EXIN DevOps Master™

Preparation Guide

Edition 201711

DevOps

Content

1. Overview

EXIN DevOps Master™ (DEVOPSM.EN)

Scope

EXIN DevOps Master™ is a certification that validates a professional's knowledge about:

- DevOps adoption;
- Planning, requirements and design;
- Development and deployment;
- Operation and scaling;
- End-of-Life.

Summary

The word DevOps is a contraction of 'Development' and 'Operations'. DevOps is a set of best practices that emphasize the collaboration and communication of IT-professionals (developers, operators, and support staff) in the lifecycle of applications and services, leading to:

- Continuous Integration: merging all developed working copies to a shared mainline several times a day
- Continuous Deployment: release continuously or as often as possible
- Continuous Feedback: seek feedback from stakeholders during all lifecycle stages

DevOps changes how individuals think about their work; DevOps values the diversity of work done, supports intentional processes that accelerate the rate by which businesses realize value, and measures the effect of social and technical change. DevOps is a way of thinking and a way of working that enables individuals and organizations to develop and maintain sustainable work practices.

Successful DevOps is:

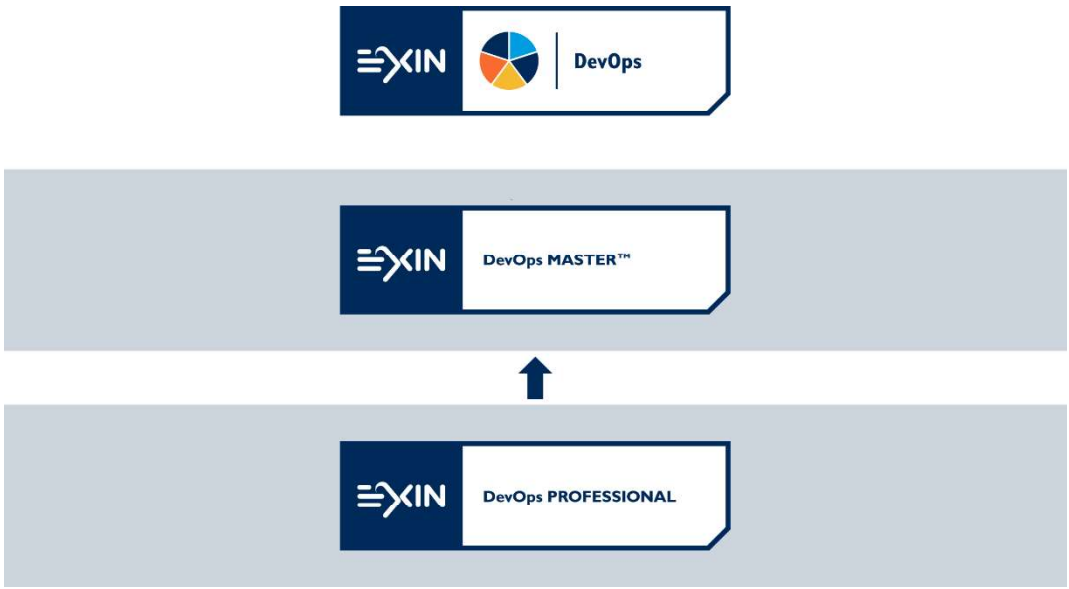
- Having a blame-free culture of sharing stories and developing empathy, enabling people and teams to practice their crafts in effective and lasting ways
- Providing applications and services for the business Just-in-Time (JiT)
- Ensuring continuity of IT services by a risk based approach of business needs
- Managing the complete lifecycle of applications and services, including end-of-life conditions

This certification focuses on adding practical skills to knowledge, enabling a DevOps Master to facilitate DevOps successfully in teams and to promote its principles in the organization.

The certification has been developed in cooperation with experts in the DevOps work field.

Context

The EXIN DevOps program:



Target group

DevOps is best known in the field of software development, but the principles are applicable in IT Service projects and other projects as well. The EXIN DevOps Master™ training and certification is aimed at professionals who are interested in facilitating DevOps practices in the organization.

The EXIN DevOps Master™ certification is meant for anyone working within a DevOps context or in an organization that considers the transition to a DevOps way of working. The target group includes: Product Owners, Agile Scrum Masters, Project Managers, Test Managers, IT Service Managers, Process Managers and Lean IT Practitioners.

