Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ЦТЕ

Кафедра цифрових технологій в енергетиці

Проектування систем з розподіленими базами даних

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	***Другий (магістерський)***
Галузь знань	*F Інформаційні технології*
Спеціальність	*F3 Комп'ютерні науки*
Освітня програма	*Технології візуальних обчислень*
Статус дисципліни	*Нормативна*
Форма навчання	*Заочна*
Рік підготовки, семестр	*1 курс, 1 семестр*
Обсяг дисципліни	*5 кредитів (150 годин) 10 лек. 6 практ. 8 лаб. 126 СРС*
Семестровий контроль/ контрольні заходи	*Екзамен, МКР*
Розклад занять	*https://schedule.kpi.ua/*
Мова викладання	*Українська*
Інформація про керівника курсу / викладачів	Лектор: *к.т.н, доц. Полягушко Любов Григорівна*, *ddpolytechnic2016@gmail.com* Лабораторні: *к.т.н, доц. Полягушко Любов Григорівна*
Розміщення курсу	Засоби Classroom та e-mail.

Програма навчальної дисципліни

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

Програму навчальної дисципліни «Проектування систем з розподіленими базами даних» складено відповідно до освітньої програми освітньо-професійної програми «Технології візуальних обчислень» другого рівня вищої освіти спеціальності F3 «Комп'ютерні науки».

Метою кредитного модуля є формування у студентів **компетентностей** у відповідності до ОПП.

ЗК 01	Здатність до абстрактного мислення, аналізу та синтезу.
ЗК 02	Здатність застосовувати знання у практичних ситуаціях.
ЗК 05	Здатність вчитися й оволодівати сучасними знаннями.
ЗК 07	Здатність генерувати нові ідеї (креативність).
ФК 01	Усвідомлення теоретичних засад комп'ютерних наук.
ФК 05	Здатність розробляти, описувати, аналізувати та оптимізувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення.
ФК 07	Здатність розробляти програмне забезпечення відповідно до сформульованих вимог з урахуванням наявних ресурсів та обмежень.
ФК 09	Здатність розробляти та адмініструвати бази даних та знань.

В результаті засвоєння кредитного модуля студенти мають продемонструвати такі програмні результати навчання:

ПРН 01	Мати спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері комп'ютерних наук і є основою для оригінального мислення та проведення досліджень, критичне осмислення проблем у сфері комп'ютерних наук та на межі

	галузей знань.
ПРН 02	Мати спеціалізовані уміння/навички розв'язання проблем комп'ютерних наук, необхідні для проведення досліджень та/або провадження інноваційної діяльності з метою розвитку нових знань та процедур
ПРН 04	Управляти робочими процесами у сфері інформаційних технологій, які є складними, непередбачуваними та потребують нових стратегічних підходів.
ПРН 06	Розробляти концептуальну модель інформаційної або комп'ютерної системи.
ПРН 07	Розробляти та застосовувати математичні методи для аналізу інформаційних моделей.
ПРН 09	Розробляти алгоритмічне та програмне забезпечення для аналізу даних (включно з великими).
ПРН 10	Проєктувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення.
ПРН 11	Створювати нові алгоритми розв'язування задач у сфері комп'ютерних наук, оцінювати їх ефективність та обмеження на їх застосування.
ПРН 12	Проєктувати та супроводжувати бази даних та знань.
ПРН 14	Тестувати програмне забезпечення.
ПРН 23	Застосовувати технології проектування систем з розподіленими базами даних, здійснювати реплікації даних та забезпечувати обробку розподілених транзакцій, розподілених обчислень в задачах енергетики.

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

знання:

- Apache Kafka - платформа для потокової обробки даних в реальному часі;
- Apache Cassandra/ScyllaDB - NoSQL бази даних для зберігання часових рядів;
- принципи CAP-теореми та архітектурні патерни розподілених систем.

вміння:

- проектувати та налаштовувати розподілені кластери для обробки потоків даних;
- розробляти схеми баз даних оптимізовані для часових рядів;
- створювати потокові додатки для real-time аналізу.

досвід:

- розробки end-to-end pipeline для обробки даних smart grid систем;
- використання Kafka Streams та Cassandra для створення моніторингових систем;
- роботи з часовими рядами великих енергетичних підприємств.

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

У структурно-логічній схемі навчальна дисципліна «Проектування систем з розподіленими базами даних» розміщена тоді, коли студенти вже прослухали навчальні дисципліни «Проектування баз даних», «Технології розробки програмного забезпечення», що достатньо для виконання лабораторних робіт з даної дисципліни.
Отримані знання можна використовувати при виконанні магістерської дисертації.

