

# DevOps Engineering

- Engineering DevOps: From Chaos to Continuous Improvement and Beyond, Marc Hornbeek
- Learning DevOps: The Complete Guide to Accelerate Collaboration with Jenkins, Kubernetes, Terraform and Azure DevOps, Mikael Krief
- DevOps for Developers: Integrate Development and Operations, The Agile Way, Michael Huttermann

	<b>Engineering DevOps</b>
	<b>Part-01: What is Engineering DevOps, and Why is It Important?</b>
	<b>What is Engineering DevOps?</b>
01	Introduction
	DevOps Engineering Blueprint
	DevOps Engineering Tenets and CALMS
	Origins of DevOps form an Engineering Point of View
	The Dilemma of Defining Engineering DevOps
	DevOps Engineering Terms
	<b>Nine Pillars of Engineering DevOps</b>
	Introduction
	Leadership Pillar
	Collaborative Culture Pillar
	Design for DevOps Pillar
	Continuous Integration (CI) Pillar
	Continuous Testing (CT) Pillar
	Elastic Infrastructure (EI) Pillar
	Continuous Monitoring (CM) Pillar
	Continuous Security Pillar
	Continuous Delivery (CD) Pillar
	<b>Why is Engineering DevOps Important?&gt;</b>
	Introduction
	Engineering DevOps Myths and Realities
	How Will I Know When I Have Engineered DevOps?
	Benefits of Well-Engineered DevOps – ½
	Benefits of Well-Engineered DevOps – ½
	Cost of Not Engineering DevOps Properly
	<b>Part-02: Engineering People, Process, and Technology for DevOps</b>
	<b>How Should People, Process, and Technology be Engineered for DevOps?</b>
	Does DevOps Engineering Require People to be Engineers?
	DevOps People, Process, and Technology Engineering Maturity Levels
	Three Dimensions of Engineering DevOps – People, Process, and Technology – People
	Three Dimensions of Engineering DevOps – People, Process, and Technology – Process
	Three Dimensions of Engineering DevOps – People, Process, and Technology – Technology
	Twenty-Seven DevOps Engineering Critical Success Factors
	Learn DevOps Value-Stream Pipeline Engineering – 1/3
	Learn DevOps Value-Stream Pipeline Engineering – 1/ 3
	Learn DevOps Value-Stream Pipeline Engineering – 1/3
	<b>Value-Stream Management (VSM)</b>
	Why Is Value-Stream Management Important to DevOps?
	How Does Value-Stream Management Work with DevOps? – ½
	How Does Value-Stream Management Work with DevOps? – ½

	What is Needed to Engineer a Value-Stream Management Solution for DevOps?
	<b>Application Release Automation (ARA)</b>
	Why is Application Release Automation Important?
	How Does Application Release Automation Work?
	What is Needed to Implement Well-Engineered ARA?
	<b>Version Management – 1/3</b>
	<b>Version Management – 1/3</b>
	<b>Version Management – 1/3</b>
	<b>Continuous Security (a.k.a. DevSecOps)</b>
	Why Is Continuous Security Important to Engineering DevOps?
	How Does Continuous Security Work with DevOps Engineering? – 1/3
	How Does Continuous Security Work with DevOps Engineering? – 1/3
	How Does Continuous Security Work with DevOps Engineering? – 1/3
	Implementing Continuous Security
	<b>Service Catalogs Facilitate DevOps Engineering</b>
	Why Is the Service Catalog Important to DevOps Engineering?
	How are DevOps Service Catalogs Engineered for DevOps?
	What is Needed to Engineer a DevOps Service Catalog?
	<b>DevOps Governance Engineering</b>
	Why Is Governance Engineering Important for DevOps Engineering?
	How Is Governance Engineered for DevOps?
	What is Needed to Engineer Governance for DevOps?
	<b>Site Reliability Engineering (SRE)</b>
	Why is SRE Important to DevOps Engineering?
	How Does SRE Work with DevOps?
	What is Needed to Engineer SRE with DevOps?
	<b>DevOps Disaster Mitigation and Recovery</b>
	<b>Part-03: Engineering Applications, Pipelines, and Infrastructures for DevOps</b>
	<b>DevOps Application Engineering</b>
	Application Design for DevOps
	Applications for Which DevOps Does Not Apply
	DevOps Applied to Enterprise Apps
	DevOps Applied to COST Systems
	DevOps Applied to Manufactured Software Embedded Products
	DevOps Applied to Software Services
	Five Levels of Application Maturity
	<b>CI/CD Pipeline Engineering Indicated in the Last Chapter...</b>
	CI/CD Tools
	CI/CD Toolchains to Realize Continuous Flow...
	CI/CD with Multitier Applications...
	CI/CD for Databases
	CI/CD for Micro-services Pipelines Micro-Services Architectures...
	The Complexity of Many Parallel Micro-services DevOps Pipelines
	CI/CD Pipelines in the Clouds Considering the Global Movement of IT Infrastructures...
	Five Levels of CI/CD Pipeline Maturity...
	<b>DevOps Elastic Infrastructures</b>
	Ephemeral Elastic Infrastructures DevOps Works Best...
	Idempotency and Immutable Infrastructure...
	Bare Metal, Virtual Machines, Containers, and Server-less The Workhorse of IT...
	Infrastructure as Code (IAC)...
	Infrastructure-as-Code Tools...

	Net DevOps
	Ad-Hoc Infrastructures
	Private Data Centers...
	Cloud Cost Management Recommended Engineering Practices
	Cloud System Performance Recommended Engineering Practices
	High-Availability Cloud Services Recommended Engineering Practices
	Cloud Change Management Recommended Engineering Practices
	Cloud Security and Compliance Recommended Engineering Practices
	DevOps Hybrid Cloud
	DevOps Hybrid Cloud Orchestration Tools
	DevOps Multi-Cloud
	DevOps Multi-Cloud Services...
	Five Levels of Infrastructure Maturity...
	<b>Continuous Test Engineering</b>
	Why Is Continuous Test Engineering Important to DevOps
	How Is Continuous Testing Engineered for DevOps
	Dev Stage Continuous Test Assessment Examples
	Integration Stage Continuous Test Assessment Examples
	Pre-Prod Stage Continuous Test Assessment Examples
	Post-Prod Stage Continuous Testing Assessment Examples
	Advanced Continuous Test Engineering
	<b>Continuous Monitoring Engineering</b>
	Why Is Continuous Monitoring Engineering Important to DevOps?
	How Continuous Monitoring Is Engineered for DevOps?
	Engineering Continuous Monitoring for Applications
	Engineering Continuous Monitoring for Pipelines
	Engineering Continuous Monitoring for Infrastructures
	<b>Continuous Delivery and Deployment Engineering</b>
	Why Is Continuous Delivery and Deployment Important to Engineering DevOps?
	How Is Continuous Delivery and Deployment Engineered?
	Blue-Green Deployments...
	Dark Launching Release Strategy...
	Feature Toggles and A/B Test Strategy...
	Feature Flag Roll-Out Deployments...
	Canary Deployments
	Deployment Database Changes...
	Micro-services Deployments Best Practice...
	Deploying Containers with Kubernetes...
	<b>Part-4: Transformation Engineering Blueprint</b>
	<b>DevOps Seven-Step Transformation Engineering Blueprint</b>
	<b>Step One: Visioning</b>
	Why is the Visioning Step Important to DevOps Engineering?
	How is the Visioning Step Accomplished?
	Overcoming Challenges with the Visioning Step...
	<b>Step Two: Alignment</b>
	Why is the Alignment Step Important to Engineering DevOps?
	How is the Alignment Step Accomplished?
	Overcoming Challenges with the Alignment Step
	<b>Step Three: Assessment</b>
	Why Is the Assessment Step Important to Engineering DevOps
	How is the Assessment Step Accomplished?

