

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»**

Кафедра програмної інженерії та інформаційних технологій управління
(назва)

«ЗАТВЕРДЖУЮ»

Голова науково-методичної комісії _____
(назва комісії)

_____ (підпис) _____ (ініціали та прізвище)

« _____ » _____ 20 _____ року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ЯКІСТЬ І ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

(назва навчальної дисципліни)

рівень вищої освіти перший (бакалаврський)
перший (бакалаврський) / другий (магістерський)

галузь знань 12 Інформаційні технології
(шифр і назва)

спеціальність 126 Інформаційні системи та технології
(шифр і назва)

вид дисципліни професійна підготовка
(загальна підготовка / професійна підготовка)

форма навчання денна
(денна / заочна)

Харків – 2017 рік

ЛИСТ ЗАТВЕРДЖЕННЯ

Робоча програма з навчальної дисципліни

ЯКІСТЬ І ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗЕЧЕННЯ

(назва дисципліни)

Розробники:

доцент, к.т.н., доц.
(посада, науковий ступінь та вчене звання)

(підпис)

Орехов С.В.
(ініціали та прізвище)

(посада, науковий ступінь та вчене звання)

(підпис)

(ініціали та прізвище)

Робоча програма розглянута та затверджена на засіданні кафедри

програмної інженерії та інформаційних технологій управління

(назва кафедри)

Протокол від « 31 » серпня 2017 року № 1

Завідувач кафедри _____ Годлевський М.Д.
(назва кафедри) (підпис) (ініціали та прізвище)

ЛИСТ ПЕРЕЗАТВЕРДЖЕННЯ РОБОЧОЇ НАВЧАЛЬНОЇ ПРОГРАМИ

Дата засідання кафедри – розробника РПНД	Номер протоколу	Підпис завідувача кафедри	Підпис голови НМК (для дисциплін загальної підготовки та дисциплін професійної підготовки за спеціальністю) або завідувача випускової кафедри (для дисциплін професійної підготовки зі спеціалізації, якщо РПНД розроблена не випусковою кафедрою)

МЕТА, КОМПЕТЕНТНОСТІ, РЕЗУЛЬТАТИ НАВЧАННЯ ТА СТРУКТУРНО-ЛОГІЧНА СХЕМА ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета навчальної дисципліни “Якість і тестування програмного забезпечення” є навчання студентів методам тестування програмного забезпечення (ПЗ) та оцінки його якості як з боку замовника так і з боку програміста. Ці методи знайшли широке використання в області розробки програмних рішень.

Компетентності:

Загальні компетентності:

- Здатність застосовувати знання у практичних ситуаціях.
- Знання та розуміння предметної області та розуміння професійної діяльності.
- Здатність вчитися і оволодівати сучасними знаннями.
- Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Спеціальні (фахові) компетентності:

- Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення якості програмного забезпечення.

Нормативний зміст підготовки здобувачів вищої освіти, сформульований у термінах результатів навчання

Класифікація компетентностей за НРК	Знання	Уміння	Комунікація	Автономія та відповідальність
ФК 11 Здатність аналізувати, вибирати і застосовувати методи і засоби для тестування програмного забезпечення.	Знати основні міжнародні та національні стандарти з якості ПЗ; принципи побудови процесу тестування ПЗ, його цілі та місце в процесі розробки ПЗ, зокрема цикл попередження дефектів; різні методи тестування ПЗ; структуру звіту про помилки.	Вміти ідентифікувати дефекти та помилки в артефактах процесу розробки ПЗ; створювати тестовий план та тестові приклади; користуватися системами обліку програмних помилок;	Здатність донесення до фахівців і нефахівців інформації, ідей, проблем, рішень та власного досвіду в галузі професійної діяльності; здатність ефективно формувати комунікаційну стратегію.	Здатність визначати необхідні об'єми тестування ПЗ; працювати на попередження виникнення дефектів в процесі розробки ПЗ; користуватися системами обліку програмних помилок.

