

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Загальна інформація про навчальну дисципліну

Повна назва навчальної дисципліни	Розробка сервіс-орієнтованого програмного забезпечення
Повна офіційна назва закладу вищої освіти	Сумський державний університет
Повна назва структурного підрозділу	Факультет електроніки та інформаційних технологій. Кафедра комп'ютерних наук
Розробник(и)	Коробов Артем Геннадійович
Рівень вищої освіти	Другий рівень вищої освіти, НРК – 7 рівень, QF-LLL – 7 рівень, FQ-EHEA – другий цикл
Семестр вивчення навчальної дисципліни	16 тижнів протягом 2-го семестру
Обсяг навчальної дисципліни	Обсяг становить 5 кред. ЄКТС, 150 год. Для денної форми навчання 32 год. становить контактна робота з викладачем (16 год. лекцій, 16 год. лабораторних занять), 118 год. становить самостійна робота.
Мова викладання	Українська

2. Місце навчальної дисципліни в освітній програмі

Статус дисципліни	Вибіркова навчальна дисципліна для освітніх програм спеціальності 122 "Комп'ютерні науки"
Передумови для вивчення дисципліни	Знання архітектури розподілених інформаційних систем, а також знання обчислювальної техніки та програмування
Додаткові умови	Додаткові умови відсутні
Обмеження	Обмеження відсутні

3. Мета навчальної дисципліни

сформувати базові знання, вміння і навички, необхідні для програмного синтезу сучасних інформаційних систем сервіс орієнтованої архітектури (SOA)

4. Зміст навчальної дисципліни

Модуль 1. Концепції науки по сервісі, менеджмент і інжиніринг
Тема 1 Визначення термінів «сервіс» і «наука про сервіси» Бізнес-орієнтований погляд на сервіси. IT-орієнтовані перспективи розвитку систем сервісів. Розгляд концепції сервісів з точки зору бізнесу. Перспективи розвитку систем сервісів в IT.

Тема 2 Загальні положення сервіс-орієнтованої архітектури (SOA). Принципи, компоненти та переваги SOA. Стандарти та протоколи.

Модуль 2. Сервіс-орієнтована архітектура і композиція сервісів

Тема 3 Впровадження SOA в розробці програмного забезпечення. Проектування сервісів. Інтеграція та взаємодія сервісів. Тестування та розгортання сервіс-орієнтованих систем. Переваги застосування SOA.

5. Очікувані результати навчання навчальної дисципліни

Після успішного вивчення навчальної дисципліни здобувач вищої освіти зможе:

PH1	Вміти обирати програмне забезпечення для організації обробки та збереження великих обсягів даних з урахуванням предметної області та ресурсних обмежень
PH2	Вміти класифікувати системи обробки даних, виявляти їх слабкі та сильні сторони
PH3	Вміти проектувати архітектуру програмного та інформаційного забезпечення розподілених інформаційно-телекомунікаційних систем
PH4	Вміти проектувати розподілені бази даних та сховища даних будь-якого об'єму

7. Роль освітнього компонента у формуванні соціальних навичок

Загальні компетентності та соціальні навички, формування яких забезпечує навчальна дисципліна:

ЗК1	Здатність до абстрактного мислення, аналізу та синтезу.
ЗК2	Здатність застосовувати знання у практичних ситуаціях.
ЗК5	Здатність вчитися і оволодівати сучасними знаннями.
ЗК7	Здатність генерувати нові ідеї (креативність)

8. Види навчальних занять

Тема 1. Визначення термінів «сервіс» і «наука про сервіси» Бізнес-орієнтований погляд на сервіси. IT-орієнтовані перспективи розвитку систем сервісів

Лк1 "Визначення термінів «сервіс» і «наука про сервіси»

Еволюція поняття сервісу: від послуги до системи сервісів; Класифікація сервісів за Лайтнером та іншими дослідниками; Наука про сервіси: історія розвитку, предмет та завдання.

