



# КУРСОВА РОБОТА З ТЕХНОЛОГІЇ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Робоча програма навчальної дисципліни (Силабус)

### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>15 Автоматизація та приладобудування</i>
Спеціальність	<i>151 Автоматизація та комп'ютерно-інтегровані технології</i>
Освітня програма	<i>Комп'ютерно-інтегровані системи та технології в приладобудуванні</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>2 курс, осінній семестр</i>
Обсяг дисципліни	<i>1 кредит (30 годин)</i>
Семестровий контроль/ контрольні заходи	<i>Залік</i>
Розклад занять	<i>Згідно з розкладом на сайті <a href="http://rozklad.kpi.ua/">http://rozklad.kpi.ua/</a></i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Викладачі: <i>к.т.н., доцент Цибульник Сергій Олексійович, <a href="mailto:tsybulnik.s.a@gmail.com">tsybulnik.s.a@gmail.com</a> Паздрій Ольга Ярославівна Кондратенко Денис Юрієвич</i>
Розміщення курсу	<a href="https://do.ipu.kpi.ua/course/view.php?id=5004">https://do.ipu.kpi.ua/course/view.php?id=5004</a>

### Програма навчальної дисципліни

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

Предметом навчальної дисципліни «Курсова робота з технології розроблення програмного забезпечення» є вивчення теорії, методики та практики розроблення програмного забезпечення, а також методології аналізу вимог до розроблюваного програмного забезпечення, його тестування та документування.

У рамках навчальної дисципліни «Курсова робота з технології розроблення програмного забезпечення» майбутні бакалаври навчаються працювати в команді, визначати вимоги до розроблюваного програмного забезпечення, проводити його розробку, тестування та документування.

Метою дисципліни є формування у студентів навичок розроблення вимог до програмного забезпечення, проектування структури програмного забезпечення, використання сучасних комп'ютерно-інтегрованих та інструментальних засобів при розробці програмного забезпечення, а також оцінювання отриманих результатів.

Результати навчання, які мають продемонструвати студенти після засвоєння дисципліни:

### Компетентності:

- ЗК01. Здатність застосовувати знання у практичних ситуаціях.
- ЗК02. Здатність спілкуватися державною мовою як усно, так і письмово. ЗК03. Здатність спілкуватися іноземною мовою
- ЗК04. Навички використання інформаційних і комунікаційних технологій.
- ЗК05. Здатність до пошуку, опрацювання та аналізу інформації з різних джерел.
- ФК 5. Здатність обґрунтовувати вибір технічних засобів автоматизації на основі розуміння принципів їх роботи аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи автоматизації і експлуатаційних умов; налагоджувати технічні засоби автоматизації та системи керування.
- ФК 6. Здатність використовувати для вирішення професійних завдань новітні технології у галузі автоматизації та комп'ютерно-інтегрованих технологій, зокрема, проектування багаторівневих систем керування, збору даних та їх архівування для формування бази даних параметрів процесу та їх візуалізації за допомогою засобів людино-машинного інтерфейсу.
- ФК 7. Здатність обґрунтовувати вибір технічної структури та вміти розробляти прикладне програмне забезпечення для мікропроцесорних систем керування на базі локальних засобів автоматизації, промислових логічних контролерів та програмованих логічних матриць і сигнальних процесорів.
- ФК 9. Здатність вільно користуватись сучасними комп'ютерними та інформаційними технологіями для вирішення професійних завдань, програмувати та використовувати прикладні та спеціалізовані комп'ютерно-інтегровані середовища для вирішення задач автоматизації.

### Програмні результати навчання:

- ПРН 3. Вміти застосовувати сучасні інформаційні технології та мати навички розробляти алгоритми та комп'ютерні програми з використанням мов високого рівня та технологій об'єктно-орієнтованого програмування, створювати бази даних та використовувати інтернет-ресурси;
- ПРН 10. Вміти обґрунтовувати вибір структури та розробляти прикладне програмне забезпечення для мікропроцесорних систем управління на базі локальних засобів автоматизації, промислових логічних контролерів та програмованих логічних матриць і сигнальних процесорів.

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

Для опанування дисципліни «Курсова робота з технології розроблення програмного забезпечення» необхідно мати знання з наступних дисциплін: «Вища математика», «Програмування» та «Технології розроблення програмного забезпечення». Додатково можуть знадобитися знання з алгоритмізації та теорії оптимізації.

У подальшому знання та вміння, одержані при вивченні цієї дисципліни, використовуються у спеціальних і професійно-орієнтованих дисциплінах, курсовому і дипломному проектуванні.

