

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ЗАТВЕРДЖУЮ

Декан факультету інформаційних технологій

Тетяна Говорущенко

Підпис: _____

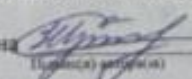
2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ Методології та технології розробки програмних систем

Галузь знань: F Інформаційні технології
Спеціальність: F2 «Інженерія програмного забезпечення»
Рівень вищої освіти: Другий (магістерський)
Освітньо-професійна програма: Інженерія програмного забезпечення
Обсяг дисципліни – 4 кредитів ЄКТС
Мова навчання: українська
Статус дисципліни: обов'язкова (фахової підготовки)

Форма здобуття освіти	Курс	Семестр	Обсяг дисципліни		Кількість годин						Форма семестрового контролю
			Кредити ЄКТС	Години	Аудиторні заняття					Самостійна робота (в т.ч. ІРС)	Іспит
					Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття		
Д	1	1	4	120	34	16	18			86	+

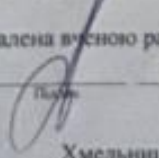
Робоча програма складена на основі освітньо-професійної програми «Інженерія програмного забезпечення» за спеціальністю F2 «Інженерія програмного забезпечення»

Робоча програма складена  к.пед.н., доцент, **Наталія ПРАВОРСЬКА**
Підпис(и) автор(ів) Науковий ступінь, учасні підпис, ім'я, ПІПСОПІД автор(ів)

Схвалена на засіданні кафедри **Інженерія програмного забезпечення**

Протокол від 28 серпня 2025 р. № 1 Зав. кафедри  **Леоніда БЕДРАТЮК**
Підпис Ім'я, ПІПСОПІД

Робоча програма розглянута та схвалена вченою радою факультету інформаційних

Голова вченої ради факультету  **Тетяна ГОВОРУЩЕНКО**
Підпис Ім'я, ПІПСОПІД

Хмельницький 2025

2 ЛИСТ ПОГОДЖЕННЯ

Посада	Назва факультету	Підпис	Ініціали, прізвище
Завідувач кафедри інженерії програмного забезпечення, доктор, фіз.-мат. наук, проф.	Факультет інформаційних технологій		Леонід БЕДРАТЮК
Гарант освітньо-професійної програми, к.т.н., доцент	Факультет інформаційних технологій		Оксана ЯШИНА

МЕТОДОЛОГІЇ ТА ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНИХ СИСТЕМ

Тип дисципліни	Вибіркова
Рівень вищої освіти	Другий
Мова викладання	Українська
Семестр	Перший
Кількість встановлених кредитів ЄКТС	4,0
Форми навчання, для яких викладається дисципліна	Денна

Результати навчання.

Студент, який успішно завершив вивчення дисципліни, має: досконало володіти професійною термінологією та основними поняттями ПС; розуміти основні етапи попередніх досліджень, визначення цілей продукту, що розробляється, зарубіжні та вітчизняні стандарти оформлення специфікації до ПС, мови специфікації вимог, проектувати ІТ-продукти, знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення, виявляти інформаційні потреби і класифікувати дані для проектування програмних систем, розробляти і оцінювати стратегії проектування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проектних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів.

Зміст навчальної дисципліни.

Основні підходи до командної розробки ПЗ, Microsoft Solutions Framework. Основні компоненти і принципи методології, управління проектами ризиками і готовністю в MSF. Методології розробки ПС. Технології програмування. Державні та міжнародні стандарти в області розробки програмного забезпечення. Елементи програмної інженерії. Організація технологічного процесу розробки ПС. Оцінка програмного продукту по СОСОМО. Проектування програмних систем. Побудова архітектури ПП. Тестування та відлагоджування ПЗ.

Запланована навчальна діяльність: лекцій 16 год., лабораторних занять 18 год., самостійної роботи 86 год.; разом 120 год.

Методи навчання: методи проблемного викладання, словесні, наочні (лекції); пояснювально-ілюстративні, проблемного викладання, дослідницькі, частково-пошукові (лабораторні заняття), проблемного викладання, дослідницькі, частково-пошукові (самостійна робота: індивідуальні завдання).

Форми і методи оцінювання результатів навчання: усне опитування, захист лабораторних робіт

Вид семестрового контролю: іспит – 1 семестр.

Навчальні ресурси:

1. Mark Richards, Neal Ford. Fundamentals of Software Architecture. – An Engineering Approach. 1st Ed. – 2020 – pp. 419.
2. Джефф Сазерленд. Scrum. Навчись робити вдвічі більше за менший час. – Клуб Сімейного Дозвілля. 2022 – 280 р.
3. Arno Lauret. The Design of Web APIs. Manning Publications Co.– 2025.– pp. 575.
4. Ali Aminian, Alex Xu. Machine Learning System Design Interview. ByteByteGo. – 2023. – pp. 294
5. Tejas Chopra, Dharendra Sinha. System Design Guide for Software Professionals: Build scalable solutions. Packt Publishing, The Limited. – 2024 – pp..384
6. Модульне середовище для навчання MOODLE. Доступ до ресурсу: <https://msn.khnu.km.ua>.