	Discover Current State
	Assess the Maturity of DevOps Practices
	Create a Current State Value-Stream Map
	Align Priorities for Solution Requirements
	Overcoming Challenges with the Assessment Step
	<b>Step Four: Solution</b>
	Why is the Solution Step Important to Engineering DevOps?
	How Is the Solution Step Accomplished?
	Create a Future State Value-Stream Map...
	Road-Mapping DevOps Transformation...
	Estimate ROI Business Managers will Not Likely Approve a Solution Unless It Will Yield a ROI.
	Solution Recommendation Alignment...
	Overcoming Challenges with the Solution Step Estimates Required for Completing...
	<b>Step Five: Realize</b>
	Why is the Realize Step Important to Engineering DeOps?
	How is the Realize Step Accomplished?
	Task Level Planning
	Proof of Concept (POC) Trials
	Implementation
	Release to Production
	Training
	Governance
	Overcoming Challenges with the Realize Step
	<b>Step Six: Operationalize</b>
	Why is the Operationalize Step Important to Engineering DevOps?
	How is the Operationalize Step Accomplished?
	Controlled Access
	Monitoring
	Governance
	Support
	Evolution
	Overcoming Challenges with the Operationalize Step
	<b>Step Seven: Expansion</b>
	Why is the Expansion Step Important to Engineering DevOps?
	How is the Expansion Step Accomplished?
	DevOps Continuous Flow Model Evolutions...
	Mastering DevOps Maturity Level 4 – Second Way (Continuous Feedback)...
	Mastering DevOps Maturity Level 5 – Third Way (Continuous Improvement)...
	Beyond DevOps Maturity Level 5 – Continuous Autonomous Improvement...
	<b>Future of Engineering DevOps – Beyond Continuous Improvement...</b>
	<b>Continuous Learning</b>
	Learning Continuous Flow...
	Why is Learning Continuous Flow Important to Engineering DevOps?
	How is Learning Continuous Flow Accomplished?
	What is Needed to Implement Learning for Continuous Flow?
	Learning Continuous Feedback...
	Why is Learning Continuous Feedback Important to Engineering DevOps?
	How is Learning Continuous Feedback Accomplished?
	What is Needed to Implement Learning for Continuous Feedback?
	Learning Continuous Improvement...
	Why is Learning Continuous Feedback Important to Engineering DevOps?

	How is Learning Continuous Feedback Accomplished?
	<b>Learning DevOps</b>
	<b>Section-01: DevOps and Infrastructure as Code</b>
	<b>DevOps Culture and Practice</b>
01	Getting Started with DevOps
	Implementing CI/CD and Continuous Deployment – Topics 4
	Understanding IaC Practices – Topics 10
	Summary
	Questions
	Further Reading
	<b>Provisioning Cloud Infrastructure with Terraform</b>
	Technical Requirements
	Installing Terraform – Topics 6
	Configuring Terraform for Azure – Topics 3
	Writing a Terraform Script to Deploy Azure Infrastructure – Topics 4
	Deploying the Infrastructure with Terraform – Topics 3
	Terraform Command Lines and Life Cycle – Topics 5
	Protecting tfstate in a Remote Backend
	Summary
	Questions
	Further Reading
	<b>Using Ansible for Configuring IaaS Infrastructure</b>
	Technical Requirements
	Installing Ansible – Topics 4
	Creating an Inventory for Targeting Ansible Host – Topics 3
	Writing the First Playbook – Topics 3
	Executing Ansible – Topics 3
	Protecting Data with Ansible Vault – Topics 2
	Using a Dynamic Inventory for Azure Infrastructure
	Summary
	Questions
	Further Reading
	<b>Optimizing Infrastructure Deployment with Packer</b>
	Technical Requirements
	An Overview of Packer – Topics 8
	Creating Packer Templates for Azure VMs with Scripts – Topics 5
	Using Ansible in a Packer Template – Topics 2
	Executing Packer – Topics 3
	Using a Packer Image with Terraform
	Summary
	Questions
	Further Reading
	<b>Section-02: DevOps CI/CD Pipeline</b>
	<b>Managing Your Source Code with Git</b>
	Technical Requirements
	Overviewing Git and Its Command Lines – Topics 4
	Overviewing Git and Its Command Lines – Topics 4
	Overviewing Git and Its Command Lines – Topics 4
	Understanding the Git Process and GitFlow Pattern – Topics 4
	Understanding the Git Process and GitFlow Pattern – Topics 4
	Summary

	Questions
	Further Reading
	<b>Continuous Integration and Continuous Delivery</b>
	Technical Requirements
	The CI/CD Principles – Topics 2
	Using a Package Manger – Topics 3
	Using Jenkins – Topics 4
	Using Azure Pipelines – Topics 3
	Using GitLab CI – Topics 4
	Summary
	Questions
	Further Reading
	<b>Section-03: Containerized Applications with Docker and Kubernetes</b>
	<b>Containerizing Your Application with Docker</b>
	Technical Requirements
	Installing Docker – Topics 3
	Creating a Dockerfile – Topics 2
	Building and Running a Container on a Local Machine – Topics 3
	Pushing an Image to Docker Hub
	Deploying a Container to ACI with a CI/CD Pipeline – Topics 2
	Summary
	Questions
	Further Reading
	<b>Managing Containers Effectively with Kubernetes</b>
	Technical Requirements
	Installing Kubernetes – Topics 3
	First Example of Kubernetes Application Deployment
	Using HELM as a Package Manger
	Using AKS – Topics 3
	Creating a CI/CD Pipeline for Kubernetes with Azure Pipelines – Topics 2
	Summary
	Questions
	Further Reading
	<b>Section-04: Testing Your Application</b>
	<b>Testing APIs with Postman</b>
	Technical Requirements
	Creating a Postman Collection with Request – Topics 3
	Using Environments and Variable to Dynamize Requests
	Writing Postman Tests
	Executing Postman Request Tests Locally
	Understanding the Newman Concept
	Preparing Postman Collections for Newman – Topics 2
	Running the Newman Command Line
	Integration of Newman in the CI/CD Pipeline Process – 5
	Summary
	Questions
	Further Reading
	<b>Static Code Analysis with SonarQube</b>
	Technical Requirements
	Exploring SonarQube
	Installing SonarQube – Topics 5

	Real-time Analysis with SonarLint
	Executing SonarQube in Continuous Integration – Topics 2
	Summary
	Questions
	Further Reading
	<b>Security and Performance Tests</b>
	Technical Requirements
	Applying Web Security and Penetration Testing with ZAP – Topic 2
	Running Performance Tests with Postman
	Summary
	Questions
	Further Reading
	<b>Section-05: Taking DevOps Further</b>
	<b>Security in the DevOps Process with DevSecOps</b>
	Technical Requirements
	Testing Azure Infrastructure Compliance with Chef InSpec – 7
	Using the Secure DevOps Kit for Azure – Topics 3
	Preserving Data with HashiCorp’s Vault – Topics 6
	Summary
	Questions
	Further Reading
	<b>Reducing Deployment Downtime</b>
	Technical Requirements
	Reducing Deployment Downtime with Terraform
	Understanding Blue-green Deployment Concepts and Patterns – Topics 3
	Applying Blue-Green Deployments on Azure – Topics 2
	Introducing Feature Flags
	Using an Open Source Framework for Feature Flags
	Using the LaunchDarkly Solution
	Summary
	Questions
	Further Reading
	<b>DevOps for Open Source Projects</b>
	Technical Requirements
	Storing the Source Code in GitHub – Topics 2
	Contributing Using Pull Requests
	Managing the Changelog and Release Notes
	Sharing Binaries in GitHub Releases
	Using Travis CI for Continuous Integration
	Getting Started with GitHub Actions
	Analyzing Code with SonarCloud
	Detecting Security Vulnerabilities with WhiteSource Bold
	Summary
	Questions
	Further Reading
	<b>DevOps Best Practices</b>
	Automating Everything
	Choosing the Right Tool
	Writing All Your Configuration in Code
	Designing the System Architecture
	Building a Good CI/CD Pipeline

	Integrating Tests
	Applying Security with DevSecOps
	Monitoring Your System
	Evolving Project Management
	Summary
	Questions
	Further Reading
	<b>DevOps for Developers</b>
	<b>Part-01: Fundamentals</b>
	<b>Beginning DevOps for Developers</b>
01	The Definition for DevOps – Topics 4
	What DevOps is Not – Topics 2
	Structure of This Book
	Conclusion
	<b>Introducing DevOps</b>
	Traditional Project Settings
	Agile Project Settings
	Blame Game: Dev vs. Ops – Topics 3
	Operations as Bottleneck – Topics 2
	DevOps to the Rescue
	The Essence of DevOps – Topics 3
	Conclusion
	<b>Building Blocks of DevOps</b>
	Measurement and Metrics – Topics 2
	Improving Flow of Features – Topics 2
	Improve and Accelerate Delivery – Topics 2
	Conclusion
	<b>Part-02: Metrics and Measurement View</b>
	<b>Quality and Testing</b>
	What is Quality? – Topics 1
	Patterns for Improving Quality – Topics 5
	Conclusion
	<b>Introduce Shared Incentives</b>
	War Study: Magic Kingdoms
	What is a Team? – Topics 2
	Becoming a Team – Topics 2
	Success Stories – Topics 2
	Conclusion
	<b>Part-03: Process View</b>
	<b>Gain Fast Feedback</b>
	The DevOps Area Matrix
	Extend Development to Operations
	Extend Operations to Development
	Embed Development into Operations
	Embed Operations into Development
	Starting with Kanban – Topics 3
	Example Use Case – Topics 2
	Conclusion
	<b>Unified and Holistic Approach</b>
	Getting Started with Concepts – Topics 1
	Origins of Conceptual Deficits – Topics 4



