

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

Кафедра «Інженерії програмного забезпечення» (№ 603)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



(підпис)

Ілона ШЕВЧЕНКО

(ім'я та прізвище)

« 29 » 08 2025 р.

СИЛАБУС ОBOB'ЯЗKОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Основи програмної інженерії

(назва навчальної дисципліни)

Галузь знань: F Інформаційні технології

(шифр і найменування галузі знань)

Спеціальність: F2 Інженерія програмного забезпечення

(код та найменування спеціальності)

Освітня програма: Інженерія програмного забезпечення

(найменування освітньої програми)

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2025 року

Харків – 2025 р.

Розробник: _____Ігор ТУРКІН, д.т.н., проф.
(прізвище та ініціали, посада, науковий ступінь та вчене звання)



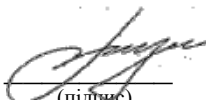
(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри інженерії програмного забезпечення

(назва кафедри)

Протокол № 1 від «29» серпня 2025 р.

Завідувач кафедри _____д-р техн. наук., проф.
(науковий ступінь і вчене звання)



(підпис)

І.Б. Туркін
(ініціали та прізвище)

Погоджено з представником здобувачів освіти:
Представник студентського самоврядування _____



(підпис)

Дикун Д.В.
(ініціали та прізвище)

1. Загальна інформація про викладача



ПІБ: Туркін Ігор Борисович

Посада: завідувач кафедри інженерії програмного забезпечення

Науковий ступінь: доктор технічних наук

Вчене звання: професор.

Перелік дисциплін, які викладає:

- Основи програмної інженерії
- Екосистеми програмного забезпечення
- Системи реального часу
- НДР магістра.

Напрями наукових досліджень: інженерія програмного забезпечення, екосистеми програмного забезпечення та цифрові платформи, програмне забезпечення бортових мікроконтролерів супутників та БПЛА.

Контактна інформація:

E-mail: i.turkin@khai.edu

Signal: IhorTurkin.89

2. Опис навчальної дисципліни

Форма здобуття освіти	<i>Денна</i>
Семестр	1 семестр
Мова викладання	Українська
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС/ кількість годин	<u>денна</u> : 5 кредитів ЄКТС / 150 годин (64 аудиторних, з яких: лекції – 32, практичні – 32; СРЗ – 86)
Види навчальної діяльності	Лекції, практичні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	<i>немає</i>
Кореквізити	Комп'ютерна дискретна математика, Основи програмування
Постреквізити	UI/UX дизайн, Аналіз вимог до програмного забезпечення, Економіка ІТ-проектів, Менеджмент ІТ-проектів, Тестування та верифікація програмного забезпечення

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

Мета - засвоєння загальних знань щодо:

- процесів створення програмного забезпечення (ПЗ),
- інженерії програмного забезпечення,
- організації навчального процесу в ХАІ зі спеціальності
- структури навчальної програми і місця кожної з дисциплін, що вивчаються, у загальній схемі навчання;
- можливостей працевлаштування після закінчення навчання.

Завдання - сформувати професійний світогляд майбутнього фахівця з інженерії програмного забезпечення і цілісного уявлення про його сутність і роль у сучасному суспільстві.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.

Загальні компетентності (ЗК):

Після закінчення цієї програми здобувач освіти буде здатен:

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК02. Здатність застосовувати знання у практичних ситуаціях.

ЗК03. Здатність спілкуватися державною мовою як усно, так і письмово.

ЗК04. Здатність спілкуватися іноземною мовою як усно, так і письмово

Фахові компетентності:

Після закінчення цієї програми здобувач освіти буде здатен:

ФК04. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

ФК05. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

ФК11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.

Програмні результати навчання:

ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН02. Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.

ПРН03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПРН04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПРН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПРН08. Вміти розробляти людино-машинний інтерфейс.

ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПРН16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.

ПРН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

ПРН20. Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.

ПРН22. Знати та вміти застосовувати методи та засоби управління проектами.

ПРН23. Вміти документувати та презентувати результати розробки програмного забезпечення.

4. Зміст навчальної дисципліни

Змістовий модуль 1. Інформаційні технології та інженерія програмного забезпечення. Стандарти та стандартизація. Професійна підготовка фахівців програмної інженерії

Тема 1. Вступ. Місце та значення курсу для інженерів-програмістів

Анотація: Місце та значення курсу. Загальні поняття та терміни. Історія. Програмні продукти, алгоритмічні мови Plancalcul, FORTRAN, Basic, Simula. Інженерія програмного забезпечення, життєвий цикл програмного продукту, кризи програмування, модульне, структурне, об'єктно-орієнтоване програмування. Сучасні та перспективні технології: інтерфейси, криптовалюта, Інтернет речей, доповнена реальність, гейміфікація.

