

ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«ПРИДНІПРОВСЬКА ДЕРЖАВНА АКАДЕМІЯ БУДІВНИЦТВА ТА АРХІТЕКТУРИ»

КАФЕДРА КОМП'ЮТЕРНИХ НАУК, ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
ТА ПРИКЛАДНОЇ МАТЕМАТИКИ

(повна назва кафедри)

«ЗАТВЕРДЖУЮ»

Проректор з навчально-виховної  
роботи  
Галина ЄВСЄЄВА



« 01 » вересня 2021 року

## СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### Архітектура та проектування програмного забезпечення

спеціальність	122 «Комп'ютерні науки»
освітньо-професійна програма	«Комп'ютерні науки»
освітній ступінь	бакалавр
форма навчання	денна
розробники	Шибко Оксана Миколаївна, Семенець Сергій Миколайович

#### 1. АНОТАЦІЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Дисципліна «Архітектура та проектування програмного забезпечення» є варіативною компонентою циклу професійної підготовки бакалаврів за спеціальністю 122 «Комп'ютерні науки». Предметом вивчення дисципліни є структура та методи проектування програмного забезпечення (ПЗ). Вивчення даної дисципліни дає знання, що використовуються на етапі проектування ПЗ при розробці програмного дизайну.

Згідно з навчальною програмою дисципліни розглядаються наступні основні питання: структура ПЗ; системне, прикладне, сервісне, службове та інструментальне ПЗ; етапи та моделі життєвого циклу програмного продукту; управління процесом проектування ПЗ; класифікація вимог до ПЗ; атрибути якості; аналіз вимог та документування; стратегії і методи проектування ПЗ; об'єктно-орієнтоване, функціонально-орієнтоване та компонентне проектування ПЗ; архітектурні фреймворки; case- технології проектування ПЗ; принципи об'єктно-орієнтованого проектування; аналіз та оцінка якості програмного дизайну; уніфікована мова моделювання UML; діаграми UML; патерни проектування ПЗ; підвищення проектної надійності інформаційних систем.

## 2. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

	Години	Кредити	Семестр
			IV
Всього годин за навчальним планом, з них:	135	4,5	135
<b>Аудиторні заняття, у т.ч:</b>	38		38
лекції	16		16
лабораторні роботи			
практичні заняття	22		22
<b>Самостійна робота, у т.ч:</b>	97		97
підготовка до аудиторних занять	12		12
підготовка до контрольних заходів	12		12
виконання курсового проекту	30		30
опрацювання розділів програми, які не викладаються на лекціях	13		13
підготовка до екзамену	30		30
<b>Форма підсумкового контролю</b>			Екзамен

## 3. СТИСЛИЙ ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

**Мета дисципліни** - теоретична та практична підготовка фахівців з питань структури і проектування програмного забезпечення інформаційних систем.

**Завдання дисципліни** - опанування студентами сучасними методами проектування програмного забезпечення; отримання знань про архітектуру, методи аналізу, побудови та тестування програмних засобів; отримання практичних навичок з розробки програмних продуктів.

**Пререквізити дисципліни.** «Основи обчислювальної техніки», «Алгоритмізація і програмування», «Конструювання програмного забезпечення».

**Постреквізити дисципліни.** «Об'єктно-орієнтоване програмування», «Проектування інформаційних систем», «Основи програмної інженерії»

### Компетентності.

#### Інтегральна компетентність:

**ІК.** Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов

#### Загальні компетентності:

**ЗК-2.** Здатність застосовувати знання у практичних ситуаціях.

**ЗК-3.** Знання та розуміння предметної області та розуміння професійної діяльності.

#### Спеціальні (фахові, предметні) компетентності:

**СК-8.** Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: структурного, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами та алгоритмами обчислень, структурами даних і механізмами управління.

#### Програмні результати навчання :

**ПР1.** Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.

**ПР2.** Використовувати сучасний математичний апарат неперервного та дискретного аналізу, лінійної алгебри, аналітичної геометрії, в професійній діяльності для розв'язання задач

теоретичного та прикладного характеру в процесі проектування та реалізації об'єктів інформатизації.

**ПР3.** Використовувати знання закономірностей випадкових явищ, їх властивостей та операцій над ними, моделей випадкових процесів та сучасних програмних середовищ для розв'язування задач статистичної обробки даних і побудови прогнозних моделей.

**ПР4.** Використовувати методи обчислювального інтелекту, машинного навчання, нейромережевої та нечіткої обробки даних, генетичного та еволюційного програмування для розв'язання задач розпізнавання, прогнозування, класифікації, ідентифікації об'єктів керування тощо.

