

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ВЕБ-ДОДАТКІВ»



| | |
|-----------------------|--|
| Ступінь освіти | бакалавр |
| Спеціальність | 121 Інженерія програмного забезпечення |
| Освітня програма | Інженерія програмного забезпечення |
| Загальний обсяг | 8 кредитів ЄКТС (240 годин) |
| Тривалість викладання | 8-й семестр (15 чверть) |
| Заняття: | |
| лекції: | 2 години/тиждень |
| лабораторні заняття: | 2 години/тиждень |
| Мова викладання | українська |

Сторінка курсу в СДО НТУ «ДП»: <https://do.nmu.org.ua/course/view.php?id=5336>

Кафедра, що викладає Програмного забезпечення комп'ютерних систем



Викладач:

Спирінцев В'ячеслав Васильович

Доцент, к.т.н.

Персональна сторінка

<https://pzks.nmu.org.ua/ua/teachers/teachers.php>

E-mail: Spirintsev.V.V@nmu.one

1. Анотація до курсу

Якість та успішність реалізації програмного проекту суттєво залежить від ефективного тестування. Тестування дозволяє зробити процес розробки програмного забезпечення прозорим і керованим для всіх учасників проекту. Автоматизоване тестування програмного забезпечення – частина процесу тестування на етапі контролю якості в процесі розробки програмного забезпечення. Воно використовує програмні засоби для виконання тестів і перевірки результатів виконання, що допомагає скоротити час тестування і спростити його процес. Веб-додатки дуже відрізняються від традиційного програмного забезпечення, оскільки сучасні веб-додатки побудовані на основі багаторівневої архітектури, включають в себе динамічне створення та інтерпретацію коду, а також реалізацію конкретного режиму взаємодії на основі навігаційної структури проекту. Однією з головних проблем автоматизованого тестування є його трудомісткість, оскільки для перевірки загальної поведінки веб-додатків, необхідно скласти комплекс методів тестування. Тому, автоматизація тестування не може бути реалізована без

відповідних інструментів. Саме вони визначають, як буде здійснюватися тестування та чи можуть бути досягнуті переваги автоматизації. Інструменти автоматизації тестування є найважливішим компонентом у інструментальному ланцюжку розробки. Попри те, що автоматизоване тестування дозволяє усунути частину рутинних операцій і прискорити виконання тестів, великі ресурси можуть витрачатися на оновлення самих тестів. В рамках курсу розглядаються такі питання: системи контролю версій (VCS). GIT; принципи проектування та архітектури програмного забезпечення; процеси тестування і розробки ПЗ; тестування документації та вимог; види та напрямки тестування; Check-list, Test Case, набори Test Cases; звіти про дефекти, оцінка трудовитрат, планування та звітність; автоматизація тестування; автоматизація тестування веб-орієнтованих додатків Selenium WebDriver; бібліотека REST Assured для тестування REST API; огляд веб-сервісів, тестування SOAP сервісів, введення в SoapUI; веб-сервіси RESTful, огляд, тестування Rest API, введення в Postman.

2. Мета та завдання курсу

Мета дисципліни – ознайомити студентів з основними підходами до методів та засобів тестування програмних систем; сформулювати принципи побудови та функціонування процесу тестування програмного забезпечення; сформулювати компетентності та надати сучасні практики щодо автоматизованого тестування веб-додатків.

Завданнями дисципліни є:

- опанування теоретико-понятійної бази курсу;
- опанування підходів, методів та засобів тестування програмних систем, принципами побудови та функціонування процесу тестування програмного забезпечення;
- ознайомлення здобувачів з сучасними практиками та інструментальними засобами щодо автоматизованого тестування веб-додатків.

3. Дисциплінарні результати навчання

Дисциплінарні результати навчання:

- знати основи принципи розробки програмного забезпечення та дизайн-патерни, основні підходи до методів та засобів тестування програмних систем, принципи побудови та функціонування процесу тестування програмного забезпечення;
- використовувати знання сучасних технологій для автоматизованого тестуванні веб-ресурсів;
- володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника;
- знати підходи щодо оцінки та забезпечення якості програмного забезпечення;
- знати і застосовувати на практиці інструментальні засоби для автоматизованого тестування програмного забезпечення.

Дисциплінарні результати навчання сформовано на основі ПРН освітньо-професійної програми «Інженерія програмного забезпечення» першого (бакалаврського) рівня вищої освіти (ПР14, ПР19, ПР20, ПР25).

