



Силабус освітнього компонента Програма навчальної дисципліни



Якість, тестування та підтримка програмного забезпечення

Шифр та назва спеціальності
122 – Комп'ютерні науки

Інститут
ННІ Комп'ютерних наук та інформаційних
технологій

Освітня програма
Комп'ютерні науки та інтелектуальні системи

Кафедра
Програмна інженерія та інтелектуальні
технології управління (321)

Рівень освіти
Бакалавр

Тип дисципліни
Спеціальна (фахова), Обов'язкова

Семестр
5

Мова викладання
Українська, англійська

Викладачі, розробники



Орехов Сергій Валерійович

Serhii.Oriekhov@khi.edu.ua

Доктор філософії (Ph.D.), доцент, доцент кафедри програмної інженерії та інтелектуальних технологій управління

Google Scholar: <https://scholar.google.com/citations?user=OIJJoUi4AAAA>

ORCID: <https://orcid.org/0000-0002-5040-5861>

Scopus: <https://www2.scopus.com/authid/detail.uri?authorId=57210618400>

[Детальніше про викладача на сайті кафедри](#)

Загальна інформація

Анотація

Завданням дисципліни є засвоєння студентами знань та умінь, необхідних для перевірки якості програмних моделей та тестування головного артефакту циклу розробки програмного забезпечення – програмного коду. В курсі студенти отримують навички створення тестових планів та вибору тестових стратегій. Також вони закріплюють ці навички шляхом практичного використання систем управління версіями (наприклад, GitHub), інструментів структурного, функціонального та інтеграційного тестування для мов програмування Java, C#, Python, PHP.

Мета та цілі дисципліни

Формування у студентів теоретичних та практичних знань, які необхідні для роботи з сучасними моделями якості, а також тестовими стратегіями і планами, що реалізують повний цикл тестування, супроводження та оцінки якості різних артефактів життєвого циклу розробки програмного забезпечення.

Формат занять

Лекції, лабораторні роботи, самостійна робота, консультації. Підсумковий контроль – іспит.

Компетентності

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.
ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.
ЗК6. Здатність вчитися й оволодівати сучасними знаннями.
ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
ЗК9. Здатність працювати в команді.
ЗК10. Здатність бути критичним і самокритичним.
ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.
СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.

Результати навчання

ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

Обсяг дисципліни

Загальний обсяг дисципліни 120 год. (4 кредити ECTS): лекції – 16 год., лабораторні роботи – 32 год., самостійна робота – 72 год.

Передумови вивчення дисципліни (пререквізити)

Основи програмування
Основи інженерії програмного забезпечення
Архітектура ЕОМ та операційні системи

Особливості дисципліни, методи та технології навчання

Методи викладання та навчання:

інтерактивні лекції з презентаціями, дискусії, лабораторні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання.

Форми оцінювання:

письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), експрес-опитування (CAS), онлайн-тести (CAS), підсумковий/семестровий контроль у формі семестрового екзамену, відповідно до графіку навчального процесу (FAS).

Програма навчальної дисципліни

Теми лекційних занять

Тема 1. Поняття якості та надійності програмного забезпечення

Визначення поняття якості програмного забезпечення. Огляд найбільш поширених моделей надійності програмного забезпечення та аналіз можливості їх впровадження.

Тема 2. Моделі якості програмного забезпечення на рівня ІТ компанії, процесу та ІТ продукту

Модель якості ISO. Модель якості ІТ компанії СММІ. Огляд стандарту якості ІТ продукту SWEBOK.

Тема 3. Характеристики якості програмного забезпечення

Основи метричної теорії програм. Метрики якості Холстеда та МакКейба.

Тема 4. Місце процесів верифікації та тестування у життєвому циклі програмного забезпечення

Поняття верифікації та валідації програм. Процес верифікації програм. Ролі в процесі верифікації. Документація в процесі верифікації.

Тема 5. Критерії тестування (структурні, функціональні, мутаційні, випадкові та інші)

Етапи процесу тестування. Визначення доброго тесту. Структурні критерії тестів. Функціональні критерії тестів. Мутаційні критерії тестів. Рандомні критерії тестів.

Тема 6. Класифікація програмних помилок

Визначення терміну програмна помилка. Класи програмних помилок. Звіт про програмну помилку. Системи реєстрації програмних помилок.

