Microsoft 365 Certified: Developer Associate – Skills Measured

This document contains the skills measured on the exams associated with this certification. It does not include any upcoming or recent changes that have been made to those skills. For more information about upcoming or recent changes, see the associated exam details page(s).

NOTE: The bullets that follow each of the skills measured are intended to illustrate how we are assessing that skill. This list is NOT definitive or exhaustive.

NOTE: Most questions cover features that are General Availability (GA). The exam may contain questions on Preview features if those features are commonly used.

Exam MS-600: Building Applications and Solutions with Microsoft 365 Core Services

Implement Microsoft identity (20-25%)

Register an application

- determine the supported account type
- select authentication and client credentials for app type and authentication flow
- define app roles

Implement authentication

- configure the JavaScript implementation of Microsoft Authentication Library (MSAL) for endpoint and token cache
- plan and configure scopes for dynamic or static permissions
- use the MSAL (JavaScript) login method

Configure permissions to consume an API

- configure delegated permissions for the app
- configure application permissions for the app
- identify admin consent requirements

Implement authorization to consume an API

- configure incremental consent scopes
- call MSAL (JavaScript) using aquireTokenSilent/aquireToken pattern

Implement authorization in an API

- validate access token
- configure effective permissions for delegated scopes
- implement app permissions using roles
- use a delegated access token to call a Microsoft API

Create a service to access Microsoft Graph

- configure client credentials using a certificate
- configure application permissions for the app
- acquire an access token for Microsoft Graph using an application permission and client credential certificate
- acquire an access token using the client secret

Build apps with Microsoft Graph (15-20%)

Optimize data usage with query parameters

- use $filter query parameter
- use $select query parameter
- order results using $orderby query parameter
- set page size of results using $skip and $top query parameters
- expand and retrieve resources using $expand query parameter
- retrieve the total count of matching resources using $count query parameter
- search for resources using $search query parameter
- determine the appropriate Microsoft Graph SDK to leverage

Optimize network traffic

- monitor for changes using change notifications
- combine multiple requests using $batch
- get changes using a delta query
- detect and handle throttling

Access user data from Microsoft Graph

- get the signed in user's profile
- get a list of users in the organization
- get the user's profile photo
- get the user object based on the user's unique identifier
- get the user's manager's profile

Access files with Microsoft Graph

- get the list of files in the signed in user's OneDrive
- download a file from the signed in user's OneDrive using file unique id
- download a file from a SharePoint Online site using the relative path to the file
- get the list of files trending around the signed in user
- upload a large file to OneDrive
- get a user object from an owner list in a group and retrieve that user's files

Manage a group lifecycle on Microsoft Graph

- get the information on a group by id
- get the list of members in a group
- get the list of owners of a group
- get the list of groups where the signed in user is a member
- get the list of groups where the signed in user is an owner
- provision a group
- provision a team
- delete a group

Extend and customize SharePoint (15-20%)

Describe the components of a SharePoint Framework (SPFx) web part

- identify the appropriate tool to create an SPFx web part project
- describe properties of client-side web parts
- describe Fluent UI (Office UI Fabric) in client-side web parts
- describe when to use an app page
- differentiate between app page and web part
- describe rendering framework options
- describe branding and theming in SharePoint Online

Describe SPFx Extensions

- identify the appropriate tool to create an SPFx Extension project
- describe page placeholders from Application Customizer
- describe the ListView Command Set extension
- describe the Field Customizer extension

Describe the process to package and deploy an SPFx solution

- describe the options for preparing a package for deployment
- describe the options for packaging a solution

- describe the requirements of tenant-scoped solution deployment
- describe the requirements of isolated web parts
- describe the options to deploy a solution

Describe the consumption of Microsoft Graph

- describe the purpose of the MSGraphClient object
- describe the methods for granting permissions to Microsoft Graph

Describe the consumption of third-party APIs secured with Azure AD from within an SPFx solution

- describe the purpose of the AadHttpClient object
- describe the methods for granting permissions to consume a third-party API

Describe Web Parts as Teams Tabs

- describe how to build a Microsoft Teams tab by using SPFx
- describe the considerations for creating an SPFx Web Part to be a Teams Tab
- describe the options for deploying an SPFx Web Part as a Teams Tab

Extend Teams (20-25%)

Create a Microsoft Teams app manifest

- configure an app manifest using App Studio or Developer Portal
- manually create an app manifest to deploy an SPFx Web Part to Teams
- create an app package for a Microsoft Teams app

Deploy a Teams app

- describe the options for deploying a Teams app
- sideload an app in Microsoft Teams
- publish a Teams app to an organization app catalog

Create and use task modules

- create a card-based task module
- create an iframe-based task module
- invoke a task module from a tab
- invoke a task module from a bot
- chain task module invocations

Create a webhook

- create an outgoing webhook
- create an incoming webhook

Implement custom Teams tabs

- create a personal tab
- create a channel/group tab
- create a tab with a deep link
- implement single sign-on authentication for a tab

Create a messaging extension

- create a messaging extension search command
- create a messaging extension action command using an Adaptive Card
- create a messaging extension action command using parameters

Create a conversational bot

- create a personal bot
- create a group/channel bot
- use proactive messaging with a bot
- send Actionable Messages from a bot
- implement single sign-on authentication for a bot

Use Microsoft Graph to work with Teams

- create a team
- list teams
- configure a built-in tab
- send activity feed notification to user
- enable resource

Extend Office (15-20%)

Describe fundamental components and types of Office Add-ins

- describe task pane and content add-ins
- describe dialog boxes
- describe custom functions
- describe add-in commands
- describe the purpose of the Office Add-ins manifest

Describe Office JS APIs

- describe the Office Add-in programming model
- describe Office Add-in developer tools
- describe the capabilities of the Excel JavaScript API
- describe the capabilities of the Outlook JavaScript API
- describe the capabilities of the Word JavaScript API
- describe the capabilities of the PowerPoint JavaScript API
- describe the capabilities of Excel custom functions

Describe development of Add-ins

- describe the options for persisting state and settings
- describe Fluent UI (Office UI Fabric) in Office Add-ins
- describe when to use Microsoft Graph in Office Add-ins
- describe authorization when using Microsoft Graph in Office Add-ins

Describe testing, debugging, and deployment options

- select deployment options based on requirements
- describe testing and debugging concepts for Office Add-ins

Describe actionable messages

- describe the features of actionable messages with an Adaptive Card
- describe the scenarios for refreshing an actionable message