



**Guia de Preparação**

Edição 201810

Copyright © EXIN Holding B.V. 2018. 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.



# Conteúdo

1. Visão geral	4
2. Requisitos do exame	8
3. Lista de conceitos básicos	12
4. Literatura do exame	15

# 1. Visão geral

EXIN DevOps Master™ (DEVOPSM.PR)

## Escopo

A certificação EXIN DevOps Master™ valida o conhecimento de um profissional sobre:

- Adoção do DevOps;
- Planejamento, requisitos e desenho;
- Desenvolvimento e implantação;
- Operação e Dimensionamento;
- Fim da vida.

## Resumo

A palavra DevOps é a abreviação de 'Desenvolvimento' e 'Operações'. DevOps é um conjunto de melhores práticas que enfatizam a colaboração e a comunicação de profissionais de TI (desenvolvedores, operadores e pessoal de suporte) no ciclo de vida de aplicativos e serviços, o que leva a:

- Integração Contínua: mesclar todas as cópias de trabalho desenvolvidas para uma linha principal compartilhada várias vezes ao dia.
- Implantação Contínua: liberação contínua ou tão frequente quanto possível.
- Feedback Contínuo: buscar feedback das partes interessadas durante todas as fases do ciclo de vida.

DevOps muda o modo de pensar dos indivíduos em relação ao seu trabalho; DevOps valoriza a diversidade do trabalho realizado, apoia processos intencionais que aceleram a taxa por meio da qual as empresas percebem o valor e mede o efeito das mudanças nos âmbitos social e técnico. DevOps é uma forma de pensar e uma forma de trabalhar que permite que indivíduos e organizações desenvolvam e mantenham práticas de trabalho sustentáveis.

O sucesso para DevOps é:

- Ter uma cultura que compartilha, sem culpa, histórias e desenvolve a empatia, permitindo que pessoas e equipes coloquem em prática suas habilidades de maneira eficaz e de maneira duradoura.
- Fornecer aplicativos e serviços para o negócio de forma Just-in-Time (JiT).
- Assegurar a continuidade dos serviços de TI por meio de uma abordagem das necessidades do negócio com base nos riscos.
- Gerenciar todo o ciclo de vida de aplicativos e serviços, incluindo condições para o final do ciclo de vida.

O foco da presente certificação é aumentar habilidades práticas ao conhecimento, permitindo que um DevOps Master™ aumente as facilidades de sucesso do DevOps em uma equipe e além disso, consiga promover seus princípios na organização.

Esta certificação foi desenvolvida em cooperação com especialistas no campo de trabalho de DevOps.

## Contexto

O programa EXIN DevOps:



## Grupo alvo

O DevOps é mais conhecido na área de desenvolvimento de software, mas os princípios são aplicáveis a projetos de Serviços de TI e também a outros projetos. O treinamento e certificação EXIN DevOps Master™ são destinados a profissionais interessados em facilitar práticas de DevOps na organização.

A certificação EXIN DevOps Master™ destina-se a qualquer pessoa trabalhando em um contexto de DevOps ou em uma organização que considere a transição para uma forma de trabalho DevOps. O grupo-alvo inclui: Proprietários de Produto, Agile Scrum Master, Gerentes de Projetos, Gerentes de Testes, Gerentes de Serviços de TI, Gerentes de Processo e Praticantes de Lean IT.

## Requisitos para a certificação

1. Conclusão bem-sucedida de um treinamento EXIN DevOps Master™ certificado pela EXIN, incluindo Tarefas Práticas.
2. Realização bem-sucedida do exame DevOps Master™.

## Detalhes do exame

Tipo de exame:	Perguntas de múltipla escolha
Número de perguntas:	50
Nota de aprovação:	65%
Com consulta/observações:	Não
Equipamentos eletrônicos/acessórios permitidos	Não
Tempo designado para o exame:	120 minutos

As Regras e Regulamentos dos exames EXIN aplicam-se a este exame.

