



Guide de Préparation

Édition 201811

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.



Table de matières

1. Résumé	4
2. Conditions de l'examen	8
3. Liste des concepts de base12	
4. Bibliographie	15

1. Résumé

EXIN DevOps Master™ (DEVOPSM.FR)

Portée

EXIN DevOps Master™ vérifient les connaissances du candidat des :

- Adoption de DevOps ;
- Planification, exigences et conception ;
- Développement et déploiement ;
- Exploitation (Fonctionnement) et Mise à l'échelle ;
- Fin de vie.

Sommaire

DevOps est la concaténation des trois premières lettres du mot anglais development (développement) et de l'abréviation usuelle (ops) du mot anglais operations (exploitation). DevOps est un ensemble de bonnes pratiques qui met l'accent sur la collaboration et la communication des professionnels de l'informatique (développeurs, exploitants et équipe de support) dans le cycle de vie des applications et des services, qui conduit à :

- Intégration continue : la fusion, plusieurs fois par jour, d'une branche sur laquelle un développeur travaille dans la branche master principale partagée
- Déploiement continu : mises en production continue ou aussi souvent que possible
- Evaluation continu : solliciter des retours des parties prenantes à travers toutes les étapes du cycle de vie

DevOps modifie la perception que les individus ont de leur travail ; DevOps valorise la diversité des travaux réalisés, favorise les processus intentionnels (volontaires) qui accélèrent le rythme auquel les entreprises produisent de la valeur, et mesure l'effet des changements sociaux et techniques. DevOps est un mode de pensée et une manière de travailler qui permet aux individus et aux organisations d'élaborer et d'entretenir des pratiques durables en matière de travail.

Un DevOps réussi :

- Se caractérise par une culture de partage de récits exempte de blâme, et le développement de l'empathie, permettant aux personnes et aux équipes d'exercer leur métier de manière efficace et durable
- Fournit des applications et des services juste-à-temps (JiT)
- Assure la continuité des services informatiques par une approche des besoins de l'entreprise basée sur le risque
- Gère l'ensemble du cycle de vie des applications et des services, incluant les conditions de fin de vie

Cette certification porte sur l'ajout de compétences pratiques aux connaissances, permettant à un DevOps Master™ d'intégrer DevOps avec succès au sein d'une équipe ainsi que de promouvoir ses principes dans l'organisation.

Cette certification a été élaborée en collaboration avec des experts dans le domaine de DevOps.

Contexte

Le programme DevOps d'EXIN :



Groupe cible

DevOps est connu dans le domaine du développement logiciel, mais ses principes sont applicables aux projets de services informatiques autant qu'à d'autres projets. La formation et la certification DevOps Master™ de l'EXIN visent les professionnels qui souhaitent promouvoir les pratiques DevOps dans leur organisation.

La certification EXIN DevOps Master™ s'adresse à toute personne qui évolue dans un contexte DevOps ou dans une organisation qui envisage la transition vers une approche DevOps. Le groupe cible comprend : les propriétaires de produits, les Scrum Masters Agile, les gestionnaires de projet, les gestionnaires des tests, les gestionnaires des services informatiques, les gestionnaires des processus et les praticiens Lean IT.

Exigences de la certification

- Le suivi d'une formation DevOps Master™ agréée par l'EXIN incluant exercices pratiques.
- La réussite à l'examen DevOps Master™.

Précisions sur l'examen

Type d'examen :	Questions à choix multiples
Nombre de questions :	50
Note minimale pour réussir :	65%
Accès à notes / manuel :	Non
Matériel / aides électronique autorisés :	Non
Durée de l'examen :	120 minutes

Les règles et règlements de l'EXIN en matière d'examens s'appliquent à cet examen.

