

1. Record Nr.	UNINA9911006678303321
Autore	Douglass Bruce Powel
Titolo	Agile Model-Based Systems Engineering Cookbook : Improve System Development by Applying Proven Recipes for Effective Agile Systems Engineering
Pubbl/distr/stampa	Birmingham : , : Packt Publishing, Limited, , 2021 ©2021
ISBN	9781523151424 1523151420 9781803234304 180323430X
Edizione	[2nd ed.]
Descrizione fisica	1 online resource (601 pages)
Collana	Expert insight
Altri autori (Persone)	HolstChristian von
Disciplina	620/.001171
Soggetti	Systems engineering Agile software development
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Cover -- Contributors -- Table of contents -- Preface -- Chapter 1: Basics of Agile Systems Modeling -- What's agile all about? -- Model-Based Systems Engineering (MBSE) -- Managing your backlog -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Measuring your success -- How to do it -- Example -- Some considerations -- Managing risk -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Product roadmap -- Purpose -- Inputs and preconditions -- How to do it -- Example -- Release plan -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Iteration plan -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Estimating Effort -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- How it works -- Example -- Work item prioritization -- Purpose -- Inputs and preconditions -- How to do it -- How it works -- Example -- Iteration 0 -- Purpose -- Inputs and preconditions -- Outputs and

postconditions -- How to do it -- Example -- Architecture 0 -- Subsystem and component view -- Concurrency and resource view -- Distribution view -- Dependability view -- Deployment view -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Additional note -- Organizing your models -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- How it works -- Example -- Managing change -- Purpose -- Inputs and preconditions -- How to do it -- Example -- Chapter 2: System Specification -- Recipes in this chapter -- Why aren't textual requirements enough? -- Definitions -- Functional Analysis with Scenarios -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it. Example -- Functional analysis with activities -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Functional analysis with state machines -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Functional Analysis with User Stories -- A little bit about user stories -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Model-Based Safety Analysis -- A little bit about safety analysis -- Some Profiles -- Hazard analysis -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Model-Based Threat Analysis -- Basics of Cyber-Physical Security -- Modeling for Security Analysis -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Specifying Logical System Interfaces -- A Note about SysML Ports and Interfaces -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Creating the Logical Data Schema -- Definitions -- Example -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Chapter 3: Developing System Architectures -- Recipes in this chapter -- Five critical views of architecture -- General architectural guidelines -- Architectural trade studies -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Architectural merge -- Example -- Pattern-driven architecture -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Subsystem and component architecture -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Architectural allocation -- Creating subsystem interfaces from use case scenarios -- Purpose -- Inputs and preconditions. Outputs and postconditions -- How to do it -- Specializing a reference architecture -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Chapter 4: Handoff to Downstream Engineering -- Recipes in this chapter -- Activities for the handoff to downstream engineering -- Starting point for the examples -- Preparation for Handoff -- Federating Models for Handoff -- Logical to Physical Interfaces -- Deployment Architecture I: Allocation to Engineering Facets -- Deployment Architecture II: Interdisciplinary Interfaces -- Chapter 5: Demonstration of Meeting Needs: Verification and Validation -- Recipes in this chapter -- Verification and validation -- Model simulation -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Model-based testing -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Computable constraint modeling -- Purpose -- Inputs and preconditions -- How to do it -- Example -- Traceability -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Effective

Reviews and walkthroughs -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Managing Model Work Items -- Purpose -- Inputs and preconditions -- How to do it -- Example -- Test Driven Modeling -- Purpose -- Inputs and preconditions -- Outputs and postconditions -- How to do it -- Example -- Appendix A: The Pegasus Bike Trainer -- Overview -- Pegasus High-Level Features -- Highly customizable bike fit -- Monitor exercise metrics -- Export/upload exercise metrics -- Variable power output -- Gearing emulation -- Controllable power level -- Incline control -- User interface -- Online training system compatible -- Configuration and OTA firmware updates -- Packt page -- Other Books You May Enjoy.

Index.

Sommario/riassunto

Agile MBSE can help organizations manage change while ensuring system correctness and meeting customers' needs. But deployment challenges have changed since our first edition. The Agile Model-Based Systems Engineering Cookbook's second edition focuses on workflows – or recipes – that will help MBSE practitioners and team leaders address practical situations that are part of deploying MBSE as part of an agile development process across the enterprise. In this 2nd edition, the Cameo MagicDraw Systems Modeler tool – the most popular tool for MBSE – is used in examples (models are downloadable by readers). Written by a world-renowned expert in MBSE, this book will take you through systems engineering workflows in the Cameo Systems Modeler SysML modeling tool and show you how they can be used with an agile and model-based approach. You'll start with the key concepts of agile methods for systems engineering. Next, each recipe will take you through initiating a project, outlining stakeholder needs, defining and analyzing system requirements, specifying system architecture, performing model-based engineering trade studies, all the way to handling systems specifications off to downstream engineering. By the end of this MBSE book, you'll learn how to implement systems engineering workflows and create systems engineering models.
