

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра радіоелектронних і комп'ютерних систем

Затверджено

На засіданні
кафедри радіоелектронних і
комп'ютерних систем
факультету електроніки та комп'ютерних
технологій
Львівського національного університету
імені Івана Франка
(протокол № 1/24 від 28.08, 2023 р.)

Завідувач кафедри:



Ігор ОЛЕНИЧ

Силабус з навчальної дисципліни
“Якість та тестування ПЗ”,
що викладається в межах ОПП
“ Високопродуктивний комп'ютинг ”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 121 – Інженерія програмного забезпечення

Львів 2023 р.

Назва дисципліни	Якість та тестування ПЗ
Адреса викладання дисципліни	Корпус факультету електроніки та комп'ютерних технологій, Львівський національний університет імені Івана Франка, м. Львів, вул. Драгоманова, 50, 79005
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій Кафедра радіоелектронних і комп'ютерних систем
Галузь знань, шифр та назва спеціальності	12 – інформаційні технології 121 – Інженерія програмного забезпечення
Викладачі дисципліни	Оленич Ігор Богданович, д-р фіз.-мат. наук Припула Маріанна Миколаївна, асистент
Контактна інформація викладачів	igor.olenych@lnu.edu.ua https://electronics.lnu.edu.ua/employee/olenych-i-b/marianna.prytula@lnu.edu.ua https://electronics.lnu.edu.ua/employee/prytula-marianna-mykolaivna
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекцій/лабораторних занять (за попередньою домовленістю): ауд. 101, корпус факультету електроніки та комп'ютерних технологій, вул. Драгоманова 50, м. Львів
Коротка анотація дисципліни	Дисципліна «Якість та тестування ПЗ» є нормативною дисципліною з спеціальності 121 – Інженерія програмного забезпечення для освітньої програми “Високопродуктивний комп'ютинг”, яка викладається в 5-му семестрі в обсязі 3,5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Інформація про дисципліну	Навчальну дисципліну розроблено таким чином, щоб надати студентам необхідні теоретичні знання та практичні навички в області забезпечення якості та тестування програмного забезпечення. У дисципліні представлено огляд концепцій та методик тестування, включаючи як мануальне, так і автоматизоване тестування.
Мета та цілі дисципліни	<i>Мета:</i> ознайомлення студентів та одержання знань про основні види та методи тестування програмного забезпечення, критеріїв тестування, видів тестування, процесів управління якістю ПЗ; набуття студентами практичних навичок використання інструментів автоматизації тестування, проведення unit-тестування, тестування баз даних та веб-додатків. <i>Цілі:</i> забезпечити знайомство студентів з загальною теорією забезпечення якості та тестування програмного забезпечення; навчити студента вибирати і обґрунтувати види тестування відповідно до поставлених цілей, використовувати методи мануального та автоматизованого тестування, створювати набори тестів.
Література для вивчення дисципліни	Основна література: 1. Крепич С.Я., Співак І.Я. "Якість програмного забезпечення та тестування: базовий курс", 2020. — 479 с. 2. Svyatoslav Kulikov Software Testing. Base Course. (3rd edition), 2022. — 280 p. 3. Lee Copeland A Practitioner's Guide To Software Test Design, 2004. — 355 p. 4. Glenford J.Myers The Art of Software Testing, 2016. — 1512 с. 5. Lisa Crispin, Janet Gregory Agile Testing: A Practical Guide for Testers and Agile Teams, 2017. — 768 p. 6. Andreas Spillner, Tilo Linz , Software Testing Foundations: A Study Guide for the Certified Tester Exam- Foundation Level - ISTQB, 5th Edition, 2021. — 390 p. 7. Pajankar Ashwin, Python Unit Test Automation: Automate, Organize, and

	<p>Execute Unit Tests in Python, 2022. — 209 p.</p> <p>8. Tanveer, Oracle 19c SQL For Developers, Testers Analyst, 2021. — 473 p.</p> <p>9. Basu, Selenium with Python Simplified For Beginners - Simple, Concise & Easy guide to Automation Testing using Python and Selenium WebDriver, 2020. — 126 p.</p> <p>Додаткова література (Інтернет-ресурси):</p> <p>10. unittest — Unit testing framework Documentation. Available at: https://docs.python.org/3/library/unittest.html</p> <p>11. Full pytest Documentation. Available at: https://docs.pytest.org/en/7.1.x/contents.html</p> <p>12. Selenium WebDriver API. Available at: https://selenium-python.readthedocs.io/api.html</p>
<p>Обсяг курсу</p>	<p>Загальний обсяг: 105 годин. Аудиторних занять: 64 год., з них 32 год. лекційних та 32 годин лабораторних робіт. Самостійної роботи: 41 год.</p>
<p>Очікувані результати навчання</p>	<p>У результаті вивчення даного курсу студент буде:</p> <p>знати: сучасні тенденції розвитку інформаційних систем і технологій, методи гарантування якості і верифікації, рівні тестування та техніки тест-дизайну, інструменти тестування, особливості тестування веб-додатків.</p> <p>вміти: інспектувати процес розробки програмного забезпечення, розробляти тестову документацію, проводити тестування на основі специфікацій та сценаріїв, використовувати інструменти автоматизованого тестування.</p> <p>Після вивчення даного курсу «Якість та тестування ПЗ» здобувачі набудуть таких Загальних(ЗК)/Фахових(ФК) компетентностей та Програмних результатів навчання (ПРН):</p> <p>ЗК01. Здатність до абстрактного мислення, аналізу та синтезу. ЗК02. Здатність застосовувати знання у практичних ситуаціях. ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово. ФК14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування. ФК16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами. ФК17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу. ФК24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення. ФК27. Здатність розробляти високопродуктивні програмні комплекси для вирішення задач наук про дані, систем штучного інтелекту, вбудованих та інших інноваційних систем. ФК28. Володіння методами розроблення і впровадження систем підвищеної продуктивності, серверних, мікросервісних, хмаркових, розподілених та інших новітніх технологій. ФК29. Здатність здійснювати розробку програмних продуктів використовуючи різні методології і технології із забезпеченням їх інноваційності та високої продуктивності.</p> <p>ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для</p>

