

**ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ**

Система забезпечення якості освітньої діяльності та якості вищої освіти
сертифікована на відповідність ДСТУ ISO 9001:2015 / ISO 9001:2015

Кафедра інженерії програмного забезпечення та кібербезпеки

ЗАТВЕРДЖЕНО
вченою радою ДТЕУ
(пост. п. 7 від «21» 02 2023 р.)
Ректор  Анатолій МАЗАРАКІ



**АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ПРОГРАМНИХ
ПРОДУКТІВ/
TEST AUTOMATION SOFTWARE PRODUCT**

**ПРОГРАМА /
COURSE SUMMARY**

Київ 2023

Розповсюдження і тиражування без офіційного дозволу ДТЕУ заборонено

Автори: Т.О. ЖИРОВА, кандидат педагогічних наук, старший викладач кафедри інженерії програмного забезпечення та кібербезпеки
Н.О. КОТЕНКО, кандидат педагогічних наук, доцент кафедри інженерії програмного забезпечення та кібербезпеки
В.Р. ЯРЕМИЧ, асистент кафедри інженерії програмного забезпечення та кібербезпеки

Програму розглянуто і затверджено на засіданні кафедри інженерії програмного забезпечення та кібербезпеки «30» травня 2023 р., протокол №36.

Рецензенти: ДЕСЯТКО А.М., PhD., доц.кафедри інженерії програмного забезпечення та кібербезпеки
ЖИРОВ Д.М., к.т.н, с.н.с. ІЕЗ ім. Є.О. Патона НАНУ,
Software Engineer – Testing компанії «EPAM»

АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ПРОГРАМНИХ ПРОДУКТІВ/ TEST AUTOMATION SOFTWARE PRODUCT

ПРОГРАМА / COURSE SUMMARY

ВСТУП

Програма дисципліни «Автоматизоване тестування програмних продуктів» призначена для студентів денної та заочної форми навчання за освітнім ступенем «магістр» галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення», ОПП «Управління проектами програмних продуктів».

Програму підготовлено відповідно до Стандарту вищої освіти України для спеціальності 121 «Інженерія програмного забезпечення» (Нак. МОН №1424 від 17.11.20) та освітньо-професійної програми підготовки магістрів спеціальності 121 «Інженерія програмного забезпечення».

Програма складається з таких частин:

1. Мета, завдання та предмет дисципліни.
2. Передумови вивчення дисципліни як вибіркової компоненти освітньої програми.
3. Результати вивчення дисципліни.
4. Зміст дисципліни.
5. Список рекомендованих джерел.

1. МЕТА, ЗАВДАННЯ ТА ПРЕДМЕТ ДИСЦИПЛІНИ

Мета дисципліни «Автоматизоване тестування програмних продуктів»: засвоєння необхідних знань щодо сучасних інструментальних засобів тестування ПП, технологій побудови автоматизованих тестів, а також набуття практичних навичок з використання сучасних методів та засобів автоматизованого тестування програмного забезпечення.

Завдання дисципліни «Автоматизоване тестування програмних продуктів»: теоретична та практична підготовка майбутніх фахівців з таких питань:

- класифікація задач, які підлягають автоматизації тестування;
- налагодження та використання інструментальних засобів автоматизованого тестування ПП;
- розробка автоматизованих тестів;

Предметом дисципліни є сукупність теоретичних і практичних проблем, які пов'язані з інструментальними засобами та методами реалізації автоматизованого тестування.

2. ПЕРЕДУМОВИ ВИВЧЕННЯ ДИСЦИПЛІНИ ЯК ВИБІРКОВОЇ КОМПОНЕНТИ ОСВІТНЬОЇ ПРОГРАМИ

Знання: основ програмування мовами C#/Java, знання ООП, основи HTML та CSS, принцип роботи HTTP протоколу, основи SQL.

Вміння: працювати з офісними програмами MS Word, MS Excel (створення, редагування, рецензування, робота з документами в online режимі); ґрунтовне володіння ООП; створювати для готових БД запити мовою SQL.