4. Структура курсу

| Види та тематика навчальних занять | Внесок в загальну оцінку, % |
|--|-----------------------------|
| ЛЕКЦІЇ | 40 |
| <p>1. Системи контролю версій (VCS). GIT Вступ. Історія GIT. Основи GIT. Галуження в GIT. GIT на сервері. Розподілений GIT. GitHub.</p> | |
| <p>2. Принципи розробки програмного забезпечення та дизайн-патерни Основні принципи проектування. DRY. KISS. YAGNI. SOLID. Патерни проектування. Creational, Structural, Behavioral Patterns. Архітектура веб-додатків. Клієнт-серверна архітектура. Автоматизоване тестування.</p> | |
| <p>3. Процеси тестування і розробки ПЗ Життєвий цикл тестування ПЗ. Основні моделі розробки: Code and fix, Waterfall Model, V-model, Incremental Model, Iterative Model, Spiral Model, Chaos model, Prototype Model, RAD-Model. Гнучка методологія розробки програмного забезпечення Agile (Kanban, Scrum, XP, Lean, FDD, TDD, Cleanroom Software Engineering, OpenUP, MSF, DSDM). Основні принципи тестування.</p> | |
| <p>4. Тестування документації та вимог Загальні підходи до визначення вимог. Класифікація вимог. Аналіз і збирання вимог. Інженерія вимог. Властивості якісних вимог. Техніки тестування вимог. Типові помилки при аналізі та тестуванні вимог.</p> | |
| <p>5. Види та напрямки тестування Основні поняття тестування. Класифікація тестування. Підходи до тестування.</p> | |
| <p>6. Check-list, Test Case, набори Test Cases Поняття Check-list, правила його створення. Test Case і його життєвий цикл. Атрибути (поля) test case. Інструментальні засоби управління тестуванням. Властивості якісних test case. Набори test case. Логіка створення ефективних перевірок.</p> | |
| <p><i>Тестова контрольна робота №1 (за темами 1-6).</i></p> | 20 |
| <p>7. Звіти про дефекти. Оцінка трудовитрат, планування та звітність Основна термінологія: помилки, дефекти, збої, відмови тощо. Звіт про дефект та його життєвий цикл. Атрибути (поля) звіту про дефекти. Властивості якісних звітів про дефекти. Логіка створення ефективних звітів про дефекти. Типові помилки під час написання звітів про дефекти.</p> | |
| <p>8. Автоматизація тестування Переваги та недоліки автоматизації тестування. Галузі застосування автоматизації. Особливості автоматизованого тестування. Технології автоматизації тестування.</p> | |
| <p>9. Автоматизація тестування веб-орієнтованих додатків Інструментальні засоби для автоматизованого тестування веб-орієнтованих додатків. Введення в Selenium. Огляд Selenium WebDriver.</p> | |
| <p>10. Огляд веб-сервісів. Тестування SOAP сервісів. Введення в SoapUI.</p> | |
| <p>11. Веб-сервіси RESTful. Огляд. Тестування Rest API. Введення в Postman.</p> | |

| Види та тематика навчальних занять | Внесок в загальну оцінку, % |
|--|------------------------------------|
| <i>Тестова контрольна робота №2 (за темами 7-11).</i> | 20 |
| ЛАБОРАТОРНІ ЗАНЯТТЯ | 60 |
| Лабораторна робота 1 Системи контролю версій (VCS). GIT | |
| <i>Звіт з роботи № 1 та захист лабораторної роботи.</i> | 12 |
| Лабораторна робота 2 Автоматизація тестових сценаріїв за допомогою Selenium Web Driver | |
| <i>Звіт з роботи № 2 та захист лабораторної роботи.</i> | 12 |
| Лабораторна робота 3 Бібліотека REST Assured для тестування REST API. | |
| <i>Звіт з роботи № 3 та захист лабораторної роботи.</i> | 12 |
| Лабораторна робота 4 Автоматизація тестування Rest API. Postman | |
| <i>Звіт з роботи № 4 та захист лабораторної роботи.</i> | 12 |
| Лабораторна робота 5 Тестування веб-сервісів. SoapUI. | |
| <i>Звіт з роботи № 5 та захист лабораторної роботи.</i> | 12 |
| РАЗОМ | 100 |

5. Технічне обладнання та/або програмне забезпечення

Використовуються лабораторії кафедри програмного забезпечення комп'ютерних систем (комп'ютерне та мультимедійне обладнання). Дистанційна платформа Moodle, MS Office 365, Microsoft Teams, GIT, IntelliJ IDEA, Selenium WebDriver, Java, Maven, REST Assured, Postman.

6. Система оцінювання та вимоги

6.1. Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:

| Рейтингова шкала | Інституційна шкала |
|------------------|--------------------|
| 90 – 100 | відмінно |
| 74 – 89 | добре |
| 60 – 73 | задовільно |
| 0 – 59 | незадовільно |

6.2. Здобувачі вищої освіти можуть отримати підсумкову оцінку з навчальної дисципліни на підставі поточного оцінювання знань за умови, якщо набрана кількість балів захисту лабораторних робіт складатиме не менше 60 балів.

