

# Password Strength Checking with jQuery

April 12, 2012 14 comments



10



20



6

Tutorials

jQuery

Password strength checking is an easy way to show the strength of user password on the registration forms. It helps users to choose more secure password when filling the forms.

In this tutorial we will build a basic form field that provides live feedback to the users about the password strength. The basic idea is to evaluate the password string every time a user enters a character. We will check the password strength using a few regular expressions with jQuery.

Password :



Strong

[Demo](#)[Download](#)

## Calculating the Password Strength:

We need to think of some rules for the password, so that we can evaluate the password strength based on those rules. So let's add some basic rules, for example:

If password is less than 6 characters, don't accept.

If the length of password is more than 6 characters, increase the strength value by +1.

## Popular Topics

beginners

cheatsheet

CodeIgniter

css

html

html5

JavaScript

jQuery

OOP

Photoshop

php

tips

video tutorial

Web Design

WordPress

workflow

## Premium Themes

**Vienna**  
Responsive  
Magazine

WordPress

**Eris**  
Responsive  
Magazine

WordPress

## Popular Posts

**silverOrchid - WordPress Magazine Theme**

April 27 with 1,381 comments

**gazpoMag: Magazine Style WordPress Theme**

September 15 with 879 comments

**Max Magazine: A Responsive WordPress Theme**

If the password contains lower and uppercase characters, +1.  
If the password contains characters and numbers, +1.  
If the password contains one special character, +1.  
If the password contains two special characters, +1.  
Based on above rules, we can calculate the password strength value and return the feedback to the user. Initially the password strength value is 0 and for each rule it fulfills, its value is increased by 1. We can assume that if it fulfills at least two rules (ie. its value is 2), it is good password. If its value is less than 2, we assume it is a weak password and if its value is more than 2, it is a strong password.

Let's start with the code.

### Markup for the form:

Here is the basic HTML markup for the form having only one password field. The span #result contains the feedback of the password.

```
1 <form id="register">
2   <label for="password">Password:</label>
3   <input name="password" id="password" type="password"/>
4   <span id="result"></span>
5 </form>
```

### CSS Styling:

Basic CSS code for the form layout:

```
1 #register {
2   margin-left:100px;
3 }
4
5 #register label{
6   margin-right:5px;
7 }
8
9 #register input {
10  padding:5px 7px;
11  border:1px solid #d5d9da;
12  box-shadow: 0 0 5px #e8e9eb inset;
```



July 18 with 816 comments



**Curiosity: WordPress Theme**  
December 21 with 139 comments



**Smooth Page Scroll to Top with jQuery**  
February 21 with 127 comments

### Recent Comments



**test...** testing 123...

Smooth Page Scroll to Top with jQuery · 1 day ago



**Linda K** Can anyone tell me where to find the line I see when I view source:...

Max Magazine: A Responsive WordPress Theme · 1 day ago



**María López Montero** Hello! Thanks for the web design. I have some questions, I want to...

Max Magazine: A Responsive WordPress Theme · 1 day ago



**davhe** Use jquery's addClass or removeClass

HTML5 In-line text editing and storing to database · 1 day ago



**Miranda Paquet** Hello! Would you be able to tell me a way to change the headings...

Max Magazine: A Responsive WordPress

```

13     width:250px;
14     font-size:1em;
15     outline:0;
16 }
17
18 #result{
19     margin-left:5px;
20 }
21
22 #register .short{
23     color:#FF0000;
24 }
25
26 #register .weak{
27     color:#E66C2C;
28 }
29
30 #register .good{
31     color:#2D98F3;
32 }
33
34 #register .strong{
35     color:#006400;
36 }

```

We have defined CSS classes short, weak, good and strong to show the feedback messages in different colors.

## The jQuery Code:

The jQuery code starts with following:

```

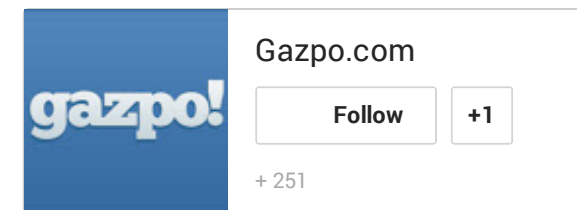
1  $(document).ready(function() {
2      //Code here
3  });

```

## Keyup event

We need to evaluate the password string as soon as the user enters a characters in the password field. So we will use a keyup event for the password input field, which is s triggered whenever the user pushes a key on the keyboard. It calls a

Theme · 3 days ago



function checkStrength by passing the password field value parameter, and appends the return value in the #result.

```
1 $( '#password' ).keyup( function() {  
2     $( '#result' ).html( checkStrength( $( '#password' ).val() ) )  
3 }
```

## checkStrength Function

As we have used a function checkStrength above, lets start that function:;

```
1 function checkStrength( password ) {  
2     //Code here  
3 }
```

Set the initial value 0 for the strength:

```
1 var strength = 0
```

## Checking the password rules

Now lets check if the first rule, if password length is less than 6 characters, we will just return the message that its too short.

