

Course Name: A Practical Approach to Controlling Your Software Development, Maintenance & Management Processes

Duration: 1 Day

Medium of Instruction: Cantonese (with handout in English)

Award of Certificate: Certificate of Attendance

Nature and Objectives:

Software development, maintenance and management are important tasks to many IT shops. In fact, these tasks are complicated processes. Poor skills and bad management of these processes usually lead to mismatching of user requirements, schedule slippage, uncontrolled budgets and eventually project failures. Though there are a lot of literature and models addressing the issues, they are either too theoretical or only applicable to very large-scale projects. The course aims to provide participants with a practical methodology to develop, implement, maintain and enhance a software application system. Throughout the course, a case study with a set of templates will be used to demonstrate the methodology.

The course will introduce necessary knowledge and skills in practical, flexible and systematic ways for participants to plan, manage and implement software development and maintenance processes. In addition to concepts, cases studies, workshops and experience sharing will be used to demonstrate the ways in which the concepts and theory are implemented.

Who Should Attend:

IT Practitioners who are involved in software development and maintenance will find this course valuable.

Course Outline:

Managing a Software Project

- Project Management Processes
- Software Processes
- Measures, Metric and Estimation
- Software Project Planning
- Risk Analysis and Management
- Project Scheduling, Tracking and Control
- Software Quality Assurance
- Software Configuration Management and Change Control

System and Software Requirement Analysis

- Identification of Needs
- Feasibility Study and Technical Analysis
- Analysis Methodologies (Conventional and Object Oriented)
- Requirements Analysis and Definition
- Converting Requirements into Functional Design
- Process Reengineering/Improvement

Design and Implementing the Design

- System Planning
 - Establishing Design Standards
 - Design Concepts and Principles
- Design
 - Software Architecture Design
 - User Interface Design
 - Database Design
 - Component-Level Design
- Implementing the Design: the Translation Process

- Software Testing

System Implementation

- Data Migration
- Train the Users and User Acceptance
- Defining Implementation Strategies

Case Study

Maintenance

- The Maintenance Cycle
- The Maintenance Characteristics

Software Configuration Management

- Configurable Items
- Definitive Software Library
- Different Environments
- Version control and management

Change Management

- Type of changes and Classification
- The change advisory board (CAB)
- Change Management Activities

Release Management

- Basic Concept
- Release Policy and Planning
- Testing and Release Acceptance
- Rollout Planning
- Communication. Preparation and Training
- Distribution and Installation