Final exam

- **For which of the following architecture requirements would you use CQRS? Select all architectural requirements so that if this architectural requirements applies, you would apply the CQRS pattern.**
    - When read performance is critical and need to be extremely fast for your microservice.
    - When queries and commands have different scaling requirements for your microservice
    - When your screens start to look very different than your table in your microservice
- **Explain clearly the difference(s) between a Service Oriented Architecture (SOA) and a Microservice architecture.**
    - A SOA uses an ESB to connect the services (orchestration), while in a microservice architecture you do not use an ESB but chereography.
    - SOA have course grained services while microservices have small services.
- **Explain how orchestration works in a distributed system**
    - You have one central orchestrator that talks to other systems
- **Give the advantages and disadvantages of orchestration**
    - Advantages:
        - Easy to monitor and process
        - Full control over synchronous flow
    - Disadvantages
        - Orchestrator is a single point of failure
        - Coupling
        - No parallel processing
- **Explain how choreography works in a distributed system**
    - You don't have one central orchestrator that talks to all the other services. The individual services knows how to talk to other services
- **Give the advantages and disadvantages of choreography**
    - Advantages:
        - Fast. Parallel processing
        - Less coupling
        - No single point of failure
    - Disadvantages
        - Hard to monitor and process
- **Explain clearly Conways law**
    - Organizations which design systems are constrained to produce designs which are copies of the communication structure of these organizations

- **Explain clearly what Conways law means for a microservice architecture.**
    - If you want to use a microservice architecture in your organization, you have to change the structure of the organization so that the structure of the organization is the same as the structure of the microservice architecture. Every Scrum/DevOps team controls one or more microservice.
- **Explain clearly why local logging does not work in a microservice architecture**
    - Containers come and go dynamically
- **Explain clearly how logging should be implemented in a microservice architecture.**
    - Collec all logging to a central place. You can do thaet using ELK stack. You can also publish the logs to kafka and collect it centerally.
- **Explain clearly how a circuit breaker works in a microservice.**
    - We set a timeout on every remote call.
    - If the call takes longer than the timeout or throws exception. The next time we call the service we fail fast without waiting for the timeout to occur. We call an alternative method. The circuit breaker keeps checking if the remote service will come online.
- **Explain the 3 techniques that a circuit breaker uses.**
    - Fail fast
    - Fail gracefully
    - Recover seamlessly
- **Explain what does event sourcing means**
    - Event sourcing means that we store events instead of the state
- **Explain clearly why kafka uses event sourcing. What are the advantages.**
    - Because messages are never changed or deleted. it doesn't matter how many publishers or subscribers we have on one topic
- **Explain the difference between Oauth2 and JWT.**
    - OAuth2 is a standard for authorization. But it doesn't define a standard for tokens.
    - JWT defines a standard for tokens
- **What does JWT add to Oauth2/ In other words, why do we need JWT if we have Oauth2?**
    - JWT defines a standard for tokens. The token contains the role of the user and JWT will sign and encrypt the token, so that the receiver of the token can securely retrieve the role of the user without having to call the authentication server.