

**Інформаційний пакет освітніх компонент навчального плану
освітньо-професійної програми «Комп'ютерні науки»**
(назва)

Освітнього рівня бакалавр

Спеціальності 122 Комп'ютерні науки та інформаційні технології (Комп'ютерні науки)

Галузь знань 12 «Інформаційні технології»

1. Назва освітньої компоненти Якість програмного забезпечення та тестування
(назва дисципліни)

2. Тип вибіркова

3. Обсяг:	Кредитів ECTS	Годин	За видами занять:				
			Лекцій	Семинар	Практичних занять	Лабораторних занять	Самостійна підготовка
	5	150	18		18	18	96
4. Взаємозв'язок у структурно-логічній схемі							
Освітні компоненти, які передують вивченню	<ol style="list-style-type: none"> 1. Прикладне програмування–JAVA. 2. Групова динаміка і комунікації. 3. Вища математика. 4. Прикладні алгоритми та структури даних. 5. Програмування мобільних пристроїв. 						
Освітні компоненти для яких є базовою	<ol style="list-style-type: none"> 1. Спеціалізовані мови програмування. 2. Мережне програмування. 3. Серверне програмування. 4. Штучний інтелект. 						
5. Компетенції відповідно до ОПШ та вимог роботодавців:							
Компетенції відповідно до ООП							
<p>ПП5. Здатність до опанування сучасної мови програмування Java Enterprise Edition та широкого спектру мов програмування; оволодіння навичками програмування, методами проектування та створення програм згідно сучасних технологій програмування, розробляти документацію до програмного проекту та реалізувати положення проекту на практиці.</p> <p>ППк3. Володіння технологіями з проектування розробки та тестування програмного забезпечення для сучасних технічних засобів, інформаційних систем, Інтернету речей (IoT) із застосуванням різних парадигм програмування.</p>							

Компетенції відповідно до вимог роботодавців

Знати	Вміти
<p>1. Основні міжнародні та національні стандарти з якості програмного забезпечення (ПЗ).</p> <p>2. Принципи побудови процесу тестування ПЗ, його цілі та місце в процесі розробки ПЗ.</p> <p>3. Методи тестування ПЗ.</p> <p>4. Структуру звіту про помилки.</p>	<p>1. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.</p> <p>2. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.</p> <p>3. Здатність розробляти архітектури, модулі та компоненти програмних систем.</p> <p>4. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.</p> <p>5. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>6. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.</p> <p>7. Здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.</p> <p>8. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.</p> <p>9. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.</p> <p>10. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.</p>

6. Результати навчання відповідно до ОПП

ПРН10. Застосовувати теорію алгоритмів, чисельні методи розрахунків та математичні методи дослідження операцій для вирішення практичних задач, будувати алгоритми та математичні моделі проблемних ситуацій.

ПРН16. Аналізувати завдання з програмування і розробляти постановку задачі, підбирати типові алгоритми та структури даних і використовувати їх при алгоритмізації обчислювальних задач, складати власні алгоритми і структури даних, проектувати структуру програми, користуючись прийомами структурного програмування, проектувати компоненти програмного забезпечення, здійснювати написання коду, передбачати появу додаткових викликів при використанні архітектури програм, що орієнтована на події, узагальнювати функціональність класів для використання компонентного підходу при розробці програм, формувати та здійснювати систематичні процедури тестування програм щодо відповідності проектним специфікаціям.

7. План вивчення освітньої компоненти

Змістовний розділ	Вид заняття	Тема	Знати	Вміти	План заняття	Лекція, методична розробка
Розділ 1						
	Лекція 1	Тема: Основи якості програмного забезпечення	особливості програмного продукту та програмного проекту, суть якості програмного продукту та методи управління нею, стандарти в інженерії якості, інструменти аналізу якості.	розбивати проект на етапи, забезпечувати належну якість програмного продукту, розробляти форми документів процесу контролю якості, описувати процеси підвищення якості, складати план якості.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Лекція 2	Тема: Характеристики і моделі якості	метрики як основу вимірювання, класифікацію мір якості та метрик якості, ключові метрики для контролю розробки програмного забезпечення, узагальнені моделі якості, класифікацію моделей оцінки надійності.	реалізувати і застосовувати метрики як основу вимірювання, застосовувати ключові метрики для контролю розробки програмного забезпечення, узагальнені моделі якості та моделі оцінки надійності.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Лекція 3	Тема: Процеси управління якістю програмного забезпечення	оцінку управління, технічні оцінки, прогонки, аудити, мету і задачі верифікації та валідації, управління верифікацією та валідацією, класифікацію методів перевірки програмних систем.	здійснювати перевірку тексту програм за столом і наскрізь; розробляти вимоги до робочих продуктів, що проходять процес інспектування; здійснювати процеси перевірки в життєвому циклі та їх призначення.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Лекція 4	Тема: Якість у життєвому циклі розробки	моделі життєвого циклу (ЖЦ) та стандарти розробки	визбирати стратегії для планування життєвого	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

