



Leading a Development Team

COURSE DESCRIPTION

This three-day course is designed to teach first-line managers or team leaders how to manage projects quantitatively in order to complete projects on schedule, within budget, and with all requirements met. The course covers the knowledge and skills leaders need to effectively lead and coach development teams.

Quality is not achieved by accident. Quality must be planned for and tracked to ensure project goals are achieved. This course provides examples of practical leading and lagging quality indicators, which leaders can use in guiding their decisions. By controlling quality, schedule and budgets become more predictable. The relationship between quality and its impact on cost and schedule is also discussed, along with quantitative techniques used to manage them.

Software is developed by teams of individuals, thus processes are needed in order to successfully develop quality software. However, organizations still struggle when trying to apply disciplined methods in the software process. Historically, this struggle has been due to a lack of operational procedures that teams and individuals can use to develop software in a disciplined fashion. Through the use of the Team Software ProcessSM(TSPSM) leaders will be given an example of both a strategy and a set of operational procedures using disciplined software process methods at the individual and team levels. At the individual level, the Personal Software Process (PSPSM) shows managers and engineers how and why to plan and track their projects, demonstrates the principles and benefits of effective quality management, and involves the engineers in process measurement, management, and improvement. At the team level, the TSP builds accurate, achievable plans for software project teams, provides a formal team-building process, and provides the mechanism for tracking progress against project plans.

AUDIENCE

First-level software managers who directly manage software development, such as

- software project managers
- software team leaders
- supervisors



PREREQUISITES

There are no prerequisites for this course.

TOPICS

- Discuss how to measure and manage development quality and produce quality products using quantitative methods
- Provide an overview of the team leader's responsibilities and ways to lead and motivate the team
- Discuss how to effectively build work plans and negotiate commitments with management
- Understand quantitative process data at the team level and how to lead the team based on the data
- Discuss how to track the team's status against its plan and maintain its commitment to management
- Discuss maintaining quality during software development
- Show how to report status to management
- Discuss how to handle leadership challenges
- Provide an overview of the TSP and how it works to build self-directed, high-performance teams
- Show how team management works on a TSP team
- Discuss the role of the team leader

OBJECTIVES

Successful completion of this course enables participants to

- describe how the use of operational procedures and data can be effectively used by teams and individuals, can improve software development activities and provide positive motivation for engineers and teams
- demonstrate key behaviors for successfully leading and coaching teams
- show how to quantitatively manage projects through the use of effective cost, schedule and quality measures
- describe the basic concepts on which the TSP is built
- provide a working-level understanding of how individuals and teams apply the TSP



PSP for Engineers I: Planning

PSP for Engineers II: Quality

COURSE DESCRIPTION

This two-week course composed of the Personal Software ProcessSM (PSPSM) for Engineers I: Planning course and the Personal Software Process (PSP) for Engineers II: Quality course, teaches software engineers the principles, concepts, and benefits of the PSP, a process-based approach for developing software. Students who complete the course will be able to apply the PSP methods to their own personal work process and participate on a Team Software ProcessSM (TSPSM) team. Students learn how to measure and analyze their personal software processes, use process data to improve their personal performance, and apply PSP methods to other structured tasks.

To be considered a PSP-trained engineer, a student must complete both PSP for Engineers I: Planning and PSP for Engineers II: Quality. Students learn the PSP by completing the assigned reading, attending the class lectures, and completing the assignments. Students should plan to spend from 100 to 125 hours in total to complete PSP for Engineers I and II. The total course completion time includes two weeks of class time plus additional time to complete the pre-reading and course homework.

AUDIENCE

- software engineers
- software engineering instructors
- software quality management instructors
- third-party vendors of software engineering or quality training

PREREQUISITES

Before registering for these courses, participants must be proficient in at least one programming language. Due to the quick pace of the course, students will not be successful if they attempt to learn a new programming language while attending. Students must bring with them the following portable computing equipment:



- Laptop computer with 110 megabytes or greater of free hard disk space
- Microsoft Office with Access 2003
- Microsoft Office Service Pack 2
- Programming language development environment (compiler, editor, etc.)
- Zip file capability
- CD-ROM or memory stick
- a simple 4-function calculator
- Please note: Beta versions of Microsoft Office products are not supported.

TOPICS

- Introduction to PSP and TSP
- Process and Size Measurement
- size and resource estimating
- proxy-based estimating (PROBE)
- schedule planning and tracking
- Using PSP Data
- Software Quality
- Defect Management
- The Design Process
- Design Verification
- Using the PSP

OBJECTIVES

- understand the PSP process-based approach for developing software
- measure and analyze their personal software processes
- use process data to improve their personal performance
- apply PSP methods to other structured tasks.