Question 2: Concept or approach to be mentioned.

(i) In addition to the continuous integration environments, it is a general practice for enterprise applications to also have a monitoring environment. This monitoring application helps us to keep track of what builds are running, statistics about how many builds have passed for the day/month/year and how many have failed, average runtime of a build, etc.

Approach:

- There are lot of wonderful opensource monitoring tools available in the market, I would suggest using Grafana monitoring and alerting solution.
- Grafana supports almost every corner of the monitoring metrics (including custom metrics) and can visualize in wide varieties of graphs and custom dashboards.
- Grafana supports alerting to its wide varieties of channels (Pages duty, Teams, Emails, Calls, SMS, etc.) when monitoring threshold reaches to trigger points.
- Alternative solution: Elasticsearch with Kibana dashboards

(ii) Given that the source is Jenkins (the CI environment) and the destination is some sort of data store (which candidate has to propose) and assuming that a data processing layer is also required to process data between the source and destination, kindly propose a solution on how this system could be designed.

Approach:

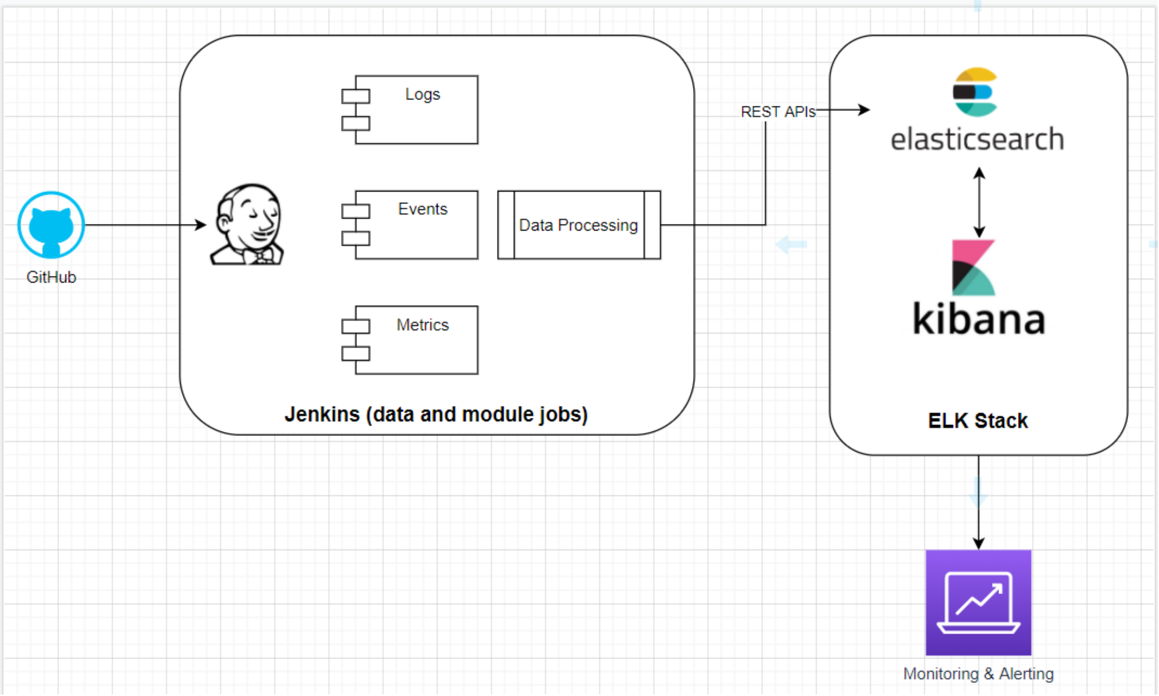
- I am assuming the data store is some kind of SQL/NoSQL based server or it can a file storage/sFTP server.
- Source is Jenkins CI env:
    - We can implement any kind of data processing modules/logic in any of the programming lang (java/python/etc.) inside the same project.
    - At the CI builds, we can run these data processing logics and store data in destinations.
    - The sensitive information/configurations can be stored in Jenkins credentials or some sort of secrets store (remote)
- Again, there are lot of ways to do this, when comes to best practices the actual requirement needs to be understand first.

(iii) Please delve into as much detail as possible, for example, name the open source components that would satisfy the condition of data store and data visualization, how would you move the data from source to destination, will the data be processed synchronously or asynchronously, etc. A sketch of the system architecture would be appreciated.

Approach:

- Tools:
    - Hadoop, one of the best distributed storage platforms for data processing

  - PCF Cloud Platform: Storage services, comes with lot of integrations.
  - OpenIO: very good object storage solution for on-prem, cloud or Edge locations
  - And Many more are there, need to understand first what type of data and it’s usage, to provide better solution
- Visualization:
  - ELK, Dynatrace and Grafana
- Data processing can happen in both async/sync, depends on amount of data and source/destination capabilities.
- We can also use streaming capabilities (Kafka) to process more real-time data synchronously.
- I’m assuming below architecture to process the logs/events/metrics data collected from Jenkins CI env to Elastic search and visualize them in Kibana dashboards: (excuse me for very high level details)



(iv) The sketch could be done using MS paint, PPT or could be drawn in a book and a snapshot of that can be attached. Please note that we are not looking for any fancy diagram. It can be as rough a sketch as possible as long as the architecture works. The expectation is just a description in candidate’s own words of the approach they would take.

Approach:

- I hope above response is more sufficient to this question, Thanks 😊