

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра системного проектування

Затверджено

На засіданні кафедри системного проектування факультету електроніки та комп'ютерних технологій Львівського національного університету імені Івана Франка (протокол № 1 від 28.08 2023 р.)

Завідувач кафедри:



Роман ШУВАР

Силабус з навчальної дисципліни
“Хмарні обчислення”,
що викладається в межах ОПП “Комп'ютерні науки”
першого (бакалаврського) рівня вищої освіти для здобувачів з
спеціальності 122 – Комп'ютерні науки

Львів 2023 р.

Назва дисципліни	Хмарні обчислення
Адреса викладання дисципліни	м. Львів, вул. Драгоманова, 50
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра системного проектування
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології 122 Комп'ютерні науки
Викладачі дисципліни	Ляшкевич Марія Юріївна, асистент
Контактна інформація	mariia.liashkevych@lnu.edu.ua , https://electronics.lnu.edu.ua/employee/liashkevych-m-yu
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення лекційних занять (за попередньою домовленістю). Також можливі он-лайн консультації через MS Teams або систему електронного навчання Moodle. Для погодження часу онлайн консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://moodle.elct.lnu.edu.ua/course/view.php?id=179
Інформація про дисципліну	Дисципліна “Хмарні обчислення” є дисципліною з циклу професійної та практичної підготовки спеціальності 122 Комп'ютерні науки та інформаційні технології, яка викладається в 5 семестрі в обсязі 3.5 кредитів (за Європейською Кредитно-Трансферною Системою ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено таким чином, щоб оволодіти базовими поняттями, пов'язаними з організацією та використанням хмарних технологій, управлінні ресурсами віддалених розподілених систем, використанням технологій розподілених обчислень. Тому у дисципліні представлено як огляд базових понять та інструментів хмарних технологій, так і засобів, які потрібні для вирішення типових завдань при використанні, налаштуванні та управлінні хмарними сервісами, розробки програм та програмних інтерфейсів для хмарних додатків.
Мета та цілі дисципліни	Метою вивчення нормативної дисципліни “Хмарні обчислення” є оволодіння базовими поняттями, теоретичними знаннями та практичними навичками використання хмарних технологій в різних галузях людської діяльності, а також налаштування сервісів хмарних обчислень, проектуванні корпоративних хмарних систем та застосунків.
Література для вивчення	Основна література:

