Green Eyes

『D4C』+『IoT』+『農業』
～スマートアグリへの挑戦～

平成28年度オープンデータコンテスト応募作品

アグリ　太郎

◆Agenda

- 背景
- アプリの説明
- 課題と今後やってみたいこと

背景

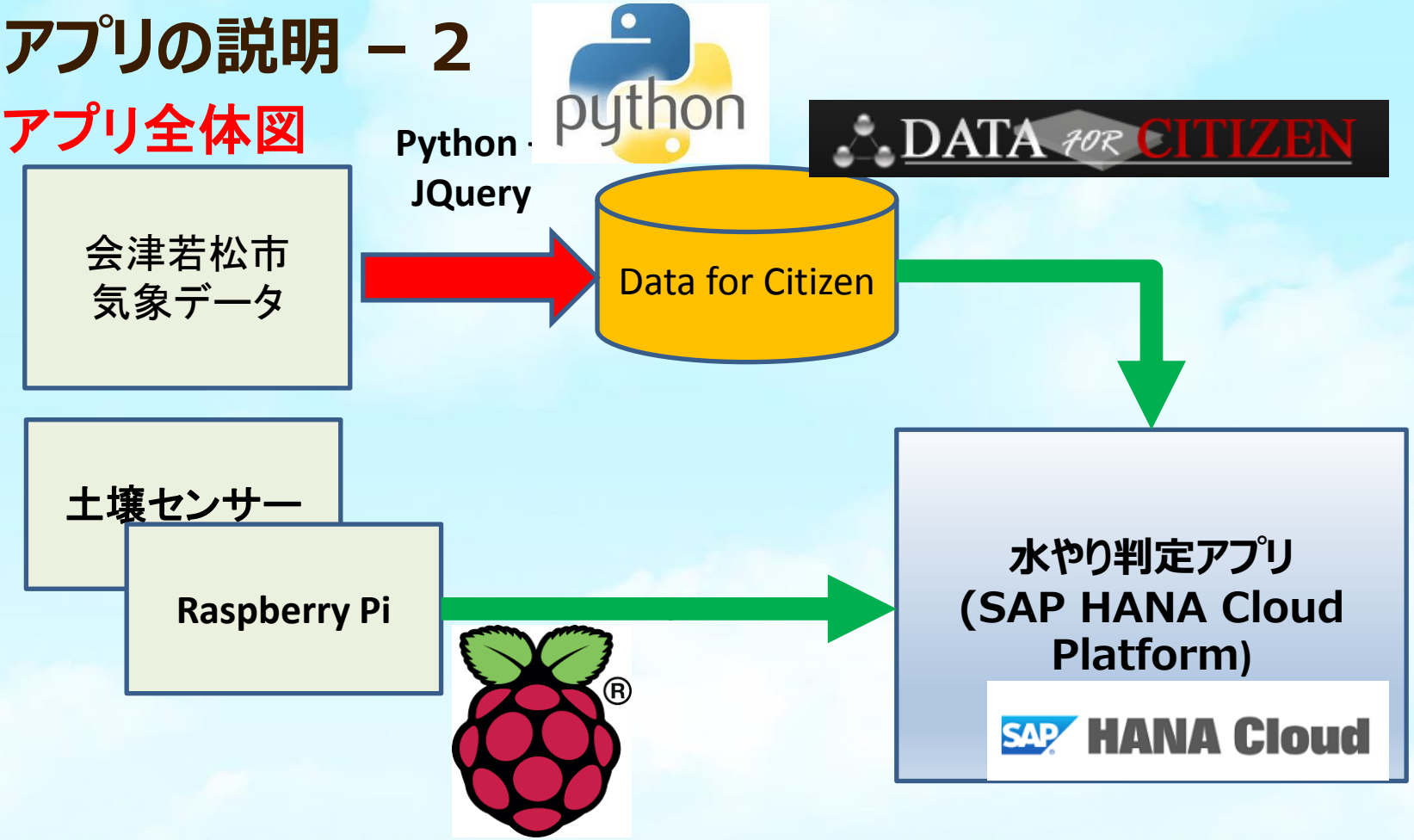
- スマートシティを目指している会津若松市、そんな会津若松の皆さんの瞳が自然の緑色に映える環境にやさしいスマートシティになるようにと願いを込めてタイトルを『緑の会津』、Green Eyesと付けました。
- IoTが脚光を浴びてきている昨今、私も何かできないかと今回の企画を思いつきました。
- 会津若松市にはData for Citizenの基盤がある、オープンデータへの取り組みが盛んである、また、9月に行われたIoTワークショップのノウハウを生かして何かできないかと思ったのがきっかけです。

