



Training

A1.SW1 - Software Engineering Foundations

Keyfacts

Duration

3 days

Language

English or German

Setting

On-site or remote

Target Group

Engineers who are working in the field of Software Engineering without having a theoretical background in Software Engineering

Training Goals

The ultimate goal of this training is to help engineers working in the field of software engineering to further improve their software engineering skills. In other words, it aims at turning "Programmers" into "Software Engineers".

This training will take you on a journey through all steps of Software Engineering, helping you to identifying blind-spots and providing you the fundamental concepts for all steps of Software Engineering. The particular learning goals of this training are:

- #1 - Understanding the challenge: Fundamental aspects of Software and Software Engineering**
Participants are aware of the delta between "programming" and "Software Engineering" in an industrial environment. They understand the specific challenges introduced by Software and know approaches how to deal with them
- #2 - Holistic Picture of Software Engineering Tasks and Methods**
Participants gain a holistic picture of all different tasks that have to be solved in Software Engineering. They know fundamental concepts and are provided with pointers for individual topics to further deepen their individual learning
- #3 - Reflection on the Role of a Software Engineer**
Participants understand the role and responsibilities of Software Engineers as part of a greater team. They are aware of concepts (i.e., Clean Coder) how to continuously improve the performance of their Software Engineering team.



Training Content

Foundations on Software and Software Engineering

- What makes Software special? (It is; really!)
- Which challenges need to be addressed by Software Engineering, besides programming?

Requirements Engineering

- Requirements Engineering as engineering discipline
- Fundamental concepts and methods of Requirements Engineering
- Specification of Quality Characteristics and Dependability

Software Architecture

- Fundamental concepts of architecture
- Architecture Description, Viewpoints and Model Kinds
- Architecture Development Methods (Twin-Peaks, Zig-Zag Pattern, FAS)
- Making good Architecture

Software Design and Design for X

- Designing Software to fit Architectural Drivers
- Challenges in interdisciplinary Software Design



Training Content	<p>Modern Software Engineering Methods</p> <ul style="list-style-type: none">• Agile, Continuous-X, and DevOps• Organizational aspects of Software Engineering <p>Integration, Verification, and Validation (IVV)</p> <ul style="list-style-type: none">• Increasing importance of integration• Constructive, Analytic, and Organizational Quality Assurance• "Shift Left" as key to quality <p>Platform und Variante-Management</p> <ul style="list-style-type: none">• Fundamental Concepts for Platform and Variant-Management• Feature Trees und Variable Architekturen <p>The Role of Software Engineers</p> <ul style="list-style-type: none">• Tasks, Responsibilities, and Authorization• The concept of a Clean Coder
Learning Methods and Didactics	Combination of theory inputs, discussions/reflections, examples and small awareness exercises

Your Benefit

In context of Cyber-Physical Systems, many people working with Software have a non-Software background. As such, it is difficult to achieve high quality Software in the face of increasing complexity. This training helps you to identify existing Blindspots and to sketch a way on how to evolve from a "Programmer" into a "Software Engineer" by learning all the necessary concepts around programming that are required to achieve high quality software.

Your Trainer

FH-Prof. Dr. Christian Neureiter
neureiter@successfactory.cc



Christian is Professor at the School of Information Technology and Digitalisation at Salzburg University of Applied Sciences. As head of the "Center for Dependable Systems Engineering" he is an expert in this field and has profound knowledge on the matter.

Asides his academic role, Christian has 10+ years of experience as consultant and trainer

at the Successfactory Consulting group with a particular focus on Leadership, Software, and Systems Engineering related topics.