

← outstanding men

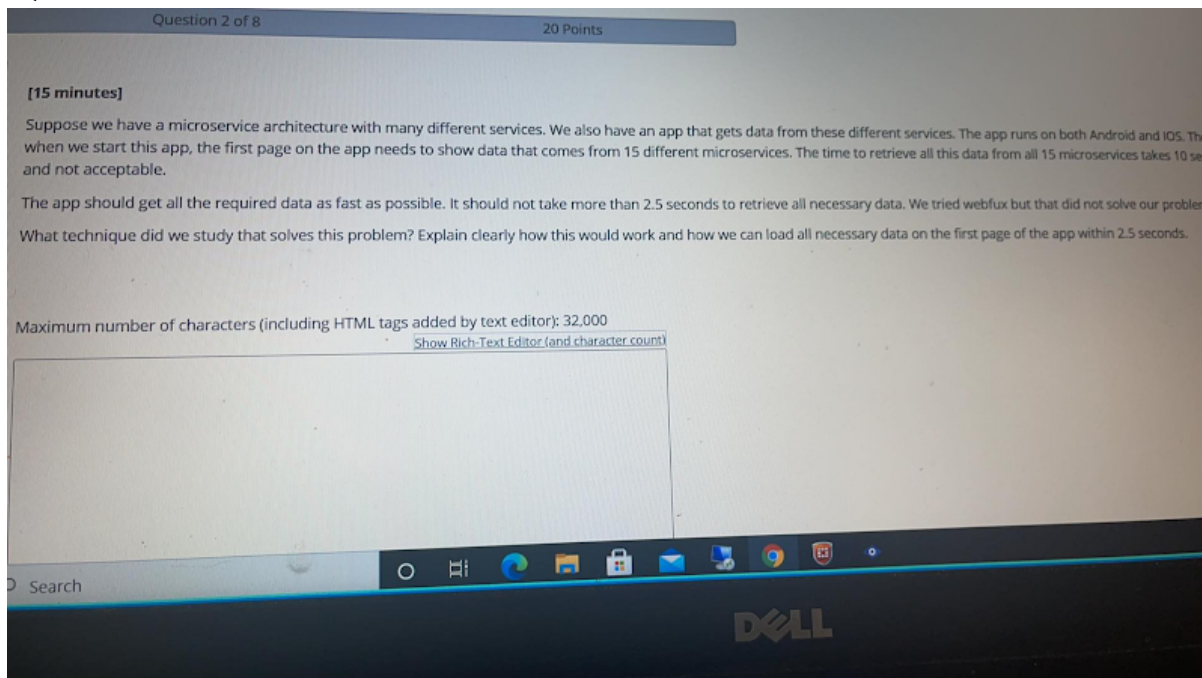
Question 1

Explain Conways law

Explain Conways law means for microservice architecture

- Organizations which design systems are constrained to produce designs which are copies of the communication structure of these organizations
- 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

Question 2



Solution is :

- Kafka as a distributed messaging queue.
- Each service subscribes to the broker and publishes the data that it has.
- Has integration service collect all of those data from different service and get one response to the client side service.

