

Wstęp

Technologie Internetowe

Projekt „SezAM wiedzy, kompetencji i umiejętności” jest współfinansowany przez Unię Europejską ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Wiedza Edukacja Rozwój

Połączenie przez Internet

- Łączą się dwie aplikacje działające na dwóch hostach (komputerach posiadających adres IP).
- Jedna z aplikacji czeka na żądanie od drugiej.
- Aplikacja czekająca na żądanie nazywana jest serwerem (ang. server) lub „nasłuchiwarem” (ang. Listener).
- Klient wysyła żądanie do konkretnej aplikacji na zdalnym hoście, host jest identyfikowany przez adres IP, aplikacja przez numer portu.
- Numer IP + numer portu identyfikuje punkt końcowy (endpoint) połączenia.
- Asocjacja to piątka: protokół, adres lokalny, port lokalny, adres zdalny, port zdalny.
- Gniazdo (półasocjacja)
 - (protokół, adres lokalny, port lokalny) albo,
 - (protokół, adres zdalny, port zdalny).



Modele komunikacji sieciowej

- Peer-to-peer
 - Wszystkie hosty mają te same uprawnienia.
- Klient-serwer
 - Serwer jest zwykle uruchomiany na hoście z publicznym adresem IP (chyba, że ma być osiągalny tylko lokalnie).
 - Klienci nie muszą mieć publicznych adresów IP.
- Pobliskacja-subskrypcja
 - Komunikacja za pośrednictwem serwera (brokera) z publicznym adresem IP.
 - Możliwość przesyłania wiadomości pomiędzy klientami bez publicznego adresu IP.

Zagadnienia sieciowe

- Bezpieczeństwo pracy w sieci.
- Przesyłanie danych między składowymi systemu rozproszonego.
- Formaty przesyłania danych.
- Obciążenie sieci.
- Łączność internetowa z urządzeniami mobilnym.
- Łączność u urządzeniami nie posiadającymi publicznego adresu IP.

Wieloplatformowość

- Przez Internet mogą przesyłać dane aplikacje programowane w różnych językach i działające pod kontrolą różnych systemów operacyjnych – Windows, Linux, Android, ...
- Języki programowania wyposażone są w mechanizmy (biblioteki) do łączenia przez Internet.
- Środowiska uruchomieniowe i platformy programistyczne działające w różnych systemach operacyjnych: Node.js (JavaScript), .NET Core (C#), Django (Python), ...



Aplikacja sieciowa

- **Aplikacja** – konkretny ze względu na oferowane użytkownikom funkcjonalność element oprogramowania użytkowego.
- **Aplikacja sieciowa** – aplikacja umieszczona na serwerze aplikacji;
 - użytkownik łączy się z aplikacją za pomocą przeglądarki internetowej,
 - aplikacja obsługuje żądania klienta,
 - przetwarzanie odbywa się na serwerze,
 - aplikacja dostarcza interfejs użytkownika,
 - protokół HTTP (metody GET, POST).



Usługa sieciowa

- Usługa sieciowa dostarcza metody sieciowe, które może wywoływać aplikacja klienta.
- API (ang. *Application Programming Interface*) – Interfejs programowania aplikacji, interfejs programistyczny aplikacji.
- Usługa sieciowa nie dostarcza interfejsu użytkownika.
- Usługi sieciowe nie są ograniczone do jednego systemu operacyjnego, ale są oparte na otwartych standardach.
- Protokół SOAP (ang. *Simple Object Access Protocol*) – komunikacja między klientem (konsumentem) i usługą sieciową odbywa za pomocą dokumentów bazujących na XML.
- REST (ang. *Representational State Transfer*) – styl architektury usług sieciowych wykorzystujących protokół HTTP (metody GET, POST, PUT, DELETE) i format danych JSON (JavaScript Object Notation).
- Można stosować i inne protokoły.

