

Advanced Design and Programming

Prof. Dr. Dirk Riehle

Friedrich-Alexander University Erlangen-Nürnberg

ADAP A01

Licensed under [CC BY 4.0 International](https://creativecommons.org/licenses/by/4.0/)

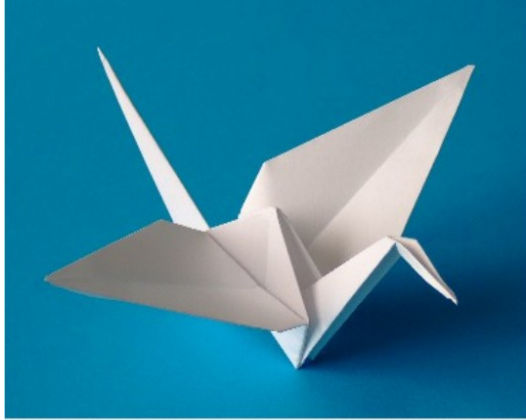
Course Learning Goals

- To acquire competence in ...
 - domain-independent
 - principles and practices of
 - advanced object-oriented
 - design and programming

Wahlzeit / Flowers Application

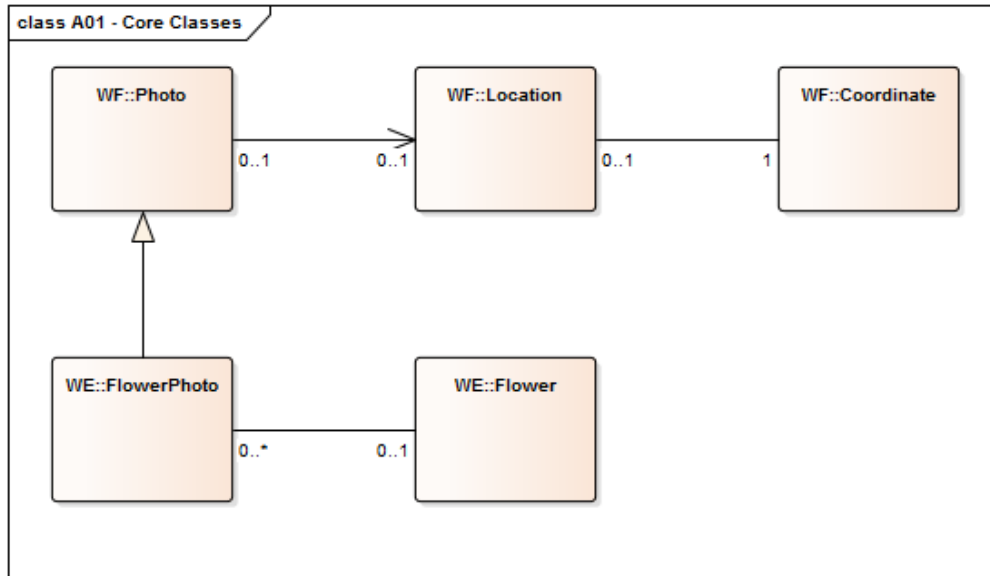


Student Choice of Project 1 / 2



Student Choice of Project 2 / 2

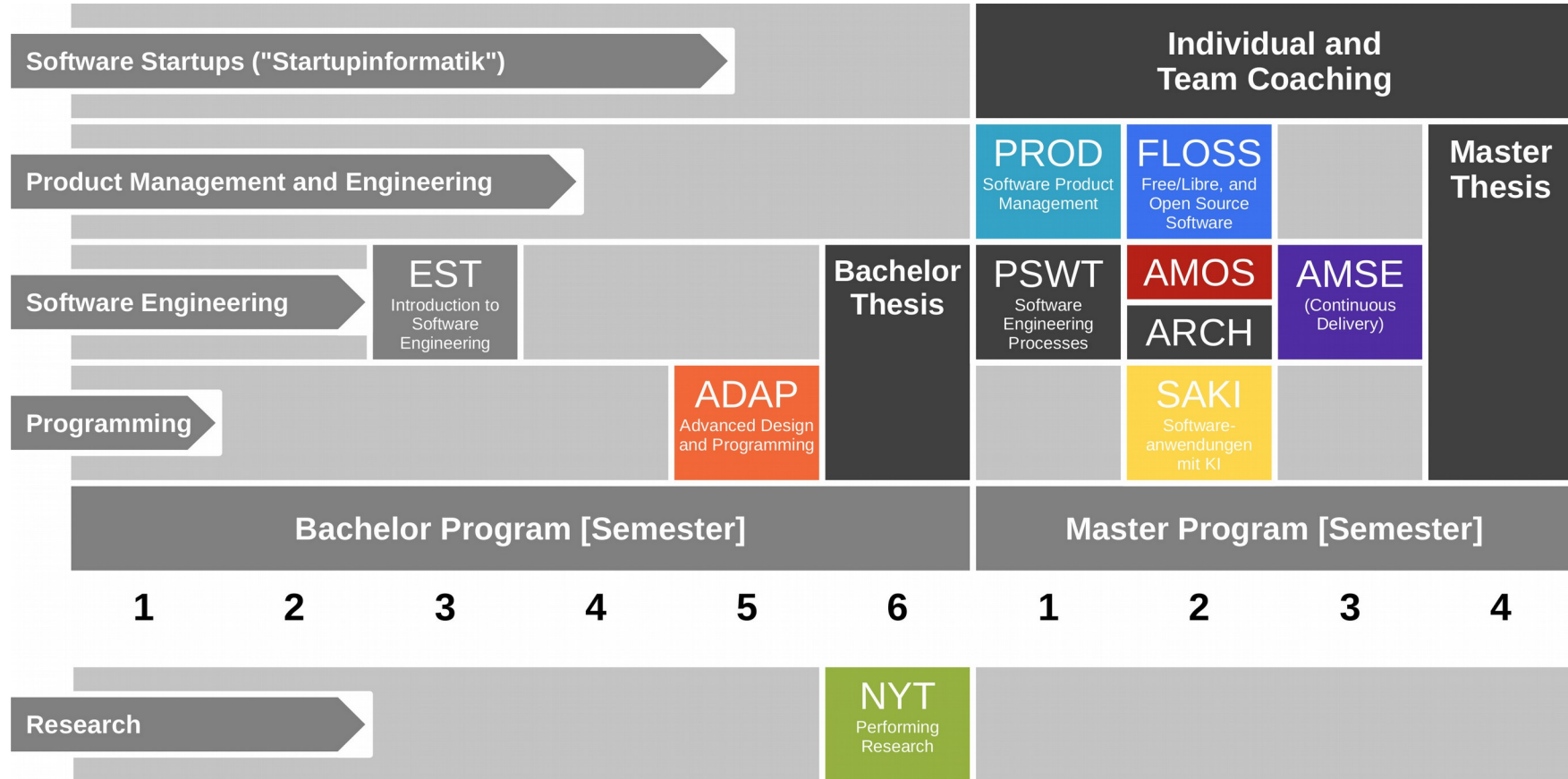