**ПР5.** Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

**ПР6.** Використовувати методи чисельного диференціювання та інтегрування функцій, розв'язання звичайних диференціальних та інтегральних рівнянь, особливостей чисельних методів та можливостей їх адаптації до інженерних задач, мати навички програмної реалізації чисельних методів.

**ПР7.** Розуміти принципи моделювання організаційно-технічних систем і операцій; використовувати методи дослідження операцій, розв'язання одно- та багатокритеріальних оптимізаційних задач лінійного, цілочисельного, нелінійного, стохастичного програмування.

**ПР8.** Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проектування динамічних процесів в макроекономічних, технічних, технологічних і фінансових об'єктах.

**ПР9.** Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

**ПР10.** Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

**ПР11.** Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

**ПР12.** Застосовувати методи та алгоритми обчислювального інтелекту та інтелектуального аналізу даних в задачах класифікації, прогнозування, кластерного аналізу, пошуку асоціативних правил з використанням програмних інструментів підтримки багатовимірного аналізу даних на основі технологій DataMining, TextMining, WebMining.

**ПР13.** Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

**ПР14.** Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

**ПР15.** Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

**ПР16.** Розуміти концепцію інформаційної безпеки, принципи безпечного проектування програмного забезпечення, забезпечувати безпеку комп'ютерних мереж в умовах неповноти та невизначеності вихідних даних.

**ПР17.** Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення.

У результаті вивчення навчальної дисципліни студент повинен **знати:**

- структуру ПЗ;
- моделі життєвого циклу програмного продукту;
- вимоги до програмного забезпечення;
- стратегії і методи проектування програмного забезпечення;
- архітектурні фреймворки та CASE- технології проектування;
- принципи об'єктно-орієнтоване проектування;
- уніфіковану мову моделювання UML;
- критерії оцінювання якості дизайну програмного забезпечення;
- патерни проектування програмного забезпечення;
- методи підвищення проектної надійності інформаційних систем.

**вміти:**

- орієнтуватися в структурі сучасного програмного забезпечення;
- оцінювати якість програмного дизайну;
- аналізувати вимоги та розробляти відповідний дизайн програмного забезпечення;
- розробляти проектну документацію для опису та підтримки програмного забезпечення;
- управляти процесом проектування програм;
- проектувати програмні продукти згідно з наданим технічним завданням.

**Методи навчання** – практичний, наочний, словесний, робота з книгою. Практичних навичок студенти набувають на аудиторних лекційних та практичних заняттях.

**Форми навчання** – аудиторна, позааудиторна, індивідуальна.

#### 4. СТРУКТУРА (ТЕМАТИЧНИЙ ПЛАН) ДИСЦИПЛІНИ

Назва змістових модулів і тем	Кількість годин, у тому числі				
	усього	л	п	лаб	с/р
<b>Змістовий модуль 1. Структура, етапи розробки та вимоги до ПЗ</b>					
Структура програмного забезпечення.	8	2			6
Етапи розробки програмного продукту.	9	2	2		5
Вимоги до ПЗ.	10	2	4		4
Стратегії і методи проектування ПЗ.	10	2	4		4
<b>Разом за змістовим модулем 1</b>	<b>37</b>	<b>8</b>	<b>10</b>		<b>19</b>
<b>Змістовий модуль 2. Проектування ПЗ</b>					
Принципи об'єктно - орієнтованого проектування.	7	2			5
Уніфікована мова моделювання UML.	12	2	6		4
Патерни проектування.	10	2	4		4
Підвищення проектної надійності ІС .	9	2	2		5
<b>Разом за змістовим модулем 2</b>	<b>38</b>	<b>8</b>	<b>12</b>		<b>18</b>
<b>Змістовий модуль 3. Курсовий проект</b>					
Аналіз вимог та декомпозиція системи.	4				4
Обрання та освоєння патерну проектування з урахуванням вимог до системи.	4				4
Побудова діаграми варіантів використання UML.	2				2



Побудова діаграми класів UML.	4			4
Побудова діаграми послідовності UML.	4			4
Аналіз та оцінка якості програмного дизайну.	2			2
Розрахунок показників надійності системи.	2			2
Розробка моделі оптимального резервування системи з урахуванням надійності.	2			2
Розробка оптимальної структурної схеми надійності системи.	2			2
Оформлення пояснювальної записки.	4			4
<b>Разом за змістовим модулем 3</b>	<b>30</b>			<b>30</b>
<b>Підготовка до екзамену</b>	<b>30</b>			<b>30</b>
<b>Усього годин</b>	<b>135</b>	<b>16</b>	<b>22</b>	<b>97</b>

