

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

Кафедра інженерії програмного забезпечення (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми

 І.Б. Шевченко
(підпис) (ініціали та прізвище)

« 31 » 08 2021 р.

**РОБОЧА ПРОГРАМА ОБОВ'ЯЗКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Архітектура та проектування програмного забезпечення .Net
(назва навчальної дисципліни)

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

Спеціальність: 121 «Інженерія програмного забезпечення»
(код та найменування спеціальності)

Освітня програма: «Інженерія програмного забезпечення»
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: початковий рівень (короткий цикл) вищої освіти

Харків 2021 рік

Розробник: Лучшев П.О., доц. кафедри №603, канд. техн. наук
(прізвище та ініціали, посада, науковий ступінь та вчене звання)



(підпис)

Робочу програму розглянуто на засіданні кафедри інженерії програмного забезпечення

(назва кафедри)

Протокол № 2 від 31.08.2021 р.

Завідувач кафедри д-р техн. наук., проф.
(науковий ступінь і вчене звання)



(підпис)

І.Б. Туркін
(ініціали та прізвище)

1. Опис навчальної дисципліни

Найменування показника	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – 5	<p>Галузь знань <u>12 «Інформаційні технології»</u> <small>(шифр і найменування)</small></p> <p>Спеціальність <u>121 «Інженерія програмного забезпечення»</u> <small>(код і найменування)</small></p> <p>Освітня програма <u>«Інженерія програмного забезпечення»</u> <small>(найменування)</small></p> <p>Рівень вищої освіти: початковий рівень (короткий цикл)</p>	Обов'язкова
Кількість модулів – 2		Навчальний рік
Кількість змістовних модулів – 4		2022/2023
Індивідуальне завдання «Патерни проектування» <small>(назва)</small>		Семестр
Загальна кількість годин – 94/150		<u>4</u> -й
Кількість тижневих годин для денної форми навчання: аудиторних – 3,5 самостійної роботи студента – 5,9		Лекції*
		<u>24</u> години
	Практичні, семінарські*	
	_____ годин	
	Лабораторні*	
	<u>32</u> годин	
	Самостійна робота	
	<u>94</u> годин	
	Вид контролю	
	модульний контроль, іспит	

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: 56/94.

*Аудиторне навантаження може бути зменшене або збільшене на одну годину залежно від розкладу занять.

2. Мета та завдання навчальної дисципліни

Мета вивчення: ознайомлення та засвоєння студентами існуючих архітектур програмного забезпечення, принципами проектування розподілених систем з використанням проміжного/сполучного програмного забезпечення та шаблонів проектування.

Завдання: в результаті навчання студенти матимуть практичні навички з розроблення складних програмних проектів з використанням однієї або декількох добре відомих архітектур та компонентного підходу у середовищі Visual Studio.

Компетентності, які набуваються:

Загальні компетентності

- ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.
- ЗК2. Здатність застосовувати знання у практичних ситуаціях.
- ЗК5. Здатність вчитися і оволодівати сучасними знаннями.
- ЗК6. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Фахові компетентності

- ФК1. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.
- ФК2. Здатність розробляти модулі та компоненти програмних систем.
- ФК6. Здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.
- ФК7. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.
- ФК9. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
- ФК11. Здатність до алгоритмічного та логічного мислення.

Програмні результати навчання:

- ПРН2. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.
- ПРН3. Знати і застосовувати основні професійні стандарти та інші нормативно-правові документи в галузі інженерії програмного забезпечення.
- ПРН4. Знати і застосовувати відповідні математичні поняття, методи об'єктно-орієнтованого аналізу для розробки програмного забезпечення.
- ПРН6. Вміти розробляти людино-машинний інтерфейс.
- ПРН9. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН10. Застосовувати на практиці інструментальні програмні засоби проектування, тестування, візуалізації та документування програмного забезпечення.

ПРН12. Вміти застосовувати методи компонентної розробки програмного забезпечення.

