PeopleCert DevOps.

PeopleCert DevOps Leadership

December 2019
Syllabus v1.1

PeopleCert
All talents, certified.

e-mail: info@peoplecert.org, www.peoplecert.org

1. Introduction

DevOps (a clipped compound of "development" and "operations") is a software development methodology that combines software development (Dev) with information technology operations (Ops). The goal of DevOps is to shorten the systems development life cycle while delivering software releases frequently in close alignment with business objectives in a faster, better, and cheaper fashion. The DevOps approach to work covers three critical success factors: cultural shift, improvement of practices and processes, and leverage of automation technologies.

Due to the proven success of the DevOps approach to work, **DevOps** oriented roles have become a sought-after resource across industries by a constantly growing number of organizations, who need DevOps-savvy people to make the cultural shift, work comfortably in a DevOps environment, and manage an array of technology and tools —referred to collectively as "toolchain" – for process, including but not limited to, coding, building, testing, packaging, releasing, configuring and monitoring.

It's in this continuously evolving context, that PeopleCert designed a suite of DevOps qualifications that reflects the market need to close the skills-gap that many organizations are facing and support the realization of their business objectives and to improve communication, standardization, collaboration and automation for the delivery of quality software products better, faster and with a lower cost.

To reflect the need of this evolving work approach, the PeopleCert DevOps qualification has been structured as follows:

- **PeopleCert DevOps Fundamentals** (14-18 training hours) – Candidates gain practical guidance on how to deliver effective DevOps through 15 essential practices, especially in relation to well-established ITSM practices.
- **PeopleCert DevOps Leadership** (14-18 training hours) – Candidates get guidance on how to lead the implementation of DevOps practices and how to achieve a cultural shift for better collaboration and communication.
- **PeopleCert DevOps Engineering** (coming soon) – Candidates will gain practical guidance on automation tools use and how these tools support DevOps culture and practices.

The **PeopleCert DevOps Fundamentals** certification covers the fundamental **knowledge** required for a candidate to build their knowledge and skills regarding DevOps principles and practices. In addition, the **PeopleCert DevOps Leadership** certification covers more advanced skills, practices and knowledge about DevOps.

The body of knowledge underlying these skills are presented in the official courseware provided by PeopleCert to accredited ATOs. The primary purpose of the syllabus is to provide a basis for accreditation of people involved with DevOps. It documents the learning outcomes related to the qualification and describes the requirements a candidate is expected to meet to demonstrate that these learning outcomes have been achieved at the specific qualification level.

2. PeopleCert DevOps Leadership

2.1. Purpose of the Leadership Qualification

The purpose of this qualification is to confirm that a candidate has sufficient knowledge, understanding of fundamental DevOps skills and is able to work effectively with, or as a leading member of, a DevOps environment, analyzing and applying these skills and knowledge. The **PeopleCert DevOps Fundamentals** certification is a pre-requisite for the **PeopleCert DevOps Leadership** certification.

2.2. Target Group/Audience

This certification is the **second (Leadership) level** of the **PeopleCert DevOps** qualification scheme provided by PeopleCert and is aimed at anyone who wishes to become an efficient leading member of a DevOps environment and requires candidates to have and demonstrate a solid **knowledge** and **understanding** of DevOps terms, principles, tools and practices, as well as demonstrate their **application** skills of how to use tools efficiently and effectively. The certification can also cater for candidates seeking personal certification.

This certification will provide the **Leadership** level of knowledge to its holders and will certify that they have a solid understanding of DevOps Leadership practices using various tools and are able to apply these in everyday work involving DevOps practices. This basic level of skills & knowledge is covered in the **PeopleCert DevOps Fundamentals** level of the qualification scheme provided by PeopleCert.

3. Learning Objectives

At the **Leadership** level course, candidates will be introduced to concepts, terms, principles and tools used by DevOps Leadership to clarify, plan and approach a DevOps Transformation, validate and sustain a DevOps transformation, how the DevOps Full Stack approach can be engaged and implemented within the organization, as well as why DevOps Leadership is needed in modern enterprises, and how it can be aligned to value delivery. In addition, Scrum methodology, people and culture implications as well as the practices, processes, automation and technology used for adapting DevOps within an organization.

Holders of **PeopleCert DevOps Leadership** certification**,** will be able to demonstrate their knowledge, understanding and practical application of:

- What is leadership and what is leadership for DevOps
- How to lead an organization through a DevOps transformation
- Establish the need for urgency for DevOps
- Clarify & align business objectives
- Create a vision and strategy for transformation
- Identify and influence the vital stakeholders
- Lead a culture of self-organized, cross-functional teams
- Gather, broadcast and Implement feedback
- Enable flow across the value stream
- Break work into iterations to accelerate learning and experimentation

- Lead for continuous delivery
- Lead a culture of continual improvement
- Lead for innovation
- Improve organizational resilience & sustainability

3.1. Qualification Scheme Level

Through the above learning objectives, candidates will demonstrate relevant knowledge skills in the following areas:

Main Topics
Introduction to DevOps Leadership Clarifying & Aligning the DevOps Transformation to Value Delivery Planning & Approaching the DevOps Transformation Engaging & Implementing the DevOps Full Stack Validating Results & Sustaining the DevOps Transformation

4. Examination

The **PeopleCert DevOps Leadership** certification exam is designed to validate a candidate's knowledge, and understanding, of DevOps leadership principles, as detailed above as well as how to be able to apply and analyze this knowledge in real situations assisting DevOps transformation within a modern enterprise.

The **Leadership** exam focuses on the following **four** (**4**) categories in the cognitive domain of **Bloom's taxonomy**[1] which is a reference for different levels of learning:

- **Knowledge**
- **Comprehension**
- **Application**
- **Analysis**

4.1. Assessment Approach

The assessment approach used for the **PeopleCert DevOps Leadership** certification focuses on the **four basic** categories of Knowledge, Comprehension, Application and Analysis.

Knowledge is defined as recalling previously learned material, from facts to theories and represents the lowest level of learning outcomes in the cognitive domain. Such learning outcomes are turned in assessment objectives that include **knowing and recalling** such as:

- Common and/or basic terms, definitions, concepts and principles
- Specific processes
- Processes, procedures and project management methods

Comprehension is the lowest level of understanding and entails the ability to grasp the meaning of the

[1] *The Bloom's taxonomy defines **six** (6) levels of learning in the **cognitive** domain (know, comprehend, apply, analyze, evaluate, create), which are both sequential and cumulative and move from the simple to the complex. In order to achieve the 6th level of learning, it must be ensured that the previous five levels have been mastered.*

material taught, including some elements of interpretation, translation or estimation during the process. Such learning outcomes and in turn assessment objectives go beyond simply recalling information and may include:

- Understanding facts, concepts and principles
- Interpreting material (i.e. code, charts, graphs, text, diagrams)
- Justifying a process, procedure and method used

Application is a level where candidates need to combine their knowledge and understanding/comprehension on a subject and be able to create an abstraction. More specifically, candidates are expected to apply their knowledge and understanding so that abstractions, general principles, or methods to specific concrete situations are made. Such learning outcomes and in turn assessment objectives go beyond simply recalling information and may include:

- Use ideas, principles and theories in new, specific and concrete situations
- Being able to choose appropriate methods and tools, apply principles, use a specific approach or identify the selection of options at a given situation
- Apply what is learnt into a new situation
- Apply rules, methods, concepts, principles, and theories
- Learning outcomes in this area require a higher level of understanding than those under comprehension

Analysis is the level that goes beyond application as the candidates need to be able to break down information into its component parts so that its organizational structure may be understood and to make inferences. More specifically, candidates need to break down, discriminate, diagram, detect, differentiate and illustrate which are all important tasks at this level of learning and include the previous levels of knowing, comprehending and applying. Such learning outcomes and in turn assessment objectives go beyond knowing, understanding and applying and may include:

- see patterns that they can use to analyze a problem
- develop divergent conclusions by identifying motives or causes
- make inferences
- find evidence to support generalizations
- identify parts, analyze the relationship between parts, and recognize the organizational principles involved

The assessment incorporates the above learning outcomes as it uses assessment objectives that cater for the above cognitive domain categories.