### 5. ЛЕКЦІЙНИЙ КУРС

№ заняття	Тема занять	Кількість годин
1	<b>Структура програмного забезпечення.</b> Предмет та задачі дисципліни. Програмне забезпечення (ПЗ) комп'ютера. Системне ПЗ. Операційна система. Драйвери. Планування процесів. Синхронізація та обмін даними між процесами. Керування пам'яттю. Керування файлами. Системи керування вводом-виводом. Переривання. Огляд операційних систем Windows, Linux, Macintosh. Прикладне ПЗ. Браузери. Текстові і графічні редактори. Пакети презентацій. Табличні процесори, системи управління базами даних, мультимедійні програми та САПР. Службове і сервісне ПЗ. Мережеві протоколи. Утиліти. Архіватори та антивірусні програми. Інструментальне ПЗ. Редактори, компілятори та бібліотеки мов програмування. Асемблери.	2
2	<b>Етапи розробки програмного продукту.</b> Життєвий цикл програмного продукту (ПП). Основні етапи життєвого циклу ПП. Планування. Розробка технічного завдання (формування вимог до системи). Аналіз вимог та проектування. Етап конструювання. Інтеграція і тестування. Супровід і експлуатація. Моделі життєвого циклу ПП. Каскадна, спіральна та інкрементна моделі. Стороння розробка ПП. Заміна версії та управління конфігурацією системи.	2
3	<b>Вимоги до ПЗ.</b> Класифікація вимог до програмного забезпечення. Бізнес вимоги та бізнес правила. Вимоги користувача. Функціональні і не функціональні вимоги. Ефективність, надійність та безпека системи. Атрибути якості. Вимоги до даних. Вимоги до інтерфейсів. Аналіз вимог та документування.	2
4	<b>Стратегії і методи проектування ПЗ.</b> Загальні стратегії проектування ПЗ. Функціонально-орієнтоване (структурне) проектування ПЗ. Проектування «зверху-вниз» і «знизу-вгору». Компонентне проектування ПЗ. Об'єктно-орієнтоване проектування ПЗ. Архітектурні фреймворки. CASE - технології проектування. Нотації проектування.	2
5	<b>Принципи об'єктно-орієнтованого проектування.</b> Принципи SOLID як базові принципи об'єктно-орієнтованого проектування ПЗ. Принцип єдиної відповідальності. Принцип відкритості / закритості. Принцип підстановки. Принцип поділу інтерфейсу. Принцип інверсії залежностей. Інкапсуляція, абстракція, поліморфізм, спадкування та принципи SOLID. Критерії якісної та неякісної архітектури. Розширюваність, гнучкість, масштабованість, тестованість та ремонтпридатність програми. Можливість	2

	перевикористання модулів. Жорсткість, крихкість, нерухомість. Аналіз та оцінка якості програмного дизайну.	
6	<b>Уніфікована мова моделювання UML.</b> Мова UML як засіб моделювання ПЗ. Основні будівельні блоки UML. Структурні та поведінкові сутності. Сутності, що групують та сутності, що анотують. Діаграми UML. Діаграма класів як засіб моделювання статичної структури програмної системи в термінах класів та відношень між ними. Графічне зображення класів. Атрибути та операції (методи) класу. Абстрактні класи. Інтерфейси. Умовне графічне зображення відносин залежності, асоціації, композиції, агрегації, узагальнення та реалізації. Приклади складання діаграм класів. Діаграми варіантів використання, послідовності, станів та діяльності. Інші діаграми.	2
7	<b>Патерни проектування.</b> Призначення та класифікація патернів проектування. Породжуючі, структурні та поведінкові дизайн-патерни. Огляд дизайн - патернів Builder, Facade, Adapter, Decorator, Observer, Mediator.	2
8	<b>Підвищення проектної надійності інформаційних систем.</b> Визначення надійності ІС. Розрахунок показників надійності систем з логічно послідовним, логічно паралельним та змішаним з'єднанням елементів. Метод структурного резервування для підвищення надійності складних систем. Оцінка надійності резервованої системи.	2
	<b>Усього годин</b>	16

### 6. ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

№ занять	Тема занять	Кількість годин
1-3	Розробка UML-діаграм.	6
4-6	Проектування уніфікованого програмного інтерфейсу.	6
7-9	Розробка програмного дизайну.	6
10-11	Проектування ІС з урахуванням надійності.	4
	<b>Усього годин</b>	22

### 7. ТЕМИ ЛАБОРАТОРНИХ ЗАНЯТЬ

Лабораторні заняття навчальним планом не передбачені.