Викладач: кандидат педагогічних наук, доцент Праворська Н.І

3. Пояснювальна записка

Дисципліна «Методології та технології розробки програмних систем» є обов'язковою дисципліною фахової підготовки і займає провідне місце у підготовці здобувачів другого (магістерського) рівня вищої освіти за освітньо-професійною програмою «Інженерія програмного забезпечення» в межах спеціальності F2 «Інженерія програмного забезпечення». У рамках дисципліни «Методології та технології розробки програмних систем» вивчається процес конструювання та розробки програмних систем, розповсюдженні методики і практики побудови надійного програмного забезпечення.

Пререквізити – вихідна

Кореквізити – ОФП.04 Машинне навчання, ОФП.05 Управління проектами і якістю програмних систем, ОФП.07 Переддипломна практика, ОФП.08 Кваліфікаційна робота

Відповідно до освітньої програми дисципліна має забезпечити:

- **компетентності:** (ІК) Здатність особи розв'язувати складні задачі і проблеми у певній галузі професійної діяльності або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог; (ФК.01) Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення; (ФК.02) Здатність розробляти і реалізовувати наукові та/або прикладні проєкти у сфері інженерії програмного забезпечення; (ФК.03) Здатність проєктувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів; (ФК.04) Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення; (ФК.05) Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення; (ФК.06) Здатність ефективно керувати фінансовими, людськими, технічними та іншими проєктними ресурсами у сфері інженерії програмного забезпечення; (ФК.07) Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах; (ФК.08) Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення; (ФК.09) Здатність забезпечувати якість програмного забезпечення;

- **програмні результати навчання:** (ПРН.01) Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення; (ПРН.02) Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу; (ПРН.03) Будувати і досліджувати моделі інформаційних процесів у прикладній області; (ПРН.04) Виявляти інформаційні потреби і класифікувати дані для проєктування програмного забезпечення; (ПРН.05) Розробляти, аналізувати, обґрунтовувати та систематизувати вимоги до програмного забезпечення; (ПРН.06) Розробляти і оцінювати стратегії проєктування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проєктних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів; (ПРН.07) Аналізувати, оцінювати і застосовувати на системному рівні сучасні програмні та апаратні платформи для розв'язання складних задач інженерії програмного забезпечення; (ПРН.08) Розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника; (ПРН.13) Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу; (ПРН.15) Здійснювати реінжиніринг програмного забезпечення відповідно до вимог замовника.

Мета дисципліни: формування теоретичних та практичних навичок в здобувачів другого (магістерського) рівня вищої освіти з питань проєктування та розробки інформаційних систем та вивчення основних шляхів організації і виконання успішних проєктів в області розробки програмного забезпечення.

Предмет дисципліни – охоплює теоретичні знання, задачі, функції та вимоги до програмного забезпечення та систем. Проводиться розгляд технологічних основ процесу розробки програмного забезпечення та систем; розроблення формальних вимог до програмної системи, основаних на потребах замовника; розроблення проєкту програмної системи з урахуванням можливостей її подальшого розвитку, а також повторного використання її компонент в інших проєктах; вивчення основ уніфікованої мови UML для візуального моделювання елементів наочної області в рамках проєктування програмної системи і її основних компонент; отримання практичного досвіду роботи із

застосуванням методології MSF; розвиток навиків аналізу, проектування, документування і розробки програмних комплексів середньої складності.

Завдання дисципліни: отримання студентами знань, умінь і навичок в області застосування програмних систем, оволодіння дослідницькими навичками, уміння працювати самостійно, здатністю породжувати нові ідеї (креативністю), оволодіння основними методами, способами й засобами одержання, зберігання, переробки інформації з використанням комп'ютерної техніки, уміннями аналізувати й оцінювати зібрані дані.

- **Результати навчання:** після вивчення дисципліни студент повинен аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки; уміти вибирати та використовувати відповідну задачі методологію створення програмного забезпечення; проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування; вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання; застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення; знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань; застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення; мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації; вміти застосовувати методи компонентної розробки програмного забезпечення; знати та вміти застосовувати методи верифікації та валідації програмного забезпечення; розробляти і модифікувати архітектуру програмного забезпечення для реалізації вимог замовника; виконувати конфігурацію програмного забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу; здійснювати реінжиніринг програмного забезпечення відповідно до вимог замовника.