<p>дисципліни</p>	<ol style="list-style-type: none"> 1. Renhman T.B., Cloud Computing Basics: Mercury Learning and information, 2019. - 198 p. - Режим доступу: https://terrorgum.com/tfox/books/cloudcomputingbasics_aselfteachingintroduction.pdf 2. Duncan C. E. Winn. Cloud Foundry: The Definitive Guide: Develop, Deploy, and Scale 1st Edition: O'Reilly, 2017. - 324 p. 3. K. Chandrasekaran. Essentials of CLOUD COMPUTING: CRC Press, 2015. - 40 p. 4. Kief Morris. Infrastructure as Code: O'Reilly, 2021. - 430 p. - Режим доступу: https://dl.ebooksworld.ir/books/Infrastructure.as.Code.2nd.Edition.Kief.Morris.OReilly.9781098114671.EBooksWorld.ir.pdf 5. Lee Chao. Computing Networking Theory, Practice, and Development: CRC Press, 2016. - 59. - Режим доступу: https://sadir.ws/bitstream/handle/123456789/1658/CloudComputing_Ebook.pdf?sequence=1&isAllowed=y 6. Amazon. Welcome to AWS Documentation. - [Режим доступу]: https://docs.aws.amazon.com/ 7. Amazon Cloud Design pattern. - [Режим доступу]: https://en.cloud.designpattern.org/index.php/Main_Page 8. Mark Vilkins. Learning Amazon Web Services (AWS): A HandsOn Guide to the Fundamentals of AWS Cloud 1st Edition. 9. Google App Engine [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу: https://cloud.google.com/appengine/. 10. Michael Collier, Robin Shashan. Fundamentals of Azure. Second Edition. Microsoft Azure Essentials: Microsoft, 2016. - 263p. 11. Essentials of Application Development on IBM Cloud. - [Режим доступу]: https://www.redbooks.ibm.com/redbooks/pdfs/sg248374.pdf 12. Michael Crump, Chris Pietschmann, Vahe Minasyan. The Developer's Guide to Azure. Microsoft Press, A division of Microsoft Corporation One Microsoft Way, Redmond, Washington 98052-6399. 13. Kai Hwang, Min Chen. Big-Data Analytics for Cloud, IoT and Cognitive Computing: Willey, 2017. - 428 p. 14. Designing Distributed System. - [Режим доступу]: https://azure.microsoft.com/mediahandler/files/resourcefiles/designing-distributed-systems/Designing_Distributed_Systems.pdf 15. Integrity Systems. Хмарні обчислення. - [Режим доступу]: http://integritysys.com.ua/solutions/privatecloud-solution/ 16. Building Secure & Reliable Systems. - [Режим доступу]: https://static.googleusercontent.com/media/sre.google/en//static/pdf/building_secure_and_reliable_systems.pdf 17. Cloud Computing for Science and Engineering. - [Режим доступу]: https://cloud4scieng.org/chapters/ 18. Cloud Computing. - [Режим доступу]: http://eddiejackson.net/web_documents/The_Definitive_Guide_to_Cloud_Computing.pdf 19. Analytic Architecture for Capabilities-Based Planning, Mission-System Analysis, and Transformation. - [Режим доступу]: https://www.researchgate.net/profile/Paul-Davis-10/publication/228769350_Analytic_Architecture_for_Capabilities-Based_Planning_Mission-System_Analysis_and_Transformation/links/00b7d5242ef5be1c78000000/Analytic-Architecture-for-Capabilities-Based-Planning-Mission-System-Analysis-and-Transformation.pdf 20. James Turnbull. The Docker Book, 2016. - 410 p. - [Режим доступу]: https://github.com/AngelSanchezT/books-1/blob/master/docker/the-docker-book.pdf 21. Nigel Poulton. The Kubernetes Book, 2021. - 243 p. - [Режим доступу]: https://github.com/rohitg00/DevOps_Books/blob/main/The%20Kubernetes%20Book%20(Nigel%20Poulton)%20(z-lib.org).pdf
<p>Обсяг курсу</p>	<p>Кількість кредитів ЄКТС: 3,5 (105 год), з них: 64 годин аудиторних занять (лекції: 32 год, лабораторні: 32 год.) та 41 год. самостійної роботи.</p>
<p>Очікувані результати навчання</p>	<p>Після завершення цього курсу студент буде знати та вміти:</p> <ul style="list-style-type: none"> Здатність до абстрактного мислення, аналізу та синтезу. Здатність застосовувати знання у практичних ситуаціях. Здатність спілкуватися іноземною мовою як усно, так і письмово. Здатність вчитися і оволодівати сучасними знаннями. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

	<p>Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами 7 замовника, технічним завданням та стандартами.</p> <p>Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.</p> <p>Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).</p> <p>Здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.</p> <p>Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.</p> <p>Здатність розробляти високопродуктивні програмні комплекси для вирішення завдань наук про дані, систем штучного інтелекту, вбудованих та інших інноваційних систем.</p> <p>Володіння методами розроблення систем підвищеної продуктивності, серверними та розподіленими технологіями, інструментальними засобами проектування та розробки веб-застосувань і нових технологій.</p> <p>Здатність здійснювати розробку програмного забезпечення використовуючи різні методології та засоби програмування з метою забезпечення їх високої надійності та продуктивності в роботі.</p> <p>Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.</p> <p>Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.</p> <p>Вміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.</p> <p>Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.</p> <p>Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.</p> <p>Вміти застосовувати методи компонентної розробки програмного забезпечення.</p> <p>Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.</p> <p>Знати та вміти застосовувати методи та засоби управління проектами.</p> <p>Вміти застосовувати сучасні технологічні рішення щодо розробки програмно-апаратних систем та їх компонентів.</p> <p>Знати засоби інтеграції, розгортання та підтримки спеціалізованих програмних компонентів, розроблених на основі інноваційних технологій для вирішення завдань високопродуктивних обчислень.</p> <p>Знати основи інженерії даних і конструювання конвеєрів даних та</p>
--	---

