I do not guarantee the correctness of the answers and calculation methods.

True or False

- With services computing, loose coupling indicates the degree of dependency. Web services are loosely coupled.
- Docker is not an example of an OS-Level Virtualization technology.
- Cloud computing is beneficial when considering the use case of batch computational workloads.
- Security concerns in IoT systems include non-repudiation.
- Multi-access Edge Computing (MEC) is typically characterised by low reliability and volatility.
- Software-as-a-Service and Platform-as-a-Service are better in terms of vendor lock-in than infrastructure-as-a-Service.
- Federated learning provides a solution to the problem of providing incentives to devices to behave well - for example by contributing local model updates to a global model.
- Within federated learning, submission of data by end-devices in a timely manner (often referred to as timeliness) is not an issue.
- A great advantage of Software-as-a-Service (SaaS) is that it can be always be customized to one's needs.
- Within an IoT-Cloud application and from a service provider perspective, the edge should typically be considered as trusted.
- Within services computing, a Message-oriented Middleware is an infrastructure for passing of data between applications using a common

communication channel where messages are sent and received in an asynchronous manner.

- It is more typical to execute a microservice in a virtual machine than in a container.

- LPWAN offers connectivity with higher throughput compared to WAN. Wi-Fi is an example of an LPWAN technology.

- Consider a supply-chain management system which involves interactions between multiple entities (e.g., customer, supplier, etc.), each one hosting a different set of web services. Web service choreography refers to managing the control and data flow within each entity's organizational boundaries.

- Within containerization, drivers for hardware are not shared.

- When network neutrality is not enforced, IoT service providers can team up with network operators so that their services' traffic is preferentially treated.

- According to the ETSI standard specifications for Multi-access Edge Computing, when a customer selects to create an application instance via a Customer Facing Service portal, the application package should have already been onboarded.

- Xen and Docker are both examples of OS-Level Virtualization technologies. (no Xen is Hardware-Level)

- Microservice architectures are characterized by a small invocation depth.

Single Choice

This specification allows to synthesize complex Web Services from smaller ones which are interacting.

- WS-Coordination
- **BPEL**
- UDDI

An IoT application is in charge of continuously collecting data from thousands of environmental sensors distributed over a country, performing some initial filtering of the data to detect faulty sensor readings, and storing them for future processing. Where would you host the data filtering functionality?

- **at the edge**
- on the cloud

A robot control application works as follows: It receives sensing information from a mobile robot which is connected to the Internet over a Wi-Fi link, processes the sensing data, and decides in real time on an appropriate robot control action which it sends to the robot (e.g., instructs the robot to change its direction to avoid an obstacle). Where would you execute the robot control application.

- **on an edge computer or the robot itself**
- on a cloud virtual machine

A cancer research institute wants to build a digital library of DNA (genome) samples from tens of thousands of patients for long-term research usage. This requires running a genome sequencing analysis on each of the samples independently. Assume that each individual sample analysis can be handled by a single large memory Amazon EC2 VM instance. A sample analysis can last for hours, but can be stopped and resumed arbitrarily. The institute's primary concern is to complete the sample library at the smallest possible cost. Based on the provided information, which EC2 instance type would you use to implement the described use case? Select the one that is most applicable.

- On-demand
- Dedicated
- Reserved
- Spot instance

Consider a smart parking service which works as follows. A camera is capturing images from parking spots and transmits them to a remote server at a rate of 10 images/s. The server performs an image processing task where it identifies if the parking spot is occupied and, if so, it detects the license plate of the respective car. Which networking technology would you select to connect the camera device to the remote server?

- 4G
- NB-IoT
- LoRaWAN

Multiple Choice

A provider of computational resources (e.g., the operator of an IT center offering computational resources to customers) would rather provide virtualized resources like Virtual Machines instead of offering Physical Machines for reasons of (select all statements below that apply):

- Backend parallelization
- Customer security
- Consolidation for energy consumption
- Multitenancy

Microservices are a variant of the service-oriented architecture (SOA) structural style. Select all of the statements below that are applicable.

- Microservices may feature individual data backends.
- Increased latency is a problem when adopting microservices.
- Microservices can be easily scaled.
- Adopting microservices eases the burden of integration testing.
- Microservices feature good fault isolation.

