



ЗАТВЕРДЖУЮ

Декан факультету інформаційних технологій

Тетяна ГОВОРУЩЕНКО
Ім'я, ПРІЗВИЩЕ

09 2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Технології створення програмних продуктів

Назва дисципліни

Галузь знань – 12 Інформаційні технології

Спеціальність – 122 Комп'ютерні науки

Рівень вищої освіти – Перший (бакалаврський)

Освітньо-професійна програма – Комп'ютерні науки

Обсяг дисципліни – 6 кредитів ЄКТС

Шифр дисципліни – ОПП.09

Мова навчання – Українська

Статус дисципліни: Обов'язкова (професійної підготовки)

Факультет – Інформаційних технологій

Кафедра – Комп'ютерних наук

Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Курсовий проект	Курсова робота	Форма семестрового контролю	
					Аудиторні заняття					Самостійна робота, у т.ч. ІРС			Залік	Іспит
			Кредити ЄКТС	Години	Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття					
Д	4	7	6	180	66	32	34			114				+
Разом ДФН			6	180	66	32	34			114				7

Робоча програма складена на основі освітньо-професійної програми «Комп'ютерні науки» за спеціальністю 122 «Комп'ютерні науки»

Робоча програма складена


Підпис автора

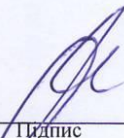
д-р. техн. наук, проф. Едуард МАНЗЮК

Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ автора

Схвалена на засіданні кафедри Комп'ютерних наук

Протокол від 29.08.2025 № 1.

Зав. кафедри


Підпис

Олександр БАРМАК

Ім'я, ПРІЗВИЩЕ

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

Голова вченої ради факультету


Підпис

Тетяна ГОВОРУЩЕНКО

Ім'я, ПРІЗВИЩЕ

ЛИСТ ПОГОДЖЕННЯ

Посада	Назва кафедри	Підпис	Ініціали, ПРІЗВИЩЕ
Завідувач кафедри, д-р. техн. наук, проф.	Комп'ютерних наук		Олександр БАРМАК
Гарант освітньо-професійної програми, канд. техн. наук, доц.	Комп'ютерних наук		Олександр МАЗУРЕЦЬ

3. Пояснювальна записка

Дисципліна «Технології створення програмних продуктів» є однією із дисциплін фахової підготовки і займає провідне місце у підготовці здобувачів першого (бакалаврського) рівня вищої освіти, очної (денної) (далі – денної) форми здобуття вищої освіти, які навчаються за освітньо-професійною програмою «Комп'ютерні науки» в межах спеціальності 122 «Комп'ютерні науки».

Пререквізити – вебтехнології (ОПП.02), технології розподілених систем та паралельних обчислень (ОПП.11), теорія алгоритмів (ОЗП.08), об'єктно-орієнтоване проектування (ОПП.03), алгоритмізація та програмування (ОЗП.05).

Кореквізити – управління ІТ-проєктами (ОПП.04), інформаційні технології хмарних обчислень (ОПП.13), кваліфікаційна робота (ОПП.17), професійна практика (ОПП.16).

Відповідно до освітньої програми дисципліна сприяє забезпеченню:

компетентностей: здатність до аналізу та функціонального моделювання бізнес-процесів, побудови та практичного застосування функціональних моделей організаційно-економічних і виробничотехнічних систем, методів оцінювання ризиків їх проектування (ФК 15);

програмних результатів навчання: розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук (ПРН 9); виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення (ПРН 16).

Мета дисципліни. Формування у здобувачів вищої освіти компетентностей, необхідних для розв'язання складних спеціалізованих задач у сфері створення програмних продуктів, розвитку навичок проектування програмного забезпечення із застосуванням шаблонних методів та архітектурних рішень, аналізу та моделювання програмних систем.

Предмет дисципліни. Шаблони проектування програмних систем, принципи розробки програмного забезпечення, архітектурні рішення побудови програмних продуктів, методи та технології створення програмних систем у сфері комп'ютерних наук.

