



## **MS-600T00: Building Applications and Solutions with Microsoft 365 Core Services**

**Course Duration:** 4 Days

**Class times:** 9am-4pm

**Course Level:** Intermediate

**Language:** English

**Mode of Training:** Virtually Instructor-Led

### **Prerequisites**

Job Role: Administrator

Related Exam: MS-600

### **Audience Profile:**

Students in this course are interested in the Microsoft 365 development platform or in passing the Microsoft Teams Application Developer Associate certification exam. Students should also have 1-2 years' experience as a developer. This course assumes students know how to code and have a basic understanding of REST APIs, JSON, OAuth2, OData, OpenID Connect, Microsoft identities including Azure AD and Microsoft accounts, Azure AD B2C, and permission/consent concepts.

### **Course Outline**

#### **Module 1: Getting Started with Microsoft Identity**

Microsoft identity platform is an evolution of the Azure Active Directory (Azure AD) developer platform. It allows developers to build applications that sign in users, and access resources in both external applications such as Microsoft Office 365, the Azure portal, and thousands of other SaaS applications as well as internal resources, such as apps on your corporate network and intranet, along with any cloud apps developed by your own organization. In this module, you will learn the basics of Microsoft identity including the different types of tokens, account types, and supported topologies.

#### **Learning objectives**

At the end of this module, you should be able to:

- Explain the different types of tokens used in Microsoft Identity solutions



- Compare and contrast the different account types.
- Compare and contrast the supported Microsoft Identity Topologies

## Module 2: Application types in Microsoft identity

The Microsoft Identity platform enables developers to build many different types of applications to satisfy diverse business requirements and different scenarios. By supporting multiple OAuth 2.0 standard authentication protocols, developers can create different types of applications that meet business needs including single page applications, web apps, mobile or native apps, and services or daemon apps. In this module, you'll learn how you can implement different OAuth 2.0 protocol grant types (flows) in popular application types.

### Learning objectives

At the end of this module, you should be able to:

- Compare and contrast different grant types (flows) in the OAuth 2.0 protocol.
- Evaluate which authentication protocol to use based on the app scenario.
- Implement different authentication protocols in different application types.

## Module 3: Permissions and Consent Framework

The Microsoft identity platform implements the OAuth 2.0 authorization protocol. This protocol is a method that a third-party app can access web-hosted resources on behalf of a user. The web-hosted resources can define a set of permissions that you can use to implement functionality in smaller chunks. Developers can leverage one of two types of permissions supported by the Microsoft identity platform depending on the app scenario. In this module, you'll learn the different types of permissions and consent framework models for obtaining permissions from users to use them in apps.

### Learning objectives

At the end of this module, you should be able to:

- Compare and contrast different permission types supported by the Microsoft identity platform.
- Compare and contrast the difference between static and dynamic consent in user permissions.



- Create an app that implements dynamic consent for incrementally obtaining permissions as needed from users.

#### **Module 4: Secure custom APIs with Microsoft Identity**

Many solutions involve creating web APIs to expose functionality to different clients and consumers. Developers can secure these APIs using Microsoft identity to ensure only approved apps can access the web APIs provided they've been granted the necessary permissions. In this module, you'll learn how to secure a web API with Microsoft identity and how to call it from another application.

##### **Learning objectives**

By the end of this module, you'll be able to:

- Create a custom web API that is secured with Microsoft identity.
- Create a custom web app that calls a custom web API secured with Microsoft identity.
- Create a custom daemon app that calls a custom web API secured with Microsoft identity.

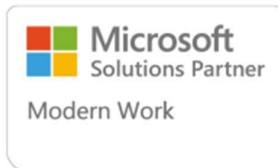
#### **Module 5: Work with users, groups, and roles in custom apps and APIs**

Role-based access control (RBAC) is a popular mechanism to enforce authorization in applications. The administrator assigns roles to different users and groups to control who can access to what content and functionality. Using RBAC with Application Roles and Role Claims, developers can securely enforce authorization in their apps with little effort on their part. Another approach is to use Azure AD Groups and Group Claims. In this module, you'll learn how to use both Azure AD Groups and Application Roles to provide fine grained access control to an application.

##### **Learning objectives**

By the end of this module, you'll be able to:

- Create a custom ASP.NET web app that is secured with Microsoft identity.
- Demonstrate how to obtain security groups as a claim in the token and use them in the app.
- Demonstrate how to use app roles to grant users access to an app.



## **Module 6: Optimize data usage when using Microsoft Graph with query parameters.**

Microsoft Graph provides a unified programmability model that you can use to build apps for organizations and consumers that interact with the data of millions of users. The Microsoft Graph REST APIs implements many of the OData protocol's query parameters. In this module, you'll learn how to manipulate queries using query parameters.

### **Learning objectives**

By the end of this module, you'll be able to:

- Demonstrate how to manipulate REST queries with query parameters.
- Create queries that expand complex entities.
- Demonstrate how to search for content with Microsoft Graph
- Optimize Microsoft Graph queries with batching.