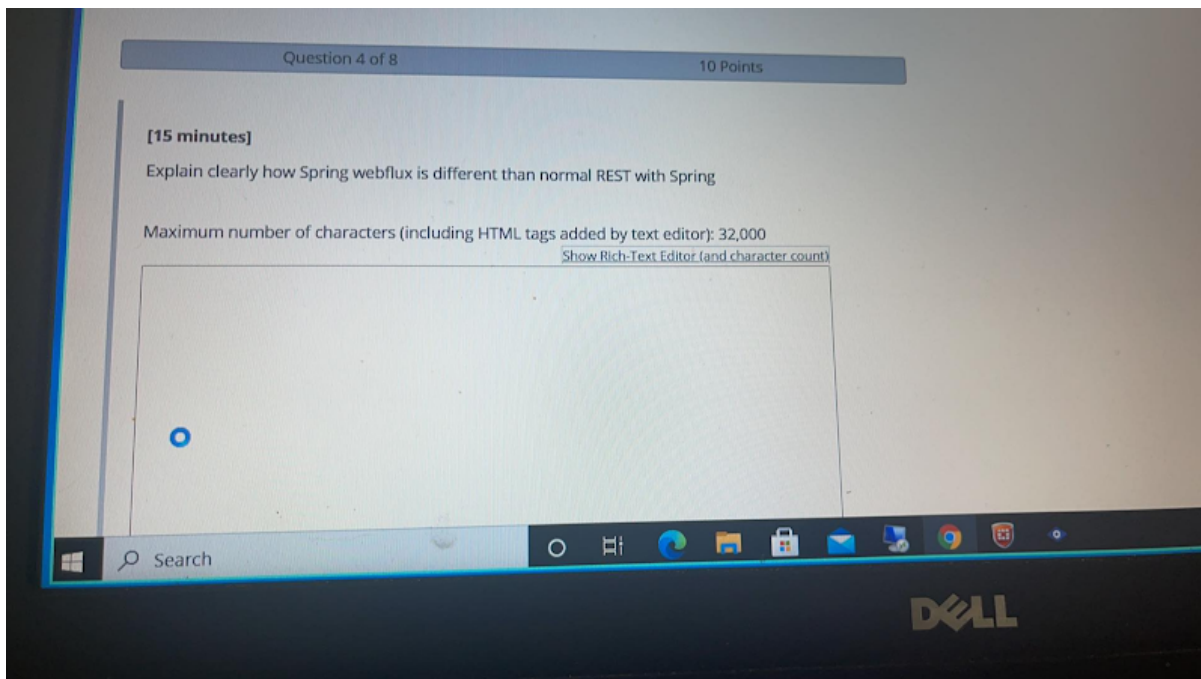
← outstanding men

- Fail fast : Service A gets an immediate response
- Fail gracefully: Service A can fall back to an alternative
- Recover Seamlessly: Circuit breaker will periodically check if service B is back online.

=====

=====

Question 4 basiouny



Spring WebFlux => Allows to build reactive web(REST) applications
provides reactive, async, non-blocking programming support for applications

Advantage:

- Non-Blocking : form application server thread pool perspective instead of take thread and hold it till the web service gets back with result it returns an optional empty response and register the client side reference and release then once the webservice has response it returns back to the client.