Попередні дисципліни:	Наступні дисципліни:
“Вища математика”	
“Математичне програмування”	
“Інформатика та комп’ютерна техніка”	

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

(розподіл навчального часу за семестрами та видами навчальних занять)

Семестр	Загальний обсяг (годин) / кредитів ECTS	З них		За видами аудиторних занять (годин)			Індивідуальні завдання студентів (КП, КР, РГ, Р, РЕ)	Поточний контроль	Семестровий контроль	
		Аудиторні заняття (годин)	Самостійна робота (годин)	Лекції	Лабораторні заняття	Практичні заняття, семінари			Контрольні роботи (кількість робіт)	Залік
1	2	3	4	5	6	7	8	9	10	11
7	150 /5	48	102	16	32			2	4	
8	90 /3	30	60	10	20			2	4	

Співвідношення кількості годин аудиторних занять до загального обсягу складає 48% (%):

СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

№ з/п.	Види навчальних занять (Л, ЛЗ, ПЗ, СР)	Кількість годин	Номер семестру (якщо дисципліна викладається у декількох семестрах). Назви змістових модулів. Найменування тем та питань кожного заняття. Завдання на самостійну роботу.	Рекомендована література (базова, допоміжна)
1	2	3	4	5
Семестр 7. Змістовий модуль 1. Якість програмного забезпечення				
	Л	4	Тема 1. Управління якістю програмного забезпечення Поняття якості програмного забезпечення. Поняття надійності програмного забезпечення. Модель	
	ЛР	6		
	СР	22		

№ з/п.	Види навчальних занять (Л, ЛР, ПЗ, СР)	Кількість годин	<p>Номер семестру (якщо дисципліна викладається у декількох семестрах). Назви змістових модулів. Найменування тем та питань кожного заняття. Завдання на самостійну роботу.</p>	Рекомендована література (базова, допоміжна)
	Л ЛР СР	4 10 20	<p>комплексного управління якістю ISO. Модель управління якістю процесу розробки програмного забезпечення СММІ.</p> <p>Тема 2. Метрична теорія програм Модель оцінки якості програми ISO. Основи метричної теорії оцінки якості програми. Огляд метрик оцінки якості на рівні функціонального програмування.</p>	
Семестр 7. Змістовий модуль 2. Процес верифікації програмного забезпечення				
	Л ЛР СР	4 6 30	<p>Тема 3. Цикл попередження дефектів. Огляд моделей життєвого циклу розробки програмного забезпечення. Цикл попередження дефектів компанії ІВМ. Вартість усунення дефектів програмного забезпечення на різних стадіях життєвого циклу розробки. Задачі валідації та верифікації, які вирішуються на різних етапах життєвого циклу. Перелік звітної документації..</p> <p>Тема 4. Процедура верифікації програмного забезпечення Визначення, ролі, обов'язки, етапи. Огляд методів верифікації. Метрики процесу верифікації. Документація про процес верифікації..</p>	
Семестр 8. Змістовий модуль 3. Тестування програмного забезпечення				
	Л ЛР СР	4 4 20	<p>Тема 5. Основні поняття тестування Термінологія. Проблеми тестування. Фази тестування. Критерії вибору тестів. Типи тестування. Рівні тестування. Методи тестування. Модульне тестування. Інтеграційне тестування. Функціональне тестування. Тестування „чорного ящика”. Компонентне тестування. Системне тестування.</p> <p>Тема 6. Програмні помилки Критерії та класифікація програмних помилок. Аналіз помилки, що повторюється. Звіт про програмну помилку. Системи відстеження програмних помилок.</p> <p>Тема 7. Розробка тестів Характеристики доброго тесту. Класи еквівалентності та граничні умови. Тестування переходів між станами. Тестовий план. Тестовий приклад. Виконання тестів.</p>	

№ з/п.	Види навчальних занять (Л, ЛЗ, ПЗ, СР)	Кількість годин	<p>Номер семестру (якщо дисципліна викладається у декількох семестрах). Назви змістових модулів. Найменування тем та питань кожного заняття. Завдання на самостійну роботу.</p>	Рекомендована література (базова, допоміжна)
Разом (годин)		240		

САМОСТІЙНА РОБОТА

№ з/п	Назва видів самостійної роботи	Кількість годин
1	Опрацювання лекційного матеріалу	60
2	Підготовка до практичних(лабораторних, семінарських) занять	40
3	Самостійне вивчення тем та питань, які не викладаються на лекційних заняттях	40
4	Виконання індивідуального завдання:	20
5	Інші види самостійної роботи	2
	Разом	162