Requirements for certification

- Successful completion of an EXIN Accredited EXIN DevOps Master™ training including Practical Assignments.
- Successful completion of the DevOps Master™ exam.

Examination details

Examination type:	Computer-based or paper-based multiple-choice questions
Number of questions:	50
Pass mark:	65%
Open book/notes:	No
Electronic equipment/aides permitted:	No
Time allotted for examination:	120 minutes

The Rules and Regulations for EXIN's examinations apply to this exam.

Bloom level

The EXIN Agile Scrum Master certification tests candidates at Bloom Level 3 and 4 according to Bloom's Revised Taxonomy:

- Bloom Level 3: Applying – shows that candidates have the ability to make use of information in a context different from the one in which it was learned.
  This type of questions aims to demonstrate that the candidate is able to solve problems in new situations by applying acquired knowledge, facts, techniques and rules in a different, or new way. The question usually contains a short scenario.
- Bloom level 4: Analyzing – shows that candidates have the ability to break learned information into its parts to understand it. This Bloom level is mainly tested in the Practical Assignments. The Practical Assignments aim to demonstrate that the candidate is able to examine and break information into parts by identifying motives or causes, make inferences and find evidence to support generalizations.

Training

Training is a mandatory part of the certification. The candidate is expected to have basic knowledge of DevOps principles and Lean and Agile concepts before attending the training. This knowledge can be acquired:

- through certification in EXIN DevOps Professional

Or

- by reading The Phoenix Project (see literature list).

Contact hours

The minimum number of contact hours for this training course is 24. This includes practical assignments, exam preparation and short breaks. This number of hours does not include homework, the logistics related to the exam session, the exam session and lunch breaks.

Guidelines for Practical Assignments can be found on Partnernet. The recommended number of hours for the Practical Assignments is a maximum of 8. The Practical Assignments can be completed outside of the training. For the Practical Assignments, trainers can choose between:

- The GamingWorks simulation (the trainer needs a GamingWorks accreditation)
- The game from the book: Introduction to DevOps with Chocolate, LEGO and Scrum Game[1]
- Develop their own Practical Assignments

[1] **Introduction to DevOps with Chocolate, LEGO and Scrum Game**
Dana Pylayeva
ISBN-13: 978-1-4842-2565-3
ISBN-10: 1484225643
Apress: 2017

Training length may vary with the amount of work on the practical assignments within the scope of the training. Possibilities include, but are not limited to:

- 2 days theoretical classroom training & 1 day classroom practical assignments
- 2 days theoretical e-learning training & 8 hours practical assignments in the workplace; the candidate studies through e-learning and prepares the practical assignments; the trainer assesses the work on the practical assignments
- 2 days theoretical classroom training & 8 hours practical assignments in the workplace; the trainer assesses the work on the practical assignments

Indication study load

120 hours, depending on existing knowledge. The literature matrix in chapter *4. Literature* in this Preparation Guide references the body of knowledge that is tested in the exam.

Training provider

You can find a list of our accredited training providers at www.exin.com.

2. Exam requirements

Exam requirement	Exam specification	Weight %
1. DevOps adoption		28%
	1.1 DevOps Mindset and Benefits	
	1.2 Organizational Culture	
	1.3 Principles & Concepts	
2. Planning, requirements, and design		18%
	2.1 Application or Service Lifecycle Management	
	2.2 Project Charter (Defining Scope) & Visual Control	
	2.3 Infrastructure and Architecture Design	
	2.4 Service Level Requirements and Agreements	
	2.5 Implementing a Testing Strategy	
3. Development and deployment		30%
	3.1 Continuous Delivery & Continuous Integration	
	3.2 Deployment Pipeline	
	3.3 Continuous Deployment	
	3.4 Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress and One-piece-flow	
	3.5 Automation , Tools and Testing	
4. Operation and Scaling		22%
	4.1 Managing Data; Infrastructure and Environments; and Components and Dependencies	
	4.2 Configuration Management and Version Control	
	4.3 Cloud and Immutable Infrastructure	
	4.4 Business Continuity	
	4.5 Scaling	
5. End-of-Life		2%
	5.1 Conditions for End-of-Life of a product or service	
	Total	**100%**

