Handling Requests & Responses
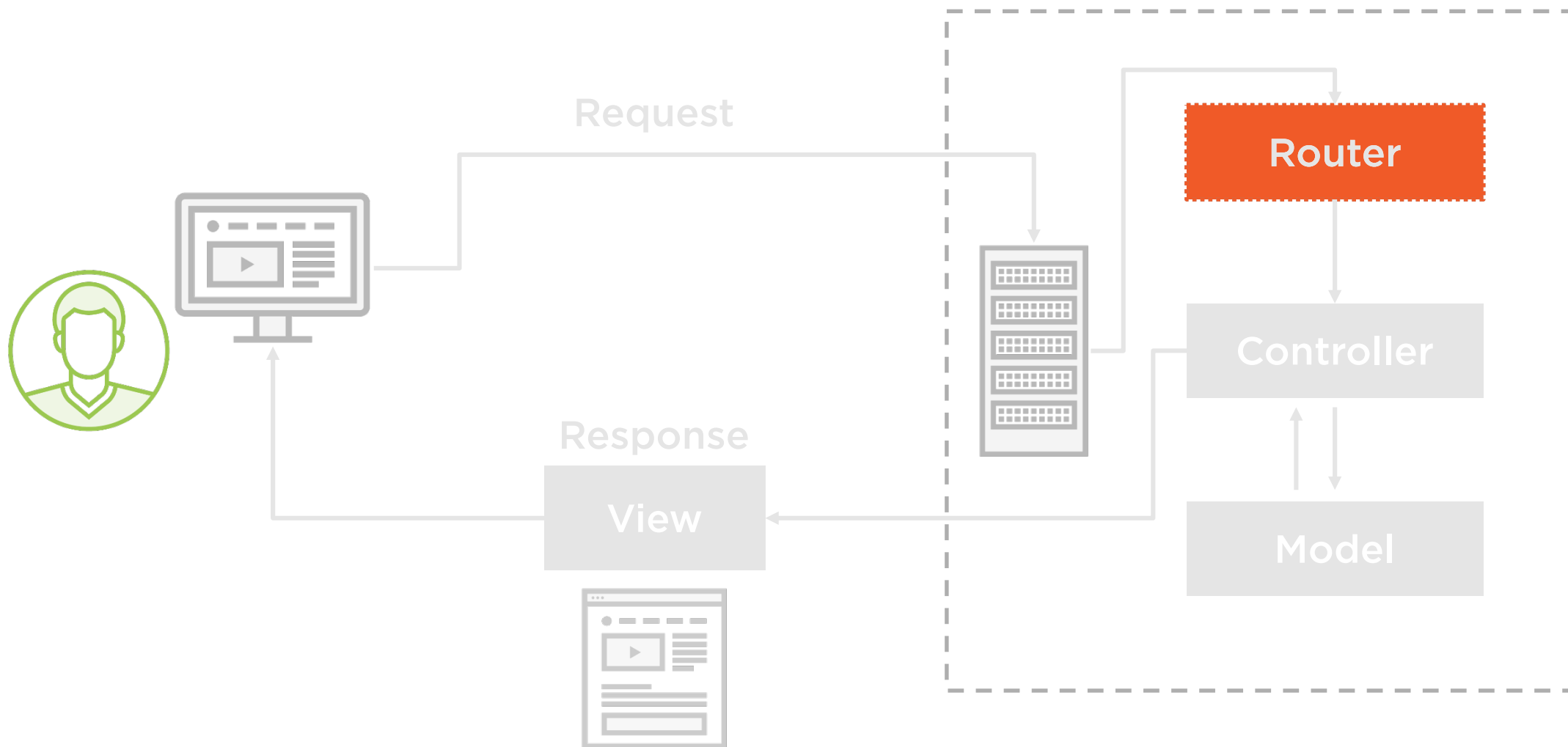
