



SOURCE CODE QUALITÄT

Sichern Sie nachhaltig das technische und wirtschaftliche Überleben Ihres Softwaresystems

Codequalität ist primär auf nichtfunktionale Anforderungen wie z.B. Modularität, Wiederverwendbarkeit, Analysierbarkeit, Modifizierbarkeit, Testbarkeit, Verständlichkeit des Quellcodes sowie auf formale Anforderungen wie Programmierstil und Einhaltung von Best-Practices (z.B. Patterns, Style-Guides, etc.) fokussiert. Durch immer komplexere Softwaresysteme stellt Codequalität eine fundamentale Säule der Softwareentwicklung dar.

- Statische Codeanalyse
- Code-Reviews
- Softwaremetriken
- Coding Guidelines
- Coding Process
- Clean Code
- Developer-/Unit-Tests
- Behaviour-/Test-Driven



"Any fool can write code that a computer can understand. Good programmers write code that humans can understand."

Martin Fowler

Ihr Nutzen

Wirtschaftlichkeit

Höhere Wirtschaftlichkeit der Softwareentwicklung durch guten Quellcode

Risikoreduktion

Identifizierung von Risikostellen im Code bevor Qualitätsprobleme auftreten, Reduzieren der persönlichen Haftung im Schadensfall

Qualitätsverbesserung

Einheitlichkeit und Lesbarkeit des Codes über Personen/ Teamgrenzen hinweg, mehr Codequalität bedeutet weniger Fehler und bessere Wartbarkeit, Feststellen von wichtigen Veränderungen durch Trendanalysen

Prozess

Erfahrungen von guten Programmierern in den Entwicklungsprozess einbringen, Erfüllen von Compliance-Anforderungen und Safety-Life-Cycle-Anforderungen

Unsere Leistungen

- Initialanalyse zur Feststellung der aktuellen Codequalität Ihrer Software
- Aufbau und Einführung eines statischen Codeanalyse- und Metriksystems
- Schulungen der Entwickler (Code-Reviews, Clean Code, Unit-Testen, TDD, BDD, Usability, Architektur, etc.)
- Operatives Coaching der Entwickler und regelmäßige Impuls-Workshops
- Definition und Institutionalisierung von Coding-Guidelines und passende Tools für statische Analyse und Metriken
- Z Einführung von Verfahren zur Quellcodeverbesserung (z.B. Refactoring, Patterns)
- Code-Reviews durch Experten zur regelmäßigen Prüfung der Codequalität
- Coding-Prozess strukturieren und modernisieren und Etablieren von Developer-Tests (z.B. TDD, Unit-Tests)