



Інженерія програмного забезпечення. Курсова робота

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерні системи та мережі</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Заочна</i>
Рік підготовки, семестр	<i>2 курс, весняний семестр</i>
Обсяг дисципліни	<i>1 кредит (30 годин – СРС)</i>
Семестровий контроль/ контрольні заходи	<i>Залік/ захист курсової роботи</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>старший викладач, Васильєва Марія Давидівна, vasylieva.mariia@iit.kpi.ua</i>
Розміщення курсу	<i>На платформі дистанційного навчання «Сікорський»: https://classroom.google.com/c/NzcwMzQzNzQ5OTYy?cjc=ah5rdqlh</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою виконання курсової роботи з інженерії програмного забезпечення є закріплення теоретичних знань та набуття практичних навичок у проектуванні, розробці, тестуванні та документуванні програмних систем. В рамках роботи студенти повинні продемонструвати вміння застосовувати сучасні методи та інструменти розробки програмного забезпечення, вирішувати інженерні завдання, враховувати вимоги до якості, масштабованості та надійності програмних продуктів, а також ефективно організовувати роботу в команді або індивідуально. Виконання курсової роботи також передбачає розвиток аналітичних навичок та здатності до системного мислення, що є важливими складовими професійної діяльності в сфері ІТ.

Предметом виконання курсової роботи з інженерії програмного забезпечення процеси, методи та технології, що застосовуються на різних етапах життєвого циклу програмного забезпечення. Це включає аналіз вимог, проектування архітектури, розробку коду, тестування, інтеграцію, впровадження та супровід програмних систем. Крім того, досліджуються питання забезпечення якості, управління конфігурацією, автоматизації процесів розробки, а також

застосування шаблонів проектування, методологій управління проектами та інших практик, спрямованих на підвищення ефективності й надійності програмного забезпечення.

Вивчення дисципліни забезпечує наступні загальні та фахові компетенції:

- ЗК03. Здатність застосовувати знання у практичних ситуаціях.
- ЗК07. Вміння виявляти, ставити та вирішувати проблеми.
- ФК02. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.
- ФК03. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.
- ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науковотехнічних звітів.
- ФК12. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних та кіберфізичних систем, мереж та їхніх компонентів шляхом використання аналітичних методів і методів моделювання.
- ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.
- ФК15. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.
- ФК18. Здатність розробляти, адаптувати, використовувати програмно-технічні засоби для збирання, оброблення та інтелектуального аналізу даних в комп'ютерних системах: паралельних, розподілених, хмарних, безпечних, інтелектуальних, розумних, Інтернет речей, зокрема, з використанням штучного інтелекту.

У відповідності до вищезазначеного, студенти отримають наступні результати навчання:

- ПРН02. Мати навички проведення експериментів, збирання даних та моделювання в комп'ютерних системах.
- ПРН07. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.
- ПРН08. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.
- ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.
- ПРН11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.
- ПРН12. Вміти ефективно працювати як індивідуально, так і у складі команди.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

При вивченні дисципліни «Інженерія програмного забезпечення» доцільно використовувати знання, отримані при вивченні попередніх дисциплін:

- Програмування. Частина 1
- Програмування. Частина 2
- Інженерія програмного забезпечення
- Організація баз даних

Дисципліна є базовою для курсів:

- Комп'ютерні системи
- Паралельні та розподілені обчислення
- Організація баз даних

3. Зміст навчальної дисципліни

Курсова робота виконується за індивідуальним завданням і готується до захисту в завершальний період теоретичного навчання. Курсова робота повинна бути підготовлена до захисту в термін, обумовлений у завданні та погодженому з викладачем. До захисту курсової роботи представляється пояснювальна записка у складі:

Титульний лист

Зміст

Вступ

Розділи:

1.1 Розділ 1. Аналітичний огляд (теоретична частина).

1.2 Розділ 2. Проєктування системи.

1.3 Розділ 3. Реалізація програмного продукту.

