



Компоненти програмної інженерії 1. Вступ до програмної інженерії

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>бакалавр</i>
Галузь знань	<i>Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інформаційних систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>денна</i>
Рік підготовки, семестр	<i>Перший курс, другий семестр</i>
Обсяг дисципліни	<i>96</i>
Семестровий контроль/ контрольні заходи	<i>залік</i>
Розклад занять	
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор М.О.Сидоров Лабораторні:
Розміщення курсу	https://ipi.kpi.ua/navchannya/silabusy/

Програма навчальної дисципліни

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

1.1. Метою дисципліни є формування у студентів здатності:

виконувати шляхом моделювання та аналізу діяльності спрямованої на

- створення і супроводження програмного забезпечення в контексті обраної моделі життєвого циклу;
- відновлення інформації з програмного забезпечення за методами реверсивної інженерії;
- метричне визначення властивостей компонентів програмного забезпечення (вимог, проекту, коду, документації) за методами емпіричної інженерії програмного забезпечення;
- визначення економічних показників щодо створення та супроводження програмного забезпечення;
- визначення екосистем програмного забезпечення.

1.2. Основні завдання кредитного модуля.

Згідно з вимогами програми навчальної дисципліни студенти після засвоєння дисципліни мають продемонструвати такі результати навчання:

знання :

- історичного контексту щодо виникнення та розвитку інженерії програмного забезпечення;
- загальних положень інженерії програмного забезпечення;
- теоретичних основ моделювання життєвого циклу програмного забезпечення та його складових (продукт, процес, ресурс);
- теоретичних основ складових інженерії програмного забезпечення, а саме прямої, зворотної та

емпіричної інженерії програмного забезпечення;

- загальних положень щодо економіки програмного забезпечення;
- теоретичних основ культури програмного забезпечення і моделювання зрілості процесів програмного забезпечення;
- теоретичних основ моделювання екосистем програмного забезпечення.

уміння:

- проектувати, моделювати та аналізувати продукти і процеси життєвих циклів програмного забезпечення;
- користуватися ресурсами щодо проектування, аналізу та моделювання продуктів і процесів життєвих циклів програмного забезпечення;
- користуватися методами та засобами розрахування метричних характеристик складових моделей життєвого циклу програмного забезпечення;
- користуватися методами та засобами економіки програмного забезпечення;
- проектувати, моделювати та аналізувати екосистеми програмного забезпечення.

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

Вивчення дисципліни спирається на знання, отримані студентами при вивченні дисциплін «Об'єктно-орієнтоване програмування», та є базою для вивчення всіх інших професійних дисциплін, які є в навчальному плані.

Набуті знання та навички можна використати при працевлаштуванні.

3. Зміст навчальної дисципліни

Тема 1. Підґрунтя для появи інженерії програмного забезпечення. Інженерія програмного забезпечення в обличчях та подіях.
Тема 2. Інженерія програмного забезпечення, як одна з інженерних галузей.
розділ 2
Тема 1. Складові інженерії програмного забезпечення.
Тема 2. Поняття програми, програмного забезпечення, програмного продукту, продукції.
Тема 3. Види інженерії програмного забезпечення - пряма, зворотна, емпірична інженерії.
розділ 3
Тема 1. Життєвий цикл та його складові..
Тема 2. Моделювання життєвого циклу програмного забезпечення.
Тема 3. Допоміжні процеси життєвого циклу програмного забезпечення.
розділ 4
Тема 1. Створення та використання програмних компонентів.
Тема 2. Підходи до побудови програмного забезпечення
розділ 5
Тема 1. Вимірювання у програмному забезпеченні
Тема 2. Метричне забезпечення
розділом 6
Тема 1. Культура інженерії програмного забезпечення
Тема 2. Моделювання зрілості процесів програмного забезпечення.
розділом 7
Тема 1. Економіка програмного забезпечення, загальні положення..