Пререквізити – «Основи програмної інженерії», «Основи програмування», «Програмування мовою C#».

Кореквізити – «Бази даних».

3. Програма навчальної дисципліни

Змістовний модуль №1 Технології розробки ПЗ

ТЕМА 1. Об'єкт, предмет, метод і значення дисципліни. Ефективні способи вирішення задач проектування програмного забезпечення.

ТЕМА 2. Розробка та застосування DLL. Створення та налагодження програм декількома засобами розробки.

ТЕМА 3. Компоненти. Розробка архітектури складного програмного забезпечення.

ТЕМА 4. Створення, завершення і безпосереднє управління потоками. Налagodження багатопоточних програм.

ТЕМА 5. Об'єкти та методи синхронізації. Проблеми спільного доступу до даних.

ТЕМА 6. Використання об'єктів синхронізації: Event, Critical section, Semaphore, Mutex.

Змістовний модуль №2 Структура та архітектура ПЗ

ТЕМА 7. Поняття шаблону проектування. Типи шаблонів GOF.

ТЕМА 8. Породжуючі шаблони: Абстрактна фабрика (Abstract Factory), Будівник (Builder), Одинак (Singleton), Прототип (Prototype), Фабричний метод (Factory Method)

ТЕМА 9. Структурні шаблони: Адаптер (Adapter), Декоратор (Decorator), Замісник (Proxy), Компонувальник (Composite), Міст (Bridge), Легковаговик (Flyweight), Фасад (Facade)

ТЕМА 10. Поведінкові шаблони: Відвідувач (Visitor), Інтерпретатор (Interpreter), Ітератор (Iterator), Команда (Command), Ланцюг обов'язків (Chain of Responsibility), Посередник (Mediator), Спостерігач (Observer), Стан (State), Стратегія (Strategy), Знімок (Memento), Шаблонний метод (Template Method).

ТЕМА 11. Шаблони багатозадачності: Обмін повідомленнями (Messaging design pattern), Блокування з подвійною перевіркою (Double-checked locking), Монітор (Monitor object), Планувальник (Scheduler), Однопоточні виконання (Single Thread Execution).

Модульний контроль.**Змістовний модуль №3 СОМ-технології**

ТЕМА 12. Основи поняття СОМ-технології. Інтерфейси. Створення і використання ідентифікаторів об'єктів. Засоби реєстрації та обліку об'єктів.

ТЕМА 13. Створення та використання СОМ-об'єктів.

ТЕМА 14. Застосування СОМ-об'єктів на платформі .Net.

Змістовний модуль №4 Аналіз якості та оцінка програмного дизайну

ТЕМА 15. Методи та засоби вимірювання внутрішніх параметрів і складності програмного забезпечення

ТЕМА 16. Попередня оцінка складності проекту

ТЕМА 17. Метрики об'єктно-орієнтованого проектування програмного забезпечення

ТЕМА 18. Еволюція та реінжиніринг програмного забезпечення

Модульний контроль.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин				
	денна форма				
	усього	у тому числі			
л		п	лаб	с.р.	
Змістовний модуль №1					
1. Об'єкт, предмет, метод і значення дисципліни	2	2			
2. Розробка та застосування DLL	5	1		2	2
3. Компоненти. Архітектура складного ПЗ	13	2		6	5
4. Багатопоточні програми	9	2		2	5
5. Засоби синхронізації	9	2		2	5
6. Використання об'єктів синхронізації	7	2			5
Разом за змістовим модулем 1	45	11		12	22
Змістовний модуль №2					
7. Шаблони проектування	9	1		2	6
8. Породжуючі шаблони	9	1		2	6
9. Структурні шаблони	9	1		2	6
10. Поведінкові шаблони	9	1		2	6
11. Шаблони багатозадачності	9	1		2	6
Разом за змістовим модулем 2	45	5		10	30
<i>Модульний контроль</i>	4				
Змістовний модуль №3					
12. Основи СОМ-технології	9	1		2	6
13. Створення та використання СОМ-об'єктів	9	1		2	6
14. Застосування СОМ-об'єктів на платформі .Net	10	2		2	6
Разом за змістовим модулем 3	28	4		6	18
Змістовний модуль №4					
15. Методи та засоби вимірювання внутрішніх параметрів ПЗ	8	1		1	6
16. Оцінка складності проекту	8	1		1	6
17. Метрики ООП	8	1		1	6
18. Еволюція та реінжиніринг програмного забезпечення	8	1		1	6
Разом за змістовим модулем 4	32	4		4	24
<i>Модульний контроль</i>	4				
<i>Контрольний захід</i>	8				
Усього годин	150	24		32	94

