



**Preparation Guide**

Editie 201901

Copyright © EXIN Holding B.V. 2019. All rights reserved.  
EXIN® is a registered trademark.  
DevOps Master™ is a registered trademark.

No part of this publication may be published, reproduced, copied or stored in a data processing system or circulated in any form by print, photo print, microfilm or any other means without written permission by EXIN.



# Inhoud

1. Overzicht	4
2. Exam requirements	8
3. Begrippenlijst	12
4. Literatuur	15

# 1. Overzicht

EXIN DevOps Master™ (DEVOPSM.NL)

## Scope

De certificering EXIN DevOps Master™ bevestigt het kennisniveau van een professional op het gebied van:

- DevOps toepassing;
- Planning, vereisten en ontwerp;
- Ontwikkeling en deployment (uitrol);
- Operations en schalen;
- Uitmaken.

## Samenvatting

De term DevOps is een samenvoeging van 'Development' en 'Operations'. DevOps bestaat uit een reeks best practices waarin de nadruk wordt gelegd op samenwerking en communicatie tussen IT-professionals (ontwikkelaars, operators en ondersteuningsteam) tijdens de levensduur van applicaties en services, die moet leiden tot:

- Continuous integration (continue integratie): meerdere keren per dag alle ontwikkelde werkende versies samenvoegen tot een gedeelde centrale lijn;
- Continuous deployment (continue uitrol): doorlopend releasen of zo vaak als mogelijk is releasen;
- Continuous feedback (continue feedback): feedback van stakeholders vragen tijdens alle fasen van de levensduur.

DevOps verandert de manier waarop mensen over hun werk denken. DevOps houdt rekening met de diversiteit van uitgevoerd werk, ondersteunt doelbewuste processen waarmee bedrijven waarde realiseren en meet het effect van sociale en technische veranderingen. DevOps is een manier van denken en werken die individuen en organisaties in staat stelt duurzame oplossingen te ontwikkelen en te onderhouden.

Succesvol DevOps wordt gekenmerkt door:

- Een niet-verwijtende cultuur van uitwisseling van ervaringen en het ontwikkelen van empathie, om mensen en teams in staat te stellen hun vaardigheden effectief en blijvend uit te oefenen;
- Het Just-in-Time (JiT) leveren van applicaties en services aan het bedrijf;
- Continuïteit van IT-services door een op risico gebaseerde benadering van wat het bedrijf nodig heeft;
- Het beheer van de volledige levensduur van applicaties en services, inclusief de voorwaarden voor het uitfasen (end-of-life).

Deze certificering richt zich op het toevoegen van praktische vaardigheden aan kennis, zodat een DevOps Master™ op succesvolle wijze DevOps kan inzetten in een team en de principes ervan kan doorvoeren in de organisatie.

De certificering is ontwikkeld in samenwerking met deskundigen op het gebied van DevOps..

## Context

Het EXIN DevOps programma:



## Doelgroep

DevOps is het meest bekend op het gebied van softwareontwikkeling, maar de principes zijn net zo goed toepasbaar in IT-serviceprojecten en projecten op andere gebieden. De training en certificering EXIN DevOps Master™ zijn bedoeld voor alle professionals die geïnteresseerd zijn in het faciliteren van DevOps-werkwijzen in hun organisatie.

De EXIN DevOps Master™-certificering is bedoeld voor iedereen die werkt binnen een DevOps-context of in een organisatie waar de overstap naar een DevOps-werkwijze wordt overwogen. Tot de doelgroep behoren: Product Owners, Agile Scrum Masters, projectmanagers, testmanagers, IT-servicemanagers, procesmanagers en Lean IT practitioners.

## Certificeringseisen

- Het succesvol afsluiten van een geaccrediteerde training EXIN DevOps Master™ met praktijkopdrachten.
- Het behalen van het examen EXIN DevOps Master™.

## Examendetails

Examenvorm:	Multiple-choicevragen
Aantal vragen:	40
Cesuur:	65%
Open boek/notities:	Nee
Elektronische hulpmiddelen toegestaan:	Nee
Examenduur	: 60 minuten

Op dit examen is het Reglement voor de examens van EXIN van toepassing.

