

Шифр та назва спеціальності	122 Комп'ютерні науки ▾
Назва освітньої програми	Комп'ютерні науки. Штучний інтелект та управління проектами
Назва дисципліни	Якість, тестування та підтримка програмного забезпечення
Вид (основна, вибіркова)	Основна ▾
Блок дисципліни (залишити блок, що відповідає дисципліні)	5.Технології та методології розробки, управління проектами, тестування ▾
Кількість студентів (поточний рік)	18
Курс/Семестр	4/8

ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО ДИСЦИПЛІНУ

Анотація	Дисципліна спрямована на опанування теоретичних основ методики, яка реалізується через лекційний матеріал та лабораторні роботи, тестування програмних продуктів; підготувати тестувальників програмного проєкту; підготувати студентів до самостійного вивчення довідкової та науково-технічної літератури за фахом. Розглянуто взаємозв'язок та особливості окремих методів, обговорено їх можливості та обмеження, показано важливу роль методів досліджень тестування програмного забезпечення.								
Мета навчальної дисципліни	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 вебдодатків.								
Типи занять та контролю	Лекції, лабораторні роботи, самостійна робота. Підсумковий контроль – іспит								
Загальний обсяг (кредитів)	4	Лекції (занять)	32	Лабораторні (занять)	32	Практичні (занять)		Самостійна (годин)	56
Попередні дисципліни									

МАТЕРІАЛЬНО-ТЕХНІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

Комп'ютер, ноутбук, Visual Studio (або Eclipse), JMeter, Postman, Microsoft office, Trello, Jira

СТРУКТУРА ТА ЗМІСТ ДИСЦИПЛІНИ

№	Теоретична складова <i>Назва, перелік питань або анотація лекції</i>	Годин	Практична складова <i>Опис та приклад завдання, а також посилання на методичні матеріали</i>	Годин	Інструменти, засоби та технології
Тема 1 – Основи тестування та якість програмного забезпечення. Артефакти тестування					
1	Основи оцінки якості програмного забезпечення (ПЗ). Основні поняття тестування ПЗ. Поняття оцінювання якості ПЗ та тестування.	2	Вибрати свій продукт (будь-який приклад). На цьому прикладі розписати основні завдання за	2	Microsoft office

			трьома видами: Тестування, Контролю якості та Забезпечення якості		
2	Основні поняття тестування: термінологія тестування, розбіжності тестування та налагодження. Мета тестування. Життєвий цикл тестування.	2	Фази тестування та технологія тестування (Scrum-дошка).	2	Trello
3	Основні фази тестування та технологія тестування. Проблеми тестування систем. Рівні тестування ПЗ.	2	Метрики та оцінювання якості ПЗ. Покер оцінювання	2	Microsoft office
4	Класифікація тестування. Види і типи тестування. Планування тестування. Проектна та продуктна документація. Особливості процесу та технології індустріального тестування.	2	Види тестування та планування тестування, розробити перевірки для різних видів тестування, навчитися планувати тестові активності.	2	Microsoft office
5	Документування тестування, огляд та метрики. Тестування вимог. Робота з замовником. Типи та види ризиків.	2	Виявлення гарних та поганих вимог. Ризики написання неякісних вимог	2	Microsoft office
6	Класифікація критеріїв вибору тестів. Структурні критерії тестування ПЗ. Розробка тестового покриття. Розробка тест-плану.	2	Створення тест-кейсів для тестування вимог за правилами за темою занятті.	2	Microsoft office
7	Структура та розробка тест-кейсів. Техніки тест-дизайну. Класи еквівалентності та граничні значення. Практичні навички та скіли.	2	Створення тест-кейсів для тестування ПЗ за темою занятті.	2	Microsoft office, TestRail
8	Основні баг трекінгові системи і створення баг репорту. Основні правила занесення багів та характеристики.	2	Створення баг репортів в баг трекінгові системи при тестуванні вебзастосунка за темою занятті.	2	Microsoft office, Trello, Jira, TestRail

Тема 2 – Основи автоматизованого тестування. Фреймворки автоматизованого тестування