4. СТРУКТУРА ЗАЛІКОВИХ КРЕДИТІВ ДИСЦИПЛІНИ “Методології та технології розробки програмних систем”

Теми	Лекції	Лаб. роб.	Самостійна робота студентів
Тема 1. Процес розробки програмних систем Бізнес і IT-проекти. Ринок ПЗ.	2	2	10
Тема 2. Міжнародні та національні стандарти розробки складних програмних продуктів	2		10
Тема 3. Організація технологічного процесу розробки ПЗ. Підстави аналізу вимог. Встановлення вимог Принципи специфікації вимог.	2	2	10
Тема 4. Оцінка програмного проекту по СОСОМО	2		10
Тема 5. Системне проектування. Проектування баз даних.	2	2	10
Тема 6. Проектування програмної системи	2	4	12
Тема 7. Побудова архітектури ПП. Розробка програм, модульне програмування	2	4	12
Тема 8. Тестування і відлагодження ПС. Тестування та управління змінами. Види тестування	2	4	12
Разом за 1-й семестр	16	18	86

5. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ “Методології та технології розробки програмних систем”

5.1. Зміст лекційного курсу

№ лекції	Перелік тем лекцій, їх анотація	Кількість годин
Тема 1. Процес розробки програмних систем. Бізнес і IT-проекти. Ринок ПЗ.		
1	Лекція 1. Процес розробки програмних систем. Бізнес і IT-проекти. Ринок ПЗ. Стандарт ISO 9000. UML.CASE-засоби та вдосконалення процесу. Планування розроблення системи. Підхід SWOT. Підхід VCM. Підхід BPR. Підхід ISA. Системи для трьох рівнів керування. Етапи життєвого циклу програмного забезпечення. Планування проекту протягом життєвого циклу ПЗ. Бізнес і IT-проекти. Ринок ПЗ. Причини невдач IT-проектів. Виникнення Технології розробки ПЗ. Література: [1] стор. 13-24; [2] стор. 16-35. [10]; [12].	2
2	Тема 2. Міжнародні та національні стандарти розробки складних програмних продуктів. Методологія розробки інформаційних та програмних систем у вітчизняній та зарубіжній літературі. Державні та міжнародні стандарти в області розробки програмного забезпечення. Міжнародний стандарт ISO / ІЕС 12207: 1995-08-01. Стандарти комплексу ГОСТ 1934. Стандарти комплексу ГОСТ 19. Забезпечення надійності розробки ПЗ. Складність програмної системи. Література: [1] стор. 29-45; [4] стор. 158-195, [6] стор. 113-201.	2
3	Тема 3. Організація технологічного процесу розробки ПЗ. Підстави аналізу вимог. Встановлення вимог. Принципи специфікації вимог. Основи об'єктної технології. Аналіз об'єктів. Моделювання прецедентів. Моделювання видів діяльності. Моделювання класів. Моделювання взаємодій. Діаграма станів. Принципи встановлення вимог. Виявлення вимог. Прототипування. Спільна розробка програм (JAD-метод). Системні	2

	послуги. Системні обмеження. Специфікація вимог. Специфікації станів. Моделювання класів. Виявлення класів. Підхід з урахуванням використання іменних груп. Підхід з урахуванням використання загальних шаблонів класів. Підхід з урахуванням використання прецедентів. Комплексний підхід. Література: [2] стор. 57-98; [5] стор. 135-149; [7] стор. 113-156; [9]; [10]; [11].	
4	Тема 4. Оцінка програмного проекту по СОСОМО. Процес управління проектом: початок проекту; оцінювання, заходи і метрики; процес оцінки; аналіз ризиків; планування; трасування і контроль. Планування проектних завдань. Розмірно-орієнтовані метрики. Функціонально-орієнтовані метрики. Оцінка проекту на основі LOC- і FP-метрик. Література: [3] стор. 141-165; [5] стор. 132-156; [6] стор. 271-328; [8]	2
5	Тема 5. Системне проектування. Проектування баз даних. Архітектура програмного забезпечення. Розподілена архітектура. Триланкова архітектура. BCED рівнів. Стратегія повторного використання компоненти. Діаграма компонентів. Розгортання. Реалізація Web-додатків. Проект розгортання. Розгортання Web-додатків. Рівень незмінних об'єктів бази даних. Моделі даних. Відображення об'єктів у базі даних. Модель об'єктної бази даних. Об'єктно-реляційна модель бази даних. Модель реляційної бази даних. Елементарні типи моделі РБД. Реляційні таблиці. Література: [5] стор. 163-171; [7] стор. 149-163; [11].	2
6	Тема 6. Проектування програмної системи. Особливості процесу синтезу програмних систем. Особливості етапу проектування. Структуризація системи. Моделювання управління. Декомпозиція підсистем на модулі. Модульність. Інформаційна закритість. Зв'язність модуля. Зчеплення модулів. Складність програмної системи. Характеристики ієрархічної структури програмної системи. Зв'язність та ув'язування класів. Види ув'язування класів. Закон Деметра. Методи відкриття доступу. Динамічна класифікація та зв'язність класів зі змішаними екземплярами. Проектування клієнт-серверних кооперативних взаємодій. Збереження процедур. Тригери. Проектування транзакцій. Короткі транзакції. Рівень ізоляваності. Автоматичне відновлення. Програмоване відновлення. Точка збереження. Тригерний відкат. Література: [2] стор. 312-397; [3] стор. 170-189; [4] стор. 162-175; [8]; [9]	2
7	Тема 7. Побудова архітектури ПП. Розробка програм, модульне програмування. Поняття архітектури і завдання її опису. Основні класи архітектури програмних засобів. Взаємодія між підсистемами і архітектурні функції. Контроль архітектури програмних засобів. Література: [4] стор. 175-209; [8]; [34];	2
8	Тема 8. Тестування і відлагодження ПС. Тестування та управління змінами. Види тестування. Основні поняття тестування і від лагодження. Основні підходи і принципи відлагодження. Організація процесу тестування ПЗ. Види тестування. Тестування системних послуг. Наскрізний контроль. Інспекція. Тестування по відношенню до специфікації. Тестування стосовно програмного коду. Тестування системних обмежень. Тестування інтерфейсу користувача. Тестування бази даних. Тестування авторизації. Тестування інших обмежень. Документація з тестування та управління змінами. Література: [6] стор. 439-561; [9]; [15]; [35].	2
Разом за 1-й семестр		16