### 8. САМОСТІЙНА РОБОТА

№ п/п	Назва теми	Кількість годин
1	підготовка до аудиторних занять	12
2	підготовка до контрольних заходів	12
3	виконання курсового проекту	30
4	опрацювання розділів програми, які не викладаються на лекціях	13
3	підготовка до екзамену	30
	<b>Усього годин</b>	97

### 9. МЕТОДИ КОНТРОЛЮ

Письмовий, усний, практична перевірка.

## 10. ПОРЯДОК ТА КРИТЕРІЇ ОЦІНЮВАННЯ

Контроль успішності студента здійснюється за допомогою 100-бальної системи оцінювання, що має відповідні оцінки в національній шкалі і шкалі ECTS.

### Змістовий модуль 1. Структура, етапи розробки та вимоги до ПЗ.

#### *Самостійна домашня робота* (максимальна кількість балів – 100):

##### «Домашнє завдання №1»

- виконання самостійної роботи та її оформлення (самостійна робота повинна бути виконана в повному обсязі відповідно до завдання) – 60 балів;
- відповідь на теоретичне питання №1, №2 при захисті самостійної роботи (максимальна кількість балів на одне питання - 20 балів): 40 балів;
  - правильна відповідь на питання 20 балів;
  - відповідь на питання потребує деяких уточнень щодо принципів конвеєрної та паралельної розробки 16 – 19 балів;
  - відповідь на питання із незначними помилками (сутність розкрита, але були помилки в місцях паралельних обчислень) 6 -15 балів;
  - відповідь на питання зі значними помилками (не розкрита сутність питання) 1 - 5 балів;
  - неправильна відповідь або немає відповіді 0 балів.

Підсумкова оцінка зі змістового модуля 1 визначається як оцінка за самостійну домашню роботу змістового модуля 1.

### Змістовий модуль 2. Проектування ПЗ.

#### *Самостійна домашня робота* (максимальна кількість балів – 100):

##### «Домашнє завдання №2»

- виконання самостійної роботи та її оформлення (самостійна робота повинна бути виконана в повному обсязі відповідно до завдання) – 60 балів;
- відповідь на теоретичне питання №1, №2 при захисті самостійної роботи (максимальна кількість балів на одне питання - 20 балів): 40 балів;
  - правильна відповідь на питання (правильна розробка моделі) 20 балів;
  - відповідь на питання потребує деяких уточнень щодо моделі для оцінки можливого підвищення продуктивності 16 – 19 балів;
  - відповідь на питання із незначними помилками (сутність розкрита, але були помилки в проектуванні програмного забезпечення) 6 -15 балів;
  - відповідь на питання зі значними помилками (не розкрита сутність питання) 1 - 5 балів;
  - неправильна відповідь або немає відповіді 0 балів.

Підсумкова оцінка зі змістового модуля 2 визначається як оцінка за самостійну домашню роботу змістового модуля 2.

### Змістовий модуль 3. Курсовий проект

#### Курсовий проект (максимальна кількість балів – 100):

- правильне виконання курсового проекту та його оформлення (курсний проект повинен бути виконана в повному обсязі відповідно до завдання) – 60 балів;

При захисті курсового проекту студент повинен відповісти на 2 теоретичних питання і пояснити розрахункове завдання з кожного розділу проекту.

- відповідь на теоретичні питання №1, №2, (максимальна кількість балів на одне питання - 10 балів): 20 балів;

- правильна відповідь на питання 10 балів;
- відповідь на питання потребує деяких уточнень щодо розробки схеми або аналізу ефективності 6 – 9 балів;
- відповідь на питання із незначними помилками (сутність розкрита, але були помилки в формулах або їх застосуванні, висновки недостатньо повні) 3 – 5 балів;

- відповідь на питання зі значними помилками (помилки в формулах, неповне пояснення сутності питання, висновки поверхневі або відсутні) 1 – 2 балів;
- неправильна відповідь або немає відповіді 0 балів.
- виконання розрахункових завдань (максимальна кількість балів на одне завдання - 10 балів):
- правильне виконання завдання 10 балів;
- виконання завдання потребує деяких уточнень щодо реалізації паралельної або послідовної програм 6 – 9 балів;
- виконання завдання із незначними помилками (незначні помилки в розрахунках або в графічному відображенні результатів, неповне пояснення результату) 3 – 5 балів;
- виконання завдання зі значними помилками (неправильні розрахункові формули або їх застосування, неповне або неправильне обґрунтування розв'язання завдання) 1 - 2 балів;
- неправильне виконання завдання або немає розв'язання 0 балів.