	вміти обирати оптимальні алгоритми і технології розробки інноваційних рішень, зокрема для вирішення задач наук про дані та вбудованих систем.
Ключові слова	Хмарні обчислення, хмарні технології, cloud computing, big data, розподілені системи, хмара, віртуалізація.
Формат курсу	Очний. Проведення лекцій, практичних робіт та консультації для кращого розуміння тем.
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Залік в кінці семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Комп'ютерні інформаційні мережі»
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Презентація, лекції, лабораторні роботи, обговорення, дискусія.
Необхідне обладнання	Мультимедійне обладнання, комп'ютерний клас, програми та сервіси MS Teams, Moodle, Хмарна платформа AWS (EC2, S3, IAM, Slack, Facebook Application, Facebook Messenger)
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться упродовж семестру за 100-бальною шкалою, де враховано бали за два контрольні заміри (по 50 балів за кожний замір). На останньому занятті студент отримує залік на основі набраних балів.</p> <p>Бали нараховуються за такими видами робіт:</p> <ul style="list-style-type: none"> - самостійні роботи: 20% оцінки контрольного заміру, тобто 10 балів. Контрольний замір здійснюється у вигляді тестування (20 запитань по 0.5 балів за кожний тест). Сумарно, за два заміри (16 самостійних робіт), отримуємо 20 балів від загальної оцінки за дисципліну. - лекційні заняття: 32% оцінки контрольного заміру, тобто 16 балів. Контрольний замір здійснюється у вигляді тестування (32 запитання по 0.5 балів за кожний тест). Сумарно, за два заміри (16 лекційних занять), отримуємо 32 бали від загальної оцінки за дисципліну. - лабораторні роботи: 48% оцінки контрольного заміру, тобто 24 бали. Контрольний замір здійснюється у вигляді результатів виконання роботи та у вигляді опитування чи тестування. Сумарно, за два заміри (11 лабораторних робіт), отримуємо 48 балів від загальної оцінки за дисципліну. <p>Бали за лабораторними роботами розподіляються так:</p> <ul style="list-style-type: none"> - лаб. робота 1, 2, 3, 7, 8 оцінюється в 3 бали, де від 0 до 2 балів за виконання та від 0 до 1 балу за тестування чи

опитування.

- лаб. робота 4, 5, 6 оцінюється в 4 бали, де від 0 до 3 балів за виконання та від 0 до 1 балу за тестування чи опитування.
- лаб. робота 9, 10, 11 оцінюється в 5 балів, де від 0 до 4 бали за виконання та від 0 до 1 балу за тестування чи опитування.

При тестування, вага питання має 0.2 бала (5 питань для одного лабораторної роботи). Критерії виконання та оцінювання більш детально розписані у завданнях для лабораторних робіт.

Академічна доброчесність: Очікується, що лабораторні та контрольні роботи студентів будуть їх оригінальними дослідженнями чи міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату чи обману.

Відвідання занять є важливою складовою навчання.

Очікується, що всі студенти відвідають усі лекції і лабораторні заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття.

Студенти зобов'язані дотримуватися усіх термінів визначених для виконання усіх видів робіт, передбачених курсом.

Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її передачі третім особам. Студенти заохочуються до використання також й іншої літератури та джерел, яких немає серед рекомендованих.

Політика виставлення балів. Враховуються бали набрані на поточному тестуванні, самостійній роботі та бали підсумкового тестування. При цьому обов'язково враховуються присутність на заняттях та активність студента під час практичного заняття; недопустимість пропусків та запізнь на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття в цілях не пов'язаних з навчанням; списування та плагіат; несвоєчасне виконання поставленого завдання. Жодні форми порушення академічної доброчесності не толеруються.

Критерії оцінювання результатів неформальної освіти:

Нарахування балів відбувається за написання студентом тез доповідей на конференціях, наукових статей, участь у діяльності наукових гуртків, участь у наукових семінарах та круглих столах, конкурсах, участь у заходах неформальної освіти за отримання сертифікатів про проходження навчання на різних освітніх платформах (Coursera, Prometheus тощо),