## Bloom level

Het EXIN DevOps Master™ examen toetst kandidaten op Bloom Levels 3 en 4 volgens de Bloom's Revised Taxonomy:

- Bloom Level 3: Toepassen – laat zien dat kandidaten in staat zijn om informatie in een andere context te gebruiken dan die waarin deze is geleerd. Dit type vragen onderzoekt of de kandidaat in staat is problemen in nieuwe situaties op te lossen door verworven kennis, feiten, technieken en regels op een andere of nieuwe manier toe te passen. Deze vragen bevatten meestal een korte voorbeeldsituatie.
- Bloom Level 4: Analyseren – laat zien dat kandidaten in staat zijn geleerde informatie in stukjes op te breken om hem te begrijpen. Dit Bloom Level wordt voornamelijk getest tijdens de praktijkopdrachten. De praktijkopdrachten zijn bedoeld om te kijken of de kandidaat kan onderzoeken en informatie in delen kan opbreken door redenen of oorzaken te herkennen, zelf dingen uit de informatie af te leiden en bewijs te vinden voor generalisaties.

## Training

Het volgen van een training is een verplicht onderdeel van de certificering. Van de kandidaten wordt verwacht dat zij basiskennis hebben van DevOps-principes en Lean- en Agile-concepten voordat zij deze training beginnen. Deze kennis kan worden verworven:

- door de certificering EXIN DevOps Foundation;
- Of
- door de certificering EXIN DevOps Professional;
- Of
- door het lezen van The Phoenix Project (zie literatuurlijst).

## Contacturen

Het minimum aantal contacturen tijdens de training is 24. Dit omvat praktijkopdrachten, voorbereiding op het examen en korte pauzes. Dit aantal uren is exclusief huiswerk, logistieke voorbereiding van het examen en lunchpauzes.

Richtlijnen voor praktijkopdrachten zijn te vinden op Partnernet. Het aanbevolen aantal uren voor de praktijkopdrachten is 8. De praktijkopdrachten kunnen buiten de training worden uitgevoerd. Voor de praktijkopdrachten kunnen trainers kiezen uit:

- De GamingWorks-simulatie *The Phoenix Project* (de trainer moet hiervoor een GamingWorks-accreditatie hebben);
- Het spel uit het boek: *Introduction to DevOps with Chocolate, LEGO and Scrum Game*<sup>1</sup>;
- Praktijkopdrachten die zij zelf hebben ontwikkeld.

De duur van de training kan variëren afhankelijk van de hoeveelheid werk die is besteed aan de praktijkopdrachten. De mogelijkheden omvatten, maar zijn niet beperkt tot:

- 2 dagen theoretische klassikale training & 1 dag klassikale praktijkopdrachten.
- 2 dagen theoretische e-learningtraining & 8 uur praktijkopdrachten op de werkplek. De kandidaat bestudeert de e-learning en maakt de praktijkopdrachten. De trainer beoordeelt het werk aan de praktijkopdrachten.
- 2 dagen theoretische klassikale training & 8 uur praktijkopdrachten op de werkplek. De trainer beoordeelt het werk aan de praktijkopdrachten.

---

<sup>1</sup> **Introduction to DevOps with Chocolate, LEGO and Scrum Game**

Dana Pylayeva

ISBN-13: 978-1-4842-2565-3

ISBN-10: 1484225643

Apress: 2017

### Indicatie studielast

120 uur, afhankelijk van bestaande kennis. De literatuurmatrix in hoofdstuk 4, *Literatuur*, van deze Preparation Guide verwijst naar de kennis die tijdens het examen wordt getest.

### Training provider

Een lijst van geaccrediteerde training providers kunt u vinden op de website van EXIN [www.exin.com](http://www.exin.com).

## 2. Exameneisen

De exameneisen zijn gespecificeerd in de examenspecificaties. De onderstaande tabel bevat de onderwerpen van de module (exameneisen) en de subonderwerpen (examenspecificaties).

Exameneis	Examenspecificaties	Gewicht
1. DevOps toepassing		28%
	1.1 DevOps-mindset en voordelen	
	1.2 Organisatiecultuur	
	1.3 DevOps-principes en -concepten	
2. Planning, vereisten en ontwerp		18%
	2.1 Application- of Service Lifecycle Management	
	2.2 Projectcharter en visuele beheersing	
	2.3 Infrastructuur- en architectuurontwerp	
	2.4 Service level requirements en service level agreements	
	2.5 Een teststrategie invoeren	
3. Ontwikkeling en deployment (uitrol)		30%
	3.1 Continuous delivery (continue levering) en continuous integration (continue integratie)	
	3.2 Deployment Pipeline (uitrolpijplijn)	
	3.3 Continuous deployment (continue uitrol)	
	3.4 Ji-Kotei-Kanketsu (JKK), ritme, Werk-in-Uitvoering (WiP) en one-piece-flow	
	3.5 Automatisering, tools en testen	
4. Operations en schalen		22%
	4.1 Databeheer; infrastructuur en omgevingen; en componenten en afhankelijkheden	
	4.2 Configuration management en versiebeheer	
	4.3 Cloud en onveranderlijke infrastructuur	
	4.4 Bedrijfscontinuïteit	
	4.5 Schalen	
5. Uitmaken		2%
	5.1 Voorwaarden voor het uitmaken van een product of service	
<b>Totaal</b>		<b>100%</b>



