

Міністерство освіти і науки України  
Національний аерокосмічний університет ім. М. Є. Жуковського  
«Харківський авіаційний інститут»

Кафедра «Інформаційних технологій проектування» (№ 105)

**ЗАТВЕРДЖУЮ**

Гарант освітньої програми

 Свген ДРУЖИНИН  
(підпис) (ініціали та прізвище)

«30» серпня 2024 р.

**СИЛАБУС *ОБОВ'ЯЗКОВОЇ*  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

СУЧАСНІ ТЕХНОЛОГІЇ ТА ІНСТРУМЕНТАРІЙ ПРОГРАМУВАННЯ  
(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»  
(шифр і найменування галузі знань)

Спеціальність: 126 «Інформаційні системи та технології»  
(код і найменування спеціальності)

Освітня програма: «Інформаційні системи та технології підтримки  
віртуальних середовищ»  
(найменування освітньої програми)

**Форма навчання: денна**

**Рівень вищої освіти: другий (магістерський)**

**Вводиться в дію з «01» вересня 2024 р.**

**Харків 2024**

Розробник: доцент каф 105, к.т.н., доцент Олена ЯШИНА  
(посада, науковий ступінь і вчене звання, ім'я та прізвище)

  
(підпис)

Силабус «Сучасні технології та інструментарій програмування» розглянуто  
на засіданні кафедри (№ 105) інформаційних технологій проектування  
(назва кафедри)

Протокол № 1 від « 30 » серпня 2024 р.

В.о. зав. кафедри

к.т.н., доцент

(науковий ступінь і вчене звання)



(підпис)

Аліна АРТЬОМОВА

(ім'я та прізвище)

Погоджено з представником здобувачів освіти:



(підпис)

Арсеній ПИМАЧКОВІЙ

(ім'я та прізвище)

## 1. Загальна інформація про викладача



Яшина Олена Сергіївна, доцент кафедри інформаційних технологій проектування, кандидат технічних наук, доцент.

---

Перелік дисциплін, які викладає: Структури даних, Проектно-орієнтоване управління створенням комп'ютерних систем, Управління ІТ-проектами

---

Напрями наукових досліджень: проектування інформаційних систем, методи та технології обробки даних, розробка баз даних та програмного забезпечення

---

## 1. Опис навчальної дисципліни

**Форма навчання** – денна.

**Семестр, в якому викладається дисципліна** – 1, 2.

**Дисципліна** обов'язкова.

**Загальна кількість годин за навчальним планом** - 195 годин / 6,5 кредитів ЄКТС. 96 годин аудиторної та 99 годин самостійної роботи здобувачів (1 семестр: 135 годин / 4,5 кредитів ЄКТС, в т.ч. 64 години аудиторної та 71 самостійної роботи; 2 семестр (курсове проектування): 60 годин/2 кредити ЄКТС, в т.ч. 32 години аудиторної та 28 самостійної роботи).

**Види занять** – лекції, лабораторні, самостійна робота, курсовий проект.

**Вид контролю** – поточний, модульний та підсумковий (семестровий) контроль іспит та диференційований залік.

**Мова викладання** – українська.

## 2. Мета та завдання навчальної дисципліни

**Мета:** надання слухачам знань, умінь, навичок, методичних прийомів та засобів, що необхідні створення сучасного програмного забезпечення інформаційних систем.

**Завдання:** вивчити загальні підходи, методи, інструментальні засоби та технології створення програмного забезпечення інформаційних систем на платформі .NET, забезпечувати доступ до даних та ефективну обробку даних.

### **Компетентності, які набуваються:**

#### **Загальні компетентності (ЗК):**

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК03. Здатність спілкуватися з представниками інших професійних груп різного рівня (з експертами з інших галузей знань/видів економічної діяльності).

ЗК04. Здатність розробляти проекти та управляти ними.

ЗК05. Здатність оцінювати та забезпечувати якість виконуваних робіт.

#### **Спеціальні (фахові) компетентності (СК):**

СК01. Здатність розробляти та застосувати ІСТ, необхідні для розв'язання стратегічних і поточних задач.