ІНДИВІДУАЛЬНІ ЗАВДАННЯ

Курсовий проект

(вид індивідуального завдання)

Модуль 1. Якість програмного забезпечення

Аналіз метрик якості програмного забезпечення

Аналіз надійності програмного забезпечення

Побудова метричної моделі програмного забезпечення

Побудова моделі якості в ІТ компанії

Побудова моделі якості ІТ проекту

Використання метрик для оцінки якості програмного коду

Література: основна [1 – 4]; додаткова [5 – 15].

Модуль 2. Процес верифікації програмного забезпечення

Аналіз методів верифікації програмного забезпечення

Аналіз методів валідації програмного забезпечення

Побудова циклу попередження дефектів в ІТ компанії

Використання метрик для верифікації програмного забезпечення

Література: основна [1 – 4]; додаткова [5 – 15].

Модуль 3. Тестування програмного забезпечення

Побудова системи відстеження програмних помилок

Використання модульного тестування

Використання інтеграційного тестування

Використання системного тестування

Використання тестування навантаженням

Використання альфа-тестування

Використання бетта-тестування

Використання регресійного тестування

Література: основна [1 – 4]; додаткова [5 – 15].

МЕТОДИ НАВЧАННЯ

При викладанні навчальної дисципліни для активізації навчального процесу передбачено застосування сучасних навчальних технологій, таких, як: проблемні лекції; робота в малих групах; семінари-дискусії; кейс-метод; ділові ігри.

Проблемні лекції спрямовані на розвиток логічного мислення студентів. Коло питань теми лекції обмежується двома-трьома ключовими моментами, увага студентів концентрується на матеріалі, що не знайшов широкого відображення в підручниках, використовується досвід закордонних навчальних закладів з роздаванням студентам під час лекцій друкованого матеріалу та виділенням головних висновків з питань, що розглядаються. При викладанні лекційного матеріалу студентам пропонуються питання для самостійного розмірковування. При цьому лектор задає запитання, які спонукають студента шукати розв'язання проблемної ситуації. Така система примушує студентів сконцентруватися і почати активно мислити в пошуках правильної відповіді.

На початку проведення проблемної лекції необхідно чітко сформулювати проблему, яку необхідно вирішити студентам. При викладанні лекційного матеріалу слід уникати прямої відповіді на поставлені запитання, а висвітлювати лекційний матеріал таким чином, щоб отриману інформацію студент міг використовувати при розв'язанні проблеми.

Міні-лекції передбачають викладання навчального матеріалу за короткий проміжок часу й характеризуються значною ємністю, складністю логічних побудов, образів, доказів та узагальнень. Міні-лекції проводяться, як правило, як частина заняття-дослідження. На початку проведення міні-лекції за вказаними темами лектор акцентує увагу студентів на необхідності представити викладений лекційний матеріал у так званому структурно-логічному вигляді. На розгляд виносяться питання, які зафіксовані у плані лекцій, але викладаються вони стисло. Лекційне заняття, проведене у такий спосіб, пробуджує у студента активність та увагу при сприйнятті матеріалу, а також спрямовує його на використання системного підходу при відтворенні інформації, яку він одержав від викладача. Проблемні лекції та міні-лекції доцільно поєднувати з такою формою активізації навчального процесу, як робота в малих групах.

Робота в малих групах дає змогу структурувати лекційні або лабораторні заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду соціального спілкування. Після висвітлення проблеми (при використанні проблемних лекцій) або стислого викладання матеріалу (при використанні міні-лекцій) студентам пропонується об'єднуватися у групи по 5-6 осіб та презентувати наприкінці заняття своє бачення та сприйняття матеріалу.

Презентації – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи, звіту про виконання індивідуальних завдань. Однією з позитивних рис презентації та її переваг при використанні в навчальному процесі є обмін досвідом, який здобули студенти при роботі у певній малій групі.

