

**Чернівецький національний університет імені Юрія Федьковича**

(повне найменування закладу вищої освіти)

**Навчально-науковий інститут фізико-технічних і комп'ютерних наук**

(назва інституту/факультету)

**Кафедра комп'ютерних систем та мереж**

(назва кафедри)

## **СИЛАБУС**

**навчальної дисципліни**

### ***Технології проектування програмних систем***

(вказати назву навчальної дисципліни (іноземною, якщо дисципліна викладається іноземною мовою))

#### **Обов'язкова**

(обов'язкова чи вибіркова)

**Освітньо-професійна програма – “Комп'ютерна інженерія”**

**Спеціальність 123 – Комп'ютерна інженерія**

(шифр і назва спеціальності)

**Галузь знань 12 – Інформаційні технології**

(шифр і назва галузі знань)

**Рівень вищої освіти – другий (магістерський)**

(вказати: перший (бакалаврський)/другий (магістерський)/третій (освітньо-науковий))

**Навчально-науковий інститут фізико-технічних і комп'ютерних наук**

(назва факультету / інституту, на якому здійснюється підготовка фахівців за вказаною освітньо-професійною програмою)

**Мова навчання – українська**

(мова, на якій читається дисципліна)

**Розробники: Двірничук Костянтин Васильович, асистент кафедри КСМ, канд. фіз.-мат. наук,**

(вказати авторів (викладач (ів)), їхні посади, наукові ступені, вчені звання)

**Профайл викладача (-ів)** <https://csn.chnu.edu.ua/>,  
<https://csn.chnu.edu.ua/employees/dvirnychuk-kostyantyn-vasylovych/>

**Контактний тел.** +(38) 0372 50 94 32 (кафедра КСМ) – Двірничук К.В.

**E-mail:** [k.dvirnychuk@chnu.edu.ua](mailto:k.dvirnychuk@chnu.edu.ua)

**Сторінка курсу в Moodle** <https://moodle.chnu.edu.ua/course/view.php?id=3145>

**Консультації**  
*on-line: понеділок з 17.00 до 18.00*  
*Очні консультації: четвер з 17.00 до 18.00*

## 1. Анотація дисципліни

Курс «Технології проектування програмних систем» призначений для надання навиків створення програмних додатків з використанням шаблонів (патернів) проектування програмного забезпечення, для вміння провести тестування розроблених програмних додатків, для розуміння основних принципів архітектурних шаблонів ПЗ.

Вивчення даної вибіркової дисципліни надає студентам ряд переваг, оскільки сучасне програмування не можна представити собі без проектування структури самої програми на основі архітектурних шаблонів, патернів проектування. Зокрема одним із популярних рішень архітектурного шаблону є MVC-шаблон, основи якого і подаються в рамках в цієї дисципліни.

**1.1. Мета навчальної дисципліни:** надати студентам систематизовані знання про мету, завдання та принципи технологій проектування програмних систем. Навчити студентів використовувати шаблони проектування (патерни) в своїх програмних розробках. Пояснити студентам суть архітектурних шаблонів та більш детально розібратись з MVC-програмуванням. Надати можливість студентам освоїти деякі сучасні технології та надати їм поняття та підходи тестування та відлагодження програм.

**1.2. Завдання** – на основі отриманих теоретичних знань виробити у студентів уміння розробляти власні програмні продукти, використовувати в своїх програмах шаблони проектування, будувати MVC-моделі програм, тестувати програми. Створювати програмні додатки з допомогою сучасних технологій.

**1.3. Пререквізити.** Для коректного розуміння і засвоєння матеріалу даного курсу слухачі повинні попередньо пройти курси з: Програмування, Об'єктно-орієнтоване програмування, Інженерія програмного забезпечення.

## 2. Результати навчання

У результаті вивчення навчальної дисципліни студент повинен

**2.1. Знати:** технології програмування, паттерни проектування, будову MVC-моделей програм, методи тестування програмних розробок.