СК02. Здатність формулювати вимоги до етапів життєвого циклу сервіс-орієнтованих інформаційних систем.

СК05. Здатність використовувати сучасні технології аналізу даних для оптимізації процесів в інформаційних системах.

СК06. Здатність управляти інформаційними ризиками на основі концепції інформаційної безпеки.

СК07. Розробляти і реалізовувати інноваційні проекти у сфері ІСТ.

### **Очікувані результати навчання:**

РН06. Обґрунтовувати вибір технічних та програмних рішень з урахуванням їх взаємодії та потенційного впливу на вирішення організаційних проблем, організувати їх впровадження та використання.

РН10. Забезпечувати якісний кіберзахист ІСТ, планувати, організувати, впроваджувати та контролювати функціонування систем захисту інформації.

РН11. Розв'язувати задачі цифрової трансформації у нових або невідомих середовищах на основі спеціалізованих концептуальних знань, що включають сучасні наукові здобутки у сфері інформаційних технологій, досліджень та інтеграції знань з різних галузей.

**Пререквізити:** Сучасні технології та інструментарій програмування.

**Кореквізити:** Основи ІТ консалтингу.

**Постреквізити:** дипломне проектування.

### **3. Зміст навчальної дисципліни**

#### **Модуль 1.**

#### **Змістовний модуль 1. Основи проектування програмного забезпечення інформаційних систем**

##### ***Тема 1. Сутність, зміст і завдання технологій та інструментальних засобів створення програмного забезпечення.***

*Загальна кількість годин за темою:* 6 годин.

*Анотація:*. Предмет вивчення і задачі дисципліни. Зв'язок з іншими дисциплінами спеціальності. Сучасні методологічні підходи та технології створення програмного забезпечення (ПЗ)

*Тема лекції:* Вступ до дисципліни.

*Тема лабораторного заняття:* -

*Обсяг самостійної роботи здобувача:* 4 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* методи створення програмного забезпечення.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання

##### ***Тема 2. Стили та методи проектування архітектури ПЗ.***

*Загальна кількість годин за темою:* 10 годин.

*Анотація:* Системний підхід до проектування ПЗ. Критерії декомпозиції систем. Стили проектування ПЗ. Стили проектування «згори донизу» та «знизу догори». Повторне використання коду та архітектурних рішень. Архітектурне та деталізоване проектування. Структурний, об'єктно-орієнтований, компонентний та сервісний підходи до створення ПЗ.

*Тема лекції:* Системний підхід до створення ПЗ.

*Тема лабораторного заняття:* -

*Обсяг самостійної роботи здобувача:* 4 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Задачі та особливості структурного, об'єктно-орієнтованого, компонентного та сервісного підходів до створення ПЗ.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання

##### ***Тема 3. Методи та підходи до створення програмного забезпечення.***

*Загальна кількість годин за темою:* 12 годин.

*Анотація:* Підходи та методи деталізованого проектування ПЗ. Створення ПЗ згідно з методологією Agile. Підходи Test Driven Development, Behavior Driven Development, Domain Driven Development та інші. Безперервна інтеграція та попереджувальне тестування. Інструментальні засоби автоматизації тестування.

*Тема лекції:* Підходи та методи деталізованого проектування ПЗ.

*Тема лабораторного заняття:* - Інструментальні засоби платформи .NET.

*Обсяг самостійної роботи здобувача:* 6 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Встановлення інструментальних засобів. Основні можливості MS Visual Studio, MS SQL Server, MS Management Studio.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

##### ***Тема 4. Принципи створення об'єктно-орієнтованого програмного забезпечення.***

*Загальна кількість годин за темою:* 12 годин.

*Анотація:* Принцип єдиної відповідальності (Single Responsibility Principle, SRP), принцип відкритості/закритості (Open/Closed Principle, OCP), принцип заміщення Лісков

(Liskov Substitution Principle, LSP), принцип розділення інтерфейсу (Interface Segregation Principle, ISP), принцип інверсії залежності (Dependency Inversion Principle, DIP). Додаткові принципи: You Aren't Gonna Need It (YAGNI), Don't Repeat Yourself (DRY), Keep it short and simple (KISS).