Теоретична частина оцінюється за результатами здачі двох контрольних тестових робіт, кожна з яких містить тестові закриті запитання з однією вірною відповіддю (максимальна кількість – 20 балів за кожною тестовою роботою).

Загалом за дві контрольні тестові роботи отримується максимум 40 балів, тобто 40% від оцінки за дисципліну.

Лабораторні роботи (п'ять робіт – у вигляді індивідуального завдання з кожної, розподіл % див. в таблиці розділу 4) виконуються у письмовому вигляді (звіт з кожної роботи оцінюється в межах балів, представлених в таблиці розділу 4, загалом лабораторні враховуються як 60% (максимум 60 балів). Отримані бали за теоретичну частину та лабораторні роботи додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за поточною успішністю здобувач вищої освіти може набрати 100 балів.

6.3. Критерії оцінювання підсумкової роботи. У випадку якщо здобувач вищої освіти за поточною успішністю отримав менше 60 балів та/або прагне поліпшити оцінку, проводиться підсумкове оцінювання.

Підсумкове оцінювання за дисципліною проводиться у вигляді комплексної контрольної роботи, яка включає запитання з теоретичної та практичної частини курсу. Білет складається з 30 тестових завдань з чотирма варіантами відповідей, одна правильна відповідь оцінюється в 2 бали (разом 60 балів) та 2 завдань з практичної частини, кожне з запитань оцінюється максимум у 20 балів (разом 40 балів), причому:

- 20 балів – відповідність еталону;
- 15 балів – відповідність еталону з незначними помилками;
- 10 балів – часткова відповідність еталону, питання повністю не розкриті;
- 5 балів – невідповідність еталону, але відповідність темі запитання;
- 0 балів – відповідь не наведена або не відноситься до теми запитання.

Отримані бали за тестові завдання та завдання з практичної частини додаються і є підсумковою оцінкою за вивчення навчальної дисципліни. Максимально за підсумковою роботою здобувач вищої освіти може набрати 100 балів.

7. Політика курсу

7.1. Політика щодо академічної доброчесності. Академічна доброчесність здобувачів вищої освіти є важливою умовою для опанування результатами навчання за дисципліною і отримання задовільної оцінки з поточного та підсумкового контролів. Академічна доброчесність базується на засудженні практик списування (виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання), плагіату (відтворення опублікованих текстів інших авторів без зазначення авторства), фабрикації (вигадування даних чи фактів, що використовуються в освітньому процесі). Політика щодо академічної доброчесності регламентується положенням "Положення про систему запобігання та виявлення плагіату у Національному технічному університеті "Дніпровська політехніка" <https://cutt.ly/MCfh5kv>

У разі порушення здобувачем вищої освіти академічної доброчесності (списування, плагіат, фабрикація), робота оцінюється незадовільно та має бути виконана повторно. При цьому викладач залишає за собою право змінити тему завдання.

7.2. Комунікаційна політика. Здобувачі вищої освіти повинні мати активовану корпоративну університетську пошту.

Усі письмові запитання до викладачів стосовно курсу мають надсилатися на університетську електронну пошту.

7.3. Політика щодо перескладання. Роботи, які здаються із порушенням термінів без поважних причин оцінюються на нижчу оцінку. Перескладання підсумкового оцінювання відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

7.4 Політика щодо оскарження оцінювання. Якщо здобувач вищої освіти не згоден з оцінюванням його знань він може опротестувати виставлену викладачем оцінку у встановленому порядку.

8. Рекомендовані джерела інформації

1. Авраменко А.С., Авраменко В.С., Косенюк Г.В. Тестування програмного забезпечення. Навчальний посібник/ А.С.Авраменко, В.С.Авраменко, Г.В. Косенюк. – Черкаси: ЧНУ імені Богдана Хмельницького, 2017. – 284 с.
2. Svyatoslav Kulikov. Software testing. Base course, 3rd Edition/ EPAM Systems, 2022.-278p.
3. Selenium documentation [Electronic resource]. – Access mode: <https://www.selenium.dev/documentation/>-Title from the screen.
4. Scott Chacon, Ben Straub. Pro Git, Second Edition/Apress, 2014.- 495p.
5. Daniel Galin Software Quality Concepts and Practice / USA.: Wiley-IEEE Press, 2018. – 711 p.
6. Claude Y. LaporteAlain Software Quality Assurance, First Edition/ USA: Wiley-IEEE Press, 2017. – 596 p.