Exam specifications

1. **DevOps Adoption (28%)**
   - 1.1 DevOps Mindset and Benefits (10%)
     - 1.1.1 analyze DevOps anti-patterns in a scenario
     - 1.1.2 explain the benefits of DevOps
     - 1.1.3 explain why DevOps fits the current software development process so well
     - 1.1.4 explain why DevOps needs a specific mindset to work
     - 1.1.5 explain how DevOps fits with Lean and Agile Scrum practices
   - 1.2 Organizational Culture (12%)
     - 1.2.1 explain why the 4 Pillars of Effective DevOps (Collaboration, Affinity, Tools, and Scaling) are so important
     - 1.2.2 analyze a scenario for missing parts of the DevOps mindset
     - 1.2.3 explain how to create a team from a group of people, through fostering collaboration, a DevOps mindset, and empathy and trust
     - 1.2.4 analyze a situation where there is a misconception regarding collaboration and identify the correct troubleshooting method
     - 1.2.5 analyze a situation where there is a need for conflict management and identify the best solution
     - 1.2.6 explain how human resource management can foster diversity and which benefits this brings to the organization
   - 1.3 DevOps Principles and Concepts (6%)
     - 1.3.1 explain the use and usefulness of different software development methodologies (Waterfall, Agile, Scrum, etc.) and their basic principles
     - 1.3.2 explain the use and usefulness of different operations methodologies (IT Service Management)
     - 1.3.3 explain the use and usefulness of the Lean systems methodology

2. Planning, Requirements, and Design (18%)

2.1 Application or Service Lifecycle Management (4%)

- 2.1.1 explain how DevOps adds value to modern Application Lifecycle Management
- 2.1.2 explain why DevOps improves customer experience when used for Service Lifecycle Management

2.2 Project Charter and Visual Control (4%)

- 2.2.1 explain how a DevOps project's scope should be determined
- 2.2.2 explain why Visual Control over a DevOps project facilitates DevOps practices

2.3 Infrastructure and Architecture Design (4%)

- 2.3.1 explain how DevOps changes or influences the design of IT infrastructure and architecture
- 2.3.2 explain why Cloud computing and virtualization techniques make integrating Dev and Ops easier

2.4 Service Level Requirements and Agreements (2%)

- 2.4.1 explain how DevOps changes Service Level Requirements and Agreements

2.5 Implementing a Testing Strategy (4%)

- 2.5.1 explain why and how the Testing Strategy needs to be changed when transitioning to DevOps
- 2.5.2 analyze User Stories for completeness

3. Development and Deployment (30%)

3.1 Continuous Delivery and Continuous Integration (12%)

- 3.1.1 explain why Continuous Delivery is essential for Effective DevOps
- 3.1.2 analyze how to integrate Continuous Delivery in a scenario
- 3.1.3 analyze how to solve problems with Continuous Delivery in a scenario
- 3.1.4 explain why Continuous Integration is essential for Effective DevOps
- 3.1.5 analyze how to achieve Continuous Integration in a scenario with a distributed team or a distributed version control system
- 3.1.6 analyze how to solve problems with Continuous Integration in a scenario

3.2 Deployment Pipeline (4%)

3.2.1 explain the logic of the anatomy of a DevOps deployment pipeline

3.2.2 explain how to use build and deployment scripting

3.3 Continuous Deployment (4%)

3.3.1 explain why the iteration plan and the release plan should be changed for effective DevOps

3.3.2 analyze how to implement Continuous Deployment in a scenario

3.4 Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress and One-piece-flow (4%)

3.4.1 explain the concepts Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress and One-piece-flow

3.4.2 analyze a scenario for a problem with Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress or One-piece-flow and find a suitable solution

3.5 Automation, Tools and Testing (6%)

3.5.1 explain why automation is important for effective DevOps

3.5.2 explain how to use tools to facilitate DevOps in general

3.5.3 explain how to use tools to support DevOps mindset and culture

3.5.4 explain why it is important that DevOps testing is automated

3.5.5 analyze a scenario and choose the correct way of automating an acceptance test

4. Operation and Scaling (22%)

4.1 Managing Data; Infrastructure and Environments; and Components and Dependencies (10%)

4.1.1 explain which problems can be encountered when managing data in databases within DevOps

4.1.2 analyze a scenario where a database is used in DevOps and provide the best solution to a problem

4.1.3 analyze a scenario and identify the best way to prepare an infrastructure environment for deployment or manage it after deployment