*Тема лекції:* Принципи SOLID.

*Тема лабораторного заняття:* -

*Обсяг самостійної роботи здобувача:* 6 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Особливості застосування SOLID. Приклади та контрприклад програмного коду. Основні помилки при застосуванні принципів SOLID та способи їх виправлення.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання

### ***Тема 5. Багаторівнева архітектура інформаційних систем.***

*Загальна кількість годин за темою:* 12 годин.

*Анотація:*. Багаторівнева архітектура ІС. Рівні (шари) та ланки системи. Відкриті та закриті шари. Типовий розподіл функцій між рівнями. Доступ до даних у багаторівневій архітектурі.

*Тема лекції:* Багаторівнева архітектура ІС.

*Тема лабораторного заняття:* - Доступ до даних у багаторівневій архітектурі.

*Обсяг самостійної роботи здобувача:* 6 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Налаштування доступу до даних. Контекст даних та сутнісні класи. Виконання запитів до БД.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

### ***Тема 6. Технології доступу до даних.***

*Загальна кількість годин за темою:* 12 годин.

*Анотація:* Технології доступу до даних. Задачі доступу до даних. Огляд існуючих технологій. Особливості доступу до даних у WEB-розробках. Технології об'єктно-реляційного перетворення (Object-Relational Mapping, ORM). Технологія ADO.NET Entity Framework Core. Побудова сутнісної моделі Entity Data Model. Проміжні рівні взаємодії прикладного програмного забезпечення та бази даних. Засоби об'єктно-реляційного зіставлення даних. Клас DBNull. Перетворення звичайних та Nullable типів даних.

*Тема лекції:* Технології доступу до даних .NET.

*Тема лабораторного заняття:* - Технологія ADO.NET Entity Framework Core.

*Обсяг самостійної роботи здобувача:* 6 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Завдання ORM технологій. Побудова Entity Data Model. Зіставлення типів даних C# та SQL.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

## **Модуль 2. Технології та інструментальні засоби проектування інформаційних систем**

### ***Тема 7. Доступ до даних за допомогою мови LINQ.***

*Загальна кількість годин за темою:* 8 годин.

*Анотація:* Мова Language Integrated Query (LINQ). Особливості LINQ to Objects та LINQ to SQL. Структура запиту LINQ. Інтерфейс IEnumerable<>. Виконання запитів. Синтаксис виразів та синтаксис методів. Проекції на іменовані та анонімні класи. Фільтрація та сортування. Методи розширення. Лямбда-вирази.

*Тема лекції:* Основи мови LINQ.

*Тема лабораторного заняття:* - LINQ для доступу до даних.

*Обсяг самостійної роботи здобувача:* 5 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Запити LINQ to SQL. Запити із підзапитами. Використання методів розширення. Написання делегатів та лямбда-функцій.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..5 балів.

### ***Тема 8. Поглиблені можливості LINQ.***

*Загальна кількість годин за темою:* 10 годин.

*Анотація:* Складні запити LINQ. Запити до кількох наборів даних. Навігаційні властивості. З'єднання, групування та агрегування даних. Дерева виразів.

*Тема лекції:* Складні запити LINQ.

*Тема лабораторного заняття:* - Багатотабличні запити LINQ.

*Обсяг самостійної роботи здобувача:* 5 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Використання навігаційних властивостей. Створення запитів з із групуванням та агрегуванням даних.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..5 балів.

### ***Тема9. Програмування на боці серверу БД.***

*Загальна кількість годин за темою:* 9 годин.

*Анотація:* Особливості діалекту Transact-SQL. Програмування на боці сервера БД. Створення збережених процедур та користувацьких функцій. Системні збережені процедури і функції. Тригери БД. Використання SQL у Entity Framework Core. Виконання запитів, що повертають набори даних, скалярних запитів та запитів на модифікацію даних. Виклик збережених процедур та користувацьких функцій із коду C#.

*Тема лекції:* Запити Transact-SQL.

