Ihr Webentwickler
Inhaber: Gunnar von Spreckelsen

www.ihrwebentwickler.de
service@ihrwebentwickler.de
Tel. 01525 3479709

Checkliste Code-Review
1. Ausgabe Februar 2015

Die Checkliste wird verwendet, um die Qualität von Code und Systemen zu überprüfen.

1.) Primäre Richtlinien

1.1 Sicherheit: Ist gewährleistet, das alle verwendeten Variablen in Front-End-Templates escaped sind (cross-site-scripting), wenn kein HTML-Inhalt benötigt wird?
1.2 Sicherheit: Sind alle Eingabe-Parameter (get, request, post, put, cookie) escaped?
1.3 Sicherheit: Werden bei SQL oder DQL-Statements durchgängig Prepared-Statements verwendet?
1.4 Performance: Sollten Daten oder Datenbanken in systemlastigen Umgebungen gecached werden?
1.5 Internationalität (I18N): Sind alle Inhalte übersetzt und nicht hardcoded verankert?
1.6 Qualität: Tauchen Syntax-, Runtime- oder Warnungs-Fehler-Ausgaben auf?
1.7 Qualität: Sind alle Test-Bestandteile aus dem Code entfernt, inclusive Degugging-Ausgabe-Anweisungen wie „die, „print, „echo“ oder „var_dump“?
1.8 Sicherheit: Existieren sensitive Daten in Logs?
1.9 Sicherheit: Ist die Sicherheitsumgebung, insbesondere das Session-Management, sauber implementiert?

2.) Sekundäre Richtlinien

2.1 Ist der Code entsprechend nach den Design-Vorgaben umgesetzt?
2.2 Sind die Coding-Guidelines respektiert wurden?
2.3 Speichert das Debugging nur wichtige und sinnvolle Daten?
2.4 Ist die Funktionalität des Codes hochwertig und gekapselt (Methoden sind für eine Aufgabe zuständig, Klassen verfolgen einen definierten Zweck) ?
2.5 Sind recursive Methoden/ Funktionen durchgängig getestet?
2.6 Als Code-Reviewer soll man den Code auf Anieb verstehen können. Falls der Code-Reviewer einen unverständlichen Code vorfindet, ist der Review nicht abgeschlossen.

3.) PHP/ Zend Framework speziell

3.1 Sind Unit-Tests für neuen oder geänderten Code geschrieben und laufen die Tests erfolgreich?

3.2 Wurde das Exception-Handling sauber umgesetzt?

3.3 Existiert Programm-Logik in Views?

3.4 Wurden deprecated PHP-Funktionen oder Bibliotheken verwendet?

3.5 Wurden immer Zend Framework-Bibliotheken und Methoden der SPL verwendet anstatt eigene Implementierungen, falls verfügbar?