	Attributes of a Unified Approach – Topics 3
	Conclusion
	<b>Part-04: Technical View</b>
	<b>Automatic Releasing</b>
	Prerequisites for Automatic Releasing
	Patterns with Appropriate Tools – Topics 7
	Conclusion
	<b>Infrastructure as Code</b>
	Starting with Infrastructure as Code – Topics 2
	Test Environments with Vagrant – Topics 4
	Provisioning with Puppet – Topics 2
	Use Case: Open Source Infrastructure with Puppet – Topics 2
	Where to Look Next? – Topics 1
	Conclusion
	Specification by Example
	Getting Started with Acceptance Tests – Topics 1
	Defining Your Acceptance Criteria – Topics 8
	Reporting on the Test Outcomes – Topics 1
	Conclusion
	Moving On

# Cloud Computing and Virtualization

- Cloud Computing: Concepts, Technology and Architecture, Thomas Erl, Zaigham, and Ricardo
- Virtualization Essential, Matthew Portnoy

	Cloud Computing
	<b>Introduction</b>
01	Objectives of This Book
	What This Book Does Not Cover
	Who This Book is for
	How This Book is Organized
	Part-1: Fundamental Cloud Computing
	Understanding Cloud Computing
	Fundamental Concepts and Models
	Cloud-Enabling Technology
	Fundamental Cloud Security
	Part-2: Cloud Computing Mechanisms
	Cloud Infrastructure Mechanisms
	Specialized Cloud Mechanisms
	Cloud Management Mechanisms
	Cloud Security Mechanisms
	Part-3: Cloud Computing Architecture
	Fundamental Cloud Architectures
	Advanced Cloud Architectures
	Specialized Cloud Architectures
	Part-4: Working with Clouds
	Cloud Delivery Model Considerations
	Cost Metrics and Pricing Models
	Service Quality Metrics and SLAs
	Part-5: Appendices
	Case Study Conclusions
	Mapping Mechanisms to Characteristics
	Data Center Facilities (TIA-942)
	Emerging Technologies
	Cloud Provisioning Contracts
	Cloud Business Case Template
	<b>Case Study Background</b>
	Case Study 1: ATN
	Technical Infrastructure and Environment
	Business Goals and New Strategy
	Roadmap and Implementation Strategy
	Case Study 2: DTGOV
	Technical Infrastructure and Environment
	Business Goals and New Strategy
	Roadmap and Implementation Strategy
	Case Study 3: Innovartus Technologies Inc.
	Technical Infrastructure and Environment
	Business Goals and Strategy
	Roadmap and Implementation Strategy
	<b>Part-01: Fundamental Cloud Computing</b>

	<b>Understanding Cloud Computing</b>
	Origins and Influences
	A Brief History
	Definition
	Business Drivers – Topics 3
	Technology Innovations – Topics 4
	Basic Concepts and Terminology
	Cloud
	IT Resource
	On-Premise
	Cloud Consumers and Cloud Providers
	Scaling – Topics 2
	Cloud Service
	Cloud Service Consumer
	Goals and Benefits
	Reduced Investments and Proportional Cost
	Increased Scalability
	Increased Availability and Reliability
	Risks and Challenges
	Increased Security Vulnerabilities
	Reduced Operational Governance Control
	Limited Portability Between Cloud Providers
	Multi-Regional Compliance and Legal Issues
	<b>Fundamental Concepts and Models</b>
	Roles and Boundaries
	Cloud Provider
	Cloud Consumer
	Cloud Service Owner
	Cloud Resource Administrator
	Additional Roles
	Organizational Boundary
	Trust Boundary
	Cloud Characteristics
	On-Demand Usage
	Ubiquitous Access
	Multitenancy (and Resource Pooling)
	Elasticity
	Measured Usage
	Resiliency
	Cloud Delivery Models
	Infrastructure-as-a-Service (IaaS)
	Platform-as-a-Service (PaaS)
	Software-as-a-Service (SaaS)
	Comparing Cloud Delivery Models
	Combining Cloud Delivery Models – Topics 2
	Cloud Deployment Models
	Public Clouds
	Community Clouds
	Private Clouds
	Hybrid Clouds
	Other Cloud Deployment Models

	<b>Cloud-Enabling Technology</b>
	Broadband Networks and Internet Architecture
	Internet Service Providers (ISPs)
	Connectionless Packet Switching (Datagram Networks)
	Router-Based Interconnectivity – Topics 3
	Technical and Business Considerations – Topics 3
	Data Center Technology
	Virtualization
	Standardization and Modularity
	Automation
	Remote Operation and Management
	High Availability
	Security-Aware Design, Operation, and Management
	Facilities
	Computing Hardware
	Storage Hardware
	Network Hardware – Topics 5
	Other Considerations
	Virtualization Technology
	Hardware Independence
	Server Consolidation
	Resource Replication
	Operating System-Based Virtualization
	Hardware-Based Virtualization
	Virtualization Management
	Other Considerations
	Web Technology
	Basic Web Technology
	Web Application
	Multitenant Technology
	Service Technology
	Web Services
	REST Services
	Service Agents
	Service Middleware   Case Study Example
	<b>Fundamental Cloud Security</b>
	Basic Terms and Concepts
	Confidentiality
	Integrity
	Authenticity
	Availability
	Threat
	Vulnerability
	Risk
	Security Controls
	Security Mechanisms
	Security Policies
	Threat Agents
	Anonymous Attacker
	Malicious Service Agent
	Trusted Attacker

	Malicious Insider
	Cloud Security Threats
	Traffic Eavesdropping
	Malicious Intermediary
	Denial of Service
	Insufficient Authorization
	Virtualization Attack
	Overlapping Trust Boundaries
	Additional Considerations
	Flawed Implementations
	Security Policy Disparity
	Contracts
	Risk Management   Case Study Example
	<b>Part-02: Cloud Computing Mechanisms</b>
	<b>Cloud Infrastructure Mechanisms</b>
	Logical Network Perimeter   Case Study Example
	Virtual Server   Case Study Example
	Cloud Storage Device
	Cloud Storage Levels
	Network Storage Interfaces
	Object Storage Interfaces
	Database Storage Interfaces – Topics 2
	Case Study Example
	Cloud Usage Monitor
	Monitoring Agent
	Resource Agent
	Polling Agent   Case Study Example
	Resource Replication   Case Study Example
	Ready-Made Environment   Case Study Example
	<b>Specialized Cloud Mechanisms</b>
	Automated Scaling Listener   Case Study Example
	Load Balancer   Case Study Example
	SLA Monitor   Case Study Example – Topics 2
	Pay-Per-Use Monitor   Case Study Example
	Audit Monitor   Case Study Example
	Failover System
	Active-Active   Case Study Example
	Hypervisor   Case Study Example
	Resource Cluster   Case Study Example
	Multi-Device Broker   Case Study Example
	State Management Database   Case Study Example
	<b>Cloud Management Mechanisms</b>
	Remote Administration System   Case Study Example
	Resource Management System   Case Study Example
	SLA Management System   Case Study Example
	Billing Management System   Case Study Example
	<b>Cloud Security Mechanisms</b>
	Encryption
	Symmetric Encryption
	Asymmetric Encryption   Case Study Example
	Hashing   Case Study Example

	Digital Signature   Case Study Example
	Public Key Infrastructure (PKI)   Case Study Example
	Identity and Access Management (IAM)   Case Study Example
	Single Sign-On (SSO)   Case Study Example
	Cloud-Based Security Groups   Case Study Example
	Hardened Virtual Server Images   Case Study Example
	<b>Part-03: Cloud Computing Architecture</b>
	<b>Fundamental Cloud Architectures</b>
	Workload Distribution Architecture
	Resource Pooling Architecture
	Dynamic Scalability Architecture
	Elastic Resource Capacity Architecture
	Service Load Balancing Architecture
	Cloud Bursting Architecture
	Elastic Disk Provisioning Architecture
	Redundant Storage Architecture Fundamental Cloud Architectures
	<b>Advanced Cloud Architectures</b>
	Hypervisor Clustering Architecture
	Load Balanced Virtual Server Instances Architecture
	Non-Disruptive Service Relocation Architecture
	Zero-Disruptive Service Relocation Architecture
	Cloud Balancing Architecture
	Resource Reservation Architecture
	Dynamic Failure Detection and Recovery Architecture
	Bare-Metal Provisioning Architecture
	Repaid Provisioning Architecture
	Storage Workload Management Architecture
	Case Study Example
	Specialized Cloud Architectures
	Direct I/O Access Architecture
	Direct LUN Access Architecture
	Dynamic Data Normalization Architecture
	Elastic Network Capacity Architecture
	Cross-Storage Device Vertical Tiering Architecture
	Intra-Storage Device Vertical Tiering Architecture
	Load Balanced Virtual Switches Architecture
	Multipath Resource Access Architecture
	Persistent Virtual Network Configuration Architecture
	Redundant Physical Connection for Virtual Servers Architecture
	<b>Part-04: Working with Clouds</b>
	<b>Cloud Delivery Model Considerations</b>
	Cloud Delivery Models: The Cloud Provider Perspective
	Building IaaS Environments – Topics 4
	Equipping PaaS Environments – Topics 3
	Optimizing SaaS Environments   Security
	<b>Cloud Delivery Models: The Cloud Consumer Perspective</b>
	Working with IaaS Environments
	IT Resource Provisioning Considerations
	Working with PaaS Environments
	IT Resource Provisioning Considerations
	Working with SaaS Services

