



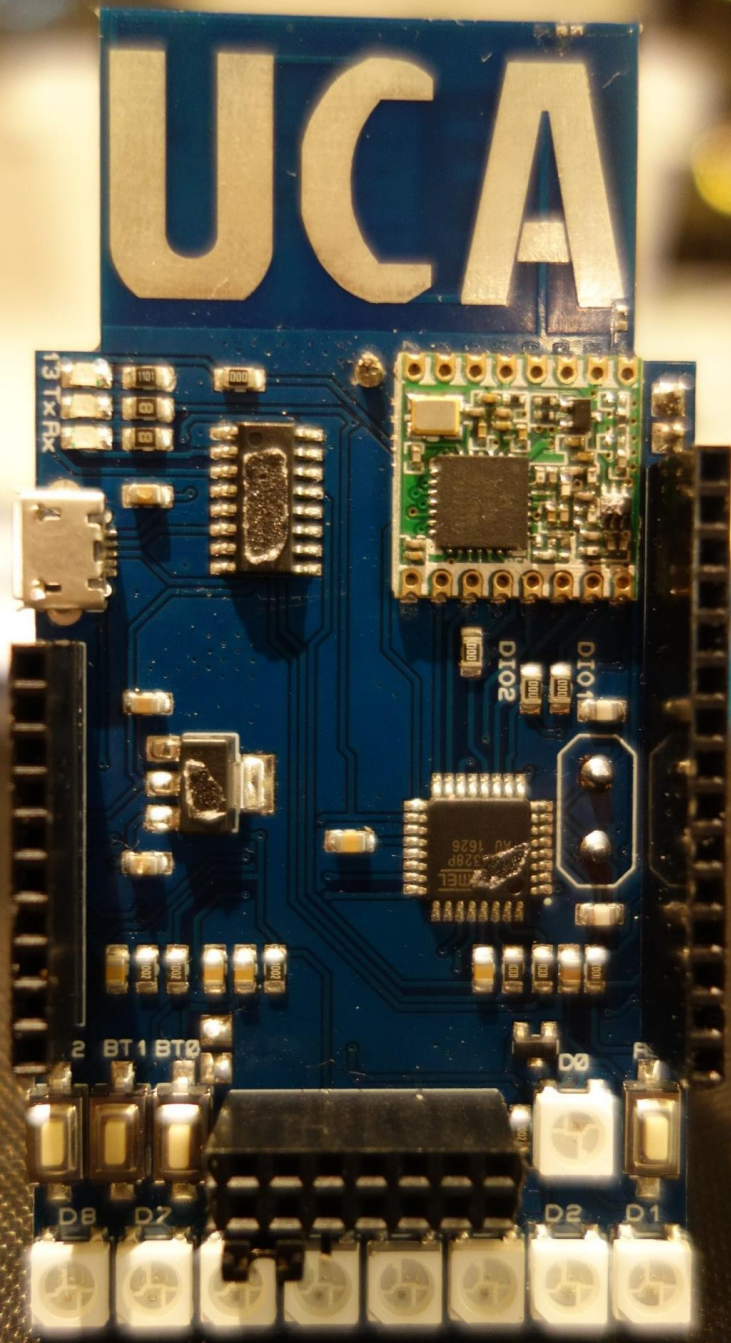
HomeConnect

ESCOBAR QUENTIN

VIVALDI ADRIEN

Sommaire

- Présentation du projet
- Problématique
- Listes des composants
- Réseau et communication
- Code et Explications
- Design



Présentation du projet

Problématique

Récupérer les données depuis TTN

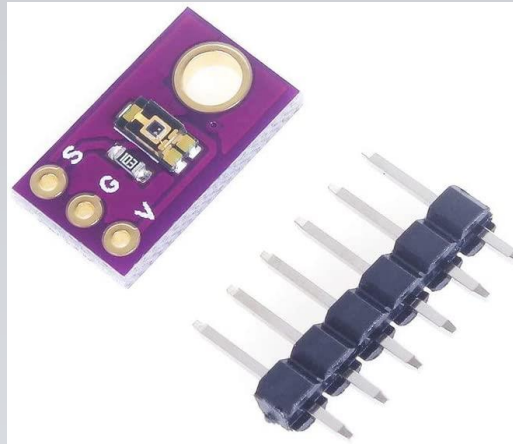
Détecteur de mouvement non précis (faux positif)

Traiter les données

Listes des composants



Capteur de mouvement



Capteur de luminosité



Capteur de
température
& humidité

Réseau et communications



```
mirror_mod = modifier_ob.  
#set mirror object to mirror.  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
#selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
print("please select exactly  
-- OPERATOR CLASSES ----  
types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
context):  
context.active_object is not
```

