



СУЧАСНІ ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	123 Комп'ютерна інженерія
Освітня програма	Комп'ютерні системи та мережі
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)/заочна
Рік підготовки, семестр	4 курс, весняний семестр
Обсяг дисципліни	4 кредити, 120 один
Семестровий контроль/ контрольні заходи	Залік
Розклад занять	//rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: асистент Шевело О. П. alex.shevelo@gmail.com Лабораторні: асистент Шевело О.П. alex.shevelo@gmail.com
Розміщення курсу	https://comsys.kpi.ua

Програма навчальної дисципліни

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

Метою викладання дисципліни є отримання знань, вмінь та навичок, необхідних фахівцю, який спеціалізується в області розробки та проектуванню сучасного програмного забезпечення особливо бекенду.

Завданнями вивчення навчальної дисципліни є:

- ознайомлення з концепціями, та основними підходами до проектування сучасного програмного забезпечення;
- вивчення принципів та методів обирання технологій під конкретну поставлену задачу;
- вивчення технологій та принципів побудови високонавантажених сервісів;
- ознайомлення з організацією, та методологією організації роботи команд розробників.

У результаті вивчення навчальної дисципліни студент повинен:

Знати:

- У яких випадках достатньо базових знань по програмуванню (ООП, ASID, SOLID, etc), а коли потрібно створювати власний підхід ;

- Як правильно проводити Trade-of аналіз і для чого він потрібен;
- Повний обсяг елементів, які вкладаються в термін «технології» при розробці сучасного ПЗ;
- Принципи роботи з ризиками та закладання запасів на модифікацію.

Вміти:

- орієнтуватися у методах виборах технологій при розробці ПЗ;
- визначати оптимальні технології в залежності від проекту;
- орієнтуватися в питаннях проектування, побудови, експлуатації проектів в довгостроковій перспективі.

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

Необхідні дисципліни: “Програмування”, “Об’єктна - орієнтоване програмування”, “Системне програмування”, “Структури даних та алгоритми”, “Інженерія програмного забезпечення”, “Алгоритми та методи обчислень”

Дисципліни, які базуються на результатах навчання з даної дисципліни: “Системне програмне забезпечення”, “Комп’ютерні системи”

3. Зміст навчальної дисципліни

Розділ 1. Поняття технологій

Тема 1.1. Визначення «технологія»

Тема 1.2. Основні елементи системи які впливають на вибір технологій

Тема 1.3. Вимоги до обраних технологій, які мають не допустити заходження проекту в глухий кут

Розділ 2. Методи аналізу системи

Тема 2.1. Схеми опису системи і їх типи

Тема 2.2. Аналіз ризиків та Trade-of аналіз

Розділ 3. Робота з вимогами

Тема 3.1. Що таке вимоги, їх типи і чому це важливо

Тема 3.2. Методи отримання вимог

Розділ 4 . Огляд найпопулярніших технологій.

Тема 4.1. Типи Архітектурних шаблонів та їх основне призначення.

Тема 4.2. Моноліти і їх призначення та які технології підходять для них.

Тема 4.3. Мікросервіси і їх призначення та які технології підходять для них.

Тема 4.4. Serverless системи і їх призначення та які технології підходять для них.

Тема 4.5. Типи баз даних та методи вибору типу для проекту

Розділ 5 . Технології подальшого розвитку та підтримки проекту

Тема 5.1. Як не допустити глухого кута вибравши невірні технології

Тема 5.2. Рефакторинг\реінженірінг\оптимізація і технології потрібні для них

Розділ 6. Технології серверного забезпечення.

Тема 6.1. Хмарні сервіси і їх призначення

Тема 6.2. Методи вибору хмарних сервісів

Тема 6.3. Причини коли не можна використовувати хмарні сервіси

Розділ 7. Сучасні технології організації команд розробки.

Тема 7.1. База AGILE

Тема 7.2. SCRUM vs CANBAN

4. Навчальні матеріали та ресурс.