Niveau Bloom

La certification d'EXIN DevOps Master™ teste les candidats aux niveaux 3 et 4 de la taxonomie révisée de Bloom :

- Niveau 3 : Application - indique que les candidats ont la capacité d'utiliser des informations dans un contexte différent de celui dans lequel elles ont été apprises. Ce type de questions vise à démontrer que le candidat est en mesure de résoudre des problèmes dans de nouvelles situations, en utilisant des connaissances acquises, des faits, des techniques et des règles d'une manière nouvelle ou différente. Les questions comportent généralement un bref scénario.
- Niveau 4 : Analyse - indique que les candidats sont aptes à décomposer les différents composants de l'information apprise afin de les comprendre. Ce niveau de la taxonomie de Bloom est principalement évalué par des exercices pratiques. Les exercices pratiques visent à démontrer que le candidat est apte à examiner et à décomposer les différents composants de l'information en identifiant les causes, à tirer des conclusions et à trouver des preuves appuyant des généralisations.

Formation

La formation est prérequis au passage de la certification. Les candidats doivent avoir des connaissances de base des principes DevOps et des concepts Lean et Agile. Celles-ci peuvent être acquises :

- par le biais de la certification EXIN DevOps Professional

Ou

- En lisant The Phoenix Project (voir bibliographie).

Heures de contact

Le nombre minimum d'heures de contact pour cette formation est de 24. Cela comprend les exercices pratiques, la préparation aux examens et de brèves pauses. Ce nombre d'heures n'inclut pas les devoirs, la logistique liée à la session de l'examen, la session de l'examen ni les pauses déjeuner.

Vous trouverez sur Partnernet des orientations relatives aux exercices pratiques. Il est recommandé d'y consacrer au maximum 8 heures. Les exercices pratiques peuvent être faits en dehors de la formation. Les formateurs, alors, peuvent opter pour :

- Une simulation GamingWorks (le formateur a alors besoin d'une accréditation GamingWorks)
- Le jeu du livre : Introduction to DevOps with Chocolate, LEGO and Scrum Game¹
- Le développement de leurs propres exercices pratiques

La durée de la formation peut varier selon la quantité de travail à fournir sur les exercices pratiques dans le cadre de la formation. Les options possibles sont :

- 2 journées de formation théorique en classe et 1 journée d'exercices pratiques en classe.
- 2 jours de formation théorique en ligne et 8 heures d'exercices pratiques sur le lieu de travail ; le candidat étudie en ligne et prépare les exercices pratiques ; le formateur évalue le travail fourni par le biais des exercices pratiques
- 2 journées de formation théorique en classe et 8 heures d'exercices pratiques sur le lieu de travail ; le formateur évalue le travail fourni par le biais des exercices pratiques

Charge de travail estimée

120 heures, en fonction de connaissances existantes. Dans les références de ce Guide de préparation, le terme *Ouvrage* fait référence à l'ensemble de connaissances testé pendant l'examen. Le tableau des ouvrages est présenté au chapitre 4.

Organisme de formation

Vous trouverez une liste des organismes de formation accrédités sur www.exin.com.

¹ **Introduction to DevOps with Chocolate, LEGO and Scrum Game**

Dana Pylayeva

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

ISBN-10: 1484225643

Apress: 2017

2. Conditions de l'examen

Les conditions de l'examen sont détaillées dans les spécifications de l'examen. Le tableau suivant énumère les sujets du module (conditions de l'examen) et les sous-rubriques (spécifications de l'examen).

Exigences de l'examen	Spécifications de l'examen	Importance
1. Adoption de DevOps		28%
	1.1 Esprit et bénéfices de DevOps	10%
	1.2 Culture organisationnelle	12%
	1.3 Principes et concepts de DevOps	6%
2. Planification, exigences et conception		18%
	2.1 Gestion du cycle de vie des applications ou des services	4%
	2.2 Charte projet (définition du périmètre) et Visual Control	4%
	2.3 Conception de l'infrastructure et de l'architecture	4%
	2.4 Exigences et Conventions de niveau de service	2%
	2.5 Mise en œuvre d'une stratégie de test	4%
3. Développement et déploiement		30%
	3.1 Livraison continue et Intégration continue	12%
	3.2 Pipeline de déploiement	4%
	3.3 Déploiement continu	4%
	3.4 Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress et One-piece-flow	4%
	3.5 Automatisation, Outils et Tests	6%
4. Exploitation (Fonctionnement) et Mise à l'échelle		22%
	4.1 Gestion des données ; Infrastructure et environnements et Composants et dépendances	10%
	4.2 Gestion de configuration et Contrôle de version	4%
	4.3 Infrastructure cloud et Infrastructure immutable	2%
	4.4 Continuité des activités (des affaires)	2%
	4.5 Mise à l'échelle	4%
5. Fin de vie		2%
	5.1 Conditions de fin de vie d'un produit ou d'un service	2%
Total		100%