## Examenspecificaties

### 1 DevOps toepassing (28%)

#### 1.1 DevOps-mindset en –voordelen (10%)

De kandidaat kan...

- 1.1.1 DevOps-antipatronen in een casus analyseren.
- 1.1.2 uitleggen wat de voordelen van DevOps zijn.
- 1.1.3 uitleggen waarom DevOps goed past binnen het bestaande software-ontwikkelproces.
- 1.1.4 uitleggen waarom DevOps om een bepaalde mindset vraagt om het werkend te krijgen.
- 1.1.5 uitleggen hoe DevOps past binnen de Lean- en Agile praktijk.

#### 1.2 Organisatiecultuur (12%)

De kandidaat kan...

- 1.2.1 uitleggen waarom de vier pijlers van effectief DevOps (samenwerking, affiniteit, tools en schalen) zo belangrijk zijn.
- 1.2.2 een casus analyseren op basis van missende delen uit de DevOps-mindset.
- 1.2.3 uitleggen hoe een team kan worden samengesteld uit een groep mensen, door het bevorderen van samenwerking, een DevOps-mindset, empathie en vertrouwen.
- 1.2.4 een situatie analyseren waarin een verkeerde opvatting over samenwerking leeft en de juiste probleemoplossende methode identificeren.
- 1.2.5 een situatie analyseren waarin er noodzaak is voor conflicthantering en het identificeren van de juiste oplossing.
- 1.2.6 uitleggen hoe human resource management diversiteit kan bevorderen en welke voordelen dit voor de organisatie biedt.

#### 1.3 DevOps-principes en -concepten (6%)

- 1.3.1 het gebruik en toegevoegde waarde van verschillende software-ontwikkelmethodologieën (Waterval, Agile, Scrum, etc.) en hun basisprincipes uitleggen.
- 1.3.2 het gebruik en toegevoegde waarde van verschillende methodologieën voor operations (IT-servicemanagement) uitleggen.
- 1.3.3 het gebruik en toegevoegde waarde van Lean systeemmethodologie uitleggen.

### 2 Planning, vereisten en ontwerp (18%)

#### 2.1 Application of Service Lifecycle Management (4%)

De kandidaat kan...

- 2.1.1 uitleggen hoe DevOps waarde toevoegt aan hedendaags Application Lifecycle Management.
- 2.1.2 uitleggen waarom DevOps de customer experience verbetert als het gebruikt wordt bij Service Lifecycle Management.

#### 2.2 Projectcharter en visuele beheersing (4%)

De kandidaat kan...

- 2.2.1 uitleggen hoe de scope van een DevOps-project dient te worden vastgesteld.
- 2.2.2 uitleggen waarom visuele controle in een DevOps-project de DevOps-aanpak ondersteunt.

#### 2.3 Infrastructuur- en architectuurontwerp (4%)

De kandidaat kan...

- 2.3.1 uitleggen hoe DevOps het ontwerp van de IT-infrastructuur en -architectuur verandert of beïnvloedt.
- 2.3.2 uitleggen waarom cloud computing- en virtualisatietechnieken de integratie van Dev en Ops vereenvoudigt.

- 2.4 Service level requirements en service level agreements (2%)  
De kandidaat kan...
  - 2.4.1 uitleggen hoe DevOps service level requirements en -agreements verandert.
- 2.5 Een teststrategie invoeren (4%)  
De kandidaat kan...
  - 2.5.1 uitleggen waarom en uitleggen hoe de teststrategie zou moeten veranderen wanneer men overgaat op DevOps.
  - 2.5.2 user story's analyseren op volledigheid.