Open Questions

Name 2 advantages and 2 disadvantages of microservices.

Advantages

- Small understandable, clearly separated functionality units
- Lightweight, quick to deploy and high scalability
- Technology and team independence for each microservice

Disadvantages

- High inter-service communication (e.g., increased latency)
- Individual data backends can lead to duplication of data
- Automated deployment requires tooling support

Explain Loose Coupling and Stateless Service in 1-2 sentences each.

Loose Coupling: Coupling indicates the degree of dependency any two systems have on each other. For example, web services are loosely coupled since they connect and interact across the internet with different applications, but without the knowledge how those other applications behave or are implemented.

Stateless Service: Services can be invoked repeatedly without having to maintain context or state. For example, if the data was sent via a REST API and should be saved, then a database is needed since the application does not maintain a state.

Name 3 benefits of fog/edge computing.

Low latency, decentralization, less signaling and communication overhead

Name key differences between fog and edge computing.

Fog: wider scope, deeply-hierarchical, computation anywhere along collaborating entities all along the IoT cloud continuum.

Edge: typically spans up to the edge of the operators network, computation on-device or offloaded 1 hop away (other local devices, base stations)

Why should a provider use virtualization? What are the benefits for it?

It enables the provider to abstract the view on his real resources.

- Higher degree of capacity utilization
  - Multitenancy: resources are shared between users
  - Backend parallelization
- Consolidation
  - Many different classes of application onto different VMs in the same data center has positive effects for energy consumption and space usage.
- Fault tolerance
  - Save VM state and therefore possibility to migrate to somewhere else

You are the provider of a large social networking service. One of the many functions of this service is the generation of JSON Web Tokens, i.e., cryptographically-signed objects that can be used, e.g., for authenticating user requests to various API endpoints of your service. This function requires 100 MB RAM and is invoked 2.5 million times per month. Each time it is invoked, it runs for 0.1s. As a service provider, you have the following two options to host this function:

- Option 1: Lease an on-demand virtual machine from an laas provider and nost your function there. The cheapest offering which can fit your memory and other requirements costs 0.15 EUR/hour.
- Option 2 Deploy your function at a Faas provider in this case, the following charges apply: Compute charges 0.000001 EUR per MB-s. Request charges 1 EUR per million requests.

You need to select the option with the minimum monthly cost (Assumptions: A month has 30 days and both options there is no free tier.)

(1 point) Which of the two options has the minimum monthly cost? (1 or 2)

(3 points) What is the minimum monthly cost? Please enter a numerical value.

You have one authentication interface with an availability of 0.99, one authentication server with an availability of 0.98 and one database server with an availability of 0.8.

a.) Calculate the composite availability.

0.99 * 0.98 * 0.8 = 0.77616

b.) What could you improve to get a composite availability of higher than 0.95?
Add two more database servers (goes from 0.8 to ~0.99)

The figure shows a redundant cloud deployment for a web application. The application can function properly if the load balancer is working, as well as at least one web server and at least one database server. (figure shows 1 load balancer, 2 web servers and 3 database clusters).

Assume that the individual component availabilities are as given below:

- availability(load_balancer)=0.98
- availability(web_server)=0.99
- availability(database)=0.93

What is the overall availability (A) of the system?

- A > 0.78 and A < 0.97
- A < 0.78
- **A > 0.97**

Calculation:

av(2*web) = 1 – 0.01^2

av(3*database) = 1 – 0.07^3

av(all) = 0.98 * 0.9999 * 0.999657 = 0.9795

Explanation of Terms

Universal Description, Discovery and Integration (UDDI) is a platform-independent, Extensible Markup Language protocol that includes a (XML-based) registry by which businesses worldwide can list themselves on the Internet, and a mechanism to register and locate web service applications. UDDI is an open industry initiative, sponsored by the Organization for the Advancement of Structured Information Standards (OASIS), for enabling businesses to publish service listings and discover each other, and to define how the services or software applications interact over the Internet.

Web Services Coordination (WS-Coordination)

- General mechanism for coordinating multiparty, multi message Web service tasks.
- describes an extensible framework for providing protocols that coordinate the actions of distributed applications. The framework enables existing transaction processing, workflow, and other systems for coordination to hide their proprietary protocols and to operate in a heterogeneous environment. Additionally this specification describes a definition of the structure of context and the requirements for propagating context between cooperating services.