Завдання дисципліни. Формування практичних навичок проектування програмних систем із застосуванням шаблонних методів, розвитку здатностей до функціонального моделювання предметних середовищ, опанування технологій паралельних та розподілених обчислень при створенні програмних продуктів.

Результати навчання. Після вивчення дисципліни студент повинен: розробляти програмні моделі предметних середовищ із застосуванням відповідних шаблонів проектування; обирати оптимальні парадигми програмування та архітектурні рішення з позицій якості та зручності застосування; реалізовувати породжуючі, поведінкові та структурні шаблони проектування; застосовувати принципи проектування та методи responsibility-driven design; проектувати програмні системи з урахуванням паралельних та розподілених обчислень; використовувати шаблони для конкуруючих операцій при розробці багатопотокового програмного забезпечення; аналізувати та моделювати бізнес-процеси через призму архітектури програмних продуктів; оцінювати ризики проектування та обирати оптимальні технологічні рішення для реалізації програмних систем.

4. Структура залікових кредитів дисципліни

Назва розділу (теми)	Кількість годин, відведених на:		
	лекції	лабор. роботи	СРС
Тема 1. Шаблони проектування. Вступ в шаблони проектування. Класифікація шаблонів проектування, (шаблони GoF). Породжуючі шаблони Abstract Factory, Builder.	2	8	6
Тема 2. Породжуючі шаблони Factory Method, Prototype, Singleton.	2	6	6
Тема 3. Поведінкові шаблони проектування Chain of Responsibility, Command, Interpreter.	2	6	6
Тема 4. Поведінкові шаблони Iterator, Mediator, Memento.	2	6	6
Тема 5. Поведінкові шаблони Observer, State, Strategy.	2	6	6
Тема 6. Поведінкові шаблони Visitor, Template Method, структурний шаблон Adapter.	2	6	6
Тема 7. Структурні шаблони Bridge, Composite, Decorator.-	2	6	6
Тема 8. Структурні шаблони Facade, Flyweight, Virtual Proxy.	2	7	6
Тема 9. Структурні шаблони Cache Management, Proxy, породжуючий шаблон Object Pool.	2	-	6
Тема 10. Структурний шаблон Dynamic Linkage, шаблони для конкуруючих операцій Single Threated Execution, Lock Object.	2	-	6
Тема 11. Шаблони для конкуруючих операцій Guarder Suspension, Balking, Scheduler.	2	-	6
Тема 12. Шаблони для конкуруючих операцій Read/Write Lock, Producer-Consumer, Two-Phase Termination.	2	-	6
Тема 13. Шаблони для конкуруючих операцій Double Buffering, Asynchronous Processing, Future	2	-	6
Тема 14. Проектування на базі обов'язків (Responsibility-Driven Design). Принципи GRASP. Шаблони Low Coupling, High Cohesion, Information Expert.	2	-	6
Тема 15. Шаблони Creator, Pure Fabrication, Indirection, Polymorphism, Protected Variations, Controller.	2	-	6
Тема 16. Шаблони домена Domain Model, Service Layer, Table Module, Transaction Script, шаблони джерела даних Active Record, Data Mapper	4	-	5
Разом:	34	51	95

