Blekinge Institute of Technology
Department of Software Engineering

Revision: 3
Reg.no: BTH-4.1.14-1505-2022

COURSE SYLLABUS

Testning av mjukvara

Software Testing

6 credits (6 högskolepoäng)

Course code: PA1465
Main field of study: Software Engineering, Technology
Disciplinary domain: Technology
Education level: First cycle
Specialization: GXX - First cycle, in-depth level of the course cannot be classified

Language of instruction: The course is taught primarily in English, but teaching in Swedish may occur.
Applies from: 2023-08-28
Approved: 2022-12-15

1. Decision

This course is established by Dean 2022-12-15. The course syllabus is approved by Head of Department of Software Engineering 2022-12-15 and applies from 2023-08-28.

2. Entry requirements

Admission to the course requires 12 credits completed in programming and completed course Software Engineering, 6 credits.

3. Objective and content

3.1 Objective

Testing is vital software engineering activity, but it is often challenging to perform software testing in an efficient and effective manner. The objectives of this course are for participants to appreciate how testing can improve software quality when effectively integrated into the software engineering processes, understand how this can be achieved using both established and novel software testing techniques, and gain practical experience of tools that support and automate these techniques.

3.2 Content

The following core contents will be discussed during the course:

1. Basic concepts and theories for software testing
2. Manual testing
3. Automated testing
4. Testing in continuous development
5. Application domains

4. Learning outcomes

The following learning outcomes are examined in the course:

4.1 Knowledge and understanding

On completion of the course, the student will be able to:

- Describe quality models, cost of quality, and continuous quality improvement and metrics to assess the quality of a test suite.
- Describe methods to generate tests from requirements (e.g., behavior modeling and test generation using FSM, input space modeling using combinatorial designs, combinatorial test generation).

4.2 Competence and skills

On completion of the course, the student will be able to:

- Apply different testing techniques, tools and libraries, from test harnesses to assertion libraries.
- Generate software test plans with well-defined objectives and targets (including design, implementation, and maintenance) while taking into account practical considerations (e.g., time, budget, profiles).

5. Learning activities

- Discussion sessions/debates on different topics in testing

- Team projects
- The course summarizes basic topics of software quality and testing; and introduces the students to more advanced ones through lectures mixing theory and discussion-based approaches. A team project, ongoing for the duration of the course, gives the students the opportunity to link the techniques, tools/framework, and methods learned to complex settings similar to real-world projects. The project results are summarized in a written report and presented to the class.

6. Assessment and grading

Modes of examinations of the course

Code	Module	Credits	Grade
2105	Written assignment	2 credits	AF
2115	Project assignment	4 credits	AF

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Fail, supplementation required, F Fail.

The final course grade is the average of the grades in the individual modules. To get a passing grade for the course, all modules must be approved.

The information before a course occasion states the assessment criteria and make explicit in which modes of examination that the learning outcomes are assessed.

An examiner can, after consulting the Disability Advisor at BTH, decide on a customized examination form for a student with a long-term disability to be provided with an examination equivalent to one given to a student who is not disabled.

7. Course evaluation

The course evaluation should be carried out in line with BTH:s course evaluation template and process.

8. Restrictions regarding degree

The course can form part of a degree but not together with another course the content of which completely or partly corresponds with the contents of this course.

9. Course literature and other materials of instruction

- Jorgensen, P.C., 2013, Software Testing: A Craftsman's Approach, 4th Ed, Auerbach Publications. ISBN: 978-1466560680
- Myers, G.J., Sandler, C., Badgett, T., 2011, The Art of Software Testing, 3rd Ed, Wiley. ISBN: 978-1-119-20248-6
- McCaffrey, J.D., 2009, Software Testing: Fundamental Principles and Essential Knowledge, BookSurge Publishing
- Graham D., Veenendaal E.V., Evans I., Black R., Foundations of Software Testing (ISTQB certification). ISBN: 978-1844809899
- Davis J., Daniels R., Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale, 1st Ed. ISBN: 978-1491926307
- Torbjörn Ryber, Essential Test Design. 2007. ISBN: 978-9185699032