PeopleCert DevOps.

PeopleCert DevOps Fundamentals

September 2019
Syllabus v1.0

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 (which is the next level of the qualification) 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 Fundamentals

2.1. Purpose of the Fundamentals (Foundation level) Qualification

The purpose of this foundation level qualification is to confirm that a candidate has sufficient knowledge, understanding and application of fundamental DevOps skills and are be able to work effectively with, or as a member of, a DevOps environment. The Fundamentals qualification is also a pre-requisite for the **PeopleCert DevOps Leadership** qualification.

2.2. Target Group/Audience

This certification is the **first (Fundamentals) level** of the **PeopleCert DevOps** qualification scheme provided by PeopleCert and is aimed at anyone who wishes to become an efficient 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 **fundamental** level of knowledge to its holders and will certify that they have a solid understanding of DevOps using various tools.

This **Advanced/Leadership** level of skills and knowledge is covered in the next level of the **PeopleCert DevOps** qualification scheme provided by PeopleCert.

3. Learning Objectives

As the **Fundamentals** level course, candidates will be introduced to basic concepts, terms, principles and tools used for DevOps as well as why DevOps is needed in modern enterprises, Scrum methodology, people and culture implications as well as the practices, processes, automation and technology used for adapting DevOps within an organization.

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

- History and Need of DevOps
- Key concepts of DevOps
- Business Value of DevOps
- Concepts underlying C.A.L.M.S., The Three Ways, continuous delivery, and the scope of the full stack
- DevOps culture, transformational leadership and DevOps structure and teaming
- Evolution of the DevOps Practices and the 15 Essential Practices of DevOps
- Agile Project Management and Scrum Methodology
- Key concepts of cloud technology and virtualization, automation for deployment pipeline, and architecting for continuous delivery

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	Agile Project Management
The Urgency for DevOps	Scrum
Key DevOps Principles & Concepts	IT service Management (ITSM)
The Full Stack – People & Culture	Virtualization
The Full Stack – Processes & Practices	
The Full Stack – Technology & Automation	

4. Examination

The **PeopleCert DevOps Fundamentals** Certification Exam is designed to validate a candidate's knowledge and understanding of DevOps basics as detailed above as well as how to be able to apply this knowledge through tools and practices in a modern enterprise.

The **Fundamentals** exam focuses on the following **two** (**2**) categories in the cognitive domain of **Bloom's taxonomy**[1] which is a reference for different levels of learning:

- **Knowledge**
- **Comprehension**

4.1. Assessment Approach

The assessment approach used for the **Fundamentals** certification focuses on the **two basic** categories of Knowledge and Comprehension.

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 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

The assessment incorporates the above learning outcomes as it uses assessment objectives that cater

[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.*

for the above cognitive domain categories.

4.2. Entry Criteria/Training Requirements

For this examination, there **are no** formal entry criteria or training requirements.

In order to be eligible for the **PeopleCert DevOps Fundamentals** level examination a candidate must be able to demonstrate knowledge and understanding of basic DevOps terms, principles, processes, practices and tools and it is recommended that the candidate has received **Accredited Training** by a PeopleCert accredited training partner.

4.3. Examination Format

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

Delivery	Computer (web proctored or classroom)
Type	**40 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% (28 marks out of 40)**
Invigilator / Supervisor / Proctor	**Yes** *Physical or Online Proctoring*
Open Book	**No** *No materials are allowed in the examination room*
Prerequisites	**None**
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. A total of **fourteen to eighteen (14-18) hours** of accredited training is **recommended**.

Category	Topic	Skill Set
1.0 Introduction to DevOps	1.1 What Is DevOps	1.1.1 Defining DevOps
		1.1.2 The History of DevOps
		1.1.3 The DevOps Full Stack
2.0 The Urgency for DevOps	2.1 The Business Value Delivery Problem	2.1.1 Dimensions of Business Value
		2.1.2 External Drivers of Change
		2.1.3 The Organizational Goals of DevOps
	2.2 The IT Value Delivery Problem	2.2.1 IT & Silos
		2.2.2 The Negative Influence of Bureaucracy
		2.2.3 Complex Processes Result in Waste
		2.2.4 A Big Shift is Required
		2.2.5 IT Downward Spiral
		2.2.6 IT & DevOps
3.0 Key DevOps Principles & Concepts	3.1 C.A.L.M.S. & The Three Ways	3.1.1 C.A.L.M.S.
		3.1.2 The Three Ways
	3.2 The Deployment Pipeline	3.2.1 Continuous Delivery Across the Deployment Pipeline
	3.3 The Scope of DevOps	3.3.1 DevOps Transforms People & Culture
		3.3.2 DevOps Transforms Processes & Practices
		3.3.3 DevOps Transforms Technology & Automation
4.0 The Full Stack – People & Culture	4.1 Defining the DevOps Culture	4.1.1 Organizational Culture
		4.1.2 Importance of Collaboration
	4.2 Transformational Leadership	4.2.1 What is Transformational Leadership
		4.2.2 Transformation Leadership Principles
		4.2.3 Teams
	4.3 DevOps Structures & Teaming	4.3.1 Organizational Models for DevOps
		4.3.2 The Evolution of DevOps Teams
		4.3.3 DevOps Requires Practical Specialists
		4.3.4 A Shift to Product & Platform Teams
		4.3.5 Building a Cross-Functional Team
5.0 The Full Stack – Processes & Practices	5.1 The Evolution of DevOps Practices	5.1.1 DevOps and Other Methodologies
	5.2 The 15 Essential Practices of DevOps	5.2.1 Voice of The Customer
		5.2.2 Relationship Management
		5.2.3 Lean Process Optimization
		5.2.4 Value Stream Mapping
		5.2.5 Knowledge Management
		5.2.6 Visual Management
		5.2.7 Agile Project Management & Scrum

