Distributed Systems Programming (DSP620S)

Assignment

Due on 21/10/2019

Problem 1.

Mark: 50

In this project, your task is to design and implement `Notes`, a notice board application using a micro-service architectural style. The application should *store* new notices as well as *display* the notices for a given day, week or month.

The application will use one single type of service, `Noter`. We will deploy five instances of the service, each with its own data store. The service instances will communicate with an API gateway and execute the aforementioned operations.

To submit a notice, a user sends an API call to the API gateway with a notice object. The latter comprises of the identifier of the notice, its topic, description and date of submission of the notice. Once the API gateway receives a call to submit a notice, it randomly identifies an instance of `Noter` and directs the call to it. When a service receives a call to submit a notice, it generates a ledger from the received notice. A ledger consists of a piece of data (here the notice), a hash[1] of the data, and the hash of the previous ledger. After creating the ledger, the service stores it locally and then start a *gossip* protocol until all five instances have received and validated[2] the new ledger. Once a service validates the ledger, it stores it locally. During the gossip protocol, a service can only interact with a single service at a time using an `HTTP` protocol.

To retrieve the notices of a day, week or month, a specific API call is placed, where a random service is selected to answer the call.

You will design `Notes` and implement it using the *Ballerina* programming language. You will use `Graphql` for your API gateway. You will also choose

[1] You will use the $sha - 512$ algorithm

[2] The validation entails checking the hash of the previous ledger and the signature of the current one

and configure a storage system for each service. Note that the storage systems might differ per service.

You will deploy your application using a `Kubernetes` cluster, with each service running in a `Docker` instance.

2. Submission Instructions

- This assignment is to be completed by groups of four (4) students each.
- You will submit your assignment in the folder named DSP620S_2019_Assignment2 on the E-Learning platform.
- The submission date is Monday, October 21^{st}, 2019. No late submission will be accepted, and no other submission mode will be considered valid.
- Any student who fails to submit on time will be awarded the mark 0.
- Your submission should only contain the source code files and the additional resources needed to run your program.
- There should be no assumption about the execution environment of your code. It could be run using a specific framework or simply on the command line.
- In the case of plagiarism, all submissions involved will be awarded the mark 0, and each student will receive a warning.