

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА  
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ  
КАФЕДРА ІНТЕЛЕКТУАЛЬНИХ ПРОГРАМНИХ СИСТЕМ**

**«ЗАТВЕРДЖУЮ»**

Заступник декана  
з навчальної роботи

\_\_\_\_\_ Кашпур О.Ф.

«\_\_» \_\_\_\_\_ 2019 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ  
МЕТОДИ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ  
ПРОГРАМНИХ СИСТЕМ  
для студентів**

галузь знань	<b>12 Інформаційні технології</b>
спеціальність	<b>121 Інженерія програмного забезпечення</b>
освітній рівень	<b>магістр</b>
освітня програма	<b>Програмне забезпечення систем</b>
вид дисципліни	<b>обов'язкова</b>

Форма навчання	<b>денна</b>
Навчальний рік	<b>2019/2020</b>
Семестр	<b>1</b>
Кількість кредитів ECTS	<b>3</b>
Мова викладання, навчання та оцінювання	<b>українська</b>
Форма заключного контролю	<b>іспит</b>

Викладач: **к. т. н., доцент Демківський Є.О.** (лекції, лабораторні заняття).

Пролонговано: на 20\_\_/20\_\_ н. р. \_\_\_\_\_ (\_\_\_\_\_) «\_\_» 20\_\_ р.

на 20\_\_/20\_\_ н. р. \_\_\_\_\_ (\_\_\_\_\_) «\_\_» 20\_\_ р.

Розробник: Демківський Євген Олександрович, к. т. н., доцент, доцент кафедри інтелектуальних програмних систем.

ЗАТВЕРДЖЕНО

Завідувач кафедри інтелектуальних програмних систем

\_\_\_\_\_ О.І. Провотар

Протокол № \_\_ від «\_\_» \_\_\_\_\_ 2019 р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «\_\_» \_\_\_\_\_ 2019 року №\_\_

Голова науково-методичної комісії \_\_\_\_\_ Л.Л. Омельчук

«\_\_» \_\_\_\_\_ 2019 року

Затверджено вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «\_\_» \_\_\_\_\_ 2019 року №\_\_

Голова вченої ради факультету \_\_\_\_\_ А.В. Анісімов

**1. Мета дисципліни** – ознайомлення студентів з актуальними практичними підходами до тестування і забезпечення якості програмного забезпечення.

**2. Попередні вимоги до опанування навчальної дисципліни:** відсутні.

**3. Анотація навчальної дисципліни.** Навчальна дисципліна «Методи забезпечення якості програмних систем» є складовою освітньо-наукової програми підготовки фахівців за другим (магістерським) рівнем вищої освіти у галузі знань 12 Інформаційні технології за спеціальністю 121 Інженерія програмного забезпечення в рамках освітньо-наукової програми «Програмне забезпечення систем».

Дана дисципліна є нормативною навчальною дисципліною в рамках освітньої програми «Програмне забезпечення систем». Викладається у **1 семестрі** в обсязі – **90 год. (3 кредити ECTS)**, зокрема: лекції – 18 год., лабораторні – 10 год., самостійна робота – 60 год. У курсі передбачено 2 змістовних частини. Завершується дисципліна – **іспитом**.

Структура курсу. В рамках вивчення дисципліни розглядаються: основні поняття забезпечення якості; ручне тестування; класифікація, локалізація та документування дефектів програмного забезпечення (ПЗ); тестова документація; основні підходи до тест-дизайну; методології оцінювання і застосування тестового покриття; багтрекінг; інструменти для автоматизації тестування; менеджмент тестування. Виступає базовою для дисципліни «Актуальні проблеми об'єктно-орієнтованого програмування».

**4. Завдання (навчальні цілі).** Основними завданнями дисципліни «Методи забезпечення якості програмних систем» є набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень в області якості програмних систем відповідно до освітньої кваліфікації магістр з інженерії програмного забезпечення. Зокрема, розвивати:

- Здатність мотивувати людей та рухатися до спільної мети, працювати в команді співробітників (ЗК-4).
- Здатність спілкуватися з представниками інших професійних груп різного рівня (з експертами з інших галузей знань/видів економічної діяльності) (ЗК-5).
- Здатність аналізувати предметні області, формувати, аналізувати та моделювати вимоги до програмного забезпечення (СК-1).
- Здатність ідентифікувати, класифікувати та описувати проектні завдання, знаходити раціональні методи й підходи до їх розв'язання (СК-2).
- Здатність проектувати програмне забезпечення, включаючи проведення моделювання його архітектури, поведінки та процесів функціонування окремих підсистем і модулів (СК-3).
- Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення (СК-4).
- Здатність оцінювати ступінь обґрунтованості застосування специфікацій, стандартів, правил і рекомендацій в професійній галузі та дотримуватися їх при реалізації процесів життєвого циклу програмного забезпечення (СК-5).
- Здатність систематизувати професійні знання щодо створення і супроводження програмного забезпечення (СК-7).
- Здатність розробляти і координувати процеси, фази та ітерації життєвого циклу програмних систем на основі застосування відповідних моделей, методів та технологій розробки програмного забезпечення (СК-8).

## 5. Результати навчання за дисципліною.

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати принципи проектування та забезпечення якості ПЗ.	Лекції, лабораторні заняття.	Тест, 60% правильних відповідей, іспит.	12%
РН1.2	Знати принципи тестування ПЗ.	Лекції, лабораторні заняття.	Тест, 60% правильних відповідей, іспит.	15%
РН1.3	Знати основні види тестування ПЗ.	Лекції, лабораторні заняття.	Тест, 60% правильних відповідей, іспит.	15%
РН2.1	Вміти використовувати методи і прийоми тестування для поширених типів ПЗ.	Лабораторні заняття, самостійна робота.	Захист лабораторної роботи, іспит.	20%
РН2.2	Вміти самостійно тестувати ПЗ як вручну, так і за допомогою сучасних інструментальних засобів автоматизації тестування.	Лабораторні заняття, самостійна робота.	Захист лабораторної роботи.	20%
РН3.1	Обґрунтовувати власний погляд на задачу, спілкуватися з колегами з питань тестування та розробки тестів, складати письмові звіти.	Лабораторні заняття.	Поточне оцінювання, захист лабораторної роботи.	5%
РН4.1	Ініціювання інноваційних комплексних проектів, лідерство та повна автономність під час їх реалізації.	Самостійна робота.	Поточне оцінювання, захист лабораторної роботи.	8%
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість.	Лабораторні заняття.	Захист лабораторної роботи.	5%

## 6. Співвідношення результатів навчання дисципліни із програмними результатами навчання.

Результати навчання дисципліни	РН1.1	РН1.2	РН1.3	РН2.1	РН2.2	РН3.1	РН4.1	РН4.2
	Програмні результати навчання							
<b>ПРН-2.</b> Обґрунтовувати вибір методів формування вимог до програмної системи, розробляти, аналізувати та систематизувати вимоги	+						+	+
<b>ПРН-5.</b> Розробляти і оцінювати стратегії проектування програмних засобів; обґрунтовувати, аналізувати і оцінювати прийняті проектні рішення з точки зору якості кінцевого програмного продукту		+	+		+			
<b>ПРН-6.</b> Аналізувати, оцінювати і обирати методи, сучасні програмно-апаратні інструментальні та обчислювальні засоби, технології, алгоритмічні та програмні рішення для ефективного виконання конкретних виробничих задач з програмної інженерії				+	+	+		
<b>ПРН-8.</b> Проводити аналітичне дослідження параметрів функціонування програмних систем для їх валідації та верифікації, а також проводити аналіз обраних методів, засобів автоматизованого проектування та реалізації програмного забезпечення		+			+			

## 7. Схема формування оцінки.

### 7.1 Форми оцінювання студентів.

#### Семестрове оцінювання:

Захист лабораторної роботи:

1. Презентація теми проекту: РН3.1., РН4.2 – **5 балів/3 бали.**
2. Розробка технічного завдання: РН3.1, РН4.2 – **5 балів/3 бали.**
3. Розробка макету інтерфейсу: РН3.1, РН4.2 – **5 балів/3 балів.**
4. Реферування літератури з тематики курсу: РН1.1, РН1.2, РН1.3 – **10 балів/6 балів.**
5. Програмна реалізація прототипу ПЗ: РН4.1, РН4.2 – **25 балів/15 балів.**
6. Оцінка якості, тестування та пробна експлуатація ПЗ: РН2.1, РН2.2 – **10 балів/6 балів.**

#### Підсумкове оцінювання (у формі іспиту):

- Максимальна кількість балів які можуть бути отримані студентом: 40 балів.
- Результати навчання які будуть оцінюватись: РН1.1, РН1.2, РН1.3, РН2.1.
- Форма проведення і види завдань: письмова робота.
- Види завдань: 20 тестових завдань.

