



Certificate of Advanced Studies

Microservices Integration and Operations

Moderne Unternehmensanwendungen weisen oft eine Microservice-Architektur auf, weil sich damit neue Anforderungen effizienter umsetzen lassen. Im CAS Microservices Integration and Operations lernen Sie, wie man Microservices in einer Anwendung integrieren und auf einer Cloud-Plattform betreiben kann.

Inhaltsverzeichnis

1	Umfeld	3
2	Zielpublikum	3
3	Ausbildungsziele	3
4	Voraussetzungen	3
5	Durchführungsort	3
6	Kompetenzprofil	4
7	Kursübersicht	4
8	Kursbeschreibungen	5
	8.1 Microservices Integration	5
	8.2 Event-Driven Architecture	5
	8.3 Continuous Integration and Delivery	6
	8.4 Microservices Operations	6
9	Kompetenznachweis	7
10	Lehrmittel	7
11	Dozierende	8
12	Organisation	8

Stand: 27.01.2023

1 Umfeld

Immer mehr Firmen setzen bei der Entwicklung Ihrer Unternehmensanwendungen auf eine Microservice-Architektur, weil diese durch ihre Modularität verschiedene Vorteile bietet. Doch bereits bei der Entwicklung solcher Anwendungen stellen sich zur Integration unter anderem folgende Fragen:

- Sollen die Microservices synchron, asynchron oder Event-basiert miteinander kommunizieren?
- Wie werden redundante Daten der Microservices konsistent gehalten?
- Wie kann eine hohe Resilienz erreicht werden, auch wenn einzelne Microservices vorübergehend nicht zur Verfügung steht?

Aber auch der Betrieb von Microservice-basierten Anwendungen stellt neue Herausforderungen an das entsprechende Team und die verwendete Infrastruktur:

- Wie können Microservices effizient gestartet und zuverlässig heruntergefahren werden?
- Wie kann die Inbetriebnahme neuer Versionen von Microservices automatisiert werden?
- Mit welchen Metriken kann das Laufzeitverhalten von Microservices überwacht werden?
- Wie können Microservices bei Bedarf horizontal skaliert werden?

2 Zielpublikum

Das CAS richtet sich an Personen aus der Software-Entwicklung oder dem Betrieb, die Microservice-basierte Unternehmensanwendungen für Cloud-Plattformen entwickeln und betreiben wollen.

3 Ausbildungsziele

- Sie kennen die verschiedenen Integrationsmöglichkeiten von Microservices mit ihren Vor- und Nachteilen.
- Sie verstehen die Funktionsweise und Eigenschaften einer Ereignis-gesteuerten Architektur.
- Sie sind in der Lage, Microservice-basierte Anwendungen effizient, zuverlässig und skalierbar auf einer Cloud-Plattform zu betreiben.

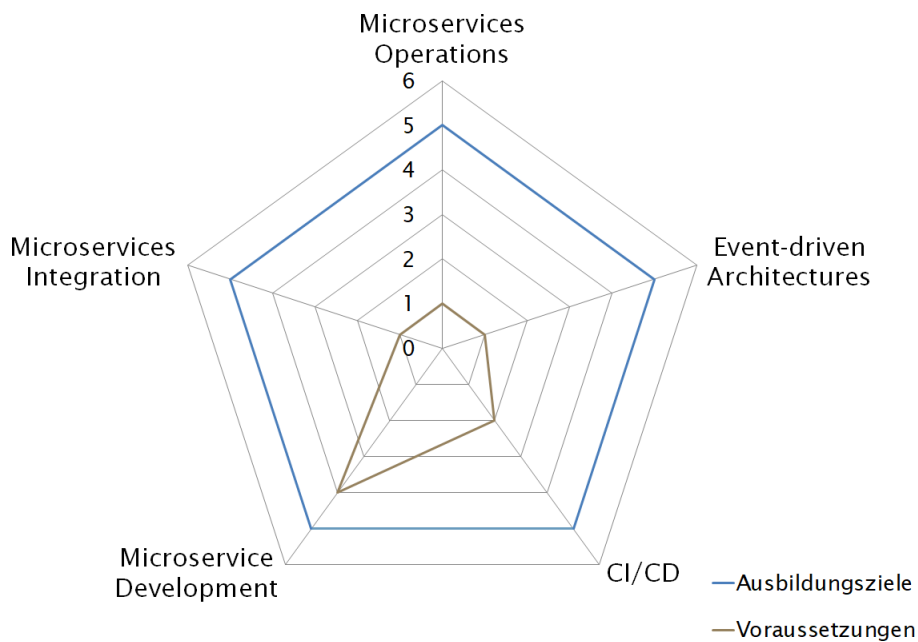
4 Voraussetzungen

Sie bringen Erfahrung in der Entwicklung von Java-basierten Microservices, zum Beispiel mit dem Spring Boot Framework mit.