4.2. Entry Criteria/Training Requirements

For this examination, the candidate should hold the **PeopleCert DevOps Fundamentals** certification, however, there are no training requirements.

In order to be eligible for the **PeopleCert DevOps Leadership** level examination a candidate must be able to demonstrate knowledge and understanding of basic DevOps Leadership terms, principles, processes, practices and be able to apply and analyze this knowledge in real situations assisting DevOps transformation within a modern enterprise and it is recommended that the candidate has received related **Accredited Training** by a PeopleCert accredited training partner.

4.3. Examination Format

The following table details the examination format for the **Leadership** exam:

Delivery	Computer (web proctored or classroom)
Type	**20 Multiple Choice Questions (MCQ)** *Each question is awarded one (1) mark*
Duration	**1 hour (60 minutes)** *For non-native speakers or candidates with a disability, an additional 30 minutes of extra time is allowed.*
Pass Mark	**70% (14 marks out of 20)**
Invigilator / Supervisor / Proctor	**Yes** *Physical or Online Proctoring*
Open Book	**No** *No materials are allowed in the examination room*
Prerequisites	**PeopleCert DevOps Fundamentals certificate holder**
Distinction	**N/A**
Certification validity	**Perpetual**

The tests are derived from a regularly updated question test bank (QTB) based on the test specification detailed below. Questions are used interchangeably among test sets. The overall difficulty level of each test is the same with any other test. A candidate is never assigned the same test in the case of multiple examination attempts.

5. Detailed Syllabus

The syllabus is structured into sections relating to the **major subject headings** and numbered with a single digit section number. The **recommended training hours, per Syllabus Category** are also provided in this table. A total of **eighteen (18) hours** of accredited training is recommended.

Category	Topic	Skill Set
1.0 Introduction to DevOps Leadership	1.1 What is Leadership?	1.1.1 What does it mean to lead?
	1.2 Key Principles of DevOps	1.2.1 Review: What is DevOps?
		1.2.2 The DevOps Full Stack
		1.2.3 Key Principles of DevOps
		1.2.4 15 Essential DevOps Practices
		1.2.5 Leveraging Technology & Automation
	1.3 Leading the Organization through Transformation	1.3.1 Lewin's Model for Change
		1.3.2 Continual Improvement
		1.3.3 A Clear 20/20 Vision for Transformation
		1.3.4 The 20/20 Change Model
2.0 Clarifying & Aligning the DevOps Transformation to Value Delivery	2.1 Establish the need for urgency for DevOps	2.1.1 The IT Value Delivery Problem
		2.1.2 Drivers of Change
		2.1.3 Technology Adoption & Change
		2.1.4 Complexity Creates Fragility & Debt
		2.1.5 The Need for Standardization
		2.1.6 Standardization vs. Complexity
		2.1.7 Gleicher's Formula for Change
	2.2 Clarifying & Aligning Business Objectives	2.2.1 Review: What is Business Value?
		2.2.2 What Happens Without Value Alignment?
		2.2.3 The Importance of True North
		2.2.4 Establish True North Values & Principles

Category	Topic	Skill Set
		2.2.5 Defining Mission vs. Vision
		2.2.6 Building a True North Alignment System
		2.2.7 Example of Business Objectives
		2.2.8 The Planned Enterprise Backlog
		2.2.9 Unplanned Work & the Team Backlog
		2.2.10 Sources of demand
		2.2.11 Building Visibility into All Work Types
		2.2.12 Case Study of True North
3.0 Planning & Approaching the DevOps Transformation	3.1 Creating a Vision & Strategy for Transformation	3.1.1 The importance of vision
		3.1.2 The evolution of a DevOps transformation
		3.1.3 Bi-Modal or Variable Speed IT
		3.1.4 Patterns for scaling DevOps Teams
		3.1.5 Communities of Practice to Bridge Silos
		3.1.6 Clarifying your current state
		3.1.7 Systems Thinking
		3.1.8 Iceberg Model
		3.1.9 Example of Current state
		3.1.10 Current State Assessment
		3.1.11 Example of Future state for the DevOps transformation
		3.1.12 Mental models and structures for DevOps
	3.2 Identifying & Influencing the Vital Stakeholders	3.2.1 Map the critical stakeholders in the DevOps transformation
		3.2.2 The stakeholder management process
		3.2.3 Key considerations when identifying different groups of stakeholders during DevOps transformation
		3.2.4 Ways to overcome resistance and influence critical stakeholders to participate fully in developing the vision and strategy for DevOps in your organization.
		3.2.5 Estimating Stakeholder Support
4.0 Engaging & Implementing the	4.1 Leading a Culture of Self-Organized,	4.1.1 Breaking Down the Wall of Confusion

Category	Topic	Skill Set
DevOps Full Stack	Cross-Functional Teams	
		4.1.2 Pathological culture, Bureaucratic culture and generative culture
		4.1.3 Task Specialization vs. Cross-Functional
		4.1.4 The importance of cross-functional teams
		4.1.5 Enabling Self-organization
		4.1.6 Agile Scrum Teams
		4.1.7 Agile vs. DevOps Teams
		4.1.8 Leadership and Team Authority
		4.1.9 The importance of balancing generalists and pure specialists within DevOps teams
		4.1.10 Phases of evolution in DevOps teaming
		4.1.11 The structure of a functional silo with platform and product teams
		4.1.12 Teaming Changes
		4.1.13 Cultural and Structural Changes
		4.1.14 Trust-Ownership Model
		4.1.15 Knowledge and Skills Planning
		4.1.16 Knowledge and Skills
		4.1.17 Workforce & Talent Management
		4.1.18 Skills & Knowledge Matrix Development
		4.1.19 Published IT competence frameworks: ECF and SFIA
	4.2 Gathering, Broadcasting & Implementing Feedback	4.2.1 Inputs for Identifying the Future State
		4.2.2 Value as the VOC & CTQ
		4.2.3 Measuring Critical to Quality
		4.2.4 Variation Indicates Control
		4.2.5 Customer Engagement Roles - Delivery
		4.2.6 Product/Service Owner Considerations
		4.2.7 Role of the Relationship Manager
		4.2.8 Engagement Roles & Build

Category	Topic	Skill Set
	4.3 Enabling Flow Across the Value Stream	4.3.1 Complexity Impacts Flow & Time
		4.3.2 The Three Types of Lean Work
		4.3.3 A Strategic Perspective on Standardization
		4.3.4 Complexity & Impact of Unplanned Work
		4.3.5 Value Stream Improvement Phases
		4.3.6 Value Stream Mapping
		4.3.7 Waste in a Process
		4.3.8 Metrics - What Should It Measure?
		4.3.9 Leader's Use of Visual Management
		4.3.10 Examples of Visual Management
		4.3.11 Kanban with Scrum (“Scrumban”)
		4.3.12 Making Unplanned Work Visible
		4.3.13 The importance of creating common communication channels
		4.3.14 Communication & Transparency Solutions
		4.3.15 Communication Considerations
	4.4 Breaking Work into Iterations to Accelerate Learning & Experimentation	4.4.1 Agile vs. Waterfall Project Management
		4.4.2 Iterative Product Management
		4.4.3 The Pillars of Agile & Scrum
		4.4.4 Scrum - A Leadership Perspective
		4.4.5 Enabling the Shift Left with Agile XP
		4.4.6 Agile XP Practices
		4.4.7 Visualizing Velocity Improvement
	4.5 Leadership for Continuous Delivery	4.5.1 Requirements for Automation
		4.5.2 Continuous Delivery
		4.5.3 Applying Continuous Delivery at Scale
		4.5.4 Continuous Testing
		4.5.5 Component Testing
		4.5.6 Subsystem or Application Testing