5. Програма навчальної дисципліни

5.1 Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
1	Породжуючі шаблони Abstract Factory, Builder. Літ.: [1] с. 65 – 100; [2] с. 24 – 65; [5] с. 9-23.	2
2	Породжуючі шаблони Factory Method, Prototype, Singleton. Літ.: [1] с. 65 - 100 ; [2] с. 24 – 65; [5] с. 23-37.	2
3	Поведінкові шаблони проектування Chain of Responsibility, Command, Interpreter. Літ.: [1] с. 139 – 176; [2] с. 113 – 192; [5] с. 38-63.	2
4	Поведінкові шаблони Iterator, Mediator, Memento. Літ.: [1] с. 139 – 176; [2] с. 113 – 192; [5] с. 63-72.	2
5	Поведінкові шаблони Observer, State, Strategy. Літ.: [1] с. 139 – 176; [2] с. 113 – 192; [5] с. 72-83.	2
6	Поведінкові шаблони Visitor, Template Method. Структурний шаблон Adapter. Літ.: [1] с. 139 – 176; [2] с. 113 – 192; [5] с. 83-94.	2
7	Структурні шаблони Bridge, Composite, Decorator. Літ.: [1] с. 101 – 138; [2] с. 65 – 104; [5] с. 94-102.	2
8	Структурні шаблони Facade, Flyweight, Proxy. Літ.: [1] с. 101 – 138; [2] с. 65 – 104; [5] с. 102-114.	2
9	Структурні шаблони Cache Management, Proxy, породжуючий шаблон Object Pool Літ.: [1] с. 101 – 138; [2] с. 65 – 104; [5] с. 121-141.	2
10	Структурний шаблон Dynamic Linkage, шаблони для конкуруючих операцій Single Threaded Execution, Lock Object Літ.: [1] с. 101 – 138; [2] с. 65 – 104; [5] с. 141-160.	2
11	Шаблони для конкуруючих операцій Guarder Suspension, Balking, Scheduler Літ.: [1] с. 179 – 213; [5] с. 160-175.	2
12	Шаблони для конкуруючих операцій Read/Write Lock, Producer-Consumer, Two-Phase Termination Літ.: [1] с. 179 – 213; [5] с. 175-185.	2
13	Шаблони для конкуруючих операцій Double Buffering, Asynchronous Processing, Future Літ.: [1] с. 179 – 213; [5] с. 185-202.	2
14	Проектування на базі обов'язків (Responsibility-Driven Design). Шаблони Low Coupling, High Cohesion, Information Expert Літ.: [3] с. 111 – 128; [5] с. 204-218.	2
15	Шаблони Creator, Pure Fabrication, Indirection, Polymorphism, Protected Variations, Controller. Літ.: [3] с. 111 – 128; [5] с. 218-236.	2
16	Шаблони домена Domain Model, Service Layer, Table Module, Transaction Script, шаблони джерела даних Active Record, Data Mapper Літ.: [4] с. 171 – 185; [5] с. 238-258.	4
Разом:		34

5.2 Зміст лабораторних занять

№ п/п	Тема лабораторного заняття	Годин
1	Практична реалізація шаблонів Abstract Factory, Builder. Літ.: [1] с. 71 – 78; [2] с. 24 – 38; [5] с. 264-267.	8
2	Практична реалізація шаблонів Factory Method, Prototype, Singleton. Літ.: [1] с. 67 – 71, 78 – 85; [2] с. 38 – 57; [5] с. 267-271.	6
3	Практична реалізація шаблонів Chain of Responsibility, Command, Interpreter. Літ.: [1] с. 143 – 150; [2] с. 114 – 142; [5] с. 271-276.	6
4	Практична реалізація шаблонів Iterator, Mediator, Memento. Літ.: [1] с. 150 – 160; [2] с. 142 – 162; [5] с. 276-280.	6

5	Практична реалізація шаблонів Observer, State, Strategy. Літ.: [1] с. 160 – 169; [2] с. 162 – 179; [5] с. 280-284.	6
6	Практична реалізація шаблонів Visitor, Template Method, Adapter. Літ.: [1] с. 169 – 176, 102 – 106; [2] с. 66 – 72, 186 – 202; [5] с. 284-287.	6
7	Практична реалізація шаблонів Bridge, Composite, Decorator. Літ.: [1] с. 106 – 115; [2] с. 72 – 94; [5] с. 287-291.	6
8	Практична реалізація шаблонів Facade, Flyweight, Proxy. Літ.: [1] с. 115 – 135; [2] с. 94 – 113; [5] с. 291-294.	7
	Разом:	51