Category	Topic	Skill Set
		5.2.8 Shift Left Testing
		5.2.9 Change Control
		5.2.10 Service Configuration Management
		5.2.11 Release & Deployment Management
		5.2.12 Incident Management
		5.2.13 Problem Management & Kaizen
		5.2.14 Continual Improvement
		5.2.15 Antifragility
6.0 The Full Stack – Technology & Automation	6.1 Automation for the Deployment Pipeline	6.1.1 Automation & The Full Stack
		6.1.2 Deployment Pipeline Automation
		6.1.3 Tools for A DevOps Toolchain
		6.1.4 Stages of DevOps Automation
	6.2 Cloud Technology and Virtualization	6.2.1 The Emergence of Cloud Technology & Computing
		6.2.2 Cloud Computing Deployment & Service Models
	6.3 Architecting for Continuous Delivery	6.3.1 Product & Platform Teams
		6.3.2 Containerization & Microservices
		6.3.3 Application Programming Interfaces (APIs)

6. Test Specification

The **PeopleCert DevOps Fundamentals** examination will consist of **six** (**6**) sections with the following structure:

Category	Description	Exam (%)
1.0	Introduction to DevOps	5.0%
2.0	The Urgency for DevOps	10.0%
3.0	Key DevOps Principles & Concepts	15.0%
4.0	The Full Stack – People & Culture	20.0%
5.0	The Full Stack – Processes & Practices	30.0%
6.0	The Full Stack – Technology & Automation	20.0%
	Total	**100.0%**

7. Recommended Reading

1. 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

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

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

4. 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/

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

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

7. 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

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

9. 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

10. 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

11. 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/

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

13. 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

14. Conway, Melvin. "*Conway's Law*." Accessed June 25, 2019. http://www.melconway.com/Home/Conways_Law.html

15. 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).

16. 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/

17. 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

18. 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/

19. 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/

20. 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

21. 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

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

23. 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

24. 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

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

26. 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.

27. 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.

28. 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

29. 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

30. Humble, Jez, and David Farley. *“Continuous Delivery: Anatomy of the Deployment Pipeline.” informIT.* September 7, 2010. Accessed May 8, 2018. http://www.informit.com/articles/article.aspx?p=1621865&seqNum=2

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

32. Humble, Jez, n.d. *“What Is Continuous Delivery?” Continuous Delivery.* Accessed May 28,

2018. https://continuousdelivery.com/

33. 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/

34. 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

35. 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).

36. 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.

37. 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/

38. 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/

39. 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/

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

41. 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/

42. 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

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

44. 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

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

46. 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

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

48. 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/

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

50. 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

51. 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

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

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

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

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

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

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

58. 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/

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

60. 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/

61. 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/

62. 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/

63. 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/

64. 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/.

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

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

67. 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.

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

69. 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

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

71. AXELOS Limited. *ITIL® Foundation ITIL 4 Edition.* TSO, 2019.

72. "*2016 State of DevOps Report." Puppet*. Accessed May 28, 2018. https://puppet.com/resources/whitepaper/2016-state-of-devops-report

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

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

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

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

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

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

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

80. *"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

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

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

8. Glossary

Term	Definition
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 Manifesto.
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.
Business Value	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.†
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 ITIL® practice of aligning an organization's practices and services with changing business needs through the ongoing identification and alignment 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.
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 (CTQ)	According to the principles of Lean, the value items that should be prioritized and focused on.
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.‖

* 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.

Term	Definition
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.
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 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.
Incident	According to ITIL®, an unplanned interruption to a service, or a failure of a component of a service that hasn't yet impacted service.[†]
Incident Management	The ITIL®, practice of minimizing the negative impact of incidents by restoring normal service operation as quickly as possible.[†]
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.[†]
Lead Time	The time between input and output. It triggers the receipt of value.
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.
Microservice Architecture (MSA)	An architecture in which one function is associated with one service which is scaled by distributing services across nodes.
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.
Non-Functional Testing	Testing of system operation rather than specific outcomes, including performance, security, compliance or capacity testing.

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

Term	Definition
Non-Value Add Work	According to Lean, work in a process that should be removed.
Organizational Culture	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	The Scrum artifact consisting of the items completed based upon the Sprint Backlog according to the agreed 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	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 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.‖
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.
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 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)	According to ITIL®, a written agreement between an IT service provider and the business which defines key service targets and responsibilities, as well as the expected warranty and utility of a 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-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

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

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

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

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

Term	Definition
	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.
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.[ii]
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.[ii]
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.[ii]
Sprint Retrospective	The Scrum event that provides an opportunity for the Scrum Team to inspect itself and create a plan for improvements to be enacted during the next Sprint.[ii]
Sprint Review	The Scrum event which is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed.[ii]
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.
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 Values	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.

[ii] Schwaber, Ken, and Jeff Sutherland. "The Scrum Guide™." Scrum.Org and ScrumInc., 2014.

Term	Definition
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 functional requirements of a service. Utility describes those requirements of a service which are fit for purpose – does the service do what it is supposed to do?
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.

- **This page is left intentionally blank -**

PeopleCert

DevOps.