**2.2. Вміти:** будувати програмні додатки, використовувати для цього шаблони проектування та архітектурні шаблони, проводити тестування програмних додатків.

## 2.3. Набути компетентностей:

### ЗК - загальних

- ЗК1. Здатність до адаптації та дій в новій ситуації.
- ЗК2. Здатність до абстрактного мислення, аналізу і синтезу.
- ЗК3. Здатність проводити дослідження на відповідному рівні.
- ЗК4. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- ЗК5. Здатність генерувати нові ідеї (креативність).
- ЗК6. Здатність виявляти, ставити та вирішувати проблеми.
- ЗК7. Здатність приймати обґрунтовані рішення.

### СК – фахових (спеціальних)

- СК1. Здатність до визначення технічних характеристик, конструктивних особливостей, застосування і експлуатації програмних, програмно-технічних засобів, комп'ютерних систем та мереж різного призначення.
- СК2. Здатність розробляти алгоритмічне та програмне забезпечення, компоненти комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем з використанням сучасних методів і мов програмування, а також засобів і систем автоматизації проектування.
- СК3. Здатність проектувати комп'ютерні системи та мережі з урахуванням цілей, обмежень, технічних, економічних та правових аспектів.

- СК4. Здатність будувати та досліджувати моделі комп'ютерних систем та мереж.
- СК5. Здатність будувати архітектуру та створювати системне і прикладне програмне забезпечення комп'ютерних систем та мереж.
- СК12. Здатність вирішувати завдання комп'ютерної інженерії з використанням апаратно-програмної обробки даних, засобів штучного інтелекту, хмарних технологій, Інтернету речей та комп'ютеризованих інформаційно-вимірювальних систем і комплексів.

*ПРН – програмних результатів навчання*

- РН1. Застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії.
- РН2. Знаходити необхідні дані, аналізувати та оцінювати їх.
- РН3. Будувати та досліджувати моделі комп'ютерних систем і мереж, оцінювати їх адекватність, визначати межі застосовності.
- РН4. Застосовувати спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері комп'ютерної інженерії, необхідні для професійної діяльності, оригінального мислення та проведення досліджень, критичного осмислення проблем інформаційних технологій та на межі галузей знань.
- РН5. Розробляти і реалізовувати проекти у сфері комп'ютерної інженерії та дотичні до неї міждисциплінарні проекти з урахуванням інженерних, соціальних, економічних, правових та інших аспектів.
- РН6. Аналізувати проблематику, ідентифікувати та формулювати конкретні проблеми, що потребують вирішення, обирати ефективні методи їх вирішення.
- РН8. Застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення складних задач комп'ютерної інженерії та дотичних проблем.
- РН9. Розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем.
- РН10. Здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії, аналізувати та оцінювати цю інформацію.
- РН11. Приймати ефективні рішення з питань розроблення, впровадження та експлуатації комп'ютерних систем і мереж, аналізувати альтернативи, оцінювати ризики та імовірні наслідки рішень.

### 3. Опис навчальної дисципліни

#### 3.1. Загальна інформація

Назва навчальної дисципліни <i>ОКЗ Технології проектування програмних систем</i>													
Форма навчання	Рік підготовки	Семестр	Кількість				Кількість годин						Вид підсумкового контролю
			кредитів	годин	змістових модулів	лекцій	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання		
Денна	1(5)	1(9)	3	90	1	15	-	-	15	60	-	іспит	
Заочна	1(5)	1(9)	3	90	1	4	-	-	4	82	-	іспит	

**Примітка.** Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить: для денної форми навчання – 0,5 (30/60);  
для заочної форми навчання – 0,1 (8/82).