5.2. Зміст лабораторних занять

№ з/п	Теми лабораторних занять	Години
1	Планування процесу розробки ПС	2
2	Організація розробки ПС	2

3	Аналіз вимог та моделювання Розробка та специфікація вимог	2
4	Об'єктно-орієнтований аналіз систем	4
5	Побудова тимчасової діаграми етапів робіт для заданого процесу	4
6	Розробка проектної моделі	4
Разом за 1-й семестр		18

5.3. Зміст самостійної роботи

Об'єм самостійної роботи з дисципліни “Методології та технології розробки програмних систем” становить 86 години. Вони включають опрацювання лекційного матеріалу, теоретичних і лабораторних завдань, підготовку до виконання лабораторних робіт, їх захисту і поточне тестування.

Номер тижня	Назва теми	Години
1	Тема 1. Процес розробки програмних систем. Бізнес і IT-проекти. Ринок ПЗ. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №1. Література: [1] стор. 13-24; [2] стор. 16-35. [10]; [12].	4
2	Тема 1. Процес розробки програмних систем. Бізнес і IT-проекти. Ринок ПЗ. Захист лабораторної роботи № 1. Література: [1] стор. 13-24; [2] стор. 16-35. [10]; [12].	6
3	Тема 2. Сучасні технології та інструментарій програмування. Призначення і роль новітніх інформаційних систем та технологій. Класифікація та можливості. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №2. Література: [1] стор. 29-45; [4] стор. 158-195, [6] стор. 113-201.	10
4	Тема 3. Організація технологічного процесу розробки ПЗ. Підстави аналізу вимог. Встановлення вимог. Принципи специфікації вимог. Опрацювання лекційного матеріалу, виконання лабораторної роботи №2. Література: [2] стор. 57-98; [5] стор. 135-149; [7] стор. 113-156; [9] ; [10]; [11].	4
5	Тема 3. Організація технологічного процесу розробки ПЗ. Підстави аналізу вимог. Встановлення вимог. Принципи специфікації вимог. Опрацювання лекційного матеріалу. Захист лабораторної роботи №2. Література: [2] стор. 57-98; [5] стор. 135-149; [7] стор. 113-156; [9] ; [10]; [11].	6
6	Тема 4. Оцінка програмного проекту по СОСОМО. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №3. Література: [3] стор. 141-165; [5] стор. 132-156; [6] стор. 271-328; [8]	4
7	Тема 4. Оцінка програмного проекту по СОСОМО. Опрацювання лекційного матеріалу. Виконання лабораторної роботи №3. Література: [3] стор. 141-165; [5] стор. 132-156; [6] стор. 271-328; [8]. Підготовка до поточного контролю.	6
8	Тема 5. Системне проектування. Проектування баз даних. Опрацювання лекційного матеріалу, захист лабораторної роботи №3. Література: [5] стор. 163-171; [7] стор. 149-163; [11].	4
9	Тема 5. Системне проектування. Проектування баз даних. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №4. Література: [5] стор. 163-171; [7] стор. 149-163; [11].	6
10	Тема 6. Проектування програмної системи. Опрацювання лекційного матеріалу, захист лабораторної роботи №4. Література: [2] стор. 312-397;	6

	[3] стор. 170-189; [4] стор. 162-175; [8]; [9]	
11	Тема 6. Проектування програмної системи. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №5. Література: [2] стор. 312-397; [3] стор. 170-189; [4] стор. 162-175; [8]; [9]	6
12	Тема 7. Побудова архітектури ПП. Розробка програм, модульне програмування. Опрацювання лекційного матеріалу, продовження виконання лабораторної роботи №5. Література: [4] стор. 175-209; [8]; [34];	6
13	Тема 7. Побудова архітектури ПП. Розробка програм, модульне програмування. Опрацювання лекційного матеріалу, підготовка до виконання лабораторної роботи №6. Література: [4] стор. 175-209; [8]; [34];	6
14	Тема 8. Тестування і відлагодження ПС. Тестування та управління змінами. Види тестування. Опрацювання лекційного матеріалу, виконання лабораторної роботи №6. Література: [24; 27].	2
15	Тема 8. Тестування і відлагодження ПС. Тестування та управління змінами. Види тестування. Опрацювання лекційного матеріалу. Захист лабораторної роботи №6. Література: [6] стор. 439-561; [9]; [15]; [35].	4
16	Тема 8. Тестування і відлагодження ПС. Тестування та управління змінами. Види тестування. Опрацювання лекційного матеріал. Підготовка до іспиту. Література: [6] стор. 439-561; [9]; [15]; [35].	6
Разом за 1-й семестр		86

