Edge and fog Computing Assignment 1 - m22cs007 A.Jyothi

**Q.1 SUMRAM**

Ans1.

Concerns:

Since there are many natural disasters, Network connectivity is often lost. Due to this, the systems will not be available.

1. Since there is no reliable connection, the data can also be lost.
2. Since it is a poor country, there are not enough resources nor can they spend on setting up and managing the systems with backup.
3. We cannot scale the systems.
4. There is a chance to lose important information. So, privacy and security concerns arise.
5. In order to enhance the educational sector, we need to accommodate the systems with good backups and scale them to the maximum extent possible. But scalability is an issue in such cases.
6. Data collection strategies will be different if there are frequent losses of data, and thereby, to award degrees, it becomes difficult to compare.
7. If they are using blockchain technology, it should be carefully designed, knowing all the drawbacks.
8. The system should be designed in compliance with SUMRAM country's rules and regulations.
9. There are no rules as to who will maintain the systems, or take charge of the data governance. That is very important for any system maintenance.

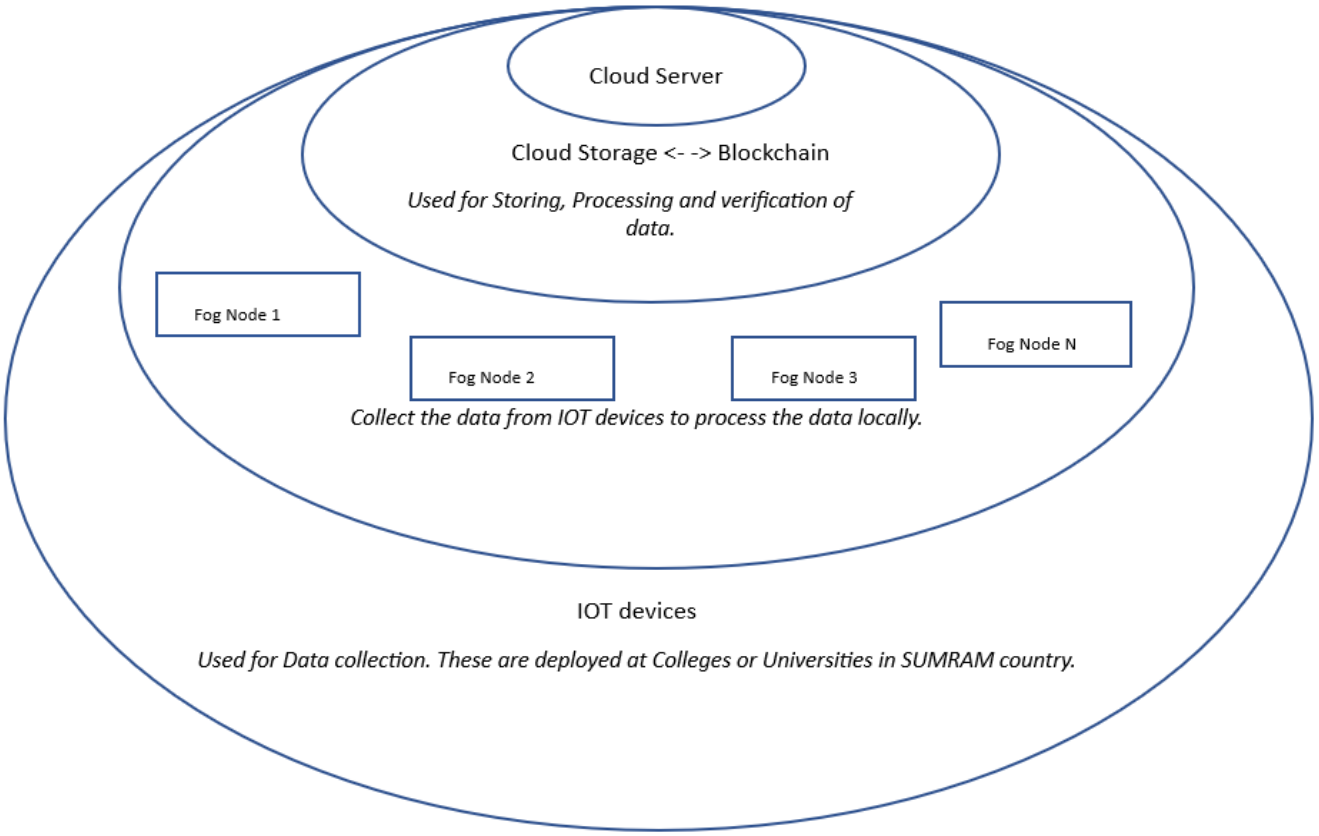
Assumptions:

> As an Enterprise cloud data architect, I need to design a solution which will satisfy all the SUMRAM's education requirements. So, these are what I assume that SUMRAM country have in available.