<p>Лк2 "Бізнес-орієнтований погляд на сервіси. IT-орієнтовані перспективи розвитку систем сервісів"</p> <p>Визначення основних бізнес-цілей та бізнес-сервісів; IT-орієнтований погляд: створення IT-сервісів для підтримки бізнесу; Методи та засоби проектування системи сервісів на різних рівнях архітектури.</p>
<p>Лб1 "Класифікація і вибір сервісів для заданого бізнес-процесу"</p> <p>Розробка програми на Python для аналізу бізнес-процесу, визначення необхідних операцій; Вибір відповідних сервісів з каталогу, фільтрація за критеріями; Генерування звіту з переліком обраних сервісів з описом функціональності</p>
<p>Лб2 "Розробка IT-архітектури та дизайну системи сервісів для заданого бізнесу"</p> <p>Розробка програми на Python для аналізу вимог бізнесу до IT системи; Проектування архітектури: веб/мобільні додатки, API, мікросервіси; Документування дизайну системи, ER-діаграми БД, діаграми сервісів</p>
<p>Тема 2. Загальні Положення SOA</p>
<p>Лк3 "Модель і опис сервісу для його пошуку"</p> <p>Визначення поняття сервісу, його інтерфейсу та функціональності; Структура опису сервісу для пошуку та ідентифікації; Мови опису веб-сервісів: WSDL, WSCL, OWL-S; Реєстрація та публікація описів сервісів; Механізми пошуку та відбору сервісів на основі описів; Семантичне анотування описів для розширення можливостей пошуку; Використання UDDI реєстрів та anderen каталогів описів сервісів; Приклади описів реальних веб-сервісів та їх застосування; Переваги використання повних описів сервісів у SOA та Web.</p>
<p>Лк4 "Концепції Сервіс-Орієнтованої Архітектури"</p> <p>Визначення та основні принципи сервіс-орієнтованої архітектури (SOA); Компоненти SOA: постачальники сервісів, сервісні реєстри, споживачі сервісів; Сервіси в SOA: бізнес-сервіси та інфраструктурні сервіси, основні властивості; Базові стандарти та протоколи SOA (REST, SOAP, XML, HTTP, WSDL, UDDI); Способи взаємодії сервісів: парадигми запит-відповідь, публікація-підписка тощо; Проектування, розробка та впровадження SOA: етапи, архітектурні шаблони; Інтегрування застарілих систем із сервісно-орієнтованою архітектурою підприємства; Технології реалізації SOA (J2EE, .NET, Web Services, BPEL); Переваги, сценарії застосування та приклади використання SOA</p>
<p>Лб3 Розробка та використання опису веб-сервісу на основі OpenAPI</p> <p>створення специфікації OpenAPI для простого REST сервісу, генерування клієнтського коду на основі специфікації, тестування роботи сервісу за допомогою створеного клієнта</p>
<p>Лб4 Реалізація базових концепцій SOA з використанням мікросервісів</p> <p>Розробка простих мікросервісів, їх комунікація через API шину, демонстрація можливості лабільного масштабування та оновлення сервісів</p>
<p>Тема 3. Імплементация Сервіс-орієнтованої архітектури в розробці програмного забезпечення</p>
<p>Лк5 "Оркестрування і хореографія сервісів"</p> <p>Визначення понять оркестрування та хореографії сервісів; Призначення та цілі</p>

