SANDER HOOGENDOORN

Designing, developing and deploying a Microservices Architecture

JUNE 7, 2018

Introduction to Agile, Scrum, XP, Kanban and Continuous Delivery in practice

JUNE 8, 2018

RESIDENZA DI RIPETTA - VIA DI RIPETTA, 231
ROME (ITALY)

info @ technologytransfer.it
www.technologytransfer.it

ABOUT THIS SEMINAR

The development and maintenance of monoliths presents organizations with increasing challenges, resulting in high costs and a slow time-to-market. More and more organizations are therefore attempting to componentize their applications. The latest and greatest paradigm Microservices finally seems to deliver on the promises of Service Oriented Architecture: shortening time-to-market, scalability, autonomy, and exchangeability of technology and databases. The challenges of delivering Microservices however are equally big.

What makes a component a Microservice? How to design, develop and deploy these small services? How does collaboration between analysts, developers, testers and operations change the organizations?

During this one day course, from his years of experience in Agile, Scrum, requirements, architecture and code Sander Hoogendoorn shares the voyage that implementing a Microservices landscape is. He will answer questions on modeling and designing Microservices, the granularity of applications and services, the communication between services, design patterns, polyglot persistence, testing services and setting up deployment pipelines. Richly illustrated with real-life examples, this course gives a perfect introduction into this promising technology.

BENEFITS OF ATTENDING

- Get a clear understand of the strengths and weaknesses of using Microservices
- Learn to understand which problems Microservices can solve and which not
- Help you decide whether Microservices will solve your current problems
- Learn how your software architecture will evolve when you move towards Microservices and how to keep your architecture flexible
- Learn how to design and model applications, Microservices and resources in this new architecture
- Get understanding of the more explicit role of testing in a Microservices architecture, and which technique apply well
- Learn about continuous integration and continuous delivery, and how you design your deployment pipelines

WHO SHOULD ATTEND

- Managers who have clear issues with current applications, such as poor maintainability, long time-to-market, moving away from legacy code, and spaghetti architecture, and are considering Microservices
- Managers who want to understand the challenges and opportunities Microservices architectures
- Architects who want to define software architecture for successful Microservices implementations
- IT Professionals who want to understand who Microservices will influence the existing infrastructure architecture
- Business and functional Analysts who want to obtain an overview of the different design techniques used in Microservices, or who want to understand who Microservices will change their jobs
- Developers and Testers who require an overview of all new technology, frameworks and tools that are used in Microservices
- IT Managers and IT Strategists selecting new technologies
- IT Architects and Managers who need to develop an integration strategy for their company
- Consultants who need to recommend different strategies for implementing integration scenario's

OUTLINE

1. An introduction to components and services

• Introduction
• Monoliths versus Microservices
• Strengths and weaknesses of monolithic software
• A brief history of components and services

2. Introducing Microservices

• Are we beyond the hype yet?
• A definition of Microservices
• Characteristics of Microservices
• Containers and scalability
• Polyglot persistence
• Promises of Microservices
• Challenges in Microservices
• How big or small are Microservices?

3. Greenfield or brownfield?

• Presenting two real world cases
• Guiding principles from both cases
• A business process first approach
• Different levels of business processes
• An architecture first approach
• How to split up your existing code base?
• Brownfield migration to Microservices

4. Evolutionary software architecture

• Where to start?
• Introducing design patterns
• Applications, workers and services
• Service consuming applications
• Service delivering components
• Dealing with communication and REST
• Authentication and tokens

5. Designing and building Microservices

• Why modular design is key
• Guidelines for design Microservices
• The Single Responsibility Principle (SRP)
• Introducing domain driven design
• Explaining bounded contexts

• Modeling micro-applications
• Wireframes
• Smart use cases
• Mapping bounded contexts around resources
• Working with HTTP verbs
• Creating a RESTful API for you services
• Explaining Postel's law
• Introducing the resource model

6. Testing Microservices

• An overview of test techniques for Microservices
• Why manual testing isn't efficient
• Unit testing
• Behavior driven design with examples
• Introducing service contracts
• QA (with SonarQube)
• Integration testing
• Acceptance testing
• When to test what in your deployment pipelines

7. Deployment of Microservices

• Continuous integration
• Designing your deployment pipelines
• Moving towards continuous delivery
• Minimal viable products (MVP)
• Agile, Kanban and Microservices
• Microservices and DevOps
• Do Microservices change your organization?

