

Continuous Integration: Softwarequalität dauerhaft sicherstellen

Continuous Integration ist der Schlüsselprozess agiler Softwareentwicklung, die von immer mehr Firmen umgesetzt wird. Anstatt viele Softwareteile nach längerer, unabhängiger Entwicklung in einer großen Integration zusammenzuwerfen, setzt CI auf möglichst häufige Integration kleiner Bausteine. Dieser praxisorientierte Workshop befasst sich mit dem Prozess der fortlaufenden Integration von Komponenten zu einem System mit dem Ziel, die Softwarequalität zu steigern. Dabei wird typischerweise durchgehend überprüft, ob sich die Anwendung noch erstellen lässt. Um die Qualität der Software noch weiter zu erhöhen und beständig zu gewährleisten, sollten sich diesem Prozess noch weitere Schritte wie automatisierte Tests anschließen.

Ziele - Ihr Nutzen

Durch Continuous Integration wird die entstehende Software fortlaufend getestet, d.h. die Qualität wird gemessen, und Fehler werden früh erkannt. Dadurch werden spätere Kosten für Fehlersuche und -behebung erheblich reduziert.

Teilnehmer

Testingenieure, Test-Manager, Software-Entwickler, Software-Architekten, System-Architekten, Teamleiter, Projektleiter

Voraussetzungen

Praktische Kenntnisse in der Programmierung mit C/C++. Vorkenntnisse im Testen sind nicht notwendig.

Continuous Integration: Softwarequalität dauerhaft sicherstellen

Inhalt

Begriffsklärung: Softwarequalität und Versionsverwaltung

Was ist Continuous Integration?

- Warum Continuous Integration?
- Wie setze ich Continuous Integration ein?

Vorstellung des Beispielprojekts und Vorbereitungen zu CI

- Einrichten der Infrastruktur für den Praxisteil
- Aufsetzen eines VCS am Beispiel Subversion
- Erstellung des Testframeworks Google Test
- Projekterstellung und Cross-Compilation mit NetBeans
- Probelauf des Beispielprojektes mit BeagleBone und Workstation

Vorstellung des CI-Servers

- Vorstellung CI-Server Jenkins
- Automatisierung des Projekt-Buildprozesses mit Jenkins

Grundlagen professioneller Softwaretests

- Notwendigkeit des Testens
- Was ist ein statischer Softwaretest und welche Arten gibt es?

Statischer Test

- Wie bereite ich einen statischen Test für CI vor (am Beispiel CppCheck)?
- Durchlauf mehrerer CI-Zyklen mit statischen Tests

Dynamischer Softwaretest

- Grundlagen und Begriffsklärung

Unit Test

- Vorbereitung, Erläuterung und Einstieg in die Übung Unit Test

Dynamischer Test und Einbindung CI

- Fertigstellung der Übung zum Unit Test
- Erläuterung und Übung zum Integrationstest
- Erläuterung und Übung zum Systemtest

Gruppenübung CI

- Erweiterung der Sourcen in Teamarbeit
- Durchlauf möglichst mehrerer CI-Zyklen

Präsenz-Training

Preis *	Dauer
-	2 Tage

Anmeldecode: INTEG

* Preis je Teilnehmer, in Euro zzgl. USt.

Coaching

Unsere Coaching-Angebote bieten den großen Vorteil, dass unsere Experten ihr Wissen und ihre Erfahrungen direkt in Ihren Lösungsprozess einbringen und damit unmittelbar zu Ihrem Projekterfolg beitragen.

Für Ihre Anfrage oder weiterführende Informationen stehen wir Ihnen gern zur Verfügung.