アプリの説明 - 1

- まずは身近にできるスマートアグリのIoTアプリを作ろうと考えました。
- 会津若松市の気象データと土壌センサーを用いて、『水やり』のタイミングを知らせるアプリを作ることとしました。
- 会津若松市の気象データは、気象庁のページよりPytonとJQueryでData for Citizenにアップロードします。
- 土壌センサーの値は、土壌センサーと**Raspberry Pi**（ラズベリー パイ）を用いて取得します。
- 取得した気象データと土壌センサーの値を元に『水やり』が必要かどうかをプログラムで判断します。
- アプリ開発はクラウド環境(SAP HANA Cloud Platform)で行います。

アプリの説明 – 2

アプリ全体図



アプリの説明 - 3

使用した技術やもの

1. Python（無料）
2. JQuery（無料）
3. Raspberry Pi（Amazonで購入）
4. 土壌センサー(Amazonで購入)
5. A/Dコンバータ（Amazonで購入）
6. SAP HANA Cloud Platform(無料のトライアル環境)
7. Data for Citizen
8. 会津若松市の気象データ

   http://www.jma.go.jp/jp/amedas_h/today-36361.html

アプリの説明 - 4

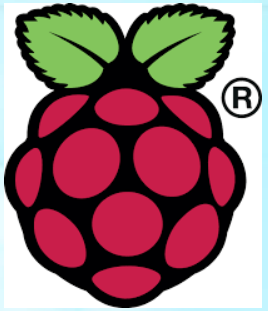
会津若松市の気象データ(http://www.jma.go.jp/jp/amedas_h/today-36361.html)を
PythonとJQueryを用いてData for Citizenにアップします。テーブル名：O_WETHER_DATA
定期的に実行し、DataForCitizenにアップロード(※PCに電源がはいってる時のみ)

◆プログラムの一部を紹介

```
#!/usr/bin/env python
# coding: UTF-8
#
# @see http://www.crummy.com/software/BeautifulSoup/documentation.html
#
import os

#import urllib2
try:
    import urllib.request as urllib2
except ImportError:
    import urllib2

from bs4 import BeautifulSoup

def str2float(str):
    try:
        return float(str)
    except:
       return 0.0

#会津若松市のアメダスデータ（表形式）
if __name__ == "__main__":
    url = 'http://www.jma.go.jp/jp/amedas_h/today-36361.htm

    # サーバーから気象データのページを取得
    html = urllib2.urlopen(url).read()

    soup = BeautifulSoup(html,"html.parser")

    trs = soup.find('table', { 'id' : 'tbl_list' })
```

時刻	気温	降水量	風向	風速	日照時間	積雪深	湿度	気圧
時	℃	mm	16方位	m/s	h	cm	%	hPa
1	0.6	0.0	北西	1.5		0	95	1027.7
2	-0.4	0.0	南南西	1.0		0	96	1027.3
3	-0.6	0.0	北北西	1.6		0	97	1027.9
4	-0.7	0.0	南南西	1.4	0.0	0	99	1028.2
5	-1.2	0.0	東南東	1.3	0.0	0	98	1028.6
6	-1.4	0.0	南東	1.7	0.0	0	97	1028.6
7	-1.4	0.0	東南東	0.9	0.0	0	98	1029.0
8	-0.6	0.0	北北東	1.1	0.0	0	100	1029.1
9	0.0	0.0	北西	1.1	0.0	0	100	1028.7
10	2.2	0.0	北	1.9	0.7	0	100	1028.2
11	4.6	0.0	北西	1.1	1.0	0	76	1026.6

アプリの説明 - 5



Raspberry Piと土壌センサーを用いて土の湿度を取得。
自宅の植木鉢を利用。（急だったので花はありません。。とほほ）
下記のサイトを参考に突貫で構築。（ほぼコピーでいけました）

http://dev.classmethod.jp/server-side/node-js-server-side/raspberry-pi-2-ad-convert-nodejs/

