D U A L E H O C H S C H U L E
Baden-Württemberg
Studienbereich Technik

Software Engineering I (T4INF2003)

Software Engineering I (isced-f/0613)

Formal Details of the Module				
moduleno.	**location in course of study**	**duration**	**responsibility**	**language**
T4INF2003	2. Year	2 Semester	Prof. Till Hänisch	Deutsch/Englisch

Type of module
Core module

Teaching methods	
Teaching forms	**Teaching methods**
lecture, tutorial, lab work	Lecture, discussion, group work

Forms of examination		
Examination form	**Exam duration (in minutes)**	**grading**
program design		yes

Workload and ECTS credit points			
total workload (in hours)	**of which in classroom**	**Of which self study**	**ECTS**
270	96	174	9

Qualification goals and competences
professional competentce
Students know the basics of the software development process. They know the methods of the respective project phases and can apply them.
methodical competences
The students are able to analyze a given problem statement. They can select and apply appropriate methods for specific problems. They can design and implement a computer-based solution. They can make corrective adjustments to solution proposals. They can use tools for collaboration and problem-solving.
personal and social competences
The students can competitively evaluate solution proposals for a given problem and justify their designs and solutions. They can competitively assess, select, and critically reflect upon solution proposals for a given problem. The students can engage with domain experts in discussions about problem analyses and solution proposals, as well as about the interconnections of individual phases. They can orally and in writing present their designs and solutions.

During the discussion, they can critically engage with various perspectives. They can build and further develop teams.
interdisciplinary competence
The students can integrate interdisciplinary skills, such as combining the software development process with project management techniques and considering time and cost factors during the project. They can independently familiarize themselves with tools. They can recognize their own strengths and weaknesses in the project and strive for improvement. They can handle conflicts and resolve them constructively. They can pass on and support skills. They can provide each other with constructive feedback. They can effectively collaborate within a team in complex projects.

Learning Units and Contents		
Teaching and learning units	**classroom**	**self study**
Basics of Software-Engineering	**96**	**174**
Alternative:		
Specification	**32**	**58**

- Information and Communication Technologies (ICT)
- Project Management Methods
- Phases of Software Engineering and their Interconnections
- Requirements Engineering and Use Cases
- Analysis and Design Models (e.g., Modeling Techniques like UML or SADT)
- Different types of documentation are addressed in phase-specific manner

Skills and competences in this unit:

- The students are familiar with the fundamentals of the software development process.
- They can analyze a given problem statement.
- They can use tools for collaboration and problem-solving.
- The students can competitively evaluate solution proposals for a given problem and justify their designs and solutions.
- They can competitively assess, select, and critically reflect upon solution proposals for a given problem.
- The students can engage with domain experts in discussions about problem analyses and solution proposals, as well as about the interconnections of individual phases.
- They can present their designs and solutions verbally and in writing.
- In discussions, they can critically engage with various perspectives.
- They can build and further develop teams.
- They can independently familiarize themselves with tools.
- They can recognize their own strengths and weaknesses in the project and strive for improvement.
- They can handle conflicts and resolve them constructively.
- They can pass on and support skills.
- They can provide each other with constructive feedback.
- They can effectively collaborate within a team in complex projects.

Design	**32**	**58**

As in specification, but additionally:

- Requirements Management
- Software Architectures, Interface Design, Software Design, and Design Patterns
- Version Control

- Incorporation of Existing Software Libraries
- Software Development Environments

Additional skills and competences in this unit:

- They can design and implement a computer-based solution.
- They can make corrective adjustments to solution proposals.
- The students can integrate interdisciplinary skills, such as combining the software development process with project management techniques and considering time and cost factors during the project.

Implementation	**32**	**58**

As in design, but additionally:

- Coding guidelines and code quality and reviewing
- Testing levels, planning, and assessment
- Continuous Integration
- Operation and Maintenance

Additional skills and competences in this unit:

- They are familiar with the methods and supporting technologies of the respective project phases.
- They can select and apply appropriate methods for specific problems.

Specifics

The individual contents of the course are to be deepened by means of a project. In the individual project phases, the use of suitable methods, documentation and quality assurance should be dealt with. Suitable tools are to be used. In the group-oriented laboratory exercises, extracurricular qualifications are practised and (partial) results are presented. This module also includes up to 24 hours of guided self-study in the form of practice hours, labs or projects. Here the students work on exercises and/or in-depth assignments.

Prerequisites

-

Literature

- Helmut Balzert: Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb, Spektrum akademischer Verlag
- Helmut Balzert: Lehrbuch der Softwaretechnik: Softwaremanagement, Spektrum akademischer Verlag
- Ian Sommerville: Software Engineering, Pearson Studium
- Chris Rupp: Requirements-Engineering und -Management: Aus der Praxis von klassisch bis agil, Carl Hanser Verlag GmbH & Co. KG