Тема 2. Моделі та засоби оцінки вартості програмного забезпечення.
розділом 8
Тема 1. Моделювання екосистем програмного забезпечення..
Тема 2 Засоби моделювання екосистем програмного забезпечення.
Модульна контрольна робота

4. Навчальні матеріали та ресурси

Рекомендована література

9.1. Базова

1. СИДОРОВ М.О. Вступ до інженерії програмного забезпечення. – К.: НАУ, 2008. – 65с.
3. СИДОРОВ Н.А. Основи програмування – К.-НАУ. - 2018. - 435 с.
4. ISO/IEC 19501:2005. Unified Modeling Language Specification. Version 1.4.2
5. Report on a conference sponsored by the NATO science committee, Garmisch, Germany, 7th to 11th October 1968, Editors: Peter Naur and Brian Randell
6. Lehman M.M. "Programs, life cycles, and laws of software evolution", Proceedings of the IEEE, September 1980. P. 1060–1076.
7. Humphrey, Watts S. Managing the Software Process. Reading, MA: Addison-Wesley, 1989.
8. SWEBOK Guide V3.06 2014, IEEE Society.
9. СИДОРОВ Н.А. Software ecology - К.-№ 1- 2010.- С 1-10.

9.2. Допоміжна

1. N.SIDOROV. Software Engineering. –К.: НАУ, 2007. – 130 p.
2. Software engineering code of ethics and professional practice (Version 5.2) as recommended by the IEEE-CS/ACM Joint Task Force on Software Engineering Ethics and Professional Practices <https://www.ics.uci.edu/~redmiles/ics131-FQ03/week08ethics/> IEEE-ACM-Ethics
3. Chikofsky E.J Foreword, Comm. Of ACM. Vol. 37, N 5. 1994. P. 24.
4. Сидоров Н.А. Инженерия программного обеспечения – учебная дисциплина или подготовка бакалавра? Управляющие системы и машины. 2006. № 2. С. 25–34.
5. Сидоров Н.А. Повторное использование, переработка и восстановление программного обеспечения Управляющие системы и машины. 2000. № 3, 4. С. 27–37.
6. M.F. Lungu. Reverse Engineering Software Ecosystems. – Doct. Diss. – USI.- 2009. –208 p.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

5.1. Тематика лекцій

Розділ 1. Історичний аспект.

Тема 1.1 Підґрунтя для появи інженерії програмного забезпечення. Інженерія програмного забезпечення в обличчях та подіях.

Тема 1.2 Інженерія програмного забезпечення, як одна з інженерних галузей.

Література: [9.1.1., с. 4-11, 9.1.2., 9.1.3, с. 11-17, 9.2.1., с. 7-11.].

Розділ 2. Основні поняття інженерії програмного забезпечення.

Тема 2.1 Складові інженерії програмного забезпечення.

Тема 2.2 Поняття програми, програмного забезпечення, програмного продукту, продукції.

Тема 2.3. Види інженерії програмного забезпечення - пряма, зворотна, емпірична інженерії. Реінженерія. Реструктурування. Редокументування.

Література: [9.1.1., с. 23-30, 59-73, 9.1.3, с. 15-23].

Розділ 3. Моделювання життєвого циклу програмного забезпечення.

Тема 3.1. Життєвий цикл та його складові.

Процеси. Продукти. Ресурси. Послідовність процесів життєвого циклу. Доменний аналіз. Специфікування вимог. Проектування. Конструювання. Супроводження. Утилізація.

Тема 3.2. Моделювання життєвого циклу програмного забезпечення.

Поняття моделі життєвого циклу. Моделі життєвого циклу.

Тема 3.3. Допоміжні процеси життєвого циклу програмного забезпечення. Управління персоналом. Управління якістю. Документування.

Література: [9.1.1., с. 31-57, 9.1.6., 9.1.7., 9.1.8. с. 1-1 - 5-12, 8-1 – 9-10., 9.2.1, с. 82 - 120].

Розділ 4. Підходи щодо реалізації життєвого циклу програмного забезпечення.

