



"Charting the Course ...

protechtraining.com

... to Your Success!"

## Build and Release Management using TFS 2015

### Course Summary

#### Description

Team Foundation Server 2015 and Visual Studio Team Services offer a completely redeveloped build system and Microsoft is in the process of rolling out a new system for Release Management.

This 2-day course focuses on building and releasing .NET applications using these new systems. It focuses on the new scriptable, cross-platform build system introduced in TFS 2015 and how to use it. The course briefly looks at the current version of Release Management for Visual Studio 2015 but then deep dives into Release Management vNext. Release Management vNext is currently in preview in Visual Studio Team Services and in TFS 2015 Update 2.

If you are eager to learn how to implement DevOps practices designed to streamline your build and release processes using TFS or VSTS, this is the course for you.

The content in this course is a subset of our 3-day "DevOps using Visual Studio 2015" course.

This course includes hands-on labs to reinforce practical skills and ensure you're ready to use the tools on your return to your workplace.

#### Objectives

At the end of this course, students will be able to:

- Describe the current features in TFS or VSTS focused on build and release management
- Plan and configure a scalable build environment including pools and queues
- Configure appropriate security on pools and queues
- Author build definitions
- Configure Continuous Integration (CI) builds as well as scheduled builds
- Use existing build templates as well as create new build templates
- Create custom build workflow using build steps
- Automate the execution of tests as part of our build
- Understand the features of the current version of Release Management for Visual Studio
- Use Release Management vNext to create and manage releases
- Use Release Management vNext to create environments
- Manage configuration information in each environment
- Deploy software to both on-premises servers as well as to Azure
- Configure Continuous Release by triggering releases from builds
- Create approval workflows and track releases across environments

#### Topics

- A Lap Around the Microsoft Visual Studio 2015 Family
- Configuring your Build environment
- Authoring Our First Build Definition
- A Closer Look at Builds
- Customizing Your Build Process
- Other Build Enhancements and Features
- Overview of Release Management
- Release Management for Visual Studio 2015
- Release Management vNext
- Configuring Environments
- Defining a Release Workflow for On-Premises Servers
- Defining a Release Workflow for Azure



*"Charting the Course ...*

*... to Your Success!"*

## **Build and Release Management using TFS 2015**

### **Course Summary (cont'd)**

#### **Audience**

This course is designed for .NET developers, TFS Administrators and Build masters. It is a hands-on look at various technologies rather than a high-level management overview.

#### **Prerequisites**

Attendees should be familiar with Visual Studio and the basic use of Team Foundation Server or Visual Studio Team Services.

#### **Duration**

Two days

## Build and Release Management using TFS 2015

### Course Outline

- I. **A Lap Around the Microsoft Visual Studio 2015 Family**
  - A. Overview of the Visual Studio 2015 family
  - B. Overview of product features
  - C. The need for centralized builds
  - D. The importance of release management
- II. **Configuring Your Build Environment**
  - A. Understanding the new 2015 Build infrastructure
  - B. Creating Agent pools and queues
  - C. Installing Build Agents
  - D. Specifying capabilities on your Build Agents
  - E. Configuring security
  - F. Overview of non-Windows Build Agents
  - G. Overview of hosted build infrastructure
- III. **Authoring Our First Build Definition**
  - A. Creating a new build definition
  - B. Understanding Build templates
  - C. Selecting the solution to build
  - D. Running tests as part of your build
  - E. Queuing a build
- IV. **A Closer Look At Builds**
  - A. Selecting our source repository (TFVC/Git/GitHub/Subversion)
  - B. Running multiple concurrent builds
  - C. Working with Build variables
  - D. Configuring Continuous Integration (CI) and scheduled builds
  - E. Using Gated Check-ins
  - F. Using demands to influence agent selection
  - G. Build retention policies
- V. **Customizing Your Build Process**
  - A. A lap around the available build tasks
  - B. Using Build Tasks to deploy your application components
  - C. Extending builds using scripts
- VI. **Other Build Enhancements and Features**
  - A. Viewing build status
  - B. Controlling build outputs and understanding artifacts
  - C. Packaging your .NET application
  - D. Examining test results from a build
- VII. **Overview of Release Management**
  - A. What is Release Management
  - B. Key concepts in Release Management
  - C. The current state of Release Management in TFS/VSTS
- VIII. **Release Management for Visual Studio 2015**
  - A. Release Management for Visual Studio 2015
  - B. Creating environments and releases
  - C. Configuring approval workflows
  - D. Triggering and tracking releases
- IX. **Release Management vNext**
  - A. Authoring a new release definition
  - B. Linking your build definition to a release
  - C. Determining what to deploy using artifacts
  - D. Defining and using configuration variables
  - E. Configuring Continuous Release
  - F. Naming your release and using retention policies
- X. **Configuring Environments**
  - A. Understanding environments
  - B. Configuring your approval workflow
  - C. Notifying approvers
  - D. Selecting your deployment queue
  - E. Defining custom variables for your environments
  - F. Configuring the environment owner
  - G. Understanding deployment conditions
  - H. Cloning environments
- XI. **Defining a Release Workflow for On-Premises Servers**
  - A. Overview of release tasks
  - B. Copying files and running scripts
  - C. Releasing a web application to an IIS server
  - D. Executing tests as part of our on-premises release
  - E. Tracking our release
- XII. **Defining a Release Workflow for Azure**
  - A. Understanding Azure resources
  - B. Understanding Service Endpoints
  - C. Configuring endpoints for connecting to Azure
  - D. Releasing an Azure Web Application
  - E. Executing tests as part of our Azure release