

Modulhandbuch

# Bachelor »Code & Context«

TH Köln

Modulhandbuch 2019-2024

Version: 09e488a / Mon May 4 13:55:44 2026

# Inhaltsverzeichnis

<b>I Module</b>	<b>1</b>
Übersicht über die Module	2
Modul »Coding Software 1« (CS1)	3
Modul »Designing Futures 1« (DF1)	4
Modul »Project Explore 1« (PE1)	5
Modul »Community & Reflection 1« (CR1)	6
Modul »Coding Software 2« (CS2)	7
Modul »Creating Impact 1« (CI1)	8
Modul »Project Explore 2« (PE2)	9
Modul »Community & Reflection 2« (CR2)	10
Modul »Developing Things 1« (DT1)	11
Modul »Creating Impact 2« (CI2)	12
Modul »Project Make 1« (PM1)	13
Modul »Community & Reflection 3« (CR3)	14
Modul »Developing Things 2« (DT2)	15
Modul »Designing Futures 2« (DF2)	16
Modul »Project Make 2« (PM2)	17
Modul »Community & Reflection 4« (CR4)	18
Modul »Designing Futures 3« (DF3)	19
Modul »Creating Impact 3« (CI3)	20
Modul »Project Launch« (PL1)	21
Modul »Community & Reflection 5« (CR5)	22
Modul »Optionales Praxissemester« (OPT)	23
Modul »Praxisprojekt« (PP)	25
Modul »Bachelorarbeit« (BA)	26
Modul »Bachelor Kolloquium« (BK)	27
Modul »Community & Reflection 6« (CR6)	28
<b>II Kurse</b>	<b>29</b>
Übersicht über Kurse und Katalog	30
Kurs »Computational Thinking«	31
Kurs »Coding Essentials 1«	32
Kurs »Coding Essentials 2«	34

Kurs »Client Server Basics«	36
Kurs »Design Thinking«	38
Kurs »Generative Gestaltung«	40
Kurs »Product Dimensions«	41
Kurs »Advanced Clients«	43
Kurs »Agile Team Coding«	45
Kurs »Application Design«	47
Kurs »Clean Code«	49
Kurs »Entrepreneurial Thinking«	51
Kurs »Rapid Prototyping«	52
Kurs »Social Hubs«	53
Kurs »Applied AI«	54
Kurs »Immersive Applications«	55
Kurs »Microservice Architectures«	56
Kurs »Trend and Market Research«	58
Kurs »Open Ecosystems«	60
Kurs »Business Economics«	62
Kurs »Internet of Things«	63
Kurs »Connected Products«	65
Kurs »DevOps«	66
Kurs »Design as a Language«	68
Kurs »Interaction Design«	69
Kurs »Users and Situated Action«	70
Kurs »Global Citizenship«	71
Kurs »Open Design«	72
Kurs »Business Models and Plans«	73
Kurs »Entrepreneurial Marketing«	74
<b>III Kompetenzorientierung</b>	<b>76</b>
Handlungsfelder	77
Handlungsfeld »Coding Software« (CS) . . . . .	77
Handlungsfeld »Developing Things« (DT) . . . . .	77
Handlungsfeld »Designing Futures« (DF) . . . . .	78
Handlungsfeld »Creating Impact« (CI) . . . . .	78
Kompetenzcluster	80
Kompetenzcluster »Software Development and Architecture« (SDA) . . . . .	80
Kompetenzcluster »Dual Reality and Smart Spaces« (DRSS) . . . . .	80
Kompetenzcluster »Interaction, Empathy and Emotion« (IEE) . . . . .	80
Kompetenzcluster »Product Dimensions« (PROD) . . . . .	81
Kompetenzcluster »Agile Methoden« (AGILE) . . . . .	81
Kompetenzcluster »Communication« (COMM) . . . . .	82
Kompetenzcluster »Technical Society, Culture and Ethics« (TSCE) . . . . .	82

---

Kompetenzcluster »Creative Coding« (CREA) . . . . .	83
Kompetenzcluster »Entrepreneurship and Social Innovation« (ENSI) . . . . .	83
Kompetenzcluster »Soziale Kompetenz und Selbstkompetenz« (SOSE) . . . . .	84

Module

## Übersicht über die Module

Modul	Coding Software	Developing Things	Designing Futures	Creating Impact	SDA	DRSS	IEE	PROD	AGILE	COMM	TSCE	CREA	ENSI	SOSE	Sem.	CP
Coding Software 1	x				x										1	12
Designing Futures 1			x				x	x	x			x			1	9
Project Explore 1					x		x	x		x		x		x	1	4
Community & Reflection 1														x	1	5
Coding Software 2	x				x	x			x						2	12
Creating Impact 1				x		x	x		x		x	x	x		2	9
Project Explore 2					x	x	x		x	x	x		x	x	2	4
Community & Reflection 2														x	2	5
Developing Things 1		x			x	x		x							3	9
Creating Impact 2				x				x			x		x		3	9
Project Make 1					x	x		x		x	x		x	x	3	7
Community & Reflection 3														x	3	5
Developing Things 2		x			x	x		x	x						4	9
Designing Futures 2			x				x	x				x			4	9
Project Make 2					x	x	x	x		x		x		x	4	7
Community & Reflection 4														x	4	5
Designing Futures 3			x				x				x	x			5	6
Creating Impact 3				x						x			x		5	6
Project Launch							x	x		x	x	x	x	x	5	13
Community & Reflection 5														x	5	5
Optionales Praxissemester					x	x	x	x	x	x	x	x	x	x	0/6	30
Praxisprojekt					x	x	x	x	x	x	x	x	x	x	6/7	10
Bachelorarbeit					x	x	x	x	x	x	x	x	x	x	6/7	12
Bachelor Kolloquium					x	x	x	x	x	x	x	x	x	x	6/7	3
Community & Reflection 6														x	6/7	5

Tabelle 0.1: Vernetzung der Module mit den Handlungsfeldern und Kompetenzclustern.

---

# Modul »Coding Software 1« (CS1)

**Verantwortlich:** Prof. Dr. Frank Schimmel

**Studiensemester:** 1

**Häufigkeit:** jährlich

**Kreditpunkte:** 12 ECTS

**Arbeitsaufwand:** 160h Kontaktzeit / 200h Selbstlernzeit

**Kompetenzcluster:** SDA

**Modulprüfung:** Teilmodulprüfungen, siehe Kurse (benotet)

## Learning Outcome / Qualifikationsziele des Moduls

Studierende können in Einzelarbeit auf Basis exakter jedoch veränderlicher Anforderungen eine mittelgroße, web-basierte Client-Server-Applikation kodieren, indem sie Programmiersprachenkonzepte im Kleinen wie Großen, sowie gängige Libraries, Frameworks und Standard-Server anwenden, um später in Projekten einfache Entwicklungsaufgaben übernehmen zu können.

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
CS11	Computational Thinking	40h / 50h	SDA	3	benotet
CS12	Coding Essentials 1	40h / 50h	SDA	3	benotet
CS13	Coding Essentials 2	40h / 50h	SDA	3	benotet
CS14	Client Server Basics	40h / 50h	SDA	3	benotet

---

# Modul »Designing Futures 1« (DF1)

**Verantwortlich:** Prof. Dr. Laura Popplow

**Studiensemester:** 1

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Kompetenzcluster:** AGILE / CREA / IEE / PROD

**Modulprüfung:** Teilmodulprüfungen, siehe Kurse (benotet)

## Learning Outcome / Qualifikationsziele des Moduls

Absolvent\*innen dieses Moduls beherrschen grundlegende Methoden des integrierten Designs im digitalen Kontext. Sie konzeptionieren radikal innovative Anwendungen, indem sie

- Szenarien für zukünftige Produkte und Dienstleistungen unter Berücksichtigung gesellschaftlicher, kultureller und ästhetischer Kontexte entwickeln und bewerten,
- in Prototyping und Modellbau Konzepten und Ideen materielle Form geben und zugleich neue Konzepte und Ideen im Dialog mit Werkzeug und Material entwickeln,
- sowie Programmierung als künstlerisch-gestalterisches Feld begreifen und Methoden des Creative Coding und der generativen Gestaltung im Spannungsfeld von Design und Kunst einsetzen,

um später in interdisziplinären Projekten gestalterische Experimente und spekulative Exploration durchzuführen, die der Ideen- und Formfindung dienen.

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
DF11	Design Thinking	40h / 50h	AGILE	3	benotet
DF12	Generative Gestaltung	40h / 50h	CREA	3	benotet
DF13	Product Dimensions	40h / 50h	IEE / PROD	3	benotet

---

## Modul »Project Explore 1« (PE1)

**Verantwortlich:** Prof. Dr. Jonas Schild, Prof. Dr. Frank Schimmel, Prof. Dr. Lasse Scherffig, Prof. Dr. René Wörzberger

**Studiensemester:** 1

**Häufigkeit:** jährlich

**Kreditpunkte:** 4 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 100h Selbstlernzeit

**Kompetenzcluster:** SDA / IEE / PROD / COMM / CREA / SOSE

**Modulprüfung:** Präsentation, Ende Kursblock (benotet)

### Learning Outcome / Qualifikationsziele des Moduls

Studierende können auf der Basis von Client-Server-Applikation, gestalterischer Experimente und spekulativer Exploration ein problem-basiertes Szenario analysieren und mögliche Lösungen explorieren, indem sie

- sich in Teams zusammenfinden,
- projekt-basierte Organisationsformen anwenden,
- die Kompetenzen der begleitenden Kurse anwenden und in einen Zusammenhang bringen
- eine umfangreiche Aufgabenstellung (die auf die begleitenden Kurse zugeschnitten ist) selbstständig bearbeiten, in geeignete Teilaufgaben zerlegen und diese im Team verteilen,

damit sie

- die in den Kursen erworbenen Kenntnisse durch praktische Anwendung und eigene Erfahrungen weiter vertiefen und verinnerlichen,
- erste Erfahrungen in der Teamdynamik bei größeren Aufgabenstellungen sammeln,
- und dies in spätere Module einbringen können.

### Prüfungsformen

- Präsentation von Endergebnissen

---

## Modul »Community & Reflection 1« (CR1)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 1

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** Portfolio, individuelle Fristen (unbenotet)

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

---

# Modul »Coding Software 2« (CS2)

**Verantwortlich:** Prof. Dr. Stefan Bente

**Studiensemester:** 2

**Häufigkeit:** jährlich

**Kreditpunkte:** 12 ECTS

**Arbeitsaufwand:** 160h Kontaktzeit / 200h Selbstlernzeit

**Empfohlene Voraussetzungen:** Coding Software 1

**Kompetenzcluster:** DRSS / AGILE / SDA

**Modulprüfung:** benotet

## Learning Outcome / Qualifikationsziele des Moduls

Studierende können im Team realitätsnahe digitale Produkte gestalten, deren Anforderungen sich dynamisch verändern und aus realen Quellen stammen, indem sie

- Methoden, Regeln und Werkzeuge agiler Entwicklung anwenden,
- die Architektur mittlerer bis größerer Anwendung sinnvoll strukturieren,
- Prinzipien für nachhaltige Code-Qualität anwenden, sowie
- die Anwendungen in geeigneter Weise hosten,

so dass sie ein Projekt in Ende-zu-Ende-Verantwortung im Team umsetzen können.

## Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- mündliche Prüfungen

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
CS21	Agile Team Coding	40h / 50h	AGILE / SDA	3	benotet
CS22	Clean Code	40h / 50h	SDA	3	benotet
CS23	Application Design	40h / 50h	SDA	3	benotet
CS24	Advanced Clients	40h / 50h	DRSS	3	benotet

---

# Modul »Creating Impact 1« (CI1)

**Verantwortlich:** Prof. Dr. Jonas Schild

**Studiensemester:** 2

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Kompetenzcluster:** ENSI / AGILE / DRSS / CREA / IEE / TSCE

**Modulprüfung:** benotet

## Learning Outcome / Qualifikationsziele des Moduls

Die Absolvent\*innen des Moduls können soziale Kontexte und ökonomische Anforderungen für digitalen Produkten und Applikationen untersuchen, indem sie

- die Grundlagen des Entrepreneurship verstehen und diese anwenden,
- digitale soziale Kontexte von Produkten und Anwendungen nutzen und gestalten,
- und auf Basis von Material und Code Prototyping als einen »maker-based« Entwurfsprozess anwenden,

um erste grundlegende Kompetenzen in der Realisierung nachhaltiger Wirkungen digitaler und sozialer Produkte zu erlangen.

## Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- Demonstration von Prototypen
- mündliche Prüfungen

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
CI11	Entrepreneurial Thinking	40h / 50h	ENSI / AGILE	3	benotet
CI12	Rapid Prototyping	40h / 50h	DRSS / CREA	3	benotet
CI13	Social Hubs	40h / 50h	IEE / TSCE	3	benotet

---

## Modul »Project Explore 2« (PE2)

**Verantwortlich:** Prof. Dr. Christian Faubel, Prof. Dr. Laura Popplow

**Studiensemester:** 2

**Häufigkeit:** jährlich

**Kreditpunkte:** 4 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 100h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / AGILE / ENSI / COMM / TSCE / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Studierende können problem-basierte Szenarien in sozialen und ökonomischen Kontexten analysieren und explorieren. Durch eine agile und Realisierung realitätsnaher digitaler "Minimum Viable Products (MVP)" können die Studierenden Lösungen für dynamische Anforderungen finden, indem sie

- sich in selbstorganisierten Teams zusammenfinden,
- dabei agile Methoden für Teamorganisation, Qualitätssicherung und Aufgabenplanung anwenden,
- die Kompetenzen der begleitenden Kurse (insb. Entwicklung einer sinnvollen Produktidee sowie Coding und Architektur für bei der Implementierung) anwenden und in einen Zusammenhang bringen,
- eine Aufgabenstellung (die auf die begleitenden Kurse zugeschnitten ist) selbstständig bearbeiten, in geeignete Teilaufgaben zerlegen und diese im Team verteilen,

damit sie

- die in den Kursen erworbenen Kenntnisse durch praktische Anwendung und eigene Erfahrungen weiter vertiefen und verinnerlichen,
- Erfahrungen in der Teamdynamik bei der Lösung größerer Aufgabenstellungen mit agilen Methoden sammeln,
- und dies in spätere Module einbringen können.