3. Зміст навчальної дисципліни

Розділ 1. Теоретичні основи розподілених систем
Тема 1.1. Фундаментальні принципи розподілених систем
Тема 1.2. Архітектурні рішення
Розділ 2. Технології потокової обробки та nosql системи
Тема 2.1. Платформи потокової обробки
Тема 2.2. Спеціалізовані системи та потокова обробка
Розділ 3. Розподілені операції та управління даними

Тема 3.1. Транзакційні моделі та консистентність
Тема 3.2. Стратегії партиціонування даних
Розділ 4. Надійність, продуктивність та безпека систем
Тема 4.1. Забезпечення надійності та продуктивності
Тема 4.2. Інформаційна безпека систем
Розділ 5. Хмарні технології та перспективи розвитку
Тема 5.1. Сучасні технології та майбутні перспективи

4. Навчальні матеріали та ресурси

Базова

1. Apache Kafka Documentation. Офіційна документація Apache Kafka. URL: https://kafka.apache.org/documentation/
2. Apache Cassandra Documentation. Офіційна документація Apache Cassandra. URL: https://cassandra.apache.org/doc/latest/
3. Kafka Streams Documentation. Документація для розробки потокових додатків. URL: https://kafka.apache.org/documentation/streams/
4. Kafka Connect Documentation. Документація для інтеграції систем через Kafka Connect. URL: https://kafka.apache.org/documentation/#connect
5. Cassandra Query Language (CQL) Reference. Довідник мови запитів CQL. URL: https://cassandra.apache.org/doc/latest/cql/

Допоміжна

6. Confluent Education. Безкоштовні курси з Apache Kafka та потокової обробки. URL: https://www.confluent.io/training/
7. DataStax Academy. Безкоштовні курси з Apache Cassandra. URL: https://academy.datastax.com/
8. IEEE Power & Energy Society. Стандарти та публікації з енергетичних систем. URL: https://www.ieee-pes.org/
9. IEC 61850 Standard. Міжнародний стандарт комунікаційних протоколів для електричних підстанцій. URL: https://www.iec.ch/dyn/www/f?p=103:23:0::::FSP_ORG_ID:1231
10. NERC CIP Standards. Стандарти кібербезпеки критичної інфраструктури. URL: https://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx
11. US Energy Information Administration (EIA). Відкриті енергетичні дані та API. URL: https://www.eia.gov/opendata/
12. Міненерго України. Відкриті дані енергетичної галузі України. URL: https://mev.gov.ua/timeline/Vidkriti-dani.html
13. НЕК "Укренерго". Операційні дані енергосистеми України. URL: https://ua.energy/diyalnist/dyspetcherska-informatsiya/

Навчальний контент

5. Методика опанування навчальної дисципліни(освітнього компонента)

Лекція 1. Основи розподілених баз даних та Apache Kafka

Еволюція систем від централізованих до розподілених архітектур на прикладі енергетичних мереж. Вивчення CAP-теореми та її практичного застосування на прикладі систем рендерингу й потокового відео. Огляд різних архітектур розподілених баз даних та їх використання для енергетичних додатків. Знайомство з Apache Kafka як основною платформою для потокової обробки даних - на прикладі енергетичної телеметрії.

Лекція 2. NoSQL бази даних та Apache Cassandra

Порівняння NoSQL та SQL підходів для розподілених систем з урахуванням специфічних вимог галузі. Детальне вивчення архітектури Apache Cassandra та її особливостей для роботи з часовими рядами — на прикладі метрик рендер-ферм (frame time, GPU load). Методи проектування

оптимальних схем даних для зберігання енергетичних метрик з урахуванням patterns доступу та масштабування.

Лекція 3. Потокова обробка даних та event-driven архітектури

Використання Kafka Streams для створення аналітичних додатків на прикладі енергетичної сфери. Вивчення технік Complex Event Processing на прикладі детекції аномалій у відеопотоках. Real-time обробка великих обсягів IoT-даних від енергетичних установок. Робота з віконними функціями та stateful обробкою для потокових даних.

Лекція 4. Інтеграція систем та забезпечення надійності

Побудова ETL конвеєрів для інтеграції різнорідних енергетичних даних. Використання Kafka Connect для підключення до SCADA систем та інших промислових протоколів. Розподілені транзакції на прикладі багатоетапних рендер-пайплайнів. Стратегії disaster recovery та забезпечення безперервності бізнесу для критично важливих систем реального часу (енергетика, cloud gaming, live-трансляції).

Лекція 5. Безпека, моніторинг та compliance в розподілених системах

Імплементація комплексних заходів безпеки для розподілених баз даних. Забезпечення відповідності енергетичним регуляціям, включаючи стандарти NERC CIP. Методи моніторингу та оптимізації продуктивності на прикладі рендер-кластерів і медіапайплайнів. Використання хмарних технологій та перспективи розвитку розподілених БД в енергетичній галузі та у сфері візуальних обчислень (нейрорендеринг, digital twins).