Category	Topic	Skill Set
		4.5.7 End-to-End enterprise system testing
		4.5.8 Isolated feature branch
		4.5.9 Continuous Integration
		4.5.10 The importance of Continuous Improvements
		4.5.11 Trunk Management & Gated Commits
		4.5.12 Release & Deployment Cadences
		4.5.13 Trunk Management & Release Strategies
		4.5.14 Blue-Green Deployment
		4.5.15 The importance of understanding the capabilities and role of each tool in the DevOps toolchain
		4.5.16 Features of DevOps Toolchain
		4.5.17 Orchestration & Integration
		4.5.18 Advantages and disadvantages of Open Source software
		4.5.19 The Larger Tool Ecosystem
		4.5.20 Automation & Tooling Strategies
5.0 Validating Results & Sustaining the DevOps Transformation	5.1 Leading a Culture of Continual Improvement	5.1.1 The Continual Improvement Mindset
		5.1.2 Creating a Culture of Experimentation
		5.1.3 Leaders & Employee Engagement
		5.1.4 Lean Leadership & Gemba
		5.1.5 Leading at Every Level
		5.1.6 The Nature of Change
		5.1.7 Proactive Problem Solving & Continual Improvement
		5.1.8 The DMAIC Model
		5.1.9 Kaizen Events
	5.2 Leading for Innovation	5.2.1 What is Innovation?

Category	Topic	Skill Set
		5.2.2 The Paradox of Collaboration
		5.2.3 The Paradox of Innovation
		5.2.4 Leading Innovation
		5.2.5 Innovation Continuum
		5.2.6 Leadership Paradoxes of Collaboration
	5.3 Improving Organizational Resilience & Sustainability	5.3.1 Antifragility
		5.3.2 From Response to Learning
		5.3.3 DevOps Practices & Disaster Recovery
		5.3.4 DevOps Practices & Resilience
		5.3.5 DevOps Practices & Antifragility
		5.3.6 Embracing Failure as a Learning Tool

6. Test Specification

The **PeopleCert DevOps Leadership** examination will consist of **five** (**5**) sections with the following structure:

Category	Description	Exam (%)
1.0	Introduction to DevOps Leadership	5.0%
2.0	Clarifying & Aligning the DevOps Transformation to Value Delivery	15.0%
3.0	Planning & Approaching the DevOps Transformation	15.0%
4.0	Engaging & Implementing the DevOps Full Stack	50.0%
5.0	Validating Results & Sustaining the DevOps Transformation	15.0%
	Total	**100.0%**

7. Recommended Reading

1. APMG and Smith, Richard et al. *The Effective Change Manager's Handbook.* London, UK: Kogan Page. 2014, page 196.

2. AXELOS. ITIL Continual Service Improvement Model, 2011. https://www.axelos.com

3. Basiri, Ali, Lorin Hochstein, Abhijit Thosar, and Casey Rosenthal. "*Chaos Engineering Upgraded," The Netflix Tech Blog*, September 24, 2015. Accessed May 22, 2018. https://medium.com/netflix-techblog/chaos-engineering-upgraded-878d341f15fa

4. Bass, Len; Weber, Ingo; Zhu, Liming (2015). *DevOps: A Software Architect's Perspective*. ISBN 978-0134049847.

5. Baukes, Mike. *"The Best DevOps Resources Online," UpGuard* (blog), April 4, 2017. Accessed May 8, 2018. https://www.upguard.com/blog/devops-resources-online

6. Beckhard, Richard, and Harris, Reuben T. *Organizational Transitions*. Massachusetts: Addison-Wesley Publishing Company, 1987.

7. Beedle, Mike, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, Ron Jeffries, et al., n.d. "Manifesto for Agile Software Development." *agilemanifesto*. Accessed May 9, 2018. http://agilemanifesto.org/

8. Bhargava, Rajat. *"Is DevOps a Title?" DevOps.com* (blog), March 20, 2014. Accessed May 8, 2018. https://devops.com/is-devops-a-title/

9. Birrell, N.D., Ould, M.A. "*A Practical Handbook for Software Development"*. Cambridge, UK: Cambridge University Press, 1985. I(SBN 0-521-25462-0).

10. Blazeclan, Gurmeet. *Blazeclan Technologies* (Blog). A Guide to TCO for Open Source Software – 7 Reasons Why Open Source Is Not Free. Accessed January 13, 2019. https://24by7masti.wordpress.com/2013/01/20/a-guide-to-tco-of-opensource-software-7-reasons-why-open-source-is-not-free

11. Blessing-White. *"X Model of Employee Engagement."* GP Strategies. (Article). March 2016. Accessed January 13, 2019. https://blessingwhite.com/the-x-model-of-employee-engagement/

12. Boutelle, Jonathan. "Understanding Organizational Stakeholders for Design Success." *Boxes and Arrows*, 2004.

13. Brown, Kyle G. *"Apply the Strangler Application Pattern to Microservices Applications."* IBM. February 13, 2017. Accessed May 28, 2018. https://www.ibm.com/developerworks/cloud/library/cl-strangler-application-pattern-microservices-apps-trs/index.html

14. Cannon, David. *ITIL Service Strategy.* London: TSO (The Stationery Office), 2011.

15. Carr, Nicholas, W.W Norton and Company, n.d. "*Book Review: The Big Switch: Rewiring the World, From Edison to Google.*" Review of *The Big Switch: Rewiring the World, From Edison to Google*, by Mike Evans. *e-Zine*. Accessed May 9, 2018.http://www.cambashi.com/contentmgr/showdetails.php/id/1865/page/3

16. Carr, Nicholas. www.nicholascarr.com (Blog). Accessed January 13, 2019. http://www.nicholascarr.com/

17. Caum, Carl. "*Continuous Delivery Vs. Continuous Deployment: What's the Diff?" Puppet* (blog), August 30, 2013. Accessed May 8, 2018. https://puppet.com/blog/continuous-delivery-vs-continuous-deployment-what-s-diff

18. Ceresani, Necco. *"9 Companies You Wouldn't Expect To Be Using DevOps," XebiaLabs* (blog),

September 11, 2015. Accessed May 8, 2018. https://blog.xebialabs.com/2015/09/11/9-companies-you-wouldnt-expect-to-be-using-devops/

19. Chui, Michael et al. *The Social Economy: Unlocking Value and Productivity Through Social Technologies.* McKinsey Global Institute, 2012. Accessed January 13, 2019. https://www.mckinsey.com/~/media/McKinsey/Industries/High%20Tech/Our%20Insights/The%20social%20economy/MGI_The_social_economy_Full_report

20. Cockcroft, Adrian, MesosCon. *Keynote: Cloud Trends, DevOps, and Microservices*.YouTube, 44:06. August 26, 2015. https://www.youtube.com/watch?v=c0wSmr-u5vQ

21. Cohn, Mike. *"Two Types of Authority Leaders Must Give to Self-Organizing Teams," Mountain Goat Software (blog)*, August 15, 2017. Accessed May 8, 2018. https://www.mountaingoatsoftware.com/blog/two-types-of-authority-leaders-must-give-to-self-organizing-teams

22. Collins, Jim. *Level 5 Leadership: Good to Great: Why Some Companies Make the Leap* and *Others Don't.* New York, NY: Harper Collins, 2001.

23. Complexity Labs, *Systems Thinking*. YouTube, 2015. https://www.youtube.com/watch?v=Miy9uQcwo3U&feature=youtu.be

24. Connors, Roger, et al. *The Oz Principle: Getting Results Through Individual and Organizational Accountability*. New York, N.Y: Portfolio, 2004.