	Case Study Example
	<b>Cost Metrics and Pricing Models</b>
	Business Cost Metrics
	Up-Front and On-Going Costs
	Additional Costs   Case Study Example
	Product Catalog Browser – Topics 4
	Client Database – Topics 4
	Cloud Usage Cost Metrics
	Network Usage – Topics 3
	Server Usage – Topics 2
	Cloud Storage Device Usage – Topics 2
	Cloud Service Usage – Topics 3
	Cost Management Considerations
	Pricing Models
	Additional Considerations   Case Study Example
	Virtual Server On-Demand Instance Allocation
	Virtual Server Reserved Instance Allocation
	Cloud Storage Device
	WAN Traffic
	<b>Service Quality Metrics and SLAs</b>
	Service Quality Metrics
	Service Availability Metrics – Topics 2
	Service Reliability Metrics – Topics 2
	Service Performance Metrics – Topics 7
	Service Scalability Metrics – Topics 3
	Service Resiliency Metrics – Topics 2
	Case Study Example
	SLA Guidelines
	Case Study Example – Topics 5

# UML

- Software Engineering with UML, Bhuvan Unhelkar
- Software Development with UML, Ken Lunn

Software Engineering with UML	
	<b>Software Engineering Fundamentals with Object Orientation</b>
01	Learning Objectives
	Introduction to Software Engineering
	Learning and Adopting Software Engineering
	Importance of Modeling
	Software Engineering Fundamentals – Topics 2
	Classification (Grouping)
	Abstraction (Representing)
	Encapsulation (Modularizing)
	Association (Relating)
	Inheritance (Generalizing)
	Polymorphism (Executing)
	Software Engineering: A Historical Perspective   Evolution of Modeling
	About the UML and Its Purpose   UML Usage
	Common Errors in Interpreting Software Engineering Fundamentals and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Review of 14 Unified Modeling Language Diagrams</b>
	Learning Objectives
	List and Nature of UML Diagrams
	Nature and Basics of UML Diagrams
	Brief Review of UML Diagrams
	Use Case Diagrams
	Activity Diagrams
	Class Diagrams
	Sequence Diagrams
	Interaction Overview Diagrams
	Communication Diagrams
	Object Diagrams
	State Machine Diagram
	Composite Structure Diagrams
	Component Diagrams
	Deployment Diagrams
	Package Diagrams
	Timing Diagrams
	Profile Diagrams
	Difference in List of UML Diagrams
	Common Errors in Understanding UML Diagrams and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Software Projects and Modeling Spaces: Package Diagrams</b>
	Learning Objectives



	Understanding Different Types of Sizes of UML-Based Projects – Topics 2
	Organizing the Project – Topics 3
	The Three Modeling Spaces in Software Engineering – Topics 3
	Mapping UML to Modeling Spaces
	Package Diagrams – Topics 5
	Common Errors in Organizing Project Packages and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>The Software Development Life Cycle and Agility</b>
	Learning Objectives
	Process in Developing Software
	UML and Process
	Process Elements
	Software Development Life Cycles
	Iterative, Incremental, and Parallel Process in Software Development – Topics 4
	Agile in Software Development – Topics 2
	Roles, Ceremonies, and Artifacts – Topics 4
	Disciplined Agile Development
	Composite Agile Method and Strategy
	Common Errors in SDLC and Agile Use and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Use Case Models-1: Actors and Use Cases</b>
	Learning Objectives
	Use Case Modeling in the Problem Space
	Actors – Topics 5
	Clarifying Actor – Class Confusion
	Actor Documentation – Topics 2
	Use Cases – Topics 5
	Example: Use Cases in the Hospital Management System – Topics 6
	Strengths and Weaknesses of Use Cases and Actors – Topics 2
	Relating Use Cases to Packages
	Relating Use Cases to Functional Testing
	Common Errors in Modeling Actors and Use Cases and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Use Case Models-2: Use Case Diagrams and Requirements Modeling</b>
	Learning Objectives
	Use Case Diagrams – Topics 6
	Use Case Relationships – Topics 3
	Naming a Use Case Diagram
	Use case Diagrams for Hospital Management System – Topics 4
	Strengths and Weaknesses of Use Case Diagrams – Topics 2
	Common Errors in Use Case Diagrams and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Activity Diagrams, Interaction Overview Diagrams, and Business Process Models</b>

	Learning Objectives
	Introduction
	Activity Diagrams – Topics 2
	Activity Diagrams for Hospital Management System – Topics 4
	Strengths and Weakness of Activity Diagrams – Topics 2
	Interaction Overview Diagrams – Topics 4
	Strength and Weaknesses of Interaction Overview Diagrams – Topics 2
	Business Process Modeling
	Common Errors in Activity Diagrams, Interaction Overview Diagrams, and Business Process Models...
	Discussion Questions
	Team Project Case Study
	<b>Class Models-1: Classes and Business Entities</b>
	Learning Objectives
	Understanding Business Entities, Classes, and Objects
	Classes and Business Entities
	Identifying and Naming Classes – Topics 2
	Analyzing the “Registers Patient” Use Case to Identify Classes/Business Entities
	Class Definitions – Topics 6
	Visibilities on a Class
	Designing a Class in the Solution Space
	Class Identification in Design (MOSS)
	Strengths and Weaknesses of Classes – Topics 2
	Common Errors in Classes and Business Entities and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	<b>Class Model-2: Basic Class Diagram</b>
	Learning Objectives
	Class Diagrams
	Notations of Class Diagrams
	Inheritance Relationship in a Class Diagram
	Association Relationship in a Class Diagram
	Aggregation Relationship in a Class Diagram
	Multiplicities in Class Diagrams
	Class Diagrams for Hospital Management System – Topics 4
	Strengths of Class Diagrams – Topics 2
	Common Errors in Basic Class Diagram and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	<b>UML’s Extensibility Mechanisms: Notes, Stereotypes, Constraints, and Tags</b>
	Learning Objectives
	UML’s Extensibility Mechanisms
	Notes
	Stereotypes – Topics 8
	Stereotypes for Attributes and Operations – Topics 6
	Profile Diagram
	Constraints
	Tagged Value
	Common Errors in UML’s Extensibility Mechanisms and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnote

	<b>Class Model-3: Advanced Class Designs</b>
	Learning Objectives
	Introduction
	Understanding Class Relationships – Topics 2
	Advanced Relationships in a Class Diagram in Design
	Association Relationship in Design
	Dependency Relationship in Design
	Interface and Realization Relationship in Design
	Aggregation Relationship in Design
	Implementing the Relationships: By References and By Value
	Parameter Visibility
	Multiplicities and Object Diagrams – Topics 3
	Inheritance and Polymorphism in Design – Topics 2
	Incorporating Errors and Exceptions in Design
	Attribute Identification, Naming, and Definition – Topics 5
	Operation Identification, Naming, and Signature – Topics 3
	Common Errors in Modeling Advance Class Designs and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	<b>Interaction Modeling with Sequence Diagrams</b>
	Learning Objectives
	Interaction Modeling
	About Sequence Diagrams
	Sequence Diagrams in Detail – Topics 6
	Sequence Diagrams in Hospital Management System
	Sequence Diagrams in the Problem Space
	Design-Level Sequence Diagrams in the Solution Space – Topics 4
	Strengths and Weaknesses of Sequence Diagrams
	Common Errors in Interaction Modeling with Sequence Diagrams and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	<b>Database Modeling with Class and Sequence Diagrams</b>
	Learning Objectives
	Introduction to Persistence
	Persistence Mechanisms – Databases – Topics 4
	Using Relational Databases in Object-Oriented Designs – Topics 3
	Robustness in Persistence Design – Topics 2
	Inheritance Relationship and Relational Tables
	Mapping Associations in Relational Tables – Topics 3
	Persistence in Practice for HMS – Topics 2
	Incorporating Database Interface Pattern in HMS Persistence Design
	Common Errors in Interpreting Database Modeling and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnote
	<b>Dynamic Modeling with State Machine Diagrams</b>
	Learning Objectives
	Introduction to Dynamic Modeling with State Machine Diagrams
	State Machine Diagrams for Dynamic Modeling
	Notations for State Machine Diagrams
	State Machine Diagrams for Patient Object in Problem Space – Topics 3