3. РЕЗУЛЬТАТИ ВИВЧЕННЯ ДИСЦИПЛІНИ

Дисципліна «Автоматизація тестування програмних продуктів» як обов'язкова компонента освітньої програми, забезпечує оволодіння здобувачами вищої освіти загальними та фаховими компетентностями і досягнення ними програмних результатів навчання за відповідною освітньо-професійною програмою:

Управління проектами програмних продуктів (ОС магістр)

Номер в освітній програмі	Зміст компетентності	Номер теми, що розкриває зміст компетентності
<i>Загальні компетентності за освітньою програмою</i>		
ЗК01.	Здатність до абстрактного мислення, аналізу та синтезу.	3-12
ЗК02.	Здатність спілкуватися іноземною мовою як усно, так і письмово.	1-12
ЗК03.	Здатність проводити дослідження на відповідному рівні.	5, 6
ЗК05.	Здатність генерувати нові ідеї (креативність).	4-10
<i>Фахові компетентності за освітньою програмою</i>		
СК01.	Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.	1, 2, 3
СК07.	Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах.	4, 5, 6
СК08.	Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та	4, 5, 11

	технологій розроблення програмного забезпечення.	
СК09.	Здатність забезпечувати якість програмного забезпечення.	1-9
<i>Програмні результати навчання за освітньою програмою</i>		
РН01	Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення	1, 2, 5
РН02	Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.	4, 12
РН11	Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення.	1-12
РН16	Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення.	1-12

4. ЗМІСТ ДИСЦИПЛІНИ

Тема 1. Вступ до автоматизованого тестування програмних продуктів

Основні поняття та принципи автоматизації тестування. Тестові сценарії (Test Scripts). Інструменти автоматизації (Automation Tools), особливості та області застосування. Принципи автоматизації тестування програмних продуктів: модульність (Modularity), повторне використання (Reusability), незалежність даних (Data Independence), прозорість та чіткість (Transparency and Clarity).

Переваги та недоліки автоматизації тестування. Виклики автоматизації тестування програмних продуктів.

Список рекомендованих джерел:

Основний: 2 [с. 13-52]

Додатковий: 8 [с. 21-46], 11 [с. 1-96], 12 [с. 12-64]

Інтернет-ресурси: 14, 16, 17, 18, 19

Тема 2. Тестові артефакти

Поняття Check-list, правила його створення. TestCase і його життєвий цикл. Атрибути (поля) testcase. Інструментальні засоби управління тестуванням. Властивості якісних testcase. Набори testcase. Логіка створення ефективних перевірок. Типові помилки під час розробки check-list, testcase і наборів test cases. Особливості test cases в автоматизації.

Основна термінологія: помилки, дефекти, збої, відмови тощо. Звіт про дефект та його життєвий цикл. Атрибути (поля) звіту про дефекти. Інструментальні засоби керування звітами про дефекти. Властивості якісних дефектів. Логіка створення ефективних звітів про дефекти. Типові помилки під час написання звітів про дефекти.

Список рекомендованих джерел:

Основний: 2 [с. 52-79]

Додатковий: 8 [с. 46-61], 12 [с. 64-93]

Інтернет-ресурси: 14, 16, 17, 18, 19

Тема 3. Технології автоматизації тестування програмних продуктів

Піраміда тестування. Рівні тестування: Unit tests; Integration tests; GUI tests; Manual tests. Складові частини автоматизації тестування. Обмеження автоматизації тестування. Підходи до автоматизації тестування: тестування на рівні коду і GUI-тестування.

Аналітичний огляд еволюції високорівневих технологій тестування. Підходи автоматизації тестування: частинні рішення, тестування під

керуванням даними (DDT), тестування під керуванням ключовими словами (KDT), використання фреймворків, запис та відтворення, тестування під керуванням поведінкою.

