MCIA-LEVEL-1 Q&As

MuleSoft Certified Integration Architect - Level 1

Pass Mulesoft MCIA-LEVEL-1 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/mcia-level-1.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Mulesoft Official Exam Center

QUESTION 1

An organization designing a hybrid, load balanced, single cluster production environment. Due to performance service level agreement goals, it is looking into running the Mule applications in an active-active multi node cluster configuration. What should be considered when running its Mule applications in this type of environment?

A. All event sources, regardless of time , can be configured as the target source by the primary node in the cluster

B. An external load balancer is required to distribute incoming requests throughout the cluster nodes

C. A Mule application deployed to multiple nodes runs in an isolation from the other nodes in the cluster

D. Although the cluster environment is fully installed configured and running, it will not process any requests until an outage condition is detected by the primary node in the cluster.

Correct Answer: B

QUESTION 2

One of the backend systems involved by the API implementation enforces rate limits on the number of request a particle client can make.

Both the back-end system and API implementation are deployed to several non-production environments including the staging environment and to a particular production environment. Rate limiting of the back-end system applies to all non-production environments.

The production environment however does not have any rate limiting.

What is the cost-effective approach to conduct performance test of the API implementation in the non-production staging environment?

A. Including logic within the API implementation that bypasses in locations of the back-end system in the staging environment and invoke a Mocking service that replicates typical back-end system responses Then conduct performance test using this API implementation

B. Use MUnit to simulate standard responses from the back-end system. Then conduct performance test to identify other bottlenecks in the system

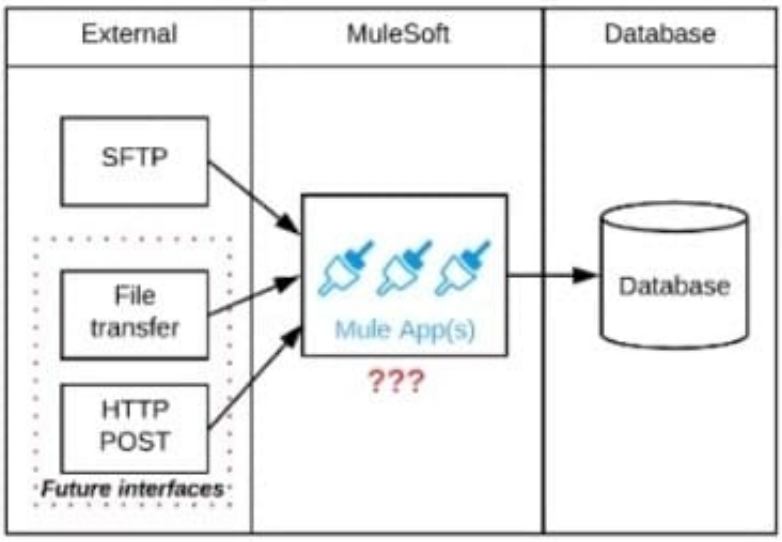
C. Create a Mocking service that replicates the back-end system\\'s production performance characteristics Then configure the API implementation to use the mocking service and conduct the performance test

D. Conduct scaled-down performance tests in the staging environment against rate-limiting back-end system. Then upscale performance results to full production scale

Correct Answer: C

QUESTION 3

Refer to the exhibit.



A business process involves the receipt of a file from an external vendor over SFTP. The file needs to be parsed and its content processed, validated, and ultimately persisted to a database. The delivery mechanism is expected to change in the future as more vendors send similar files using other mechanisms such as file transfer or HTTP POST.

What is the most effective way to design for these requirements in order to minimize the impact of future change?

A. Use a MuleSoft Scatter-Gather and a MuleSoft Batch Job to handle the different files coming from different sources

B. Create a Process API to receive the file and process it using a MuleSoft Batch Job while delegating the data save process to a System API

C. Create an API that receives the file and invokes a Process API with the data contained In the file, then have the Process API process the data using a MuleSoft Batch Job and other System APIs as needed

D. Use a composite data source so files can be retrieved from various sources and delivered to a MuleSoft Batch Job for processing

Correct Answer: C

Explanation:

*

Scatter-Gather is used for parallel processing, to improve performance. In this scenario, input files are coming from different vendors so mostly at different times. Goal here is to minimize the impact of future change. So scatter Gather is not the correct choice.

