AZ-400$^{\text{Q\&As}}$

Designing and Implementing Microsoft DevOps Solutions

Pass Microsoft AZ-400 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.passapply.com/az-400.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft Official Exam Center

QUESTION 1

You have a private distribution group that contains provisioned and unprovisioned devices.

You need to distribute a new iOS application to the distribution group by using Microsoft Visual Studio App Center.

What should you do?

A. Select Register devices and sign my app.

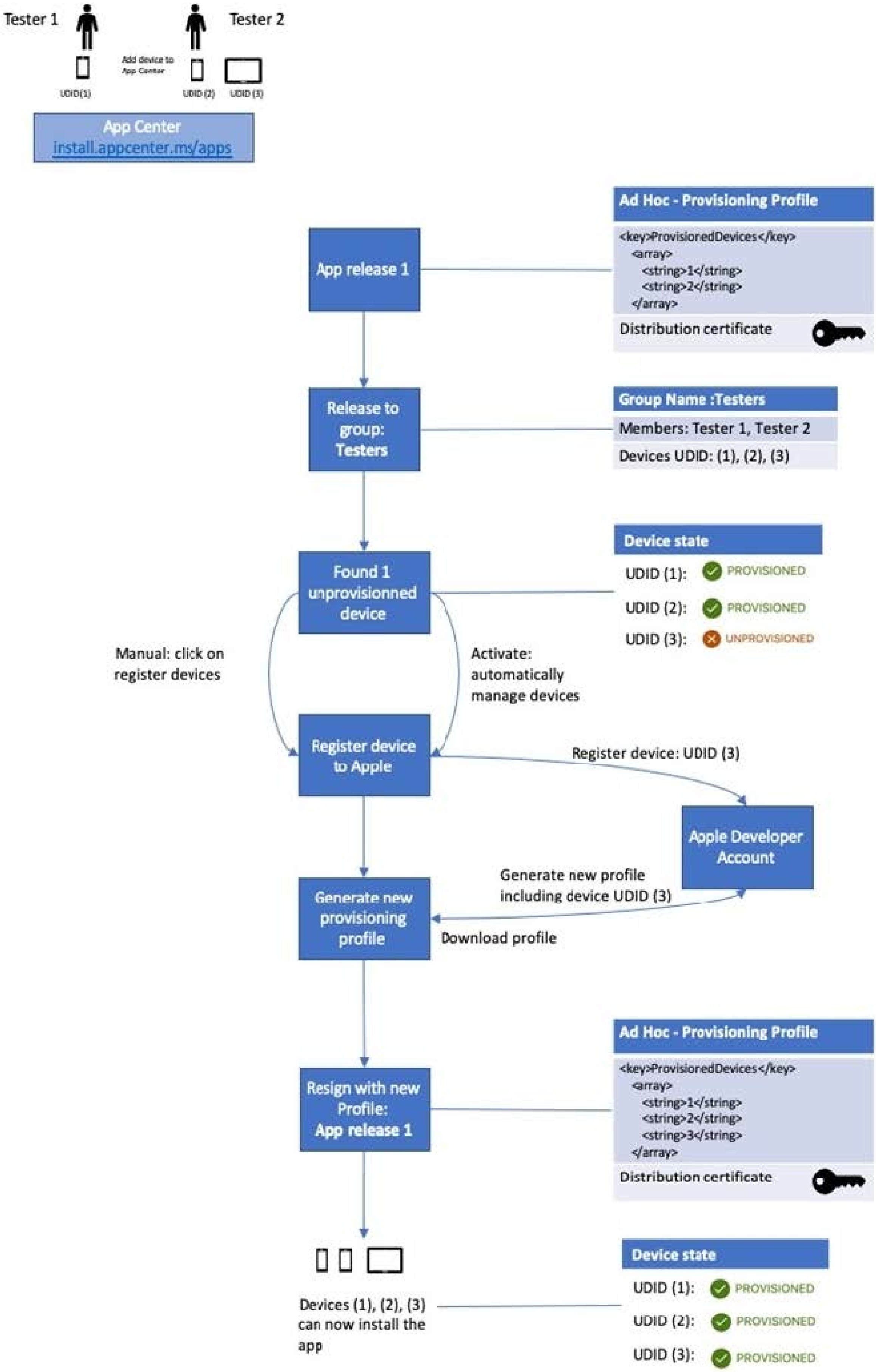
B. Generate a new .p12 file for each device.

