



Berner
Fachhochschule



Certificate of Advanced Studies

Industrie 4.0 – Software Engineering (CAS I4SE)

Digitalisierung und Industrie 4.0 stellen neue Herausforderungen an das Software Engineering in den Unternehmen. Gefordert sind kurze Zyklen mit schneller Entwicklung, sofortiger Wirkung und optimaler Verbreitung von neuen Produkten. In diesem CAS lernen Sie, wie man die Applikationsentwicklung im Spannungsfeld eines modernen, vernetzten industriellen Umfeldes beherrscht.

Zielpublikum

Das CAS «Software Engineering Industrie 4.0» richtet sich an Entwickler und Fach- und Führungskräfte in Unternehmen und IT-Bereichen, die mit Entwicklung von Software der Industrie und der Dienstleistungsbereiche betraut sind.

Voraussetzungen

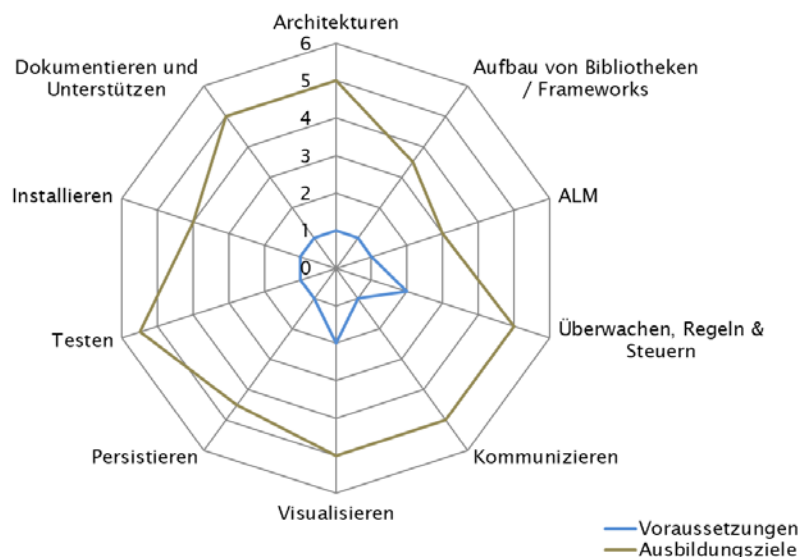
Die Teilnehmenden bringen IT- Vorkenntnisse im Rahmen einer Grundausbildung in Informatik-, Engineering (mit Software Entwicklung im Lehrplan) oder Wirtschaftsinformatik mit. Für das Verständnis von Beispielen sind Programmierkenntnisse in C#.NET von Vorteil.

Ausbildungsziele

Die Teilnehmenden erlangen Kenntnisse in folgenden Gebieten:

- Einfluss der Digitalisierung auf die Softwareentwicklung
- Standards von Industrie 4.0, eCH und ISO
- Wichtige grundlegende Architekturmuster für Desktop- und verteilte Anwendungen
- Konzeption von Schnittstellen zu Umsystemen und Wahl der korrekten Persistenztechnik
- Einsatz und Realisierung von Datenforensik
- Überwachung und Feedback der Anwendung
- Lebenszyklus der Anwendungsentwicklung
- Dokumentation von Architekturen und Software.

Kompetenzprofil



Kompetenzstufen

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

Termine, Anmeldung und Durchführungsort

Kursstart Kalenderwoche 43. Anmeldeschluss Ende Kalenderwoche 40. Das CAS dauert ein Semester und findet an einem Tag pro Woche statt. Kurszeiten sind von 08:30h bis 16:15h.

Berner Fachhochschule, Weiterbildung, Wankdorffeldstrasse 102, 3014 Bern,
Telefon +41 31 848 31 11, E-Mail office.ti-be@bfh.ch.



