






LARAVEL FRAMEWORK

BÀI 5: VALIDATION




- ⊙ Tổng quan Validation
- ⊙ Cấu hình sử dụng Validation
- ⊙ Form Request Validation
- ⊙ Custom Validation Rules



Phần I: Tổng quan Validation

-  Giới thiệu Validation
-  Các thành phần trong Validation
-  Cấu hình sử dụng Validation

Phần II: HTTP Basic Authentication

-  Tạo Routes
-  Tạo Controller
-  Thêm các xác thực tự định nghĩa



- ❑ Validation là một công đoạn quan trọng khi phát triển ứng dụng, thường là kiểm tra dữ liệu đầu vào.
- ❑ Validation là tiến trình xác thực, ràng buộc dữ liệu đến ứng dụng của bạn có tuân theo quy tắc nhất định



- ❑ Laravel cung cấp nhiều cách để Validate dữ liệu ứng dụng.
- ❑ Mặc định, class base controller của Laravel sử dụng ValidatesRequests trait cung cấp phương thức khá thuận tiện cho việc validate HTTP request đến với đa dạng quy định validation.

Laravel Form Validate

❌ Input field !

This field cannot be empty.

❌ Valid input ✓

- ❑ Với ValidatesRequests trong Laravel thì các dữ liệu không hợp lệ luôn được kiểm tra và thông báo tới user thông qua biến \$error
- ❑ Danh sách “validation rules” trong Laravel:

Available Validation Rules in Laravel		
Accepted	Active URL	After (Date)
Alpha	Alpha Dash	Alpha Numeric
Array	Before (Date)	Between
Boolean	Confirmed	Date
Date Format	Different	Digits
Digits Between	E-Mail	Exists (Database)
Image (File)	In	Integer

❑ Danh sách “validation rules” trong Laravel:

IP Address	JSON	Max
MIME Types(File)	Min	Not In
Numeric	Regular Expression	Required
Required If	Required Unless	Required With
Required With All	Required Without	Required Without All
Same	Size	String
Timezone	Unique (Database)	URL

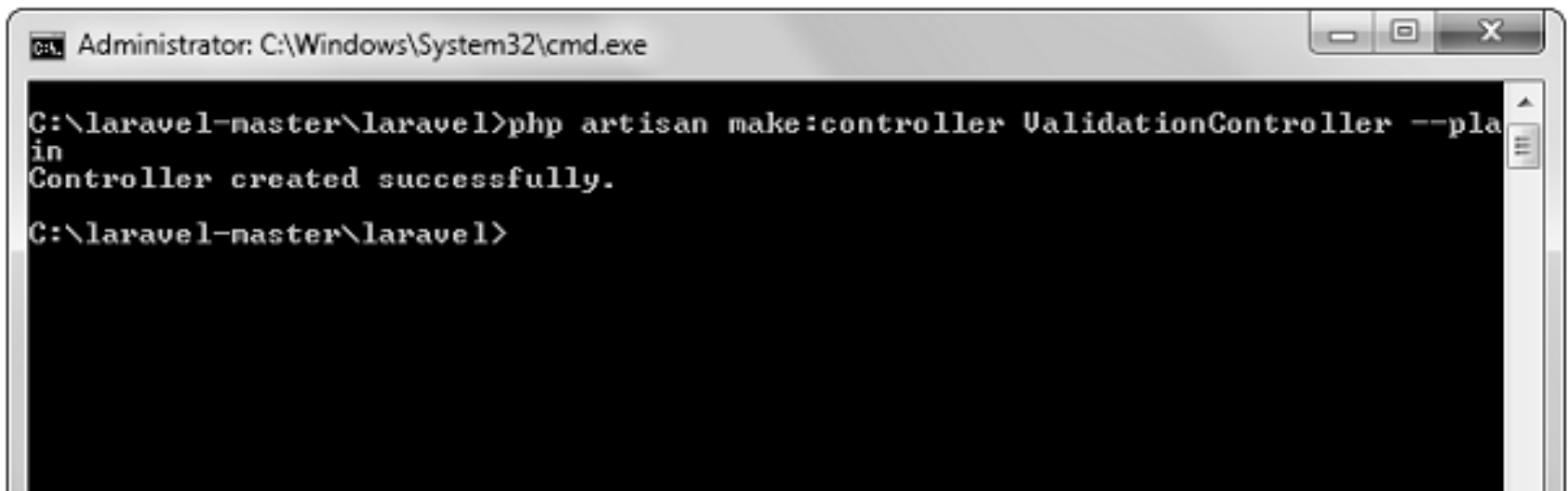
- ❑ Sử dụng biến `$errors` được cung cấp trong `Illuminate\Support\MessageBag` để hiển thị ra view

```
@if (count($errors) > 0)
    <div class = "alert alert-danger">
        <ul>
            @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
            @endforeach
        </ul>
    </div>
@endif
```


