



Software quality in eCommerce projects

Agenda

- What?
- Why?
- Where?
- How?
- Questions?

| **What?**

the totality of features and characteristics of a product or service that bears its ability to satisfy stated or implied needs

ISO 8402-1986

Wordpress: a fractal of bad design

Everyone knows about WordPress, and everyone knows WordPress is just awful. If you happen to think WordPress is awesome, hopefully your mindset will change by the end of this post. I'll just state this now — I hate WordPress. I despise its popularity but I understand why it is so, I want to shoot myself when I look at the source code, and I try my hardest to avoid any WordPress related projects. But the worst problem of all is it breeds ignorance (e.g., "WordPress developers"), promotes backwards standards and implements inefficient design patterns.

Inspired by eevee's post on [PHP: a fractal of bad design](#), this post will attempt to do the same about WordPress. Any point eevee made about PHP, double that for WordPress. I'll try my best not to bash everything, but it's really hard not to. I've literally revised this post 4 times to tone it down.

I've worked with WordPress through the years, even way back in 1.x days. I recently had to maintain a few 3.5 and 3.6 installs for my previous employer, as well as interact with their VIP service (but that's a whole topic in itself). I've written plugins, widgets, modified the admin, yadda yadda. I'd say I have a good grasp on WordPress as a whole, and that's why I'm writing this post. I always get asked by other developers why I hate WordPress, so here's a definite list of the largest problems with the WordPress codebase, but I'm also **not the only one who thinks so**.

Before I begin, I would like to state that popularity does not equate to quality. Stop using that as an excuse to use WordPress.

