

TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

---

# BÀI 05:

## FORM VALIDATION & ANIMATION

## MỤC TIÊU (3-3)

Giải thích validation form trong AngularJS

Giải thích về form states và input states

Sử dụng Animations trong AngularJS

Sử dụng css classes cho Angular Animations

# FORM VALIDATION

Form là một thành phần không thể thiếu cho một trang web hay ứng dụng, thông qua đó người dùng có thể tiếp cận và tương tác trực tiếp với ứng dụng

Form được sử dụng để thực hiện dữ liệu từ người dùng

Dữ liệu thu thập được xác thực trước khi gửi đến máy chủ

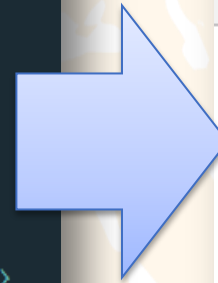
**Angular liên tục giám sát tình trạng của form** khi người dùng nhập dữ liệu trên các trường nhập liệu và thông báo cho người dùng về tình trạng hiện tại

# Form validation Required

Sử dụng thuộc tính **required** của html 5 input

Thuộc tính **required** sẽ bắt buộc người dùng nhập dữ liệu rồi mới có thể submit form

```
<!DOCTYPE html >
<html ng-app>
<head>
  <meta charset="utf-8">
  <title>Title Page</title>
</head>
<body>
  <form action="" method="POST" name="myForm" >
    <p>
      <label class="sr-only" for="">Full name</label>
      <input type="text" name="full_name" ng-model="full_name" required>
    </p>
    <p>Trang thái form khi chưa nhập full_name</p>
    <h1>{{myForm.full_name.$valid}}</h1>
  </form>
  <script src="js/angular.min.js"></script>
</body>
</html>
```



Browser window showing the form validation result:

Full name

Trang thái form khi chưa nhập full\_name

**false**

# Form validation Required

```
<!DOCTYPE html >
<html ng-app>
<head>
  <meta charset="utf-8">
  <title>Title Page</title>
</head>
<body>
  <form action="" method="POST" name="myForm" >
    <p>
      <label class="sr-only" for="">Full name</label>
      <input type="text" name="full_name" ng-model="full_name" required>
    </p>
    <p>Trang thái form khi chưa nhập full_name</p>
    <h1>{{myForm.full_name.$valid}}</h1>
  </form>
  <script src="js/angular.min.js"></script>
</body>
</html>
```



Title Page x Nested Scope Demo

file:///E:/giao-trinh-hoc-tap/LAP\_T

Full name

Trang thái form khi nhập full\_name


**true**

# FORM VALIDATION REQUIRED

Xem xét ví dụ sau

```
<!DOCTYPE html >
<html ng-app>
<head>
  <meta charset="utf-8">
  <title>Title Page</title>
</head>
<body>
  <form action="" method="POST" name="myForm" >
    <p>
      <label class="sr-only" for="">Full name</label>
      <input type="text" name="full_name" ng-model="full_name" required>
    </p>
    <p ng-show="myForm.full_name.$dirty && myForm.full_name.$invalid">
      Họ và tên không được để rỗng
    </p>
  </form>
  <script src="js/angular.min.js"></script>
</body>
</html>
```

Gõ ký tự khoảng trắng

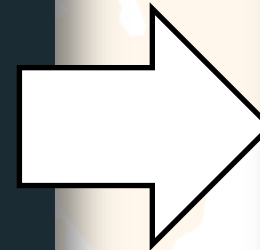


The screenshot shows a web browser window with the title 'Title Page'. The address bar displays 'file:///E:/giao-trinh-hoc-tap/LAP'. The form contains a text input field labeled 'Full name'. Below the input field, a red error message is displayed: 'Họ và tên không được để rỗng' (Full name cannot be empty). A blue arrow points from the code editor on the left to the browser window.