- ❑ Tạo controller ValidationController bằng php artisan:

```
php artisan make:controller ValidationController --plain
```

- ❑ Kết quả tạo thành công:



```
Administrator: C:\Windows\System32\cmd.exe

C:\laravel-master\laravel>php artisan make:controller ValidationController --plain
Controller created successfully.

C:\laravel-master\laravel>
```

- ❑ Tiến hành xác thực kiểm tra dữ liệu bên trong `app/Http/Controllers/ValidationController.php`

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Http\Requests;
use App\Http\Controllers\Controller;

class ValidationController extends Controller {
    public function showform(){
        return view('login');
    }
    public function validateform(Request $request){
        print_r($request->all());
        $this->validate($request,[
            'username'=>'required|max:8',
            'password'=>'required'
        ]);
    }
}
```

- ❑ Tạo View để tiếp nhận thông tin từ user trong `resources/views/login.blade.php`

```
<html>

<head>
    <title>Login Form</title>
</head>

<body>

    @if (count($errors) > 0)
        <div class = "alert alert-danger">
            <ul>
                @foreach ($errors->all() as $error)
                    <li>{{ $error }}</li>
                @endforeach
            </ul>
        </div>
    @endif

    <?php
        echo Form::open(array('url'=>'/validation'));
    ?>
```

- ❑ Tạo View để tiếp nhận thông tin từ user trong `resources/views/login.blade.php`

```
<table border = '1'>
  <tr>
    <td align = 'center' colspan = '2'>Login</td>
  </tr>
  <tr>
    <td>Username</td>
    <td><?php echo Form::text('username'); ?></td>
  </tr>
  <tr>
    <td>Password</td>
    <td><?php echo Form::password('password'); ?></td>
  </tr>
  <tr>
    <td align = 'center' colspan = '2'
      ><?php echo Form::submit('Login'); ? ></td>
    </tr>
</table>

<?php
  echo Form::close();
?>

</body>
</html>
```

- ❑ Định nghĩa Routes trong app/Http/routes.php

```
Route::get('/validation','ValidationController@showform');  
Route::post('/validation','ValidationController@validateform');
```

- ❑ Chạy <http://localhost:8000/validation> và kiểm tra kết quả

- The username field is required.
- The password field is required.

Login	
Username	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="Login"/>	



DEMO

Demo xác thực dữ liệu form login





LARAVEL FRAMEWORK

BÀI 5: VALIDATION (PHẦN 2)



- ❑ Trong Laravel đã tồn tại sẵn nhiều "validation rule" hữu dụng, tuy nhiên ta vẫn có nhu cầu định nghĩa ra thêm các "rule" riêng cho ứng dụng.

- ❑ Ví dụ tạo form xác thực có phải số điện thoại ở Việt Nam hay không

Laravel 5 : Custom Validation Rule

Enter Indian phone number

Save

- ❑ Laravel chưa hỗ trợ xác thực riêng cho số điện thoại VietNam. Ta tạo rule bắt buộc phải tồn tại "+84"

- ❑ Tạo 2 Routes, route thứ nhất sinh view và route thứ 2 xử lý nghiệp vụ khi người dùng post dữ liệu
- ❑ Tại file routes.php thêm đoạn code:

app/Http/routes.php