Performance : No need to wait till all results are available

Scaling :Less threads needed

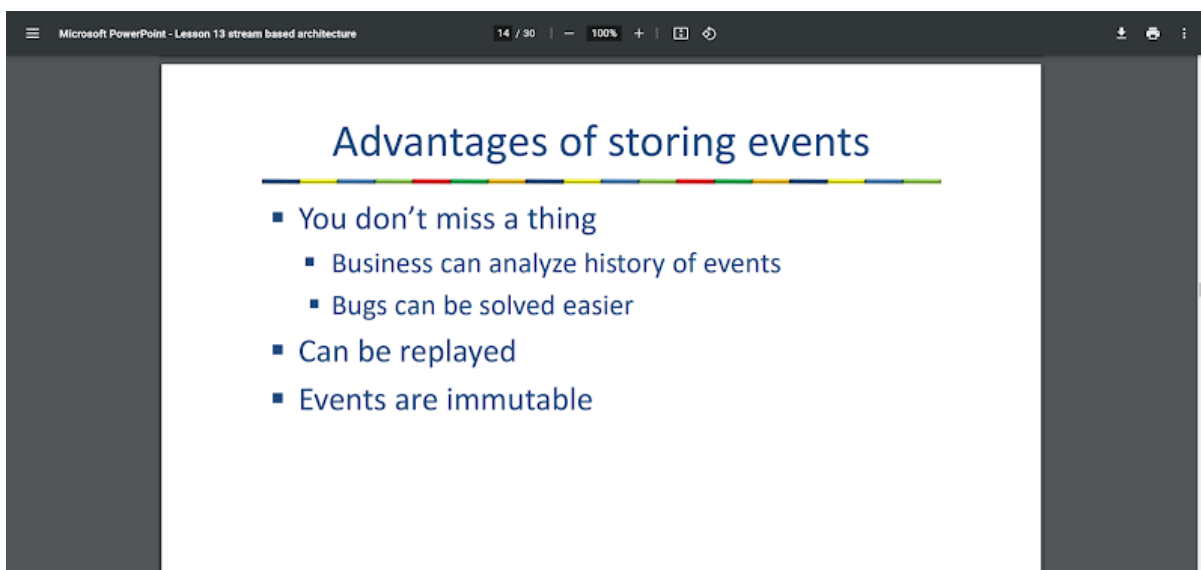
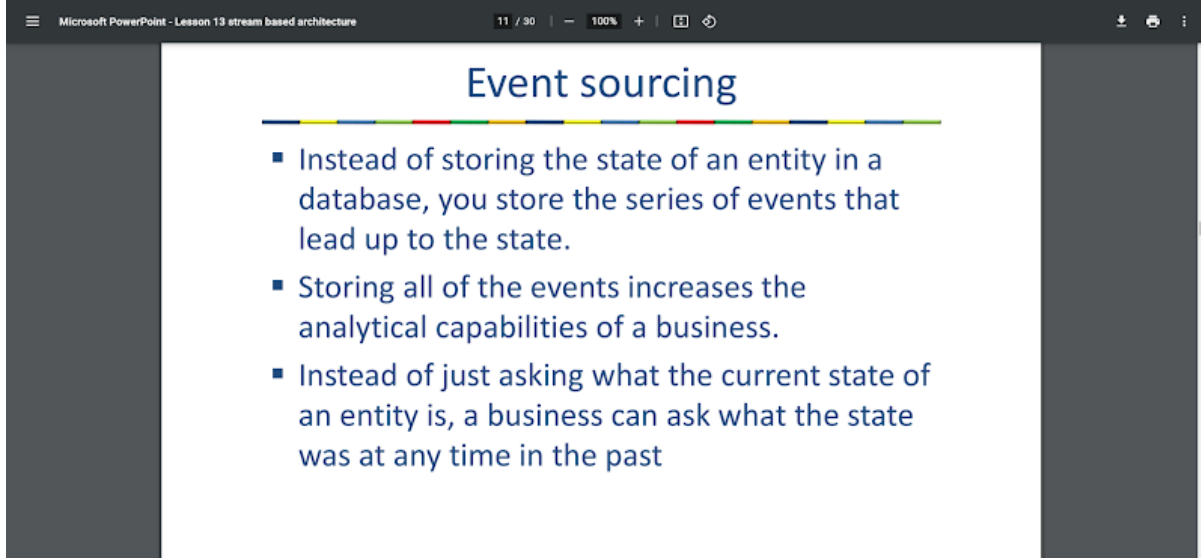
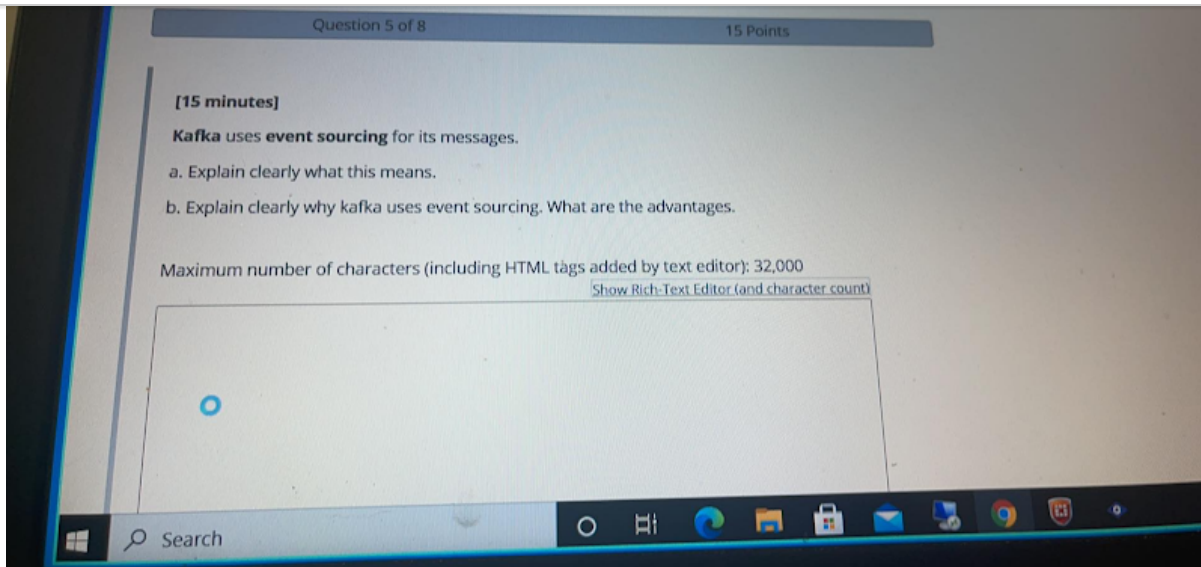
Rest => the thread will block until the web client receives the response.