Тема 7. Тестовий план

Структура та призначення. Алгоритм створення тестового плану. Системи контролю версій та тестування.

Теми практичних занять

Практичні заняття в рамках дисципліни не передбачені.

Теми лабораторних робіт

Тема 1. Створення тестової програмної системи для подальшого експерименту з тестування на мові програмування Java, C#, Javascript, Python, або PHP

Тема 2. Підготовка документації для створення тест-плану на мові UML та з використанням методології IDEF

Тема 3. Розрахунки значень типових метрик якості на основі розроблених артефактів програмного забезпечення

Тема 4. Використання метричної теорії програм для проведення тестових експериментів

Тема 5. Застосування структурних критеріїв тестування

Тема 6. Особливості застосування критеріїв тестування в межах ООП парадигми

Тема 7. Використання компонентів для автоматизації тестування в середовищах програмування на мові Java, C#, Javascript, Python, або PHP

Тема 8. Створення тест-планів. Контроль усунення програмних помилок за допомогою спеціального програмного забезпечення

Самостійна робота

Індивідуальних завдань не передбачено навчальним планом.

Студентам рекомендуються додаткові матеріали (відео, статті) для самостійного вивчення та опрацювання.

Література та навчальні матеріали

Основна література

1. S. B. Vardeman, J. M. Jobe. Statistical Methods for Quality Assurance. Basics, Measurement, Control, Capability, and Improvement. Springer, 2016.
2. R.J. Leach. Introduction to Software Engineering. CRC Press, 2016.
3. P. Ammann, J. Offutt. Introduction to Software Testing. Cambridge University Press. 2008.
4. D. Graham, E. Veenendaal, I. Evans, R. Black. Foundations of Software Testing. Istqb Certification. Thomson. 2018.
5. M. Pezzè, M. Young. Software Testing and Analysis: Process, Principles, and Techniques. 2008.
6. K. Naik, P. Tripathy. Software Testing and Quality Assurance. Theory and Practice. John Wiley & Sons, Inc. 2008.
7. B. Rumpe. Agile Modeling with UML. Code Generation, Testing, Refactoring. Springer, 2017.
- 8 A. Takanen, J. DeMott, C. Miller, A. Kettunen. Fuzzing for Software Security Testing and Quality Assurance. Artech House, 2018.
9. O. Filipova, R. Vilao. Software Development From A to Z. A Deep Dive into all the Roles Involved in the Creation of Software. Apress, 2018.

Додаткова література

1. Введення в програмну інженерію і управління життєвим циклом програмного забезпечення Guide to Software Engineering Base of Knowledge (SWEBOOK): Пер. з англ. С.Орлик [Електронний ресурс] - Режим доступу: sorlik.blogspot.com/.
2. D. Winkler, S. Biffl, J. Bergsmann. Software Quality. The Complexity and Challenges of Software Engineering and Software Quality in the Cloud. Springer, 2019.
3. J.F. Dooley. Software Development, Design and Coding. With Patterns, Debugging, Unit Testing, and Refactoring. Apress, 2017.
4. A. Mili, F. Tchier. Software Testing Concepts and Operations John Wiley & Sons, Inc. 2015.
5. Quality Assurance. Software Quality Assurance Made Easy. Solis Tech, 2018.

6. McCabe T.J. A Complexity Measure // IEEE Transactions on Software Engineering. – V.2, № 4, 1976. – pp.308 – 320.
7. Y. Singh. Software Testing. Cambridge University Press. 2012.
8. A. Pajankar Python Unit Test Automation. Practical Techniques for Python Developers and Testers. Apress, 2018.
9. Standard for Software Verification and Validation Plans (ANSI / IEEE standard 1012). 19. M. Hutcheson Software Testing Fundamentals. Methods and Metrics. Wiley Publishing Inc. 2003.

Система оцінювання

Критерії оцінювання успішності студента та розподіл балів

100% підсумкової оцінки складаються з результатів оцінювання у вигляді екзамену (40%) та поточного оцінювання (60%):

- 6 лабораторних робіт (по 6%);
- 2 контрольні роботи (по 12%).

Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту. Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Погодження

Силабус погоджено	08.06.2023	Завідувач кафедри Ігор ГАМАЮН
	08.06.2023	Гарант ОП Андрій КОПП