

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

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

ЗАТВЕРДЖУЮ

Гарант освітньої програми


(підпис) Ольга МОРОЗОВА
(ім'я та ПРІЗВИЩЕ)

« 29 » _____ серпня _____ 2025 р.

**СИЛАБУС ОБОВ'ЯЗКОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Технології проектування програмних систем
(назва навчальної дисципліни)

Галузь знань: F «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: F7 «Комп'ютерна інженерія»
(код і найменування спеціальності)

Освітньо-професійна програма: «Системне програмування»
(найменування освітньої програми)

Рівень вищої освіти: другий (магістерський)

Силабус введено в дію з 01.09.2025

Харків – 2025 р.

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


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри _____
комп'ютерних систем, мереж і кібербезпеки
(назва кафедри)

Протокол № 1 від « 29 » 08 2025 р.

Завідувач кафедри _____
(науковий ступінь і вчене звання)


(підпис)

Вячеслав ХАРЧЕНКО
(ім'я та ПРІЗВИЩЕ)

Погоджено з представником здобувачів освіти:


(підпис)

Дмитро ВАСИК
(ім'я та ПРІЗВИЩЕ)

1. Загальна інформація про викладача



ПІБ: Орехов Олександр Олександрович

Посада: Професор

Науковий ступінь: Кандидат технічних наук

Вчене звання: Доцент

Перелік дисциплін, які викладає:

Технології проектування програмних систем
Технології забезпечення якості програмно-технічних комплексів

Напрями наукових досліджень:

Моделювання та оцінювання якості розроблення програмного забезпечення для критичних галузей.

Контактна інформація:

a.orehov@csn.khai.edu

2. Опис навчальної дисципліни

Форма здобуття освіти	Денна
Семестр	2
Мова викладання	Українська
Тип дисципліни	Обов'язкова
Обсяг дисципліни: кредити ЄКТС/ кількість годин	4 кредити ЄКТС / 120 годин (48 аудиторних, з яких: лекції – 32, лабораторні – 16; СРЗ – 72)
Види навчальної діяльності	Лекції, лабораторні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	Дисципліна є обов'язковим компонентом освітньої програми і базується на знаннях, отриманих під час вивчення дисциплін у циклі загальної і професійної підготовки, передбачених навчальним планом спеціальності

3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – отримання студентами, 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.

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.

Компетентності, які набуваються:

Загальні компетентності (ЗК):

- 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.

Спеціальні компетентності (СК):

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.

Програмні результати навчання (ПРН):

ПРН1. Застосовувати загальні підходи пізнання, методи математики, природничих та інженерних наук до розв'язання складних задач комп'ютерної інженерії.

ПРН2. Знаходити необхідні дані, аналізувати та оцінювати їх.

ПРН3. Будувати та досліджувати моделі комп'ютерних систем і мереж, оцінювати їх адекватність, визначати межі застосовності.

ПРН4. 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.

ПРН5. Розробляти і реалізовувати проекти у сфері комп'ютерної інженерії та дотичні до неї міждисциплінарні проекти з урахуванням інженерних, соціальних, економічних, правових та інших аспектів.

ПРН6. Аналізувати проблематику, ідентифікувати та формулювати конкретні проблеми, що потребують вирішення, обрати ефективні методи їх вирішення.

ПРН7. Вирішувати задачі аналізу та синтезу комп'ютерних систем та мереж.

ПРН8. 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.

ПРН9. Розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем.

ПРН11. Приймати ефективні рішення з питань розроблення, впровадження та експлуатації комп'ютерних систем і мереж, аналізувати альтернативи, оцінювати ризики та імовірні наслідки рішень.

ПРН12. 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.

ПРН13. 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.

4. Зміст навчальної дисципліни

МОДУЛЬ 1

Змістовний модуль 1. Методи розробки програмного забезпечення

Тема 1. Вступ

Стисла анотація: 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.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 2. Гнучкі методи розробки програмного забезпечення.

Стисла анотація: 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.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 3. Модель проектної групи.

Стисла анотація: 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.

РГР: Сформувати проектну групу

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 4. Управління ризиками та модель процесів.

Стисла анотація: 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.

РГР: Вибрати та описати процес розробки. Описати можливі ризики.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 5. Вироблення концепції і планування.

Стисла анотація: 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.

РГР: Сформулювати концепцію проекту.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 6. Розробка, стабілізація та впровадження.

Стисла анотація: 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.

РГР: Розробити рішення, виконати стабілізацію, запровадити першу версію рішення.

Самостійна робота: Опрацювання матеріалу лекцій

Тема 7. Керування проектами програмних систем.

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

РГР: Вибрати систему управління проектами.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 8. Планування проекту.

Стисла анотація: Створення структури поопераційного переліку робіт. Методи створення структури проекту. Визначення стадій проекту. Ідентифікація завдань і дій.

Самостійна робота: Опрацювання матеріалу лекцій

РГР: Створення структури поопераційного переліку робіт.

Тема 9. Мережеве планування.

Стисла анотація: Мережеве планування та управління (мережевий аналіз). Визначення мережевого планування. Можливості мережевого

планування. Етапи мережевого планування. Структурне планування. Календарне планування. Оперативне планування.