C. Create an active subscription in App Center Test.

D. Add the device owner to the collaborators group.

Correct Answer: A

The following diagram displays the entire app re-signing flow in App Center.


Tester 1
Tester 2
Add device to App Center
UDID (1)
UDID (2)
UDID (3)
App Center
install.appcenter.ms/apps
App release 1
Ad Hoc - Provisioning Profile
Distribution certificate
Release to group: Testers
Group Name :Testers
Members: Tester 1, Tester 2
Devices UDID: (1), (2), (3)
Found 1 unprovisionned device
Device state
UDID (1): ✓ PROVISIONED
UDID (2): ✓ PROVISIONED
UDID (3): ✗ UNPROVISIONED
Manual: click on register devices
Activate: automatically manage devices
Register device to Apple
Register device: UDID (3)
Apple Developer Account
Generate new profile including device UDID (3)
Download profile
Generate new provisioning profile
Resign with new Profile: App release 1
Ad Hoc - Provisioning Profile
Distribution certificate
Devices (1), (2), (3) can now install the app
Device state
UDID (1): ✓ PROVISIONED
UDID (2): ✓ PROVISIONED
UDID (3): ✓ PROVISIONED

```
<key>ProvisionedDevices</key>
  <array>
    <string>1</string>
    <string>2</string>
  </array>
```

```
<key>ProvisionedDevices</key>
  <array>
    <string>1</string>
    <string>2</string>
    <string>3</string>
  </array>
```

Incorrect Answers:

B: Only one .p12 file for the app, not one for each device.

Reference: https://docs.microsoft.com/hu-hu/appcenter/distribution/auto-provisioning

QUESTION 2

Your company is building a new solution in Java.

The company currently uses a SonarQube server to analyze the code of .NET solutions.

You need to analyze and monitor the code quality of the Java solution.

Which task types should you add to the build pipeline?

A. Chef

B. Gradle

C. Octopus

D. Gulp

Correct Answer: C

SonarQube is a set of static analyzers that can be used to identify areas of improvement in your code. It allows you to analyze the technical debt in your project and keep track of it in the future. With Maven and Gradle build tasks, you can run SonarQube analysis with minimal setup in a new or existing Azure DevOps Services build task.

References: https://docs.microsoft.com/en-us/azure/devops/java/sonarqube?view=azure-devops

QUESTION 3

SIMULATION

You need to create an instance of Azure Application Insights named az400-9940427-main and configure the instance to receive telemetry data from an Azure web app named az400-9940427-main.

To complete this task, sign in to the Microsoft Azure portal.

Correct Answer: See solution below.

Step 1: Create an instance of Azure Application Insights

1.

Open Microsoft Azure Portal

2.

Log into your Azure account, Select Create a resource > Developer tools > Application Insights.

3.

Enter the following settings, and then select Review + create. Name: az400-9940427-main

Step 2: Configure App Insights SDK

4.

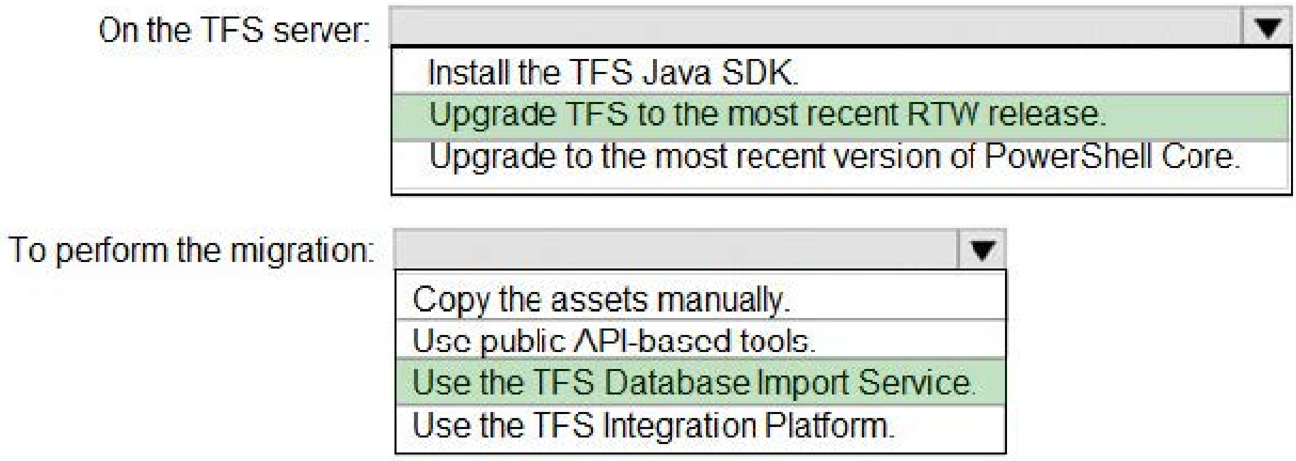
Open your ASP.NET Core Web App project in Visual Studio > Right-click on the AppName in the Solution Explorer > Select Add > Application Insights Telemetry.