### 3 Ontwikkeling en deployment (uitrol) (30%)

- 3.1 Continuous delivery (continue levering) en continuous integration (continue integratie) (12%)  
De kandidaat kan...
  - 3.1.1 uitleggen waarom continuous delivery essentieel is voor effectief DevOps.
  - 3.1.2 in een casus analyseren hoe continuous delivery geïntegreerd kan worden.
  - 3.1.3 in een casus analyseren hoe problemen met continuous delivery opgelost kunnen worden.
  - 3.1.4 uitleggen waarom continuous integration essentieel is voor effectief DevOps.
  - 3.1.5 in een casus analyseren hoe continuous integration bereikt kan worden met een gedistribueerd team of een gedistribueerd versiebeheersysteem.
  - 3.1.6 in een casus analyseren hoe problemen met continuous integration opgelost kunnen worden.
- 3.2 Deployment pipeline (uitrolpijplijn) (4%)  
De kandidaat kan...
  - 3.2.1 uitleggen wat de logica van de anatomie van een DevOps-deployment pipeline is.
  - 3.2.2 uitleggen hoe build- en uitrolscripten gebruikt moeten worden.
- 3.3 Continuous deployment (continue uitrol) (4%)  
De kandidaat kan...
  - 3.3.1 uitleggen waarom het iteratieplan en het releaseplan dienen te worden aangepast voor effectief DevOps.
  - 3.3.2 in een casus analyseren hoe continuous deployment kan worden geïmplementeerd.
- 3.4 Ji-Kotei-Kanketsu (JKK), ritme, Werk-in-Uitvoering (WiP) en one-piece-flow (4%)  
De kandidaat kan...
  - 3.4.1 uitleggen wat de concepten Ji-Kotei-Kanketsu, ritme, Werk-in-Uitvoering en one-piece-flow inhouden.
  - 3.4.2 een casus analyseren waarin een probleem met Ji-Kotei-Kanketsu, ritme, Werk-in-Uitvoering of one-piece-flow bestaat en een geschikte oplossing vinden.
- 3.5 Automatisering, tools en testen (6%)  
De kandidaat kan...
  - 3.5.1 uitleggen waarom automatisering belangrijk is voor effectief DevOps.
  - 3.5.2 uitleggen hoe tools moeten worden gebruikt om DevOps in het algemeen te ondersteunen.
  - 3.5.3 uitleggen hoe tools moeten worden gebruikt om de DevOps-mindset en -cultuur te ondersteunen.
  - 3.5.4 uitleggen waarom het belangrijk is dat testen in DevOps geautomatiseerd plaatsvindt.
  - 3.5.5 een casus analyseren en de juiste wijze van automatiseren van een acceptatietest kiezen.

#### 4 Operations en schaling (22%)

##### 4.1 Databeheer; infrastructuur en omgevingen; en componenten en afhankelijkheden (10%)

De kandidaat kan...

- 4.1.1 uitleggen welke problemen kunnen optreden wanneer data wordt beheerd in databases binnen DevOps.
- 4.1.2 een casus analyseren waarin een database wordt gebruikt in DevOps en de beste oplossing voor een probleem geven.
- 4.1.3 een casus analyseren en de beste manier van het voorbereiden van een infrastructuuromgeving voor deployment of beheer na deployment, identificeren.
- 4.1.4 een casus analyseren en een algemeen gebruikte strategie voorstellen om componenten te beheren.
- 4.1.5 uitleggen hoe afhankelijkheden moeten worden beheerd.

##### 4.2 Configuration management en versiebeheer (4%)

De kandidaat kan...

- 4.2.1 uitleggen waarom versiebeheer een sleutel is tot effectief DevOps.
- 4.2.2 uitleggen hoe versiebeheer voor data, infrastructuur en componenten wordt gevoerd.
- 4.2.3 een casus analyseren en de beste strategie voor het beheren van een probleem van configuration management voorstellen.

##### 4.3 Cloud en onveranderlijke infrastructuur (2%)

De kandidaat kan...

- 4.3.1 uitleggen wanneer het wel en wanneer het niet nodig is om gebruik te maken van infrastructuur in de cloud voor effectief DevOps.
- 4.3.2 uitleggen hoe infrastructuur in de cloud moet worden beheerd in DevOps.

##### 4.4 Bedrijfscontinuïteit (2%)

De kandidaat kan...

- 4.4.1 uitleggen hoe DevOps de bedrijfscontinuïteitspraktijken kan faciliteren.

##### 4.5 Schalen (4%)

De kandidaat kan...

- 4.5.1 een casus analyseren, uitleggen of en waarom het belangrijk is om op of af te schalen in die situatie en de beste manier identificeren om dat te doen.
- 4.5.2 een casus analyseren op wat er fout ging met schalen en het identificeren van een goede manier om het probleem op te lossen.
- 4.5.3 uitleggen hoe sociaal beleid en rekruteringspraktijken het schalen van DevOps ondersteunen.