		програмного забезпечення	ПЗ; етапи життєвого циклу програмного забезпечення їх види та способи використання.	циклу ПЗ; обирати та використовувати моделі ЖЦ і стандарти розробки ПЗ.		
	Лекція 5	Тема: Вступ до тестування програмного забезпечення	модель та структуру процесу тестування, ефективність тестування, теоретичні і практичні обмеження, зв'язок тестування з іншими видами діяльності.	створювати групи з тестування, аналізувати ризики, визначати об'єкти тестування, розробляти план тестування та тестів; розподіляти обов'язки по виконанню процесу тестування.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Лекція 6	Тема: Види і рівні тестування	автономне та інтеграційне тестування, тестування програмного забезпечення системи, системне тестування, модульне тестування, тестування методами «білого ящика», «сірого ящика» і «чорного ящика», альфа- і бета-тестування, регресійне і повторне тестування, функціональне та нефункціональне тестування.	застосовувати і розрізняти рівні і види тестування, обирати інструментальні засоби і проводити тестування та валідацію ПЗ.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Лекція 7	Тема: Техніки тестування	класифікацію технік тестування; техніки, що базуються на досвіді й інтуїції, техніки, що базуються на специфікації, техніки, що базуються на аналізі коду, техніки, орієнтовані на код	проводити тестування веб-застосовань, графічного інтерфейсу користувача, систем реального часу, критичних систем; вибирати інструменти тестування, методи аналізу показників	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

			та ті, що базуються на блок-схемі.	функціональності та вимірювання, що базуються на концепції функціонального розміру.		
Лекція 8	Тема: Аналіз результатів тестування		Аналіз причин і наслідків відмови, підготовку рішень і рекомендацій по результатах тестування, структуру звіту про результати тестування, структуру циклу управління якістю, поняття статистичного управління процесами.	вимірювати результати тестування, аналізувати причин і наслідків відмови, розробляти рішення і рекомендацій по результатах тестування, звіт про результати тестування; пропонувати перевірку рішень і звіту групою з тестування, розробниками і замовниками.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Лекція 9	Тема: Методології покращення якості в сучасній парадигмі		процеси розробки та методології підготовки програмних рішень, розробки в Agile-методологіях; екстремальне програмування, методологію SCRUM.	використовувати процеси розробки та різних методологій підготовки програмних рішень; проводити порівняння методологій покращення процесу розробки.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Практичне заняття 1	Тема: Характеристики і моделі якості		метрики як основу вимірювання, класифікацію мір якості та метрик якості, ключові метрики для контролю розробки програмного забезпечення, узагальнені моделі якості, класифікацію моделей оцінки надійності.	реалізувати і застосовувати метрики як основу вимірювання, застосовувати ключові метрики для контролю розробки програмного забезпечення, узагальнені моделі якості та моделі оцінки надійності.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Практичне заняття 2	Тема: Якість у життєвому циклі розробки		моделі життєвого циклу (ЖЦ) та стандарти розробки ПЗ; етапи життєвого циклу	визбирати стратегії для планування життєвого циклу ПЗ; обирати та	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

	програмного забезпечення	програмного забезпечення їх види та способи використання.	використовувати моделі ЖЦ і стандарти розробки ПЗ.		
Практичне заняття 3	Тема: Вступ до тестування програмного забезпечення	модель та структуру процесу тестування, ефективність тестування, теоретичні і практичні обмеження, зв'язок тестування з іншими видами діяльності.	створювати групи з тестування, аналізувати ризики, визначати об'єкти тестування, розробляти план тестування та тестів; розподіляти обов'язки по виконанню процесу тестування.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Практичне заняття 4	Тема: Види і рівні тестування	автономне та інтеграційне тестування, тестування програмного забезпечення системи, системне тестування, модульне тестування, тестування методами «білого ящика», «сірого ящика» і «чорного ящика», альфа- і бета-тестування, регресійне і повторне тестування, функціональне та нефункціональне тестування.	застосовувати і розрізняти рівні і види тестування, обирати інструментальні засоби і проводити тестування та валідацію ПЗ.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Практичне заняття 5	Тема: Методології покращення якості в сучасній парадигмі	процеси розробки та методології підготовки програмних рішень, розробки в Agile-методологіях;	використовувати процеси розробки та різних методологій підготовки програмних рішень; проводити порівняння методологій покращення процесу розробки.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Лабораторне заняття 1	Тема: Техніки тестування	класифікацію технік тестування; техніки, що базу	проводити тестування веб-застосунків, графічного	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

