

Міністерство освіти і науки України
Львівський національний університет природокористування
Факультет механіки, енергетики та інформаційних технологій
Кафедра інформаційних технологій



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ

(назва навчальної дисципліни)

спеціальність: 122 Комп'ютерні науки

перший (бакалаврський) рівень вищої освіти

Робоча програма з дисципліни якість програмного забезпечення та тестування для здобувачів першого (бакалаврського) рівня вищої освіти ОПП «Комп'ютерні науки» спеціальності 122 Комп'ютерні науки

Розробник: Олег КОВАЛИШИН, к.т.н., в.о. доцента

Робочу програму схвалено на засіданні кафедри інформаційних технологій

Протокол від “12” серпня 2024 року № 1

Завідувач кафедри інформаційних технологій



(підпис)

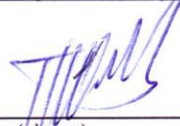
(Анатолій Тригуба)

(ім'я та прізвище)

Робочу програму схвалено на засіданні методичної комісії факультету механіки, енергетики та інформаційних технологій

Протокол від “29” серпня 2024 року № 1

Заступник голови методичної комісії факультету механіки, енергетики та інформаційних технологій



(підпис)

(Олексій ШВЕЦЬ)

(ім'я та прізвище)

1. Опис навчальної дисципліни

Галузь знань 12 Інформаційні технології

Спеціальність 122 Комп'ютерні науки

Рівень вищої освіти перший (бакалаврський)

Характеристика навчальної дисципліни:

Обов'язкова

Кількість кредитів 4

Загальна кількість годин – 120

Вид контролю: екзамен

Тижневих аудиторних годин для денної форми навчання – 3

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить (%):

для денної форми навчання – 40 %

для заочної форми навчання – 16,6 %

2. Програма навчальної дисципліни

План лекційних занять з дисципліни

1. Вступ до якості програмного забезпечення

1.1 Визначення та важливість якості програмного забезпечення

1.2 Основні аспекти якості програмного забезпечення з перспективи зацікавлених сторін та кінцевих користувачів

1.3 Виміри якості програмного забезпечення: Функціональність, Надійність, Зручність використання, Ефективність, Підтримуваність та Переносимість

2. Основи тестування програмного забезпечення.

2.1 Поняття, характеристика, мета та завдання тестування

2.3 Основні принципи проведення тестування

2.3 Основні фази життєвого циклу тестування. Типові активності життєвого циклу тестування

3. Основні типи та підходи до проведення тестування.

3.1 Типи тестування відносно об'єкту тестування

3.2 Типи тестування відносно цілей тестування

3.3 Типи тестування відносно рівня доступу

3.4 Типи тестування за ступенем автоматизації тестування

3.5 Типи тестування відносно рівня тестування

3.6 Поняття ціни якості

4. Артефакти фаз життєвого циклу тестування.

4.1 Політика тестування та тестова стратегія

4.2 План тестування

4.3 Специфікація тестового дизайну. Тестові випадки. Чеклісти. Тестові сценарії.

4.4 Звіти про дефекти.

4.5 Підсумкові звіти про проведене тестування.

5. Техніки проектування тестових випадків

5.1 Класифікація технік тестового дизайну.

5.2 Статичні техніки тестового дизайну.

- 5.3 Динамічні техніки тестового дизайну.
 5.4 Динамічні техніки тестового дизайну базовані на специфікації.
 5.5 Динамічні техніки тестового дизайну базовані на досвіді.

6. Поглиблені техніки тестування

- 6.1 Динамічні техніки тестового дизайну базовані на структурі.
 6.2 Динамічні техніки тестового дизайну базовані на моделях.
 6.3 Взаємозв'язок між техніками тестового дизайну.
 6.4 Аналіз покриття коду
 6.5 Мутаційне тестування
 6.6 Fuzz тестування

7. Основи автоматизованого тестування.

- 7.1 Сучасні проблеми класичного процесу тестування.
 7.2 Поняття автоматизації тестування.
 7.3 Етапи розвитку автоматизації тестування.

