating Routes in Laravel

```
// app/Http/routes.php

Route::post('/post', [
    'uses' => 'PostController@create',
    'as' => 'post.create',
    'middleware' => 'auth',
]);
```



RESTful Routes in Laravel

```
// app/Http/routes.php
```

```
Route::resource('post', 'PostController', [  
    'only' => ['index', 'show']  
]);
```

```
Route::resource('post', 'PostController', [  
    'except' => ['create', 'store', 'update', 'destroy']  
]);
```



RESTful Controllers in Laravel

HTTP Method & URL

GET /post

GET /post/create

POST /post

GET /post/{post}

GET /post/{post}/edit

PUT/PATCH /post/{post}

DELETE /post/{post}

Controller Action

index [show all posts]

create [get creation form]

store [store post on server]

show [get single post]

edit [get edit form]

update [save update on server]

destroy [delete post on server]



Creating a RESTful Controller

```
php artisan make:controller PostController --resource
```

```
// app/Http/Controllers/PostController.php
```

```
// ...
```

```
public function store(Request $request) {
```

```
    // Controller Logic
```

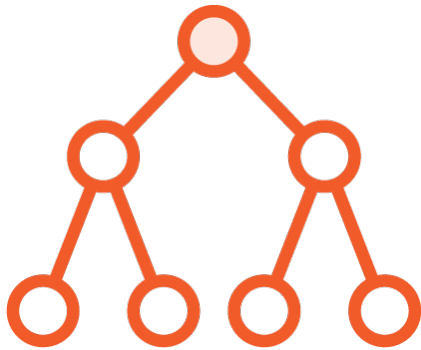
```
}
```



Route Grouping & Middleware



Route Similarities



Prefix

Routes share the same prefix (e.g. /api/v1)



Protection

Routes share the same protection level

Route Grouping

```
// app/Http/routes.php

Route::group(['prefix' => 'api/v1'], function() {
    Route::post('/post', [
        // Route Configuration
    ]);
});
```



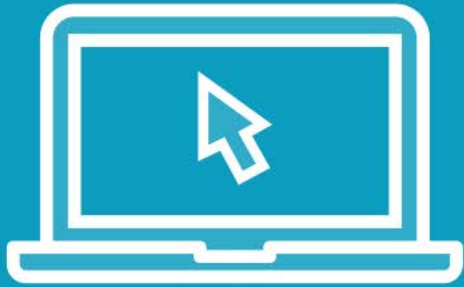
Controller Middleware

```
// app/Http/Controllers/PostController.php

public function __construct()
{
    $this->middleware('auth');
    $this->middleware('log', [
        'only' => ['store', 'update']
    ]);
}
```



Demo



Create RESTful routes & controller

Group routes and set up middleware



POST

Create Meeting

↑ Title, Description, Time, User ID
↓ Message, Summary, URL, Users

/api/v1/meeting

PUT

Update Meeting

↑ Meeting Data, User ID, Meeting ID
↓ Message, Summary, URL, Users

/api/v1/meeting

DELETE

Delete Meeting

↑ Meeting ID, User ID
↓ Message,

/api/v1/meeting

POST

Register

↑ User ID, Meeting ID
↓ Message, User, Meeting, URL

/api/v1/meeting/registration

DELETE

Unregister

↑ User ID, Meeting ID
↓ Message, User, Meeting, URL

/api/v1/meeting/registration

POST

Create User

↑ Name, E-Mail, Password
↓ Message, User, Meeting, URL

/api/v1/user

GET

Get List of all Meetings

↑ (null)
↓ List of Meetings, URL

/api/v1/meeting

GET

Get Meeting

↑ Meeting ID
↓ Meeting Info, URL

/api/v1/meeting

POST

User Signin

↑ E-Mail, Password
↓ ???

/api/v1/user/signin

Authentication required

No authentication required