5 Durchführungsort

Berner Fachhochschule Technik und Informatik, Weiterbildung Aarbergstrasse 46, 2503 Biel.
Telefon +41 31 848 31 11, E-Mail weiterbildung.ti@bfh.ch

6 Kompetenzprofil



Kompetenzstufen

1. Kenntnisse/Wissen
2. Verstehen
3. Anwenden
4. Analyse
5. Synthese
6. Beurteilung

7 Kursübersicht

Das CAS umfasst insgesamt 12 ECTS-Credits. Für die einzelnen Kurse ist entsprechend Zeit für Selbststudium, Prüfungsvorbereitung etc. einzurechnen.

Kurs / Lehreinheit	Lektionen	Stunden	Dozierende
Microservices Integration	32	24	Simon Martinelli
Event-Driven Architecture	32	24	Guido Schmutz
Continuous Integration/Delivery	32	24	Sebastian Graf
Microservices Operations	32	24	Mathias Herzog
Total	128	96	

8 Kursbeschreibungen

Nachfolgend sind die einzelnen Kurse dieses Studienganges beschrieben.

Der Begriff Kurs schliesst alle Veranstaltungstypen ein, es ist ein zusammenfassender Begriff für verschiedene Veranstaltungstypen wie Vorlesung, Lehrveranstaltung, Fallstudie, Living Case, Fach, Semesterarbeiten usw.

8.1 Microservices Integration

Lernziele	Die Studierenden kennen die Integrationsprobleme einer Microservice-basierten Applikation und die zugehörigen Lösungsmuster. Sie können die Muster mittels Spring Cloud umsetzen.
Themen und Inhalte	<ul style="list-style-type: none">– Monolith vs Microservices– Kommunikation: Remote Procedure Invocation vs Messaging vs Event-Driven– Adressierung: Service Registry, API Gateway, Backends for Frontends– Verteilte Daten: Datenbank pro Service, API Composition, CQRS, Saga, Outbox Pattern– Zuverlässigkeit/Resilienz: Retry, Circuit Breaker, Bulk Head– Beobachtbarkeit: Audit Logging, Distributed Tracing, Health Checks, Metrics– Token-basierte Security– Monolithische vs Micro-Frontends
Lehrmittel	<ul style="list-style-type: none">– Skript, das alle wesentlichen Lerninhalte umfasst

8.2 Event-Driven Architecture

Lernziele	Die Studierenden verstehen die Funktionsweise und Eigenschaften einer Ereignis-gesteuerten Architektur und können eine solche mit einer entsprechenden Plattform (z.B. Kafka) umsetzen.
Themen und Inhalte	<ul style="list-style-type: none">– Vor-/Nachteile synchroner und asynchroner Kommunikation– Event-First Thinking– Patterns: Domain Event, Event Notification, Event-carried State Transfer– Messaging: Publish/Subscribe, Point-to-Point– Zuverlässigkeit und Skalierbarkeit mittels Event Broker (Kafka)– Event-getriebene Datenverträge mit Avro, Protobuf und JSON– Basic Producer und Consumer Microservices mit Java, Python und .NET– Integration von Anwendungen mittels Change Data Capture (CDC)– Event Streaming und Stream Processing Capabilities– Event Sourcing und systemweite CQRS (Event Store, Snapshots)
Lehrmittel	<ul style="list-style-type: none">– Skript, das alle wesentlichen Lerninhalte umfasst

8.3 Continuous Integration and Delivery

Lernziele	Die Studierenden kennen die Prinzipien von Continuous Integration und Continuous Delivery und können diese auf einer entsprechenden Plattform (z.B. GitLab) effizient umsetzen.
Themen und Inhalte	<ul style="list-style-type: none"> – Repository pro Service vs Mono-Repository – Automatisierter Build-Prozess und Testen von Software (CI) – Automatisierter Release- und Delivery-Prozess (CD) – Staging und wiederverwendbare Processing Pipelines – Konfiguration von Containern mittels Templates (Helm) – Implementierung von GitOps-Paradigmen – DevOps-Metriken (MTTD/MTTR, Error Budgets, SLIs/SLOs) – Sicherheitsaspekte (Security Compliance, Library Scans) – Immutable Software
Lehrmittel	– Skript, das alle wesentlichen Lerninhalte umfasst