<p>оркестрування сервісів; Типи зв'язків між сервісами: послідовні та паралельні; Централізовані і децентралізовані підходи до оркестрування; Огляд популярних технологій оркестрування (BPEL, BPMN, WS-CDL тощо); Особливості моделювання процесів з використанням нотаций хореографії; Порівняння хореографії та оркестрування сервісів; Переваги використання оркестрування та хореографії; Приклади застосування оркестрування та хореографії на практиці;</p>
<p>Лк6 "Мікросервіси і контейнери"</p> <p>Визначення Сервіс-Орієнтованої Архітектури. Основні принципи та характеристики. Поняття сервісу в SOA. Типи сервісів. Переваги застосування. Взаємодія компонентів в SOA за допомогою інтерфейсів та протоколів. REST, SOAP та інші стандарти. Проектування SOA. Компоненти та шаблони. Платформи і фреймворки (WCF, Spring). Переваги SOA: масштабованість, гнучкість, повторне використання. Можливі проблеми та ризики при реалізації SOA. SOA на сучасному етапі. Перспективи розвитку.</p>
<p>Лк7 "«Сервіс-орієнтована архітектура з змішаною композицією сервісів. Агентнобазована сервіс-орієнтована архітектура"</p> <p>Поняття змішаної композиції сервісів в SOA; Переваги використання змішаної композиції; Способи компонування сервісів: оркестрування, хореографія, координація за допомогою агентів; Особливості агентно-базованої реалізації SOA; Архітектура агентно-базованої SOA; Порівняння агентно-базованої та сервісно-орієнтованої архітектур; Переваги агентно-базованого підходу до побудови SOA; Недоліки та обмеження агентно-базованої SOA; Приклади застосування агентно-базованої SOA на практиці</p>
<p>Лк8 "«Обчислювальна незалежна модель CIM. Незалежна від платформи модель PIM. Модель для конкретної платформи PSM"</p> <p>Обчислювально незалежна модель (CIM) - призначення, складові, переваги; Незалежна від платформи модель (PIM) - особливості, зв'язок з CIM; Модель для конкретної платформи (PSM) - призначення, зв'язок з PIM, реалізація; Перетворення CIM в PIM - методи, підходи, інструменти; Трансформація PIM в PSM - способи, засоби реалізації; Мови моделювання та стандарти для опису CIM, PIM та PSM (UML, QVT, OMG); Переваги використання CIM, PIM та PSM при розробці програмного забезпечення; Приклади трансформацій між моделями в реальних проектах; Інструментальні засоби підтримки моделей (Rational Rose, Eclipse Modeling Framework тощо)</p>
<p>Лб5 Розробка простого оркестратора сервісів на Python</p> <p>Використання фреймворку Flask для створення веб-оркестратора. Взаємодія із зовнішніми сервісами за допомогою бібліотеки Requests. Демонстрація основних патернів оркестрування (ланцюжок, паралельне виконання тощо)</p>
<p>Лб6 "Створення та розгортання мікросервісів у контейнерах Docker з використанням Python.</p> <p>Розробка простих мікросервісів на Python, створення Docker-контейнерів, рапук та тестування</p>
<p>Лб7 Розробка сервіс-орієнтованої системи з використанням WSGI та Flask для створення REST сервісів на Python.</p>

9. Стратегія викладання та навчання

9.1 Методи викладання та навчання

Дисципліна передбачає навчання через:

МН1	Лекційне навчання.
МН2	Проблемне навчання.
МН3	Практикоорієнтоване навчання

Лекції надають студентам теоретичні матеріали з тем дисципліни, що є основою для проблемного навчання здобувачів вищої освіти (РН1-РН4). Лекції доповнюються практичними заняттями, що надають студентам можливість застосовувати теоретичні знання на практичних прикладах (РН1-РН4). Проблемному навчанню сприятиме підготовка до лекцій. В особливих умовах застосовуються методи та засоби електронного навчання на базі платформи mix.sumdu.edu.ua Під час проведення занять студенти отримують навички комунікації, вміння працювати в команді, здатність логічно і системно мислити, аргументовано висловлювати свої думки. Виконання лабораторних робіт допоможе студентам розвивати та реалізувати навички логічного та системного мислення, тайм-менеджменту, самостійного опрацювання матеріалу

9.2 Види навчальної діяльності

НД1	Інтерактивні лекції.
НД2	Підготовка до лекцій.
НД3	Виконання лабораторних робіт

10. Методи та критерії оцінювання

10.1. Критерії оцінювання

Визначення	Чотирибальна національна шкала оцінювання	Рейтингова бальна шкала оцінювання
Відмінне виконання лише з незначною кількістю помилок	5 (відмінно)	$90 \leq RD \leq 100$
Вище середнього рівня з кількома помилками	4 (добре)	$82 \leq RD < 89$
Загалом правильна робота з певною кількістю помилок	4 (добре)	$74 \leq RD < 81$
Непогано, але зі значною кількістю недоліків	3 (задовільно)	$64 \leq RD < 73$
Виконання задовольняє мінімальним критеріям	3 (задовільно)	$60 \leq RD < 63$
Можливе повторне складання	2 (незадовільно)	$35 \leq RD < 59$
Необхідний повторний курс з навчальної дисципліни	2 (незадовільно)	$0 \leq RD < 34$

10.2 Методи поточного формативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МФО1 Проміжний модульний контроль	Проміжний модульний контроль призначений для перевірки рівня засвоєння теоретичного матеріалу. Проводиться у форматі тестування засобами системи mix.sumdu.edu.ua. Оцінка за проміжний модульний контроль не перескладається.	згідно графіку навчального процесу	https://mix.sumdu.edu.ua , telegram
МФО2 Опитування та усні коментарі викладача за його результатами	Призначені для контролю засвоєння теоретичних знань поточної і минулих лекцій. Проводиться протягом дискусій і обговорень проблематики лекційного заняття.	Протягом лекційного заняття	Google meet, telegram