5.3 Зміст самостійної (у т.ч. індивідуальної) роботи здобувача вищої освіти

Самостійна робота студентів усіх форм здобуття освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до виконання і захисту лабораторних робіт. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Номер тижня	Вид самостійної роботи	Кількість годин
1	Опрацювання лекційного матеріалу. Шаблони проектування. Вступ в шаблони проектування. Класифікація шаблонів проектування, (шаблони GoF). Породжуючі шаблони Abstract Factory, Builder. Виконання індивідуального завдання.	6
2	Опрацювання лекційного матеріалу. Породжуючі шаблони Factory Method, Prototype, Singleton.	6
3	Опрацювання лекційного матеріалу. Поведінкові шаблони проектування Chain of Responsibility, Command, Interpreter. Підготовка до захисту лабораторної роботи №1. Виконання індивідуального завдання.	6
4	Опрацювання лекційного матеріалу. Поведінкові шаблони Iterator, Mediator, Memento. Виконання індивідуального завдання.	6
5	Опрацювання лекційного матеріалу. Поведінкові шаблони Observer, State, Strategy. Підготовка до захисту лабораторної роботи №2. Виконання індивідуального завдання.	6
6	Опрацювання лекційного матеріалу. Поведінкові шаблони Visitor, Template Method, структурний шаблон Adapter. Виконання індивідуального завдання.	6
7	Опрацювання лекційного матеріалу. Структурні шаблони Bridge, Composite, Decorator. Підготовка до захисту лабораторної роботи №3. Виконання індивідуального завдання.	6
8	Опрацювання лекційного матеріалу. Структурні шаблони Facade, Flyweight, Virtual Proxy. Виконання індивідуального завдання. Виконання індивідуального завдання.	6
9	Опрацювання лекційного матеріалу. Структурні шаблони Cache Management, Proxy, породжуючий шаблон Object Pool. Підготовка до захисту лабораторної роботи №4. Виконання індивідуального завдання. Виконання індивідуального завдання.	6
10	Опрацювання лекційного матеріалу. Структурний шаблон Dynamic Linkage, шаблони для конкуруючих операцій Single Threated Execution, Lock Object. Виконання індивідуального завдання.	6
11	Опрацювання лекційного матеріалу. Шаблони для конкуруючих операцій Guarder Suspension, Balking, Scheduler. Підготовка до захисту лабораторної роботи №5. Виконання індивідуального завдання.	6
12	Опрацювання лекційного матеріалу. Шаблони для конкуруючих операцій Read/Write Lock, Producer-Consumer, Two-Phase Termination. Виконання індивідуального завдання.	6
13	Опрацювання лекційного матеріалу. Шаблони для конкуруючих операцій Double Buffering, Asynchronous Processing, Future. Підготовка до захисту лабораторної роботи №6. Виконання індивідуального завдання.	5
14	Опрацювання лекційного матеріалу. Проектування на базі обов'язків (Responsibility-Driven Design). Принципи GRASP. Шаблони Low Coupling, High Cohesion, Information Expert. Виконання індивідуального завдання.	5
15	Опрацювання лекційного матеріалу. Шаблони Creator, Pure Fabrication, Indirection, Polymorphism, Protected Variations, Controller. Підготовка до захисту лабораторної роботи №7. Виконання індивідуального завдання.	5
16	Опрацювання лекційного матеріалу. Шаблони домена Domain Model, Service Layer, Table Module, Transaction Script, шаблони джерела даних Active Record, Data Mapper. Підготовка до захисту лабораторної роботи №8. Виконання індивідуального завдання. Захист індивідуального завдання. Підготовка до підсумкового контрольного заходу.	8
Разом:		95

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи індивідуальні завдання (ІЗ) (з відповідних тем. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання індивідуального завдання викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного навчання, мотиваційних прийомів,

інформаційно-комунікаційних технологій); лабораторні заняття: з використанням методів комп'ютерного моделювання, аналіз проблемних ситуацій, пояснення, дискусія тощо, самостійна робота: робота над засвоєнням теоретичного матеріалу, виконання індивідуальних та домашніх завдань, підготовка до поточного та підсумкового контролю тощо), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

7. Методи контролю

Поточний контроль здійснюється під час аудиторних лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, в т.ч. з використанням Модульного середовища для навчання. При цьому використовуються такі методи поточного контролю:

- оцінювання результатів захисту лабораторних робіт;
- оцінювання індивідуальних завдань.