## **Module 7: Optimize network traffic with Microsoft Graph**

Microsoft Graph provides a unified programmability model that you can use to build apps for organizations and consumers that interact with the data of millions of users. In this module, students will learn how Microsoft has implemented throttling to Microsoft Graph to limit the overuse of Microsoft Graph resources. Students will learn how to avoid requests from being throttled, and how to properly handle scenarios when Microsoft Graph throttles high user traffic in a graceful way.

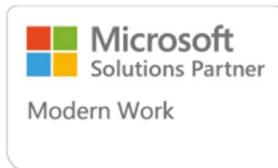
### **Learning objectives**

By the end of this module, you'll be able to:

- Explain how Microsoft Graph maintains resource health.
- Identify when Microsoft Graph throttles requests.
- Decide the appropriate pattern to address throttled requests.
- Create queries that mitigate throttling scenarios.

## **Module 8: Access User Data from Microsoft Graph**

Users are the core of most operations in Microsoft 365. Microsoft Graph enables developers full control over the lifecycle of users in Microsoft 365 including creating, updating, and deleting users



and to listing users in the organization. In this module, you'll learn how to use Microsoft Graph to work with users in Microsoft 365 including the required permissions.

### Learning objectives

At the end of this module, you should be able to:

- Demonstrate how to get a list of users.
- Demonstrate how to get details, including a profile picture, of a user.
- Demonstrate how to manage the lifecycle of a user from creation to deletion.

### Module 9: Manage Group Lifecycle with Microsoft Graph

Groups are collections of users who share access to resources in Microsoft services or in your apps. Developers can use Microsoft Graph to create and manage different types of groups. In this module, you'll learn how to manage the lifecycle of groups, the different types of groups and obtain information about the users associated with a group using Microsoft Graph.

### Learning objectives

At the end of this module, you should be able to:

- Demonstrate how to get information about a group.
- Demonstrate how to get information about a user's groups.
- Demonstrate how to manage the lifecycle of a group.

### Module 10: Access Files with Microsoft Graph

OneDrive is the files hub for Office 365. OneDrive enables users to access and collaborate on files no matter where they're stored. Microsoft Graph enables developers to use a single API to work with the files in OneDrive. Files in Office 365 are stored in drives. Users can store files in a personal drive, their OneDrive, or in a shared drive powered by a SharePoint document library. In this module, you'll learn how to access files in OneDrive, both reading and writing files, and how to traverse relationships between files and users.

### Learning objectives

At the end of this module, you should be able to:



- Demonstrate how to retrieve a list of files OneDrive.
- Demonstrate to download a file from OneDrive using a unique ID.
- Get a list of files trending around the signed in user.
- Demonstrate how to upload a large file to OneDrive.

### **Module 11: Use Change Notifications and Track Changes with Microsoft Graph**

Microsoft Graph enables developers to consume user information stored in Microsoft 365 in custom applications. Data is retrieved from Microsoft Graph through a REST API or using one of the various native SDKs provided by Microsoft. In this module, you'll learn how to work with change notifications (webhooks) & track changes (delta query) in the Microsoft Graph.

#### **Learning objectives**

At the end of this module, you should be able to:

- Demonstrate how to use the Microsoft Graph .NET SDK.
- Demonstrate how to subscribe and receive notifications when entities change with Microsoft Graph
- Demonstrate how to retrieve a list of all entities that have changed using delta query.

### **Module 12: Introduction to building apps for Microsoft Teams**

Microsoft Teams is an extensible platform you can build custom apps on. Understand what is possible with a Microsoft Teams custom app and determine if it's right for you.

#### **Learning objectives**

At the end of this module, you should be able to:

- Describe the components of a Microsoft Teams app.
- List the options for distributing a Microsoft Teams app.
- List the extensibility points available in Microsoft Teams
- Choose the correct extensibility point(s) based on your scenario.



### **Module 13: Task-oriented interactions in Microsoft Teams with messaging extensions**

In this module, you'll learn how to create different types of messaging extensions in a custom Microsoft Teams app.

#### **Learning objectives**

At the end of this module, you should be able to:

- Demonstrate how to create an action-based messaging extension for Microsoft Teams
- Demonstrate how to create search-based messaging extension for Microsoft Teams
- Demonstrate how to unfurl a URL with messaging extension for Microsoft Teams

### **Module 14: Create embedded web experiences with tabs for Microsoft Teams**

Tabs are Microsoft Teams-aware webpages embedded in Microsoft Teams. They can be added as part of a channel or a group chat inside a team or as a personal app for an individual user. As part of your app, you can add custom tabs to embed your own web content in Teams. Two types of tabs are available in Teams, channel/group and personal. In this module, you'll learn how to create tabs and add them to your Microsoft Teams app.

#### **Learning objectives**

At the end of this module, you should be able to:

- Evaluate the differences between personal and channel/group tabs.
- Create a channel/group tab with a configuration experience.
- Create a tab that uses authentication to call a protected REST API.