5. Теми семінарських занять

№ п/п	Назва теми	Кількість годин
1		
	Разом	

6. Теми практичних занять

№ п/п	Назва теми	Кількість годин
1		
	Разом	

7. Теми лабораторних занять

№ п/п	Назва теми	Кількість годин
1.	Розробка компонентів	4
2.	Розробка компонентів UI. Прив'язка даних	4
3.	Прив'язка та цілісність даних	4
4.	Серіалізація і десеріалізація даних	4
5.	Використання COM-технології. Експорт даних в документи MS Office	6
6.	Створення розподілених додатків	6
7.	Аналіз якості та продуктивності	4
	Разом	32

8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Об'єкт, предмет, метод і значення дисципліни	4
2	Шаблони проектування	4
3	Породжуючі шаблони	6
4	Структурні шаблони	6
5	Поведінкові шаблони	6
6	Шаблони багатозадачності	6
7	Розробка та застосування DLL	4
8	Архітектура складного ПЗ	4
9	Багатопоточні програми	4
10	Засоби синхронізації	4
11	Використання об'єктів синхронізації	4
12	Основи COM-технології	6
13	Створення та використання COM-об'єктів	8
14	Створення та використання компонентів .Net	6
15	Методи та засоби вимірювання параметрів ПЗ	4
16	Оцінка складності проекту	6
17	Метрики ООП	6
18	Еволюція та реінжиніринг програмного забезпечення	6
	Разом	94

9. Індивідуальне завдання

Виконання розрахунково-графічної роботи на тему «Патерни проектування».

10. Методи навчання

За джерелами придбання знань – словесні: лекція (вступна, традиційна, проблемна, з помилками), бесіда (евристична), диспут, дискусія, робота з друкованими та інтернет-джерелами; наочні: ілюстрація, спостереження; практичні: вправа, лабораторна робота.

За характером пізнавальної діяльності тих, хто навчається – інформаційно-репродуктивний, репродуктивний, проблемне викладання, частково-пошуковий.

За логікою пізнання – індуктивний, дедуктивний, аналогій, вивідних знань.

Методи перевірки й оцінки знань, умінь, навичок: спостереження, усне опитування, контрольні роботи, програмований контроль, тестування (традиційне та машинне).

11. Методи контролю

1. Опитування.
2. Тестування.
3. Лабораторні роботи.
4. Модульні контрольні роботи.

Форма підсумкового контролю успішності навчання: іспит (тестування).

12. Критерії оцінювання та розподіл балів, які отримують студенти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1,2			
Робота на лекціях	0...1	8	0...8
Виконання і захист лабораторних (практичних) робіт	1...5	3	3...15
Модульний контроль	0...20	1	0...20
Змістовний модуль 3,4			
Робота на лекціях	0...1	4	0...4
Виконання і захист лабораторних (практичних) робіт	1...5	4	3...20
Модульний контроль	0...20	1	0...20
Виконання і захист РГР (РР, РК)			1...13
Усього за семестр			60...100

Елементи самостійної роботи студента в складі практичних завдань можуть бути оцінені додатковими балами на розсуд викладача в рамках максимальної сумарної кількості балів.