### Prüfungsformen

- Präsentation von Zwischen- und Endergebnissen
- Bewertung der Qualität von Produktidee, Code, Architektur sowie ggfs. physischer Umsetzung anhand von Dokumentationen, Entwürfen oder prototypischen Entwicklungen zu vereinbarten Meilensteinen

---

## Modul »Community & Reflection 2« (CR2)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 2

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** unbenotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

### Prüfungsformen

Präsentation der Ergebnisse, schriftliche Dokumentation

---

# Modul »Developing Things 1« (DT1)

**Verantwortlich:** Prof. Dr. Christian Faubel

**Studiensemester:** 3

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Empfohlene Voraussetzungen:** Coding Software 1 / Coding Software 2

**Kompetenzcluster:** SDA / DRSS / PROD

**Modulprüfung:** benotet

## Learning Outcome / Qualifikationsziele des Moduls

Die Studierenden können in dynamischen Entwicklungsumfelder und komplexen Anwendungsdomänen eine anwenderfreundliche, sichere, skalier- und erweiterbare Client-Server-Applikation entwickeln, indem sie

- moderne Ansätze des Entwurfs sowie Frameworks zur Realisierung anwenderfreundlicher und sicherer Systeme anwenden, sowie
- Grundsätze von Datenschutz, Sicherheit und Privacy beherzigen,

um später in gestaltender Rolle in dynamischen Entwicklungsprojekten mitwirken zu können.

## Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- mündliche Prüfungen

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
DT11	Microservice Architectures	40h / 50h	SDA / DRSS	3	benotet
DT12	Applied AI	40h / 50h	SDA / DRSS	3	benotet
DT13	Immersive Applications	40h / 50h	SDA / DRSS / PROD	3	benotet

---

## Modul »Creating Impact 2« (CI2)

**Verantwortlich:** Prof. Dr. Laura Popplow

**Studiensemester:** 3

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Kompetenzcluster:** PROD / ENSI / TSCE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Die Studierenden können Konsequenzen des eigenen Handelns sowie die Wirkungen hergestellter Artefakte in ökonomischen sowie kulturellen Kontexten analysieren und Wechselwirkungen herstellen, indem sie

- Ideen für innovative Produkte, Dienste und Geschäftsmodelle auf Trend- und Marktforschung basieren,
- die wichtigsten Entscheidungsbereiche wirtschaftlichen Handelns auf Unternehmensebene und in gesellschaftlichen Organisationen kennen,
- Wechselwirkungen ihrer Vorhaben mit offenen Ökosystemen erkennen, nutzen und herstellen,

um wirtschaftliche Potenziale für Produkte zu erkennen, diese mit offenen Prozessen in kulturelle Zusammenhänge zu setzen und Geschäfts- oder Betreibermodelle betriebswirtschaftlich zu fundieren.

### Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- mündliche Prüfungen

### Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
CI21	Trend and Market Research	40h / 50h	PROD / ENSI / TSCE	3	benotet
CI23	Open Ecosystems	40h / 50h	PROD / ENSI / TSCE	3	benotet
CI23	Business Economics	40h / 50h	ENSI	3	benotet

---

## Modul »Project Make 1« (PM1)

**Verantwortlich:** Prof. Dr. Frank Schimmel, Prof. Dr. Christian Faubel, Prof. Dr. Laura Popplow, Prof. Dr. Jonas Schild

**Studiensemester:** 3

**Häufigkeit:** jährlich

**Kreditpunkte:** 7 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 190h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / PROD / ENSI / COMM / TSCE / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Studierende können ein digitales Produkt mit einem komplexen Software-Anteil entwerfen und im Team umsetzen. Dabei erschließen sie sich die ökonomischen, sozialen, rechtlichen und kulturellen Aspekte einer gegebenen realistischen Anwendungsdomäne und setzen die Produkthanforderungen in einem dynamischen Umfeld um, indem sie

- in Projektteams mit agilen Organisationsform arbeiten,
- der Komplexität und Dynamik von Domäne und technischen Lösungen mit erlernten Methoden und Konzepten begegnen,
- dabei die Kompetenzen der begleitenden Kurse anwenden und in einen Zusammenhang bringen,
- ihre entwickelten Artefakte (Code, Dokumentation) so gestalten, dass eine nachhaltige Weiterentwicklung als Open-Source-Projekt (oder einer anderen Form eines digitalen Ökosystems) möglich ist, sowie
- ihr Produkt in angemessener Weise den Stakeholdern präsentieren,

damit sie

- die in den Kursen erworbenen Kenntnisse durch praktische Anwendung und eigene Erfahrungen im Rahmen eines realistischen Projekts weiter vertiefen und verinnerlichen,
- Erfahrungen mit dynamischen und komplexen Anforderungen aus ökonomischen und kulturellen Zusammenhängen machen,
- und dies in spätere Module sowie den Beruf einbringen können.

### Prüfungsformen

- Präsentation von Zwischen- und Endergebnissen
- Bewertung der Qualität von Produktidee, Code, Architektur sowie ggfs. physischer Umsetzung anhand von Dokumentationen, Entwürfen oder prototypischen Entwicklungen zu vereinbarten Meilensteinen

---

## Modul »Community & Reflection 3« (CR3)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 3

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** unbenotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

### Prüfungsformen

Präsentation der Ergebnisse, schriftliche Dokumentation

---

## Modul »Developing Things 2« (DT2)

**Verantwortlich:** Prof. Dr. Matthias Böhmer

**Studiensemester:** 4

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Empfohlene Voraussetzungen:** Developing Things 1

**Kompetenzcluster:** DRSS / SDA / PROD / AGILE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer und Teilnehmerinnen dieses Moduls können verteilte und smarte Anwendungen für hardwarenahe Systeme entwickeln, indem sie

- Mikrocontroller und Einplatinencomputer programmieren, um die dingliche Welt durch Sensoren zu erfassen und durch Aktoren zu beeinflussen,
- die Daten der Sensoren und weitere Informationen mit Verfahren der künstlichen Intelligenz verarbeiten und automatisierte Aktionen auslösen,
- und verteilte Informationsarchitekturen entwerfen und diese über verschiedene Devices synchronisieren,

um später hardwarenahe, smarte und verteilte Anwendungen für dinglich-digitale Produkte in Software und Hardware zu entwickeln.

### Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- mündliche Prüfungen

### Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
DT21	Internet of Things	40h / 50h	DRSS / SDA	3	benotet
DT22	DevOps	40h / 50h	AGILE	3	benotet
DT23	Connected Products	40h / 50h	DRSS / PROD	3	benotet

---

## Modul »Designing Futures 2« (DF2)

**Verantwortlich:** Prof. Dr. Lasse Scherffig

**Studiensemester:** 4

**Häufigkeit:** jährlich

**Kreditpunkte:** 9 ECTS

**Arbeitsaufwand:** 120h Kontaktzeit / 150h Selbstlernzeit

**Kompetenzcluster:** IEE / PROD / CREA

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Absolvent\*innen dieses Moduls können Interaktionen und digitale Handlungsfelder gestalten und reflektieren, indem sie

- Ideen und Szenarien spekulativ entwickeln, darstellen und kritisch hinterfragen,
- interaktive Systeme unter Berücksichtigung der Konventionen und Geschichte des Interaktionsdesigns und unter Einsatz experimenteller Designmethoden gestalten,
- und dabei Methoden anwenden, die ihnen erlauben, die Wahrnehmung, Gewohnheiten und kontextuelle Situierung ihrer Nutzer\*innen zu berücksichtigen und neu auszurichten,

um später interaktive Produkte und ihre Wirkung ganzheitlich gestalten zu können.

### Prüfungsform

- TBD

### Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
DF21	Design as a Language	40h / 50h	IEE / PROD	3	benotet
DF22	Interaction Design	40h / 50h	IEE / CREA	3	benotet
DF23	Users and Situated Action	40h / 50h	IEE / CREA	3	benotet

---

## Modul »Project Make 2« (PM2)

**Verantwortlich:** Prof. Dr. Frank Schimmel, Prof. Dr. Christian Faubel, Prof. Dr. Laura Popplow, Prof. Dr. Jonas Schild

**Studiensemester:** 4

**Häufigkeit:** jährlich

**Kreditpunkte:** 7 ECTS

**Arbeitsaufwand:** 0h Kontaktzeit / 210h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / PROD / COMM / CREA / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Studierende können hardware-basierte, smarte und verteilte interaktive Produkte und ihre materiell-digitale Wirkung ganzheitlich gestalten, indem sie

- hardwarenahe, verteilte und smarte Anwendungen realisieren,
- Interaktionen gestalten und reflektieren,
- in Projektteams mit agilen Organisationsform arbeiten,
- der Komplexität und Dynamik mit Methoden und Konzepten begegnen,
- und die Kompetenzen der begleitenden Kurse anwenden und in einen Zusammenhang bringen,

damit sie

- die in den Kursen erworbenen Kenntnisse durch praktische Anwendung und eigene Erfahrungen weiter vertiefen und verinnerlichen,
- und dies in spätere Module sowie dem Beruf einbringen können.

### Prüfungsformen

- Präsentation von Zwischen- und Endergebnissen
- Bewertung von Dokumentationen, Entwürfen oder prototypischen Entwicklungen und Ergebnissen zu vereinbarten Meilensteinen

---

## Modul »Community & Reflection 4« (CR4)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 4

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** unbenotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

### Prüfungsformen

Präsentation der Ergebnisse, schriftliche Dokumentation

---

## Modul »Designing Futures 3« (DF3)

**Verantwortlich:** Prof. Nina Juric

**Studiensemester:** 5

**Häufigkeit:** jährlich

**Kreditpunkte:** 6 ECTS

**Arbeitsaufwand:** 80h Kontaktzeit / 100h Selbstlernzeit

**Kompetenzcluster:** IEE / TSCE / CREA

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Studierende kennen und verstehen die Entwicklung des Open Design und Open Web als Triebfeder digitaler Innovation und sind in der Lage, mit eigenen Beiträgen dessen Weiterentwicklung zu fördern. Sie basieren dies auf ihrem Bewusstsein einer Handlungsethik, die die Werte freiheitlicher, sozialer und geschlechtergerechter Weiterentwicklung von Gemeinwesen ins Zentrum setzt.

Hierzu führen sie

- Projekte mit Nutzung und zur Erweiterung der Open xx Plattformen
- Studien zu den Implikationen von Coded Artefacts auf die Gesellschaft und die Verantwortung des Entwicklers

durch, um

- zum Gemeineigentum im globalen Netz beizutragen und
- als Entwickler globaler Applikationen ethisch verantwortbar handeln zu können.

### Prüfungsform

- TBD

### Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
DF31	Open Design	40h / 50h	IEE / TSCE / CREA	3	benotet
DF32	Global Citizenship	40h / 50h	IEE / TSCE	3	benotet

---

# Modul »Creating Impact 3« (CI3)

**Verantwortlich:** Prof. Dr. Monika Engelen

**Studiensemester:** 5

**Häufigkeit:** jährlich

**Kreditpunkte:** 6 ECTS

**Arbeitsaufwand:** 80h Kontaktzeit / 100h Selbstlernzeit

**Kompetenzcluster:** ENSI / COMM

**Modulprüfung:** benotet

## Learning Outcome / Qualifikationsziele des Moduls

Die Studierende können die Realisierung einer Lösung organisatorisch, betriebswirtschaftlich und rechtlich holistisch beschreiben, analysieren und selbst entwerfen. Indem sie die organisatorischen, betriebswirtschaftlichen und rechtlichen Aspekte insb.

- die organisatorischen und rechtlichen Grundlagen zur Gründung eines Unternehmens o.ä.,
- Geschäfts- und Betreibermodelle, also betriebswirtschaftliche Konzepte zum Aufbau und der Führung von Unternehmen, Communities und anderer Strukturen für ein Produkt oder eine Dienstleistung,
- die kunden- und marktorientierte Ausrichtung entlang der 4 Ps des Marketingmixes (Product, Price, Promotion, Place) für eine innovative Idee und spezifische Zielgruppe
- und die Optionen für den erfolgreichen Eintritt in einen neuen Markt unter Einbeziehung der zukünftigen Nutzer verstehen, bewerten und anwenden können,
- sowie ihre Ideen zielkundengerecht (Investoren oder Kunden) überzeugend zu präsentieren (Pitch),

um Lösungen als kommerzielle Produkte, Dienstleistungen, freie Software oder kollaborative Anstrengungen durch die Gründung eines Startups und alternativer Strukturen zu realisieren und eine Geschäftsidee erfolgreich am Markt zu platzieren und zu etablieren.

## Prüfungsform

- schriftliche Klausurarbeiten
- semesterbegleitende Projektaufgaben und Dokumentation
- Präsentation (insb. Pitch)

## Teilmodule

Nr.	Kurs aus Katalog	Kontakt-/Selbstl.	Kompetenzcluster	ECTS	Prüfung
CI31	Business Models and Plans	40h / 50h	ENSI	3	benotet
CI32	Entrepreneurial Marketing	40h / 50h	ENSI / COMM	3	benotet

---

## Modul »Project Launch« (PL1)

**Verantwortlich:** Prof. Dr. Frank Schimmel, Prof. Dr. Christian Faubel, Prof. Dr. Laura Popplow, Prof. Dr. Jonas Schild

**Studiensemester:** 5

**Häufigkeit:** jährlich

**Kreditpunkte:** 13 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 370h Selbstlernzeit

**Kompetenzcluster:** IEE / PROD / ENSI / COMM / TSCE / CREA / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Studierende können Ideen für digitale Produkte als Minimum Viable Product an den Markt bringen, indem sie

- eine innovative Projektidee von ersten Prototypen zum Minimum Viable Product entwickeln,
- Betreiber- oder Geschäftsmodelle für kommerzielle Produkte, Dienstleistungen, freie Software oder kollaborative Anstrengungen beschreiben,
- Nutzer in eine zielkundengerechte Produktentwicklung einbeziehen,
- und die Kompetenzen der begleitenden Kurse anwenden und in einen Zusammenhang bringen,

damit Sie

- die in den Kursen erworbenen Kenntnisse durch praktische Anwendung und eigene Erfahrungen weiter vertiefen und verinnerlichen,
- und später innovative digitale Produkte erfolgreich an den Markt bringen.