Maximilian Schwarzmüller
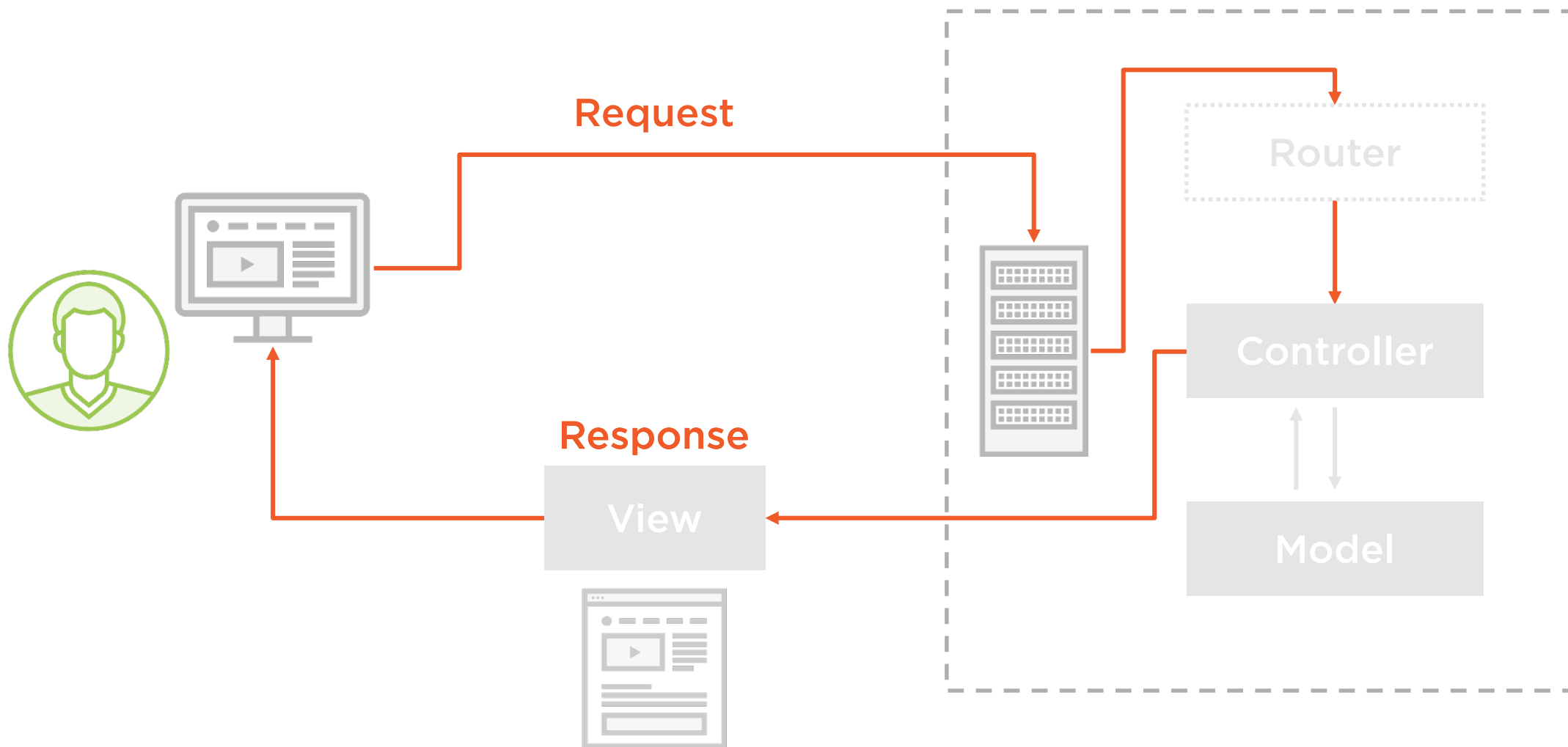
AUTHOR