Семестровий контроль проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Критерії оцінювання роботи студента протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Уявляти принципи та застосовувати компонентне-орієнтований підхід у проектуванні ПЗ; базові принципи СОМ-технології; засоби вимірювання і використання метрик в проектуванні ПЗ; особливості створення складних програмних систем.

Добре (75-89). Твердо знати мінімум, здати тестування та поза аудиторну самостійну роботу. Досконало розуміти принципи застосування СОМ-технології, проектувати і реалізовувати програмне забезпечення, використовуючи технології проміжного програмного забезпечення; використовувати на практиці принципи об'єктно-орієнтованого програмування.

Відмінно (90-100). Здати всі роботи та контрольні точки з оцінкою «відмінно». Виставляється якщо при відповіді на питання виявлено всебічні, систематизовані, глибокі знання матеріалу, який виноситься на контроль, уміння вільно виконувати завдання, передбачені програмою. Студент вміє аналізувати, цілеспрямовано шукати і вибирати необхідні шаблони проектування для вирішення практичних задач. Досконально знати всі теми та уміти застосовувати їх на практиці.

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

13. Методичне забезпечення

- Архітектура та проектування програмного забезпечення [Текст]: навч. посібник з виконання лаб. робіт / П.О. Лучшев. –Харків: Нац. аерокосм. ун-т ім. М.Є. Жуковського «Харків. авіація. ін-т », 2021. - 90 с.
- Дистанційний курс дисципліни розроблено у системі дистанційного навчання Mentor, яку впроваджено в Національному аерокосмічному університеті ім. М.Є. Жуковського «ХАІ», доступ до курсу за посиланням: <https://mentor.khai.edu/enrol/index.php?id=5814>

14. Рекомендована література

Базова література

1. Табунщик Г. В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с. http://eir.zntu.edu.ua/bitstream/123456789/1824/1/Tabunshchik_Software_Design.pdf
2. Designing Applications on the .NET Platform. Application Architecture Guide 2.0 patterns & practices / J.D. Meier, Alex Homer, David Hill, Jason Taylor, Prashant Vansode, Lonnie Wall, Rob Boucher Jr, Akshay Bogawat – Microsoft press, – 381 p.
3. Опорний конспект лекцій. Архітектура та проектування програмного забезпечення.
4. «Дизайн-патерни — просто, як двері» by Andriy Buday is licensed under a Creative Commons Attribution-NonCommercial 3.0 Unported License <http://designpatterns.andriybuday.com>. 2012. – 90 с.

Допоміжна

1. Мартін Роберт Чиста архітектура: мистецтво розробки програмного забезпечення / Роберт Мартін – К.: Фабула, 2019. – 368 с. <https://fabulabook.com/product/chysta-arhitektura/>
2. Design Patterns: Elements of Reusable Object-Oriented Software. Erich Gamma, Richard Helm, Ralph Johnson and John Vliss. Addison-Wesley Professional, 1994.
3. Мартін Роберт Чистий код: створення, аналіз, рефакторинг / Роберт Мартін – К.: Фабула, 2019. – 416 с. <https://fabulabook.com/product/chystyj-kod/>
4. E. Gamma, R. Helm, R. Johnson, J. Willsides. Tricks of object-oriented design. Design Patterns. Peter 2007, - 366 pp.
5. Martin Fowler. Templates for corporate applications. Williams, 2009, - 544 pp.

15. Інформаційні ресурси

1. ASP.NET MVC: The Official Microsoft ASP.NET Site <http://www.asp.net/mvc>
2. <http://www.aivosto.com/project/help/pm-oo-mood.html> MOOD & MOOD2 metrics.
3. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.48.2017&rep=rep1&type=ps> Fernando Brito e Abreu. Toward the Design Quality Evaluation of Object-Oriented / Software Systems. Fernando Brito e Abreu, Miguel Goulão, Rita Esteves /