

2017

Internet des objets : Domotique



ARAS Bilal

KHALFAT Mossaab

LY Abdoul Yaya

DUT RT AN 2

PROJET TUTURE 2017

MR J.M FEYBESSE

Remerciements :

Nous tenons à remercier notre tuteur de projet Mr FEYBESSSE pour son soutien et son aide dans le cadre de notre projet.

Nous tenons aussi à remercier tous les professeurs du DUT RT qui nous ont permis d'obtenir les connaissances afin de pouvoir réaliser ce projet concernant L'Internet des objets.

Table des matières

Introduction	4
I-L'Internet des objets.....	5
A-La Domotique.....	5
B-Les Capteurs	5
II-La raspberry pi 3.....	8
A-Présentation de La raspberry pi 3	8
B-Les Pins et le GPIO.....	9
C-Les Systèmes d'exploitations	10
D-Installation de Raspbian :.....	10
E-Les circuits de notre projet	11
III-Activation de la connexion wifi de la carte Raspberry Pi.....	13
A-Configuration des interfaces	13
B-Probleme rencontres : RF-kill	13
C-Activation de la Wi-Fi.....	14
IV-Activation du serveur SSH de la Raspberry Pi	15
A-Installer un client SSH sur Linux	15
B-Ajout d'un nouvel utilisateur.....	15
C- Configuration de SSH	15
V-WiringPi.....	17
A-Installation.....	18
B-Les commandes de base.....	18
VI-Utilisation de DHT_11	19
A-Installation de de Adafruit et python	19
B-Code python	19
VII-Interface web	20
A-Installation d'un serveur web	20
B-Mise en place de l'interface web	21
CONCLUSION	22
GLOSSAIRE.....	23

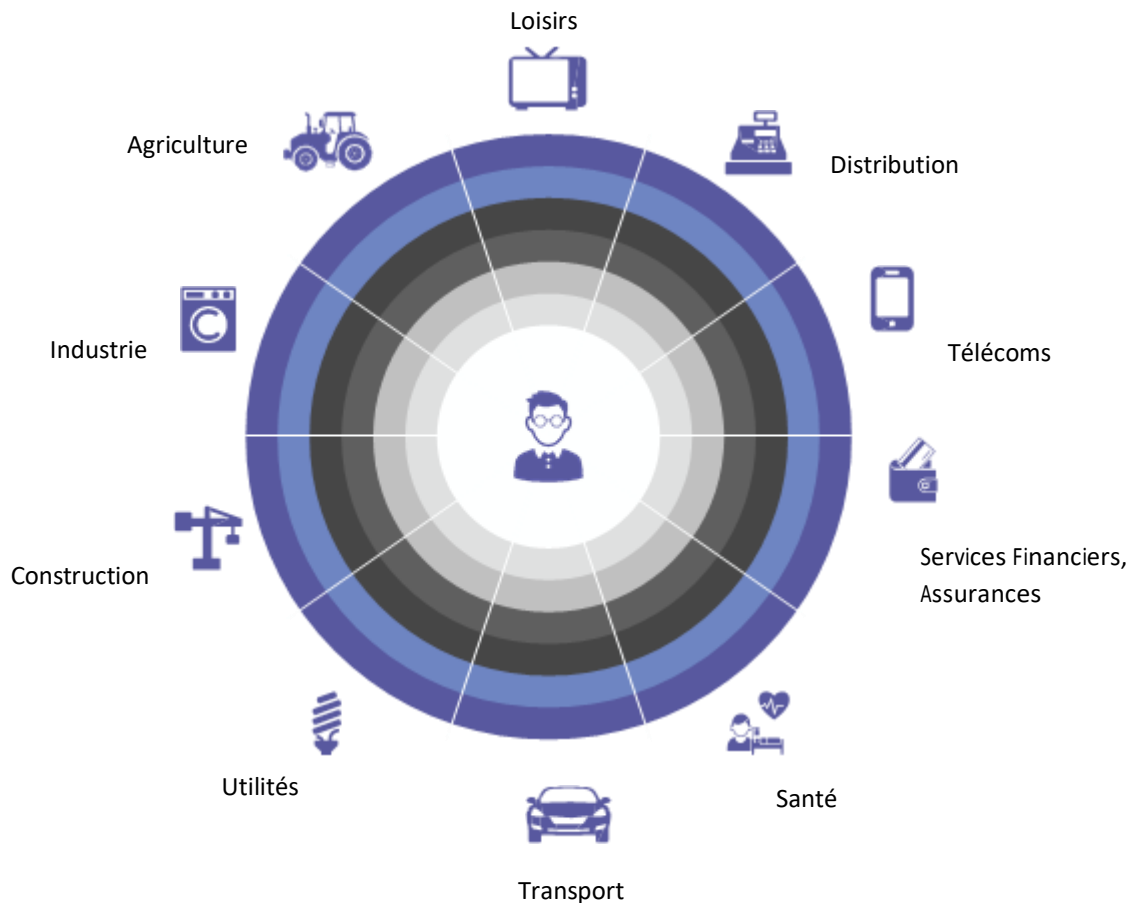
Introduction

Le terme « internet des objets » remonte, autant qu'on puisse en témoigner, à l'année 1999 où **Kevin Ashton**, un employé de Procter & Gamble, a utilisé le terme en question pour désigner le **lien qui existe entre la technologie RFID et l'internet**.

Aujourd'hui, l'Internet des Objets désigne avant tout le futur d'un monde où les appareils interagissent entre eux via le cloud, le réseau internet et cellulaire pour analyser de grandes quantités de données via le Big Data.

A l'heure actuelle nous pouvons trouver des objets connectés dans plusieurs domaines notamment les plus connus qui sont l'automobile, les maisons intelligentes, l'industrie et la santé.

Aujourd'hui l'Internet des objets a un domaine d'utilisation très vaste et continue de se répandre.



Problématique : Qu'est-ce que l'IoT ?

Afin de répondre à cette question, premièrement nous allons définir l'IoT et donner un exemple concret dans le cas de la domotique.

I-L'Internet des objets

L'Internet des objets (*Internet of Things* : IoT) est une technologie qui permet de connecter sur divers réseaux des appareils d'enregistrement des données. Il existe différents usages variés en fonctions des objets dans le domaine de la e-santé ou domotique par exemple.

A-La Domotique

La domotique a pour principe de contrôler et d'ajuster le fonctionnement des appareils électriques en fonction de certains paramètres.

En domotique l'IoT recouvre les appareils électroménagers communiquant, les capteurs (thermostat, détecteur de fumée ou de présence) ou bien les systèmes de sécurité connectés des appareils de type domotique.

B-Les Capteurs

Pour pouvoir mesurer les paramètres dont nous avons besoin pour contrôler et d'ajuster le fonctionnement des appareils électriques, les capteurs sont nécessaires.