			ються на досвіді й інтуїції, техніки, що базуються на специфікації.	інтерфейсу користувача, систем реального часу, критичних систем; вибрати інструменти тестування, методи аналізу показників функціональності та вимірювання, що базуються на концепції функціонального розміру.		
Лабораторне заняття 2	Тема: Техніки тестування		техніки, що базуються на аналізі коду, техніки, орієнтовані на код та ті, що базуються на блок-схемі.	проводити тестування веб-застосунків, графічного інтерфейсу користувача, систем реального часу, критичних систем; вибрати інструменти тестування, методи аналізу показників функціональності та вимірювання, що базуються на концепції функціонального розміру.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
Лабораторне заняття 3	Тема: Аналіз результатів тестування		Аналіз причин і наслідків відмови, підготовку рішень і рекомендацій по результатах тестування, структуру звіту про результати тестування, структуру циклу управління якістю, поняття статистичного управління процесами.	вимірювати результати тестування, аналізувати причин і наслідків відмови, розробляти рішення і рекомендацій по результатах тестування, звіт про результати тестування; пропонувати перевірку рішень і звіту групою з тестування, розробниками і замовниками.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

	Лабораторне заняття 4	Тема: Методології покращення якості в сучасній парадигмі	процеси розробки та методології підготовки програмних рішень, методологію SCRUM.	використовувати процеси розробки та методологій SCRUM для підготовки програмних рішень; проводити порівняння методологій покращення процесу розробки.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/
	Самостійна робота	Тема 1. Класифікація методів перевірки програмних систем. Тема 2. Різні види тестування Тема 3. Інструменти тестування. Тема 4. Процеси розробки та методології підготовки програмних рішень, процеси розробки в Agile-методологіях. Тема 5. Екстремальне програмування. Методологія SCRUM.	1. аналітичні методи та методи колективної перевірки. 2. структуру регресійного і повторного, функціонального тестування, тестування безпеки та ергономічності, технічних характеристик, тестування на надійність, швидкодії та конфігурації. 3. класифікацію інструментів тестування, стандартизацію методів вимірювання розміру, основні методи оцінки затрат. 4. процеси розробки та методології підготовки програмних рішень та процеси розробки в Agile-методологіях. 5. процеси екстремального програмування. Методологію SCRUM.	1. застосовувати аналітичних методів та методи колективної перевірки. 2. застосовувати різні види тестування до розробки програмного забезпечення. 3. проводити вибір інструментів тестування, визначати розміри веб-застосувань, застосовувати основні методи оцінки затрат. 4. організувати процес розробки по методології Agile. 5. організувати процес розробки процеси екстремального програмування. Методологію SCRUM.	http://dl.dut.edu.ua/	http://dl.dut.edu.ua/

8. Мова вивчення освітньої компоненти

(українська, англійська, розділи, що викладаються англійською мовою)

українська

9. Інформаційне забезпечення освітньої компоненти

Рекомендовані джерела та інші навчальні ресурси: вказати підручники, навчальні посібники не пізніше 2010 року видання, які є у нас у бібліотеці на державній мові; електронні ресурси, посилання, електронна бібліотека ДУТ, іншомовні джерела
1. Левус Є.В., Мельник Н.Б. Вступ до інженерії програмного забезпечення: навч. посіб.– Л.:Видав. Львівської політехніки, 2018. – 246с.
2. Левус Є.В., Марусенкова Т.А., Нитребич О.О. Життєвий цикл програмного забезпечення: навч. посіб.– Л.:Видав. Львівської політехніки,2017.–208с.
3. Табунщик Г.В, Кудерметов Р.К., Брагіна Т.І. Інженерія якості програмного забезпечення [Електронний ресурс]:навчальний посібник. – Запоріжжя: ЗНТУ, 2013. – 180 с
10. Методи оцінювання, підсумкові звітності за освітньою компонентою
(заліки, екзамени, курсові проекти, тестування)
Екзамен
11. Матеріально-технічне забезпечення освітньої компоненти