### Критерії оцінювання на іспиті.

Завдання	Тема завдання	Вага складових у відсотках
Завдання 1-3	Ручне тестування	16%
Завдання 4-7	Дефекти	16%
Завдання 8-12	Тестова документація	20%
Завдання 13-15	«Багтрекінг»	14%
Завдання 16-18	Інструменти для автоматизації тестування	14%
Завдання 19	Інші види тестування	10%
Завдання 20	Менеджмент тестування	10%
		<b>100%</b>

### Запитання для підготовки до екзамену

1. Особливості розробки великих промислових програмних продуктів.
2. Які характеристики притаманні одноразовій стратегії конструювання ПЗ?
3. Які характеристики притаманні інкрементній стратегії конструювання ПЗ?
4. Які характеристики притаманні еволюційній стратегії конструювання ПЗ?
5. Від яких факторів залежить вибір методології проектування ПЗ?
6. Критерії успішності програмного проекту.
7. Які заходи відносяться до процедур внутрішнього забезпечення якості?
8. Які заходи відносяться до процедур зовнішнього забезпечення якості?
9. Дайте визначення поняттю «Unit-тести».
10. Дайте визначення поняттю «Code-review».
11. Дайте визначення поняттю «Менеджерське приймання».
12. Які процедури забезпечення якості ПЗ проводять в рамках функціонального тестування?
13. Які процедури забезпечення якості ПЗ проводять в рамках тестування інтерфейсу користувача?
14. Які процедури забезпечення якості ПЗ проводять в рамках тестування специфікації?
15. Особливості тестування методом «чорної скриньки».
16. Особливості тестування методом «сірої скриньки».
17. Особливості тестування методом «білої скриньки».
18. Ручне, напівавтоматизоване та автоматизоване тестування.
19. Які процедури забезпечення якості ПЗ проводять в рамках модульного тестування?
20. Які процедури забезпечення якості ПЗ проводять в рамках компонентного тестування?
21. Які процедури забезпечення якості ПЗ проводять в рамках інтеграційного тестування?
22. Які процедури забезпечення якості ПЗ проводять в рамках системного тестування?
23. Специфіка проведення тестування при прийманні.
24. В чому суть регресійного тестування?
25. Залежність вартості виправлення дефекту від етапу розробки ПЗ.
26. Дайте визначення поняттю «дефект ПЗ».
27. Перелічіть основні типи дефектів ПЗ.
28. Дайте визначення поняттю «функціональні помилки ПЗ».
29. Наведіть приклади функціональних помилок у ПЗ.

