

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

(повне найменування вищого навчального закладу)

Кафедра Програмних засобів

“ЗАТВЕРДЖУЮ”

Ректор (перший проректор)

_____ В.Г. Прушківський
“ _____ ” _____ 2017 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

3.14 ПРОЕКТНИЙ ПРАКТИКУМ

рівень вищої освіти перший (бакалаврський)

спеціальність 121 «Інженерія програмного забезпечення»

(код і назва спеціальності)

інститут, факультет Інститут інформатики та радіoeлектроніки, факультет
комп'ютерних наук і технологій

(назва інституту, факультету)

Робоча програма «Проектний практикум» для студентів бакалаврів
(назва навчальної дисципліни)
 за спеціальністю 121 «Інженерія програмного забезпечення». „___” _____,
 2017 року – 8 с.

Розробники: (вказати авторів, їхні посади, наукові ступені та вчені звання)
 Федорончак Тетяна Василівна, доц. каф. ПЗ, к.т.н., доц.

Робоча програма затверджена на засіданні кафедри
програмних засобів

Протокол від. “28” серпня 2017 року № 1

Завідувач кафедри програмних засобів

_____ (Субботін С.О.)
(підпис) (прізвище та ініціали)
 “___” _____ 2017 року

Схвалено методичною комісією вищого навчального закладу за напрямом підготовки
 (спеціальністю) 121 «Інженерія програмного забезпечення»
(код, назва)

Протокол від. “___” _____ 2017 року № ___

“___” _____ 2017 року Голова _____ (М.М. Касьян)
(підпис) (прізвище та ініціали)

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 3	Галузь знань 12 Інформаційні технології (шифр і назва)	Нормативна	
	Спеціальність 121 «Інженерія програмного забезпечення» (шифр і назва)		
Модулів – 2	Спеціальність (освітня програма, спеціалізація): _____	Рік підготовки:	
Змістових модулів – 3		3-й	
Індивідуальне науково-дослідне завдання _____ (назва)		Семестр	
Загальна кількість годин – 90		6-й	
Тижневих годин для денної форми навчання: аудиторних – 3 самостійної роботи студента – 4	Освітній ступінь: бакалавр	Лекції	
		12 год.	
		Практичні, семінарські	
		--	
		Лабораторні	
		26 год.	
		Самостійна робота	
		52 год.	
		Індивідуальні завдання: --	
		Вид контролю: залік	

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:

для денної форми навчання – 0,75

для заочної форми навчання -

2. Мета та завдання навчальної дисципліни

Мета – надання майбутньому фахівцю чіткого розуміння про методи і засоби створення програмного проекту на всіх етапах життєвого циклу та їх взаємозв'язок, можливості практично засвоїти їх в проектуванні програмного забезпечення.

Завдання дисципліни полягає в тому, що студенти повинні опанувати основні принципи проектування класів в об'єктно-орієнтованому програмуванні та шаблони проектування.

У результаті вивчення навчальної дисципліни студент повинен

знати:

– теоретичні засади використання шаблонів проектування при розробці архітектури програмного продукту;

– принципи проектування SOLID;

– загальні шаблони розподілу обов'язків GRASP;

– шаблони проектування GOF;

– переваги та недоліки використання шаблонів проектування;

– анти-шаблони в програмуванні;

вміти:

– застосовувати та створювати компоненти багаторазового використання;

– проектувати архітектуру системи з використанням шаблонів проектування.

3. Програма навчальної дисципліни

Змістовий модуль 1. Шаблони SOLID та GRASP

Тема 1. Основні принципи проектування класів SOLID.

Принципи єдиної відповідальності, відкритості/закритості, підстановки Барбери Лісков, розділення інтерфейсів, інверсії залежностей.

Тема 2. Загальні шаблони розподілу обов'язків GRASP.

Основні шаблони: інформаційний експерт, творець, контролер, низька зв'язність, високе зачеплення. Додаткові шаблони: чиста вигадка, посередник, поліморфізм, приховування реалізації.

Змістовий модуль 2. Шаблони проектування GOF

Тема 1. Основи шаблонів і їх структури.

Огляд шаблонів проектування. Структура шаблону з чотирьох частин: імені, формулювання завдання, описів рішення і результатів. Переваги використання шаблонів проектування при проектуванні структури ПЗ.

Тема 2. Класифікація шаблонів. Використання шаблонів в проектуванні структури ПЗ.

Класифікація шаблонів проектування: Структурні шаблони, шаблони поведінки та породжуючі шаблони. Сумісне використання шаблонів.

Змістовий модуль 2. Анті-шаблони.