	Advanced State Machine Diagram for Patient Object in HMS in Solution Space – Topics 4
	Common Errors in Modeling State Machine Diagrams and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnote
	<b>Advanced Software Engineering Design Concepts: Reuse, Granularity, Patterns, and Robustness</b>
	Learning Objectives
	Introduction
	Reusability in Software Engineering – Topics 4
	Reuse Strategies in Software Projects – Topics 3
	Granularity in Object-Oriented Design
	Design Patterns in Software Design Engineering – Topics 4
	Robustness in Design – Topics 4
	System Architecture and Design Process
	Common Error in Reuse, Granularity, Patterns, and Robustness and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Interface Specifications Prototyping</b>
	Learning Objectives
	Introduction to Interfaces
	Specifying Interface Requirements
	Interface Specifications of HMS – Topics 3
	Examples of User Interface Designs for HMS (Initial Iteration)
	Specifying the Flow of User Interfaces (HMS Example)
	Mobile Applications Interfaces
	Printer Interfaces
	User Interface Design Considerations – Topics 3
	Prototyping – Topics 4
	Common Errors in Interface Specifications and Prototyping and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Implementation Modeling with Component, Deployment, and Composite Structure Diagrams</b>
	Learning Objectives
	Introduction
	Component Diagrams – Topics 5
	Component Diagrams for HMS
	Practical Component Diagram Showing Interdependencies and Packages for HMS
	Strengths and Weaknesses of Component Diagram
	Composite Structure Diagram
	Deployment Diagrams
	UML Notations on a Deployment Diagram
	Process Around Implementation Diagrams
	Common Errors in Implementation Modeling with Component, Deployment, and Composite Structure...
	Discussion Questions
	Team Project Case Study
	Endnote
	<b>Quality of UML Models with Syntax, Semantic, and Aesthetic Checks</b>
	Learning Objectives
	Introduction

	Quality Management, Assurance, and Control (Testing) – Topics 2
	Syntax, Semantics, and Aesthetics Verify and Validate Artifacts, Diagrams, and Models – Topics 4
	Quality Techniques and V&V Checks – Topics 4
	Common Errors in Quality Assurance and Testing of UML Models and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnote
	<b>Software Testing: Plan, Design and Execute</b>
	Learning Objectives
	Introduction – Topics 3
	Organizing the Testing of Software – Topics 3
	Test Approaches – Topics 4
	Test Architecture
	Test Designs – Topics 4
	Test Cases in Solution Space – Topics 3
	Acceptance Test Cases for Hospital Management System – Topics 6
	Class-Based Approach to Test Cases in the Solution Space – Topics 2
	Operational (NFR) Testing
	Some Operational Tests
	Common Errors in Testing in Solution Space and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Nonfunctional (Operational) Requirements Specification and Application</b>
	Learning Objectives
	Nonfunctional (Operational) Requirements – Topics 4
	NFR Categories: Qualities and Constraints – Topics 2
	NFR Levels
	Performance – Topics 3
	Scalability – Topics 2
	Volume
	Operating System
	Mobile OS
	Accessibility
	Reliability and Maintenance
	Environment
	Legal and Compliance
	Security
	Usability and User Experience – Topics 2
	Big Data (Velocity, Variety)
	Cloud
	Common Errors in Handling NFRs and How to Rectify Them
	Discussion Questions
	Team Project Case Study
	Endnotes
	<b>Emerging Information Technologies and Modeling</b>
	Learning Objectives
	Emerging Information Technologies and Modeling
	SMAC Significance
	Service Orientation (Analytic, Utilities)
	Internet of Things

	Mobile and Social Media Applications
	Cloud Integration
	Virtual and Augmented Reality
	Robotics and Machine Learning
	Modeling the Not Only SQL Databases
	Service Orientation Based on the Cloud
	Designing with Services
	Core Elements of Web Services
	XML/SOAP
	Web Services and Modeling – Topics 4
	Discussion Questions
	Team Project Case Study
	Endnotes

# Software Documentation

- Docs for Developers: An Engineer's Field Guide to Technical Writing, Jared Bhattin, Zachary, Jen, David, Heidi

➤	Docs for Developers
	<b>Understanding Your Audience</b>
01	Corg.ly: One month to launch
	The Curse of Knowledge
	Creating and Initial Sketch of Your Users – Topics 3
	Validate Your User Understanding – Topics 2
	Condensing User Research Findings – Topics 3
	Creating a Friction Log
	Summary
	<b>Planning Your Documentation</b>
	Corg.ly: Creating a Plan
	Plans and Patterns
	Content Types – Topics 7
	Planning Your Documentation
	Summary
	<b>Drafting Documentation</b>
	Corg.ly: First Drafts
	Confronting The Blank Page (or Screen)
	Setting Yourself Up for Writing Success – Topics 3
	Creating Your Outline – Topics 2
	Creating Your Draft – Topics 5
	Writing for Skimming – Topics 4
	Getting Unstuck – Topics 5
	Working From Templates
	Finishing Your First Draft
	Summary
	<b>Editing Documentation</b>
	Corg.ly: Editing Content
	Editing to Meet Your User's Needs
	Different Approaches to Editing – Topics 4
	Creating an Editing Process – Topics 3
	Receiving and Integrating Feedback
	Giving Good Feedback
	Summary
	<b>Integrating Code Samples</b>
	Corg.ly: Showing How It Works
	Using Code Samples
	Types of Code Samples
	Principles of Good Code Samples – Topics 5
	Designing Code Samples – Topics 3
	Tooling for Code Samples – Topics 3
	Summary
	<b>Adding Visual Content</b>
	Corg.ly: Worth a Thousand Words
	When Words aren't Enough
	Why Visual Content is Hard to Create – Topics 3

	Using Screenshots
	Common Types of Diagrams – Topics 3
	Drawing Diagrams – Topics 7
	Creating Video Content
	Reviewing Visual Content
	Maintaining Visual Content
	Summary
	<b>Publishing Documentation</b>
	Corg.ly: Ship it
	Putting Your Content Out There
	Building a Content Release Process
	Creating a Publishing Timeline – Topics 4
	Planning for the future
	Summary
	<b>Gathering and Integrating Feedback</b>
	Corg.ly: Initial Feedback
	Listening to Your Users
	Creating Feedback Channels – Topics 5
	Converting Feedback into Action – Topics 2
	Summary
	<b>Measuring Documentation Quality</b>
	Corg.ly: Tuesday After the Launch
	Is My Documentation Any Good?
	Understanding Documentation Quality – Topics 3
	Creating a Strategy for Analytics – Topics 3
	Tips for Using Document Metrics – Topics 5
	Summary
	<b>Organizing Documentation</b>
	Corg.ly: The Next Release
	Organizing Documentation for Your Readers
	Helping Your Readers Find Their Way – Topics 3
	Organizing Your Documentation – Topics 4
	Summary
	<b>Maintaining and Deprecating Documentation</b>
	Corg.ly: A Few Releases Later
	Maintaining Up-to-Date Documentation
	Planning for Maintainability – Topics 3
	Automating Documentation Maintenance – Topics 4
	Removing Content From Your Docset – Topics 2
	Summary



# Software Testing

- Software Testing: Testing Across the Entire Software Development Life Cycle, Geraled D. Everett, Raymond

	Software Testing
	<b>Overview of Testing</b>
01	Introduction
	Objectives and Limits of Testing
	The Value Versus Cost of Testing
	Relationship of Testing to the Software Development Life Cycle
	Tester Versus Developer Roles in Software Testing
	Putting Software Testing in Perspective
	Summary
	<b>The Software Development Life Cycle</b>
	Introduction
	Methodologies and Tools
	The Evolution of System Development Life Cycles
	The Phased Development Methodology
	The Preliminary Investigation Stage
	The Analysis Stage
	The Design Stage
	The Preliminary Construction Stage
	The Final Construction Stage
	The Installation Stage
	Putting Phased Development in Perspective
	Summary
	<b>Overview of Structured Testing</b>
	Introduction
	Checklist Mentality for Software Testers
	SPRAE – A Generic Structured Testing Approach
	Putting the Overview of Structured Testing in Perspective
	<b>Testing Strategy</b>
	Introduction
	The Chess Pieces for Testing Strategies
	The Two-Dimensional Testing Strategy Chess Board
	The Three-Dimensional Testing Strategy Chess Board
	Putting the Testing Strategy into Perspective
	<b>Test Planning</b>
	Introduction
	The Test Plan
	Test Cases
	Writing Your Test Plan and Test Cases in the Real World
	Test Document Standards
	Putting Test Planning in Perspective
	<b>Static Testing</b>
	Introduction
	Goal of Static Testing
	Candidate Documents for Static Testing
	Static Testing Techniques
	Tracking Defects Detected by Static Testing

