



REFERENTIEL DE COMPETENCES ET D'ÉVALUATION DE LA CERTIFICATION

METTRE EN ŒUVRE DEVOPS POUR LE CLOUD MICROSOFT AZURE

(Titre en anglais : Microsoft Certified Azure DevOps Engineer Expert)

MODALITES D'ÉVALUATION

Pour obtenir cette certification, il est proposé au candidat d'évaluer leurs compétences à travers un (1) examen en ligne, supervisé par l'organisme Pearson VUE, délivré dans un centre d'examen accrédité (ou via de la surveillance à distance).

L'examen dure environ deux (2) heures – livres fermés - et comprend une variété de questions appelant différentes formes de réponse* : Écran actif, Meilleure réponse, Construction de liste, Études de cas, Glisser-déplacer, Zone réactive, Choix multiple, Plusieurs choix de réponse, Réponse courte, Révision des notes, Écran de révision.

*Détaillées à cette URL : <https://www.microsoft.com/en-us/learning/certification-exams.aspx?types=true>

Le seuil de réussite est fixé à environ 70% de bonnes réponses, qui correspond à un score de passage de 700 points (score à l'échelle). Le pourcentage réel varie d'un examen à l'autre. La note de passage est basée sur l'apport d'experts en la matière, le niveau de compétence requis pour être considéré comme compétent dans le domaine du contenu, et la difficulté des questions livrées pendant l'examen. Les pourcentages dans le tableau des compétences évaluées indiquent le poids relatif de chaque sujet principal de l'examen. Plus le pourcentage est élevé, plus les candidats devront répondre à des questions sur cette zone de contenu. La liste des tâches évaluées n'est pas exhaustive et peut couvrir d'autres tâches dans le cadre des compétences évaluées.

Le contenu des tests est réévalué régulièrement par les équipes Microsoft (dernière mise à jour le 15 juin 2020).

COMPETENCES GENERALES :

Les candidats à cet examen sont des professionnels de la culture DevOps qui combinent des personnes de différents services, des processus et des technologies pour fournir en permanence des produits et des services qui répondent aux besoins des utilisateurs finaux et aux objectifs commerciaux de l'entreprise. Les professionnels de DevOps rationalisent la livraison en optimisant les pratiques, en améliorant les communications et la collaboration et en créant une automatisation. Ils conçoivent et mettent en œuvre des stratégies de code d'application et d'infrastructure qui permettent une intégration, des tests, une livraison, une surveillance et un retour d'informations continus.



Les candidats doivent maîtriser les pratiques en gestion de projet « Agiles ». Ils doivent être familiarisés avec l'administration de Microsoft Azure et le développement Azure. [Les professionnels Azure DevOps sont évalués sur leur capacité à concevoir et mettre en œuvre des pratiques DevOps pour le contrôle de version d'application, la conformité, l'infrastructure en tant que code, la gestion de la configuration, la génération, la publication et les tests à l'aide des technologies Azure.](#)

COMPETENCES MOBILISABLES EVALUEES		Nature des tâches qui constituent la compétence	EVALUATION		
Sur 100% :			% de l'évaluation globale	Modalités d'évaluation	Critères
Concevoir une stratégie d'instrumentation DevOps			20-25%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 30 minutes (pour 14 à 16 questions) sont consacrées à cette compétence	Le taux de bonnes réponses doit être au global de 70% minimum
Recommander une stratégie de migration et de consolidation pour les outils DevOps	<ul style="list-style-type: none"> analyser l'artefact existant (par exemple, les packages de déploiement, NuGet, Maven, npm) et le conteneur référentiels analyser les outils de gestion des tests existants recommander des stratégies de migration et d'intégration pour les référentiels d'artefacts, le contrôle des sources, la gestion des tests et la gestion du travail 				
Concevoir et mettre en œuvre une approche de gestion du travail Agile	<ul style="list-style-type: none"> identifier et recommander des mesures de projet, des KPI et des mesures DevOps (par exemple, cycle temps, délai, limite WIP) mettre en œuvre des outils et des processus pour soutenir la gestion du travail Agile encadrer les membres de l'équipe sur les techniques et pratiques Agiles recommander une structure organisationnelle qui prend en charge la mise à l'échelle des pratiques Agile recommander des mécanismes de collaboration en équipe et entre équipes 				

