

Designing and Implementing Microsoft DevOps solutions



Przeznaczenie szkolenia

Szkolenie skierowane do osób zainteresowanych podniesieniem swojej wiedzy i umiejętności w zakresie realizacji procesów DevOps oraz do osób zainteresowanych zdaniem egzaminu certyfikacyjnego Microsoft Azure DevOps Solutions.

Kurs polecany:

- deweloperom
- administratorom biorącym udział w projektach związanych z tworzeniem nowoczesnych aplikacji.

Kurs obejmuje takie zagadnienia jak:

- korzystanie z kontroli kodu źródłowego, skalowanie Git dla przedsiębiorstwa oraz wdrażanie i zarządzanie infrastrukturą.
- wdrażanie ciągłej integracji w potoku Azure DevOps, zarządzanie jakością kodu i zasadami bezpieczeństwa oraz wdrażaniem strategii budowania kontenerów.
- projektowanie strategii uwalniania, konfigurowanie przepływu pracy zarządzania wersjami i zaimplementowanie odpowiedniego wzoru wdrożenia.
- projektowanie strategii zarządzania zależnościami i zarządzanie bezpieczeństwem i zgodnością.
- wdrażanie infrastruktury zarządzania kodami i konfiguracją, zapewnienie infrastruktury Azure za pomocą popularnych narzędzi automatyzacji, wdrażanie infrastruktury aplikacji przy użyciu różnych usług platformy Azure i metod wdrażania, integrowanie zewnętrznych narzędzi do wdrażania z platformą Azure, takie jak Chef i Puppet, by włączyć zgodność i bezpieczeństwo do potoku wydania.
- projektowanie mechanizmów sprzężenia zwrotnego systemu, wdrażanie procesu przekazywania opinii systemowych zespołom programistycznym i optymalizowanie mechanizmów sprzężenia zwrotnego.
- planowanie transformacji, wybranie projektu i stworzenie struktury zespołu, rozwijanie strategii jakości i bezpieczeństwa, planowanie migracji i konsolidacji artefaktów i kontroli źródeł



Korzyści wynikające z ukończenia szkolenia

Uzyskanie wiedzy i praktycznych umiejętności w zakresie Office 365.

W tym zapoznanie się z:

- Zaletami korzystania z kontroli źródła, migracją z TFVC do Git, skalowaniem Git dla Enterprise DevOps, wdrażaniem i zarządzaniem infrastrukturą kompilacji, zarządzaniem konfiguracją aplikacji i przechowywaniem danych wrażliwych, mobilną strategią DevOps
- Dlaczego ciągła integracja ma znaczenie, wdrażaniem ciągłej integracji za pomocą Azure DevOps, konfigurowaniem kompilacji i dostępnych opcji, tworzeniem przepływów pracy zautomatyzowanej, integracją innych narzędzi do kompilacji z platformą Azure DevOps, tworzeniem hybrydowych procesów budowania, jakości kodu i sposobach jego pomiaru, wykrywaniem podejrzanego kodu, integrowaniem automatycznych testów jakości kodu, raportowaniem dotyczącym zasięgu kodu podczas testowania, narzędziami do mierzenia długu technicznego, wykrywaniem problemów z licencjami Open Source i innymi kwestiami licencjonowania, implementowaniem strategii budowania kontenerów
- Różnicami pomiędzy wydaniem a wdrożeniem, zdefiniowaniem komponentów potoku wydania, o rzeczach na które należy zwrócić uwagę kiedy projektujemy strategię wydania, klasyfikowaniem wydania w porównaniu z procesem wydawania i kontrolowaniem ich jakości, radzenia sobie z uwagami do wydania i dokumentacją, wzorcami wdrażania, zarówno w sensie tradycyjnym, jak i we współczesnym znaczeniu, wybraniem narzędzi do zarządzania wydaniem, terminologią używaną w Azure DevOps i innych narzędziach do zarządzania wydaniem, zrozumienia co to Buduj i wydaj, klasyfikacją agenta, kolejką agentów i pulą agentów, potrzebą wielu zadań zwolnienia w jednym z potoków wydań, rozróżnianiem zadań zwolnienia dla wielu agentów i wielu konfiguracji, wykorzystaniem zmiennych wersji i zmiennych etapowych w potoku wydawania, zdalnym wdrażaniem w środowisku przy użyciu połączenia usługowego, osadzeniem testowania w przygotowaniu, różnymi sposobami sprawdzania kondycji potoku i uwalniania przy użyciu, alertów, haków serwisowych i raportów, tworzeniem bramki zwolnienia, wzorcami wdrażania, implementowaniem wdrożenia Blue Green, wdrożeniem Canary Release, implementowaniem wdrażania progresywnej ekspozycji
- Narzędziami i praktykami zarządzania składnikami, uogólnianiem wspólnych pakietów umożliwiających udostępnianie i ponowne użycie, sprawdzaniem bazy kodów do identyfikacji zależności kodu, które można przekonwertować na pakiety, identyfikowaniem i polecaniem standardowych typów i wersji pakietów w całym rozwiązaniu, istniejącymi potokami do budowania refaktora w celu wdrożenia strategii wersji, która publikuje pakiety, zarządzaniem bezpieczeństwem i zgodnością, sprawdzaniem pakietów oprogramowania open source pod kątem bezpieczeństwa i zgodności z licencją, by dostosować je do standardów korporacyjnych, konfigurowaniem budującego potoku, by uzyskać dostęp do pakietu bezpieczeństwa i oceny licencji, konfigurowaniem bezpiecznego dostępu do kanałów pakietowych
- Zastosowaniem infrastruktury i konfiguracji jako zasady kodu, wdrażaniem infrastruktury i zarządzanie nią przy użyciu technologii automatyzacji firmy Microsoft, takich jak szablony ARM, PowerShell i interfejs wiersza polecenia Azure, modelami i usługami wdrażania dostępnymi w usłudze Azure, wdrażaniem i konfiguracją klastra Managed Kubernetes, rozmieszczaniem i konfigurowaniem infrastruktury przy użyciu narzędzi i usług innych firm za pomocą platformy Azure, takich jak Chef, Puppet, Ansible, SaltStack i Terraform, strategią infrastruktury i konfiguracją oraz odpowiednimi zestawami narzędzi dla potoku wydań i infrastruktury aplikacji, wdrażaniem zgodności i bezpieczeństwa w infrastrukturze aplikacji