	Putting Static Testing in Perspective
	<b>Functional Testing</b>
	Introduction
	Functional Test Cases from Use Cases
	An Approach to Functional Testing
	An Approach to Regression Testing
	Detailed White Box Testing Techniques
	Detailed Black Box Testing Techniques
	Summary
	Putting Functional Testing in Perspective
	<b>Structural (Non-Functional) Testing</b>
	Introduction
	Interface Testing
	Security Testing
	Installation Testing
	The Smoke Test
	Administration Testing
	Backup and Recovery Testing
	Putting Structural Testing in Perspective
	Summary
	<b>Performance Testing</b>
	Introduction
	Workload Planning Techniques
	Workload Execution Techniques
	Component Performance Testing
	Round Trip Performance
	Putting Performance Testing in Perspective
	Summary
	<b>The Testing Environment</b>
	Introduction
	Simulations
	Benchmarking
	Testing Environments
	The Goal of a Testing Environment
	Good Testing Environments and Why They Should Be Used
	Bad Testing Environments and Why They Should Be Avoided
	Putting the Testing Environment in Perspective
	Summary
	<b>Automated Testing Tools</b>
	Introduction
	Brief History of Automated Testing Tools for Software
	Test Tool Record/Playback Paradigm
	Test Tool Touchpoint Paradigms
	Test Tool Execution Paradigm
	The Benefits that Testing Tools Can Provide
	The Liabilities that Testing Tools Can Impose
	Putting Automated Testing Tools in Perspective
	Summary
	<b>Analyzing and Interpreting Test Results</b>
	Introduction
	Test Cases Attempted Versus Successful

	Defect Discovery Focusing on Individual Defects
	Defect Discovery Focusing on the Defect Backlog
	Defect Discovery Focusing on Clusters of Defects
	Prior Defect Discovery Pattern Usefulness
	The Rayleigh Curve – Gunsights for Defect Discovery Patterns
	More Defect Tracking Metrics
	Putting Test Results in Perspective
	Summary
	<b>A Full Software Development Lifecycle Testing Project</b>
	Introduction
	Preliminary Investigation Stage
	Analysis Stage
	Design Stage
	Preliminary Construction Stage
	Final Construction Stage
	Implementation Stage
	Post Implementation Stage
	Case Study Closure
	<b>Testing Complex Applications</b>
	Introduction
	1-Tier Applications
	2-Tier Applications
	3-Tier Applications
	n-Tier Applications
	Putting Testing Complex Applications in Perspective
	Summary
	<b>Future Directions in Testing</b>
	Introduction
	Future Directions in Software Development That Could Increase the Need for Testing Professionals
	Software Testing Challenges Already Upon Us
	Software Testing Near Future Challenges
	Software Testing Challenges to Come
	Putting Future Testing Directions in Perspective
	Summary

# Software Quality Assurance

- Software Quality Assurance: From Theory to Implementation, Daniel Galin

	Software Quality Assurance
	Part-01: Introduction
	The Software Quality Challenge
01	The Uniqueness of Software Quality Assurance
	The Environments for Which SQA Methods and Developed
	Summary
	Review Questions
	Topics for Discussion
	What is Software Quality?
	What is Software?
	Software Errors Faults and Failures
	Classification of the Causes of Software Errors
	Software Quality – Definition
	Software Quality Assurance – Definition and Objectives
	Software Quality Assurance and Software Engineering
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Software Quality Factors
	The Need for Comprehensive Software Quality Requirements
	Classifications of Software Requirements into Software Quality Factors
	Product Operation Software Quality Factors
	Product Revision Software Quality Factors
	Product Transition Software Quality Factors
	Alternative Models of Software Quality Factors
	Who is Interested in the Definition of Quality Requirements?
	Software Compliance with Quality Factors
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	The Components of the Software Quality Assurance System – Overview
	The SQA System – An SQA Architecture
	Pre-project Components
	Software Project Life Cycle Components
	Infrastructure Components for Error Prevention and Improvement
	Management SQA Components
	SQA Standards, System Certification, and Assessment Components
	Organizing for SQA – The Human Components
	Considerations Guiding Construction of an Organization's SQA System
	Part-02: Pre-project Software Quality Components
	Contract Review
	Introduction: The CFV Project Completion Celebration
	The Contract Review Process and Its Stages
	Contract Review Objectives

	Implementation of a Contract Review
	Contract Review Subjects
	Contract Reviews for Internal Projects
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 5A: Proposal Draft Reviews – Subjects Checklist
	Appendix 5B: Contract Draft Review – Subjects Checklist
	<b>Development and Quality Plans</b>
	Development Plan and Quality Plan Objectives
	Elements of the Development Plan
	Elements of the Quality Plan
	Development and Quality Plans for Small Projects and for Internal Projects
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 6A: Software Development Risks and Software Risk Management
	<b>SQA Components in the Project Life Cycle</b>
	<b>Integrating Quality Activities in the Project Life Cycle</b>
	Classic and Other Software Development Methodologies
	Factors Affecting Intensity of Quality Assurance Activities in the Development Process
	Verification, Validation and Qualification
	A Model for SQA Defect Removal Effectiveness and Cost
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Reviews</b>
	Review Objectives
	Formal Design Reviews (DRs)
	Peer Reviews
	A Comparison of the Team Review Methods
	Expert Opinions
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 8A: DR Report Form
	Appendix 8B: Inspection Session Findings Report Form
	Appendix 8C: Inspection Session Summary Report
	<b>Software Testing – Strategies</b>
	Definition and Objectives
	Software Testing Strategies
	Software Test Classifications
	White Box Testing
	Black Box Testing
	Summary
	Selected Bibliography
	Review Questions

	Topics for Discussion
	Software Testing – Implementation
	The Testing Process
	Test Case Design
	Automated Testing
	Alpha and Beta Site Testing Programs
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Assuring the Quality of Software Maintenance Components
	Introduction
	The Foundations of High Quality
	Pre-Maintenance Software Quality Components
	Maintenance Software Quality Assurance Tools
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Assuring the Quality of External Participants' Contributions</b>
	Introduction: The HealthSoft Case
	Types of External Participants
	Risks and Benefits of Introducing External Participants
	Assuring Quality of External Participants Contributions: Objectives
	SQA Tools for Assuring the Quality of External Participants; Contributions
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>CASE Tools and Their Effect on Software Quality</b>
	What is a CASE tool?
	The Contribution of CASE Tools to Software Production Quality
	The Contribution of CASE Tools to Software Maintenance Quality
	The Contribution of CASE Tools to Improved Project Management
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Part-04: Software Quality Infrastructure</b>
	<b>Procedures and Work Instructions</b>
	The Need for Procedures and Work Instructions
	Procedures and Procedures Manuals
	Work Instructions and Work Instruction Manuals
	Procedures and Work Instructions: Preparation, Implementation and Updating
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 14A: Design Review Procedure
	<b>Supporting Quality Devices</b>
	Templates

	Checklists
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Staff Training and Certification</b>
	Introduction: Surprises for the “3S” Development Team
	The Objectives of Training and Certification
	The Training and Certification Process
	Determining Professional Knowledge Requirements
	Determining Training and Updating Needs
	Planning Training and Updating Programs
	Defining Positions Requiring Certification
	Planning The Certification Processes
	Delivery of Training and Certification Programs
	Follow-up Subsequent to Training and Certification
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Corrective and Preventive Actions</b>
	Introduction: The “3S” Development Team Revisited
	Corrective and Preventive Actions – Definitions
	The Corrective and Preventive Actions Process
	Information Collection
	Analysis of Collected Information
	Development of Solutions and Their Implementation
	Follow-up of Activities
	Organizing for Preventive and Corrective Actions
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Configuration Management</b>
	Software Configuration, Its Items and its Management
	Software Configuration Management – Tasks and Organization
	Software Change Control
	Release of Software Configuration Versions
	Provision of SCM Information Services
	Software Configuration Management Audits
	Computerized Tools for Managing Software Configuration
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Documentation Control</b>
	Introduction: Where is the Documentation?
	Controlled Documents and Quality Records
	The Controlled Documents List
	Controlled Document Preparation
	Issues of Controlled Document Approval

	Issues of Controlled Document Storage and Retrieval
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Part-05: Management Components of Software Quality</b>
	<b>Project Progress Control</b>
	The Components of Project Progress Control
	Progress Control of Internal Projects and External Participants
	Implementation of Project Progress Control Regimes
	Computerized Tools for Software Progress Control
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Software Quality Metrics</b>
	Objectives of Quality Measurement
	Classification of Software Quality Metrics
	Process Metrics
	Product Metrics
	Implementation of Software Quality Metrics
	Limitations of Software Metrics
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 21A: The Function Point Method
	<b>Costs of Software Quality</b>
	Objectives of Cost of Software Quality Metrics
	The Classic Model of Cost of Software Quality
	An Extended Model for Cost of Software Quality
	Application of a Cost of Software Quality System
	Problems in the Application of Cost of Software Quality Metrics
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>Part-06: Standards, Certification and Assessment</b>
	<b>Quality Management Standards</b>
	The Scope of Quality Management Standards
	ISO 9001 and ISO 9000-3
	Certification According to ISO 9000-3
	Capability Maturity Models – CMM and CMMI Assessment Methodology
	The Bootstrap Methodology
	The SPICE Project and the ISO/IEC 15504 Software Process Assessment Standard
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 23A: CMMI Process Areas
	Appendix 23B: ISO/IEC 15504 Model Processes



