

Міністерство освіти і науки України  
Львівський національний університет імені Івана Франка  
Кафедра радіофізики та комп'ютерних технологій

**“ЗАТВЕРДЖУЮ”**

Завідувач кафедри радіофізики та  
комп'ютерних технологій

\_\_\_\_\_ Болеста І.М.  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2019 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**  
**ТЕХНОЛОГІЇ СТВОРЕННЯ ПРОГРАМНИХ ПРОДУКТІВ**

підготовки бакалаврів  
галузь знань 12 Інформаційні технології  
спеціальність 122 Комп'ютерні науки  
факультет електроніки та комп'ютерних технологій

2019-2020 навчальний рік

Робоча програма навчальної дисципліни “**Технології створення програмних продуктів**”  
для студентів за спеціальністю 122 Комп’ютерні науки

Розробники:

**Кушнір О.О.**, канд. фіз.-мат. наук, доцент кафедри радіофізики та комп’ютерних технологій  
факультету електроніки та комп’ютерних технологій

Робочу програму схвалено на засіданні кафедри радіофізики та комп’ютерних технологій

“   28   ”    серпня    2019 року, протокол №   19  

Завідувач кафедри  
радіофізики та комп’ютерних технологій

Болеста І.М.

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни
		<i>денна форма навчання</i>
Кількість кредитів – <b>5</b>	Галузь знань <b>12</b> <b>Інформаційні технології</b>	Нормативна
Модулів – <b>2</b>	Спеціальність <b>122 – Комп’ютерні науки</b>	<i>Рік підготовки:</i> <b>3-й</b>
Змістових модулів – <b>4</b>		<i>Семестр</i> <b>6-й</b>
Індивідуальне науково-дослідне завдання –		
Загальна кількість годин – <b>150</b>		
Тижневих годин для денної форми навчання: аудиторних – <b>4</b> самостійної роботи студента – <b>2,7</b>	Освітньо-кваліфікаційний рівень: <b>бакалавр</b>	<i>Лекції</i> <b>32 год.</b>
		<i>Практичні, семінарські</i> -----
		<i>Лабораторні</i> <b>32 год.</b>
		<i>Самостійна робота</i> <b>86 год.</b>
		<i>Вид контролю:</i> залік

## **2. Мета та завдання навчальної дисципліни**

Курс «**Технології створення програмних продуктів**» є нормативною дисципліною циклу професійної та практичної підготовки.

**Мета:** полягає у ознайомленні студентів із сукупністю виробничих процесів створення програмних засобів, які можуть ефективно використовуватися в процесах розробки програмного забезпечення, зокрема при аналізі вимог, моделюванні та проектуванні програмних систем.

**Завдання:** полягають у формуванні у бакалаврів знань щодо основних процесів бізнес розробки програмних продуктів належної якості.

В результаті вивчення даного курсу студент повинен:

**знати** принципи складання вимог до програмних продуктів, моделі життєвого циклу розробки програмних продуктів, концепції проектування програмних продуктів, принципи конструювання, техніки досягнення гарантії якості та принципи керування різними етапами створення програмних продуктів.

**вміти:** вибирати стратегії для планування життєвого циклу системи, визначати організаційну, економічну, технічну та операційну здійсненність проекту, проектувати компоненти програмного забезпечення, проектувати інтерфейс інформаційних систем, реалізовувати прототипи архітектури програмного забезпечення, реалізовувати та тестувати компоненти програмного забезпечення, інтегрувати компоненти в систему.

Для вивчення дисципліни необхідні знання з таких курсів, як «Алгоритмізація та проектування», «Об'єктно-орієнтовне програмування», «Кросплатформне проектування», «Теорія алгоритмів».

У подальшому, знання отримані студентами з курсу «Технології створення програмних продуктів» будуть використовуватись при вивченні наступних дисциплін: «Управління ІТ проектами», «Інновації у підприємстві».

## **3. Програма навчальної дисципліни**

### **МОДУЛЬ 1**

#### **Змістовий модуль 1. Вимоги та проектування програмних продуктів.**

##### **Тема 1. Поняття програмного продукту, основна термінологія.**

Вступ і загальний огляд дисципліни. Основна термінологія курсу. Етапи створення програмного продукту.

##### **Тема 2. Вимоги до програмних продуктів.**

Поняття вимог. Напрямки розробки вимог. Класифікація вимог. Інженерія, специфікація та трасування вимог. Вимоги до інтерфейсу користувача.

##### **Тема 3. Засоби контролю версій.**

Історія виникнення засобів контролю версій. Принципи роботи та сучасні варіації. Використання Git та його модифікацій.

##### **Тема 4. Життєві цикли програмних продуктів.**

Поняття життєвого циклу програмного продукту. Процеси життєвого циклу. Моделі життєвого циклу програмних продуктів: каскадна модель, інкрементна модель, ітераційна модель, еволюційна модель, V-подібна модель.