## **3. Зміст навчальної дисципліни**

### **Розділ 1. Створення програмного продукту з графічним інтерфейсом**

#### **○ Тема 1. Планування проекту.**

Стратегії розроблення: каскадна, інкрементна, еволюційна. Моделі життєвого циклу. Створення групи. Організація роботи. Групова робота. Рішення задач. Мотивація. Аналіз

вимог до проекту. Нотації. Формалізація вимог. Створення макету. Формування технічного завдання. Документування.

○ **Тема 2.** Архітектурне проектування. Реалізація нефункціональних вимог. Архітектура. Архітектурне проектування системи. Підсистема. Модуль. Сервіс. Функціональні та нефункціональні вимоги до програмного забезпечення. Основні класи архітектур.

○ **Тема 3.** Реалізація функціональних вимог. Інкапсуляція. Поліморфізм. Спадковість. Принципи SOLID. Принцип єдиної відповідальності. Принцип відкритості-закритості. Принцип підстановки Барбара Лісков. Принцип розділення інтерфейсу. Принцип інверсії залежностей. Аналоги зазначених принципів у відомих шаблонах. Створення елементів графічного інтерфейсу. Контейнери. Елементи керування. Таблиці. Виникнення та обробка подій.

○ **Тема 4.** Тестування програмного забезпечення. Написання авто тестів. Юніт-тести. Інтеграційне тестування. Тестування графічного інтерфейсу користувача. Ручне тестування.

#### **4. Навчальні матеріали та ресурси**

##### Базова література:

1. Мартін Р. Чистий код. Харків, Фабула, 2019. 416 с.
2. Мартін Р. Чиста архітектура. Харків, Фабула, 2019. 368 с.
3. Мартін Р. Чистий Agile. Харків, Фабула, 2021. 224 с.
4. Цибульник, С. О. Технології розробки програмного забезпечення-1. Комп'ютерний практикум [Електронний ресурс]: навчальний посібник для студентів спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології», освітньо-професійної програми «Комп'ютерно-інтегровані технології та системи навігації і керування» / С. О. Цибульник ; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2021. – 125 с.
5. Технології програмування та створення програмних продуктів: конспект лекцій /укладач О. В. Алексенко. Суми, Сумський державний університет, 2013. 133 с.
6. Карпенко М.Ю., Манакова Н.О., Гавриленко І.О. Технології створення програмних продуктів та інформаційних систем : навч. Посібник. Харків, ХНУМГ ім. О. М. Бекетова, 2017. 93 с.
7. Конспект лекцій з дисципліни «Аналіз вимог до програмного забезпечення» для студентів напряму підготовки 6.050103 «Програмна інженерія». Укладач: канд. фіз.-мат. наук, доцент Божуха Л.М. Дніпродзержинськ: ДДТУ, 2015. 94 с.
8. Цибульник, С. О. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного забезпечення. Підручник [Електронний ресурс]: підручник для здобувачів ступеня бакалавра за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології» / Цибульник С. О., Барандич К. С. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 270 с.

##### Допоміжна література:

9. Freeman E., Robson E. Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software. Sebastopol, O'Reilly Media, 2020. 672 p.
10. Richards M., Ford N. Fundamentals of Software Architecture: An Engineering Approach. Sebastopol, O'Reilly Media, 2020. 432 p.
11. Fowler M. Refactoring: Improving the Design of Existing Code. Boston, Addison-Wesley Professional, 2018. 448 p.
12. Gaddis T. Starting Out with Java: From Control Structures through Objects. London, Pearson, 2018. 1168 p.

13. Bloch J. *Effective Java*. Boston, Addison-Wesley Professional, 2018. 412 p.
14. Schildt H. *Java: A Beginner's Guide*. NY, McGraw-Hill Education, 2017. 752 p.

Інформаційні ресурси:

15. Цибульник С.О. Технології розроблення програмного забезпечення  
<https://do.ipk.kpi.ua/course/view.php?id=4999>
16. Цибульник С.О. Курсова робота з технологій розроблення програмного забезпечення  
<https://do.ipk.kpi.ua/course/view.php?id=5004>

## Навчальний контент

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

Основні форми навчання - самостійна робота студентів.

На лекціях студентам викладаються теоретичні основи дисципліни.

### 6. Самостійна робота студента

Самостійна робота студентів передбачає самостійне вивчення окремих питань за рекомендованими навчально-методичними матеріалами.

