



# КОМПОНЕНТИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ. КУРСОВА РОБОТА

## Робоча програма навчальної дисципліни (Силабус)

### Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Назва
Статус дисципліни	Нормативна
Форма навчання	заочна
Рік підготовки, семестр	3 курс, осінній семестр
Обсяг дисципліни	30 годин / 1 кредит ECTS, з яких 30 годин становить самостійна робота
Семестровий контроль/ контрольні заходи	Захист курсової роботи/залік
Розклад занять	<a href="http://rozklad.kpi.ua/">http://rozklad.kpi.ua/</a>
Мова викладання	Українська
Інформація про керівника курсу / викладачів	доцент, к.т.н., Варава Іван Андрійович, <a href="mailto:varava_ivan@iit.kpi.ua">varava_ivan@iit.kpi.ua</a> , тел. 098-432-29-33 (у робочий час)
Розміщення курсу	Кампус

### Програма навчальної дисципліни

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

Силабус освітнього компонента «Компоненти програмної інженерії. Курсова робота» складено відповідно до освітньої програми підготовки бакалаврів «Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці» спеціальності 121 «Інженерія програмного забезпечення».

Кредитний модуль «Компоненти програмної інженерії. Курсова робота» є завершальним етапом вивчення циклу дисциплін «Компоненти програмної інженерії». В ході виконання курсової роботи студент узагальнює і поглиблює засвоєні знання щодо життєвого циклу розробки програмного забезпечення, починаючи із аналізу вимог та завершуючи тестуванням програмного забезпечення.

**Метою** дисципліни «Якість та тестування програмного забезпечення. Курсова робота» є закріплення у студентів наступних компетенцій:

- Здатність застосовувати знання у практичних ситуаціях (ЗК 2);
- Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення (ФК1);
- Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування (ФК2);
- Здатність розробляти архітектури, модулі та компоненти програмних систем (ФК3);

–Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами (ФК 4);

–Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу (ФК 5);

– Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних(ФК 7);

–Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення (ФК 8);

–Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення тестування і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя (ФК 10);

–Здатність реалізувати фази та ітерації життєвого циклу програмних система інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення (ФК 11);

–Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення (ФК 12);

–Здатність обґрунтовано обирати та освоювати інструментарій з розробки тестування та супроводження програмного забезпечення (ФК 13);

**Предмет дисципліни** – сучасні методики та інструментальні засоби оцінки якості програмного забезпечення та його тестування.

**Програмні результати навчання**, на формування та покращення яких спрямована дисципліна: (ПРН1) Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки; (ПРН 3) Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення; (ПРН 4) Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення; (ПРН5)Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення; (ПРН 6) Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення; (ПРН 7) Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення; (ПРН 8) Вміти розробляти людино-машинний інтерфейс; (ПРН 9) Знати та вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення; (ПРН 10) Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування; (ПРН11) Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання; (ПРН 12) Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення; (ПРН 13) Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань. (ПРН 14) Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення; (ПРН15) Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення; (ПРН 16) Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації; (ПРН 17) Вміти застосовувати методи компонентної розробки програмного забезпечення; (ПРН 18) Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних; (ПРН 19) Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення; (ПРН 20) Знати підходи щодо оцінки та забезпечення якості програмного забезпечення; (ПРН23) Вміти документувати та презентувати результати розробки програмного забезпечення.

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

**Пререквізити дисципліни.** Знання, отримані при вивченні дисциплін: «Бази даних», «Проектування інформаційних систем», «Основи web-програмування», «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення», «Компоненти програмної інженерії. Частина 3. Архітектура програмного забезпечення».

**Постреквізити дисципліни.** Отримані знання при вивченні дисципліни використовуються для виконання дипломної роботи та безпосередньої професійної діяльності.

## 3. Зміст навчальної дисципліни

Етап	Найменування та зміст	Результат	Години
1	<i>Підготовчий етап</i> Вибір теми КР, затвердження ТЗ	Затверджене ТЗ	2
2	<i>Етап виконання</i> Аналіз вимог до ПЗ, проектування і розробка ПЗ, тестування ПЗ.	Оформлена тестова документація	23
3	<i>Заключний етап</i> Оформлення пояснювальної записки	Пояснювальна записка до КР	4,5
4	<i>Захист КР</i>		0,5

*Структура пояснювальної записки до курсової роботи:*

*Вступна частина:*

- титульний аркуш;
- завдання на курсову роботу;
- календарний план-графік виконання роботи;
- анотація українською мовою;
- зміст;
- перелік скорочень, умовних позначень, термінів; – вступ.

*Основна частина:*

- аналіз вимог до програмного забезпечення;
- проект ПЗ
- тест-план;
- тест-кейси;
- модульні тести;
- інтеграційні тести;
- тестування навантаження.

*Додатки.*

## 4. Навчальні матеріали та ресурси

### Основна література:

1. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник / За ред. Крепич С.Я., Співак І.Я. / для бакалаврів галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення». – Тернопіль: ФОП Паляниця В.А., 2020. – 478с.
2. Svyatoslav Kulikov Software Testing. Base Course. 3rd ed. EPAM Systems, 2015–2023. – 278p.
3. Myers, Glenford J The art of software testing / Glenford J. Myers, Corey Sandler, Tom Badgett. — 3rd ed. – Wiley, 2013. – 256 p.

4. Roy Osherove The Art of Unit Testing with examples in C#, 2th ed. – Manning, 2013. – 292 p.
5. James Whittaker How Google Tests Software / James Whittaker, Jason Arbon, Jeff Carollo – Addison-Wesley, 2012. – 320 p.

