

# Projet TIC - Documentation

Damien Goetschi

3 octobre 2016

## 1 Introduction

Cette documentation sert à configurer la machine virtuelle qui communiquera avec le robot Thymio, et d'installer le serveur avec lequel il communiquera.

### 1.1 Glossaire

**Thymio-II** : Robot éducatif créé par l'EPFL

**Aseba** : Language de programmation créé par l'EPFL et utilisé de base pour le Thymio-II

**AsebaHttp2** : Programme permettant d'accéder au code Aseba à travers Http. Utilisé par la librairie Python thymiolib fournie

## 2 Configuration de la machine virtuelle

Login : projettic

Password : tic2016

Vous pouvez changer le mot de passe en utilisant la commande `passwd`

### 2.1 Avec VirtualBox

Si vous n'avez pas VirtualBox sur votre machine, vous pouvez le télécharger ici :


<https://www.virtualbox.org/wiki/Downloads>

Pensez à aussi télécharger l'Extension Pack qui est indispensable pour gérer les périphériques USB.

Une fois VirtualBox installé, vous pouvez ouvrir le fichier VBox\_UbuntuProjetTIC.ova, la machine virtuelle va être importée dans VirtualBox.

Il faut encore ajouter le dongle Wireless du Thymio dans la machine virtuelle, pour le faire, branchez le dongle Wireless et sélectionnez la machine virtuelle qui a été importée, puis allez dans la partie USB des paramètres.

Activez le contrôleur USB et sélectionnez "USB 2.0 (EHCI) Controller"

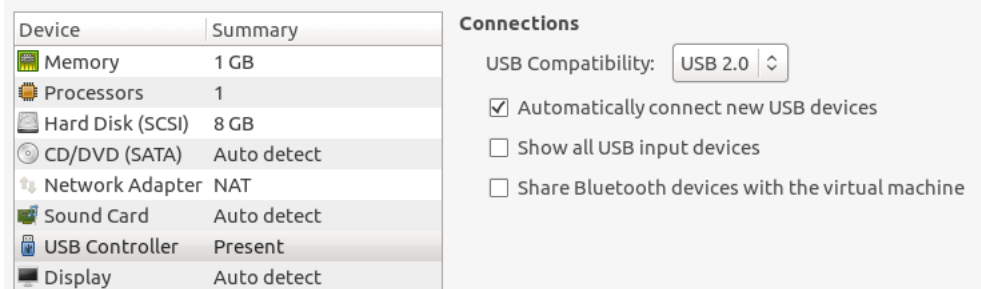
Cliquez sur le bouton  pour ajouter une nouvelle règle et sélectionnez "Mobsya.org Thymio-II Wireless". Si votre machine hôte est un Windows, il se peut que le nom soit "USB Serial Port Vous pouvez fermer les paramètres et démarrer la machine virtuelle.

## 2.2 Avec VMWare

Si vous n'avez pas VMWare sur votre machine vous pouvez le télécharger en passant par le site du service informatique (<https://sinfo.hefr.ch/Pages/default.aspx>)

Une fois VMWare installé, ouvrez le puis aller dans "File - Open" et choisissez le fichier VMWare\_UbuntuProjetTIC.ova puis l'endroit où vous voulez l'importer.

Dans les configurations de la machine virtuelle, vérifiez que le controller USB soit présent et qu'il est configuré pour que les nouveaux périphériques soient connectés à la machine virtuelle. Ainsi dès qu'on branchera le dongle wireless du Thymio, il se connectera à la machine virtuelle et pas à la machine hôte.



Ensuite, vous pouvez fermer les paramètres et démarrer la machine virtuelle.

