



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА
ФАКУЛЬТЕТ УПРАВЛІННЯ ФІНАНСАМИ ТА БІЗНЕСУ

ЗАТВЕРДЖУЮ
Декан
доц. Андрій СТАСИШИН
“30” січня 2024 р.

РОБОЧА
ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об’єктно-орієнтований дизайн

(назва навчальної дисципліни)

галузь знань: 05 “Соціальні та поведінкові науки”
(шифр та найменування галузі знань)

спеціальність: 051 “Економіка”
(код та найменування спеціальності)

освітня програма: Інформаційні технології в бізнесі
(найменування спеціалізації)

освітній ступінь: магістр
(бакалавр/магістр)

форма навчання: денна
(денна, заочна)

ЛЬВІВ 2024

КАФЕДРА ЦИФРОВОЇ ЕКОНОМІКИ ТА
БІЗНЕС-АНАЛІТИКИ

Робоча програма навчальної дисципліни “Об’єктно-орієнтований дизайн” для студентів, які навчаються за галуззю знань 05 “Соціальні та поведінкові науки” спеціальністю 051 “Економіка” освітньою програмою “Інформаційні технології в бізнесі” освітнього ступеня магістр.

“15” січня 2024 року – 24 с.

Розробник: Шевчук І.Б., завідувач кафедри цифрової економіки та бізнес-аналітики, д.е.н., професор.

Розглянуто та ухвалено на засіданні кафедри цифрової економіки та бізнес-аналітики

Протокол № 6 від “16” січня 2024 р.

Завідувач кафедри


(підпис)

Ірина ШЕВЧУК

(прізвище, ініціали)

Розглянуто та ухвалено Вченою радою факультету управління фінансами та бізнесу

Протокол № 6 від “30” січня 2024 р.

© Шевчук І.Б., 2024 рік

© ЛНУ імені Івана Франка, 2024 рік

ЗМІСТ

1. ПОЯСНЮВАЛЬНА ЗАПИСКА	4
2. ОПИС ПРЕДМЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	7
3. ТЕМАТИЧНИЙ ПЛАН НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	8
4. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	8
5. СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ	10
6. ГРАФІК РОЗПОДІЛУ НАВЧАЛЬНОГО ЧАСУ ЗА ОСВІТНЬОЮ ПРОГРАМОЮ ТА ВИДАМИ НАВЧАЛЬНОЇ РОБОТИ	11
7. КАЛЕНДАРНО-ТЕМАТИЧНИЙ ПЛАН АУДИТОРНИХ ЗАНЯТЬ	12
7.1. Календарно-тематичний план лекційних занять	12
7.2. Календарно-тематичний план лабораторних занять, заліків по модулях, контрольних робіт	14
7.3. Графік консультацій	15
8. ПЕРЕЛІК ПИТАНЬ, ЩО ВІНОСЯТЬСЯ НА ПІДСУМКОВИЙ КОНТРОЛЬ	15
9. МЕТОДИ ОЦІНЮВАННЯ ЗНАТЬ СТУДЕНТІВ	17
9.1. Таблиця оцінювання (визначення рейтингу) навчальної діяльності студентів	18
9.2. Система нарахування рейтингових балів та критерії оцінювання знань студентів	18
9.3. Шкала оцінювання успішності студентів за результатами підсумкового контролю	20
10. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	20
11. МЕТОДИКИ АКТИВІЗАЦІЇ ПРОЦЕСУ НАВЧАННЯ	21
12. РЕСУРСИ МЕРЕЖІ ІНТЕРНЕТ	23
13. ЗМІНИ І ДОПОВНЕННЯ ДО РОБОЧОЇ ПРОГРАМИ	24

1. ПОЯСНОВАЛЬНА ЗАПИСКА

Навчальна програма дисципліни “Об’єктно-орієнтований дизайн” є складовою методичного забезпечення навчального процесу для підготовки магістрів галузі знань 05 «Соціальні та поведінкові науки» за спеціальністю 051 «Економіка» освітньою програмою «Інформаційні технології в бізнесі».

Предмет навчальної дисципліни

Предметом дисципліни “Об’єктно-орієнтований дизайн” є методи та прийоми розробки складних програмних систем з сучасним інтерфейсом.

Мета навчальної дисципліни

Мета вивчення дисципліни “Об’єктно-орієнтований дизайн” – ознайомлення студентів із сучасними методами проектування програмного забезпечення, що дозволяють здійснювати розробку програмних систем середньої та високої складності.

Основні завдання

Основні завдання дисципліни “Об’єктно-орієнтований дизайн”: ознайомлення студентів із сучасними процесами розробки програмного забезпечення (ПЗ), орієнтованими на наскрізне проектування; вивчення основ мови моделювання UML; вивчення принципів аналізу та проектування об’єктно-орієнтованих систем; вивчення найбільш поширених шаблонів проектування та архітектурних шаблонів.