#### **Допоміжна література:**

6. Програмна інженерія. Якість продукту. Частина 1. Модель якості (ISO/IEC 9126-1:2001, IDT): ДСТУ ISO/IEC 9126-1:2013 – [Чинний від 2014-07-01]. – К.: МІНЕКОНОМПРОЗВИТКУ України, 2014. – 20 с. – (Національний стандарт України).
7. Програмна інженерія. Якість продукту. Частина 2. Зовнішні метрики (ISO/IEC TR 9126-2:2003, IDT): ДСТУ ISO/IEC TR 9126-2:2008 – [Чинний від 2010-07-01]. – К.: Держспоживстандарт України, 2011. – 85 с. – (Національний стандарт України).
8. Програмна інженерія. Якість продукту. Частина 3. Внутрішні метрики (ISO/IEC TR 9126-3:2003, IDT): ДСТУ ISO/IEC TR 9126-3:2012 – [Чинний від 2013-05-01]. – К.: МІНЕКОНОМПРОЗВИТКУ України, 2013. – 46 с. – (Національний стандарт України).
9. Програмна інженерія. Якість продукту. Частина 4. Метрики якості під час використання (ISO/IEC TR 9126-4:2004, IDT): ДСТУ ISO/IEC TR 9126-4:2012 – [Чинний від 2013-05-01]. – К.: МІНЕКОНОМПРОЗВИТКУ України, 2013. – 49 с. – (Національний стандарт України).
10. Lisa Crispin, Janet Cregory More Agile Testing: Learning Journeys for the Whole Team // AddisonWesley Professional – 2014. – 532 p.
11. Humble, Jez. Continuous delivery : reliable software releases through build, test, and deployment automation / Jez Humble, David Farley // Addison-Wesley Professional; 1st edition. – 2010. – 484p.
12. Jonathan Rasmusson. The Way of the Web Tester. A Beginner’s Guide to Automating Tests / J. Rasmusson// Pragmatic Bookshelf. – 2016. – 258p.

### **Навчальний контент**

#### **5. Методика опанування навчальної дисципліни (освітнього компонента)**

Освітній компонент «Інженерія програмного забезпечення. Курсова робота» за порядком слідування та розподілом годин розміщується на третьому курсі в осінньому семестрі.

#### **Календарний план запланованих заходів на час проходження студентами освітнього компонента**

<b>Тиждень</b>	<b>Назва етапу роботи</b>
1 тиждень	Погодження теми на курсову роботу та формулювання основного завдання
2 тиждень	Опис проекту. Формулювання вимог.
3 тиждень	Тестування вимог до програмного забезпечення.
4-5 тиждень	Проектування програмного забезпечення. Розробка діаграми прецедентів, діаграми класів, схеми бази-даних
6-8 тиждень	Розробка програмного забезпечення. Вибір інструментальних засобів, фреймворків, бібліотек. Реалізація алгоритмів.
9 тиждень	Розробка тест-плану
10 тиждень	Розробка чек-листу та матриці відстеження вимог

11-12 тиждень	Функціональне тестування
13-14 тиждень	Модульне тестування
15-16 тиждень	Навантажувальне тестування.
17 тиждень	Оформлення записки до курсової роботи
18 тиждень	Захист курсової роботи

## Політика та контроль

### 6. Політика навчальної дисципліни (освітнього компонента)

Студент повинен виконувати КР протягом семестру, дотримуючись затвердженого календарного плану.

Усі етапи виконання КР студент має виконувати самостійно.

Курсова робота вважається виконаною, якщо студент надав текст пояснювальної записки до КР у вигляді файлу (та в друкованому вигляді) та захистив її перед комісією (продемонстрував працездатність ПЗ; привів результати тестування; відповів на усі питання).

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

#### 7.1 Поточний контроль

Поточний контроль виконання КР здійснюється шляхом перевірки готовності розділів КР згідно затвердженого графіку виконання.

#### 7.2 Система рейтингових балів та критерії оцінювання

Загальна оцінка з курсової роботи складається із двох компонентів:

1. Виконання курсової роботи – 80 балів.
2. Захист КР – 20 балів.

При оцінці виконання КР враховуються:

- Розробка технічного завдання – до 5 балів;
- Проектування системи – до 10 балів;
- Розробка програмного забезпечення – до 10 балів;
- Розробка тест-плану – до 10 балів;
- Розробка тест-кейсів – до 10 балів;
- Тестування вимог до програмного забезпечення – до 5 балів;
- Розробка програмного забезпечення – до 10 балів;
- Розробка модульних тестів – до 10 балів;
- Розробка інтеграційних тестів – до 10 балів;
- Проведення навантажувального тестування – до 10 балів.

При оцінці захисту КР беруть до уваги:

- якість доповіді – до 10 балів; –
- відповіді на питання – до 10 балів.

Сума балів за виконання КР та її захист переводиться до залікової оцінки згідно таблиці відповідності рейтингових балів оцінкам за університетською шкалою:

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно

**Робочу програму навчальної дисципліни (силабус) «Компоненти програмної інженерії. Курсова робота»:**

Складено доцентом кафедри ІПЗЕ, к.т.н., Варавою Іваном Андрійовичем

Ухвалено кафедрою ІПЗЕ (протокол № 34 від 10.05.2024 р.)

Погоджено Методичною комісією ННІАТЕ КПІ ім. Ігоря Сікорського (протокол №9 від 31.05.2024 р.)