```
01. Route::get('custom-validation',['as'=>'custom-validation.get','uses'=>'CustomValidatorController@getCustomValidation']);

02. Route::post('custom-validation',['as'=>'custom-validation.post','uses'=>'CustomValidatorController@postCustomValidation']);
```

❑ Tạo controller bên trong CustomValidatorController.php

app/Http/Controllers/CustomValidatorController.php

```
01. <?php
02. namespace App\Http\Controllers;
03.
04. use Illuminate\Http\Request;
05. use App\Http\Controllers\Controller;
06.
07. class CustomValidatorController extends Controller {
08.
09.     public function getCustomValidation(){
10.
11.         return view('customvalidation');
12.     }
13.     public function postCustomValidation(Request $request){
14.         $this->validate($request, [
15.             'phone' => 'required|in_phone',
16.         ]);
17.         return 'successfully';
18.     }
19. }
```

- ❑ Hiện thực code xác thực mẫu số điện thoại VietNam bắt đầu bằng "+84"
- ❑ Mở rộng phương thức Validator bên trong AppServiceProvider với 4 đối số:
 - ❖ \$attribute
 - ❖ \$value
 - ❖ \$parameters
 - ❖ \$validator

```
Validator::extend('in_phone', function($attribute, $value, $parameters)
```

- ❑ "in_phone" là tên quy luật để sử dụng về sau khi cần

```
01. <?php
02.
03. namespace App\Providers;
04.
05. use Illuminate\Support\ServiceProvider;
06. use Validator;
07. class AppServiceProvider extends ServiceProvider
08. {
09.     public function boot()
10.     {
11.         Validator::extend('in_phone', function($attribute, $value, $parameters) {
12.             return substr($value, 0, 3) == '+84';
13.         });
14.     }
15.     public function register()
16.     {
17.     }
18. }
```

- ❑ Tạo các thông báo lỗi khi xác thực thất bại
- ❑ Các thông báo này do người dung định nghĩa

```
01. 'custom' => [  
02.   'phone' => [  
03.     'in_phone' => 'Please enter Viet Nam phone number which  
    starts with +84',  
04.   ],  
05. ],
```

❑ Tạo view trong resoueces/view/

resoueces/view/customvalidation.blade.php

```
01. <html lang="en">
02. <head>
03.     <title>Laravel 5 : Custom Validation Rule</title>
04.     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css" >
05. </head>
06. <body>
07.     <div class="container">
08.         {!! Form::open(array('route' => 'custom-validation.post','method'=>'POST')) !!}
09.         @if (count($errors) > 0)
10.             <div class="alert alert-danger">
```

❑ Tạo view trong resoueces/view/

```
11.      <ul>
12.          @foreach ($errors->all() as $error)
13.              <li>{{ $error }}</li>
14.          @endforeach
15.      </ul>
16.  </div>
17.  @endif
18.  {!! Form::text('phone', old('phone'), ['placeholder' => 'Enter Indian phone number']) !!}
19.  <br/>
20.  {!! Form::submit('Save') !!}
21. </form>
22. </div>
23. </body>
24. </html>
```




DEMO


-Demo xác thực số điện thoại
VietNam bắt đầu “+84”



Phần I: Tổng quan Validation

 Giới thiệu Validation

 Các thành phần trong Validation

 Cấu hình sử dụng Validation

Phần II: HTTP Basic Authentication

 Tạo Routes

 Tạo Controller

 Thêm các xác thực tự định nghĩa





Cảm ơn