25. Davis, Jennifer and Ryn Daniels. (2016). *Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale.* Sebastopol, CA: O'Reilly Media. ISBN 978-149192630).

26. DevOps Enterprise Forum 2016, n.d. *"Evaluating Organizational Models With a DevOps Lens." IT Revolution*. Accessed May 2, 2018. http://itrevolution.com/devops_enterprise_forum_guidance/

27. Earnshaw, Aliza. "Why I Hired a DevOps Engineer: Bess Sadler of Stanford University," *Puppet* (blog), July 30, 2013. Accessed May 8, 2018. https://puppet.com/blog/why-i-hired-a-devops-engineer-bess-sadler-of-stanford-university

28. Edwards, Damon. (2013-07-23). *"Integrating DevOps tools into a Service Delivery Platform".* http://dev2ops.org/2012/07/integrating-devops-tools-into-a-service-delivery-platform-video/

29. Eoyang, Glenda. Containers, Significant Differences & Transforming Exchanges. In *Conditions for Self-Organization in Human Systems*. Ph.D. Dissertation, Union Institute, Cincinnati, Ohio, 2001.

30. Floyd, David. "Who Killed Sears? 50 Years on the Road to Ruin." *Investopedia*. Last modified April 23, 2018. Accessed May 9, 2018. https://www.investopedia.com/news/downfall-of-sears/

31. Forsgren, Nicole, Gene Kim, Jez Humble, Alanna Brown, and Nigel Kersten. *"2017 State of DevOps Report. See the Difference DevOps Makes—and How to Get There." Puppet*. 2017. Accessed May 8, 2018. https://puppet.com/resources/whitepaper/state-of-devops-report

32. Fowler, Martin. "*How to Break a Monolith into Microservices."* MARTINFOWLER.COM. April 24, 2018. Accessed May 28, 2018. https://martinfowler.com/articles/break-monolith-into-microservices.html

33. Fowler, Martin. *"InfrastructureAsCode."* MARTINFOWLER.COM. March 1, 2016. Accessed May 22, 2018. https://martinfowler.com/bliki/InfrastructureAsCode.html

34. Gilbert, Clark, and Joseph L. Bower. “Disruptive Change When Trying Harder Is Part of the Problem.” *Harvard Business Review*. May 2002. Accessed May 9, 2018. https://hbr.org/2002/05/disruptive-change-when-trying-harder-is-part-of-the-problem

35. Gleeson, Brent, and Megan Rozo. “The Silo Mentality: How to Break Down the Barriers.” Forbes. October 2, 2013. Accessed May 9, 2018. https://www.forbes.com/sites/brentgleeson/2013/10/02/the-silo-mentality-how-to-break-down-the-barriers/#5783e43a8c7e

36. Goldratt, Eliyahu, and Jeff Cox. *The Goal: A Process of Ongoing Improvement*. Massachusetts: The North River Press, 2004.

37. Goodman, M. *The Iceberg Model*. Hopkinton, MA: Innovation Associates Organizational Learning. Accessed January 13, 2019. http://www.ascd.org/ASCD/pdf/journals/ed_lead/el200910_kohm_iceberg.pdf

38. Gruver, Gary, Tommy Mouser and Gene Kim (foreword). (2015). *Leading the Transformation: Applying Agile and DevOps Principles at Scale*. IT Revolution Press. ISBN 978-1942788010.

39. Hackman, J. Richard. *“The Psychology of Self-Management in Organizations.” In Psychology and Work: Productivity, Change, and Employment*, by Pallack M.S. and Perloff R.O. Washington, DC: American Psychological Association, 1986. http://dx.doi.org/10.1037/10055-003

40. Harvey, Cynthia (2017-09-01). "10 Ways DevOps is Changing the Enterprise". https://www.datamation.com/data-center/slideshows/10-ways-devops-is-changing-enterprise-it.html

41. Hassan, Qusay F. “Demystifying Cloud Computing.” *CrossTalk*, Jan/Feb 2011. Accessed May 9, 2018. https://static1.1.sqspcdn.com/static/f/702523/10181434/1294788395300/201101-Hassan.pdf

42. Hill, Linda A., et al. *Collective Genius: The Art and Practice of Leading Innovation*. Boston: Harvard Business Review Press, 2014.

43. Holman, Peggy, et al. *The Change Handbook: The Definitive Resource on Today’s Best Methods for Engaging Whole Systems.* San Francisco: Berrett-Koehler Publishers, 2007, 1C2.1, Page 36-38.

44. Humble, Jez, and David Farley. *“Continuous Delivery: Anatomy of the Deployment Pipeline.” informIT.* September 7, 2010. Accessed January 13, 2019. http://www.informit.com/articles/article.aspx?p=1621865&seqNum=2 https://puppet.com/blog/continuous-delivery-vs-continuous-deployment-what-s-diff

45. Humble, Jez, Joanne Molesky, and Barry O’Reilly. *Lean Enterprise—How High Performance Organizations Innovate at Scale*. California: O’Reilly Media, Inc., 2015.

46. Humble, Jez, n.d. *“What Is Continuous Delivery?” Continuous Delivery.* Accessed May 28, 2018. https://continuousdelivery.com/

47. Hunter, Tim. “A Personal Reinterpretation of the Three Ways.” IT Revolution. July 10, 2013. Accessed May 8, 2018. https://itrevolution.com/a-personal-reinterpretation-of-the-three-ways/

48. Hybels, Bill. *Courageous Leadership*. Grand Rapids, Mich.: Zondervan, 2002.

49. Jesuthasan, Ravin, and John Boudreau. “Thinking Through How Automation Will Affect Your Workforce.” *Harvard Business Review*. April 20, 2017. Accessed May 9, 2018. https://hbr.org/2017/04/thinking-through-how-automation-will-affect-your-workforce

50. Kim, Gene and Kevin Behr (2018). *The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win 5th Anniversary Edition*. IT Revolution Press. ISBN 978-194278829).

51. Kim, Gene, Patrick Debois, Hohn Willis, Jez Humble (2016). *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. Portland, OR: IT Revolution Press. ISBN 978-1942788003.

52. Kim, Gene. *"The Three Ways: The Principles Underpinning DevOps." IT Revolution*. August 22, 2012. Accessed May 8, 2018. https://itrevolution.com/the-three-ways-principles-underpinning-devops/

53. Kotter, John. *Leading Change*. Boston: Harvard Business Review Press, 2012. p. 59.

54. Krstic, Natasa. "Stakeholder Management from the Business Perspective." *Megatrend Review* 11 (2014): 165-182.

55. Kumar, Ajoy. *"How 'Everything as Code' Changes Everything?" WordPress* (blog). March 9, 2016. Accessed May 28, 2018. https://cloudmatters.wordpress.com/2016/03/09/how-everything-as-code-changes-everything/

56. Kumar, Ajoy. *"Infrastructure and How 'Everything as Code' Changes Everything." BMC* (blog). August 17, 2016. Accessed May 28, 2018. https://www.bmc.com/blogs/infrastructure-everything-code-changes-everything/

57. Loukides, Mike (2012-06-07). *"What is DevOps?"* http://radar.oreilly.com/2012/06/what-is-devops.html

58. Mann, Andi. *"Four Key Vectors for Closing the DevOps Feedback Loop," DevOps.com* (blog), September 29, 2014. Accessed May 8, 2018. https://devops.com/four-key-vectors-closing-devops-feedback-loop/

59. Mayer, Chris. *"S Is for Sharing Tells the Father of DevOps, Patrick Debois,"* JAXenter. October 17, 2012. Accessed May 8, 2018. https://jaxenter.com/s-is-for-sharing-tells-the-father-of-devops-patrick-debois-105077.html

