

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

ЗАТВЕРДЖЕНО

на засіданні кафедри
інформаційних систем.
Протокол № 1 від 22.08.2023 р.

ПОГОДЖЕНО

Проректор з навчально-методичної роботи



Карина НЕМАШКАЛО

ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ

робоча програма навчальної дисципліни (РПНД)

Галузь знань 12 "Інформаційні технології"
Спеціальність 121 "Інженерія програмного забезпечення"
Освітній рівень перший (бакалаврський)
Освітня програма "Інженерія програмного забезпечення"

Статус дисципліни обов'язкова
Мова викладання, навчання та оцінювання англійська

Розробник:
к.е.н., доцент

підписано КЕП

Ірина УШАКОВА

Завідувач кафедри
інформаційних систем

Дмитро БОНДАРЕНКО

Гарант програми

Олег ФРОЛОВ

Харків
2024

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
SIMON KUZNETS KHARKIV NATIONAL UNIVERSITY
OF ECONOMICS

APPROVED

at the meeting of the Information
Systems Department

Protocol No. 1 of 22.08.2023



AGREED

Vice-rector for educational and methodological
work

Karina NEMASHKALO

SOFTWARE QUALITY ASSURANCE AND TESTING
Program of the course

Branch of knowledge **12 "Information technologies"**
Specialty **121 "Software engineering"**
Study cycle **first (bachelor)**
Study program **"Software Engineering"**

Course status
Language

mandatory
English

Developer:
Professor,
PhD (Economics)

digital signature

Iryna USHAKOVA

Head of Information
Systems Department

Dmytro BONDARENKO

Head of Study program

Oleg FROLOV

Kharkiv
2024

Handwritten signature and date: 22.08.2023

INTRODUCTION

The course "Software quality assurance and testing" belongs to the mandatory component of the professional cycle of the educational program 121 "Software engineering" of the first (bachelor) level. The content of the course covers a range of issues related to determining the quality of software (software), its control and testing, which includes activities for work planning (Test Management), test design (Test Design), test execution (Test Execution) and analysis of the obtained results (Test Analysis).

The purpose of the course "Software quality assurance and testing" is to provide students with higher education with a system of special knowledge to provide basic profiling training in a specialty, the formation of theoretical knowledge and practical skills to ensure and control the quality of software during its development.

The tasks of the course are:

- mastering the theoretical foundations of software quality and software testing by students of higher education;
- mastering software quality management processes;
- basic methods of building tests and software testing;
- approaches to creating reports on identified problems;
- familiarization with quality control tools.

The subject of the course is a variety of operating systems, their architecture and main components and objects, which are considered in the form of sets of characteristics.

The object of the course is modern theoretical concepts and methodologies, principles of operation, provision of basic profiling training in the specialty, formation of theoretical knowledge and practical skills in ensuring and controlling the quality of software during its development and testing.

The learning outcomes and competence formed by the course are defined in the table. 1.

Table 1

Learning outcomes and competences formed by the course

Learning outcomes	Competences
LO03	GC07, SC5, SC11, SC12
LO09	GC02, GC03, SC01, SC04, SC05
LO14	GC03, SC02, SC13
LO19	GC02, SC04, SC05
LO20	GC02, SC04, SC09
LO23	SC10

where, LO03. Know the basic processes, phases, and iterations of the software lifecycle.

LO09. Know and be able to use methods and tools for collecting, formulating and analyzing software requirements.

- LO14. Apply in practice software tools for domain analysis, design, testing, visualization, measurement and documentation of software.
- LO19. Know and be able to apply software verification and validation methods.
- LO23. Be able to document and present the results of software development.
- GC02. Ability to apply knowledge in practical situations.
- GC03. Ability to communicate in the state language both orally and in writing.
- GC07. Ability to work in a team.
- SC01. Ability to identify, categorize and formulate software requirements.
- SC02. Ability to participate in the design of software, including modelling (formal description) of its structure, behaviour and processes of operation.
- SC04. Ability to formulate and ensure software quality requirements in accordance with customer requirements, terms of reference and standards.
- SC05. Ability to comply with specifications, standards, rules and guidelines in the professional field when implementing life cycle processes.
- SC09. Ability to evaluate and take into account economic, social, technological and environmental factors affecting the field of professional activity.
- SC09. Ability to evaluate and take into account economic, social, technological and environmental factors affecting the field of professional activity.
- SC10. The ability to accumulate, process, and systematize professional knowledge about creating and maintaining software and recognize the importance of life long learning.
- SC11. Ability to implement phases and iterations of the life cycle of software systems and information technologies based on appropriate software development models and approaches.
- SC12. Ability to carry out the system integration process, apply change management standards and procedures to maintain the integrity, overall functionality and reliability of the software.
- SC13. Ability to reasonably choose and master tools for software development and maintenance.

COURSE CONTENT

Content module 1. Fundamentals of software quality assurance and testing.

Topic 1. Introduction to software quality assurance and testing.