Ces capteurs permettent de convertir une mesure en un signal, transmis ensuite aux appareils électriques pour contrôler leur fonctionnement.

Les capteurs ne réagissent pas de la même manière on distingue cinq types de capteurs pour une installation domotique :

a) Le capteur logique, ou capteur TOR (Tout Ou Rien) :

C'est le type de capteur le plus basique. Il détecte une action ou un évènement. Le capteur met en marche ou arrête l'appareil sur lequel il est relié. Il traduit l'action observée par deux possibilités, 0 et 1 (traduit en marche / arrêté par exemple).



Figure 1 Capteur de présence, un bel exemple de capteur TOR

b) Le capteur analogique :

Un capteur analogique peut être physique ou chimique, selon la grandeur qu'il mesure. Pour un capteur de température ou un capteur acoustique, on parle de capteur analogique physique. Si on mesure le taux d'humidité par exemple, on parle alors de capteur analogique chimique. Le capteur analogique transmet une valeur proportionnelle à la grandeur mesurée.

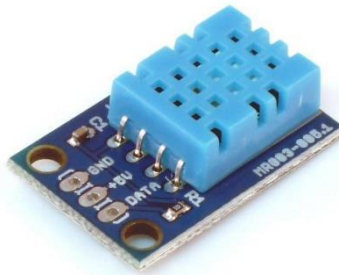


Figure 2 Capteur de température

c) Le capteur numérique :

Un capteur numérique mesure une grandeur et transmet une valeur chiffrée précise. Le pluviomètre est un exemple classique de capteur numérique.



Figure 3 Pluviomètre

d) Le capteur digital :

Un capteur digital utilise des données binaires (comme un capteur logique) combinatoires. Il peut donc associer deux états ou actions. Les volets électriques par exemple peuvent utiliser ce type de capteur : les volets se ferment s'ils ne sont pas déjà complètement fermés et s'il y a du soleil.

e) Le capteur biométrique :

La biométrie est la mesure d'une grandeur humaine. Ces capteurs sont modernes et font appel à des technologies innovantes. Un capteur d'empreinte digitale par exemple est un capteur biométrique.



Figure 4 Capteur d'empreintes digitales

II-La raspberry pi 3

Le Raspberry Pi est un nano-ordinateur monocarte à processeur ARM conçu par le créateur de jeux vidéo David Braben, dans le cadre de sa fondation Raspberry Pi.

Cet ordinateur, qui a la taille d'une carte de crédit, est destiné à encourager l'apprentissage de la programmation informatique ; il permet l'exécution de plusieurs variantes du système d'exploitation libre GNU/Linux et des logiciels compatibles. Il est fourni nu (carte mère seule, sans boîtier, alimentation, clavier, souris ni écran) dans l'objectif de diminuer les coûts et de permettre l'utilisation de matériel de récupération.

Depuis sa création la raspberry a connu trois générations : le modèle A, le A+, le modèle 1B, 1B+ ensuite les modèles 2B et la dernière que nous utilisons dans le cadre de notre projet à savoir la Raspberry Pi 3B lancé en Février 2016.

A-Présentation de La raspberry pi 3

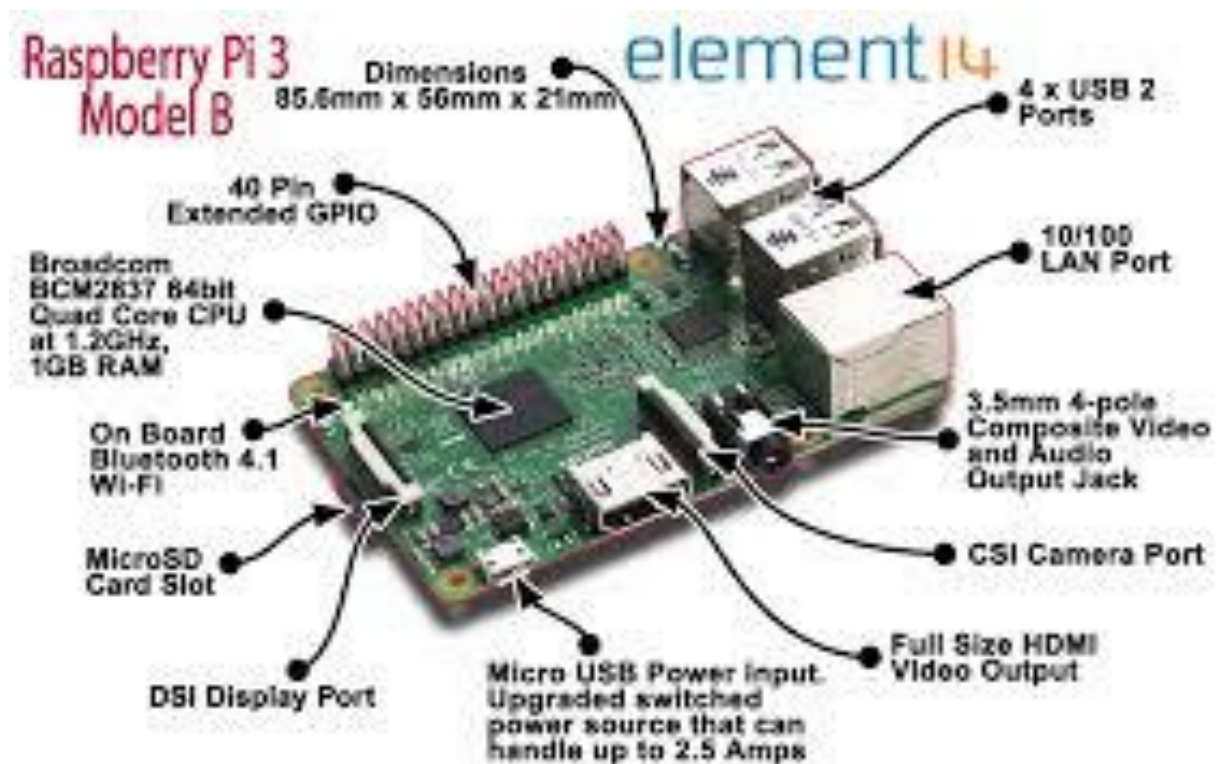


Figure 5 Raspberry Pi 3B

La Raspberry Pi est composée d'un micro USB pour l'alimentation et de 4 ports USB2 pour pouvoir brancher la souris et le clavier et d'autres périphériques. Dans notre cas on les utilise que pour le clavier et la souris.

Nous avons aussi un port LAN 10/100, et une carte wifi pour la connexion internet. Notons aussi le Bluetooth 4.1.

Elle est composée aussi d'une sortie HDMI et d'un composite Vidéo et audio.

N'étant pas composée de mémoire, le support de carte microSD permet de compenser ce manque et nous permet d'y installer un système d'exploitation. La carte mémoire la plus adaptée à notre projet est la 16 GB.

