

Course

EKS - Development of Complex Software Systems

Version: 1 | Last Change: 03.09.2019 11:28 | Draft: 0 | Status: vom verantwortlichen Dozent freigegeben

^ General information

Long name	Development of Complex Software Systems
Approving CModule	EKS_BaTIN
Responsible	Prof. Dr. Hans Nissen Professor Fakultät IME
Level	Bachelor
Semester in the year	winter semester
Duration	Semester
Hours in self-study	60
ECTS	5
Professors	Prof. Dr. Hans Nissen Professor Fakultät IME
Requirements	Specification and modeling of systems and software with UML, modularization in Java, simple design patterns, basic software testing methode, various architectures of systems and software, basic quality assurance concepts, version management skills, very good practical and theoretical knowledge of the programming language Java
Language	German
Separate final exam	Yes

Final exam

Details

oral exam, with many students written exam

The oral exam or written exam ensures that each student also individually fulfills the goals of the Learning Outcome has reached, through tasks of the following types:

questions about basic knowledge of design principles, architectural concepts, test procedures, application of design patterns to given problem cases, design or extension of a modularized system architecture with supporting specified non-functional properties, creation of suitable logical test specifications and concrete test cases.

Minimum standard

At least 50% of the total number of points.

Exam Type

oral exam, with many students written exam

The oral exam or written exam ensures that each student also individually fulfills the goals of the Learning Outcome has reached, through tasks of the following types:

questions about basic knowledge of design principles, architectural concepts, test procedures, application of design patterns to given problem cases, design or extension of a modularized system architecture with supporting specified non-functional properties, creation of suitable logical test specifications and concrete test cases.

^ Lecture / Exercises

Learning goals

Knowledge

design patterns

modularization

professional code development

advanced Java concepts

Module-oriented architectural principles

complex test procedures

Skills

apply and evaluate design patterns

Apply and evaluate approaches to professional code development

Apply automated code analysis methods and interpret the results

Design and implement modularized architectures

use complex test procedures

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Lecture	2
Exercises (whole course)	1
Exercises (shared course)	1
Tutorial (voluntary)	0

Separate exam

none

^ Practical training

Learning goals

Skills

implementation of design pattern

Create modularized architectures for large-scale applications

apply automated code review and static code analysis

Select test method and apply to programs

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Practical training	1
Tutorial (voluntary)	0

Separate exam

Exam Type

working on practical scenarion (e.g. in a lab)

Details

Students work in small teams.

Each group completes several internship sessions with assigned laboratory appointments.

In each session, programming tasks are solved.

To prepare for a lab appointment, a homework sheet has to be solved.

The developed solutions must be submitted by the students before the lab date and

explained and defended during the appointment to the supervisor (K.16).

If this test is not passed, then a repetition task must

be edited and presented until a follow-up appointment;

in the case of recurrence, this leads to failure of the internship.

In addition, during the laboratory appointment, an attendance sheet with further tasks has to be solved under supervision (and possibly with assistance) in a controlled environment.

Minimum standard

Successful participation in all laboratory appointments, i.e. in particular independent solution (or with some assistance if necessary) of the assignments.