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 он-лайн режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до лабораторного заняття (вивчення теоретичного матеріалу з теми роботи, попередню підготовку протоколу роботи, (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, якісно підготувати звіт (протокол роботи відповідно до теми), захистити результати виконаної роботи, брати участь у дискусіях щодо прийнятих конструктивних рішень при виконанні здобувачами лабораторних робіт тощо.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи відповідно до робочої програми навчальної дисципліни. Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. Пропущене лабораторне заняття студент 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 захисту лабораторних робіт та контрольної роботи.

Здобувач вищої освіти, виконуючи самостійну роботу або індивідуальну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, плагіат (в т.ч. із використанням мобільних девайсів)). У разі виявлення 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 *не допускаються*.

У межах вивчення навчальної дисципліни здобувачам вищої освіти передбачено визнання і зарахування результатів навчання, набутих шляхом неформальної освіти, що розміщені на доступних платформах, які сприяють формуванню 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).

Окрім теми курсу можуть бути зараховані у випадку отримання студентом **результатів навчання у неформальній освіті**, що підтверджені відповідним документом (сертифікат, свідоцтво, освітня програма тощо) та відповідно до Положення про порядок визнання та перерахування результатів навчання здобувачів вищої освіти у ХНУ (<https://khmnua.edu.ua/wp-content/uploads/normativni-dokumenty/polozhennya/pro-poryadok-vyznannya-ta-pererakhuvannya-rezultativ-navchannya.pdf>). Перелік лабораторних робіт, що можуть бути перераховані та приклади відповідних курсів на освітніх ресурсах:

Лабораторні роботи 1-3: Design Patterns – <https://www.coursera.org/learn/design-patterns>

Лабораторні роботи 1-6: Software Design and Architecture – <https://www.coursera.org/specializations/software-design-architecture>

Лабораторні роботи 1-3: Software Design Patterns: Best Practices for Software Developer – <https://www.educative.io/courses/software-design-patterns-best-practices>

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 встановлених Робочою програмою для цього виду роботи. При цьому кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

При оцінюванні результатів навчання здобувачів вищої освіти з будь-якого виду навчальної роботи (структурної одиниці) рекомендується використовувати наведені нижче узагальнені критерії:

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>похибки</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

Лабораторна робота								Самостійна робота	Семестровий контроль	Разом
								ІЗ	Іспит	Сума балів
1	2	3	4	5	6	7	8	1		
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	15-20	24-40	
24-40								12-20	24-40	60-100*

Примітки: *За набрану з будь-якого виду навчальної роботи з дисципліни кількість балів, нижче встановленого мінімуму, здобувач отримує незадовільну оцінку і має її перездати у встановлений викладачем (деканом) термін. Інституційна оцінка встановлюється відповідно до таблиці «Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС».

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; знання теоретичного матеріалу; здатність аргументувати прийняті рішення; повнота та якість отриманих результатів; дотримання вимог до оформлення та структурування роботи; уміння виправляти виявлені помилки у ході захисту.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці "Критеріїв оцінювання навчальних досягнень здобувача вищої освіти" (мінімальний позитивний бал – 3 бали, максимальний – 5 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 виконання індивідуального завдання

Виконане та оформлене відповідно до вимог, визначених Методичними рекомендаціями, індивідуальне завдання (ІЗ) комплексно оцінюється викладачем з урахуванням таких критеріїв: самостійність виконання; правильність розв'язання поставлених задач; обґрунтованість вибору методів розв'язання; повнота пояснень та аргументованість відповідей; якість оформлення та дотримання вимог до структури і змісту роботи.