BPEL (Business Process Execution Language)

- Provides orchestration for Web services
- Definition and execution of business processes
- <u>Allows the recursive creation of larger Web services from smaller ones</u>
- is an OASIS[1] standard executable language for specifying actions within business processes with web services. Processes in BPEL export and import information by using web service interfaces exclusively. WS-BPEL aims to model the behavior of processes,[2] via a language for the specification of both Executable and Abstract Business Processes. By doing so, it extends the Web Services interaction model and enables it to support business transactions.

WS CDL (Web Services Choreography Description Language)

- describes how services work together

WS Addressing

- Interoperable, transport independent way of identifying message senders and receivers.

WS Policy

- Define constraints, conditions, service level assurances and requirements
- Attach these policies to Web services using WS PolicyAttachment

Use Cases for Cloud Computing

- Demand for a service varies with time (peak loads)
- Demand is unknown in advance (e.g. startup)
- Batch workloads (e.g. 1000 virtual machine instances for one hour cost the same as one instance for 1000 hours)

Security concepts

- **Non-repudiation**
  - Information and actions cannot be denied, once performed (e.g., payments)
  - A service that provides proof of the integrity and origin of data.
  - An authentication that can be said to be genuine with high confidence.
- **Confidentiality**
  - Information should be kept secret (e.g., between IoT devices, IoT edge, IoT-cloud)
- **Integrity**
  - Information should not be modified (e.g., in transit, at rest)
- **Availability**
  - Service or data are provided when needed (e.g., due to DoS attacks)

    - Information originates from an identified source (e.g., app users)

Multi-access Edge Computing (MEC)

- Architectures and interfaces to standardize:
    - The deployment of 3rd party applications at the edge of the 4G/5G mobile network
    - Ways these apps can provide and consume edge services

- Tailored to a network operator-controlled edge infrastructure

- Consumer-oriented services
    - AR/VR, computation offloading (e.g., mobile gaming), multimedia services, …
- Operator and 3rd party-oriented services
    - Radio network monitoring/troubleshooting, video analytics, connected vehicles, …

- Critical enabler for 5G. Why?
    - Enables Ultra Reliable and Low Latency Communication (URLLC) and enhanced Mobile BroadBand (eMBB) services

- Important difference with traditional clouds: end-user context available to edge apps

IaaS vs. PaaS vs. SaaS

- SaaS is already there to use, but cannot always be customized fully to one's needs
- PaaS applications can be developed faster than IaaS (e.g., one does not need to care about load balancing and networking)
- but IaaS offers highest customization possibilities
- **Cost:** PaaS can be cheaper to run, because a cloud system is optimized for efficiency (multi-tenancy)

- **Vendor lock-in:** SaaS and PaaS worse than IaaS

Federated Learning

<u>Research Challenges</u>

- **Device recruitment strategies:** Which subset of the devices to assign a learning task at any given round? Processing, storage, battery, trustworthiness, data quality and other criteria to consider
- **Volatility:** Devices can "disappear" or join at any time
- **Asynchrony:** Algorithms face challenges when end devices do not submit their data in a timely manner
- **Non independent and identically distributed data:** inaccuracies, personalization lost
- **Heterogeneity in the volume of training data per device:** A device that contributes a lot may lead to a biased model
- **Preventing privacy leaks:** Some private information may be inferred even if devices do not transmit the actual data
- **Incentives to misbehave:** Why waste battery when I can let the others do all the work?

Low Power Wide Area Networking (LPWAN)

- Event-driven or periodic data transmission
    - Small packets, typically infrequently (e.g, order of seconds, minutes, or even more)
    - Low bandwidth requirements
- Very large number of devices -> networking hardware should be cheap
- May need long-range wireless connectivity
    - Bluetooth, Wi-Fi, ZigBee not enough
- Should operate at (very) low power
    - 4G more power-hungry

1. **LoRa**
    - Proprietary PHY
    - low cost
    - operates in unlicensed spectrum
    - LoRaWAN: open specs for the higher network layer

2. **SigFox**
    - Propriertary PHY
    - Very low bandwidth
    - Unlicensed spectrum

3. **LTE-M, NB-IoT**
    - Response of the operators
    - 4G / 5G support