#### 5 Uitfaseren (2%)

##### 5.1 Voorwaarden voor het uitfaseren van een product of service (2%)

De kandidaat kan...

- 5.1.1 uitleggen aan welke voorwaarden dient te zijn voldaan voordat een service of product wordt uitgefaseerd.

### 3. Begrippenlijst

Dit hoofdstuk bevat de begrippen en afkortingen die kandidaten moeten kennen.

Let op! Uitsluitend kennis van deze termen is niet voldoende voorbereiding voor het examen; de kandidaten moeten de begrippen begrijpen en in staat zijn om voorbeelden te geven.

#### Engels

A/B testing  
acceptance tests  
affinity (in DevOps)  
Agile  
anti-pattern  
application deployment  
artifact management  
(artifact) repository  
ATAM  
automated testing  
automation  
binary files  
blamelessness  
blue-green deployment  
build (management)  
build-time  
canary releasing  
capacity testing  
change management  
check-in  
Cloud Computing  
collaboration (in DevOps)  
Commit (stage)  
communication styles  
compact  
component(tests)  
configuration management  
containers  
Continuous Delivery  
Continuous Deployment  
Continuous Integration  
cycle time  
Definition of Done (in Agile Scrum)  
  
dependency  
(Deployment) Pipeline  
Development team  
DevOps Engineer  
Disciplined Agile

#### Nederlands

A/B-testen  
acceptatietests  
affiniteit (in DevOps)  
Agile  
antipatroon  
Applicatie deployment (uitrol)  
artefactbeheer  
artefactopslagplaats  
ATAM  
geautomatiseerd testen  
automatisering  
binaire bestanden  
niet-verwijten  
blue-green deployment (uitrol)  
build (management)  
build-tijd  
Kanarie releasen  
capaciteitstesten  
change management  
check-in  
cloud computing  
samenwerking (in DevOps)  
commit (fase)  
communicatiestijlen  
compact  
component(tests)  
configuration management  
containers  
continuous delivery (continue levering)  
continuous deployment (continue uitrol)  
continuous integration (continue integratie)  
cyclustijd  
Definition-of-Done (definitie van klaar) (in Agile Scrum)  
afhankelijkheid  
(deployment) pipeline (uitrolpijplijn)  
Development-team  
DevOps Engineer  
gedisciplineerde Agile

distributed Team	gedistribueerd team
effective DevOps	effectief DevOps
event management	event management
exploratory testing	verkennend testen
flow	flow
functional acceptance tests	functionele acceptatie testen
forensic tools	forensische tools
Gatekeeper	Gatekeeper
happy path	succesvol pad (happy path)
human error	menselijke fout
incident management	incident management
information radiators	informatieradiatoren
infrastructure management	infrastructure management
infrastructure automation	automatiseren van infrastructuur
INVEST	INVEST
integration tests	integratietesten
Iteration	Iteratie
ITSM (IT Service Management)	ITSM (IT-servicemanagement)
Ji-Kotei-Kanketsu (JKK)	Ji-Kotei-Kanketsu (JKK)
Just-in-Time (JiT)	Just-in-Time (JiT)
Kaizen (in Lean)	Kaizen (in Lean)
Lean	Lean
libraries	bibliotheken
light-weight ITSM	lichtgewicht ITSM
manual testing	handmatig testen
Minimum Viable Product	minimaal levensvatbaar product
monitoring strategy	monitoringsstrategie
negotiation styles	onderhandelingsstijlen
non-functional testing	niet-functionele testen
Obeya	Obeya
orchestration	orkestratie (orchestration)
one-piece-flow	one-piece-flow
Operations team	Operations-team
Organizational Learning	organisatieleren
Plan-Do-Check-Act cycle (PDCA cycle)	Plan-Do-Check-Act cyclus (PDCA cyclus)
Process Master	Process Master
(Product) Backlog	(Product) Backlog
Product Owner (in Agile Scrum)	Product Owner (in Agile Scrum)
project charter	projectcharter
pull system	pull-systeem
Quality Assurance (QA)	Kwaliteitsbewaking (QA)
regression testing	regressietesten
Release Coordinator	Release Coordinator
Reliability Engineer	Reliability Engineer
retrospective	retrospective
rhythm (in Lean)	ritme (in Lean)
Root Cause Analysis	root cause analysis
run time	looptijd