### Prüfungsformen

- Präsentation von Zwischen- und Endergebnissen
- Bewertung von Dokumentationen, Entwürfen oder prototypischen Entwicklungen und Minimum Viable Product zu vereinbarten Meilensteinen

---

## Modul »Community & Reflection 5« (CR5)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 5

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** unbenotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

### Prüfungsformen

Präsentation der Ergebnisse, schriftliche Dokumentation

---

## Modul »Optionales Praxissemester« (OPT)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 0 (6 bei optionalem Praxissemester)

**Häufigkeit:** fortwährend

**Kreditpunkte:** 30 ECTS

**Arbeitsaufwand:** 0h Kontaktzeit / 900h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / PROD / AGILE / ENSI / COMM / TSCE / CREA / SOSE

**Modulprüfung:** unbenotet

### Qualifikationsziele des Moduls

Im optionalen Praxissemester haben die Student\*innen die Wahl zwischen drei Alternativen mit jeweils unterschiedlichen Qualifikationszielen zur Ausgestaltung des Moduls:

- Internship
- Self-Initiated Project
- Auslandssemester

Die Wahl lässt den Studierenden Freiraum zur Selbstgestaltung eines eigenen Profils. Das Praxissemester dient zudem als Mobilitätsfenster. Es ist unbenotet.

### Modulvariante »Praktikum«

In der Modulvariante »Praktikum« haben die Studierenden die Möglichkeit, ihre bisher erlangten Kompetenzen in verschiedenen Bereichen praktisch zu erproben und zu vertiefen. Mögliche Bereiche sind beispielsweise Institutionen innerhalb von Communities, Vereine, NGOs, Unternehmen oder Startups. Die Studierenden erlangen ein vertieftes Verständnis ihrer berufspraktischen Tätigkeit und steigern ihre Kompetenzen hinsichtlich der Softskills und Querschnittsqualifikationen, bspw. Kommunikation, Organisation, Zeit- und Selbstmanagement.

### Modulvariante »Self-Initiated Project«

In der Modulvariante »Self-Initiated Project« haben die Studierenden die Möglichkeit, ein eigenes Projekt mit einer hohen Workload von 900h in einem Team zu initialisieren und zu realisieren. Die Studierenden wenden die bisher gebildeten Kompetenzen an und können sich für ihre eigene neue Aufgabenstellung neue Kompetenzen und Technologien aneignen. Das Projekt soll das Potential haben, ein Minimum Viable Product zu realisieren, welches das Potenzial für eine weitere Verwertung hat (bspw. Pitch vor Investoren, Crowd-Funding, Bildung einer Community, Open Source Beitrag). Die Studierenden erlangen ein vertieftes Verständnis ihrer berufspraktischen Tätigkeit und steigern ihre Kompetenzen hinsichtlich der Softskills und Querschnittsqualifikationen, bspw. Kommunikation, Organisation, Projekt-, Konflikt, Zeit- und Selbstmanagement.

In der Modulvariante »Auslandssemester« haben die Studierenden die Möglichkeit, ihre Fachkompetenzen und wissenschaftlichen Fähigkeiten zu vertiefen und akademische Perspektive zu erweitern, bspw. durch den Besuch interdisziplinärer Lehrveranstaltungen im Ausland. Zudem können die Studierenden ihre inter-kulturellen Kompetenzen und Sprachkompetenzen stärken.

**Prüfungsformen**

- Anerkennung der erbrachten Leistungen beim Auslandssemester
- Projektdokumentation und Vortrag

---

## Modul »Praxisprojekt« (PP)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 6 (7 bei optionalem Praxissemester)

**Häufigkeit:** fortwährend

**Kreditpunkte:** 10 ECTS

**Arbeitsaufwand:** 0h Kontaktzeit / 300h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / PROD / AGILE / ENSI / COMM / TSCE / CREA / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Die Studierenden

- können Methoden und Techniken, die sie im Studium erlernt haben, in realitätsnahen Projekten weitgehend selbstständig anwenden
- haben erste Erfahrungen mit der Selbststeuerung und proaktiven Kommunikation in einem Projekt mittlerer Größe gesammelt

### Lehrform/SWS

Angeleitetes, eigenverantwortliches Arbeiten

### Inhalt

Modulinhalte des ersten bis fünften Semesters anhand von realen Anforderungen in einem praxisrelevanten Kontext anwenden und den Studierenden durch die Betreuung des Dozenten an eine selbstständige Projektdurchführung und Kommunikation heranführen. Das Praxisprojekt kann beispielsweise in Communities, Vereinen, NGOs, Unternehmen, Startups oder in der Hochschule - dann eingebettet in Forschungsprojekte - erfolgen.

### Studien-/Prüfungsleistungen

- Projektdokumentation
- Vortrag

---

## Modul »Bachelorarbeit« (BA)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 6 (7 bei optionalem Praxissemester)

**Häufigkeit:** fortwährend

**Kreditpunkte:** 12 ECTS

**Arbeitsaufwand:** 0h Kontaktzeit / 160h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / PROD / AGILE / ENSI / COMM / TSCE / CREA / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Die Bachelorarbeit soll zeigen, dass der Prüfling befähigt ist, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus seinem Fachgebiet sowohl in ihren fachlichen Einzelheiten als auch in den fachübergreifenden Zusammenhängen nach wissenschaftlichen, fachpraktischen und gestalterischen Methoden selbstständig zu bearbeiten. Die Bachelorarbeit ist in der Regel eine eigenständige Untersuchung mit einer Aufgabenstellung aus der aktuellen Themen des Studiengangs und einer ausführlichen Beschreibung und Erläuterung ihrer Lösung. In fachlich geeigneten Fällen kann sie auch eine schriftliche Hausarbeit mit fachliterarischem Inhalt sein.

---

## Modul »Bachelor Kolloquium« (BK)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 6 (7 bei optionalem Praxissemester)

**Häufigkeit:** fortwährend

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 0h Kontaktzeit / 90h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / IEE / PROD / AGILE / ENSI / COMM / TSCE / CREA / SOSE

**Modulprüfung:** benotet

### Learning Outcome / Qualifikationsziele des Moduls

Das Kolloquium dient der Feststellung, ob der Prüfling befähigt ist, die Ergebnisse der Bachelorarbeit, ihre fachlichen Grundlagen, ihre fachübergreifenden Zusammenhänge und ihre außerfachlichen Bezüge mündlich darzustellen und selbständig zu begründen und ihre Bedeutung für die Praxis einzuschätzen. Dabei soll auch die Bearbeitung des Themas der Bachelorarbeit mit dem Prüfling erörtert werden.

### Lehrform/SWS

- Angeleitetes, eigenverantwortliches Arbeiten

### Inhalt

- Vortrag über das Thema der Bachelorarbeit
- Fachdiskussion und mündliche Verteidigung der Arbeit

### Prüfungsform

- mündliche Prüfung, Vortrag und Fachdiskussion

---

## Modul »Community & Reflection 6« (CR6)

**Verantwortlich:** alle Professor\*innen im Studiengang

**Studiensemester:** 6 (7 bei optionalem Praxissemester)

**Häufigkeit:** fortwährend

**Kreditpunkte:** 5 ECTS

**Arbeitsaufwand:** 20h Kontaktzeit / 130h Selbstlernzeit

**Kompetenzcluster:** SOSE

**Modulprüfung:** unbenotet

### Learning Outcome / Qualifikationsziele des Moduls

Teilnehmer\*innen der »Community & Reflection«-Module können an der Organisation, Selbstverwaltung, Kommunikation und Weiterentwicklung des Studiengangs teilhaben, in dem sie:

- organisatorische und institutionelle Strukturen, Abläufe und Zusammenhänge kennen lernen und ein kritisches Verständnis für selbige entwickeln
- Projekte und Events im Team planen und durchführen
- Aufgabenstellungen in heterogenen Teams eigenständig entwickeln
- Teamprozesse lösungsorientiert moderieren und Konflikte managen
- neuen Themen (Technologien, Methoden, Werkzeuge) erschließen und das neue Wissen bzw. neue Kompetenzen an Peers vermitteln
- Arbeitsgemeinschaften und Kommunikationsformate weiterführen und entwickeln
- Verantwortung für das eigene Studiumfeld übernehmen

um später im beruflichen und gesellschaftlichen Kontext professionell handeln zu können und das Arbeitsumfeld als auch die Disziplin als solches zu hinterfragen und mitgestalten zu können.

### Voraussetzungen für die Vergabe von Credits

Aktive Teilnahme an den Events und Veranstaltungen, sowie die Erstellung eigenständiger mündlicher und schriftlicher Beiträge und Dokumentationen.

### Prüfungsformen

Präsentation der Ergebnisse, schriftliche Dokumentation



## Übersicht über Kurse und Katalog

Teilmodul	Kurs / Lehrveranstaltung	SDA	DRSS	IEE	PROD	AGILE	COMM	TSCE	CREA	ENSI	SOSE	CP	Prüfung
CS11	Computational Thinking	x										3	benotet
CS12	Coding Essentials 1	x										3	benotet
CS13	Coding Essentials 2	x										3	benotet
CS14	Client Server Basics	x										3	benotet
DF11	Design Thinking					x						3	benotet
DF12	Generative Gestaltung								x			3	benotet
DF13	Product Dimensions			x	x							3	benotet
CS24	Advanced Clients		x									3	benotet
CS21	Agile Team Coding	x				x						3	benotet
CS23	Application Design	x										3	benotet
CS22	Clean Code	x										3	benotet
CI11	Entrepreneurial Thinking					x				x		3	benotet
CI12	Rapid Prototyping		x						x			3	benotet
CI13	Social Hubs			x				x				3	benotet
DT12	Applied AI	x	x									3	benotet
DT13	Immersive Applications	x	x		x							3	benotet
DT11	Microservice Architectures	x	x									3	benotet
CI21	Trend and Market Research				x			x		x		3	benotet
CI23	Open Ecosystems				x			x		x		3	benotet
CI23	Business Economics									x		3	benotet
DT21	Internet of Things	x	x									3	benotet
DT23	Connected Products		x		x							3	benotet
DT22	DevOps					x						3	benotet
DF21	Design as a Language			x	x							3	benotet
DF22	Interaction Design			x					x			3	benotet
DF23	Users and Situated Action			x					x			3	benotet
DF32	Global Citizenship			x				x				3	benotet
DF31	Open Design			x				x	x			3	benotet
CI31	Business Models and Plans									x		3	benotet
CI32	Entrepreneurial Marketing						x			x		3	benotet

Tabelle 0.1: Übersicht über die Teilmodule und Vernetzung der Kurse mit den einzelnen Kompetenzclustern.

---

# Kurs »Computational Thinking«

**Verantwortlich:** Prof. Dr. Frank Schimmel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** Essay, am 2. Freitag im Block (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können eine gegebene Problemstellung ohne Programmierkenntnisse lösen, indem sie bewusst Prinzipien des "Computational Thinkings" anwenden, um diese später bei der Entwicklung eines (Software-)Produkts anwenden zu können.

## Inhalt

- wichtige Prinzipien des CT
  - Abstraktion und Modellbildung
  - Dekomposition
  - Mustererkennung
  - Design von Algorithmen
  - Datenerhebung/-analyse/-repräsentation
  - Automatisierung
  - Parallelisierung
  - Simulation
- Kognitive Grundlagen
  - Chunking/Recoding, Hierarchisierung, Schemabildung

## Lehr- und Lernformen

- Vorlesung
- Gastvortrag externer Referenten
- Planspiele
- praktische Übungen

## Materialien/ Ressourcen

- [https://docs.google.com/document/d/1i0wg-BMG3TdwsShAyH\\_0Z1xpFnpVcMvpYJceHGWex\\_c/edit](https://docs.google.com/document/d/1i0wg-BMG3TdwsShAyH_0Z1xpFnpVcMvpYJceHGWex_c/edit)

---

# Kurs »Coding Essentials 1«

**Verantwortlich:** Prof. Dr. Christian Faubel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** Produktabgabe, am 2. Freitag im Block (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können in Einzelarbeit und auf Basis exakt spezifizierter Aufgabenstellungen kleine, konsolenbasierte Programme kodieren, indem sie Programmiersprachenkonzepte für das "Programmieren im Kleinen" anwenden, um später große Systeme auch auf Detailebene kodieren zu können.

## Inhalt

- Grundlegende Prozesse (Codierung, Kompilation/Interpretation)
- "Programmieren im Kleinen", z.B.
  - Expressions
  - Variablen, elementare Typen
  - Funktionen, Parameter
  - Verzweigungen, Schleifen
  - komplexe Typen (Arrays, Maps etc.)
- Grundlegende Tools
  - Compiler
  - IDE (z.B. Visual Studio Code)
    - \* Projekte verwalten (Classpath etc.)
    - \* Debugger bedienen
  - Build-Management-Tools (z.B. Maven)

## Lehr- und Lernformen

- Vorlesung
- Übungen zu einzelnen Inhalten
- begleitende Projektaufgabe:
  - Ausgangssituation: Exakt spezifizierte Anforderungen an ein sehr kleines System (Kommandozeilen-tool zu String-Manipulationen, Taschenrechner etc.)
  - Zielsituation: lauffähiges System

## Materialien/ Ressourcen

- Liste ausgewählter Literatur und Web-Ressourcen
- Vorlesungsunterlagen

- 
- Übungsunterlagen inkl. Lösungen
  - Projekt-Meilensteinbeschreibungen
  - vorkonfigurierte Entwicklungsumgebungen
  - vorgefertigte Entwicklungsstände

---

# Kurs »Coding Essentials 2«

**Verantwortlich:** Prof. Dr. Frank Schimmel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** Dokumentation, gesonderte Deadline (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können in Einzelarbeit und auf Basis exakter jedoch veränderlicher Anforderungen größere Programme mit graphischen Benutzeroberflächen kodieren, indem Sie Programmiersprachenkonzepte für das "Programmieren im Großen" sowie gängige Frameworks und Libraries verwenden, um später große und wartbare Systeme kodieren zu können.