30. Дайте визначення поняттю «візуальні помилки ПЗ».
31. Наведіть приклади візуальних помилок у ПЗ.
32. Дайте визначення поняттю «логічні помилки ПЗ».
33. Наведіть приклади логічних помилок у ПЗ.
34. Дайте визначення поняттю «помилки зручності використання ПЗ».
35. Наведіть приклади помилок у зручності використання ПЗ.
36. Дайте визначення поняттю «помилки безпеки ПЗ».
37. Наведіть приклади помилок безпеки у ПЗ.
38. Навіщо документувати дефекти?
39. Перелічіть основні правила оформлення дефектів.
40. Перелічіть основні типи документів, які тестувальники складають до початку тестування.
41. Перелічіть основні типи документів, які тестувальники складають після проведення тестування.
42. Що повинен містити правильно складений тест-план?
43. Що повинен містити правильно складений чек-ліст?
44. Що повинен містити правильно складений тест-кейс?
45. Дайте визначення для поняття «Тест-дизайн».
46. Перелічіть та охарактеризуйте основні техніки тест-дизайну.
47. Для чого використовуються тестові матриці?
48. Чим займається тест-дизайнер?
49. Назвіть основні методики оцінки тестового покриття.
50. Охарактеризуйте техніку вимірювання тестового покриття на базі аналізу потоку управління.
51. Назвіть можливі сценарії використання інформації про тестове покриття.
52. Основні критерії для вибору багтрекера.
53. Основні критерії для вибору інструмента для управління тестами.
54. Дайте визначення поняттю «ROI».
55. Які заходи здатні позитивно вплинути на ROI?
56. Яким критеріям повинні відповідати тести, щоб їх можна було назвати «вигідними для автоматизації» тестами?
57. Дайте визначення поняттю «локатор» в контексті автоматизації тестування.
58. Перелічіть переваги та недоліки, які забезпечує автоматизація тестування ПЗ.
59. Дайте визначення поняттю «ефективні тестові прогони».
60. Дайте визначення поняттю «тестування зручності використання».
61. Які задачі вирішуються за допомогою юзабіліті-тестування?
62. Опишіть механізми проведення юзабіліті-тестування.
63. Перелічіть та охарактеризуйте основні види тестування продуктивності.
64. Опишіть механізми проведення тестування навантаженням.
65. Опишіть механізми проведення стрес-тестування.
66. Опишіть механізми проведення тестування стабільності.
67. Опишіть механізми проведення конфігураційного тестування.
68. Назвіть основні цілі тестування продуктивності.
69. Назвіть основні метрики продуктивності ПЗ.
70. Який вид тестування проводиться для визначення граничних значень навантаження досліджуваного об'єкта, при яких він задовольняє вимогам по продуктивності?
71. Який вид тестування проводиться для оцінки ступеня відповідності досліджуваного об'єкта вимогам при заданому очікуваному навантаженні протягом тривалого проміжку часу?

72. Для чого збираються метрики під час тестування ПЗ?
73. Які метрики є найважливішими з точки зору менеджера тестування?
74. Перелічіть основні етапи найму співробітників у відділ тестування ПЗ.
75. Чому важливо визначити індивідуальні мотиватори і стимулятори кожного співробітника?

**Студенти не допускаються до екзамену, якщо під час семестру вони набрали менше ніж 36 балів та/або не виконали всі передбачені планом роботи.**

## 7.2 Організація оцінювання.

### Терміни проведення форм оцінювання:

1. Презентація теми проекту: до 3 тижня семестру.
2. Розробка технічного завдання: до 5 тижня семестру.
3. Розробка макету інтерфейсу: до 7 тижня семестру.
4. Реферування літератури з тематики курсу: на протязі семестру (визначається індивідуально, за згодою сторін, але не пізніше 12 тижня).
5. Програмна реалізація прототипу ПЗ: до 10 тижня семестру.
6. Оцінка якості, тестування та пробна експлуатація ПЗ: до 12 тижня семестру.

Студенти мають право на одне перескладання кожної роботи із можливістю отримання максимально 80% початково визначених за цю роботу балів. Термін перескладання визначається викладачем.

У випадку відсутності студентів з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до «Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу» від 1 жовтня 2010 року.

У разі неякісного виконання визначених вище робіт, викладач має право не зарахувати роботу, або знизити за неї бали.

Студенти мають право здавати роботи після закінчення визначеного для них терміну, але з втратою 20% від максимальної оцінки за кожен тиждень, який пройшов з моменту закінчення терміну її здачі.

## 7.3 Шкала відповідності оцінок.

<b>Відмінно / Excellent</b>	90-100
<b>Добре / Good</b>	75-89
<b>Задовільно / Satisfactory</b>	60-74
<b>Незадовільно / Fail</b>	0-59

## 8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять.

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабораторні заняття	Самостійна робота
<b>Частина 1. Основні поняття забезпечення якості ПЗ.</b>				
1	<b>Тема 1. Мета і базові визначення курсу.</b>	2		2



2	<b>Тема 2.</b> Ручне тестування.	2	2	6
3	<b>Тема 3.</b> Дефекти.	2		4
4	<b>Тема 4.</b> Тестова документація.	2	2	12
5	<b>Тема 5.</b> Тест-дизайн.	2		10
Всього по частині 1		10	4	34
<b>Частина 2. Інструменти управління тестами та інші види тестування.</b>				
9	<b>Тема 6.</b> Багтрекінг.	2	2	4
10	<b>Тема 7.</b> Інструменти для автоматизації тестування.	2	2	4
11	<b>Тема 8.</b> Інші види тестування.	2	2	6
12	<b>Тема 9.</b> Менеджмент тестування.	2	2	12
Всього по частині 2		8	6	26
Консультація		2		
<b>ВСЬОГО</b>		<b>18</b>	<b>12</b>	<b>60</b>

Загальний обсяг – **90** год., в тому числі:  
 Лекцій – **18** год.  
 Лабораторні заняття – **10** год.  
 Консультації – **2** год.  
 Самостійна робота – **60** год.

