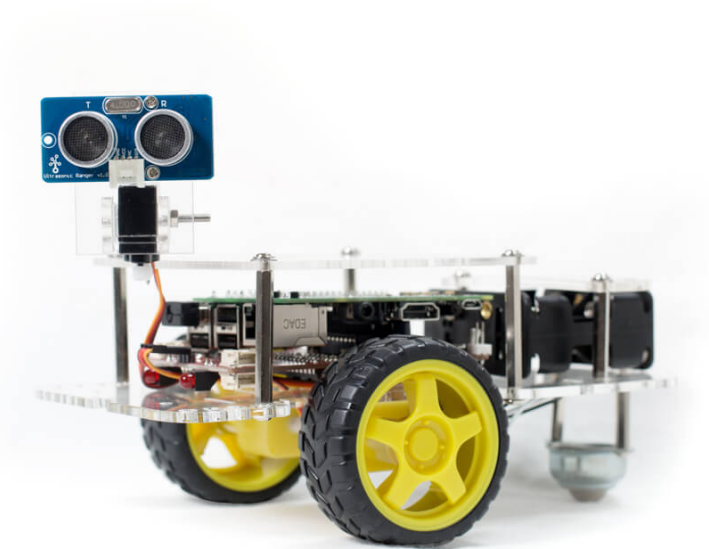


# Grand Projet Robot Gopigo

---



## Objectif

---

L'objectif du projet est de réaliser un système automatisé de contrôle du robot et de réaliser une présentation du projet. Les robots Gopigo sont équipés de moteurs et d'un capteur de distance afin de détecter les obstacles. L'objectif de ce projet sera de programmer un algorithme de contrôle automatique du robot. Pour ce faire, les étudiants devront concevoir et définir un protocole expérimental, développer, tester leur programme et présenter leur travail. Les étudiants organisés en binômes commenceront par étudier le robot, puis choisiront un projet de contrôle automatique et feront la description du protocole expérimental.

## Séances

---

Vous déposerez sur Moodle > Rendus 15 minutes avant la fin des séances le zip `gp-nom1-nom2-sX.zip` en remplaçant `x` par le numéro de séance. Par exemple, avant la fin de la 1ère séance vous déposerez `gp-nom1-nom2-s1.zip`, avant la fin de la séance 2 vous déposerez `gp-nom1-nom2-s2.zip`.

1. Mise en place de l'environnement de développement, prise en main du Gopigo, documentation et tutoriels (2h).  
Choix du projet, du protocole expérimental et validation. Réalisation et rendu définitif du slide de présentation "Objectif".
2. Conception: Réalisation du protocole expérimental, des diagrammes d'activité et de classes.
3. Développement du programme en langage Python.
4. Application du protocole expérimental. Mesures, résultat. Rendu final du programme dans `gp-nom1-nom2-s4.zip`.
5. Présentation et rendu final de la présentation dans `gp-nom1-nom2-s5.zip` avant de présenter.

# Protocole expérimental

---

Le protocole expérimental est réalisé pendant la phase de conception avant de commencer le développement. Il définit la méthode qui sera utilisée pour vérifier le bon fonctionnement du projet. Il définit les résultats attendus avec leur marge d'erreur en fonction de la précision des mesures des capteurs, du matériel et de l'environnement. Il définit aussi Le nombre d'essais à réaliser pour avoir des informations statistiques sur les résultats.

Pour un système de freinage d'urgence, le robot avance à vitesse maximal pendant 10m et freine. On mesure la distance d'arrêt qui doit correspondre avec la distance calculée avec sa marge d'erreur.

## Présentation

---

Vous écrierez votre présentation dans un fichier Powerpoint `gp-gpg-nom1-nom2.pptx` de exactement 7 diapositives et durera 10 minutes au maximum. Choisissez un thème le plus neutre possible, sans image, avec un fond blanc. La taille des polices de caractères doit être suffisamment grande pour être lisible de loin.

1. Page de garde: Noms, groupe, module, enseignant, date, titre, promotion
2. Objectifs: Description des étapes (>2) avec durées prévues et réelles (mis à jour pour la séance 5).
3. Protocole expérimental (avec calculs).
4. Statistiques: nombre de lignes non vides (précision de 10 lignes), fichiers, classes et methodes.
5. Diagramme de classe Uml.
6. Diagramme d'activité Uml.
7. Une vidéo de l'expérience de 1 minute en résolution 1920×1080.