4.1.4 analyze a scenario and suggest a commonly used strategy to manage components

4.1.5 explain how to manage dependencies

4.2 Configuration Management and Version control (4%)

4.2.1 explain why version control is a key to effective DevOps

4.2.2 explain how to keep version control over data, infrastructure and components

4.2.3 analyze a scenario and suggest the best strategy to manage a configuration problem

4.3 Cloud and Immutable Infrastructure (2%)

4.3.1 explain when it is and when it is not necessary to move to Cloud-based infrastructure for effective DevOps

4.3.2 explain how Cloud-based infrastructure should be managed within DevOps

4.4 Business Continuity (2%)

4.4.1 explain how DevOps can facilitate Business Continuity practices

4.5 Scaling (4%)

4.5.1 analyze a scenario, explain if and why it is important to scale up or down in that situation, and identify the best way to do that

4.5.2 analyze a scenario for what went wrong with scaling, and identify a good way to solve the problem

4.5.3 explain how social policy and hiring practices support scaling DevOps

5. End-of-Life (2%)

5.1 Conditions for End-of-Life of a product or service (2%)

5.1.1 explain which conditions should be met before terminating a service or product

3. List of Basic Concepts

This chapter contains the terms and abbreviations with which candidates should be familiar.

Please note that knowledge of these terms alone does not suffice for the exam; the candidate must understand the concepts and be able to provide examples.

A/B testing	Continuous Delivery
Acceptance Tests	Continuous Deployment
Affinity *(in DevOps)*	Continuous Integration
Agile	Cycle time
Anti-pattern	Definition of Done *(in Agile Scrum)*
Application Deployment	Dependency
Artifact Management	(Deployment) Pipeline
(Artifact) repository	Development Team
ATAM	DevOps Engineer
Automated testing	Disciplined Agile
Automation	Distributed Team
Binary files	Effective DevOps
Blamelessness	Event Management
Blue-Green deployment	Exploratory testing
Build (Management)	Flow
Build-time	Functional acceptance tests
Canary releasing	Forensic tools
Capacity testing	Gatekeeper
Change Management	Happy path
Check-in	Human error
Cloud computing	Incident Management
Collaboration *(in DevOps)*	Information radiators
Commit (stage)	Infrastructure management
Communication styles	Infrastructure Automation
Compact	INVEST
Component (tests)	Integration tests
Configuration Management	Iteration
Containers	ITSM (IT Service Management)

Ji-Kotei-Kanketsu (JKK)	Run time
Just-in-Time (JiT)	Sad path
Kaizen *(in Lean)*	Scaling *(in DevOps or Agile Scrum)*
Lean	Scrum
Libraries	Scrum Master *(in Agile Scrum)*
Light-weight ITSM	Service Continuity
Manual testing	Service Level Agreement (SLA)
Minimum Viable Product	Service Master
Monitoring strategy	Silos
Negotiation styles	Sprint
Non-functional testing	System of Engagement (SoE)
Obeya	System of Record (SoR)
Orchestration	System tests
One-piece-flow	Test-Driven Development
Operations Team	Tools
Organizational Learning	Toyota Production System (TPS)
Plan-Do-Check-Act cycle (PDCA cycle)	Unit Test
Process Master	Usability tests
(Product) Backlog	User Acceptance Testing (UAT)
Product Owner *(in Agile Scrum)*	User Story
Project charter	Value Stream Mapping (VSM)
Pull system	Velocity *(in Agile Scrum)*
Quality Assurance (QA)	(Vendor) lock-in
Regression testing	Version Control
Release Coordinator	Virtualization
Reliability Engineer	Waste *(in Lean)*
Retrospective	Waterfall
Rhythm *(in Lean)*	Work-in-Progress (WiP)
Root Cause Analysis	

4. Literature

The knowledge required for the EXIN DevOps Master™ exam is covered in the following literature.

Exam literature

A **Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale**
Jennifer Davis, Katherine Daniels
ISBN-13: 978-1491926307
ISBN-10: 1491926309
O'Reilly Media; 1 edition (June 25, 2016)

B **Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation**
Jez Humble, David Farley
ISBN-13: 978-0321601919
ISBN-10: 0321601912
Addison-Wesley Professional; 1 edition (August 6, 2010)

C **Success with Enterprise DevOps**
Koichiro (Luke) Toda, President Strategic Staff Services Corporation and Director of TPS Certificate Institution
Nobuyuki Mitsui, CTO of Strategic Staff Services Corporation
White Paper; June 2016 (download from EXIN DevOps Master™ product page)

Additional literature

The Phoenix Project is strongly recommended reading before the training.

