

Załącznik nr 1
do Zapytania ofertowego ILZ/261-0142/19/2801

1. Szkolenie – Podstawy Powershell
3 x 7 godzin (3 dni) (21 godzin)
20 osób, 2 grupy

wymagane miejsce realizacji szkolenia w Olsztynie

Izba Administracji Skarbowej w Olsztynie dysponuje własną, 13 osobową salą szkoleniową

- zakres szkolenia oparty na szkoleniu „Automatyzacja administracji z Windows Power Shell (MS-10961)”,
- szkolenie prowadzone w formie wykładu i prezentacji min. 2/3 czasu oraz ćwiczeń praktycznych max. 1/3 czasu szkolenia
- materiały dydaktyczne w formie pisemnej w języku polskim obejmujące część teoretyczną oraz praktyczną szkolenia
- zamawiający nie wymaga autoryzowanych materiałów szkoleniowych oraz certyfikatu Microsoft

L.p.	Obszar	Zakres tematyczny
1	Początek pracy z Windows PowerShell	<ul style="list-style-type: none"> • informacje ogólne i historia • wyszukiwanie i uruchamianie komend
2	Praca z Windows PowerShell Pipeline	<ul style="list-style-type: none"> • praca z wykorzystaniem pipeline • eksport, import i konwertowanie danych • filtrowanie obiektów z pipeline • wyliczanie obiektów przy wykorzystaniu pipeline
3	Zrozumienie funkcjonowania pipeline w Windows PowerShell	<ul style="list-style-type: none"> • przekazywanie danych w pipeline wg wartości • przekazywanie danych w pipeline wg nazwy właściwości
4	Używanie PSProviders i PSDrives	<ul style="list-style-type: none"> • zrozumienie PSProviders i PSDrives • stosowanie PSDrives
5	Formatowanie efektu wyjściowego	<ul style="list-style-type: none"> • stosowanie formatowania podstawowego • stosowanie formatowania zaawansowanego • przekierowanie sformatowanego efektu wyjściowego
6	Stosowanie Windows Management Instrumentation / Common Information Model (WMI/CIM)	<ul style="list-style-type: none"> • zrozumienie WMI/CIM • wysyłanie kwerend dzięki WMI/CIM • dokonywanie zmian przy użyciu WMI/CIM
7	Przygotowanie do wykonywania skryptów	<ul style="list-style-type: none"> • wykorzystanie zmiennych • aspekty bezpieczeństwa podczas wykonywania skryptów • praca z poświadczeniami alternatywnymi
8	Przejsie od komendy do skryptu do modułu	<ul style="list-style-type: none"> • przejście od komendy do skryptu • przejście od skryptu do funkcji do modułu • dołączenie podstawowej obsługi błędów • stosowanie podstawowych konstrukcji skryptowych
9	Administracja komputerami zdalnymi	<ul style="list-style-type: none"> • podstawowe funkcje zdalne • używanie sesji zdalnych • funkcje zdalne dla administracji delegowanej

2. Szkolenie – Windows PowerShell Scripting and Toolmaking (oparte na szkoleniu MS 55039)

5 x 7 godzin (5 dni) (35 godzin)

1 osoba

- zamawiający nie wymaga autoryzowanych materiałów szkoleniowych oraz certyfikatu Microsoft