Spécifications de l'examen

1. Adoption de DevOps

1.1 Esprit et bénéfices de DevOps

Le candidat est capable de ...

- 1.1.1 Analyser des erreurs de DevOps dans un scénario
- 1.1.2 Expliquer les avantages de DevOps
- 1.1.3 Expliquer pourquoi DevOps s'adapte si bien au processus de développement logiciel actuel
- 1.1.4 Expliquer pourquoi l'adoption de DevOps exige un état d'esprit spécifique
- 1.1.5 Expliquer comment DevOps s'adapte aux pratiques du Lean et d'Agile Scrum

1.2 Culture organisationnelle

Le candidat est capable de ...

- 1.2.1 Expliquer l'importance des 4 piliers d'un DevOps efficace (Collaboration, Affinité, Outils et Mise à l'échelle)
- 1.2.2 Analyser un scénario pour les éléments manquants de l'esprit DevOps
- 1.2.3 Expliquer comment créer une équipe à partir d'un groupe de personnes, en favorisant la collaboration, la façon de penser DevOps, et l'empathie et la confiance
- 1.2.4 Analyser une situation où il existe une perception erronée concernant la collaboration et identifier la méthode corrective appropriée
- 1.2.5 Analyser une situation où il y a un besoin pour la gestion des conflits et identifier la meilleure solution
- 1.2.6 Expliquer comment la gestion des ressources humaines peut favoriser la diversité et quels avantages cela apporte à l'organisation

1.3 Principes et concepts de DevOps

Le candidat est capable de ...

- 1.3.1 Expliquer l'usage et l'utilité de différentes méthodologies de développement logiciel (en Cascade (Waterfall), Agile, Scrum) et leurs principes de base
- 1.3.2 Expliquer l'usage et l'utilité des différentes méthodologies d'exploitation (Gestion des services informatiques - IT Service Management)
- 1.3.3 Expliquer l'usage et l'utilité de la méthodologie des systèmes Lean

2. Planification, exigences et conception

2.1 Gestion du cycle de vie des applications ou des services

Le candidat est capable de ...

- 2.1.1 Expliquer comment DevOps ajoute de la valeur à la Gestion du cycle de vie des applications modernes
- 2.1.2 Expliquer pourquoi DevOps améliore l'expérience client lorsqu'il est utilisé pour la Gestion du cycle de vie des services

2.2 Charte projet et Visual Control

Le candidat est capable de ...

- 2.2.1 Expliquer comment le périmètre d'un projet DevOps devrait être déterminé
- 2.2.2 Expliquer pourquoi Visual Control facilite les pratiques de DevOps sur un projet DevOps

2.3 Conception de l'infrastructure et de l'architecture

Le candidat est capable de ...

- 2.3.1 Expliquer comment DevOps modifie ou influence la conception de l'infrastructure et de l'architecture informatique
- 2.3.2 Expliquer pourquoi le cloud computing et les techniques de virtualisation facilitent l'intégration des Dev et Ops

- 2.4 Exigences et Accords de niveau de service
 - Le candidat est capable de ...
 - 2.4.1 Expliquer comment DevOps modifie les Exigences et les Accords de niveau de service (SLR et ANS)
- 2.5 Mise en œuvre d'une stratégie de test
 - Le candidat est capable de ...
 - 2.5.1 Expliquer pourquoi et comment la stratégie de test doit être modifiée lors de la transition vers DevOps
 - 2.5.2 Analyser les Récits utilisateur pour en vérifier l'exhaustivité