Список рекомендованих джерел:

Основний: 2 [с. 79-112]

Додатковий: 8 [с. 61-84], 10 [с. 34-100]

Інтернет-ресурси: 14, 16, 17, 18, 19

Тема 4. Інструментальні засоби роботи з проєктом

Робота з системами контролю версіями Git. Коротка історія Git. Установка та налаштування Git. Основи Git: створення Git-репозиторію, запис змін в репозиторії, перегляд історії комітів, робота з віддаленим репозиторієм. Розгалуження в Git. Git на сервері. Розподілений Git. Налаштування і конфігурація облікового запису GitHub, супровід проєкту. Інструменти та налаштування Git.

Вступ в Enterprise-розробку. Основи Maven. Файл POM. Життєвий цикл збірки проєктів. Профілі збірки. Репозиторії. Плагіни. Налаштування Maven Environment. Засоби обробки проєктів з використанням Maven. Збірка і тестування проєкта.

Системи підтримки безперервної інтеграції TeamCity, Hudson (Jenkins), TFS, Atlassian Bamboo та їх порівняння. Jenkins – призначення та налаштування Tomcat. Налаштування Jenkins і Maven, конфігурація Maven. Налаштування збірки. Модульне тестування, автоматизація тестування, звітність. GitLab CI/CD: система безперервної інтеграції в репозиторії. Azure DevOps: безперервна інтеграція для Microsoft-стеку.

Список рекомендованих джерел:

Основний: 2 [с. 153-160]

Додатковий: 7 [с. 137-159]

Інтернет-ресурси: 14, 20

Тема 5. Планування та стратегія автоматизації тестування програмних продуктів

Розробка стратегії автоматизації. Визначення цілей автоматизації: чітке формулювання цілей, вимірювані цілі, аналіз поточних процесів тестування, оцінка ручних процесів, виявлення слабких місць.

Вибір інструментів для автоматизації: огляд ринку, оцінка інструментів. Розробка архітектури автоматизації: структура тестового середовища, інтеграція з існуючими системами.

Планування ресурсів: людські ресурси, технічні ресурси. Створення документації: стратегія автоматизації, план тестування. Вибір тестів для автоматизації. Визначення критеріїв для вибору тестів: частота виконання, критичність функціональності, час та ресурси.

Класифікація тестових сценаріїв: функціональні тести, нефункціональні тести. Пріоритезація тестів. Аналіз повернення інвестицій (ROI): вартість автоматизації, економія часу, якість продукту.

Виконання пілотних тестів та аналіз результатів, розширення покриття.

Список рекомендованих джерел:

Основний: 2 [с. 181-213]

Додатковий: 7 [с. 160-173], 9 [с. 271-298]

Інтернет-ресурси: 14, 18, 19, 23, 25, 26

Тема 6. Основи написання автоматизованих тестів. Модульне тестування та розробка через тестування

Основи автономного тестування. Вибір мови програмування. Структура автоматизованих тестів.

Розробка через тестування. Каркаси автономного тестування, каркаси сімейства xUnit. Фреймворк NUnit/JUnit. Установка та налаштування NUnit/JUnit, використання їхніх атрибутів. Властивості NUnit/JUnit. Клас Assert. Стилiстичне оформлення тестового коду. Рефакторинг – параметризовані тести. Перевірка очікуваних виключень. Визначення категорій тестів. Використання заглушок для розриву залежностей. Тестування взаємодії за допомогою підставних об'єктів. Ієрархія і організація тестів.

Список рекомендованих джерел:

Основний: 3 [с. 1-304]

Додатковий: 5 [с. 4-320], 12 [с.105-134], 13 [с. 93-98]

Інтернет-ресурси: 22, 23, 24

Тема 7. Автоматизація тестування веб-додатків

Використання Selenium для автоматизації тестування веб-додатків.

Огляд Selenium: коротка історія, компоненти (Selenium WebDriver, Selenium Grid, Selenium IDE). Переваги використання Selenium: Відкритий вихідний код, підтримка багатьох мов програмування (Java, Python, C#, Ruby, JavaScript), сумісність з різними браузерами та операційними системами.

