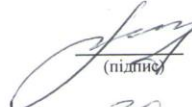


Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. С. Жуковського
«Харківський авіаційний інститут»

Кафедра інженерії програмного забезпечення (№ 603)

ЗАТВЕРДЖУЮ

Керівник проектної групи/


(підпис) І.Б. Туркін
(ініціали та прізвище)
«30» 08 2019 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: перший (бакалаврський)

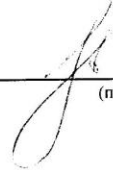
Харків 2019 рік

Робоча програма «Архітектура та проектування програмного забезпечення .Net» для студентів за спеціальністю: 121 «Інженерія програмного забезпечення» освітньою програмою «Інженерія програмного забезпечення»

«20» 04 2019 р, – 10 с.

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)

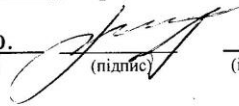
Протокол № 1 від «20» 05 2019 р.

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> (шифр і найменування)</p> <p>Спеціальність <u>121 «Інженерія програмного забезпечення»</u> (код і найменування)</p> <p>Освітня програма <u>«Інженерія програмного забезпечення»</u> (найменування)</p> <p>Рівень вищої освіти: перший (бакалаврський)</p>	Цикл професійної підготовки (2.1. Дисципліни загально-професійної підготовки)
Кількість модулів – 2		Навчальний рік
Кількість змістовних модулів – 4		2019/2020
Індивідуальне завдання «Патерни проектування» (назва)		Семестр
Загальна кількість годин – 48/150		_6_ -й
Кількість тижневих годин для денної форми навчання: аудиторних – 3 самостійної роботи студента – 6,375		Лекції*
		<u>24</u> години
		Практичні, семінарські*
	_____ годин	
	Лабораторні*	
	<u>24</u> годин	
Самостійна робота		
<u>102</u> годин		
Вид контролю		
модульний контроль, іспит		

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: **48/102.**

*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.

Результати навчання: студент має:

знати: компонентне-орієнтований підхід до проектування ПЗ; базові принципи СОМ-технології; засоби вимірювання і використання метрик в проектуванні ПЗ; особливості створення складних програмних систем в операційних системах Microsoft Windows за допомогою сучасних середовищ розробки програм (Visual Studio та ін.).

вміти: застосовувати різноманітні шаблони проектування, середовища розробки та архітектури в проектуванні різноманітного програмного забезпечення; проектувати і реалізовувати програмне забезпечення, використовуючи декілька різних технологій проміжного програмного забезпечення; використовувати адекватні метрики якості як засіб оцінки якості проектування, оцінювати відповідність результатів проектування поставленим цілям; модифіковані проекти, використовуючи продумані підходи до управління змінами; використовувати методи зворотної інженерії (reverse engineering) для відновлення дизайну програмного забезпечення; використовувати на практиці принципи об'єктно-орієнтованого програмування.

мати уявлення: з використання сучасних засобів розробки програмного забезпечення для створення та проектування компонентів архітектури програмного продукту; практичне застосування шаблони проектування.

Міждисциплінарні зв'язки: дисципліні передують курси: «Алгоритми та структури даних», «Людино-машинна взаємодія», «Об'єктно-орієнтоване програмування», «Інструментальні засоби підтримки колективної розробки», «Конструювання ПЗ»

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. 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.

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	4	1		2	1
3. Компоненти. Архітектура складного ПЗ	8	2		2	4
4. Багатопоточні програми	8	2		2	4
5. Засоби синхронізації	8	2		2	4
6. Використання об'єктів синхронізації	8	2		2	4
Разом за змістовим модулем 1	38	11		10	17
Змістовний модуль №2					
7. Шаблони проектування	8	1			7
8. Породжуючі шаблони	8	1		1	6
9. Структурні шаблони	8	1		1	6
10. Поведінкові шаблони	10	1		1	8
11. Шаблони багатозадачності	10	1		1	8
Разом за змістовим модулем 2	44	5		4	35
<i>Модульний контроль</i>	5				
Змістовний модуль №3					
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
Разом за змістовим модулем 3	24	4		6	14
Змістовний модуль №4					
15. Методи та засоби вимірювання внутрішніх параметрів ПЗ	6	1		1	4
16. Оцінка складності проекту	6	1		1	4
17. Метрики ООП	6	1		1	4
18. Еволюція та реінжиніринг програмного забезпечення	6	1		1	4
Разом за змістовим модулем 4	24	4		4	16
<i>Модульний контроль</i>	5				5
<i>Контрольний захід</i>	10				10
Усього годин	150	24		24	102