- Praktykami projektowymi do pomiaru satysfakcji użytkownika końcowego, projektowaniem procesów do przechwytywania i analizy opinii użytkowników ze źródeł zewnętrznych, projektowaniem routingu dla danych raportu awarii aplikacji klienta, polecanymi narzędziami i technologiami monitorowania, zalecanymi narzędziami do śledzenia wykorzystania systemu i funkcji, konfiguracją integracji raportów awarii dla aplikacji klienckich, opracowywaniem panelów monitorowania i stanu, implementacją routingu dla danych raportu awarii aplikacji klienta, implementacją narzędzi do śledzenia wykorzystania systemu, wykorzystaniem funkcji i przepływu, integracją i konfiguracją systemów biletowych z systemem zarządzania pracą zespołu programistycznego, analizą alertów do ustalenia linii bazowej, analizą telemetry do ustalenia linii bazowej, wykonaniem aktualnych recenzji witryn i przechwytywaniem opinii o awariach systemu, wykonaniem ciągłego strojenia, by zmniejszyć nieważne lub nie nadające się do działania ostrzeżenia
- Planowaniem transformacji ze wspólnymi celami i osiami czasu, wybieranie projektu i określenie wskaźników projektu oraz wskaźników KPI, tworzeniem zespołu i sprawną strukturę organizacyjną, opracowaniem strategii jakości projektu, zaplanowaniem bezpiecznej praktyki programistycznej i zasady zgodności, migracją i konsolidacją artefaktów, migracją i integracją środków kontroli źródła



Metoda egzaminowania

Egzamin w formie **on-line**. Zapis na stronie <https://home.pearsonvue.com/Clients/Microsoft.aspx>
<https://www.altkomakademia.pl/vouchery-microsoft-promocja/>



Opis egzaminu

Po kursie **AZ-400** można przystąpić do certyfikowanego egzaminu **Microsoft**. Szczegóły na: <https://docs.microsoft.com/en-us/learn/certifications/exams/az-400>



Oczekiwane przygotowanie słuchaczy

- Podstawowa wiedza na temat platformy Azure, kontroli wersji, rozwoju oprogramowania Agile oraz podstawowych zasad tworzenia oprogramowania. Wskazane jest doświadczenia w organizacjach dostarczających oprogramowanie
- Zalecane jest doświadczenie w pracy w środowisku IDE, a także pewną wiedzę na temat portalu Azure
- Umiejętność korzystania z anglojęzycznych materiałów
- szkolenia poprzedzające: **AZ-104, MS 10961**

