OST Ostschweizer Fachhochschule

MSE | MASTER OF SCIENCE IN ENGINEERING

**Modulbeschrieb**

# Advanced Software Architecture

## Allgemeine Angaben

### Modulbezeichnung

**Advanced Software Architecture**

### Modulkategorie

Fachliche Vertiefung

### Anzahl der Credits

3

### Modulverantwortliche/r

olaf.zimmermann@ost.ch

### Durchführungssetting

| | | | |
|---|---|---|---|
| Campus | ☐ Buchs | x Rapperswil-Jona | ☐ St. Gallen |
| Online Teilnahme | x keine Onlineteilnahme möglich | ☐ hybrid | ☐ ausschliesslich online |
| Durchführung | ☐ wöchentlich | ☐ als Blockwoche | x nach Absprache |

## Ziele, Inhalt und Methoden

### Learning objectives

Having attended this supplemental course, students:

- Are able to compare and assess emerging architecture-centric practices and patterns (e.g., in the form of SWOT analyses)
- Are able to apply lightweight approaches to architecture design and documentation (e.g., capturing architectural decisions)
- Can engineer method content, write up practices and patterns according to standard templates (e.g., EuroPLoP and DPR)
- Can report research problems, directions and results in written and oral form (e.g., tool design challenges)
- Are able to conduct research independently and take responsibility for managing the study process

### Module content

One or more advanced and/or emerging topics from the software architecture domain such as:

- Agile architecting practices such as agile modeling and C4, including comparison with other notations and viewpoint models
- Architectural Decision Records (ADRs) in support of documentation-as-code
- Strategic domain-driven design for module and service decomposition; coupling and granularity concerns
- Cloud-native application architectures (CNAs) and serverless cloud computing
- Event-driven architectures and messaging as examples of emerging and mature integration styles
- Architectural refactoring and interface refactoring
- Ethics as a design concern

Each participant will study a single topic that fits the term and master thesis topic(s), starting with an article or book chapter.

### Teaching and learning methods

Participants are expected to present their topic and their analysis results in writing and in a final presentation:

- Guided self-study (dt. begleitetes Selbststudium)
- Analysis or creation of an empirical case study
- Incremental feedback on emerging practice descriptions and/or scientific essay, small writer's workshops

The exact deliverables and grading scheme will be defined at the start of this supplemental course.

### Prerequisite skills and experience (competencies)

Centrale module «Software Engineering and architecture» or equivalent knowledge and skills; practical experience as software developer or junior architect welcome, but not required

### Bibliography

Books:

- N. Ford et al, “Software Architecture: The Hard Parts”, O’Reilly.
- M. Keeling, “Design It!”, Pragmatic Programmers
- E. Woods et al, “Continuous Software Architecture in Practice”, Addison Wesley

Articles:

- G. Hohpe et al, “The Role of the Software Architect in the Digital Age, IEEE Software
- I. Ozkaya, “Ethics as a Design Concern”, IEEE Software
- C. Pautasso et al, “The Web as a Software Connector”, IEEE Software

Online resources:

- O. Zimmermann and M. Stocker, “Design Practice Reference”, GitHub repository and pages

## Leistungsbewertung

**Bewertet werden die abgegebenen Berichte und die Präsentation des Abschlussberichts.**

Prüfungsdauer

**30 Minuten**

Bewertungskriterien

Planung, Durchführung und Ergebnis der Seminararbeit.

Inhalt und Form der abgegebenen Berichte (Tipps: siehe z.B. https://ozimmer.ch/categories/#Authoring).

Präsentation, insb. Präsentationsunterlagen und Eingehen auf Fragen.

**Aktive Mitarbeit in Vorträgen und Diskussionen zu den Seminararbeiten der anderen Teilnehmenden.**

Bewertet werden die abgegebenen Berichte und die Präsentation des Abschlussberichts.

**Prüfungsdauer**

30 Minuten