## Nível Bloom

A certificação EXIN DevOps Master™ testa os candidatos nos Níveis Bloom 3 e 4 de acordo com a Taxonomia Revisada de Bloom:

- **Nível Bloom 3: Aplicação** – mostra que os candidatos têm a capacidade de utilizar as informações em um contexto diferente daquele em que elas foram aprendidas. Este tipo de pergunta pretende demonstrar que o candidato é capaz de resolver problemas em novas situações, aplicando o conhecimento adquirido, fatos, técnicas e regras de um modo novo ou diferente. A pergunta geralmente contém um breve cenário.
- **Nível Bloom 4: Análise** – mostra que os candidatos têm a capacidade de decompor as informações aprendidas em suas partes para compreendê-las. Este nível Bloom é testado principalmente nos exercícios práticos. Os exercícios práticos têm o objetivo de demonstrar que o candidato é capaz de examinar e decompor a informação em partes, identificando motivos ou causas, fazer inferências e encontrar evidências para respaldo de generalizações.

## Treinamento

O treinamento é uma parte obrigatória da certificação. Os candidatos deverão ter conhecimentos básicos de princípios DevOps e de conceitos Lean e Agile antes de participar do treinamento. Este conhecimento pode ser adquirido:

- Através da certificação em EXIN DevOps Professional.
- Ou
- lendo The Phoenix Project (consulte a lista de literatura).

## Horas de contato

O número mínimo de horas de contato para esse treinamento é de 24 horas. Isso inclui atribuições práticas, preparação para o exame e paradas curtas (breaks). Este número de horas não inclui tarefas para casa, a logística (preparação) relacionada à sessão do exame, a sessão do exame e intervalos de almoço.

Diretrizes para atribuições práticas podem ser encontradas na Partnet. O número de horas recomendado para as atribuições práticas é de 8, no máximo. As atribuições práticas podem ser realizadas fora do treinamento. Para as atribuições práticas, os instrutores podem escolher entre:

- A simulação GamingWorks (o instrutor precisa de um credenciamento em GamingWorks)
- O jogo do livro: Introduction to DevOps with Chocolate, LEGO and Scrum Game (Introdução a DevOps com Chocolate, LEGO e Jogo Scrum)<sup>1</sup>
- Desenvolver suas próprias atribuições práticas

A duração do treinamento pode variar com a quantidade de trabalho nas atribuições práticas dentro do escopo do treinamento. Essas possibilidades incluem, mas não se limitam a:

- 2 dias de treinamento teórico em sala de aula e 1 dia de atribuições práticas em sala de aula
- 2 dias de treinamento teórico em e-learning e 8 horas de atribuições práticas no local de trabalho; o candidato estuda por meio de e-learning e prepara as atribuições práticas; o instrutor avalia o trabalho nas atribuições práticas
- 2 dias de treinamento teórico em sala de aula e 8 horas de atribuições práticas no local de trabalho; o instrutor avalia o trabalho nas atribuições práticas

### Carga de estudos indicada

120 horas, dependendo do conhecimento existente. A matriz de literatura no capítulo 4. A literatura neste Guia de Preparação faz referência ao conjunto de conhecimento que é testado no exame.

### Provedores de Treinamentos

Você encontrará uma lista de nossos provedores de treinamento credenciados em [www.exin.com](http://www.exin.com).

---

#### <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



## 2. Requisitos do exame

Os requisitos do exame são definidos nas especificações do exame. A tabela a seguir lista os tópicos do módulo (requisitos do exame) e subtópicos (especificações do exame).

Requisitos do exame	Especificações do exame	Peso
<b>1. Adoção do DevOps</b>		<b>28%</b>
	1.1 Mentalidade (Mindset) DevOps e seus Benefícios	10%
	1.2 Cultura Organizacional	12%
	1.3 Princípios e Conceitos DevOps	6%
<b>2. Planning, requirements, and design</b>		<b>18%</b>
	2.1 Gerenciamento do Ciclo de Vida de Aplicativos ou Serviços	4%
	2.2 Termo de Abertura do Projeto (Definição de escopo) e Controle Visual	4%
	2.3 Desenho da Infraestrutura e Arquitetura	4%
	2.4 Requisitos e acordos de nível de serviço	2%
	2.5 Implementando uma Estratégia de Testes	4%
<b>3. Development and deployment</b>		<b>30%</b>
	3.1 Entrega Contínua e Integração Contínua	12%
	3.2 Pipeline de Implantação	4%
	3.3 Implantação contínua	4%
	3.4 Ji-Kotei-Kanketsu, Ritmo, Trabalho em Andamento e Fluxo Único (Fluxo Contínuo)	4%
	3.5 Automação, Ferramentas e Testes	6%
<b>4. Operation and Scaling</b>		<b>22%</b>
	4.1 Gerenciamento de Dados; Infraestrutura e Ambientes; e Componentes e Dependências	10%
	4.2 Gerenciamento de Configuração e Controle de Versão	4%
	4.3 Infraestrutura em Nuvens e Imutável	2%
	4.4 Continuidade do Negócio	2%
	4.5 Escala	4%
<b>5. End-of-Life</b>		<b>2%</b>
	5.1 Condições de Fim de Vida de um produto ou serviço	2%
<b>Total</b>		<b>100%</b>