Berner
Fachhochschule

Inhaltsverzeichnis

1	Umfeld	4
2	Zielpublikum	4
3	Voraussetzungen	4
4	Ausbildungsziele	5
5	Kursübersicht	6
6	Kompetenzprofil	7
7	Kursbeschreibungen	8
7.1	Software-Entwicklung heute	8
7.2	Architekturen	9
7.3	Aufbau der Bibliotheken & Frameworks	10
7.4	ALM	11
7.5	Verarbeiten, Überwachen, Regeln und Steuern	11
7.6	Kommunizieren	12
7.7	Visualisieren	12
7.8	Persistieren	13
7.9	Testen	13
7.10	Installieren	14
7.11	Dokumentieren und Unterstützen	14
7.12	Semesterarbeit	15
8	Kompetenznachweise	15
9	Lehrmittel	16
10	Dozierende	16
11	Organisation	16
12	Termine	16

1 Umfeld

Im Rahmen der Digitalisierung und dem Gedankengut rund um Industrie 4.0 stehen Industriebetriebe aber auch der Dienstleistungssektor grossen Herausforderungen im Software Engineering gegenüber. Gefordert sind kurze Zyklen mit schneller Entwicklung, sofortiger Wirkung und optimaler Verbreitung von neuen Produkten.

Diese neuen Anforderungen benötigen Weitblick in der Entwicklung und viel methodisches Knowhow, sowie einen geübten Umgang mit den Werkzeugen, die diese Art der Entwicklung unterstützen. Erst die perfekte Abstimmung aller Komponenten, und dazu gehört die Softwarearchitektur, das Requirements Engineering, die Abschätzung der zeitlichen Verhältnisse für Entwicklung und Betrieb der Lösung (DevOps), sowie die Managementunterstützung führen letztendlich zum Erfolg.



Dieses CAS zeigt auf, wie man heute die Entwicklung von Anwendungen im Spannungsfeld der modernen Industrie und deren Vernetztheit in den Griff bekommen kann.

2 Zielpublikum

Das CAS «Software Engineering Industrie 4.0» richtet sich an Entwickler und Fach- und Führungskräfte in Unternehmen und IT-Bereichen, die mit Entwicklung von Software der Industrie und der Dienstleistungsbereiche betraut sind.

3 Voraussetzungen

Die Teilnehmenden bringen IT-Vorkenntnisse im Rahmen einer Informatik-, Engineering (mit Software Entwicklung im Lehrplan) oder Wirtschaftsinformatik-Ausbildung mit. Insbesondere sind Erfahrungen in der Mitarbeit und Umsetzung von Informatik-Projekten erforderlich.

Für das Verständnis von Beispielen sind Programmierkenntnisse in C#.NET von Vorteil. Die Semesterarbeit kann aus dem eigenen Umfeld oder aber vom Dozententeam vorgegeben werden. Arbeiten aus dem eigenen Bereich können mit beliebigen Technologien realisiert werden. In diesem Fall kann das Dozententeam aber die umfassende technische Betreuung nicht garantieren, wohl aber die konzeptuelle Aufsicht übernehmen.

Es werden Grundkenntnisse der folgenden Technologien vorausgesetzt:

- TCP-IP Kommunikation
- HTTP Kommunikation
- HTML Aufbau und Anwendung
- .NET Technologie Basiswissen (inkl. C#)
- Datenbanktechnik und SQL
- Grundkenntnisse UML
- Kenntnisse über grundlegende Entwurfsmuster (Singleton, Composite, MVC, MVVM, Factory, Facade, Observable ...)

4 Ausbildungsziele

Nach Absolvierung des CAS sind die Teilnehmenden in der Lage Anwendung im heutigen Spannungsfeld der Software-Entwicklung nach modernsten Grundsätzen zu entwickeln. Dazu erlangen die Teilnehmenden Kenntnisse in folgenden Gebieten:

- Einfluss der Digitalisierung auf die Softwareentwicklung
- Standards von Industrie 4.0, eCH und ISO
- Wichtige grundlegende Architekturmuster für Desktop- und verteilte Anwendungen
- Konzeption von Schnittstellen zu Umsystemen
- Wahl der korrekten Persistenztechnik
- Einsatz und Realisierung von Datenforensik
- Überwachung und Feedback der Anwendung
- Lebenszyklus der Anwendungsentwicklung
- Dokumentation von Architekturen und Software

