

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

Кафедра комп'ютерних систем, мереж і кібербезпеки (№ 503)

**ЗАТВЕРДЖУЮ**

Голова НМК



(підпис)

Д.М. Крицький

(ініціали та прізвище)

« 31 » серпня 2022р.

**РОБОЧА ПРОГРАМА ОБОВ'ЯЗКОВОЇ  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Технології проектування програмних систем

(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»

(шифр і найменування галузі знань)

Спеціальність: 123 «Комп'ютерна інженерія»

(код та найменування спеціальності)

Освітня програма: Комп'ютерні системи та мережі

Освітня програма: Програмовні мобільні системи та Інтернет речей

Освітня програма: Системне програмування

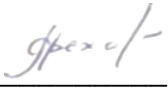
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: другий (магістерський)

**Харків 2022 рік**

Розробник: Орехов О.О., професор каф.503, к.т.н, доцент  
(прізвище та ініціали, посада, науковий ступінь та вчене звання)


  
(підпис)

Робочу програму розглянуто на засіданні кафедри комп'ютерних систем, мереж і кібербезпеки

(назва кафедри)

Протокол № 1 від «30» серпня 2022 р.

Завідувач кафедри д.т.н., професор  
(науковий ступінь та вчене звання)

  
(підпис)

В.С. Харченко  
(ініціали та прізвище)

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – 4	<p style="text-align: center;"><b>Галузь знань</b> <u>12 «Інформаційні технології»</u> (шифр та найменування)</p> <p style="text-align: center;"><b>Спеціальність</b> <u>123 «Комп'ютерна інженерія»</u> (код та найменування)</p> <p style="text-align: center;"><b>Освітня програма</b> <u>«Комп'ютерні системи та мережі»</u></p> <p style="text-align: center;"><u>«Програмовні мобільні системи та Інтернет речей»</u></p> <p style="text-align: center;"><u>«Системне програмування»</u> (найменування освітньої програми)</p> <p style="text-align: center;"><b>Рівень вищої освіти:</b> другий (магістерський)</p>	Обов'язкова
Кількість модулів – 2		<b>Навчальний рік</b>
Кількість змістових модулів – 2		2022/ 2023
Індивідуальне завдання <u>розрахункова робота</u> <b>Колективне проектування ПЗ за допомогою гнучких методів</b> (назва)		<b>Семестр</b>
Загальна кількість годин – денна –48/120		2-й
Кількість тижневих годин для денної форми навчання: аудиторних – 3 самостійної роботи здобувача – 3		<b>Лекції<sup>1)</sup></b>
		32 годин
		<b>Практичні, семінарські<sup>1)</sup></b>
		0 годин
		<b>Лабораторні<sup>1)</sup></b>
	16 годин	
	<b>Самостійна робота</b>	
72 годин		
<b>Вид контролю</b>	іспит	

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: –48/72;

<sup>1)</sup> Аудиторне навантаження може бути зменшене або збільшене на одну годину в залежності від розкладу занять.

## 2. Мета та завдання навчальної дисципліни

**1. Мета вивчення:** отримання здобувачами, необхідних компетенцій для командної роботи над створення програмних систем різного призначення, а також полягає у оволодінні методологією, методами та сучасним інструментарієм проектування програмних систем для ефективного виконання дипломного проекту в галузі комп'ютерної інженерії.

**2. Завдання:** вивчення принципів гнучких процесів керування проектами, командної розробки програмних систем, архітектурних патернів проектування та практичні навички розробки інтерфейсів, а також оволодіння сучасними гнучкими методами та засобами проектування програмних систем.

### 3. Компетентності, які набуваються:

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.

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.

СК6. Здатність використовувати та впроваджувати нові технології, включаючи технології розумних, мобільних, зелених і безпечних обчислень, брати участь в модернізації та реконструкції комп'ютерних систем та мереж, різноманітних вбудованих і розподілених додатків, зокрема з метою підвищення їх ефективності.