## Exam specifications

### 1 Adoção do DevOps

#### 1.1 Mentalidade (Mindset) DevOps e seus Benefícios

O candidato é capaz de...

- 1.1.1 analisar anti-padrões DevOps em um cenário.
- 1.1.2 explicar os benefícios do DevOps.
- 1.1.3 explicar por que o DevOps se adequa tão bem ao atual processo de desenvolvimento de software.
- 1.1.4 explicar por que DevOps precisa de um mentalidade específico para trabalhar.
- 1.1.5 explicar como o DevOps se encaixa com as práticas Lean e Agile Scrum.

#### 1.2 Cultura Organizacional

O candidato é capaz de...

- 1.2.1 explicar por que os 4 pilares de DevOps Eficaz (Colaboração, Afinidade, Ferramentas e Dimensionamento) são tão importantes.
- 1.2.2 analisar um cenário com partes faltantes da mentalidade DevOps.
- 1.2.3 explicar como criar uma equipe com um grupo de pessoas, focando fomentar a colaboração, a mentalidade DevOps, a empatia e a confiança.
- 1.2.4 analisar uma situação que tenha um equívoco sobre a colaboração e identificar o método correto de solução do problema.
- 1.2.5 analisar uma situação em que há necessidade de gestão de conflitos e identificar a melhor solução.
- 1.2.6 explicar como o gerenciamento de recursos humanos pode fomentar a diversidade e quais benefícios isso traz para a organização.

#### 1.3 Princípios e Conceitos DevOps

O candidato é capaz de...

- 1.3.1 explicar o uso e a utilidade de diferentes metodologias de desenvolvimento de software (Cascata (Waterfall), Agile, Scrum) e seus princípios básicos.
- 1.3.2 explicar o uso e utilidade de diferentes metodologias de operações (Gerenciamento de Serviços de TI).
- 1.3.3 explicar o uso e a utilidade da metodologia de sistemas Lean.

### 2 Planejamento, Requisitos e Desenho

#### 2.1 Gerenciamento do Ciclo de Vida de Aplicativos ou Serviços

O candidato é capaz de...

- 2.1.1 explicar como o DevOps agrega valor ao Gerenciamento do Ciclo de Vida do Aplicativo moderno.
- 2.1.2 explicar por que a DevOps melhora a experiência do cliente quando usada para o Gerenciamento do Ciclo de Vida do Serviço.

#### 2.2 Termo de Abertura do Projeto (Definição de escopo) e Controle Visual

O candidato é capaz de...

- 2.2.1 explicar como o escopo do projeto DevOps deve ser determinado.
- 2.2.2 explicar por que o Controle Visual em um projeto DevOps facilita as práticas DevOps.

#### 2.3 Desenho da Infraestrutura e Arquitetura

O candidato é capaz de...

- 2.3.1 explicar como o DevOps muda ou influencia o projeto de infraestrutura e arquitetura de TI.
- 2.3.2 explicar por que a Computação em nuvem e as técnicas de virtualização tornam a integração de Dev e Ops mais fácil.

#### 2.4 Requisitos e acordos de nível de serviço

O candidato é capaz de...

- 2.4.1 explicar como o DevOps altera os requisitos e os acordos de nível de serviço.

- 2.5 Implementando uma Estratégia de Testes
  - O candidato é capaz de...
  - 2.5.1 explicar por que e como a Estratégia de Teste precisa ser alterada ao fazer a transição para o DevOps.
  - 2.5.2 analisar Histórias de Usuário para completude.