Dla zwiększenia komfortu pracy oraz efektywności szkolenia zalecamy skorzystanie z dodatkowego ekranu. Brak dodatkowego ekranu nie jest przeciwwskazaniem do udziału w szkoleniu, ale w znaczący sposób wpływa na komfort pracy podczas zajęć

Informacje oraz wymagania dotyczące uczestniczenia w szkoleniach w formule zdalnej dostępne na:

<https://www.altkomakademia.pl/distance-learning/#FAQ>



Język szkolenia

- **Szkolenie:** polski
- **Materiały:** angielski

Szkolenie obejmuje

* **podręcznik** w formie elektronicznej dostępny na platformie:

<https://learn.microsoft.com/pl-pl/training/>

* dostęp do portalu słuchacza AltKom Akademii

Czas trwania

4 dni / 28 godzin

Agenda szkolenia

1. Implementing DevOps Development Processes
2. Pierwsze kroki z kontrolą źródła
 - Co to jest kontrola źródła?
 - Korzyści z kontroli źródła
 - Rodzaje systemów kontroli źródła
 - Wprowadzenie do **Azure Repos**
 - Migracja z **TFVC** do Git
 - Uwierzytelnianie do repozytorium **Git**

3. Skalowanie git dla DevOps dla przedsiębiorstw

- Jak zorganizować repozytorium git
- Rozgałęzione przepływy pracy Git
- Współpraca z wnioskami ściąganyymi
- Dlaczego dbać o **GitHooks**?
- Wspieranie wewnętrznego otwartego źródła
- Wersja Git
- Projekty publiczne
- Pliki w Git

4. Wdrażanie i zarządzanie infrastrukturą kompilacji

- Koncepcja potoków w DevOps
- Potoki w Azure
- Ocena użycia hostowanych prywatnych agentów
- Pule agentów
- Potokii i współbieżność
- Projekty Azure DevOps i **Open Source**
- Azure Pipelines YAML vs Visual Designer
- Konfiguracja agentów prywatnych
- Integracja Jenkinsa z Azure Pipelines
- Integracja zewnętrznego źródła sterowania z Azure Pipelines
- Analiza i integracja wielostopniowej kompilacji Docker

5. Zarządzanie konfiguracją aplikacji i tajnikami

- Wprowadzenie do bezpieczeństwa
- Wdrożenie bezpiecznego i zgodnego proces rozwoju
- Ponowne przeanalizowanie danych konfiguracji aplikacji
- Zarządzanie tajnikami, tokenami i certyfikatami
- Wdrożenie narzędzi do zarządzania bezpieczeństwem i zgodnością w przygotowaniu

6. Wdrożenie mobilnej strategii DevOps

- Wprowadzenie do Mobile DevOps
- Wprowadzenie do Centrum aplikacji **Visual Studio**
- Zarządzanie mobilnymi zestawami urządzeń docelowych i grupami dystrybucyjnymi
- Zarządzanie docelowymi zestawami urządzeń testujących interfejs użytkownika
- Urządzenia testujące do wdrożenia
- Tworzenie publicznych i prywatnych grup dystrybucyjnych