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи питання з кожної теми. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання контрольної роботи (для студентів заочної форми здобуття освіти) та індивідуального домашнього завдання (для студентів денної форми здобуття освіти) викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

6. Технології та методи навчання

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: методи навчання за джерелом передачі і сприймання інформації (словесні (пояснення, дискусія, консультування), практичні (інструктування, розв'язування ситуаційних задач), наочні (демонстрування, ілюстрування, спостереження); за логікою передачі і сприймання навчальної інформації; за рівнем самостійності пізнавальної діяльності (методи проблемного викладу, частково пошукові, дослідницькі); методи стимулювання і мотивації учіння, інтерактивні; метод аналізу конкретних ситуацій (case-study) з використанням технологій візуалізації, інформаційно-комунікаційних та технології дистанційного навчання (сервіс для проведення онлайн конференцій Zoom, Модульне середовище для навчання тощо).

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій, зокрема:

- лекції (з використанням мультимедійних презентацій, методів візуалізації, пояснення, проблемного й інтерактивного навчання, методів стимулювання і мотивації, **інформаційно-комунікаційних технологій**, інтенсифікації та індивідуалізації навчання, тощо);

- лабораторні заняття (з використанням методів комп'ютерного моделювання, методів проектної діяльності, тренінгових вправ, аналіз проблемних ситуацій, пояснення, дискусія, бесіда, демонстрування, спостереження, з використанням кейсів, розв'язування задач, презентацій);

- самостійна робота (робота над засвоєнням теоретичного матеріалу, виконання індивідуальних та домашніх завдань, підготовка до поточного та підсумкового контролю тощо).

7. Методи контролю

Поточний контроль здійснюється під час практичних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу.

При цьому використовуються такі методи поточного контролю:

- тестовий контроль теоретичного матеріалу;
- оцінювання результатів роботи на лабораторних заняттях (розв'язування задач, участь у обговоренні ситуацій, самостійні роботи);

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, не допускається до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який має академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в індивідуальному режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до практичних занять (вивчення теоретичного матеріалу з теми, активно працювати на занятті, розв'язувати задачі, брати участь у дискусіях щодо прийнятих рішень при виконанні здобувачами задач).

Здобувачі вищої освіти зобов'язані дотримуватися термінів виконання усіх видів робіт у встановлені терміни, передбачених робочою програмою навчальної дисципліни. Пропущене практичне заняття здобувач зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами опитування під час практичних занять, тестування й виконання індивідуального домашнього завдання. Виконання індивідуального завдання завершується його здачею на перевірку у терміни, встановлені графіком самостійної роботи. У якості ІДЗ здобувач може підготувати реферат або тези доповіді на конференцію за однією з тем навчальної дисципліни при дотриманні узгоджених з викладачем термінів його виконання.

Здобувач вищої освіти, виконуючи самостійну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, підказки, плагіат, використання штучного інтелекту (без вірного цитування)). У разі порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності під час вивчення навчальної дисципліни не допускаються та не толеруються.

У межах вивчення навчальної дисципліни здобувачам вищої освіти передбачено визнання і зарахування результатів навчання, набутих шляхом неформальної освіти, що розміщені на доступних платформах (<https://github.com>, <https://docs.oracle.com/javase/tutorial/java/index.html>, <http://www.bruceeckel.com/>, <http://java.sun.com/products/jdk/1.3/docs/index.html>, <http://java.sun.com/products/jdk/1.3/devdocs-vs-specs.html>, <http://developer.java.sun.com/developer/onlineTraining/>, <http://java.sun.com/products/java-media/jmf/2.1.1/guide/JMFTOC.html>), які сприяють формування компетентностей і поглибленню результатів навчання, визначених робочою програмою дисципліни, або забезпечують вивчення

відповідної теми та/або виду робіт з програми навчальної дисципліни (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із призначених робочою програмою для цього виду роботи. При цьому кожна структурна одиниця (робота) може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

Будь-які форми порушення академічної доброчесності **не допускаються та не толеруються**.

Отриманий здобувачем бал за зарахований вид навчальної роботи (структурну одиницю) після її оцінювання викладач виставляє в електронному журналі обліку успішності здобувачів вищої освіти. За умови виконання усіх видів навчальної роботи за результатами поточного контролю протягом вивчення навчальної дисципліни, встановлених її Робочою програмою, здобувач денної форми здобуття освіти з навчальної дисципліни, підсумковим контролем для якої є іспит, може набрати до 60 балів (здобувач заочної форми – до 50 балів). Позитивну підсумкову оцінку здобувач може отримати, якщо за результатами поточного та підсумкового контролів набере від 60 до 100 балів. Семестрова підсумкова оцінка розраховується в автоматизованому режимі в інформаційній підсистемі «Електронний журнал» (ІС «Електронний університет») і відповідно до накопиченої суми балів визначається оцінка за інституційною шкалою та шкалою ЄКТС (див. таблицю Співвідношення...), яка заноситься в екзаменаційну відомість, а також до Індивідуального навчального плану здобувача вищої освіти.