- 1.1. The concept of quality, quality control and software testing.
- 1.2. Software quality metrics.
- 1.3. History of testing development.
- 1.4. The role of testing in software quality assurance.
- 1.5. Software life cycle models.
- 1.6. Life cycle of testing.
- 1.7. Technical skills and personal qualities of the tester.

Topic 2. Types and directions of testing.

- 2.1. Classification of testing.
- 2.2. Alternative and additional types of testing.
- 2.3 Testing according to the method of white and black boxes.

Topic 3. Testing documentation and requirements.

- 3.1. Definition and importance of software requirements.
- 3.2 Sources and ways of identifying requirements.
- 3.3. Levels and types of requirements.

- 3.4. Properties of quality requirements.
- 3.5. Requirements testing techniques.
- 3.6. An example of requirements analysis and testing.
- 3.7. Typical errors in requirements analysis and testing.

Topic 4. Concepts and properties of checklists, test cases, sets of test cases.

- 4.1. Concept and properties of check letters.
- 4.2. The concept of a test case and its life cycle.
- 4.3. Attributes (fields) of the test case.
- 4.4. Test management tools.
- 4.5. Properties of quality test cases.
- 4.6. Sets of test cases.
- 4.7. The logic of creating effective checks.
- 4.8. Typical mistakes when developing checklists, test cases and sets of test cases.

Topic 5. Finding and documenting defects

- 5.1. Definition of basic concepts related to software defects.
- 5.2. Report on the defect and its life cycle.
- 5.3. Attributes (fields) of the defect report.
- 5.4. Defect report management tools.
- 5.5. Properties of quality defect reports.
- 5.6. The logic of creating effective defect reports.
- 5.7. Common mistakes when writing defect reports.

Content module 2. Organization of software testing processes

Topic 6. Planning the testing process

- 6.1 Products to be tested.
- 6.2. Tasks and advantages of quality planning.
- 6.3. Test plan and test results report.
- 6.4. Planning and estimation of labor costs based on requirements and test cases.

Topic 7. Peculiarities of testing mobile applications.

- 7.1. Features of mobile applications.
- 7.2. Types of mobile application testing.
- 7.3. Peculiarities of testing mobile applications.
- 7.4. Approaches to testing.

Topic 8. Fundamentals of test automation.

- 8.1. Key concepts of test automation.
- 8.2. Fields of application of test automation.
- 8.3. Factors of test automation.
- 8.4. Advantages and disadvantages of test automation.
- 8.5. Technical features of testing automation.
- 8.6. Tools for test automation. Selenium.

Topic 9. Performance testing.

- 9.1. Key concepts of performance testing.

- 9.2. Performance testing stages.
- 9.3. Performance testing reporting.
- 9.4. Apache Jmeter 15.
- 9.5. Additional means of collecting information about application activity.

Topic 10. Use of various testing techniques.

- 10.1. Positive and negative test cases.
- 10.2. Equivalence classes and boundary conditions.
- 10.3. Domain testing and parameter combinations.
- 10.4. Paired testing and search for combinations.
- 10.5. Exploratory testing.
- 10.6. Finding the causes of defects.

The list of laboratory classes by course is given in the table. 2.

Table 2

List of laboratory classes

Topic name	Content
Topic 2. Laboratory work 1.	Types of testing
Topic 3. Laboratory work 2.	Software requirements testing
Topic 4. Laboratory work 3.	Creation of checklists
Topic 4. Laboratory work 4.	Creation of test cases
Topic 5. Laboratory work 5.	Finding and documenting defects.
Topic 5. Laboratory work 6.	Track defects in Jira
Topic 6. Laboratory work 7.	Development of a test plan
Topic 8. Laboratory work 8.	Automated testing in Selenium

The list of self-studies in the course is given in table 3.

Table 3

List of self-studies

Topic name and/or task	Content
Topic 1 - 10	Study of theoretical material
Topic 2 - 6, 8	Completing tasks for laboratory work

The number of hours of lecture and laboratory studies and hours of self-study is given in the technological card of the course.

TEACHING METHODS

In the process of teaching the course "Software quality assurance and testing" for the acquisition of specified learning outcomes and the activation of the educational process, the use of productive learning methods aimed at activating and stimulating the educational and cognitive activity of higher education students is foreseen , such as: verbal and visual presentations to lectures (Topic 1, Topic 2, Topic 7, Topic 9, Topic 10), case studies (Topic 3, Topic 4, Topic 5, Topic 6, Topic 8).

FORMS AND METHODS OF ASSESSMENT

The University uses a 100-point cumulative system for assessing the learning outcomes of students.

Current control is carried out during lectures and laboratory classes and is aimed at checking the level of readiness of a higher education student to perform a specific job and is evaluated by the sum of points scored:

– for courses with a form of semester credit control: the maximum amount is 100 points; the minimum amount is 60 points.

The final control includes the semester control and certification of the student of higher education.

Semester control is conducted in the form of credit. The score is set as the total number of points obtained during the current control.