Techniki tworzenia aplikacji i usług internetowych

- Program w języku C, C#, Java, Perl, ... generujący strony lub usługi.
- Skrypty osadzone na stronie np. PHP, C# i Visual Basic.
- ASP.NET Web Forms, ASP.NET MVC, JSP; aplikacje sieciowe.
- AJAX – *Asynchronous JavaScript and XML*.
- Usługi SOAP – *Simple Object Access Protocol*.
- RESTFull API – Representational State Transfer.
- Node-RED – graficzne środowisko programistyczne, ważne zastosowanie to Internet rzeczy, szczególnie Raspberry Pi; pozwala tworzyć usługi i aplikacje.
- Programy w języku JavaScript działające na platformie Node.js lub inne techniki lekkie.
- Programy serwerowe (obsługujące żądania http) napisane w języku C#,

Narzędzia programistyczne wybrane do przedmiotu

- Środowiska programistyczne
 - Visual Studio, Visual Studio Code (wieloplatformowe), pozwalają na kodowanie w różnych językach, w tym C# i JavaScript, są dostępne bezpłatnie,
 - Node-Red, działa w przeglądarce internetowej.
- Podstawowy język programowania C# z uwagi na jego znajomość.
- JavaScript powszechnie wykorzystywany w aplikacjach internetowych, w tym w aplikacjach Node-RED.



Fundusze
Europejskie
Wiedza Edukacja Rozwój



Rzeczpospolita
Polska

Unia Europejska
Europejski Fundusz Społeczny



Platforma .NET

- Języki programowania C#, VB, J#, C++.
- Wspólne środowisk uruchomieniowe (ang. *Common Language Runtime* - CLR).
- Wspólny system typów.
- Biblioteka klas.
- Pozwala kompilować i uruchamiać aplikacje Windows Forms oraz sieciowe aplikacje ASP.NET Web Form, MVC i usługi Web Services.
- Najnowsza wersja 4.8.



Visual Studio

- Zintegrowane środowisko programistyczne firmy Microsoft.
- Wersja w lab. C-135A: 10 (2010) związana z platformą .NET4.0.
- Najnowsza wersja 15 (2019) związana z platformą .NET4.7.
- Visual Studio składa się z narzędzi typu RAD (ang. *Rapid Application Development*) służących do „szybkiego tworzenia aplikacji”.
- Podstawowe zastosowania
 - tworzenie aplikacji Windows Forms,
 - tworzenie aplikacji i usług sieciowych ASP.NET Web Forms i MVC.
- Główne języki C#, C++, VisualBasic, J#.
- Dostępna bezpłatna wersja *Community Edition*.
- Umożliwia programowanie w innych językach w tym Java Script na platformę Node.js

Strona internetowa (WWW)

- Dokument tekstowy, który może być zinterpretowany i wyświetlony przez przeglądarkę.
- Strony internetowe są najczęściej tworzone w języku HTML (ang. *HyperText Markup Language*).
- Aktywna (dynamiczna) strona internetowa to strona tworzona w części lub w całości przez program działający na serwerze internetowym.
- Do tworzenia stron aktywnych wykorzystywane są technologie ASP, ASP.NET Web Forms i MVC, JSP, PHP,



Protokół HTTP

- HTTP (ang. *Hypertext Transfer Protocol*) protokół przesyłania dokumentów hipertekstowych.
- Protokół określa format żądania klienta wysyłanego na serwer i format odpowiedzi serwera.
- Żądanie poza adresem zwrotnym może zawierać parametry przekazywane do aplikacji serwerowej np. dane formularza przesłane metodą GET lub POST.
- Protokół jest bezstanowy – nie zapamiętuje stanu sesji z klientem.



Serwer WWW

- Program działający na serwerze internetowym.
- Obsługuje żądania HTTP, odsyła do klienta stronę HTML lub dane w dowolnym formacie np. XML, JSON, YAML, ...
- Odpowiedź może być statycznym plikiem HTML zapisanym na dysku lub kodem wygenerowanym przez program umieszczony na serwerze.
- Najczęściej używane serwery WWW to
 - Apache – Open Source, Linux, Windows,
 - Internet Information Services (IIS) – tylko Microsoft Windows.

