

# **Free Questions for Architecture-Specialist-11**

**Shared by McClain on 04-10-2024**

**For More Free Questions and Preparation Resources**

**Check the Links on Last Page**

## Question 1

---

**Question Type:** MultipleChoice

---

The Architecture Design Process has three steps. Which of the options below is NOT one of those steps?

**Options:**

---

**A-** Organize

**B-** Plan

**C-** Assemble

**D-** Disclose

**Answer:**

---

B

## Question 2

---

**Question Type:** MultipleChoice

---

There are generally 4 common style guide scenario. Which of the below is NOT a common style guide scenario?

### Options:

---

**A-** Specialize a Built-in: Use when introducing changes to an existing theme for multiple apps. First, create a custom theme that REFERENCE the Base Theme. Second, clone a Custom Template from the Base Template but this custom theme will reference the custom theme created previously. Third, create new applications based on the Custom Theme and Custom Template

**B-** Built-in Style Guide: Uses Built-in style guide from Outsystems UI. Minor customizations to the Base Theme should be done inside the App Theme.

**C,** Build Your Own: Use when custom style guide cannot benefit from any existing theme. Just like Clone a Built In scenario, clone both Custom Theme and Custom Template from the Base Theme and Base Template, however clone the Base Theme from Outsystems UI.

**D-** Since the theme is very basic, front-end developer will have to expand and design the theme from there.

Modify Built-in Style Guide: Use when only small changes are required for single application. First, create a custom theme that REFERENCE the Base Theme. Second, clone a Custom Template from the Base Template but this custom theme will reference the custom theme created previously. Third, reference that application to the custom Theme and Template.

**E-** Clone a Built-in: Use when introducing extensive changes to existing theme. First, CLONE a custom theme from a Base Theme. Second, clone a Custom Template from the Base Template but this custom theme will reference the custom theme created previously. Third, create new applications based on the Custom Theme and Custom Template.  
Base Theme can be Outsystems UI or your own custom theme.

### Answer:

---

D

## Question 3

---

**Question Type:** MultipleChoice

---

What is NOT a best practice for Mobile Application Architecture: Limit server side calls?

### Options:

---

**A-** Limit Server side calls to online transactions or Synchronization. Do the bulk of the operations on the Client Side before passing information to the Server Side.

**B-** Synchronize on every screen or online event for better granularity

No direct calls to backend systems. Do call to Outsystems Core Services and Outsystems will handle the interaction with external backend systems.

**C-** Direct calls is bad from Security, Performance and Maintainability standpoint. Direct calls will require javascript Code and javascript Code is harder to maintain.

### Answer:

---

B

## Question 4

---

**Question Type:** MultipleChoice

---

The Architecture Canvas is a ...

### Options:

---

- A- framework to support Architecture Conventions for Modules in Outsystems
- B- framework to support application architecture design in Outsystems
- C- framework to support Architecture Design Process in Outsystems

### Answer:

---

B

## Question 5

---

**Question Type:** MultipleChoice

---

What is NOT a best practice for Mobile Application Architecture: transactions & granularity?

### Options:

---

- A-** Have long synchronizations in a single transaction. Better UX as app does not need to sync all the time. Is prepared for constant offline or device standby
- B-** Ensure order and sync granularity. Sync incrementally by entity with partial commit. This way O synchronizations is prepared for constant interruptions and allow retries without repeating the entire synchronization from the start.

### Answer:

---

A

## Question 6

---

**Question Type:** MultipleChoice

---

Themes and Layouts are an important element of an application. In which of the following would you define these elements?

### Options:

---

- A-** MyApp\_CS Module, in the Foundation layer

- B-** MyApp\_BL, in the Foundation Layer.
- C-** MyApp\_Th Module, in the End-User layer
- D-** MyApp\_MTh, in the Foundation layer.

**Answer:**

---

C

## Question 7

---

**Question Type:** MultipleChoice

---

Considering Architecture Dashboard(AI Mentor), which of the following sentences is FALSE?

**Options:**

---

- A-** Architecture Dashboard allows you to automatically apply a solution to solve a found code pattern.
- B-** Architecture Dashboard performs a code analysis that uncovers patterns related to performance, security, architecture and maintainability.

**Answer:**

---

A

## Question 8

---

**Question Type:** MultipleChoice

---

What is the new name for Architecture Dashboard?

**Options:**

---

**A-** Architecture Validation

**B-** Architecture Canvas

**C-** Architecture Framework

**D-** AI Mentor

**Answer:**

---

D



## Question 9

---

**Question Type:** MultipleChoice

---

\_Sync module is for

### Options:

---

- A-** Reusable Core Services with public entities, actions, and blocks.
- B-** Isolated Business Logic (Actions) or Core Widgets (blocks), to manage complexity, composition or to have its own lifecycle.
- C-** 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.
- D-** 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.
- E-** Technical wrapper to expose an API to External consumers, keeping core services system agnostic and supporting multiple versions of the API.

### Answer:

---

D

## Question 10

---

Question Type: MultipleChoice

---

Which of the below is NOT a reason for Validating a Modules' Architecture?

### Options:

---

- A- Optimizes lifecycle independence
- B- Promotes microservices
- C- Minimizes impact of changes
- D- Promotes abstraction of reusable services
- E- Stop unmanageable dependencies
- F- Prevent slow deployments, inflexible legacy systems
- G- Avoid poor service abstraction

### Answer:

---

B

**To Get Premium Files for Architecture-Specialist-11 Visit**

**<https://www.p2pexams.com/products/architecture-specialist-11>**

**For More Free Questions Visit**

**<https://www.p2pexams.com/outsystems/pdf/architecture-specialist-11>**