СК8 Здатність забезпечувати якість продуктів і сервісів інформаційних технологій на протязі їх життєвого циклу.

СК9 Здатність представляти результати власних досліджень та/або розробок у вигляді презентацій, науково-технічних звітів, статей і доповідей на науково-технічних конференціях.

СК10 Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних систем, мереж та їхніх компонентів;

#### **. 4. Очікувані результати 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.

РН7. Вирішувати задачі аналізу та синтезу комп'ютерних систем та мереж.

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 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.

РН12 Вільно спілкуватись усно і письмово українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською) при обговоренні професійних питань, досліджень та інновацій в галузі інформаційних технологій.

РН13 Зрозуміло і недвозначно доносити власні знання, висновки та аргументацію з питань інформаційних технологій і дотичних міжгалузевих питань до фахівців і нефахівців, зокрема до осіб, які навчаються.

**5. Пререквізити** - дисципліна є вибіркоvim компонентом освітньої програми і базується на знаннях, отриманих під час вивчення дисциплін у циклі вибіркових компонент ОП, передбачених навчальним планом спеціальності.

**Кореквізити** – «Науково-педагогічне стажування».

### **3. Зміст навчальної дисципліни**

#### **Модуль 1.**

##### **Змістовний модуль 1. Методи розробкою програмного забезпечення.**

###### **Тема 1. Вступ до навчальної дисципліни.**

Предмет, мета вивчення і задачі дисципліни. Структура та зміст дисципліни і методичні рекомендації щодо її вивчення. Місце дисципліни у навчальному процесі (зв'язок даного курсу з іншими дисциплінами). Вимоги до знань та вмінь тих, хто навчається. Характеристика рекомендованих під час вивчення дисципліни джерел інформації. Стислий екскурс в історію розробки програмного забезпечення.

###### **Тема 2. Гнучкі методи розробки програмного забезпечення.**

Поняття технології проектування ПЗ. Основні терміни технології проектування ПО. Складові частини програмної інженерії (продукт, персонал, процес, проект). Компетенції програмної інженерії. Організації-розробники стандартів з програмної інженерії. Гнучкі процеси розробки ПЗ. Принципи гнучкої розробки. Різновиди гнучких підходів. Екстремальне програмування. Гнучкий процес SCRUM.

###### **Тема 3. Модель проектної групи.**

Введення в методологію MSF. Основні концепції методології. MSF for Agile Software Development. Формування команди. Модель проектної групи. Основні принципи побудови команди. Рольові групи і ролі. Зони відповідальності рольових груп. Завдання рольових груп і взаємодія із зацікавленими особами. Формування команди. Рекомендації щодо можливого об'єднання ролей.

###### **Тема 4. Управління ризиками та модель процесів.**

Управління ризиками. Дисципліна управління ризиками MSF. Основні відомості про ризики. Планування управління ризиками. Процес управління ризиками. Управління ризиками як складова частина життєвого циклу проекту. Модель процесів. Принципи моделі процесів. Управління компромісами. Схема процесу розробки.

###### **Тема 5. Вироблення концепції і планування.**

Фаза вироблення концепції. Бачення. Рамки. Основні завдання фази. Виявлення та аналіз бізнес-вимог. Завдання рольових груп на фазі вироблення

концепції. Віхи фази вироблення концепції. Результати фази вироблення концепції. Планування проекту. Фаза планування. Основні завдання фази. Календарний графік проекту. Завдання рольових груп на фазі планування. Віхи фази планування. Результати фази планування.

### **Тема 6. Розробка, стабілізація та впровадження.**

Розробка рішення. Фаза розробки. Основні завдання фази. Завдання рольових груп на фазі розробки. Віхи фази розробки. Результати фази розробки. Стабілізація рішення. Фаза стабілізації. Основні завдання фази. Завдання рольових груп на фазі стабілізації. Результати фази стабілізації. Впровадження рішення. Фаза впровадження. Завдання рольових груп на фазі впровадження. Віхи фази впровадження. Результати фази впровадження.