3. Développement et déploiement

- 3.1 Livraison Continue et Intégration Continue
 - Le candidat est capable de ...
 - 3.1.1 Expliquer pourquoi la Livraison continue est essentielle pour DevOps efficace
 - 3.1.2 Analyser comment intégrer la Livraison Continue dans un scénario
 - 3.1.3 Analyser comment résoudre des problèmes de livraison continue dans un scénario
 - 3.1.4 Expliquer pourquoi l'Intégration continue est essentielle pour un DevOps efficace
 - 3.1.5 Analyser comment réaliser l'Intégration continue dans un scénario avec une équipe distribuée ou un système de contrôle de version distribué
 - 3.1.6 Analyser comment résoudre des problèmes d'intégration continue dans un scénario
- 3.2 Pipeline de déploiement
 - Le candidat est capable de ...
 - 3.2.1 Expliquer la logique de l'anatomie d'un Pipeline de déploiement DevOps
 - 3.2.2 Expliquer comment utiliser les scripts de réalisation et de déploiement
- 3.3 Déploiement continu
 - Le candidat est capable de ...
 - 3.3.1 Expliquer pourquoi le plan d'itération et le plan de libération doivent être modifiés pour un DevOps efficace
 - 3.3.2 Analyser comment implémenter le Déploiement continu dans un scénario
- 3.4 Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress et One-piece-flow
 - Le candidat est capable de ...
 - 3.4.1 Expliquer les concepts Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress et One-piece-flow
 - 3.4.2 Analyser un scénario pour un problème avec Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress or One-piece-flow et trouver une solution appropriée
- 3.5 Automatisation, Outils et Tests
 - Le candidat est capable de ...
 - 3.5.1 Expliquer pourquoi l'automatisation est importante pour un DevOps efficace
 - 3.5.2 Expliquer comment utiliser les outils pour faciliter DevOps en général
 - 3.5.3 Expliquer comment utiliser les outils pour soutenir l'esprit et la culture DevOps
 - 3.5.4 Expliquer pourquoi il est important que les tests DevOps soient automatisés
 - 3.5.5 Analyser un scénario et choisir la manière correcte d'automatiser un test d'acceptation

4. Exploitation (Fonctionnement) et Mise à l'échelle

4.1 Gestion des données ; Infrastructure et environnements et Composants et dépendances

Le candidat est capable de ...

- 4.1.1 Expliquer quels problèmes peut être rencontrés lors de la gestion des données dans les bases de données selon DevOps
- 4.1.2 Analyser un scénario où une base de données est utilisée dans DevOps et fournir la meilleure solution à un problème
- 4.1.3 Analyser un scénario et identifier la meilleure façon de préparer un environnement d'infrastructure pour le déploiement ou de le gérer après le déploiement
- 4.1.4 Analyser un scénario et suggérer une stratégie couramment utilisée pour gérer les composants
- 4.1.5 Expliquer comment gérer les dépendances

4.2 Gestion de configuration et Contrôle de version

Le candidat est capable de ...

- 4.2.1 Expliquer pourquoi le contrôle de version est essentiel pour un DevOps efficace
- 4.2.2 Expliquer comment garder le contrôle de version des données, de l'infrastructure et des composants
- 4.2.3 Analyser un scénario et suggérer la meilleure stratégie pour gérer un problème de configuration

4.3 Infrastructure cloud et Infrastructure immuable

Le candidat est capable de ...

- 4.3.1 Expliquer quand il est nécessaire, ou pas, de passer à une infrastructure basée sur le cloud pour un DevOps efficace
- 4.3.2 Expliquer comment l'infrastructure cloud doit être gérée selon DevOps

4.4 Continuité des activités (des affaires)

Le candidat est capable de ...

- 4.4.1 Expliquer comment DevOps facilite les pratiques de Continuité des Activités

4.5 Mise à l'échelle (Scaling)

Le candidat est capable de ...

