

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

WiFi Car Revolutions (partea a II-a)

Recunoașterea limbajului natural a devenit încet, încet o funcție obișnuită prin care putem comanda diverse dispozitive inteligente precum telefonul mobil sau calculatorul de bord al automobilul personal. Este posibil să implementăm o funcționalitate asemănătoare pentru WiFi Car? Este posibil să revoluționăm telecomanzile RC uzuale și să adăugăm comandă prin limbaj natural?



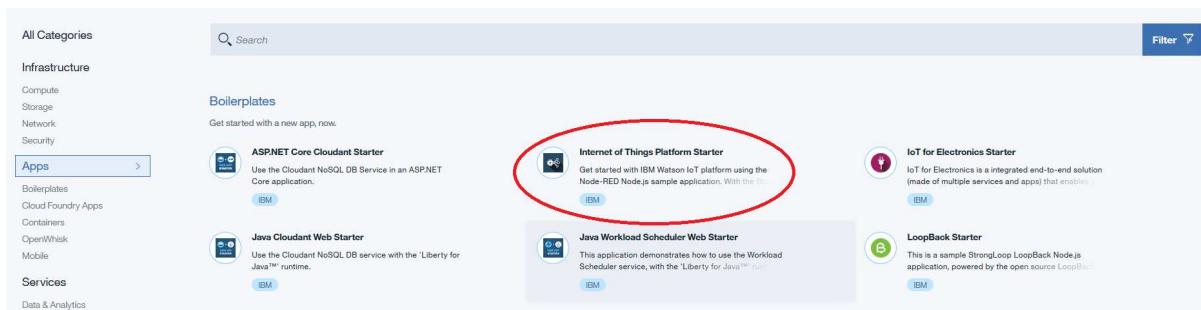
Da!

Pentru a face acest lucru vom exploata conexiunea de rețea, prezentată în prima parte, pentru a conecta WiFi Car la un serviciu cloud de recunoaștere vocală. Puterea de calcul a plăcii de dezvoltare nu permite implementarea algoritmului de recunoaștere vocală local, pe placa de dezvoltare a WiFi Car, din această cauză vom apela la resursele de calcul a unui sistem de cloud. Sistemul de cloud utilizat este IBM BlueMix™, serviciu cloud comercial dar cu posibilitatea de evaluare gratuită de 30 de zile fără a fi nevoie să introducem informații legate de un card bancar. Mai mult decât atât, în anumite centre universitare din țară accesul studenților și profesorilor este gratuit nelimitat. Pentru familiarizarea cu serviciul de cloud IBM BlueMix™ se poate revedea și lecția [How sunny is the Blue? \(I, II\)](#). Înregistrarea pe platformă se face la adresa:

Sign up for IBM Bluemix

<https://console.ng.bluemix.net/registration/>

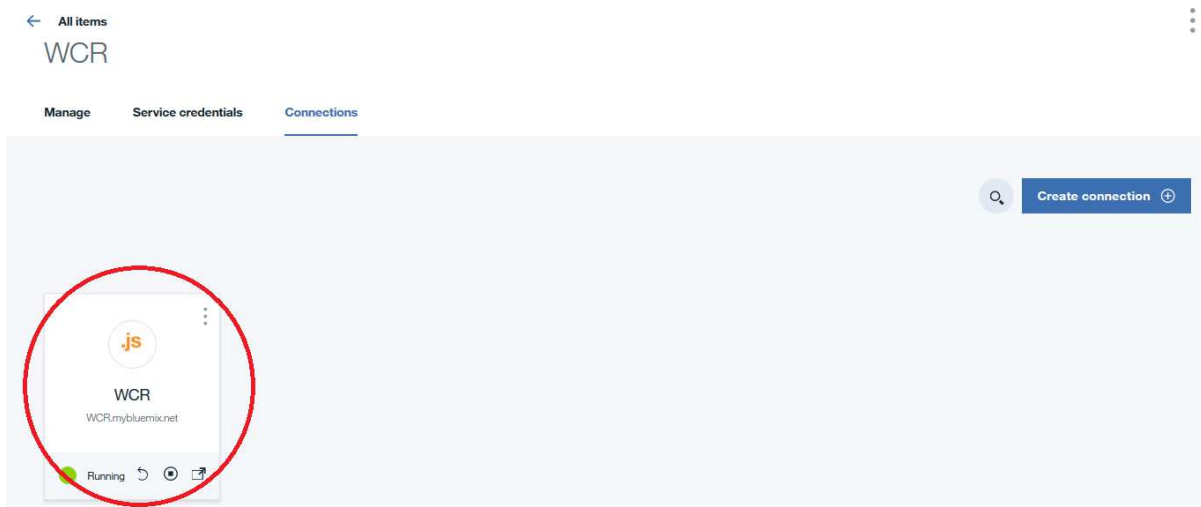
După înregistrare și conectare la platformă se va naviga în zona de aplicații și se va crea o aplicație nouă (Create Application) de tipul Boilerplates / Internet of Things Platform Starter:



Se va alege un nume pentru aplicație (în cadrul lecției se va utiliza numele *WCR* dar puteți alege orice alt nume) și se va da comanda (*Create*) de alocare de resurse cloud pentru această aplicație nouă. Alocarea de resurse poate dura câteva minute sau zeci de minute. Finalizarea operației este marcată de schimbarea stării aplicației în *Running*.



În plus, trebuie să adăugăm aplicației noastre un serviciu de tip Speech to Text (serviciul de recunoaștere vocală): Services / Watson / Speech to Text. Noul serviciu trebuie asociat cu aplicația creată anterior (*Connections / Create connection*):



După definirea conexiunii aplicația *WCR* se va restarta. Pentru a putea conecta WiFi Car la aplicația nou creată trebuie să mergem în Dashboard-ul de administrare a serviciului *WCR-iotf-service* (*Connect your devices / Launch dashboard*). Comanda ne va direcționa către portalul IBM Watson IoT Platform unde vom crea un nou dispozitiv (*Devices / Add Device*). Nu trebuie să completăm decât tipul dispozitivului (în exemplul nostru *Adafruit_HUZZAH*) și numele dispozitivului (în exemplul nostru *WCR*). La finalizarea adăugării noului dispozitiv trebuie să notăm informațiile de autentificare furnizate de platformă pentru a putea să le folosim în programul

dispozitivului (*Organization ID, Device Type, Device ID, Authentication Method și Authentication Token*).

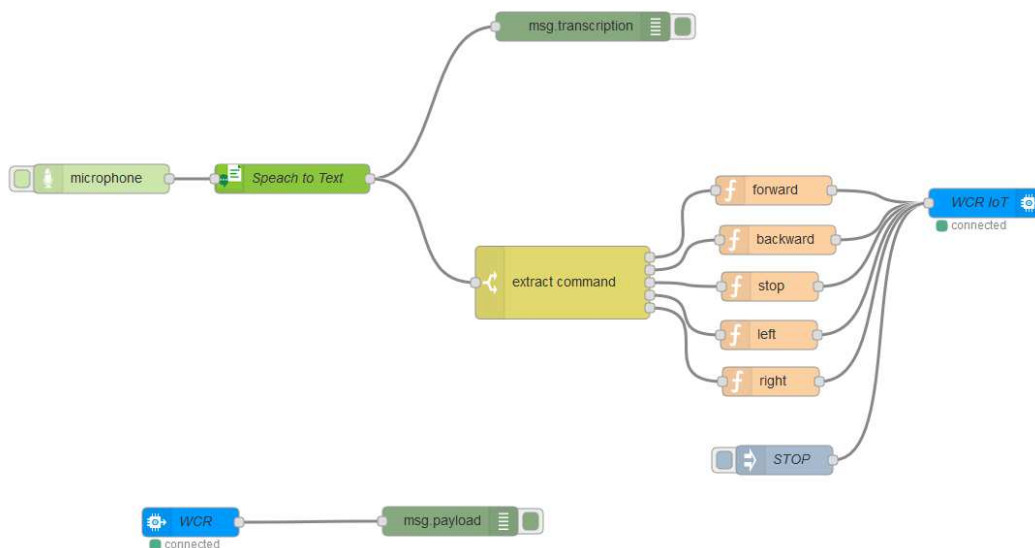
Devices

Browse | Diagnose | Action | Device Types | Manage Schemas

Refresh + Add Device

<input type="checkbox"/>	Device ID	Device Type	Class ID	Date Added	Location
Results 1-1 of 1					
<input type="checkbox"/>	WRC	Adafruit_HUZZAH	Device	25 mar. 2017 11:10:57	

