

Module description: Software Project 3

Module Code	t.BA.IT.PM3.19HS
ECTS Credits	4
Language of Instruction/Examination	German
Organizational Unit	InIT
Module Coordinator	Kurt Bleisch
Legal Framework	The module description is part of the legal basis in addition to the general academic regulations. It is binding. During the first week of the semester a written and communicated supplement can specify the module description in more detail.
Module Characteristic	Type 4* 4 lab lessons per semester week and half-class
Module Description	Students develop a fairly large object-oriented software application. In groups, they implement a self-selected project idea from the vision through to an executable application. A predefined iterative-incremental software development process is applied (the basis of agile software development).

Module description: Software Project 3

Module Content

The students develop in teams of 5 (+/-1) persons a larger software application in this module. In addition to the application of analysis and design competencies (from SWEN1), the realization and project management competencies are also practiced and the social and personal competencies are strengthened.

The ability to professionally create and carefully test larger software systems is an essential part of the realization competence.

Project management competence is needed to design work processes and, in particular, to organise one's own work and that of others. For this purpose, the ability to work in a team and the constructive examination of concepts and proposed solutions are essential. Students have learned to develop solutions, even with limited resources, that meet generally accepted quality standards and are accepted by all participants.

The teams are drawn by lot, choose a project manager and jointly develop a project idea that they want to realise. The milestones and the development and project management artefacts to be created are predefined. Otherwise, the team organises itself.

The students use a predefined iterative-incremental software development process (basis of agile software development) with the following predefined milestones and artefacts.

M1: Project Outline

1 Iteration (2 weeks)

Evaluation: Stakeholder agreement on vision, project objectives and scope, agreed initial phase/iteration plan, initial risks identified.

Artefacts: Project outline (vision, competitor analysis, initial requirements, profitability), project management (phase/iteration plan, iteration plan next iteration, risk list).

M2: Solution Architecture

3 Iterations (4 - 6 weeks)

Evaluation: Vision, goals and requirements clearly documented and stable (80%), documented and verified software architecture (with code) that meets the requirements.

Artefacts: Use case model, domain model, non-functional requirements, software architecture, selected designs, code, project management (phase/iteration plan, iteration plans, iteration assessments, risk list)

M3: Prototype

2 Iterations (4 weeks)

Evaluation: Feature-complete, prototype is stable

Artefacts: Application (software), technical report (summary of analysis, design and implementation), project management (phase/iteration plan, iteration plans, iteration assessments, risk list)

A software increment is implemented in each iteration (except the first iteration). The results of the milestones are presented in plenary sessions for each team. In between, status meetings are held per team with the lecturer for each iteration.

Throughout the project, the students use version management (Git) for the code as well as build and test tools (JUnit, Mockito, Gradle, or similar).

Module description: Software Project 3

Prerequisite Knowledge	-				
Learning Objectives (Competences)	Students...	Competencies	Taxonomies		
	You can realize a larger object-oriented software application from the vision to the executable application with a given iterative-incremental software development process.	M, F	K3		
	You can incrementally implement, test and continuously improve an object-oriented design (refactoring).	M, F	K3, K4		
	You can carry out a technical research (competition analysis, technology) and record it in accordance with standards.	F, M	K2, K3		
	You can independently create the development and project management artifacts defined in the software development process.	M, F	K3		
	You can model, display, and communicate the created development artifacts using a standardized notation (such as UML).	F, M	K3		
	You can consistently use analysis, architecture and design patterns as well as appropriate frameworks and tools (for versioning, build, etc.).	M, F	K3		
	You can plan and execute the project iteratively and risk-driven as part of a team.	F, M	K3		
You can communicate technical content orally and in writing in a way that is appropriate for the addressee.	SO	K3			
Performance Assessment	End-of-module exam	Assessment	Length (min.)	Weighting	Form
	other	Grade		100	
	Performance assessment during the semester	Assessment	Length (min.)	Weighting	Form
	-	-	-	-	-
Classroom Attendance Requirement	None				
Learning material					
Comments					