- 4.5.1 Analyser un scénario, expliquer si et pourquoi il est important d'intensifier ou de réduire l'ampleur dans cette situation, et d'identifier la meilleure façon de le faire
- 4.5.2 Analyser un scénario pour ce qui ne va pas avec la mise à l'échelle, et identifier une bonne façon de résoudre le problème
- 4.5.3 Expliquer comment la politique sociale et les pratiques de recrutement soutiennent la mise à l'échelle DevOps

5. Fin de vie

5.1 Conditions de fin de vie d'un produit ou d'un service

Le candidat est capable de ...

- 5.1.1 Expliquer quelles sont les conditions requises pour le retrait d'un service ou un produit

3. Liste des concepts de base

Ce chapitre contient les termes et abréviations avec lesquels le candidat doit se familiariser.

Veillez noter que la connaissance de ces termes seule ne suffit pas pour l'examen ; le candidat doit comprendre le concept et être en mesure de fournir des exemples.

English	French
A/B testing	test A / B
acceptance Tests	tests d'acceptation
affinity (<i>in DevOps</i>)	affinité (<i>dans DevOps</i>)
Agile	Agile
anti-pattern	contre-modèle
application deployment	déploiement d'application
artifact management	gestion des artefacts
(artifact) repository	répertoire (des artefacts)
ATAM	ATAM
automated testing	test automatisé
automation	automatisation
binary files	fichiers binaires
blamelessness	irréprochabilité
Blue-Green deployment	déploiement bleu/vert
build (management)	(gestion de) version (build)
check-in	enregistrement
Cloud Computing	Cloud Computing
collaboration (<i>in DevOps</i>)	collaboration (<i>dans DevOps</i>)
commit (stage)	commissionnement (étape)
communication styles	styles de communication
compact	Compact
component (tests)	composant (tests)
configuration management	gestion de configuration
containers	conteneurs
continuous delivery	livraison continue
continuous deployment	déploiement continu
continuous integration	intégration continue
cycle time	durée d'un cycle
Definition of Done (<i>in Agile Scrum</i>)	Définition d'achevé (Definition of Done) (<i>dans Agile Scrum</i>)
dependency	dépendance
(deployment) pipeline	pipeline (de déploiement)
Development team	équipe de Développement
DevOps Engineer	Ingénieur DevOps
Disciplined Agile	Agile discipliné (Disciplined Agile)
distributed Team	équipe dispersée
effective DevOps	DevOps efficace
event management	gestion des événements
exploratory testing	tests exploratoires
flow	flux
functional acceptance test	test d'acceptation fonctionnelle
forensic tools	outils médico-légaux
Gatekeeper	Gardien

happy path	chemin heureux (happy path)
human error	erreur humaine
incident management	gestion des incidents
information radiators	diffuseurs d'information
infrastructure management	gestion de l'infrastructure
infrastructure automation	automatisation de l'infrastructure
INVEST	INVEST
integration tests	tests d'intégration
iteration	itération
ITSM (IT Service Management)	Gestion des Services Informatiques (ITSM)
Ji-Kotei-Kanketsu (JKK)	Ji-Kotei-Kanketsu (JKK)
Just-in-Time (JiT)	juste-à-temps (Just-in-Time ; JiT)
Kaizen (<i>in Lean</i>)	Kaizen (<i>dans Lean</i>)
Lean	Lean
libraries	bibliothèques
light-weight ITSM	ITSM léger
manual testing	test manuel
minimum viable product	produit minimum viable
monitoring strategy	stratégie de surveillance
negotiation styles	styles de négociation
non-functional testing	tests non fonctionnels
Obeya	Obeya
orchestration	orchestration
one-piece-flow	flux pièce par pièce (One-Piece-Flow)
Operations team	Équipe d'exploitation
organizational learning	apprentissage organisationnel
Plan-Do-Check-Act cycle (PDCA cycle)	Roue de Deming (PDCA)
Process Master	Maître de processus
(Product) Backlog	Carnet de Produit
Product Owner (<i>in Agile Scrum</i>)	Propriétaire de produit (<i>dans Agile Scrum</i>)
project charter	charte de projet
pull system	système de traction
Quality Assurance (QA)	Assurance qualité (QA)
regression testing	tests de régression
Release Coordinator	Coordinateur de mise en production
Reliability Engineer	ingénieur fiabilité
retrospective	rétrospective
rhythm (<i>in Lean</i>)	rythme / cadence (<i>en Lean</i>)
Root Cause Analysis	analyse de la cause originelle
run time	temps d'exécution
sad path	chemin triste (sad path)
scaling (<i>of DevOps or Agile Scrum</i>)	mise à l'échelle (Scaling) (<i>de DevOps ou Agile Scrum</i>)
Scrum	Scrum
Scrum Master (<i>in Agile Scrum</i>)	Scrum Master (<i>dans Agile Scrum</i>)
service continuity	continuité de service
Service Level Agreement (SLA)	Accord des niveaux de service (ANS)
Service Master	Maître de service
silos	silos
Sprint	Sprint
System of Engagement (SoE)	système d'engagement (SoE)
System of Record (SoR)	système d'enregistrement (SoR)
system tests	tests système
test-driven development	développement piloté par les tests

