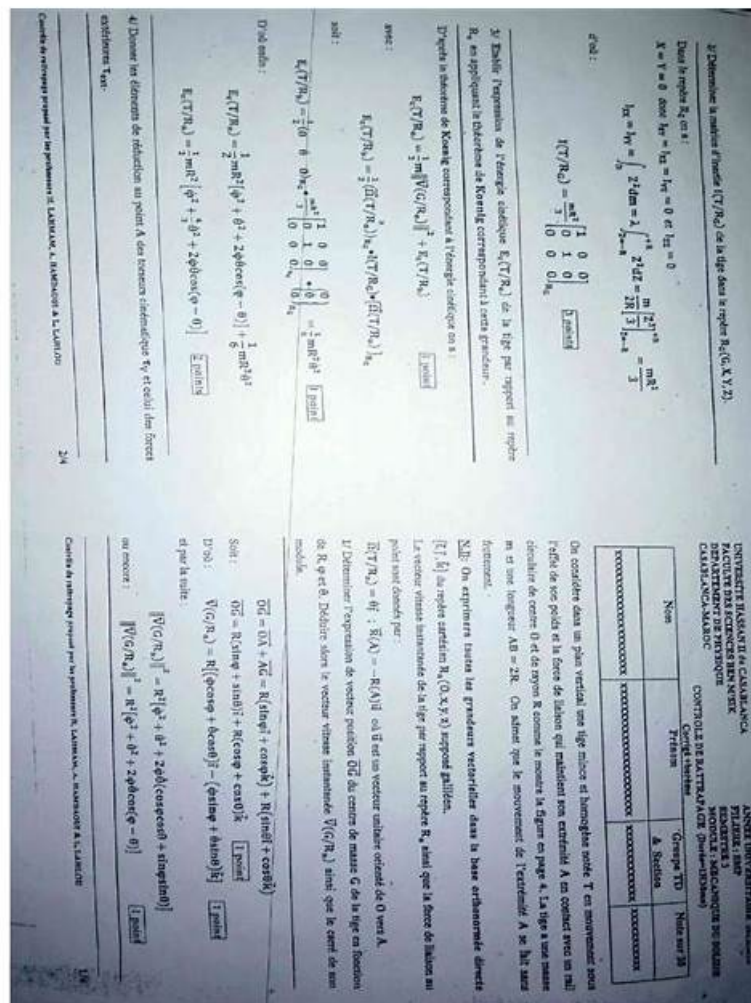


Mule-Arch-201 Exam | Mule-Arch-201 Formal Test



FreeDumps's Salesforce Mule-Arch-201 Exam Training materials is no other sites in the world can match. Of course, this is not only the problem of quality, it goes without saying that our quality is certainly the best. More important is that FreeDumps's exam training materials is applicable to all the IT exam. So the website of FreeDumps can get the attention of a lot of candidates. They believe and rely on us. It is also embodied the strength of our FreeDumps site. The strength of FreeDumps is embodied in it. Our exam training materials could make you not help recommend to your friends after you buy it. Because it's really a great help to you.

Our FreeDumps can help you realize your dream to pass Mule-Arch-201 certification exam by providing Mule-Arch-201 test training materials. Because it concludes all training materials you need to Pass Mule-Arch-201 Exam. Choosing our FreeDumps can absolutely help you pass Mule-Arch-201 test easily, and make you become a member of elite in IT. What are you waiting for? Hurry up!

<< Mule-Arch-201 Exam >>

First-Grade Mule-Arch-201 Exam & Leader in Qualification Exams & Useful Mule-Arch-201: Salesforce Certified MuleSoft Platform Architect

With the help of the Salesforce Mule-Arch-201 brain dumps and preparation material provided by FreeDumps, you will be able to get Salesforce Salesforce MuleSoft certified at the first attempt. Our Salesforce experts have curated an amazing Mule-Arch-201 exam guide for passing the Mule-Arch-201 Exam. You can get the desired outcome by preparing yourself from the Mule-Arch-201 exam dumps material provided by FreeDumps. We frequently update our Mule-Arch-201 exam preparation material to reflect the latest changes in the Mule-Arch-201 exam syllabus.

Salesforce Certified MuleSoft Platform Architect Sample Questions (Q77-Q82):

NEW QUESTION # 77

An organization wants MuleSoft-hosted runtime plane features (such as HTTP load balancing, zero downtime, and horizontal and vertical scaling) in its Azure environment. What runtime plane minimizes the organization's effort to achieve these features?

