



# Critical Path Training Course Description

## Modern SharePoint and Teams Development

### Mastering Today's Best Practices in Web and Mobile Development

Course Code	MSTDEV
Audience	Professional Developers
Format	Self-paced
Length	24 Hours
Course Description	<p><b>Modern SharePoint and Teams Development</b> is an intensive self-paced training course designed to teach professional developers and architects how to create custom solutions for SharePoint Online and Microsoft Teams using modern developer tools and today's best practice techniques. This course provides deep coverage of the SharePoint Framework and the Microsoft Teams development model. However, the course first spends the time to teach the essential developer prerequisites including TypeScript, Node.js, NPM, Gulp, Webpack, Visual Studio Code and the ever-popular React.js library.</p> <p>This course is well suited for experienced SharePoint developers looking to move beyond legacy development models such as SharePoint Farm Solutions and SharePoint Add-ins. In addition to learning to develop with the SharePoint Framework and Microsoft Teams, this course teaches students how to secure custom applications with Azure Active Directory and how to write code to authenticate users, acquire access tokens using the Microsoft Authentication Library (MSAL) to execute authorized web service calls against commonly-used Microsoft APIs including the SharePoint Rest API, the Microsoft Graph API and the Power BI Service API.</p> <p>This course prepares students for the Microsoft exam <b>MS-600: Building Applications and Solutions with Microsoft 365 Core Services</b>. By taking this training course and working through the lab exercises, student will learn all they need to pass this exam to become a Microsoft 365 certified developer.</p>
Student Prerequisites	Each attendee requires their own Windows PC to complete lab exercises. Attendees should have professional development experience with Visual Studio, JavaScript, C#, the .NET Framework and ASP.NET.

### Course Modules

1. Introduction to Modern Software Development
2. Developing with Node.js and Visual Studio Code
3. Developing SPAs with React and JSX/TSX
4. Introduction to the SharePoint Framework
5. Developing React Web Parts
6. Packaging and Deploying SharePoint Framework Solutions
7. Developing Secure Applications using Azure AD
8. Programming with the Microsoft Graph API
9. Developing for Microsoft Teams

## Course Module Detailed Outline

### Module 01: Introduction to Modern Software Development

This module examines SharePoint Online development strategies and discusses the evolution of modern pages and Microsoft Teams into the SharePoint user experience. Students will learn to program using the Client-side Object Model (CSOM) using C# to automate remote provisioning tasks such as creating SharePoint sites and customized lists. Next, the module provides an overview of Microsoft Azure as a cloud-based development platform and demonstrates how to create essential on-demand services such as Azure Web Apps and Azure SQL databases. The module concludes with a TypeScript primer where students will learn to develop with TypeScript in Visual Studio 2017 to write client-side code using interfaces and strongly-typed programming.

#### Topics Covered

- Understanding SharePoint Development Strategies
- Creating a SharePoint Development Environment
- Programming the Client-side Object Model (CSOM)
- Understanding Azure as a Development Platform
- Developing with TypeScript and Interfaces

#### Hands-on Lab: Creating an Office 365 Development Environment

- Exercise 1: Download a Local Copy of the Student Lab Files
- Exercise 2: Create an Office 365 Trial Account
- Exercise 3: Create New Office 365 User Accounts using a PowerShell Script
- Exercise 4: Navigate to the Root SharePoint Site for Your Tenant
- Exercise 5: Create a New SharePoint Site using the SharePoint Admin Center
- Exercise 6: Create a New SharePoint Site using PowerShell
- Exercise 7: Creating SharePoint Lists with the Client-side Object Model (CSOM)
- Exercise 8: Getting Started with an Azure Subscription
- Exercise 9: Connect to Your Microsoft Azure Subscription using PowerShell

### Module 02: Developing with Node.js and Visual Studio Code

This module introduces students to developing software projects using Node.js and Node Package Manger (npm). Students will learn to use npm together with Visual Studio Code to manage software projects and to install external packages. The module explains how to configure a new project with support for TypeScript and how to add the packages for typed definition files for popular JavaScript libraries such as jQuery. The module demonstrates configuring a project with a local web server to test and debug project files using the HTTP protocol. The module introduces students to Gulp and demonstrates how to write and run Gulp tasks to automate developer tasks such as compiling TypeScript into JavaScript and starting up a debugging session. The module concludes by examining Webpack and teaching students how to use Webpack to bundle a project's source files into a single file for distribution.