Результат виконання здобувачем вищої освіти кожного ІЗ оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти з урахуванням рівня досягнення запланованих програмних результатів навчання та сформованих компетентностей. За підсумками захисту присвоюється відповідна сума балів (мінімальний позитивний бал – 12 балів, максимальний – 20 балів).

У разі, якщо здобувач вищої освіти виявив рівень знань і виконання ІЗ, що нижчий ніж 60 відсотків від максимальної кількості балів, встановленої Робочою програмою для цієї структурної одиниці, завдання не зараховується. У такому випадку студент має повторно опрацювати зміст завдання, усунути помилки та здати на перевірку доопрацьоване ІЗ у терміни, погоджені з викладачем.

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	3	4	5
Теоретичне питання № 2	3	4	5
Практичне завдання	18	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).

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

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		
D	66-72		Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
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 самоконтролю результатів навчання

1. Для чого використовуються GoF шаблони?
2. Де можна використовувати породжуючі шаблони?
3. Що таке шаблон Abstract Factory і як його використовують?
4. Для чого використовують шаблон Builder?
5. Яке призначення шаблону Factory Method і де його можна застосовувати?
6. Що таке шаблон Prototype і як він використовується?
7. Як працює шаблон Singleton і де його можна застосовувати?
8. Для чого потрібний шаблон Chain of Responsibility і де його використовують?
9. Що таке шаблон Command і в яких сценаріях його варто застосовувати?
10. Який зміст шаблону Interpreter і де його можна застосовувати?
11. Де можна застосовувати поведінкові шаблони?
12. Як використовується шаблон Iterator і де його можна застосувати?
13. Для чого використовується шаблон Mediator і в яких областях його можна використовувати?
14. Що таке шаблон Memento і як його застосовують?
15. Де застосовують шаблон Observer і як він працює?
16. Яким чином працює шаблон State і де його можна використовувати?
17. Для чого використовується шаблон Strategy і в яких сферах його можна використовувати?
18. Що таке шаблон Visitor і як він використовується?
19. Яким чином використовується шаблон Template Method і де його можна застосовувати?
20. Де можна застосовувати структурні шаблони?
21. Що таке шаблон Adapter і в яких сценаріях його варто використовувати?
22. Для чого потрібний шаблон Bridge і в яких областях його використовують?
23. Як використовується шаблон Composite і де його можна застосовувати?
24. Що таке шаблон Decorator і в яких сферах його використовують?
25. Для чого використовується шаблон Facade і де його можна використовувати?
26. Як працює шаблон Flyweight і в яких сферах його можна застосовувати?
27. Для чого використовується шаблон Proxy і в яких сценаріях його варто використовувати?
28. Як використовується діаграма класів у шаблонах проектування?
29. Як використовується діаграма кооперації у шаблонах проектування?
30. Як використовується діаграма послідовності у шаблонах проектування?
31. Що означає шаблон Cache Management і в яких сферах його використовують?
32. Як використовується шаблон Virtual Proxy і в яких сценаріях його варто використовувати?
33. Для чого використовується шаблон Object Pool і де його можна застосовувати?
34. Що означає шаблон Dynamic Linkage і в яких сферах його використовують?
35. Для чого використовується шаблон Single Threaded Execution і в яких сценаріях його варто використовувати?
36. Що таке шаблон Lock Object і в яких сферах його використовують?
37. Як використовується шаблон Guarder Suspension і в яких сценаріях його варто використовувати?
38. Для чого використовується шаблон Balking і в яких областях його можна застосовувати?
39. Що означає шаблон Scheduler і в яких сферах його використовують?
40. Як використовується шаблон Read/Write Lock і в яких сценаріях його варто використовувати?
41. Для чого використовується шаблон Producer-Consumer і в яких сферах його використовують?
42. Що означає шаблон Two-Phase Termination і в яких сценаріях його варто використовувати?
43. Як використовується шаблон Double Buffering і в яких сферах його використовують?