## Inhalt

- "Programmieren im Großen", insbesondere
  - Klassen, Interfaces
  - Packages
  - Visibilities
  - Generizität
  - Lambdas
  - Ausnahmebehandlung
- Grundlegende Muster wie z.B.
  - Singletons
  - Callbacks
  - Observer
- Frameworks und Libraries
  - ... zur Serialisierung / Deserialisierung (z.B. java.io)
  - ... zur Erzeugung von graphischen Benutzeroberflächen (z.B. JavaFX)
  - ... zum Logging
- Umgang mit Versionsverwaltungs-Tools wie Git bei Einzelarbeit
  - Git-Basics (Repositories, Commits etc.)
  - Repositories aufsetzen (git clone, git init)
  - Versionen erzeugen (git commit)
  - Versionen markieren (git tag)

## Lehr- und Lernformen

- Vorlesung

- 
- Übungen zu einzelnen Inhalten
  - begleitende Projektaufgabe:
    - Ausgangssituation: Exakt spezifizierte Anforderungen an ein kleines System (z.B. Addressverwaltung mit JavaFX-GUI)
    - Zielsituation: lauffähiges System

#### **Materialien/ Ressourcen**

- Liste ausgewählter Literatur und Web-Ressourcen
- Vorlesungsunterlagen
- Übungsunterlagen inkl. Lösungen
- Projekt-Meilensteinbeschreibungen
- vorkonfigurierte Entwicklungsumgebungen
- vorgefertigte Entwicklungsstände

---

# Kurs »Client Server Basics«

**Verantwortlich:** Prof. Dr. René Wörzberger

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** Coding Session, am 2. Freitag im Block (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können in Einzelarbeit auf Basis exakter und vollständiger Anforderungen eine webbasierte Client-Server-Applikation codieren, indem sie einen gegebenen Entwicklungs-Stack einsetzen, um später bei der Codierung großer derartiger Applikationen mitwirken zu können sowie alternative Frameworks und Libraries einordnen zu können.

## Inhalt

- Der Browser als Client
  - Grundlegende Protokolle und Standards wie z.B.
    - \* das Hypertext Transfer Protocol (HTTP)
    - \* der Domain Name Service (DNS)
    - \* Uniform Resource Identifier (URI)
  - Auszeichnungen und Strukturierungen von Inhalten mit der Hypertext Markup Language (HTML)
  - Layoutdefinitionen mit Cascading Stylesheets (CSS)
  - Dynamisierung mit JavaScript (ECMAScript), Asynchronous JavaScript and XML (Ajax) und dem JavaScript Object Notation (JSON)
- Web-Server
  - Konfiguration von Web-Servern wie Nginx
  - Serving von statischen Inhalten
  - Caching als wichtiges Mittel des World-Wide-Web
- Application-Server
  - Wichtige Spezifikationen wie z.B.
    - \* Servlets als Bindeglied zwischen Clients und Application-Servern
    - \* Java Persistence API (JPA) als Bindeglied zwischen Application-Servern und Datenbank-Servern
  - industriell relevante Implementierungen
    - \* Apache Tomcat
    - \* Spring (Boot)
- Datenbanken
  - Relationale Datenbanken

- 
- \* Grundlegende Konzepte wie Attribute, Schlüssel, Tupel, Beziehungen und Transaktionen
  - \* Datenbankzugriffe mit der Structured Query Language (SQL)
  - \* Konfiguration einer MySQL-Datenbank
  - NoSQL-Datenbanken
    - \* Vor- und Nachteile von NoSQL-Datenbanken
    - \* Konfiguration z.B. einer MongoDB
  - Programmatischer Zugriff auf Datenbanken (mittel z.B. JPA)

### Lehr- und Lernformen

- Vorlesung
- Übung
- begleitende Projekt-Aufgabe:
  - Ausgangssituation: exakt ausformulierte, jedoch auf wesentliche Use-Cases beschränkte Anforderungen an ein webbasiertes System wie z.B. Vorlesungsverwaltung, Warenwirtschaftssystem oder Issue-Tracking
  - Zielsituation: abnahmefähiges webbasiertes System

### Materialien/ Ressourcen

- Liste ausgewählter Literatur und Web-Ressourcen
- Vorlesungsunterlagen
- Übungsunterlagen inkl. Lösungen
- Projekt-Meilensteinbeschreibungen
- vorkonfigurierte Entwicklungsumgebungen
- vorgefertigte Entwicklungsstände

---

# Kurs »Design Thinking«

**Verantwortlich:** Prof. Dr. Ivonne Preusser

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** AGILE

**Teilmodulprüfung:** Präsentation, in Prüfungswoche (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Studierende können Design Thinking als einen Ansatz zur Entwicklung von innovativen Ideen für Produkte anwenden, um später nach dieser Methode Ideen für eigene Projekte zu explorieren.

Absolvent\*innen des Kurses lernen Design Thinking als agile, nutzerzentrierte Arbeitsweise zur Gestaltung innovativer Lösungen kennen und Methoden anwenden, indem sie

- die Design Thinking Grundlagen (z.B. „human-centered-Design“, Elemente, Mind-Set) kennenlernen,
- einen Design Thinking Prozess umsetzen und anhand realer Beispiele durchlaufen,
- Zielgruppen und ihre Bedürfnisse verstehen und dies in den Entwicklungsprozess einbeziehen,
- Methoden der Nutzer- und Kundenforschung anwenden,
- Kreativmethoden nutzen, um Ideenfindung voranzutreiben,
- Problem- und Lösungsräume miteinander verknüpfen,
- Ideen bis hin zu einem ersten Prototyp entwickeln,
- agile inkrementell-iterative Vorgehensmodelle anwenden,
- in selbstorganisierten Teams Lösungen entwickeln
- und Einsatzmöglichkeiten reflektieren

um Ideen für eigene Projekte zu explorieren und Lösungsideen evaluieren zu können.

## Inhalt

Die Teilnehmer\*innen des Kurses durchlaufen einen typischen Design Thinking Prozess mit den Phasen: Verständnis, Erkundung, Synthese, Ideenfindung, Prototypen und Test.

Lernziel ist es, Ideen unter Anwendung einer bestimmten Systematik (Design Thinking Prozess) reifen zu lassen und Prototypen zu entwickeln, bei dem der Mensch im Fokus der Betrachtung steht. Dabei unterstützen Kreativmethoden, iterative Testung und interviewgeleitetes Feedback.

- Design Thinking-Prozess
- Kreativmethoden
- Iterative Testung
- Prototypen
- Nutzer- und Kundenforschung (z.B. qualitative und quantitative Verfahren, Interviews, Befragung, Methoden der Marktforschung)

- 
- Bedürfnisse der Nutzer eruieren und prüfen
  - Persona erstellen, Methode um Kundenbedürfnisse zu identifizieren

#### **Lehr- und Lernformen**

- Workshop
- Impuls-Vorträge
- Interaktives Seminar
- Kollaborative Teamarbeit
- Praxis-Übungen
- Projektarbeit

#### **Materialien/ Ressourcen**

- Brenner, W.; Uebernickel, F. (Hrsg.) (2016): Design Thinking for Innovation: Research and Practice. Cham: Springer International Publishing, 2016
- Brown, T. (2009): Change by design – How design thinking transforms organizations and inspires innovation. New York: Harper Business
- Plattner, H., Meinel, C., & Leifer, L. (Hrsg.). (2015). Understanding Innovation. Design thinking research: Building innovators. Cham, s.l.: Springer International Publishing. Retrieved from <http://dx.doi.org/10.1007/978-3-319-06823-7>

---

# Kurs »Generative Gestaltung«

**Verantwortlich:** Prof. Christian Noss

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** CREA

**Teilmodulprüfung:** Coding Session, am 2. Freitag im Block (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Teilnehmer\*innen des Kurses können eigene Konzepte und Code für regelbasierte und generierte Artefakte analysieren und entwickeln, indem sie

- einen Überblick über die gestalterischen Möglichkeiten kennen lernen
- Tools und Techniken kennen und einsetzen können
- die Grundprinzipien Wiederholung, Zufall und Logik verstehen und anwenden können

so dass sie generative Gestaltung im Entwurfsprozess nutzen zu können.

## Inhalt

- Inspiration und Projekte
- Farbe, Form, Typographie & Bild
- Zufall und Rauschen
- Attraktoren & Agenten
- Bäume

## Lehr- und Lernformen

Workshop

## Materialien/ Ressourcen

- Kostas Terzidis: Algorithms for visual design using the processing language; Wiley Publishing, 2009
- Benedikt Groß, Hartmut Bohnacker, Julia Laub, Claudius Lazzeroni: Generative Gestaltung: Creative Coding im Web; Entwerfen, Programmieren und Visualisieren mit Javascript in p5.js; Verlag Hermann Schmidt, 2018
- p5.js

---

# Kurs »Product Dimensions«

**Verantwortlich:** Prof. Dr. Laura Popplow

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / PROD

**Teilmodulprüfung:** Klausur, in Prüfungswoche (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Studierende lernen, was Design im Kontext eines Produktrahmens bedeutet, indem sie ...

- den Ansatz "Design as a language" kennenlernen: (> mindset > jackpot)
- Denkweisen - Herangehensweisen- Wirkungsweisen unterscheiden lernen,
- Szenarien zur Produktentwicklung in unterschiedlichen Kontexten entwickeln (Storylines / Storytelling / Text)
- Arten von Prototypes kennenlernen und verstehen (Material / Materialität)
- unterschiedliche Perspektiven einzunehmen lernen (Roleplay)
- unterschiedliche Ebenen eines Produktes multiperspektivisch interpretieren (Communication)
- Ideen, Bilder, Texte und Sounds, Filme bzw. ganze Kommunikationskampagnen dazu entwickeln,
- Sustainability, durability and recycling hinterfragen
- Methoden kennenlernen um social layer neu zu denken

um ...

- interaktive Umgebungen zu verstehen darin ein Produkt zu gestalten
- die Entwicklungsschritte zu kennen
- die Relevanz & Qualität eines Produktes einzuschätzen lernen,
- Pipelines zu verstehen und Konsequenzen abzusehen
- Prozesse innerhalb der Gestaltung zu steuern, zu planen
- "relational Environments: Connected products and ecosystems" kennenzulernen
- Konzeptionstärke für originellen Produktideen freizusetzen
- Gestaltung von Kommunikation und Erfahrungen zu bewerkstelligen

## Inhalt

- What is a product? What is the context? What is a digital product within a certain context? What is virtual product?
- What means product design, when a product is not visible anymore? What are relational interactive environments? What is the difference between a feature and a bug?
- Genres & Formatfragen

---

### Lehr- und Lernformen

- tbd

### Materialien/ Ressourcen

- tbd

---

# Kurs »Advanced Clients«

**Verantwortlich:**

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** DRSS

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können auf Basis gegebener Anforderungen und eines vorhandenen Servers einer Client-Server-Applikation einen dem Stand der Technik entsprechenden Client entwickeln, indem sie fortgeschrittene Techniken und Frameworks für Browser-basierte Clients (Web-Apps) oder native Apps einsetzen, um später Clients für vermarktungsfähige Software-Produkte entwickeln zu können.

## Inhalt

- Web-Apps
  - Browser-Konzepte und -APIs
    - \* Document Object Model
    - \* History
    - \* Storage
    - \* Service Worker
    - \* Web Sockets
    - \* ...
  - Advanced JavaScript (EcmaScript)
    - \* Lexical Environments
    - \* Execution Contexts
    - \* Scope Chain
    - \* Closures
    - \* Prototypes
    - \* Classes
    - \* Promises
    - \* Type Script
    - \* ...
  - Responsive Web Design
    - \* Media Queries
    - \* Frameworks (z.B. Bootstrap)
  - Single Page Applications
    - \* Grundlegende Konzepte
      - HTML-Rahmen mit Platzhalter
      - Routing innerhalb einer Web-Page

- 
- ViewModels im Browser
  - Server-API-Calls
  - \* Frameworks
    - React, Angular oder vue.js
  - native Apps
    - Aufsetzen einer Entwicklungsumgebung
    - Grundlagen von Android-Native-Apps
      - \* Components
        - Activities
        - Services
        - Broadcast Receivers
        - Content Providers
      - \* APIs
        - Kamera
        - Standort
        - weitere Sensor-APIs

#### **Lehr- und Lernformen**

- Vorlesung
- Seminar
- Übungen

#### **Materialien/ Ressourcen**

- tbd.

---

# Kurs »Agile Team Coding«

**Verantwortlich:** Prof. Dr. Frank Schimmel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** AGILE / SDA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden verstehen und beherrschen die Entwicklungsdynamik bei Coding im Team, indem sie

- mittels agilen Prozessen und Werkzeugen neue Anforderungen selbstorganisiert und arbeitsteilig planen und umsetzen,
- agile Methoden und Rollen nutzen, um dabei effizient und effektiv vorzugehen,
- Regeln zur Qualitätssicherung im Team formulieren und umsetzen,
- teamorientierte Entwicklungs-, Test- und Auslieferungsprozesse durch Automatisierung beschleunigen,

damit sie später in digitale Produkte in einem agilen Prozess schnell und in hoher Qualität umsetzen können.