##### **Тема 5. Стандарти розробки програмних продуктів.**

Історія виникнення стандартів до розробки програмних продуктів. ДСТУ для розробки програмних продуктів. Стандарт ISO: процеси життєвих циклів за ISO, зміст основних процесів, стадії створення, оцінка якості за стандартом ISO. Галузевий стандарт IEEE – SWEBOK: стандарти процесів підприємництва за SWEBOK, модель зрілості компаній-розробників, визначення рівнів зрілості, області процесів розробки для різних рівнів зрілості. Переваги та недоліки різних стандартів розробки програмних продуктів.

#### **Тема 6. Проектування програмних продуктів.**

Поняття проектування та архітектури програмного продукту. Основні поняття універсальної мови моделювання UML. Рекомендації щодо побудови архітектури.

#### **Тема 7. Архітектура програмних засобів.**

Стандартизований та загальносистемний підходи до проектування архітектури програмних продуктів. Види архітектур, основні архітектурні шаблони.

### **МОДУЛЬ 2**

#### **Змістовий модуль 2. Розробка та тестування програмних продуктів.**

#### **Тема 8. Автоматизація розробки програмних продуктів.**

Класифікація інструментальних середовищ розробки програмних продуктів. Комп'ютерні технології розробки програмних продуктів – CASE: класифікація CASE-технологій, їхні переваги, основні зміни життєвого циклу при використанні CASE, зміна трудовитрат по фазах життєвого циклу програмних продуктів. Структурний аналіз: засоби функціонального, інформаційного та подійного моделювання.

#### **Тема 9. Патерни проектування.**

Історія виникнення шаблонів проектування – патернів. Класифікація патернів. Структурні патерни. Патерни поведінки класів/об'єктів. Твірні патерни.

#### **Тема 10. Контроль якості програмних продуктів.**

Поняття якості. Шаблони рапорту про помилку (баг-репорту). Тестування програмних систем: класифікація видів тестування, статичні методи тестування, динамічні методи тестування, функціональне тестування.

#### **Тема 11. Тестування програмних продуктів.**

Рівні тестування: компонентне або модульне тестування, інтеграційне тестування, системне тестування, приймальне тестування. Автоматизоване тестування. Засоби автоматизації тестування Навантажувальне тестування.

#### **Тема 12. Верифікація та валідація інформаційних систем.**

Валідація програмних продуктів. Верифікація програмних продуктів: верифікація об'єктних моделей, базові методи доведення правильності програм, модель доведення програми за твердженнями, верифікація композиції компонентів, майбутнє верифікації програм.

#### **Тема 13. Засоби построзробки (підтримки) програмних систем**

Підтримка програмних продуктів після розробки та її складові. Сучасні засоби автоматизації збору та опрацювання даних від користувачів. Оновлення програмних продуктів.

#### 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	Денна форма					
	Усього	у тому числі				
Л		П	лаб	інд	Ср	
1	2	3	4	5	6	7
<b>МОДУЛЬ 1</b>						
<b>Змістовий модуль 1. Вимоги та проектування програмних продуктів.</b>						
Тема 1. Поняття програмного продукту, основна термінологія.	2	2		-		-
Тема 2. Вимоги до програмних продуктів.	16	2		8		6
Тема 3. Засоби контролю версій.	11	2		-		9
Тема 4. Життєві цикли програмних продуктів.	12	2		4		6
Тема 5. Стандарти розробки програмних продуктів.	12	4				8
Тема 6. Проектування програмних продуктів.	8	2		4		2
Тема 7. Архітектура програмних засобів.	6	2		4		
<i>Разом – МОДУЛЬ 1</i>	<b>57</b>	<b>16</b>		<b>10</b>		<b>31</b>
<b>МОДУЛЬ 2</b>						
<b>Змістовий модуль 2. Розробка та тестування програмних продуктів.</b>						
Тема 8. Автоматизація розробки програмних продуктів.	10	2				8
Тема 9. Патерни проектування	23	6		8		9
Тема 10. Контроль якості програмних продуктів.	17	2				15
Тема 11. Тестування програмних продуктів.	15	2		4		9
Тема 12. Верифікація та валідація інформаційних систем.	6	2		4		
Тема 13. Засоби построзробки (підтримки) програмних систем	16	2				14
<i>Разом – МОДУЛЬ 2</i>	<b>83</b>	<b>16</b>		<b>12</b>		<b>55</b>
<b>Усього годин</b>	<b>150</b>	<b>32</b>		<b>32</b>		<b>86</b>

#### 5. Темі лекційних занять

№ з/п	Назва теми	Кількість годин
1	Поняття програмного продукту, основна термінологія.	2
2	Вимоги до програмних продуктів.	2
3	Засоби контролю версій.	2
4	Життєві цикли програмних продуктів.	2

5	Стандарти розробки програмних продуктів.	4
6	Проектування програмних продуктів.	2
7	Архітектура програмних засобів.	2
8	Автоматизація розробки програмних продуктів.	2
9	Патерни проектування.	6
10	Контроль якості програмних продуктів.	2
11	Тестування програмних продуктів.	2
12	Верифікація та валідація інформаційних систем.	2
13	Засоби построзробки (підтримки) програмних систем	2
	<b>Разом</b>	<b>32</b>

### **6. Теми семінарських занять**

Семінарські заняття в курсі не передбачені.