Місце навчальної дисципліни в структурно-логічній схемі

Дисципліна “Об’єктно-орієнтований дизайн” взаємопов’язана з такими дисциплінами як “Технології створення програмних та інтелектуальних систем”, “Проектний менеджмент”, “ІТ-стратегії бізнесу”, “Ефективність інформаційних систем”, “Сучасні інформаційні технології”.

Вимоги до знань і умінь

При вивченні дисципліни «Об’єктно-орієнтований дизайн» **здобувачі вищої освіти набувають такі компетентності (здатність):**

ІК1 – Здатність визначати та розв’язувати складні економічні задачі та проблеми, приймати відповідні аналітичні та управлінські рішення у сфері економіки або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій за невизначених умов та вимог.

ЗК1 – Здатність генерувати нові ідеї (креативність).

ЗК4 – Здатність спілкуватися з представниками інших професійних груп

різного рівня (з експертами з інших галузей знань/видів економічної діяльності).

ЗК5 – Здатність працювати в команді.

ЗК6 – Здатність розробляти та управляти проектами.

СК4 – Здатність використовувати сучасні інформаційні технології, методи та прийоми дослідження економічних та соціальних процесів, адекватні встановленим потребам дослідження.

СК11 – Здатність планувати і розробляти проекти у сфері економіки, здійснювати її інформаційне, методичне, матеріальне, фінансове та кадрове забезпечення.

СК12 – Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.

СК14 – Здатність формувати нові конкурентоспроможні ідеї й реалізовувати їх у проектах (стартапах).

СК15 – Здатність самостійно опановувати нові знання, використовуючи сучасні освітні та дослідницькі технології у сфері економіки й інформаційних технологій та приймати ефективні управлінські рішення.

Програмні результати навчання:

ПР03 – Вільно спілкуватися з професійних та наукових питань державною та іноземною мовами усно і письмово.

ПР06 – Оцінювати результати власної роботи, демонструвати лідерські навички та уміння управляти персоналом і працювати в команді.

ПР17 – Керувати технологічними процесами ІТ виробництва та здійснювати контроль якості програмної продукції.

ПР18 – Виявляти основні тенденції в ІТ-сфері та бізнес-перспективи використання конкретних інформаційних технологій в умовах динамічності ринкового середовища.

ПР19 – Розробляти управлінські рішення щодо забезпечення розвитку державних, муніципальних, комерційних організацій та бізнес-структур в усіх галузях економіки через призму інформаційних технологій; усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань.

Вивчення навчальної дисципліни передбачає досягнення такого кваліфікаційного рівня підготовки студента, за якого він повинен:

а) знати

- сучасні засоби моделювання і документування архітектури ПЗ;

- методи побудови та аналізу формальних моделей предметних областей;
- основні фази процесу розробки ПЗ;
- основи мови UML;
- принципи об'єктно-орієнтованого проектування;
- принципи і методи створення складних програмних засобів на основі концепції та стандартів відкритих систем, CASE-систем, мови UML;
- функціональні можливості однієї з програм-редакторів мови UML;
- основні архітектурні шаблони, які використовуються при розробці програмних систем;
- шаблони проектування, що використовуються при розробці систем;

б) уміти

- розробляти моделі компонентів інформаційних систем, включаючи моделі баз даних;
- обґрунтовувати прийняті проектні рішення, здійснювати постановку і виконувати експерименти з перевірки їх коректності та ефективності;
- моделювати, аналізувати та документувати вимоги до коректності та ефективності ПЗ;
- методи і засоби організації колективної роботи над проектом по розробці ПЗ;
- застосовувати на практиці фундаментальні концепції об'єктно-орієнтованого підходу: абстракцію, інкапсуляцію, успадкування, поліморфізм;
- проводити аналіз вимог і проектувати, на основі даних аналізу, системи і підсистеми з використанням мови моделювання UML;
- розробляти логічну модель програмної системи;
- розробляти компонентну модель програмної системи;
- розробляти модель розміщення компонентів програмної системи;
- застосовувати шаблони проектування та архітектурні шаблони.

Опанування навчальною дисципліною повинно забезпечувати необхідний рівень сформованості вмінь:

Назва рівня сформованості вміня	Зміст критерію рівня сформованості вміня
1. Репродуктивний	Вміння відтворювати знання, передбачені даною програмою
2. Алгоритмічний	Вміння використовувати знання в практичній

	діяльності при розв'язуванні типових ситуацій
3. Творчий	Здійснювати евристичний пошук і використовувати знання для розв'язання нестандартних завдань та проблемних ситуацій

Навчальна програма складена на **3 кредити**.

Форми контролю – проміжний модульний контроль, залік.

