Training Courses

2017-2018

T-AirTech
Software Designer

- **Title: Introduction to Requirements Engineering**
- *Description*
  - Objective: Introduce the notion of requirements, methods to identify, define, approve and manage requirements.
  - Audience: People interested in requirement management and traceability methodologies
  - Duration: 1 day.
  - Prerequisites: None.
- *Course overview*
  - Introduction to Requirements
    - What are requirements?
    - Quality of requirements
    - Stakeholders in the requirement lifecycle
  - Requirements Engineering Activities
    - Eliciting, analyzing, documenting and checking completeness of requirements
    - Refining, categorizing, decomposing, validating and tracing requirements
  - Requirements Evolving
    - Process: from customer to design requirements, modeling, verification
    - Changes management and traceability (vertical vs horizontal)
  - Examples and exercises.

Requirements Engineering

- **Title: Learn Reqtify**

- *Description*
  - Objective: Learn to setup a project, to run analysis on requirements traceability, and to correctly interpret the results.
  - Audience: Anyone starting with Reqtify.
  - Duration: 2 days.
  - Prerequisites: Basic knowledge in requirements management and traceability.
- *Course overview*
  - Introduction & Setup
    - Requirements management and traceability
    - Reqtify architecture, functionalities and quick setup
  - Explore & Analyze
    - Data and computation: explore the various types of analysis
    - Use the rules checker to validate the data consistency
    - Understand how to perform an impact analysis
  - Maximize the use of the software
    - Analysis filtering, changes management and reports generation
    - Additional useful Reqtify tools and features: collaboration, tagger, reviewer, and more
  - Exercises may be adapted to the customer needs.

Learn Reqtify

- **Title: Manage Reqtify**

- *Description*
  - Objective: Learn to configure Reqtify: projects setup, export and reports generation.
  - Audience: Anyone wishing to setup and configure Reqtify projects.
  - Duration: 2 days.
  - Prerequisites: Good knowledge of Reqtify. For beginners, "Learn Reqtify" course is recommended first.
- *Course overview*
  - Project Setup
    - Customization of the pre-processing phase: types of analysis and Regular Expressions
    - Process rules, data accessibility, import (binary/text) and synchronization
  - Export and Reporting
    - Customization of the post-processing phase: export analysis
    - Data export and customized reports: build models
  - Quick introduction to scripting
    - Conditions to generate report elements
    - Create custom rules
  - Exercises may be adapted to the customer needs.

Manage Reqtify

- **Title: Reqtify - OTScript language**

- *Description*
  - Objective: Maximize the Reqtify experience by exploring OTScript language capabilities.
  - Audience: Anyone using Reqtify.
  - Duration: 1 day.
  - Prerequisites: Advanced knowledge of Reqtify. Programming skills recommended.
- *Course overview*
  - What is the OTScript?
    - Principles and concepts: object oriented language.
    - Software architecture, methods and API.
  - When should we use the OTScript?
    - Advanced reports generation: collection add-ons, embedded algorithms
    - Advanced rules customization: methods dependencies
    - Advanced usage:
      - Analysis automation: batch scripting
      - Customized converters and post-processing algorithms
      - User interface customization
  - Exercises may be adapted to the customer needs.

OTScript language

- **Title: Reqtify - Stay Up to Date**
- *Description*
  - Objective: Discover the latest features of Reqtify, and the solutions on known and "unknown" issues.
  - Audience: Anyone using Reqtify.
  - Duration: 2 hours.
  - Location: Web based training
  - Prerequisites: Knowledge of Reqtify is recommended.
- *Course overview*
  - What's New in version 2018 FD01?
    - Presentation of the latest features and interfaces.
    - Examples.
  - Known and "unknown" issues?
    - Identification of issues for the 2018 FD01 version
    - Workarounds and solutions
  - Questions and Answers

Stay up to date with Reqtify

- **Title: Co-simulation Introduction**
- *Description*
  - Objective: Learn how and when the co-simulation becomes a real benefit for complex systems simulation.
  - Audience: Anyone involved in system/sub-system design and simulation.
  - Duration: 2 days.
  - Prerequisites: Knowledge of system design and simulation processes.
- *Course overview*
  - Why the co-simulation?
    - Multi-domains simulation: understanding of physics interactions
    - Principles of co-simulation: loose/strong coupling, communication steps.
    - Co-simulation applications: bridge, bus and API.
  - Interoperability with a Simulation environment
    - Simulation engine: event-based, solver.
    - Solver: ordinary differential (ODE) and differential algebraic equations (DAE).
    - Interoperability: synchronization methods, accuracy and stability.
  - Application
    - Quick introduction on FMI (Functional Mock-up Interface) for co-simulation.
    - Use case: a 3 instances co-simulation platform: 2 FMI, 1 Simulink.
  - Conclusion and Perspectives
  - Exercises may be adapted to the customer needs.

