Requirements Determination (code RD)

Course Contents

Part I: Skills and Techniques

Interviewing Techniques
- The Inquiry-based Model
- Storyboards
- Other sources for Requirements Capture

The Inquiry-based Model in Detail
- Introduction to the inquiry-based model
- Starting an interview; the body of an interview
- Closing an interview
- Information capturing

Viewpoints and Stakeholders
- Kinds of stakeholders
- Stakeholders as sources of requirements
- Viewpoint-oriented requirements determination
- Viewpoints and business concerns
- Mapping viewpoints to requirements

Requirements
- Functional requirements
- Non-functional requirements
- Usability requirements
- Goal modelling

Requirements and Use Cases
- Why a use case is not a requirement
- Stakeholders and UML Actors
- Functional and non-functional requirements

Service Oriented Architectures
- What is a service?
- *Provides* and *requires* services
- Discovering requirements from services
- Documenting services

Part II: Requirements Elicitation

The Customer Requirements Document
- Initial features and requirements
- Interviewing key stakeholders
- Is the project feasible?
- Creating a Proof-Of-Concept (POC) system

Indentifying Stakeholders
- Preparing for interviews
- Applying the Inquiry-based model
- Hidden and real goals
- Other sources of requirements

Risk-Reduction during Elicitation
- The major business concerns
- Business processes
- Creating a context diagram
- Major functional (FRs) and non-functional requirements (NFRs)

Part III: Requirements Analysis and Negotiation

Requirements Analysis
- Sharpening our understanding of the initial requirements
- Requirements' priorities and risks
- Checklist questions and requirements classification
- Inter-requirement correlation (negative, positive)

Requirements Negotiation
- Customer Importance Levels
- Conflicts and Overlaps
- Describing requirements quantitatively
- Classification of requirements for IT stakeholders

System Structure and Decomposition
- Defining system boundaries; updated context diagram
- Removing superfluous requirements
- The Single Responsibility Principle (SRP) and unique requirements
- Aligning behaviour and structure

Domain Architectures
- What is a DA?
- The 5 categories: MAN, RAT, MIS, PCS, ACS
- Behavioural and structural aspects of DA
- The role of DAs in requirements engineering
- DAs and *analogical reasoning*

Scenarios: DAs to Requirements Determination

- Finding the DA corresponding to my system
- What are the important requirements in a DA?
- Finding the important stakeholders in a DA

Part IV: Requirements Description and Documentation

General Rules

- Standard templates
- Use of diagrams (UML class and component diagrams)
- Creating the requirements document; structure
- Requirements validation

Documentation using Services

- Determining *'provides'* and *'requires'* services
- Services and requirements
- Aligning services and systems

Requirements Management

- What is requirements management?
- Defining management policies
- Change management policies
- Management and requirements

Your Trainer

Daniel J. Duffy started the company Datasim in 1987 to promote C++ as a new object-oriented language for developing applications in the roles of developer, architect and requirements analyst to help clients design and analyse software systems for Computer Aided Design (CAD), process control and hardware-software systems, logistics, holography (optical technology) and computational finance. He used a combination of top-down functional decomposition and bottom-up object-oriented programming techniques to create stable and extendible applications (for a discussion, see Duffy 2004 where we have grouped applications into domain categories). Previous to Datasim he worked on engineering applications in oil and gas and semiconductor industries using a range of numerical methods (for example, the finite element method (FEM)) on mainframe and mini-computers.

Daniel Duffy has BA (Mod), MSc and PhD degrees in pure and applied mathematics and has been active in promoting partial differential equation (PDE) and finite difference methods (FDM) for applications in computational finance. He was responsible for the introduction of the Fractional Step (Soviet Splitting) method and the Alternating Direction Explicit (ADE) method in computational finance. He is also the originator of the exponential fitting method for time-dependent partial differential equations.

He is also the originator of two very popular C++ online courses (both C++98 and C++11/14) on www.quantnet.com in cooperation with Quantnet LLC and Baruch College (CUNY), NYC. He also trains developers and designers around the world. He can be contacted dduffy@datasim.nl for queries, information and course venues, in-company course and course dates