## Inhalt

- Konzepte
  - Agiles Vorgehen im Projekt, agile Ansätze (Scrum, XP, Kanban, ...)
  - Scrum-Rollen und -Prozesse
  - agiles Anforderungsmanagement (Backlog)
  - Kernkonzepte wie DoD, MVP
- Agiles Lego-Planspiel
  - Einüben des agilen Prozesses anhand eines nicht-technischen Planspiels
- Agile Planung
  - Backlog-Management, Definieren von Anforderungen in Epics, User Stories und Tasks
  - Sprintplanung
  - agile Aufwandsschätzung
- Build-Management und Versionierung (
  - Einführung in gängige Tools: IDEs, Git, Maven, Gradle, Jenkins
  - Elementare Continuous Integration
  - Branching, Dokumentation
- Testen im Team
  - Testarten (Module, Integration, Acceptance)
  - Teststrategien im Team

- 
- Einführung in gängige Testing-Tools
  - Dokumentation im Team

#### **Lehr- und Lernformen**

- Vorlesung
- Gastvortrag externer Referenten
- Planspiel
- praktische Übung
- Anwendung auf laufendes Projekt:
  - MVP, DoD: Produktdefinition und DoD definieren
  - Team-Build für ein laufendes Projekt aufsetzen
  - Teststrategie in der DoD für das Team abbilden, stichprobenartig umsetzen

#### **Materialien/ Ressourcen**

- Lego (für das agile Planspiel)
- Methodenkarten wie z.B. Planning Poker Sets

---

# Kurs »Application Design«

**Verantwortlich:** Prof. Dr. Stefan Bente

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden sind in der Lage, effizient und nachhaltig Softwareapplikationen zu erstellen, indem sie

- in einer bewussten, kriteriengestützten Entscheidung das beste Werkzeug (Sprache, Tool, Methode, Framework) für die aktuelle Aufgabe auswählen,
- bekannte Paradigmen, Technologien, Patterns und Architekturstile für den Entwurf und die Umsetzung eines Softwaresystems nutzen können, sowie
- in der Lage sind, diese problembezogen und pragmatisch anzupassen,

so dass sie digitale Produkte mit hohem Softwareanteil schnell und mit Blick auf spätere Weiterentwicklung herstellen können.

## Inhalt

- Konzept von Software-Patterns verstehen
  - Idee von Patterns
  - Verwendung von Pseudocode
- Schichtenarchitektur
  - Rolle der einzelnen Schichten
  - Patterns pro Schicht
- Gängige Patterns erkennen, einschätzen können und nutzen
  - Basispatterns
  - Dependency Injection
  - Inversion of Control
- Beyond OOP
  - Composition over Inheritance
  - Vererbungstiefen
- Weiterführende Konzepte
  - Kernprinzipien (Separation of Concerns, ...)
  - Konzept von Architekturstilen
  - Modularisierung, Austauschbarkeit

- 
- Re-Use von Software-Komponenten
  - Event-Handling, Event-Patterns, Event-driven Architecture

#### **Lehr- und Lernformen**

- Vorlesung
- Gastvortrag externer Referenten
- Seminar
- Übung
- Anwendung auf konkretes Projekt:
  - Patterns: gezielt Muster im eigenen Code identifizieren und/oder einbauen; über den Nutzen gemeinsam reflektieren
  - Modularisierung für ein Code-Beispiel (eigenes oder vom Dozent vorgegeben) analysieren / umsetzen
  - Messaging für eigenes Projekt aufsetzen

#### **Materialien/ Ressourcen**

- Blaupausen und Tutorials für Software-Entwicklung (IDEs, Sourcecode-Management, ...)
- Vorgefertigte Umgebungen für das Deployment von Software-Prototypen und -Lösungen (insb. Infrastruktur für Continuous Integration / Deployment)
- Modellierungswerkzeuge (digital und/oder Whiteboard)

---

# Kurs »Clean Code«

**Verantwortlich:** Prof. Dr. Frank Schimmel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden sind in der Lage, qualitativ hochwertigen Code zu produzieren, indem sie

- die Code-Qualität ex-ante durch passende Coding-Guidelines sicherstellen,
- die Prinzipien des Clean Code (Bob Martin) verinnerlichen und anwenden,
- die Qualität des Codes ex-post durch passende Techniken sicherstellen (TDD) und verbessern (Refactoring),
- die Qualität des Codes in "-ities" messen und bewerten (Usability, Maintainability, Scalability, Security, Reliability, ...),

so dass der Code langfristig wartbar bleibt, selbsterklärend ist und effiziente Teamarbeit ermöglicht.

## Inhalt

- Clean-Code als elementares Leitmotiv beim Coding
  - Konzepte "Bad Smells" und Antipatterns
  - Gängige, oft auftretende Antipatterns / Bad Smells
- Refactoring als Methode und Haltung
  - Richtiges Vorgehen beim Refactoring
- Test-Driven- und Business-Driven Development als Paradigmen
  - Zusammenspiel mit Clean Code Methoden
  - Praktische Hinweise zu einem pragmatischen Umgang

## Lehr- und Lernformen

- Vorlesung
- Gastvortrag externer Referenten
- Seminar
- Übung
  - TDD: Dozenten stellen Unit Tests zur Verfügung, Studierende implementieren als erste Übung dagegen
- Anwendung auf konkretes Projekt:
  - Refactoring: Bad Smell oder Antipattern im eigenen Projekt finden und durch Refactoring beheben
  - TDD: neues Feature zu eigenem Code hinzufügen

---

## Materialien/ Ressourcen

- Blaupausen und Tutorials für Software-Entwicklung (IDEs, Sourcecode-Management, ...)

---

# Kurs »Entrepreneurial Thinking«

**Verantwortlich:** Prof. Dr. Jonas Schild

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** ENSI / AGILE

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierende verstehen die Grundlagen des Entrepreneurship und können diese anwenden, indem sie

- die Herkunft und wirtschaftliche Bedeutung von Gründungen kennen,
- die betriebswirtschaftliche Sichtweise von Gründer, Investoren und Kunden auf innovative Produkte verstehen, deren Sprache verstehen und sprechen können,
- bestehende Gründungen betriebswirtschaftlich analysieren und bewerten können
- etablierte und moderne Konzepte und Ansätze der Gründung beschreiben und anwenden können

um Ideen für innovative Produkte und Geschäftsmodelle in einem Gründungskontext fundiert zu diskutieren und selbst zu entwickeln.

## Inhalt

- Grundlagen Entrepreneurship – Herkunft, Entwicklung, Volkswirtschaftliche Bedeutung, Abgrenzung Corporate/Social/Intrapreneurship
- Etablierte Sichtweisen auf Gründung – Businessplan, Investoren Pitch
- Moderne Ansätze – Effectuation, Lean Start Up, Crowdfunding/funding
- Ansätze zur Opportunity Recognition

## Lehr- und Lernformen

- Seminaristischer Unterricht zu Grundlagen
- Analyse eines bestehenden StartUps (Geschichte, Erfolgsfaktoren, Geschäftsmodell) z.B. Fond of Bags, True fruits, My Muesli, (weitere aus der IT Branche)

## Materialien/ Ressourcen

- Ries (201x): Lean StartUp
- tbd

---

# Kurs »Rapid Prototyping«

**Verantwortlich:** Prof. Dr. Christian Faubel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** DRSS / CREA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Absolvent\*innen des Kurses beherrschen den Umgang mit materiellen und Code-basierten Entwurfsprozessen, indem sie

- Prototypen mit Methoden des Physical Computing entwickeln und dabei das Ineinandergreifen von Code und Material als gestalterisches Handlungsfeld nutzen,
- zur Realisierung dieser Prototypen sowohl klassischen Modellbau als auch digitale Fabrikation einsetzen und
- Gestaltung als iterativen Prozess betreiben, der den ständigen Dialog von Material, Code, Funktion und Ästhetik sucht

um später Prototypen unterschiedlichster Art (formal, funktional, narrativ) in den verschiedenen Stadien interdisziplinärer Projekte zu realisieren.

## Inhalt

- Grundlagen des Physical Computing im Design-Kontext
- Modellbau
- digitale Fabrikation
- Formen von Prototypen
- Iterative Gestaltungsprozesse
- Code und Material

## Lehr- und Lernformen

- Vorlesung
- Praktische Übung

## Materialien/ Ressourcen

- Modellbauwerkstatt
- Werkzeuge digitaler Fabrikation
- 2D und 3D Entwurfssoftware

---

# Kurs »Social Hubs«

**Verantwortlich:** Prof. Dr. Laura Popplow

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / TSCE

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Absolvent\*innen des Kurses gestalten und nutzen digitale soziale Kontexte, indem sie

- unterschiedliche soziale Technologien, wie Wikis, Instant Messenger, Image Boards, oder Online-Spiele und die darin gängigen Kulturtechniken nutzen,
- neue soziale Organisationsformen wie Open Source, Crowd Sourcing, Remote Work, Hackathons, Hackerspaces oder Co-working projektbezogen einsetzen,
- und beim Entwurf neuer Anwendungen die sozialen Gefüge berücksichtigen, die im Umfeld von Software und Computertechnologie entstehen,

um später den Community-Kontext digitaler Produkte und Anwendungen absehen, planen und gezielt aufbauen zu können.

## Inhalt

- Social Computing
- Open Source, F(L)OSS, Creative Commons
- New work, remote work, collaboration
- Hackathons, Maker Spaces
- Digitale „Parallelwelten“ und ihre Kulturtechniken, von Online-Spielen bis Image Boards
- Geschichte sozialer Hubs: Startups, Networking
- Social Hubs als Gefüge von Personen, Unternehmungen, Netzwerken
- Persönlichkeitsbildung

## Lehr- und Lernformen

- Kurzprojekt: Organisation eines sozialen Kontexts (Event, Open-Source-Projekt, ...)

## Materialien/ Ressourcen

- Biella Coleman: Coding Freedom: The Ethics and Aesthetics of Hacking, University Press Group Ltd., 2013

---

# Kurs »Applied AI«

**Verantwortlich:** Prof. Dr. Christian Faubel

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden kennen Methoden der künstlichen Intelligenz und können diese anwenden, indem sie...

- ausgehend vom Geschäftsmodell und den Daten die Ziele eines KI-Projektes präzisieren,
- Anwendungsfälle hinsichtlich ihrer Daten aufbereiten und modellieren,
- intelligentes Systemverhalten formalisieren,
- geeignete Methode und Verfahren zur Realisierung auswählen,
- state-of-the-art Frameworks und Tools einsetzen (bspw. TensorFlow),
- und smarte KI-Komponenten evaluieren,

um später smarte digitale Produkte entwickeln zu können.

## Inhalt

- Historie Künstlicher Intelligenz
- Smartes Produktverhalten
- Daten verstehen, prüfen und aufbereiten
- Methoden der künstlichen Intelligenz
  - Supervised Learning
  - Reinforcement Learning
  - Deep Learning
  - GAN (Generative Adversarial Networks): "KI und Kunst"
  - Collective Intelligence
  - Best practices
- State-of-the-art Frameworks und Tools
- Case Studies in TensorFlow/Keras
- KI und Gesellschaft

## Lehr- und Lernformen

- Vorlesungen
- Workshops TensorFlow/Keras

## Materialien/ Ressourcen

- tbd.

---

# Kurs »Immersive Applications«

**Verantwortlich:** Prof. Dr. Jonas Schild

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS / PROD

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden sind in der Lage, ein digitales Produkt hinsichtlich immersiver Wirkungsweise zu konzipieren und technisch umzusetzen, indem sie

- die wahrnehmungspsychologischen Voraussetzungen für immersive Systeme und Dimensionen der User Experience verstehen
- Wichtige Grundlagen z.B. technischer, mathematischer und gestalterischer Art kennen und anwenden können,
- Basisprinzipien der Interaktion in immersiven Umgebungen wie Selektion, Manipulation, Navigation, Kollaboration verstehen und konzeptuell wie programmiertechnisch umsetzen können,
- Umsetzungswerkzeuge insb. Game Engines kennenlernen und für die Erstellung eigener interaktiver, ggf. vernetzter Prototypen anwenden können,

so dass sie Konzepte zu immersiven Produkten in eigenen interaktiven Prototypen umsetzen können.

## Lehr- und Lernformen

- Vorlesung
- Gastvortrag externer Referenten
- Seminar
- Praktische Übung

## Materialien/ Ressourcen

- Dörner R., Broll, W., Grimm, P, Jung, B., Virtual und Augmented Reality (VR/AR). Grundlagen und Methoden der Virtuellen und Augmentierten Realität, Springer Vieweg Berlin, Heidelberg, 2019, <https://link.springer.com/book/10.1007/978-3-662-58861-1>
- LaViola Joseph J. Jr., Kruijff Ernst, McMahan Ryan P., Bowman Doug, Poupyrev Ivan P., 3D User Interfaces 2nd ed., Theory and Practice, Pearson ITP, 2017
- <https://aframe.io/>

---

# Kurs »Microservice Architectures«

**Verantwortlich:** Prof. Dr. Stefan Bente

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** SDA / DRSS

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können mittlere bis große Software- und IT-Systeme konzipieren und umsetzen, indem sie

- die Fachlichkeit der Anwendung gemäß der Methoden des Domain-Driven Design analysieren und dekomponieren,
- diese in Services schneiden, deren Kernfunktionalitäten über REST-APIs und Messaging nach außen angeboten werden,
- dazu eine unter Anwendung von gebräuchlichen Architekturpatterns eine Microservice-Applikationsarchitektur entwerfen,
- für diese Servicelandschaft ein Hostingkonzept konzipieren, und
- die Umsetzung des Systems in einen agilen Entwicklungsprozess einbetten können, bei dem die Entwicklungsteams jeweils autark an einzelnen Services arbeiten.

um später große digitale Produkte entwickeln zu können, die ein komplexes, verteiltes Backend erfordern.