Le processeur est le Broadcom BCM2837 64bit ARMv8 quad core cadencé à 1.2 GHz et une RAM de 1Gb.

Elle est composée de 40 PIN GPIO nous permettant de communiquer avec les équipements externes. Nous allons expliquer ce qu'est le GPIO.

B-Les Pins et le GPIO

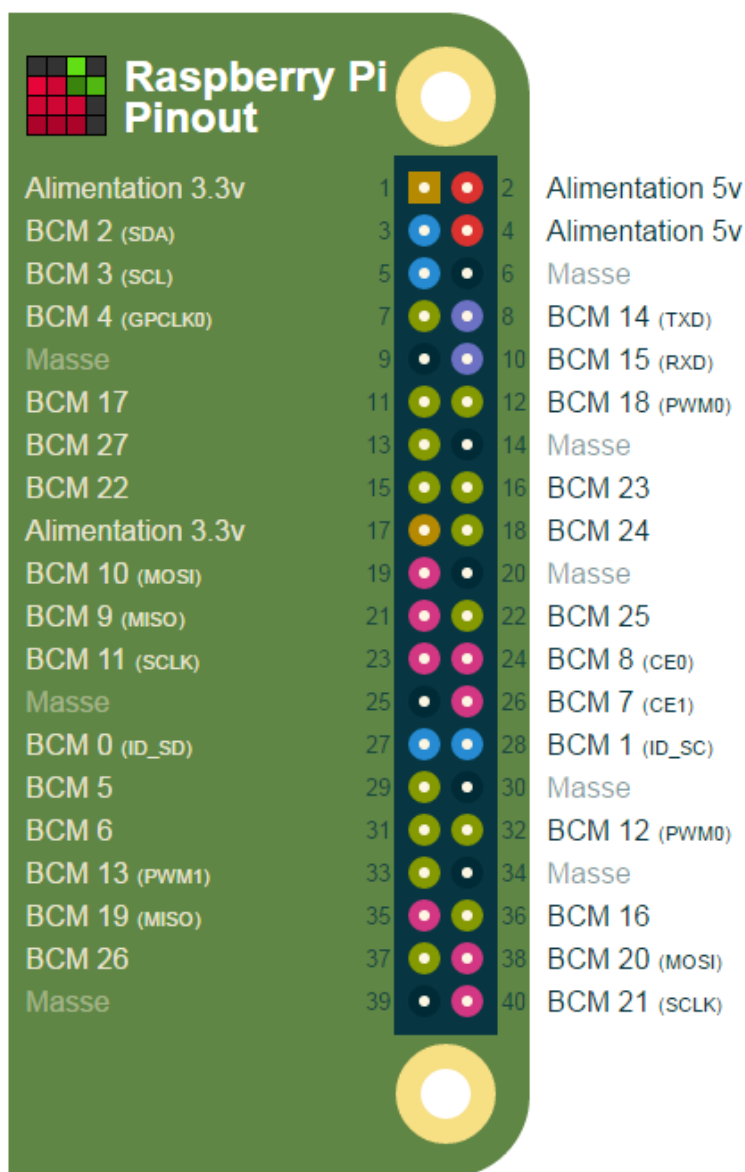


Figure 6 Pins et gpio

La plupart des pins correspondent à des BCM c –numérotation Broadcom comme noté sur la figure 2. Les BCM sont plus communément appelées « GPIO pour General Purpose Input/Output ». La figure 2 nous montre la correspondance de chaque pin et son numéro GPIO. IL faut remarquer aussi que le pin 1 correspond à une alimentation de 3.3V et les pins 2 et 4 à des alimentations de 5V. Les pins 6, 9, 14, 20, 25, 30, 34 et 39 sont des masses.

Dans le cadre de notre projet nous utilisons les pins 1 ou 2 pour l'alimentation, le pin 6 pour la masse et les pins 7 (GPIO4) et 11 (GPIO 17) pour commander notre led et récupérer les informations sur la température et l'humidité.

C-Les Systèmes d'exploitations

Nous avons beaucoup de possibilité pour notre raspberry pi 3 on va citer quelques-uns

- On a UBUNTU MATE 15.10: système basé sur Ubuntu très fluide et très bien adapter à notre projet même s'il présente un problème au niveau du Bluetooth
- Il y'a aussi le OSMC Open Source Media Center : basé sur DEBIAN et comme son nom l'indique il permet à notre raspberry pi de faire office de media center
- On a aussi OpenELEC qui est un système d'exploitation ultraléger basé sur Kodi, compatible avec toutes les versions du Raspberry Pi (dont le Pi3). Il est dédié, comme OSMC, à la création de « Media Centers ».
- Et le RASPBIAN : une distribution comme le nom le sous entend, il est basé sur DEBIAN. Le plus connu et le plus fluide pour notre raspberry 3 et exploite non seulement la wifi et le Bluetooth intégrer mais les ports GPIO qui nous permettrons de réussir notre projet. Nous allons utiliser cette distribution pour des raisons d'habitude et de conformabilité à DEBIAN utilisé tout au long de notre formation.

D-Installation de Raspbian :

Pour l'installer on a besoin d'un PC sous Windows et d'une carte SD d'au moins 16 Go pour le bon fonctionnement de notre précieux. Les étapes à suivre sont :

- L'Installer Raspbian Jessie sur le site officielle de raspberry pi <https://www.raspberrypi.org/downloads/raspbian/>
- Il nous faut un logiciel pour booter l'image sur la carte. Il s'agit de WIN32 DISK IMAGER. On peut l'avoir sur <http://win32-disk-imager.fr.uptodown.com/windows>
- Il faut extraire raspbian car étant en zip
- Une fois les deux outils mis en point on peut maintenant mettre notre Raspbian sur la carte mémoire. IL faudra lancer Win 32 et lui indiquer le chemin ou se trouve l'image raspbian. Si la carte sd est le seul périphérique branché il le met en sortie par default et on clique sur write et confirmer malgré le l'avertissement.

Une fois le boot terminer notre carte mémoire devient bootable. Et on peut voir cette dernière sur l'ordinateur comme le montre l'image suivante.



