

ЯКІСТЬ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

СИЛАБУС

Шифр і назва спеціальності	122 — комп'ютерні науки	Інститут / факультет	Комп'ютерних наук і програмної інженерії
Назва програми	«Інженерія програмного забезпечення». (Innovation Campus)	Кафедра	програмної інженерії та інформаційних технологій управління
Тип програми	Освітньо-професійна	Мова навчання	Українська, англійська

Викладач

Орехов Сергій Валерійович

Serhii.Oriekhov@khpі.edu.ua



к.т.н. доцент, доцент кафедри програмної інженерії та інформаційних технологій управління. Підготував і опублікував понад 30 публікацій, 5 статей у виданнях, індексованих у Scopus. (ідентифікатор ORCID-<https://orcid.org/0000-0002-5040-5861>).
Провідний лектор з курсів: *Інформаційний пошук та семантичний веб (бакалаври) (англійською)*, *Інтернет речей (бакалаври) (англійською)*

Загальна інформація про курс

Анотація	Дисципліна «ЯКІСТЬ, ТЕСТУВАННЯ ТА ПІДТРИМКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ» є навчальною дисципліною з циклу професійної обов'язкової підготовки за спеціальністю 121 «Інженерія програмного забезпечення». Вона викладається у шостому семестрі в обсязі 90 год. (3 кредити ECTS), зокрема: лекції – 32 год., лабораторні – 32 год., самостійна робота – 84 год. У курсі передбачено два змістових модулі та одна модульна контрольна робота. Завершується дисципліна екзаменом.
Цілі курсу	Навчання студентів методології аналізу і оцінки якості програмного забезпечення, що дозволяє на етапі створення програмного забезпечення (ПЗ) вирішити такі основні завдання: обрати модель якості програмного забезпечення на рівні ІТ компанії, ІТ продукту або ІТ процесів; підготувати тест стратегію та тест план, що реалізуються в системі автоматизації тестування; обрати конкретний набір інструментів тестування та підтримки для реалізації стратегії або плану тестування.
Формат	Лекції, лабораторні роботи, самостійна робота. Підсумковий контроль — іспит.
Семестр	6

Обсяг (кредити) / Тип курсу (обов'язковий / вибірковий)	3 / Обов'язковий	Лекції (години)	32	Практичні заняття (години)	32	Самостійна робота (години)	26
---	------------------	------------------------	----	-----------------------------------	----	-----------------------------------	----

Програмні компетентності	<p>ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.</p> <p>ЗК2. Здатність застосовувати знання у практичних ситуаціях.</p> <p>ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.</p> <p>ЗК6. Здатність вчитися й оволодівати сучасними знаннями.</p> <p>ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.</p> <p>ЗК9. Здатність працювати в команді.</p> <p>ЗК10. Здатність бути критичним і самокритичним.</p> <p>ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.</p> <p>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.</p> <p>СК13. Здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп'ютерні системи і мережі передачі даних та аналізує якість роботи комп'ютерних мереж.</p>
---------------------------------	--

Результати навчання	Методи викладання та навчання	Форми оцінювання (поточне оцінювання CAS, підсумкове оцінювання FAS)
<p>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.</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>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)</p>
<p>ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).</p>	<p>Інтерактивні лекції з презентаціями, дискусії, практичні заняття, командна робота, кейс-метод, метод зворотного зв'язку з боку студентів, проблемне навчання</p>	<p>Письмові індивідуальні завдання до лабораторних робіт (CAS), оцінювання знань на лабораторних заняттях (CAS), онлайн -тести (CAS), підсумковий/семестровий контроль у формі семестрового екзамену, відповідно до графіку навчального процесу (FAS)</p>

СИСТЕМА ОЦІНЮВАННЯ

оцінювання успішності	Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	Нарахування	100% підсумкове оцінювання у вигляді
------------------------------	---	--------------------	--------------------------------------	--------------------	---

	90-100	A	відмінно	балів	іспиту (30%) та поточного оцінювання (70%). 30% іспит: семестровий екзамен, відповідно до графіку навчального процесу 70% поточне оцінювання: <ul style="list-style-type: none"> 40% оцінювання завдань на лабораторних роботах; 30% проміжний контроль (1 модульна контрольна робота)
	82-89	B	добре		
	74-81	C			
	64-73	D	задовільно		
	60-63	E			
	35-59	FX			
	0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни		

Політика курсу Студенти зобов'язані відвідувати заняття згідно розкладу та дотримуватися етики поведінки. У разі відсутності студентам необхідно буде виконати всі завдання, щоб компенсувати пропущені заняття. Участь у практичних заняттях вимагає попередньої підготовки та завчасного опрацювання всіх необхідних матеріалів для продуктивних дискусій під час заняття. Письмові завдання повинні бути подані до встановлених строків.

