ACD300 Q&As

Appian Certified Lead Developer

Pass Appian ACD300 Exam with 100% Guarantee

QUESTION 1

You are running an inspection as a part of the first deployment process from TEST to PROD. You receive a notice that one of your objects will not deploy because it is dependent on an object from an application owned by a separate team.

What should be your next step?

A. Create your own object with the same code base, replace (he dependent object in the application. and deploy to PROO.

B. Halt the production deployment and contact the other team tor guidance on promoting the object to PROD

C. Check the dependencies of the necessary object Deploy w PROO if there are few dependencies and it is low risk

D. Push a functionally viable package to PROD without the dependencies, and plan the rest o! the deployment accordingly with the other team\\'s constraints

Correct Answer: B

Deploying an object that is dependent on another object from a different application can cause errors and inconsistencies in the production environment. The best practice is to halt the production deployment and contact the other team for guidance on how to promote the object to PROD. The other team may have a different deployment schedule, or they may have some dependencies or customizations that need to be considered. By communicating with the other team, you can ensure that the object is deployed in a safe and coordinated manner, and avoid any potential conflicts or issues. Verified References: [Appian Deployment Guide], [Appian Best Practices]

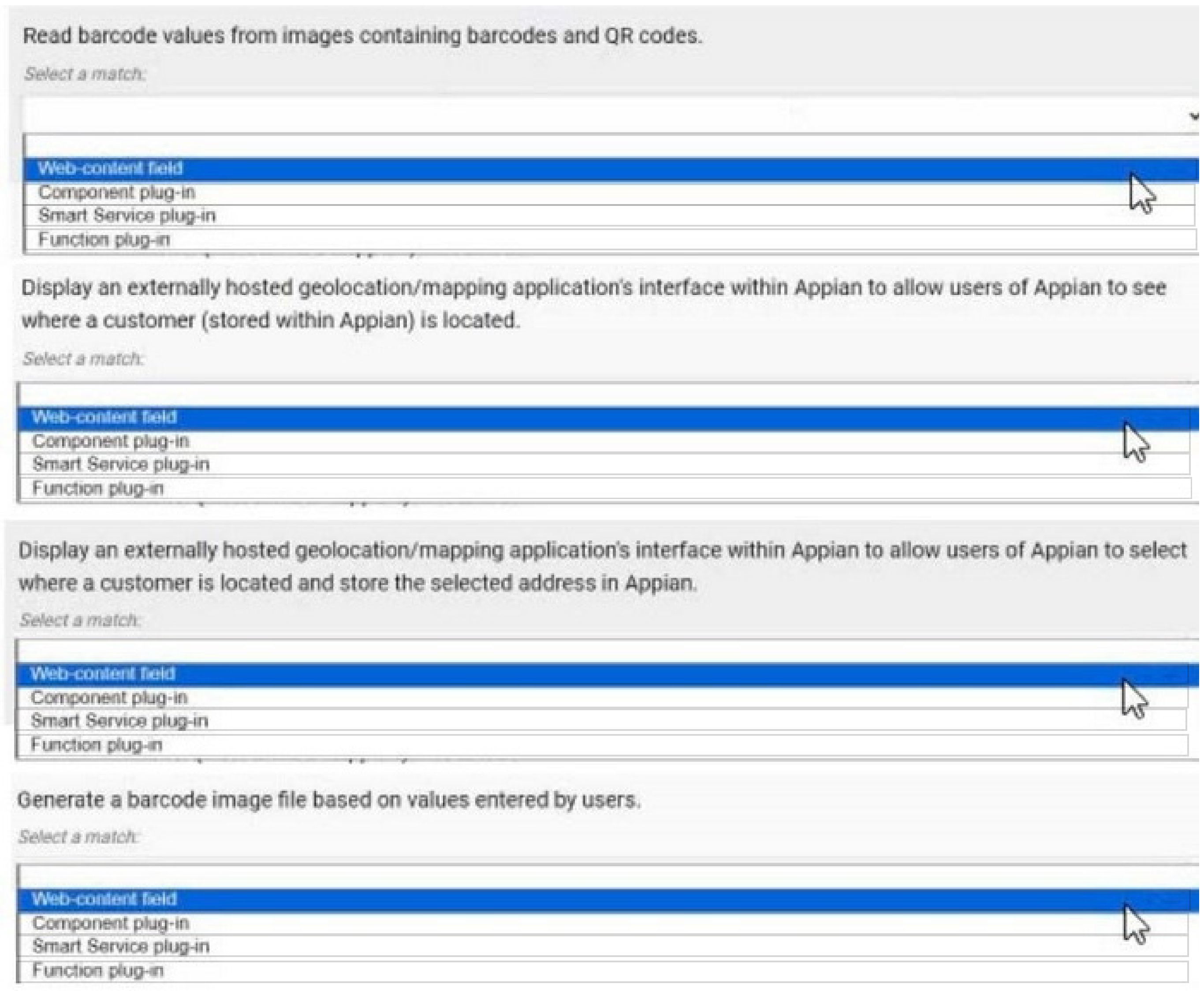
QUESTION 2

HOTSPOT

For each requirement, match the most appropriate approach to creating or utilizing plug-ins Each approach will be used once.

Note: To change your responses, you may deselect your response by clicking the blank space at the top of the selection list.

Hot Area:



Read barcode values from images containing barcodes and QR codes.

Select a match:

- Web-content field
- Component plug-in
- Smart Service plug-in
- Function plug-in

Display an externally hosted geolocation/mapping application's interface within Appian to allow users of Appian to see where a customer (stored within Appian) is located.

Select a match:

- Web-content field
- Component plug-in
- Smart Service plug-in
- Function plug-in

Display an externally hosted geolocation/mapping application's interface within Appian to allow users of Appian to select where a customer is located and store the selected address in Appian.

Select a match:

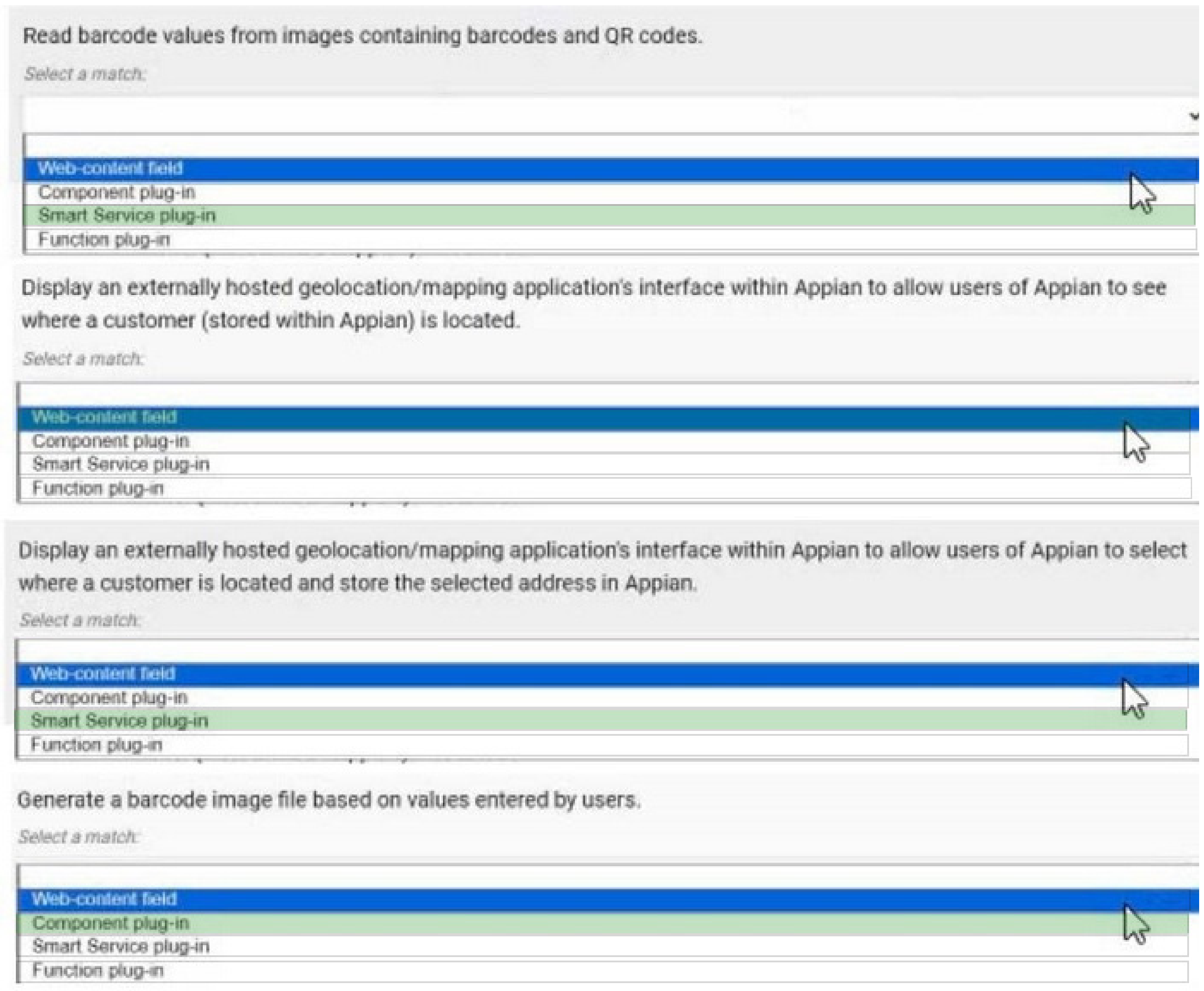
- Web-content field
- Component plug-in
- Smart Service plug-in
- Function plug-in