- Choose something that will provide you with domain classes
 - Food → Type of food, price, location
 - Landscapes → Location, season, camera
 - Postage stamps → Stamp, country, age
- You will need these domain classes for your homework



Skills Required for Course

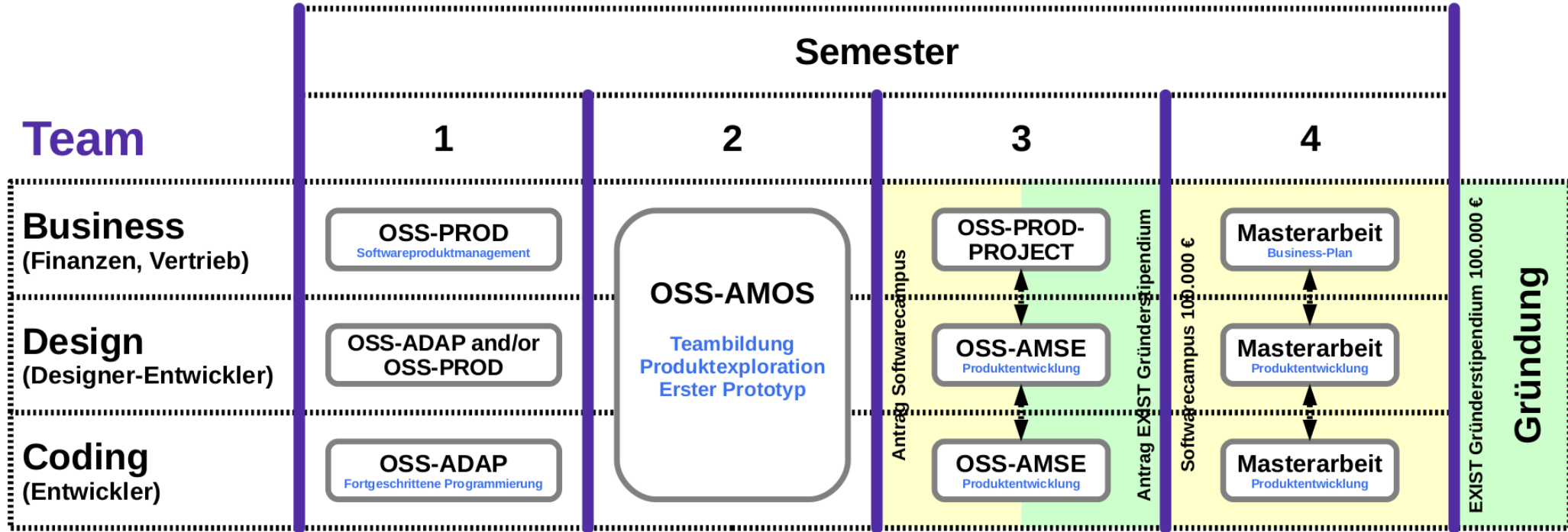
- Required prior student skills
 - Programming in Java
 - Working with git
- Familiar with at least one
 - Operating system (default: Linux)
 - Command line (default: bash)
 - Development environment (default: Eclipse)
- Expected of student
 - Willingness to learn by themselves