## Inhalt

- Einführung und Konzepte
  - Unterschiedliche Ansätze: MS vs. SOA
  - Prinzipien von Microservice-Architekturen
  - CAP-Theorem und Eventual Consistency
- Business Architecture
  - Domänen- und Bounded-Context-Analyse nach Methoden des Domain Driven Design (Evans)
  - Entities und Aggregates identifizieren
- Application Architecture
  - Good Practices für den Serviceschnitt
  - Gängige MS-Patterns (Sidecar, BFF, API Gateway, ...)
  - Whitebox-Sicht auf Microservices (Schichtenarchitektur innerhalb eines Services)
  - Anemic vs. Rich Domain Model
- Integration Architecture
  - API Design, API First vs. Code First
  - Zusammenspiel REST und Messaging

- 
- REST-API-Regeln, REST Level 2 vs. 3 vs. GraphQL
  - Umsetzung von Transaktionen, Orchestrierung vs. Choreographie
  - Resilienz großer Systeme
  - Infrastructure Architecture
    - Containerschnitt und –Orchestrierung
    - Testautomatisierung, CI / CD
    - Load Balancing

#### **Lehr- und Lernformen**

- Vorlesung
- Gastvortrag externer Referenten
- Seminar
- Übung
- Anwendung auf konkretes Projekt:
  - Domäne und BC analysieren, Entities und Aggregates
  - Serviceschnitt machen
  - einen Service beispielhaft umsetzen

#### **Materialien/ Ressourcen**

- Vorgefertigte Umgebungen für das Deployment von Software-Prototypen und -Lösungen (insb. Infrastruktur für Continuous Integration / Deployment)
- Modellierungswerkzeuge (digital und/oder Whiteboard)

---

# Kurs »Trend and Market Research«

**Verantwortlich:** Prof. Dr. Monika Engelen

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** PROD / ENSI / TSCE

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierende können Trend- und Marktforschung durchführen, indem Sie

- Sekundär- und Primärdatenerhebungen und -auswertung durchführen,
- diese bewerten, analysieren und interpretieren können
- und die strukturierten Ansätze zur Opportunity Recognition anwenden können,

um Ideen für innovative Produkte und Geschäftsmodelle zu finden, zu konkretisieren und zu bewerten.

## Inhalt

- Grundlagen Trends und aktuelle Megatrends und Technologietrends
- Grundlagen Marktforschung (Sekundär/Primärdatenerhebungen insb. Umfragen und Fokusgruppen für Kundenprofile, Analyse, Interpretation)
- Ansätze zur Opportunity Recognition

## Lehr- und Lernformen

- Seminaristischer Unterricht zu Grundlagen
- Recherche und Referat zu einem Trend-Thema z.B. einem Megatrend (Sustainability, New Work, Urbanisierung) oder Technologietrend (z.B. Blockchain)
- Durchführung einer Opportunity Recognition Methode (z.B. Analyse der Wertschöpfungskette einer Branche oder der Wertversprechen einer Produktkategorie)
- Durchführung einer Marktforschung (Umfrage oder Fokusgruppe) z.B. für das Projekt
- Eigene Trend/Marktrecherche, Formulierung und Bewertung eines eigenen innovativen Konzept bzw. Bewertung des Projekts

## Materialien/ Ressourcen

- Zukunftsinstitut (2018): Megatrends
- Homburg, Christian (2014): Grundlagen des Marketingmanagments, 4. Auflage
- Berekoven, Ludwig (2009): Marktforschung: Methodische Grundlagen und praktische Anwendung, 12. Auflage
- Kuß, Alber, Wildner, Raimund, Kreis, Raimund (2014): Marktforschung: Grundlagen der Datenerhebung und Datenanalyse, 5. Auflage
- Meffert, H. (1992), Marketingforschung und Käuferverhalten, 2. Auflage

- 
- Engelen, von Gaggern (2017): Opportunity Recognition: 15 Ansätze für mehr Unternehmenswachstum

---

# Kurs »Open Ecosystems«

**Verantwortlich:** Prof. Dr. Laura Popplow

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** PROD / ENSI / TSCE

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Absolvent\*innen können Wechselwirkungen eigener Artefakte und Ergebnisse mit offenen Ökosystemen erkennen, nutzen und herstellen, indem Sie

- offen zur Verfügung stehende relevante Artefakte identifizieren und nutzen,
- und die Funktionsweisen und kulturelle Zusammenhänge offener Prozesse und Communities analysieren,

um später eigene Ergebnisse und Projekte in offenen und partizipatorischen Prozessen herzustellen und zur Verfügung zu stellen.

## Inhalt

- Open Source und Open Web
  - Communities und Movements: Wer stellt Dinge zur Verfügung?
  - Wer sind die Leute, die die Sachen zur Verfügung stellen und was sind deren Motivationen?
  - Evaluation und Kriterien zur Auswahl, bspw. Qualität und Quantität (Committer, Historie)
  - Entscheidungen: was nutzen?
  - Historie zu Industrien, die sich entwickelt haben (bspw. Gaming)
  - Authorship und Copycats
  - github: Dinge anderer wie Code und Libs
- Open Design
  - partizipatorische Prozesse
  - social Design
  - durch Gesellschaft inspiriertes Design
  - Dialoge zwischen Gesellschaft und Design und Code (als Teilbereiche im Design, die mit code zu tun haben)
  - Demo-Szene / gaming
  - Ästhetik / Materialität
  - Co-Working-Spaces

## Lehr- und Lernformen

- tbd

---

## Materialien/ Ressourcen

- tbd

---

# Kurs »Business Economics«

**Verantwortlich:** Prof. Dr. Stefan Eckstein

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** ENSI

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden kennen und verstehen die wichtigsten Entscheidungsbereiche wirtschaftlichen Handelns, indem Sie

- grundlegenden Entscheidungen im Rahmen einer Unternehmensgründung treffen,
- die Aufgaben der Unternehmensführung wie die Konzeption einer tragfähigen Strategie kennen,
- die Aufgaben der Teilbereiche Produktion, Absatz und Marketing sowie Investition und Finanzierung beschreiben,
- Investitionsentscheidungen informationsgestützt treffen,
- und Kalkulationsverfahren der Investitionsrechnung anwenden und auswerten

um wirtschaftliche Konzepte im Unternehmenskontext anzuwenden.

## Inhalt

- Grundlagen
- Unternehmensführung: Ziele, Planung und Entscheidung, Ausführung und Kontrolle
- Investition und Finanzierung
- Konstitutive Entscheidungen
- Produktion
- Absatz und Marketing
- Sicht Gesamtunternehmen
- Zahlenwelt
- Lean Startup
- Plan Build Measure Learn
- Marktabschätzungen
- Szenariorechnungen
- Unternehmensplanspiel

## Lehr- und Lernformen

- tbd

## Materialien/ Ressourcen

- tbd

---

# Kurs »Internet of Things«

**Verantwortlich:** Prof. Dr. Matthias Böhmer

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** DRSS / SDA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können hardwarenah vernetzte Geräte entwickeln, die ihre physikalische Umgebung wahrnehmen und verändern, indem sie

- Sensoren und Aktoren zur Messung und Veränderung der Umwelt auswählen,
- hardwarenahe Software für Mikrocontroller und Einplatinencomputer entwickeln,
- geeignete Protokolle zur Vernetzung kennen und nutzen,
- und Konzepte und Architekturen zur Einbindung der IoT einsetzen,

um später Anwendungen und Produkte zu realisieren, bei denen digitale und dingliche Welten miteinander wechselwirken.

## Inhalt

- Physical Computing
- Retrofitting
- Device Kategorien
- Eigenschaften von Einplatinencomputern
- Programmierung von Mikrocontrollern (bspw. Arduino)
- Entwicklungsumgebung für IoT
- Sensoren und Aktoren
- Analogere und Digitaler Input und Output
- Low level communication protocols (bspw. I2C, SPI, UART)
- Architektur Paradigmen (requests/response vs. pub/sub)
- TCP/IP und Protokolle (bspw. MQTT, CoAP)
- Non-TCP/IP und Protokolle (bspw. NB-IoT, LoRa, LPWAN)
- Verteilte Architekturen (bspw. local gateway, remote gateway)
- Ausgewählte Technologien (bspw. NFC, RFID, Beacons)
- Cloud Computing und Edge Computing

## Lehr- und Lernformen

- Vorlesung
- Seminar
- Übungen

---

## Materialien/ Ressourcen

- Mattern: Die Informatisierung des Alltags. Springer, 2007.
- Andelfinger und Hänisch: Internet der Dinge - Technik, Trends und Geschäftsmodelle. Springer, 2015.
- Adryan: The Technical Foundations of IoT. Artech House Publishers, 2017.
- McEven: Designing the Internet of Things. Wiley, 2013.
- IoT Kits (bspw. Microcontroller, Sensoren, Aktoren)

---

# Kurs »Connected Products«

**Verantwortlich:** Prof. Dr. Matthias Böhmer

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** DRSS / PROD

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können verteilte Informationsarchitekturen entwickeln, die sich über mehrere Geräte verschiedener Gattungen erstrecken (bspw. Web, Mobile, IoT, TV, Speech), indem sie

- Paradigmen zur Gestaltung von multi-device Experience kennen und adaptieren,
- Techniken zur Synchronisation von verteilter Interaktion entwerfen und implementieren.

Dies versetzt sie in die Lage, Use Cases und User Journeys von vernetzten Produkten über verschiedene Interaktionspunkte zu verteilen.

## Inhalt

- Grundlagen verteilter Systeme
- Plattformen und Ökosysteme
- Context Awareness
- Multi-Device Experiences (Consistent Design, Continuous Design, Complementary Design)
- Authentication, Login und Pairing
- Herausforderung Konnektivität
- Architekturen (insb. Event-basiert)
- Synchronisation in verteilten Informationsarchitekturen
- aktuelle Technologien und Frameworks
- Fallstudien von vernetzten Produkten

## Lehr- und Lernformen

- Vorlesung
- Seminaristischer Unterricht zu Fallstudien
- Übungen zum Transfer in Projekte

## Materialien / Ressourcen

- Rowland et al.: Designing Connected Products - UX for the consumer internet of things. O'Reilly, 2015.

---

# Kurs »DevOps«

**Verantwortlich:** Prof. Dr. René Wörzberger

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** AGILE

**Teilmodulprüfung:** Coding Session, am 2. Freitag im Block (benotet)

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können fortwährend entwickelte Software-Systeme automatisiert bauen und automatisiert in von ihnen aufgesetzten virtualisierten Cloud-Umgebungen installieren, indem sie

- Grundkenntnisse in Linux (als dem marktbeherrschenden Server-Betriebssystem) besitzen und zur Administration einsetzen können,
- Build, Test und Deployment der Software-Systeme über eine CI/CD-Pipeline automatisieren,
- grundlegende Administrationskenntnisse (Firewall-, Proxy-, Domain-Konfiguration, ...) nutzen,
- und Werkzeuge zur Virtualisierung und zum Cloud-Computing anwenden,

damit sie digitale Produkte agil über den gesamten Lebenszyklus und im Sinne einer Ende-zu-Ende-Verantwortung ("you build it - you run it") realisieren können.

## Inhalt

- Basiswissen Linux und Netzwerke
  - Grundlegende Verfahren wie z.B. Public-Key-Authentifizierung
  - Grundlegende Werkzeuge wie z.B. bash und ssh
- Weiterführende Continuous Integration und Delivery
- Betrieb
  - klassische Data-Center-Infrastrukturen (Firewalls, Load-Balancer, Web-/App-/DB-Server etc.)
  - Virtualisierungsformen
  - Container-Virtualisierung mit z.B. Docker
  - Container-Orchestrierung mit z.B. Kubernetes
  - Monitoring mit z.B. Prometheus
  - Cloud-Computing mit z.B. Google Cloud

## Lehr- und Lernformen

- Vorlesung
- Gastvortrag externer Referenten
- Seminar
- Übung
- Projekt:

- 
- Ausgangssituation: existierendes System
    - \* mit vorhandener, jedoch manuell versionierter und zu übersetzender Code-Base,
    - \* mit manuellem Build, Test und Deployment auf verschiedenen, divergierenden Umgebungen,
    - \* mit klassischer, nicht-virtualisierter, unüberwachter Produktionsumgebung
  - Zielsituation: überarbeitetes System
    - \* unter vollständiger Versionskontrolle,
    - \* mit automatisiertem Build, Test und Deployment mittels einer CI/CD-Pipeline,
    - \* betrieben auf container-virtualisierten Umgebungen in einer Cloud

#### **Materialien / Ressourcen**

- Vorlesungsunterlagen
- Übungsunterlagen inkl. Lösungen
- Projekt-Meilensteinbeschreibungen
- Zugänge, Lizenzen und Kontingente für
  - gemeinsame Code-Repositories (z.B. GitLab)
  - Cloud-Computing-Provider (z.B. Google Cloud)
  - Entwicklungsumgebungen (z.B. IntelliJ IDEA)

---

## Kurs »Design as a Language«

**Verantwortlich:** Prof. Nina Juric

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / PROD

**Teilmodulprüfung:** benotet

**Angestrebte Lernergebnisse / Learning Outcomes**

tbd.

**Inhalt**

- Produktdimensionen (sozio, äst, techn grafik)
- 7 product dimensions
- Ästhetik / Materialität

**Lehr- und Lernformen**

- tbd.

**Materialien/ Ressourcen**

- tbd.

---

# Kurs »Interaction Design«

**Verantwortlich:** Prof. Dr. Lasse Scherffig

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / CREA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Absolvent\*innen des Kurses können die Interaktion mit digitalen Artefakten und Prozessen gestalten, indem sie

- Interaktionen mit GUIs und TUIs technisch realisieren,
- diese Interaktionen unter dem Einsatz von Prototypen und Methoden des Creative Coding experimentell gestalten,
- und Methoden und Konventionen aus der Geschichte der Interaktionsgestaltung einzusetzen, aber auch bewusst zu unterlaufen wissen

um später neuartige Interaktionsformen und User Experiences in interdisziplinären Projekten zur Anwendung zu bringen.

