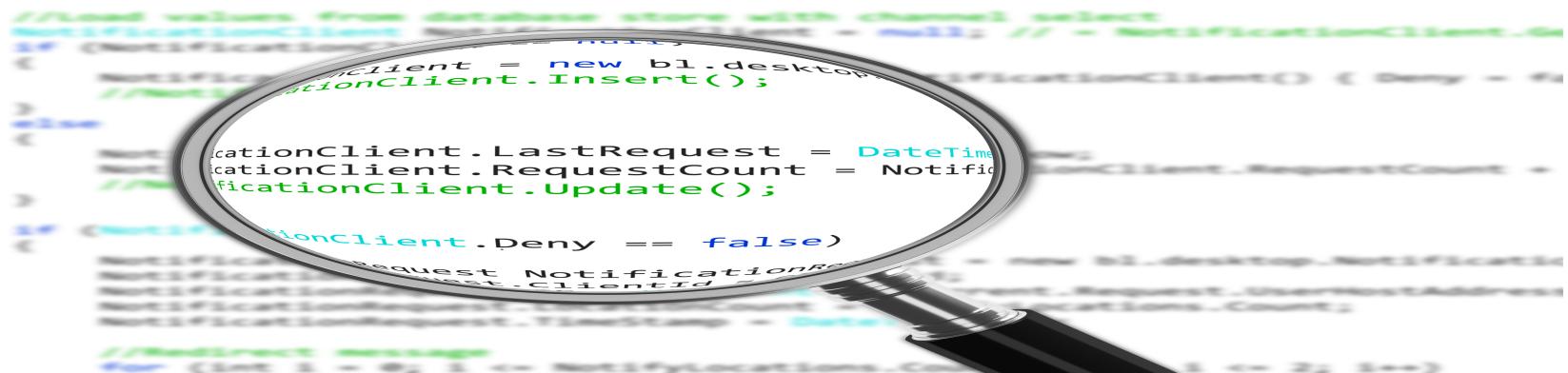


Conquering Uncomfortable Code Reviews



Patrick Schwisow

Chicago PHP User Group—October 19, 2015

What is a Code Review?

“Code review is systematic examination...of computer source code.

It is intended to find and fix mistakes overlooked in the initial development phase,

improving both the overall quality of software and the developers' skills.”

(http://en.wikipedia.org/w/index.php?title=Code_review&oldid=530204872)

Who does Code Reviews?

- Peers (other developers)
- Managers / Team Leads / Senior Developers
- Dedicated QA / Testing Personnel
- All of the above

Code Reviews in Your Workflow

- Code review is a required step
- Review as soon as possible
- Functional testing—together or separate?
- Include reviews in all estimation

Code Review Prerequisites

- Issue Tracking
 - JIRA
 - Trac
 - GitHub Issues
- Version Control System
 - Git
 - Mercurial
 - Subversion



Code Review Tools

- GitHub Pull Requests
- Bitbucket Pull Requests
- Atlassian Stash
- Atlassian Crucible
- Review Board
- Anything that gives you a diff

Getting Ready

- Agree on process
- Agree on standards
- Automate where possible
- Plan for disagreements
- Everything is subject to future re-evaluation



Rules of Code Reviews

- Review *ALL* the changes.

A screenshot of a Twitter post from the account "I Am Devloper" (@iamdevloper). The post contains the following text:
10 lines of code = 10 issues.
500 lines of code = "looks fine."
Code reviews.
The post was made at 3:58 AM - 5 Nov 2013. It has 6,749 retweets and 3,033 likes. There is a "Follow" button next to the author's name.

I Am Devloper
@iamdevloper

Follow

10 lines of code = 10 issues.

500 lines of code = "looks fine."

Code reviews.

3:58 AM - 5 Nov 2013

6,749 3,033

Rules of Code Reviews

- Review all the agreed upon criteria
 - Functionality
 - Architecture
 - Efficiency
 - Code style
 - Readability

Rules of Code Reviews

- If it looks wrong, ASK!!!
- If you are unsure, ASK!!!
- Not all comments / questions require a code change.



Rules of Code Reviews

- It's OK not to say anything.



Rules of Code Reviews

- Correct is good enough.
- Facts are better than opinions.
- Be specific.
- Cite sources whenever possible.
- Discuss “off-line” if needed, but document the resolution.

Rules of Code Reviews

- What is missing?
 - Completeness
 - Consistency
 - Error handling
 - Edge cases
 - Test coverage
 - Documentation
 - Side effects?



Rules of Code Reviews

- Nothing goes to prod until it's been reviewed.
- Nothing goes to prod until it's been tested.



Rules of Code Reviews

- If it's out of scope, create a ticket.



Rules of Code Reviews

*"Don't see code reviews as attacks on your person.
They are lessons."*



–Stefan Koopmanschap

Wisdom of the ElePHPant

wisdomoftheelephpant.com

Recommended Reading

- Mark Chu-Carroll:
[Things Everyone Should Do: Code Review](#)
- Jeff Atwood: [Code Reviews: Just Do It](#)
- Steve McConnell: *Code Complete: A Practical Handbook of Software Construction*
(<http://amzn.com/0735619670>)
- Karl Wiegers: *Peer Reviews in Software: A Practical Guide* (<http://amzn.com/0201734850>)

Who Am I?

Feedback / Contact / Slides

- Software Engineer at [Shutterstock](#)
- Zend Certified Engineer – PHP 5 & Zend Framework
- Founder / Organizer of [Lake / Kenosha PHP](#)
- Email: patrick.schwisow@gmail.com
- Twitter: [@PSchwisow](#)
- Slides: <https://github.com/PSchwisow/Miscellaneous/>