	Concevoir un processus de développement sécurisé	<ul style="list-style-type: none"> inspecter et valider la base de code pour la conformité inspecter et valider la conformité de l'infrastructure recommander une stratégie de développement sécurisée recommander des outils et des pratiques pour intégrer la validation de la sécurité du code (par ex., l'analyse de code statique) et de la sécurité de l'infrastructure 			
	Concevoir une stratégie d'intégration d'outils	<ul style="list-style-type: none"> concevoir une stratégie de gestion des licences (par exemple, utilisateurs VSTS, pipelines simultanés, test environnements, licences de logiciels open source, outils et services DevOps tiers, licence de gestion des packages) concevoir une stratégie de traçabilité de bout en bout des éléments de travail au logiciel de travail concevoir une stratégie pour intégrer le suivi et le feedback aux équipes de développement concevoir une stratégie d'authentification et d'accès concevoir une stratégie d'intégration des ressources locales et cloud 			
Mettre en œuvre des processus de développement DevOps			20-25%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 30 minutes (pour 14 à 16 questions) sont consacrées à	Le taux de bonnes réponses doit être au global de 70% minimum
	Concevoir une stratégie de contrôle de version	<ul style="list-style-type: none"> recommander des modèles de branchement recommander des systèmes de contrôle de version recommander une stratégie de flux de code 			
	Mettre en œuvre et intégrer le contrôle des sources	<ul style="list-style-type: none"> Intégrer un contrôle de source externe intégrer un contrôle de source en systèmes d'intégration continue et de déploiement continu (CI/CD) de tiers 			
	Mettre en œuvre et gérer l'infrastructure de construction	<ul style="list-style-type: none"> implémenter des agents privés et hébergés intégrer des systèmes de construction tiers recommander une stratégie pour les pipelines simultanés gérer la configuration du pipeline Azure (par exemple, les files d'attente d'agents, les points de terminaison de service, les pools, webhooks) 			

	Implémenter le flux de code	<ul style="list-style-type: none"> • mettre en œuvre des stratégies de demande de pull • mettre en œuvre des stratégies de succursales et d'embranchements • configurer les politiques de succursales 		cette compétence	
	Gestion de la configuration et des secrets d'application	<ul style="list-style-type: none"> • mettre en œuvre un processus de développement sécurisé et conforme • implémenter des données de configuration générales (non secrètes) • gérer les secrets, les jetons et les certificats • implémenter des configurations d'applications (par exemple, Web App, Azure Kubernetes Service, conteneurs) • implémenter la gestion des secrets (par exemple, Web App, Azure Kubernetes Service, conteneurs, Azure Key Vault) • implémenter des outils pour gérer la sécurité et la conformité dans le pipeline 			
Mettre en œuvre un processus d'intégration continue					
	Gérer la qualité du code et les politiques de sécurité	<ul style="list-style-type: none"> • surveiller la qualité du code • configurer la construction pour rendre compte de la couverture du code • gérer la qualité des tests automatisés • gérer les suites de tests et les catégories • surveiller la qualité des tests • intégrer des outils d'analyse de sécurité (par exemple, SonarQube, White Source Bolt, Open Web Projet de sécurité des applications) 	10-15%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 20 minutes (pour une 10aine questions) sont consacrées à cette compétence	Le taux de bonnes réponses doit être au global de 70% minimum
	Mettre en œuvre une stratégie de construction de conteneurs	<ul style="list-style-type: none"> • créer des images déployables (par exemple, Docker, Hub, Azure Container Registry) • analyser et intégrer les constructions à étapes multiples Docker 			

	Mettre en œuvre une stratégie de construction	<ul style="list-style-type: none"> • concevoir des déclencheurs de construction, des outils, des intégrations et un flux de travail • mettre en œuvre un processus de construction hybride • mettre en œuvre des constructions à agents multiples • recommander des outils de construction et la configuration (Azure Pipelines, Jenkins) • définir un flux de travail de construction automatisé 			
Mettre en œuvre un processus de livraison continue			10-15%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 20 minutes (pour une 10aine questions) sont consacrées à cette compétence	Le taux de bonnes réponses doit être au global de 70% minimum
	Concevoir une stratégie de publication	<ul style="list-style-type: none"> • identifier et recommander les autorisations de sortie et les portes • recommander une stratégie pour mesurer la qualité de la version et du processus de version • recommander une stratégie pour les notes de publication et la documentation • sélectionner le modèle de déploiement approprié 			
	Configurer un workflow de gestion des versions	<ul style="list-style-type: none"> • automatiser l'inspection des signaux de santé pour les approbations de libération en utilisant des portes de libération • configurer l'intégration automatisée et l'exécution des tests fonctionnels • créer un pipeline de versions (par exemple, Azure Kubernetes Service, Service Fabric, WebApp) • créer des pipelines de libération en plusieurs phases • intégrer des secrets avec le pipeline de versions • approvisionner et configurer les environnements • gérer et modulariser les tâches et les modèles (par exemple, les groupes de tâches et de variables) 			