Co-simulation Introduction

- **Title: Matlab Introduction**

- *Description*
  - Objective: Discovery of the software suite: user environment, programming, graphics, data management.
  - Audience: People starting with Matlab.
  - Duration: 1 day.
  - Prerequisites: Basic knowledge in mathematics and computer programming.
- *Course overview*
  - Introduction to the software suite
    - Description of the programming environment, getting help, files and directories, workspaces.
    - Libraries, toolboxes and blocksets, a quick overview of Simulink.
  - Matlab, a programming language
    - Data types, variables allocation, accessing and modifying the values, operations.
    - Vectors and matrices: operations and functions, calculation, statistics.
    - Additional languages elements and functions.
  - Data analysis: plotting and visualization
    - Creation of 2D/3D graphs, edition, exportation and saving.
    - Use of external data: CSV, Excel files.
  - Exercises may be adapted to the customer needs.

Matlab Introduction

- **Title: Matlab Introduction Plus**

- *Description*
  - Objective: Learn more about scripting and programming features with Matlab.
  - Audience: Anyone using Matlab.
  - Duration: 1 day.
  - Prerequisites: Basic knowledge of Matlab. "Matlab Introduction" course recommended.
- *Course overview*
  - Introduction to the Matlab scripts
    - M files structure, running the scripts.
    - Publish and deploy scripts.
    - Functions and code sections in scripts.
  - More programming features
    - Strings: declaration, build strings, comparison, conversion.
    - I/Os functions: read/write binary and ASCII files.
    - Code optimisation: pre-allocation, profiling.
  - Interactions with the user
    - Creation and management of default and customized dialog boxes.
  - Exercises may be adapted to the customer needs.

Matlab Introduction Plus

- **Title: Advanced concepts of Matlab**

- *Description*
  - Objective: Master Matlab and its programming to design complex algorithms.
  - Audience: Anyone wishing to program with Matlab.
  - Duration: 1 day.
  - Prerequisites: Good knowledge of Matlab. "Matlab Introduction" and "Matlab Introduction Plus" courses recommended.
- *Course overview*
  - Advanced data types
    - Cell arrays to merge data with different data types.
    - Structures of data: declaration, management and access.
  - Advanced functions
    - Arrays extension
    - Functions with variable parameters, generic functions.
  - Robustness, code optimization and parallel computing
    - Errors management and programming advices.
    - Code optimization: pre-allocation, profiling.
    - Introduction of the parallel computing toolbox: notion of workers.
  - Exercises may be adapted to the customer needs.

Advanced concepts of Matlab

- **Title: Simulink Introduction**

- *Description*
  - Objective: Discovery of the graphical environment of Matlab: Simulink.
  - Audience: People starting with Simulink.
  - Duration: 1 day.
  - Prerequisites: Basic knowledge of Matlab. "Matlab Introduction" course recommended.
- *Course overview*
  - Model design and simulation introduction
    - Notion of model: creation, blocks assembly, simulation parameters.
    - Start the simulation and display the results.
  - Discrete vs continuous simulations
    - Notion of solvers.
    - Discrete and continuous blocks.
    - Linear equations, differential equations: 1st and 2nd orders.
  - Advanced design
    - Blocks customization, sub-systems, analysis elements.
  - Exercises may be adapted to the customer needs.

Simulink Introduction

- **Title: Matlab/Simulink API**

- *Description*
  - Objective: Maximize the Matlab/Simulink experience by connecting with external programs.
  - Audience: Anyone using Matlab/Simulink.
  - Duration: 1 day.
  - Prerequisites: Advanced knowledge of Matlab/Simulink. Programming skills required.
- *Course overview*
  - Matlab engine API
    - Engine control from external C/C++ program.
  - Matlab COM Interface
    - Calling Matlab functions from external C/C++ program.
    - Data management.
  - MEX files API and Simulink S-Functions
    - Calling C/C++ programs from Simulink.
    - I/Os management.
  - MAT files API
    - Read and write MAT files from external C/C++ programs.
  - Exercises may be adapted to the customer needs.

Matlab/Simulink API

- **Title: Tests Management - TestLink**
- *Description*
  - Objective: Understand what is testing, why tests are important, and how to manage them. Work with TestLink, a testing software solution.
  - Audience: Anyone starting with tests management.
  - Duration: 8 hours.
  - Prerequisites: None.
- *Course overview*
  - Introduction to testing
    - Tests through the system life cycle
    - Testing categories, types and principles
  - Testing life cycle
    - Requirement analysis, test planning and test case development
    - Environment setup, test execution and test cycle closure
  - Build a test case
    - Composition of a test case
    - Testing techniques
  - Software solution
    - Introduction to TestLink
    - Hands On with TestLink

Tests Management