*

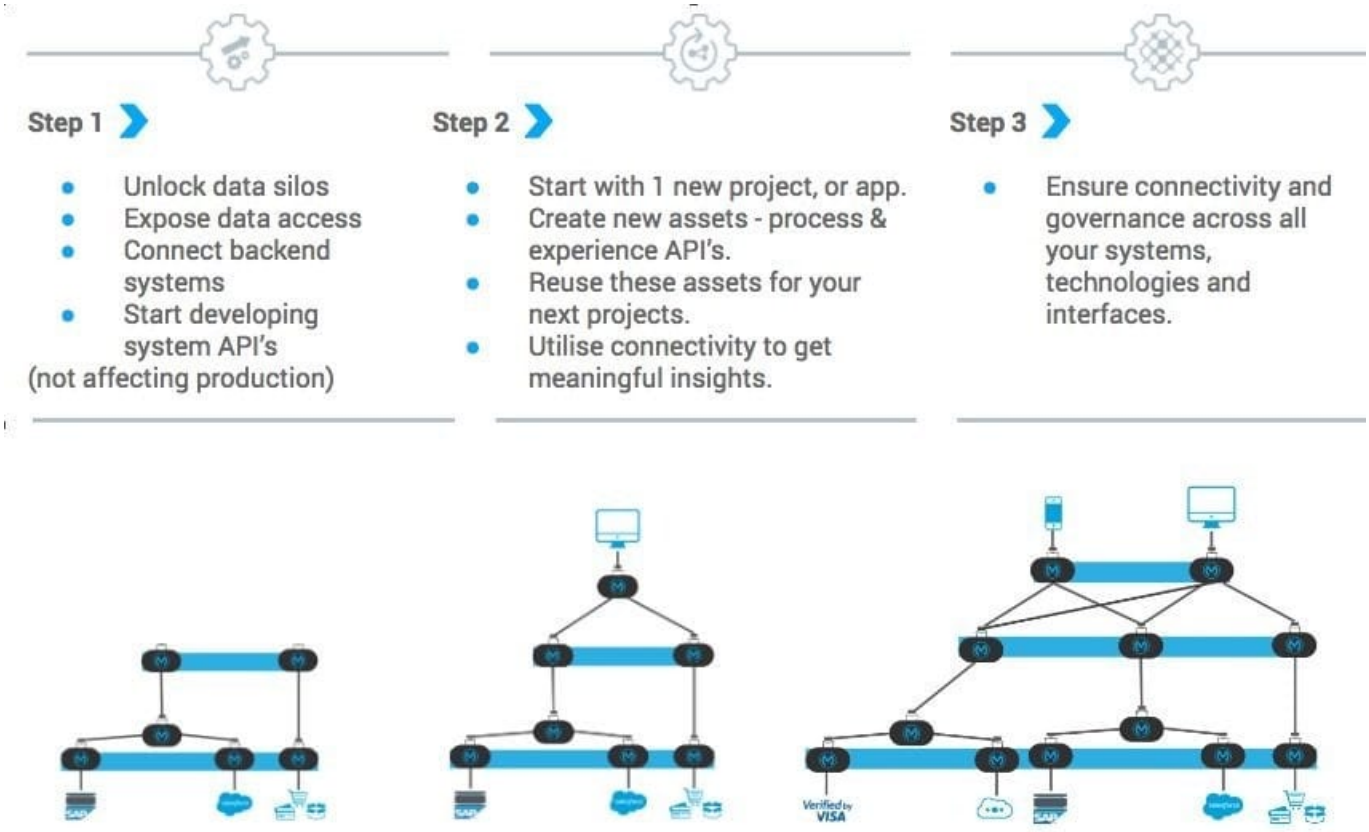
If we use 1 API to receive all files from different Vendors, any new vendor addition will need changes to that 1 API to accommodate new requirements. So Option A and C are also ruled out.

*

Correct answer is Create an API that receives the file and invokes a Process API with the data contained in the file, then have the Process API process the data using a MuleSoft Batch Job and other System APIs as needed. Answer to this question lies in the API led connectivity approach.

*

API-led connectivity is a methodical way to connect data to applications through a series of reusable and purposeful modern APIs that are each developed to play a specific role ?unlock data from systems, compose data into processes, or deliver an experience. System API : System API tier, which provides consistent, managed, and secure access to backend systems. Process APIs : Process APIs take core assets and combines them with some business logic to create a higher level of value. Experience APIs : These are designed specifically for consumption by a specific end-user app or device. So in case of any future plans , organization can only add experience API on addition of new Vendors, which reuse the already existing process API. It will keep impact minimal.



QUESTION 4

In which order are the API Client, API Implementation, and API interface components called in a typical REST request?

A. API Client > API implementation > API Interface

B. API interface > API Client > API Implementation

C. API Client > API Interface > API implementation

D. API Implementation > API Interface > API Client

Correct Answer: A

QUESTION 5

An organization has implemented the cluster with two customer hosted Mule runtimes is hosting an application.

This application has a flow with a JMS listener configured to consume messages from a queue destination. As an integration architect can you advise which JMS listener configuration must be used to receive messages in all the nodes of the cluster?

A. Use the parameter primaryNodeOnly= "false" on the JMS listener

B. Use the parameter primaryNodeOnly= "false" on the JMS listener with a shared subscription

C. Use the parameter primaryNodeOnly= "true" on the JMS listener with a non-shared subscription

D. Use the parameter primaryNodeOnly= "true" on the JMS listener

Correct Answer: A