Free Questions for Architecture-Specialist-11 by certsinside

Shared by Mcknight on 24-05-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Which of the below matches the most to Library Module Pattern - Extension Pattern...

Options:

A- ... is a pattern with two modules, a connector module that can be used to encapsulate an external API with the input/output structures and a wrapper module to expose the normalized API to the consumers.

B- Same as ECS with local replica but API module is provided. So any changes to the external system can notify OS, which OS then gets update from the ERP system (subscription system)

C- ... tries to fetch data from local cache entity, if not there, get single entry from the external system. Cache only that record (read-through caching) Use when whole database too big or costly to synchronize. Integration only touches a small portion of the database. Avoid if access to lists of data is needed up front

D- ... is needed if data is coming from MULTIPLE external systems. IS will decide which driver to use depending on the data.

E- ... caches only summary data that is frequently lister, joined or searched. Full detail for a) single entry is fetched directly from external system. Use when whole database too big or costly to synchronize. Details are only required for single entities (not lists)

F- Entity is exposed as read-only and API is available to centralize business logic for entity creation/update

G- Same as ECS with local replica but synchronization logic is separated. Pro: Code independence. Consumers of CS is not affected by Sync. Sync can orchestrate several CS

H- Entity is not in Outsystems but in an external ERP system. IS just makes remote call to v external system/database. No data is being kept inside OS. Data retrieval may not be optimized as it needs to traverse two different systems to get the information back. Con: Integration API must support all use cases

I- ... a wrapper used to contain the logic, actions and data that will expose code that is inside of) external library or to inspect external database and import the data structures so they can be used as entities inside of OS

J- Same as Base ECS pattern, but have a local replica. Store data to serve as a local cache. Pro: Leverage Entity Use, Simpler Integration API. Con: Less impact on source system

Answer:

I

Question 2

Question Type: **MultipleChoice**

_API module is for

Options:

A- Logic to Synchronize data in CS's with an external system. Isolating this logic makes the CS completely system agnostic and it's easier to decouple or replace the external system.

B- A BL becomes a Calculation Engine if it performs complex calculations, (e.g. an invoice calculation engine or an insurance simulator). Engines are usually subject to versions.

C- Technical wrapper to expose an API to External consumers, keeping core services system agnostic and supporting multiple versions of the API.

D- Reusable Core Services with public entities, actions, and blocks.

E- Isolated Business Logic (Actions) or Core Widgets (blocks), to manage complexity, composition or to have its own lifecycle.

Answer:

C

Question 3

Question Type: **MultipleChoice**

Which of the below is NOT a suitable advice for designing a Parallel mobile local storage?

Options:

A- Allow table dependency : Normalize tables to promote parallel data fetch

B- Use Fetch Data : Avoid cascading aggregates in Onlnitialize and OnReady

C- Avoid generic tables : Contains too much data, not all are relevant

Answer:

A

Question 4

Question Type: MultipleChoice

Which of the below are not recommendations for Architecture Validations

Options:

A- All public entities in Core Modules should be set to Read-only

B- No business logic in the Foundational Modules

C- No core entities in Foundational Modules

D- No screens in the End User Layer

E- No front-end screens in Core Modules

Answer:

D

Question 5

Question Type: MultipleChoice

Which of the below is NOT a disadvantage of having a fragmented system or microservices?

Options:

A- All of the above

B- Monitoring & Logging : for effective monitoring & logging, requires centralized service

C- Inter-process communication : network latency and hiccups

D- Complicated Debugging and troubleshooting : root cause may be deep inside the chain of services

E- Security : need to manage credentials and access management

F- Limited data mashup in memory and limited to APIs

G- Multiple transactions committed independently

H- Fault tolerance : communication errors, service consistency

Answer:

A

Question 6

Question Type: MultipleChoice

Which are NOT part of the major concepts that needs to be investigated in the Disclose phase in Architecture Design Process?

Options:

A- User Experience Expectations

B- Integration Technology

C- User Stories, Personas and Roles

D- Architecture Canvas Modules

E- Information Architecture

Answer:

D

Question 7

Question Type: MultipleChoice

Which of the below is not a best practice for mobile security:authentication?

Options:

A- Store password in local storage

B- Encrypt (only) sensitive data

C- Authentication : Use google or facebook for online authentication or fingerprint or pin for offline authentication

Answer:

A

Question 8

Question Type: MultipleChoice

Which of the below matches the most to Core Module Pattern - Core Entity Pattern...

Options:

A- is a pattern with two modules, a connector module that can be used to encapsulate an O external API with the input/output structures and a wrapper module to expose the normalized API to the consumers.

B- ... is needed if data is coming from MULTIPLE external systems. IS will decide which driver to use depending on the data.

C- ... tries to fetch data from local cache entity, if not there, get single entry from the external system. Cache only that record (read-through caching) Use when whole database too big or costly to synchronize. Integration only touches a small portion of the database. Avoid if access to lists of data is needed up front

D- ... Entity is exposed as read-only and API is available to centralize business logic for entity creation/update

E- Same as Base ECS pattern, but have a local replica. Store data to serve as a local cache. Pro: Leverage Entity Use, Simpler Integration API. Con: Less impact on source system

F- Same as ECS with local replica but API module is provided. So any changes to the external system can notify OS, which OS then gets update from the ERP system (subscription system)

G- ... a wrapper used to contain the logic, actions and data that will expose code that is inside of external library or to inspect external database and import the data structures so they can be used as entities inside of OS

H- ... caches only summary data that is frequently lister, joined or searched. Full detail for a single entry is fetched directly from external system. Use when whole database too big or costly to synchronize. Details are only required for single entities (not lists)

I- Same as ECS with local replica but synchronization logic is separated. Pro: Code independence. Consumers of CS is not affected by Sync. Sync can orchestrate several CS

J- ... Entity is not in Outsystems but in an external ERP system. IS just makes remote call to external system/database. No data is being kept inside OS. Data retrieval may not be optimized as it needs to traverse two different systems to get the information back. Con:

Integration API must support all use cases

Answer:

D

Question 9

Question Type: MultipleChoice

In OutSystems, a Foundation Application can NOT contain ...

Options:

A- Core Modules and Foundation Modules.

B- End-User and Foundation Modules.

C- End-user and Core Modules.

Answer:

C

Question 10

Question Type: MultipleChoice

In which Architecture Canvas layer do you expect to have a higher reusability rate?

Options:

A- End-User layer

B- Core layer

C- Foundation layer

Answer:

C

Question 11

Question Type: MultipleChoice

Which of the below best matches this statement : "Needed if data is coming from MULTIPLE external systems. IS will decide which driver to use depending on the data."

Options:

A- Transparency Service Pattern 1 way (simplification)

B- Library : Extension Pattern

C- ECS with isolated synchronization logic

D- ECS Lazy Load variation

E- ECS with publish/subscribe through an ESB

F- ECS Summary cache only variation

G- Library : Connector Pattern

H- ECS with direct integration

I- Base ECS pattern

J- Transparency Service Pattern

Answer:

J