9	Основні підходи та методи до автоматизованого тестування. Тестування продуктивності. Навантажувальне тестування.	2	Створення тестових випадків для тестування ПЗ продуктивності за темою занятті.		Visual Studio (або Eclipse)
10	Фреймворки Unittest, pytest. Test fixtures. Assertions. Exception, errors, warnings. Skipping tests. Parametrized test module.	2	Створення тестових випадків для тестування ПЗ навантаженням за темою занятті. Застосування JMeter.	2	Visual Studio (Python або Eclipse). JUnit, NUnit, pytest, JMeter
11	Модульне тестування систем. Інтеграційне тестування. Особливості інтеграційного тестування систем. Регресійне тестування.	2	Створення класів для модульного тестування для розроблених класів за темою занятті. Інтеграційне тестування. Створення класів тестування за темою занятті.	4	Visual Studio (або Eclipse)
12	Набори тестів (Test suites). Test Runner. Logger. Рівні логування. Mocks. Огляд SQL / No-SQL (Redis та MongoDB). Mock-и для тестування баз даних. SQLAlchemy.	2	Робота з SQL. Тестування баз даних (функціональне та нефункціональне).	2	Visual Studio (або Eclipse), бібліотеки SQL
13	Тестування API. Клієнт-серверна архітектура. Теорія API. REST/SOAP. HTTP методи. Фреймворки для тестування API (Swagger, Postman, SoapUI).	2	Створення тестових випадків для тестування API ПЗ за темою занятті. Застосування Postman (Swagger)	2	Postman, Swagger

14	Selenium WebDriver. Locators. XPath, CSS, DOM. Selenium команди. POM. Data -driven testing.	2	Написання автоматизованих тестів з використанням Selenium WebDriver. Створення логіна	2	Selenium WebDriver, allure, logging
15	TDD. Test Coverage. Static code analysis. BDD. Мова Gherkin. Behave. RobotFramework	2	Розробка тестових сценаріїв на Gherkin з реалізацією кроків з behave.	2	
16	Основи тестування мобільних застосунків. Огляд основних фреймворків з тестування мобільних застосунків.	2	Створення тестових випадків для тестування мобільного додатку за темою занятті. Застосування Framework для тестування мобільних додатків	2	

ТЕМИ ТА ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

№	Назва та опис завдання	Методи контролю та критерії оцінювання	Годин
1	Вичерпне тестування. Оптимізація бізнес-технологій.		2
2	Тестування, як процес, направлений на демонстрацію коректності		2
3	Підтримка тест-кейса. Кількість ідей, очікуваних результатів у тест-кейсі.		6
4	Документація різних стадій тестування		6
5	Тестування Desktop додатків		8
6	Security testing		8
7	Тестування великих обсягів даних.		5
8	Проектна документація		5
9	Основи тестування iOS та Android mobile.		14

ПРОЄКТ (за наявністю)

№	Назва та опис завдання	Метод контролю та захисту	Строки виконання
1	Автоматизоване тестування. Розробка власного фреймворку для тестування public API. Взяти існуючу документацію і розробити framework з тестами, що розробляються безпосередньо для нього		4 тижня

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ ТА НАВЧАЛЬНІ МАТЕРІАЛИ

Основні

№	Назва	До теми (вказати номер)
---	-------	-------------------------

1	Канер Кем, Фолк Джек, Нгуєн Енг Кек Тестування програмного забезпечення. Фундаментальні концепції менеджменту бізнес-додатків. – К.: ДіаСофт – 2001. – 544 с	1-7
2	Джон Макгрегор, Девід Сайкс Тестування об'єктно-орієнтованого програмного забезпечення. Практичний посібник. — К.: ДіаСофт – 2002. — 432 с.	5-10
3	Patton R. Software Testing [Text] / R. Patton. -- 2nd Edn. -- Indianapolis: Sams, 2005. -- 408p.	1-7
4	Jonathan Rasmusson. Penetration Testing: A Hands-On Introduction to Hacking, 2016	
5	Arnon Axeirod. Complete guide to test automation: techniques, practices and patterns for building and maintaining effective software projects. 2018	7-8
6	Cohn M. Agile Estimating and Planning [Text] / Mike Cohn. -- Prentice Hall, 2005. -- 368 p.	5
7	Andreas Spillner, Tilo Linz, Software Testing Foundations: A Study Guide for the Certified Tester Exam-Foundation Level — ISTQB, 5th Edition, 2021. — 390 p.	8-13
8	Pajankar Ashwin, Python Unit Test Automation: Automate, Organize, and Execute Unit Tests in Python, 2022.	8-13
9	Basu, Selenium with Python Simplified For Beginners — Simple, Concise & Easy guide to Automation Testing using Python and Selenium WebDriver, 2020. — 126 p.	14-15
10	Tanveer, Oracle 19c SQL For Developers, Testers Analyst, 2021. — 473 p	12

Додаткові

№	Назва	До теми (вказати номер)
1	Білас О. Є. Якість програмного забезпечення та тестування. Навчальний посібник. Львів : Львівська політехніка, 2011. — 216 с.	1-7
2	Svyatoslav Kulikov, Software Testing. Base Course. (3rd edition), 2022. — 280 p.	1-7
3	https://behave.readthedocs.io/en/stable/gherkin.html	8-16

