

- [Home](#)

# [Blufusion.net](#)

php javascript jquery codeigniter web development

Subscribe by: [RSS](#)

- [CodeIgniter](#)
- [CSS](#)
- [jQuery](#)
- [php](#)
- [javascript](#)
- 

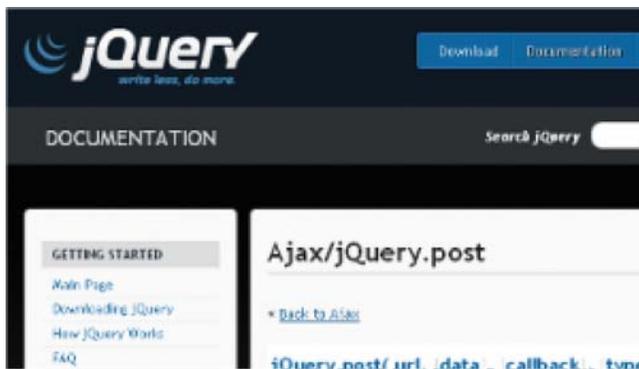
[TeamViewer vs Local Web Server](#) →

← [PHP/JS Contest: Win an iPhone 3GS at Nettuts](#)

## [Easy AJAX Post with jQuery and CodeIgniter](#)

Written by [dferg](#)

Topics: [CodeIgniter](#), [jQuery](#)



### jQuery Post

I've seen quite a few people come into the CodeIgniter forums asking how to use jQuery's built in AJAX functionality to post to a CodeIgniter function. In this tutorial I will walk you through creating the form, writing the jQuery, posting to the function, and working with handling the result. It's actually really easy.

In this example, I'll create a small, basic form that will allow user's to input their username and email address into the form and submit it via AJAX. The input will be validated using the CodeIgniter [form validation library](#), and an appropriate response will be displayed to the user.

Let's check it out.

## Creating Your Form

If you've ever submitted a form to a CodeIgniter controller before this should be very straight forward. If you haven't, it's as simple as changing the value of the *action* property of the form to point to the controller and function of choice.

```

1 <form id="ajax-form" action="/forms/submit" method="post">
2 <label for="username">Username</label>
3 <input id="username" name="username" type="text">
4
5 <label for="email">Email Address</label>
6 <input id="email" name="email" type="text">
7 <input id="form_submit" type="submit" value="submit">
8 </form>

```

Here, we will be submitting our form to a controller named *forms* and a function named *submit* within that controller. Easy stuff. Let's move on to the *forms* controller.

## Setting Up Your Controller

We're only going to be looking at the *submit* function in *forms* controller. There are no other functions used in this tutorial. In the *submit* function, we will load the form validation library, set the rules for our input, run the validation, and then display the response to the user.

There is one additional constant provided in this example that is not part of the default CodeIgniter installation. The constant I am referring to is the *IS\_AJAX* constant. This constant is used to determine whether or not the request that was received was an AJAX request, or a standard HTTP request. I have been told that this constant can be found in the CodeIgniter forums, however, I found out about it in one of [Weblee](#)'s screencasts. Simply paste this into the bottom of the *constants.php* file found at CodeIgniter *install > application > config* folder.

```

1 // Define Ajax Request
2 define('IS_AJAX', isset($_SERVER['HTTP_X_REQUESTED_WITH']) &&
  strtolower($_SERVER['HTTP_X_REQUESTED_WITH']) == 'xmlhttprequest');

```

Now let's get on to that *submit* function.

```

01 function submit() {
02     //load form validation
03     $this->load->library('form_validation');
04
05     //set form rules
06     $this->form_validation->set_rules('username', 'Username',
  'required|min_length[5]|max_length[15]');
07     $this->form_validation->set_rules('email', 'Email',
  'required|max_length[20]|min_length[6]|valid_email');
08
09     //run form validation
10     $success = $this->form_validation->run();
11
12     //set username variable

```

```

13     $username = $this->input->post('username');
14
15     //if the validation was a success
16     if ((IS_AJAX && $success) || (!IS_AJAX && $success)) { echo
"Congradulations {$username}, your form was accepted!"; }
17
18     //if validation failed
19     else { echo strip_tags(validation_errors()); }
20 }

```

You can see from the comments within the code what each portion does. When posting via AJAX, anything that is echoed, is returned back to the jQuery function. In a real world example, you would probably want to handle the result a little better by redirecting to another page if the request were not an AJAX request. The current code would only echo a response back to a blank page for a non AJAX request. If the validation fails, form validation error message is returned as HTML. Since we will be using that result in an alert dialog, I added the strip tags function to remove the paragraph tags from around the result. As mentioned, this would be handled differently in a production level app. For more information on the form validation process, check the CodeIgniter User Guide.

