Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Department of Applied Radio Electronics

[RE-8] SOFTWARE QUALITY CONTROL

Curriculum (Syllabus)

Course details

Level of higher education	First (bachelor's)
Field of knowledge	17 - Electronics, Automation, and Electronic Communications
Specialization	172 - Electronic Communications and Radio Engineering
Educational program	All educational programs
Discipline status	Elective (F-catalog)
Form of higher education	Full-time
Year of training, semester	Available for selection starting from the 2nd year, spring semester
Scope of the discipline	4 credits (Lectures 18 hours, Practical 36 hours, Lab 0 hours, Independent work 66 hours)
Semester Control/control measures	Credit
Class schedule	https://rozklad.kpi.ua
Language of instruction	Ukrainian
Information about the course leader/teachers	Lecturer: Nikitchuk A. V., Practical: Nikitchuk A. V.,
Course location	https://do.ipo.kpi.ua/course/view.php?id=5221

Curriculum

1. Description of the academic discipline, its purpose, subject matter, and learning outcomes

The discipline provides students with an understanding of the principles, methods, and means of ensuring the quality of software systems used in modern radio-electronic equipment and communication devices.

The aim of the discipline is to provide knowledge and practical skills in assessing, controlling, and improving the quality of software throughout its life cycle.

The subject of study is the processes, methodologies, and tools for ensuring software quality (quality models, metrics, testing, standards, defect management).

After studying the discipline, the student should:

Know:

- the basic concepts, models, and factors of software quality;
- the stages of the software life cycle and the role of quality control at each stage;
- the specifics of quality requirements for software;
- types, levels, and methods of testing;
- agile software development methodologies;
- tools for testing, quality assurance, and teamwork; •

principles of building automated testing systems.

Be able to:

- plan the quality assurance process for a specific project;
- create test plans, test cases, defect reports;
- work in a quality control team;
- apply various types of testing and test design methods;
- document identified defects;
- analyze code and evaluate the quality of the developed product.
- use tools for test automation.

2. Prerequisites and post-requisites of the discipline (place in the structural-logical scheme of training under the relevant educational program)

Prerequisites. For successful completion of the course, students must have basic knowledge of:

- Computer science.
- Programming fundamentals.
- English.

Post-requisites. The acquired knowledge and skills can be used not only for software quality control, but also for ensuring the quality and reliability of electronic devices and systems at all stages of their life cycle. The acquired knowledge and skills are used in the following educational components:

- .NET technologies for software development. •

Reliability of radio-electronic equipment.

- Diploma design.

3. Contents of the academic discipline

- Topic 1. Software quality and requirements specifications •

Topic 2. Software life cycle

- Topic 3. Levels and types of testing •

Topic 4. Test planning and design

- Topic 5. Bug reports. Agile development methodologies. Popular tools •

Topic 6. Automated testing

- Topic 7. Features of web application testing
- Topic 8. Features of mobile application testing

4. Training materials and resources

Basic literature

1. Krepych S.Ya., Spivak I.Ya. Software Quality and Testing: Basic Course / Ternopil 2020.
2. DSTU 2469-24. Certification. Basic concepts, terms, definitions.
3. DSTU 2850-94. Computer software. Quality indicators and assessment methods.

Supplementary literature

1. BLACK, Rex. *Critical testing processes: plan, prepare, perform, perfect*. Addison-Wesley Professional, 2004.
2. BRAUDE, Eric J.; BERNSTEIN, Michael E. *Software engineering: modern approaches*. Waveland Press, 2016.
3. DUSTIN, Elfriede; GARRETT, Thom; GAUF, Bernie. *Implementing automated software testing: How to save time and lower costs while raising quality*. Pearson Education, 2009.
4. KANER, Cem; FALK, Jack; NGUYEN, Hung Q. *Testing computer software*. John Wiley & Sons, 1999.
5. MCGREGOR, John D.; SYKES, David A. *A practical guide to testing object-oriented software*. Addison-Wesley Professional, 2001.

Information resources

1. https://qalight.ua/

Educational content

5. Methodology for mastering the academic discipline (educational component)

Classes Description

Topic 1. Software quality and requirements specifications

Introduction. Software quality. Software characteristics. Quality models and metrics.