Următorul pas este scrierea aplicație Node-RED care va face conversia între comenzile vocale date de utilizator și comenzile WiFi Car. Vom naviga la adresa <https://wcr.mybluemix.net/> (adresa diferă în funcție de numele ales al aplicației) și vom accesa Node-RED flow editor (editorul vizual ce ne va permite scrierea aplicației). Aplicația are diagrama următoarea:



Pentru a putea utiliza elementul de intrare de tip microfon trebuie să instalăm următoarele două extensii Node-RED: *node-red-contrib-browser-utils* și *node-red-contrib-media-utils* (extensiile se instalează accesând din meniul principal opțiunea *Manage palette*). Elementul de intrare de tip *microphone* va permite înregistrarea comenzilor vocale direct din browser și le va transmite automat către serviciul de recunoaștere vocală *Speech to Text*. Ieșirea serviciului de recunoaștere vocală se va conecta la un bloc decizional cu următoarea configurație:

Edit switch node

Delete Cancel Done

Name

Property

contains	a_z forward	→ 1	x
contains	a_z backward	→ 2	x
contains	a_z stop	→ 3	x
contains	a_z left	→ 4	x
contains	a_z right	→ 5	x

Rolul blocului decizional este de a identifica comenzile vocale transcrise de serviciul de recunoaștere vocală (forward, backward, stop, left, right). Din păcate serviciul de recunoaștere vocală nu este disponibil în limba română. Ieșirile blocului decizional conduc la blocuri distincte de construire a comenzii în format JSON către serviciul IoT:

Delete Cancel Done

Name

Function

```

1 msg.payload = "{\"command\":\"i\"}";
2 return msg;

```

Serviciul IoT care va trimite, prin intermediul serviciului IBM Watson, comenzile către WiFi Car va avea următoarea configurație:

Edit ibmiot out node

Delete Cancel Done

Authentication

Output Type

Device Type

Device Id

Command Type

Format

Data

QoS

Name

Pentru siguranță am adăugat un bloc de intrare pentru comanda de *STOP* care permite comanda manuală pentru cazuri de urgență. Restul componentelor din aplicație sunt utile doar pentru depanare: blocul msg.transcription va afișa textul de ieșire al serviciului de recunoaștere vocală iar blocurile WCR + msg.payload ne arată mesajele ce ajung efectiv la serviciul IoT.

Ultimul pas al implementării este reprogramarea modulului Adafruit HUZZAH pentru a prelua comenzile de la serviciul IBM Watson (pentru reprogramare modulul trebuie deconectat de la placa Arduino Uno). Programul a fost dezvoltat și testat utilizând Arduino IDE 1.8.1 și extensia esp8266 2.3.0 precum și bibliotecile MQTT 1.11.0 și ArduinoJson 5.8.3.

```
#include <ESP8266WiFi.h>
#include <MQTTClient.h>
#include <ArduinoJson.h>
```