- A. Anypoint Platform for Pivotal Cloud Foundry
- B. CloudHub
- C. A hybrid combination of customer-hosted and MuleSoft-hosted Mule runtimes
- **D. Anypoint Runtime Fabric**

Answer: D

Explanation:

Correct Answer: Anypoint Runtime Fabric

>> When a customer is already having an Azure environment, It is not at all an ideal approach to go with hybrid model having some Mule Runtimes hosted on Azure and some on MuleSoft. This is unnecessary and useless.

>> CloudHub is a Mulesoft-hosted Runtime plane and is on AWS. We cannot customize to point CloudHub to customer's Azure environment.

>> Anypoint Platform for Pivotal Cloud Foundry is specifically for infrastructure provided by Pivotal Cloud Foundry

>> Anypoint Runtime Fabric is right answer as it is a container service that automates the deployment and orchestration of Mule applications and API gateways. Runtime Fabric runs within a customer-managed infrastructure on AWS, Azure, virtual machines (VMs), and bare-metal servers.

-Some of the capabilities of Anypoint Runtime Fabric include:

-Isolation between applications by running a separate Mule runtime per application.

-Ability to run multiple versions of Mule runtime on the same set of resources.

-Scaling applications across multiple replicas.

-Automated application fail-over.

-Application management with Anypoint Runtime Manager.

NEW QUESTION # 78

What is a typical result of using a fine-grained rather than a coarse-grained API deployment model to implement a given business process?

- A. An overall tower usage of resources because each fine-grained API consumes less resources
- B. A decrease in the number of connections within the application network supporting the business process
- **C. A higher number of discoverable API-related assets in the application network**
- D. A better response time for the end user as a result of the APIs being smaller in scope and complexity

Answer: C

Explanation:

Correct Answer: A higher number of discoverable API-related assets in the application network.

>> We do NOT get faster response times in fine-grained approach when compared to coarse-grained approach.

>> In fact, we get faster response times from a network having coarse-grained APIs compared to a network having fine-grained APIs model. The reasons are below.

Fine-grained approach:

1. will have more APIs compared to coarse-grained

2. So, more orchestration needs to be done to achieve a functionality in business process.

3. Which means, lots of API calls to be made. So, more connections will need to be established. So, obviously more hops, more network i/o, more number of integration points compared to coarse-grained approach where fewer APIs with bulk functionality embedded in them.

4. That is why, because of all these extra hops and added latencies, fine-grained approach will have bit more response times compared to coarse-grained.

5. Not only added latencies and connections, there will be more resources used up in fine-grained approach due to more number of APIs.

That's why, fine-grained APIs are good in a way to expose more number of reusable assets in your network and make them

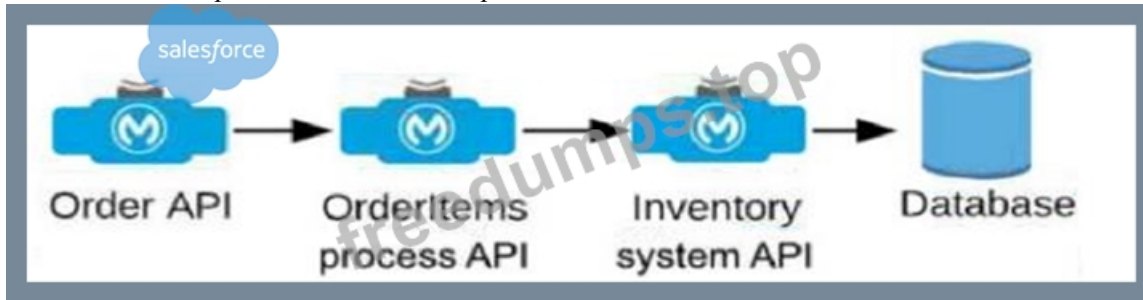
discoverable. However, needs more maintenance, taking care of integration points, connections, resources with a little compromise w.r.t network hops and response times.

NEW QUESTION # 79

An Order API triggers a sequence of other API calls to look up details of an order's items in a back-end inventory database. The Order API calls the OrderItems process API, which calls the Inventory system API. The Inventory system API performs database operations in the back-end inventory database.

The network connection between the Inventory system API and the database is known to be unreliable and hang at unpredictable times.

Where should a two-second timeout be configured in the API processing sequence so that the Order API never waits more than two seconds for a response from the OrderItems process API?



- A. In the Order API implementation
- B. In the inventory database
- C. In the Inventory system API implementation
- **D. In the OrderItems process API implementation**

Answer: D

Explanation:

Understanding the API Flow and Timeout Requirement:

The Order API initiates a call to the OrderItems process API, which in turn calls the Inventory system API to fetch details from the inventory database.

The requirement specifies that the Order API should not wait more than two seconds for a response from the OrderItems process API, even if there are delays further down the chain (between Inventory system API and the database).