Практичні роботи

Практична робота 1. Розгортання кластера Kafka-Cassandra для моніторингу енергетичних систем та рендер-ферми	2 год.
Практична робота 2. Потокова аналітика даних з Kafka Streams	2 год.
Практична робота 3. Побудова комплексного конвеєра обробки даних для інтеграції енергетичних систем та систем візуальних обчислень	2 год.

Лабораторні роботи

Лабораторна робота 1. Оптимізація схем Cassandra для енергетичних часових рядів та метрик рендерингу	2 год.
Лабораторна робота 2. Розширена потокова аналітика з Kafka Streams	2 год.
Лабораторна робота 3. Відновлення після катастроф та забезпечення безперервності бізнесу	2 год.
Модульна контрольна робота	2 год.

6. Самостійна робота студента/аспіранта

Тема 1.1. Фундаментальні принципи розподілених енергетичних систем

Проаналізуйте вплив мікросервісної архітектури на проектування розподілених систем моніторингу smart grid та обґрунтуйте переваги цього підходу для інтеграції різнорідних енергетичних підсистем.

Тема 1.2. Архітектурні рішення для енергетичних систем

Дослідіть архітектурні патерни CQRS та Event Sourcing в контексті енергетичних систем та визначте їх потенціал для розділення операцій читання/запису в системах управління енергетичними ресурсами.

Тема 2.1. Платформи потокової обробки для енергетичних застосувань

Порівняйте архітектурні підходи Apache Kafka та Apache Pulsar для обробки потокових даних в енергетичних системах, враховуючи їх можливості multi-tenancy та geo-replication.

Тема 2.2. Спеціалізовані системи та потокова обробка

Дослідіть концепцію Digital Twin в енергетичних системах та проаналізуйте, як спеціалізовані системи часових рядів можуть забезпечити синхронізацію між фізичними енергетичними об'єктами та їх цифровими копіями.

Тема 3.1. Транзакційні моделі та консистентність

Вивчіть застосування теорії ігор для координації розподілених енергетичних ресурсів в умовах eventual consistency.

Тема 3.2. Стратегії партиціонування енергетичних даних

Проаналізуйте концепцію consistent hashing та її модифікації для динамічного партиціонування IoT-даних в умовах змінної топології енергетичної мережі та непередбачуваного навантаження.

Тема 4.1. Забезпечення надійності та продуктивності

Дослідіть принципи chaos engineering та їх застосування для тестування відмовостійкості критичних енергетичних систем, включаючи методики контрольованого введення відмов.

Тема 4.2. Інформаційна безпека енергетичних систем

Вивчіть стандарт IEC 62351 для кібербезпеки енергетичних систем та проаналізуйте його інтеграцію з сучасними протоколами шифрування в розподілених базах даних.

Тема 5.1. Сучасні технології та майбутні перспективи

Дослідіть алгоритми для оптимізації розподілу енергетичних ресурсів.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

- Відвідування лекцій та лабораторних занять, а також відсутність на них, не оцінюється.
- При захисті лабораторних робіт студент має продемонструвати розроблений програмний код та результати його виконання на тестах, як заздалегідь підготованих, так і запропонованих викладачем. У випадку дистанційної форми навчання захист відбувається на відповідній конференції шляхом демонстрації екрана.
- Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code.
- Норми етичної поведінки Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: https://kpi.ua/code.

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

1. Стартовий рейтинг (виконання робіт в семестрі) оцінюється у 50 балів. Розподіл балів наведено в таблиці:

Види робіт	**Бали**
Практична робота 1. Розгортання кластера Kafka-Cassandra для моніторингу енергетичних систем та рендер-ферми	2
Практична робота 2. Потокова аналітика даних з Kafka Streams	2
Практична робота 3. Побудова комплексного конвеєра обробки даних для інтеграції енергетичних систем та систем візуальних обчислень	2
Лабораторна робота 1. Оптимізація схем Cassandra для енергетичних часових рядів та метрик рендерингу	10
Лабораторна робота 2. Розширена потокова аналітика з Kafka Streams	12
Лабораторна робота 3. Відновлення після катастроф та забезпечення безперервності бізнесу	12
Модульна контрольна робота	10
Всього	50

Критерії оцінювання заходів поточного контролю:

- 90–100% від максимуму – повне виконання, коректний код, вичерпні відповіді
- 70–89% – виконано з незначними недоліками
- 50–69% – виконано з помилками та неточностями
- 10–49% – часткове виконання, суттєві помилки
- 0% – не виконано або не захищено

За несвоєчасне подання лабораторної роботи (після встановленого терміну) нараховується штраф у розмірі 20% від отриманої оцінки.

Модульна контрольна робота оцінюється у 10 балів і проводиться у формі тесту з 20 питань по 0,5 бала кожне. Відповідь на кожне питання оцінюється як правильна (0,5 бала) або неправильна (0 балів). За несвоєчасне виконання МКР нараховується штраф у розмірі 20% від отриманої оцінки.

