



## Introduction to Visual Studio Team System

This 4-day instructor led class is designed to teach students how to adopt Visual Studio Team System within their organizations.

**Audience:** Experienced Visual Studio users interested in learning about the features of Visual Studio Team System.

# Team System

Notion Solutions, Inc is a leading provider of consulting, training and application development for Microsoft's Visual Studio Team System Tools.

### Course Description:

This course covers the major topics for using Team System for software projects. This is an introductory level course and is focused on teaching all aspects of Team Foundation Server and the various Visual Studio Team Editions.

### At Course Completion:

Students will understand the capabilities of the tools in Team System and how they support the SDLC. Students will learn how to plan and create a team project, setup & modify work items, administer source control, use the design and modeling tools, write better code by setting up unit tests and analyzing their code, make and test database changes, setup an automated build, create and run manual tests, web tests and load tests. As well as modify the process template and run reports from the data warehouse.

### Team System Overview

Development challenges  
Features  
Role-based approach  
Packaging  
[LAB](#)

### Team Projects and Process

Overview of Team Projects  
How process is integrated with Visual Studio Team System  
Overview of MSF for Agile Software Development  
Overview of MSF for CMMI Process Improvement  
[LAB](#)

### Version Control Overview

Architecture overview  
Integrated check-in  
Parallel development  
Atomic check-in  
Remote development  
Check-in notes and notification  
Check-in policy

### Design and modeling tools

Using the Logical Datacenter Designer  
Using the Application Designer  
Using the System Designer  
Using the Deployment Designer  
Using the Class Designer  
[LAB](#)

### Managing Database Change

Creating database projects  
Using rename refactoring  
Using data generation plans  
Database unit testing  
Comparing data  
Comparing schema  
Deploying database changes  
[LAB](#)

### Writing Quality Code

Tools for writing good tests  
Code review  
Static code analysis  
Application verifier  
[LAB](#)

### Unit Tests

Test driven development  
Unit testing  
Code coverage  
[LAB](#)

### Code Profiling

Using the code profiler to find performance problems  
[LAB](#)

### Build Process

Team Build architecture  
Creating build scripts  
Executing builds  
Executing command-lines tasks  
Scheduling builds  
Continuous integration  
[LAB](#)

### Quality Assurance & Testing

Overview of Testing  
Test Manager  
Writing manual tests  
Executing manual tests  
Reporting defects  
[LAB](#)

### Web Testing

Creating and executing web application tests  
Using extraction and validation rules  
Creating & executing web services tests  
Test Run Configurations  
Data Driven Web Tests  
[LAB](#)

### Load Testing

Creating & executing web application load tests  
Using the load agent  
Analyzing results  
Creating & executing web service tests  
[LAB](#)

### Advanced Version Control

Shelving  
Branching and merging  
Multiple Solutions  
Common Admin Tasks  
[LAB](#)

### Modifying Work Item Definitions

Changing work item definitions on live projects  
Adding workflow  
Controlling field access  
[LAB](#)

### Modifying Process Templates

Modifying document templates  
Modifying work items  
Modifying MS Project synchronization  
Using custom processes with team projects  
[LAB](#)

### Building Reports

[LAB](#)