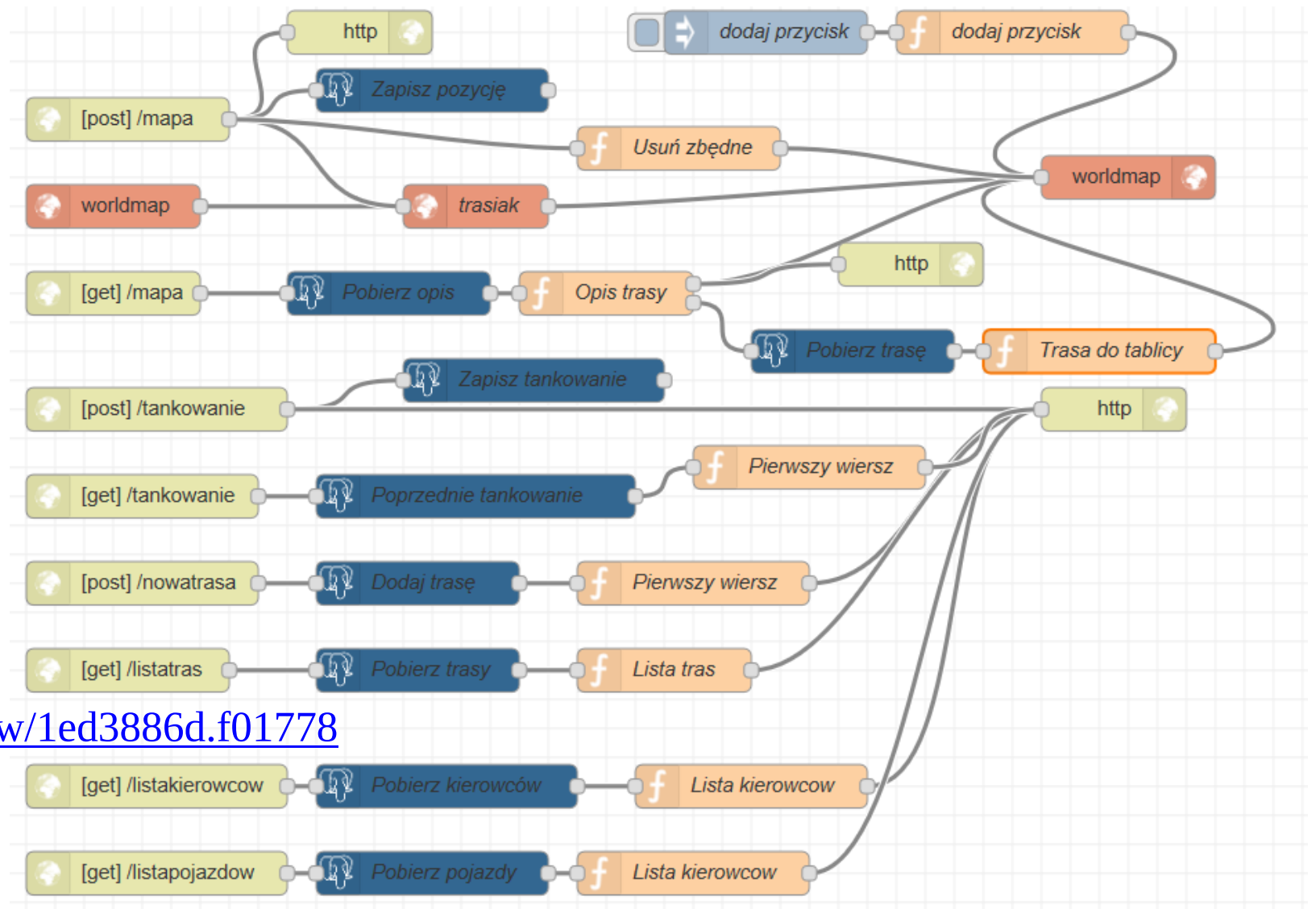


Node-RED

- Node-RED to graficzne środowisko programistyczne, w którym za pomocą grafu przepływu danych można tworzyć algorytm aplikacji.
- Projekt został zapoczątkowany przez IBM, przekazany społeczności – obecnie rozwija się bardzo dynamicznie.
- Pierwotnie był dedykowany do Internetu Rzeczy, ale obecnie ma o wiele szersze zastosowania.
- Node-RED działa na platformie node.js i jest programowany w języku JavaScript a JSON jest naturalnym sposobem zapisu danych.
- Programowanie odbywa się za pomocą przeglądarki internetowej.
- W środowisku Node-RED bardzo łatwo tworzy się usługi REST.
- W Node-RED dostępna jest mapa sygnowana przez IBM o podobnej funkcjonalności co mapa Google, mapa ma wiele przydatnych warstw w tym OSM, Esri, Esri Satellite,
- Mapa nie pyta o numer karty kredytowej, a można z niej korzystać bez ograniczeń, póki co.
- Platformę Node-RED można instalować na komputerach Windowsowych i Linuksowych, jest dedykowana do Raspberry Pi, gdzie ma łatwy dostęp do GPIO i wszystkich interfejsów.



Diagram przepływu danych Node-RED



<http://lukan.sytes.net:1880/#flow/1ed3886d.f01778>



Możliwości Node-RED

- Komunikacja sieciowa
 - TCP, UDP, Http, websocket, MQTT, Email, SMS, Twiter, Telegram,
- Czytanie wskazań czujników i przetworników A/C, włączanie i wyłączanie urządzeń – przede wszystkim na Raspberry Pi.
- Komunikacja szeregową – serial port.
- Dostęp do baz danych SQL i NoSQL.
- Korzystanie z bibliotek uczenia maszynowego SciKit-Learn, Tensorflow i
- Współpraca z IBM Watson, Google services.
- Tworzenie graficznego interfejsu użytkownika, zaawansowane wykresy
<https://flows.nodered.org/node/node-red-contrib-chartjs>.

Tworzenie usług REST w Node-RED

- Praktycznie gotowe punkty końcowe usług, wymagają tylko prostej konfiguracji.