1. First, the Government is having a clear picture of how to design their system, like what data needs to be collected, where they will be using it etc.
2. They are not that poor to have backups, or maintaining and setting up the systems and resources, and that they can provide necessary infrastructure.
3. People are educated enough to use the systems, and maintain them also.
4. The technologies we use are reliable and secure. (Security and privacy concern)
5. The system developed will be on par with existing educational systems, and we can modify or update the system when and where required. (Network and availability concern). Like discussed in class, we can use SDN for dynamic network changes.

Ans2.

Blockchain and cloud-based solution:




(Screenshot taken from my drafted word document, attaching the same for reference)

First, we have to collect the data (like biometrics) from IOT devices and text data for personal information.These will be installed in colleges or universities to collect the student details like marks, name etc.

Once these details are collected, we have to send them to local Fog devices. They will process the data. Here processing means filtering the important information from the data collected, and sending it to cloud storage. These nodes will also have the copies of blockchain. These local nodes can reduce the data transmission over the network to a centralised server. Thus it will decrease the network disruptions and improve the performance of the system. In this layer, the networking also is taken care of.

Now that we have filtered it, we can store this data as a backup in the cloud storage locally, at different locations, and then , using Blockchain we can verify the details of the students. It serves as a security tool. Since it is a blockchain, a distributed ledger, all the fog nodes will have access to

it, so even in case of data loss, or network disruptions, we can make use of it.Thus we can encrypt the student data and store it to local cloud storage and blockchain.

Then, the cloud server can have the global storage of all the college details and student details, it also uses blockchain to verify student data globally and award degrees. As well it can be used to store standardised data and process it. This can be maintained by the Ministry of Education.

Maintaining huge servers may be a complex task, but if , as assumed, there are highly skilled professionals and resources for funding IT infrastructure, and technical expertise in these fields, then it can be managed.Since we are using Blockchain technology, multiple parties will have the copies of student records., and sharing this student data should be taken care by colleges or Universities.

Ans 3:
Features offered by this solution:

1. IOT devices: Since we are using IoT devices to collect the data, we can capture Real time data easily and effectively. IoT includes sensors, actuators etc. So, using them, we can capture students' face images, biometrics, etc easily. And then link these to the textual data to generate records. This will be a secure way of capturing the details.
2. Not only grades are important for awarding degrees, many aspects like attendance and activeness in the class are also considered in many institutions, so using these IoT devices, smart classrooms can be designed.
3. Accuracy and scalability can be improved.
4. Fault tolerance can be achieved using blockchain - IoT-cloud solution.
5. Fog Nodes: Using them, we can compute and process the data easily and locally. Data need not be transferred to the central server for processing. So this way, data transmission and processing time can be reduced, which implies Quick Processing.
6. And since multiple fog nodes are being used, the Availability factor is always met.
7. Cloud storage and Blockchain: Security aspect can be met using Blockchain.
8. And, incase of data loss or power outages, this distributed ledger helps.
9. Cloud is central storage with distributed computing. So, different types of colleges will have different ways of marking and awarding degrees, so that can be possible with cloud.
10. Since it is managed by the central Government, the security is met.
11. Integrity and authenticity is met.
12. There will be a structure to maintain all this.

Ans 4:

Limitations:

1. This can be complex to build.
2. As I assumed that the Government of SUMRAM will give enough resources for development, I came up with an architecture that supports many features, but if it's not the case, it will be an expensive solution to build.
3. Fog nodes and IOT devices have less computational power, so if data flows in highly, it can't handle the loads properly.
4. If many colleges get added up, the system cant scale to large extents.
5. Blockchain might be secure, but sometimes there can be security issues also
6. Data can be encrypted and securely stored in blockchain, but privacy issues may arise.
7. Sometimes, we can't overcome network connectivity issues or power outages.

Ans 5:

1. Requesting users to “Register” to the site developed.
2. In the Registration step, we can ask for their personal details like phone number, email address, and even the biometrics ( for extra security).
3. When a phone/mobile number is asked, we can generate OTPs and ensure that the provided number is valid, and belongs to the user., like Multi- factor authentication.
4. This step can be repeated for every login. So, we can generate different OTPs to phone and email separately. Ask to sign in using different OTPs.
5. We can encrypt the data while using the app or site. If there is sensitive data, then we can use even more security steps in variations and combinations of OTP with data of birth etc.
6. However, we should allow only Read only access. If any modifications are required, that can be done in person only. Thus we can ensure that the marks or grades cant be updated or changed by the users.
7. In these ways, we can ensure security and authentication to our sites or apps that we develop.

Ans 6:

Tracking:

1. We can monitor the log in activities from different devices, locations and IP addresses.
2. Not only locations, we can keep track of updates that are being made in the system by the user.
3. Any suspicious activity can be notified to the team that handles all these changes.

Controlling:

1. Educating people about not sharing otps , passwords etc.
2. Firewalls protection for unauthorised access.
3. We can keep checking log in activities on the website.

4. Allowing only some functionalities to be updated by the 3rd party users, like name, phone number etc (personal details only), and not giving options to edit or update their marks and educational details.

Charging:

1. We can ask users to agree to terms and conditions in order to use the app or website.
2. In the terms and conditions, we can mention about the tracking, controlling and payment measures.
3. If the user wants to update their own details, they will be charged and will be required to upload the documents related to the updations.
4. If they want to see additional features as to the app, they will be charged accordingly.