Налаштування середовища. Встановлення Selenium WebDriver: Завантаження та налаштування необхідних бібліотек та драйверів для різних браузерів (ChromeDriver, GeckoDriver для Firefox). Налаштування проекту: Створення проекту в IDE (IntelliJ IDEA, Eclipse, Visual Studio Code), підключення необхідних бібліотек через Maven або Gradle.

Основи написання тестів на Selenium. Локатори в Selenium: використання різних стратегій для знаходження елементів на веб-сторінці (By.id, By.name, By.className, By.xpath, By.cssSelector).

Автоматизація основних сценаріїв. Навігація та взаємодія з елементами, чекери та затримки.

Розширені можливості Selenium. Обробка спливаючих вікон та діалогових вікон. Робота з фреймами та вкладками. Знімки екрану.

Інтеграція з тестовими фреймворками: JUnit/TestNG та Page Object Model (POM).

Тестування кроссбраузерної сумісності. Інструменти для кроссбраузерного тестування: Selenium Grid, Хмарні сервіси. Налаштування середовища для кроссбраузерного тестування. Створення та виконання кроссбраузерних тестів. Аналіз результатів та усунення несправностей.

Список рекомендованих джерел:

Основний: 2 [с. 312-367]

Додатковий: 6 [с. 23-40]

Інтернет-ресурси: 14, 17, 19, 21

Тема 8. Автоматизація тестування мобільних додатків

Інструменти для автоматизації тестування мобільних додатків. Appium, його архітектура, переваги та недоліки. Espresso, його архітектура, переваги та недоліки. Особливості тестування на різних платформах: Android. Емулятори та реальні пристрої. Розробка кроссплатформених тестів. Контейнеризація та віртуалізація. Паралельне тестування. Безперервна інтеграція.

Список рекомендованих джерел:

Основний: 2 [с. 367-412]

Додатковий: 6 [с. 23-40]

Інтернет-ресурси: 14, 17, 19, 21

Тема 9. Тестування API та мікросервісів

Види тестів в тестуванні API: оглядове тестування, юзабіліті тестування, тестування безпеки, тестування безпеки, автоматизація тестування, тестування документації.

Дефекти, які виявляються під час API тестування: збій обробки помилкових умов; невикористані прапори; відсутній або дублюється функціонал; питання надійності: труднощі при підключенні і отриманні відповіді від API; проблеми з безпекою; питання багатопоточності; проблеми з продуктивністю: час відгуку API дуже високо; помилкові дефекти; некоректна обробка валідних значень; дані відповіді некоректно структуровані (JSON або XML).

Введення в архітектуру мікросервісів. Визначення та характеристики: розуміння того, що таке мікросервіси, чим вони відрізняються від монолітних архітектур, а також їхні ключові

характеристики, такі як децентралізоване керування даними, незалежне розгортання та масштабованість. Стратегії тестування мікросервісів.

Інструменти для роботи з API: Postman, jMeter, Fiddler, SoapUI, Runscope, Advanced REST Client.

Список рекомендованих джерел:

Основний: 2 [с. 412-453]

Додатковий: 6 [с. 23-40]

Інтернет-ресурси: 14, 17, 19, 21

Тема 10. Тестування продуктивності та навантаження

Тестування продуктивності. Види тестування продуктивності: Load Testing (тестування навантаження), тестування стабільності, тестування відмовостійкості, тестування відновлення, стресове тестування, тестування масштабованості, тестування конфігурації.

Етапи тестування продуктивності: аналіз системи і вимог, підготовка стратегії, налаштування генератора навантаження, моніторинг серверів та генератора навантаження, підготовка тестових даних, розробка скрипів навантаження, тестування, аналіз результатів і підготовка звітів.