*Тема лабораторного заняття:* - Використання Transact-SQL у Entity Framework Core.

*Обсяг самостійної роботи здобувача:* 5 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:*

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

### ***Тема 10. Моделювання даних в Entity Framework Core.***

*Загальна кількість годин за темою:* 10 годин.

*Анотація:* Технології Code First та Database First. Контекст даних та сутнісні класи. Створення моделі існуючої бази даних, застосування скафолдінгу. Створення нової моделі даних за допомогою Code First. Налаштування бази даних. Анотації даних. Програмний інтерфейс Fluent API.

*Тема лекції:* Моделювання даних за підходом Code First.

*Тема лабораторного заняття:* - Створення моделі бази даних за підходом Code First.

*Обсяг самостійної роботи здобувача:* 5 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Створення та налаштування контексту даних. Створення сутнісних класів. Застосування методів Fluent API.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

### ***Тема 11. Супроводження та модифікація БД.***

*Загальна кількість годин за темою:* 10 годин.

*Анотація:* Міграції. Створення схеми БД за допомогою міграцій. Управління міграціями. Застосування та видалення міграцій. Отримання скрипта міграції. Модифікація схеми БД за допомогою міграцій: переіменування сутностей, додавання та видалення стовпців. Можливі проблеми та їх розв'язання. Застосування SQL.

*Тема лекції:* Міграції в Entity Framework Core.

*Тема лабораторного заняття:* - Створення БД за допомогою міграцій.

*Обсяг самостійної роботи здобувача:* 6 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Створення та застосування міграцій. Застосування Fluent API в міграціях.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання. Виконання та захист лабораторної роботи – 0..10 балів.

### **Тема 12. Управління транзакціями.**

*Загальна кількість годин за темою:* 8 годин.

*Анотація:*. ACID властивості транзакцій: атомарність (Atomicity), узгодженість (Consistency), ізолюваність (Isolation), довговічність (Durability). Блокування. Тупикові блокування (Dead Locks), способи їх уникнення. Рівні ізоляції транзакцій. Управління транзакціями в Transact-SQL. Журнал транзакцій. Фіксація і відкат транзакцій. Управління транзакціями в Entity Framework Core. Розподілені транзакції в мікросервісній архітектурі.

*Тема лекції:* Управління транзакціями.

*Тема лабораторного заняття:* - Транзакції в Entity Framework Core.

*Обсяг самостійної роботи здобувача:* 5 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Управління транзакціями. Створення, фіксація та відкат транзакцій. Точки збереження. Конфлікти. Управління рівнями ізоляції.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання

### **Тема 13. Архітектурний паттерн Model-View-Controller.**

*Загальна кількість годин за темою:* 8 годин.

*Анотація:*.

*Тема лекції:* Паттерн Model-View-Controller (MVC). Розвиток та інтерпретація. Компоненти: модель (Model), подання (View), контролер (Controller). Активна та пасивна модель. Різновиди: Model-View-ViewModel, Model-View-Presenter. Антіпаттерн FSUC (Fat Stupid Ugly Controllers). MVC у .NET. Платформа ASP.NET Core MVC.

*Тема лабораторного заняття:* -

*Обсяг самостійної роботи здобувача:* 4 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Функції компонентів паттерну MVC. Можливості ASP.NET Core MVC.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання

### **Тема 14. Мікросервісна архітектура.**

*Загальна кількість годин за темою:* 8 годин.

*Анотація:* Сервіси. Розвиток сервіс-орієнтованої архітектури (Service-Oriented Architecture, SOA). Мікросервіси. Взаємодія мікросервісів через REST API. Розподілені БД та розподілені транзакції. Паттерн SAGA. Аутентифікація та авторизація в мікросервісних додатках. Протокол OAuth2.

*Тема лекції:* Особливості мікросервісної архітектури.

*Тема лабораторного заняття:* -

*Обсяг самостійної роботи здобувача:* 4 години.

*Теми, види робіт, що належать до самостійної роботи здобувачів:* Відмінності між монолітною та мікросервісною архітектурою. Переваги та недоліки мікросервісів. Реалізація розподілених транзакцій.