We want to display this error message in red font which we have defined in the css class .weak. So, we will remove any class attached to #result and add the css class .weak.

```
1 if ( password.length < 6 ) {  
2     $( '#result' ).removeClass()  
3     $( '#result' ).addClass( 'short' )  
4     return 'Too short'  
5 }
```

If the length is valid, we can continue to the next rule. As we have set the minimum requirement for the password 6 characters, so if the user password is 6 or 7 characters, we do not increase the strength value, but if it is more than 7 characters, we increase the strength of password by 1.

```
1 if ( password.length > 7 ) strength += 1
```

Now, check if the password contains lower and uppercase characters, increase its value by 1.

```
1 if ( password.match( /[a-zA-Z]/ ) && password.match( /[0-9]/ ) ) strength += 1
```

If it has one special character, increase strength value by 1

```
1  if (password.match(/([!,%,&,@,#,$,^,*,?,_~])) strength += 1
```

If the password contains two special characters, increase strength value by 1 more

```
1  if (password.match(/(.*[!,%,&,@,#,$,^,*,?,_~].*[!,%,&,@,#,$,^,*,?,_~])) strength += 1
```

Ok, now we have calculated password strength value according to the above rules. We can return feedback messages according to the value. As I mentioned above that we will assume that password is weak if it has value less than 2. If it has value 2, it is good, and if it is more than 2, it is strong.

Following code checks if the strength is less than 2, it adds css class weak and returns the message 'Weak'. Similarly, if the value is 2, it adds the css class good, and returns the message 'Good'. If the value is more than two, it adds css class strong and returns the message 'Strong'.

```
1  if (strength < 2 ) {  
2      $('#result').removeClass()  
3      $('#result').addClass('weak')  
4      return 'Weak'  
5  } else if (strength == 2 ) {  
6      $('#result').removeClass()  
7      $('#result').addClass('good')  
8      return 'Good'  
9  } else {  
10     $('#result').removeClass()  
11     $('#result').addClass('strong')  
12     return 'Strong'  
13 }
```

That's all. Here's the full jQuery code:

```
1  $(document).ready(function() {  
2  
3      $('#password').keyup(function() {  
4          $('#result').html(checkStrength($('#password').val()))  
5      })  
6  })
```

```
6
7  function checkStrength(password) {
8
9  //initial strength
10 var strength = 0
11
12 //if the password length is less than 6, return message.
13 if (password.length < 6) {
14     $('#result').removeClass()
15     $('#result').addClass('short')
16     return 'Too short'
17 }
18
19 //length is ok, lets continue.
20
21 //if length is 8 characters or more, increase strength value
22 if (password.length > 7) strength += 1
23
24 //if password contains both lower and uppercase characters, increase strength value
25 if (password.match(/[a-z].*[A-Z]|([A-Z].*[a-z])/)) strength += 1
26
27 //if it has numbers and characters, increase strength value
28 if (password.match(/[a-zA-Z]/) && password.match(/[0-9]/)) strength += 1
29
30 //if it has one special character, increase strength value
31 if (password.match(/[!%,&,@,#,$,^,*,?,_~]/)) strength += 1
32
33 //if it has two special characters, increase strength value
34 if (password.match(/(.*[!%,&,@,#,$,^,*,?,_~].*[!%,&,@,#,$,^,*,?,_~])/)) strength
35
36 //now we have calculated strength value, we can return messages
37
38 //if value is less than 2
39 if (strength < 2 ) {
40     $('#result').removeClass()
41     $('#result').addClass('weak')
42     return 'Weak'
```


```
43     } else if (strength == 2 ) {
44         $('#result').removeClass()
45         $('#result').addClass('good')
46         return 'Good'
47     } else {
48         $('#result').removeClass()
49         $('#result').addClass('strong')
50         return 'Strong'
51     }
52 }
53 });
```


That's all. We have used only a few basic rules to evaluate password strength, you can change them and add more complex rules if you want to.

12 comments


 Join the discussion...

Newest ▾ Community Share ↗ L

 **Guest** • 4 months ago  
Showing me Parser Error , How can i resolve it



1 ^ | ▾ • Reply • Share ›

 **FriedFish** • 5 months ago  
when i erase all the text from the password text box, still i get a message of Too Short. What should i do so the erases the message too at the end. I hope you got my question.

^ | ▾ • Reply • Share ›



Ahmed FriedFish • a month ago

You may use this code:

```
//if the password length is less than 6, return message.  
if (password.length < 6 && password.length != 0) {  
  $('#passstrength_result').removeClass()  
  $('#passstrength_result').addClass('short')  
  return 'Too short'  
}
```

```
//if no password is given.  
if (password.length == 0) {  
  $('#passstrength_result').removeClass()  
  return "  
}
```

^ | v • Reply • Share ›



**xavor** Mod FriedFish • 5 months ago

You can make it return empty string :)

1 ^ | v • Reply • Share ›



**sandeep** • 6 months ago

thanks a lot....

^ | v • Reply • Share ›



**Neel** • 6 months ago

really simplifies and gives great control over valiadtion

^ | v • Reply • Share ›



**Anna Khiger** • 8 months ago

Thank you!

^ | v • Reply • Share ›



**jahdakine** • 10 months ago

Plugged this into Bootstraps progress bars. Cool!

^ | v • Reply • Share ›







love this. made a few modifications, do you have a git repo setup for this?

^ | v • Reply • Share ›



**Guest** • a year ago

It accepts space?

^ | v • Reply • Share ›



**mintik** • a year ago

Very useful for login form. Thanks.

^ | v • Reply • Share ›



**yashwanth** • a year ago

good

^ | v • Reply • Share ›



Subscribe



Add Disqus to your site

DISQ