2. ОПИС ПРЕДМЕТА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ “ОБ’ЄКТНО-ОРІЄНТОВАНИЙ ДИЗАЙН”

Характеристика навчальної дисципліни							
Шифр та найменування галузі знань: 05 „Соціальні та поведінкові науки”				Цикл дисциплін за навчальним планом: Цикл професійної та практичної підготовки			
Код та назва спеціальності: 051 „Економіка”				Освітній ступінь: магістр			
Освітня програма: „Інформаційні технології в бізнесі”							
Курс: _____ 1 _____ Семестр: _____ II _____				Методи навчання: Лекції, лабораторні заняття, самостійна робота, робота в бібліотеці, Інтернеті тощо.			
Кількість кредитів ECTS	Кількість годин	Кількість аудиторних годин	Лекції	Семинари, практичні, лабораторні	Заліки по модулях (контрольні роботи)	Самостійна робота студента (СРС)	Індивідуальна робота студента (ІНДЗ)
3	90	32	16	14	2	30	28
Кількість тижневих годин		Кількість змістових модулів (тем)		Кількість заліків по модулях /контрольних робіт		Вид контролю	
2		8		1		ПМК, залік	

3. ТЕМАТИЧНИЙ ПЛАН НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Номер теми	Назва теми
Тема 1.	Еволюція методологій програмування
Тема 2.	Об'єкти та класи
Тема 3.	Об'єктно-орієнтований аналіз
Тема 4.	Принципи об'єктно-орієнтованого дизайну
Тема 5.	Уніфікована мова моделювання UML
Тема 6.	Об'єктно-реляційне відображення
Тема 7.	Шаблони проектування
Тема 8.	Метрики

4. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Еволюція методологій програмування

Покоління мов програмування. Об'єктна модель. Об'єктні мови програмування. Об'єктно-орієнтовані мови. Об'єктно-орієнтований аналіз, дизайн та проектування. Парадигми програмування.

Складові частини об'єктного підходу. Абстрагування. Інкапсуляція. Модульність. Ієрархія. Типізація. Паралелізм. Збережуваність.

Тема 2. Об'єкти та класи

Поняття об'єкту в ООП. Стан та поведінка об'єкту. Ідентичність. Життєвий цикл об'єкту. Відношення між об'єктами. Взаємодія клієнт-сервер. Ієрархія об'єктів.

Структура класів. Абстрактні класи, інтерфейси та класи-утиліти. Відношення між класами: асоціація, агрегація та композиція, використання, наслідування. Інстанціювання.

Тема 3. Об'єктно-орієнтований аналіз

Цілі і завдання фази об'єктно-орієнтованого аналізу. Процес переходу від фази збору вимог до фази аналізу. Аналітична модель. Аналітичні класи і відносини між ними. Реалізації варіантів використання. Колаборації. Опис колаборацій за допомогою діаграм взаємодії.

Тема 4. Принципи об'єктно-орієнтованого дизайну

Принципи дизайну класів. Правило цілісності абстракції. Закон Деметри. Принцип закритості. Принцип підстановки (Liskov). Принцип розділення інтерфейсів.

Принципи проектування залежностей. Принцип інверсії залежності. Принцип ациклічності залежностей.

Принципи проектування пакетів. Загальний принцип повторного використання. Принцип локалізації змін. Принцип стабільності залежностей. Стабільність. Принцип стабільних абстракцій. Абстрактність. Генеральна послідовність.

Тема 5. Уніфікована мова моделювання UML

Мова UML. Основні функції UML. Загальна структура мови UML. Типи схем UML. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень. Типи відношень в UML. Механізми розширення UML.

Види діаграм в UML. Діаграма класів. Діаграма компонентів. Діаграма композитної / складеної структури. Діаграма кооперації (UML2.0). Діаграма розгортання. Діаграма об'єктів. Діаграма пакетів. Діаграма діяльності. Діаграма станів (або скінчених автоматів). Діаграма прецедентів (або варіантів використання). Діаграма кооперації (UML1.x) / комунікації (UML2.0). Діаграма огляду взаємодії (UML2.0). Діаграма послідовності. Діаграма синхронізації (UML2.0). Особливості графічного зображення діаграм мови UML. Пакети в UML. Редактор UML-діаграм.

Тема 6. Об'єктно-реляційне відображення

Об'єктно-орієнтовані БД. Об'єктно-реляційна БД. Що таке ORM? Парадигма «невідповідності». Принцип роботи ORM. Переваги і недоліки використання. Список програм об'єктно-реляційного відображення.

Тема 7. Шаблони проектування

Визначення, класифікація та опис шаблонів проектування. Архітектурні шаблони. Породжуючі шаблони. Структурні шаблони. Шаблони поведінки.

Тема 8. Метрики

Кількісні метрики. Метрики складності управління програмами. Метрики складності потоку даних програми. Комбіновані метрики складності управління і даних. Об'єктно-орієнтовані метрики і метрики надійності. Гібридні метрики.

5. СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

Основна:

1. Ajit Singh, Ms. Anamika. Object Oriented Modeling and Design Using UML. 2nd Edition. Independently published. 2023. 153 p.
2. Craig Larman. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Pearson, 2014. 616 p.
3. Dean Leffingwell, Don Widrig. Managing Software Requirements: A Unified Approach. Addison-Wesley Professional, 2003. 560 p.
4. Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Addison-Wesley Professional, 2005. 496 p.
5. Grady Booch, Robert A. Maksimchuk, Michael W. Engel, Bobbi J. Young, Jim Conallen, Kelli A. Houston. Object-Oriented Analysis and Design with Applications. Addison-Wesley Professional, 2007. 720 p.
6. James Rumbaugh, Ivar Jacobson, Grady Booch The Unified Modeling Language Reference Manual 2nd Edition. Addison-Wesley Professional; 2nd edition. 2004. 721 p.
7. Janis Osis, Uldis Donins. Approach for Domain Modeling and Software Development. Elsevier Science Publishing Co Inc. 2017. 276 p.
8. Jean-Marc Nerson. Enterprise Model Patterns: Describing the World. Technics Publications, 2014. 648 p.
9. Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language. Addison-Wesley Professional, 2003. 208 p.
10. Pender T.A. UML Weekend Crash Course. Wiley Publishing Inc., 2002. 358 p.
11. Stephen J. Mellor, Marc J. Balcer, Ivar Jacobson. Executable UML: A Foundation for Model-Driven Architecture. Independently published. 2022. 402 p.
12. Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник / В.С. Авраменко, А.С. Авраменко. Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с.
13. Омельчук Л.Л. Об'єктно-орієнтоване програмування. Лабораторний практикум: навчальний посібник / Л.Л. Омельчук, А.С. Белова. Київ: 2022. 271 с.
14. Роберт С. Мартін. Чиста архітектура. Фабула. 2019. 368 с.

Додаткова:

1. Dean Leffingwell, Don Widrig. Managing Software Requirements: A Unified Approach. Addison-Wesley Professional, 2003. 560 p.
2. Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. Addison-Wesley Professional, 2005. 496 p.
3. Krishna Rungta. UML 2.0 : Learn UML in 1 Day. Independently Published. 2019. 112 p.

4. Larman Craig. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Pearson, 2004. 616 p.
5. Norman Daoust. UML Requirements Modeling for Business Analysts : Steps to Modeling Success. Technics Publications LLC. 2012. 268 p.
6. Pender T.A. UML Weekend Crash Course. Wiley Publishing Inc., 2002. 358 p.
7. Scott W. Ambler. The Object Primer : Agile Model-Driven Development with UML 2.0. Cambridge University Press. 2015. 572 p.
8. Suriya Sundaramoorthy. UML Diagramming: A Case Study Approach Auerbach Publications; 1st edition. 416 p.
9. Інструментальні програмні засоби розробки ІУС. Методичні вказівки до виконання лабораторних робіт. / уклад.: К.І. Київська. – Київ: КНУБА, 2018. 40 с.
10. Райчев І.Е., Харченко О.Г., Замковий В.В. Принципи проектування відкритих розподілених систем : навч. посіб. -К.: Вид-во Нац. авіац. ун-ту "НАУ-друк", 2010. 240 с

6. ГРАФІК РОЗПОДІЛУ НАВЧАЛЬНОГО ЧАСУ ЗА ОСВІТНЬОЮ ПРОГРАМОЮ ТА ВИДАМИ НАВЧАЛЬНОЇ РОБОТИ

№ розділу, теми (змістові модулі)	Назва розділу, теми (змістового модуля)	Кількість годин за ОПІ			Розподіл аудиторних годин		
		всього	у тому числі		лекції	лабораторні	заліки по модулях, контрольні (контрольні роботи)
			аудиторні	СРС/ІР			
ЗАЛКОВИЙ МОДУЛЬ № 1							
Тема 1.	Еволюція методологій програмування	4	2	2/-	1	1	-
Тема 2.	Об'єкти та класи	6	2	4/-	1	1	-
Тема 3.	Об'єктно-орієнтований аналіз	8	4	4/-	2	2	-
Тема 4.	Принципи об'єктно-орієнтованого дизайну	8	4	4/-	2	2	-
Тема 5.	Уніфікована мова моделювання UML	12	8	4/-	4	8	-
Тема 6.	Об'єктно-реляційне відображення	6	2	4/-	2	-	-

Тема 7.	Шаблони проектування	6	2	4/-	2	-	-
Тема 8.	Метрики	8	4	4/-	2	-	2
Індивідуальне навчально-дослідне завдання (ІНДЗ)		28	-	-/28	-	-	-
Разом годин		90	32	58	16	14	2