Базова:

1. Pierre Bourque, Richard Fairley, Guide to the Software Engineering Body of Knowledge, Version 3.0 SW
2. I. Sommerville, Software Engineering, 10th ed., Addison-Wesley, 2016
3. K.E. Wiegers, Software Requirements, 3rd ed., Microsoft Press, 2013..
4. P. Clements et al., Documenting Software Architectures: Views and Beyond, 2nd ed., Pearson Education
5. D. Budgen, Software Design, 2nd ed., Addison-Wesley, 2003
6. L. Bass, P. Clements, and R. Kazman, Software Architecture in Practice, 3rd ed., Addison-Wesley Prof

Додаткова:

1. Steve McConnell, Software Estimation: Demystifying the Black Art
2. John Dooley, Software Development and Professional Practice
3. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Очна форма навчання

Назви розділів, тем	Кількість годин			
	Всього	У тому числі		
		Лекції	Лабораторні заняття	СРС
Розділ 1. Поняття технологій Тема 1.1. Визначення «технологія» Тема 1.2. Основні елементи системи які впливають на вибір технологій	7	2	1	4

Тема 1.3. Вимоги до обраних технологій, які мають не допустити заходження проекту в глухий кут				
Розділ 2. Методи аналізу системи Тема 2.1. Схеми опису системи і їх типи Тема 2.2. Аналіз ризиків та Trade-of аналіз	10	4	2	4
Розділ 3. Робота з вимогами Тема 3.1. Що таке вимоги, їх типи і чому це важливо Тема 3.2. Методи отримання вимог	7	2	1	4
Розділ 4 . Огляд найпопулярніших технологій. Тема 4.1. Типи Архітектурних шаблонів та їх основне призначення. Тема 4.2. Моноліти і їх призначення та які технології підходять для них. Тема 4.3. Мікросервіси і їх призначення та які технології підходять для них. Тема 4.4. Serverless системи і їх призначення та які технології підходять для них. Тема 4.5. Типи баз даних та методи вибору типу для проекту	40	12	8	20
Розділ 5 . Технології подальшого розвитку та підтримки проекту Тема 5.1. Як не допустити глухого кута вибравши невірні технології Тема 5.2. Рефакторинг\реінженірінг\оптимізація і технології потрібні для них	15	4	1	10
Розділ 6. Технології серверного забезпечення. Тема 6.1. Хмарні сервіси і їх призначення Тема 6.2. Методи вибору хмарних сервісів Тема 6.3. Причини коли не можна використовувати хмарні сервіси	11	4	3	4
Розділ 7. Сучасні технології організації команд розробки. Тема 7.1. База AGILE Тема 7.2. SCRUM vs CANBAN	12	5	2	5
МКР	6	1		5
Залік	12	2		10
Разом	120	36	18	66

Заочна форма навчання

Назви розділів, тем	Кількість годин			
	Всього	У тому числі		
		Лекції	Лабораторні заняття	СРС
Розділ 1. Поняття технологій Тема 1.1. Визначення «технологія» Тема 1.2. Основні елементи системи які впливають на вибір технологій	10	1	1	8

Тема 1.3. Вимоги до обраних технологій, які мають не допустити заходження проекту в глухий кут				
Розділ 2. Методи аналізу системи Тема 2.1. Схеми опису системи і їх типи Тема 2.2. Аналіз ризиків та Trade-of аналіз	9		1	8
Розділ 3. Робота з вимогами Тема 3.1. Що таке вимоги, їх типи і чому це важливо Тема 3.2. Методи отримання вимог	5	1		4
Розділ 4 . Огляд найпопулярніших технологій. Тема 4.1. Типи Архітектурних шаблонів та їх основне призначення. Тема 4.2. Моноліти і їх призначення та які технології підходять для них. Тема 4.3. Мікросервіси і їх призначення та які технології підходять для них. Тема 4.4. Serverless системи і їх призначення та які технології підходять для них. Тема 4.5. Типи баз даних та методи вибору типу для проекту	25	1	2	22
Розділ 5 . Технології подальшого розвитку та підтримки проекту Тема 5.1. Як не допустити глухого кута вибравши невірні технології Тема 5.2. Рефакторинг\реінженірінг\оптимізація і технології потрібні для них	17	1		16
Розділ 6. Технології серверного забезпечення. Тема 6.1. Хмарні сервіси і їх призначення Тема 6.2. Методи вибору хмарних сервісів Тема 6.3. Причини коли не можна використовувати хмарні сервіси	18	1	2	15
Розділ 7. Сучасні технології організації команд розробки. Тема 7.1. База AGILE Тема 7.2. SCRUM vs CANBAN	18	1	2	15
МКР	6	1		5
Залік	12	1		11
Разом	120	8	8	104