Code et Explications

```

float readLight(){

    float result;
    float sensorValue = analogRead(A3);
    float voltage = sensorValue * (3.30 / 1023.0);
    result = voltage * 2000;
    return result;
}

/*void movementReader(){

    if (digitalRead(2) == LOW){
        Serial.println("Mouvement non détecté");
        pir = false;
        return pir;
    }
    else if (digitalRead(2) == HIGH){
        Serial.println("Mouvement détecté");
        pir = true;
        return pir;
    }
}
*/
void updateEnvParameters()
{
    int temperature = sensor.getCelsiusHundredths();
    temp = temperature / 100;
    humidity = sensor.getHumidityPercent();
    light = readLight();
    Serial.print(temp); Serial.println("°C");
    Serial.print(humidity); Serial.println("%");
    Serial.print(light); Serial.println("lux");
}


```

Code Arduino

INTEGRATION OVERVIEW

Process ID home_connect

Status ● Running

Platform  HTTP Integration (v2.6.0) [documentation](#)

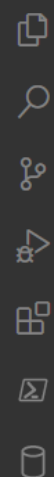
Author The Things Industries B.V.

Description Sends uplink data to an endpoint and receives downlink data over HTTP.

TTN vers Web

Récupération en PHP

```
mirror_mod = modifier_ob.  
set mirror object to mirror  
mirror_mod.mirror_object  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
print("please select exactly  
-- OPERATOR CLASSES ----  
types.Operator):  
on X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
context):  
context.active_object is not
```



ttn.php × notification.php

ttn.php > ...

```
1  <?php
2  /*****
3   * Projet Communication Sans Fil
4   * HomeConnect
5   * -----
6   * Page de réception des données de The Things Network via fichier JSON,
7   * et envoi des data reçu sur une Base de données en MySQL.
8   * -----
9   * Créé par Adrien Vivaldi et Quentin Escobar
10  * Version 1.0
11  *****/
12
13  require_once('Extensions/config.php'); // Connexion à la bdd
14  ini_set('display_errors', 'off'); // On retire l'affichage des erreurs si vrai
15  $write_log = true; // Autorise ou non l'écriture de log
16
17  $ttn_post = file('php://input'); // Récupération des données
18  $data = json_decode($ttn_post[0]); // Récupération de l'encodage JSON et conversion en variable PHP sur $data
19
20  // Création des variables PHP via le JSON decoder
21  $sensor_temperature = $data->payload_fields->temperature_1; // La valeur du capteur de température s'inscrit sur la variable PHP sensor_temperature
22  $sensor_humidity = $data->payload_fields->relative_humidity_2; // La valeur du capteur d'humidité s'inscrit sur la variable PHP sensor_temperature
23  $sensor_luminosity = $data->payload_fields->luminosity_3; // La valeur du capteur de luminosité s'inscrit sur la variable PHP sensor_luminosity
24
25  $ttn_app_id = $data->app_id; //
26  $ttn_dev_id = $data->dev_id; //
27  $ttn_time = $data->metadata->time; //
28
29  // Récupère des informations complémentaire
30  $server_datetime = date("Y-m-d H:i:s");
31  $jour = strftime('%A');
32  $heure = date("H");
33
34  // On sécurise les données récupérée via la fonction PHP htmlspecialchars
35  $ttn_app_id = htmlspecialchars($ttn_app_id);
36  $ttn_dev_id = htmlspecialchars($ttn_dev_id);
37  $ttn_time = htmlspecialchars($ttn_time);
38  $sensor_temperature = htmlspecialchars($sensor_temperature);
39  $sensor_humidity = htmlspecialchars($sensor_humidity);
40  $sensor_luminosity = htmlspecialchars($sensor_luminosity);
41
42  // Insertion des variables dans la table homeconnect_data de notre base de données MySQL via requêtes préparées : fonctions prepare et execute (propre au PDO)
43  // pour plus de sécurité (injection SQL) et plus de rapidité.
44  $insertbdd = $bdd->prepare("INSERT INTO homeconnect_data(datetime, app_id, dev_id, ttn_time, jour,heure, temperature, humidity, luminosity) VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?)");
45  $insertbdd->execute(array($server_datetime, $ttn_app_id, $ttn_dev_id, $ttn_time, $jour,$heure, $sensor_temperature, $sensor_humidity, $sensor_luminosity));
46
47  // -----
48  // Vérification capteurs et serveur
49
```



```
mirror_mod = modifier_ob.  
set mirror object to mirror.  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
print("please select exactly  
-- OPERATOR CLASSES ----  
types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
context):  
context.active_object is not
```