3. Структура навчальної дисципліни

Назви тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
л		п	лаб.	інд.	с. р.	л		п	лаб.	інд.	с. р.	
1	2	3	4	5	6	7	8	9	10	11	12	13
Рік підготовки 3 Семестр 6												
Тема 1.	14	4	4			6	1	1			5	1
Тема 2.	12	2	4			6	1	1			10	1
Тема 3.	12	2	4			6	1	1			11	1
Тема 4.	11	2	4			5	1	1			15	1
Тема 5.	15	2	8			5	2	2			10	2
Тема 6.	13	2	4			7	2	2			10	2
Тема 7.	13	2	4			7	2	2			9	2
Іспит	30					30	1	1			5	1
Всього	120	16	32			72	120	10	10			100

4. Теми практичних занять

План практичних занять з дисципліни

№ з/п	Назва практичної роботи	К-сть годин
1	3	4
1	Ознайомлення з основами тестування та світовими проблемами з якістю ПЗ.	8
2	Практичне визначення атрибутів та вимог до якості програмного забезпечення	8
3	Практична розробка тестового плану додатку	10
4	Побудова тестових сценаріїв та вимог	8
5	Практичне виконання тестування програмного забезпечення	8

	оформлення дефектів	
6	Базові підходи до автоматизації тестування Record-and-Play	8
	Разом за семестр	50

5. Теми винесені на самостійне вивчення

№ з/п	Назва теми
1	Тестування на основі ризику: Пріоритезація тестових сценаріїв в залежності від потенційних ризиків для бізнесу.
2	Сіре тестування: Тестування, що комбінує елементи білого та чорного ящика для ефективною перевірки.
3	Тестування середовища та конфігурації: Перевірка працездатності програмного забезпечення в різних умовах та налаштуваннях.
4	Техніки невизначеного тестування: Перевірка аспектів програмного забезпечення, які не стосуються функціональності.
5	Тестові оракули та евристика: Використання методів для визначення, чи пройшов тест, або ні.
6	Тестування зі зсувом вліво: Виконання тестування на ранніх етапах розробки для швидкого виявлення дефектів.
7	Тестування візуальної валідації: Верифікації, що інтерфейс відображається належним чином на різних пристроях та браузерах.

6. Методи навчання

1. Словесні методи: розповідь, пояснення, бесіда, лекція.

2. Наочні методи:

– ілюстрація (картинки, таблиці, моделі, муляжі, малюнки тощо);

– демонстрування (навчальне відео чи його фрагменти, інтерактивні презентації в PowerPoint, компіювання та моделювання, експеримент, спостереження, досліди і аналіз результатів тощо).

3. Практичні методи: досліди, вправи, самостійна робота. Лабораторні та практичні роботи, розрахункові, реферати. Розв'язування тестових завдань.

7. Методи контролю

1. Усне опитування: фронтальне, індивідуальне детальний аналіз відповідей студентів.

2. Письмова аудиторна та поза аудиторна перевірка: розрахункові, вирішення задач і прикладів, виконання графічних матеріалів, схем, підготовка різних відповідей, рефератів, контрольні роботи, розв'язування тестових завдань на платформі moodle тощо.

3. Практична перевірка: проведення різних вимірів, збір, систематизація та опрацювання даних, рішення професійних завдань, ділові ігри і т.д.

4. Стандартизований контроль: екзамен.

Види контролю: поточний контроль, проміжна та семестрова атестація.

8. Результати навчання

У результаті засвоєння тем із дисципліни Якість програмного забезпечення та тестування здобувачі першого (бакалаврського) рівня

вищої освіти набувають знання, уміння та компетентності, що відповідають вимогам ОП «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки».