5 Kursübersicht

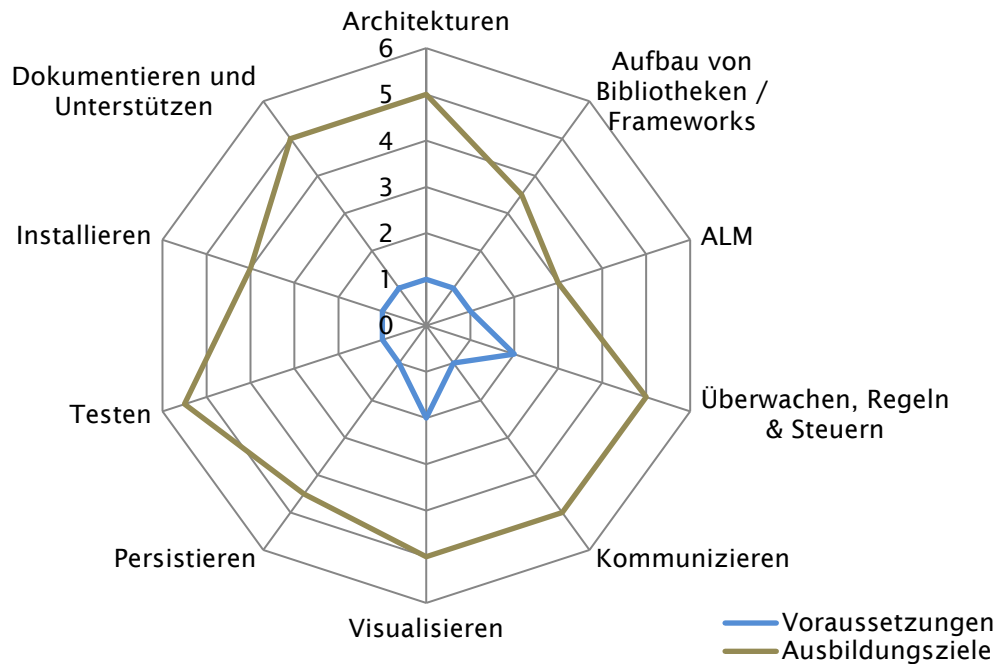
Die Kursübersicht zeigt die verschiedenen Module und Themen in der zeitlichen Abfolge und deren Umfang.

Kurs/Lehreinheit	Lektionen	Stunden
Software-Entwicklung heute	8	
Architekturen	16	
Aufbau von Bibliotheken & Frameworks	4	
ALM	8	
Verarbeiten, Überwachen, Regeln & Steuern	16	
Kommunizieren	12	
Visualisieren	24	
Persistieren	16	
Testen	8	
Installieren	4	
Dokumentieren und Unterstützen	4	
Semesterarbeit	24	90
Total	144	90

Das CAS umfasst insgesamt 12 ECTS Punkte. Für die einzelnen Kurse ist entsprechend Zeit für Selbststudium, Prüfungsvorbereitung etc. einzurechnen. Der zeitliche Aufwand des Studiums, bei einer Dauer von 18 Wochen, liegt bei ca. 14.5h pro Woche.

6 Kompetenzprofil

Die untenstehende Grafik zeigt die Steigerung der Kompetenzen, die dieses CAS anstrebt.



Legende:

1. Kenntnisse von Begriffen, Definitionen und Regeln; Faktenwissen
2. Verstehen von Zusammenhängen, Sachverhalte erklären können
3. Anwendung des Wissens in einfachen Situationen
4. Analyse der eigenen Lösung
5. Synthese neuer Lösungen und Anwendung in komplexen Situationen
6. Beurteilung der Anwendbarkeit für bestimmte Probleme und Situationen, methodische Abwägung und Evaluation von Alternativen, Beziehungen zu anderen Fachgebieten

7 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, Studienreise, Semesterarbeiten usw.