	<p>курсах на провідних ІТ компаніях за тематикою навчальної дисципліни. Кількість балів визначається відсотком покриття результатів відповідної активності до вимог результатів навчання з навчальної дисципліни.</p>
<p>Питання до модульного контролю</p>	<p>Перелік питань та завдань для проведення підсумкової оцінки знань певних тем:</p> <ol style="list-style-type: none"> 1. <i>Поняття хмарних застосунків (Основні концепції розробки хмарних застосунків. Концепція хмарних застосунків. Формалізація бізнес проблем. Аналіз вимог. Декомпозиція вимог. Історія користувача.)</i> 2. <i>Визначення бізнес-вимог до веб-застосунків (Планування на основі можливостей. Сценарії та прототипи. Моделювання бізнес-рішень. Проектування хмарних застосунків. Оцінка даних. Потіки даних.)</i> 3. <i>Можливості AWS (CloudFormation, EC2, S3, IAM, Databases, etc.)</i> 4. <i>Поняття архітектури хмарних застосунків (Поняття архітектури в розробці хмарних застосунків. Архітектурні фреймворки. TOGAF. RUP. Архітектурні активності. Нефункціональні вимоги. Функціональна відповідність.)</i> 5. <i>Нефункціональні вимоги до хмарних застосунків (Ефективність продуктивності. Використання. Надійність. Захист. Підтримка. Переносимість. Контроль на рівні архітектури. Agile архітектура. Типи архітектури. Поняття інтерпрайс архітектури.)</i> 6. <i>Поняття інфраструктури в хмарі (Хмарна інфраструктура. Пропозиції на ринку хмарних послуг. Провайдери хмарних засобів. Зв'язок та мережі. Поняття безпеки. Поняття інтеграції. Дані у хмарах.)</i> 7. <i>Проектування веб-застосунків (Рівні програмного забезпечення. Поняття балансиру завантаження. Інфраструктура. Сервіси та послуги. Безсерверні технології. API. Безсерверні рішення на Azure. Мікросервіси.)</i> 8. <i>Поняття мікросервісів (Мікросервісна архітектура. Декомпозиція функціональних вимог. Комунікація. Синхронізація. Вибір мікросервісної архітектури. Поняття докерної технології. Приклад мікросервісів на платформі Azure.)</i> 9. <i>Проектування архітектури даних (Сучасні шаблони і найкращі практики. Керування даними. Доступність даних. Проектування та розробка. Пересилання повідомлень.)</i> 10. <i>Поняття безпеки і моніторингу системи (Управління та моніторинг. Продуктивність та масштабування. Стійкість та надійність. Безпека.)</i>

11. *Поняття нереляційних баз даних (Поняття NoSQL. Модель даних. Типи моделей даних. Логічна модель. Створення фізичної моделі даних. Фізична модель даних для реляційних баз даних. Інструменти для моделювання даних.)*
12. *MongoDB (Гнучке моделювання даних за допомогою MongoDB Atlas. Приклад моделювання бази даних. Запити і агрегування. Детальніше про мову запитів в MongoDB.)*
13. *MongoDB SQL (Розуміння зіставлення MongoDB та SQL. MongoDB SQL: термінологія. MongoDB SQL: Виконувани файли бази даних. MongoDB SQL: команди. Приклад CRUD функцій в MongoDB. Побудова запитів в MongoDB.)*
14. *Графові бази даних (Що таке є мова запитів GraphQL. Переваги GraphQL. GraphQL в порівнянні REST. Схеми GraphQL. Схема визначення. Вирішувачі (Resolvers). Запит даних. Повернення даних. Apollo GraphQL. Приклад з NodeJS та Express. Приклад запиту для React client.)*
15. *Резидентна система управління базами даних Redis (Розподілене сховище пар ключ-значення. Конфігурації Redis. Типи даних Redis. Команди з використанням ключа. Redis server. Redis on python. Backup. Безпека Redis. Redis benchmark. Транзакції Redis. Redis pipelining. Redis на docker.)*
16. *Концепція розподілених систем (Приклади типових розподілених систем. Компоненти розподілених систем. Інтернет та інтранет. Використання пристроїв у розподілених системах. Обмін ресурсів в WWW. Веб-сервери та веб-браузери. HTTP/HTTPS протоколи. Програмні та апаратні сервісні рівні. Клієнт-серверні моделі. Веб-прохі сервери.)*
17. *Веб-застосунки (Веб-застосунки. Мобільні агенти. Комп'ютерні мережі. Синхронні та асинхронні розподілені системи. Події в системі. Порядок подій в реальному часі. Канали зв'язку. Питання захисту.)*
18. *Технологія Docker (Контейнерні технології. Контейнеризація програмного забезпечення. Контейнерні технології. Docker – основні компоненти та принцип роботи. Використання Docker в розробці хмарних вебсервісів.)*
19. *Ознайомлення з Kubernetes (Архітектура. Концепція технології Kubernetes. Компонент Master. Компонент Node. Додаткові сервіси. Мережі. Розгортання кластеру. Розгортання від початку до кінця.)*
20. *Розгортання веб-застосунку (Побудова конвеєру застосунку. Скрипти. Конфігурації. Інфраструктура. Індикатори атрибутів якості. Безпека. Автоматична побудова. Процеси. Схеми автоматичної побудови. Тестування. Види тестів. Автоматичне розгортання.)*