	<b>SQA Project Process Standards – IEEE Software Engineering Standards</b>
	Structure and Content of IEEE Software Engineering Standards
	IEEE/EIA Std 12207 – Software Life Cycle Processes
	IEEE Std 1012 – Verification and Validation
	IEEE Std 1028 – Reviews
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	Appendix 24A: IEEE Software Engineering Standards
	Appendix 24B: MIL-STD-498: List of Data Item Descriptions (DIDs)
	Appendix 24C: Task Structure for a Primary Process According to IEEE/EIA Std 12207 – Example
	<b>Part-07: Organizing for Quality Assurance</b>
	<b>Management and Its Role in Software Quality Assurance</b>
	Top Management’s Quality Assurance Activities
	Department Management Responsibilities for Quality Assurance
	Project Management Responsibilities for Quality Assurance
	Summary
	Selected Bibliography
	Review Questions
	Topics for Discussion
	<b>The SQA Unit and Other Actors in the SQA System</b>
	The SQA Unit
	SQA Trustees and THEIR Tasks
	SQA Committees and Their Tasks
	SQA Forums – Tasks and Methods for Operation
	Summary
	Review Questions
	Topics for Discussion

# Software Project Management

- Software Project Management: A Process-Driven Approach, Ashfaque Ahmed
- Essential Scrum: A Practical Guide to the Most Popular Agile Process, Kenneth S. Rubin

	Software Project Management
	Part-01: Project Management Fundamentals
	Introduction to Software Project Management
01	Introduction
	What is Project Management?
	What is Software Project Management?
	Importance of Software Projects
	Problems in Project Management
	Processes in Software Projects
	Project Processes, People, and Technology
	Successful Software Project Manager
	Project Management Processes
	Software Project Initiation – Topics 4
	Software Project Planning   Components of Project Planning
	Software Project Monitoring and Control
	Software Project Closure
	Configuration and Version Control Management
	Management Metrics
	Case Study – Topics 3
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	Project Initiation Management
	Introduction
	Define Project Charter
	Define Project Scope
	Define Project Objectives
	Practical Considerations
	Estimate Initial Project Size
	Estimate Initial Project Effort and Costs
	Estimate Initial Project Schedule
	Create Initial Project Plan
	Project Initiation in Iterative Model
	Stakeholder Influence
	Quality Planning
	Feasibility Study
	Project Division
	Artifacts of Project Initiation
	Case Study – Topics 3
	Chapter Summary
	Exercise
	Review Questions
	Recommended Readings
	Software Project Effort and Cost Estimation

	Introduction
	Effort Estimation Techniques
	Choosing a Suitable Effort Estimate Technique
	Function Point Analysis   Function Point Analysis Usage
	Wide Band Delphi
	COCOMO – Topics 4
	Effort Estimation for Waterfall Model-Based Planning
	Effort Estimation for Iterations Model-Based Planning
	Cost Estimation
	Cost Factor Analysis
	Activity-Based Cost Estimation
	Cost Estimation for Iterations-Based Planning
	Schedule Estimation
	Schedule Estimation for Waterfall Model-Based planning
	Resource Estimation
	Artifacts of Effort and Cost Estimates
	Practical Considerations in Effort and Cost Estimates
	Effort and Cost in Product Development
	Case Study – Topics 3
	Exercise
	Review Questions
	Recommended Readings
	<b>Risk Management</b>
	Introduction
	Causes of Risks – Topics 5
	Causes of Risks – Topics 5
	Risk Categories – Topics 5
	Risk Analysis
	Balancing Act
	Project Risk Management in Agile Models
	Artifacts of Project Risk Management
	Case Study   Risks on This Project
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	<b>Configuration Management</b>
	Introduction
	Configuration Management
	Configuration Management Techniques
	Artifacts of Configuration Management
	Configuration Management Case Study
	Configuration Management for an Incremental Iteration Development Environment
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	<b>Project Planning</b>
	Introduction
	Project Planning Fundamentals – Topics 5
	Project Planning Fundamentals – Topics 5

	Project Planning Fundamentals – Topics 4
	Project Planning Techniques – Topics 2
	Project Planning Artifacts
	Project Planning in Agile Models   Iteration Planning
	Planning at Project Management Office
	Case Study – Topics 2
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	<b>Project Monitoring and Control</b>
	Introduction
	Project Monitoring – Topics 5
	Project Monitoring – Topics 5
	Project Monitoring – Topics 3
	Project Control Techniques – Topics 5
	Project Monitoring and Control Artifacts
	Project Monitoring and Control in Iterative Model – Topics 2
	Case Study – Topics 2
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	<b>Project Closure</b>
	Introduction
	Source Code Management
	Project Data Management
	Project Closure in Iterative Model
	Lessons Learned
	Resource Release
	Data Structures
	Case Study
	Chapter Summary
	Exercises
	Review Questions
	Recommended Readings
	<b>Part-02: Software Life-Cycle Management</b>
	<b>Introduction to Software Life-Cycle Management</b>
	Introduction
	Software Engineering Management
	Software Life-Cycle Management Processes
	Software Life-Cycle in Waterfall Model
	Software Life-Cycle in Iterative Model
	Moving From Waterfall Model
	Software Life-Cycle in Concurrent Engineering Model
	Software Life-Cycle Processes – Topics 6
	Software Life-Cycle Metrics
	Work Products
	Quality Assurance
	Case Study
	Chapter Summary

	Exercises
	Review Questions
	Recommended Readings
	<b>Software Requirement Management</b>
	Introduction
	Software Requirements Development – Topics 2
	Software Requirements Management – Topics 2
	Requirement Life-Cycle Management – Topics 2
	Software Requirements Practical Strategy
	Software Requirements Artifacts
	Software Requirements Quality Control
	Case Study – Topics 3
	Chapter Summary
	Review Questions
	Recommended Readings
	<b>Software Design Management</b>
	Introduction
	Software Design Fundamentals – Topics 3
	Software Design Methods – Topics 2
	Design Version Control   Subversions
	Design Characteristics
	Software Design Techniques – Topics 6
	Software Design for Internet
	Software Design Quality
	Concurrent Engineering in Software Design
	Design Life-Cycle Management
	Module Division (Refactoring)
	Module Coupling
	Case Study – Topics 2
	Chapter Summary
	Review Questions
	Recommended Readings
	<b>Software Construction</b>
	Introduction
	Coding Standards – Topics 6
	Coding Framework
	Reviews (Quality Control) – Topics 4
	Coding Methods – Topics 6
	Configuration Management
	Unit Testing
	Integration Testing
	Software Construction Artifacts
	Software Construction in Iterative Model
	Case Study   Continuous Integration
	Chapter Summary
	Review Questions
	Recommended Readings
	<b>Software Testing</b>
	Introduction
	Problems with Traditional Development Model
	Verification and Validation

	Test Strategy and Planning – Topics 4
	Test Automation – Topics 2
	Test Project Monitoring and Control – Topics 6
	Test Reporting
	Test Artifacts   Management Artifacts
	Practical Considerations
	Software Testing in Iterative Model
	Case Study
	Chapter Summary
	Review Questions
	Recommended Reading
	<b>Production Release and Maintenance</b>
	Introduction
	Product Release Management
	Product Implementation
	User Training
	Maintenance Introduction
	Maintenance Types – Topics 4
	Maintenance Cost
	Maintenance Process
	Maintenance Life Cycle
	Maintenance Techniques – Topics 3
	Case Study – Topics 2
	Chapter Summary
	Review Questions
	Recommended Readings
	<b>Part-03: Software Engineering Management</b>
	<b>Process Standards Introduction</b>
	Introduction
	Root Cause of Problems in Software Projects
	Solutions for Problems in Software Projects
	Standard Process for Software Projects   Process Tailoring
	Standard Process Across Software Projects
	Program Management
	Portfolio Management
	Statistical Process Control on Software Projects
	Cost of Nonstandard Processes
	Organization Training
	Software Project Abandonment
	Defect Prevention
	Software Project without Process
	Process Improvement
	Final Word
	Review Questions
	Recommended Readings
	<b>Software Process Standards and Process Improvement</b>
	Introduction
	CMMI Standards   CMMI Standards in a Nutshell
	ISO Standards   ISO Standards in a Nutshell
	IEEE Standards   IEEE Standards in a Nutshell
	Rational Unified Process   RUP in a Nutshell