*Види контролю, критерії оцінювання:* присутність на заняттях та відповідь на питання.



## 4. Індивідуальне завдання (курсний проект)

Мета виконання – набуття практичних навичок самостійного розроблення сучасних програмних засобів, призначених для вирішення задач, що виникають під час комп'ютеризації інженерної діяльності. У процесі виконання курсового проекту студенти також закріплюють ті практичні знання та навички, набуті при вивченні дисципліни.

Індивідуальні теми курсових робіт надаються студентам у окремому електронному документі і студенти мають можливість самостійно обрати тему проекту із запропонованих або запропонувати свою тему, що відповідає вимогам до проекту.

Склад пояснювальної записки повинен бути таким:

1. Титульний аркуш
2. Зміст
3. Вступ
4. Аналіз предметної області. Формулювання функціональних вимог.
5. Побудова структури БД для підтримки класів з використанням технології Code First та міграцій Entity Framework Core.
6. Проектування та реалізація користувацького інтерфейсу.
7. Програмна реалізація функціональних вимог, реалізація доступу до даних та управління даними.
8. Тестування.
9. Висновок
10. Перелік посилань

Обсяг записки не обмежується і, в середньому, сягає 20-30 сторінок.

Термін виконання курсового проекту – тижні 1 – 14. Трудомісткість – 32 годин аудиторних занять, 28 година самостійної роботи. Типовий план-графік виконання курсового проекту такий:

| № | Етап   | Термін<br>(тиждень<br>семестру) | Вид звіту          | Процент<br>готовності |
|---|--|---------------------------------|--------------------|-----------------------|
| 1 | Аналіз предметної області,<br>визначення вимог | 2                               | Розділ записки     | 15                    |
| 2 | Проектування структури<br>класів               | 4                               | Розділ записки     | 25                    |
| 3 | Створення БД                                   | 6                               | Розділ записки     | 40                    |
| 4 | Проектування<br>користувацького інтерфейсу     | 8                               | Розділ записки     | 55                    |
| 5 | Реалізація функціональних<br>вимог             | 12                              | Розділ записки     | 70                    |
| 6 | Тестування програмного<br>забезпечення         | 14                              | Розділ записки     | 80                    |
| 7 | Оформлення записки                             | 16                              | Записка            | 90                    |
| 8 | Захист курсового проекту                       | 17                              | Записка і програма | 100                   |

Варіанти курсових проектів надані на сервері кафедри. Для магістрів та студентів, залучених до виконання науково-дослідних робіт, можливе виконання курсових проектів дослідницького спрямування, узгоджених з індивідуальними планами підготовки.

## 5. Методи навчання

Лекції проводяться з використанням основних розділів конспекту лекцій в електронній формі, елементів мультимедійної підтримки курсу (відео фрагментів), демонстрацій окремих прийомів роботи з інструментальним середовищем та/або наочних

матеріалів у вигляді схем та діаграм. Лабораторні роботи виконуються з використанням загальнодоступних безплатних (Community Edition) або ліцензованих робочих версій сучасного програмного забезпечення. Самостійна робота включає підготовку до лабораторних робіт, модульного контролю та іспиту, виконання поза-аудиторної частини індивідуального завдання і вивчення вказаних вище тем за конспектом, літературними джерелами та програмною документацією.

## 6. Методи контролю

Контроль здійснюється згідно з «Положенням про рейтингове оцінювання досягнень студентів». Поточний контроль – відповідно до повноти, якості та своєчасності виконання лабораторних робіт, індивідуального завдання; проміжний (модульний) контроль – письмові контрольні роботи на 8-му та 16-му тижнях; підсумковий контроль – письмовий іспит.

Поточний контроль здійснюється під час лабораторних занять, у процесі розгляду й оцінювання виконаних завдань, спілкування викладача зі студентом на індивідуальних консультаціях. Він має на меті перевірку рівня підготовленості студента до виконання конкретної роботи. Засоби проведення поточного контролю включають тести для виявлення ступеня оволодіння студентами необхідними теоретичними положеннями.