7. Implementing Continuous Integration

8. Wdrażanie ciągłej integracji w potoku Azure DevOps

- Omówienie ciągłej integracja
- Wdrażanie strategii tworzenia

9. Zarządzanie jakością kodu i strategię bezpieczeństwa
 - Zarządzanie jakością kodu
 - Zarządzanie zasadami bezpieczeństwa
10. Wdrażanie strategii tworzenia kontenerów
 - Wdrażanie strategii budowania kontenerów
11. Implementing Continuous Delivery
12. Projekt strategii Release
 - Wprowadzenie do ciągłej dostawy
 - Zalecenia dotyczące strategii uwalniania
 - Budowanie potoku o wysokiej jakości
 - Wybór wzorca wdrażania
 - Wybór odpowiedniego narzędzia do zarządzania wersjami
13. Konfiguracja przepływu pracy zarządzania wersjami
 - Tworzenie potoku wydania
 - Dostarczanie i konfigurowanie środowisk
 - Zarządzanie i modularyzacja zadań i szablonów
 - Integracja tajników z potokiem wydania
 - Konfiguracja zautomatyzowanej integracji i automatyzacji testów funkcjonalnych
 - Automatyzacja inspekcję kondycji
14. Implementacja odpowiedniego wzoru wdrożenia
 - Wprowadzenie do wzorców wdrażania
 - Implementacja wdrożenia Blue Green
 - Przełączniki funkcji
 - Canary Release
 - Dark Launching
 - Testowanie AB
 - Progresywna ekspozycja
15. Implementing Dependency Management
16. Projektowanie strategii zarządzania zależnościami
 - Wprowadzenie
 - Pakowanie zależności
 - Zarządzanie pakietami
 - Wprowadzenie strategii wersjonowania
17. Zarządzanie bezpieczeństwem i zgodnością
 - Wprowadzenie
 - Zabezpieczenie pakietu
 - Oprogramowanie open source
 - Integracja skanowań licencji i luk w zabezpieczeniach

- Sprawdzanie pakietów oprogramowania open source pod kątem bezpieczeństwa i zgodności z licencją, aby dostosować je do standardów korporacyjnych
- Konfiguracja potoku budującego, by uzyskać dostęp do pakietu bezpieczeństwa i oceny licencji
- Konfiguracja bezpiecznego dostępu do kanałów pakietowych

18. Implementing Application Infrastructure

19. Infrastruktura i konfiguracja Narzędzia Azure

- Infrastruktura jako kod i zarządzanie konfiguracją
- Tworzenie zasobów Azure Resources przy użyciu szablonów ARM
- Tworzenie zasobów platformy Azure za pomocą interfejsu wiersza polecenia Azure
- Tworzenie zasobów Azure za pomocą Azure PowerShell
- Dodatkowe narzędzia automatyzacji
- Kontrola wersji

20. Modele i usługi wdrażania platformy Azure

- Modele i opcje wdrażania
- Usługi Azure Infrastructure-as-a-Service (IaaS)
- Azure Automation z DevOps
- Żądana konfiguracja stanu (DSC)
- Usługi Azure Platform-as-a-Service (PaaS)
- Azure Service Fabric

21. Tworzenie i zarządzanie infrastrukturą usługową Kubernetes

- Usługa Azure Kubernetes

22. Narzędzia innych firm i narzędzia Open Source dostępne z Azure

- Chef
- Puppet
- Ansible
- Cloud-Init
- Terraform

23. Wdrażanie zgodności i bezpieczeństwa w swojej infrastrukturze

- Zasady bezpieczeństwa i zgodności z DevOps
- Centrum zabezpieczeń Azure

24. Podsumowanie kursu

25. Implementing Continuous Feedback

26. Rekomendowane i projektowane mechanizmy sprzężenia zwrotnego systemu

- Wewnętrzna pętla
- Ciągły eksperyment doświadczalny
- Praktyki projektowe do pomiaru satysfakcji użytkownika końcowego
- Projektowanie procesów do przechwytywania i analizowania opinii użytkowników
- Proces projektowania w celu zautomatyzowania analizy aplikacji

27. Wdrażanie procesów przekazywania opinii systemowych do zespołów programistycznych
 - Implementacja narzędzi do śledzenia wykorzystania systemu, wykorzystania funkcji i przepływu
 - Implementacja routingu dla danych raportu awarii aplikacji mobilnych
 - Opracowanie kokpitów monitorowania i stanu
 - Integracja i konfiguracja systemów biletowych
28. Optymalizacja mechanizmów sprzężenia zwrotnego
 - Inżynieria niezawodności witryny
 - Analiza telemetrii do ustalenia linii bazowej
 - Wykonanie ciągłego strojenia, by zmniejszyć nieważne lub nie nadające się do działania ostrzeżenia
 - Analiza alertów, by ustalić linię bazową
 - Bezpłomieniowe postmortemy i uczciwa kultura
29. Designing a DevOps Strategy
30. Planowanie DevOps
 - Planowanie transformacji
 - Wybór projektu
 - Struktury zespołu
31. Planowanie jakości i bezpieczeństwa
 - Planowanie strategii jakości
 - Planowanie bezpiecznego rozwoju
32. Migracja i konsolidacja artefaktów i narzędzi
 - Migrowanie i konsolidacja artefaktów
 - Migracja i integracja kontroli źródła