### **Лабораторна робота.**

- Завдання.1:** Презентація теми проекту.
- Завдання.2:** Розробка технічного завдання.
- Завдання.3:** Розробка макету інтерфейсу.
- Завдання.4:** Реферування літератури з тематики курсу.
- Завдання.5:** Програмна реалізація прототипу ПЗ.
- Завдання.6:** Оцінка якості, тестування та пробна експлуатація ПЗ.

### **9. Рекомендовані джерела.**

#### **Основні:**

1. Black, Rex. Foundations of Software Testing. Salt Lake City: Thomson. January 2008. ISBN 1-84480-989-7.
2. Black, Rex. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, 3e. Hoboken, NJ: Wiley. April 2009. ISBN 0-470-40415-9.
3. Роман Савин. Тестирование Дот Ком, или Пособие по жестокому обращению с багами в интернет-стартапах. – М.: Дело, 2007. – 312 с. – ISBN 978-5-7749-0460-0.

4. Гленфорд Майерс, Том Баджетт, Кори Сандлер. Искусство тестирования программ, 3-е издание. – М.: «Диалектика», 2012. – 272 с. – ISBN 978-5-8459-1796-6.
5. Lee Copeland. A Practitioner's Guide to Software Test Design. – Artech House (January 2004). – 300p. – ISBN-10: 158053791X, ISBN-13: 978-1580537919.
6. Ron Patton. Software Testing (2nd Edition). – Sams Publishing (August 5, 2005). – 408 p. – ISBN-10: 0672327988, ISBN-13: 978-0672327988.
7. James A. Whittaker. Exploratory Software Testing: Tips, Tricks, Tours, and Techniques to Guide Test Design. – Addison-Wesley Professional (September 4, 2009). – 256 p. – ISBN-10: 0321636414, ISBN-13: 978-0321636416.

#### **Додаткові:**

1. Алан Купер. Психбольница в руках пациентов. – СПб.: Символ-Плюс, 2009. – 336 с. – ISBN 978-5-93286-168-4, ISBN 0-672-31649-8.
2. Дэвид С. Платт. Софт отстой и что с этим делать. – СПб.: Символ-Плюс, 2007. – 248 с. – ISBN 5-93286-097-9, ISBN 978-5-93286-097-7, ISBN 0-321-46675-6.
3. Святослав Куликов. Тестирование программного обеспечения. Базовый курс. ([http://svyatoslav.biz/software\\_testing\\_book/](http://svyatoslav.biz/software_testing_book/)).
4. Джоэл Спольски. Джоэл о программировании. – СПб.: Символ-Плюс, 2006. – 352 с. – ISBN 5-93286-063-4, ISBN 1-59059-389-8.
5. Mike Andrews, James A. Whittaker. How to break web software. – Addison-Wesley Professional (February 12, 2006). – 240 p. – ISBN-10: 0321369440, ISBN-13: 978-0321369444.
6. Б. Фитцпатрик, Б. Коллинз-Сассмэн. Идеальная IT-компания. Как из гиков собрать команду программистов. – СПб.: Питер, 2014. – 208 с. – ISBN 978-5-496-00949-2.
7. Том Демарко. Deadline. Роман об управлении проектами. – М.: Манн, Иванов и Фербер, 2011. – 352 с. – ISBN 978-5-91657-284-1.
8. Джефф Сазерленд. Scrum. Революционный метод управления проектами. – М.: Манн, Иванов и Фербер, 2015. – 288 с. – ISBN 978-5-00057-722-6.
9. Кармин Галло. iПрезентация. Уроки убеждения от лидера Apple Стива Джобса. – М.: Манн, Иванов и Фербер, 2010. – 224 с. – ISBN 978-5-91657-329-9.
10. Ненси Дуарте. Slide:ology. Искусство создания выдающихся презентаций. – М.: Манн, Иванов и Фербер, 2012. – 288 с. – ISBN 978-5-91657-241-4.