#### 3.2. Дидактична карта 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
<b>Змістовий модуль 1. ТППС</b>												
Тема 1. Поняття архітектури ПЗ.	30	3		7		20	30	1		2		27
Тема 2. Шаблони проектування програмного забезпечення.	32	8		4		20	30	2		1		27
Тема 3. Тестування та відлагодження програмних систем.	28	4		4		20	30	1		1		28
Разом за змістовим модулем 1	90	15		15		60	90	4		4		82
<b>Усього годин</b>	90	15		15		60	90	4		4		82

### 3.2.1. Теми семінарських або практичних, або лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Аналіз архітектурного шаблону ПЗ	3
2	Розробка MVC-додатку	4
3	Шаблони (патерни) проектування ПЗ	4
4	Тестування ПЗ	4
	Разом	<b>15</b>

### 3.2.2. Тематика індивідуальних завдань

В даному курсі виконання індивідуальних завдань не передбачено.\*

\* ІНДЗ – може бути рекомендовано в окремих випадках для студентів, які успішно освоїли основний навчальний матеріал, з метою поглибленого вивчення чи удосконалення матеріалів певного змістового модуля, або в цілому для навчальної дисципліни за рішенням кафедри чи викладача.

### 3.2.3. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	CMF та CMS	10
2	Розробка HMVC-, MVVM-, NA-, MVP-, PAC-моделей програм	10
3	Антипаттерни	10
4	Породжуючі паттерни	10
5	Класифікація та проведення тестування	20
	Разом	<b>60</b>

### 3.3. Форми і методи навчання

**Форми навчання** – це проблемні й оглядові лекції, лабораторні заняття, заняття із застосуванням комп'ютерної та телекомунікаційної техніки, інтерактивні заняття з навчанням одних студентів іншими, інтегровані заняття, проблемні заняття, відеолекції, відеозаняття і відеоконференції засобами Google Meet, Zoom, Cisco Webex, заняття з використанням системи електронного навчання Moodle.

**Методи:** проблемний виклад матеріалу, частково-пошукові та дослідницькі лабораторні практикуми, презентації, кейс-стаді, консультації і дискусії, робота в інтернет-класі: електронні лекції, лабораторні роботи, дистанційні консультації та ін., спрямовані на активізацію і стимулювання навчально-пізнавальної діяльності студентів.

**Підходи до навчання:** використовуються студентоцентрикований, проблемно-орієнтований, діяльнісний, комунікативний, професійно-орієнтований, міждисциплінарний підходи.

**Реалізація навчального процесу** здійснюється під час лекційних, лабораторних занять, самостійної позааудиторної роботи з використанням сучасних інформаційних технологій навчання, консультацій з викладачами.

Для **формувань умінь та навичок** застосовуються такі **методи навчання:**

- вербальні/словесні (лекція, пояснення, розповідь, бесіда, інструктаж);
- наочні (спостереження, ілюстрація, демонстрація);
- практичні (проведення експерименту, практики);
- пояснювально-ілюстративний або інформаційно-рецептивний, який передбачає пред'явлення готової інформації викладачем та її засвоєння студентами;
- репродуктивний (виконання лабораторних завдань за зразком);
- метод проблемного викладу матеріалу на лекційних заняттях.

### 3.4. Технічне й програмне забезпечення/обладнання.

Комп'ютери в комп'ютерних класах №307, №311, №312, №313 8к. ЧНУ –кафедра КСМ.

Програмне забезпечення: особливих вимог до ПЗ немає. Лабораторні роботи можуть бути виконані на будь-якій операційній системі та в будь-якому середовищі розробки програм.

## 4. Система контролю та оцінювання

### 4.1. Розподіл максимально можливої кількості балів, які отримують студенти за виконання всіх видів навчальної діяльності

Підсумковий контроль				Сумарна кількість балів	
Змістовий модуль 1				іспит	
T1	T2	T3	M		
20	15	15	10	40	<b>100</b>

#### Змістовий модуль 1.

- T1. Поняття архітектури ПЗ (виконання лабораторних робіт №1-2 – 20 балів)  
 T2. Шаплони проектування ПЗ (виконання лабораторної роботи №3 – 15 балів)  
 T3. Тестування програмного додатку (виконання лабораторних робіт №4 – 15 балів)  
 M. Модульна контрольна робота – 10 балів

### 4.2. Шкала оцінювання: національна та ЄКТС

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

80 – 89	<b>B</b>	добре	
70 – 79	<b>C</b>		
60 – 69	<b>D</b>	задовільно	
50 – 59	<b>E</b>		
35 – 49	<b>FX</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0 – 34	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

### 4.3. Засоби оцінювання

Засобами оцінювання результатів навчання студента є: завдання для виконання лабораторних робіт, а також модульні контрольні роботи.