#### Topics Covered

- Introduction to Node.JS and NPM
- Installing and Updating Packages in Visual Studio Code
- Adding TypeScript Support to a Node.js Project
- Configuring Node.js with Server-side Debugging Support
- Using Gulp to Automate Running Development Tasks
- Developing Projects using Webpack

#### Hands-on Lab: Developing with Node.js, NPM and Visual Studio Code

- Exercise 1: Getting Started with Node.js, NPM and Visual Studio Code
- Exercise 2: Moving from JavaScript to TypeScript
- Exercise 3: Integrating Gulp Tasks into a Project
- Exercise 4: Using WebPack to Bundle Your Project Files for Distribution

### Module 03: Developing SPAs with React and JSX/TSX

This module introduces students to React.js and examines how React.js uses a component-based architecture and a virtual DOM to optimize performance. Students will learn to create and configure new Node.js projects as a Single Page Application (SPA) using React.js together with TypeScript and Webpack. The module walks through how to design an SPA by creating a hierarchy of React components which define properties, state and event handlers. The module introduces JSX and teaches students the essential concepts and syntax for writing TypeScript code in a TSX file to generate the HTML for a React component. Along the way, students will learn how to integrate the React Router into an SPA project to provide navigation across multiple views.

#### Topics Covered

- Getting Started with React.js
- Creating SPAs using React.js, TypeScript and Webpack
- Designing a React Component Hierarchy

- Extending a React Project with the React Router
- Understanding the React Component Lifecycle Methods
- Calling Across the Network using the Fetch API

---

**Hands-on Lab: Developing with React.js, TypeScript and Webpack**

---

- Exercise 1: Create a New Project with React.js, TypeScript and Webpack
- Exercise 2: Create a React Component Hierarchy
- Exercise 3: Extend Your React Project using the React Router
- Exercise 4: Build a User Interface to Search and View Customer Data
- Exercise 5: Retrieving Data from an OData Web Service using the React Fetch API

## Module 04: Introduction to the SharePoint Framework

This module introduces students to the SharePoint Framework (SPFX) and the extensive API it provides for client-side development. Students will learn to create new SPFX projects using the Yeoman generator and to develop SPFX projects using Visual Studio Code. The module examines the different types of components that can be created with SPFX including Web Parts, Application Customizers, Field Customizers and Command Sets. Students will learn how to extend a Web Part with custom properties that can be viewed and edited by users in the Web Part Properties Pane. The module also teaches best practices for managing CSS styles in an SPFX project using SCSS files and CSS modules. Students will learn how to test and debug SPFX projects in the local SharePoint Workbench as well as in the hosted SharePoint Workbench running inside a test site in SharePoint Online.

---

**Topics Covered**

---

- Introduction to the SharePoint Framework
- Creating SPFX Projects using the Yeoman Generator
- Testing & Debugging Webparts in SharePoint Workbench
- Managing Styles using SCSS Files and CSS Modules
- Creating a Web Part with Custom Properties
- Creating Application Customizers

---

**Hands-on Lab: Getting Started with SharePoint Framework**

---

- Exercise 1: Create an SPFX Project using the Yeoman Generator
- Exercise 2: Test and Debug a Webpart in the Local SharePoint Workbench
- Exercise 3: Extend an SPFX Webpart with Custom Properties
- Exercise 4: Test a SharePoint Framework Webpart in SharePoint Online
- Exercise 5: Create an Application Customizer using SPFX

## Module 05: Developing React Web Parts

This module moves beyond SPFX fundamentals to examine the process of developing SPFX Web Parts for real-world scenarios. The module explains how to develop React Web Parts with a design that coordinates the movement of data between the Web Part class which defines persistent properties and the React component class which defines state. The module also teaches students how to leverage the Office UI Fabric React library to create a user interface using standard React components such as PrimaryButton, TextField and DetailsList. The module demonstrates how to develop Web Parts which use the SharePoint REST API to create new SharePoint lists and to update and query SharePoint list items. Along the way, students will learn essential SPFX programming techniques for executing calls asynchronously and for updating the user interface with a loading indicator whenever the user is waiting for a call to return from across the network.

---

**Topics Covered**

---

- Designing and Developing React Web Parts
- Web Part Properties versus React Component State
- Leveraging the Office UI Fabric React Library
- Developing Web Parts using the SharePoint REST API

---

**Hands-on Lab: Developing React WebParts**

---

- Exercise 1: Create a React WebPart Styled using the Office UI Fabric
- Exercise 2: Create a React Webpart with a Synchronized Property
- Exercise 3: Extend The React Webpart using the DetailsList React Component
- Exercise 4: Call the SharePoint REST API from a SharePoint Framework Webpart
- Exercise 5: Extend The Webpart to Dynamically Switch Between Contacts Lists

## Module 06: Packaging and Deploying SharePoint Framework Solutions

This module examines the process of application lifecycle management (ALM) with SharePoint Framework solutions. The module explains the role of the App Catalog and walks through how to create a new App Catalog site for a SharePoint Online tenant. Students will learn how to bundle and package a SPFX solution for distribution and to optimize SPFX builds for a production environment. The module explains how to package third-party JavaScript libraries as external references and how to deploy SharePoint Framework solution resources to the Office 365 CDN. Students will also learn how to publish SharePoint Framework solution packages in the App Catalog as well as how to install a SPFX solution in a SharePoint site. The module also explains the process of upgrading an SPFX solution after it's has already been deployed to a production environment.