L.p.	Obszar	Zakres tematyczny
1	Projektowanie narzędzia	<ul style="list-style-type: none"> Narzędzia do wykonywania jednej rzeczy Uniwersalne narzędzia Wbudowane narzędzia
2	Budowa podstawowych funkcji i modułów	<ul style="list-style-type: none"> Tworzenie modułu Sprawdzanie wymagań Uruchamianie nowych komend
3	Dodawanie CmdletBinding i parametrów	<ul style="list-style-type: none"> Omówienie CmdletBinding i Common Parametes Akceptowanie parametrów w potoku Parametry obowiązkowe Walidacja parametrów Aliasy parametrów
4	Obiekt jako wynik	<ul style="list-style-type: none"> Składanie informacji Konstruowanie i wysyłanie danych wyjściowych Szybkie testy
5	Zmiana podejścia do skryptowania	<ul style="list-style-type: none"> Omawianie skryptów Aktualizacja skryptu
6	Wyjście danych: szczegółowe, ostrzeżenia, informacyjne	<ul style="list-style-type: none"> Dane szczegółowe i ostrzeżenia Verbose Dane informacyjne
7	Obsługa błędów	<ul style="list-style-type: none"> Zrozumienie błędów i wyjątków Przechwytywanie wyjątku Obsługa wyjątków
8	Podstawy debugowania	<ul style="list-style-type: none"> Write-Debug Set-PSBreakpoint PowerShell ISE
9	Więcej o parametrach	<ul style="list-style-type: none"> Pozycje parametrów Walidacja Zestaw parametrów Aliasy CmdletBinding
10	Pisanie pełnej pomocy	<ul style="list-style-type: none"> Zewnętrzna pomoc Pomoc online Tematy pomocy „About” Tworzenie pomocy którą można uaktualniać
11	Testowanie kodu	<ul style="list-style-type: none"> Wzorce do testowania Rozszerzanie testów Korzystanie z Pester
12	Typy wyjściowe	<ul style="list-style-type: none"> Zrozumienie typów Rozszerzanie obiektu Korzystanie z Update-TypeData
13	Analizowanie skryptu	<ul style="list-style-type: none"> Wykonywanie podstawowej analizy Analizowanie analizy
14	Publikowanie narzędzia	<ul style="list-style-type: none"> Tworzenie Manifestu Publikowanie w galerii PowerShell Publikowanie do prywatnych repozytoriów
15	Podstawy sterowania skryptem: tworzenie menu	<ul style="list-style-type: none"> Tworzenie menu Korzystanie z UIChoice

16	Funkcje proxy	<ul style="list-style-type: none">• Przykłady proxy• Tworzenie proxy• Modyfikowanie proxy• Dodawanie lub usuwanie parametrów
17	Praca z formatem XML	<ul style="list-style-type: none">• CliXML• Import formatu XML• ConvertTo-XML• Tworzenie natywnego XML od zera
18	Praca z danymi JSON	<ul style="list-style-type: none">• Konwersja na JSON• Konwersja z JSON
19	Praca z SQL Server	<ul style="list-style-type: none">• Terminologia SQL Server• Łączenie z serwerem i bazą• Pisanie zapytań• Uruchamianie zapytań• Invoke-SqlCmd

3. Szkolenie – PowerShell - Practical application for advanced users

2 x 7 godzin (2 dni) (14 godzin)

4 osoby

- zamawiający nie wymaga autoryzowanych materiałów szkoleniowych oraz certyfikatu Microsoft

L.p.	Obszar	Zakres tematyczny
1	Projektowanie narzędzi	<ul style="list-style-type: none">• Zasady projektowania narzędzi• Parametryzacja• Budowa podstawowych funkcji i modułów• Definiowanie obiektów na wyjściu
2	Analizowanie skryptów	<ul style="list-style-type: none">• Wykonywanie podstawowej analizy• Automatyczne poprawianie formatowania kodu
3	Połączenia zdalne	<ul style="list-style-type: none">• Połączenia typu “Ad-hoc” i oparte na tworzeniu sesji• Połączenia typu “jeden do wielu”• Przesyłanie lokalnych zmiennych do zdalnego komputera• Podłączanie i rozłączanie sesji
4	Budowanie interfejsu graficznego	<ul style="list-style-type: none">• Projektowanie i budowa interfejsu• Przykłady wykorzystanie klas .Net Framework do budowy interfejsu graficznego• Budowa własnego graficznego systemu pomocy do programu – opartego o JSON• Włączanie i wyłączanie części interfejsu w zależności od uprawnień
5	Konfiguracja i wdrożenie JEA	<ul style="list-style-type: none">• Przygotowanie pliku konfiguracyjnego roli JEA• Przygotowanie pliku konfiguracyjnego sesji JEA• Wdrożenie i przetestowanie konfiguracji JEA

4. Szkolenie – Wzorce projektowe w TypeScript
3 x 7 godzin (3 dni) (21 godzin)
3 osoby

L.p.	Obszar	Zakres tematyczny
1	Charakterystyka języka TypeScript w odniesieniu do standardu ES6+	
2	Narzędzia developerskie na platformie Node.js	<ul style="list-style-type: none">• Charakterystyka, architektura i zastosowania• Kompilator Babel i konwersja kodu do postaci JS• Statyczna analiza kodu z TSLint• Automatyzacja procesu budowania z WebPack
3	Wzorce modularnej budowy aplikacji	
4	Elementy programowania funkcyjnego	
5	Inżynieryjne wzorce projektowe	<ul style="list-style-type: none">• Kreacyjne: Singleton, Factory, Builder, Prototype• Strukturalne: Facade, Proxy, Adapter, Bridge, Decorator, Composite• Zachowań: Command, Observer, Iterator, State, Strategy
6	Wzorce związane z implementacją warstwy widoków	<ul style="list-style-type: none">• MVC, MVP, MVVM
7	Implementacja i zastosowania CQRS i EventSourcing	
8	Architektura aplikacji SPA	
9	Kontenery stanowe, Fux, Redux	

5. Szkolenie – Mikroserwisy w Node.js z wykorzystaniem Docker

3 x 7 godzin (3 dni) (21 godzin)

3 osoby

L.p.	Obszar	Zakres tematyczny
1	Mikroserwisy	<ul style="list-style-type: none"> • Architektura mikrosług i odniesienie do architektury monolitycznej • Zasady projektowania mikroserwisów, wydzielanie mikrosług • Projektowanie baz danych, warstwy dostępu do danych pod architekturę mikroserwisową • Kryterium stosowalności, mocne i słabe strony mikroserwisów • Typowe wzorce użycia
2	Platforma Node.js	<ul style="list-style-type: none"> • Charakterystyka i zastosowania • Zarządzanie pakietami • Konfiguracja i użycie narzędzi do kompilacji, budowania i testowania projektu
3	Konteneryzacja z Docker	<ul style="list-style-type: none"> • Architektura środowiska konteneryzacji • Praca z obrazami i kontenerami • Wykorzystanie Docker Compose w procesie uruchamiania wielu aplikacji
4	Wprowadzenie do tematu orkiestracji i charakterystyka popularnych narzędzi	
5	Implementacja mikroserwisów	<ul style="list-style-type: none"> • Architektura message-based z frameworkiem Seneca • Service discovery, message routing, load balancing z framwokiem Express i Hydra • Komunikacja pomiędzy mikrosługami, HTTP, WebSocket • Implementacja autentykacji i autoryzacji

6. Szkolenie – Programowanie reaktywne w TypeScript z użyciem RxJS

3 x 7 godzin (3 dni) (21 godzin)

3 osoby

L.p.	Obszar	Zakres tematyczny
1	Programowanie funkcyjne	<ul style="list-style-type: none"> • Założenia i idea programowania funkcyjnego • Funkcje wyższego rzędu • Zastosowania
2	Charakterystyka języka JavaScript	<ul style="list-style-type: none"> • Architektura i standard ES6+ • Praca z kodem asynchronicznym, Promises
3	ReactiveX	<ul style="list-style-type: none"> • Architektura zastosowania i dostępne implementacje • hot vs. cold observable • Użycie subject • Tworzenie strumieni zdarzeń, subskrypcja
4	Operatory ReactiveX	<ul style="list-style-type: none"> • Transformujące, filtrujące, łączące, warunkowe • Obsługi błędów • Zarządzanie strumieniami
5	Przetwarzanie danych z biblioteką RxJS	
6	Budowa Rich UI, wzorzec MVVM z biblioteką RxJS	
7	Refaktoryzacja kodu i włączenie biblioteki RxJS do istniejącego projektu	
8	Testowanie kodu opartego o RxJS	

7. Szkolenie – Zaawansowane aspekty języka JavaScript

3 x 7 godzin (3 dni) (21 godzin)

1 osoba

L.p.	Obszar	Zakres tematyczny
1	Zarządzanie pamięcią, Garbage Collector	
2	Architektura obiektu	<ul style="list-style-type: none"> • Kontekst this, binding • Inicjalizacja obiektów • Mechanizm dziedziczenia i prototype chain • Klonowanie obiektów • Interfejs getter/setter, stan obiektu • Enkapsulacja, metody implementacji prywatności danych
3	Definiowanie stanu obiektu, obiekty mutable i immutable, stan danych	
4	Modularna budowa kodu, import, export	
5	Elementy programowania funkcyjnego i kolekcje	<ul style="list-style-type: none"> • Generatory i iteratory • Filtrowanie, mapowanie, redukcja • Praktyczne wykorzystanie collection API • Weak collections, WeakMap, WeakSet
6	Praca z kodem asynchronicznym, Promise, EventLoop	
7	Wykorzystanie symboli	
8	Praktyki związane z wykorzystaniem obiektu Proxy	