### **Тема 7. Керування проектами програмних систем.**

Поняття управління проектом. Складові управління проектом. Основні параметри. Схема процесу управління проектом. Виявлення та зменшення ризиків.

### **Тема 8. Планування проекту.**

Створення структури поопераційного передня робіт. Методи створення структури проекту. Визначення стадій проекту. Ідентифікація завдань і дій.

### **Тема 9. Мережеве планування.**

Мережеве планування та управління (мережевий аналіз). Визначення мережевого планування. Можливості мережевого планування. Етапи мережевого планування. Структурне планування. Календарне планування. Оперативне планування.

### **Тема 10. Розрахунок мережевого графіка.**

Мережевий графік. Функції мережевого графіка. Робота. Види робіт. Синтаксис робіт. Подія. Шлях. Типи шляхів. Критичний шлях. Розробка мережевого графіка. Етапи складання графіка. Визначення передня робіт. Оцінка тривалості операцій. Виявлення залежностей робіт. Ресурси проекту. Типи мережевих графіків. Правила побудови мережевих графіків.

### **Модульний контроль**

## **Змістовний модуль 2. Патерни проектування**

### **Тема 1. Архітектурні системне патерни.**

Архітектура програми. Цілі вибору архітектур. Типи архітектур і їх моделі. Класифікація архітектур. Архітектури, засновані на потоках даних. Репозиторні архітектури. Рівневі архітектури. Процедура вибору архітектури.

### **Тема 2. Виробничі шаблони.**

Класифікація породжують патернів. Патерн Abstract Factory (Абстрактна фабрика). Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація.

### **Тема 3. Шаблони поведінки.**

Класифікація поведінкових шаблонів. Патерн Chain of Responsibility (Ланцюжок обов'язків). Призначення. Структура патерну на мові UML. Учасники. Відносини між учасниками. Мотивація. Застосування патерну. Результати. Реалізація.

### **Тема 4. Структурні шаблони.**

Класифікація структурних шаблонів. Патерн Adapter (Адаптер). Призначення. Структура патерну мовою UML. Учасники. Взаємини між учасниками. Мотивація. Застосовність патерну. Результати. Реалізація.

### **Тема 5. Принципи проектування інтерфейсів користувача.**

Концепції проектування інтерфейсів. Основні поняття. Класифікація інтерфейсів. Якість інтерфейсу. Моделі якості призначених для користувача інтерфейсів. Стандарти юзабіліті ISO.

### **Тема 6. Процес проектування.**

Підходи до проектування інтерфейсів. Сценарії та історії використання. Оцінка якості інтерфейсів. Класифікація методів оцінки юзабіліті. Метод на основі метричного підходу. Метод "Думки вголос". Евристична оцінка. Фокус групи. Контрольний список інтерфейсу. Макетування. Кількісний аналіз інтерфейсів. Тестування інтерфейсу.

### **Тема 7. Введення в мову XAML.**

Мета і призначення мови XAML. Синтаксис XAML. Базові правила XAML.



Мови опису інтерфейсів. Мова розмітки графічного інтерфейсу XAML.  
Інструментальні засоби проектування інтерфейсів.

### **Тема 8. Мова XAML.**

Еліпс. Окружність. Еліпс з мінімумом атрибутів. Лінія. Товщина лінії за замовчуванням. Ламана лінія. Багатокутник. Вертикальний порядок послідовності об'єктів.

### **Тема 9. Технології проектування.**

Технологія проектування інтерфейсів. Технологія WPF і Silverlight. Модель програмування та подання. Архітектура.