Лабораторні заняття (з елементами семінарської дискусії) дозволяють формувати у студентів навички особистого експериментального дослідження фізичних процесів що відбуваються під час роботи компонентів операційної системи, проводити аналіз умов її функціонування, а також розробляти нові елементи та системні компоненти відповідно до вимог, що пред'являються до них, узагальнювати отримані результати, формулювати висновки та думки, вести подальший обмін думками та поглядами з іншими учасниками щодо отриманих результатів досліджень з даної теми, а також розвивають творче мислення, допомагають формувати погляди і переконання, вчать об'єктивно оцінювати результати і пропозиції опонентів, критично підходити до власних результатів та поглядів.

Ділові та рольові ігри – форма активізації студентів, за якої вони задіяні в процесі інсценізації певної виробничої ситуації у ролі безпосередніх учасників подій. Наприклад, при проведенні лабораторного заняття за темою “Безпечність персональних конфіденціальних даних на базі секретного диску та захищеної електронної пошти PGP” слід поділити аудиторію на групи, кожній з яких дати завдання використовуючи поштові протоколи.

Кейс-метод – метод аналізу конкретних ситуацій, який дає змогу наблизити процес навчання до реальної практичної діяльності спеціалістів і передбачає розгляд виробничих, управлінських та інших ситуацій, складних конфліктних випадків, проблемних ситуацій, інцидентів у процесі вивчення навчального матеріалу.

Розподіл форм та методів активізації процесу навчання за темами навчальної дисципліни

Тема	Практичне застосування навчальних технологій
ТЕМА 1. УПРАВЛІННЯ ЯКІСТЮ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	Проблемна лекція “Визначення базових засад якості програмного забезпечення, ІТ проекту, процесу розробки програмного забезпечення”
ТЕМА 3. ЦИКЛ ПОПЕРЕДЖЕННЯ ДЕФЕКТІВ	Кейс "Аналіз наукових результатів інституту інформаційних технологій компанії ІВМ". Міні-лекція “Класифікація програмних помилок”
ТЕМА 6. ПРОГРАМНІ ПОМИКЛИ	

МЕТОДИ КОНТРОЛЮ

Система оцінювання знань, вмінь та навичок студентів передбачає виставлення оцінок за усіма формами проведення занять. Перевірка та оцінювання знань студентів може проводитись у таких формах:

1. Оцінювання роботи студентів у процесі лабораторних занять.
2. Проведення проміжного контролю.
3. Проведення модульного контролю.

Загальна модульна оцінка складається з поточної оцінки, яку студент отримує під час лабораторних занять та оцінки за виконання модульної контрольної роботи.

Загальна оцінка з дисципліни визначається як середнє арифметичне модульних оцінок.

Порядок поточного оцінювання знань студентів

Поточне оцінювання здійснюється під час проведення лабораторних занять і має на меті перевірку рівня підготовленості студента до виконання конкретної роботи. Об'єктами поточного контролю є:

- 1) активність та результативність роботи студента протягом семестру над вивченням програмного матеріалу дисципліни; відвідування занять;
- 2) виконання проміжного контролю;
- 3) виконання модульного контрольного завдання.

Контроль систематичного виконання самостійної роботи та активності на лабораторних заняттях

Оцінювання проводиться за 5-бальною шкалою за такими критеріями:

- 1) розуміння, ступінь засвоєння теорії та методології проблем, що розглядаються;
- 2) ступінь засвоєння матеріалу дисципліни;
- 3) ознайомлення з рекомендованою літературою, а також із сучасною літературою з питань, що розглядаються;
- 4) уміння поєднувати теорію з практикою при розгляді виробничих ситуацій, розв'язанні задач, проведенні розрахунків при виконанні завдань, винесених для самостійного опрацювання, та завдань, винесених на розгляд в аудиторії;
- 5) логіка, структура, стиль викладу матеріалу в письмових роботах і при виступах в аудиторії, вміння обґрунтовувати свою позицію, здійснювати узагальнення інформації та робити висновки.

Оцінка "відмінно" ставиться за умови відповідності виконаного завдання студента або його усної відповіді до всіх п'яти зазначених критеріїв.

Відсутність тієї чи іншої складової знижує оцінку на відповідну кількість балів.

При оцінюванні практичних завдань увага приділяється також їх якості та самостійності, своєчасності здачі виконаних завдань викладачу (згідно з графіком навчального процесу). Якщо якась із вимог не буде виконана, то оцінка буде знижена.