	<p>вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.</p> <p>ПРН03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.</p> <p>ПРН09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.</p> <p>ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.</p> <p>ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.</p> <p>ПРН16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.</p> <p>ПРН19. Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.</p> <p>ПРН20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.</p> <p>ПРН26. Знати засоби інтеграції, розгортання та підтримки спеціалізованих програмних компонентів, розроблених на основі інноваційних технологій для вирішення завдань високопродуктивних технологій.</p>
Ключові слова	Якість програмного забезпечення, тестування програмного забезпечення, Test cases, Checklists, Jira, TestRail, техніки тест-дизайну, unit-тестування, SQL, Selenium WebDriver.
Формат курсу	Очний
	Проведення лекцій, лабораторних робіт та консультації для кращого розуміння тем.
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Залік у кінці семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін: основи програмування, бази даних. веб-програмування.
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	<p>Для проведення лекційних занять: доступ до мережі Internet, засоби мультимедіа (в т.ч. проектор).</p> <p>Для проведення лабораторних занять: Комп'ютер (мінімальні характеристики: процесор Intel Core i3(4 ядра/8 потоків), 8ГБ оперативної пам'яті, 50ГБ вільного місця на диску, відеокарта Nvidia GEFORCE GT1030 2048MB). Необхідне програмне забезпечення включає в себе Jira, TestRail, PyCharm Community Edition, бібліотеки unittest, pytest, SQLAlchemy, requests, Selenium WebDriver, allure, logging, behave.</p>
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться упродовж семестру за 100-бальною шкалою. Бали нараховуються за такими видами робіт з наступним співвідношенням:</p> <ul style="list-style-type: none"> • лабораторні роботи: 50% семестрової оцінки; максимальна кількість балів 50. • контрольні заміри (2 модулі): 50% семестрової оцінки; максимальна кількість балів 50. <p>Загалом упродовж семестру 100 балів.</p>

Звітність за курс – залік в кінці семестру.

Академічна доброчесність: Очікується, що роботи студентів будуть оригінальними дослідженнями чи міркуваннями. Списування та втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в написанні завдань є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману. Жодні форми порушення академічної доброчесності не толеруються.

Відвідання занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі лекції та лабораторні заняття курсу. Студенти повинні інформувати викладача про неможливість відвідати заняття. У будь-якому випадку студенти зобов'язані дотримуватися термінів визначених для виконання всіх видів робіт, передбачених курсом.

Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

Політика виставлення балів. Враховуються бали, набрані при поточному контролі та бали за виконання лабораторних робіт. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання і т. ін.

Оцінювання лабораторних робіт (8 лабораторних робіт, максимальна кількість балів: 50) відбувається шляхом оцінки захисту звіту по виконаній лабораторній роботі:

- лабораторні роботи 1-5 (0-5 балів за одну роботу – загалом 25 балів)
- лабораторні роботи 6-7 (0-5 балів за одну роботу помножені на коефіцієнт 1.6 – загалом 16 балів)
- лабораторна робота 8 - (0-5 балів помножені на коефіцієнт 1.8 – 9 балів).

Бали оцінювання лабораторних робіт нараховуються за наступним співвідношенням:

5 – студент в повному обсязі володіє навчальним матеріалом, має повне розуміння розглянутої теми, надає правильні відповіді на запитання по темі, код програми функціонує відповідно до завдання;

4 – студент достатньо розуміє розглянутий матеріал та принципи написаного ним коду програми, присутні неточності та незначні помилки у відповідях на запитання по темі, код програми функціонує відповідно до завдання (або з несуттєвими недоліками);

3 – студент не досить добре розуміє розглянутий матеріал та написаний ним код програми, вагається та надає неточні/не конкретні відповіді на запитання по темі, код програми функціонує неточно, або з помірними недоліками;

2 – студент погано розуміє розглянутий матеріал та написаний ним код програми, студент в більшості надає помилкові відповіді на питання по темі, код програми функціонує з суттєвими недоліками;