## The jQuery

The jQuery code is also extremely short and to the point. This script will listen for the form submit event. When the event is triggered, the event object is passed in and uses the *preventDefault* function to stop the regular submission so that we can submit via AJAX instead. Next, it will use the built-in [\\$.post function](#) to submit the data to the specified location. The function accepts parameters of *url*, *data*, *callback*, and request *type*. In this example, we were submitting the *username* and *email* fields and posting to */forms/submit*. Here's the code.

```

1  $(document).ready(function() {
2      $("form").submit(function(e) {
3          e.preventDefault();
4          $.post("/forms/submit", { username: $("#username").val(),
email: $("#email").val() }, function (data) {
5              alert(data);
6          });
7      });
8  });

```

That's it. I think we had a little less than 30 lines total even with all the extra spacing and such.

Check out the demo [here](#).



Own a business?  
Be found on 

Claim your free listing today!

[Start Now](#)

## Like My Blog Design?

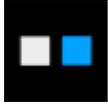
[Get your own free ThrillingTheme](#) infused with goodness by [WooThemes](#) and [FreelanceWP](#).

## Bookmark and Share



Posted on 30 July 2009

## 9 Comments Comments For This Post I'd Love to Hear Yours!



1. [Montana Flynn](#) says:  
[September 1, 2009 at 9:59 pm](#)

Thanks a bunch, you are extremely helpful in learning Codeignitor to a PHP novice like myself.

Cheers,  
Montana Flynn



2. [Will](#) says:  
[September 23, 2009 at 2:55 pm](#)

Really good job, which respect the MVC architecture proposed by CodeIgniter using JQuery methods. Moreover, it allows a user to navigate without JavaScript activated, the accessibility is respected.

Thanks,  
Will



3. [loop](#) says:  
[December 3, 2009 at 9:23 am](#)

This was a great read though! Thanks..



4. [Justme](#) says:  
[January 5, 2010 at 5:24 pm](#)

Very nice, but, if i have several forms on page how should i change the jquery code?

Thanks a lot, i hope you will help me, i really appreciate.



5. [Pr0v4](#) says:  
[February 10, 2010 at 6:50 am](#)

@Justme: Hey man! Have you solved your problem?

And I've got only another question... If I want to put the files in a database, respecting the MVC pattern I supposed to use a model to do that, but with jQuery shall I send those data straight with it?

Thanks in advance for the help!



6. *tuakaka* says:

[April 11, 2010 at 11:32 pm](#)

Thanks for your post! How can you deal with this error: uncaught exception: [Exception... "Component returned failure code: 0x80070057 (NS\_ERROR\_ILLEGAL\_VALUE) [nsIXMLHttpRequest.open]" nsresult: "0x80070057 (NS\_ERROR\_ILLEGAL\_VALUE)" location: "JS frame :: <http://localhost/mysite/js/jquery.js> :: anonymous :: line 4389" data: no]. That's what I got when clicked on submit button. Please help me with that!



7. *knp* says:

[April 30, 2010 at 12:23 am](#)

I like this tutorial thanks



8. *Peter* says:

[May 13, 2010 at 4:00 am](#)

Good job. Just one thing: use jQuery's serialize function for collecting data from the form. With this solution you don't have to manipulate your submit code when you want to use it for other forms.



9. *dim* says:

[June 8, 2010 at 7:44 pm](#)

You have to change the title to "The EASIEST AJAX Post with jQuery and CodeIgniter"

u r a god!  
thanks.

## 9 Comments Trackbacks For This Post

1. [9 Ways to Integrate Ajax with CodeIgniter | Choose Daily](#)

**Leave a Comment Here's Your Chance to Be Heard!**

Name (required)

Email (will not be published) (required)

Website

## Subscribe to Blufusion.net:



[Grab the RSS feed for Free Updates!](#)

(What's this? — [Learn more about RSS](#))

- [Featured](#)
- [Popular](#)
- [Comments](#)
- [Topics](#)
  
- [Tutorial Ideas](#)
- [TeamViewer vs Local Web Server](#)
- [Easy AJAX Post with jQuery and CodeIgniter](#)
- [PHP/JS Contest: Win an iPhone 3GS at Nettuts](#)
- [Creating A Simple Tooltip Using jQuery and CSS](#)
- [CodeIgniter From Scratch: Day 2](#)
- [50 Great CSS Tricks](#)
- [CodeIgniter: Installation and Setup](#)



### Search the Site:



## Twitter Updates

[@macappstorm](#) Does it matter? There are several I've mentioned before and they were never reviewed.



[Follow @jdfwarrior on Twitter](#)→

### Other Resources

- [Nettuts](#) Nettuts+ is a site aimed at web developers and designers offering tutorials and articles on technologies, skills and techniques to improve how you design and build websites.
- [Noupe](#) NOUPE passionately delivers stylish and dynamic news for designers and web-developers on all subjects of design, ranging from; CSS, Ajax, Javascript, web design, graphics, typography, advertising & much more.
- [Six Revisions](#) Useful Information for Web Developers and Designers



Copyright © 2010 Blufusion.net. [Thrilling Theme](#) from [FreelanceWP](#)

Proudly powered by [WordPress](#) 

☺