<p>Mettre en œuvre un modèle de déploiement approprié</p>	<ul style="list-style-type: none"> • implémenter des déploiements bleu-vert • implémenter des déploiements canaris • implémenter des déploiements d'exposition progressive • faire évoluer un pipeline de versions à déployer sur plusieurs points de terminaison (par exemple, des groupes de déploiement, Azure Service Kubernetes, Service Fabric) 			
<p>Implémenter la gestion des dépendances</p>		<p>5-10%</p>	<p>Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 10 minutes (pour répondre à 5 questions) sont consacrées à cette compétence</p>	<p>Le taux de bonnes réponses doit être au global de 70% minimum</p>
<p>Concevoir une stratégie de gestion des dépendances</p>	<ul style="list-style-type: none"> • recommander des outils et des pratiques de gestion des artefacts (Azure Artifacts, npm, Maven, Nuget) • activer le partage et la réutilisation avec des progiciels communs abstraits • inspecter la base de code pour identifier les dépendances de code qui peuvent être converties en packages • identifier et recommander des types et versions de packages standardisés dans la solution • refactoriser les pipelines de construction existants pour implémenter une stratégie de version qui publie des packages 			
<p>Gérer la sécurité et la conformité</p>	<ul style="list-style-type: none"> • inspecter les packages de logiciels open source pour la sécurité et la conformité des licences pour normes d'entreprise (par exemple, GPLv3) • configurer le pipeline de construction pour accéder à la sécurité du package et à la classification des licences (par exemple, Black Duck, Source blanche) • configurer un accès sécurisé aux flux de packages 			

Mettre en œuvre l'infrastructure d'application					
	Concevoir une stratégie de gestion de l'infrastructure et de la configuration	<ul style="list-style-type: none"> • analyser l'infrastructure d'hébergement existante et future • analyser l'infrastructure existante en tant que technologies de code (IaC) • concevoir une stratégie de gestion de la dette technique sur des modèles • concevoir une stratégie d'utilisation de l'infrastructure transitoire pour certaines parties d'un cycle de vie de livraison • concevoir une stratégie pour atténuer la dérive de l'état des infrastructures 	15-20%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 25 minutes (pour 10-12 questions) sont consacrées à cette compétence	Le taux de bonnes réponses doit être au global de 70% minimum
	Mettre en œuvre l'infrastructure en tant que code (IaC)	<ul style="list-style-type: none"> • créer des modèles de ressources imbriqués • gérer les secrets dans les modèles de ressources • provisionner des ressources Azure • recommander une stratégie Infrastructure as Code (IaC) • recommander des technologies appropriées pour la gestion de la configuration (par exemple, ARM Modèles, Terraform, Chef, Marionnette, Ansible) 			
	Mettre en œuvre la conformité et la sécurité de l'infrastructure	<ul style="list-style-type: none"> • implémenter l'analyse de conformité et de sécurité • empêcher la dérive en utilisant des outils de gestion de configuration • automatiser la gestion de la configuration à l'aide de la configuration de l'état souhaité de PowerShell (DSC) • automatiser la gestion de la configuration à l'aide d'un agent de VM en utilisant des extensions de script personnalisées • mettre en place un pipeline automatisé pour inspecter la sécurité et la conformité 			

Mettre en œuvre un processus de feedback continu					
	Recommander et concevoir des mécanismes de feedback	<ul style="list-style-type: none"> • concevoir des pratiques de mesure de satisfaction de l'utilisateur final (par exemple, Send a Smile, analytiques d'applications) • concevoir des processus pour capturer et analyser les commentaires des utilisateurs provenant de sources externes (par ex. Twitter, Reddit, Help Desk) • concevoir le routage des données de rapport de plantage de l'application client • recommander des outils et des technologies de surveillance • recommander des outils de suivi de l'utilisation du système et des fonctionnalités 	10-15%	Examen en ligne avec une variété de questions (cf détail plus haut) ** Environ 18 minutes (pour 9-11 questions) sont consacrées à cette compétence	Le taux de bonnes réponses doit être au global de 70% minimum
	Mettre en œuvre un processus pour acheminer les commentaires du système aux équipes de développement	<ul style="list-style-type: none"> • configurer l'intégration des rapports de plantage pour les applications clientes • développer des tableaux de bord de suivi et d'état • implémenter le routage pour les données de rapport de plantage de l'application client • implémenter des outils pour suivre l'utilisation du système, l'utilisation des fonctionnalités et le flux • intégrer et configurer les systèmes de tickets avec le système de gestion du travail de l'équipe de développement (par exemple, connecteur de gestion du service IT, gestion ServiceNow Cloud, les articles de travail de App Insights) 			
	Optimiser les mécanismes de feedback	<ul style="list-style-type: none"> • analyser les alertes pour établir une base de référence • analyser la télémétrie pour établir une base de référence • effectuer des révisions en direct du site et recueillir des commentaires sur les pannes du système • effectuer un réglage continu pour réduire les alertes inutiles ou non exploitables 			