Programul trebuie particularizat cu datele de conectare la rețeaua WiFi locală (*ssid* și *password*) precum și cu datele de autentificare la serviciul de cloud (*OrgID*, *DeviceT*, *DeviceID*, *AuthenticationToken* și *OrgID*):

```
const char* ssid      = "...";
const char* password = "...";

WiFiClient net;
MQTTClient client;

void connect_mqtt() {
    while (!client.connect("d:OrgID:DeviceT:DeviceID", \
        "use-token-auth", "AuthenticationToken")) {
        delay(1000);
    }
    client.subscribe("iot-2/cmd/+/fmt/json");
}

void setup() {
    Serial.begin(9600);
```

```

    delay(10);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500); }
    client.begin("OrgID.messaging.internetofthings. \
        ibmcloud.com", 1883, net);
    connect_mqtt();
}

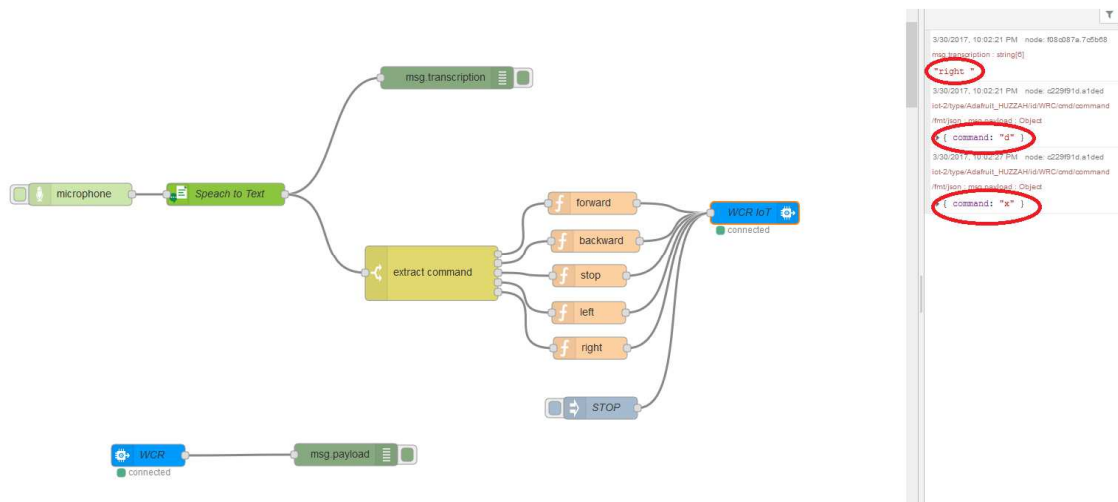
void loop() {
    client.loop();
    delay(10);
    if (!client.connected()) {
        connect_mqtt();
    }
}

void messageReceived(String topic, String payload, char *
bytes, unsigned int length) {
    StaticJsonBuffer<200> jsonBuffer;
    JsonObject& root = jsonBuffer.parseObject(payload);
    if (!root.success()) {
        return;
    }
    String command = root["command"];
    Serial.println(command);
}

```

Programul de pe placa Arduino Uno rămâne neschimbat. După reconectarea modulului Adafruit HUZZAH la WiFi Car putem să testăm comanda vocală. Din aplicația Node-RED se apasă butonul blocului microphone și se dă comanda (se înregistrează o comandă vocală care să conțină unul dintre cuvintele cheie forward, backward, stop, left sau right). Atenție, trebuie să permitem la nivel de browser ca aplicația Node-RED să acceseze dispozitivul hardware microfon al sistemului de calcul. După rostirea

comenzii se apasă din nou butonul blocului microphone pentru a opri înregistrarea și a trimite comanda vocală către serviciul de recunoaștere vocală.



În captura de ecran anterioară se pot vedea mesajele din consola debug pentru o comandă vocală (*right*) și o comandă manuală (*stop*). Pentru comanda vocală se poate vedea atât recunoașterea vocală a comenzii cât și mesajul obținut ca feedback de la serviciul IoT. Pentru comanda manuală avem doar feedback-ul afișat.

Bineînțeles, ca alternativă la comanda vocală din browser există posibilitatea contruirii unui sistem portabil de înregistrare vocală (telecomandă vocală). Proiectul de mai jos explorează această posibilitate utilizând o placă de dezvoltare Raspberry Pi:

Control your Robosapien Humanoid Robot using IBM Watson IoT Platform, Raspberry Pi and Node-RED (Part 1)

<https://developer.ibm.com/recipes/tutorials/control-your-robosapien-humanoid-robot-using-ibm-watson-iot-platform-raspberry-pi-and-nodered/>

Control the Humanoid Robot with voice commands using IBM Watson Speech to Text service (Part 2)

<https://developer.ibm.com/recipes/tutorials/control-the-humanoid-robot-with-voice-commands-using-ibm-watson-speech-to-text-service-part-2/>