



Web Application Development

BY ATA SASMAZ



Log UI Errors

JavaScript allows exceptions to be caught and it's feasible to send them to an error logging service via ajax requests. Otherwise it's difficult to intercept UI errors in web environments.



Interchangeable data layer

The data layer should be detachable and exchangeable with another data layer that conforms to the same contracts.



Automated deployment process

Deployment process should be automated and project files for production environment should be generated by a deployment server and be deployed automatically without human touch.



Use VCS

A version control system keeps a history of all code changes (thus serving as code backup) and keeps track of why certain pieces of code have been added – and hence is essential for collaboratively working on a codebase. While [git](#) is currently the most popular VCS, you might also try [SVN](#) or [TFS](#). [GitHub](#) is the most popular VCS provider for open source projects; a free alternative is [BitBucket](#) or [GitLab](#) (for self-hosting). Microsoft has [Team Foundation](#) with extra features for collaboration.



Code review

Nobody writes perfect code all the time, so a code reviewing system is necessary to keep code quality high. Code review also allows more than one developer to become familiar with the code – so if the original author of the code is not available, another developer can make changes easily. [GitHub](#), [GitLab](#) and [Team Foundation](#) (among others) provide code reviewing features.



Permissions and roles system

Every application needs permissions and roles. Application admins and user organization admins are a minimum. For other roles, a flexible, global roles system is required.



Log all unhandled errors

All errors should be logged globally for future inspection. No error should be able to pass the global error logger.

✓	
<input type="radio"/>	<h3>Automated Testing</h3> <p>Include as many as you can: Unit tests, End to End tests, Integration tests, Component interface tests, System tests, Performance tests, Acceptance tests.</p> <p>Apply these tests to all levels of your application – especially to your JavaScript code, if you’re writing a SPA.</p>
<input type="radio"/>	<h3>Continuous Integration</h3> <p>Setup a continuous integration environment (e.g. with Jenkins) which, on every commit to your VCS, fetches the latest source code, builds it, executes all kind of automated tests and then deploys it to a staging environment.</p> <p>Never deploy from your workstation; always deploy from your CI server.</p>
<input type="radio"/>	<h3>Guidelines for developer machine configuration</h3> <p>One of the most time consuming (and pointless) problems occurs when different developers have different development environments. Let people know what should they install, which version, with which components, and how.</p>
<input type="radio"/>	<h3>Business layer should work in different environments</h3> <p>The code in the business layer must be generic. Even if it’s targeted for web environments, it should also work in desktop, server, and mobile environments, with a different user interface and data layer, without changing any code.</p>
<input type="radio"/>	<h3>Define standards for coding</h3> <p>A well-defined coding standard plays an important part in the future of the project. Does every method need a comment? What are the naming conventions? Where should the sample code usages go?</p>
<input type="radio"/>	<h3>Use and know the tooling you already have</h3> <p>Know the tools already available, e.g., Chrome DevTools and Firebug. Use them to inspect your web application, find performance hits and other potential problems. Also check out popular productivity extensions for your browser like PostMan (or Advanced REST client), various JSON output prettifiers, etc.</p>

ABOUT THE AUTHOR



ATA SASMAZ is a software engineer specializing in web application architectures. He blogs regularly at www.ata.io.

DZONE RESOURCES

BROWSE OUR COLLECTION OF 250+ FREE RESOURCES, INCLUDING:

RESEARCH GUIDES: Unbiased insight from leading tech experts

REFCARDZ: Library of 200+ reference cards covering the latest tech topics

COMMUNITIES: Share links, author articles, & engage with other tech experts

[JOIN NOW](#)



DZone communities deliver over 6 million pages each month to more than 3.3 million software developers, architects and decision makers. DZone offers something for everyone, including news, tutorials, cheat sheets, research guides, feature articles, source code and more.

"DZone is a developer's dream," says PC Magazine.

Copyright © 2014 DZone, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by means electronic, mechanical, photocopying, or otherwise, without prior written permission of the publisher.

DZone, Inc.
150 Preston Executive Dr.
Suite 201
Cary, NC 27513
888.678.0399
919.678.0300

Refcardz Feedback Welcome
refcardz@dzone.com
Sponsorship Opportunities
sales@dzone.com

ISBN-13: 978-1-936502-77-6
ISBN-10: 1-936502-77-1