60. Mello Jr., John P., n.d. *"The Best DevOps Conferences of 2018." TechBeacon*. Accessed May 8, 2018. https://techbeacon.com/best-devops-conferences-2018

61. Meyer, Bertrand. *"Practice to Perfect: The Quality First Model." EiffelSoft*, (1997): 102–106. Accessed May 22, 2018. http://se.inf.ethz.ch/~meyer/publications/computer/quality_first.pdf

62. Middelburg, Jan-Willem. *Service Automation Framework*. Netherlands: Van Haren Publishers, 2016.

63. Mirchandani, Sanjay. *The State of DevOps in 2017: Transformational Leadership Is Key.* By Chris Cancialosi. Forbes, June 19, 2017. Accessed May 8, 2018. https://www.forbes.com/sites/chriscancialosi/2017/06/19/the-state-of-devops-in-2017-transformational-leadership-is-key/#7cb8b17c63dc

64. Mueller, Ernest. *"What Is DevOps?" the agile admin* (blog), July 24, 2017. Accessed May 8, 2018. https://theagileadmin.com/what-is-devops/

65. Newman, Lily Hay. *"How Netflix DDoS'd Itself to Help Protect the Entire Internet." WIRED*. Last modified July 28, 2017. Accessed May 22, 2018. https://www.wired.com/story/netflix-ddos-attack/

66. Nickolaisen, Neil, and Pixton, Pollyanna. *The Agile Culture: Leading Through Trust and Ownership.* Massachusetts: Addison-Wesley Publishing Company, 2014.

67. Null, Christopher, n.d. "*10 Companies Killing It at Devops." TechBeacon*. Accessed May 8, 2018. https://techbeacon.com/10-companies-killing-it-devops

68. Null, Christopher, n.d. "*Infrastructure as Code: The Engine at the Heart of DevOps."* TechBeacon. Accessed May 22, 2018. https://techbeacon.com/infrastructure-code-engine-heart-devops

69. Nygard, Michael. *"The New Normal: Embracing Failure With Netflix, The Chaos Monkey, and Chaos Kong." cognitect* (blog), March 24, 2016. Accessed May 22, 2018. http://blog.cognitect.com/blog/2016/3/24/the-new-normal-embracing-failure-with-netflix-the-chaos-monkey-and-chaos-kong

70. O' Reilly FREE media/e-books available @ https://www.oreilly.com/webops/free/

71. Panko, Riley. "*The Elusive Definition of DevOps," DevOps.com* (blog), February 14, 2017. Accessed May 8, 2018. https://devops.com/elusive-definition-devops/

72. Pink Think Tank*, Leadership, Structural & Cultural Enablers for DevOps.*

73. Rafferty, Alannah Eileen, and Mark Griffin. *"Dimensions of Transformational Leadership: Conceptual and Empirical Extensions." The Leadership Quarterly* 15, no. 3 (2004): 329–354.

74. Rivington, Mark. "*Secrets to Collaboration in DevOps." DevOps.com.* April 7, 2014. Accessed May 8, 2018. https://devops.com/secrets-collaboration-devops/

75. Schien, Edgar H. Humble *Inquiry: The Gentle Art of Asking Instead of Telling*. California: Berrett-Koehler Publishers, Inc., 2013.

76. Schien, Edgar H. *The Corporate Culture Survival Guide.* California: Jossey-Bass, 2009.

77. Schouten, Edwin. "*Cloud Computing Defined: Characteristics and Service Levels." IBM* (blog). January 31, 2014. Accessed May 28, 2018. https://www.ibm.com/blogs/cloud-computing/2014/01/31/cloud-computing-defined-characteristics-service-levels/

78. Schwaber, Ken & *Jeff* Sutherland, "*SCRUM Guide*", http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100, 2016.

79. Senge, Peter M. *The Fifth Discipline: The Art and Practice of the Learning Organization*. New York, NY: Doubleday/Currency, 1990.

80. Shafer, Andrew Clay, n.d. *"Eating Sushi With Andrew Clay Shafer."* Produced by Arrested DevOps. Podcast, MP3 audio, 41:10. Accessed May 9, 2018. https://www.arresteddevops.com/eating-sushi-with-andrew-clay-shafer/

81. Skelton, Matthew. "*What Team Structure Is Right for DevOps to Flourish?" Matthew Skelton* (blog), October 22, 2013. Accessed May 8, 2018. https://blog.matthewskelton.net/2013/10/22/what-team-structure-is-right-for-devops-to-flourish/

82. Stanley, Andy. *Visioneering: Your Guide for Discovering and Maintaining Personal Vision*. New York, NY: Penguin Random House LLC, 2016.

83. Steinberg, Scott. *"Twenty Surprising Mobile Stats for 2016: The Smartphone Takeover." Mobile Business Insights*. June 29, 2016. Accessed May 28, 2018. https://mobilebusinessinsights.com/2016/06/twenty-surprising-mobile-stats-for-2016-the-smartphone-takeover/

84. Sward, David S. *Measuring the Business Value of Information Technology: Practical Strategies for IT and Business Managers*. Hillsboro, Oregon: Intel Press, 2006.

85. Taylor, Twain. "*The Top 10 Thought Leaders in DevOps," Sweetcode* (blog), May 8, 2017. Accessed May 8, 2018. https://sweetcode.io/top-10-thought-leaders-devops/

86. Thompson, Lee. "*Q&A: Lee Thompson, Former Chief Technologist of E*TRADE Financial."* By

Damon Edwards. *dev2ops*, September 28, 2009. Accessed May 9, 2018. http://dev2ops.org/2009/09/qa-lee-thompson-former-chief-technologist-of-etrade-financial/.

87. Vehent, Julien. (2018). *Securing DevOps: Security in the Cloud.* Manning Publications. ISBN 978-1617294136.

88. Walls, Mandy. (2013). *Building a DevOps Culture*. O'Reilly Media. ISBN 978-1-449-36893-7

89. Watkins, Michael D. *"What Is Organizational Culture? And Why Should We Care?" Harvard Business Review*. May 15, 2013. Accessed May 8, 2018. https://hbr.org/2013/05/what-is-organizational-culture.

90. Westrum, R. *"A Typology of Organisational Cultures." Qual Saf Health Care* 13, suppl. II (2004): ii22–ii27.

91. Wilsenach, Rouan, n.d. *"3 DevOps Techniques for Stress-Free Release Management." TechBeacon.* Accessed May 22, 2018. https://techbeacon.com/3-devops-techniques-stress-free-release-management

92. Womack, James P., and Daniel T. Jones. *Lean Thinking: Banish Waste and Create Wealth in Your Corporation.* New York: Free Press, 2003.

93. "*2016 State of DevOps Report." Puppet*. Accessed May 28, 2018. https://puppet.com/resources/whitepaper/2016-state-of-devops-report

94. *"Amazon." Wikipedia.* Accessed May 9, 2018.https://en.wikipedia.org/wiki/Amazon_(company)

95. *"Fordism." Wikipedia.* Accessed May 9, 2018. https://en.wikipedia.org/wiki/Fordism

96. "*Internet Usage Statistics." Internet World Stats.* Accessed May 28, 2018. http://www.internetworldstats.com/stats.htm

97. *"Kano Model." Wikipedia.* Accessed May 9, 2018. https://en.wikipedia.org/wiki/Kano_model

98. *"Microservices." Wikipedia*. Accessed May 25, 2018. https://en.wikipedia.org/wiki/Microservices

99. *"Principles of Chaos Engineering." principlesofchaos*. Last updated May 2018. Accessed May 22, 2018. http://principlesofchaos.org/

100. *"Scientific Management." The Economist*. February 9, 2009. Accessed May 9, 2018.https://www.economist.com/node/13092819

101. *"The Netflix Simian Army." The Netflix Tech Blog*, July 19, 2011. Accessed May 22, 2018. https://medium.com/netflix-techblog/the-netflix-simian-army-16e57fbab116

102. *"The Netflix Simian Army"*, The Netflix Tech Blog. https://medium.com/netflix-techblog/the-netflix-simian-army-16e57fbab116

103. *"The World Is Flat." Wikipedia*. Accessed May 9, 2018. https://en.wikipedia.org/wiki/The_World_Is_Flat

104. "*What Is the Kano Model?" kanomodel.com*. Accessed May 9, 2018. https://www.kanomodel.com/.