### 4.4. Форми поточного та підсумкового контролю

Формами поточного контролю рівня знань є усна та письмова відповідь студента при захисті виконаних лабораторних робіт, а також письмова відповідь при написанні модульних контрольних робіт.

Формами підсумкового контролю рівня знань є усна та письмова відповідь студента при здачі заліку.

### 4.5. Політика дисципліни.

Визначається системою вимог викладача щодо рівня знань і засвоєння матеріалу студентом при вивченні дисципліни, та ґрунтується на засадах академічної доброчесності з урахуванням норм законодавства України щодо академічної доброчесності та Статуту, положень Університету, й інших нормативних документів, які регламентують організацію освітнього процесу при вивченні дисципліни.

Вимоги стосуються заохочень і нарахування додаткових балів за активну участь у дискусіях щодо аналізу і обговорення тематичного матеріалу на лекціях і лабораторних заняттях, ґрунтовної підготовки до занять, відсутності пропусків без поважних причин, виявлення поглиблених знань під час захисту звітів з лабораторного практикуму і модульного контролю.

## 5. Рекомендована література

### 5.1. Базова (основна)

1. Двірничук К.В. Конспект лекцій з навчальної дисципліни «Технології проектування програмних систем» (електронне видання). – Чернівці, 2022. – 100с.
2. Будаї А. Дизайн-патерни – просто, як двері. – електронний ресурс, 2019. – 90 с.
3. Книга Head First. Патерни проектування / Ерік Фрімен, Елізабет Робсон, Берт Бейтс, Кеті Сієрра. – Фабула, 2020. – 672 с.

4. Бородкіна І. Інженерія програмного забезпечення. Посібник для студентів вищих навчальних закладів / І. Бородкіна, Г. Бородкін. – Центр навчальної літератури, 2018. – 204 с.
5. Грицюк Ю. Аналіз вимог до програмного забезпечення / Ю. Грицюк. – Львівська політехніка, 2018. – 456 с.
6. Мельник Н. Вступ до інженерії програмного забезпечення / Н. Мельник, Є. Левус – Львівська політехніка, 2018. – 248 с.
7. Мова програмування С# [електронний ресурс]. Режим доступу: <https://abitap.com/category/c/>
8. Патерни проектування [електронний ресурс]. Режим доступу: <https://abitap.com/category/paterny-proektuvannya/>.

### **5.2. Допоміжна**

1. Патерни проектування [електронний ресурс]. Режим доступу: <https://metanit.com/sharp/patterns/>.
2. Мова програмування С# [електронний ресурс]. Режим доступу: <https://metanit.com/sharp/tutorial/>.
3. Троелсен Е. Мова Програмування С# 6.0 і платформа .NET 4.6 / Е. Троелсен, Ф. Джемекс. – Книжка лавка, 2019. – 800 с.
4. Роберт С. Мартін. Чиста архітектура / Роберт С. Мартін. – Фабула, 2019. – 368 с.
5. Роберт С. Мартін. Чистий код / Роберт С. Мартін. – Фабула, 2019. – 416 с.
6. Мельник Р.А. Програмування веб-застосувань (фронт-енд та бек-енд) / Р.А. Мельник. – Львівська політехніка, 2018. – 248 с.

### **6. Інформаційні ресурси**

1. <https://csn.chnu.edu.ua/about-us/ok-rivni/>
2. <https://csn.chnu.edu.ua/spetsialnist-123-komp-yuterna-inzheneriya-opp-komp-yuterna-inzheneriya-magistratura-1-5-r/>