7. КАЛЕНДАРНО-ТЕМАТИЧНИЙ ПЛАН АУДИТОРНИХ ЗАНЯТЬ

7.1. Календарно-тематичний план лекційних занять

№ заняття	Тема та короткий зміст заняття	Кількість годин
1	2	3
ЗАЛКОВИЙ МОДУЛЬ № 1		
Тема 1. Еволюція методологій програмування		1
1.	Покоління мов програмування. Об'єктна модель. Об'єктні мови програмування. Об'єктно-орієнтовані мови. Об'єктно-орієнтований аналіз, дизайн та проектування. Парадигми програмування. Складові частини об'єктного підходу. Абстрагування. Інкапсуляція. Модульність. Ієрархія. Типізація. Паралелізм. Збережуваність.	1
Тема 2. Об'єкти та класи		1
	Поняття об'єкту в ООП. Стан та поведінка об'єкту. Ідентичність. Життєвий цикл об'єкту. Відношення між об'єктами. Взаємодія клієнт-сервер. Ієрархія об'єктів. Структура класів. Абстрактну класи, інтерфейси та класи-утиліти. Відношення між класами: асоціація, агрегація та композиція, використання, наслідування. Інстанціювання.	1
Тема 3. Об'єктно-орієнтований аналіз		2
2.	Цілі і завдання фази об'єктно-орієнтованого аналізу. Процес переходу від фази збору вимог до фази аналізу. Аналітична модель. Аналітичні класи і відносини між ними. Реалізації варіантів використання. Колаборації. Опис колаборацій за допомогою діаграм взаємодії.	2

Тема 4. Принципи об'єктно-орієнтованого дизайну		2
3.	Принципи дизайну класів. Правило цілісності абстракції. Закон Деметри. Принцип закритості. Принцип підстановки (Liskov). Принцип розділення інтерфейсів. Принципи проектування залежностей. Принцип інверсії залежності. Принцип ациклічності залежностей. Принципи проектування пакетів. Загальний принцип повторного використання. Принцип локалізації змін. Принцип стабільності залежностей. Стабільність. Принцип стабільних абстракцій. Абстрактність. Генеральна послідовність.	2
Тема 5. Уніфікована мова моделювання UML		4
4.	Мова UML. Основні функції UML. Загальна структура мови UML. Типи схем UML. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень. Типи відношень в UML. Механізми розширення UML. Види діаграм в UML. Діаграма класів. Діаграма компонентів. Діаграма композитної / складеної структури. Діаграма кооперації (UML2.0). Діаграма розгортання. Діаграма об'єктів.	2
5.	Діаграма пакетів. Діаграма діяльності. Діаграма станів (або скінчених автоматів). Діаграма прецедентів (або варіантів використання). Діаграма кооперації (UML1.x) / комунікації (UML2.0). Діаграма огляду взаємодії (UML2.0). Діаграма послідовності. Діаграма синхронізації (UML2.0). Особливості графічного зображення діаграм мови UML. Пакети в UML. Редактор UML-діаграм.	2
Тема 6. Об'єктно-реляційне відображення		2
6.	Об'єктно-орієнтовані БД. Об'єктно-реляційна БД. Що таке ORM? Парадигма «невідповідності». Принцип роботи ORM. Переваги і недоліки використання. Список програм об'єктно-реляційного відображення.	2
Тема 7. Шаблони проектування		2
7.	Визначення, класифікація та опис шаблонів проектування. Архітектурні шаблони. Породжуючі шаблони. Структурні шаблони. Шаблони поведінки.	2
Тема 8. Метрики		2
8.	Кількісні метрики. Метрики складності управління програмами. Метрики складності потоку даних програми.	2

Комбіновані метрики складності управління і даних. Об'єктно-орієнтовані метрики і метрики надійності. Гібридні метрики.	
Разом годин	16

7.2. Календарно-тематичний план лабораторних занять, заліків по модулях, контрольних робіт

№ заняття	Тема лабораторного заняття. Контрольні роботи (заліки по модулях)	Кількість годин
1	2	3
ЗАЛІКОВИЙ МОДУЛЬ № 1		
	Тема 1. Еволюція методологій програмування. Тема 2. Об'єкти та класи	2
1.	Дослідження предметної області. Створення глосарію.	2
	Тема 3. Об'єктно-орієнтований аналіз	2
2.	Визначення функціональних та нефункціональних вимог до програмного продукту, виділення дійових осіб та їх інтересів	2
	Тема 4. Принципи об'єктно-орієнтованого дизайну	2
3.	Складання описів варіантів використання	2
	Тема 5. Уніфікована мова моделювання UML	8
4.	Діаграма класів. Діаграма об'єктів	2
5.	Діаграма використання	2
6.	Діаграма послідовності	2
7.	Діаграма діяльності	2
	Тема 8. Метрики	2
8.	Контрольна робота (залік по модулю)	2
	Разом лабораторних занять	14
	Разом контрольні роботи, заліки по модулях (ЗМ)	2
	Разом годин	16

7.3. Графік консультацій

№ з/п	Назва розділу, теми, зміст консультації	К-ть годин
1.	Консультація до тем 1-8	2
2.	Консультації по виконанню лабораторних робіт	2
3.	Консультація по виконанню ІЗ.	1
4.	Консультація по організації та виконанню самостійної роботи студентів	0,5
	Разом годин	5,5