Індекс в матриці ОПП	Програмні компоненти
1	2
ЗК01	Здатність до абстрактного мислення, аналізу та синтезу.
ЗК05	Здатність оцінювати та забезпечувати якість виконуваних робіт.
СК03	Здатність проектувати інформаційні системи з урахуванням особливостей їх призначення, неповної / недостатньої інформації та суперечливих вимог.
СК04	Здатність розробляти математичні, інформаційні та комп'ютерні моделі об'єктів і процесів інформатизації.
СК05	Здатність використовувати сучасні технології аналізу даних для оптимізації процесів в інформаційних системах.
РН01	Відшуковувати необхідну інформацію в науковій і технічній літературі, базах даних, інших джерелах, аналізувати та оцінювати цю інформацію.
РН08	Розробляти моделі інформаційних процесів та систем різного класу, використовувати методи моделювання, формалізації, алгоритмізації та реалізації моделей з використанням сучасних комп'ютерних засобів.
РН11	Розв'язувати задачі цифрової трансформації у нових або невідомих середовищах на основі спеціалізованих концептуальних знань, що включають сучасні наукові здобутки у сфері інформаційних технологій, досліджень та інтеграції знань з різних галузей.

9. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота (разом 50 балів)							Підсумковий тест (екзамен)	Сума
T1	T2	T3	T4	T5	T6	T7	50 балів	100
7	7	7	7	7	8	7		

T1, T2 ... T7 – теми

10. Методичне забезпечення

Навчально-методичні матеріали до семінарських, практичних і лабораторних занять; підручники і навчальні посібники; індивідуальні навчально-дослідні завдання; контрольні роботи; текстові та електронні варіанти тестів для поточного і підсумкового контролю, методичні матеріали для організації самостійної роботи студентів, виконання індивідуальних завдань, курсових і дипломних робіт.

11. Рекомендована література

Базова

1. Білас О.Є. Верифікація і тестування програмного забезпечення. Конспект лекцій. — Львів: Видавництво Національного університету «Львівська політехніка», 2008.
2. Авраменко А.С., Авраменко В.С., Косенюк Г.В. Тестування програмного забезпечення. Навчальний посібник. – Черкаси: ЧНУ імені Богдана Хмельницького, 2017. – 284 с.
3. Крепич, С. Я. Якість програмного забезпечення та тестування : базовий курс [Електронний ресурс] : навч. посіб. / С. Я. Крепич, І. Я. Співак. - Тернопіль : ФОП Паляниця В. А., 2020. - 479 с.

1.

Допоміжна

4. ДСТУ 2873-94. Системи обробки інформації. Програмування. Терміни та визначення. К.: Держстандарт України, 1994.
5. ДСТУ 2941-94. Системи оброблення інформації. Розроблення систем. Терміни та визначення. - К.: Держстандарт України, 1994.
6. ДСТУ 4302:2004. Інформаційні технології. Настанови щодо документування комп'ютерних програм. К.: Держстандарт України, 2004.
7. ДСТУ ISO/IEC 12119:2003. Інформаційні технології. Пакети програм тестування і вимоги до якості. К.: Держстандарт України, 2003.
8. ДСТУ ISO/IEC 14764:2002. Інформаційні технології. Супроводження програмного забезпечення. К.: Держстандарт України, 2002.
9. ДСТУ ISO/IEC 90003:2006. Програмна інженерія. Настанови щодо застосування ISO 9001:2000 до програмного забезпечення (ISO/IEC 90003:2004, IDT) К.: Держстандарт України, 2006.
10. ДСТУ ISO/IEC TR 12182:2004. Інформаційні технології. Класифікація програмних засобів (ISO/IEC TR 12182:1998, IDT) К.: Держстандарт України, 2004.
11. ДСТУ ISO/IEC 14598-1:2004. Інформаційні технології. Оцінювання програмного продукту. Частина 1. Загальний огляд (ISO/IEC 14598-1:1999, IDT) К.: Держстандарт України, 2004. ДСТУ ISO/IEC 15288:2005. Інформаційні технології. Процеси життєвого циклу системи (ISO/IEC 15288:2002, IDT) - К.: Держстандарт України, 2005.
12.). ДСТУ ISO/IEC 15939:2008. Інженерія систем і програмних засобів. Процес вимірювання. К.: Держстандарт України, 2008.
13. ДСТУ 3327-96. Методика випробування процесорів мов програмування. Загальні вимоги. К.: Держстандарт України, 1996.
14. . ДСТУ ISO/IEC TR 14369:2003. Інформаційні технології. Мови програмування, їхнє середовище та системний інтерфейс. Настанова щодо підготовки незалежних від мов специфікацій послуг. К.: Держстандарт України, 2003.