Тема лекції 1: Вступ. Місце та значення курсу для інженерів-програмістів. Критерії оцінювання успішності студентів. Механічний етап розвитку обчислювачів.

Тема лекції 2: ЕОМ, мікроелектроніка, персональні комп'ютери, хмарні обчислення, інформаційно-комунікаційні технології.

Тема лекції 3: Історія розвитку програмного забезпечення.

Тема лекції 4: Сьогодення та майбутнє інженерії програмного забезпечення.

Тема практичного завдання 1: Оформлення технічної документації в текстовому процесорі Word.

Тема практичного завдання 2: Аналіз на релевантність результатів запиту пошукових ресурсів.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування питань до викладача.

Тема 2. Професійна підготовка фахівців з програмної інженерії

Анотація: Основні стандарти програмної інженерії. Організації - автори стандартів. Промислові стандарти. Project Management Body of Knowledge - Звід знань з управління проектами. IEEE SWEBOOK - Software Engineering Body of Knowledge (Звід знань з програмної інженерії): галузі знань програмної інженерії. Академічні освітні стандарти в області комп'ютерних наук. ACM / IEEE Computing Curricula. Стандарти Євросоюзу: «European Qualifications Framework for Lifelong Learning (EQF)», Engineering Competency Model. Компетентнісна модель. European e-Competence Framework 3.0 (Європейські рамкові вимоги до компетенцій фахівців у сфері IT). CEPIS (Council of European Professional Informatics Societies) e-Competence Benchmark - <http://www.cepis.org> , Національні кваліфікаційні вимоги: Стандарти вищої освіти України. Професійні та етичні вимоги. Кодекс етики. Академічна доброчесність. Відповідальне використання ІІІ.

Тема лекції 1: Стандартизація, стандарти, сертифікація на відповідність стандартам.

Тема лекції 2: Стандарти життєвого циклу програмного забезпечення.

Тема лекції 3: Академічні освітні стандарти в області комп'ютерних наук.

Тема лекції 4: Професійні та етичні вимоги. Кодекс етики. Академічна доброчесність. Відповідальне використання ІІІ.

Тема практичного завдання 1: Реферат з історії інженерії програмного забезпечення.

Тема практичного завдання 2: Використання Microsoft PowerPoint для розробки презентацій.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування питань до викладача.

Модульний контроль 1

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).

Змістовий модуль 2. Етапи та моделі життєвого циклу програмного забезпечення. Управління програмним проектом

Тема 1. Етапи та моделі життєвого циклу програмного забезпечення.

Анотація: Методи програмної інженерії - визначення. CASE-засоби та їх класифікація. Еволюція CASE. Властивості хорошої програми. Основні труднощі універсального методу. Вимоги до програмного забезпечення. Три завдання - 3 рівня вимог. Джерела вимог. Методи збору і поліпшення вимог. П'ять вимог до написання вимог. Атрибути хороших вимог. Моделі представлення архітектури програмного забезпечення. Статична структурна модель, динамічна модель процесів. Проектування інтерфейсу користувача. Стили взаємодії користувача з комп'ютером. Альтернативні представлення даних. Приклади повідомлень про помилки. Ітераційність процесу проектування інтерфейсу користувача. Принципи проектування: взаємодія з користувачем, подання інформації, використання кольорів, засоби підтримки користувача, повідомлення про помилки, проектування довідкової системи, документація користувача. Вибір мови реалізації. Проблемно (предметно) – орієнтовані мови програмування. Приклади хороших і поганих імен в програмі. Угорська нотація. Правила хорошого оформлення коду. Класифікація помилок. Налагодження. Помилки в ПЗ: defect, failure, fault, error. Об'єкти та рівні тестування. Система менеджменту багів (Bug Tracking System). Підхід «розробки від тестування» (test-driven development). Рівні інтеграційного тестування. Підходи до інтеграційного тестування. Підходи до системного тестування. Середовище тестування. Автоматизоване тестування програмного забезпечення (Software Automation Testing). Головний закон контролю якості ПЗ. Супровід (maintenance, sustaining). Причини необхідності супроводу програм. Типи супроводу. Фактори, що впливають на витрати при супроводі. Реінженіринг.

Тема лекції 1: Модель, метод, CASE. CASE-засоби і їх класифікація. Еволюція CASE. Властивості хорошої програми.

Тема лекції 2: Інженерія вимог.

Тема лекції 3: Проектування.

Тема лекції 4: Кодування та налагодження, тестування, верифікація, валідація, супровід.