- Bardzo proste debugowanie.
- Łatwy dostęp do stanu wszystkich elementów diagramu.
- Przykłady
 - Dioda,
 - Bme280,
 - MQTT,
 - Pomiar wilgotności,
 - Czas reakcji.



Formaty kodowania danych stosowane w usługach sieciowych

JSON

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumber": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "fax",
      "number": "646 555-4567"
    }
  ],
  "gender": {
    "type": "male"
  }
}
```

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<root>
  <firstName>John</firstName><lastName>Smith</lastName>
  <age>25</age>
  <address>
    <streetAddress>21 2nd Street</streetAddress>
    <city>New York</city>
    <state>NY</state>
    <postalCode>10021</postalCode></address>
  <phoneNumber>
    <type>home</type><number>212 555-1234</number></phoneNumber>
    <phoneNumber>
      <type>fax</type><number>646 555-4567</number></phoneNumber>
    <gender><type>male</type></gender>
  </root>
```



Formaty kodowania danych stosowane w usługach sieciowych

JSON

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumber": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "fax",
      "number": "646 555-4567"
    }
  ],
  "gender": {
    "type": "male"
  }
}
```

YAML

```
firstName: John
lastName: Smith
age: 25
address:
  streetAddress: 21 2nd Street
  city: New York
  state: NY
  postalCode: '10021'
phoneNumber:
  - type: home
    number: 212 555-1234
  - type: fax
    number: 646 555-4567
gender:
  type: male
```

Zastosowanie Node-RED

- Prototypowanie i budowa systemów rozproszonych.
- Prototypowanie i budowa systemów Internetu Rzeczy.
- Prototypowanie i budowa systemów wbudowanych.
- Node-RED jest przede wszystkim środowiskiem testowym, ale sprzedawane są rozwiązania przemysłowe oparte na Node-RED.