1.4 Розділ 4. Тестування та оцінка якості.

Висновки

Список використаних джерел

Додатки

Під час написання курсової роботи студенти дотримуються наступних вимог:

1. Програма має бути розроблена з урахуванням основних правил ООП, composition over inheritance, loose coupling – high cohesion, inversion of control (IoC).

2. Створюється специфікація вимог програмного забезпечення. Вимоги повинні описувати функціональність програмного продукту.

3. Слід використовувати паттерни проєктування та обґрунтувати доцільність їх використання в коді.

4. Використовувати принципи SOLID при проєктуванні додатку.

5. Написати модульні тести для бізнес-логіки.

3.1 Етапи виконання курсової роботи

Основні етапи виконання курсової роботи:

1. Отримання теми та завдання

2. Підбор та вивчення літератури

3. Формування технічного завдання

4. Огляд методів колективної розробки програмного забезпечення

5. Розробка прецедентів розроблюваної програмного додатка

6. Розробка програмного додатка

7. Тестування програмного додатку

8. Оформлення пояснювальної записки

9. Подання курсового проєкту (роботи) на перевірку

10. Захист курсового проєкту (роботи)

4. Навчальні ресурси та матеріали

4.1. Базова література

1. Методичні вказівки до виконання лабораторних робіт з дисципліни «інженерія програмного забезпечення» [Електронний ресурс] : Методичні вказівки. для студ. Спеціальності 123

«Комп'ютерна інженерія» /А.І. Антонюк, А.О. Болдак; КПІ ім. Ігоря Сікорського. - Київ : КПІ ім. Ігоря Сікорського, 2022. - 51 с.

2. Інженерія програмного забезпечення: Посібник для студентів вищих навчальних закладів/І. Л. Бородкіна, Г. О. Бородкін. – К.: Центр учбової літератури, 2020. – 204 с. ISBN 978-611-01-1232-1

3. С.В.Баран. Розробка програмного забезпечення з використанням патернів проектування: Навчальний посібник. – Кривий Ріг: Державний університет економіки і технологій, 2023. – 203 с.

4. Чиста архітектура: Мистецтво створення програмного забезпечення. Видання друге / пер. з англ. І.Бондар-Терещенко. – Харків. Вид-во «Ранок»: Фабула, 2023. – 368 с. ISBN 978-617-09-5286-8

4.2. Додаткова література

1. Head First. Патерни проектування / Ерік Фрімен, Елізабет Робсон, Кеті Сьєрра і Берт Бейтс; пер. з англ. Г. Якубовська — Харків : ВД «Фабула», 2020. — 672 с. ISBN 978-617-09-6159-4

2. Design Patterns: elements of reusable object-oriented software / Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Indianapolis: - Addison-Wesley, 1994. -417 p. ISBN: 0201633612.

3. Head First. Java. Легкий для сприйняття довідник / Кетті Сьєрра, Берт Бейтс. - Харків: ВД «Фабула», 2022. – 720 с. ISBN 978-617-522-033-7

Інженерія програмного забезпечення: навч. посібник / Д. П. Кучеров, Є. Б. Артамонов. – К. : НАУ, 2017. – 388 с.

4. Mark Grand. Patterns in Java: A Catalog of Reusable Design Patterns Illustrated with UML, Volume 1, 2nd Edition. Wiley Publishing, 2002. 480 p. ISBN: 0471258393

5. Theo Mandel. The Elements of User Interface Design. - John Wiley & Sons, 2005. - 468 p. ISBN: 0471162671

6. Bruce Eckel. Thinking in Java, 4th ed. – Printice hall, 2006. – 1057 p. ISBN: 0131872486

7. Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.

8. Бандура В.В., Храбатин Р.І. Архітектура та проектування програмного забезпечення: конспект лекцій. — Івано-Франківськ: ІФНТУНГ, 2012. — 240 с.