## Inhalt

- GUIs, TUIs und andere Paradigmen der Interaktion
- Geschichte des Interaction Design
- Materialität von Schnittstellen
- Experimentelle Interaktionsgestaltung

## Lehr- und Lernformen

- Vorlesung
- Praktische Übung
- Kurzprojekt

## Materialien/ Ressourcen

- Prototyping-Materialien
- Labor

---

# Kurs »Users and Situated Action«

**Verantwortlich:** Prof. Dr. Lasse Scherffig

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / CREA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Absolvent\*innen des Kurses können Interaktion für bestimmte Gruppen von Nutzer\*innen gestalten, indem sie

- die Wahrnehmung von Nutzer\*innen im Kontext verschiedener Interfaces und der damit zusammenhängenden Handlungszusammenhänge berücksichtigen,
- Interfaces im Sinne ihrer kontextuellen Situierung verstehen und gestalten
- und Methoden der Erfassung von solcher Kontexten einsetzen

um Interaktion und die durch Interfaces erzeugten Handlungsräume ganzheitlich zu entwerfen und umzusetzen.

## Inhalt

- Prototyping als Vermittlung zwischen (technischen, anwenderbezogenen, administrativen, ...) Kulturen und Kontexten
- Paradigmen des Umgangs mit Benutzer\*innen: User Models, Mental Models, Embodiment, Situated Action
- Technology Probes

## Lehr- und Lernformen

- Vorlesung
- Praktische Übung
- Kurzprojekt

## Materialien/ Ressourcen

- Prototyping-Materialien
- Labor

---

## Kurs »Global Citizenship«

**Verantwortlich:** Prof. Dr. Jonas Schild, Prof. Dr. Carl Georg Hartung

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / TSCE

**Teilmodulprüfung:** benotet

### Angestrebte Lernergebnisse / Learning Outcomes

An ausgewählten Problemen zur Fragestellung "Ethische Fragestellungen im Coding", die in der wissenschaftlichen Literatur beschrieben sind, führen die Studierenden Untersuchungen durch, z.B. durch eigene weitere Literaturrecherche, Interviews mit Akteuren im C&C-Umfeld usw. Sie erarbeiten sich damit die Fähigkeit, ethische Problemstellungen in ihrer beruflichen Praxis zu erkennen und sie diskutieren zu können.

- Literaturrecherche,
- Diskussion digitaler Produkte unter ethischen Aspekten,
- Interviews mit Akteuren im C&C-Umfeld usw.

Sie erarbeiten sich damit die Fähigkeit, eine auf ethischen Prinzipien beruhende Folgenabschätzung in ihrer beruflichen Praxis vorzunehmen und damit als "Global Citizen" verantwortungsvoll handeln zu können.

### Inhalt

- Grundlagen: Begriffe aus der Handlungsethik, einfache Beispiele
- Durchführung einer Projektstudie

### Lehr- und Lernformen

- Vortrag
- Seminar

### Materialien/ Ressourcen

- Kornfeld: Philosophie für Ingenieure, Hanser Verlag

---

# Kurs »Open Design«

**Verantwortlich:** Prof. Nina Juric

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** IEE / TSCE / CREA

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Studierende können offene gestalterische Prozesse initiieren und auswerten, indem sie

- das Open-Source-Projekte untersuchen,
- Methoden des Social und Participatory Design einsetzen
- die zur Entwicklung eigener Konzepte führen

um später Anwendungen entwickeln zu können, die relevant, sozialverträglich und innovativ sind.

## Inhalt

- partizipatorische prozesse
- social design
- durch gesellschaft inspiriert im design
- dialog gesellschaft und design plus code, als teilbereiche im design, die mit code zu tun haben
- demo szene / gaming
- Ästhetik / Materialität
- Co Working Spaces

## Lehr- und Lernformen

- Flashmobs
- Interviews
- Seminaristische Inputs
- Case studies
- Subversive Intervention
- Performance
- spielerische Formate, etc.

## Materialien/ Ressourcen\* Open Design Networks

- Fablabs
- Design Noir - The secret Live of electronic objects
- tbd.

---

## Kurs »Business Models and Plans«

**Verantwortlich:** Prof. Dr. Stefan Eckstein

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** ENSI

**Teilmodulprüfung:** benotet

### Angestrebte Lernergebnisse / Learning Outcomes

Die Studierenden können Geschäftsmodelle entwickeln, um eigene Projekte in erfolgreiche Unternehmungen zu überführen.

### Inhalt

- Geschäftsmodell und Breit-BWL
- Personal und Orga
- rechtliche Aspekte: Rechtsformen (welche? warum?)
- Schutzrechte, bspw. Erfindungen
- Liquidität, Investitionen
- Steueraspekte (Arbeitsrecht für Dummies, „zum Steuerberater“ für Dummies)
- BWL für Gründer
- ...

### Lehr- und Lernformen

- tbd

### Materialien/ Ressourcen

- tbd

---

# Kurs »Entrepreneurial Marketing«

**Verantwortlich:** Prof. Dr. Monika Engelen

**Sprache:** deutsch

**Kreditpunkte:** 3 ECTS

**Arbeitsaufwand:** 40h Kontaktzeit / 50h Selbstlernzeit

**Kompetenzcluster:** **ENSI** / **COMM**

**Teilmodulprüfung:** benotet

## Angestrebte Lernergebnisse / Learning Outcomes

Die Studierende können den Marketingmix und die Markteintrittsstrategie für ein innovatives Produkt entwerfen, indem sie die 4 Ps des Marketingmixes (Product, Price, Promotion, Place) für ein spezifisches Produkt und Zielgruppe analysieren und gestalten können und den Markteintritt konkret planen insb.

- Produktgestaltung, -komponenten sowie Marke basierend auf Kundenprofilen und -wertversprechen (Value Proposition Map) ableiten
- Preissetzung und -strategie sowie Erlösmodell orientiert an den Nachfragern, Wettbewerb und den eigenen Preisen definieren
- Basierend auf den Zielgruppen und Zielsetzungen, geeignete Kommunikationsmedien, -träger und -botschaften definieren und eine Kommunikationskampagne zum Markteintritt planen
- die geeigneten Vertriebsstrategie und -kanäle auswählen, die Stufen des Sales Processes zu planen und geeignete CRM Systeme einsetzen
- sowie Ihre Ideen zielkundengerecht (Investoren oder Kunden) überzeugend zu präsentieren (Pitch)

um Ideen für innovative Produkte und Geschäftsmodelle erfolgreich am Markt zu platzieren und zu etablieren.

## Inhalt

- Grundlagen Entrepreneurial Marketing
- Marketingstrategie, Zielkundenanalyse (Value Proposition Maps)
- Produkt-, Preis-, Kommunikation- und Vertriebspolitik für innovative Produkte/StartUps
- Erfolgreich Präsentieren – Storytelling und Pitchtraining

## Lehr- und Lernformen

- Seminaristischer Unterricht zu Grundlagen
- Eigene Analyse der Markteintrittsstrategie eines (erfolgreichen) StartUps (z.B. Tesla, fruitcore, Jimdo, true fruits, etc.)
- Erstellung eines Markteintrittskonzept entlang der 4 Ps für das Projekt
- Workshop Storytelling und Pitchtraining (z.B. in Kooperation mit GTC Gründer- und Technologiezentrum)

## Materialien/ Ressourcen

- NUK (2018): Handbuch zum Businessplan
- Homburg (2014): Grundlagen des Marketingmanagements, 4. Auflage

- 
- Freiling, Kollmann (2015): Entrepreneurial Marketing

# Kompetenzorientierung

---

# Handlungsfelder

## Handlungsfeld »Coding Software« (CS)

Im Bereich der Software Entwicklung modellieren und abstrahieren Absolvent\*innen die fachlichen Aspekte von Anwendungssystemen. Sie dekomponieren Problemstellungen in Teilprobleme, die in dedizierten Software-Komponenten durch eine Implementierung gelöst werden. Das Handlungsfeld »Coding Software« fasst die späteren beruflichen Tätigkeiten in diesem Bereich zusammen. Weitere Aktivitäten der Absolvent\*innen in diesem Handlungsfeld werden beispielsweise sein:

- das Schreiben von Source Code,
- das Implementieren von Software in Teams,
- das Anwenden agiler Vorgehensmodelle,
- das Entwerfen von Software Architekturen,
- das Entwickeln von Services im Backend und von Interfaces,
- das Testen von Source Code und Komponenten,
- das Absichern von Software Systemen,
- das Bereitstellen und in Betrieb nehmen von Software-Systemen.

Studierende des Studiengangs bilden in diesem Bereich Kompetenzen, um Software von einem ersten low-fidelity Prototypen bis zum Minimal Viable Product entwickeln zu können. Diese Kompetenzen werden insbesondere den Kompetenzclustern »Software Development and Architecture« und »Agile Methoden« zugeordnet.

## Handlungsfeld »Developing Things« (DT)

Im Bereich der Entwicklung digitaler Produkte realisieren die Absolvent\*innen smarte Objekte, die nicht nur ein digitales Abbild, sondern auch eine dingliche Repräsentanz besitzen, bzw. mit der realen Welt korrespondieren. Zum Handlungsfeld »Developing Things« gehören des Weiteren folgende exemplarische Tätigkeiten:

- das Prototyping verschiedener Alternativen im Entwicklungsprozess,
- die Verwendung von Sensoren, um Input der dinglichen Realität zu messen,
- die Nutzung von Aktoren, um eine physikalische Wirkung zu erzeugen,
- das Implementieren von Software für Mikrocontroller, Kleinstcomputer und mobile Devices,
- die gestalterische Arbeit mit den Wechselbeziehungen programmierter und materieller Eigenschaften smarter Objekte,
- das integrative Entwerfen von Code und Material,
- das Reflektieren von Funktionalität und Ästhetik,
- das Implementieren von smartem Systemverhalten auf Basis künstlicher Intelligenz,
- die Vernetzung physikalisch verteilter Komponenten,
- die Realisierung von Clients auf Basis einer natürlichen Mensch-Computer-Interaktion.

---

Für das berufliche Handlungsfeld »Developing Things« bilden die Studierenden im Studiengang Kompetenzen, um dingliche und realweltliche Artefakte mit digitalen Systemen vernetzen zu können. Diese Kompetenzen werden insbesondere den Kompetenzclustern »Dual Reality and Smart Spaces« und »Creative Coding zugeordnet«.

## Handlungsfeld »Designing Futures« (DF)

Absolvent\*innen erfassen die ganzheitliche Wirkung von interdisziplinären Ideen, Phänomenen, Szenarien, Visionen, Hypothesen, Produkten und deren Wirkungsweisen. Sie stellen sich den Herausforderungen einer sich rapide und nachhaltig wandelnden Gesellschaft, Kultur, Politik und Wirtschaft. Ein ganzheitliches Designverständnis und Ansätze wie Integrated Design, Critical Design, Speculative Design und Design Research befähigen sie, Erfahrungen in mehreren Dimensionen - auch jenseits von UI-, UX- oder HCI-Themen - zu gestalten und die kritische Auseinandersetzung mit relevanten Themen zu suchen. Zu den Tätigkeiten in diesem Handlungsfeld gehören exemplarisch:

- das Erfassen von Kontexten,
- das kritische Hinterfragen und spekulative Entwickeln neuer Produktideen,
- der Einsatz künstlerischer Ansätze zur kreativen Gestaltung,
- die Stimulation kreativen Denkens zur Ideenbildung und Förderung von Kreativprozessen,
- das Beobachten und Verstehen sozio-kultureller Phänomene,
- die Entwicklung und Darstellung zukunftsrelevanter Szenarien, basierend auf der spekulativen Wahrnehmung der Bedürfnisse der Nutzer\*innen von morgen,
- Design im Dialog mit verschiedenen Stakeholdern, durch den Einsatz von Participatory-Design- und Co-Design-Methoden,
- das Prototyping in verschiedenen Dimensionen unter Verwendung verschiedener Prototyping-Methoden,
- das Experimentieren mit Material, Modell, Methoden, Wissenschaft, Ethik und Philosophie.

Für das berufliche Handlungsfeld »Designing Futures« entwickeln die Studierenden im Studiengang Kompetenzen, um innovative Produkte und ihre Wirkung ganzheitlich gestalten zu können. Diese Kompetenzen lassen sich den Clustern »Interaction, Empathy and Emotion«, »Technical Society, Culture and Ethics«, »Communication« und »Creative Coding« zuordnen.

## Handlungsfeld »Creating Impact« (CI)

Absolvent\*innen entwerfen, entwickeln und evaluieren digitale Produkte und Applikationen, die auf soziale, kulturelle und ökonomische Phänomene und Entwicklungen reagieren und einwirken. Sie erkennen gesellschaftliche und wirtschaftliche Strömungen, können diese bewerten und daraus Potenziale für digitale Produkte und soziale Innovationen ableiten. Zu den Tätigkeiten in diesem Gebiet gehören:

- das Entwickeln und Bewerten von Ideen für neuartige Produkte und Prozesse,
- das Reflektieren dieser Produkte und Prozesse in ihrer gesellschaftlichen Rolle (Technikfolgenabschätzung, STS: Science Technology and Society),
- das Erproben von Zwischenständen mit realen Nutzern (Prototyping),
- das Entwickeln von Lösungen unter Einbeziehung dieser Nutzer (partizipative Strategien),
- die Nutzbarmachung dieser Lösungen als kommerzielles Produkt, Dienstleistung, freie Software oder kollaborative Anstrengung einer Community,
- das Entwickeln von Geschäfts- und Betreibermodellen, also betriebswirtschaftlichen Konzepten zum Aufbau von Unternehmen, Communities und anderer Strukturen für ein Produkt oder eine Dienstleistung,

- 
- Konzeption, Design und Entwicklung von Kommunikation und Produkten aus den Perspektiven relevanter Stakeholder (wie bspw. User\*innen, Kund\*innen, Designer\*innen, Coder\*innen, Bürger\*innen, Staat, Börse),
  - das Hervorbringen sozialer Innovationen auf Basis vernetzter digitaler Applikationen,
  - das Gründen von Start-ups und alternativer Strukturen (non-profit, NGO) zum Erproben von Geschäftsideen,

Für das berufliche Handlungsfeld »Creating Impact« entwickeln die Studierenden im Studiengang Kompetenzen, um relevante Produkte und Applikation zu entwickeln, ihre Ergebnisse nachhaltig Dritten zur Verfügung zu stellen und diese in ökonomischen und kulturellen Kontexten wirksam werden zu lassen. Daher steht dieses Handlungsfeld mit den Kompetenzclustern »Product Dimensions«, »Technical Society, Culture and Ethics« und »Entrepreneurship and Social Innovation« in Beziehung.