8.4 Microservices Operations

Lernziele	Die Studierenden kennen die verschiedenen Möglichkeiten, Microservices effizient, zuverlässig und skalierbar zu betreiben.
Themen und Inhalte	<ul style="list-style-type: none"> – Containerisierung (OCI, Docker) – Orchestrierung (Kubernetes, OpenShift) – Software Defined Networks, Service Meshs (Istio oder Consul) – Betrieb On-Premise vs Cloud (PaaS) – Deployment-Prinzipien: Rolling/Recreate, Blue/Green, Canary – Monitoring und Metriken (Prometheus, Grafana) – Automatische Skalierung – Testing: Zuverlässigkeit (Chaos Monkey), Performanz, Last
Lehrmittel	– Skript, das alle wesentlichen Lerninhalte umfasst

9 Kompetenznachweis

Für die Anrechnung der 12 ECTS-Credits ist das erfolgreiche Bestehen der Qualifikationsnachweise (Prüfungen, Projektarbeiten) erforderlich, gemäss folgender Aufstellung:

Kompetenznachweis	Gewicht	Art der Qualifikation	Erfolgsquote
Microservices Integration	2.5	Schriftliche Prüfung	0 - 100 %
Event-Driven Architecture	2.5	Schriftliche Prüfung	0 - 100 %
Continuous Integration/Delivery	2.5	Schriftliche Prüfung	0 - 100 %
Microservices Operations	2.5	Schriftliche Prüfung	0 - 100 %
Gesamtgewicht / Erfolgsquote	10		0 - 100 %

In jedem Kompetenznachweis kann eine Erfolgsquote von 0 bis 100% erreicht werden. Die gewichtete Summe aus den Erfolgsquoten der Kompetenznachweise und deren Gewichten ergibt eine Gesamterfolgsquote zwischen 0 und 100%. Diese wird in eine Note zwischen 3 und 6 umgerechnet. Die Note 3 (Gesamterfolgsquote weniger als 50%) ist ungenügend. Die Noten 4, 4.5, 5, 5.5 und 6 (Gesamterfolgsquote zwischen 50% und 100%) sind genügend.

10 Lehrmittel

Ergänzende Lehrmittel sind Empfehlungen, um den Stoff zu vertiefen oder zu erweitern. Die Beschaffung liegt im Ermessen der Studierenden.

Nr	Titel	Autoren	Verlag	Jahr	ISBN
1	Microservices Patterns	Chris Richardson	Manning	2018	9781617294549
2	Building Event-Driven Microservices	Adam Bellemare	O'Reilly	2020	9781492057895
3	SRE with Java Microservices	Jonathan Schneider	O'Reilly	2020	9781492073925
4	Building Secure and Reliable Systems	Heather Adkins et al.	O'Reilly	2020	9781492083122
5	Kubernetes in Action	Marko Luksa	Manning	2018	9781617293726
6	The Kubernetes Book	Nigel Poulton		2020	9781521823637

11 Dozierende

Vorname Name	Firma	E-Mail
Simon Martinelli	72 Services GmbH	simon@72.services
Mathias Herzog	Peak Scale	mathias.herzog@peakscale.ch
Sebastian Graf	FHNW	sebastian.graf@fhnw.ch
Guido Schmutz	Trivadis – Part of Accenture	guido.schmutz@trivadis.com

12 Organisation

CAS-Leitung:

Prof. Dr. Stephan Fischli

Tel: +41 31 848 32 74

E-Mail: stephan.fischli@bfh.ch

CAS-Administration:

Andrea Moser

Tel: +41 31 84 83 211

E-Mail: andrea.moser@bfh.ch

Während der Durchführung des CAS können sich Anpassungen bezüglich Inhalte, Lernzielen, Dozierenden und Kompetenznachweisen ergeben. Es liegt in der Kompetenz der Dozierenden und der Studienleitung, aufgrund der aktuellen Entwicklungen in einem Fachgebiet, der konkreten Vorkenntnisse und Interessenslage der Teilnehmenden, sowie aus didaktischen und organisatorischen Gründen Anpassungen im Ablauf eines CAS vorzunehmen.

Berner Fachhochschule

Technik und Informatik

Weiterbildung

Aarbergstrasse 46

CH-2503 Biel

Telefon +41 31 848 31 11

Email: weiterbildung.ti@bfh.ch

bfh.ch/ti/weiterbildung

bfh.ch/ti/cas-mio