44. Для чого використовується шаблон Asynchronous Processing і в яких областях його можна застосовувати?
45. Що таке шаблон Future і як він використовується?
46. Для чого потрібен шаблон Low Coupling і в яких сферах його використовують?
47. Як використовується шаблон High Cohesion і в яких сценаріях його варто використовувати?
48. Для чого використовується шаблон Information Expert і в яких областях його можна застосовувати?
49. Що означає шаблон Creator і в яких сферах його використовують?
50. Для чого використовується шаблон Pure Fabrication і в яких сценаріях його варто використовувати?
51. Що означає шаблон Indirection і в яких сферах його використовують?
52. Як використовується шаблон Polymorphism і в яких областях його можна застосовувати?
53. Для чого використовується шаблон Protected Variations і в яких сценаріях його варто використовувати?
54. Що таке шаблон Controller і в яких сферах його використовують?
55. Для чого потрібен шаблон Domain Model і в яких областях його використовують?
56. Як використовується шаблон Service Layer і в яких сценаріях його варто використовувати?
57. Що означає шаблон Table Module і в яких сферах його використовують?
58. Для чого використовується шаблон Transaction Script і в яких сценаріях його варто використовувати?
59. Як використовується шаблон Active Record і в яких областях його використовують?
60. Для чого використовується шаблон Data Mapper і в яких сценаріях його варто використовувати?

11. Навчально-методичне забезпечення

Освітній процес з дисципліни «Технології створення програмних продуктів» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафедри підготовлені і видані такі роботи:

1. Манзюк Е.А., Бармак О.В., Радюк П.М., Молчанова М.О., Пасічник О.А., Багрій Р.О. Технології створення програмних продуктів. Теоретичний курс та лабораторний практикум: навч. посібник. Хмельницький: ХНУ, 2024р. 306с.
2. Курс «Технології створення програмних продуктів». <https://msn.khmnmu.edu.ua/course/view.php?id=2609>

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

Необхідні інструменти, обладнання, програмне забезпечення: комп'ютер (надається для використання в лабораторіях кафедри комп'ютерних наук), Eclipse IDE має ліцензію Eclipse Public License (EPL), версії 2.0 (це ліцензія з відкритим кодом, яка дозволяє вільно використовувати, модифікувати і поширювати програмне забезпечення), текстовий редактор (пропонується використання безкоштовних онлайн-сервісів).

13. Рекомендована література:

Основна

1. Wengner M. Practical Design Patterns for Java Developers. – Packt Publishing, 2023. – 266p.
2. Jain H., Semwal V. Design Patterns using Java Paperback. – Hemant Jain, 2023. – 289p.
3. Bass J.M. Agile Software Engineering Skills. – Springer International Publishing, 2023 – 317p.
4. Khononov V. Learning Domain-Driven Design. – O'Reilly Media, 2021 – 342p
5. Манзюк Е.А., Бармак О.В., Радюк П.М., Молчанова М.О., Пасічник О.А., Багрій Р.О. Технології створення програмних продуктів. Теоретичний курс та лабораторний практикум: навч. посібник. Хмельницький: ХНУ, 2024р. 306с.

Додаткова

1. Carducci M. Mastering Software Architecture A Comprehensive New Model and Approach. – Apress, 2025. – 451p.
2. Varyani D. Design Patterns in Action Implementing Effective Design Patterns in Java. – Notion Press, 2025. – 238p.