# FORM VALIDATION EMAIL

Tương tự như required thì email validation sử dụng thuộc tính `type="email"` của input

```
<!DOCTYPE html >
<html ng-app>
<head>
  <meta charset="utf-8">
  <title>Title Page</title>
</head>
<body>
  <form action="" method="POST" name="myForm" >
    <p>
      <label class="sr-only" for="">Email address</label>
      <input type="email" name="email" ng-model="email">
    </p>
    <p ng-show="myForm.email.$dirty && myForm.email.$invalid">
      Email không đúng định dạng
    </p>
  </form>
  <script src="js/angular.min.js"></script>
</body>
</html>
```



The screenshot shows a web browser window with two tabs, both titled 'Title Page'. The address bar displays the file path 'file:///E:/giao-trinh-hoc-tap...'. Below the address bar, there is a form with a label 'Email address' and an input field containing the text 'semail'. Below the input field, a validation message is displayed: 'Email không đúng định dạng'.

# FORM VALIDATION – CÁC VALIDATION KHÁC

Ngoài required và email thì trong html5 còn có các thuộc tính khác như

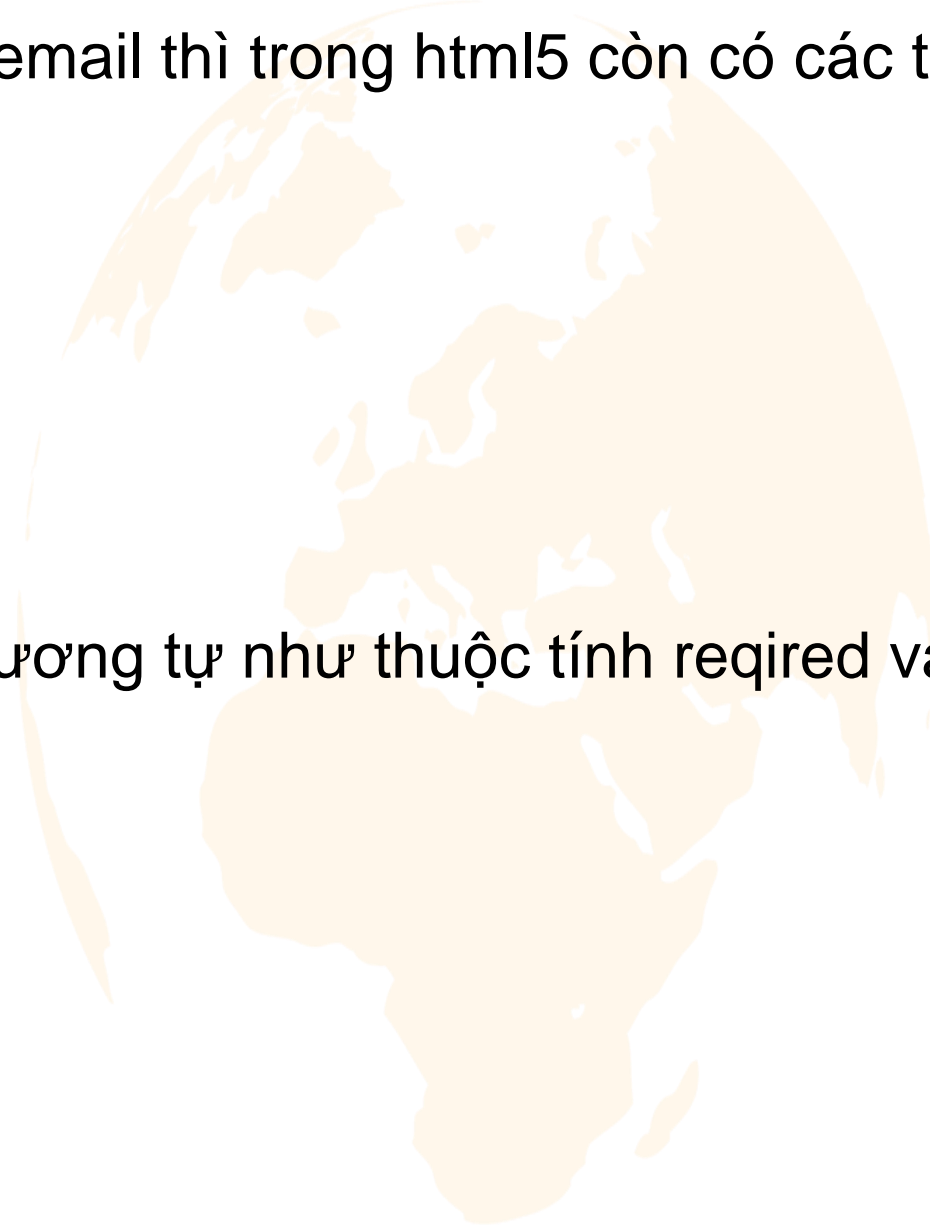
**Maxlength**

**Minlength**

**Number**

**Pattern**

Có cách sử dụng tương tự như thuộc tính required và email





# FORM VALIDATION – FORM STATES

Form là tập hợp của các nhóm điều khiển có liên quan với nhau để thực hiện một chức năng nào đó và cho phép người dùng tương tác với nó

VD: Form đăng nhập bao gồm username và password là thông tin tài khoản của người dùng cần chứng thực để đăng nhập vào hệ thống