8. ПЕРЕЛІК ПИТАНЬ, ЩО ВІНОСЯТЬСЯ НА ПІДСУМКОВИЙ КОНТРОЛЬ

1. Покоління мов програмування.
2. Об'єктна модель.
3. Об'єктні мови програмування.
4. Об'єктно-орієнтовані мови.
5. Об'єктно-орієнтований аналіз, дизайн та проектування.
6. Парадигми програмування.
7. Складові частини об'єктного підходу.
8. Абстрагування.
9. Інкапсуляція.
10. Модульність.
11. Ієрархія.
12. Типізація.
13. Паралелізм.
14. Збережуваність.
15. Поняття об'єкту в ООП.
16. Стан та поведінка об'єкту. Ідентичність.
17. Життєвий цикл об'єкту.
18. Відношення між об'єктами.
19. Взаємодія клієнт-сервер.
20. Ієрархія об'єктів.
21. Структура класів.
22. Абстрактні класи, інтерфейси та класи-утиліти.
23. Відношення між класами: асоціація, агрегація та композиція, використання, наслідування.
24. Інстанціювання.
25. Цілі і завдання фази об'єктно-орієнтованого аналізу.

26. Процес переходу від фази збору вимог до фази аналізу.
27. Аналітична модель.
28. Аналітичні класи і відносини між ними.
29. Реалізації варіантів використання.
30. Колаборації.
31. Опис колаборацій за допомогою діаграм взаємодії.
32. Принципи дизайну класів.
33. Правило цілісності абстракції.
34. Закон Деметри.
35. Принцип закритості.
36. Принцип підстановки (Liskov).
37. Принцип розділення інтерфейсів.
38. Принципи проектування залежностей.
39. Принцип інверсії залежності.
40. Принцип ациклічності залежностей.
41. Принципи проектування пакетів.
42. Загальний принцип повторного використання.
43. Принцип локалізації змін.
44. Принцип стабільності залежностей. Стабільність.
45. Принцип стабільних абстракцій. Абстрактність.
46. Генеральна послідовність.
47. Мова UML.
48. Основні функції UML.
49. Загальна структура мови UML.
50. Типи схем UML.
51. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень.
52. Типи відношень в UML.
53. Механізми розширення UML.
54. Види діаграм в UML.
55. Діаграма класів.
56. Діаграма компонентів.
57. Діаграма композитної / складеної структури.
58. Діаграма кооперації (UML2.0).
59. Діаграма розгортання.
60. Діаграма об'єктів.
61. Діаграма пакетів.
62. Діаграма діяльності.
63. Діаграма станів (або скінченних автоматів).
64. Діаграма прецедентів (або варіантів використання).

65. Діаграма кооперації (UML1.x) / комунікації (UML2.0).
66. Діаграма огляду взаємодії (UML2.0).
67. Діаграма послідовності.
68. Діаграма синхронізації (UML2.0).
69. Особливості графічного зображення діаграм мови UML.
70. Пакети в UML.
71. Редактор UML-діаграм.
72. Об'єктно-орієнтовані БД.
73. Об'єктно-реляційна БД.
74. Що таке ORM?
75. Парадигма «невідповідності».
76. Принцип роботи ORM.
77. Переваги і недоліки використання.
78. Визначення, класифікація та опис шаблонів проектування.
79. Архітектурні шаблони.
80. Породжуючі шаблони.
81. Структурні шаблони.
82. Шаблони поведінки.
83. Кількісні метрики.
84. Метрики складності управління програмами.
85. Метрики складності потоку даних програми.
86. Комбіновані метрики складності управління і даних.
87. Об'єктно-орієнтовані метрики і метрики надійності.
88. Гібридні метрики.

9. МЕТОДИ ОЦІНЮВАННЯ ЗНАТЬ СТУДЕНТІВ

Методи оцінювання знань студентів: поточний контроль, оцінка за індивідуальне завдання та виконану самостійну роботу, контрольну роботу та підсумковий модульний контроль.

Бали студентам нараховуються за:

1. здачу лабораторних робіт,
2. виконання самостійних робіт та індивідуального завдання,
3. написання контрольної роботи та контрольної модульної роботи.

Порядок вивчення та оцінювання дисципліни доводиться до відома студентів протягом семестру.

**9.1. Таблиця оцінювання (визначення рейтингу)
навчальної діяльності студентів**

Поточний та модульний контроль Заліковий модуль № 1		СР	ІНДЗ	РАЗОМ – 100 балів
Лабораторні роботи	КМР			
36	40	10	14	

**9.2. Система нарахування рейтингових балів та критерії оцінювання
знань студентів**

№ з/п	Види робіт. Критерії оцінювання знань студентів	Бали рейтингу	Максимальн а кількість балів
1. Бали поточної успішності за участь у лабораторних заняттях			
Критерії оцінювання		6 балів	
	лабораторна робота виконана у зазначений термін, у повному обсязі, без помилок	6	
	лабораторна робота виконана у зазначений термін, у повному обсязі, але є незначні помилки	4	
	лабораторна робота виконана у неповному обсязі, або (та) з порушенням терміну її виконання, або (та) при наявності значних помилок	3	
	виконання пропущеної без поважних причин лабораторної роботи або повторне виконання незарахованої лабораторної роботи	2	
	лабораторна робота не виконана або не зарахована	0-1	
2. Самостійна робота студентів (СРС)			
Критерії оцінювання		10 балів	
	завдання зроблене повністю та здане вчасно, якісно оформлено звіт	10	
	завдання зроблене, але є незначні помилки в процесі побудови коду програми або неналежно оформлений звіт;	8	
	завдання зроблене, але є незначні помилки в процесі побудови коду програми та немає звіту;	6	
	завдання зроблене, але є суттєві помилки в процесі побудови коду програми та немає звіту;	4	
	завдання не виконане або тільки розпочато процес побудови коду програми.	0-2	