7.1 Software-Entwicklung heute

Der Kurs bildet den Einstieg in das CAS und holt die Teilnehmenden in ihrem Alltag bezüglich der Themen des Kurses ab.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können den Begriff Digitalisierung und Industrie 4.0 beschreiben– kennen die Auswirkung der «Nicht-Teilnahme» an der Digitalisierung– können die Anforderungen an die moderne Software-Entwicklung erläutern– kennen die Vorteile aber auch die Risiken der «modernen Software-Entwicklung»
Themen	<ul style="list-style-type: none">– Definition und Abgrenzung von Digitalisierung & Industrie 4.0– Eckpfeiler der heutigen Software-Entwicklung– Chancen und Risiken der heutigen Software-Entwicklung
Kursform	Vorlesung
Lehrmittel	Folien

7.2 Architekturen

Dieser Kurs widmet sich dem Grundthema Architekturen für Systeme. Neben der Klassifizierung der verschiedenen Systeme und der Herstellung eines Bezugs zu den Abbildungsformen von Architekturen, steht der Aufbau einer Architektur, die auch betriebliche und unterstützende Aspekte abbildet im Vordergrund.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können den Begriff Architektur in Bezug auf Software, Hardware und Lösungsarchitektur differenzieren und erklären– kennen die wichtigsten Frameworks zur Abbildung von Architekturen– können die Kriterien für ein gezieltes «right-sizing» einer Architektur einer Lösung erklären– können ein Architekturdokument erstellen und wissen welche Aspekte für die Kommunikation einer Architektur wichtig sind
Themen	<ul style="list-style-type: none">– Begriffe Architektur und Design– Die zeitliche Anordnung der Findung der Architektur im Projekt– Der Einfluss der Technik und des Markts auf die Architektur– Architektur-Frameworks und deren Ausdrucksform– Synchrone und asynchrone Verarbeitung und deren Auswirkung auf die Architektur– Persistieren von Daten mit Veränderungsnachweis– Sicherheit und Architektur– Standards für Industrie 4.0, eCH, ISO– Mehrsprachige Anwendungen konzipieren
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.3 Aufbau der Bibliotheken & Frameworks

Dieser Kurs beleuchtet isoliert das Architektur-Thema «Aufbau von Bibliotheken und Frameworks». Es zeigt die geltenden Besonderheiten auf und begründet wieso Bibliotheken eingesetzt werden sollten.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– lernen die Prinzipien des Stream- und Event-Processing kennen– können die Komponenten einer Event-Driven Architecture (EDA) beschreiben– lernen die Unterschiedlichen Sprachen für die Erkennung und Verarbeitung von Events kennen– können Probleme mit Hilfe von Event-Processing lösen– können abschätzen, wann sich der Einsatz von Event-Processing lohnt– kennen die Positionierung von Event Processing interhalb einer Big Data Architektur
Themen	<ul style="list-style-type: none">– Was ist ein Event, was ist eine Message?– Was ist Complex Event Processing (CEP)?– Historie und Prinzipien von Stream- und Complex Event Processing– Event Processing Design Patterns– Erkennen von Events– Aggregation von Events – wie können Business Events von den Raw Events abgeleitet werden– Internet of Things und Machine to Machine (M2M) – was hat dies mit Event-Processing zu tun?– Welche Sprachen für das Event-Processing gibt es?– Plattformen und Frameworks für Stream Processing: Apache Storm, Apache Flink, Kafka Streams, Spark Streaming usw.
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.4 ALM

In der ganzheitlichen Betrachtung der Software-Entwicklung ist ein Blick auf ALM (Application Life-Cycle Management) unabdingbar. Dieser Kurs zeigt in einer Zusammenfassung auf was es heute ankommt.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können ALM erklären– können das Delta der Entwicklung ihres eigenen Umfelds zur ALM Theorie beschreiben (wo bin ich stark und wo schwach Analyse)
Themen	<ul style="list-style-type: none">– ALM Grundlagen– Aspekte des ALM– Entwicklungsprozesse im Vergleich– Verwendung eines ALM Werkzeuges– Wichtigkeit von Qualitätssicherung– Funktion und Idee von DevOps
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.5 Verarbeiten, Überwachen, Regeln und Steuern