D **The Phoenix Project**
Gene Kim, Kevin Behr, George Spafford
ISBN-10: 0988262576
ISBN-13: 978-0988262577
IT Revolution Press (January 10, 2013)

E **The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations**
Gene Kim, Jez Humble, Patrick Debois, John Willis
ISBN-10: 1942788002
ISBN-13: 978-1942788003
IT Revolution Press; 1 edition (2016)

F Other sources:
http://newrelic.com/devops
http://devops.com/

Comment

Additional literature is for reference and depth of knowledge only.

Literature matrix

Exam requirement			Literature	Size
1. DevOps Adoption				28%
	1.1	DevOps Mindset and Benefits		10%
	1.1.1	analyze DevOps anti-patterns in a scenario	A: Chapter 5 B: Chapter 1 C: Chapter 2	
	1.1.2	explain the benefits of DevOps	B: Chapter 1 C: Chapter 4	
	1.1.3	explain why DevOps fits the current software development process so well	A: Chapter 3 B: Chapter 1 C: Chapter 4	
	1.1.4	explain why DevOps needs a specific mindset to work	A: Chapter 1, 2, 4 C: Chapter 4	
	1.1.5	explain how DevOps fits with Lean and Agile Scrum practices	B: Chapter 15 C: Chapter 1, 4	
	1.2	Organizational Culture		12%
	1.2.1	explain why the 4 Pillars of Effective DevOps (Collaboration, Affinity, Tools, and Scaling) are so important	A: Chapter 6, 7	
	1.2.2	analyze a scenario for missing parts of the DevOps mindset	A: Chapter 7	
	1.2.3	explain how to create a team from a group of people, through fostering collaboration, a DevOps mindset, and empathy and trust	A: Chapter 7, 9	
	1.2.4	analyze a situation where there is a misconception regarding collaboration and identify the correct troubleshooting method	A: Chapter 7, 8, 9, 16	
	1.2.5	analyze a situation where there is a need for conflict management and identify the best solution	A: Chapter 7, 14	
	1.2.6	explain how human resource management can foster diversity and which benefits this brings to the organization	A: Chapter 7	
	1.3	DevOps Principles and Concepts		6%
	1.3.1	explain the use and usefulness of different software development methodologies (Waterfall, Agile, Scrum, etc.) and their basic principles	A: Chapter 4 B: Chapter 1 C: Chapter 5	
	1.3.2	explain the use and usefulness of different operations methodologies (IT Service Management)	C: Chapter 3, 4	
	1.3.3	explain the use and usefulness of the Lean systems methodology	A: Chapter 4	

2. Planning, Requirements, and Design				18%
	2.1	Application or Service Lifecycle Management		4%
	2.1.1	explain how DevOps adds value to modern Application Lifecycle Management	C: Chapter 1	
	2.1.2	explain why DevOps improves customer experience when used for Service Lifecycle Management	C: Chapter 4	
	2.2	Project Charter and Visual Control		4%
	2.2.1	explain how a DevOps project's scope should be determined	B: Chapter 10 C: Chapter 8	
	2.2.2	explain why Visual Control over a DevOps project facilitates DevOps practices	B: Chapter 5 C: Chapter 7	
	2.3	Infrastructure and Architecture Design		4%
	2.3.1	explain how DevOps changes or influences the design of IT infrastructure and architecture	B: Chapter 11	
	2.3.2	explain why Cloud computing and virtualization techniques make integrating Dev and Ops easier	B: Chapter 11	
	2.4	Service Level Requirements and Agreements		2%
	2.4.1	explain how DevOps changes Service Level Requirements and Agreements	B: Chapter 12 C: Chapter 4	
	2.5	Implementing a Testing Strategy		4%
	2.5.1	explain why and how the Testing Strategy needs to be changed when transitioning to DevOps	B: Chapter 4	
	2.5.2	analyze User Stories for completeness	B: Chapter 4	