| **Why?**

01

Solid

Stable

Scalable

Performant

Flexible

Extendable

02

Predictable

Less re-work

Less variation in productivity

Better performance

03

Reputation

Building quality software

Part of the brand

Customers seek out

Hard to establish

04

Employee Morale

Proud of work

Higher productivity

Happy

05

Customer satisfaction

Long term relationship

Comes back with more

Positive referrals

| **Where?**

Coding styles

Design patterns

Law of demeter

Single responsibility

Open/closed

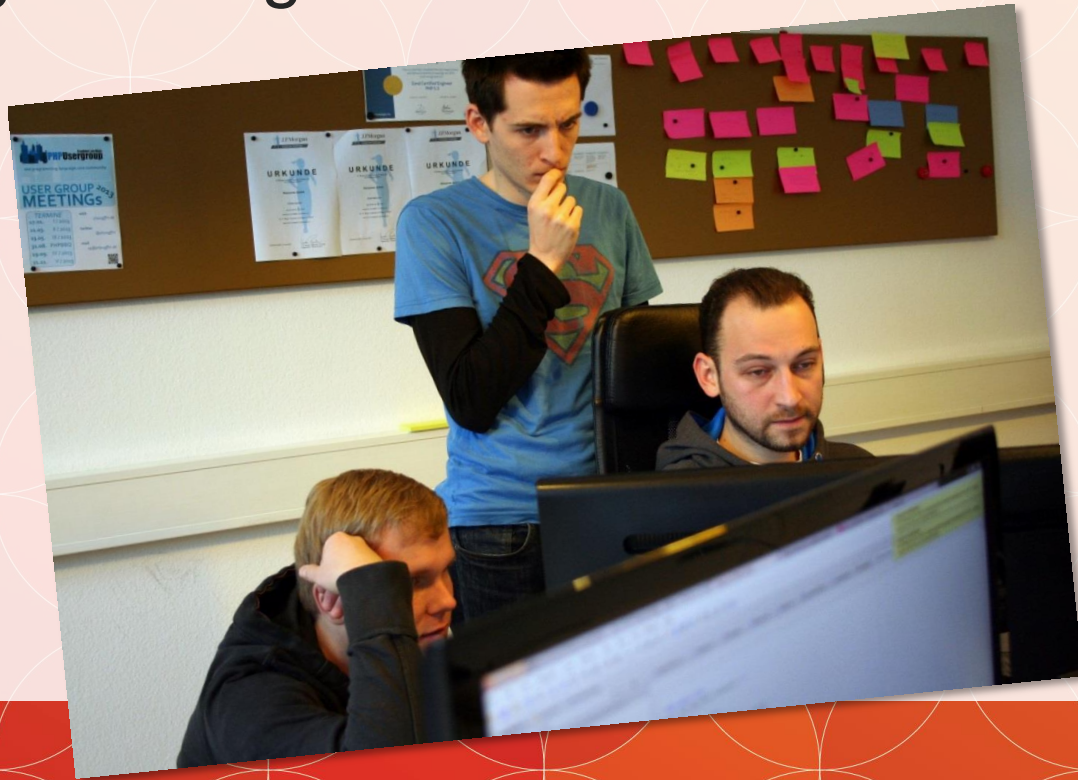
Liskov substitution

Interface segregation

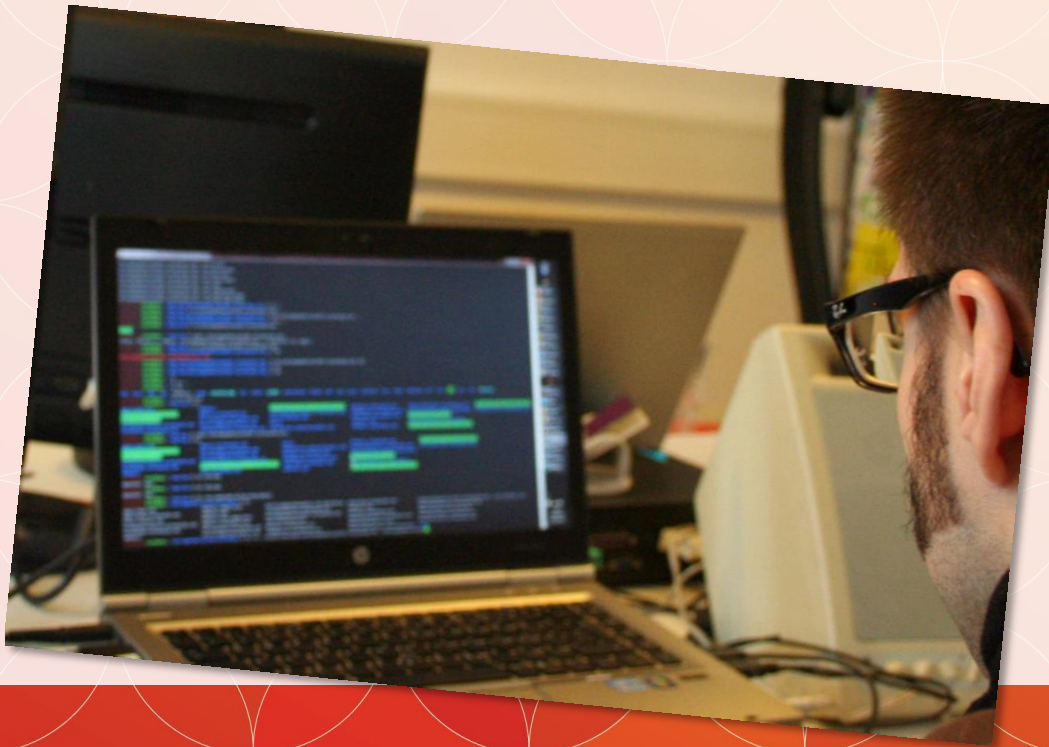
Dependency inversion

| **How?**

Pair Programming



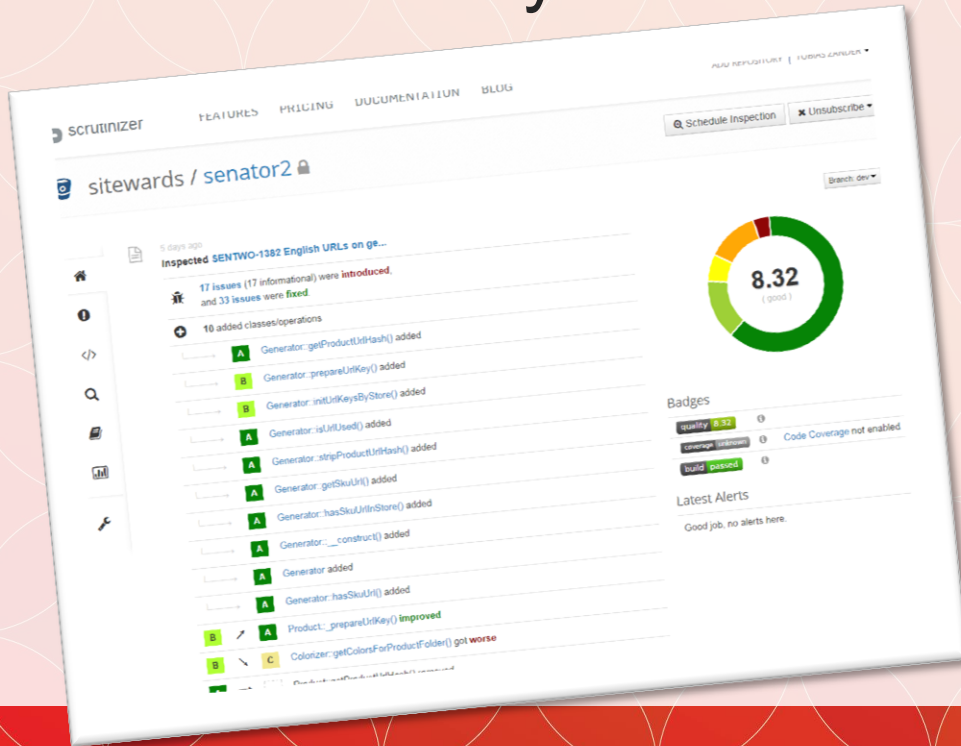
Code Reviews



Automate code inspection

- PHPUnit
- Selenium
- PHP Code Sniffer
- PHP Code Similarity Analyzer
- PHP Change Tracking Analyzer
- PHP Mess Detector
- PHP PDepend
- PHP Analyzer
- Security Advisor Checker
- JSLint
- ...

Nice and sexy ...



Performance Monitoring



Train developers

Certifications

Conferences

Usergroups

www.firegento.com

Knowledge Sharing

| Questions?



Thanks!

tobias.zander@sitewards.com