	Agile Methodologies   Extreme Programming in a Nutshell
	Test Process Improvement Techniques
	Deming's PDCA Technique
	Test Maturity Model – Topics 6
	Test Process Improvement
	Critical Testing Process
	Systematic Test and Evaluation Process
	Process Improvement Life-Cycle
	Process Standard Certifications
	Benefits of Certification
	How to Apply for a Certification – Topics 2
	Future of Certifications
	Review Questions
	Recommended Readings
	<b>Process Selection</b>
	Introduction
	History of Plan-Driven Model
	Strengths of Plan-Driven Model
	Limitations of Plan-Driven Model
	History of Agile Methods
	Strengths of Agile Methods
	Limitations of Agile Methods
	Once and for All
	Best Practice for Process Selection
	Converting Traditional to Agile Model
	Case Study
	Exercise
	Review Questions
	Recommended Readings
	<b>Part-04: People Management</b>
	<b>Introduction to People Management</b>
	Introduction
	People Management
	Team Management
	Supplier Management
	Customer Management
	Communication Management
	Review Questions
	<b>Team Management</b>
	Introduction
	Organization Structure and Policies – Topics 4
	Motivating the Team
	Team Effectiveness – Topics 3
	Training
	Nurturing
	Conflict Management
	Knowledge Management
	Communication Management
	Case Study
	Review Questions
	Recommended Readings

	<b>Customer Management</b>
	Introduction
	Customer Expectation Management
	Negotiation Management
	Rapport Building Management
	Reporting Management
	Return on Investment
	Bottom Line
	Case Study
	Review Questions
	Recommended Readings
	<b>Supplier Management</b>
	Introduction
	Supplier Search Management – Topics 3
	Supplier Agreement Management – Topics 2
	Supplier Communication management
	Organization Structure
	Account Management
	Project Offshore Transition
	Case Study
	Review Questions
	Recommended Readings
	<b>Part-05: Tools and Techniques</b>
	<b>Software Project Management Tools Introduction</b>
	Introduction
	Compatibility with Environment
	Cost of Tool
	Data Integration Among Tools
	Existing Skills on Tools
	Tool Obsolescence
	Scale of Operation
	Review Questions
	<b>Project Management and Software Life-Cycle Tools</b>
	Introduction
	Requirement Management Tools
	Software Design Management Tools – Topics 2
	Software Build Management Tools – Topics 3
	Software Testing Management Tools – Topics 3
	Project Management Tools
	Project Planning Tools – Topics 2
	Review Questions
	Recommended Readings
	<b>Software Project Templates</b>
	Introduction
	Software Life-Cycle Template Guidelines – Topics 4
	Project Management Template Guidelines – Topics 3
	Recommended Readings
	<b>Future Tools and Techniques</b>
	Introduction
	Software Industry Trends – Topics 5
	Software Industry Trends – Topics 4



	Software Requirement Management Tools
	Software Design Management Tools
	Software Build Management Tools – Topics 3
	Software Testing Management Tools – Topics 5
	Software Project Management Tools
	Recommended Readings
	<b>Essential Scrum</b>
	<b>Introduction</b>
01	What is Scrum?
	Scrum Origins
	Why Scrum?
	Genomica Results
	Can Scrum Help You? – Topics 6
	Closing
	<b>Part-01: Core Concepts</b>
	<b>Scrum Framework</b>
	Overview
	Scrum Roles – Topics 3
	Scrum Activities and Artifacts – Topics 4
	Scrum Activities and Artifacts – Topics 4
	Closing
	<b>Agile Principles</b>
	Overview
	Variability and Uncertainty – Topics 4
	Prediction and Adaptation – Topics 5
	Validate Learning – Topics 3
	Work in Process (WIP) – Topics 4
	Progress – Topics 3
	Performance – Topics 3
	Closing
	<b>Sprints</b>
	Overview
	Timeboxed – Topics 6
	Short Duration – Topics 6
	Consistent Duration – Topics 2
	No Goal-Altering Changes – Topics 6
	Definition of Done – Topics 4
	Closing
	<b>Requirements and User Stories</b>
	Overview
	Using Conversations
	Progressive Refinement
	What are User Stories? – Topics 3
	Level of Detail
	INVEST in Good Stories – Topics 6
	Nonfunctional Requirements
	Knowledge-Acquisition Stories
	Gathering Stories – Topics 2
	Closing
	<b>Production Backlog</b>
	Overview

	Product Backlog Items
	Good Product Backlog Characteristics – Topics 4
	Grooming – Topics 3
	Definition of Ready
	Flow Management – Topics 2
	Which and How Many Product Backlogs? – 4
	Closing
	<b>Estimation and Velocity</b>
	Overview
	What and When We Estimate – Topics 3
	PBI Estimation Concepts – Topics 4
	PBI Estimation Units – Topics 2
	Planning Poker – Topics 3
	What is Velocity?
	Calculate a Velocity Range
	Forecasting Velocity
	Affecting Velocity
	Misusing Velocity
	Closing
	<b>Technical Debt</b>
	Overview
	Consequences of Technical Debt – Topics 5
	Consequences of Technical Debt – Topics 4
	Causes of Technical Debt – Topics 4
	Technical Debt Must Be Managed
	Managing the Accrual of Technical Debt – Topics 3
	Making Technical Debt Visible – Topics 2
	Servicing the Technical Debt – Topics 5
	Closing
	<b>Part-02: Roles</b>
	<b>Product Owner</b>
	Overview
	Principal Responsibilities – 6
	Characteristics/Skills – Topics 4
	A Day in the Life
	Who Should Be a Product Owner? – Topics 4
	Product Owner Combined with Other Roles
	Product Owner Team – Topics 2
	Closing
	<b>ScrumMaster</b>
	Overview
	Principal Responsibilities – Topics 6
	Characteristics/Skills – Topics 6
	A Day in the Life
	Fulfilling the Role – Topics 3
	Closing
	<b>Development Team</b>
	Overview
	Role-Specific Teams
	Principal Responsibilities – Topics 5
	Characteristics/Skills – Topics 5

	Characteristics/Skills – Topics 5
	Closing
	<b>Scrum Team Structures</b>
	Overview
	Feature Teams Versus Component Teams
	Multiple-Team Coordination – Topics 2
	Closing
	<b>Managers</b>
	Overview
	Fashioning Teams – Topics 5
	Nurturing Teams – Topics 4
	Aligning and Adapting the Environment – Topics 4
	Managing Value – Creation Flow – Topics 3
	Project Managers – Topics 2
	Closing
	<b>Part-03: Planning</b>
	<b>Scrum Planning Principles</b>
	Overview
	Don't Assume We Can Get the Plans Right Up Front
	Up-Front Planning Should Be Helpful without Being Excessive
	Keep Planning Options Open Until the Last Responsible Moment
	Focus More on Adapting and Re-planning Than on Conforming to a Plan
	Correctly Manage the Planning Inventory
	Favor Smaller and More Frequent Releases
	Plan to Learn Fast and Pivot When Necessary
	Closing
	<b>Multilevel Planning</b>
	Overview
	Portfolio Planning
	Product Planning (Envisioning) – Topics 3
	Release Planning
	Sprint Planning
	Daily Planning
	Closing
	<b>Portfolio Planning</b>
	Overview – Topics 3
	Scheduling Strategies – Topics 3
	Inflow Strategies – Topics 4
	Outflow Strategies – Topics 3
	In-Process Strategies   Use Marginal Economics
	Closing
	<b>Envisioning (Product Planning)</b>
	Overview – Topics 3
	SR4U Example
	Visioning
	High-Level Product Backlog Creation
	Product Roadmap Definition
	Other Activities
	Economically Sensible Envisioning – Topics 6
	Closing
	<b>Release Planning (Longer-Term Planning)</b>

	Overview – Topics 3
	Release Constraints – Topics 6
	Grooming the Product Backlog
	Refine Minimum Releasable Features (MRFs)
	Sprint Mapping (PBI Slotting)
	Fixed-Date Release Planning
	Fixed-Scope Release Planning
	Calculating Cost
	Communicating – Topics 2
	Closing
	<b>Part-04: Sprinting</b>
	Overview – Topics 3
	Approaches to Sprint Planning – Topics 2
	Determining Capacity – Topics 3
	Selecting Product Backlog Items
	Acquiring Confidence
	Refine the Sprint Goal
	Finalize the Commitment
	Closing
	<b>Sprint Execution</b>
	Overview – Topics 3
	Sprint Execution Planning
	Flow Management – Topics 5
	Daily Scrum
	Task Performance – Technical Practices
	Communicating – Topics 3
	Closing
	<b>Sprint Review</b>
	Overview
	Participants
	Pework – Topics 5
	Approach – Topics 4
	Sprint Review Issues – Topics 3
	Closing
	<b>Sprint Retrospective</b>
	Overview
	Participants
	Pework – Topics 4
	Approach – Topics 5
	Follow Through
	Sprint Retrospective Issues
	Closing
	<b>The Path Forward</b>
	There is No End State
	Discover Your Own Path
	Sharing Best Practices
	Using Scrum to Discover the Path Forward
	Get Going