## **Модульний контроль**

### **4. Структура навчальної дисципліни**

Назви змістових модулів і тем	Кількість годин				
	Усього о	У тому числі			
		л	п	лаб.	с. р.
1	2	3	4	5	6
<b>Модуль 1</b>					
<b>Змістовий модуль 1. Методи розробки програмного забезпечення</b>					
Тема 1. Вступ до навчальної дисципліни	2	1		1	
Тема 2. Гнучкі методи розробки програмного забезпечення	2	1			1
Тема 3. Модель проектної групи	2	1			1
Тема 4. Управління ризиками та модель процесів.	2	1			1
Тема 5. Вироблення концепції і планування	2	1			1
Тема 6. Розробка, стабілізація та впровадження	3	2			1

Тема 7. Керування проектами програмних систем	2	1			1
Тема 8.Планування проекту	5	2		1	2
Тема 9. Мережеве планування	4	2			2
Тема 10. Розрахунок мережевого графіка.	5	2		1	2
Модульний контроль	1	1			
Разом за змістовим модулем 1	<b>30</b>	<b>15</b>		<b>3</b>	<b>12</b>
<b>Змістовний модуль 2. Патерни проектування</b>					
Тема 1. Архітектурні системне патерни	5	2		1	2
Тема 2. Виробничі шаблони.	6	2		2	2
Тема 3. Шаблони поведінки	4	2			2
Тема 4.Структурні шаблони	4	2			2
Тема 5. Принципи проектування інтерфейсів користувача	6	2		2	2
Тема 6. Процес проектування	6	2		2	2
Тема 7. Введення в мову XAML	5	1		2	2
Тема 8.Мова XAML	5	1		2	2
Тема 9.Технології проектування	6	2		2	2
Модульний контроль	1	1			
Індивідуальне завдання	42	-	-	-	42
Разом за змістовим модулем 2	<b>90</b>	<b>17</b>		<b>13</b>	<b>60</b>
<b>Усього годин</b>	<b>120</b>	<b>32</b>		<b>16</b>	<b>72</b>

### 5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
1		
	<b>Разом</b>	

### 6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1		
	<b>Разом</b>	

### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Практична програма курсу	1
2	Створення найпростіших тривимірних фігур в WPF	3
3	Побудова поверхонь	3
4	Робота з текстурою тривимірного об'єкту	3
5	Анімація тривимірних об'єктів	6
	<b>Разом</b>	<b>16</b>

### 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Розробка специфікації проекту	4
2	Розробка архітектури проекту	4
3	Розробка плану проекту	4
4	Розробка плану тестування	4
5	Вибір та використання системи керування програмними проектами	4
6	Вибір та використання систем контролю версій	4
7	Вибір та використання bug tracking систем	3

8	Вибір та використання інструментальних засобів дизайнера	3
9	Виконання розрахунково-графічної роботи	42
	<b>Разом</b>	<b>72</b>

## 9. Індивідуальні завдання

Розрахунково-графічна робота на тему Колективне проектування ПЗ за допомогою гнучких методів (42 год)

## 10. Методи навчання

Проведення аудиторних лекцій, лабораторних занять, консультацій, а також самостійна робота здобувачів за матеріалами (п.14, 15).

## 11. Методи контролю

Проведення поточного контролю, письмового модульного контролю, підсумковий контроль у вигляді іспиту.

## 12. Критерії оцінювання та розподіл балів, які отримують здобувачі

### 12.1 Розподіл балів, які отримують здобувачі (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
<b>Змістовий модуль 1</b>			
Робота на лекціях	0,5...1	10	5...10
Виконання і захист лабораторних робіт	3...5	3	9...15
Модульний контроль	15...25	1	15...25
<b>Змістовий модуль 2</b>			
Робота на лекціях	0,5...1	9	5...9
Виконання і захист лабораторних робіт	3...5	2	6...10
Модульний контроль	15...25	1	15...25
Виконання і захист розрахункової роботи	5...6	1	5...6
<b>Усього за семестр</b>			<b>60...100</b>

Семестровий контроль у вигляді іспиту проводиться у разі відмови здобувача від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту здобувач має можливість отримати максимум 100 балів.

Білет для іспиту складається з двох теоретичного та одного практичного запитань, максимальна кількість за кожне із запитань, складає 25 балів за теоретичний та 50 за практичний запит.