Модульний контроль проводиться у формі контрольних робіт за окремими частинами (модулями) дисципліни. Для проміжного контролю дисципліна поділена на дві частини (модулі): 1 модуль – теми 1-6; 2 – теми 7-14.

Підсумковий контроль проводиться у формі письмового іспиту. Студенти допускаються до іспиту, якщо він виконав усі обов'язкові роботи і завдання з дисципліни протягом семестру. Перелік таких робіт і завдань доводиться до студентів на початку семестру. Своєчасність і якість їх виконання з'ясовуються в процесі поточного контролю.

Якщо студент одержав незадовільну оцінку за контрольне оцінювання з певної теми курсу, або був відсутній на контрольному оцінюванні, то він зобов'язаний ліквідувати поточну заборгованість, що утворилася, під час найближчої консультації у викладача, який проводить лабораторні заняття або лекції з даної дисципліни.

Студент, який має заборгованості з контрольних оцінювань з окремих тем курсу «Сучасні технології та інструментарій програмування», вважається таким, що не виконав всі види робіт, передбачені навчальним планом на семестр з цієї навчальної дисципліни, і до іспиту не допускається.

## 7. Критерії оцінювання та розподіл балів, які отримують здобувачі

### 7.1. Розподіл балів, які отримують здобувачі (кількісні критерії оцінювання)

| Складові навчальної роботи            | Бали за одне заняття (завдання) | Кількість занять (завдань) | Сумарна кількість балів |
|---------------------------------------|---------------------------------|----------------------------|-------------------------|
| <i>Змістовний модуль 1</i>            |                                 |                            |                         |
| Виконання і захист лабораторних робіт | 0...10                          | 3                          | 0...30                  |
| Модульний контроль                    | 0...15                          | 1                          | 0...15                  |
| <i>Змістовний модуль 2</i>            |                                 |                            |                         |
| Виконання і захист лабораторних робіт | 0...10                          | 4                          | 0...40                  |
| Модульний контроль                    | 0...15                          | 1                          | 0...15                  |
| <i>За семестр 1</i>                   |                                 |                            | <b>0...100</b>          |
| Курсовий проект                       |                                 |                            | 0...100                 |
| <i>За семестр 2</i>                   |                                 |                            | <b>0...100</b>          |

Семестровий контроль (іспит) проводиться за наявності допуску до іспиту. При складанні семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту складається з двох блоків теоретичних питань у вигляді тестів за кожним модулем та однієї практичної задачі. Тест складається з 30 питань закритого типу (максимальна кількість балів за відповідь на одне питання – 2 або 3 бала залежно від складності) За теоретичні питання сумарна кількість балів 60, за практичну задачу – 40 балів, загалом 100 балів.