### **7. Теми практичних занять**

Практичні заняття в курсі не передбачені.

### **8. Теми лабораторних занять**

№ з/п	Назва теми	Кількість годин
1	Витягнення та опрацювання вимог клієнта.	4
2	Написання технічного завдання.	4
3	Вибір моделі життєвого циклу програмного продукту.	4
4	Побудова загальної архітектури програмного засобу.	4
5	Проектування модулів та структури класів.	4
6	Розробка модулів з використанням патернів проектування.	4
7	Покриття модулів тестами.	4
8	Збірка та реліз програмного засобу	4
	<b>Разом</b>	<b>32</b>

### **9. Самостійна робота**

№ з/п	Назва теми	Кількість Годин
1	Опрацювання лекційного матеріалу.	16
2	Підготовка до лабораторних робіт, оформлення звітів.	16
3	Підготовка до модульних контрольних робіт.	8
4	Ознайомлення із засобами контролю версій на практиці.	9
5	Ознайомлення із засобами автоматизованого високорівневого тестування.	15
6	Підготовка модульних тестів.	9
7	Робота із засобами підтримки програмних продуктів.	14
	<b>Разом</b>	<b>86</b>

### **10. Методи навчання**

Навчальні заняття проводяться у формі лекційних та лабораторних робіт. Лекція – основна форма проведення навчальних занять, призначена для засвоєння теоретичного матеріалу. Під час лабораторних занять студент під керівництвом викладача набуває

практичних навичок зі створення прикладних програм комплексів та систем. Лабораторні заняття проводяться у навчальному комп'ютерному класі.

Самостійна робота студента є основним засобом оволодіння навчальним матеріалом у час, вільний від обов'язкових навчальних занять. Самостійна робота студентів складається з таких видів робіт:

- підготовка до лекцій та лабораторних робіт,
- оформлення звітів про виконані лабораторні роботи,
- самостійне опрацювання окремих тем навчальної дисципліни,
- підготовка до модульних контрольних робіт.

### **11. Методи контролю**

Оцінка якості засвоєння навчальної дисципліни включає поточний контроль успішності та складання підсумкового екзамену.

Для поточного контролю засвоєння студентами навчального матеріалу передбачається виконання та захист 8 обов'язкових лабораторних робіт та написання семестрового проекту.

Екзамен проводиться в письмово-усній формі.

### **12. Розподіл балів, що присвоюється студентам для іспиту**

Контроль засвоєння матеріалу включає поточний контроль (семестровий проект 20 балів), лабораторні роботи та оцінка відповідей (30 балів) та іспит (50 балів). Сумарна оцінка виставляється за 100-бальною шкалою.

При оцінюванні лабораторної роботи враховується підготовка до виконання лабораторної роботи, хід виконання лабораторної роботи, оформлення звіту, отримані результати та захист звіту про виконану лабораторну роботу.

Поточне тестування та самостійна робота		Лабораторні роботи	Іспит	Сума
Модуль 1	Модуль 2			
T1 – T9	T10 – T18			
10	10	30	50	100

#### **Шкала оцінювання: Університету, національна та ECTS**

Сума балів за всі види навчальної діяльності	Оцінка за ECTS	Оцінка за національною шкалою		
			іспит	залік
90-100	<b>A</b>	<b>5</b>	відмінно	зараховано
81-89	<b>B</b>	<b>4</b>	добре	
71-80	<b>C</b>			
61-70	<b>D</b>	<b>3</b>	задовільно	
51-60	<b>E</b>			
21-50	<b>FX</b>	<b>2</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-20	<b>F</b>	<b>2</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни



### ***13. Рекомендована література***

#### **Базова**

1. Макконнелл С. Совершенный код. Мастер-класс. / С. Макконнелл – М.: Microsoft Press, 2017. – 896 с.
2. Лавріщева К. М. Програмна інженерія / К. М.-Лавріщева Підручник. – К.: Академперіодика, 2008.-319 с.
3. Guckkenheimer S. Software Engineering With Microsoft Visual Studio. Team System. S. Guckkenheimer, J. Peter – Adison Wesley, 2006. – 273 p.
4. Abran A. Guide to the Software Engineering Body of Knowledge / A. Abran, J. W. Moore, P. Bourque, R. Dupuis – 2004. IEEE Computer Society. ISBN 0-7695-2330-7.
5. ISO/IEC 12207: 1995–0801: Informational Technology – Software life cycle processes.
6. Будаї А. Дизайн патерни – просто, як двері / А, Будаї – 2012 – 90 с.
7. Канер К. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений. / К. Канер, Д. Фолк, Е. К. Нгуен – Киев : ДиаСофт, 2001. – 544 с. – ISBN 9667393879.

### ***14. Інформаційні ресурси***

1. <https://developer.android.com/ndk/>