3. Development and Deployment				30%
	3.1	Continuous Delivery and Continuous Integration		12%
	3.1.1	explain why Continuous Delivery is essential for Effective DevOps	B: Chapter 13, 15	
	3.1.2	analyze how to integrate Continuous Delivery in a scenario	B: Chapter 3, 5, 15	
	3.1.3	analyze how to solve problems with Continuous Delivery in a scenario	B: Chapter 15	
	3.1.4	explain why Continuous Integration is essential for Effective DevOps	B: Chapter 3	
	3.1.5	analyze how to achieve Continuous Integration in a scenario with a distributed team or a distributed version control system	B: Chapter 3	
	3.1.6	analyze how to solve problems with Continuous Integration in a scenario	B: Chapter 3	
	3.2	Deployment Pipeline		4%
	3.2.1	explain the logic of the anatomy of a DevOps deployment pipeline	B: Chapter 5, 8 C: Chapter 7	
	3.2.2	explain how to use build and deployment scripting	B: Chapter 1, 6	
	3.3	Continuous Deployment		4%
	3.3.1	explain why the iteration plan and the release plan should be changed for effective DevOps	C: Chapter 5, 7	
	3.3.2	analyze how to implement Continuous Deployment in a scenario	B: Chapter 10	
	3.4	Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress and One-piece-flow		4%
	3.4.1	explain the concepts Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress and One-piece-flow	C: Chapter 4, 7	
	3.4.2	analyze a scenario for a problem with Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress or One-piece-flow and find a suitable solution	A: Chapter 1, 2, 9 B: Chapter 1 C: Chapter 4	
	3.5	Automation, Tools and Testing		6%
	3.5.1	explain why automation is important for effective DevOps	B: Chapter 1, 8	
	3.5.2	explain how to use tools to facilitate DevOps in general	B: Chapter 8	
	3.5.3	explain how to use tools to support DevOps mindset and culture	A: Chapter 12	
	3.5.4	explain why it is important that DevOps testing is automated	A: Chapter 1 B: Chapter 3	

	3.5.5	analyze a scenario and choose the correct way of automating an acceptance test	B: Chapter 8	
4. Operation and Scaling				22%
	4.1	Managing Data; Infrastructure and Environments; and Components and Dependencies		10%
	4.1.1	explain which problems can be encountered when managing data in databases within DevOps	B: Chapter 12	
	4.1.2	analyze a scenario where a database is used in DevOps and provide the best solution to a problem	B: Chapter 12	
	4.1.3	analyze a scenario and identify the best way to prepare an infrastructure environment for deployment or manage it after deployment	B: Chapter 11	
	4.1.4	analyze a scenario and suggest a commonly used strategy to manage components	B: Chapter 13	
	4.1.5	explain how to manage dependencies	B: Chapter 13	
	4.2	Configuration Management and Version Control		4%
	4.2.1	explain why version control is a key to effective DevOps	B: Chapter 2	
	4.2.2	explain how to keep version control over data, infrastructure and components	B: Chapter 10, 11	
	4.2.3	analyze a scenario and suggest the best strategy to manage a configuration problem	B: Chapter 2	
	4.3	Cloud and Immutable Infrastructure		2%
	4.3.1	explain when it is and when it is not necessary to move to Cloud-based infrastructure for effective DevOps	A: Chapter 17 B: Chapter 11 C: Chapter 5, 7	
	4.3.2	explain how Cloud-based infrastructure should be managed within DevOps	B: Chapter 11	
	4.4	Business Continuity		2%
	4.4.1	explain how DevOps can facilitate Business Continuity practices	C: Chapter 4	
	4.5	Scaling		4%
	4.5.1	analyze a scenario, explain if and why it is important to scale up or down in that situation, and identify the best way to do that	A: Chapter 14	
	4.5.2	analyze a scenario for what went wrong with scaling, and identify a good way to solve the problem	A: Chapter 15 B: Chapter 11	
	4.5.3	explain how social policy and hiring practices support scaling DevOps	A: Chapter 6, 7, 14	

5. End-of-life				2%
	5.1	Conditions for End-of-Life of a product or service		
	5.1.1	explain which conditions should be met before terminating a service or product	C: Chapter 7	
Total				**100%**

Note: Reading literature source D The Phoenix Project, will especially benefit the understanding of the following specifications:

- *1.1*
- *1.2*
- *3.1*
- *3.3*
- *3.4*
- *4.4*

Contact EXIN

www.exin.com

YOUR ICT COMPETENCE PARTNER