1. Record Nr. UNINA9910784543603321

Autore	Arnowitz Jonathan
Titolo	Effective prototyping for software makers [[electronic resource] /] / Jonathan Arnowitz, Michael Arent, Nevin Berger
Pubbl/distr/stampa	San Francisco, Calif., : Morgan Kaufmann London, : Elsevier Science [distributor], c2007
ISBN	1-280-75139-8 9786610751396 0-08-046896-9
Edizione	[1st edition]
Descrizione fisica	1 online resource (625 p.)
Collana	The Morgan Kaufmann series in interactive technologies
Altri autori (Persone)	ArentMichael BergerNevin
Disciplina	005.1
Soggetti	Computer software - Development Computers
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Front Cover; Effective Prototyping for Software Makers; Copyright Page; Dedications; Table of Contents; Acknowledgments; Preface; CHAPTER 1 WHY PROTOTYPING?; What Is a Prototype?; An Historical Perspective of Prototyping; Leonardo da Vinci: The Thinking Man's Inventor; Thomas Alva Edison: Inventor Prototyper; Henry Dreyfuss: Designer Prototyper; The Purpose of Prototyping Software; Will the Design Work Properly?; Can the Design Be Produced Economically?; How Will Users and Other Stakeholders Respond to the Design?; Which Approach Can Be Taken to Get From Concept to Product? How Can Prototyping Support Product Design Specification?How Can Prototyping Contribute to Better Product Scheduling and Budget Planning?; Summary; References; CHAPTER 2 THE EFFECTIVE PROTOTYPING PROCESS; Phase I: Plan (Chapters 3-5); Step 1: Verify the Requirements (Chapter 3); Step 2: Create a Task/Screen Flow (Chapter 4); Step 3: Specifying Content and Fidelity (Chapter 5); Phase II: Specification (Chapters 6-8); Step 4: Determine the Right Prototyping Characteristics (Chapter 6); Step 5: Choose a Prototyping Method (Chapter 7); Step 6: Choose a Prototyping Tool (Chapter 8)

Phase III: Design (Chapters 9 and 10)Step 7: Formulate Design Criteria (Chapter 9); Step 8: Create the Prototype (Chapter 10); Phase IV: Results (Chapters 11-13); Step 9: Review the Prototype (Chapter 11); Step 10: Validate the Design (Chapter 12); Step 11: Implement the Design (Chapter 13); Summary; PHASE I PLAN YOUR PROTOTYPE; CHAPTER 3 VERIFY PROTOTYPE ASSUMPTIONS AND REQUIREMENTS; Prototyping Requirements Are Not Software Requirements; Transformation of Assumptions to Requirements; Step 1: Gather Requirements; Step 2: Inventorize the Requirements
Step 3: Prioritize Requirements and AssumptionsRequirements and the Big Picture; Iteration 1: From Idea to First Visualization; Iteration 2: From Quick Wireframe to Wireframe; Iteration 3: From Wireframe to Storyboard; Iteration 4: From Storyboard to Paper Prototype; Iteration 5: From Paper Prototype to Coded Prototype; Iteration 6: From Coded Prototype to Software Requirements; Summary; References; CHAPTER 4 DEVELOP TASK FLOWS AND SCENARIOS; Task Flow; Task Layer Maps; Step 1: Create List of Tasks; Step 2: Identify Dependencies; Step 3: Layer Task Items; Step 4: Remove Redundant Dependencies Dependency DiagramStep 1: Prioritize Requirements; Step 2: Highlight Key Tasks; Step 3: Identify Needs; Swim Lane Diagrams; Step 1: Identify User Tasks; Step 2: Identify User Roles; Step 3: Layout User Roles and Task Flows; Step 4: Identify and Visualize Interrelationships; Usage Scenarios; Step 1: Sketch Out Plot; Step 2: Choose Cast; Step 3: Outline Plot; Step 4: Mark Points in Outline; Summary; References; CHAPTER 5 DEFINE PROTOTYPE CONTENT AND FIDELITY; Prototype Fidelity; Low Fidelity; High Fidelity; Prototype Content; Information Design Techniques to Adjust the Fidelity of Information Design

Sommario/riassunto

Much as we hate to admit it, most prototyping practice lacks a sophisticated understanding of the broad concepts of prototyping-and its strategic position within the development process. Often we overwhelm with a high fidelity prototype that designs us into a corner. Or, we can underwhelm with a prototype with too much ambiguity and flexibility to be of much use in the software development process. This book will help software makers-developers, designers, and architects-build effective prototypes every time: prototypes that convey enough information about the product at the appropriat