



ЗАТВЕРДЖУЮ  
Декан

доц. Андрій СТАСИШИН

“30” січня 2024 р.

## 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 освітньої програми)

освітній ступінь: магістр

(бакалавр/магістр)

форма навчання: денна

(денна, заочна)

ЛЬВІВ 2024

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 р. – 10 с.

**Розробник:** Шевчук І.Б., зав. 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

# 1. ПОЯСНЮВАЛЬНА 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):*

КК1 – 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:

<b>Назва рівня сформованості вміння</b>	<b>Зміст критерію рівня сформованості вміння</b>
<b>1. Репродуктивний</b>	Вміння відтворювати знання, передбачені даною програмою

<b>2. Алгоритмічний</b>	Вміння використовувати знання в практичній діяльності при розв'язуванні типових ситуацій
<b>3. Творчий</b>	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. 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

## 3. 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. Абстрактну 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.

## **4. 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

## **5. INTERNET САЙТИ**

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.ucalgary.ca/~far/Lectures/SENG609-23/PDF/uml\\_refman.pdf](https://people.ucalgary.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](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](http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML)