1. Який принцип лежить в основі мікросервісної архітектури?

* Об'єднання сервісів
* Розбиття на сервіси
* Побудова монолітної архітектури
* Збільшення залежностей між сервісами
* Зменшення функціональності системи

2. Яка перевага мікросервісної архітектури з точки зору розробки та підтримки системи?

* Збільшення складності
* Збільшення залежностей між компонентами
* Зменшення залежностей між сервісами
* Побудова більш монолітної системи
* Зменшення функціональності системи

3. Як повинна здійснюватися комунікація між сервісами?

* За допомогою телефонів
* За допомогою API
* За допомогою пошти
* За допомогою кур'єра
* За допомогою смс

4. Яку комунікацію рекомендується використовувати для забезпечення надійності та масштабованості?

* Синхронну комунікацію
* Асинхронну комунікацію
* Чергову комунікацію
* Багатопотокову комунікацію
* Статичну комунікацію

5. Які переваги має використання асинхронної комунікації між сервісами?

* Забезпечує надійність
* Забезпечує масштабованість
* Забезпечує швидкість
* Всі відповіді правильні
* Ні одна відповідь не правильна

6. Яка перевага у використанні окремої бази даних для кожного сервісу?

* Збільшення ризику збоїв у базі даних
* Забезпечення незалежності між сервісами
* Зменшення масштабованості
* Використання однієї бази даних для всіх сервісів
* Забезпечення недостатньої кількості ресурсів.

7. Що рекомендується для забезпечення масштабованості та надійності сервісів?

* Використання одного контейнера для всіх сервісів
* Використання різних баз даних для кожного сервісу
* Використання контейнерних технологій, таких як Docker
* Використання великої кількості монолітних сервісів
* Використання тільки віртуалізації

8. Що таке управління конфігурацією в мікросервісній архітектурі?

* Збереження даних про користувачів
* Управління секретами та ключами
* Управління налаштуваннями кожного сервісу
* Організація комунікації між сервісами
* Створення інтерфейсів для користувачів

9. Що є важливими аспектами мікросервісної архітектури?

* моніторинг та логування
* розробка інтерфейсів
* розгортання на хмарних платформах
* настройка баз даних
* створення зовнішніх API

10. Як забезпечити надійність та якість роботи системи в мікросервісній архітектурі?

* використовувати більше серверів;
* мати велику кількість сервісів у системі;
* автоматизувати тестування та релізи;
* збільшувати потужність обладнання;
* зменшувати кількість користувачів системи.

12. Яку роль відіграє моніторинг у мікросервісній архітектурі?

* забезпечення надійності та якості роботи системи
* автоматизація тестування
* швидка ідентифікація та вирішення проблем
* налагодження баз даних
* створення зовнішніх API

13. Як можна забезпечити надійність та якість роботи системи в мікросервісній архітектурі?

* за допомогою автоматизації тестування та релізів
* розробка інтерфейсів
* розгортання на хмарних платформах
* настройка баз даних
* створення зовнішніх API

14. Який інструмент дозволяє швидко та безпечно внести зміни у систему в мікросервісній архітектурі?

* автоматизація тестування
* логування подій
* моніторинг
* реліз-менеджмент
* настройка баз даних

15. Які переваги має використання контейнерів та інфраструктури як коду?

* Забезпечення безпеки;
* Швидке розгортання та налаштування сервісів;
* Автоматизація процесів розгортання;
* Відповідь на зміни у системі;
* Усі відповіді правильні.

16. Які механізми можна використовувати для забезпечення безпеки в мікросервісній архітектурі?

* Автентифікація;
* Авторизація;
* Шифрування даних;
* Усі відповіді правильні;
* Ні одна відповідь не правильна.

17. Що потребує мікросервісна архітектура від підтримки та супроводу?

* Розробка нових сервісів;
* Розгортання та налаштування сервісів;
* Внесення змін у систему;
* Моніторинг системи;
* Усі відповіді правильні.

18. Що таке Docker та Kubernetes, яка їхня роль у розгортанні сервісів?

* Docker - платформа для обміну повідомленнями між сервісами, Kubernetes - інструмент для оркестрації контейнерів
* Docker - інструмент для оркестрації контейнерів, Kubernetes - технологія контейнеризації
* Docker - інструмент для розгортання та налаштовування сервісів у контейнерах, Kubernetes - інструмент для оркестрації контейнерів
* Docker - моніторинговий інструмент, Kubernetes - інструмент для збору метрик з сервісів
* Docker - інструмент для збору метрик з сервісів, Kubernetes - інструмент для оркестрації мікросервісів

19. Що таке Apache Kafka та яку функцію вона виконує у системі?

* Apache Kafka - технологія контейнеризації, яка дозволяє розгортати та налаштовувати сервіси у контейнерах
* Apache Kafka - платформа для обміну повідомленнями між сервісами, яка забезпечує високу доступність та масштабованість
* Apache Kafka - інструмент для моніторингу та логування, який дозволяє збирати, обробляти та відправляти логи
* Apache Kafka - інструмент для збору метрик з сервісів, який забезпечує потужний пошуковий двигун та аналітику даних
* Apache Kafka - інструмент для моніторингу мікросервісів, який забезпечує збір метрик з сервісів

20. Що таке Docker та Kubernetes?

* Технологія контейнеризації та інструмент для оркестрації контейнерів
* Інструмент для моніторингу та логування
* Розподілена платформа для обміну повідомленнями
* Інструменти для моніторингу мікросервісів
* Нічого з вище перерахованого