Anwendungen beinhalten in der Regel einen verarbeitenden Teil. Dieser Kurs beschäftigt sich mit Entwurfsmustern und Implementierungsdetails für diese Softwareebene.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können die wichtigsten Entwurfsmuster für Verarbeitungen nennen– können die Kernteile in einer Architektur so einbringen, dass diese isoliert sind– kennen die Vorteile aber auch die Gefahren von asynchronen Verarbeitungen
Themen	<ul style="list-style-type: none">– Geschäftslogik entwerfen– Synchrone Verarbeitungen– Direkte und indirekte asynchrone Verarbeitung– Isolation der Geschäftslogik von UI und Persistenz– Sicherheitsaspekte in der Geschäftslogik
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.6 Kommunizieren

Anwendungen müssen Daten zwischen der Benutzeroberfläche und dem Speicher übertragen. Dabei kommen je nach Architektur und Umfeld unterschiedliche Techniken zum Einsatz. Dieser Kurs zeigt auf, was heute angesagt ist.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können die heute verwendeten Standards für Kommunikation nennen– kennen den Unterschied der Protokolle– kennen die unterschiedlichen Formen wie synchrone, asynchrone Kommunikation
Themen	<ul style="list-style-type: none">– TCP-IP basierte Kommunikationen– Verwendete Systeme in der Industrie– Synchrone und asynchrone Kommunikation– WCF, die Kommunikationsmaschine– Web-Services– REST-Services– Sicherheit in der Kommunikation
	Vorlesung
Lehrmittel	Folien

7.7 Visualisieren

Anwendungen brauchen eine Mensch-Maschinen-Schnittstelle. Dieser Kurs zeigt die unterschiedlichen Konzepte von Desktop und Web-Anwendungen auf und beleuchtet Aspekte wie Technik, gestalterische Möglichkeiten und Aufwand.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können die Konsequenzen der unterschiedlichen Technologien erklären– wissen wie man eine Benutzeroberfläche lose an eine Geschäftslogik koppelt– Kennen die wichtigsten Entwurfsmuster für UI-Technik und können diese erklären
Themen	<ul style="list-style-type: none">– Anwendungen mit WPF– Anwendungen mit Web-Technologien (ASP.NET)– Anwendungen für Smart-Geräte– Sicherheitsaspekte im UI– Reaktive UI– Design ist nicht gleich Design
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.8 Persistieren

Anwendungen benötigen Daten und speichern diese in der Regel auch ab. Dieser Kurs zeigt auf wie das richtige Datendesign erreicht wird.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– können die Entscheidungskriterien für die Wahl der richtigen Speichertechnologie erklären– können eine Persistenzebene lose gekoppelt entwerfen– kennen das Factory-Entwurfsmuster– können die Performance aufzeigen
Themen	<ul style="list-style-type: none">– Speichertechnologien File, Datenbank, Cloud– .NET Entity Framework– Persistenz und Sicherheit– Datenverschlüsselung
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.9 Testen

Da Software heute iterativ ausgerollt wird, ist das Automatisieren des Testens von Komponenten und Anwendungen noch wichtiger denn je geworden. Dieser Kurs zeigt auf wieso und wie man das in den Griff bekommt.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– kennen die Testpyramide– kennen die Möglichkeiten der Testautomation
Themen	<ul style="list-style-type: none">– Testpyramide– Unit-Tests– Integrations-Tests– Akzeptanz-Tests– Test von Benutzerschnittstellen– Test Automatisierung
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.10 Installieren

Anwendungen herstellen ist eines. Die Anwendung auf dem Kundensystem zum Einsatz zu bringen erfordert eine Installation. Dahinter verstecken sich oft unerkannte Anforderungen. Dieser Kurs geht auf diese Problematik ein und zeigt die Strategien für die Installation von Software auf Kundensystemen auf.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– kennen die Strategie für eine Kundeninstallation einer Anwendung– kennen die technischen Möglichkeiten für die Installation von Anwendungen unter Berücksichtigung heterogener Kundenumgebungen– können Abläufe für die korrekte Installation rechtzeitig in der Architektur einbringen
Themen	<ul style="list-style-type: none">– Grundlagen für Installation von Anwendungen– Strategien für Anwendungsinstallation– Rückkopplung mit der Architektur– Was ist zuerst, die Installation oder die Anwendung
Kursform	<ul style="list-style-type: none">– Vorlesung– Übungen
Lehrmittel	Folien

7.11 Dokumentieren und Unterstützen

Sind ihre Produkte heute dokumentiert? Dieser Kurs zeigt auf wie die richtige Dokumentation in welcher Form erstellt wird.