### 3 Desenvolvimento e implantação

- 3.1 Entrega Contínua e Integração Contínua
  - O candidato é capaz de...
  - 3.1.1 explicar por que a Entrega Contínua é essencial para DevOps eficaz.
  - 3.1.2 analisar como integrar a Entrega Contínua em um cenário.
  - 3.1.3 analisar como resolver problemas com a Entrega Contínua em um cenário.
  - 3.1.4 explicar por que a Integração Contínua é essencial para DevOps eficaz.
  - 3.1.5 analisar como alcançar a Integração Contínua em um cenário com uma equipe distribuída ou um sistema de controle de versão distribuído.
  - 3.1.6 analisar como resolver problemas com Integração Contínua em um cenário.
- 3.2 Pipeline de implantação
  - O candidato é capaz de...
  - 3.2.1 explicar a lógica da anatomia de um pipeline de implantação DevOps.
  - 3.2.2 explicar como usar scripts de criação e implantação.
- 3.3 Implantação contínua
  - O candidato é capaz de...
  - 3.3.1 explicar por que o plano de iteração e o plano de liberação devem ser alterados para um DevOps eficaz.
  - 3.3.2 analisar como implementar a implantação contínua em um cenário.
- 3.4 Ji-Kotei-Kanketsu, Ritmo, Trabalho em Andamento e Fluxo Único (Fluxo Contínuo) (4%)
  - O candidato é capaz de...
  - 3.4.1 explicar os conceitos Ji-Kotei-Kanketsu, Ritmo, Trabalho em Andamento e Fluxo Único (Fluxo Contínuo).
  - 3.4.2 analisar um cenário para um problema com Ji-Kotei-Kanketsu, Ritmo, Trabalho em andamento ou Fluxo Único e encontrar uma solução adequada.
- 3.5 Automação, Ferramentas e Testes
  - O candidato é capaz de...
  - 3.5.1 explicar por que a automação é importante para o DevOps eficaz.
  - 3.5.2 explicar como usar ferramentas para facilitar DevOps em geral.
  - 3.5.3 explicar como usar ferramentas para dar suporte à mentalidade e cultura do DevOps.
  - 3.5.4 explicar por que é importante que o teste de DevOps seja automatizado.
  - 3.5.5 analisar um cenário e escolher a maneira correta de automatizar um teste de aceitação.

### 4 Operação e Dimensionamento

- 4.1 Gerenciamento de Dados; Infraestrutura e Ambientes; e Componentes e Dependências
  - O candidato é capaz de...
  - 4.1.1 explicar quais problemas podem ser encontrados ao gerenciar dados em bancos de dados dentro do DevOps.
  - 4.1.2 analisar um cenário onde um banco de dados é usado em DevOps e fornecer a melhor solução para um problema.
  - 4.1.3 analisar um cenário e identificar a melhor maneira de preparar um ambiente de infraestrutura para implantação ou gerenciá-lo após a implantação.
  - 4.1.4 analisar um cenário e sugerir uma estratégia comumente usada para gerenciar componentes.
  - 4.1.5 explicar como gerenciar dependências.

- 4.2 Gerenciamento de Configuração e Controle de Versão  
O candidato é capaz de...
  - 4.2.1 explicar por que o controle de versão é uma chave para o DevOps eficaz.
  - 4.2.2 explicar como manter o controle de versão sobre dados, infraestrutura e componentes.
  - 4.2.3 analisar um cenário e sugerir a melhor estratégia para gerenciar um problema de configuração.
- 4.3 Infraestrutura em Nuvens e Imutável  
O candidato é capaz de...
  - 4.3.1 explicar quando é e quando não é necessário mover para a infraestrutura baseada em nuvem para ter um DevOps eficaz.
  - 4.3.2 explicar como a infraestrutura baseada em nuvem deve ser gerenciada dentro do DevOps.
- 4.4 Continuidade do Negócio  
O candidato é capaz de...
  - 4.4.1 explicar como o DevOps pode facilitar práticas de continuidade de negócios.
- 4.5 Dimensionamento  
O candidato é capaz de...
  - 4.5.1 analisar um cenário, explicar se e por que é importante dimensionar para cima ou para baixo nessa situação, e identificar a melhor maneira de fazer isso.
  - 4.5.2 analisar um cenário que deu errada a dimensionamento, e identificar uma boa maneira de resolver o problema.
  - 4.5.3 explicar como a política social e práticas de contratação suportam escalonamento DevOps.