8. Further Reading

1. *2017* State of DevOps Report – Puppet Labs. https://puppet.com/resources/whitepaper/state-of-devops-report

2. "*Branching* (Version Control)." *Wikipedia.com*. Accessed January 13, 2019. https://en.wikipedia.org/wiki/Branching_(version_control)

3. *Cunliff*, Ed. "Connecting Systems Thinking and Action." *The Systems Thinker* (Article). *n.d.* Accessed January 13, 20 19. https://thesystemsthinker.com/connecting-systems-thinking-and-action/

4. *Cohn*, Mike. "Two Types of Authority Leaders Must Give to Self-organizing Teams." *Mountain Goat Software* (Article). August 15, 2017. Accessed January 13, 20 19. https://www.mountaingoatsoftware.com/blog/two-types-of-authority-leaders-must-give-to-self-organizing-teams

5. *Fowler*, Martin. "Branch by Abstraction." *MartinFowler.com* (Article). January 7, 2014. Accessed January 13, 2019. https://martinfowler.com/bliki/BranchByAbstraction.html

6. *Fowler*, Martin. "Feature Branch." *MartinFowler.com* (Article). September 3, 2009. Accessed January 13, 2019. https://martinfowler.com/bliki/FeatureBranch.html

7. *Fowler*, Martin. "Feature Toggle." *MartinFowler.com* (Article). October 29, 2010. Accessed January 13, 2019. https://martinfowler.com/bliki/FeatureToggle.html

8. Glesson, Brent. "The Silo Mentiality: How to Break Down the Barriers." *Forbes.com* (Blog). October 2, 2013. Accessed January 13, 2019. https://www.forbes.com/sites/brentgleeson/2013/10/02/the-silo-mentality-how-to-break-down-the-barriers/#7505326e8c7e

9. *Nayar*, Vineet. Three Differences Between Managers and Leaders. *Harvard Business Review* https://hbr.org/2013/08/tests-of-a-leadership-transiti

10. *Pink* Think Tank, 2017 White Paper – *Critical Success Factors for DevOps.*

11. *Pink* Think Tank, 2017 White Paper #2 – *Leadership, Structural & Cultural Enablers for DevOps*.

12. *Newman*, Lily H. "How Netflix DDOS'd Itself To Help Protect The Entire Internet." *Wired.com* (Blog). July 28, 2017. Accessed January 13, 2019. https://www.wired.com/story/netflix-ddos-attack/

13. *Nygard*, Michael. "The New Normal: Embracing Failure With Netflix, The Chaos Monkey, And Chaos Kong." *Cognitect.com* (Blog). March 24, 2016. January 13, 2019. http://blog.cognitect.com/blog/2016/3/24/the-new-normal-embracing-failure-with-netflix-the-chaos-monkey-and-chaos-kong

14. "The Elusive Definition of DevOps." *DevOps.com* (Blog). February 14, 2017. Accessed January 13, 2019. https://devops.com/elusive-definition-devops/

15. *Westrum*, R. "A Typology of Organisational Cultures." *Qual Saf Health Care* 2004;13(Suppl II):ii22–ii27. doi: 10.1136/qshc.2003.009522 Accessed January 13, 2019. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765804/pdf/v013p0ii22.pdf

16. "*You're* Doing it Wrong." *Truckbaseddevelopment.com* (Article). *n.d*. Accessed January 13, 2019. https://trunkbaseddevelopment.com/youre-doing-it-wrong/

9. Glossary

Term	Definition
20/20 Model	The 20/20 Change Model from Pink Elephant uses widely accepted Organizational Change Management principles to define the steps needed to achieve true organizational change in a corporate IT environment.
Agile Manifesto	Individuals and interactions over processes and tools Working software over comprehensive documentation Customer collaboration over contract negotiation Responding to change over following a plan*
Agile Principles	The twelve principles that comprise the Agile Manifest.
Agile Project Management	An adaptive approach to project management that implies that requirements – made at the very start of a project – are expected to change and evolve as the project advances. It uses continual iterations to adapt and incorporate these changes.
Antifragility	The means to not only respond to and resist incidents and disruptions of all kinds but to use them as an opportunity for learning and adaptation.
Automation	The technique, method, or system of operating or controlling a process by highly automatic means, as by electronic devices, reducing human intervention to a minimum.
Bi_modal IT	From Gartner: Bimodal is the practice of managing two separate but coherent styles of work: one focused on predictability; the other on exploration. Mode 1 (Business As Usual) is optimized for areas that are more predictable and well-understood. Mode 2 is exploratory, experimenting to solve new problems and optimized for areas of uncertainty. Both play an essential role in the digital transformation.
Blue-Green Deployment	In DevOps, the concept of operating two identical cloud-based infrastructures, one designated as Blue, the other Green. To facilitate zero-downtime deployments, i.e., Blue is the current production environment while Green incorporates new changes & updates and allows for production-level testing. When the test environment is stable, the production environment is switched and the process repeats itself.
Business Value	Business Value is the level to which a service meets a customer's expectations or exceeds them.
C.A.L.M.S	An acronym for the values of DevOps: Culture, Automation, Lean, Measurement, and Sharing.
Change Control	The ITIL® practice of ensuring that risks are properly assessed, authorizing changes to proceed and managing a change schedule in order to maximize the number of successful service and product changes. †
Complexity	In the IT/DevOps context, complexity refers to much more than technology. Over time, IT has solved problems with a very narrow or local focus due to urgency and (often) customer pressure. This "quick fix" mode has resulted in entire systems that cannot integrate plus a patchwork of local and 3rd party suppliers that cannot integrate or even communicate effectively.
Containerization	The bundling of an entire runtime environment – into one package or "container" so that the application platform and its dependencies, differences in OS distributions and underlying infrastructure are abstracted away.
Continual Improvement	The practice of aligning an organization's practices and services with changing business needs through the ongoing identification and improvement of all elements involved in the effective management of products and services. †
Continuous Delivery	A set of practices designed to ensure that code is always able to be deployed rapidly and safely throughout its lifecycle to production, achieved by pushing the executables into a production-like environment and conducting automated testing to detect problems.