Структура та зміст курсу

Тема 1	Поняття якості та надійності програмного забезпечення	Лабораторна робота 1	Створення тестової програмної системи для подальшого експерименту з тестування на мові програмування Java, C#, Javascript, або PHP.	Самостійна робота	Вивчення моделей надійності програмного забезпечення та аналіз можливості їх впровадження.
Тема 2	Моделі якості програмного забезпечення на рівня IT компанії, процесу та IT продукту	Лабораторна робота 2	Підготовка документації для створення тест-плану у вигляді програмних вимог на мові UML та з використанням методології IDEF		Огляд існуючих моделей якості IT продукту
Тема 3	Характеристики якості програмного забезпечення	Лабораторна робота 3	Розрахунки значень типових метрик якості на основі розроблених артефактів програмного забезпечення		Аналіз метрик якості програмного забезпечення згідно моделі SWEBOOK
Тема 4	Місце процесів верифікації та тестування у життєвому циклі програмного забезпечення	Лабораторна робота 4	Використання метричної теорії програм для проведення тестових експериментів		Вивчення існуючих методів тестування „біла скриня”
Тема 5	Критерії тестування (структурні, функціональні, мутаційні, випадкові та інші)	Лабораторна робота 5	Застосування структурних критеріїв тестування		Вивчення існуючих методів тестування „чорна скриня”
Тема 6	Класифікація програмних помилок	Лабораторна робота 6	Особливості застосування критеріїв тестування в межах ООП парадигми		Аналіз особливостей та реалізація експерименту щодо тестування веб проекту он-лайн
Тема 7	Тестові плани. Структура та призначення. Алгоритм побудови	Лабораторна робота 7	Використання компонентів для автоматизації тестування в середовищах програмування на мові Java, C#, Javascript, або PHP		Огляд систем автоматизації тестування
Тема 8	Системи підтримки процесів верифікації та тестування програмного забезпечення	Лабораторна робота 8	Створення тест-планів. Контроль усунення програмних помилок за допомогою спеціального програмного забезпечення.		Вивчення систем контролю усунення помилок он-лайн

Література

Обов'язкова

1. A. Mili, F. Tchie. Software Testing Concepts and Operations John Wiley & Sons, Inc. 2015.
2. Y. Singh. SOFTWARE TESTING. Cambridge University Press. 2012.
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. M. Hutcheson Software Testing Fundamentals. Methods and Metrics. Wiley Publishing Inc. 2003.
- 8 Standard for Software Verification and Validation Plans (ANSI / IEEE standard 1012-1986).

Додаткова

9. С. Куликов. Тестирование программного обеспечения. Базовый курс. EPAM Systems, 2021.
10. Введення в програмну інженерію і управління життєвим циклом програмного забезпечення Guide to Software Engineering Base of Knowledge (SWEBOK): Пер. з англ. С.Орлик [Електронний ресурс] - Режим доступу: sorlik.blogspot.com/.
- 11.Г. Майерс. Надежность программного обеспечения. М.: МИР, 1980.
12. Г. Майерс. Искусство тестирования программ. М.: Финансы и статистика, 1982.
13. К.Бек. Экстремальное программирование: разработка через тестирование. – СПб.: Питер, 2003.
14. Б.Бейзер Тестирование черного ящика. Технологии функционального тестирования ПО и систем. – СПб.: 2004.
15. McCabe T.J. A Complexity Measure // IEEE Transactions on Software Engineering. – V.2, № 4, 1976. – pp.308 – 320.
16. С. Канер. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес приложений. К.: Диасофт, 2001.
17. Л. Тамре. Введение в тестирование программного обеспечения. М.: Вильямс, 2003

Норми академічної етики

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність

Силабус за змістом повністю відповідає робочій програмі курсу.