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>помилки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у

	відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з вивчення навчальної дисципліни.

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у семестрі

Аудиторна робота								Контрольні заходи	Семест- ровий контроль	Разом
<i>Другий семестр</i>										
Лабораторні роботи №:								Тестовий контроль:	Іспит	Сума балів
1	2	3	4	5	6	7	8	Т		
Кількість балів за вид навчальної роботи (мінімум-максимум)										
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5		
24-40								12-20	24-40	60-100

Умовні позначення: Т – тема навчальної дисципліни

Оцінювання результатів захисту лабораторної роботи

Виконана й оформлена відповідно до встановлених Методичними рекомендаціями вимог лабораторна робота комплексно оцінюється викладачем при її захисті з урахуванням таких критеріїв: самостійність та правильність виконання; повнота відповіді та знання методики проектування та розробки коду (застосунок); наявність відповідей на контрольні питання до відповідної лабораторної роботи, наявність звіту зі скріпками коду та результатів його роботи, тощо.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти та рівня досягнення здобувачем запланованих ПРН та сформованих компетентностей з присвоєнням йому відповідної суми балів.

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, встановленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому не зараховується і для її захисту він має детальніше опрацювати матеріал з теми роботи, методику її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання результатів навчання з теоретичного матеріалу (тестовий контроль)

Тест, передбачений Робочою програмою, відбувається за допомогою системи Strum, розробленою компанією Genesis в рамках вивчення дисципліни «Створення та розвиток ІТ-продуктів»

Розподіл балів в залежності від наданих правильних відповідей на тестові завдання

Кількість пойдених модулів	0-2	3-4	5-9	10-12
Відсоток правильних відповідей	10%	25%	50%	100%
Отримана оцінка	-	3	4	5 та отримано сертифікат
Набрана кількість балів	>6	12	16	20

Оцінювання результатів підсумкового семестрового контролю (іспит)

Освітня програма передбачає підсумковий семестровий контроль з дисципліни у формі іспиту, завданням якого є системне й об'єктивне оцінювання як теоретичної, так і практичної

підготовки здобувача з навчальної дисципліни. Складання іспиту відбувається за попередньо розробленими і затвердженими на засіданні кафедри білетами. Відповідно до цього в екзаменаційному білеті пропонується поєднання питань як теоретичного (в т.ч. у тестовій формі), так і практичного характеру.

Таблиця – Оцінювання результатів підсумкового семестрового контролю здобувачів денної форми навчання (40 балів для підсумкового контролю)

Види завдань	Для кожного окремого виду завдань		
	Мінімальний (достатній) бал (задовільно)	Потенційні позитивні бали* (середній бал) (добре)	Максимальний (високий) бал (відмінно)
Теоретичне питання № 1	6	4	5
Теоретичне питання № 2	6	4	5
Практичне завдання	6	24	30
Разом:	24		40

Примітка. *Позитивний бал за іспит, відмінний від мінімального (24 бали) та максимального (40 балів), знаходиться в межах 25-39 балів та розраховується як сума балів за усі структурні елементи (завдання) іспиту.

Для кожного окремого виду завдань підсумкового семестрового контролю застосовуються критерії оцінювання навчальних досягнень здобувача вищої освіти, наведені вище (**Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти**).

Семестровий іспит виставляється, якщо загальна сума балів, яку набрав студент з дисципліни за результатами поточного контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «відмінно/добре/задовільно», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення.

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	Відмінно/Excellent – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		Добре/Good – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
D	66-72		
E	60-65		
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. ПИТАННЯ ДЛЯ ПІДСУМКОВОГО КОНТРОЛЮ З ДИСЦИПЛІНИ

“Методології та технології розробки програмних систем”