tools	outils
Toyota Production System (TPS)	Système de Production Toyota (TPS)
unit test	test unitaire
usability tests	tests d'utilisabilité
User Acceptance Testing (UAT)	test d'acceptation utilisateur (UAT)
User Story	Récit Utilisateur
Value Stream Mapping	cartographie des chaînes de valeur
velocity (<i>in Agile Scrum</i>)	vélocité (<i>dans Agile Scrum</i>)
(vendor) lock-in	verrouillage fournisseur
Version Control	contrôle des versions
virtualization	virtualisation
Waste (<i>in Lean</i>)	gaspillage (<i>dans Lean</i>)
Waterfall	Cascade
Work-in-Progress (WiP)	Travail en Cours (Work-in-Progress, WiP)

4. Bibliographie

Ouvrages pour l'examen

Les connaissances requises pour le passage de l'examen EXIN DevOps Master™ sont couvertes par les ouvrages suivants.

- A. Jennifer Davis, Katherine Daniels
AEffective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale
O'Reilly Media; 1ère édition (juin 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; 1ère édition (août 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; juin 2016 (téléchargé de la page produit EXIN DevOps Master™)

Bibliographie complémentaire

- 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/>

Commentaire

La littérature supplémentaire est pour la référence et la profondeur de la connaissance seulement. Il est fortement recommandé de lire The Phoenix Project avant la formation.