Figure 7 Etat de la carte SD après boot

La carte est ainsi chargée de fichier système :

Icon	File Name	Date	Type	Size
Folder	overlays	25/11/2016 17:24	Dossier de fichiers	
File	bcm2708-rpi-b.dtb	22/09/2016 09:07	Fichier DTB	14 Ko
File	bcm2708-rpi-b-plus.dtb	22/09/2016 09:07	Fichier DTB	14 Ko
File	bcm2708-rpi-cm.dtb	22/09/2016 09:07	Fichier DTB	14 Ko
File	bcm2709-rpi-2-b.dtb	22/09/2016 09:07	Fichier DTB	15 Ko
File	bcm2710-rpi-3-b.dtb	22/09/2016 09:07	Fichier DTB	16 Ko
File	bcm2710-rpi-cm3.dtb	24/10/2016 12:41	Fichier DTB	15 Ko
File	bootcode.bin	22/06/2016 08:06	Fichier BIN	18 Ko
File	cmdline	25/11/2016 17:52	Document texte	1 Ko
File	config	25/11/2016 17:24	Document texte	2 Ko
File	COPYING.linux	21/08/2015 17:04	Fichier LINUX	19 Ko
File	fixup.dat	25/11/2016 16:35	Fichier DAT	7 Ko
File	fixup_cd.dat	25/11/2016 16:35	Fichier DAT	3 Ko
File	fixup_db.dat	25/11/2016 16:35	Fichier DAT	10 Ko
File	fixup_x.dat	25/11/2016 16:35	Fichier DAT	10 Ko
File	issue	25/11/2016 18:09	Document texte	1 Ko
File	kernel	25/11/2016 16:35	Fichier d'image di...	4 032 Ko
File	kernel7	25/11/2016 16:35	Fichier d'image di...	4 133 Ko
File	LICENCE.broadcom	18/11/2015 16:01	Fichier BROADCOM	2 Ko
File	LICENSE.oracle	25/11/2016 18:09	Fichier ORACLE	19 Ko
File	start.elf	25/11/2016 16:35	Fichier ELF	2 756 Ko
File	start_cd.elf	25/11/2016 16:35	Fichier ELF	619 Ko
File	start_db.elf	25/11/2016 16:35	Fichier ELF	4 839 Ko
File	start_x.elf	25/11/2016 16:35	Fichier ELF	3 813 Ko

Figure 8 Contenu de la carte après boot

Après avoir finie cette partie, on peut insérer notre carte sur la raspberry et ainsi cette dernière devient un ordinateur prêt à être utiliser pour le projet.

E-Les circuits de notre projet

- Pour l'allumage et l'éteinte de led

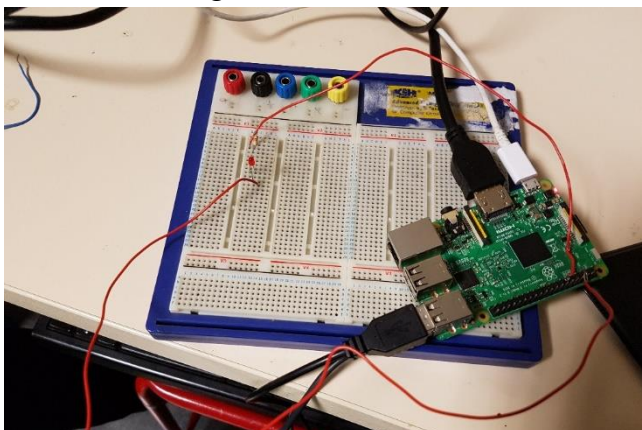


Figure 9 Circuit de led

Sur la figure 5, nous avons un circuit alimenté par la raspberry par un courant de 3.3 V (pin 11). Le courant passe par une résistance de 270 Ohm car l'intensité du courant étant de l'ordre de 10 mA alors $R=U/I$ nous donne un résultat de l'ordre de 270. La sortie du circuit est reliée à la masse (pin 6).

- Capteur de température et d'humidité

Pour le capteur de température et d'humidité nous utilisons un capteur DHT11.

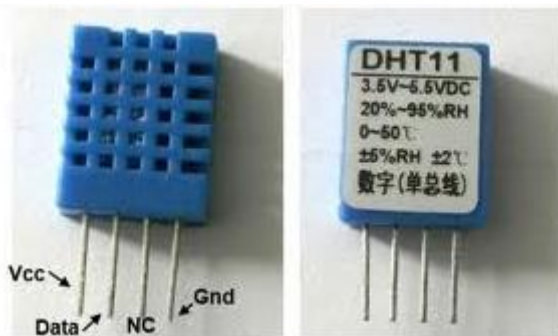


Figure 10:DHT11

Ce capteur est composé de 4 broches. La une pour l'alimentation entre 3.5V et 5.5 VDC. Cette broche est notée VDC sur l'image ci-dessus. La broche 2 qui permet de récupérer les mesures de température et d'humidité noté Data sur l'image. La 3eme broche est le NC qu'on n'utilise pas dans notre projet et la broche 4^{ème} est la masse.

Pour le circuit du projet le Vcc est relié au pin 1, le Data au pin 11 et Gnd à l'une des masses. Entre le Vcc et le Data se trouve une résistance de 270 Ohm pour les mêmes raisons que précédemment.

Voici l'image du circuit

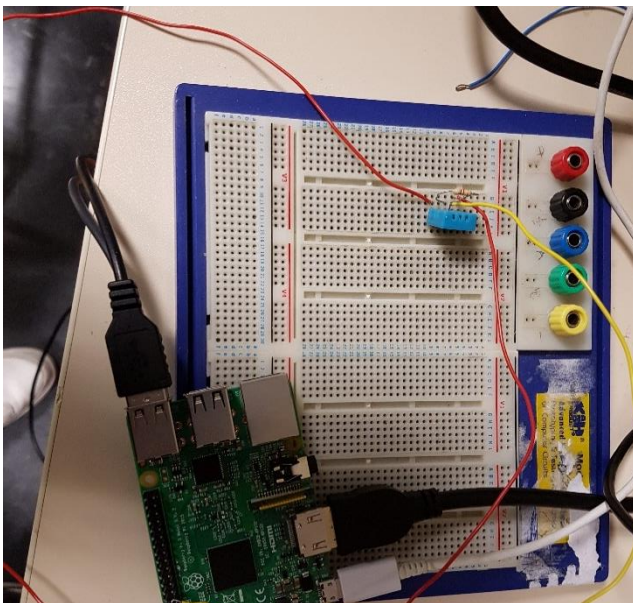


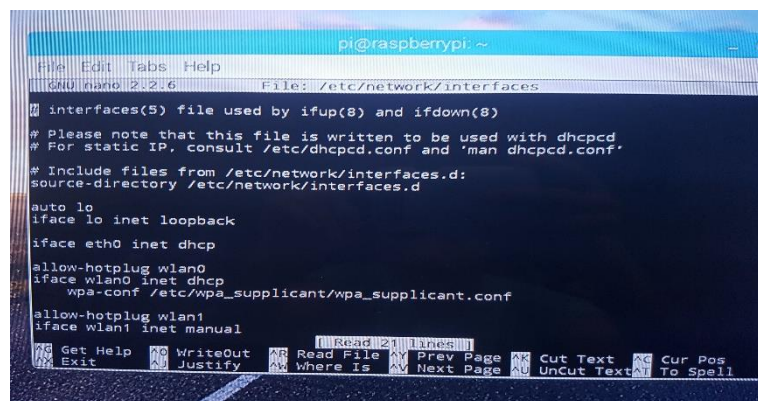
Figure 11 Capteur de température et d'humidité

III-Activation de la connexion wifi de la carte Raspberry Pi

A-Configuration des interfaces

Après l'installation de Raspbian sur notre carte il faudra configurer les interfaces quand va utiliser pour se connecter à internet.