Term	Definition
Continuous Deployment	An extension of the concept of Continuous Delivery in which all changes that pass automated tests are automatically pushed into production. It automates the step that was previously manual in Continuous Delivery, and enables multiple deployments per day.
Continuous Integration	The practice of merging all developer working copies into a shared mainline (a code repository or a main code trunk) throughout the day. Within an automated Continuous Delivery process, continuous integration covers mainly the build stage. Usually, continuous integration applies to integrating, building and testing code within the development environment.
Continuous Testing	The execution of automated tests at every stage of the deployment pipeline. It provides immediate feedback at each stage to mitigate risk. Automated continuous testing is a key component of continuous integration and delivery. It ensures that the code and environment operate properly and remain in a deployable state.
Conway's Law	States that “organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.” ‡
Critical To Quality	According to the principles of Lean, the value items that should be prioritized and focused on.
CSF	Critical Success Factor. Any team, process, outcome, etc. that must be in place to succeed.
CTQ	Critical To Quality. From Lean VOC, which of the many items on the customer’s “wish list” are absolutely critical to achieve value for the customer.
Daily Scrum	A 15-minute time-boxed Scrum event for the Development Team to synchronize activities and create a plan for the next 24 hours.‖
Daily Scrum Board	A Scrum Visual Management tool that makes transparent what each team member is working on, and captures their attitude towards each day.
Definition of Done	In Agile/Scrum, the list of criteria which must be met before a product increment is considered "done". In DevOps, the product is error-free, deployed into production, released to the customer and is in use and providing value.
Deployment Management	The ITIL® practice of moving new or changed hardware, software, documentation, processes, or any other service component to live environments. †
Deployment Pipelines	Model your current delivery process as stages that then allow you to examine your end to end delivery for bottlenecks, opportunities for automation, and opportunities for collaboration.
Development Team	The Agile Scrum team which consists of professionals who do the work of delivering a potentially releasable Increment of “Done” product at the end of each Sprint. ‖
DevOps	A coined noun used to describe an evolution of existing IT best practices from ITIL, Lean and Agile into a development and operations approach that supports automation and continuous delivery, and encourages a culture of collaboration and learning to help IT deliver business value better, faster and cheaper than ever before.
DevOps Team	One view of DevOps re-structures existing vertical IT technical domains and departments into Product Teams, who create, deploy and manage the products and Platform Teams, who create and manage the various platforms necessary to deliver value to the business on a consistent basis.
Digital Transformation	A profound transformation that encompasses all organizational activities, processes, skills and cultural attitudes.
Disaster Recovery	The means to respond to worst-case scenarios and protect critical systems from incidents and disruptions.
DMAIC	From Six Sigma: DMAIC (Define, Measure, Analyze, Improve, Control) refers to a data-driven improvement cycle used for improving, optimizing and stabilizing business processes and designs.

Term	Definition
DMAIC Cycle	A model which provides guidance for continual improvement throughout five stages: Define, Measure, Analyze, Improve and Control.
Feedback	Occurs when outputs are routed back as inputs as part of a chain of cause-and-effect that forms a loop.
Flow	The way people, information and products move through a process.
Functional Testing	Testing of the features which are necessary for the product to work, including unit, API, integration or systems testing.
Gemba	From Lean: the place where work happens. Lean states that leaders and managers should perform a “Gemba walk” at least once a day. Go to where the work actually happens and learn what people actually do.
Iceberg Model	Most (80%) of an iceberg is invisible and below the waterline. This model uses the iceberg as an analogy to understand that there is “far more than meets the eye” when examining a DevOps transformation.
Incident	According to ITIL® an unplanned interruption to a service or reduction in the quality of a service. †
Incident Management	The ITIL® practice of minimizing the negative impact of incidents by restoring normal service operation as quickly as possible. †
ITIL Service Owner	A role within ITIL® that is accountable for the delivery of a specific service. †
ITIL®	Best-practice guidance for IT service management.
Kanban	Kanban emerged in the 1940s as part of the initial evolution of Lean manufacturing. It provided a way for assembly line workers to notify downstream partners of demand for parts or other work. This allowed for transparency and increased communication, and it standardized processes.
Knowledge Management	The ITIL® practice of maintaining and improving the effective, efficient, and convenient use of information and knowledge across an organization. †
KPI	Key Performance Indicators. Trends. Metrics taken over time and compared will generate trends.
Lead Time	The time between input and output. It triggers the receipt of value.
Leadership	Leadership is the art of motivating a group of people to act towards achieving a common goal. In a business setting, this can mean inspiring workers and colleagues with a vision and strategy to meet the organization's needs.
Lean	A quality system focused on flow. It is all about increasing customer value, eliminating waste, and continuous improvement.
Lean Kaizen	A structured approach for solving problems that is about improving flow and processes incrementally, with an attitude or mindset that encourages everyone at every level of an organization to look for small ideas which, if possible, can be implemented easily and quickly. Kaizen should be part of the daily culture of an organization.
Local Optimization	An environment that is structured and built to produce the best results for the individual or team. While it is important to create local efficiencies, you should be able to see how process design that is locally optimized within a silo can potentially be a problem.
Metric	A data point. Metrics are a series of data points taken at specific times. Also referred to as a “snapshot”.
Microservice Architecture (MSA)	An architecture in which one function is associated with one service which is scaled by distributing services across nodes.
Muda	Waste in processes identified by Lean, including defects and rework, over-processing and overproduction.
Mura	Variances and variation in processes identified by Lean, including variability in volume and spread in the outcome of processes.
Muri	Overburden and inflexibility in processes identified by Lean, including inability to scale up or down with demand.
Necessary Non-Value Add Work	According to Lean, work in a process that should be minimized. This is work that is not value-add, but that must be done.

Term	Definition
Non-Functional Testing	Testing of system operation rather than specific outcomes, including performance, security, compliance or capacity testing.
Non-Value Add Work	According to Lean, work in a process that should be removed.
Operational Level Agreement (OLA)	A written agreement between an IT service provider and another area of IT that assists with the provision of services. Its targets should underpin those within a Service Level Agreement.
Organizational Culture	Organizational Culture is the pattern of shared assumptions and values learned within an organization. It takes its cue from observable patterns of behavior in an organization that are picked up on over time by those who work there.
Potentially Shippable Product Increment	A Scrum artifact – the items completed based upon the Sprint Backlog according to the agreed Definition of Done, the acceptance criteria around both functional and non-functional requirements.
Problem	According to ITIL®, a cause, or potential cause, of one or more incidents. †
Problem Management	The ITIL® practice of reducing the likelihood and impact of incidents by identifying actual and potential causes of incidents and managing workarounds and known errors. †
Process	A structured set of activities designed to accomplish a specific objective. They take one or more defined inputs and turn them into defined outputs.
Process Time	The total time spent actually creating products or services.
Product Backlog (Scrum team)	An ordered list of everything that is known to be needed in the product. It is the single source of requirements for any changes to be made to the product.
Product Backlog	The Scrum artifact consisting of an ordered list of everything that might be needed in the product which is the single source of requirements for any changes to be made to the product. ‖
Product Backlog Board	A Scrum Visual Management tool that makes the project goal visible, and tracks the Work Items being moved to the top and which are completed.
Product Owner	The Scrum role responsible for creating and maintaining the Product Backlog. They are in constant communication with the customer and collaborate with the team. ‖
Product Owner (Scrum team)	Scrum Product Owner is responsible for maximizing the value of the product resulting from the work of the Development Team and is is the sole person responsible for managing the Product Backlog.
Pull System	A system in which products or services are pulled through the process by customer demand. This contrasts with a Push System, in which products or services are pushed through the process based on forecasted demand. Pull systems reduce waste in processes.
Relationship Management	Relationship Management stimulates, surfaces and shapes business demand for a provider's products and services and ensures the potential value from those products and services is captured and optimized.
Release Burndown Chart	A Scrum Visual Management tool which shows the items at the start of the project and the progress of items completed on a Sprint by Sprint basis.
Release Management	The ITIL® practice of making new and changed services and features available for use. †
Resilience	The means to respond to, but also resist, incidents and disruptions of all kind.
Scrum	An adaptable Agile process framework within which various processes, tools and techniques can then be applied. It promotes the development of products in an iterative way that results in more frequent releases with the highest quality outcomes possible.
Scrum Development Team	A Scrum Development Team consists of professionals who do the work of delivering a potentially releasable Increment of "Done" product at the end of each Sprint.

