



How I Follow a Request Through Laravel

Find where it entered.
Find where it stopped.
Follow the response back out.

IN

Setup and middleware



THROUGH

Route and handler



OUT

Response and outer layers

A request goes in. A response comes out.

Some requests receive a response before reaching the route handler.

- | | | |
|---|---|--|
| <p>01</p> <p>Entry point</p> <p>public/index.php</p> | <p>02</p> <p>Bootstrap</p> <p>Prepare the application</p> | <p>03</p> <p>Global middleware</p> <p>Before route matching</p> |
| <p>04</p> <p>Router</p> <p>Match method and URL</p> | <p>05</p> <p>Route middleware</p> <p>Check the matched route</p> | <p>06</p> <p>Handler</p> <p>Controller or closure</p> |

Read stages 1–6 in order. The response returns through middleware that was entered.

Get the application ready before doing the work.

For requests handled by Laravel, public/index.php is the entry point.

- | | | |
|----|-----------------------------|---|
| 01 | Load the application | Load Composer's autoloader and get the app from bootstrap/app.php. |
| 02 | Capture the request | The site hands Request::capture() to \$app->handleRequest(). |
| 03 | Prepare services | The HTTP kernel bootstraps configuration and service providers before application work. |

A static file served directly by Nginx may never enter Laravel.

Check scope and matching before the controller.

A request must reach the router before there is a matched route to inspect.

01 Global middleware

Inspect, change, continue, or return an immediate response across the HTTP pipeline.

02 Route matching

The router uses the URL and HTTP method. GET and POST can reach different handlers.

03 Placement matters

Do not assume global middleware already has a matched route or route-bound models.

The site's global security-header middleware also works on the response coming back out.

Reaching a route is not reaching its method body.

Checks can answer the request before your controller's first line runs.

01 Route and group middleware

A web route's configured group supplies concerns such as sessions and CSRF protection.

02 Early responses

Authentication can redirect a browser to login.
A request expecting JSON can get a different response.

03 Handler dependencies

Form Request resolution can validate and authorize before the method body executes.

A handler can return HTML, JSON, or a redirect. A Blade template is not required.

Middleware wraps the work inside it.

The layer entered first receives the response back last on the usual successful path.

```
Global: before
  Route: before
    Handler responds
  Route: after
Global: after
```

Response work

An entered middleware can add a header after the inner layer responds.

An early response returns through the outer layers that were entered.

An early stop skips middleware further inside and the route handler.

Use the usual path without treating it as universal.

Bring the extra detail in when it explains the problem you are investigating.

01 Exceptions

Handled exceptions become responses.
They can interrupt the ordinary return path.

02 Termination

Sending the response and termination work
are distinct from outgoing middleware changes.

03 Long-lived workers

Octane reuses a booted application.
Do not assume a fresh bootstrap for every request.

When the client sees a response also depends on the server runtime.

Start with how far the request got.

Pick one route and trace both the incoming request and the outgoing response.

- ☐ **Entry** Was this served as a file or forwarded to Laravel?
- ☐ **Setup** Were the required dependencies registered?
- ☐ **Route** Does the method and URL match the intended handler?
- ☐ **Before the handler** Did middleware, binding, or a Form Request stop it?
- ☐ **Response** Which entered middleware changes the result on its way out?

[Read the full article: How I Follow a Request Through Laravel](#)