◆プログラムの一部を紹介

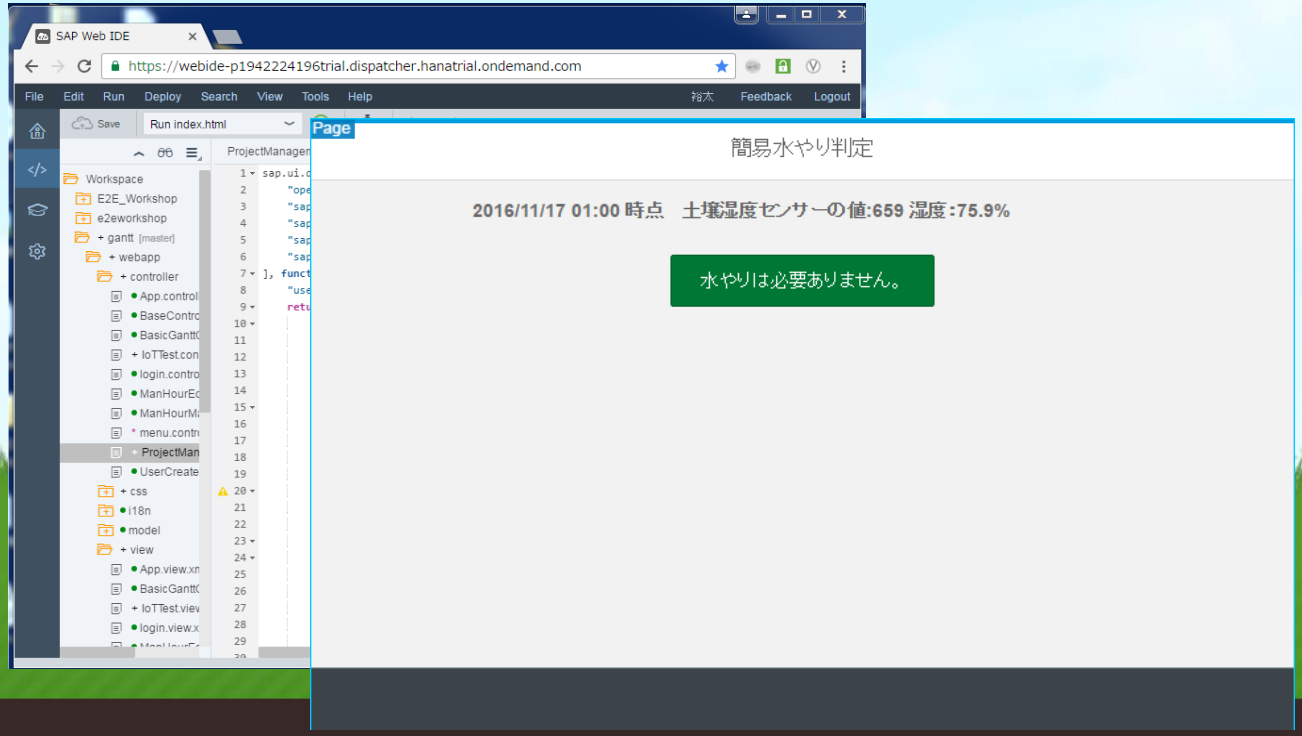
```
var raspi    = require('raspi');
var gpio     = require('raspi-gpio');

const HIGH   = gpio.HIGH;
const LOW    = gpio.LOW;
const Input  = gpio.DigitalInput;
const Output = gpio.DigitalOutput;

var initHandler = function() {

    var d_in  = new Output('GPIO10'); //SPI_MOSI
    var d_out = new Input('GPIO9');   //SPI_MISO
    var clk   = new Output('GPIO11'); //SPI_CSLK
    var cs    = new Output({ 'pin' : 'GPIO8', 'pullResistor' : gpio.PULL_UP });  //SPI_CE0

    var clock = function(count) {
        for(var i = 0; i < count; i++) {
            clk.write(HIGH);
            clk.write(LOW);
        }
    };

    var readValue = function() {
        cs.write(LOW); //start

        var i;
        var ch  = 0; // 0 ch. (0~7)
        var cmd = ( ch | 0x18 ) << 3;
        for(i = 0; i < 5; i++) {
            d_in.write( ( cmd & 0x80 ) ? HIGH : LOW );
```

アプリの説明 － 6

水やり判定のWebアプリをSAP HANA Cloud Platformのトライアル環境を使用して作成。
AWSのIoTサービスがあるが今回は、2016年9月に受講したIoTワークショップのノウハウを利用。

◆プログラムの一部を紹介

会津若松市の気象データ(湿度)と土壌湿度センサー(1～1023)の値をもとに判定。
土壌湿度センサーと気象データの湿度を元に算出。
※土壌湿度センサーの値(1～1023)×気象データ湿度（%）の値が８００を超えたら水やりが必要と判定する。



課題と今後やってみたいこと

今回のアプリ作成で感じたスマートアグリの課題。

畑や田んぼに電源がない・・・今回は、自宅の植木鉢を用いたが、畑にセンサーを設置する場合に電源がない。電源が合って遠い。。。

Wifi環境がない・・・当然、畑や田んぼにWifi環境がなく、センサーで取得した値をデータベース等に登録する手段がない。

上記の問題があり、なかなかリアルタイムのスマートアグリは難しいと思いました。

⇒リアルタイムではなく、１日数回のセンサーの値の取得などになってしまう。

・スマートアグリには、電源環境やWifi環境が必要であると感じました。

スマートシティを構築していく中で、都市部だけではなく田畑にある程度、

電源やWiFi環境があると便利かと思いました。

・今後やってみたいこと（やってほしいこと）としては、ドローンを利用し、畑や田んぼの情報をオープンデータ化し、会津若松のスマートアグリ化をしてみたいと思いました。

ありがとうございました。