РГР: Розробити календарний графік виконання проєкту.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 10. Розрахунок мережевого графіка.

Стисла анотація: 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 проєкту. Типи мережевих графіків. Правила побудови мережевих графіків.

РГР: Виконати оптимізацію плану робіт.

Самостійна робота: Опрацювання матеріалу лекцій.

Модульний контроль

Змістовний модуль 2. Патерни проектування

Тема 1. Архітектурні системне патерни.

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

РГР: Розробити системну архітектуру проєкту.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 2. Виробничі шаблони.

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

РГР: Обґрунтувати можливість використання патернів, що породжують.

Самостійна робота: : Опрацювання матеріалу лекцій.

Тема 3. Шаблони поведінки.

Стисла анотація: 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.

РГР: Обґрунтувати можливість використання патерну поведінки.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 4. Структурні шаблони.

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

РГР: Обґрунтувати можливість використання структурного патерну.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 5. Принципи проектування інтерфейсів користувача

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

РГР: Розробити прототип інтерфейсу.

Самостійна робота: : Опрацювання матеріалу лекцій.

Тема 6. Процес проектування.

Стисла анотація: 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.

РГР: Спроекувати інтерфейс.

Самостійна робота: Опрацювання матеріалу лекцій.

Тема 7. Введення в мову XAML.

Стисла анотація: Мета і призначення мови XAML. Синтаксис XAML. Базові правила XAML. Мови опису інтерфейсів. Мова розмітки графічного інтерфейсу XAML. Інструментальні засоби проектування інтерфейсів.

Самостійна робота: : Опрацювання матеріалу лекцій.

Тема 8. Мова XAML.

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

РГР: Зібрати версію проекту.

Самостійна робота: : Опрацювання матеріалу лекцій.

Тема 9. Технології проектування.

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

РГР: Захист проекту.

Самостійна робота: Опрацювання матеріалу лекцій.

Модульний контроль

5. Індивідуальні завдання

Розрахунково-графічна робота на тему Колективне проектування ПЗ за допомогою гнучких методів

6. Методи навчання

Проведення аудиторних лекцій, лабораторних робіт, консультацій, а також самостійна робота здобувачів з використанням відповідних матеріалів (п.11, 12).

7. Методи контролю

Проведення поточного контролю, підсумковий контроль у вигляді іспиту.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовий модуль 1			
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
Змістовий модуль 2			
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
Усього за семестр			60...100

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 студент 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.

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача освіти протягом семестру

Задовільно (60-74) – Показати мінімум знань та умінь. Захистити не менше 75% від усіх завдань лабораторних занять. Знати основні складові індустріального Інтернету речей. Уміти створювати та налаштовувати конфігурацію складових індустріального Інтернету речей.

Добре (75-89) – Твердо знати мінімум, захистити не менше 90% завдань лабораторних занять. Знати ключові принципи індустріального Інтернету речей та типові архітектури. Уміти розробляти програмне забезпечення для складових IoT з використанням спеціалізованих мов програмування, мов програмування високого рівня або low-code мов.

Відмінно (90-100) – Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти їх застосовувати. Уміти налаштовувати обмін даними між складовими IoT-систем через MQTT та OPC UA. Вміти використовувати бази даних часових рядів. Уміти створювати дашборди для візуалізації технологічної інформації. Уміти ефективно використовувати джерела даних. Знати вимоги до налаштування безпеки складових індустріального Інтернету речей.

9. Політика навчального курсу

Відвідування занять. Інтерактивний характер курсу передбачає обов'язкове відвідування лабораторних занять. Здобувачі освіти, які за певних обставин не можуть відвідувати лабораторні заняття регулярно, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після їх пропуску. Відпрацювання занять здійснюється усно у формі співбесіди за питаннями, визначеними планом заняття. В окремих випадках дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених

Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>). Очікується, що роботи здобувачів освіти будуть їх оригінальними дослідженнями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu/ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

10. 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>

11. 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.Р.А. Мельник. Програмування веб- застосувань (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 «Харьк. авиац. ин-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. ДСТУ ISO/IEC 25010:2011. Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) -System and software quality models: ISO/IEC, 2016. – 34p.
3. ECSS-E-ST-10C. REF1 REF2 Space engineering – System engineering general requirements: ESA Requirements and Standards Division, 2017. 2019 – 100 p.
4. ECSS-Q-ST-80C. REF1 Product assurance. Software product assurance: ESA Requirements and Standards Division, 2017. – 113p.

12. Інформаційні ресурси

1. Iterative and incremental development. [Ел. ресурс]. Режим доступу: <http://en.wikipedia.org/wiki/>
2. Iterative_and_incremental_development.Spiral model. [Ел. ресурс]. Режим доступу: http://en.wikipedia.org/wiki/Spiral_model
3. Introduction to the Microsoft Solutions Framework. [Ел. ресурс]. Режим доступу: <http://technet.microsoft.com/en-us/library/bb497060.aspx>.
4. Agile manifesto. [Ел. ресурс]. Режим доступу: <http://agilemanifesto.org>.
5. Agile software development. [Ел. ресурс]. Режим доступу: <http://en.wikipedia.org/wiki/>