На самостійну роботу студентів виділяється 30 годин, з яких 8 годин – на підготовку до заліку і 22 години на опрацювання навчальної літератури відповідно до структури дисципліни та виконання поставленого завдання.

## Політика та контроль

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

Перед студентом ставляться наступні вимоги:

- **правила виконання роботи:**

- курсова робота виконується командами з 4-5 студентів;
- учасники кожної команди визначаються або затверджуються викладачем для запобігання випадків нерівномірного розподілу (наприклад, усі «сильні» студенти збираються в одну команду);
- кожна окрема команда у визначений викладачем спосіб отримує одне завдання на всіх її учасників (орієнтовний перелік завдань наведено в **Додатку А**);
- мова програмування та перелік технологій обираються або затверджуються викладачем;
- мова програмування обов'язково має бути зі статичною сильною типізацією, об'єктно-орієнтованою та високорівневою (Java або C#);
- учасники кожної команди ділять між собою наступні ролі без повторення: аналітик вимог, архітектор, програміст, документатор, тестувальник. Додатково в кожній команді визначається *Team Leader*, який буде керувати процесом виконання поставленого завдання та нести відповідальність за всіх учасників своєї команди;
- у разі необхідності один тестувальник може бути назначений в декілька команд одночасно;
- кожен з учасників має виконати частину завдання, яка безпосередньо пов'язана з його роллю у команді: формування вимог та технічного завдання, архітектура системи, програмування, документування, тестування;
- завершено роботу необхідно завантажити у систему Moodle.

- **правила призначення заохочувальних та штрафних балів:**

- штрафні бали призначаються за несвоєчасний захист роботи, заохочувальні – за виконання творчих робіт з дисципліни;
- максимальна кількість заохочувальних та штрафних балів визначається відповідно до Положення про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського, Положення про поточний, календарний та семестровий контроль результатів навчання в КПІ ім. Ігоря Сікорського, а також інших Положень та рекомендацій, які діють в КПІ ім. Ігоря Сікорського.
- **політика дедлайнів та перескладань:**
  - перескладання передбачено тільки за наявності документально підтверджених вагомих причин;
  - вчасним вважається захист протягом виділеного навчального часу відповідно до силабусу та/або календарного плану;
  - порушення даного дедлайну вважається незадовільним захистом і оцінюється в 0 балів.
- **політика округлення рейтингових балів:**
  - округлення рейтингового балу відбувається до цілого числа;
  - при округленні до цілого числа всі цифри, що йдуть за наступним розрядом замінюються нулями;
  - якщо цифра розряду, що залишився, 5 або більша, то ціле число збільшується на одиницю, а розряд прирівнюється до нуля;
  - якщо цифра розряду, що залишився, менша за 5, то ціле число не змінюється, а розряд прирівнюється до нуля.
- **політика оцінювання:**
  - оцінювання роботи відбувається комісією відповідно до Положення про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського, Положення про поточний, календарний та семестровий контроль результатів навчання в КПІ ім. Ігоря Сікорського, а також інших Положень та рекомендацій, які діють в КПІ ім. Ігоря Сікорського;
  - нижня межа позитивного оцінювання має бути не менше 60% від балів, визначених для цієї роботи;
  - негативний результат оцінюється в 0 балів;
  - бали за контрольний захід «Виконання роботи» (пункт 8 силабусу, PCO) виставляються один раз після останнього етапу з календарного плану (пункт 5 силабусу). Забороняється виставляти будь-які бали за даний контрольний захід до дати завершення останнього етапу з календарного плану;
  - кожен член комісії проводить оцінювання кожної роботи за наступними напрямками (за кожним окремо) по 90-бальній шкалі (від 0 до 90 балів, за «Виконання роботи» пункт 8 силабусу, PCO): формування вимог та технічного завдання, архітектура системи, програмування, документування, тестування. Оцінка за роботу визначається як середнє арифметичне оцінок за всіма напрямками;
  - загальна оцінка за кожну окрему роботу визначається за 100-бальною шкалою як середнє арифметичне оцінок від усіх членів комісії з додаванням балів за «Дотримання етапів календарного плану» (пункт 8 силабусу, PCO);
  - загальна оцінка за роботу виставляється кожному студенту/студентці з відповідної команди незалежно від його/її ролі (аналітик вимог, архітектор, програміст, документатор, тестувальник) та незалежно від оцінки за окремим напрямком. Виключенням є Team Leader, який може отримати додаткові заохочувальні або штрафні бали (у визначених даним силабусом межах) за керівництво своєю командою.
- **політика вирішення виняткових ситуацій:**

- на основі письмової скарги Team Leader'a викладач може виключити студента з відповідної команди з наступною заміною його завдання на інше незалежно від поточного етапу виконання курсової роботи;
- на основі письмової скарги тестувальника (наприклад, у разі невиконання відповідною командою завдання у визначені силабусом строки) викладач може переглянути зміст та оцінку його роботи.

