



Ruhr Master School
of Applied Sciences

Dieses Wahlpflichtmodul ist ein Angebot der:

**Fachhochschule
Dortmund**

University of Applied Sciences and Arts

Master Digital Transformation

Software Architectures

daniel.vonfalkenhayn@fh-dortmund.de

Prof. Dr. Sabine Sachweh
sabine.sachweh@fh-dortmund.de

Hochschule Bochum
Bochum University
of Applied Sciences



Fachhochschule
Dortmund
University of Applied Sciences and Arts



Westfälische
Hochschule
Gelsenkirchen Bocholt Recklinghausen
University of Applied Sciences

STIFTUNG
MERCATOR



Software Architectures (MOD1-02)					
Code Number	Workload	Credits	Semester	Frequency	Duration
48020/21	180 h	6	1	winter semester	1 Semester
1	Course Title Software Architectures		Contact hours 4 SWS / 60 h	Self-Study 120 h	Planned Group Size 25 students
2	<p>Course Description</p> <p>In recent years' new architectural styles have emerged to cope with the increasing need of highly scalable and distributed systems. Among them are Microservices and Self-Contained Systems. The resulting systems are characterized by being componentized into independent services which communicate using well-defined interfaces.</p> <p>This course the students learn about modern software architecture paradigms, both conceptually and practically. Additionally, subjects related to the operation of such systems are covered, such as infrastructure-technologies and particular challenges of operation like scaling or load balancing.</p> <p>In addition to the lectures, the students have the opportunity to apply their knowledge in project-based group activities.</p>				
3	<p>Course Structure</p> <ul style="list-style-type: none"> • Historical development of software architecture paradigms. • Characteristics of modern architectural styles. • Designing Microservices and Self-Contained Services. • Developing Microservices and Self-Contained Services. • Infrastructure, deployment, and operation: Methods, technologies and challenges. 				
4	<p>Application Focus</p> <p>To complement the lectures and enable the students to apply the covered topics, there will be a project. Their students will work in small groups to participate in the design and development of a significant application.</p>				
5	<p>Scientific Focus</p> <p>Written assignment: literature review in the style of a scientific paper up to 10 pages Performing a survey based on relevant scientific methods</p>				
6	<p>Parameters</p> <ul style="list-style-type: none"> • ECTS: 6 • Hours of study in total: 180 • Weekly hours per semester: 4 <ul style="list-style-type: none"> - Contact hours: 60 - Self-Study hours: 120 • Course characteristics: compulsory • Course frequency: every year – winter semester 				

	<ul style="list-style-type: none"> • Maximal capacity: 25 students • Course admittance prerequisites: none • Skills trained in this course: theoretical knowledge, practical skills and scientific competencies • Assessment of the course: Theoretical knowledge (40%): Theoretical knowledge (40%): Written Exam at the end of the course, Practical Skills (40%): Individual programming task, realizing a small real-world project within the lecture related topics of software architectures and Scientific Competences (20%): written paper (literature review, approx. 10 pages) and presentation (in class or at a student conference, e.g. International Research Conference Dortmund) • Teaching staff: Prof. Dr. Sabine Sachweh, external lecturers from industry and/or partner universities, PhD students from IDiAL
7	<p>Learning outcomes</p> <p>7.1 Knowledge</p> <ul style="list-style-type: none"> • Knows the concepts and structure of modern software architecture styles • Knows technologies and tools related to the operation of modern software architectures • Knows particular challenges of operating distributed systems • Knows how to analyze an application by different metrics • Knows to maintain and operate a distributed system • Knows how to distributed a system based on workload of particular components <p>7.2 Skills</p> <ul style="list-style-type: none"> • Can critically evaluate the suitability of an architectural style given a particular problem • Can design, develop and operate leveraging the lecture topics • Can assess and improve an existing software architecture • Can analyze a distributed system by different application metrics • Can distribute a container-based system by workload <p>7.3 Competence – attitude</p> <ul style="list-style-type: none"> • Can discuss and assess the differences between various architectural styles • Can communicate and explain architectural decisions • Can work in a team on scientific topics • Can demonstrate and discuss results in a group
8	<p>Teaching and training methods</p> <ul style="list-style-type: none"> • Theoretical knowledge: e-learning modules on software architecture models, tool tutorials • Practical Skills: Projects, Labs & Exercises, small project • Scientific Competences: student research group on SW Architectures
9	<p>Course mapping</p> <p>Input for:</p> <p>MOD2-02 – Software-intensive Solutions</p> <p>MOD-E01 – Software Engineering Project</p> <p>Input from:</p> <p>None</p>
10	<p>References</p> <p><u>Basics & Practitioner</u></p>

Newman, S.; Building Microservices, O'Reilly Media, 2016

Newman, S.; Monolith to Microservices: Evolutionary Patterns to Transform Your Monolith, O'Reilly Media, 2019

Wolff, E.; Microservices: Flexible Software Architecture, Addison-Wesley, 2016

Chris Richardson, Microservices Patterns: With examples in Java, Manning Publications, 2018

Research (Journals, Conferences & selected papers)

Microservices Conference, <https://www.conf-micro.services/> , e.g. 2019, FH Dortmund

IEEE Software

IEEE Transactions on Software Engineering

F. Rademacher, J. Sorgalla and S. Sachweh, "Challenges of Domain-Driven Microservice Design: A Model-Driven Perspective," in IEEE Software, vol. 35, no. 3, pp. 36-43, May/June 2018

L. De Lauretis, "From Monolithic Architecture to Microservices Architecture," 2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), Berlin, Germany, 2019