---

# Kompetenzcluster

## Kompetenzcluster »Software Development and Architecture« (SDA)

Die Absolvent\*innen können große Software-Systeme professionell designen, kontinuierlich weiterentwickeln und neue Versionen zügig produktiv setzen, indem Sie

- Grundprinzipien des »Computational Thinking« kennen und anwenden,
- Systeme mit fortschrittlichen Mitteln gängiger Programmiersprachen, Libraries, Frameworks und Tools und Beachtung akzeptierter Richtlinien und Best Practices codieren,
- Techniken zur Realisierung webbasierter, verteilter und mobiler Systeme anwenden, die dem Stand der Technik bzgl. Usability, Sicherheit, Robustheit, Skalierbarkeit etc. entsprechen,
- Tools zur Automatisierung von Software-Builds, -Qualitätssicherung und -Deployments einsetzen,

um später qualitativ hochwertige digitale Produkte realisieren und diese ständig verbessern zu können.

## Kompetenzcluster »Dual Reality and Smart Spaces« (DRSS)

Die Absolvent\*innen können dinglich-digitale und smarte Produkte entwickeln, indem sie

- die dingliche Welt durch Sensoren erfassen und durch Aktoren beeinflussen,
- das Eigenverhalten und die »Programmierbarkeit« von Materialien ausnutzen,
- Sensoren und Aktoren mit Microcomputern steuern und der Cloud vernetzen,
- die Daten der Sensoren durch Methoden, Verfahren und Algorithmen der künstlichen Intelligenz verarbeiten und Aktoren entsprechend steuern,
- informationsverarbeitende Computer physikalisch und kognitiv in Produkte einbetten,
- digitale Produktionstechniken für die Herstellung von Prototypen anwenden,

um später smarte innovative Produkte mit einer ganzheitlichen Perspektive auf Software und Hardware zu entwickeln und technisch handlungsfähig zu sein.

## Kompetenzcluster »Interaction, Empathy and Emotion« (IEE)

Die Absolvent\*innen können Geschichten erzählen und diese emotionalen Werte aktiv gestalten, indem sie

- Theorien, Thesen, Mechanismen, Muster, Materialien und Positionen, die das Themenfeld Interaction, Empathy and Emotion ausmachen, kennenlernen,
- inter- & transdisziplinäre Konzepte zu Verhaltensmechanismen im Feld des Interaction Designs und der Interactive Environments untersuchen (Case Studies) und sich kritisch mit diesen auseinandersetzen,
- Methoden wie Design Fiction, Fiction as a Method, Storytelling und Telling in Time kennen und für Fiktionen und Narrationen anwenden
- durch Kenntnisse im Bereich Body and Embodiment eigene emotionale Interaktion durch körperliche Bewegung, darstellende Künste (Tanz, Schauspiel), Form und Materialstudien (Soft & Hard) explorieren,

- dabei Raum- & Zeitkonzepte kennenlernen und in die Gestaltung mit einfließen lassen (Animationen / Conversational Interfaces / Audio-Visualität etc.),
- Sensationen und Phänomene verbalisieren und in Materialität und Interaktionen übersetzen,
- dabei das komplexe Zusammenspiel von Wahrnehmung, Handlung und Kontext in der Interaktion und dessen Wirkung auf Ästhetik und Bedeutung digitaler Produkte berücksichtigen,
- Charakterstudien / Kriterien und formästhetische Aspekte kennenlernen,
- Prototyping als Skizzenbuch praktizieren (in Soft- & Hardware),
- Experimente mit Material hinterfragen,
- die Klaviatur der Dimensionen von Wahrnehmung, Vorstellung und Darstellung beherrschen,
- gezielte Planung und Gestaltung von Verhaltensweisen konzipieren lernen,
- in der Lage sind, zukünftige Nutzende als gleichberechtigte Partner in den Designprozess einzubeziehen und
- sich mit sozio-affektiven und sozio-kognitive Prozessen befassen / sozio-affektive als auch sozio-kognitive Funktionen als Bedingung für prosoziales Verhalten erkennen und anwenden / adaptives Sozialverhalten als ein Ergebnis des dynamischen Zusammenspiels von sozio-affektiven und sozio-kognitiven Prozessen verstehen

um Dinge, Ideen, Produkte und Geschäftsmodelle zu beseelen und soziale Innovation im Rahmen von gesellschaftlichen Kontexten mit neuen Kommunikations- und Interaktionsformen nachhaltig zu verbessern.

## Kompetenzcluster »Product Dimensions« (PROD)

Die Absolventen\*innen können einem Produkt und ihrem Ecosystem auf den Grund gehen, indem sie

- Denkweisen, Herangehensweisen, Wirkungsweisen eines Produktes oder einer Produktwelt verstehen und verändern,
- neue Dimensionen des Produktbegriffs kennen und deklinieren,
- den sozio-kulturellen Kontext der Produktentwicklung in Betracht ziehen,
- den Product-Launch in eine Zukunft abstrahieren können,
- unterschiedliche Produktebenen multiperspektivisch interpretieren, planen und gestalten,
- Sustainability, Durability und Recycling verstehen und planen,
- soziale Verantwortung als Kern jeder Produktentwicklung ansehen,
- und die Wichtigkeit dieses Umdenkens als Grundstein für soziale Innovation verstehen,

um einen Unterschied in der Zukunft von Geschäftsmodellen, unserer Gesellschaft und unserer Umwelt zu machen.

## Kompetenzcluster »Agile Methoden« (AGILE)

Die Absolvent\*innen können Prozesse zur Entwicklung innovativer digitaler Produkte in heterogenen Teams agil managen, indem sie

- agile inkrementell-iterative Vorgehensmodelle anwenden,
- Kunden in den Entwicklungsprozess durch Methoden des Design Thinking, Co-Creation und Participatory Design einbeziehen,
- Kreativtechniken zur Moderation von Gruppen zur Ideenfindung und Variantenbildung nutzen,
- in selbstorganisierten Teams mitarbeiten,
- und verschiedene Fragestellungen in Projekten durch Prototypen beantworten,

---

um Projekte mit visionären Zielen aber unklaren und vagen Anforderungen zu managen und dabei Erkenntnisse, die während des Projekts gewonnen werden, in den Entwicklungsprozess zu integrieren.

## Kompetenzcluster »Communication« (COMM)

Die Absolventen\*innen können miteinander und über ihr Tun versiert sprechen und schreiben. Sie haben ein tieferes Verständnis des Themenkomplexes der Kommunikation im Rahmen von Mensch-Maschine Interaktion. Sie können soziale Innovationen vermittelt, indem sie

- Komplexes einfach machen,
- mit entsprechendem Vokabular ausgestattet werden und dieses erlernen und anwenden,
- den Paradigmenwechsel der Terminologien und Infrastrukturen innerhalb der Handlungsfelder kritisch hinterfragen, reflektieren und mitgestalten,
- Marketingprozesse und Mechanismen verstehen, und optimieren,
- Designprozesse in Form von Kommunikation durch verschiedene (neue) Medienformate befördern
- ein sinnloses von einem sinnvollem Produkt / Prozess / Geschäftsmodell unterscheiden,
- Interdisziplinarität praktisch entdecken, ausleben und im Anschluss dokumentieren,
- soziale Kompetenzen im Sinne einer verbesserten Umwelt und Kultur ausbauen
- und Lernen das Richtige zu pitchen

um für die Notwendigkeit einer verstärkten Zusammenarbeit in der nachhaltigen Welt von morgen und am »Arbeitsplatz der Zukunft« insbesondere zwischen Kreativ-, Wirtschafts- und Technologieabteilungen gewappnet zu sein und diese Kommunikationsprozesse und -terminologien mitzuerschaffen und auszugestalten.

## Kompetenzcluster »Technical Society, Culture and Ethics« (TSCE)

Die Absolvent\*innen können ihr Handeln im Kontext einer digitalisierten, vernetzten und demokratischen Gesellschaft, Kultur und Wirtschaft nachhaltig gestalten und reflektieren, indem sie

- den Wert offener Software-Entwicklungen (Open Source) kennenlernen und durch eigene offene Projekte neue Werte schaffen,
- die Bedeutung des Open Web für demokratische Gesellschaften einschätzen und seine Bedrohung durch proprietäre Entwicklungen exemplarisch analysieren,
- soziale und kulturelle Entwicklungen im Kontext der Digitalisierung untersuchen und insbesondere auch Gefährdungen, wie den Wegfall sicherer Arbeitsverhältnisse, analysieren,
- Übersichtswissen zu wichtigen juristischen Fragestellungen im Zusammenhang mit der Erstellung digitaler Systeme (z.B. Einbindung fremder Bestandteile unter GNU-Lizenz) und ihrem Betrieb (z.B. Haftungsfragen bei Störungen) erwerben und exemplarisch auf eigene Projekte anwenden,
- ethische Fragestellungen im Zusammenhang mit neuen digitalen Entwicklungen und Produkten aufwerfen, im Lichte von bekannten Problemfällen diskutieren und Alternativen abwägen,
- durch ihr Verständnis für soziokulturelle Phänomene Zusammenhänge, Gefahren und Chancen über Fachgrenzen hinweg erkennen und bewerten,

um als soziale Innovatoren gesellschaftliche Potenziale für digitale Produkte und Services zu erkennen und diese als kritische Akteure nachhaltig zu gestalten. Sie werden damit ihrer Verantwortung als »Global Citizen« in diversen sozio-kulturellen Kontexten gerecht werden.

---

## Kompetenzcluster »Creative Coding« (CREA)

Die Absolvent\*innen haben eine kreative Arbeitsweise und können auch durch unkonventionellem Prozesse abseits der herkömmlichen Wege zu Lösungen gelangen. Sie verstehen Coding als kreative Disziplin um Klänge, Bilder, Animationen und synästhetische, oftmals echtzeitgenerierte Interaktionen zu erzeugen. Absolvent\*innen können sich kreativ mit Code ausdrücken indem sie

- künstlerische Methoden, Künstler\*innen und ihre Werke / Positionen kennenlernen,
- die Historie, Einflüsse und Strömungen von Open Source und Maker-Bewegungen kennen und daran anknüpfen,
- an offenen und kreativen Communities partizipieren und eigene Beiträge leisten,
- die Grenzen des technisch Machbaren challengen und durch unkonventionelle Herangehensweisen lösen oder umgehen,
- bestehende Tools und Werkzeuge kennen und als Baukasten nutzen und einem »Do it Yourself«-Ansatz folgend sich eigene Werkzeuge entwickeln wo nötig,
- bewusst unkonventionelle Herangehensweisen und Artefakte in geänderten Kontexten einsetzen,
- Code als Mittel zur Realisierung von Funktionalität aber auch zum Ausdruck von Ästhetik nutzen,
- ästhetische Gesichtspunkte analysieren und einschätzen,
- die Kopie vom Original unterscheiden,
- Medien flexibel ineinander verschachteln,
- alle Sinne miteinbeziehen,
- Werke audio-visueller Kunst und Design erstellen, wie z.B. Live-Visuals, Sound Art, Maschinen-Installationen, Multiscreen-Projektionen, Synthesizer oder Roboter,
- kommerzielle Medieninszenierungen und Produktinstallationen entwickeln,
- und Produktprototypen simulieren,

um eine eigene Formsprache zu entwickeln und als neugierige Gestalter\*innen kreative Lösung für herausfordernde Fragestellungen zu finden.

## Kompetenzcluster »Entrepreneurship and Social Innovation« (ENSI)

Die Absolvent\*innen können Geschäftsmodelle für digitale Produkte entwickeln, bewerten und in frühen Phasen unternehmerisch umsetzen, indem sie

- zukunftsrelevante Szenarien darstellen und spekulativ die Bedürfnisse der Nutzer\*innen von morgen wahrnehmen,
- wirtschaftliche und gesellschaftliche Potenziale für Produkte über Fachgrenzen hinweg erkennen und bewerten,
- Betriebs- und Geschäftsmodelle konzipieren und bewerten,
- Produkte oder Dienstleistungen als Minimum Viable Product früh an den Markt bringen und dort evaluieren,
- betriebswirtschaftliche Grundlagen zur Gründung und für den Betrieb von StartUps anwenden,
- ökonomisch überzeugend und informationsbelegt präsentieren,

um ihre eigenen Innovationen nachhaltig in die Welt zu bringen und als soziale Innovatoren zu wirken.

---

## Kompetenzcluster »Soziale Kompetenz und Selbstkompetenz« (SOSE)

Die Absolvent\*innen sind in der Lage sich selbstständig neue Methoden und Wissen anzueignen und zu erkennen, welches Wissen für sie relevant ist, indem sie wissenschaftlich, analytisch und reflektiert arbeiten. Sie kennen ihre berufliche Rolle in zivilgesellschaftlicher, politischer und kultureller Hinsicht, sowie die damit verbundenen Erwartungen und ggf. vorhandene Rollenkonflikte. Sie können zur Konfliktlösung beitragen, indem sie Kenntnisse zum Konfliktmanagement wirksam machen, in kontroversen Diskussionen zielorientiert argumentieren und mit Kritik sachlich umzugehen.