1 - студент погано розуміє розглянутий матеріал та написаний ним код програми, код програми не функціонує належним чином;

	<p>0 - студент зовсім не засвоїв розглянутий матеріал, написаний ним код програми не відповідає темі/не функціонує взагалі.</p> <p>Контрольні заміри проводяться у формі тестових завдань на Moodle (2 модулі, 25 балів за кожний).</p> <p>Критерії оцінювання результатів неформальної освіти: Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо), курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

**Схема курсу “Якість та тестування ПЗ”
для студентів спеціальності 121 – Інженерія програмного забезпечення**

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття) **лекція, самостійна, дискусія, групова робота)	Література. * Ресурси в інтернеті	Завдання, год	Термін виконання
1	Поняття якості програмного забезпечення. Забезпечення (QA) та контроль (QC) якості ПЗ. Основні поняття тестування ПЗ.	Лекція	[1], [2], [3], [6]	2	кінець поточного тижня
1	Вступне заняття. Техніка безпеки, перевірка обладнання та налаштування програмного забезпечення.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня
2	Життєвий цикл програмного забезпечення (SDLC). Типи та призначення моделей життєвого циклу програмного забезпечення. Стадії циклу розробки програмного забезпечення у розрізі тестування. Тестова документація (Checklists).	Лекція	[1], [2], [3], [5], [6]	2	кінець поточного тижня
2	Лаб.1. Дослідницьке тестування (Exploratory testing). Написання Checklists.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня
3	Принципи та види тестування. Підходи та рівні тестування. Тестова документація.	Лекція	[1], [4], [5], [6]	2	кінець поточного тижня
3	Лаб.2. Написання баг-репортів в Jira.	Лабораторна робота	Сайт курсу -	2	кінець поточного тижня

4	Ознайомлення з інструментами для документації та управління проектами (Jira, TestRail).	Лекція	[1], [2], [4]	2	кінець поточного тижня
4	Лаб.3. Написання тест-кейсів в TestRail і проведення тестування.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня
5	Баги. Баг-репорти та їх структура. Життєвий цикл багів. Severity, Priority.	Лекція	[1], [2], [3], [6]	2	кінець поточного тижня
5	Лаб.4. Матриця відповідності вимог. Звіти в TestRail.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня
6	Тест-план. Тестова стратегія. Тест-сценарій, тест-умова і тест-кейси. Структура тест-кейсів.	Лекція	[1], [7], [*3]	2	кінець поточного тижня
7	Техніки тест-дизайну (Equivalence Partitioning, Boundary Value Analysis, Pairwise Testing, Decision Table, State transition, Use-cases).	Лекція	[1], [2], [3], [6]	2	кінець поточного тижня
6-7	Лаб.5. Використання технік тест-дизайну.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	4	кінець поточного тижня
8	Робота з вимогами. Оцінювання задач. Метрики оцінювань.	Лекція	[1], [2], [5], [6]	2	кінець поточного тижня
8	Підсумкове заняття ЗМ 1	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня
9	Фреймворки Unittest, pytest. Test fixtures. Assertions. Exception, errors, warnings. Skipping tests. Parametrized test module.	Лекція	[7], [*10], [*11]	2	Кінець поточного тижня
10	Набори тестів (Test suites). Test Runner. Logger. Рівні логування. Mocks. SQL. Ін-тепогу бази даних. Mock-и для тестування баз даних. SQLAlchemy.	Лекція	[7], [*10], [*11]	2	кінець поточного тижня
9-10	Лаб.6. Робота з SQL.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	4	кінець поточного тижня
11	Клієнт-серверна архітектура. Теорія API. REST/SOAP.	Лекція	[7], [8], [*10], [*11]	2	кінець поточного тижня
12	HTTP методи. Статус коди. Тестування API. Бібліотека requests.	Лекція	[1]	2	кінець поточного тижня
11-12	Лаб.7. Тестування API.	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	4	кінець поточного тижня
13	TDD. Test Coverage. Static code analysis. BDD. Мова Gherkin. Behave.	Лекція	[1], [7], [*10], [*11]	2	кінець поточного тижня
14	Selenium WebDriver. Locators. XPath, CSS, DOM.	Лекція	[2], [3], [12] [*1]	2	кінець поточного тижня

15	Selenium WebDriver. Locators. XPath, CSS, DOM. Selenium команди. POM. Data-driven testing.	Лекція	[9], [*12]	2	кінець поточного тижня
13-15	Лаб. 8. Написання автоматизованих тестів з використанням Selenium WebDriver. Створення логера. Генерація звітів Allure.	Лабораторна робота	[9], [*12] Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	6	кінець поточного тижня
16	Підсумкове заняття. Подальші кроки самовдосконалення.	Лекція	[1], [2], [4], [9]	2	кінець поточного тижня
16	Підсумкове заняття ЗМ 2	Лабораторна робота	Сайт курсу - https://e-learning.lnu.edu.ua/course/view.php?id=5899	2	кінець поточного тижня