1. Планування розробки системи

2. Планування проекту протягом життєвого циклу ПЗ
3. Підхід SWOT
4. Проектування клієнт-серверних кооперативних взаємодій
5. Проектування транзакцій
6. Програмування баз даних
7. Підхід на основі використання прецедентів
8. Підхід на основі використання іменних груп
9. Підхід на основі використання загальних шаблонів для класів.
10. Прототипування.
11. Принципи встановлення вимог
12. Поясніть різницю між етапами визначення вимог і розробки специфікації.
13. Поясніть взаємозв'язок двох етапів проектування (архітектурне проектування і деталізоване проектування) з першими двома етапами життєвого циклу – етапом визначення вимог і етапом розробки специфікації.
14. Поясніть, у чому складається розходження між розподіленою системою обробки і розподіленою системою баз даних.
15. Поясніть, у чому полягають основні розходження чотирьох підходів до виявлення класів.
16. CASE-засоби та удосконалювання процесу
17. Системи для трьох рівнів керування
18. Стратегія повторного використання.
19. Специфікації вимог
20. Специфікації станів
21. Спільна розробка додатків (JAD-метод).
22. Системні сервіси
23. Суб'єкти
24. Етапи життєвого циклу програмного забезпечення.
25. Етап встановлення вимог
26. Етап специфікації вимог.
27. Етап проектування архітектури
28. Етап деталізованого проектування.
29. Етап реалізації
30. Елементарні типи моделі РБД.
31. Основи об'єктної технології
32. Об'єктно-реляційна модель бази даних
33. Опишіть види блокувань при песимістичному керуванні паралельністю.
34. Моделювання аналізу
35. Модель об'єктної бази даних.
36. Моделі даних
37. Моделювання класів.
38. Моделювання прецедентів
39. Моделювання видів діяльності
40. Динамічна класифікація і зв'язаність класів зі змішаними екземплярами
41. Діаграми прецедентів та документування прецедентів
42. Діаграма видів діяльності
43. Динамічна класифікація і зв'язаність класів зі змішаними екземплярами
44. Зв'язність і ув'язування класів
45. Збережені процедури.
46. Закон Дементра.
47. Реляційні таблиці
48. Рівень постійних об'єктів бази даних
49. Реалізація Web-додатків
50. Рівні BCED
51. Реінжиніринг бізнес-процесів (BPR) проводить ясне розходження між бізнес-процесом і бізнес-функцією. У чому полягає це розходження?
52. Виявлення вимог
53. Відображення об'єктів у базу даних
54. Тестування відносно специфікації

55. Триланкова архітектура
56. Традиційні методи виявлення вимог
57. Комплексний підхід
58. Коротко опишіть види блокувань при песимістичному керуванні паралельністю.
59. Коротко опишіть п'ять рівнів SQL-інтерфейсів.
60. Архітектура програмного забезпечення
61. Що таке точка збереження? Як її можна використовувати при проектуванні програми?
62. Що таке відношення з'єднання?
63. Що таке триланкова архітектура? У чому її переваги і недоліки?
64. Що таке домінуючий клас?
65. Які дії можливі у відповідь на відправлений запит на зміни?
66. Які об'єкти можуть виступати як цільові об'єкти для повідомлень відповідно до закону Деметра?
67. Який вплив на проектування роблять принципи, зв'язані зі зв'язаністю та ув'язуванням?
68. Які розходження між об'єктною і реляційною моделлю БД?
69. Які основні причини зрушення від структурного підходу до проектування об'єктно-орієнтованого?
70. Які принципи встановлення вимог?
71. У чому перевага виклику з клієнтської програми збереженої процедури в порівнянні з SQL-запитом, що пересилається базі даних? Чи існують ситуації, при яких ми змушені використовувати SQL-запит замість виклику вилученої процедури?
72. У чому полягає сутність підходу на основі використання загальних шаблонів для класів?
73. Чому розуміння методу ISA (архітектура інформаційної системи) важливо для системної розробки?

11. Навчально-методичне забезпечення

Освітній процес з дисципліни «Методології та технології розробки програмних систем» повністю і в достатній кількості забезпечений і представлений у модальному середовищі ХНУ <https://msn.khmnmu.edu.ua/course/view.php?id=8110>

12. Матеріально-технічне та програмне забезпечення дисципліни (за потреби)

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проектор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями.

Вивчення навчальної дисципліни не потребує використання спеціального програмного прикладного забезпечення, крім загальнонавчаних програм і операційних систем.

13. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

ОСНОВНА:

1. Mark Richards, Neal Ford. Fundamentals of Software Architecture. – An Engineering Approach. 1st Ed. – 2020 – pp. 419.
2. Джефф Сазерленд. Scrum. Навчись робити вдвічі більше за менший час. – Клуб Сімейного Дозвілля. 2022 – 280 р.
3. Ali Aminian, Alex Xu. Machine Learning System Design Interview. ByteByteGo. – 2023. – pp. 294
4. Arno Lauret. The Design of Web APIs. Manning Publications Co. – 2025. – pp. 575.
5. Tejas Chopra, Dharendra Sinha. System Design Guide for Software Professionals: Build scalable solutions. Packt Publishing, The Limited. – 2024 – pp. 384
6. Кунгурцев О.Б., Новікова Н.О. Конструювання програмного забезпечення. Об'єктно-орієнтований підхід. Видавництво Кондор. – 2024 – 228 стор.
7. Neal Ford, Mark Richards. Fundamentals of Software Architecture : An Engineering Approach. O'reilly Media, Incorporated – 2020 – pp. 400
8. Joel Natty. Software Design Patterns for Scalable Applications Independently Published. – 2025 – pp. 142
9. ДСТУ ISO 9000:2007. Системи управління якістю. Основні положення та словник термінів. – К.: Держспоживстандарт, 2008. – [Чинний від 2008-01-01] – 35 с. – (Державний стандарт).