Інструменти тестування продуктивності: JMeter, LoadRunner, Gatling, Taurus, Java Microbenchmark Harness, BenchmarkDotNet, Google Lighthouse

Список рекомендованих джерел:

Основний: 2 [с. 453-472]

Додатковий: 6 [с. 23-40]

Інтернет-ресурси: 14, 17, 19, 21

Тема 11. Розширені техніки автоматизації тестування

Вступ до ШІ та машинного навчання в тестуванні. Основні концепції штучного інтелекту (ШІ) та машинного навчання (МН). Види задач, які можуть вирішувати ШІ та МН у контексті автоматизації тестування.

Застосування ШІ та МН для генерації тестових випадків. Використання алгоритмів машинного навчання для аналізу вихідного коду і автоматичного створення тестових випадків. Приклади інструментів та фреймворків, що використовують ШІ для генерації тестів.

Автоматичне визначення та виправлення дефектів. Використання ШІ для аналізу результатів тестування та визначення потенційних дефектів. Техніки самовиправлення коду за допомогою машинного навчання.

Прогнозування та пріоритизація тестових випадків. Використання моделей машинного навчання для прогнозування ризиків та визначення пріоритету тестування на основі історичних даних. Інтеграція прогнозування ризиків у процес тестування.

Аналіз тестового покриття та оптимізація тестових наборів. Використання ШІ для аналізу тестового покриття і виявлення пробілів у тестуванні. Оптимізація тестових наборів для забезпечення максимальної ефективності тестування.

Розпізнавання та автоматизація тестування інтерфейсів користувача (UI). Використання технологій комп'ютерного зору для автоматизації тестування інтерфейсів користувача. Інструменти та методи для автоматичного розпізнавання та взаємодії з елементами UI.

Автоматичне оновлення та підтримка тестових сценаріїв. Використання ШІ для автоматичного оновлення тестових сценаріїв при змінах у програмному продукті. Підтримка актуальності тестових сценаріїв та мінімізація ручної роботи.

Список рекомендованих джерел:

Основний: 2 [с. 478-515]

Додатковий: 7 [с. 289-321]

Інтернет-ресурси: 26, 27, 29

Тема 12. Гнучке тестування в Agile

Переваги гнучкого тестування. Принципи гнучкого тестування. Роль тестувальника в Scrum. Гнучка тестова діяльність в Scrum. Відмова від традиційних способів тестування на користь Agile. Маніфест Agile-тестування. Відмінності традиційного тестування від тестування в Agile. Піраміда тестування. Shift-left testing: розподіл тестових активностей в команді; практики, «змінюють» ролі в команді: тестування вимог (Specification by Example), тестування в розробці (TDD, ATDD, BDD), планування / ретроспектива в тестуванні, парні сесії (парне програмування). Тестування безпеки, продуктивності та інших атрибутів якості. Стратегія автоматизації тестування: фреймворки автоматизації тестування, паттерни і антипаттерн в автоматизації. Тестова документація.

Список рекомендованих джерел:

Основний: 1 [с. 5-113]

Додатковий: 12 [с. 2-139]

Інтернет-ресурси: 14, 15, 23, 24

5. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ¹

Основний

1. Gregory J. Agile Testing Condensed: A Brief Introduction Paperback /Janet Gregory,Lisa Crispin – September 3, 2019. – 113 p.
2. Jorgensen P. C., DeVries B. Software testing: a craftsman's approach, fifth edition. Auerbach Publishers, Incorporated, 2021. 528 p.
3. Khorikov V. Unit Testing Principles, Practices, and Patterns: Effective testing styles, patterns, and reliable automation for unit testing, mocking, and integration testing with examples in C# 1st Edition./ Vladimir Khorikov – Apress ,January 14, 2020. – 304 p.
4. Westerveld D. API Testing and Development with Postman: A practical guide to creating, testing, and managing APIs for automated software testing. Packt Publishing, 2021. 340 с.

