

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. Туркін
(ініціали та прізвище)

31.08.2021 р.

**СИЛАБУС ОBOB'ЯЗKОВОЇ
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 і найменування спеціальності)

Освітня програма: Інженерія програмного забезпечення
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2021 року

Харків – 2021 р.

Розробник: Лучшев П.О., доц. каф. 603, к.т.н
(прізвище та ініціали, посада, науковий ступінь і вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри _____
інженерії програмного забезпечення (№ 603)
(назва кафедри)

Протокол № 2 від «31» серпня 2021 р.

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


(підпис)

І.Б. Туркін
(ініціали та прізвище)

Погоджено з представником здобувачів освіти:

Представник студентського
самоврядування


(підпис)

Р.В. Кадру
(ініціали та прізвище)

1. Загальна інформація про викладача



Лучшев Павло Олександрович, к.т.н., доцент. Викладає в університеті з 1997 року. Розробник дисциплін:

- Архітектура та проектування програмного забезпечення .Net;*
- Комп'ютерна графіка з OpenGL;*
- Теорія віртуальної реальності;*
- Програмування віртуальної реальності;*
- Патерни проектування програмного забезпечення систем критичного призначення.*

Напрями наукових досліджень: інженерія програмного забезпечення.

2. Опис навчальної дисципліни

Семестр, в якому викладається дисципліна – 4 семестр.

Обсяг дисципліни: 4.5 кредитів ЄКТС (135 годин), у тому числі аудиторних – 48 год., самостійної роботи здобувачів – 87 год.

Форма здобуття освіти: денна, дистанційна.

Дисципліна: обов'язкова.

Види навчальної діяльності: – лекції, лабораторні роботи, самостійна робота здобувача.

Види контролю: модульний та підсумковий контроль (іспит).

Мова викладання: українська

Необхідні обов'язкові попередні дисципліни (пререквізити): «Основи програмування», «Програмування мовою C#» «Алгоритми та структури даних», «Людино-машинна взаємодія», «Об'єктно-орієнтоване програмування»

Супутні дисципліни (кореквізити): «Інструментальні засоби підтримки колективної розробки», «Основи організації спільної роботи в ІТ», «Бази даних».

3. Мета та завдання навчальної дисципліни

Мета: 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.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК05. Здатність вчитися і оволодівати сучасними знаннями.

ЗК06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

Фахові компетентності.

ФК02. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

ФК03. Здатність розробляти архітектури, модулі та компоненти програмних систем.

ФК07. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

ФК10. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

ФК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

ФК14. Здатність до алгоритмічного та логічного мислення.

Очікувані результати навчання.

ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

ПРН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПРН08. Вміти розробляти людино-машинний інтерфейс.

ПРН09. Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.

ПРН10. Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.

ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.

ПРН26. Вміти конфігурувати компоненти, організувати робочий процес розгортання та тестувати рішення в його остаточному операційному середовищі.

4. Зміст 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. Налагодження 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.

Назви змістових модулів і тем	Кількість годин				
	денна форма				
	усього	у тому числі			
		л	п	лаб	с.р.
<i>Змістовний модуль №1</i>					
1. Об'єкт, предмет, метод і значення дисципліни	2	2			
2. Розробка та застосування DLL	4	1		2	1
3. Компоненти. Архітектура складного ПЗ	7	2		2	3
4. Багатопоточні програми	7	2		2	3
5. Засоби синхронізації	7	2		2	3
6. Використання об'єктів синхронізації	7	2		2	3
<i>Разом за змістовим модулем 1</i>	34	11		10	13
<i>Змістовний модуль №2</i>					
7. Шаблони проектування	6	1			5
8. Породжуючі шаблони	7	1		1	5
9. Структурні шаблони	7	1		1	5
10. Поведінкові шаблони	7	1		1	5
11. Шаблони багатозадачності	8	1		1	6
<i>Разом за змістовим модулем 2</i>	35	5		4	26
<i>Модульний контроль</i>	5				
<i>Змістовний модуль №3</i>					
12. Основи СОМ-технології	8	1		2	5
13. Створення та використання СОМ-об'єктів	8	1		2	5
14. Застосування СОМ-об'єктів на платформі .Net	8	2		2	4
<i>Разом за змістовим модулем 3</i>	24	4		6	14
<i>Змістовний модуль №4</i>					
15. Методи та засоби вимірювання внутрішніх параметрів ПЗ	5	1		1	3
16. Оцінка складності проекту	5	1		1	4
17. Метрики ООП	5	1		1	4
18. Еволюція та реінжиніринг програмного забезпечення	5	1		1	3
<i>Разом за змістовим модулем 4</i>	20	4		4	14
<i>Модульний контроль</i>	5				
<i>Контрольний захід</i>	10				
Усього годин	135	24		24	67