### **- екзамену**

Екзамен проводиться в письмовій формі у вигляді відповідей на білети, що містять 2 теоретичних питання і 1 практичного завдання. До складання екзамену допускаються студенти, які повністю виконали навчальний план дисципліни – отримали не менше 60 балів з кожного змістового модуля.

Екзамен (максимальна кількість балів – **100**):

- відповідь на теоретичні питання (максимальна кількість балів на одне питання - 25 балів):
- правильна відповідь на питання 50 балів;
- відповідь на питання потребує деяких уточнень щодо моделі або методології 25 балів;
- відповідь на питання із незначними помилками (сутність розкрита, але були помилки в визначеннях) 19 – 24 балів;
- відповідь на питання зі значними помилками (не розкрита сутність питання) 9 -18 балів;
- неправильна відповідь або немає відповіді 0 балів.
- виконання практичних завдань максимальна кількість балів: 50 балів;
- правильне виконання завдання, складена модель або побудована діаграма 50 балів;
- робота виконана повністю, але виконання завдання потребує деяких уточнень, аналіз результатів недостатній 40 – 49 балів;
- робота виконана повністю, але виконання завдання потребує деяких уточнень, аналіз результатів відсутній 30 – 39 балів
- виконання завдання із незначними помилками (сутність розкрита, але були помилки в моделюванні або аналізі паралельних обчислень) 20 -29 балів;
- виконання завдання зі значними помилками (неправильно складена модель або помилки при паралельних обчисленнях, неповне обґрунтування завдання ) 10-19 балів;
- виконання завдання зі значними помилками (розрахунки паралельних обчислень виконані зі значними помилками, немає обґрунтування завдання ) 9 - 1 балів
- неправильне виконання завдання або немає розв'язання 0 балів.

**Підсумкова оцінка з дисципліни** розраховується як середня між підсумковою оцінкою змістових модулів 1,2,3 та оцінкою екзамену.

## **11. ПОЛІТИКА КУРСУ**

Порядок зарахування пропущених занять:

- пропущена лекція відпрацьовується підготовкою конспекту відповідно до теми пропущеного заняття та його захистом;



- пропущені практичні заняття відпрацьовуються студентами виконанням відповідної практичної роботи самостійно та її захистом.

Зміни в нарахуванні балів у випадках несвоєчасного виконання завдань не відбувається.

Дотримання академічної доброчесності студента передбачає:

- самостійне та добросовісне виконання завдань, в тому числі поточного та підсумкового контролю;
- відповідальне ставлення до своїх обов'язків;
- повага до честі й гідності інших осіб;
- посилення на джерела інформації у разі запозичення ідей, розробок, тверджень, відомостей;
- використання при виконанні завдань лише перевірених та достовірних джерел інформації.

За порушення академічної доброчесності студент може бути притягнутий до академічної відповідальності (повторне проходження оцінювання).

Також неприємливим у навчальній діяльності студентів є академічний плагіат, самоплагіат, фальсифікація та інші види академічної нечесності.

## 12. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

### Основна

1. Бородкіна І., Бородкин Г. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів / І. Бородкіна, Г. Бородкин – М: Центр навчальної літератури, 2018. – 204 с.
2. Гагарина Л. Г. Введение в архитектуру программного обеспечения, Учебное пособие. - М.: Инфра-М, 2016. - 320 с
3. Завгородній В.В., Ялова К.М. Конспект лекцій з дисципліни «Архітектура та проектування програмного забезпечення» для здобувачів вищої освіти із спеціальності 121 – «Інженерія програмного забезпечення» / Укл. В.В.Завгородній, К.М.Ялова.– Кам'янське: ДДТУ, 2019.– 144с
4. Мартін Р. Чистий код / Р. Мартін – М.: Фабула, 2019. – 416 с.
5. Постіл С. Д. UML. уніфікована мова моделювання інформаційних систем / С. Д. Постіл : Ун-т держ. фіск. служби України. - Ірпінь : Ун-т держ. фіск. служби України, 2019. - 321 с.
6. Роберт М. Чиста архітектура: мистецтво розробки програмного забезпечення» / Роберт Мартін, Фабула, 2019. – 416 с.
7. Фаулер М. Архитектура корпоративных программных приложений. - М: Вильямс, 2016. - 541с.

