

# Making your javascript maintainable [closed]

Asked 15 years, 10 months ago   Modified 13 years, 7 months ago

Viewed 1k times



10



**Closed.** This question needs to be more [focused](#). It is not currently accepting answers.



**Want to improve this question?** Update the question so it focuses on one problem only by [editing this post](#).

Closed 9 years ago.

[Improve this question](#)

I am in the process of converting an internal application to use more Ajax via jQuery. I am moving away from the standard code behind of the current ASP.NET application and incorporating JavaScript methods that are going to run client side. My concern is what is the best method for allowing this application to stay maintainable for those who come behind me.

javascript

jquery

Share

asked Jan 29, 2009 at 16:16

Improve this question

Follow



Bobby Borszich

11.8k ● 9 ● 40 ● 35

I recently wrote about this

[franciscomsferreira.blogspot.com/2013/01/...](http://franciscomsferreira.blogspot.com/2013/01/...) – fmsf Jan 31, 2013 at 10:09

7 Answers

Sorted by:

Highest score (default)



9



I've found that build scripts are the key to success in maintainable Javascript. Essentially in order to have something fast and concise on the final page you need a script to make that maintainable in development.

If you use a build process you can break your JS into files based on the functionality. You can use the debug versions of libraries during development, and leave glorious code comments smattered around your code with impunity.

A build script should:

- [Lint](#) all your code
- Strip comments
- Concatenate the files together
- [Minify](#)/Obfuscate/Compress the code
- Version files for deployment
- Push the version files to the server ready to be "turned on" in production

This leaves you to work on files that make logical sense and it to tidy up after you for production.

Share Improve this answer

answered Jan 29, 2009 at 17:55

Follow



[sh1mmer](#)

1,326 ● 11 ● 20

---

awesome answer, way to sum it all up – [Bobby Borszich](#)

Jan 29, 2009 at 18:31

---

Thanks :) I'm actually planning a podcast on the subject of build scripts for next week so it was on my mind :)

– [sh1mmer](#) Jan 29, 2009 at 19:25

---



8



I would say that applying the same sort of OOP principles to your javascript that you apply to server-side code is one of the best ways to enhance maintainability. Instead of global functions, think about creating classes that encapsulate your javascript methods and data. Use javascript unit testing frameworks to test this code to make sure that it is robust.

Share Improve this answer

answered Jan 29, 2009 at 16:22

Follow



[tvanfossion](#)

532k ● 102 ● 699 ● 798



8

Run [JSLint](#) as part of your build and make sure that if it doesn't pass, it **fails your build**.



Ensuring that your JS is valid is not a key to maintainability but it will flesh out some browser issues that might have otherwise squeaked by.



If possible, start writing unit tests using a framework like Rhinounit, JSUnit or QUnit.

Share Improve this answer

answered Jan 29, 2009 at 16:26

Follow



[Kevin](#)

30.4k ● 9 ● 78 ● 83

---

thanks for providing example frameworks, very helpful

– [Bobby Borszich](#) Jan 29, 2009 at 16:40

---

JS Lint also provides notice of missing ; which can cause horrible production problems in debugging especially if you minify your code. – [sh1mmer](#) Jan 29, 2009 at 17:19

---



Create a script that runs during your build that removes all comments from javascript.

6



Then comment liberally, and encourage everyone else to also, safe with the knowledge that comments won't make it into production.



Share Improve this answer

answered Jan 29, 2009 at 16:21

Follow



[Tom Ritter](#)

101k ● 31 ● 142 ● 174



I would suggest putting as much of the javascript code as you can into external script files, to add some modularity.

4



But beyond that, comments and keeping the code and markup as clean as possible are the best that can be done in any programming project, regardless of language.

Share Improve this answer

answered Jan 29, 2009 at 16:19

Follow



[cdeszaq](#)



31.3k ● 27 ● 121 ● 175

---

Good idea for development, testing, but you may want to have combined/minimized files for use in production to reduce the number of requests and the size of responses.

– [tvanfossan](#) Jan 29, 2009 at 16:23

---

I think this is a good suggestion, just that there are more complete answers that I voted up. Thanks

– [Bobby Borszich](#) Jan 29, 2009 at 18:28

---



3



Also, if you can, try to make a bit of an object-oriented design if you can. I was once on a project that involved a lot of jQuery/Javascript, and it did not have an object-oriented design, which made it difficult for us to develop and maintain it for a while. Over time, we actually ended up rewriting parts of it to make it better.

So my best advice is to think ahead, comment (like others have said), and make sure that you code your Javascript the same way you'd code anything else (as in, follow all the good design principles that you'd use on a server-side language.)

Share Improve this answer

answered Jan 29, 2009 at 16:24

Follow



Sasha

1,190 ● 7 ● 10



3

[Nicholas Zakas: "Maintainable JavaScript" on Yahoo!](#)

Video

Share Improve this answer

edited May 11, 2011 at 9:44



Follow



answered Jan 29, 2009 at 17:09



I.devries

8,908 ● 1 ● 30 ● 29

@Downvoter, you could've just commented that the link was broken. Anyway, I put up a working link now – [I.devries](#) May 11, 2011 at 9:45