Тема 1. Анті-шаблони в управлінні розробкою програмного забезпечення. Анті-шаблони в об'єктно-орієнтованому програмуванні. Методологічні анті-шаблони.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовий модуль 1. Шаблони SOLID та GRASP												
Тема 1. Основні принципи проектування класів SOLID	12	2		-		10	12	1				11
Тема 2. Загальні шаблони розподілу обов'язків GRASP	12	2		-		10	12	1				11
Разом за змістовим модулем 1	24	4		-		20	24	2				22
Змістовий модуль 4. Шаблони проектування GOF												
Тема 1. Основи шаблонів і їх структури	4	2	-	-	-	2	4	1				3
Тема 2. Класифікація шаблонів. Використання шаблонів в проектуванні структури ПЗ	50	4	-	26	-	20	50	1		6		43
Разом за змістовим модулем 2	54	6	-	26	-	22	54	2		6		46
Змістовий модуль 3. Анті-шаблони												
Тема 1. Анті-шаблони в управлінні розробкою ПЗ. Анті-шаблони в ООП. Методологічні анті-шаблони.	12	2	-	-	-	10	12					12
Разом за змістовим модулем 3	12	2	-	-		10	12					12
Разом	90	12	-	26	-	52	90	4		6		80

5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
	Не передбачено	

6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
	Не передбачено	

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Твірні шаблони проектування	5
2	Структурні шаблони проектування	5
3	Шаблони поведінки. Частина 1	5
4	Шаблони поведінки. Частина 2	5
5	Практичне використання шаблонів проектування	6
	Разом	26

8. Самостійна робота

Самостійна робота студента:

(денна форма навчання) повторення матеріалу, засвоєного на лекціях, самостійне опанування частини теоретичного матеріалу, робота з контрольними запитаннями та завданнями, підготовка до лабораторних робіт, підготовка до заліку – 52 год.

(заочна форма навчання) повторення матеріалу, засвоєного на лекціях, самостійне опанування частини теоретичного матеріалу, робота з контрольними запитаннями та завданнями, підготовка до лабораторних робіт, виконання контрольної роботи, підготовка до заліку – 80 год.

9. Індивідуальні завдання

Студенти заочної форми навчання виконують контрольну роботу (12 год.).

10. Методи навчання

Організаційні форми навчання: лекційні заняття, лабораторні заняття, самостійна робота студентів.

Основні методи активного навчання: розповідь, бесіда, дискусія, діалог, індивідуальний/груповий проект, презентація.

11. Методи контролю

Усний контроль у вигляді індивідуального та фронтального опитування.

Письмовий контроль у вигляді самостійних письмових робіт, поточного тестування.

12. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота						Сума
ЗМ 1		ЗМ 2		ЗМ 3	Тест	
T1	T2	T1	T2	T1		
5	5	5	40	5	40	100

Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
85-89	B	добре	
75-84	C		
70-74	D	задовільно	
60-69	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	F	незадовільно з обов’язковим повторним вивченням дисципліни	не зараховано з обов’язковим повторним вивченням дисципліни

13. Методичне забезпечення

1. Конспект лекцій з дисципліни.

2. Методичні вказівки до лабораторних робіт з дисципліни “Проектний практикум” для студентів напряму підготовки 6.050103 “Програмна інженерія” всіх форм навчання / Т.В. Юр, Н.О. Миронова. – Запоріжжя: ЗНТУ, 2013. –52с.

14. Рекомендована література

Базова

1. Guide to the Software Engineering Body of Knowledge (SWEBOK) [Електронний ресурс]. – 2004. – Режим доступу: <http://www.computer.org/portal/web/swebok/html/contents>, <http://swebok.sorlik.ru>.

2. Software Engineering 2004. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering (SE 2004) [Электронный ресурс]. – 2004. – Режим доступа: <http://sites.computer.org/ccse/>

3. Гамма Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. – СПб.: Питер, 2016. – 366 с.

4. Мартин Р. К., Быстрая разработка программ. Принципы, примеры, практика / Р. К. Мартин, Дж. В. Ньюкирк, Р. С. Косс. – М.: Вильямс, 2004. – 752 с.

5. Ларман К. Применение UML и шаблонов проектирования / К. Ларман. – 2-е изд. – М.: Вильямс, 2002. – 624 с.

Допоміжна

6. Дубина О. Обзор паттернов проектирования [Электронный ресурс] / О. Дубина. – 2005. – Режим доступа: <http://citforum.ru/SE/project/pattern/#1>

7. Зандстра Мэтт. РНР: объекты, шаблоны и методики программирования / Мэтт Зандстра. – 2-е изд.. – М.: Вильямс, 2009. – 480 с. – <http://www.williamspublishing.com/PDF/978-5-8459-1586-3/part.pdf>

8. Фаулер Мартин. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / Мартин Фаулер. – 3-е издание. – СПб: Символ-Плюс, 2011. – 192 с.

9. Соммервилл Иан. Инженерия программного обеспечения / Иан Соммервилл. – 6-е изд. – М.: Вильямс, 2002. – 624 с.

10. Брауде Э. Технология разработки программного обеспечения / Эрик Дж. Брауде. – СПб: Питер, 2004. – 655 с.

11. Марк Гранд Шаблоны проектирования в JAVA. Каталог популярных шаблонов проектирования, проиллюстрированных при помощи UML. – М.: «Новое знание», 2004. – 560 с.

15. Інформаційні ресурси