### Критерії оцінювання роботи здобувача протягом семестру

**Задовільно (60-74).** Показати мінімум знань та умінь. Захистити не менше 80% від усіх завдань лабораторних занять. Уміти використовувати сучасний інструментарій гнучких процесів розробки ПЗ та знати компетенції програмної інженерії. Уміти розробляти архітектуру програмної системи з використанням шаблонів проектування. Уміти планувати та управлять проектом у вигляді мережевого графіка, а також уміти розробляти найпростіший інтерфейс користувача на мові XAML.

**Добре (75-89).** Твердо знати мінімум, захистити не менше 90% завдань лабораторних занять. Уміти використовувати сучасний інструментарій гнучких процесів розробки ПЗ та знати компетенції програмної інженерії. Уміти розробляти архітектуру програмної системи з використанням шаблонів проектування. Уміти обґрунтувати і застосовувати шаблони проектування. Уміти планувати та керувати проектом у вигляді мережевого графіка та виконувати розрахунок мережевого графіка, а також уміти розробляти 3D інтерфейс користувача на мові XAML з використанням інструментальних засобів.

**Відмінно (90-100).** Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти їх застосовувати.

### 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

Навчально-методичний комплекс дисципліни розміщений у системі дистанційного навчання «Ментор».

Сторінка дисципліни у системі дистанційного навчання «Ментор» [Ел. ресурс].  
URL: <https://mentor.khai.edu/enrol/index.php?id=1645>

## 14. Рекомендована література

### Базова

1. Ушакова І.О. Методика управління вимогами в гнучких методологіях / І.О. Ушакова. // Збірник наукових праць ХНУПС. – 2018.– Вип. 2(56). – С. 93 – 98.
2. 2.Р.А. Мельник. Програмування веб- застосунків (фронт-енд та бек-енд). Навчальний посібник. - Львів : Видавництво Львівської політехніки, - 2018. - 248 с.
3. Прикладне програмування у комп'ютерних мережах : навчальний посібник / О. Д. Азаров, О. І. Черняк., Л. А. Савицька – Вінниця : ВНТУ, 2016. – 130 с
4. Методи системного аналізу в радіоелектроніці та комп'ютерній інженерії: підручник / А.В. Горбенко, С.Ю. Даншина, В.А.Краснобаєв, О.О. Орехов и др.; за ред.. С.Ю. Даншиної, В.С. Харченка. – Х.: Нац. аерокосм. ун-т «Харк. авіац. ін-т», 2014. – 424 с.

### Допоміжна

1. ISO/IEC/IEEE 12207:2021. System and software engineering - Software life cycle processes: ISO/IEC/IEEE, 2021. – 122p.
2. ISO/IEC 25010:2011. Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models: ISO/IEC, 2010. – 34p.
3. ECSS-E-ST-10C. Space engineering – System engineering general requirements: ESA Requirements and Standards Division, 2009. – 100 p.
4. ECSS-Q-ST-80C. Product assurance. Software product assurance: ESA Requirements and Standards Division, 2009. – 113p.

## 15. Інформаційні ресурси

1. Iterative and incremental development. [Ел. ресурс]. URL: [http://en.wikipedia.org/wiki/Iterative\\_and\\_incremental\\_development](http://en.wikipedia.org/wiki/Iterative_and_incremental_development)
2. Iterative\_and\_incremental\_development.Spiral model. [Ел. ресурс]. URL: [http://en.wikipedia.org/wiki/Spiral\\_model](http://en.wikipedia.org/wiki/Spiral_model)
3. Introduction to the Microsoft Solutions Framework. [Ел. ресурс]. URL: <http://technet.microsoft.com/en-us/library/bb497060.aspx>.
4. Agile manifesto. [Ел. ресурс]. URL: <http://agilemanifesto.org>.
5. Agile software development. [Ел. ресурс]. URL: [http://en.wikipedia.org/wiki/Agile\\_software\\_development](http://en.wikipedia.org/wiki/Agile_software_development)