Kursbeschreibung	
Lernziele	Die Teilnehmenden: <ul style="list-style-type: none">– kennen die unterschiedlichen Dokumentationsstrategien– erkennen die Wichtigkeit der Dokumentation und wo man gezielt sparen kann
Themen	<ul style="list-style-type: none">– Was ist Dokumentation– Dokumentation richtig beginnen– Code-Dokumentation– Papier-Dokumentation– Vorschriften und Standards
Kursform	Vorlesung
Lehrmittel	Folien

7.12 Semesterarbeit

Die Semesterarbeit deckt neben ein paar Übungen, die eingebettet im Unterricht stattfinden, den Hauptbestandteil für die praktische Erfahrung während dem CAS.

Kursbeschreibung	
Lernziele und Themen	Die Teilnehmenden: <ul style="list-style-type: none"> – wenden gezielt Stoffgebiete und Kombinationen aus dem Kurs in einer Anwendung an – können das Thema respektive den Inhalt der Arbeit individuell festlegen
Kursform	Gruppenarbeiten zu zweit werden angestrebt, aber Einzelarbeiten sind möglich
Lehrmittel	Arbeit aus dem eigenen Arbeitsumfeld, oder ein Beispiel das die Schule abgibt.

8 Kompetenznachweise

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

Kompetenznachweis	Gewicht	Art der Qualifikation	Erfolgsquote Studierende
Nachweis 1	2	Schriftliche Prüfung	0 - 100 %
Nachweis 2	2	Schriftliche Prüfung	0 - 100 %
Semesterarbeit	6	Projektarbeit aus Praxis oder Theorie	0 - 100 %
Gesamtgewicht / Gesamterfolgsquote	10		0 - 100 %
ECTS-Note			A - F

Alle Studierenden können in einem Qualifikationsthema eine Erfolgsquote von 0 bis 100% erarbeiten. Die gewichtete Summe aus den Erfolgsquoten pro Thema und dem Gewicht des Themas ergibt eine Gesamterfolgsquote zwischen 0 und 100%. Die Gesamterfolgsquote wird in eine ECTS-Note A bis E umgerechnet, gemäss Studienreglement. Weniger als 50% Gesamterfolgsquote ergibt eine ungenügende Note F.

9 Lehrmittel

Muss noch ausgearbeitet und bestimmt werden.

Nr	Titel	Autoren	Verlag	Jahr	ISBN Nr.
[1]	bsp	Viktor Mayer-Schönberger	Redline Verlag	2013	ISBN-10: 978-3-86414-459-2
[2]					
[3]					

10 Dozierende

Vorname Name	Firma	E-Mail
Rolf Wenger	weroSoft AG	rolf.wenger@weroSoft.net
Patrick Arpagaus	weroSoft AG	patrick.arpagaus@weroSoft.net

+ Weitere Experten, Betreuer und Gastreferenten aus Unternehmen und Lehre

11 Organisation

CAS-Leitung:

Rolf Wenger

Tel: +41 79 820 61 29

E-Mail: rolf.wenger@weroSoft.net

CAS-Administration:

Andrea Moser

Tel: +41 31 84 83 211

E-Mail: andrea.moser@bfh.ch

12 Termine

Daten:

KW 43 2018 bis KW 14 2019

Montag, von 08:30 Uhr bis 16:15 Uhr

Dokumenteninformation

Study Guide CAS I4SE
2018-03-05

Dieser Study Guide gilt für die Publikation ab Herbstsemester 2018.

Während der Durchführung des CAS können sich Anpassungen bezüglich Inhalten, 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
Wankdorffeldstrasse 102
CH-3014 Bern

Telefon +41 31 848 31 11

Email: office.ti-be@bfh.ch

→ ti.bfh.ch/weiterbildung

→ ti.bfh.ch/cas-i4se