

Architektur technischer Systeme

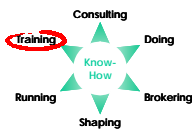
Software Engineering & Technologie

Kursziel Der Teilnehmer kennt die grundlegenden Bestandteile einer Architektur für technische Software-Systeme sowie die Mittel der UML, um solche zu dokumentieren. Er ist in der Lage einige wichtige Mechanismen und Design Pattern für technische Software-Systeme anzuwenden.

Zielpublikum Projektleiter und Software-Architekten

Voraussetzungen Kenntnisse, wie technische Software-Systeme spezifiziert werden sowie praktische Erfahrungen in einer objektorientierten Programmiersprache.

Kursinhalt Einführung Design



- Design Patterns
- Arten des Designs
- Empfohlene UML-Stereotypen im Design
- Techniken im Design

Makroarchitektur

- Design Patterns für Interrupt-Verarbeitung, parallele Tasks und Prozesse
- Design Patterns für Sicherheit und Zuverlässigkeit

Mikroarchitektur

- Design Patterns für Zustandsmaschinen
- Design Patterns für Objekt-Kollaborationen
- Design Patterns für Kommunikation
- Design Patterns für sicherheitskritische Systeme



Zusammenfassung

- Literatur

Vermittlungsmethode Vortrag, Theorieübungen anhand einer durchgängigen Fallstudie, Diskussion

Dauer 2 Tage

Teilnehmerzahl maximal 12

Kontaktadresse	KnowGravity Inc.	Telefon	+41 44 43 42 000
	Hohlstrasse 534	Fax	+41 44 43 42 009
	8048 Zürich	Internet	www.knowgravity.com
	Schweiz	E-Mail	info@knowgravity.com