Angular Form cung cấp một số tính năng bổ sung so với HTML đơn giản, cho phép các developer có thể kiểm soát nhiều hơn về cách giao tiếp trên form

# CÁC THUỘC TÍNH TRONG ANGULAR FORM

Thuộc tính FormController là:

\$pristine: Chưa có trường nào được sửa đổi

\$dirty: Đã sửa đổi một hoặc nhiều

\$invalid: Nội dung form không hợp lệ

\$valid: Nội dung form hợp lệ

\$submitted: form được gửi dữ liệu

# CÁC TRẠNG THÁI TRONG ANGULAR FORM

ST T	TRẠNG THÁI	MÔ TẢ
1	Pristine VÀ invalid	Khi form được hiển thị lần đầu và người dùng chưa tương tác với biểu mẫu.
2	dirty VÀ invalid	Người dùng đã tương tác với form nhưng hiệu lực vẫn chưa được đáp ứng.
3	dirty và valid	Người dùng đã hoàn thành điền vào form và toàn bộ quy tắc xác nhận đã được thỏa mãn.

# CÁC TRẠNG THÁI CỦA INPUT

STT	TRẠNG THÁI	MÔ TẢ
1	\$untouched	Trường chưa được chạm vào
2	\$touched	Trường đã được chạm vào
3	\$pristine	Trường đã được sửa đổi
4	\$dirty	Nội dung trường không hợp lệ
5	\$invalid	Nội dung trường là hợp lệ
6	\$valid	

# CÁC CLASS CỦA INPUT THEO TRẠNG THÁI

ST T	TRẠNG THÁI	MÔ TẢ
1	Ng-valid	AngularJS set class này nếu dữ liệu nhập trên input là hợp lệ và không có lỗi nào
2	Ng-invalid	AngularJS set class này nếu dữ liệu nhập trên input là không hợp lệ và có lỗi
3	Ng-pristine	AngularJS set class này khi người dùng chưa tương tác trên input
4	Ng-dirty	AngularJS set class này khi dữ liệu trên input đã được thay đổi/sửa đổi
5	Ng-touched	AngularJS set class này khi người dùng đã đặt ra ngoài input
6	Ng-untouched	AngularJS set class này khi người dùng chưa đặt ra ngoài input
7	Ng-submited	AngularJS set class này khi form được submit