Тема 4.1. Створення та використання програмних компонентів.

Багаторатне використання програмних компонентів. Вторинне використання програмного забезпечення у проектуванні і розробці програмного забезпечення

Тема 4.2. Підходи до побудови програмного забезпечення

Компонентне орієнтований підхід, процес орієнтований підхід, об'єктно орієнтований підхід, модельне орієнтований підхід.

Література: [9.1.8. с. 2-8 - 2-12., 3-8 – 3-12, 9.2.3, 9.2.4, 9.2.5].

Розділ 5. Метричне забезпечення моделювання та аналізу програмного забезпечення.

Тема 5.1. Вимірювання у програмному забезпеченні.

Емпіричні методи у програмному забезпеченні

Тема 5.2. Метричне забезпечення

Метрики. Вимірювальні засоби для застосуванні у програмному забезпеченні.

Література: [9.1.1., с. 66 - 74].

Розділ 6. Культура інженерії програмного забезпечення і моделювання зрілості процесів програмного забезпечення.

Тема 6.1. Культура інженерії програмного забезпечення

Загальні положення культури інженерії програмного забезпечення.

Тема 6.2. Моделювання зрілості процесів програмного забезпечення.

Загальні положення моделей CMM, CMMI, P-CMM

Література: [9.2.1., с. 16 – 30, 9.2.2].

Розділ 7. Економіка програмного забезпечення.

Тема 7.1. Економіка програмного забезпечення, загальні положення.

Методи та засоби економіки програмного забезпечення

Тема 7.2. Моделі та засоби оцінки вартості програмного забезпечення.

Моделі COCOMO, COCOMO II, SLIM, засіб Costar Estimation.

Література: [9.1.1., с. 89 – 99, 9.2.1., с. 122 – 132, 9.1.8., с. 12-1 – 12-16].

Розділ 8. Екосистеми програмного забезпечення.

Тема 8.1. Моделювання екосистем програмного забезпечення.

Поняття екосистеми програмного забезпечення.

Тема 8,2. Засоби моделювання екосистем програмного забезпечення (i*).

Література: [9.1.9., 9.2.6].

5.2. Тематика комп'ютерних практикумів/лабораторних робіт

№ з/п	Назва лабораторної роботи
1	Моделювання життєвого циклу програмного забезпечення і його складових.
2	Відновлення інформації про компоненти програмних систем методами зворотної інженерії.
3	Дослідження компонентів програмних систем методами прямої та реверсивної інженерії.
4	Дослідження метричних характеристик програмного забезпечення методами емпіричної інженерії.
5	Оцінка вартості програмного забезпечення.
6	Моделювання екосистем програмного забезпечення.

6. Самостійна робота студента/аспіранта

№ з/п	Назви тем і питань, що виносяться на самостійне опрацювання та посилання на навчальну літературу
1	<i>Розділ 1. Тема 1.</i> Підгрунтя для появи інженерії програмного забезпечення. Інженерія програмного забезпечення в обличчях та подіях.
2	<i>Розділ 1. Тема 2.</i> Інженерія програмного забезпечення, як одна з інженерних галузей.
3	<i>Розділ 2. Тема 1.</i> Складові інженерії програмного забезпечення.
4	<i>Розділ 2. Тема 2.</i> Поняття програми, програмного забезпечення, програмного продукту, продукції.
5	<i>Розділ 2. Тема 3.</i> Види інженерії програмного забезпечення - пряма, зворотна, емпірична інженерії.
6	<i>Розділ 3. Тема 1.</i> Життєвий цикл та його складові..
7	<i>Розділ 3. Тема 2.</i> Моделювання життєвого циклу програмного забезпечення.
8	<i>Розділ 3. Тема 3.</i> Допоміжні процеси життєвого циклу програмного забезпечення.
9	<i>Розділ 4. Тема 1.</i> Створення та використання програмних компонентів.
10	<i>Розділ 4. Тема 2.</i> Підходи до побудови програмного забезпечення
11	<i>Розділ 5. Тема 1.</i> Вимірювання у програмному забезпеченні
12	<i>Розділ 5. Тема 2.</i> Метричне забезпечення
13	<i>Розділ 6. Тема 1.</i> Культура інженерії програмного забезпечення
14	<i>Розділ 6. Тема 2.</i> Моделювання зрілості процесів програмного забезпечення.
15	<i>Розділ 7. Тема 1.</i> Економіка програмного забезпечення, загальні положення..
16	<i>Розділ 7. Тема 2.</i> Моделі та засоби оцінки вартості програмного забезпечення.
17	<i>Розділ 8. Тема 1.</i> Моделювання екосистем програмного забезпечення..
18	<i>Розділ 8. Тема 2.</i> Засоби моделювання екосистем програмного забезпечення.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Студент повинен вивчати дисципліну самостійно та вчасно.