## 5 Fim da vida

- 5.1 Condições de Fim de Vida de um produto ou serviço  
O candidato é capaz de...
  - 5.1.1 expliquem que condições devem ser cumpridas antes de terminar um serviço ou produto.

### 3. Lista de conceitos básicos

Este capítulo contém os termos com que os candidatos devem se familiarizar.

Por favor, note que o conhecimento destes termos de maneira independente não é suficiente para o exame; O candidato deve compreender os conceitos e estar apto a fornecer exemplos.

English	Brazilian Portuguese
A/B testing	Testes A/B
Acceptance Tests	Testes de Aceite
Affinity (in DevOps)	Afinidade (em DevOps)
Agile	Agile (Ágil)
Anti-pattern	Fora do padrão
Application Deployment	Implementação de aplicativos
Artifact Management	Gerenciamento de artefatos
(Artifact) repository	(Artefato) repositório
ATAM	ATAM
Automated testing	Testes Automatizados
Automation	Automação
Binary files	Arquivos binários
Blamelessness	Cultura livre de culpa
Blue-Green deployment	Desenvolvimento Azul-Verde
Build (Management)	(Gerenciamento de) Construção
Build-time	Tempo de construção
Canary releasing	Implantação canário
Capacity testing	Teste de capacidade
Change Management	Gerenciamento de Mudanças
Check-in	Check-In
Cloud Computing	Computação em nuvem (Cloud)
Collaboration (in DevOps)	Colaboração (em DevOps)
Commit (stage)	Commit (estágio)
Communication styles	Estilos de comunicação
Compact	Compacto (Compact)
Component (tests)	Componentes (testes)
Configuration Management	Gerenciamento de Configuração
Containers	Containers
Continuous Delivery	Entrega Contínua
Continuous Deployment	Implantação contínua
Continuous Integration	Integração Contínua
Cycle time	Ciclos de tempo
Definition of Done (in Agile Scrum)	Definição de pronto (em Agile Scrum)
Dependency	Dependência
(Deployment) Pipeline	Pipeline (de Implantação)

Development Team	Time de Desenvolvimento
DevOps Engineer	Engenheiro de DevOps
Disciplined Agile	Agile Controlado
Distributed Team	Time Distribuído
Effective DevOps	DevOps Eficiente
Event Management	Gerenciamento de Eventos
Exploratory testing	Testes Exploratórios
Flow	Fluxo
Functional acceptance tests	Testes de Aceite Funcional
Forensic tools	Ferramentas de Investigação
Gatekeeper	Ponto de Controle
Happy path	Melhor caminho
Human error	Erro Humano
Incident Management	Gerenciamento de Incidentes
Information radiators	Painel de Informações
Infrastructure Management	Gerenciamento de Infraestrutura
Infrastructure Automation	Automação de Infraestrutura
INVEST	INVEST
Integration tests	Testes integrados
Iteration	Iteração
ITSM (IT Service Management)	Gerenciamento de Serviços de TI (ITSM)
Ji-Kotei-Kanketsu (JKK)	Ji-Kotei-Kanketsu (JKK)
Just-in-Time (JiT)	Just-in-Time (JiT)
Kaizen (in Lean)	Kaizen (em Lean)
Lean	Lean
Libraries	Bibliotecas
Light-weight ITSM	ITSM leve
Manual testing	Testes manuais
Minimum Viable Product	Produto com mínima viabilidade
Monitoring strategy	Estratégia de monitoração
Negotiation styles	Estilos de negociação
Non-functional testing	Testes não-funcionais
Obeya	Sala de Guerra (Obeya)
Orchestration	Orquestração
One-piece-flow	Fluxo Único (one-piece-flow)
Operations Team	Time de Operações
Organizational Learning	Aprendizado organizacional
Plan-Do-Check-Act cycle (PDCA cycle)	Planejar-Fazer-Checar-Atuar (Ciclo PDCA)
Process Master	Master de Processo
(Product) Backlog	Backlog
Product Owner (in Agile Scrum)	Product Owner (em Agile Scrum)
Project charter	Capítulo de Projeto
Pull system	Sistema de Pull