### Topics Covered

---

- Creating the App Catalog Site in SharePoint Online
- Packaging SPFX Solutions for Distribution
- Publishing and Updating SPFX Solutions
- Deployment using a Site Collection App Catalog

### Hands-on Lab: Packaging and Deploying SPFX Solutions

---

- Exercise 1: Create an App Catalog Site Collection
- Exercise 2: Build and Deploy a Solution Package with a Webpart
- Exercise 3: Build and Deploy a Solution Package with an Application Customizer

## Module 07: Developing Secure Applications using Azure AD

This module begins with a primer on OAuth 2.0 and OpenID Connect and an overview of the Azure Active Directory security model which provides support for user authentication, application authentication and an authorization scheme based on configurable permissions. Students will learn about the differences between application permissions and delegated permissions as well as how to create and configure Azure AD applications in the new Azure portal. Students will learn programming techniques for developing secure applications which implement common authentication flows such as user credentials flow, authorization grant flow, and client credentials flow. The module examines developing secure ASP.NET MVC application by using the Active Directory Authentication Library (ADAL) together with the OWIN framework and OWIN middleware components. The module also explains how to secure client-side SPAs created Angular and AngularJS by using the ADAL.js library and the implicit grant flow to acquire access tokens.

### Topics Covered

---

- Understanding OAuth 2.0 and OpenID Connect
- Creating & Configuring Azure AD Applications
- Acquiring Access Tokens with MSAL
- Implementing Implicit Flow
- Implementing Authorization Code Flow

### Hands-on Lab: Authenticating with Azure Active Directory

---

- Exercise 1: Register a New Application with Azure Active Directory
- Exercise 2: Call the Microsoft Graph API using Direct REST Calls
- Exercise 3: Calling to the Microsoft Graph using the .NET SDK

## Module 08: Programming with the Microsoft Graph API

This module introduces the Microsoft Graph API and explains how this powerful library abstracts away the divisions between Azure Active Directory, Exchange Online, SharePoint Online and OneDrive to create a single, unified API for general application development. Students will learn how to program against the Microsoft Graph API in an SPFX Web Part using both the AadHttpClient class and the MSGraphClient class. The module also explains why it's necessary to add API permission request into the package-solution.json file for an SPFX solution and how to grant tenant-level API permissions in the SharePoint Admin Center. Along the way, students will learn how to program the Microsoft Graph API to view and create users in Azure Active Directory and to read and send email messages and calendar events using an Exchange inbox.

### Topics Covered

---

- Overview of Microsoft Graph API
- Constructing URLs for the Microsoft Graph API
- Developing Applications with the Microsoft Graph API
- Programming SPFx Webparts using MSGraphClient
- Granting Microsoft Graph API Permissions

### Hands-on Lab: Developing Webparts with the Microsoft Graph API

---

- Exercise 1: Create a React Webpart that Calls the Microsoft Graph API
- Exercise 2: Deploy the Solution Package and Grant Microsoft Graph API Permissions

## Module 09: Developing for Microsoft Teams

This module introduces the Microsoft Teams service and explains the why and the how of creating Teams and Channels when rolling out Microsoft Teams in an Office 365 organization. The module also examines the Microsoft Teams developer platform and explains how to develop Microsoft Teams apps with custom tabs, bots, connectors and messaging extensions. Students will learn how to develop and test an app for Microsoft Teams using App Studio. Students will also learn to develop Microsoft Team apps using Visual Studio 2017 and C# as well as with Node.js, Visual Studio Code and the Microsoft Teams JavaScript Client SDK. Along the way, students will learn advanced development techniques with Microsoft Teams including developing custom messaging extensions and posting dynamic content to an activity feed.

### Topics Covered

---

- Introduction to Microsoft Teams
- Developing with the NGROK Utility
- Developing Static Tabs and Dynamic Tabs
- Developing Tabs with Node.js, TypeScript and React.js
- Getting Started with App Studio
- Developing Incoming Webhooks
- Developing Bots for Microsoft Teams

### Hands-on Lab: Developing for Microsoft Teams

---

- Exercise 1: Configure Your Environment for Microsoft Teams Development
- Exercise 2: Test a Microsoft Teams App with Custom Tabs
- Exercise 3: Create a Custom Teams App using Node.js and Visual Studio Code
- Exercise 4: Working on Additional Teams Lab Exercise provided by Microsoft