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.	СОМ-технології. Експорт даних в документи MS Office	4
6.	Аналіз якості та продуктивності	4
	Разом	24

Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Об'єкт, предмет, метод і значення дисципліни	1
2	Шаблони проектування	4
3	Породжуючі шаблони	5
4	Структурні шаблони	5
5	Поведінкові шаблони	5
6	Шаблони багатозадачності	5
7	Розробка та застосування DLL	4
8	Архітектура складного ПЗ	4
9	Багатопоточні програми	6
10	Засоби синхронізації	4
11	Використання об'єктів синхронізації	4
12	Основи СОМ-технології	6
13	Створення та використання СОМ-об'єктів	6
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	87

5. Індивідуальні завдання

Виконання розрахунково-графічної роботи за темою «Патерни проектування».

Обсяг самостійної роботи здобувачів – 20 годин.

6. Методи навчання

Словесні, наочні, практичні.

7. Методи контролю

Поточний контроль (теоретичне опитування й виконання лабораторних робіт), модульний контроль (тестування за розділами курсу) та підсумковий (семестровий) контроль (іспит у виді загального тесту за усіма розділами курсу).

8. Критерії оцінювання та розподіл балів, які отримують здобувачі

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 робіт	1...5	3	3...15
Модульний контроль	0...20	1	0...20
Змістовний модуль 3,4			
Робота на лекціях	0...1	8	0...8
Виконання і захист лабораторних робіт	1...5	3	3...15
Модульний контроль	0...20	1	0...20
Виконання і захист РГР (РР, РК)			1...20
Усього за семестр			60...100

Додатково може бути оцінено як дострокове виконання кожної з лабораторних робіт (до 2 балів), так і елементи самостійної роботи в ній (до 3 балів)). Сумарно основна та додаткова оцінка за усі види робіт не може перевищувати 100 балів.

Семестровий контроль (іспит) проводиться у вигляді комп'ютерного тестування (питання відкритого типу) у разі відмови студента від балів поточного тестування та за наявності допуску до іспиту (виконання усіх лабораторних робіт та РГР). Під час складання семестрового іспиту студент має можливість отримати максимум 60 балів, які замінюють результати поточного модульного контролю.

Критерії оцінювання роботи здобувача протягом семестру

Задовільно (60-74). Здати всі лабораторні роботи (зі значною корекцією тексту лабораторної роботи викладачем), захистити розрахункову роботу (зі значною корекцією тексту розрахункової роботи викладачем), мати необхідних мінімум знань за всіма темами та мінімум вмій щодо застосування отриманих знань.

Добре (75-89). Здати всі лабораторні роботи (з мінімальною корекцією тексту лабораторної роботи викладачем), захистити розрахункову роботу (з мінімальною корекцією тексту розрахункової роботи викладачем), знати всі теми та уміти застосовувати їх.

Відмінно (90-100). Здати всі лабораторні роботи (без корекції тексту лабораторної роботи викладачем), захистити розрахункову роботу (без корекції тексту розрахункової роботи викладачем), досконально знати всі теми та уміти застосовувати їх.

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

9. 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.

10. Методичне забезпечення

Навчальні посібники, навчально-методичні посібники, конспекти лекцій, методичні рекомендації з проведення лабораторних робіт тощо, які видані в Університеті знаходяться за посиланням: <https://library.khai.edu/catalog>

Для забезпечення дистанційний доступу до курсу дисципліни застосовуються сервіси <https://mentor.khai.edu/> або <https://classroom.google.com/> за додатковою вказівкою викладача ідентифікатору курсу. Для реєстрації студентів у системах дистанційного доступу використовуються поштові адреси виключно у домені *student.khai.edu*.

11. 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 Bansode, 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 /