Проміжний модульний контроль

Проміжний модульний контроль рівня знань передбачає виявлення опанування студентом матеріалу лекційного модуля та вміння застосовувати його для вирішення практичної ситуації і проводиться у вигляді контрольної роботи за темами 1-го або 2-го модулю.

Проведення модульного контролю

Модульний контроль здійснюється та оцінюється за допомогою проведення контрольної роботи за всіма темами дисципліни.

Підсумкова оцінка з дисципліни розраховується як середня з кількох складових, що враховує оцінки кожного виду контролю (дві оцінки за результатами поточного модульного контролю, оцінку за курсовий проект і підсумкову контрольну роботу).

РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ СТУДЕНТИ, ТА ШКАЛА ОЦІНЮВАННЯ ЗНАНЬ ТА УМІНЬ (НАЦІОНАЛЬНА ТА ECTS)

Таблиця 1 – Розподіл балів

	Поточний контроль			Семестровий контроль	Всього за семестр
	КР	лр	ІНДЗ		
Підсумкові бали	75			25	100
Макс. проміжні бали	12	7	16		
Кільк. од. обліку у семестрі	2	5	1		
Макс. проміжних балів, всього	24	35	16		100
Коеф.. перерахунку	1				
Макс. кільк. підсумкових балів	24	35	16	25	100

Таблиця 2 – Розподіл балів за виконання курсового проекту

Пояснювальна записка	Ілюстративна частина	Захист роботи	Сума
до 25	до 25	до 50	100

Таблиця 3 – Шкала оцінювання знань та умінь: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою
90 ... 100	A	відмінно
82 ... 89	B	добре
74 ... 81	C	
64 ... 73	D	задовільно
60 ... 63	E	
35 ... 59	FX	незадовільно з можливістю повторного складання
0 ... 34	F	незадовільно з обов'язковим повторним вивченням дисципліни

НАВЧАЛЬНО-МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Орехов С.В. Лабораторний практикум з дисципліни “Якість ПЗ та тестування” [Електронний ресурс]. – Режим доступу: ntumoodle.com

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Базова

1. Канер Сэм. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений.: Пер. с англ. Сэм Канер, Джек Фолк, Енг Кек Нгуен. — К.: Издательство «Диасофт», 2001.
2. Бейзер Б. Тестирование черного ящика. Технологии функционального тес-тирования программного обеспечения и систем / Б. Бейзер; - Питер, 2004.
3. Сеницын С. В., Налютин Н. Ю. Верификация программного обеспечения. Учебное пособие [текст] / С. В. Сеницын, Н. Ю. Налютин; - Бинум, 2008.
4. Коликова Т.В., Котляров В.П. Основы тестирования программного обеспе-чения. Учебное пособие [текст] / Т.В. . Коликова, В.П. Котляров; - Интуит, 2006.

Допоміжна література

5. Брауде Э.Д. Технология разработки программного обеспечения / Э.Д. Брауде; - Питер, 2004.
6. Винниченко И. В. Автоматизация процессов тестирования. Питер, 2005.
7. Калбертсон Р., Браун К., Кобб Г. Быстрое тестирование – Вільямс, 2002.
8. Касперски К. Техника отладки программ без исходных текстов – СПб, БХВ - 2005.

9. Макгрегор Д., Сайкс Д. Тестирование объектно-ориентированного программного обеспечения. Практическое пособие - ТИД «ДС», 2004.
10. Плаксин М. Тестирование и отладка программ - для профессионалов будущих и настоящих - Бином. Лаборатория знаний, 2007.
11. Роберт М., Ньюкирк Д., Быстрая разработка программ: принципы, примеры, практика - Вильямс, 2004.
12. Рэшка Д., Пол Д., Дастин Э. Автоматизированное тестирование программного обеспечения - Лори, 2003.
13. Спольски Д. Лучшие примеры разработки ПО - Питер, 2007.
14. Тамре Л. Введение в тестирование программного обеспечения [текст] / Л. Тамре; - Вильямс, 2003.
15. Фолк Д., Нгуен Е. К., Канер С. Тестирование программного обеспечения - Диасофт, 2003.