10. ДСТУ ISO 9001:2009. Системи управління якістю. Вимоги. [Текст]: – К.: Держспоживстандарт, 2009. – [Чинний від 2009-06-22] – 80 с.– (Державний стандарт).
11. VanVeendabl E. Standard glossary of terms used in Software testing / E. VanVeendabl. - ISTQB. - 2010. Vol. 1,2. - June. - 30 p.
12. 1846A: Microsoft Solutions Framework Essentials. Microsoft Official Course, 2002
13. 2710B: Analyzing Requirements and Defining Microsoft .NET Solutions Architecture. Microsoft Official Course, 2003
14. MSF Process Model. White paper, 2002 Microsoft Corporation.
15. MSF Risk Management Discipline. White paper, 2002 Microsoft Corporation.
16. MSF Team Model. White paper, 2002 Microsoft Corporation.

ДОПОМІЖНА:

17. Brenda Jin, Amir Shevat, Saurabh Sahní The Design of Web APIs. O'reilly Media, Incorporated – 2018. – pp. 218
18. Robert C. Martin. Clean Architecture: A Craftsman's Guide to Software Structure and Design (Robert C. Martin Series) 1st Edition, – 2018 Pearson Education, Inc – pp.352
19. Бороджкіна І., Бороджкін Г.. Інженерія програмного забезпечення. – Центр навчальної літератури. – 2018.- 204 с.
20. Grady Booch. Object-Oriented Analysis and Design with Applications. Third Edition – Addison Wesley – 2007 – 717p.
21. Van Veendabl E. Standard glossary of term used in Software testing / E. Van Veendabl. - ISTQB. - 2007. Vol. 1,2. - June. - 30 p.
22. Sidorov N.A. Software Stylistics / N. A. Sidorov // Proceedings of NAU. - 2015. - 2(24), - P. 98 - 103.
23. Creating a software engineering culture / K. Wiegers // Dorset House Publishing. - New York, 2003. - 358 p.
24. G. Booch, J. Rumbaugh, I. Jacobson. The Unified Modeling Language Reference Manual - Second Edition, Addison-Wesley, 2004.
26. Матвєєва Л.Є. Процес розробки програмного забезпечення. Від теорії до практики / Л.Є.Матвєєва, В.А. Волков. –К. ТОВ «Інформаційні програмні системи», 2012 – 117с.
27. Перевозчикова О.Л. Інформаційні системи і структури даних. – Києво-Могилянська академія. – 2011. -287 с.
28. Steve Resnick, Aaron Bjork, Michael de la Maza. Professional Scrum with Team Foundation Server 2010. – Wrox- 2011 - 211 p.
29. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем: Навчальний посібник. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2011. - 108 с.
30. J.D. Meier, Jason Taylor, Alex Mackman, Prashant Bansode, Kevin Jones. Team Development with Visual Studio Team Foundation Server. - Microsoft Corporation.-2014.-496 p.
31. Len Bass, Paul Clements, Rick Kazman. Software Architecture in Practice (SEI Series in Software Engineering) 3rd Edition. – 2013
32. Microsoft Visual Studio Licensing. 2019. - Microsoft Corporation–34 p.
33. Charles Petzold. Code: The Hidden Language of Computer Hardware and Software (Dv- Undefined) – 2000-
34. Michael C. Feathers, Prentice Hall. Working Effectively with Legacy Code – 2013.
35. Braude Eric J. Software Design: From Programming to Architecture Gebundene Ausgabe – Illustriert, 14. Februar 2013
36. Computing Curricula 2005. The Overview Report. [Електронний ресурс]. – Режим доступу: www.acm.org/education/curricula.html.
37. Systems and software engineering – Software Life Cycle Processes. ISO 12207:2008. – [Чинний від 2008-02-01] – II, 122 с.– (Міжнародний стандарт).
38. IEEE Standard Glossary of Software Engineering Terminology, Глосарій. IEEE Std 610.12-1990. – (Галузевий стандарт).
39. Техническое задание на создание автоматизированной системы. ГОСТ 34.602-89 – [Чинний від 1990-01-01] – 12 с.– (Міждержавний стандарт).
40. Pravorska N. METHOD OF APPLYING MACHINE LEARNING TO ENHANCE THE EFFICIENCY OF DEVOPS PROCESSES – November 2024. Herald of Khmelnytskyi National University Technical sciences 343(6(1)):454-463. DOI:10.31891/2307-5732-2024-343-6-68

41. Праворська Н.І., Грига С.І. МЕТОДИ РЕАЛІЗАЦІЇ МІКРОСЕРВІСНИХ АРХІТЕКТУР: ПЕРЕВАГИ ТА НЕДОЛІКИ, ВПРОВАДЖЕННЯ ТА ТЕСТУВАННЯ ПРИ РОЗРОБЦІ ПРОГРАМНИХ ЗАСТОСУНКІВ. Вісник Хмельницького національного університету №3, Т1. 2024 (335)- стор. 404-409

14. ІНФОРМАЦІЙНІ РЕСУРСИ

1. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khmnu.edu.ua/course/view.php?id=8110>
2. Електронна бібліотека університету . Доступ до ресурсу: <http://library.khmnu.edu.ua/>
3. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khmnu.edu.ua>
4. UML [Електронний ресурс]. – Режим доступу : <http://www.uml2.ru> /<http://uml3.ru>./
5. UML2.ru. Сообщество системных аналитиков [Электронный ресурс]. – Режим доступа : <http://www.uml2.ru/>.