Тема лекції 5: Класичні моделі життєвого циклу програмного продукту.

Тема лекції 6: Моделі життєвого циклу MSF, RUP, XP, SCRUM (промислові технології розробки ПЗ).

Тема практичного завдання 1: Git Hub репозиторій. CodeSpace.

Тема практичного завдання 2: Використання онлайн сервісу Draw.io для документування розроблених програм.

Тема практичного завдання 3: Візуалізація бізнес-процесів з використанням нотації IDEF0.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування питань до викладача.

Тема 2. Управління програмним проектом

Анотація: Процеси ЖЦ ПЗ: користувач - постачальник, інженерні процеси, допоміжні процеси, процеси управління, організаційні процеси. Організація командної роботи. Вимоги до керівника програмного проекту. PMBOK: 9 областей знань про управління. SQI: 34 компетенції ІТ менеджера. Управління командою проекту. Моделі організації команд. Спілкування в команді. Корпоративна політика. Планування проекту: види планів, процес створення плану, структура плану, контрольні відмітки етапів робіт. Метрики проекту. Графік робіт: Загальні характеристики, процес складення графіку робіт, часові та мережні діаграми. Управління ризиками: типи ризиків, ризики, що можливі, процес управління ризиками, визначення, аналіз, планування та моніторинг ризиків. Інструментальні засоби керування проектом.

Тема лекції 1: Терміни та визначення: управління, проект, управління проектами.

Тема лекції 2: Планування та контроль проекту

Тема практичного завдання 1: Побудова діаграми Ганта.

Самостійна робота здобувачів: опрацювання матеріалу лекцій, підготовка до практичних занять, проходження тестування за результатами роботи на практичних заняттях, формування питань до викладача.

Модульний контроль 2

Форма занять: написання модульної роботи в аудиторії (за рішенням лектора допускається проведення у дистанційній формі).

5. Індивідуальні завдання

Не передбачено навчальним планом

6. Методи навчання

Лекції з елементами інтерактиву (пояснення з використанням презентацій, прикладів, міні-опитувань). *Практичні заняття* – розробка документації. *Проектно-орієнтоване навчання* – виконання невеликих практичних проєктів, спрямованих на закріплення знань. *Робота в малих групах* – колективний аналіз програмних фрагментів, обговорення рішень. *Використання системи онлайн-тестування.* *Самостійна робота* – індивідуальні завдання, робота з електронними матеріалами та онлайн-курсами. *Консультації* – індивідуальні та групові (очно або онлайн) для підтримки та корекції навчального процесу.

7. Методи контролю

Поточний контроль: опитування на практичних заняттях; розв’язування алгоритмічних задач та аналіз програмних фрагментів; виконання письмових контрольних робіт з окремих розділів курсу; програмований контроль (тестування, онлайн-тести); оцінювання виконання індивідуальних і групових практичних завдань.

Модульний контроль: складання модульного контролю;

Підсумковий контроль: іспит.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Тестування після лекції	0..1	8	0..8
Виконання і захист практичних робіт	2...5	4	8...20

Модульний контроль (тестування)	0...22	1	0...22
Змістовний модуль 2			
Тестування після лекції	0..1	8	0..8
Виконання і захист практичних робіт	2...5	4	8...20
Модульний контроль (тестування)	0...22	1	0...22
Усього за семестр			60...100

Семестровий контроль (іспит) проводиться у разі відмови здобувача від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту здобувач має можливість повторно пройти контроль знань та покращити свою оцінку з теорії. Тест складається з 30 питань та виконується з обмеженням часу – 30 хвилин.

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 протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Мати уявлення про інформаційні технології (ІТ) та інформаційно-комунікаційні технології (ІКТ), основні етапи розвитку ІКТ та історію розвитку програмного забезпечення, стандартизацію, стандарти та сертифікація на відповідність стандартам. Мати уявлення про типи і види стандартів. Знати основні стандарти програмної інженерії та організації - авторів стандартів. Розуміти особливості життєвого циклу програмного забезпечення. Мати уявлення про основні моделі життєвого циклу. Мати уявлення про основні академічні освітні стандарти в області комп'ютерних наук та Національні кваліфікаційні вимоги: Стандарти вищої освіти України. Розуміти основні принципи інженерії вимог, проектування, тестування, верифікації, валідації та супровіду ПЗ. Мати уявлення про промислові технології розробки ПЗ. Мати уявлення про основи управління програмним проектом.