КОНТРОЛЬНІ ЗАХОДИ

№	Назва та опис	Методи контролю та критерії оцінювання
1	Контрольна робота з теми 1 та 2	онлайн тести (по 20%)
2	Контрольна робота з теми 3 та 4	онлайн тести (по 20%)
3	100% підсумкової оцінки складаються з результатів оцінювання у вигляді іспиту (60%) та поточного оцінювання (40%). Іспит: письмове завдання (2 запитання з теорії + розв'язання задачі) та усна доповідь.	
4		
5		

6

7

РЕЗУЛЬТАТИ НАВЧАННЯ

Знати та вміти застосовувати основні методи постановки та розв'язання задач системного аналізу в умовах невизначеності цілей, зовнішніх умов та конфліктів. Проєктувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах

ЗВ'ЯЗОК ІЗ РИНКОМ ПРАЦІ

Спеціальність/професія, підготовці до діяльності в якій читається курс

Бізнес-аналітик ▾ QA Engineer

Посилання на вакансії (понад 3), та їх скріншоти, в яких вказані компетенції, що формуються на курсі, як вимоги до кандидатів

<https://www.work.ua/jobs/5644805/>

Обов'язки:

- Проведення тестових активностей;
- Забезпечення відповідності продукту стандартам якості;
- Аналіз та тестування вимог;
- Розробка та підтримка актуальності тест дизайну;
- Створення артефактів тестування, передача результатів керівництву.

Наші вимоги:

- Володіння Test Management Tools, Project Management Tools, Wiki (JIRA, Redmine, Trello, Confluence, Google Docs etc.);
- Досвід роботи з debugging tools (Android Studio, iTunes, iTools, Dev Console, Charles, Fiddler etc.);
- Навички у розробці та підтримці актуальної тестової документації;
- Знання та розуміння основних принципів клієнт-серверної архітектури додатків;
- Знання та розуміння процесу розробки та життєвого циклу ПЗ;
- Вміння працювати в режимі мультизадачності;
- Проактивність, швидке навчання і бажання розвиватися.

<https://www.work.ua/jobs/5312605/>

Обов'язки

- тестування API
- написання та підтримка в актуальному стані тест кейсів та чек листів
- налаштування Quality Gates
- аналіз і тестування вимог
- комунікація з власниками продукту та бізнес-аналітиками

Перелік компетентностей із вказаних як вимоги до вакансії, які набувають студенти, в процесі проходження дисципліни.

Здатність до абстрактного мислення, аналізу та синтезу.

Здатність застосовувати знання у практичних ситуаціях

Здатність планувати і управляти часом

Знання та розуміння предметної області та розуміння професійної діяльності

Здатність до пошуку, оброблення та аналізу інформації з різних джерел

Здатність генерувати нові ідеї (креативність)

Здатність оцінювати та забезпечувати якість виконуваних робіт

Здатність організувати роботу з аналізу та проєктування тестування програмних систем, створення відповідних інформаційних технологій та програмного забезпечення.

Інструменти оцінювання результатів навчання за дисципліною*

№	Об'єкт оцінювання (знання методів та принципів, практичні навички, командна робота тощо)	Методи контролю (тести, виконання поточних практичних завдань та їх форма: написання коду, створення діаграми Гантта, створення прототипу тощо)	Інструмент оцінювання (доступ до результатів тесту, гостьова лекція, посилання на виконані завдання, посилання на проєкт, присутність на захисті проєктів, доступ до запису захисту тощо)
1		<ul style="list-style-type: none">• тести за темою (аналогічно технічному інтерв'ю)	<ul style="list-style-type: none">• підсумковий проєкт (код, презентація, скринкаст презентації тощо)

2		<ul style="list-style-type: none"> ● поточні завдання впродовж курсу (посилання на код на GitHub-репозитарію студента тощо) 	<ul style="list-style-type: none"> ● гостьові активності (гостьові лекції, відвідування контрольних активностей тощо).
3			
4			
5			
6			
7			
8			
9			
10			

***Блок. Інструменти оцінювання результатів навчання за дисципліною** — цей блок має на меті показати можливі варіанти проведення валідації за участі експертів від ІТ-компаній. Основна умова проведення валідації це попереднє погодження всіх аспектів з експертом та викладачем. Інструменти, завдяки яким це можна зробити:

- тести за темою (аналогічно технічному інтерв'ю);
- поточні завдання впродовж курсу (посилання на код на GitHub-репозитарію студента тощо) ;
- підсумковий проєкт (код, презентація, скринкаст презентації тощо);
- гостьові активності (гостьові лекції, відвідування контрольних активностей тощо).