De PHP vers Discord

```
Fichier Edition Sélection Affichage Atteindre Exécuter Terminal Aide
index.php notif_temp.php X stat_temp.php 2 temperature.php humidite.php luminosite.php
Extensions > notif_temp.php > ...
1 <?php
2
3 $url = 'https://discord.com/api/webhooks/838792869153341440/EcNXiIHOf3bi72vHiKP_92HjKWKzFmKiSg8-FXzIM7y02';
4
5 $data = array(
6     'content' => "__**Ca brûle mon copain !**__ \n
7     Tempetrature : **$sensor_temperature**
8     ",
9     'username' => '', // Remplacer le nom du webhook, à enlever si inutilisé
10    'avatar_url' => '', // (rouge) Remplacer l'avatar webhook (doit être une URL), à enlever si inutilisé
11    'embeds' => array(
12        array(
13            'title' => '', // Intitulé du lien
14            'url' => '', // Adresse du lien
15            'description' => '', // Texte affiché après le titre
16
17            // Image, Miniature, Auteur et Footer son optionnels
18
19            /*
20             * Ajouter une image
21             */
22            'image' => array(
23                'url' => '', // (jaune) Adresse de l'image
24                'width' => 0, // Largeur de l'image
25                'height' => 0 // Hauteur de l'image
26            ),
27            /*
28             * Ajouter une miniature
29             */
30            'thumbnail' => array(
31                'url' => '', // (vert) Adresse de l'image
32                'width' => 0, // Largeur de l'image
33                'height' => 0 // Hauteur de l'image
34            ),
35            /*
36             * Ajouter un auteur
37             */
38            'author' => array(
39                'name' => 'En voir plus', // Nom de l'auteur
40                'url' => 'https://kayze-photos.fr/securise/admin_connect.php', // Adresse de l'auteur
41                'icon_url' => 'https://kayze-photos.fr/images/compensateur.jpg' // (bleu foncé) Avatar de l'auteur
42            ),
43            /*
44             * Ajouter une mention en pied de page
45             */
46            'footer' => array(
47                'text' => '' // Texte à afficher
```

```
Fichier Edition Sélection Affichage Atteindre Exécuter Terminal Aide
index.php notif_temp.php X stat_temp.php 2 temperature.php humidite.php luminosite.php
Extensions > notif_temp.php > ...
38
39     'author' => array(
40         'name' => 'En voir plus', // Nom de l'auteur
41         'url' => 'https://kayze-photos.fr/securise/admin_connect.php', // Adresse de l'auteur
42         'icon_url' => 'https://kayze-photos.fr/images/compensateur.jpg' // (bleu foncé) Avatar de l'auteur
43     ),
44     /*
45     * Ajouter une mention en pied de page
46     */
47     'footer' => array(
48         'text' => '', // Texte à afficher
49         'icon_url' => '' // (bleu clair) URL de l'image
50     )
51 ),
52 );
53
54 $context = array(
55     'http' => array(
56         'method' => 'POST',
57         'header' => "Content-type: application/json\r\n",
58         'content' => json_encode($data),
59     )
60 );
61
62 /*
63 * Attention, certains serveurs désactivent la fonction 'allow_url_fopen'.
64 * Si c'est votre cas et si vous ne pouvez pas l'activer, vous ne pourrez
65 * pas utiliser ce script directement. Vous devrez utiliser cURL !
66 */
67
68 $context = stream_context_create($context);
69 $result = @file_get_contents($url, false, $context);
70
71 if($result === false) {
72     return false;
73 }
74
75 return true;
```

Base de données

```
mirror_mod = modifier_ob.  
#set mirror object to mirror  
mirror_mod.mirror_object =  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
  
#selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
  
print("please select exactly  
  
-- OPERATOR CLASSES -----  
  
types.Operator):  
on X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
  
context):  
context.active_object is not
```

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	id			Non	Aucun(e)			 Modifier  Supprimer  Plus
<input type="checkbox"/>	2	datetime			Non	Aucun(e)			 Modifier  Supprimer  Plus
<input type="checkbox"/>	3	app_id	text	utf8_general_ci	Non				 Modifier  Supprimer  Plus
<input type="checkbox"/>	4	dev_id	text	utf8_general_ci	Non				 Modifier  Supprimer  Plus
<input type="checkbox"/>	5	ttn_time	text	utf8_general_ci	Non				 Modifier  Supprimer  Plus
<input type="checkbox"/>	6	jour	text	utf8_general_ci	Non				 Modifier  Supprimer  Plus
<input type="checkbox"/>	7	heure	int(11)		Non	Aucun(e)			 Modifier  Supprimer  Plus
<input type="checkbox"/>	8	temperature	int(11)		Non	Aucun(e)			 Modifier  Supprimer  Plus
<input type="checkbox"/>	9	humidity	int(11)		Non	Aucun(e)			 Modifier  Supprimer  Plus
<input type="checkbox"/>	10	luminosity	int(11)		Non	Aucun(e)			 Modifier  Supprimer  Plus

HomeConnect

