HAW Kiel
Hochschule für Angewandte Wissenschaften Kiel
Kiel University of Applied Sciences

MI136 - Funktionale Programmierung

MI136 - Functional Programming

General information	
Module Code	MI136
Unique Identifier	
Module Leader(s)	Lee, Michael (michael.lee@haw-kiel.de)
Lecturer(s)	Lee, Michael (michael.lee@haw-kiel.de)
Offered in Semester	Wintersemester 2019/20
Module duration	1 Semester
Occurrence frequency	Irregular
Language	Englisch
Recommended for international students	Yes
Can be attended with different study programme	Yes

Curricular relevance (according to examination regulations)
Study Subject: M.Sc. - MIE - Information Engineering (PO 2022, V3) Study Specialization: Business IT-Management Module type: Wahlmodul Semester: 1, 2, 3
Study Subject: M.Sc. - MIE - Information Engineering (PO 2022, V3) Study Specialization: Information Technology and Systems Module type: Wahlmodul Semester: 1, 2, 3
Study Subject: M.Sc. - MIE - Information Engineering (PO 2022, V3) Study Specialization: Intelligent Systems Module type: Wahlmodul Semester: 1, 2, 3
Study Subject: M.Sc. - MIE - Information Engineering (PO 2022, V3) Study Specialization: IT Security Module type: Wahlmodul Semester: 1, 2, 3

Qualification outcome
Areas of Competence: Knowledge and Understanding; Use, application and generation of knowledge; Communication and cooperation; Scientific self-understanding / professionalism.
- principles of functional programming. Differences to other programming paradigms - differences to other programming paradigms - usage of the langauges elm and haskel - understanding of the lambda calculus - separation of side effects / handling of side effects in funtional languages
In a semster project students will - create web frontends using functional programming - handle side effects for the purpose of user interaction and backend communication - connect pure functional code with non pure code in the ECMAScript browser environment

Students will work in groups of 4 where they will - have to communicate and discuss and assign tasks - use source code management to handle simultaneously working on the same code base - work as a team on a project presentation which has to involve all parts of each group
Being focused on the product (semester project) students will tackle the same challenges they would face in a commercial software product project. As part of this students will have to without supervision - face tricky problems that require intense analysis and understanding - acquire knowledge required specific problems These responsibilities are essential aspects of the self-concept of any tech professional.

Content information	
Content	Introduction to functional programming. In this class students will learn modern functional programming in practice. - the elm language and runtime will be introduced to write modern web applications - Haskell will be introduced to learn some of its advanced concepts - theoretical background will be provided through the lambda calculus As a result of this class students will have written their own web app using functional methodology.
Literature	elm tutorial: https://www.elm-tutorial.org/en/ learning Haskell: http://learnyouahaskell.com/ lecture notes on the lambda calculus: https://www.mscs.dal.ca/~selinger/papers/lambdanotes.pdf

Teaching formats of the courses	
Teaching format	**SWS**
Projekt	2
Lehrvortrag	2

Workload	
Number of SWS	4 SWS
Credits	5,00 Credits
Contact hours	48 Hours
Self study	102 Hours

Module Examination	
Examination prerequisites according to exam regulations	None
MI136 - Projektbezogene Arbeiten	Method of Examination: Projektbezogene Arbeiten Weighting: 50% wird angerechnet gem. § 11 Absatz 2 PVO: No Graded: Yes

MI136 - Klausur	Method of Examination: Klausur Duration: 90 Minutes Weighting: 50% wird angerechnet gem. § 11 Absatz 2 PVO: No Graded: Yes

Miscellaneous	
Recommended Prerequisites	Basic knowledge of web development required. Students need to be able to - create their own HTML/CSS interfaces - write a basic REST or GraphQL backend - use a document or relational DB Knowledge of React/Redux helpful but not required.
Miscellaneous	- lab attendance is mandatory - projects will be done in groups of 4 (except for special circumstances) - there will be a written exam prior to the project presentations