=====

=====

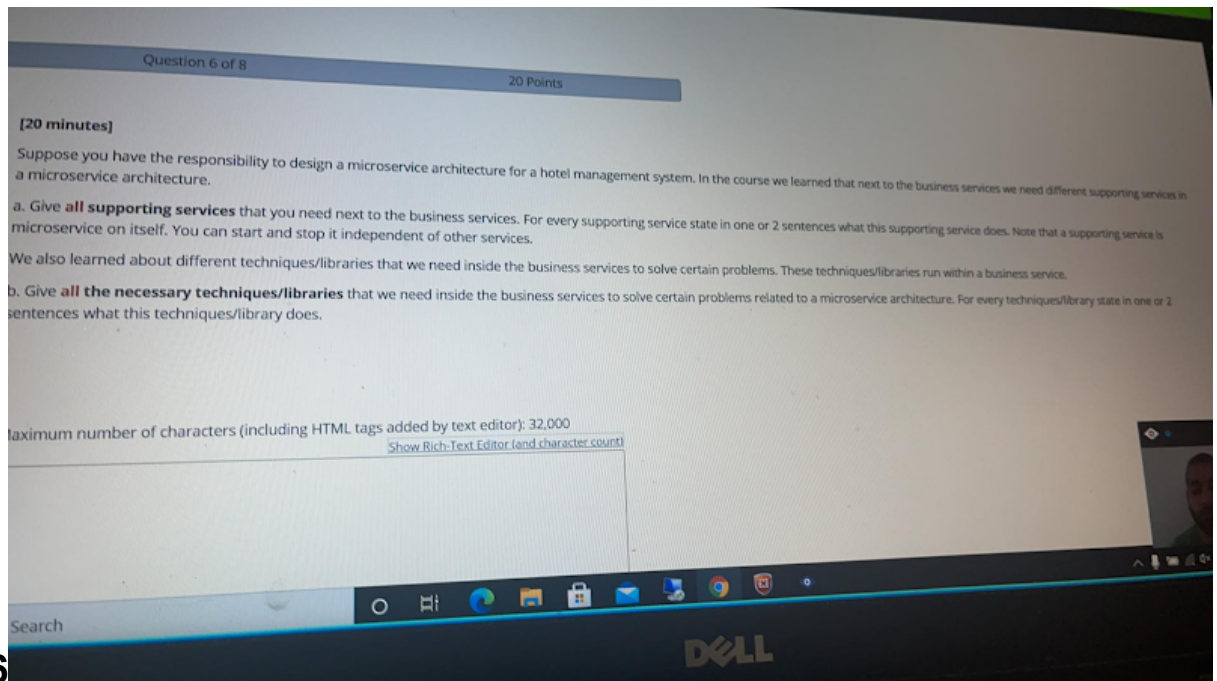
Question 5

← outstanding men



A- Event sourcing means that we store events instead of the state

← outstanding men



Question 6

- a. Hotel Service: responsible for managing hotels
 Auth Service: responsible for authorizing users based on roles
 Booking service: Responsible for managing bookings
 ((for your answer **And add more services if necessary**))

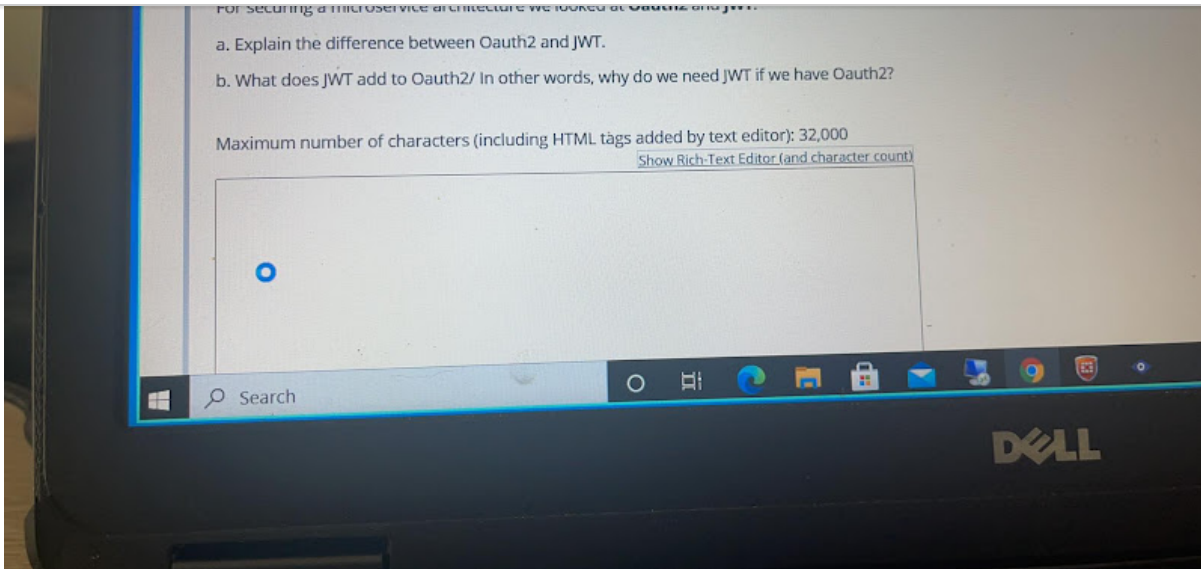
b.