Quality Assurance (QA)	Garantia de Qualidade
Regression testing	Teste de Regressão
Release Coordinator	Coordenador de Liberações
Reliability Engineer	Engenheiro de Confiabilidade
Retrospective	Retrospectiva
Rhythm (in Lean)	Ritmo (em Lean)
Root Cause Analysis	Análise de Causa Raiz
Run time	Tempo de execução
Sad path	Pior caminho
Scaling (of DevOps or Agile Scrum)	Dimensionamento (de DevOps ou Agile Scrum)
Scrum	Scrum
Scrum Master (in Agile Scrum)	Scrum Master (em Agile Scrum)
Service Continuity	Continuidade de Serviços
Service Level Agreement (SLA)	Acordo de nível de serviço (ANS)
Service Master	Master de Serviço
Silos	Silos
Sprint	Sprint
System of Engagement (SoE)	Sistema de Engajamento (SoE)
System of Record (SoR)	Sistema de Registros (SoR)
System tests	Testes de Sistemas
Test-Driven Development	Desenvolvimento orientado a Testes
Tools	Ferramentas
Toyota Production System (TPS)	Sistema Toyota de Produção
Unit tests	Testes Unitários
Usability tests	Testes de Usabilidade
User Acceptance Testing (UAT)	Testes de Aceite dos Usuários
User Story	História do usuário
Value Stream Mapping (VSM)	Mapeamento do Fluxo de Valor (VSM)
Velocity (in Agile Scrum)	Velocidade (em Agile Scrum)
(Vendor) lock-in	Dependência do fornecedor
Version Control	Controle de Versão
Virtualization	Virtualização
Waste (in Lean)	Desperdício (em Lean)
Waterfall	Cascata
Work-in-Progress (WiP)	Trabalho em Andamento (WIP)

## 4. Literatura do exame

### Exam literature

O conhecimento exigido pelo exame de EXIN DevOps Master™ exame está baseado na seguinte literatura:

- A. Jennifer Davis, Katherine Daniels  
**AEffective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale**  
O'Reilly Media; 1 edition (Junho 25, 2016)  
ISBN-13: 978-1491926307  
ISBN-10: 1491926309
- B. Jez Humble, David Farley  
**Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation**  
Addison-Wesley Professional; 1 edition (Agosto 6, 2010)  
ISBN-13: 978-0321601919  
ISBN-10: 0321601912
- C. Koichiro (Luke) Toda, President Strategic Staff Services Corporation and Director of TPS Certificate Institution, and Nobuyuki Mitsui, CTO of Strategic Staff Services Corporation  
**Success with Enterprise DevOps**  
White Paper; Junho 2016 (download disponível na página EXIN DevOps Master™)

### Additional literature

- D. Gene Kim, Kevin Behr, George Spafford  
**The Phoenix Project**  
IT Revolution Press (January 10, 2013)  
ISBN-10: 0988262576  
ISBN-13: 978-0988262577
- E. Gene Kim, Jez Humble, Patrick Debois, John Willis  
**The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations**  
IT Revolution Press; 1 edition (2016)  
ISBN-10: 1942788002  
ISBN-13: 978-1942788003
- F. Other sources:  
<http://newrelic.com/devops>  
<http://devops.com/>

### Comentários

A literatura adicional destina-se exclusivamente a referência e aprofundamento do conhecimento. A leitura de The Phoenix Project é altamente recomendável antes do treinamento