3. Залік по модулю, контрольна робота	
Критерії оцінювання	20 балів
Встановлено 3 рівні складності завдань.	
1. Перший рівень (завдання 1) – завдання із вибором відповіді – тестові завдання. Завдання з вибором відповіді на теоретичне питання вважається виконаним правильно, якщо в картці тестування записана правильна відповідь.	12*0,5=6
2. Другий рівень (завдання 2) – завдання з короткою відповіддю. Завдання з короткою відповіддю вважається виконаним правильно, якщо студент дав вірні визначення, посилання, тлумачення, короткі коментарі.	4*1,5=6
3. Третій рівень (завдання 3) – завдання із вибором відповіді – тестові завдання практичної направленості. Завдання з вибором відповіді на практичне питання вважається виконаним правильно, якщо в картці тестування записана правильна відповідь.	4*2=8
4. Індивідуальна робота студента (ІНДЗ)	
Критерії оцінювання	14 балів
завдання виконане у зазначений термін, у повному обсязі і без помилок	14
завдання виконане у зазначений термін, у повному обсязі, але є незначні помилки	11-13
завдання виконане у неповному обсязі, або (та) з порушенням терміну виконання, або (та) при наявності значних помилок	8-10
завдання виконане із суттєвими помилками	4-7
завдання не виконане або тільки розпочато процес побудови коду програми.	0-3

Підсумкова оцінка з навчальної дисципліни розраховується з урахуванням балів, отриманих під час поточного контролю за накопичувальною системою. Максимальна кількість балів за результатами поточного контролю становить 100.

Успішність навчання студентів оцінюється за шкалою: зараховано/ незараховано. Для отримання “зараховано” студент повинен під час семестру набрати 51 чи більше балів із 100 можливих. “Зараховано” заслуговує студент, який під час семестру продемонстрував знання навчального матеріалу в обсязі, потрібному для подальшого навчання та майбутньої професійної діяльності. “Незараховано” виставляється у тому випадку, коли він під час семестру набрав менше 51 балу із 100. Оцінка “Незараховано” означає, що студент не достатньо повно вивчив основний теоретичний матеріал, допускає грубі помилки у відповідях, не набув необхідних вмій та знань, передбачених програмою дисципліни

9.3. Шкала оцінювання успішності студентів за результатами підсумкового контролю

Оцінка за шкалою ECTS	Оцінка в балах	Оцінка за національною шкалою
A	90 – 100	Зараховано
B	81-89	
C	71-80	
D	61-70	
E	51-60	
FX	21-50	Не зараховано
F	0-20	Не зараховано (з повторним вивченням)

Студенти, що отримали сумарний бал в межах від 21 до 50 за національною шкалою, отримують оцінку FX за шкалою ECTS та скеровуються на повторне складання заліку.

10. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Навчально-методичне та наукове забезпечення кредитно-модульної системи організації навчального процесу з навчальної дисципліни “Алгоритмізація та програмування” включає:

- стандарти освіти (Освітньо-професійна програма);
- навчальні та робочі навчальні плани;
- силабус навчальної дисципліни;
- програму навчальної дисципліни;
- робочу програму навчальної дисципліни;
- навчально-методичні матеріали для проведення лекцій;
- плани лабораторних робіт та методичні матеріали з їх проведення;
- завдання для самостійної роботи студента і методичні рекомендації щодо їх виконання з навчальної дисципліни;
- завдання для індивідуальної роботи студента (індивідуальні навчально-дослідні завдання) і методичні рекомендації щодо їх виконання з навчальної дисципліни;
- засоби діагностики знань та умінь студентів (збірники тестових завдань);

- завдання для підсумкового модульного контролю;
- методичні, мультимедійні, опорні матеріали для лекційних, лабораторних занять;
- рекомендована література на поточний навчальний рік.

11. МЕТОДИКИ АКТИВІЗАЦІ ПРОЦЕСУ НАВЧАННЯ

Проблемні лекції направлені на розвиток логічного мислення студентів. Коло питань теми обмежується двома-трьома ключовими моментами. При читанні лекцій студентам даються питання для самостійного обмірковування. Студенти здійснюють коментарі самостійно або за участю викладача.

Робота в малих групах дає змогу структурувати лабораторні заняття за формою і змістом, створює можливості для участі кожного студента в роботі за темою заняття, забезпечує формування особистісних якостей та досвіду спілкування.