Le fichier `/etc/network/interfaces` permet de configurer les cartes réseau de manière permanente.



```
pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

iface eth0 inet dhcp

allow-hotplug wlan0
iface wlan0 inet dhcp
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

allow-hotplug wlan1
iface wlan1 inet manual

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell
```

Figure 12 Configuration des interfaces

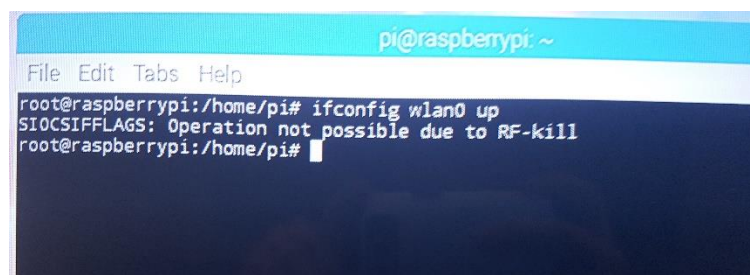
Cette configuration initialisera automatiquement les interfaces « lo » et « wlan0 ».

L'interface « lo » est souvent indispensable au système, il est important de l'initialiser. Elle aura systématiquement l'adresse IP 127.0.0.1.

L'interface « wlan0 » sera configurée avec l'adresse IP 192.168.1.2, le masque de sous réseau 255.255.255.0.

B-Probleme rencontres : RF-kill

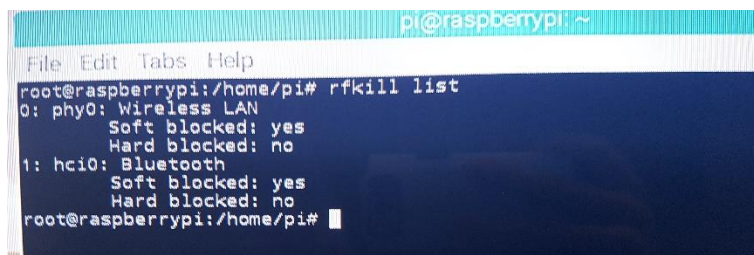
Le RF-kill est un outil qui permet d'activer et désactiver des périphériques sans fil



```
pi@raspberrypi: ~
File Edit Tabs Help
root@raspberrypi:/home/pi# ifconfig wlan0 up
SIOCSIFFLAGS: Operation not possible due to RF-kill
root@raspberrypi:/home/pi#
```

Figure 13 Problème interface dû au RF-kill

Le paramètre list passé en argument liste les périphériques sans fil du PC, c'est à dire autant le wifi que le bluetooth.

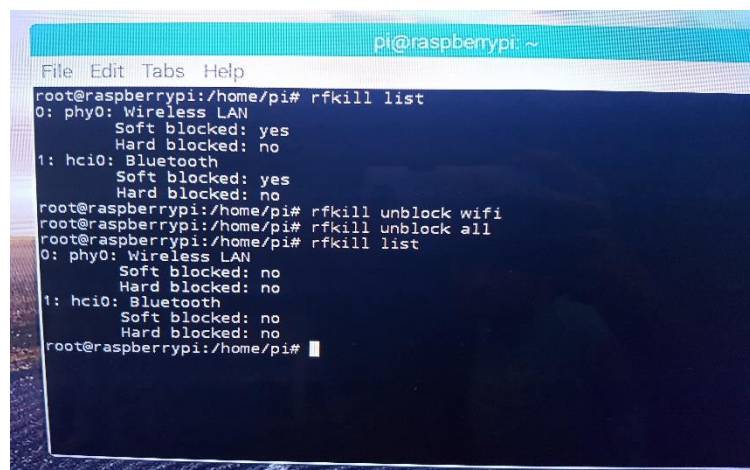


```
pi@raspberrypi: ~  
File Edit Tabs Help  
root@raspberrypi:/home/pi# rfkill list  
0: phy0: Wireless LAN  
   Soft blocked: yes  
   Hard blocked: no  
1: hci0: Bluetooth  
   Soft blocked: yes  
   Hard blocked: no  
root@raspberrypi:/home/pi#
```

Figure 14 Configuration RF-kill

Si après 'Soft blocked' vous avez un 'yes', alors la Wi-Fi (resp Bluetooth) est bloquée de façon logicielle, par contre si le 'Hard blocked' est égale à 'yes' c'est que la Wi-Fi (resp Bluetooth) est bloquée matériellement par un bouton dédié.

C-Activation de la Wi-Fi



```
pi@raspberrypi: ~  
File Edit Tabs Help  
root@raspberrypi:/home/pi# rfkill list  
0: phy0: Wireless LAN  
   Soft blocked: yes  
   Hard blocked: no  
1: hci0: Bluetooth  
   Soft blocked: yes  
   Hard blocked: no  
root@raspberrypi:/home/pi# rfkill unblock wifi  
root@raspberrypi:/home/pi# rfkill unblock all  
root@raspberrypi:/home/pi# rfkill list  
0: phy0: Wireless LAN  
   Soft blocked: no  
   Hard blocked: no  
1: hci0: Bluetooth  
   Soft blocked: no  
   Hard blocked: no  
root@raspberrypi:/home/pi#
```

Figure 15 Activation de Wi-Fi

Les paramètres 'block' et 'unblock' passés en arguments de 'rfkill' permettent d'activer/désactiver un périphérique et les mots-clés générique tel que 'all' et 'wifi' pour tout débloqué.

Pour confirmer les manipulations précédentes il faudra redémarrer la carte 'wlan0' avant d'essayer de se connecter sur internet grâce aux commandes suivantes :

- ifconfig wlan0 down
- ifconfig wlan0 up

```
collisions:0 txqueuelen:1
RX bytes:6288 (6.1 KiB) TX bytes:6288 (6.1 KiB)

wlan0    Link encap:Ethernet  HWaddr b8:27:eb:93:04:59
         inet addr:192.168.1.2   Bcast:192.168.1.255  Mask:255.255.255.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:192  errors:0  dropped:0  overruns:0  carrier:0
         TX packets:76  errors:0  dropped:81  overruns:0  frame:0
         collisions:0  txqueuelen:1000
         RX bytes:42329 (41.3 KiB)  TX bytes:12781 (12.4 KiB)

root@raspberrypi:/home/pi# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
64 bytes from 8.8.8.8: icmp_seq=1 ttl=56 time=6.99 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=56 time=6.23 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=56 time=6.17 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=56 time=6.47 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=56 time=6.50 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=56 time=6.46 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=56 time=6.72 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=56 time=4.03 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=56 time=14.6 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=56 time=11.5 ms
```

Figure 16 Essai ping pour l'accès internet

IV-Activation du serveur SSH de la Raspberry Pi

SSH (pour Secure SHell) désigne à la fois un logiciel et un protocole de communication informatiques. Ce protocole possède par ailleurs la particularité d'être entièrement chiffré. Cela signifie donc que toutes les commandes que vous exécuterez via SSH seront totalement secrètes !