8. Concluding

• Some final recommendations
• Do Microservices solve all challenges your IT department has?
• How to proceed?

ABOUT THIS SEMINAR

Since the introduction of the waterfall model in 1970 as an approach for software development projects, a lot has evolved. In the 1990's similar ideas around iterative and Agile software development where introduced simultaneously in more and more organizations and projects, resulting in The Agile Manifesto.

During this one day workshop, Sander Hoogendoorn will demonstrate how Agile projects operate, independent of any specific approach.
Next, from his twenty years of experience in Agile and iterative projects the speaker compares the principles, best practices, roles and techniques from well known approaches such as Scrum, Extreme Programming, Smart and DSDM. He will explain techniques such as stand-up meetings, retrospectives, pair programming and test driven design. He will also address how agile projects can evolve into even more effective flow-based approaches, such as Kanban, and Continuous Delivery. He touches on many of the strengths and weaknesses of each of these approaches, and will illustrate this with many, many examples from his own projects.
Moreover, he will use eye-opening hands-on exercises so participants will be able to experience the difference in effectiveness themselves real-time.

BENEFITS OF ATTENDING

• Get a clear understand of the strengths and weaknesses of the different approaches for software development
• Learn to understand how Agile works, independent of any particular approach
• Help you understand how Scrum and Extreme Programming work in practice
• Learn how your project will benefit from applying Agile best practices and techniques, such as stand-up meetings, retrospectives, pair programming and multi-disciplinary teams
• See real-life examples of using several types of Agile dashboards
• Learn how you might evolve from Agile approaches to flow-based approaches such as Kanban
• Learn about continuous integration and Continuous Delivery, and how that effects your projects

WHO SHOULD ATTEND

• Managers who have clear issues with their current approaches for software development projects, such as poor maintainability, long time-to-market, and are considering moving towards Agile, Scrum or Kanban
• Managers who want to understand the challenges and opportunities Agile approaches will offer
• Project Managers who would like to find out how their role changes completely in Agile and flow-based approaches
• IT Professionals who want to understand how Agile will influence the the ways they collaborate in teams, and are investigating DevOps
• Business Analysts, functional Analysts, Developers and for sure Testers who want to understand if and how their design techniques fit in Agile, and how their roles change
• IT Managers and IT Strategists selecting new ways of working for their organizations and projects
• IT Architects and Managers who need to develop an integration strategy for their company
• Consultants who need to recommend different strategies for implementing Agile approaches

OUTLINE

1. Why waterfall won't work

- Introducing the waterfall model
- Why waterfall won't work
- Explaining Boehm's Law
- What waterfall should have been
- Next steps in software development approaches
- The Unified Process and the Spiral Model

2. What does being Agile really mean?

- The Agile Manifesto
- What makes a project Agile?
- Introducing the product backlog
- Working in iterations
- Planning you iterations
- Retrospectives
- Dealing with new and changing requirements
- Boehm's Law applied to Agile

3. An introduction to Scrum

- Explaining Scrum ceremonies
- Scrum roles
- The development team
- What does the product owner do?
- What does a Scrum Master do?

4. Extreme Programming and Agile practices

- Extreme Programming in a nutshell
- Agile best practices
- Collaborating with the customer
- Stand-up meetings
- Distributed stand-up meetings
- Burn-down and burn-up charts
- Pair programming and side-by-side programming
- The Definition of Done

5. Using Agile dashboards

- The Scrum task board
- Stand-ups and dashboards
- Visualizing you flow
- Digital Agile dashboards
- Speedbird9, Mingle, Jira Agile

6. Agile beyond Agile: Kanban

- The red sprint anti-pattern
- Boehm's Law applied to flow
- Introducing Kanban
- Kanban boards
- Optimizing your flow
- Applying the Theory of Constraints
- Limiting your work-in-progress?