## 7.2. Якісні критерії оцінювання

### Необхідний обсяг знань для одержання позитивної оцінки:

- 1 Критерії декомпозиції інформаційної системи.
- 2 Стили проектування «згори донизу» та «знизу догори».
- 3 Технології повторного використання при створенні програмного забезпечення.
- 4 Архітектурне та деталізоване проектування ПЗ.
- 5 Структурний підхід до створення ПЗ.
- 6 Об'єктно-орієнтований підхід до створення ПЗ.
- 7 Компонентний підхід до створення ПЗ.
- 8 Сервісний підхід до створення ПЗ.
- 9 Test Driven Development.
- 10 Behavior Driven Development.
- 11 Безперервна інтеграція та попереджувальне тестування.
- 12 Принцип єдиної відповідальності (Single Responsibility Principle, SRP)
- 13 Принцип відкритості/закритості (Open/Closed Principle, OCP).
- 14 Принцип заміщення Лісков (Liskov Substitution Principle, LSP).
- 15 Принцип розділення інтерфейсу (Interface Segregation Principle, ISP).
- 16 Принцип інверсії залежності (Dependency Inversion Principle, DIP).
- 17 Багаторівнева архітектура ІС.
- 18 Типовий розподіл функцій між рівнями.
- 19 Технології доступу до даних.
- 20 Особливості доступу до даних у WEB-розробках.
- 21 Технології об'єктно-реляційного перетворення (Object-Relational Mapping, ORM).
- 22 Технологія ADO.NET Entity Framework Core.
- 23 Побудова сутнісної моделі Entity Data Model.
- 24 Проміжні рівні взаємодії прикладного програмного забезпечення та бази даних.
- 25 Засоби об'єктно-реляційного зіставлення даних.
- 26 Мова Language Integrated Query (LINQ).
- 27 Особливості LINQ to Objects та LINQ to SQL.
- 28 Структура запиту LINQ.
- 29 Інтерфейс IEnumerable<>.
- 30 Виконання запитів LINQ.
- 31 Синтаксис виразів та синтаксис методів LINQ.
- 32 Проекції на іменовані та анонімні класи в LINQ.
- 33 Фільтрація та сортування в LINQ.
- 34 Методи розширення в LINQ.
- 35 Лямбда-вирази.
- 36 Навігаційні властивості в LINQ.
- 37 З'єднання, групування та агрегування даних в LINQ.
- 38 Дерева виразів LINQ.
- 39 Створення збережених процедур та користувацьких функцій Transact-SQL.
- 40 Системні збережені процедури і функції Transact-SQL. Тригери БД.
- 41 Використання SQL у Entity Framework Core.
- 42 : Технології Code First та Database First.
- 43 Контекст даних та сутнісні класи.

- 44 Створення моделі існуючої бази даних, застосування скафолдингу.
- 45 Створення нової моделі даних за допомогою Code First.
- 46 Анотації даних в Code First.
- 47 Програмний інтерфейс Fluent API.
- 48 Міграції Entity Framework Core.
- 49 Створення схеми БД за допомогою міграцій Entity Framework Core.
- 50 Управління міграціями Entity Framework Core. Застосування та видалення міграцій.
- 51 Скрипти міграцій Entity Framework Core.
- 52 Паттерн Model-View-Controller (MVC).
- 53 Активна та пасивна модель MVC.
- 54 Model-View-ViewModel та Model-View-Presenter.
- 55 Антипаттерн FSUC (Fat Stupid Ugly Controllers).
- 56 MVC у .NET. Платформа ASP.NET Core MVC.
- 57 Мікросервісна архітектура.
- 58 Взаємодія мікросервісів через REST API.
- 59 Розподілені БД та розподілені транзакції. Паттерн SAGA.
- 60 Аутентифікація та авторизація в мікросервісних додатках. Протокол OAuth2.

### **Необхідний обсяг умінь для одержання позитивної оцінки:**

- 1 Встановлювати та налагоджувати необхідне програмне забезпечення, включаючи інсталяцію Nuget пакетів.
- 2 Налагоджувати контекст бази даних.
- 3 Конструювати та виконувати запити LINQ.
- 4 Конструювати багатотабличні запити LINQ з використанням навігаційних властивостей.
- 5 Модифікувати та зберігати дані засобами Entity Framework Core.
- 6 Використовувати методи розширення LINQ.
- 7 Використовувати делегати і лямбда-вирази.
- 8 Створювати сутнісну модель існуючої бази даних за допомогою технології Scaffolding.
- 9 Створювати модель нової бази даних за допомогою технології Code First.
- 10 Створювати та застосовувати міграції Entity Framework Core.

### **7.3 Критерії оцінювання роботи здобувача протягом семестру**

**Задовільно (60-74).** Показати мінімум знань та умінь. Виконати й захистити всі лабораторні завдання та здати тестування. Знати основні підходи до створення програмного забезпечення та основні архітектурні моделі.

**Добре (75-89).** Твердо знати мінімум, захистити всі лабораторні завдання та здати тестування. Продемонструвати в цілому високий рівень теоретичної підготовки згідно навчального плану, вміння застосовувати отриманні знання в процесі вирішення поставленої задачі тобто розуміти, як виконується об'єктно-реляційні перетворення, вміти налагоджувати контекст для з'єднання з БД, писати запити різної складності, отримувати, обробляти та відображати дані.