14. Інформаційні ресурси

1. Модульне середовище для навчання. URL: <https://msn.khmnmu.edu.ua/course/view.php?id=2609>
2. Електронна бібліотека ХНУ. URL: <http://library.khmnmu.edu.ua>
3. Інституційний репозитарій ХНУ. URL: <https://elar.khmnmu.edu.ua/home>

ТЕХНОЛОГІЇ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	7
Кількість призначених кредитів ЄКТС	6
Форми здобуття освіти, для яких викладається дисципліна	Очна (денна)

Результати навчання. Після вивчення дисципліни студент повинен: розробляти програмні моделі предметних середовищ із застосуванням відповідних шаблонів проєктування; обирати оптимальні парадигми програмування та архітектурні рішення з позицій якості та зручності застосування; реалізовувати породжуючі, поведінкові та структурні шаблони проєктування; застосовувати принципи проєктування та методи responsibility-driven design; проєктувати програмні системи з урахуванням паралельних та розподілених обчислень; використовувати шаблони для конкуруючих операцій при розробці багатопотокового програмного забезпечення; аналізувати та моделювати бізнес-процеси через призму архітектури програмних продуктів; оцінювати ризики проєктування та обирати оптимальні технологічні рішення для реалізації програмних систем.

Зміст навчальної дисципліни. Шаблони проєктування. Класифікація шаблонів проєктування, (шаблони GoF). Породжуючі шаблони. Поведінкові шаблони. Структурні шаблони. Шаблони Abstract Factory, Builder, Factory Method, Prototype, Singleton, Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Visitor, Template Method, Adapter, Bridge, Composite, Decorator, Facade, Flyweight, Proxy. Cache Management, Proxy, Object Pool, Dynamic Linkage, Single Threaded Execution, Lock Object, Guarder Suspension, Balking, Scheduler, Read/Write Lock, Producer-Consumer, Two-Phase Termination, Double Buffering, Asynchronous Processing, Future. Принципи GRASP, Шаблони Low Coupling, High Cohesion, Information Expert, Creator, Pure Fabrication, Indirection, Polymorphism, Protected Variations, Controller, Domain Model, Service Layer, Table Modyle, Transaction Script, шаблони джерела даних Active Record, Data Mapper, Row Data Gateway, Table Data Gateway.

Пререквізити – вебтехнології (ОПП.02), технології розподілених систем та паралельних обчислень (ОПП.11), теорія алгоритмів (ОЗП.08), об'єктно-орієнтоване проєктування (ОПП.03), алгоритмізація та програмування (ОЗП.05).

Кореквізити – управління ІТ-проєктами (ОПП.04), інформаційні технології хмарних обчислень (ОПП.13), кваліфікаційна робота (ОПП.17), професійна практика (ОПП.16).

Запланована навчальна діяльність: Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для *першого* (бакалаврського) рівня вищої освіти за денною формою здобуття освіти становить 10 годин на 1 кредит ЄКТС.

Форми (методи) навчання: лекції (з використанням методів візуалізації, проблемного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття: з використанням методів комп'ютерного моделювання, аналіз проблемних ситуацій, пояснення, дискусія тощо, самостійна робота: робота над засвоєнням теоретичного матеріалу, виконання індивідуальних та домашніх завдань, підготовка до поточного та підсумкового контролю тощо) з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

Форми оцінювання результатів навчання: захист лабораторних робіт.

Вид семестрового контролю: іспит.

Навчальні ресурси:

1. Wengner M. Practical Design Patterns for Java Developers. – Packt Publishing, 2023. – 266p.
2. Jain H., Semwal V. Design Patterns using Java Paperback. – Hemant Jain, 2023. – 283p.
3. Bass J.M. Agile Software Engineering Skills. – Springer International Publishing, 2023 – 317p.
4. Khononov V. Learning Domain-Driven Design. – O'Reilly Media, 2021 – 342p
5. Манзюк Е.А., Бармак О.В., Радюк П.М., Молчанова М.О., Пасічник О.А., Багрій Р.О. Технології створення програмних продуктів. Теоретичний курс та лабораторний практикум: навч. Посібник. Хмельницький: ХНУ, 2024р. 306с.
6. Модульне середовище для навчання. URL: <https://msn.khmnmu.edu.ua/course/view.php?id=2609>
7. Електронна бібліотека університету. URL: <http://library.khmnmu.edu.ua>

Викладач: д-р. техн. н., проф., Едуард МАНЗЮК