Generate a barcode image file based on values entered by users.

Select a match:

- Web-content field
- Component plug-in
- Smart Service plug-in
- Function plug-in

Correct Answer:



Read barcode values from images containing barcodes and QR codes.

Select a match:

- Web-content field
- Component plug-in
- [✓ Smart Service plug-in]
- Function plug-in

Display an externally hosted geolocation/mapping application's interface within Appian to allow users of Appian to see where a customer (stored within Appian) is located.

Select a match:

- Web-content field
- Component plug-in
- Smart Service plug-in
- Function plug-in

Display an externally hosted geolocation/mapping application's interface within Appian to allow users of Appian to select where a customer is located and store the selected address in Appian.

Select a match:

- Web-content field
- Component plug-in
- [✓ Smart Service plug-in]
- Function plug-in

Generate a barcode image file based on values entered by users.

Select a match:

- Web-content field
- [✓ Component plug-in]
- Smart Service plug-in
- Function plug-in

Requirement: Read barcode values from images containing barcodes and QR codes. Correct approach: C. Smart Service plug-in Exact explanation of correct approach taken from Appian Documentation: A smart service plug-in is a type of plug-in that allows you to create custom smart services that can be used in process models. A smart service can perform complex logic, interact with external systems, or manipulate data in Appian. A smart service plug-in can also leverage Java code to implement the functionality of the smart service. A smart service plug-in would be suitable for reading barcode values from images, as it can use Java libraries or APIs that can scan and decode barcodes and QR codes from image files. A smart service plug-in can also return the barcode values as outputs that can be used by other nodes or processes in Appian. A smart service plug-in can also be configured with input parameters, such as the image file, the barcode type, or the output format, that can customize the behavior of the smart service. A smart service plug-in can also have error handling and logging features that can handle any exceptions or failures that might occur during the barcode reading process. Requirement: Display an externally hosted geolocation mapping applications interface within Appian to allow users of Appian to see where a customer (stored within Appian)is located. Correct approach: A. Web-content field Exact explanation of correct approach taken from Appian Documentation: A web-content field is a type of user interface component that allows you to display web content from an external source in a SAIL interface. A web-content field would be suitable for displaying an externally hosted geolocation mapping applications interface, as it can embed the web content in an iframe and render it within the Appian interface. You can also pass parameters to the web content, such as the customer\\'s location, using the url parameter of the web-content field. A web- content field can also interact with other components in the Appian interface, such as buttons, grids, or forms, using the postMessage API. This way, you can create a seamless user experience that integrates the external geolocation mapping applications interface with the Appian functionality. Requirement: Display an externally hosted geolocation mapping applications interface within Appian to allow users of Appian to select where a customer is located and store the selected address in Appian. Correct approach: A. Web-content field and C. Smart Service plug-in Exact explanation of correct approach taken from Appian Documentation: A web- content field and a smart service plug-in would be suitable for displaying an externally hosted geolocation mapping applications interface within Appian to allow users of Appian to select where a customer is located and store the selected address in Appian. A web- content field would be suitable for displaying the external geolocation mapping applications interface, as explained above. A smart service plug-in would be suitable for storing the selected address in Appian, as it can use Java code to receive the address data from the web content, validate it, and write it to a data store entity or a process variable. Requirement: Generate a barcode image file based on values entered by users. Correct approach: B. Component plug-in Exact explanation of correct approach taken from Appian Documentation: A component plug-in is a type of plug-in that allows you to create custom user interface components that can be used in SAIL interfaces. A component plug-in can also leverage Java code to implement the functionality of the component. A component plug-in would be suitable for generating a barcode image file, as it can use Java libraries or APIs that can encode values into barcode formats and generate image files. A component plug-in can also display the barcode image file in the Appian interface and allow users to download or print it. A component plug-in can also interact with other components in the Appian interface, such as text fields, buttons, or forms, using the a!refreshVariable() function. This way, you can create a dynamic user experience that updates the barcode image file based on the values entered by users.