Мозкові атаки – метод розв’язання невідкладних завдань, сутність якого полягає в тому, щоб висловити якомога більшу кількість ідей за дуже обмежений проміжок часу, обговорити і здійснити їх селекцію

Кейс-метод – розгляд, аналіз конкретних ситуацій, який дає змогу наблизити процес навчання до реальної практичної діяльності.

Презентації – виступи перед аудиторією, що використовуються для представлення певних досягнень, результатів роботи групи, звіту про виконання індивідуальних завдань тощо.

Банки візуального супроводження – сприяють активізації творчого сприйняття змісту дисципліни за допомогою наочності:

- Навчально-методичні матеріали з вивчення навчальної дисципліни.
- Інтерактивні посібники, підручники.
- Лабораторний практикум.

Використання навчальних технологій для активізації процесу навчання з дисципліни

Тема 1. Еволюція методологій програмування	
Проблемні лекції	<u>Проблемні питання:</u> 1. Парадигми програмування. 2. Інкапсуляція. Паралелізм
Кейс-метод	Застосування різних методологій розробки ІТ-продукту.

Тема 2. Об'єкти та класи	
Проблемні лекції	<u>Проблемні питання:</u> 1. Абстрактні класи, інтерфейси та класи-утиліти. 2. Інстанціювання.
Мозкові атаки	Мозкова атака щодо вибору відповідної структури даних для написання програми, яка вирішуватиме реальне завдання із певної предметної області.
Тема 3. Об'єктно-орієнтований аналіз	
Проблемні лекції	<u>Проблемні питання:</u> 1. Аналітичні класи і відносини між ними. 2. Колоборації. 3. Опис колаборацій за допомогою діаграм взаємодії.
Тема 4. Принципи об'єктно-орієнтованого дизайну	
Проблемні лекції	<u>Проблемні питання:</u> 1. Принципи проектування залежностей. 2. Принцип стабільних абстракцій.
Кейс-метод	Застосування принципів ООД при оптимізації програмного коду.
Тема 5. Уніфікована мова моделювання UML	
Проблемні лекції	<u>Проблемні питання:</u> 1. Базові семантичні конструкції мови, їх опис за допомогою спеціальних позначень. 2. Діаграма компонентів. 3. Діаграма станів (або скінченних автоматів).
Презентації	Презентація можливостей програмних пакетів для створення UML-діаграм.
Тема 6. Об'єктно-реляційне відображення	
Проблемні лекції	<u>Проблемні питання:</u> 1. Об'єктно-реляційна БД. 2. Принцип роботи ORM.
Мозкові атаки	Мозкова атака щодо визначення переваг та недоліків використання об'єктно-реляційних БД.
Тема 7. Шаблони проектування	
Проблемні лекції	<u>Проблемні питання:</u> 1. Породжуючі шаблони. 2. Структурні шаблони. 3. Шаблони поведінки.

Тема 8. Метрики

Проблемні лекції	<u>Проблемні питання:</u>
	<ol style="list-style-type: none"> 1. Стабільність залежностей. 2. Стабільність абстракцій.
Кейс-метод	Обчислення різних метрик для оцінки складності та надійності програмного продукту.

12. РЕСУРСИ МЕРЕЖІ ІНТЕРНЕТ

Ресурси мережі Інтернет	Ресурси мережі Факультету з навчальної дисципліни
<ol style="list-style-type: none"> 1. Curated list of UML tools - 2022 edition. URL: https://modeling-languages.com/uml-tools/ 2. Introduction to business modeling using the Unified Modeling Language (UML) [Електронний ресурс] // IBM. - 2003. URL: http://www.ibm.com/developerworks/rational/library/360.html 3. The Unified Modeling Language Reference Manual. URL: https://people.ualgary.ca/~far/Lectures/SENG609-23/PDF/uml_refman.pdf 4. UML Java Programmers. URL: https://www.csd.uoc.gr/~hy252/references/UML_for_Java_Programmers-Book.pdf 5. UML для бізнес-моделювання: для чого потрібні діаграми процесів. URL: https://evergreens.com.ua/ua/articles/uml-diagrams.html 6. Простий посібник зі схем UML і моделювання баз даних. URL: https://www.microsoft.com/uk-ua/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling 7. Розділ 2. Основи UML. URL: https://docs.kde.org/trunk5/uk/umbrello/umbrello/uml-basics.html 8. Стандарт UML 2.2. URL: http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML 	<ul style="list-style-type: none"> – Силабус; – Програма навчальної дисципліни; – Робоча програма навчальної дисципліни; – Підручники – Навчально-методичні матеріали для проведення лекцій. Конспект лекцій з навчальної дисципліни; – Методичні рекомендації з виконання самостійної роботи (СРС) – Методичні рекомендації та ІНДЗ – Засоби діагностики знань студентів з навчальної дисципліни – Практикум для проведення лабораторних занять дисципліни

13. ЗМІНИ І ДОПОВНЕННЯ ДО РОБОЧОЇ ПРОГРАМИ

№ з/п	Зміни і доповнення до робочої програми (розділ, тема, зміст змін і доповнень)	Навчальний рік	Підпис завідувача кафедри
1	2	3	4