Matrice de littérature

Spécification de l'examen		Littérature
1. Adoption de DevOps		
1.1	Esprit et bénéfices de DevOps	
1.1.1	Analyser des erreurs de DevOps dans un scénario	A: Chapitre 5 B: Chapitre 1 C: Chapitre 2
1.1.2	Expliquer les avantages de DevOps	B: Chapitre 1 C: Chapitre 4
1.1.3	Expliquer pourquoi DevOps s'adapte si bien au processus de développement logiciel actuel	A: Chapitre 3 B: Chapitre 1 C: Chapitre 4
1.1.4	Expliquer pourquoi l'adoption de DevOps exige un état d'esprit spécifique	A: Chapitre 1, 2, 4 C: Chapitre 4
1.1.5	Expliquer comment DevOps s'adapte aux pratiques du Lean et d'Agile Scrum	B: Chapitre 15 C: Chapitre 1, 4
1.2	Culture organisationnelle	
1.2.1	Expliquer l'importance des 4 piliers d'un DevOps efficace (Collaboration, Affinité, Outils et Mise à l'échelle)	A: Chapitre 6, 7
1.2.2	Analyser un scénario pour les éléments manquants de l'esprit DevOps	A: Chapitre 7
1.2.3	Expliquer comment créer une équipe à partir d'un groupe de personnes, en favorisant la collaboration, la façon de penser DevOps, et l'empathie et la confiance	A: Chapitre 7, 9
1.2.4	Analyser une situation où il existe une perception erronée concernant la collaboration et identifier la méthode corrective appropriée	A: Chapitre 7, 8, 9, 16
1.2.5	Analyser une situation où il y a un besoin pour la gestion des conflits et identifier la meilleure solution	A: Chapitre 7, 14
1.2.6	Expliquer comment la gestion des ressources humaines peut favoriser la diversité et quels avantages cela apporte à l'organisation	A: Chapitre 7
1.3	Principes et concepts de DevOps	
1.3.1	Expliquer l'usage et l'utilité de différentes méthodologies de développement logiciel (en Cascade (Waterfall), Agile, Scrum) et leurs principes de base	A: Chapitre 4 B: Chapitre 1 C: Chapitre 5
1.3.2	Expliquer l'usage et l'utilité des différentes méthodologies d'exploitation (Gestion des services informatiques - IT Service Management)	C: Chapitre 3, 4
1.3.3	Expliquer l'usage et l'utilité de la méthodologie des systèmes Lean	A: Chapitre 4
2. Planification, exigences et conception		
2.1	Gestion du cycle de vie des applications ou des services	
2.1.1	Expliquer comment DevOps ajoute de la valeur à la Gestion du cycle de vie des applications modernes	C: Chapitre 1
2.1.2	Expliquer pourquoi DevOps améliore l'expérience client lorsqu'il est utilisé pour la Gestion du cycle de vie des services	C: Chapitre 4
2.2	Charte projet et Visual Control	
2.2.1	Expliquer comment le périmètre d'un projet DevOps devrait être déterminé	B: Chapitre 10 C: Chapitre 8
2.2.2	Expliquer pourquoi Visual Control facilite les pratiques de DevOps sur un projet DevOps	B: Chapitre 5 C: Chapitre 7

2.3	Conception de l'infrastructure et de l'architecture	
2.3.1	Expliquer comment DevOps modifie ou influence la conception de l'infrastructure et de l'architecture informatique	B: Chapitre 11
2.3.2	Expliquer pourquoi le cloud computing et les techniques de virtualisation facilitent l'intégration des Dev et Ops	B: Chapitre 11
2.4	Exigences et Accords de niveau de service	
2.4.1	Expliquer comment DevOps modifie les Exigences et les Accords de niveau de service (SLR et ANS)	B: Chapitre 12 C: Chapitre 4
2.5	Mise en œuvre d'une stratégie de test	
2.5.1	Expliquer pourquoi et comment la stratégie de test doit être modifiée lors de la transition vers DevOps	B: Chapitre 4
2.5.2	Analyser les Récits utilisateur pour en vérifier l'exhaustivité	B: Chapitre 4
3. Développement et déploiement		
3.1	Livraison Continue et Intégration Continue	
3.1.1	Expliquer pourquoi la Livraison continue est essentielle pour DevOps efficace	B: Chapitre 13, 15
3.1.2	Analyser comment intégrer la Livraison Continue dans un scénario	B: Chapitre 3, 5, 15
3.1.3	Analyser comment résoudre des problèmes de livraison continue dans un scénario	B: Chapitre 15
3.1.4	Expliquer pourquoi l'Intégration continue est essentielle pour un DevOps efficace	B: Chapitre 3
3.1.5	Analyser comment réaliser l'Intégration continue dans un scénario avec une équipe distribuée ou un système de contrôle de version distribué	B: Chapitre 3
3.1.6	Analyser comment résoudre des problèmes d'intégration continue dans un scénario	B: Chapitre 3
3.2	Pipeline de déploiement	
3.2.1	Expliquer la logique de l'anatomie d'un Pipeline de déploiement DevOps	B: Chapitre 5, 8 C: Chapitre 7
3.2.2	Expliquer comment utiliser les scripts de réalisation et de déploiement	B: Chapitre 1, 6
3.3	Déploiement continu	
3.3.1	Expliquer pourquoi le plan d'itération et le plan de libération doivent être modifiés pour un DevOps efficace	C: Chapitre 5, 7
3.3.2	Analyser comment implémenter le Déploiement continu dans un scénario	B: Chapitre 10
3.4	Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress et One-piece-flow	
3.4.1	Expliquer les concepts Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress et One-piece-flow	C: Chapitre 4, 7
3.4.2	Analyser un scénario pour un problème avec Ji-Kotei-Kanketsu, Rhythm, Work-in-Progress or One-piece-flow et trouver une solution appropriée	A: Chapitre 1, 2, 9 B: Chapitre 1 C: Chapitre 4
3.5	Automatisation, Outils et Tests	
3.5.1	Expliquer pourquoi l'automatisation est importante pour un DevOps efficace	B: Chapitre 1, 8
3.5.2	Expliquer comment utiliser les outils pour faciliter DevOps en général	B: Chapitre 8
3.5.3	Expliquer comment utiliser les outils pour soutenir l'esprit et la culture DevOps	A: Chapitre 12
3.5.4	Expliquer pourquoi il est important que les tests DevOps soient automatisés	A: Chapitre 1 B: Chapitre 3
3.5.5	Analyser un scénario et choisir la manière correcte d'automatiser un test d'acceptation	B: Chapitre 8