SSH a été créé en 1995 avec pour principal but de permettre la prise de contrôle à distance d'une machine à travers une interface en lignes de commande.

Aujourd'hui, SSH est principalement utilisé à travers l'implémentation libre OpenSSH qui est présente dans la plupart des distributions Linux.

A-Installer un client SSH sur Linux

Pour installer un client SSH, il faudra lancer la commande suivante :

- `sudo apt-get update && sudo apt-get install openssh-client`

B-Ajout d'un nouvel utilisateur

Création d'un groupe contenant des utilisateurs qui seront autorisés à se connecter via SSH:

- « `groupadd sshusers` »

Création d'un nouvel utilisateur :

- « `useradd -m projet` »
- « `passwd projet` »
- « `usermod -a -G sshusers projet` »

C- Configuration de SSH

Seul les utilisateurs appartenant au groupe sshusers sont autorisés à se connecter

- « `nano /etc/ssh/sshd_config` »
- Ajout de la ligne « `AllowGroups sshusers` »

Pour établir la connexion en SSH depuis une autre machine on pourra utiliser :

Sous Windows

Pour Windows, comme souvent, il va falloir passer par l'interface graphique. Pour s'y faire on pourra utiliser Putty :

Dans le champ « Hostname », rentrez l'IP de votre Raspberry Pi sans touches au port ; le numéro du port (22) est le port par défaut de SSH.

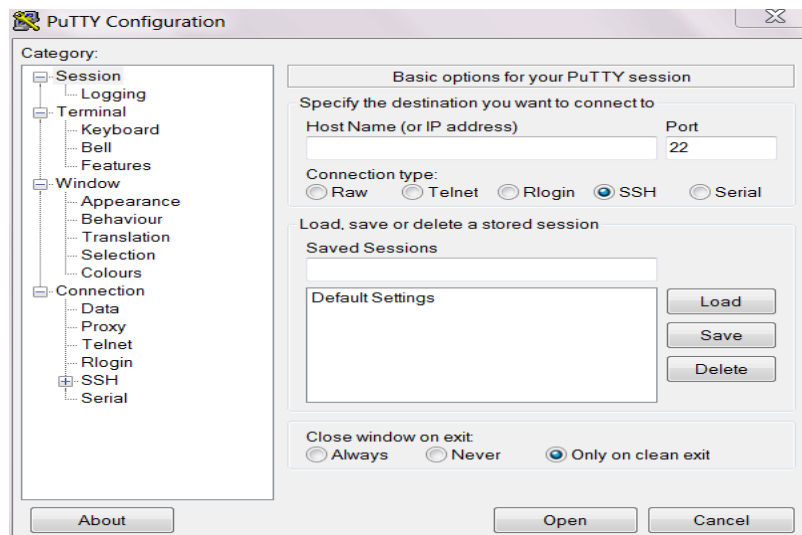


Figure 17 Interface Putty Windows

Une fois ceci fait, cliquez sur le bouton « Open ». Une fenêtre de console devrait s'ouvrir en vous demandant le nom d'utilisateur que vous souhaitez utiliser puis son mot de passe. Une fois ceci fait, vous serez connecté à la Raspberry Pi

Sous GNU/Linux et Mac OS

Ouvrez tout simplement votre terminal, et entrez

➤ `ssh [user]@[IP/Hostname]`

Dans notre cas :

➤ `ssh pi@192.168.X.X`

Entrez votre mot de passe et appuyez sur [Entrée]

Sous Android

Pour se connecter en SSH à la Raspberry via son smartphone, on pourra utiliser l'application [Juice SSH](#).

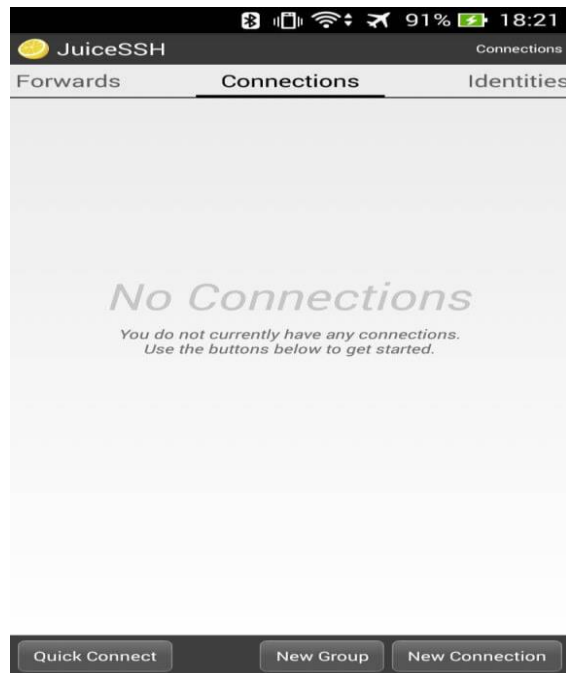


Figure 18 Interface Juice SSH Android

Il faut appuyer sur 'Quick Connect' et suivre ensuite les mêmes étapes que sur Linux.

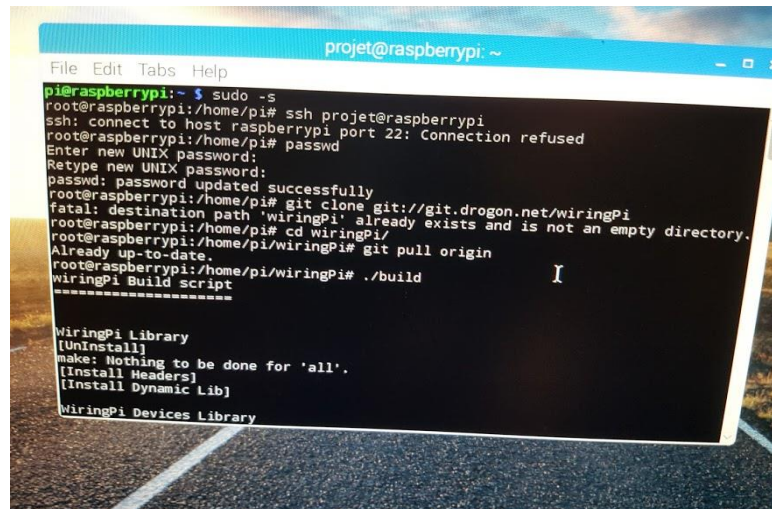
V-WiringPi

WiringPi est une bibliothèque écrite en C permettant l'accès au GPIO du BCM2835 utilisé par le Raspberry Pi.

