

Desarrollo Web Front-End con AngularJS

AngularJS Forms Validation

- Turn off HTML5 form validation:

```
<form class="form-horizontal" name="feedbackForm"  
ng-submit="sendFeedback()" novalidate>
```

Angular takes over responsibility for validation

- Make sure to give the form a name
- ng-submit directive specifies the function to call

AngularJS Forms Validation

- AngularJS offers client-side form validation.
- AngularJS monitors the state of the form and input fields (input, textarea, select), and lets you notify the user about the current state.
- AngularJS also holds information about whether they have been touched, or modified, or not.
- You can use standard HTML5 attributes to validate input, or you can make your own validation functions.

AngularJS Forms Validation Example

```
<form name="myForm">  
<input name="myInput" ng-  
model="myInput" required>  
</form>
```

```
<p>The input's valid state is:</p>  
<h1>{{myForm.myInput.$valid}}</h1>
```

Angular Form Validation Directives

- Angular validates the form fields before copying the value over to the \$scope
- Some directives of interest:
 - ng-minlength
 - ng-maxlength
 - ng-pattern: can use regex

Field Properties

- Use Field name and form name
 - e.g., `feedbackForm.firstName.$pristine` in the form, and `$scope.feedbackForm.firstName.$pristine` in the controller code

Property	Description
<code>\$pristine</code>	true if form has not been changed
<code>\$dirty</code>	reverse of <code>\$pristine</code>
<code>\$valid</code>	true if form field/whole form is valid
<code>\$invalid</code>	reverse of <code>\$valid</code>

Field Properties

- Use Field name and form name
 - e.g., `feedbackForm.firstName.$pristine` in the form, and `$scope.feedbackForm.firstName.$pristine` in the controller code

Property	Description
<code>\$pristine</code>	true if form has not been changed
<code>\$dirty</code>	reverse of <code>\$pristine</code>
<code>\$valid</code>	true if form field/whole form is valid
<code>\$invalid</code>	reverse of <code>\$valid</code>

Form classes

- `ng-pristine` - when form/input not used yet
- `ng-dirty` - when form/input has been used
- `ng-untouched` - when input has not been touched
- `ng-touched` - when input has been touched
- `ng-valid` - when a form field is valid
- `ng-invalid` - when a form field is not valid

Field Properties and CSS

- Bootstrap provides a lot of CSS classes to enable display of form validation state and messages:
 - .has-error, .has-warning, .has-success
 - .help-block to display helpful messages below the field

Form State and Input State

- AngularJS is constantly updating the state of both the form and the input fields.
- Input fields have the following states:
 - `$untouched` The field has not been touched yet
 - `$touched` The field has been touched
 - `$pristine` The field has not been modified yet
 - `$dirty` The field has been modified
 - `$invalid` The field content is not valid
 - `$valid` The field content is valid

Form State and Input State

- Forms have the following states:
- `$pristine` No fields have been modified yet
- `$dirty` One or more have been modified
- `$invalid` The form content is not valid
- `$valid` The form content is valid
- `$submitted` The form is submitted
- They are all properties of the form, and are either `true` or `false`.

Form Validation Example

http://www.w3schools.com/angular/angular_validation.asp

```
<div class="form-group" ng-class="{ 'has-error has-feedback' : feedbackForm.emailid.$invalid &&
!feedbackForm.emailid.$pristine }">
<label for="emailid" class="col-sm-2 control-label">Email</label>
<div class="col-sm-10">
<input type="email" class="form-control" id="emailid" name="emailid" placeholder="Email"
ng-model="feedback.email" required>
<span ng-show="feedbackForm.emailid.$invalid && !feedbackForm.emailid.$pristine"
class="glyphicon glyphicon-remove form-control-feedback" aria-hidden="true"></span>
<span ng-show="feedbackForm.emailid.$invalid && !feedbackForm.emailid.$pristine"
class="help-block">Enter a valid email address.</span>
<span ng-show="feedbackForm.emailid.$error.required && !feedbackForm.emailid.$pristine"
class="help-block">Enter a valid email address.</span>
</div>
</div>
```