@maxedapps   www.mschwarzmueller.com

Where Are We?

Request

Router

Response

View

Controller

Model

Where Are We?

```
Route::get('post/{id}', function($id) {

    return view('blog.post');

});
```

Handling a Get Request

Execute Route Action

Receive Parameters

Perform Logic and Send Response

Sending Responses

```php
return view('blog.post'); // Response::view()

return 'Just some text';

return Response::json(['name' => 'Max']);

return redirect()->route('index'); // Response::redirect()

...
```

```php
Route::post('create', function(Request $request) {

    return redirect()->route('admin.index');

});
```
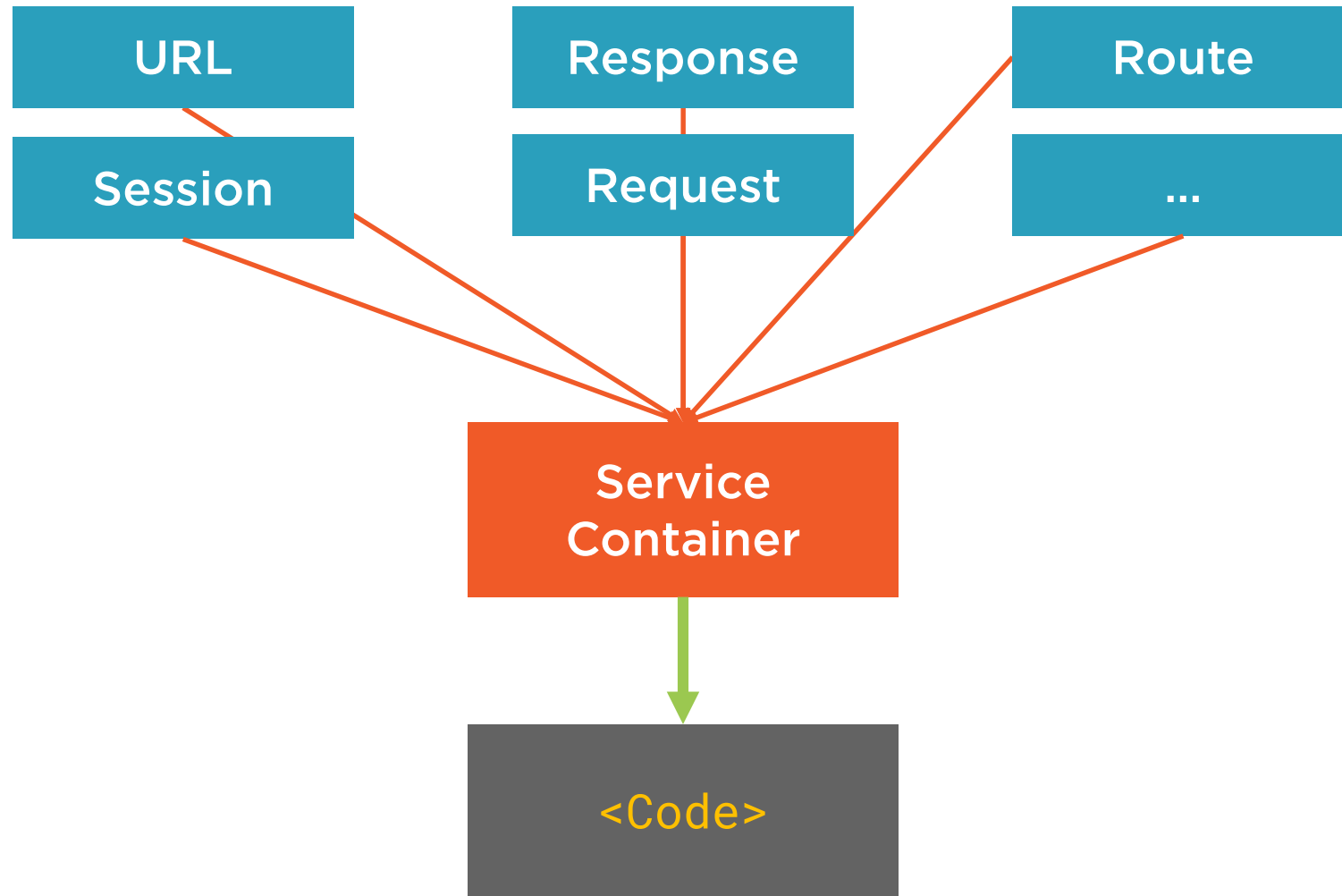
Handling a Post Request

Execute Route Action
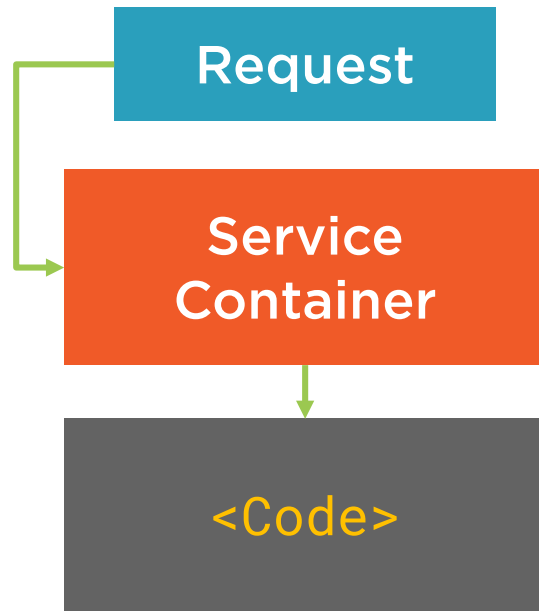
Receive injected Objects
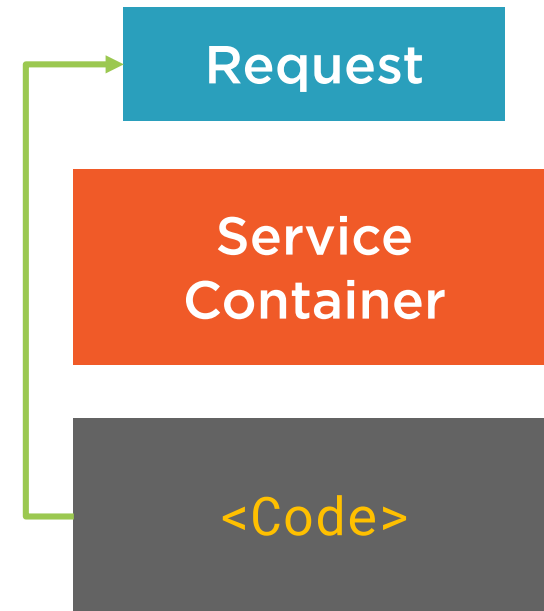
Perform Logic and Send Response