21. Безперервна інтеграція та безперервне розгортання (*Хмарне розгортання. Приклади розгортання в хмарні системи. Поняття безперервної інтеграції і безперервного розгортання. Життєвий цикл виробничих операцій. Керування розгортанням.*)
22. Поняття технологічних операцій (*Коротка історія DevOps. Спілкування, співпраця, інтеграція. Розроблення та IT операції. Переваги сучасних технологій розроблення програмного забезпечення. Agile та DevOps. Життєвий цикл для DevOps.*)
23. Технологічні операції для штучного інтелекту (*Проблеми, що вирішуються MLOps. Застосунки Штучного інтелекту в індустрії. Процес доставки ML моделі. Сервіси ШІ у хмарах. Функції ШІ у хмарах. ML засоби від Amazon, Google, IBM, Microsoft. Приклад MLOps з AWS.*)
24. Поняття DataOps (*Поява DataOps. Труднощі у виробництві. Ризиковані операції. Великі дані: парадигма взаємодії. Інтеграція даних у часі. Інтеграція даних у часі.*)
25. Керування даними у хмарі (*Керування даними. Бази даних. Раціональні бази даних. Ключ-значення. Нераціональні бази даних. Графові. Документ-бази даних. Поняття індексації. Доступність даних. Атрибути баз даних.)*
26. Розподілені бази даних (*Розподілені бази даних. Запит до баз даних. Сховища даних. Дельта-формат бази даних. Нормалізація баз даних. Поняття Data Warehouse. Data Mart.*)
27. Проектування сховищ даних (*Принципи побудови сховищ даних. Поняття зони приземлення даних. Озера даних. Архітектури сховищ даних. Схеми даних. Характеристики даних.*)
28. Безпека в хмарних застосунках (*Поняття безпеки. Моделі безпеки. Моделі зрілості безпеки. OWASP. Життєвий цикл розроблення безпеки*)
29. Приклад безпеки від Microsoft (*Стандарти безпеки. Безпека при розробленні хмарних застосунків. Процес ідентифікації чи авторизації. Поняття токенизації. Ролі та політики безпеки.*)
30. Оцінка ризиків в розробленні хмарних застосунків (*Оцінка ризиків. Потіки. Проектування поточкових процесів. Моделі і методи оцінки ризиків. Діаграми потоків даних. Компоненти діаграми потоків даних.*)
31. Засоби моделювання ризиків (*Засоби моделювання. Пом'якшення ризиків. Поняття Stride, Spoofing, Tampering, Repudiation, etc. Довірчі границі. Валідаційні моделі. Оцінка DREAD ризиків.*)

