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.pass4itsure.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

As a part of project , existing java implementation is being migrated to Mulesoft. Business is very tight on the budget and wish to complete the project in most economical way possible.

Canonical object model using java is already a part of existing implementation. Same object model is required by mule application for a business use case. What is the best way to achieve this?

A. Make use of Java module

B. Create similar model for Mule applications

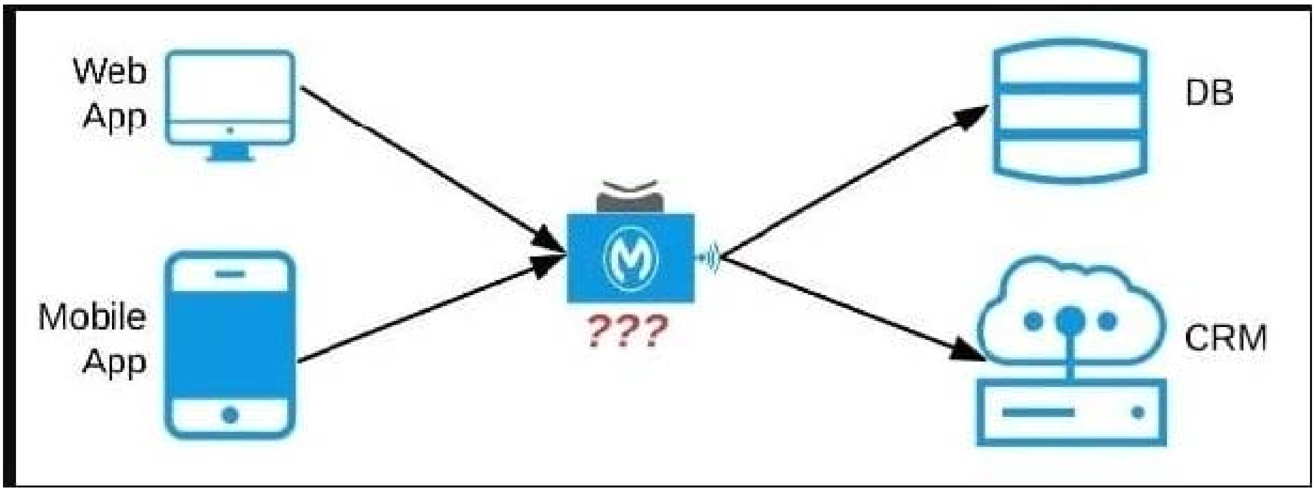
C. Create a custom application to read Java code and make it available for Mule application

D. Use Anypoint exchange

Correct Answer: A

QUESTION 2

An organization needs to enable access to their customer data from both a mobile app and a web application, which each need access to common fields as well as certain unique fields. The data is available partially in a database and partially in a 3rd-party CRM system. What APIs should be created to best fit these design requirements?



A. A Process API that contains the data required by both the web and mobile apps, allowing these applications to invoke it directly and access the data they need thereby providing the flexibility to add more fields in the future without needing API changes.

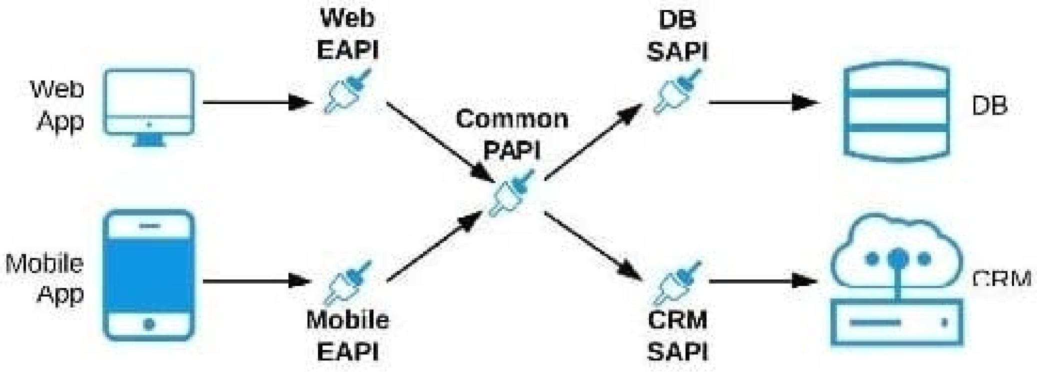
B. One set of APIs (Experience API, Process API, and System API) for the web app, and another set for the mobile app.

C. Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system

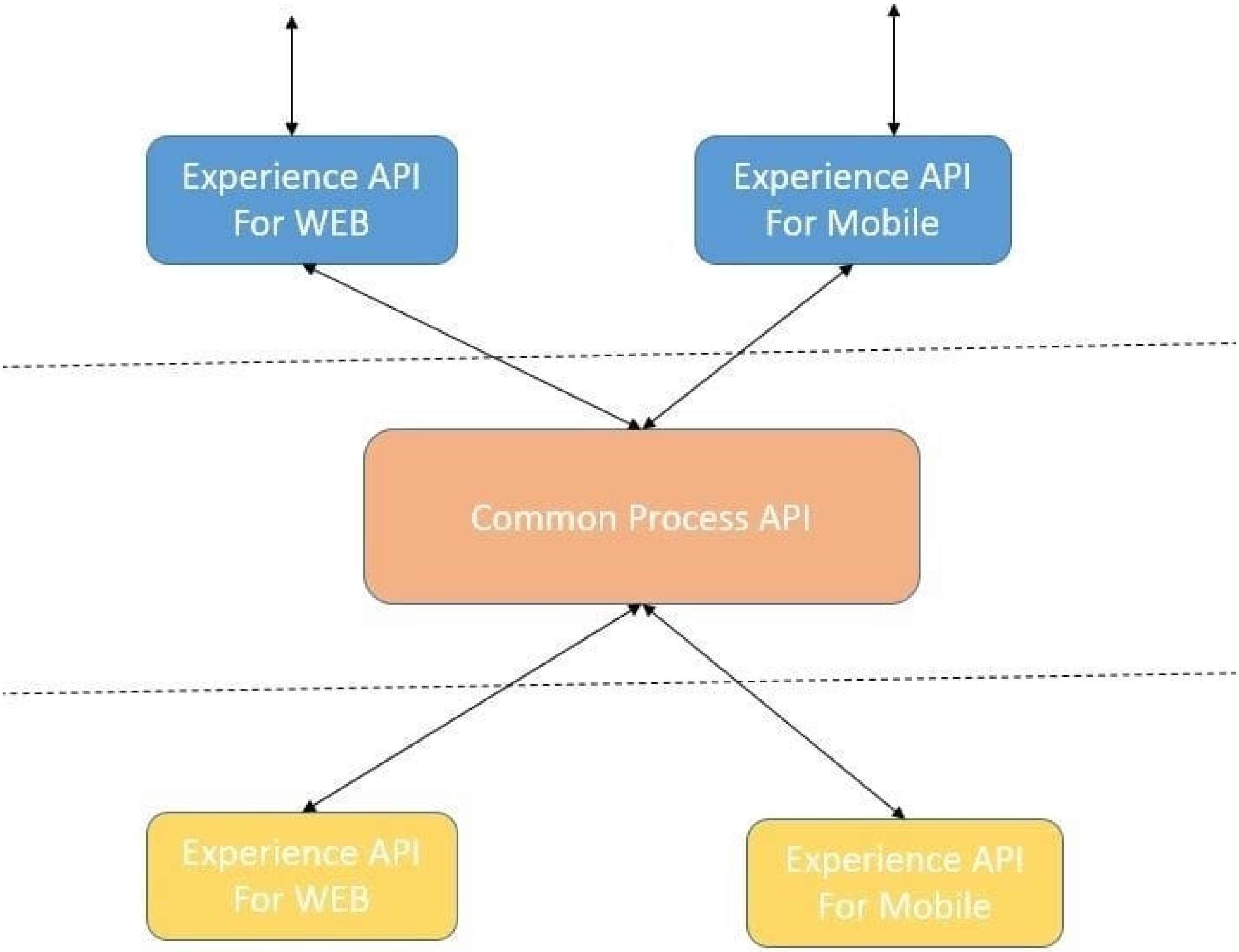
D. A common Experience API used by both the web and mobile apps, but separate Process APIs for the web and mobile apps that interact with the database and the CRM System.

Correct Answer: C

Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements which cannot be fulfilled by single experience layer API Option : Common Process API Analysis : This solution will not work because creating a common process API will impose limitations in terms of flexibility to customize API;s as per the requirements of different applications. It is not a recommended approach. Option : Separate set of API\\'s for both the applications Analysis : This goes against the principle of Anypoint API-led connectivity approach which promotes creating reusable assets. This solution may work but this is not efficient solution and creates duplicity of code. Hence the correct answer is: Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system