Dependency Injection

Dependency Injection vs Facades

Dependency Injection

Request

Service Container

\<Code\>

```php
function(Request $request)
{
    $request->input('mail');
}
```

Facades

Request

Service Container

\<Code\>

```php
function()
{
    Request::input('mail');
}
```

CSRF Attacks

Malicious Request

Response

Manipulated URL

Example: Transfer Funds

CSRF Protection

Malicious Request

Request without valid Token gets Rejected

Response + CSRF Token

Manipulated URL
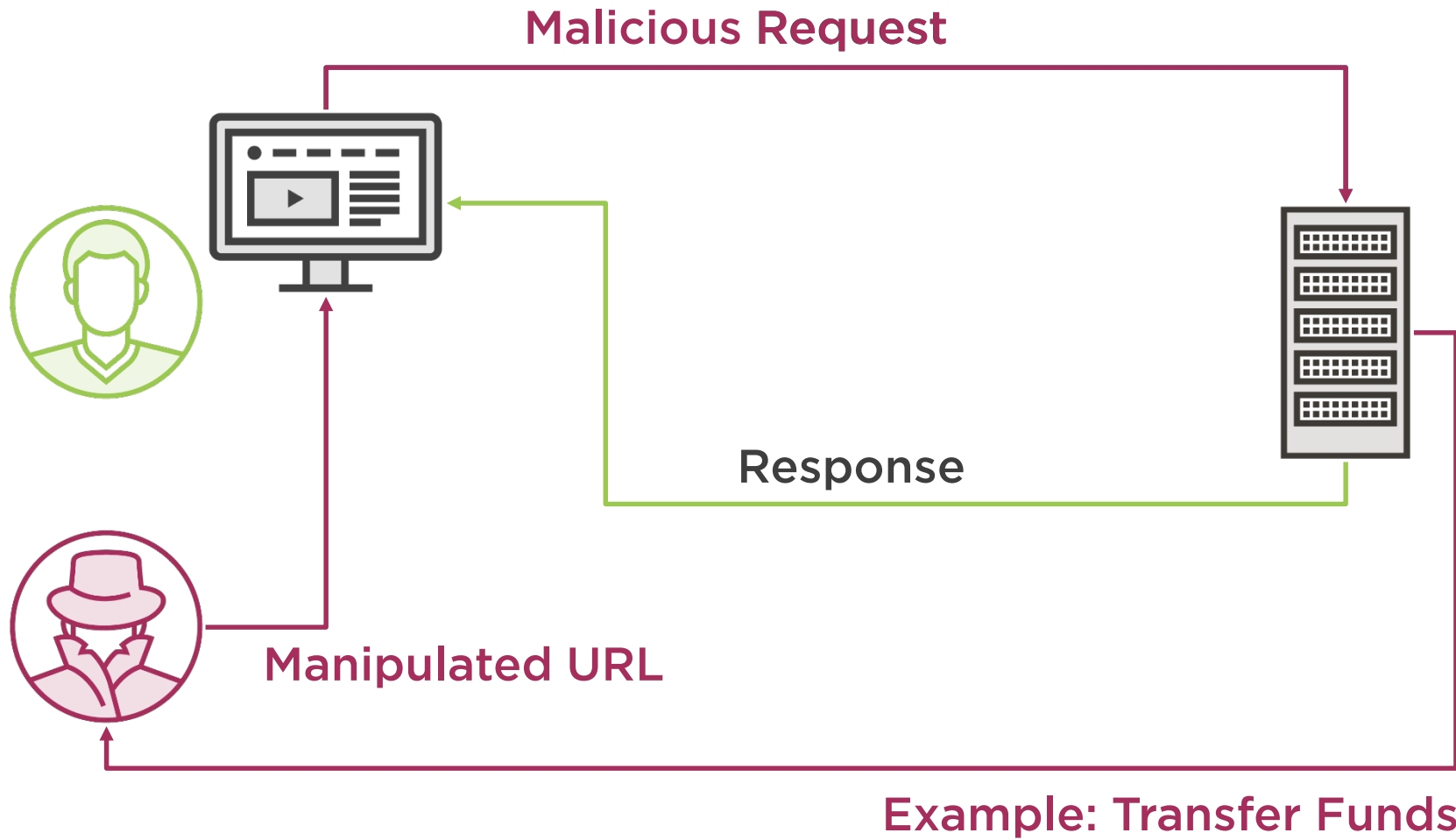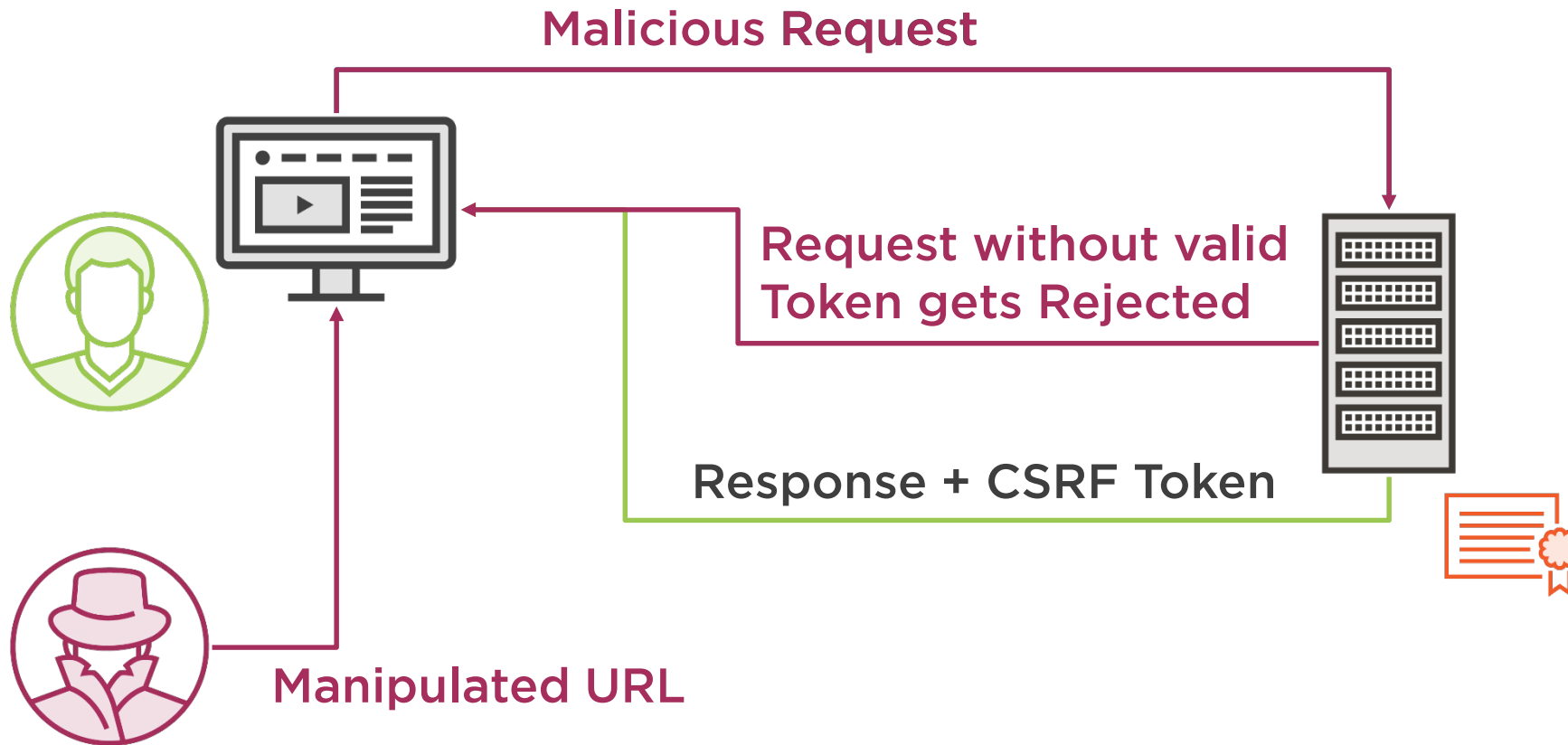
Laravel's CSRF Protection

```html
<form>

    <input type="text" name="mail">

    <input type="hidden" name="_token" value="{{csrf_token()}}">

    <button type="submit">

</form>
```

Summary

Handle GET and POST Requests

Send Responses

Use Dependency Injection

Pass Data to Views

Protect against CSRF Attacks

Use Sessions to flash Data

Validate User Input