

Modulhandbuch BaTIN2012_Software-Management

Modul

Anerkennbare Lehrveranstaltung (LV)

Organisation

Modulprüfung

Prüfungselemente

Vorlesung / Übung

Praktikum

Verantwortlich: Prof. Dr. René Wörzberger

Modul

Anerkennbare Lehrveranstaltung (LV)

- F07 SM

Organisation

| Bezeichnung | |
|-------------|-------------------------------|
| Lang | BaTIN2012_Software-Management |
| MID | BaTIN2012_SM |
| MPID | |

| Zuordnung | |
|-----------------|-----------|
| Studiengang | BaTIN2012 |
| Studienrichtung | H |
| Wissensgebiete | H_WIN |

| Einordnung ins Curriculum | |
|---------------------------|-------|
| Fachsemester | 3-6 |
| Pflicht | |
| Wahl | H_WIN |

| Version | |
|------------|------------|
| erstellt | 2018-06-14 |
| VID | 1 |
| gültig ab | WS 2018/19 |
| gültig bis | |

Zeugnistext

de

Software-Management

en

Software Management

Unterrichtssprache

Deutsch, 0...100%Englisch, 0...100%

Modulprüfung

| Form der Modulprüfung | |
|-----------------------|---|
| sMP | Regelfall (bei großer Prüfungszahl: sK) |

| Beiträge ECTS-CP aus Wissensgebieten | |
|--------------------------------------|---|
| H_WIN | 5 |
| Summe | 5 |

Aufwand [h]: 150

Prüfungselemente

Vorlesung / Übung

| Form Kompetenznachweis | |
|------------------------|-------------------------------------|
| bÜA | Präsenzübung und Selbstlernaufgaben |

| Beitrag zum Modulergebnis | |
|---------------------------|-----------|
| bÜA | unbenotet |

Spezifische Lernziele

Kenntnisse

- grundlegende Werkzeuge und Methoden der professionellen, arbeitsteiligen Entwicklung und Dokumentation großer Softwaresysteme verstehen (PFK.1,PFK.3,PFK.8,PFK.9,PFK.10,PFK.11,PSK.1)
 - relevante Betriebssystemgrundkenntnisse
 - Issue-Management
 - Dokumentationsformen
 - Kollaborationswerkzeuge
- den Nutzen, die Verwendung und Funktionsweise von Versionsverwaltungssystemen verstehen (PFK.3,PFK.8,PFK.9)
- den Nutzen von Automatisierung in der Entwicklung verstehen sowie die Verwendung und Funktionsweise von Automatisierungs-Tools (PFK.3,PFK.8,PFK.9)
- unterschiedliche Softwarequalitäts-Kriterien kennen sowie Qualitätsmessverfahren verstehen (PFK.3,PFK.8,PFK.9)
 - Qualitätstaxonomien
 - Code-Abdeckungen
 - Qualitätsmetriken
- unterschiedliche Betriebsplattformen kennen und deren Vor- und Nachteile benennen können (PFK.3,PFK.8,PFK.9)
 - Cluster dedizierter Hardware
 - Formen der Virtualisierung
 - Cloud-Computing

Fertigkeiten

- grundlegende Werkzeuge für die Entwicklung und Dokumentation großer Softwaresysteme anwenden können (PFK.1,PFK.3,PFK.8,PFK.9,PFK.10,PFK.11,PSK.1)
- kleinere Arbeitsergebnisse systematisch durch die Anwendung von Versionverwaltungssystemen sichern sowie verschiedene Arbeitsstände wiederherstellen können (PFK.3,PFK.8,PFK.9)
- Arbeitsschritte (z.B. Kompilation, Unit-Test, Deployment) mittels entsprechender Werkzeuge automatisieren können (PFK.3,PFK.8,PFK.9)
- erreichte Software-Qualität messen und nachweisen können (PFK.3,PFK.8,PFK.9)
- entwickelte Software auf unterschiedlichen Betriebsplattformen installieren und betreiben können (PFK.3,PFK.8,PFK.9)

Exemplarische inhaltliche Operationalisierung

Erweiterung eines Softwaresystems in vordefinierten, teils nebenläufigen Arbeitsschritten mit Git, Automatisierung von Entwicklungsschritten mit Maven und Ansible, Messung und Verbesserung der Qualität gegebenen Quellcode mit Sonarqube, Integration und Test von Quellcode-Änderungen mit Jenkins und Management von Issues GitLab. Der Schwerpunkt liegt nicht auf der eigentlichen Software-Konstruktion (Coding) sondern auf begleitenden Aktivitäten in der professionellen Software-Entwicklung.

Praktikum

| Form Kompetenznachweis | |
|------------------------|--|
| bSZ | Entwicklungs- und Betriebsumgebung für großes Softwaresystem aufsetzen und betreiben |

| Beitrag zum Modulergebnis | |
|---------------------------|------------------|
| bSZ | Testat unbenotet |

Spezifische Lernziele

Fertigkeiten

- größere Softwaresysteme systematisch und nachvollziehbar im Team entwickeln (PFK.1,PFK.5,PFK.6,PFK.7,PFK.8,PFK.9,PFK.11,PSK.1)
- Entwicklungsschritte größerer Softwaresysteme automatisieren (PFK.8,PFK.9)
- die Qualität eines Softwaresystems automatisiert messen und belegen können (PFK.7,PFK.8,PFK.9)
- Softwaresysteme auf unterschiedlichen Betriebsplattformen installieren und betreiben können (PFK.8,PFK.9)

Handlungskompetenz demonstrieren

- Geeignete Entwicklungs- und Betriebsumgebungen für arbeitsteilig entwickelte, komplexe Softwaresysteme aufsetzen und betreiben (PFK.1,PFK.5,PFK.6,PFK.7,PFK.8,PFK.9,PFK.11,PSK.1)
- mit der Dynamik in realistischen Entwicklungsprojekten, beispielsweise Anforderungsänderungen oder -erweiterungen, umgehen (PSK.1,PFK.11,PFK.8,PFK.9)

Exemplarische inhaltliche Operationalisierung

Aufsetzen und Verwenden einer Umgebung für die arbeitsteilige Entwicklung eines großen Softwaresystems unter Einsatz von Versionsverwaltungssystemen wie Git, Automatisierungswerkzeugen wie Maven und Ansible, Qualitätssicherungswerkzeugen wie Sonarqube, Continuous-Integration-Werkzeugen wie Jenkins und Kollaborationssystemen wie GitLab. Der Schwerpunkt liegt nicht auf der eigentlichen Software-Konstruktion (Coding) sondern auf begleitenden Aktivitäten in der professionellen Software-Entwicklung.

Das Urheberrecht © liegt bei den mitwirkenden Autoren. Alle Inhalte dieser Kollaborations-Plattform sind Eigentum der Autoren.

Ideen, Anfragen oder Probleme bezüglich Foswiki? Feedback senden