5.

Click the Get Started button

6.

Select your account and subscription > Select the Existing resource you created in the Azure portal > Click Register.

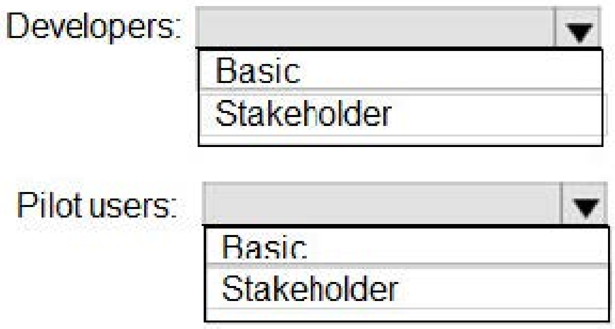


On the TFS server:

- Install the TFS Java SDK.
- [✓ Upgrade TFS to the most recent RTW release.]
- Upgrade to the most recent version of PowerShell Core.

To perform the migration:

- Copy the assets manually.
- Use public API-based tools.
- [✓ Use the TFS Database Import Service.]
- Use the TFS Integration Platform.



Developers:

- Basic
- Stakeholder

Pilot users:

- Basic
- Stakeholder

References: https://docs.microsoft.com/bs-latn-ba/azure/azure-monitor/learn/dotnetcore-quick-start?view=vs-2017

QUESTION 4

You need to configure GitHub to use Azure Active Directory (Azure AD) for authentication. What should you do first?

A. Create a conditional access policy in Azure AD.

B. Modify the Security settings of the GitHub organization.

C. Create an Azure Active Directory B2C (Azure AD B2C) tenant.

D. Register GitHub in Azure AD.

Correct Answer: D

When you connect to a Git repository from your Git client for the first time, the credential manager prompts for credentials. Provide your Microsoft account or Azure AD credentials.

Note: Git Credential Managers simplify authentication with your Azure Repos Git repositories. Credential managers let you use the same credentials that you use for the Azure DevOps Services web portal. Credential managers support multi-factor authentication through Microsoft account or Azure Active Directory (Azure AD). Besides supporting multi-factor authentication with Azure Repos, credential managers also support two-factor authentication with GitHub repositories.

Reference: https://docs.microsoft.com/en-us/azure/devops/repos/git/set-up-credential-managers

QUESTION 5

You have a public GitHub repository named Public1.

A commit is made to Public1. The commit contains a pattern that matches a regular expression.

Who is notified first when the commit is made?

A. the administrator of the GitHub organization

B. the committer

C. the owner of Public1

D. the secret scanning partner

Correct Answer: A

When a match of your secret format is found in a private repository configured for secret scanning, then repository admins and the committer are alerted and can view and manage the secret scanning result on GitHub.

Note: Secret scanning partner program

The secret scanning process

Joining the secret scanning program on GitHub

As a service provider, you can partner with GitHub to have your secret token formats secured through secret scanning, which searches for accidental commits of your secret format and can be sent to a service provider\\'s verify endpoint.

GitHub scans repositories for known secret formats to prevent fraudulent use of credentials that were committed accidentally. Secret scanning happens by default on public repositories and public npm packages. Repository administrators and organization owners can also enable secret scanning on private repositories. As a service provider, you can partner with GitHub so that your secret formats are included in our secret scanning.

When a match of your secret format is found in a public source, a payload is sent to an HTTP endpoint of your choice.

When a match of your secret format is found in a private repository configured for secret scanning, then repository admins and the committer are alerted and can view and manage the secret scanning result on GitHub.

The secret scanning process

The following diagram summarizes the secret scanning process for public repositories, with any matches sent to a service provider\\'s verify endpoint. A similar process sends service providers tokens exposed in public packages on the npm

registry.

Reference:

https://docs.github.com/en/code-security/secret-scanning/secret-scanning-partner-program