sad path	onsuccesvol pad (sad path)
scaling (of DevOps or Agile Scrum)	schaling (in DevOps of Agile Scrum)
Scrum	Scrum
Scrum Master (in Agile Scrum)	Scrum Master (in Agile Scrum)
service continuity	service continuïteit
Service level agreement (SLA)	service level agreement (SLA)
Service Master	Service Master
silos	silos
Sprint	Sprint
System of Engagement (SoE)	System of Engagement (verbindingssysteem)
System of Record (SoR)	System of Record (registratiesysteem)
system tests	systeemtesten
test-driven development	test-driven development
tools	tools
Toyota Production System (TPS)	Toyota Production System (TPS)
unit test	unittesten
usability tests	gebruikerstesten
User Acceptance Testing (UAT)	gebruikers-acceptatietesten
User Story	user story
Value Stream Mapping	value stream mapping (waardestroomschema)
velocity (in Agile Scrum)	velocity (in Agile Scrum)
(vendor) lock-in	(leveranciers)afhankelijkheid
Version Control	versiebeheer
virtualization	virtualisatie
waste ( <i>in Lean</i> )	verspilling (waste) ( <i>in Lean</i> )
Waterfall	Waterval
Work-in-Progress (WiP)	Werk in Uitvoering (WiP)

## 4. Literatuur

### Examenliteratuur

De benodigde kennis voor het EXIN DevOps Master™-examen wordt besproken in de volgende literatuur.

- 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; 1e editie (25 juni 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; 1e editie (6 augustus 2010)
  
- C**     **Success with Enterprise DevOps**  
Koichiro (Luke) Toda, President Strategic Staff Services Corporation and Director of TPS Certificate Institution  
Nobuyuki Mitsui, CTO van Strategic Staff Services Corporation  
White Paper; Juni 2016 (te downloaden op EXIN DevOps Master™ productpagina)

### Aanvullende literatuur

*Het wordt ten zeerste aangeraden om voorafgaand aan de training The Phoenix Project te lezen.*

- D**     **The Phoenix Project**  
Gene Kim, Kevin Behr, George Spafford  
ISBN-10: 0988262576  
ISBN-13: 978-0988262577  
IT Revolution Press (10 januari 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; 1e editie (2016)
  
- F**     Andere bronnen:  
<http://newrelic.com/devops>  
<http://devops.com/>

### Toelichting

De aanvullende literatuur dient alleen ter referentie en het verdiepen van kennis.

## Literatuurmatrix

Examen-eis	Examenspecificatie	Literatuur	Gewicht
<b>1. DevOps toepassing</b>			28%
	1.1 DevOps-mindset en –voordelen		10%
	1.1.1 DevOps-antipatronen in een casus analyseren	A: Hoofdstuk 5 B: Hoofdstuk 1 C: Hoofdstuk 2	
	1.1.2 uitleggen wat de voordelen van DevOps zijn	B: Hoofdstuk 1 C: Hoofdstuk 4	
	1.1.3 uitleggen waarom DevOps goed past binnen het bestaande software-ontwikkelproces	A: Hoofdstuk 3 B: Hoofdstuk 1 C: Hoofdstuk 4	
	1.1.4 uitleggen waarom DevOps om een bepaalde mindset vraagt om het werkend te krijgen	A: Hoofdstuk 1, 2, 4 C: Hoofdstuk 4	
	1.1.5 uitleggen hoe DevOps past binnen de Lean- en Agilepraktijk	B: Hoofdstuk 15 C: Hoofdstuk 1, 4	
	2.1 Organisatiecultuur		12%
	2.1.1 uitleggen waarom de vier pijlers van effectief DevOps (samenwerking, affiniteit, tools en schalen) zo belangrijk zijn	A: Hoofdstuk 6, 7	
	2.1.2 een casus analyseren op basis van missende delen uit de DevOps-mindset	A: Hoofdstuk 7	
	2.1.3 uitleggen hoe een team kan worden samengesteld uit een groep mensen, door het bevorderen van samenwerking, een DevOps-mindset, empathie en vertrouwen	A: Hoofdstuk 7, 9	
	2.1.4 een situatie analyseren waarin een verkeerde opvatting over samenwerking leeft en de juiste probleemoplossende methode identificeren	A: Hoofdstuk 7, 8, 9, 16	
	2.1.5 een situatie analyseren waarin er noodzaak is voor conflicthantering en het identificeren van de juiste oplossing	A: Hoofdstuk 7, 14	
	2.1.6 uitleggen hoe human resource management diversiteit kan bevorderen en welke voordelen dit voor de organisatie biedt	A: Hoofdstuk 7	
	1.3 DevOps-principes en -concepten		6%
	1.3.1 het gebruik en toegevoegde waarde van verschillende software-ontwikkelmethodologieën (Waterval, Agile, Scrum, etc.) en hun basisprincipes uitleggen	A: Hoofdstuk 4 B: Hoofdstuk 1 C: Hoofdstuk 5	
	1.3.2 het gebruik en toegevoegde waarde van verschillende methodologieën voor operations (IT-servicemanagement) uitleggen	C: Hoofdstuk 3, 4	
	1.3.3 het gebruik en toegevoegde waarde van Lean systeemmethodologie uitleggen	A: Hoofdstuk 4	