Term	Definition
Scrum Master	The Scrum role responsible for ensuring Scrum is understood and enacted. Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules. ‖
Scrum Sprint	The heart of Scrum, a time-box of one month or less during which a "Done", useable, and potentially releasable product Increment is created. ‖
Service Configuration Management	The ITIL® practice of ensuring that accurate and reliable information about the configuration of services, and the configuration items that support them, is available when and where needed. †
Service Level Agreement (SLA)	A documented agreement between a service provider and a customer that identifies both services required and the expected level of service. †
Service Level Management	The ITIL® practice of setting clear business-based targets for service performance so that the delivery of a service can be properly assessed, monitored, and managed against these targets. †
Service Management	A set of specialized organizational capabilities for enabling value for customers in the form of services. †
Service Owner	The person who is totally accountable for the success of a given service.
Service-Oriented Architecture (SOA)	An architecture style which separates functions into distinct units, or services, which developers make accessible over a network in order to allow users to combine and reuse them in the production of applications. These services and their corresponding consumers communicate with each other by passing data in a well-defined, shared format, or by coordinating an activity between two or more services.
Shift Left	Ensures that quality is built in earlier in the development process so that issues are detected earlier and can be resolved, and defects or errors don't impact production.
Silo Mentality	Occurs when a team or department shares a set of common tasks but operates distinctly from other groups, with their power derived from association with a function or shared technical knowledge.
Silos	Tall vertical storage enclosures on most farms. In the corporate context, a vertically-based organizational structure with little or no crossover between each silo. In many cases, all improvements and communication takes place only within the silo.
Sprint	The heart of Scrum is a Sprint, a time-box of one month or less during which a "Done", useable, and potentially releasable product Increment is created.
Sprint Backlog	The Scrum artifact consisting of a set of Product Backlog items selected for the Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal. ‖
Sprint Backlog Board	A Scrum Visual Management tool that makes the Sprint goal visible, and tracks the Work Items in progress and which are completed.
Sprint Burndown Chart	A Scrum Visual Management tool used in Sprint Planning sessions to understand the pace of the Development Team and bring the correct amount of work into the Sprint.
Sprint Execution	The Scrum event that begins after Sprint Planning and ends when time for Sprint Review and Retrospective is subtracted from the overall Sprint. During Execution, items in the Sprint Backlog become WIP and are worked on until completed in accordance with a Definition of Done.
Sprint Planning	The Scrum event that is where the work to be performed in the Sprint is planned. This plan is created by the collaborative work of the entire Scrum Team. ‖
Sprint Retrospective	A Scrum event that provides the Development Team with an opportunity to reflect on the past Sprint and make any adjustments or refinements (inspect and adapt) to how they conduct the Sprint. ‖
Sprint Review	A Scrum event that is held when Sprint Execution ends. All items completed in accord with the Definition of Done are demonstrated to the Product Owner, customer and other key stakeholders. This allows everyone to agree on accepting the final products. ‖

Term	Definition
Stakeholders	Any individual or group that will benefit or be affected by an initiative, project or change.
Standarization	Often used in the past to discuss technology, DevOps and Lean broaden the scope to include all aspects of IT value creation. Portfolio, Projects, Practices, Processes & Procedures should be standardized in addition to the technology. This provides an environment where there is a definition of Standard Working.
Strangler Application	A microservice application that is used to “strangle” or overtake a monolithic application. See also “Strangler Application pattern.”
Strangler Application Pattern	A monolithic application that is “strangled” over time by the iterative introduction of microservices—the Strangler Applications—to seamlessly replace its specific features and functions.
Systems Thinking	Understanding that your function is as an interrelated and interdependent part of a larger system, which is defined by boundaries and more than the sum of its parts.
Technical Debt	The accumulation of complicated workarounds and rework that occurs when easy solutions are consistently implemented instead of the best solutions.
Test Driven Development (TDD)	The practice of preparing test scenarios before a program is written so that the goal for the programmer is to write something that can pass the exact test.
The Full Stack	Represents the three core aspects of any DevOps implementation that represent critical factors for success: creating the right culture with the right people, putting the right processes and practices into place, and adding the level of technology and automation to that culture and those practices to streamline and accelerate them.
The Three Ways	Refers to the Three Key principles of DevOps: Flow, Feedback, Continuous experimentation and learning.
Theory of Constraints	States that no complex system or process can be more efficient or stronger than its most limiting bottleneck or constraint. It helps organizations to identify and focus the one area that is the slowest and most inefficient, because that area constrains the entire system – it sets the speed for the entire organization.
Total Cycle Time	The total time from the beginning to the end of your process.
Transformation (DevOps)	Changing the game. DevOps is not about changing the rules of the existing IT game but rather it totally changes what game IT is playing. Culture & People, Processes & Practices, Technology & Automation are ALL completely transformed under a DevOps focus.
Transformational Leadership	A leadership style that shifts the organizational culture towards something generative, reinforces a shared set of priorities and goals and that supports DevOps.
True North	Where the compass should always point towards when moving forward and making decisions. They should be established in a simple and straightforward way, and be clear and easy to summarize.
True North Values	True North Values are where the compass should always point towards when moving forward and making decisions. They should be established in a simple and straightforward way, and be clear and easy to summarize.
User Story	An informal, natural language description of one or more features of a software system. User stories are often written from the perspective of an end user or user of a system.
Utility	The functionality offered by a product or service to meet a particular need. Utility can be summarized as ‘what the service does’ and can be used to determine whether a service is ‘fit for purpose’. To have utility, a service must either support the performance of the consumer or remove constraints from the consumer. Many services do both. †

Term	Definition
Value Delivery	Value is made up of three major components: quality, speed & cost. In DevOps, delivering on all three of these is the ultimate goal. Projects, work activities and processes should all be assessed based on their contribution to the overarching goal of delivering value to the customer. In the final analysis, it is the customer who decides what is valuable to them.
Value Stream	Provides a holistic view of IT as delivering end to end business value. Understanding not only your own place within the value stream but the value stream as a whole is essential for leadership and governance.
Value-Add Work	According to Lean, work in a process that should be optimized. It is only work that the customer actually experiences and sees as value. It is what the customer is willing to pay for.
Vision	The future state of a project, an initiative or an entire organization. The long-term (years) direction that an organization needs to move in order to succeed.
VOC	Voice of the Customer. A key tool coming from Lean to help us understand just who the customer is and what exactly do they define as value.
Wall of Confusion	The virtual wall that exists between Dev and Ops. In IT, Dev and Ops often operate with completely different agendas and goals. The term derives from the feeling that Dev "throws it over the wall" to Ops.
Warranty	Assurance that a product or service will meet agreed requirements. Warranty can be summarized as 'how the service performs' and can be used to determine whether a service is 'fit for use'. Warranty often relates to service levels aligned with the needs of service consumers. This may be based on a formal agreement, or it may be a marketing message or brand image. Warranty typically addresses such areas as the availability of the service, its capacity, levels of security, and continuity. A service may be said to provide acceptable assurance, or 'warranty', if all defined and agreed conditions are met. †
Waterfall Project Management	The traditional approach to managing projects. First applied to software projects in the 1960's. The approach begins with Requirements gathering, then Design, Build, Test, Deployment and Release. The work flows left to right in a linear fashion downhill through each step like a waterfall.
WIP	Work in Progress. Lean exposes the obvious fact that there are limits to the WIP capacity for a given team or step in a process. These WIP limits are then used to define a "Pull" system.

REFERENCES

[*] Beedle, Mike, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, Jim Highsmith, Andrew Hunt, et al. “Manifesto for Agile Software Development.”, 2001. https://agilemanifesto.org/.

[†] AXELOS Limited. *ITIL® Foundation ITIL 4 Edition*. TSO, 2019.

[‡] Conway, Melvin. “Conway’s Law.” Accessed June 25, 2019. http://www.melconway.com/Home/Conways_Law.html.

[‖] Schwaber, Ken, and Jeff Sutherland. “The Scrum Guide™.” Scrum.Org and ScrumInc., 2014.

- This page is left intentionally blank -

- This page is left intentionally blank

PeopleCert

DevOps.