## Literature matrix

Exam requirement		Literature
<b>1. Adoção do DevOps</b>		
1.1	Mentalidade (Mindset) DevOps e seus Benefício	
1.1.1	Analisar anti-padrões DevOps em um cenário	A: Capítulo 5 B: Capítulo 1 C: Capítulo 2
1.1.2	Explicar os benefícios do DevOps	B: Capítulo 1 C: Capítulo 4
1.1.3	Explicar por que o DevOps se adequa tão bem ao atual processo de desenvolvimento de software	A: Capítulo 3 B: Capítulo 1 C: Capítulo 4
1.1.4	Explicar por que DevOps precisa de um mentalidade específico para trabalhar	A: Capítulo 1, 2, 4 C: Capítulo 4
1.1.5	Explicar como o DevOps se encaixa com as práticas Lean e Agile Scrum	B: Capítulo 15 C: Capítulo 1, 4
1.2	Cultura Organizacional	
1.2.1	Explicar por que os 4 pilares de DevOps Eficaz (Colaboração, Afinidade, Ferramentas e Dimensionamento) são tão importantes	A: Capítulo 6, 7
1.2.2	Analisar um cenário com partes faltantes da mentalidade DevOps	A: Capítulo 7
1.2.3	Explicar como criar uma equipe com um grupo de pessoas, focando fomentar a colaboração, a mentalidade DevOps, a empatia e a confiança	A: Capítulo 7, 9
1.2.4	Analisar uma situação que tenha um equívoco sobre a colaboração e identificar o método correto de solução do problema	A: Capítulo 7, 8, 9, 16
1.2.5	Analisar uma situação em que há necessidade de gestão de conflitos e identificar a melhor solução	A: Capítulo 7, 14
1.2.6	Explicar como o gerenciamento de recursos humanos pode fomentar a diversidade e quais benefícios isso traz para a organização	A: Capítulo 7
1.3	Princípios e Conceitos DevOps	
1.3.1	Explicar o uso e a utilidade de diferentes metodologias de desenvolvimento de software (Cascata (Waterfall), Agile, Scrum) e seus princípios básicos	A: Capítulo 4 B: Capítulo 1 C: Capítulo 5
1.3.2	Explicar o uso e utilidade de diferentes metodologias de operações (Gerenciamento de Serviços de TI)	C: Capítulo 3, 4
1.3.3	Explicar o uso e a utilidade da metodologia de sistemas Lean	A: Capítulo 4
<b>2. Planejamento, Requisitos e Desenho</b>		
2.1	Gerenciamento do Ciclo de Vida de Aplicativos ou Serviços	
2.1.1	Explicar como o DevOps agrega valor ao Gerenciamento do Ciclo de Vida do Aplicativo moderno	C: Capítulo 1
2.1.2	Explicar por que a DevOps melhora a experiência do cliente quando usada para o Gerenciamento do Ciclo de Vida do Serviço	C: Capítulo 4
2.2	Termo de Abertura do Projeto (Definição de escopo) e Controle Visual	
2.2.1	Explicar como o escopo do projeto DevOps deve ser determinado	B: Capítulo 10 C: Capítulo 8
2.2.2	Explicar por que o Controle Visual em um projeto DevOps facilita as práticas DevOps	B: Capítulo 5 C: Capítulo 7



2.3	Desenho da Infraestrutura e Arquitetura	
2.3.1	Explicar como o DevOps muda ou influencia o projeto de infraestrutura e arquitetura de TI	B: Capítulo 11
2.3.2	Explicar por que a Computação em nuvem e as técnicas de virtualização tornam a integração de Dev e Ops mais fácil	B: Capítulo 11
2.4	Requisitos e acordos de nível de serviço	
2.4.1	Explicar como o DevOps altera os requisitos e os acordos de nível de serviço	B: Capítulo 12 C: Capítulo 4
2.5	Implementando uma Estratégia de Testes	
2.5.1	Explicar por que e como a Estratégia de Teste precisa ser alterada ao fazer a transição para o DevOps	B: Capítulo 4
2.5.2	Analisar Histórias de Usuário para completude	B: Capítulo 4
<b>3. Desenvolvimento e implantação</b>		
3.1	Entrega Contínua e Integração Contínua	
3.1.1	Explicar por que a Entrega Contínua é essencial para DevOps eficaz	B: Capítulo 13, 15
3.1.2	Analisar como integrar a Entrega Contínua em um cenário	B: Capítulo 3, 5, 15
3.1.3	Analisar como resolver problemas com a Entrega Contínua em um cenário	B: Capítulo 15
3.1.4	Explicar por que a Integração Contínua é essencial para DevOps eficaz	B: Capítulo 3
3.1.5	Analisar como alcançar a Integração Contínua em um cenário com uma equipe distribuída ou um sistema de controle de versão distribuído	B: Capítulo 3
3.1.6	Analisar como resolver problemas com Integração Contínua em um cenário	B: Capítulo 3
3.2	Pipeline de implantação	
3.2.1	Explicar a lógica da anatomia de um pipeline de implantação DevOps	B: Capítulo 5, 8 C: Capítulo 7
3.2.2	Explicar como usar scripts de criação e implantação	B: Capítulo 1, 6
3.3	Implantação contínua	
3.3.1	Explicar por que o plano de iteração e o plano de liberação devem ser alterados para um DevOps eficaz	C: Capítulo 5, 7
3.3.2	Analisar como implementar a implantação contínua em um cenário	B: Capítulo 10
3.4	Ji-Kotei-Kanketsu, Ritmo, Trabalho em Andamento e Fluxo Único (Fluxo Contínuo)	
3.4.1	Explicar os conceitos Ji-Kotei-Kanketsu, Ritmo, Trabalho em Andamento e Fluxo Único (Fluxo Contínuo)	C: Capítulo 4, 7
3.4.2	Analisar um cenário para um problema com Ji-kotei-Kanketsu, Ritmo, Trabalho em andamento ou Fluxo Único e encontrar uma solução adequada	A: Capítulo 1, 2, 9 B: Capítulo 1 C: Capítulo 4
3.5	Automação, Ferramentas e Testes	
3.5.1	Explicar por que a automação é importante para o DevOps eficaz	B: Capítulo 1, 8
3.5.2	Explicar como usar ferramentas para facilitar DevOps em geral	B: Capítulo 8
3.5.3	Explicar como usar ferramentas para dar suporte à mentalidade e cultura do DevOps	A: Capítulo 12
3.5.4	Explicar por que é importante que o teste de DevOps seja automatizado	A: Capítulo 1 B: Capítulo 3
3.5.5	Analisar um cenário e escolher a maneira correta de automatizar um teste de aceitação	B: Capítulo 8

