

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем та програмної інженерії
/назва факультету/

Кафедра комп'ютерних систем та мереж
/назва кафедри/

ЗАТВЕРДЖУЮ

Декан факультету

Ігор БАРАН

« 02 » 09 2024 року



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
«Інженерія програмного забезпечення»

/назва дисципліни/

галузь знань 12 Інформаційні технології
/шифр і назва галузі знань/

рівень вищої освіти перший (бакалаврський)
/назва/

спеціальність 123 - Комп'ютерна інженерія
/шифр і назва/

освітня програма Комп'ютерна інженерія
/назва/

спеціалізація _____
/назва/

вид дисципліни обов'язкова
/обов'язкова / вибіркова/

Тернопіль - 2024 рік

Робоча програма з навчальної дисципліни

«Інженерія програмного забезпечення»

/назва дисципліни/

для студентів факультету комп'ютерно-інформаційних систем і програмної інженерії

/назва факультету(ів)/

Розробники:

доцент

/посада, науковий ступінь та вчене звання/



/підпис/

/Василь ЯЦИШИН/

/ім'я та прізвище/

/посада, науковий ступінь та вчене звання/

/підпис/

/ім'я та прізвище/

Робоча програма розглянута та схвалена

на засіданні кафедри комп'ютерних систем та мереж

/назва/

Протокол від « 27 » серпня 2024 року № 2

Завідувач кафедри

/підпис/

/Галина ОСУХІВСЬКА/

/ім'я та прізвище/

Робоча програма розглянута та схвалена НМК

факультету комп'ютерно-інформаційних систем і програмної інженерії

Протокол від « 02 » вересня 2024 року № 1

Секретар НМК

/підпис/

/Богдана МЛИНКО/

/ім'я та прізвище/

Робоча програма погоджена:

Спеціальність 123 - Комп'ютерна інженерія

/шифр і назва/

освітня програма Комп'ютерна інженерія

/назва/

Завідувач випускової кафедри

/підпис/

/Галина ОСУХІВСЬКА/

/ім'я та прізвище/

Гарант освітньої програми

/підпис/

/Андрій ПАЛАМАР/

/ім'я та прізвище/

1. Структура навчальної дисципліни

Показник	Всього годин	
	(денна форма навчання)	(заочна форма навчання)
Кількість кредитів/годин	4,5/135	4,5/135
Аудиторні заняття, год.	64	16
Самостійна робота, год.	71	119
Аудиторні заняття:		
• лекції, год.	32	8
• лабораторні заняття, год.	32	8
• практичні заняття, год.	-	-
• семінарські заняття, год.	-	-
Самостійна робота:		
опрацювання лекційного матеріалу	16	32
підготовка до лабораторних занять	16	32
опрацювання окремих розділів програми, які не виносяться на лекції	9	25
виконання контрольних завдань	-	-
виконання індивідуальних завдань	-	-
виконання курсових проектів (робіт)	-	-
підготовка та складання заліків, екзаменів, контрольних робіт, рефератів, есе, тестування	30	30
Екзамен	Е	Е
Залік	-	-

Частка годин самостійної роботи студента:

денна форма навчання – 53,0%.

заочна форма навчання – 88,0%.

2. Мета та завдання навчальної дисципліни

2.1. Мета вивчення навчальної дисципліни.

Мета вивчення дисципліни «Інженерія програмного забезпечення» полягає у формуванні знань та вмінь у студентів щодо реалізації процесів створення та управління проектами програмного забезпечення, адаптації і застосування формальних моделей і методів комп'ютерних наук у практику розробки програмного забезпечення.

2.2. Завдання навчальної дисципліни.

Завданням вивчення навчальної дисципліни є набуття студентами теоретичних та практичних навиків з основ інженерії вимог програмного забезпечення, методів проектування архітектури, принципів керування програмними проектами, атестації і впровадження програмних систем.

Вивчення навчальної дисципліни передбачає формування та розвиток у студентів загальних (ЗК) та фахових (ФК) компетентностей згідно освітньої програми.

Загальні:

ЗК2. Здатність вчитися і оволодівати сучасними знаннями.

ЗК3. Здатність застосовувати знання у практичних ситуаціях.

ЗК6. Навички міжособистісної взаємодії.

ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

ЗК8. Здатність працювати в команді.

Фахові:

ФК1. Здатність застосовувати законодавчу та нормативно-правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

ФК5. Здатність використовувати засоби і системи автоматизації проектування до розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо.

ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

ФК12. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних та кіберфізичних систем, мереж та їхніх компонентів шляхом використання аналітичних методів і методів моделювання.

ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.

ФК14. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

ФК15. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.

