1. Record Nr. UNINA9910583401203321

Autore	Boulanger Jean-Louis
Titolo	Certifiable software applications . 3 Downward cycle / / Jean-Louis Boulanger
Pubbl/distr/stampa	London, England : , : ISTE Press Kidlington, England : , : Elsevier, , 2018
ISBN	0-08-101163-6 1-78548-119-3
Descrizione fisica	1 online resource (358 pages)
Disciplina	005.1
Soggetti	Application software - Development Software engineering
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references and index
Nota di contenuto	Intro; Title page; Table of Contents; Dedication; Copyright; Introduction; Reader's guide; Acknowledgments; 1: Realization of a Software Application; Abstract; 1.1 Introduction; 1.2 Implementation process; 1.3 Quality control; 1.4 Conclusion; 2: Requirements Management; Abstract; 2.1 Introduction; 2.2 Three key concepts; 2.3 Process; 2.4 Conclusion; 3: Documentation Structure; Abstract; 3.1 Introduction; 3.2 Documentation; 3.3 Quality control; 3.4 Implementation process; 3.5 Configuration management; 3.6 Defect and non-conformity management; 3.7 Conclusion 4: Requirements Specification of a Software ApplicationAbstract; 4.1 Introduction; 4.2 Requirements management; 4.3 Specification know-how; 4.4 Quality characteristics of a software product; 4.5 Objectives of the software requirements specification phase; 4.6 Risk assessment; 4.7 System requirements specification document: SwRS; 4.8 Specifications formalization; 4.9 Considering COTS and reused components; 4.10 Specification verification; 4.11 Conclusion; 4.12 Appendix: SwRS template; 5: Modeling; Abstract; 5.1 Introduction; 5.2 Modeling; 5.3 Model transformation; 5.4 Model refinement 5.5 Methods5.6 Conclusion; 6: Formalization; Abstract; 6.1 Introduction; 6.2 Approach example; 6.3 Model verification; 6.4 Setting up of formal methods; 6.5 Implementation of formal methods; 6.6

Software application maintenance; 6.7 Conclusion; 7: Software Specification Verification Stage; Abstract; 7.1 Introduction; 7.2 Verification; 7.3 Conclusion; 8: Component Versus Module; Abstract; 8.1 Introduction; 8.2 Reusability, maintainability and continuity of service; 8.3 Module and component; 8.4 Conclusion; 9: Software Application Architecture; Abstract; 9.1 Introduction
9.2 Objective of the architecture phase of a software application9.3 Software architect; 9.4 Software architecture description document; 9.5 Verification of the software application architecture; 9.6 Consideration of COTS and reused components; 9.7 Model approach; 9.8 Conclusion; 10: Software Application Architecture Verification; Abstract; 10.1 Introduction; 10.2 Verification; 10.3 Conclusion; 11: Software Application Design; Abstract; 11.1 Introduction; 11.2 Component; 11.3 Purpose of the design phase; 11.4 Designer; 11.5 Software component specification document
11.6 Software component specification document11.7 Design verification; 11.8 Consideration of COTS and reused components; 11.9 Conclusion; 12: Software Application Component Design Phase Verification; Abstract; 12.1 Introduction; 12.2 Verification; 12.3 Conclusion; 13: Software Application Coding; Abstract; 13.1 Introduction; 13.2 Coding phase objective; 13.3 IMPlementer; 13.4 Code production; 13.5 Principles to be implemented in the context of the coding phase; 13.6 The choice of design language; 13.7 Consideration of COTS and reused components; 13.8 Coding phase verification