

1. Опишіть підходи до впровадження стійкості та відмовостійкості в мікросервісах.
2. Що слід враховувати, приймаючи рішення про перехід від монолітної архітектури до мікросервісної?
3. В чому плюси монолітної архітектури?
4. Що таке Domain Driven Design та концепція "обмеженого контексту"?
5. Як треба розподіляти бази даних між мікросервісами? Чи треба розподіляти бази взагалі? Які бази даних краще підходять до використання з мікросервісами?
6. Як може відбуватися комунікація з мікросервісною архітектурою з точки зору фронтенду?
7. Які є засоби авторизації та автентифікації користувачів в контексті роботи з монолітами та мікросервісами?
8. Для чого потрібні та принципи роботи Load Balancer
9. Як працює масштабування мікросервісів?
10. Проблеми та їх рішення моніторингу мікросервісів. Як працює визначення стану мікросервісів.
11. Засоби обробки логів в мікросервісах та відстеження помилок
12. Способи розміщення сервісів на одному або декількох хостах. Як відбувається деплоймент в таких випадках.
13. Strangler pattern. Для чого потрібен та приклад реалізації
14. Що таке CQRS та приклади його реалізації.
15. Що таке Saga та для чого вона використовується
16. Для чого існують Service Registry та методи їх впровадження в проект
17. Що таке синхронні та асинхронні виклики в контексті комунікації мікросервісів. Плюси та мінуси цих підходів.
18. Які є підходи до деплойменту мікросервісів? Наведіть плюси та мінуси цих підходів.
19. Що таке Event Sourcing та як він впроваджується в CQRS
20. Підходи (протоколи, паттерни) до комунікації між мікросервісами.
21. Що таке REST та як він може використовуватись в мікросервісній архітектурі.
22. Які загальні антипатерни в архітектурі мікросервісів? Як їх уникнути?

23. Яка роль брокерів повідомлень в архітектурі мікросервісів? Наведіть приклади.
24. Що таке Dockerfile? Наведіть приклад Dockerfile для простого мікросервісу.
25. Що таке томи Docker і коли їх слід використовувати?
26. Для чого потрібен Docker? В чому різниця між Docker та віртуальною машиною?
27. Поясніть шаблон Circuit Breaker і коли його слід застосовувати в мікросервісах.
28. Опишіть призначення та структуру файлів Docker Compose для керування кількома контейнерами.
29. Опишіть принцип єдиної відповідальності в контексті розробки мікросервісів.
30. Дайте визначення архітектурі мікросервісів і поясніть, чим вона відрізняється від монолітної архітектури.