Метою проведення циклу лабораторних робіт є набуття студентами необхідних практичних навичок використання сучасних технологій розробки програмного забезпечення.

Очна форма навчання

№	Назва лабораторної роботи	Кількість ауд. годин
1	Вибір теми проекту і первинні вимоги до нього	2

2	Основні схеми і діаграми проекту	4
3	Вибір стеку технологій для проекту	6
4	Вибір методології організації проекту	6

Заочна форма навчання

№	Назва лабораторної роботи	Кількість ауд. годин
1	Вибір теми проекту і первинні вимоги до нього	2
2	Основні схеми і діаграми проекту	2
3	Вибір стеку технологій для проекту	2
4	Вибір методології організації проекту	2
	Разом	8

6. Самостійна робота студента

Підготовка до лабораторних занять 18 годин. Підготовка до модульних контрольних робіт 8 годин.
Підготовка до заліку 10 години.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Під час занять з навчальної дисципліни студенти повинні дотримуватись певних дисциплінарних правил:

- 1) забороняється запізнюватись на заняття;
- 2) при вході викладача, на знак привітання, особи, які навчаються в КПІ ім. Ігоря Сікорського повинні встати;
- 3) не допускаються сторонні розмови або інший шум, що заважає проведенню занять;
- 4) виходити з аудиторії під час заняття допускається лише з дозволу викладача.
- 5) не допускається користування мобільними телефонами та іншими технічними засобами без дозволу викладача.

Лабораторні роботи здаються особисто з попередньою перевіркою теоретичних знань, які необхідні для виконання лабораторної роботи. Перевірка практичних результатів включає перевірку коду та виконання тестових завдань.

В процесі навчання викладач має право нарахувати до 5 заохочувальних балів за дострокове виконання лабораторної роботи, за проявлений творчий підхід при виконанні індивідуального завдання або за активну участь у обговоренні питань, що пов'язані з тематикою лекції або практичного заняття.

За виконання та здачу лабораторної роботи після зазначеного дедлайну, за значну кількість пропущених занять, або за порушення правил поведінки на заняттях викладач може призначити до 5 штрафних балів.

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

- Поточний контроль: виконання лабораторних робіт
 - Календарний контроль: виконання модульної контрольної роботи
- , провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

Семестровий контроль: залік

Умови допуску до семестрового контролю: семестровий рейтинг більше 40 балів.

Оцінювання окремих видів виконаної студентом навчальної роботи здійснюється в балах:

Вид навчальної роботи	Мах кількість балів	Сумарна кількість балів
Виконання та захист лабораторної роботи 1	15	85
Виконання та захист лабораторної роботи 2	20	
Виконання та захист лабораторної роботи 3	25	
Виконання та захист лабораторної роботи 4	25	
Виконання модульної контрольної роботи		15
Разом		100
Залік (додатково)		30

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

9. **Додаткова інформація з дисципліни (освітнього компонента)**

перелік теоретичних питань, які виносяться на семестровий контроль, наведено в Додатку 1

Умова зарахування додаткових балів.

В рамках вивчення навчальної дисципліни «Сучасні технології розробки програмного

забезпечення»

допускається зарахування балів, одержаних в результаті дистанційних курсів на платформі "Coursera", за умови попереднього погодження програми даного курсу з викладачем та за умови отримання офіційного сертифікату.

Робочу програму навчальної дисципліни (силабус):

Складено асистент кафедри обчислювальної техніки, Шевело О.П.

Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022)

Погоджено Методичною комісією факультету (протокол № 10 від 09.06.2022)

...