Le [Raspberry Pi](#) dispose d'un connecteur 26-broches d'Entrée/Sortie d'utilisation générale (appelée GPIO acronyme de "General Purpose Input/Output") qui transporte des signaux et des bus (de données).

WiringPi inclus un utilitaire GPIO en ligne de commande. Cet utilitaire peut être utilisé pour programmer et configurer les broches GPIO. Vous pouvez l'utiliser pour lire et écrire l'état des broches et même contrôler depuis des scripts shell.

A-Installation



```
projet@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~$ sudo -s  
root@raspberrypi:/home/pi# ssh projet@raspberrypi  
ssh: connect to host raspberrypi port 22: Connection refused  
root@raspberrypi:/home/pi# passwd  
Enter new UNIX password:  
Retype new UNIX password:  
passwd: password updated successfully  
root@raspberrypi:/home/pi# git clone git://git.drogon.net/wiringPi  
fatal: destination path 'wiringPi' already exists and is not an empty directory.  
root@raspberrypi:/home/pi# cd wiringPi/  
root@raspberrypi:/home/pi/wiringPi# git pull origin  
Already up-to-date.  
root@raspberrypi:/home/pi/wiringPi# ./build  
wiringPi Build script  
-----  
WiringPi Library  
[Uninstall]  
make: Nothing to be done for 'all'.  
[Install Headers]  
[Install Dynamic Lib]  
WiringPi Devices Library
```

Figure 19 Installation GIT

Il faudra tout d'abord installer GIT :

- `sudo apt-get install git-core`

Git est un système de gestion de versions conçu pour gérer de très gros projets avec rapidité et efficacité.

Git se situe dans la catégorie des outils de gestion de code source distribué. Chaque répertoire de travail Git est un dépôt à part entière avec un suivi complet des révisions et qui ne dépend pas d'un accès réseau ou d'un serveur central.

On récupère la dernière version de WiringPi :

- `git clone git://git.drogon.net/wiringPi`

On se place dans le bon répertoire et On lance la compilation :

- `cd wiringPi`
- `./build`

B-Les commandes de base

Définir un GPIO en mode sortie le GPIO 0 dans cet exemple.

- `sudo gpio mode 4 out`

Changer l'état d'un GPIO :

- `sudo gpio 4 1 (état 1)`
- `sudo gpio 4 0 (état 0)`

VI-Utilisation de DHT_11

A-Installation de de Adafruit et python

Nous allons utiliser la bibliothèque Adafruit DHT11 Python. Nous pouvons télécharger la bibliothèque à l'aide Git

- git clone https://github.com/adafruit/Adafruit_Python_DHT.git

Il faudra ensuite installer Python sur le Raspberry :

- apt-get install build-essential python-dev
- python setup.py install

B-Code python

```
#!/usr/bin/python
```

```
import sys
```

```
import Adafruit_DHT
```

```
sensor_args = { '11': Adafruit_DHT.DHT11,  
                '22': Adafruit_DHT.DHT22,  
                '2302': Adafruit_DHT.AM2302 }
```

```
if len(sys.argv) == 3 and sys.argv[1] in sensor_args:  
    sensor = sensor_args[sys.argv[1]]  
    pin = sys.argv[2]
```

```
else:
```

```
    print('usage: sudo ./Adafruit_DHT.py [11|22|2302] GPIOpin#')  
    print('example: sudo ./Adafruit_DHT.py 2302 4 - Read from an  
AM2302 connected to GPIO #4')  
    sys.exit(1)
```

```
humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
```

```
if humidity is not None and temperature is not None:
```

```
    print('Temp={0:0.1f}* Humidity={1:0.1f}%'.format(temperature,  
humidity))
```

```
else:
```

```
    print('Failed to get reading. Try again!')  
    sys.exit(1)
```

A l'exécution de ce code avec comme paramètre la version du DHT (11 dans notre cas) et le numéro du pin GPIO, la température et le taux d'humidité nous seront affichés sur le terminal.

```
pi@raspberrypi: ~
File Edit Tabs Help
AdafruitDHT.py  AdafruitDHT.py.save
root@raspberrypi:~# cd AdafruitDHT.py
AdafruitDHT.py  AdafruitDHT.py.save
root@raspberrypi:~# ls
AdafruitDHT.py  common.php  functions.php  js
AdafruitDHT.py.save  configuration.php  img  logs
action.php      css  index.php
root@raspberrypi:~# ./AdafruitDHT.py 11 4
Temp=11.0* Humidity=146.0%
root@raspberrypi:~# ./AdafruitDHT.py 11 4
Temp=11.0* Humidity=145.0%
root@raspberrypi:~# ./AdafruitDHT.py 11 17
^C
Traceback (most recent call last):
  File "/AdafruitDHT.py", line 41, in <module>
    humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
  File "build/bdist.linux-armv7l/egg/Adafruit_DHT/common.py", line 94, in read_retry
  File "build/bdist.linux-armv7l/egg/Adafruit_DHT/common.py", line 81, in read_retry
  File "build/bdist.linux-armv7l/egg/Adafruit_DHT/Raspberry_Pi_2.py", line 29, in read_retry
  File "build/bdist.linux-armv7l/egg/Adafruit_DHT/Raspberry_Pi_2.py", line 29, in read_retry
KeyboardInterrupt
root@raspberrypi:~# ./AdafruitDHT.py 11 17
Temp=23.0* Humidity=32.0%
root@raspberrypi:~#
```

Figure 20 Exécution code avec comme résultat la température et l'humidité

VII-Interface web

A-Installation d'un serveur web

Nous avons besoin d'installer un serveur Web sur le Raspberry Pi. Dans notre cas, nous n'avons pas besoin d'une base de données MySQL, juste d'un serveur HTTP et de son extension PHP.

