UNIVERSITY OF GOTHENBURG

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## DIT246 Empirical Software Engineering, 7.5 credits

Empirisk programvaruteknik, 7,5 högskolepoäng

*Second Cycle*

---

### Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2020-10-27 to be valid from 2021-08-30, autumn semester of 2021.

*Field of education:* Science 100%
*Department:* Department of Computer Science and Engineering

### Position in the educational system

The course is a compulsory course in the Software Engineering, Master's Programme. The course is offered within the framework of several degree programmes. The course is also a single subject course at the University of Gothenburg.

The course can be part of the following programmes: 1) Computer Science, Master's Programme (N2COS), 2) Applied Data Science Master's Programme (N2ADS), 3) Game Design & Technology Master's Programme (N2GDT) and 4) Software Engineering and Management Master's Programme (N2SOF)

| *Main field of studies* | *Specialization* |
|---|---|
| Mathematics | A1N, Second cycle, has only first-cycle course/s as entry requirements |
| Software Engineering | A1N, Second cycle, has only first-cycle course/s as entry requirements |

### Entry requirements

To be eligible for the course Empirical Software Engineering the student should have a bachelor degree in Software Engineering, Computer Science or equivalent.

Applicants must prove knowledge of English: English 6/English B or the equivalent level of an internationally recognized test, for example TOEFL, IELTS.

**Learning outcomes**

After completion of the course the student is expected to be able to:

*Knowledge and understanding*

- describe, understand, and apply empiricism in software engineering
- describe, understand, and partly apply the principles of case studyresearch/experiments/surveys
- describe and understand the underlying principles of meta-analytical studies
- explain the importance of research ethics
- state and explain the importance of threats to validity and how to control said threats
- describe and explain the concepts of probability space (incl. conditional probability), random variable, expected value and random processes, and know a number of concrete examples of the concepts
- describe Markov chain Monte Carlo methods such as Metropolis
- describe and explain Hamiltonian Monte Carlo
- explain and describe multicollinearity, post-treatment bias, collider bias, and confounding
- describe and explain ways to avoid overfitting

*Competence and skills*

- assess suitability of and apply methods of analysis on data
- analyse descriptive statistics and decide on appropriate analysis methods
- use and interpret code of ethics for software engineering research
- design statistical models mathematically and implement said models in a programming language
- make use of random processes, i.e., Bernoulli, Binomial, Gaussian, and Poisson distributions, with over-dispersed outcomes
- make use of ordered categorical outcomes (ordered-logit) and predictors
- assess suitability of, from a ontological (natural process) and epsitemological (maxent) perspective, various statistical distributions
- make use of and assess directed acyclic graphs to argue causality

*Judgement and approach*

- state and discuss the tools used for data analysis and, in particular, judge their output
- judge the appropriateness of particular empirical methods and their applicability to attack various and disparate software engineering problems

- question and assess common ethical issues in software engineering research
- assess diagnostics from Hamiltonian Monte Carlo and quadratic approximation using information theoretical concepts, i.e., information entropy, WAIC, and PSIS-LOO
- judge posterior probability distributions for out of sample predictions and conduct posterior predictive checks

**Course content**

This course is for students who are interested in the empirical methods applied to the field of software engineering. The course introduces quantitative and qualitative methods in software engineering with accompanying statistical methods used for analysis.

The course contains:

- Descriptive and inferential statistical methods applied to software engineering.
- Conducting qualitative and quantitative methods in software engineering.
- Methods for analyzing quantitative and qualitative data in software engineering.
- Usage of statistical tools.

*Sub-courses*

1. **Assignments** *(Inlämningsuppgifter),* 2.5 credits
   Grading scale: Pass (G) and Fail (U)

2. **Written hall examination** *(Skriftlig salstentamen),* 5 credits
   Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)

**Form of teaching**

Lectures, laboratory work, group supervision, problem-based teaching.

*Language of instruction:* English

**Assessment**

The course is examined by written lab assignment carried out in groups of normally 3-4 students (2.5 hec). The assignments are graded individually, taking into account the group work as well as the student's individual contribution to the group work.
The course is also examined by an individual written exam carried out in an examination hall (5 hec). The assignment are both theoretical and practical in nature.

If a student, who has failed the same examined component twice, wishes to change examiner before the next examination, a written application shall be sent to the

department responsible for the course and shall be granted unless there are special reasons to the contrary (Chapter 6, Section 22 of Higher Education Ordinance).

In cases where a course has been discontinued or has undergone major changes, the student shall normally be guaranteed at least three examination occasions (including the ordinary examination) during a period of at least one year from the last time the course was given.

**Grades**

The grading scale comprises: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U).
In order to pass the course both the assignments and the written hall examination have to be approved. The final grade in the course is decided from the grade of the written hall examination.

**Course evaluation**

The course is evaluated through meetings both during and after the course between teachers and student representatives. Further, an anonymous questionnaire is used to ensure written information. The outcome of the evaluations serves to improve the course by indicating which parts could be added, improved, changed or removed.

**Additional information**

The course is a joint course together with Chalmers.

Course literature to be announced the latest 8 weeks prior to the start of the course.

The course replaces the course DIT278, 7.5 credits. The course cannot be included in a degree which contains DIT278. Neither can the course be included in a degree which is based on another degree in which the course DIT278 is included.