9. Проектування інформаційних систем: Загальні питання теорії проектування ІС (конспект лекцій): навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. – Київ : КПІ ім. Ігоря Сікорського, 2020. – 192с.

10. Поморова О.В., Говорущенко Т.О. Проектування інтерфейсів користувача: навч.посібник - Хмельницький: ХНУ, 2011. - 206 с.

11. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с.

12. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія» / Дегтярьова Л.М., Гроза П.М., Сомов С.В. – Полтава: ПолтНТУ, 2017. – 218 с.

4.3. Інформаційні ресурси

1. Дистанційні курси на платформі дистанційного навчання «Сікорський» в середовищі Google Workspace for Education: Інженерія програмного забезпечення.

<https://classroom.google.com/c/NzcwMzQzMTcxMjQy?cjc=mnkvrxf>

2. <https://se-111.blogspot.com/>

3. <https://refactoring.guru/uk>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компоненту)

Таблиця 1

<i>Тиждень семестру</i>	<i>Назва етапу роботи</i>
3	Отримання теми та завдання
4-6	Підбір та вивчення літератури
7-8	Розробка прецедентів розроблюваної програмного додатка
9	Огляд архітектурного шаблону MVC
10-13	Розробка програмного додатка
14	Тестування програмного додатка
15	Оформлення пояснювальної записки
16	Подання курсової роботи на перевірку
17	Захист курсової роботи

6. Самостійна робота здобувача вищої освіти

<i>№ з/п</i>	<i>Назва теми, що виноситься на самостійне опрацювання</i>	<i>Кількість годин СРС</i>
1	Отримання теми та завдання	1
2	Підбір та вивчення літератури	5
3	Розробка прецедентів розроблюваної програмного додатка	4
4	Огляд архітектурного шаблону MVC	2
5	Розробка програмного додатка	12
6	Тестування програмного додатка	3
7	Оформлення пояснювальної записки	2
8	Захист курсової роботи	1
	Разом:	30

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Оформлення курсової роботи має відповідати вимогам до звітів про НДР (ДСТУ 3008-2015 «Державний стандарт України. Документація. Звіти в сфері науки і техніки. Структура і правила оформлення»).

Пояснювальна записка оформляється на листах формату А4 в текстовому редакторі WORD. Текст пояснювальної записки має бути набраний 14 шрифтом Times New Roman через 1,5 міжрядкових інтервали з вирівнюванням по ширині та мати наступні розміри полів: ліве – 25мм, праве – 10мм, верхнє – 20мм, нижнє – 20мм, абзац – 15мм.

Кожен розділ пояснювальної записки повинен починатися з нової сторінки. Заголовки розділів нумеруються арабськими цифрами. Між попереднім текстом і заголовком, а також між заголовком і наступним текстом потрібно вставити пустий рядок. Між заголовком і підзаголовком пропуску рядка немає. Після заголовка на сторінці має бути хоча б один рядок тексту, інакше заголовок слід перенести на наступний аркуш.

Титульний аркуш вважається першою сторінкою роботи, однак номер на ньому не проставляється. Аналогічним чином сторінка змісту формально включається до нумерації, але номер на ній також не ставиться. Видима нумерація починається зі сторінки після змісту, тобто зі вступу (сторінка 3, якщо зміст вміщується на одну сторінку). Таким чином перші сторінки такі як, титульний аркуш, завдання, анотація, перелік умовних позначень, зміст не нумерують. Номер сторінки розміщується у правому нижньому куті, арабською цифрою, без крапки в кінці.

Етапи курсової роботи повинні бути виконані згідно встановленого календарного графіку робіт.

Розроблений програмний додаток повинен бути протестований, результати тестування програмного додатку наводяться у тексті основної частини курсової роботи. Покриття тестами має становити не менше 80%.

У випадку виявлення факту академічної недобросовісності курсова робота не зараховується та перескладається з новою темою.