- `sudo apt-get install apache2 php5 libapache2-mod-php5`

```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~$ sudo -s
root@raspberrypi:/home/pi# apt-get install apache2 php5 libapache2-mod-php5
```

Figure 21 Installation serveur web

Il faudra ensuite supprimer le fichier 'index.html' qui se trouve dans '/var/www/html'

B-Mise en place de l'interface web

Dans cette partie on a utilisé une interface web trouver dans un blog (<http://blog.idleman.fr/>) en lui apportant quelque modification

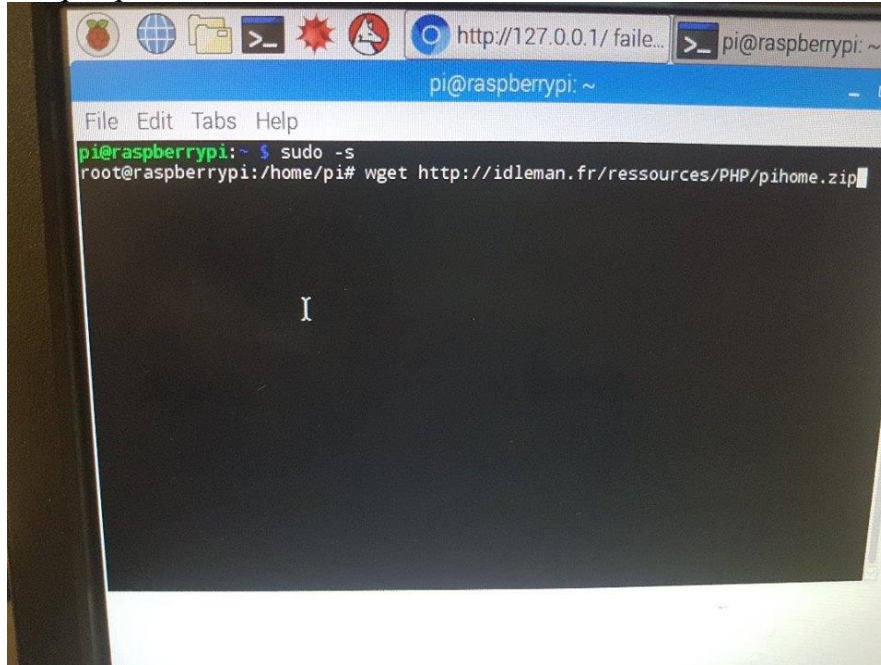


Figure 22 Téléchargement de l'interface web

Après décompression du fichier télécharger, on a déplacé les fichiers dans le répertoire de notre serveur web précédemment installé. En tapant l'adresse IP de notre Raspberry Pi (127.0.0.1) dans le navigateur on aura comme interface la suivante :

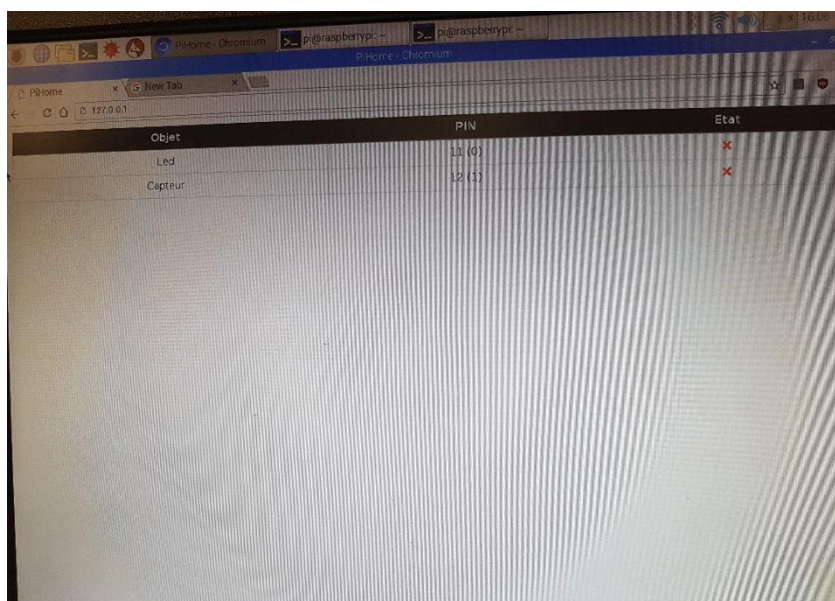


Figure 23 Interface web modifiée

CONCLUSION

Notre projet sur l'internet des objets nous a permis de nous améliorer en autonomie et pour le travail de groupe notamment.

Dans le cadre de notre projet nous avons réussi à réaliser la simulation des capteurs de présence, de températures et d'humidités.

Nous avons réussi à avoir une interface web et nous avons pu contrôler la led que nous utilisons pour simuler un capteur de présence mais malheureusement nous n'avons pas réussi à synchroniser les informations du capteur de température et d'humidité pour les afficher sur l'interface web.

Malgré les difficultés que nous avons eu à certains moments, nous avons en général bien apprécié l'idée de notre projet et le projet en lui-même.

Nous tenons encore à remercier les professeurs qui nous ont aidé et soutenu durant le projet.

GLOSSAIRE

RFID : méthode pour mémoriser et récupérer des données à distance en utilisant des [marqueurs](#) appelés « radio-étiquettes »

Big Data : ensembles de [données](#) qui deviennent tellement volumineux qu'ils en deviennent difficiles à travailler avec des outils classiques de gestion de [base de données](#) ou de gestion de l'information.

ARM : Les architectures ARM sont des architectures matérielles 2 bits (ARMv1 à ARMv7) et 64 bits (ARMv8) développées par *ARM Ltd* depuis 1990 et introduites à partir de 1983 par Acorn Computers.

Raspbian : système d'exploitation de la raspberry

Bootable : Qui contient un programme permettant de démarrer un ordinateur sur cette carte sd, clé usb ou cd-rom

SSH : Protocole de communication sécurisé et application informatique

Putty : émulateur de terminal doublé d'un client pour les protocoles SSH, Telnet, rlogin, et TCP brut

Juice SSH : émulateur de terminal doublé d'un client pour les protocoles SSH sous Android

WiringPi : librairie d'accès au GPIO utilisé par le raspberry

Tables des illustrations

Figure 1 Capteur de présence, un bel exemple de capteur TOR.....	6
Figure 2 Capteur de température.....	6
Figure 3 Pluviomètre.....	7
Figure 4 Capteur d'empreintes digitales.....	7
Figure 5 Raspberry Pi 3B.....	8
Figure 6 Pins et gpio.....	9
Figure 7 Etat de la carte SD après boot.....	11
Figure 8 Contenu de la carte après boot.....	11
Figure 9 Circuit de led.....	11
Figure 10:DHT11.....	12
Figure 11 Capteur de température et d'humidité.....	12
Figure 12 Configuration des interfaces.....	13
Figure 13 Problème interface dû au RF-kill.....	13
Figure 14 Configuration RF-kill.....	14
Figure 15 Activation de Wi-Fi.....	14
Figure 16 Essai ping pour l'accès internet.....	15
Figure 17 Interface Putty Windows.....	16
Figure 18 Interface Juice SSH Android.....	17
Figure 19 Installation GIT.....	18
Figure 20 Exécution code avec comme résultat la température et l'humidité.....	20
Figure 21 Installation serveur web.....	20
Figure 22 Téléchargement de l'interface web.....	21
Figure 23 Interface web modifiée.....	21