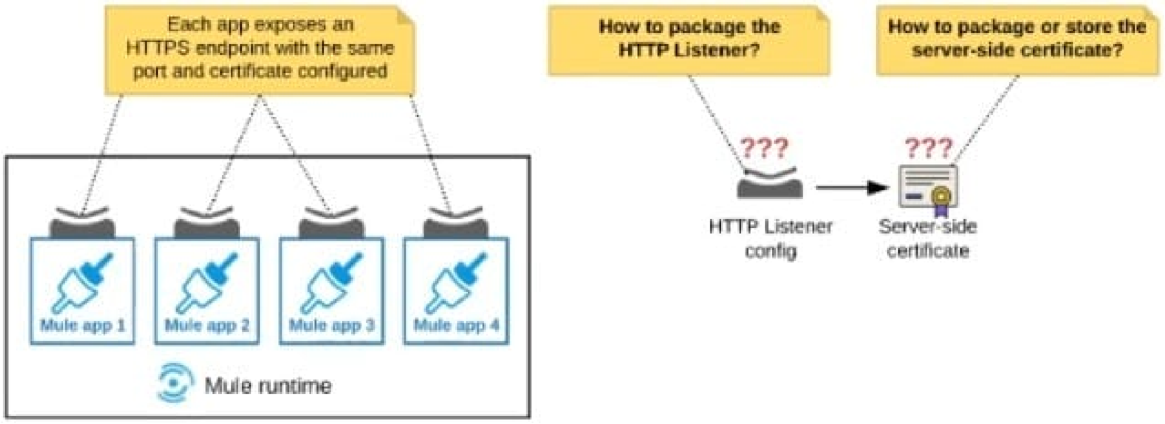


Description automatically generated with low confidence Lets analyze the situation in regards to the different options available Option : A common Experience API but separate Process APIs Analysis : This solution will not work because having common experience layer will not help the purpose as mobile and web applications will have different set of requirements which cannot be fulfilled by single experience layer API Option : Common Process API Analysis : This solution will not work because creating a common process API will impose limitations in terms of flexibility to customize API;s as per the requirements of different applications. It is not a recommended approach. Option : Separate set of API\\'s for both the applications Analysis : This goes against the principle of Anypoint API-led connectivity approach which promotes creating reusable assets. This solution may work but this is not efficient solution and creates duplicity of code. Hence the correct answer is: Separate Experience APIs for the mobile and web app, but a common Process API that invokes separate System APIs created for the database and CRM system



QUESTION 3

Refer to the exhibit.


Each app exposes an HTTPS endpoint with the same port and certificate configured
How to package the HTTP Listener?
How to package or store the server-side certificate?
???
???
HTTP Listener config
Server-side certificate
Mule app 1
Mule app 2
Mule app 3
Mule app 4
Mule runtime

An organization deploys multiple Mule applications to the same customer -hosted Mule runtime. Many of these Mule applications must expose an HTTPS endpoint on the same port using a server-side certificate that rotates often.

What is the most effective way to package the HTTP Listener and package or store the server-side certificate when deploying these Mule applications, so the disruption caused by certificate rotation is minimized?

A. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint Package the server- side certificate in ALL Mule APPLICATIONS that need to expose an HTTPS endpoint

B. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint. Store the server-side certificate in a shared filesystem location in the Mule runtime\\'s classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION

C. Package an HTTPS Listener configuration In all Mule APPLICATIONS that need to expose an HTTPS endpoint Package the server-side certificate in a NEW Mule DOMAIN project

D. Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing It from all Mule applications that need to expose an HTTPS endpoint. Package the server- side certificate in the SAME Mule DOMAIN project Go to Set

Correct Answer: B

Explanation:

In this scenario, both A and C will work, but A is better as it does not require repackage to the domain project at all.

Correct answer is Package the HTTPS Listener configuration in a Mule DOMAIN project, referencing it from all Mule applications that need to expose an HTTPS endpoint. Store the server-side certificate in a shared filesystem location in the

Mule runtime\\'s classpath, OUTSIDE the Mule DOMAIN or any Mule APPLICATION.

What is Mule Domain Project?