API gateway	Loosely coupling between clients and services and crosscutting concerns
Config server	One central place to store config data for all services
Registry	Registering and lookups of service locations (URL, portnumber)
Authorization server	Token based security with OAuth
Resilience framework (Hystrix)	Implement resilience patterns like circuit breaker and bulkheads
Tracing server (Zipkin)	Centralized tracing server
Sleuth	Add unique ID's to trace info
ELK stack	Centralized collection, searching and monitoring of logging data
Client side load balancing(Ribbon)	Load balancing between service instances

Ya

Question 7

← outstanding men



A- explain

the difference between OAuth2 and JWT

- OAuth2 is a token based authorization framework but does provide a standard for tokens
- JWT (JavaScript Web Tokens) provides a standard structure for OAuth tokens:
 - Small
 - Cryptographically signed
 - Self contained
 - Extensible

B- what does JWT add to OAuth2 in other words, why do we need JWT if we have OAuth2 ?

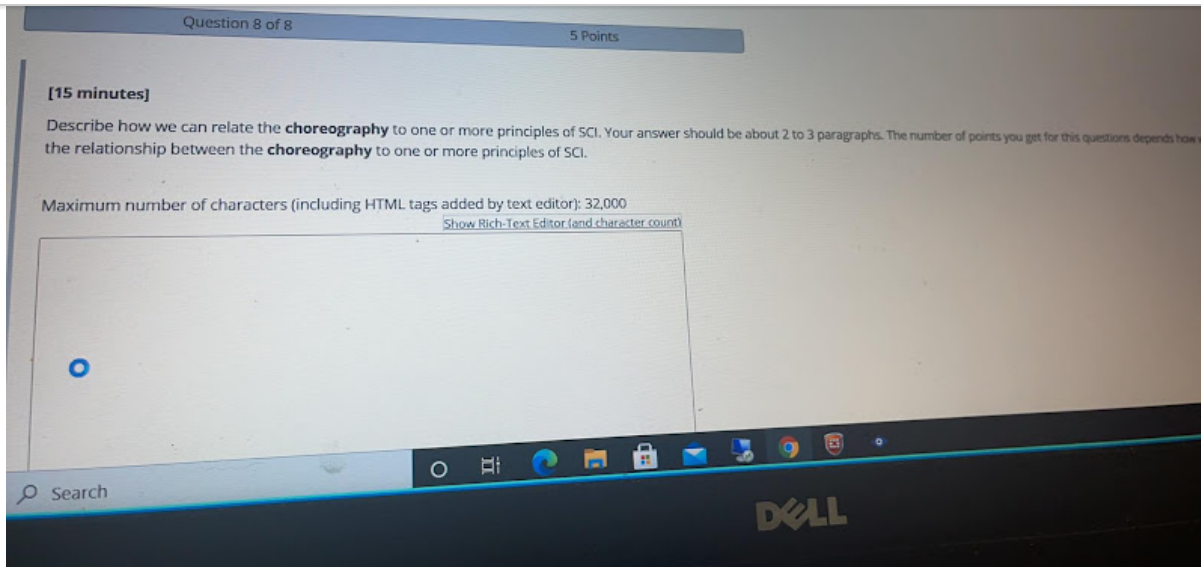
- JWT defines a standard for tokens. The token contains the role of the user and JWT will sign and encrypt the token, so that the receiver of the token can securely retrieve the role of the user without having to call the authentication server.

=====

=====

Question 8

← outstanding men



The whole is greater than the sum of its parts

Whole is contained in every part

Harmony exists in diversity

Life is found in layers

Order is present everywhere

Choreography

■ Advantage

- Less Coupling
 - Easy to add/remove services without impact on other services
- Fast: parallel processing
- No single point of failure