Choosing the Appropriate Timeout Location:

Setting the timeout at the OrderItems process API level ensures that if the Inventory system API takes longer than two seconds to respond, the OrderItems process API will terminate the request and send a timeout response back to the Order API. This prevents the Order API from waiting indefinitely due to the unreliable connection to the database.

If the timeout were set in the Inventory system API or database, it would not help the Order API directly, as the OrderItems process API would still be waiting for a response.

Detailed Analysis of Each Option:

Option A (Correct Answer): Setting the timeout in the OrderItems process API allows it to control how long it waits for a response from the Inventory system API. If the Inventory system API does not respond within two seconds, the OrderItems process API can terminate the call and return a timeout response to the Order API, meeting the requirement.

Option B: Setting the timeout in the Order API would not limit the wait time at the OrderItems process API level, meaning the OrderItems process API could still wait indefinitely for the Inventory system API, leading to a longer delay.

Option C: Setting the timeout in the Inventory system API only affects the connection to the database and does not influence how long the OrderItems process API waits for the Inventory system API's response.

Option D: Setting a timeout in the database is not feasible in this context since database timeouts are typically configured for database operations and would not directly control the API response times in the overall API chain.

Conclusion:

Option A is the best choice, as it ensures that the OrderItems process API does not hold the Order API longer than the required two seconds, even if the downstream connection to the database hangs. This configuration aligns with MuleSoft best practices for setting timeouts in API orchestration to manage dependencies and prevent delays across a chain of API calls.

For additional information on timeout settings, refer to MuleSoft documentation on handling timeouts and API orchestration best practices.

NEW QUESTION # 80

An Anypoint Platform organization has been configured with an external identity provider (IdP) for identity management and client management. What credentials or token must be provided to Anypoint CLI to execute commands against the Anypoint Platform APIs?

- A. An OAuth 2.0 token generated using the credentials provided by the IdP for client management
- B. The credentials provided by the IdP for client management
- **C. The credentials provided by the IdP for identity management**
- D. An OAuth 2.0 token generated using the credentials provided by the IdP for identity management

Answer: C

Explanation:

Correct Answer: The credentials provided by the IdP for identity management

Reference:

>> There is no support for OAuth 2.0 tokens from client/identity providers to authenticate via Anypoint CLI. Only possible tokens are "bearer tokens" that too only generated using Anypoint Organization/Environment Client Id and Secret from <https://anypoint.mulesoft.com/accounts/login>. Not the client credentials of client provider. So, OAuth 2.0 is not possible. Moreover, the token is mainly for API Manager purposes and not associated with a user. You can NOT use it to call most APIs (for example Cloudhub and etc) as per this Mulesoft Knowledge article.

>> The other option allowed by Anypoint CLI is to use client credentials. It is possible to use client credentials of a client provider but requires setting up Connected Apps in client management but such details are not given in the scenario explained in the question.

>> So only option left is to use user credentials from identity provider

NEW QUESTION # 81

A team is planning to enhance an Experience API specification, and they are following API-led connectivity design principles. What is their motivation for enhancing the API?

- A. An IP Allowlist policy is being added to the API instances in the Development and Staging environments
- **B. A Canonical Data Model is being adopted that impacts several types of data included in the API**
- C. The underlying System API is updated to provide more detailed data for several heavily used resources
- D. The primary API consumer wants certain kinds of endpoints changed from the Center for Enablement standard to the consumer system standard

Answer: B

Explanation:

In API-led design, an Experience API is enhanced to improve how data is delivered to end-user applications. One primary reason to enhance an Experience API is when new data standards, such as a Canonical Data Model, are adopted. Here's why:

Canonical Data Model (CDM):

Adopting a CDM standardizes data representations across the organization, making APIs more consistent and easier to consume across various services and applications.

Updating the Experience API ensures that it delivers data in this standardized format, improving interoperability and reusability.

of Correct Answer (D):
A CDM impacts the structure and types of data the API provides, and this update would be directly relevant to an Experience API, as it is the primary point of interaction for applications.

of Incorrect Options:

Option A involves adapting to consumer-specific standards, which is against API-led design principles.

Option B involves changes in System APIs, which don't directly mandate changes to the Experience API unless data formatting adjustments are required.

Option C (IP Allowlist) relates to security rather than API design and would not motivate a functional enhancement of the API.

Reference

For more details on the use of Canonical Data Models in API-led architecture, refer to MuleSoft's guidelines on data standardization and Experience API best practices.

NEW QUESTION # 82

.....

We are proud that our Salesforce Mule-Arch-201 exam preparation material is one of the best in the market. You should buy our