Додатковий

5. ¹Цензура М.О. Технологія JAVA : Методичні рекомендації для самостійної роботи студентів / М. О. Цензура; Кафедра програмної інженерії та інформаційних систем; КНТЕУ; авт. Цензура М.О. – Київ : КНТЕУ, 2018 – 328 с.
6. Horalek J., Urbanik P., Sobeslav V. Automated tests using selenium framework. Lecture notes of the institute for computer sciences, social informatics and telecommunications engineering. Cham, 2023. P. 23–40.
7. Parsa S. Software testing automation: testability evaluation, refactoring, test data generation and fault localization. Springer International Publishing AG, 2023. – 348 p.
8. Rankin C. The Software Testing Automation Framework. IBM Systems Journal. 2022. Vol. 41, no. 1. P. 126–139.
9. Testing software / D. Irving та ін. Research software engineering with python. Boca Raton, 2021. С. 271–298.
10. Авраменко А.С. Тестування програмного забезпечення. / Авраменко А.С., Авраменко В.С., Косенюк Г.В. – Черкаси: ЧНУ імені Богдана Хмельницького, 2017. – 284 с.
11. Вакалюк Т.А. Технології тестування програм. Навчально-методичний посібник для студентів напряму 6.040302 Інформатика. – Житомир: Вид-во ЖДУ, 2018. – 196 с.

¹Курсивом виділені джерела, наявні в бібліотеці КНТЕУ

12. Штаєр Л.О. Технології розробки програмного забезпечення: конспект лекцій / Л. О. Штаєр. – Івано-Франківськ : ІФНТУНГ, 2017. –139 с.

13. Руденська Г. Шляхи удосконалення тестування спеціального програмного забезпечення інформаційної системи управління оборонними ресурсами на етапі експлуатації. Збірник наукових праць Центру воєнно-стратегічних досліджень НУОУ імені Івана Черняхівського. 2021. С. 93–98.

Інтернет-ресурси

14. Implementing Automated Software Testing – Continuously Track Progress and Adjust Accordingly. URL: <http://www.methodsandtools.com/archive/archive.php?id=94> (дата звернення 22.02.24 р.)

15. The Agile System Development Life Cycle (SDLC). URL: <http://www.ambysoft.com/essays/agileLifecycle.html> (дата звернення 24.02.2024р.)

16. The History of Software Testing. URL: <http://www.testingreferences.com/testinghistory.php> (дата звернення 20.02.24р.)

17. Тестування програмного забезпечення. Базовий курс. URL: http://svyatoslav.biz/software_testing_book/ (дата звернення 25.02.24 р.)

18. Онлайн-курс «Software Testing Introduction». URL: <http://svyatoslav.biz/category/education/> (дата звернення 22.02.24 р.)

19. Основи тестування ПЗ. URL: <https://training.qatestlab.com/> (дата звернення 10.01.24р.)

20. Ganna Karlun, Порівнюємо інструменти для CI/CD: Teamcity, Jenkins, Bitbucket та інші. URL: <https://dou.ua/forums/topic/32724/> (дата звернення 23.03.24 р.)

21. Коваленко Д. Термінологія: тестування інтерфейсів, веб інспектор, візуальне тестування. URL: <https://www.quality-assurance-group.com/ui-user-interface-testuvannya-korystuvatskogo-interfejsu-vid-dmytra-kovalenka/> (дата звернення 10.03.24)

22. Автоматизація тестування: як уникнути поширених помилок. URL: <https://www.globallogic.com/ua/insights/blogs/qa-automation-2/> (дата звернення 10.03.24)

23. Петрук О. В., Пастух О. А. Розробка програмного забезпечення для автоматизованого тестування Інтернет ресурсу : master's thesis. 2019. URL: <http://elartu.tntu.edu.ua/handle/lib/29470> (дата звернення: 15.06.2024).

24. Bass J. M. Testing and test automation. Agile software engineering skills. Cham, 2022. С. 251–263. URL: https://doi.org/10.1007/978-3-031-05469-3_16 (дата звернення: 15.06.2024).