Добре (75-89). Твердо знати мінімум, здати тестування та поза аудиторну самостійну роботу. Розуміти визначення інформаційних технологій (ІТ) та інформаційно-комунікаційних технологій (ІКТ). Знати основні етапи розвитку ІКТ та історію розвитку програмного забезпечення, стандартизацію, стандарти та сертифікація на відповідність стандартам. Розуміти типи і види стандартів.

Знати основні стандарти програмної інженерії та організації - авторів стандартів. Розуміти особливості життєвого циклу програмного забезпечення. Розуміти основні моделі життєвого циклу. Мати уявлення про основні академічні освітні стандарти в області комп'ютерних наук та Національні кваліфікаційні вимоги: Стандарти вищої освіти України. Розуміти основні принципи інженерії вимог, проектування, тестування, верифікації, валідації та супроводу ПЗ. Знати основи промислових технологій розробки ПЗ. Знати основи управління програмним проектом

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

9. Політика навчального курсу

Процедура відпрацювання пропущених занять: здобувач самостійно ознайомлюється з пропущеним матеріалом: лекційними конспектами, навчальними презентаціями, записами занять або додатковими матеріалами, наданими викладачем. Пропущене практичне заняття відпрацьовується шляхом виконання всіх завдань, передбачених для цього. За потреби здобувач може узгодити індивідуальну консультацію з викладачем для роз'яснення складних тем або перевірки виконаних завдань. Після виконання завдань здобувач надає результати викладачу для перевірки у форматі звіту про виконану роботу. Виконане заняття оцінюється за тими ж критеріями, що і основне заняття. Відпрацювання вважається успішним після схвалення викладачем результатів та підтвердження засвоєння матеріалу.

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 розроблені здобувачами програми та звіти до них будуть оригінальними. 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 незалежно від масштабів плагіату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила етичної поведінки 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 та інформаційні ресурси

Підручники, навчальні посібники, навчально-методичні посібники, конспекти лекцій, методичні рекомендації з проведення практичних робіт тощо, які видані в Університеті знаходяться за посиланням:

1. Дистанційний курс дисципліни розроблено у системі дистанційного навчання Mentor, яку впроваджено в Національному аерокосмічному університеті ім. М.Є. Жуковського «ХАІ», доступ до курсу за посиланням: <https://mentor.khai.edu/course/view.php?id=205>

11. Рекомендована література

Базова

1. Трофименко О. Г., Манаков С. Ю., Ларін Д. Г. Основи програмної інженерії : навч.-метод. посібник [Електронне видання] / О. Г. Трофименко, С. Ю. Манаков, Д. Г. Ларін. – Одеса : Фенікс, 2022. – 197 с.

2. Основи програмної інженерії: навчальний посібник / Є. О. Зайцев – К.: КНТЕУ, 2017. – 423 с.

3. Інженерія програмного забезпечення: Посібник для студентів вищих навчальних закладів / І. Л. Бородкіна, Г. О. Бородкін ; М-во освіти і науки України, Національний університет біоресурсів та природокористування України. – Київ: , 2018. – 251 с.

4. Петрик М.Р. Моделювання програмного забезпечення: науково-методичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.

5. Вступ до інженерії програмного забезпечення / Н. Мельник, Є. Левус. – Видавництво «Львівська Політехніка», 2018. – 248 р.

6. Аналіз вимог до програмного забезпечення / Ю. Грицюк. – Видавництво «Львівська Політехніка», 2018. – 458 с.

7. Лавріщева К.М. Програмна Інженерія. – К. – 2008. – 319 с.

Допоміжна

1. SWEBOOK V3.0. The Guide to the Software Engineering Body of Knowledge. IEEE Computer Society Professional Practices Committee (Керівництво з областей знань програмної інженерії). – Tokio: OI'musha, 2014. – 335 с.

2. Бондаренко М., Сидоров М., Морозова Т., Мендзєбровський І. Модель випускника бакалаврату " Програмна інженерія " (3 досвіду роботи науково – методичної підкомісії 050103) / М. Бондаренко, М. Сидоров, Т. Морозова, І. Мендзєбровський// Вища школа. — 2009. — No 4. — С. 50—61.

3. Janarthanam S. Hands-On Chatbots and Conversational UI Development / Srinu Janarthanam. – Birmingham-Mumbai: Packt Publishing, 2017. – 373 p.

12. Інформаційні ресурси

1. Software Engineering Institute, <https://www.sei.cmu.edu/>
2. Спілка програмістів. <https://dou.ua/>
3. Стандарти вищої освіти України бакалавра та магістра з інженерії програмного забезпечення, <https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/12/21/121-inzheneriya-programnogo-zabezpechennya-bakalavr.pdf>, <https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/2020/11/17/121-inzheneriya-programnogo-zabezpechennya-mahistr.pdf>