2. Умови допуску до екзамену: відсутність заборгованостей з лабораторних робіт 1 - 5 та стартовий рейтинг не менше 30 балів.

3. Екзаменаційний рейтинг (відповідь на екзамені) оцінюється в 50 балів. Екзаменаційний білет складається з двох теоретичних питань та одного практичного завдання. Ваговий бал кожного теоретичного питання – 15 балів, завдання – 20 балів.

Теоретична частина оцінюється таким чином:

- правильна чітко викладена, повна відповідь – (не менше 90% потрібної інформації) – 13-15 балів;
- достатньо повна відповідь (не менше 75% потрібної інформації) – 11-12 балів;
- неповна відповідь (не менше 60% потрібної інформації) – 9-10 балів;
- незадовільна відповідь – 0 балів.

Практичне завдання оцінюється таким чином:

- повне, безпомилкове розв’язування завдання – 18-20 балів;
- повне, розв’язування завдання із несуттєвими невідповідностями – 15-17-балів;
- завдання виконане з певними недоліками – 12-14 балів;
- завдання не виконано – 0 балів.

4. Рейтингова оцінка за освітній компонент розраховується як сума балів стартового та екзаменаційного рейтингів (50 + 50 = 100 балів) і визначається згідно з таблицею переведення рейтингових балів у оцінку 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 питань, які виносяться на семестровий:

1. Особливості даних розподілених систем (телеметрія, потокові дані) та вимоги до СКБД
2. CAP-теорема та її застосування на прикладі систем рендерингу/стрімінгу
3. Порівняння ACID та BASE моделей для розподілених застосувань
4. Архітектурні підходи: master-slave vs peer-to-peer на прикладі GPU-кластерів рендер-ферм
5. Lambda та Kappa архітектури для потокових енергетичних даних
6. Архітектура Apache Kafka: topics, partitions, brokers
7. Kafka Connect та Kafka Streams для потокової обробки даних рендерингу
8. Порівняння Apache Cassandra та ScyllaDB для зберігання часових рядів
9. Критерії вибору NoSQL рішень для потокових задач
10. Спеціалізовані TSDB: InfluxDB, TimescaleDB, OpenTSDB для метрик продуктивності (frame time, GPU load)

11. Apache Flink vs Apache Storm для обробки подій та кадрів у реальному часі
12. Retention policies та компресія енергетичних часових рядів
13. Two-Phase Commit та Saga pattern на прикладі багатоетапних рендер-пайплайнів
14. Blockchain для енергетичної торгівлі та координації ресурсів
15. Eventual consistency vs strong consistency в розподілених системах
16. Quorum-based консистентність в Apache Cassandra
17. Multi-master реплікація для географічно розподілених дата-центрів
18. Стратегії партиціонування: часове, географічне, hash-based, просторове (tiles, octrees) для 3D-сцен
19. Динамічне перепартиціонування при зміні навантаження
20. Fault tolerance та disaster recovery в критичних системах реального часу (cloud gaming, live-трансляції)
21. Backup стратегії для енергетичних часових рядів
22. Multi-datacenter реплікація та graceful degradation
23. Метрики продуктивності та профілювання запитів
24. Оптимізація індексів для енергетичних часових рядів
25. Capacity planning для зростаючих енергетичних систем
26. Аутентифікація та авторизація в розподілених системах
27. Шифрування даних in-transit та at-rest
28. Compliance з енергетичними регуляціями та стандартами
29. Managed сервіси AWS/Azure/GCP для рендер-ферм і медіапайплайнів
30. Serverless архітектури для розподілених додатків
31. Edge computing для локальної обробки відео з камер/AR-пристроїв
32. Hybrid cloud рішення для енергетичних підприємств
33. AI/ML інтеграція з базами даних на прикладі нейрорендерингу (NeRF, генеративна графіка)
34. Quantum computing перспективи для розподілених обчислень
35. Digital twins на прикладі енергетичних об'єктів та 3D-сцен у графіці
36. Налаштування Kafka кластера для потокових даних
37. Проектування схеми Cassandra для метрик рендер-ферми
38. Реалізація потокової аналітики енергетичних даних
39. Інтеграція різнорідних систем: енергетика та візуальні обчислення
40. Моніторинг та алертинг в розподілених СКБД

Робочу програму навчальної дисципліни (силабус):

Складено доцентом каф. ЦТЕ, к.т.н., доц. Полягушко Любов’ю Григорівною

Ухвалено кафедрою ЦТЕ (протокол № 22 від 17.06.2026)

Погоджено Методичною комісією НН ІАТЕ КПІ ім. Ігоря Сікорського (протокол № 9 від 26.06.26)