#### **Академічна доброчесність**

Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі КПІ ім. Ігоря Сікорського. Детальніше: <https://kpi.ua/code>.

#### **Норми етичної поведінки**

Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі КПІ ім. Ігоря Сікорського. Детальніше: <https://kpi.ua/code>.

#### **Оскарження результатів оцінювання**

Студенти мають право підняти будь-яке питання, яке стосується процедури проведення або оцінювання роботи, та очікувати, що воно буде розглянуто комісією.

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

#### **Поточний контроль:**

№	Назва контрольного заходу	Кількість	Ваговий бал	Усього
1	Виконання роботи	1	90	90
2	Дотримання етапів календарного плану	10	1	10
			Усього:	100

#### **Календарний план:**

№	Назва етапів виконання курсової роботи	Виконати до	Бал
1	Формування вимог до програмного забезпечення	20.09.2021р.	1
2	Створення технічного завдання на розробку	27.09.2021р.	1
3	Архітектурне проектування з забезпеченням нефункціональних вимог (безпека, надійність, розширюваність, тощо)	11.10.2021р.	1
4	Документування архітектури	18.10.2021р.	1
5	Програмування основного функціоналу (графічний інтерфейс користувача, бізнес-логіка, тощо)	15.11.2021р.	1
6	Оформлення програмної документації	22.11.2021р.	1
7	Тестування	29.11.2021р.	1
8	Виправлення багів	06.12.2021р.	1
9	Написання детального керівництва користувача	13.12.2021р.	1
10	Оформлення пояснювальної записки до курсової роботи	20.12.2021р.	1

#### **Виконання роботи:**

- повне та вчасне виконання завдання без помилок – 90 балів;
- повне та вчасне виконання завдання з незначними помилками – 72-89 балів;
- повне виконання завдання з помилками – 54-71 бал.
- менше 60% правильної відповіді – 0 балів.

**Календарний контроль:** проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Для отримання «зараховано» з першої проміжної атестації (8 тиждень) студент матиме на менш ніж 2 бали (за умови, якщо на початок 8 тижня згідно з календарним планом «ідеальний» студент має отримати 4 балів).

Для отримання «зараховано» з другої проміжної атестації (14 тиждень) студент матиме не менш 4 бали (за умови, якщо на початок 14 тижня згідно з календарним планом «ідеальний» студент має отримати 8 балів).

**Семестровий контроль:** залік.

**Умови допуску до семестрового контролю:**

- виконане завдання з повним/частковим функціоналом та наявністю пояснювальної записки.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

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

**Робочу програму навчальної дисципліни (силабус):**

**Складено** доцентом, к.т.н., доцентом, Цибульником Сергієм Олексійовичем

**Ухвалено** кафедрою КІОНС (протокол № 14 від 06.07.2022 року)

**Погоджено** Методичною комісією приладобудівного факультету (протокол № 7/22 від 07.07.2022 року)

## ДОДАТОК А

### Орієнтовний перелік тем курсової роботи

1. Файловий менеджер.
2. Програмний модуль адресної книги.
3. Програмний модуль словника.
4. Програмний модуль тестування знань.
5. Автоматизована каса автовокзалу.
6. Автоматизоване робоче місце працівника бібліотеки.
7. Автоматизована система управління з/д перевезеннями.
8. Автоматизована система управління пасажирськими авіа перевезеннями.
9. Гра «Парні картинки».
10. Автоматизована система розв'язку систем лінійних рівнянь.
11. Автоматизована система розв'язку алгебраїчних рівнянь (з розпізнаванням формули).
12. Автоматизована система чисельного інтегрування із застосуванням сплайнів.
13. Автоматизована система квиткової каси кінотеатру.
14. Автоматизована пропускна система на підприємстві.
15. Калькулятор.
16. Автоматизована інформаційна система клієнта банку.
17. Програма обліку зайнятості аудиторій у вищому навчальному закладі.
18. Програмна система для автомобільного підприємства.
19. Автоматизована система керування складом аптеки.
20. Автоматизована система реєстратури поліклініки.