ФК16. Здатність застосовувати підходи та принципи проектування програмного забезпечення високопродуктивних та розподілених комп'ютерних систем, зокрема, хмарних сервісів, а також здійснювати їх супровід з урахуванням DevOps-практик.

ФК17. Здатність розробляти апаратне і програмне забезпечення для вбудованих систем та їх компонентів з використанням технологій Інтернету речей.

Вивчення навчальної дисципліни передбачає формування у студента таких програмних результатів навчання (ПР) згідно освітньої програми:

ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.

ПРН5. Мати знання основ економіки та управління проектами.

ПРН6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

ПРН7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.

ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.

ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

ПРН11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.

ПРН12. Вміти ефективно працювати як індивідуально, так і у складі команди.

ПРН14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.

ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення

ПРН18. Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях.

ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

ПРН20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення

ПРН21. Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики

ПРН22. Вміти розробляти програмне забезпечення для паралельних високопродуктивних обчислювальних систем, а також здійснювати супровід таких систем з урахуванням принципів DevOps.

ПРН23. Вміти розробляти апаратне і програмне забезпечення для вбудованих систем та їх компонентів на основі концепції Інтернету речей.

3. Опис навчальної дисципліни

3.1. Лекційні заняття

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
1.	Тема 1. Основні поняття та проблеми розробки ПЗ Поняття програмного забезпечення. Інженерія програмного забезпечення як інженерна дисципліна. Характеристика областей знань інженерії програмного забезпечення.	2	0,5
2.	Тема 2. Життєвий цикл ПЗ. Моделі та методології розробки ПЗ Поняття життєвого циклу. Міжнародний стандарт життєвого циклу ПЗ ISO/IEC 12207. Типи і характеристика процесів життєвого циклу.	2	0,5
3.	Тема 3. Моделі життєвого циклу програмного забезпечення Типи моделей життєвого циклу. Каскадна модель. Еволюційна модель. Інкрементна модель. Спіральна модель. Моделі швидкої розробки ПЗ	2	0,5
4.	Тема 4. Аналіз, специфікація, верифікація та валідація вимог до ПЗ. Функціональні та нефункціональні вимоги Визначення вимог до програмних систем. Класифікація вимог. Джерела одержання вимог та методи їх аналізу. Інженерія вимог. Фіксація вимог. Трасування і комунікація вимог до програмних систем. Об'єктно-орієнтована інженерія вимог: візуальний і текстовий підходи.	2	0,5
5.	Тема 5. Проектування архітектури ПЗ. Шаблони проектування ПЗ Загальні підходи до проектування програмних систем. Проектування різних видів архітектури програмних систем. Декомпозиція системи на підсистеми. Проектування архітектури на рівні каркасів проектування. Проектування архітектури на рівні патернів архітектури.	2	0,5

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
6.	Тема 6. Основні поняття моделювання ПЗ Методології моделювання SADT, IDEF, DFD, ELM, OOAD. Мови моделювання. Інформаційне моделювання. Діаграми сутності -зв'язок, класів. Поведінкове моделювання. Діаграми станів, діяльності, взаємодії, послідовності, часові. Моделі предметної області. Поведінкові моделі. Об'єктні моделі. Моделі даних. Створення прототипів користувацьких інтерфейсів. Технології швидкого прототипування. Засоби автоматизації моделювання (ERWin, BPWin, Enterprise Architect, IBM Rational Architect та ін.)	2	0,5
7.	Тема 7. Різновиди реалізації програмного забезпечення. Прикладне (систематичне) програмування. Структурне моделювання і програмування. Об'єктно-орієнтоване програмування. Компонентне програмування. Функційне програмування. Сервісно-орієнтоване програмування. Аспектно-орієнтоване програмування. Агентне програмування.	2	0,5
8.	Тема 8. Принципи і технології організації колективної роботи при розробці програмного забезпечення Організація моделей управління репозиторіями програмного коду. Інструменти колективної розробки програмного забезпечення та контролю версій. Git. Організація та основні функціональні можливості. Розв'язання конфліктності. Управління вітками.	2	0,5
9.	Тема 9. Проектування людино-машинних інтерфейсів Інтерфейси в сучасних середовищах. Інтерфейс між клієнтом і сервером. Стандарт ISO/IEC 11404-96 з незалежних від мов типів даних. Перетворення форматів даних. Перетворення даних з баз даних. Принципи проектування інтерфейсу користувачів. Методи візуалізації та представлення інформації. Взаємодія з користувачем. Засоби підтримки користувачів. Оцінювання користувацьких інтерфейсів.	2	0,5