25. Top 5 QA automation trends in 2024 era. QATestLab Blog | Blog About Software Testing. Quality is a Rule. URL: <https://blog.qatestlab.com/2024/01/24/title-top-5-qa-automation-trends-in-2024-era/> (дата звернення: 10.06.2024).

26. Automated testing. Encyclopedia of computer graphics and games. Cham, 2024. P. 211. URL: https://doi.org/10.1007/978-3-031-23161-2_300154 (дата звернення: 10.06.2024).

27. Artificial intelligence methods for optimization of the software testing process. Elsevier, 2022. URL: <https://doi.org/10.1016/c2021-0-00433-8> (дата звернення: 15.06.2024).

28. Chacon S. Pro Git / Scott Chacon, Ben Straub [Електронне видання] <https://github.com/progit/progit2-ru/releases/download/2.1.84/progit.pdf>

29. Трофименко О. Г., Дика А. І. Трофименко О. Г. Тестування та забезпечення якості програмних систем : навчально-методичний посібник [Електронне видання] /. Одеса : Фенікс, 2024. URL: <https://doi.org/10.32837/11300.27246> (дата звернення: 10.06.2024).

Наукові публікації відповідно до тем дисципліни:

30. Трофименко, О., Дика, А., & Лобода, Ю. (2023). АНАЛІЗ ІНСТРУМЕНТІВ ТЕСТУВАННЯ ВЕБЗАСТОСУНКІВ. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 4(20), 62–71. <https://doi.org/10.28925/2663-4023.2023.20.6271>

31. Авраменко, О. А. (2012). Організація тестування програмного забезпечення при оцінюванні його безпеки. *Eastern-European Journal of Enterprise Technologies*, 6(2(54)), 37–41. <https://doi.org/10.15587/1729-4061.2011.1923>

32. Семенов, С. Г., Liqiang, Z., Weiling, C., & Давидов, В. В. (2021). Розробка математичної моделі першого етапу тестування безпеки програмного забезпечення. *Eastern-European Journal of Enterprise Technologies*, 3(2 (111)), 24–34. <https://doi.org/10.15587/1729-4061.2021.233417>

33. Semenov, S., Sira, O., & Kuchuk, N. (2018). Розробка графоаналітичних моделей алгоритма тестування безпеки програмного забезпечення. *Eastern-European Journal of Enterprise Technologies*, 2(4 (92)), 39–46. <https://doi.org/10.15587/1729-4061.2018.127210>

34. Семенов, С. Г., Zhang, L., Cao, W., Бульба, С. С., Бабенко, В. Г., & Давидов, В. В. (2021). Розробка нечіткої GERT-моделі дослідження розповсюджених вразливостей програмного забезпечення. *Eastern-European Journal of Enterprise Technologies*, 6(2 (114)), 6–18. <https://doi.org/10.15587/1729-4061.2021.243715>

35. Р. В. Турчиняк; Н. Б. Стадник, к.т.н, Використання RPA технології для тестування програмних продуктів, Матеріали XII Міжнародної науково-практичної конференції молодих учених та студентів «Актуальні задачі сучасних технологій» – Тернопіль, 6-7 грудня 2023 року, Ст. 479

36. Антонова А. Р., & Помогаєв, Д. (2020). Аналіз основних переваг та недоліків різноманітних фреймворків автоматизації тестування. *Automation of Technological and Business Processes*, 12(4), 48-52. <https://doi.org/10.15673/atbp.v12i4.1935>

37. Щенятський, Д. (2023). СТВОРЕННЯ ТА ВИКОРИСТАННЯ ПРОГРАМНИХ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ ВЕБ-ДОДАТКІВ. *Інформаційно-комунікаційні технології в освіті*, (11). вилучено із <https://e-journals.udu.edu.ua/index.php/ikt/article/view/1340>

38. Інформаційні технології і автоматизація – 2023 / Матеріали XVI міжнародної науково-практичної конференції. Одеса, 19-20 жовтня 2023 р. - Одеса, Видавництво ОНТУ, 2023 р. – 451 с.