**Відмінно (90-100).** Безпомилково виконати й захистити всі лабораторні завдання, розрахункову роботу, здати модулі з максимальною кількістю балів. Досконально знати всі основні та додаткові теми та вміння застосувати теоретичний матеріал щодо вирішення практичної ситуації шляхом застосування отриманні знання в процесі розв'язання поставленої проблеми, тобто моделювати складні структури даних, створювати та застосовувати міграції, розв'язувати конфлікти міграцій, забезпечувати належну якість та надійність програмного забезпечення.

## Шкала оцінювання: бальна і традиційна

| Сума балів | Оцінка за традиційною шкалою  |               |
|------------|-------------------------------|---------------|
|            | Іспит, диференційований залік | Залік         |
| 90 – 100   | Відмінно                      | Зараховано    |
| 75 – 89    | Добре                         |               |
| 60 – 74    | Задовільно                    |               |
| 0 – 59     | Незадовільно                  | Не зараховано |

### 8. Політика навчального курсу

Відпрацювання пропущених занять відбувається відповідно до розкладу консультацій, за попереднім погодженням з викладачем. Питання, що стосуються академічної доброчесності, розглядає викладач або за процедурою, визначеною у Положенні про академічну доброчесність:

- самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей);
- посилення на джерела інформації у разі використання ідей, розробок, тверджень, відомостей;
- дотримання норм законодавства про авторське право і суміжні права;
- надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності, використанні методики досліджень і джерела інформації.

### 9. Методичне забезпечення

Навчальні посібники, приклади програмного коду, основні теоретичні положення та завдання для самостійного контролю у електронній формі та допоміжні матеріали надаються студентам на сервері кафедри.

### 10. Рекомендована література

#### Базова

1. Проектування багаторівневої архітектури інформаційних управляючих систем / О.Є. Федорович, О.С. Яшина, Л.М. Лутай. – Х.: Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харк. авіац. ін-т», 2012. – 132 с.
2. Проектування інформаційних систем [Електронний ресурс] : навч. посіб. до лаб. практикуму / О. С. Яшина, О. В. Коновалова, К. О. Западня. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2021. – 79 с. Режим доступу: [https://library.khai.edu/catalog?clear\\_all\\_params=0&mode=DocBibRecord&lang=ukr&caller\\_mode=DocBibRecord&print\\_basket=&ext=no&theme\\_path=0&docid=510544492](https://library.khai.edu/catalog?clear_all_params=0&mode=DocBibRecord&lang=ukr&caller_mode=DocBibRecord&print_basket=&ext=no&theme_path=0&docid=510544492)
3. Проектування інформаційних систем : навч. посіб. : гриф МОН України / Н. Б. Шаховська, В. В. Литвин ; М-во освіти і науки України ; за наук. ред. В. В. Пасічника. - Львів. - Магнолія-2006, 2011. - 380 с.
4. 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. [Електронний ресурс]: Режим доступу: <http://eprints.cdu.edu.ua/1481/1/pro.pdf>
5. Проектування інформаційних систем: Загальні питання теорії проектування ІС (конспект лекцій) [Електронний ресурс]: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. – Електронні текстові дані (1 файл: 2,02 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 192с. Режим доступу: [https://ela.kpi.ua/bitstream/123456789/33651/1/PIS\\_KL.pdf](https://ela.kpi.ua/bitstream/123456789/33651/1/PIS_KL.pdf)

## **Допоміжна**

1. Моделі автоматизованого управління ресурсами в машинобудуванні: навч. посібник / Є.А. Дружинін, О.С. Яшина. – Х.: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2008. – 46 с.
2. Introducing Microsoft SQL Server 2019. – Packt Publishing, 2019 – 489 p. ISBN: 978-1-83882-621-5

## **15. Інформаційні ресурси**

1. Портал «Документація Microsoft». <https://docs.microsoft.com/uk-ua/documentation/>
2. IEEE Computer Society. <https://www.computer.org/education/bodies-of-knowledge/software-engineering>
3. <https://dou.ua/lenta/tags/ООП/>
4. <https://stackoverflow.com/questions/tagged/architecture>
5. <https://stackoverflow.com/questions/tagged/design-patterns>