7. Moving to Continuous Delivery

- Small increments or big releases?
- Roadmap over planning
- Minimal viable products
- Continuous Delivery
- Continuous Deployment
- Those who build it, run it: DevOps

8. Retrospective

- Some final thoughts

9. Good and bad examples

Of course, all topics come with good and bad examples from Sander Hoogendoorn' fifteen years of experience in coaching and running Agile projects. He has presented this workshop over fifty times already, both in-house and through various independent organizers.

INFORMATION

PARTICIPATION FEE

1) Designing, developing and deploying a Microservices Architecture
€ 700

2) Introduction to Agile, Scrum, XP, Kanban and Continuous Delivery in practice
€ 700

The fee includes all seminar documentation, luncheon and coffee breaks.

VENUE

Roma, Residenza di Ripetta
Via di Ripetta, 231
Rome (Italy)

SEMINAR TIMETABLE

9.30 am - 1.00 pm
2.00 pm - 5.00 pm

HOW TO REGISTER

You must send the registration form with the receipt of the payment to:
TECHNOLOGY TRANSFER S.r.l.
Piazza Cavour, 3 - 00193 Rome (Italy)
Fax +39-06-6871102

within May 23, 2017

PAYMENT

Wire transfer to:
Technology Transfer S.r.l.
Banca: Cariparma
Agenzia 1 di Roma
IBAN Code:
IT 03 W 06230 03202 000057031348
BIC/SWIFT: CRPPIT2P546

GENERAL CONDITIONS

DISCOUNT

The participants who will register 30 days before the seminar are entitled to a 5% discount.

If a company registers 5 participants to the same seminar, it will pay only for 4.

Those who benefit of this discount are not entitled to other discounts for the same seminar.

CANCELLATION POLICY

A full refund is given for any cancellation received more than 15 days before the seminar starts. Cancellations less than 15 days prior the event are liable for 50% of the fee. Cancellations less than one week prior to the event date will be liable for the full fee.

CANCELLATION LIABILITY

In the case of cancellation of an event for any reason, Technology Transfer's liability is limited to the return of the registration fee only.

SANDER HOOGENDOORN

1 DESIGNING, DEVELOPING, AND DEPLOYING A MICROSERVICES ARCHITECTURE

Rome June 7, 2018
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 700

2 INTRODUCTION TO AGILE, SCRUM, XP, KANBAN AND CONTINUOUS DELIVERY IN PRACTICE

Rome June 8, 2018
Residenza di Ripetta - Via di Ripetta, 231
Registration fee: € 700

If anyone registered is unable to attend, or in case of cancellation of the seminar, the general conditions mentioned before are applicable.

first name ....

surname ....

job title ....

organisation ....

address ....

postcode ....

city ....

country ....

telephone ....

fax ....

e-mail ....

— *Stamp and signature* —

Send your registration form
with the receipt of the payment to:
Technology Transfer S.r.l.
Piazza Cavour, 3 - 00193 Rome (Italy)
Tel. +39-06-6832227 - Fax +39-06-6871102
info@technologytransfer.it
www.technologytransfer.it

Speaker

Sander Hoogendoorn is an independent dad, mentor, trainer, software architect, programmer, speaker, and writer. He is a highly appreciated catalyst in the innovation of software development at his many international clients. Well known as the author of the best-selling book **This is Agile**, Mr. Hoogendoorn coaches organizations, projects and teams, has written books on UML and Agile, and published over 250 articles in international magazines. He is an inspiring (keynote) speaker at many international conferences, and presents seminars and training courses around the world on a variety of topics such as (beyond) Agile, Scrum, Kanban, Continuous Delivery, software estimation, software architecture, Microservices, design patterns, modeling and UML, writing code, and testing.