	<p>32. Тестування безпеки у веб-застосунках (<i>Поняття тестів на проникнення. Методологія тестування на проникнення. Засоби тестування. Типи тестів на проникнення. Стратегії та фази. Google Hacking Database (GHDB). Збір інформації. Nmap сканування. Сканування вразливостей. Основні шляхи визначення потоків у застосунках.</i>)</p> <p>33. Хмарні застосунки з великими даними (<i>Поняття великих даних. Аналіз великих даних. Створення організацій з великими даними. Нові сховища для великих даних</i>)</p> <p>34. Області застосування хмарних технологій (<i>Видобуток текстів та NLP. DeepQA. Аналіз соціальних мереж. Додаткові випадки використання великих даних. Технології для аналізу великих даних.</i>)</p> <p>35. Хмарні технології для бізнесу (<i>Хмарні технології та бізнес-процеси. Сприяння бізнес-інновації за допомогою хмарних обчислень. Новий бізнес сервіс. Переваги цифрового бізнесу з хмарними технологіями.</i>)</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література. Ресурси в Інтернеті	Завдання, год	Термін виконання, тиж.
1	Визначення хмарних обчислень. Концепція хмарних обчислень і технологій. Види та варіанти хмарних систем. IAAS, PAAS, SAAS, їх основні властивості та приклади. Можливості хмарних обчислень. Концептуальний розвиток хмарних обчислень.	лекція	1-5	2	кінець поточного тижня
	Визначити навчальний шлях для хмарних обчислень	лаб. робота	6-8	2	кінець поточного тижня
	Особливості хмарної платформи від Google	сам. робота	9	2,5625	кінець поточного тижня
2	Типи обчислень. Розподілені системи. Обчислювальні платформи і технології. Мережі	лекція	2, 4-6	2	кінець поточного тижня

	CDN. Поштові служби. Сховища даних: DropBOX, Google Drive, Microsoft OneDrive. Офісні системи: Google Docs, Microsoft Office				
	Вступ до AWS	лаб. робота	6-8	2	кінець поточного тижня
	Особливості хмарної платформи від Microsoft	сам. робота	10	2,5625	кінець поточного тижня
3	Можливості Віртуалізації. Рівні віртуалізації. Еталонна модель обчислювальної машини. Основні поняття віртуалізації. Платформи для віртуалізації.	лекція	1, 3-4	2	кінець поточного тижня
	Створення EC2 екземпляру	лаб. робота	6-8	2	кінець поточного тижня
	Особливості хмарної платформи від IBM	сам. робота	11-12	2,5625	кінець поточного тижня
4	Паралелізм. Паралельна обробка. Декомпозиція. Комунікація. Синхронізація. Обмеження та попередня оцінка продуктивності. Вибір паралельної архітектури. Закон Амдала. Приклади паралелізму.	Лекція	4, 7, 8	2	кінець поточного тижня
	Робота зі сховищем AWS S3	лаб. робота	6-8	4	кінець 5-го тижня
	Робота з базами даними в хмарах: SQL	сам. робота		2,5625	кінець поточного тижня
5	Огляд платформи від Amazon та Microsoft. Огляд Amazon AWS IaaS. Обчислювальні ресурси і компоненти. Мережі і безпека. Засоби розроблення AWS IaaS. Огляд Azure IaaS. Стан хмарних технологій. Огляд. Інфраструктура	лекція	7-8, 13-15	2	кінець поточного тижня

	Azure. Перевага технології Azure IaaS.				
	Робота з базами даними в хмарах: NoSQL	сам. робота	1-3, 8, 13-16	2,5625	кінець поточного тижня
6	Огляд платформи від Google та IBM. Огляд Google Cloud Platform. Обчислювальні ресурси. Мережа. Зберігання. Великі дані. Адміністрування. Розробка. Машинне навчання. Огляд IBM Cloud. Компоненти і засоби розробки IBM Cloud. DevOps і безперервне розгортання і доставка. REST архітектури і Watson API. Сервіси даних. IBM Cloud для розробки мобільних додатків.	лекція	7-8, 14-17	2	кінець поточного тижня
	Побудова чат-бота на основі прикладу за допомогою AWS Lex	лаб. робота	6-8	4	кінець 7-го тижня
	Проектування хмарних веб-застосунків	сам. робота	14-18	2,5625	кінець поточного тижня
7	Архітектурні стилі. Архітектури, орієнтовані на дані. Архітектури потоків даних. Архітектура конвеєрів і фільтрів. Архітектура віртуальних машин. Системи, керовані подіями. Клієнт-сервер архітектура. Сервіс-орієнтована архітектура.	лекція	4, 18	2	кінець поточного тижня
	Хмарні операції: DevOps	сам. робота	2, 7, 15	2,5625	кінець поточного тижня
8	Безпечна обробка великих даних. Як безпечно обробляти дані в хмарі? Виклики та огляд існуючого стану техніки. Криптографічні методи. Аутсорсинг даних на основі шифрування. Аутсорсинг даних на основі секретного обміну даними. Використання надійних обчислювальних платформ. Захищене обладнання. Гібридна	лекція	9, 18-19	2	кінець поточного тижня