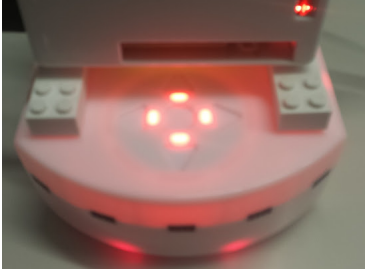
## 2.3 Tester le bon fonctionnement du Thymio

Pour tester que tout fonctionne, démarrez la machine virtuelle et allumez le robot. Une fois la machine virtuelle prête, branchez le dongle wireless du Thymio et lancez le script python d'exemple se trouvant dans le dossier **Example**.

Pour le lancer :

```
python3 Example/StandAloneExample.py
```

Si tout fonctionne, le robot va d'abord allumé les lumières sur les boutons du dessus pendant une demi-seconde (signe du chargement d'un programme)



Ensuite il doit allumer toutes ses lumières, avancer d'environ 5cm et éteindre toutes ces lumières. Il va également afficher sur la console l'état de tous ses capteurs.

```
pi@raspberrypi ~/Example $ python3 StandAloneExample.py
[397, 385]
[0, 0, 0, 0, 0, 0, 0]
[586, 742]
0
0
0
0
0
0
[2, 0, 22]
25
279
```

Une fois le serveur installé, le script ClientServerExample.py peut également être testé, mais avant ça, il faut éditer la variable IP de ce dernier à la ligne 14 par l'adresse IP du serveur.

```
ip = "192.168.1.49"
```

Ensuite, on peut lancer le script de la même manière que le précédent :

```
python3 Example/ClientServerExample.py
```

## 3 Installation Serveur

### 3.1 Fichiers nécessaires

- flask\_dev.conf (Configuration d'un site apache2 utilisant Flask)
- start.wsgi (Web Server Gateway Interface permettant d'interfacer le serveur apache2 avec Flask)
- server.py (Application Flask servant des pages pour les url données)

### 3.2 Installation de Apache2 et Flask

Il faudra copier les fichiers nécessaires sur le serveur, par exemple en utilisant SCP.

Premièrement il faut installer Apache2 ainsi que le gestionnaire de paquets Python (PIP)

```
sudo apt-get install apache2 python3-pip
```

Une fois installé, on peut installer Flask à partir de PIP :

```
sudo pip3 install Flask
```

Une fois Flask installé, il faut encore le module Apache2 qui va faire le lien entre l'application Flask et le serveur Apache2

```
sudo apt-get install libapache2-mod-wsgi-py3
```

Ensuite, on peut créer le dossier `/var/www/flask_dev` et y copier les deux fichiers `start.wsgi` et `server.py` qui sont le script d'interfaçage entre Apache2 et Flask et l'application Flask. Le fichier `flask_dev.conf` doit également être copier dans le dossier contenant les sites Apache `/etc/apache2/sites-available`

```
sudo mkdir /var/www/flask_dev
sudo cp start.wsgi server.py /var/www/flask_dev/
sudo cp flask_dev.conf /etc/apache2/sites-available/
```

Maintenant, il faut activer le module wsgi dans Apache, activer notre site flask et désactiver le site par défaut, pour ce faire :

```
sudo a2enmod wsgi
sudo a2ensite flask_dev.conf
sudo a2dissite 000-default.conf
```

Une fois fait, il faut redémarrer le serveur pour qu'il prenne en compte ces modifications

```
sudo service apache2 restart
```

### 3.3 Test du serveur

Pour tester le serveur, depuis n'importe quel navigateur rentrez l'adresse ip du serveur, si tout fonctionne la réponse reçue sera : `Hello World!`

Ensuite on peut tester également ce qui sera renvoyé au programme test client-server du robot en entrant comme url dans le browser `http://IP_SERVER/nextfunc` si tout se passe bien la réponse sera quelque chose sous cette forme mais avec des valeurs différentes :

```
{
  "funcname": "test2",
  "values": [
    21,
    2,
    11
  ]
}
```

### 3.4 Documentation

La documentation de Flask peut être retrouvée ici : <http://flask.pocoo.org/docs/0.10/>

La documentation de Apache2 peut être retrouvée ici : <https://doc.ubuntu-fr.org/apache2>