*

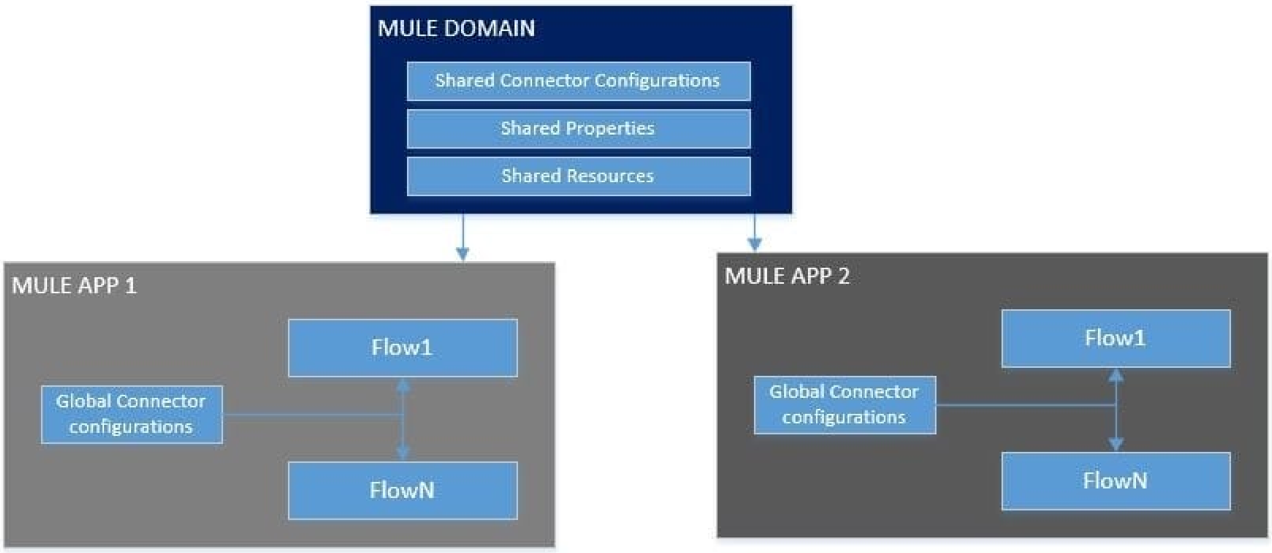
A Mule Domain Project is implemented to configure the resources that are shared among different projects. These resources can be used by all the projects associated with this domain. Mule applications can be associated with only one domain, but a domain can be associated with multiple projects. Shared resources allow multiple development teams to work in parallel using the same set of reusable connectors. Defining these connectors as shared resources at the domain level allows the team to: - Expose multiple services within the domain through the same port. - Share the connection to persistent storage. - Share services between apps through a well-defined interface. - Ensure consistency between apps upon any changes because the configuration is only set in one place.

*

Use domains Project to share the same host and port among multiple projects. You can declare the http connector within a domain project and associate the domain project with other projects. Doing this also allows to control thread settings, keystore configurations, time outs for all the requests made within multiple applications. You may think that one can also achieve this by duplicating the http connector configuration across all the applications. But, doing this may pose a nightmare if you have to make a change and redeploy all the applications.

*

If you use connector configuration in the domain and let all the applications use the new domain instead of a default domain, you will maintain only one copy of the http connector configuration. Any changes will require only the domain to the redeployed instead of all the applications. You can start using domains in only three steps: 1) Create a Mule Domain project 2) Create the global connector configurations which needs to be shared across the applications inside the Mule Domain project 3) Modify the value of domain in mule-deploy.properties file of the applications


MULE DOMAIN
Shared Connector Configurations
Shared Properties
Shared Resources
MULE APP 1
Flow1
Global Connector configurations
FlowN
MULE APP 2
Flow1
Global Connector configurations
FlowN

Use a certificate defined in already deployed Mule domain Configure the certificate in the domain so that the API proxy HTTPS Listener references it, and then deploy the secure API proxy to the target Runtime Fabric, or on-premises target. (CloudHub is not supported with this approach because it does not support Mule domains.)

QUESTION 4

An organization\\'s IT team follows an API-led connectivity approach and must use Anypoint Platform to implement a System AP\ that securely accesses customer data. The organization uses Salesforce as the system of record for all customer data, and its most important objective is to reduce the overall development time to release the System API.

The team\\'s integration architect has identified four different approaches to access the customer data from within the implementation of the System API by using different Anypoint Connectors that all meet the technical requirements of the project.

A. Use the Anypoint Connector for Database to connect to a MySQL database to access a copy of the customer data

B. Use the Anypoint Connector for HTTP to connect to the Salesforce APIs to directly access the customer data

C. Use the Anypoint Connector for Salesforce to connect to the Salesforce APIs to directly access the customer data

D. Use the Anypoint Connector tor FTP to download a file containing a recent near-real time extract of the customer data

Correct Answer: C

QUESTION 5

A leading bank implementing new mule API.

The purpose of API to fetch the customer account balances from the backend application and display them on the online platform the online banking platform. The online banking platform will send an array of accounts to Mule API get the account balances.

As a part of the processing the Mule API needs to insert the data into the database for auditing purposes and this process should not have any performance related implications on the account balance retrieval flow How should this requirement be implemented to achieve better throughput?

A. Implement the Async scope fetch the data from the backend application and to insert records in the Audit database

B. Implement a for each scope to fetch the data from the back-end application and to insert records into the Audit database

C. Implement a try-catch scope to fetch the data from the back-end application and use the Async scope to insert records into the Audit database

D. Implement parallel for each scope to fetch the data from the backend application and use Async scope to insert the records into the Audit database

Correct Answer: D