### **Module 15: Create interactive conversational bots for Microsoft Teams**

Conversational bots allow users to interact with your web service through text, interactive cards, and task modules. Conversational bots can be scoped to handling a few simple commands or complex, artificial intelligence powered and natural language processing virtual assistants. They can be one aspect of a larger application, or completely stand alone. In this module, you'll learn how to create and add bots to custom Microsoft Teams apps.



## Learning objectives

At the end of this module, you should be able to:

- Demonstrate how to create a conversation bot for Microsoft Teams
- Demonstrate how to subscribe to and respond to events in a conversation bot in Microsoft Teams
- Demonstrate how to incorporate authentication in a conversation bot in Microsoft Teams

## Module 16: Collect Input in Microsoft Teams with Task Modules

Task modules are modals that you can populate with either an Adaptive Card or an embedded web/content page for use in your custom Microsoft Teams app's user experience for workflows that require data input. Task modules allow you to gather information in a Teams-aware popup. In this module, you'll learn how to create and add task modules to your Microsoft Teams app.

## Learning objectives

At the end of this module, you should be able to:

- Create a task module for a tab.
- Demonstrate how to create a task module that includes an adaptive card.
- Create a task module for a bot.

## Module 17: Connect web services to Microsoft Teams with webhooks and Office 365 Connectors

Webhooks and connectors are a simple way to connect your web services to channels and teams inside Microsoft Teams. Outgoing webhooks allow your users to send text messages from a channel to your web services. Connectors allow users to subscribe to receive notifications and messages from your web services. There are two types of connectors available in Microsoft Teams: incoming webhooks and Office 365 connectors. In this module, you'll learn about webhooks and connectors and how to implement them in Microsoft Teams channels.

## Learning objectives

By the end of this module, you'll be able to:

- Demonstrate how to create and use an outgoing webhook for a Microsoft Teams channel.





- Demonstrate how to create and use an incoming webhook for a Microsoft Teams channel.
- Demonstrate how to create, register, and use an Office 365 Connector for Microsoft Teams

### **Module 18: Microsoft Teams – Use the teamwork Microsoft Graph endpoint.**

Microsoft Teams is the ultimate hub for teamwork & intelligent communications. Microsoft Teams delivers chat-based collaboration, meetings, calling, and enterprise voice features. Developers can tap into the power of Microsoft Teams with Microsoft Graph to integrate your custom applications. In this module, you'll learn how to use Microsoft Graph to interact with Microsoft Teams.

#### **Learning objectives**

At the end of this module, you should be able to:

- Understand how to use Microsoft Graph to communicate with Microsoft Teams
- Demonstrate how to get a list of all teams and a specific team with Microsoft Graph
- Demonstrate how to configure build-in tabs within a Microsoft Teams team with Microsoft Graph
- Demonstrate how to send a notification to a user or a channel using the Microsoft Teams activity feed with Microsoft Graph

### **Module 19: Microsoft Teams – Authentication and Single Sign-on**

Developers can create Microsoft Teams apps to create new experiences for their users and integrate with existing business solutions. When custom applications need to access user's information protected by Azure AD, and data from other services, apps will need to establish a trusted connection with these providers. In this module, you'll learn about the different authentication flows supported by Microsoft Teams that you can use in your custom apps.

#### **Learning objectives**

At the end of this module, you should be able to:

- Understand the authentication options available to developers creating custom Microsoft Teams tabs and bots.
- Create a Microsoft Teams tab that uses SSO authentication.
- Create a bot for Microsoft Teams that uses SSO authentication.



## Module 20: Create interactive meeting apps for Microsoft Teams

Meetings enable collaboration, partnership, informed communication, and shared feedback. The meeting app can deliver a user experience for each stage of the meeting lifecycle. Meeting lifecycle includes pre-meeting, in-meeting, and post-meeting app experience, depending on the attendee's status. In this module, you'll learn how to create custom apps to be used in Microsoft Teams meetings.

Learning objectives

At the end of this module, you should be able to:

- Understand the capabilities of Microsoft Teams meetings extensibility feature.
- Create a Microsoft Teams meetings app that meeting attendees can use before and during a meeting.
- Implement a summary view of the meeting after the meeting has concluded.

## Module 21: Distribute your Microsoft Teams app.

Use the Developer Portal to modify app definitions and manage app packages. Leverage the Microsoft Teams Framework (TeamsFx) to set up CI/CD pipelines and automate deployment.

Learning objectives

By the end of this module, you'll be able to:

- Manage app definitions and packages in the Developer Portal for Teams
- Upload a custom app to a single user or team.
- Upload an app to the App Catalogue for your organization.
- Automate deployment of Microsoft Teams apps

## Module 22: Monitor and maintain Microsoft Teams solutions.

Leverage Microsoft Purview's risk and compliance solutions and Application Insights to understand how users are interacting with your app. Use Azure Log Analytics and other Azure Monitor tools to analyze log data from your application.

Learning objectives



By the end of this module, you'll be able to:

- Recommend audit settings using Microsoft 365 Audit logging.
- Review and evaluate performance using App Insights
- Assess new SDK versions for impact to published apps.
- Analyze logs, usage data, and audit data.