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

Система рейтингових балів та критерії оцінювання

1. Лабораторні роботи

Вагові бали кожної лабораторної роботи наведені у таблиці 1. Сумарний ваговий бал за даний контрольний захід (r_3) складає **90 балів**.

Критерії оцінювання лабораторних робіт включають якість її виконання, захисту та оформлення звіту (таблиця 1).

Таблиця 1 – Вагові бали оцінювання лабораторних робіт

№	Назва роботи	Бали			
		Виконання	Захист	Звіт	Сума
1	Засоби моделювання життєвого циклу програмного забезпечення і його складових.	8	1	1	10
2	Моделювання процесу Специфікування вимог.	8	1	1	10
3	Моделювання процесу Проектування (Архітектурне та детальне).	8	1	1	10
4	Моделювання процесу Конструювання (Кодування та тестування).	8	1	1	10
5	Моделювання процесу Супроводження.	8	1	1	10
6	Моделювання процесів створення програмного забезпечення за гнучкою методологією.	8	1	1	10
7	Моделювання процесу Специфікування вимог за методом гнучкої методології.	8	1	1	10
8	Моделювання процесів Планування, моніторинг та виконання за методом гнучкої методології.	8	1	1	10
9	Моделювання процесу Тестування за методом гнучкої методології.	8	1	1	10
	Разом за лабораторні роботи	72	9	9	90

1. Виконання лабораторної роботи

(термін виконання та формат перевірки/захисту визначається викладачем)

- повне вчасне виконання та захист з поясненням роботи - 10;
- повне вчасне виконання та частковий захист - 5-9
- повне вчасне виконання без захисту - 5
- неповне вчасне виконання - 0-4.

2. Виконання модульної контрольної роботи

- повне виконання - 5;
- неповне виконання - 0-4.

2. Умови позитивної проміжної атестації

Для отримання “зараховано” з проміжної атестації студент повинен мати не менше ніж 50% від максимальної кількості балів.

3. Розрахунок шкали рейтингу R :

Максимальна сума вагових балів контрольних заходів протягом семестру складає:

$$R_D = 90 + 10 = 100 \text{ балів}$$

Для отримання студентом відповідних оцінок (ECTS та традиційних) його рейтингова оцінка R_D переводиться згідно з таблицею:

Таблиця 2. Рейтингова оцінка

$R_D = r_C + r_E$	Оцінка ECTS	Традиційна оцінка
95... 100	A	відмінно
85 ... 94	B	дуже добре

75 ... 84	C	добре
65 ... 74	D	задовільно
60 ... 64	E	достатньо
$R_D < 60$	FX	незадовільно
$r_c < 50$ або не виконані інші умови допуску до заліку	F	недопущений

9. Додаткова інформація з дисципліни (освітнього компонента)

Робочу програму навчальної дисципліни (силабус):

Складено проф. Сидоров М.О.

Ухвалено кафедрою ІІІ (протокол №16 від 23.06.25)

Погоджено Методичною комісією факультету (протокол №11 від 27.06.25.)