# VÍ DỤ SỬ DỤNG CSS CLASS THEO FORM STATE

```
<style type="text/css">
  input.ng-pristine.ng-valid{
    background: #ececec
  }
  input.ng-invalid{
    background: red;
    color:#fff;
  }input.ng-valid{
    background: yellow;
    color:#fff;
  }
</style>
```

```
<body>
  <h1>Nhập dữ liệu đúng</h1>
  <form action="" method="POST" name="myForm" >
    <p>
      <label class="sr-only" for="">Email address</label>
      <input type="email" name="email" ng-model="email">
    </p>
    <p ng-show="myForm.email.$dirty && myForm.email.$invalid">
      Email không đúng định dạng
    </p>
  </form>
  <script src="js/angular.min.js"></script>
</body>
```

## Form mới load

Email address

## Nhập dữ liệu sai

Email address admin

Email không đúng định dạng

## Nhập dữ liệu đúng

Email address admin@gmail.com

# ANGULARJS ANIMATION

Animation là sự chuyển đổi của một phần tử HTML cho chúng ta hiệu ứng về chuyển động

Các yếu tố animation trong ứng dụng hoặc các

trang web làm tăng thêm tính thẩm mỹ và trải nghiệm người dùng.



# ANGULARJS ANIMATION

Để sử dụng hiệu ứng animation của angularJS chúng ta cần một số bước sau

1 nhúng module ngAnimate trong thư viện angular-animate.js

2 Sử dụng css3 style hiệu ứng chuyển đổi cho các phần tử dựa vào các trạng thái css class mà angular-animate cung cấp

Sau đây là bảng liệt kê các trạng thái class của một số hiệu ứng animation



# ANGULARJS ANIMATION

Một số trạng thái class hiệu ứng khi sử dụng angular-animate

Tên hiệu ứng	Class	Mô tả
Ng-hide	ng-hide	Khi phần tử bắt đầu ẩn
	Ng-hide-remove-active	Khi phần tử bắt hiện lên
	Ng-hide-add-active	Khi phần tử bắt đầu ẩn hẳn đi
Ng-class="class_name"	<b>class_name</b> -add-active	Khi class <b>class_name</b> bắt đầu thêm vào phần tử
	<b>class_name</b> -remove-active	Khi class <b>class_name</b> bắt đầu xóa khỏi phần tử

# ANGULARJS ANIMATION

Ví dụ cụ thể với ng-hide

VD dưới cho thấy: từ class **sample-show-hide** đến khi thêm class **ng-hide** trong vòng 2s

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Example - example-animate-ng-show-production</title>
  <link href="css/style.css" rel="stylesheet" type="text/css">
  <script src="https://code.angularjs.org/snapshot/angular.min.js"></script>
  <script src="https://code.angularjs.org/snapshot/angular-animate.js"></script>
</head>
<body ng-app="ngAnimate">
  <div ng-init="checked = true">
    <label>
      <input type="checkbox" ng-model="checked" />
      Is visible
    </label>
    <div class="content-area sample-show-hide" ng-show="checked">
      Content...
    </div>
  </div>
</body>
</html>
```

```
.content-area {
  border: 1px solid black;
  margin-top: 10px;
  padding: 10px;
}

.sample-show-hide {
  transition: all linear 2s;
}

.sample-show-hide.ng-hide{
  opacity: 0;
}
```

# ANGULARJS ANIMATION

Ví dụ cụ thể với ng-class

VD dưới cho thấy: từ chuyển đổi giữa class bggreen và bgblue trong vòng 2s

```
<div class="container" ng-class="checked ? 'bgblue' : 'bggreen'">
  <div >
    <label>
      <input type="checkbox" ng-model="checked" />
      Is visible
    </label>
    <div class="content-area sample-show-hide" ng-show="checked">
      Content...
    </div>
  </div>
</div>
```

```
.bggreen{
  transition: all linear 2s;
  background: green;
}
.bgblue{
  transition: all linear 2s;
  background: blue;
}
```

TRƯỜNG ĐÀO TẠO LẬP TRÌNH VIÊN VÀ QUẢN TRỊ MẠNG QUỐC TẾ BACHKHOA-APTECH

# THANK FOR WATCH!

