

# Bachelor Studiengang: Angewandte Mathematik und Informatik

<b>Modulbezeichnung: Microservices mit GO</b>	
<b>Modulcode: xxx</b>	<b>ECTS-Credits: 5</b>
<b>Modulverantwortliche(r): Prof. Dr. Jörg Striegnitz, FH Aachen</b>	
<b>Lehr- und Lernmethoden:</b>	
Vorlesung: 1 SWS	Vor- und Nachbereitung: 35 Zeitstunden
Übung: 0 SWS	Hausarbeiten / Referate u. a.: Ca. 70 Zeitstunden
Praktikum / Seminar: 3 SWS	Gesamte Arbeitsbelastung: 150 Zeitstunden
<b>Lernergebnisse:</b>	
<p><b>Fachlich:</b> Die Studierenden verstehen das Konzept der Microservices, können einfache Microservices selbständig entwerfen, programmieren, testen und ausliefern. Sie kennen die Unterscheide zu klassischen, monolithischen Architekturen, wesentliche Herausforderungen beim Umstieg auf Microservice-Architekturen und die Vorzüge von Microservices im Kontext des Cloud-Computing.</p> <p>Die Studierenden haben Basiskenntnisse im Umgang mit der Unix-Shell. Sie verstehen die Konzepte von Container-Technologien, können Container eigenständig erstellen, orchestrieren und ausliefern.</p> <p><b>Beitrag zum Gesamtqualifikationsziel des Studiengangs:</b> Die Modulinhalte ergänzen die Informatikkenntnisse um das Gebiet des Cloud Computing. Darüber hinaus werden Themen von Grundlagenvorlesungen (z.B. Datenbanken, Rechnernetze) erweitert und an praktischen Beispielen vertieft.</p>	
<b>Inhaltsbeschreibung:</b>	
<ul style="list-style-type: none"><li>• Motivation „Microservices“ (Was ist das, wozu braucht man es?)</li><li>• Grundlagen zu Unix-Shell, Editoren und -Tools</li><li>• Container-Technologie (Grundlagen, Werkzeuge, Erstellung, Orchestrierung, Auslieferung)</li><li>• Go Primer (Schnelleinstieg in die Programmiersprache Go)</li><li>• Go IDEs</li><li>• Go Tools (Packages, Linter, Test etc.)</li><li>• Continuous Integration and -Delivery mit GitLab</li><li>• Rest-APIs und Web-Services</li><li>• Herausforderung „Datenhaltung“ im Zusammenhang mit Microservices</li><li>• <i>Event-Sourcing</i> und <i>Command Query Responsibility Segregation</i></li><li>• Kommunikation mit RabbitMQ</li></ul>	
Die Themengebiete werden überwiegend praxisnah an Fallbeispielen diskutiert.	
<b>Eingangsvoraussetzungen:</b>	
<ol style="list-style-type: none"><li>1. Grundkenntnisse der objektorientierten Programmierung (erworben z.B. durch Besuch der JAVA-Vorlesung)</li><li>2. Empfohlen: Besuch der Vorlesungen „Kommunikationssysteme“ und „Datenbanken“</li></ol>	
<b>Hinweis:</b> Die Teilnehmerzahl ist auf 20 Teilnehmer begrenzt – im Zweifel entscheidet das Los.	
<b>Art der Prüfung:</b>	
Klausur (60 Minuten)	

**Literatur und Lernunterlagen:**

- Wird in der Vorlesung bekanntgegeben