4. Exploitation (Fonctionnement) et Mise à l'échelle		
4.1	Gestion des données ; Infrastructure et environnements et Composants et dépendances	
4.1.1	Expliquer quels problèmes peuvent être rencontrés lors de la gestion des données dans les bases de données selon DevOps	B: Chapitre 12
4.1.2	Analyser un scénario où une base de données est utilisée dans DevOps et fournir la meilleure solution à un problème	B: Chapitre 12
4.1.3	Analyser un scénario et identifier la meilleure façon de préparer un environnement d'infrastructure pour le déploiement ou de le gérer après le déploiement	B: Chapitre 11
4.1.4	Analyser un scénario et suggérer une stratégie couramment utilisée pour gérer les composants	B: Chapitre 13
4.1.5	Expliquer comment gérer les dépendances	B: Chapitre 13
4.2	Gestion de configuration et Contrôle de version	
4.2.1	Expliquer pourquoi le contrôle de version est essentiel pour un DevOps efficace	B: Chapitre 2
4.2.2	Expliquer comment garder le contrôle de version des données, de l'infrastructure et des composants	B: Chapitre 10, 11
4.2.3	Analyser un scénario et suggérer la meilleure stratégie pour gérer un problème de configuration	B: Chapitre 2
4.3	Infrastructure cloud et Infrastructure immuable	
4.3.1	Expliquer quand il est nécessaire, ou pas, de passer à une infrastructure basée sur le cloud pour un DevOps efficace	A: Chapitre 17 B: Chapitre 11 C: Chapitre 5, 7
4.3.2	Expliquer comment l'infrastructure cloud doit être gérée selon DevOps	B: Chapitre 11
4.4	Continuité des activités (des affaires)	
4.4.1	Expliquer comment DevOps facilite les pratiques de Continuité des Activités	C: Chapitre 4
4.5	Mise à l'échelle (Scaling)	
4.5.1	Analyser un scénario, expliquer si et pourquoi il est important d'intensifier ou de réduire l'ampleur dans cette situation, et d'identifier la meilleure façon de le faire	A: Chapitre 14
4.5.2	Analyser un scénario pour ce qui ne va pas avec la mise à l'échelle, et identifier une bonne façon de résoudre le problème	A: Chapitre 15 B: Chapitre 11
4.5.3	Expliquer comment la politique sociale et les pratiques de recrutement soutiennent la mise à l'échelle DevOps	A: Chapitre 6, 7, 14
5. Fin de vie		
5.1	Conditions de fin de vie d'un produit ou d'un service	
5.1.1	Expliquer quelles sont les conditions requises pour le retrait d'un service ou un produit	C: Chapitre 7

Remarque : La lecture de l'ouvrage de référence D The Phoenix Project facilite grandement la compréhension des spécifications suivantes :

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

Contacter EXIN

www.exin.com