Невчасне виконання курсової роботи тягне за собою зниження отриманих за нього балів на 10%, якщо запізнення не більше двох тижнів, на 20% якщо запізнення більше двох тижнів.

При оцінюванні курсової роботи беруться до уваги наступні чинники:

- повнота виконання індивідуального завдання на курсову роботу;
- коректність розроблених прецедентів;
- своєчасність виконання курсової роботи згідно графіку;
- самостійність виконання курсової роботи та відсутність ознак плагіату;
- відповіді на питання щодо змісту курсової роботи під час її захисту.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Система оцінювання успішності студентів з дисципліни «Інженерії програмного забезпечення. Курсова робота» ґрунтується на «Положенні про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського» (https://osvita.kpi.ua/sites/default/files/downloads/Pologennia_RSO_2022.pdf), а саме Рейтингової системі оцінювання (PCO) другого типу (PCO-2).

PCO-2 курсової роботи (R_K) складається з двох складових:

- стартової (R_C);
- складова захисту (R_Z).

$$R_K = R_C + R_Z$$

Перша (стартова) складова характеризує роботу студента з курсової роботи та її результат – якість пояснювальної записки та розробленого програмного додатку. Друга складова характеризує якість захисту студентом курсової роботи.

Розмір шкали першої складової дорівнює **80 балам**, а другої складової - **20 балам**.

№ етапу	Складові роботи	Максимальна кількість балів при вчасному виконанні	Урахування вчасності виконання
1	Оформлення звіту	10	100% від оцінки при дотриманні графіку роботи
2	Огляд архітектурного шаблону MVC	5	
3	Розробка прецедентів програмного додатку	10	
4	Розробка програмного додатку	30	90% у разі затримки до 2 тижнів
5	Написання юніт тестів з відсотком покриття не менше 80%	10	

6	Виконання ручного тестування додатку. Перевірка функціональних вимог до програмної системи	10	80% у разі затримки більше 2 тижнів
7	Наявність та оформлення висновків та списку джерел	5	
8	Захист курсової роботи	20	
	Усього	100	

Студент допускається до захисту курсової роботи за умови стартової складової R_c не менше **48 балів**.

Якість захисту

Критерії оцінювання виступу з доповіддю за матеріалами КР та відповідей на питання:

- ступінь володіння теоретичним матеріалом до 10 балів;
- ступінь володіння методами розробки програмного додатку в цілому до 10 балів.

Після завершення захисту курсової роботи визначається R_k , яка у подальшому переводиться у оцінку за університетською шкалою згідно таблиці:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Нормативні положення політики навчальної дисципліни (освітнього компонента)

Політика використання штучного інтелекту. Використання штучного інтелекту регламентується «Політикою використання штучного інтелекту для академічної діяльності в КПІ ім. Ігоря Сікорського» (<https://osvita.kpi.ua/node/1225>). Усі індивідуальні завдання, мають бути результатом власної оригінальної роботи здобувача. Використання ШІ для автоматичної генерації відповідей без подальшого їх аналізу та доопрацювання заборонено.

Порушення академічної доброчесності: використання плагіату. У разі виявлення фактів використання плагіату під час виконання навчальних завдань, студент втрачає право на подальше виконання поточних завдань у межах навчальної дисципліни до з'ясування обставин порушення.

Студент зобов'язаний аргументовано пояснити причини використання плагіату, а також засвідчити готовність дотримуватись принципів академічної доброчесності в подальшому навчанні; повторно виконати завдання або, за потреби, виконати додаткове завдання відповідно до бачення викладача з урахуванням вимог щодо самостійності.

Зазначені дії регламентуються Кодексом честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (<https://kpi.ua/code>).

Робочу програму навчальної дисципліни (силабус):

Складено, старший викладач кафедри ОТ Васильєва Марія Давидівна.

Ухвалено кафедрою обчислювальної техніки (протокол No 12 від 23.06.2025).

Погоджено методичною комісією ФІОТ (протокол No 11 від 27.06.2025).