<b>2. Planning, vereisten en ontwerp</b>			18%
	2.1 Application of Service Lifecycle Management		4%
	2.1.1 uitleggen hoe DevOps waarde toevoegt aan hedendaags Application Lifecycle Management	C: Hoofdstuk 1	
	2.1.2 uitleggen waarom DevOps de customer experience verbetert als het gebruikt wordt bij Service Lifecycle Management	C: Hoofdstuk 4	
	2.2 Projectcharter en visuele beheersing		4%
	2.2.1 uitleggen hoe de scope van een DevOps-project dient te worden vastgesteld	B: Hoofdstuk 10 C: Hoofdstuk 8	
	2.2.2 uitleggen waarom visuele controle in een DevOps-project de DevOps-aanpak ondersteunt	B: Hoofdstuk 5 C: Hoofdstuk 7	
	2.3 Infrastructuur- en architectuurontwerp		4%
	2.3.1 uitleggen hoe DevOps het ontwerp van de IT-infrastructuur en -architectuur verandert of beïnvloedt	B: Hoofdstuk 11	
	2.3.2 uitleggen waarom cloud computing- en virtualisatietechnieken de integratie van Dev en Ops vereenvoudigt	B: Hoofdstuk 11	
	2.4 Service level requirements en service level agreements		2%
	2.4.1 uitleggen hoe DevOps service level requirements en -agreements verandert	B: Hoofdstuk 12 C: Hoofdstuk 4	
	2.5 Een teststrategie invoeren		4%
	2.5.1 uitleggen waarom en uitleggen hoe de teststrategie zou moeten veranderen wanneer men overgaat op DevOps	B: Hoofdstuk 4	
	2.5.2 user story's analyseren op volledigheid	B: Hoofdstuk 4	
<b>3. Ontwikkeling en deployment (uitrol)</b>			30%
	3.1 Continuous delivery (continue levering) en continuous integration (continue integratie)		12%
	3.1.1 uitleggen waarom continuous delivery essentieel is voor effectief DevOps	B: Hoofdstuk 13, 15	
	3.1.2 in een casus analyseren hoe continuous delivery geïntegreerd kan worden	B: Hoofdstuk 3, 5, 15	
	3.1.3 in een casus analyseren hoe problemen met continuous delivery opgelost kunnen worden	B: Hoofdstuk 15	
	3.1.4 uitleggen waarom continuous integration essentieel is voor effectief DevOps	B: Hoofdstuk 3	
	3.1.5 in een casus analyseren hoe continuous integration bereikt kan worden met een gedistribueerd team of een gedistribueerd versiebeheersysteem	B: Hoofdstuk 3	
	3.1.6 in een casus analyseren hoe problemen met continuous integration opgelost kunnen worden	B: Hoofdstuk 3	
	3.2 Deployment pipeline (uitrolpijplijn)		4%
	3.2.1 uitleggen wat de logica van de anatomie van een DevOps-deployment pipeline is	B: Hoofdstuk 5, 8 C: Hoofdstuk 7	
	3.2.2 uitleggen hoe build- en uitrolscripten gebruikt moeten worden	B: Hoofdstuk 1, 6	