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

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 СОМ-технології	6
13	Створення та використання СОМ-об'єктів	8
14	Створення та використання компонентів .Net	8
15	Методи та засоби вимірювання параметрів ПЗ	4
16	Оцінка складності проекту	8
17	Метрики ООП	8
18	Еволюція та реінжиніринг програмного забезпечення	8
	Разом	102

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

Опитування на лекціях. Виконання і захист лабораторних робіт. Модульні контрольні роботи.

Форма підсумкового контролю успішності навчання: іспит (письмово) у 6 семестрі

12. Критерії оцінювання та розподіл балів, які отримують студенти

12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

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	8	0...8
Виконання і захист лабораторних (практичних) 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

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

Семестровий контроль (іспит) 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.

Білет для іспиту/заліку складається з двох теоретичних питань (кожне питання 25 балів) та двох практичних питань (кожне питання 25 балів, або одного практично питання 50 балів).

12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки: принципи розробки складного ПЗ, компонентне-орієнтований підхід у проектуванні ПЗ; базові принципи СОМ-технології; засоби вимірювання і використання метрик в проектуванні ПЗ; особливості створення складних програмних систем в операційних системах Microsoft Windows за допомогою сучасних середовищ розробки програм (Visual Studio та ін.).

Необхідний обсяг вмінь для одержання позитивної оцінки: створювати програми з використанням різноманітних шаблонів проектування та архітектури; проектувати і реалізовувати програмне забезпечення, використовуючи декілька різних технологій проміжного програмного забезпечення; використовувати адекватні метрики якості як засіб оцінки якості проектування, оцінювати відповідність результатів проектування поставленим цілям;

12.3 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». 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

1. Розроблений лекційний курс та комплекс презентацій Power Point ([//master/студенти/2019-2020/6 факультет/2курс/SoftwareArchitecture/Матеріали курсу](#)).
2. Розроблені питання для модульних контрольних робіт ([//master/студенти/2019-2020/6 факультет/2курс/ SoftwareArchitecture /Модулі](#)).
3. Розроблені питання для підсумкового контролю успішності навчання ([//master/студенти/2019-2020/6 факультет/2курс/ SoftwareArchitecture /Питання](#)).
4. Лабораторні роботи ([//master/студенти/2019-2020/6 факультет/2курс/ SoftwareArchitecture /Лабораторні роботи](#)).
5. Індивідуальні розрахункові роботи (домашні завдання) ([//master/студенти/2019-2020/6 факультет/2курс/ SoftwareArchitecture /РГР](#)).
6. Дібрані матеріали для самостійної роботи студентів ([//master/студенти/2019-2020/6 факультет/2курс/ SoftwareArchitecture /Самостійна робота](#)).

14. Рекомендована література

Базова література

1. Орлов С.А. Технологии разработки программного обеспечения. – СПб.: Питер, 2002. – 464 с.
2. Соммервилл И. Инженерия программного обеспечения, 6-е издание.: Пер. с англ. – М.: Издательский дом Вильямс, 2002. – 624 с.
3. Роджерсон Дейл. Основы COM. – Microsoft Press, "Русская редакция", 2000. – 375 с.

Допоміжна

1. Елманова Н. Delphi и технология COM / Н. Елманова, С. Трепалин, А. Тенцер. – СПб.: Питер, 2003. – 698 с.
2. Рихтер Дж. Создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows. – СПб.: Питер, 2008. – 714 с.

15. Інформаційні ресурси

1. <http://rstdn.ru/summary/864.xml> Паттерны проектирования.
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 //
4. <http://rstdn.ru/summary/254.xml> ATL
5. <http://rstdn.ru/summary/247.xml> COM/DCOM/COM+
6. <http://rstdn.ru/summary/569.xml> .Net
7. <http://rstdn.ru/summary/3387.xml> Проектирование ПО.