<b>4. Operação e Dimensionamento</b>		
4.1	Gerenciamento de Dados; Infraestrutura e Ambientes; e Componentes e Dependências	
4.1.1	Explicar quais problemas podem ser encontrados ao gerenciar dados em bancos de dados dentro do DevOps	B: Capítulo 12
4.1.2	Analisar um cenário onde um banco de dados é usado em DevOps e fornecer a melhor solução para um problema	B: Capítulo 12
4.1.3	Analisar um cenário e identificar a melhor maneira de preparar um ambiente de infraestrutura para implantação ou gerenciá-lo após a implantação	B: Capítulo 11
4.1.4	Analisar um cenário e sugerir uma estratégia comumente usada para gerenciar componentes	B: Capítulo 13
4.1.5	Explicar como gerenciar dependências	B: Capítulo 13
4.2	Gerenciamento de Configuração e Controle de Versão	
4.2.1	Explicar por que o controle de versão é uma chave para o DevOps eficaz	B: Capítulo 2
4.2.2	Explicar como manter o controle de versão sobre dados, infraestrutura e componentes	B: Capítulo 10, 11
4.2.3	Analisar um cenário e sugerir a melhor estratégia para gerenciar um problema de configuração	B: Capítulo 2
4.3	Infraestrutura em Nuvens e Imutável	
4.3.1	Explicar quando é e quando não é necessário mover para a infraestrutura baseada em nuvem para ter um DevOps eficaz	A: Capítulo 17 B: Capítulo 11 C: Capítulo 5, 7
4.3.2	Explicar como a infraestrutura baseada em nuvem deve ser gerenciada dentro do DevOps	B: Capítulo 11
4.4	Continuidade do Negócio	
4.4.1	Explicar como o DevOps pode facilitar práticas de continuidade de negócios	C: Capítulo 4
4.5	Dimensionamento	
4.5.1	Analisar um cenário, explicar se e por que é importante dimensionar para cima ou para baixo nessa situação, e identificar a melhor maneira de fazer isso	A: Capítulo 14
4.5.2	Analisar um cenário que deu errada a dimensionamento, e identificar uma boa maneira de resolver o problema	A: Capítulo 15 B: Capítulo 11
4.5.3	Explicar como a política social e práticas de contratação suportam escalonamento DevOps	A: Capítulo 6, 7, 14
<b>5. Fim da vida</b>		
5.1	Condições de Fim de Vida de um produto ou serviço	
5.1.1	Expliquem que condições devem ser cumpridas antes de terminar um serviço ou produto	C: Capítulo 7

Observação: A leitura da literatura da fonte D, The Phoenix Project, será benéfica principalmente para a compreensão das seguintes especificações:

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

# Contato EXIN

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