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
10.	Тема 10. Управління ресурсами програмного забезпечення Основні поняття та задачі. Трикутник обмежень. Модель процесу керування проектом. Управління ресурсами. Планування графіку виконання проекту.	2	0,5
11.	Тема 11. Управління ризиками та станом виконання проекту Управління ризиками програмного проекту. Управління конфігураціями та змінами. Контроль та моніторинг стану проекту. Метрики контролю. Організація роботи проектної команди.	2	0,5
12.	Тема 12. Методи валідації та верифікації програмних систем Методи доведення правильності програм. Модель доведення програми за твердженнями. Підхід до валідації сценарію вимог. Верифікація об'єктних моделей. Інспектування програмних систем. Автоматичний статистичний аналіз програмних систем. Технологія Clean Room.	2	0,5
13.	Тема 13. Тестування програмного забезпечення Методи тестування програмних систем. Статичні методи тестування. Динамічні методи тестування. Функціональне тестування. Класифікація помилок і методи їхнього пошуку. Процес тестування за життєвим циклом. Інженерія керування тестуванням. Тестування дефектів. Інтеграційне тестування. Системне тестування.	2	0,5
14.	Тема 14. Інфраструктура програмного забезпечення Поняття інфраструктури та її роль у розробці та доставці програмного забезпечення до стейкхолдера. Хмарні технології. Інфраструктура як код (IaaS). Поняття DevOps. DevOps-процеси у життєвому циклі програмного забезпечення. Функції та роль DevOps фахівців.	2	0,5
15.	Тема 15. Неперервна інтеграція та доставка програмного забезпечення. Поняття неперервної інтеграції програмного забезпечення. Принципи організації CI/CD. Засоби неперервної інтеграції і доставки програмного забезпечення. Особливості налаштування. Сервер CI/CD Jenkins	2	0,5

Продовження табл.3.1

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
16.	Тема 16. Методи забезпечення та контролю якості програмного забезпечення Поняття якості програмних систем. Міжнародні стандарти у сфері забезпечення та контролю якості програмних систем. Моделі якості програмних систем. Метрики якості. Управління якістю програмних систем на етапах життєвого циклу. Технології оцінювання якості програмних систем. Сертифікація програмного забезпечення.	2	0,5
Усього годин		32	8

3.2. Лабораторні заняття

№	Тема заняття	Кількість годин	
		ДФН	ЗФН
1	Розробка специфікації вимог до програмного забезпечення.	4	1
2	Проектування об'єктно-орієнтованої архітектури ПЗ	4	1
3	Побудова UML діаграм для представлення різних аспектів ПЗ	6	1
4	Реалізація програмного забезпечення (інтернет додатки, системне програмне забезпечення та ін.) в обраному середовищі розробки з використанням системи контролю версій	8	2
5	Розробка плану тестування, виконавчих тест-кейсів та звіту тестування ПЗ	6	1
6	Контейнеризація розробленого програмного забезпечення з використанням Docker технології	4	2
Усього годин		32	8

* Усі лабораторні роботи, зазвичай, виконуються в команді з обов'язковим представленням результатів у вигляді презентацій та/або з демонстрацією роботи системи.

3.4. Самостійна робота

№	Найменування робіт	Кількість годин	
		ДФН	ЗФН
1	Опрацювання лекційного матеріалу	16	32
2	Підготовка до лабораторних робіт	16	32
3	Опрацювання окремих розділів програми, які не виносяться на лекції:	9	25
	1. Поняття інформаційних систем. Автоматизовані інформаційно-вимірjuвальні системи. Системи реального часу.	3	5
	2. RUP технологія розробки ПЗ. Формальні граматики.	3	10
	3. Методи документування системних вимог. Технології керування вимогами.	3	10
4	Підготовка та складання тестування	30	30
	– Модуль 1	10	10
	– Модуль 2	10	10
	– Екзамен	10	10
Усього годин		71	119

4. Критерії оцінювання результатів навчання студентів

Форма підсумкового семестрового контролю – екзамен.

Підсумкова семестрова оцінка складається з суми балів, отриманих студентом при проведенні проміжних (модульних) контролів рівня засвоєння теоретичних знань (за перший та другий модулі), отриманих балів за лабораторні роботи та оцінки за екзамен.

Захист звіту з лабораторної роботи оцінюється відповідною кількістю балів, поданою в таблиці.

Після проходження теоретичного матеріалу проводиться перевірка якості його засвоєння у вигляді тестування в рамках проміжного (модульного) контролю.

Контроль здійснюється засобами електронного навчального курсу (ЕНК) на сервері дистанційного навчання.