Current monitoring includes assessing students during:

- lectures - express survey (10 points), theoretical control work (20 points). The total number of points is 30;

– laboratory classes - defense of laboratory works (8 laboratory works) on the condition that the student submits a report with the results of the work performed. The total number of points is 70 points.

During the semester, two theoretical control papers are conducted in the form of tests using the distance learning system. The tests consist of multiple-choice tasks and are limited by the time of their completion. The student has one attempt to complete the test tasks. The grade on the test task is reduced if there is no answer to the question or an incorrectly given answer.

The maximum possible score for a specific task (express survey, laboratory work) is given on the condition that the student's individual task and his oral answer meet all the specified criteria. The absence of one or another component reduces the number of points.

The final grade for the course is defined as the summation of all points obtained during the current control.

More detailed information on the assessment system is provided in technological card of the course.

RECOMMENDED LITERATURE

Main

1. Ушакова І. О. Лабораторний практикум з системного аналізу та проектування інформаційних систем [Електронний ресурс] : навчальний посібник / І. О. Ушакова, І. Б. Медведєва. – Харків : ХНЕУ ім. С. Кузнеця, 2022. – 251 с. - Режим доступу : <http://repository.hneu.edu.ua/handle/123456789/27815>.

2. Козак О.Л. Опорний конспект лекцій з курсу «Аналіз вимог до програмного забезпечення» для студентів напрямку підготовки «Програмна інженерія» / О.Л. Козак. – Тернопіль, 2021. – 56 с.

3. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник / За ред. Крепич С.Я., Співак І.Я. / для бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення». – Тернопіль: ФОП Паляниця В.А., 2020. – 478с.

Additional

4. Бородкіна, І.Л. Інженерія програмного забезпечення : навч. посіб. / І.Л. Бородкіна, Г.О. Бородкін. - Київ. : ЦУЛ, 2019. - 204 с.

5. ДСТУ ISO/IEC 25000:2016. Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). Настанова до SQuaRE: чинний з 01.01.2018. – Київ. : УкрНДНЦ.

6. ДСТУ ISO/IEC 25010:2016. Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). Моделі якості системи та програмних засобів: чинний з 01.01.2018. – Київ. : УкрНДНЦ.

7. ДСТУ ISO/IEC 25020:2016. Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). Рамкова модель і настанова щодо вимірювання: чинний з 01.01.2018. – Київ. : УкрНДНЦ.

8. ДСТУ ISO/IEC 25022:2019 (ISO/IEC 25022:2016, IDT) Інженерія систем і програмних засобів. Вимоги до якості систем програмних засобів та їхнього оцінювання (SQuaRE). Вимірювання якості під час застосування. : чинний з 01.01.2020. – Київ. : УкрНДНЦ.

9. ДСТУ ISO/IEC 25023:2019. Інженерія систем і програмних засобів. Вимоги до якості систем програмних засобів та їхнього оцінювання (SQuaRE). Вимірювання якості систем та програмних продуктів: чинний з 01.11.2019. – Київ. : УкрНДНЦ.

10. ДСТУ ISO/IEC 25040:2016 (ISO/IEC 25040:2011, IDT) Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). Процес оцінювання: чинний з 01.01.2018. – Київ. : УкрНДНЦ.

11. Ushakova I. Approaches to Web Application Performance Testing And Real-Time Visualization of Results / I.O.Ushakova, O. Plokha, Yu. Skorin // Вісник ХНАДУ, вип. 96, 2022. – С.71-80. - Режим доступу: <http://repository.hneu.edu.ua/handle/123456789/27384>.

12. Ushakova I. Methods of quality assurance of software development based on systems approach / I.Ushakova, Yu. Skorin, A. Shcherbakov // Proc. of the 3rd International Conference on Information Security and Information Technologies (ISecIT 2021) co-located with 1st International Forum "Digital Reality" (DRForum 2021), Odesa, Ukraine, September 13–19, 2021. – CEUR Workshop Proceedings (CEUR-WS.org).– 2021.– Vol. 3200. – P. 158-168. – Access mode: <http://repository.hneu.edu.ua/handle/123456789/28596>.

13. Wiegers K. Software Requirements Essentials: Core Practices for Successful Business Analysis; 1st Edition / K. Wiegers, C. Hokanson. – Addison-Wesley Professional, 2022.– 208 p.

Information resources

14. Ушакова І.О. Якість програмного забезпечення та тестування: методичне забезпечення навчальної дисципліни / І.О. Ушакова [Електронний ресурс] // Сайт ПНС ХНЕУ ім. С. Кузнеця [Електронний ресурс]. – Режим доступу : <https://pns.hneu.edu.ua/course/view.php?id=2553>.

15. Try QA [Electronic recourse]. – Access mode : <http://tryqa.com/>.

16. QALight [Electronic recourse]. – Access mode : <https://qalight.ua/baza-znaniy/mobilnij-ta-veb-dodatok-u-chomu-rizniczya/>.

17. Спільнота розробників dou.ua [Електронний ресурс]. – Режим доступу : <https://dou.ua>