Lecture 1 Software life cycle. Requirements and their analysis. Requirements testing techniques.

PR 1 Requirements analysis and testing

Topic 2. Software life cycle

Software life cycle. Stages of the development cycle. Types and purposes of life cycle

Lecture 2 models. Errors in software. Beginning and end of testing.

PR 2 Defects and their life cycle

Topic 3. Levels and types of testing

Lecture 3 Testing axioms. Testing principles. Testing levels. Types of testing.
Classification of types of testing.

PR 3 Test plan

Topic 4. Test planning and design

Lecture 4 Testing axioms. Testing principles. Testing levels. Types of testing.
Classification of testing types.

PR 4 Test cases

Topic 5. Bug reports. Agile development methodologies. Popular tools

Lecture 5 Defect report (bug report). Agile development methodologies: Scrum; Kanban.
Popular tools and management systems.

PR 5 Creating and submitting defect reports in the error tracking system

Topic 6. Automated testing

Comparison of manual and automated testing. Modular testing. AAA (Arrange-Act-Assert)

Lecture 6 approach to test writing. NUnit attributes and methods.

PR 6 Automation of code testing using the nUnit framework (modular testing)

Topic 7. Features of web application testing

Lecture 7 Features of web application testing. Developer tools in the browser.
Testing automation with Selenium.

PR 7 Automating web application testing using Selenium WebDriver

Topic 8. Features of mobile application testing

Lecture 8	Aspects of mobile application development and testing. Classification of mobile application testing. Tools for automating mobile application testing.
PR 8	Automation of mobile application testing using Appium and Android SDK tools

6. Independent work by students

1. Throughout the semester:

- *Study of lecture material.*
- *Reviewing literature sources.*
- *Answering questions for self-assessment and taking tests.*

2. During the week before the scheduled date

- *Preparation for practical work.*
- *Preparation for writing a test.*
- *Preparation for completing homework assignments.* •

Preparation for the test.

Policy and control

7. Academic discipline policy (educational component)

Rules for attending classes:

- *for lectures and practical classes - attendance at classes (Zoom video conferences) according to the schedule;*
- *independent study of the material using lecture recordings and other materials posted in the relevant distance learning course is permitted;*
- *asynchronous completion of practical assignments is permitted.*

Rules of conduct in class:

- *during classes, you must use the Internet to: complete assignments in the distance learning course; familiarize yourself with the links provided; access modern, organized sources of information;*
- *The use of mobile phones, laptops, and other devices is permitted.*

Rules for performing practical work:

- *if the teacher has questions about the results obtained, it is necessary to undergo an oral defense procedure (answer questions);*
- *The defense procedure is considered timely if it is completed during the class dedicated to the work or the next class according to the schedule.*

Rules for awarding incentive points:

- *Bonus points are awarded for completing additional tasks specified in the assignments.*

Rules for assigning penalty points:

- *Penalty points may be awarded for late submission/defense of practical work.*

Deadline and retake policy:

- *tests, exams, and practical assignments must be completed by the last class of the semester.*

8. Types of control and rating system for assessing learning outcomes (RSO)

- *Ongoing assessment: quizzes (tests) on lecture topics (16 points), practical work (64 points), Module Control Work (10 points), Home Control Work (10 points).*
- *Calendar assessment: conducted twice per semester as monitoring of the current status of syllabus requirements.*
- *Semester assessment: credit.*
- *Conditions for admission to semester assessment: semester rating of more than 60 points.*

Table of correspondence between rating points and grades on the university scale

Number of points	Rating
100-95	Excellent
94	Very good
84	Good
74-65	Satisfactory
64-60	Sufficient
Less than 60	Unsatisfactory
Admission requirements not met	Not admitted

9. Additional information on the discipline (educational component)

Description of material, technical, and informational support for the discipline

Practical work is carried out on personal computers or on computers in the department's computer labs.

Work program of the academic discipline (syllabus):
Compiled by Nikitchuk A. V.;
Approved by the PRE department (protocol No. 06/2024 dated 06/27/2024)
Approved by the methodological commission of the faculty/research institute (protocol No. 06/2024 dated 28.06.2024)