Модуль 1			Модуль 2			Підсумковий контроль		Разом з дисципліни
Аудиторна та самостійна робота			Аудиторна та самостійна робота			Теоретичний курс	Практичне завдання	
Теоретичний курс (тестування)	Лабораторна робота		Теоретичний курс (тестування)	Лабораторна робота				
20	15		15	25		15	10	
№ лекцій	Вид робіт	Бал	№ лекцій	Вид робіт	Бал			
Лекція 1	Лаб.	5	Лекція 9	Лаб.	10			
Лекція 2	роб.							
Лекція 3	№1							
Лекція 4	Лаб.	5	Лекція 12	Лаб.	10			
Лекція 5	роб.							
Лекція 6	№2							
Лекція 7	Лаб.	5	Лекція 15	Лаб.	5			
Лекція 8	роб. №3		Лекція 16	роб. №6				

До підсумкового семестрового контролю (складання екзамену) допускаються студенти, які протягом семестру виконали всі види навчальної роботи, успішно пройшли проміжні (модульні) контролю і набрали не менше 45 балів семестрової бальної оцінки та за умови отримання не менше 60% балів за результатами кожного проміжного (модульного) контролю рівня знань.

Підсумкова оцінка записується за 100-бальною шкалою із подальшим переведенням її у шкалу Європейської кредитно-трансферної системи (ECTS) відповідно А, В, С, D, E, F, FX при цьому чотирибальна шкала оцінок (з записом семестрової оцінки «відмінно» - А, «добре» - В,С, «задовільно» - D, E відповідають підсумковому результату «зараховано», «незадовільно» - F, FX відповідає підсумковому результату «незараховано»).

5. Навчально-методичне забезпечення

1. Яцишин В.В. Електронний навчальний курс з дисципліни “Інженерія програмного забезпечення” (ID: 2790, сертифікат № 0415 від 21.04.2023 р.), доступний за адресою: <http://dl.tntu.edu.ua/bounce.php?course=2790>.

2. Яцишин В.В. Конспект лекцій з дисципліни «Інженерія програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія». Тернопіль, 2022. 216 с.

3. Яцишин В.В. Методичні вказівки до виконання лабораторних робіт з дисципліни «Інженерія програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія». Тернопіль, 2022. 80 с.

4. Яцишин В.В. Методичні вказівки до самостійної роботи студентів з дисципліни «Інженерія програмного забезпечення » для студентів денної та заочної форм навчання спеціальності 123 «Комп'ютерна інженерія». Тернопіль, 2022. 15 с.

6. Рекомендована література

Базова

1. Бородкіна І.Л., Бородкін Г.О. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів. К.: Центр учбової літератури. 2020. 204 с

2. Sommerville I. Software Engineering, Global Edition. Pearson Higher Ed. 2016. 816 p.

3. Мартін Р. Чистий код. Створення і рефакторинг за допомогою Agile. В-во «Фабула». 2019. 448 с.

4. Мартін Р. Чиста архітектура. Мистецтво розробки програмного забезпечення. В-во «Фабула». 2019. 368 с.

5. Лавріщева К.М. Програмна інженерія. Академперіодика. К. 2009. 371 с.

6. Wiegers K, Beatty J. Software Requirements, 3rd Edition. Microsoft Press. 2013. 672 p.

7. Yatsyshyn V., Pastukh O., Lutskev A., Tsymbalistyy V., Martsenko N. A Risks management method based on the quality requirements communication method in agile approaches / Information technologies: theoretical and applied problems, 2022, pp. 1-10.

8. Pastukh O., Yatsyshyn V. Development of software for neuromarketing based on artificial intelligence and data science using high-performance computing and

parallel programming technologies. Scientific Journal of TNTU. Tern.: TNTU, 2024. Vol 113. No 1. P. 143–149.

9. Yatsyshyn V., Kharchenko O., Lutsiv A. Maturity Requirements Model for Software Requirements with the Implementation of ISO/IEC 25010 Recommendations. International Journal "Information Models and Analyses", Vol 9, N 2, 2020. 126-143. URL: <http://www.foibg.com/ijima///vol09/ijima09-02-p02.pdf>

Допоміжна

1. Davis J., Daniels R. Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale. "O'Reilly Media, Inc.", 2016 p. - 410 p.

2. Mouat A. Using Docker: Developing and Deploying Software with Containers. "O'Reilly Media, Inc.". 2015 p. 354 p.

3. Chandrasekara Ch., Herath P. Hands-on GitHub Actions: Implement CI/CD with GitHub Action Workflows for Your Applications. Apress, 2021. 162 p.

7. Інформаційні ресурси

1. http://www.tutorialspoint.com/software_engineering/
2. <https://www.geeksforgeeks.org/software-engineering-introduction-to-software-engineering/>

8. Зміни та доповнення до робочої програми навчальної дисципліни

№	Зміст внесених змін (доповнень)	Дата і № протоколу засідання кафедри	Примітки	
1				
2				
3				
4				
5				