QUESTION 3

You are taskedto build a large scale acquisition application for a prominent customer. The acquisition process tracks the time it takes is fulfill a purchase request with an award.

The customer has structured the contract so that there are multiple application dev teams.

How should you design for multiple processes and forms, while minimizing repeated code?

A. Create a Center of Excellence (CoE)

B. Create a common objects application.

C. Create a Scrum of Scrums sprint meeting for the team leads

D. Create duplicate processes and forms as needed

Correct Answer: B

To build a large scale acquisition application for a prominent customer, you should design for multiple processes and forms, while minimizing repeated code. One way to do this is to create a common objects application, which is a shared application that contains reusable components, such as rules, constants, interfaces, integrations, or data types, that can be used by multiple applications. This way, you can avoid duplication and inconsistency of code, and make it easier to maintain and update your applications. You can also use the common objects application to define common standards and best practices for your application development teams, such as naming conventions, coding styles, or documentation guidelines. Verified References: [Appian Best Practices], [Appian Design Guidance]

QUESTION 4

Your application contains a process model that Is scheduled to run daily at a certain time, which kicks off a user input task to a specified user on the 1ST time zone for morning data collection The time zone is set to the (default) pm!timezone.

In this situation, what does the pm!tinezone reflect?

A. The time zone of the server where Applan is intuited

B. The line zone of the user who most recently published the process model

C. The default time zone for the environment as specified in the Administration Console

D. The time zone of the user who is completing the input task.

Correct Answer: C

In this situation, pm!timezone reflects the default time zone for the environment as specified in the Administration Console. pm!timezone is a process variable that returns the time zone of the process. If the time zone is not explicitly set in the process model, then pm!timezone returns the default time zone for the environment, which can be configured in the Administration Console. In this case, the time zone is set to the (default) pm!timezone, which means that the process model does not have a specific time zone, and therefore uses the default time zone for the environment. The other options are not correct. Option A, the time zone of the server where Appian is installed, is not what pm!timezone reflects, as the server time zone may not be the same as the default time zone for the environment. Option B, the time zone of the user who most recently published the process model, is not what pm!timezone reflects, as the user\\'s time zone may not be the same as the default time zone for the environment. Option D, the time zone of the user who is completing the input task, is not what pm!timezone reflects, as the user\\'s time zone may not be the same as the default time zone for the environment.

QUESTION 5

As part of an upcoming release of an application, a new nullable field is added to a table that contains customer dataThe new field is used by a reportin the upcoming release, and is calculated using data from another table.

Which two actions should you consider when creating the script to add the new field?

A. Create a script thatadds the held and leaves it null.

B. Create a rollback script that removes the field.

C. Create a script that adds the field and then populate it

D. Create a rollback script that clears the data from the field

E. Add a view that joins the customer data to the data used in calculation

Correct Answer: BC

When creating a script to add a new field to a table, you should consider two actions:

Create a rollback script that removes the field. A rollback script is a script that can undo the changes made by the original script, in case something goes wrongor the changes need to be reverted. A rollback script is a good practice to have,

as it can help to restore the previous state of the database and avoid any errors or inconsistencies. In this case, the rollback script should remove the new field from the table, and any other changes that were made by the original script.

Create a script that adds the field and then populate it. A script that adds the field and then populate it is a script that can create the new field in the table, and then fill it with data from another table or source. This way, you can ensure that the

new field has valid and consistent data, and that it can be used by the report in the upcoming release. In this case, the script should add the new field to the customer table, and then populate it with data from another table that contains the data used in the calculation.