МФОЗ Проміжне оцінювання виконання лабораторних завдань	Призначено для перевірки теоретичних та практичних знань, отриманих протягом модуля. Тестові питання та завдання для виконання рефакторингу, розробленого програмного забезпечення на лабораторних заняттях.	згідно графіку навчального процесу	https://mix.sumdu.edu.ua , особистий кабінет
--	--	------------------------------------	--

10.3 Методи підсумкового сумативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МСО1 Проміжний модульний контроль	Проміжний модульний контроль призначений для перевірки рівня засвоєння теоретичного матеріалу. Проводиться у форматі тестування засобами системи mix.sumdu.edu.ua . Оцінка за проміжний модульний контроль не перескладається.	згідно графіку навчального процесу	https://mix.sumdu.edu.ua , telegram
МСО2 Оцінювання участі в дискусії	Участь в дискусії не є обов'язковим видом завдання, але є необхідним для отримання максимальної оцінки за курс. Дискусії та обговорення направлені на отримання навичок пошуку, аналізу інформації, формулювання висновків та висловлення власної позиції щодо оголошених проблемних питань державною (чи англійською) мовою. Для отримання максимальної оцінки студент має не лише написати власний пост, а й прокоментувати, принаймі, два пости інших студентів.	В кінці лекції	Google Meet, https://mix.sumdu.edu.ua

<p>МСО3 Оцінювання звіту за результатами виконання лабораторних робіт</p>	<p>Звіт за результатами виконання лабораторних робіт повинен містити ілюстрацію основних результатів виконання завдань відповідно методичних вказівок. Для успішного зарахування необхідно виконати мінімальний рівень завдання та оформити звіт відповідно вимогам до оформлення звітів. Оцінка зі звіту може бути один раз підвищення за умови усунення зауважень до представлених результатів і надсилання звіту у вказані терміни. В разі затримки термінів виконання оцінка не може бути підвищена.</p>	<p>До початку наступного лабораторного заняття</p>	<p>Google Meet, https://mix.sumdu.edu.ua</p>
---	--	--	---

Контрольні заходи:

		Максимальна кількість балів	Мінімальна кількість балів	Можливість перескладання з метою підвищення оцінки
2 семестр		100 балів		
МСО1. Проміжний модульний контроль		20		
	2x10	20	10	Ні
МСО2. Оцінювання участі в дискусії		16		
	8x2	16	8	Ні
МСО3. Оцінювання звіту за результатами виконання лабораторних робіт		64		
	8x8	64	20	Ні

До диференційованого заліку необхідно виконати всі лабораторні роботи на мінімальний рівень складності завдань. При успішному (отримання персоналізованого сертифікату із вказівкою рівня успішності) вивченні масових відкритих онлайн курсів можуть бути реалізовані наступні варіанти перезарахувань частини кредитів: 1. Для курсу викадачем можуть бути розглянуті масові відкриті онлайн курси за умови попереднього аналізу структури курсу та узгодження зі студентом обсягу роботи та оцінювання.

11. Ресурсне забезпечення навчальної дисципліни

11.1 Засоби навчання

ЗН1	Бібліотечні фонди
ЗН2	Комп'ютери, комп'ютерні системи та мережи
ЗН3	Мультимедіа, відео- і звуковідтворювальна, проєкційна апаратура (відеокамери, проєктори, екрани, смартдошки тощо)
ЗН4	Програмне забезпечення (python, google colab, aws, pycharm)

11.2 Інформаційне та навчально-методичне забезпечення

Основна література	
1	Microservice APIs: Using Python, Flask, FastAPI, OpenAPI and more [Електронний ресурс] / Jose Haro Peralta.— Manning, 2023.
2	Building Microservices: Designing Fine-Grained Systems [Електронний ресурс] / Sam Newman.— O'Reilly, 2021.
3	Building Microservices Applications on Microsoft Azure: Designing, Developing, Deploying, and Monitoring [Електронний ресурс] / Н Chawla Н. Kathuria Newman.— APress, 2019.
4	Practical Microservices: Build Event-Driven Architectures with Event Sourcing and CQRS [Електронний ресурс] / Ethan Garofolo.— Pragmatic Bookshelf, 2020.
Допоміжна література	
5	https://microservices.io/
6	Microservices Patterns: With examples in Java [Електронний ресурс] / Chris Richardson.— Manning; First Edition (November 19, 2018).
7	https://realpython.com/python-microservices-grpc/
8	https://medium.com/@mansha99/microservices-in-python-django-rabbitmq-and-pika-fe1adb0c6a1a