

€ TRAINING

Certified Scrum Developer CSD

A photograph of four smiling professionals in a meeting. A woman in a black top and beaded necklace is in the foreground, looking towards the camera. Behind her are three other people (two men and one woman) in white shirts, looking towards the right. The background is a bright, modern office setting. A large blue curved graphic element is overlaid on the top and right sides of the image.

30 September -
4 October 2024
Amsterdam (Netherlands)



Certified Scrum Developer CSD

REF: B2216 DATE: 30 September - 4 October 2024 Venue: Amsterdam (Netherlands) - Fee: 6145 Euro

Introduction:

This training program provides participants with essential knowledge and skills to become a Certified Scrum Developer CSD. It empowers them to understand Agile principles, Scrum framework, and practices essential for effective software development within Scrum teams.

Program Objectives:

At the end of this program, participants will be able to:

- Understand Agile values and principles.
- Apply Scrum practices and ceremonies in software development.
- Collaborate effectively within Scrum teams.
- Develop high-quality software using Agile engineering practices.

Targeted Audience:

- Software Developers.
- QA Engineers.
- Technical Leads.
- Personnel involved in Agile software development processes.

Program Outline:

Unit 1:

Agile and Scrum Fundamentals:

- Introduction to Agile methodologies.
- Agile Manifesto and principles.
- Scrum framework overview: roles, events, and artifacts.
- Scrum values and their importance in team dynamics.
- Comparing Scrum with other Agile frameworks Kanban, XP.

Unit 2:

Agile Software Development Practices:

- User Stories: writing, estimating, and refining.
- Sprint Planning: preparing backlog items for development.
- Daily Stand-ups: conducting effective daily Scrum meetings.
- Sprint Review and Retrospective: evaluating sprint results and continuous improvement.
- Definition of Done DoD and its role in delivering increments.

Unit 3:

Test-Driven Development TDD:

- TDD principles and benefits.
- Red-Green-Refactor cycle in TDD.
- Writing unit tests using testing frameworks JUnit, NUnit.
- Integration of TDD with Scrum development process.
- Continuous integration and automated testing practices.

Unit 4:

Pair Programming and Code Reviews:

- Pair Programming: benefits and techniques.
- Conducting effective pair programming sessions.
- Code Reviews: principles and best practices.
- Peer feedback and collaborative code improvement.
- Tools and techniques for code collaboration GitHub, Bitbucket.

Unit 5:

Agile Software Design and Refactoring:

- Principles of Agile software design.



- Refactoring techniques and patterns SOLID principles.
- Implementing design patterns in Agile projects.
- Emergent design and evolutionary architecture.
- Balancing flexibility and maintainability in Agile development.