Course Position in Curriculum



Startupinformatik

Dein Weg zur Startup im Masterstudium, seit 2010!



<http://startupinformatik.de>

Gewinner 1ter Preis 2015
Programm zur Startup-
Förderung an Hochschulen



Courses and Modules

		Courses (Lehrveranstaltungen)		
		OSS-ADAP- VUE		Total ECTS
Modules	OSS-ADAP	x5		

Course Grading [1]

- Understanding of lecture content (4 SWS) = $60 / 150 = 40\%$
 - Graded using [0..10] each class session using StudOn quizzes
- Homework assignments = $90 / 150 = 60\%$
 - Graded using [0..10] each

Class Quizzes

- Each class session starts with a class quiz
 - A quiz will test your understanding of last session's lecture(s)
 - A quiz typically has 5 questions and will last 10 minutes
 - The overall quiz is graded using [0..10] scheme (10 points in total)
- A class quiz will open precisely when class starts
 - The quiz is administered automatically through StudOn
 - It is your job to have reliable Internet access etc.
 - There is no way to make up for a missed quiz
- Please see the StudOn quiz FAQ <https://bit.ly/2JpgjnQ>

Homework Grading

Categories	Points					Criteria
	Disagree	Disagree some	Neutral	Agree some	Agree	
Form (10%)	Does the deliverable meet formal requirements?					Does the deliverable meet page length requirements, has the right language, has required header, etc.?
Language (10%)	Is the language clear, concise, and helpful?					Are sentences complete, is the grammar correct, are statements coherent, etc.?
Solution Design (30%)	Does the solution make sense and is it explained well?					Is the answer clear, is it explained why it was chosen, are pros and cons weighed, etc.
Solution Implementation (50%)	Has the solution been implemented well? (If no implementation was required, use your explanation points for the implementation as well.)					Is the code clean, follows coding guidelines, is well tested, matches explanation, etc.?

Requirement for Passing Course

- At least 50% participation in all grade-relevant components
 - Class quizzes
 - Homework submissions

Receiving a Grade for the Course

- If you want to receive a grade
 - You must register through the course management system
- If you cannot register through the course management system
 - Please follow these instructions: <https://wp.me/PDU66-2bx>
- Otherwise: No grade

No Oral or Written Exam [1]



Course Language [1]

- Class
 - Lecturer: German
 - Student: Choice of German or English
- Homework
 - Source code: English
 - Submissions: Choice of German or English

[1] See <http://osr.cs.fau.de/2012/03/10/english-or-german-deutsch-oder-englisch/>

Course Communication

- Announcements by email (through StudOn)
- Questions and answers using FSI Forum for course
- If you must send email, use the teaching team alias from course index



Open Source Governance

- Do not add reciprocally-licensed libraries to your project
 - May make later desired license change difficult, including proprietary use
 - Check with osr-adap@group.riehle.org whether library is OK
- Rules of thumb on license choice
 - **OK: Permissive licenses (BSD, MIT, Apache)**
 - **May be OK: Weakly protective (a.k.a. “weak copyleft”)**
 - **Usually not OK: Strongly protective (a.k.a. “reciprocal” or “copyleft”)**
 - **Never OK: Public domain or no license**
- Professionals (i.e. companies) use code scanners to check
 - Black Duck, Palamida, Protecode, ...

Project Contributor Agreement

- For contributions to Wahlzeit, we require a rights grant
 - We use the Sun Contributor Agreement
 - Available here: <http://osr.cs.fau.de/teaching/contributor-agreement/>

Course Information

- Course index
 - <https://goo.gl/H9Zcmy>

Thank you! Questions?

dirk.riehle@fau.de – <http://osr.cs.fau.de>

dirk@riehle.org – <http://dirkriehle.com> – [@dirkriehle](#)

Credits and License

- Original version
 - © 2012-2019 [Dirk Riehle](#), some rights reserved
 - Licensed under [Creative Commons Attribution 4.0 International License](#)
- Contributions
 - ...