	хмара. Аутсорсинг обробки даних на основі розділення даних. Висновок і відкриті проблеми.				
	Інтеграція чат-боту з Slack	лаб. робота	6-8	2	кінець поточного тижня
	Хмарні операції: MLOps	сам. робота	2, 17	2,5625	кінець поточного тижня
9	Системи керовані подіями. Функції та обробка, керована подіями. Функції як сервіс (FaaS). Труднощі FaaS. Шаблони FaaS. Моделі публікації і підписки. Черги повідомлень. Керування чергами.	лекція	2, 4, 6, 13, 17	2	кінець поточного тижня
	Створення Facebook застосунку	лаб. робота	6-8	2	кінець поточного тижня
	Хмарні операції: DataOps	сам. робота	2, 17	2,5625	кінець поточного тижня
10	Розгортання застосунків у хмарах. Хмарне розгортання. Приклади розгортання в хмарні системи. Поняття безперервної інтеграції і безперервного розгортання.	лекція	4, 7-8, 17-18	2	кінець поточного тижня
	Інтеграція чат-боту з Facebook Messenger	лаб. робота	6-8	2	кінець поточного тижня
	Поняття Docker	сам. робота	20	2,5625	кінець поточного тижня
11	Контейнерні технології. Контейнеризація програмного забезпечення. Контейнерні технології. Docker – основні компоненти та принцип роботи. Використання Docker в розробці хмарних вебсервісів. Архітектури сховищ даних. Схеми даних. Характеристики даних.	лекція	2-4, 7-8	2	кінець поточного тижня

	Документування архітектури розробленого чат-боту	лаб. робота	6-8	4	кінець поточного тижня
	Поняття Kubernetes	сам. робота	21	2,5625	кінець поточного тижня
12	Поняття технології Kubernetes. Ознайомлення з Kubernetes. Архітектура. Концепція технології Kubernetes. Компонент Master. Компонент Node. Додаткові сервіси. Мережі. Розгортання кластера. Розгортання від початку до кінця.	лекція	4, 7, 12, 16	2	кінець поточного тижня
	Поняття безперервної інтеграції	сам. робота	2, 4, 16-17	2,5625	кінець поточного тижня
13	Виробничі операції: DevOps. Коротка історія DevOps. Спілкування, співпраця, інтеграція. Розроблення та IT операції. Переваги сучасних технологій розроблення програмного забезпечення. Agile та DevOps. Життєвий цикл для DevOps. Розуміння потоку робіт.	лекція	1-2, 4	2	кінець поточного тижня
	Робота з Docker контейнерами	лаб. робота	6-8	4	кінець поточного тижня
	Поняття безперервного розгортання	сам. робота	2, 4, 16-17	2,5625	кінець поточного тижня
14	Виробничі операції: MLOps. Проблеми, що вирішуються MLOps. Застосунки Штучного інтелекту в індустрії. Процес доставки МЛ моделі. Сервіси ШІ у хмарах. Функції ШІ у хмарах. МЛ засоби від Amazon, Google, IBM, Microsoft. Приклад MLOps з AWS.	лекція	1-2, 7-8	2	кінець 15-го тижня
	Конвеєр безперервної інтеграції та безперервного розгортання (CI/CD)	сам. робота	2, 4, 16-17	2,5625	кінець поточного тижня

15	Виробничі операції: DataOps. Поява DataOps. Труднощі у виробництві. Ризиковані операції. Великі дані: парадигма взаємодії. Інтеграція даних у часі. Інтеграція даних у часі.	лекція	7-8, 17-18	2	кінець поточного тижня
	AWS CloudFormation приклад для CI/CD конвеєру	лаб. робота	6-8	4	кінець 16-го тижня
	Використання хмарних технологій для бізнес застосунків	сам. робота		2,5625	кінець поточного тижня
16	Побудова сучасних веб-застосунків на основі мікросервісів. Мікросервіси. Докер. Мікросервіси на платформі Azure. 12-факторні додатки. Контейнери. Вичислювальна платформа Azure. Azure Service Fabric.	лекція	7-8, 17-19	2	кінець поточного тижня
	Планування на основі можливостей (Capability-based planning)	сам. робота	19	2,5625	кінець поточного тижня