

HANDBOOK

BACHELOR OF INFORMATICS PROGRAM

DEPARTMENT OF INFORMATICS

FACULTY OF INTELLIGENT ELECTRICAL AND INFORMATICS TECHNOLOGY

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

Module name	Software Quality Assurance
Module level	Undergraduate
Code	IF184972
Courses (if applicable)	Software Quality Assurance
Semester	8
Contact person	-
Lecturer	Ir. Siti Rochimah, MT.,Ph.D.
Language	Bahasa Indonesia and English
Relation to curriculum	<ol style="list-style-type: none"> 1. Undergraduate degree program; optional; 8th semester. 2. International undergraduate program; optional; 8th semester.
Type of teaching, contact hours	<ol style="list-style-type: none"> 1. Undergraduate degree program: lectures, < 60 students, 2. International undergraduate program: lectures, < 40 students
Workload	<ol style="list-style-type: none"> 1. Lectures: 3 x 50 = 150 minutes (2 hours 30 minutes) per week. 2. Exercises and Assignments: 3 x 60 = 180 minutes (3 hours) per week. 3. Private study: 3 x 60 = 180 minutes (3 hours) per week.
Credit points	3 credit points (sks).
Requirements according to the examination	A student must have attended at least 80% of the lectures to sit in the exams.

regulations	
Mandatory prerequisites	-
	After completing this module, a student is expected to:

Learning outcomes and their corresponding PLOs	CO1 Able to understand the basics of software testing.	
	CO2 Able to explain in detail, differentiate, and perform types and levels of testing (unit, integration, system, acceptance).	
	CO3 Able to explain and perform testing techniques.	
	CO4 Able to explain in detail the important test case identification techniques for unit, integration, and system testing.	
	CO5 Able to implement an inspection or review process of software source code on small or medium scale software projects.	
	CO6 Able to actively participate as part of team activities to practice the inspection process for small / medium scale source code segments.	
	CO7 Able to take measurements related to testing.	
	CO8 Able to carry out the testing process / procedure.	
	CO9 Be able to properly explain the verification and validation process for non-source code artifacts.	
	CO10 Able to use testing tools in the implementation of the testing process.	
	CO11 Be able to make good use of software defect tracking tools to manage software defects in small-scale software projects, and analyze their results.	
	CO12 Able to understand the basics of software quality assurance.	
Content	<p>Knowledge:</p> <p>Mastering the concepts and principles: designing and building software with standard and scientific methods of planning, requirements engineering, designing, implementing, testing, and launching, and producing software products that meet various technical and managerial quality parameters, and are efficient and mastering the concepts and principles: making simple programs in general programming languages and object-oriented programming languages, creating web applications and desktop applications, creating simple databases to solve problems in the context of software development in general.</p>	

	<p>Specific Skill:</p> <p>Able to analyze, design and build software using the principles of software engineering processes to produce software that meets both technical and managerial quality.</p>
Study and examination requirements and forms of examination	Mid-terms examination and Final examination.
Media employed	LCD, whiteboard, websites, books (as references), etc.
Assessments and Evaluation	
Reading List	<p>S. Naik and P. Tripathy, Software Testing and Quality Assurance: Theory and Practice, Wiley-Spektrum, 2008.</p> <p>S.H. Kan, Metrics and Models in Software Quality Engineering, 2nd ed., Addison-Wesley, 2002.</p> <p>D. Galin, Software Quality Assurance: From Theory to Implementation, Pearson Education Limited, 2004.</p>