15. ДСТУ 4072:2001. Інформаційні технології. Мови програмування, їхнє середовище та системний інтерфейс. Настанова щодо підготовки незалежних від мов виклик процедур. К.: Держстандарт України, 2001.
16. ДСТУ ISO/IEC 2382-15:2005. Інформаційні технології. Словник термінів. Мови програмування (ISO/IEC 2382-15:1999, IDT) - К.: Держстандарт України, 2005.
17. ДСТУ 3008-95. "Документація. Звіти у сфері науки і техніки Структура і правила оформлення". К.: Держстандарт України, 1995. 75 с.
18. ДСТУ ГОСТ 7.1:2006. Система стандартів з інформації, бібліотечної та видавничої справи. Загальні вимоги та правила складання. К.: Держстандарт України, 2007. 47 с.
19. ДСТУ ГОСТ 2.104:2006. ЕСКД. Основні написи. К.: Держстандарт України, 2006. ДОДАТКОВА
20. Bohm, Corrado; and Giuseppe Jacopini (May 1966). "Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules". Communications of the ACM 9 (5): 366–371. doi:10.1145/355592.365646
21. Dijkstra, E. W. (Aug 1972). "The Humble Programmer". Communications of the ACM 15 (10): 859-866. doi:10.1145/355604.361591. <http://www.cs.utexas.edu/~EWD/transcriptions/EWD03xx/EWD340.html>. (EWD340) PDF, 1972 ACM Turing Award lecture
22. Dijkstra, E.W., "Structured Programming," Software Engineering Techniques, Buxton, J.N., and Randell, B., eds. Brussels, Belgium, NATO Science Committee, 1969.
23. B. Meyer, Object-Oriented Software Construction, second ed., Prentice Hall, 1997, Chap. 6, 10, 11.
24. Guide to the Software Engineering Body of Knowledge (SWEBOK). CHAPTER 4. SOFTWARE CONSTRUCTION. <http://www.computer.org/portal/web/swebok/html/ch4K>. Beck, Test-Driven Development: By Example, Addison-Wesley, 2002.
25. McCabe : Complexity Measure, IEEE Transactions on Software Engineering, Volume 2, No 4, pp 308-320, December 1976
26. M. Fowler and al., Refactoring: Improving the Design of Existing Code, Addison-Wesley, 2002. 29. Russell Gold, Thomas Hammell, Tom Snyder. Test Driven Development: A J2EE Example. Apress, 2005. 296 pages.

12. Інформаційні ресурси

1. Бібліотечно-інформаційні ресурси – книжковий фонд, періодика та фонди на електронних носіях бібліотеки ЛНУП, державних органів науково-технічної інформації, наукових, науково-технічних бібліотек та інших наукових бібліотек України.
 2. <https://www.istqb.org/>
 3. <https://www.softwaretestinghelp.com/>
 4. <https://www.ministryoftesting.com/>
 5. https://www.tutorialspoint.com/software_testing/index.htm
 6. <https://www.qasymphony.com/>
 7. <https://www.guru99.com/>

8. <https://www.softwaretestingclass.com/>
9. <https://sqa.stackexchange.com/>
10. <https://testautomationu.applitools.com/>
11. <https://www.testingexcellence.com/>
12. <https://www.qaiworldwide.org/>
13. <https://www.associationforsoftwaretesting.org/>
14. <https://www.coursera.org/>
15. <https://www.udemy.com/>
16. <https://www.edx.org/>
17. <https://www.techwell.com/>
18. <https://www.stickyminds.com/>
19. <https://www.softwaretestpro.com/>
20. <https://www.cigniti.com/>
21. <https://www.testguild.com/>