Ans 7:

References:


1. Xiaoshuang He, Hechuan Guo, Xueyu Cheng, "Blockchain-Based Privacy Protection Scheme for IoT-Assisted Educational Big Data Management", *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 3558972, 11 pages, 2021. https://doi.org/10.1155/2021/3558972
2. Chaiyarak, Sakchai, Alongkorn Koednet, and Prachyanun Nilsook. "Blockchain, IoT and fog computing for smart education management." *International journal of education and information technologies* 14.1 (2020): 52-61.
3. https://www.geeksforgeeks.org/benefits-and-applications-of-blockchain-in-cloud-computing/
4. https://www.google.co.in/books/edition/Fog_and_Edge_Computing/gMOCDwAAQBAJ?hl=en&gbpv=1 wiley textbook preview.
5. Muhammad Rizwan Anawar, Shangguang Wang, Muhammad Azam Zia, Ahmer Khan Jadoon, Umair Akram, Salman Raza, "Fog Computing: An Overview of Big IoT Data Analytics", *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 7157192, 22 pages, 2018. https://doi.org/10.1155/2018/7157192
6. General social media working like gmail, instagram, myntra for 5th,6th questions.

Q.2:
Paper " Kubernetes in Fog Computing: Demonstration, Limitations and Improvement Scope"

This paper speaks about how the kubernetes is being used in the Fog computing environment. As there cannot be many fog nodes, we will be using containers in the fog devices to run multiple applications. To maintain, manage and run all these containers we need some tools , generally they are called container orchestration tools. Kubernetes is one such tool. This scheduling model is improved by Ant colony algorithm and particle swarm optimization algorithm. There were some alterations like KubeEdge, MicroK8s, K3S, and FLEDGE. They could not do justice as they work on limited resources. There are mainly two ways of allocating the application. They are Application deployment on a single fog node or distribution across a network of communicating fog nodes. Kubernetes deploys and manages such applications easily.It uses a master-slave architecture in which the master node creates and runs the cluster while the worker nodes are under its control. Master node acts as the entry for administrative tasks. It initialises and manages the cluster. While the slaves are controlled by the master after they join the cluster. Kubernetes nodes are physical devices and run like docker. Pods and services are the objects of K8. Pods are basic control and management units of k8. It can have multiple containers. Services abstract the logical set of pods and expose them as apps. K8s use 1 IP address per pod.Scheduling (Kube-Scheduler)and networking are the two main parts of the Kubernetes orchestration system. Nodes are filtered out and assigned some scores. The networking can be between containers or between pods.The communication between the containers placed on various nodes is made possible by the weave net CNI plugin. This existing kubernetes model is not versed with the distributed fog computing model. So we can improve it in scheduling and master-slave aspects. There are some limitations in the KS algorithm, which we can overcome. Three plugin extensions: the Queue Sort Plugin, the Split and Distribute Extender Plugin, and the Topology Aware Scoring Extender Plugin can be implemented. These improvements will provide more flexibility to application developers and support the deployment of a wide variety of IoT applications. The authors deployed a NodeJS IoT program using a master-slave Kubernetes architecture, using a PC as the master and four Raspberry Pi 3 as workers. The distribution on various nodes in various pods versus placing all Docker containers in the same pod were the two placement methods that were compared. Despite network delays, distribution induced time lags without affecting program functionality. So the authors concluded the Kubernetes extensions for containerized app deployment in fog computing. Experimental results show feasible distribution of microservices on Raspberry Pi nodes without significant penalty.

2. The article is on edge computing. It involves processing data locally on devices at the of the network, and is an important factor to take into account when administering IoT apps in settings with constrained network bandwidth and latency. This is because they are discussing

the difficulties in managing and deploying IoT applications in environments with limited resources and recommending containerization and auto-scaling methods to deal with these difficulties.

3. Containerized applications are better than traditional virtualization in the context of IoT application scaling. Few benefits are here:

1. Since containers don't need hypervisors like Vms, they need less resources and thus improve performance.
2. VMs will need a complete OS to be loaded. But the containers will need only necessary libraries to run. Thus they are faster.
3. We can run different applications on a single host in containers. In VMs we can run only one application per VM.
4. Containers can offer better isolation, like applications running in one container do not interfere with applications running on other containers.
5. Cost effective approach.
6. Containers need less memory space than virtualization
7. Containers are often short-lived, and they run directly on the kernel without booting. VM processes are long-lived, and they need booting.
8. We cannot run  more than 2,3 Vms on an average running laptop. But we can run many containers on a normal laptop.
9. Autoscaling can easily add more resources like containers or servers without affecting performance.
10. Resources are used effectively when we auto scale. Because it takes  dynamic load into account.
11. Deploying applications across regions of availability improves application reliability.
12. Better fault tolerance can be achieved through auto scaling.
13. Performance and availability are improved.

4. This is the video implementation: https://www.youtube.com/watch?v=k52aBh6fCA8