### Допоміжна

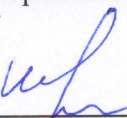
1. M. Fowler and al., Refactoring: Improving the Design of Existing Code, Addison-Wesley, 2012.
2. Russell Gold, Thomas Hammell, Tom Snyder. Test Driven Development: A J2EE Example.- Apress, 2015.- 296 pages.
3. Амблер С. Гибкие технологии: Экстремальное программирование и унифицированный процесс разработки. Библиотека программиста.—Спб.: Питер, 2015.—412 с.
4. Бородкіна І., Бородкин Г. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів. – К. «Центр навчальної літератури», 2018. – 204 с.
5. Влссидес Джон. Применение шаблонов проектирования. Дополнительные штрихи.: Пер. с англ.— М.: Издательский дом «Вильямс», 2013. — 144 с
6. Гамма Е., Хелм Р., Джонсон Р., Влссидес Д. Приемы объектно-ориентированного проектирования. Паттерны проектирования, - ДМК Пресс, 2010. -366 с.

7. Иан Соммервилл. Инженерия программного обеспечения, 6-е издание.: Пер. с англ.—М.:Издательский дом «Вильямс», 2012.—624 с.
8. Лаврищева К.М. Програма інженерія.—К.— 2018.—319 с.
9. Мартин Р. Быстрая разработка программ: принципы, примеры, практика. Пер. с англ.—М.: Издательский дом «Вильямс», 2014.—752 с.
10. Макконел С. Совершенный код. Мастер класс / Пер. с англ.—М.: Издательско-торговый дом «Русская редакция»; Спб.: Питер, 2015.—896 с.
11. Табунщик Г. В., Каплиенко Т.І., Петрова О.А. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплиенко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с
12. Фаулер М. Рефакторинг: улучшение существующего кода.—Пер. с англ.—СПб: Символ-Плюс, 2013.—432 с.
13. ДСТУ 3918-99 (ISO/IEC 12207:1995) Інформаційна технологія. Процеси життєвого циклу програмного забезпечення (Аналог – ГОСТ Р ИСО/МЭК 12207-99 Информационная технология. Процессы жизненного цикла программных средств) [Электронный ресурс] // Электронная библиотека ГОСТов. – Режим доступа: <http://www.it-gost.ru/content/view/73/40/>

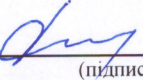
### 13. INTERNET-РЕСУРСИ

1. Цифровий репозиторій ХНУГХ ім. А. Н. Бекетова / [Електронний ресурс]. – Режим доступу: <http://eprints.kname.edu.ua/>
2. Цифровий репозиторій Харківського національного університету імені В.Н. Каразіна / [Електронний ресурс]. – Режим доступу: <http://dspace.univer.kharkov.ua/handle/123456789/568>
3. Наукова бібліотека НУВГП (м. Рівне, вул. Олекси Новака, 75) / [Електронний ресурс]. – Режим доступу: <http://nuwm.edu.ua/naukova-biblioteka> [http://nuwm.edu.ua/MySQL/page\\_lib.php](http://nuwm.edu.ua/MySQL/page_lib.php)
4. UML Tutorial [Електронний ресурс]. – Режим доступу: URL: <https://www.tutorialspoint.com/uml/index.htm>. – Назва з екрану.
5. Компонентный подход в программировании [Електронний ресурс]. – Режим доступу: [http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=2778](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=2778)
6. Архитектурное проектирование программного обеспечения [Електронний ресурс]. – Режим доступу: <http://www.lib.tpu.ru/fulltext2/m/2011/m416.pdf>
7. Обзор паттернов проектирования [Електронний ресурс]. – Режим доступу: <http://www.lib.tpu.ru/fulltext2/m/2011/m310.pdf>

Розробники

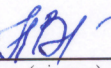
  
\_\_\_\_\_ (підпис)

(Оксана ШИБКО)

  
\_\_\_\_\_ (підпис)

(Сергій СЕМЕНЕЦЬ)

Гарант освітньої програми

  
\_\_\_\_\_ (підпис)

(Наталя ВЕЛЬМАГІНА)

Силабус затверджено на засіданні кафедри комп'ютерних наук, інформаційних технологій та прикладної математики

Протокол від «30» серпня 2021 року № 1