## Environnement de développement:

---

### Hardware

---

Raspberry Pi + GoPiGo3 + Capteur de distance à ultrason + servomoteur

### Software

---

Raspbian Linux + Python3 + bibliothèques Python GoPiGo3

### Spécifications techniques du GoPiGo3

---

<https://www.dexterindustries.com/GoPiGo/learning/hardware-port-description/>

### Connexion au réseau Wifi de la salle L20

---

- nom du réseau: L20ROBOTSWIFI
- mot de passe: WIFIROBOTSL20

### Connexion au robot

---

L'adresse ip d'un Gopigo est `192.168.1.X` où `x` est le numéro du Gopigo. Allumez le robot nommé `dex111` et allez à `http://192.168.1.111` Cliquez sur le bouton `vnc` pour accédez au bureau Linux du GoPiGo3.

### Configuration de VNC

---

Cliquez sur le petit bouton à gauche, puis sur Settings, cocher `Automatic Reconnect` . Sélectionnez `Advanced > Scaling mode > Local Scaling`

## Configuration de l'affichage

---

Pour ouvrir le Menu, cliquez dans le coin en haut à gauche de l'écran, sur la framboise. Dans `Menu > Préférence > Appearance settings > Defaults` , choisir `Set Default for medium screen` et choisir un fond d'écran foncé pour faire apparaître les textes des icônes.

## Répertoire de travail du Gopigo

---

Avec un clic droit sur le bureau, créez le répertoire `gp-gpg-nom1-nom2` .

## IDE

---

Démarrez l'éditeur de code Geany qui est installé sur le Gopigo: `Menu > Programing > Geany` Enregistrez le fichier avec le nom `led.py` dans votre répertoire `gp-gpg-nom1-nom2` avec `ctrl-s` .

## Transfert de fichiers

---

### Transfert de fichier avec WinScp

<https://winscp.net/eng/download.php>

```
Protocole: sftp
Hôte: 192.168.1.111
Port:22
Utilisateur: pi
Mot de passe: robots1234
```

### Divers

- On peut utiliser une clé usb mais les prises usb sont peu accessibles.
- On peut transférer du texte dans le clipboard de VNC (bouton à gauche de l'écran VNC).

## Faire clignoter une led

---

En utilisant le clipboard de VNC copier dans `led.py` (dans Geany) et essayez le code suivant:

```
import time
import easygopigo3

gopigo = easygopigo3.EasyGoPiGo3()

for i in range(100):
    gopigo.blinker_on('left')
    time.sleep(0.1)
    gopigo.blinker_off('left')
    time.sleep(0.1)
```

# Documentation GoPiGo3

---

<https://gopigo3.readthedocs.io>

## Tutoriels

---

Pour récupérer les programmes des tutoriels, reconnectez vous au réseau Wifi IONIS et copiez les sur le gopigo avec WinScp. Reconnectez-vous au réseau L20ROBOTSWIFI pour essayer et modifier les programmes

### Tutoriel 1. Faire clignoter une led

---

<https://gopigo3.readthedocs.io/en/master/tutorials-basic/led.html>

### Tutoriel 2. Mesurer la distance

---

[https://gopigo3.readthedocs.io/en/master/tutorials-basic/distance\\_sensor.html](https://gopigo3.readthedocs.io/en/master/tutorials-basic/distance_sensor.html)

### Tutoriel 3. Déplacer le Gopigo3

---

<https://gopigo3.readthedocs.io/en/master/tutorials-basic/driving.html>

## API GoPiGo3 Basic

---

<https://gopigo3.readthedocs.io/en/master/api-basic/index.html>

### Bibliothèque EasyGoPiGo3

---

<https://gopigo3.readthedocs.io/en/master/api-basic/easygopigo3.html>

### Bibliothèque UltrasonicSensor

---

<https://gopigo3.readthedocs.io/en/master/api-basic/sensors.html#ultrasonicsensor>

### Liste des fonctions

---

<https://gopigo3.readthedocs.io/en/master/api-basic/structure.html#functions-short-list>