	3.3 Continuous deployment (continue uitrol)		4%
	3.3.1 uitleggen waarom het iteratieplan en het releaseplan dienen te worden aangepast voor effectief DevOps	C: Hoofdstuk 5, 7	
	3.3.2 in een casus analyseren hoe continuous deployment kan worden geïmplementeerd	B: Hoofdstuk 10	
	3.4 Ji-Kotei-Kanketsu (JKK), ritme, Werk-in-Uitvoering (WiP) en one-piece-flow		4%
	3.4.1 uitleggen wat de concepten Ji-Kotei-Kanketsu, ritme, Werk-in-Uitvoering en one-piece-flow inhouden	C: Hoofdstuk 4, 7	
	3.4.2 een casus analyseren waarin een probleem met Ji-Kotei-Kanketsu, ritme, Werk-in-Uitvoering of one-piece-flow bestaat en een geschikte oplossing vinden	A: Hoofdstuk 1, 2, 9 B: Hoofdstuk 1 C: Hoofdstuk 4	
	3.5 Automatisering, tools en testen		6%
	3.5.1 uitleggen waarom automatisering belangrijk is voor effectief DevOps	B: Hoofdstuk 1, 8	
	3.5.2 uitleggen hoe tools moeten worden gebruikt om DevOps in het algemeen te ondersteunen	B: Hoofdstuk 8	
	3.5.3 uitleggen hoe tools moeten worden gebruikt om de DevOps-mindset en -cultuur te ondersteunen	A: Hoofdstuk 12	
	3.5.4 uitleggen waarom het belangrijk is dat testen in DevOps geautomatiseerd plaatsvindt	A: Hoofdstuk 1 B: Hoofdstuk 3	
	3.5.5 een casus analyseren en de juiste wijze van automatiseren van een acceptatietest kiezen	B: Hoofdstuk 8	
<b>4. Operations en schalen</b>			22%
	4.1 Databeheer; infrastructuur en omgevingen; en componenten en afhankelijkheden		10%
	4.1.1 uitleggen welke problemen kunnen optreden wanneer data wordt beheerd in databases binnen DevOps	B: Hoofdstuk 12	
	4.1.2 een casus analyseren waarin een database wordt gebruikt in DevOps en de beste oplossing voor een probleem geven	B: Hoofdstuk 12	
	4.1.3 een casus analyseren en de beste manier van het voorbereiden van een infrastructuuromgeving voor deployment of beheer na deployment	B: Hoofdstuk 11	
	4.1.4 een casus analyseren en een algemeen gebruikte strategie voorstellen om componenten te beheren	B: Hoofdstuk 13	
	4.1.5 uitleggen hoe afhankelijkheden moeten worden beheerd	B: Hoofdstuk 13	
	4.2 Configuration management en versiebeheer		4%
	4.2.1 uitleggen waarom versiebeheer een sleutel is tot effectief DevOps	B: Hoofdstuk 2	
	4.2.2 uitleggen hoe versiebeheer voor data, infrastructuur en componenten wordt gevoerd	B: Hoofdstuk 10, 11	
	4.2.3 een casus analyseren en de beste strategie voor het beheren van een probleem van configuration management vinden	B: Hoofdstuk 2	

	4.3 Cloud en onveranderlijke infrastructuur		2%
	4.3.1 uitleggen wanneer het wel en wanneer het niet nodig is om gebruik te maken van infrastructuur in de cloud in te zetten voor effectief DevOps	A: Hoofdstuk 17 B: Hoofdstuk 11 C: Hoofdstuk 5, 7	
	4.3.2 uitleggen hoe infrastructuur in de cloud moet worden beheerd binnen DevOps	B: Hoofdstuk 11	
	4.4 Bedrijfscontinuïteit		2%
	4.4.1 uitleggen hoe DevOps de bedrijfscontinuïteitspraktijken kan faciliteren	C: Hoofdstuk 4	
	4.5 Schalen		4%
	4.5.1 een casus analyseren, uitleggen of en waarom het belangrijk is om op of af te schalen in die situatie en de beste manier identificeren om dat te doen	A: Hoofdstuk 14	
	4.5.2 een casus analyseren op wat er fout ging met schalen en het identificeren van een goede manier om het probleem op te lossen	A: Hoofdstuk 15 B: Hoofdstuk 11	
	4.5.3 uitleggen hoe sociaal beleid en rekruteringspraktijken het schalen van DevOps ondersteunen	A: Hoofdstuk 6, 7, 14	
<b>5. Uutfaseren</b>			2%
	5.1 Voorwaarden voor het uutfaseren van een product of service		
	5.1.1 uitleggen aan welke voorwaarden dient te zijn voldaan voordat een service of product wordt uitgefaseerd	C: Hoofdstuk 7	
<b>Total</b>			<b>100%</b>

Opmerking: Literatuurbron D, *The Phoenix Project*, biedt de nodige informatie voor een beter begrip van met name de volgende specificaties:

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

# Contact EXIN

[www.exin.com](http://www.exin.com)

