



IOT PROJECT

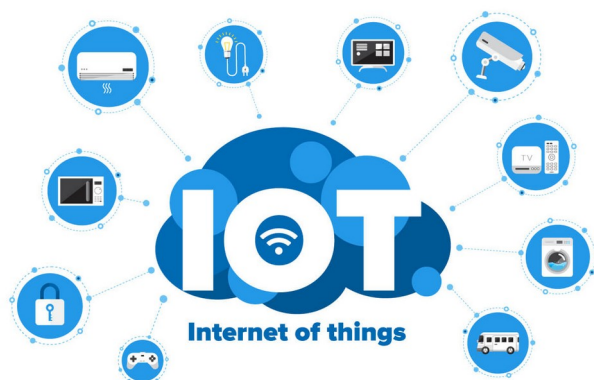
2022/2023

TP : TP 1&2 PROG. JAVA

Classe : D-EEA

CONCEPTS:

- les bases de la poo (class, attribut, méthodes, constructeurs, public, private, static),
- Héritage, redéfinition et polymorphisme.



Abréviations

IOT Internet Of Things

mvt mouvement

syst. système

TAF Travail à faire

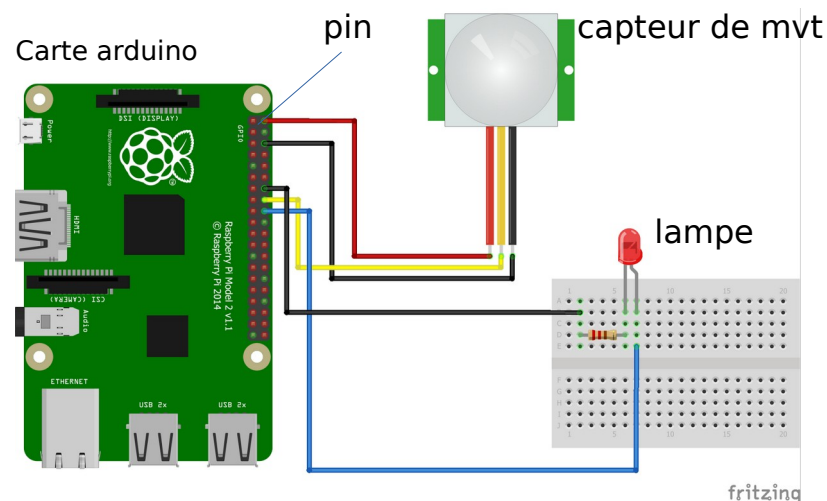
INTRODUCTION

Le projet consiste à implémenter une simulation d'un système iot avec java et en utilisant la poo. Le système dit est composé d'une carte controleur (arduino), de capteurs et d'actionneurs.



Pratiquement, les capteurs et les actionneurs sont connectés à la carte via ses pins intégrés. Tout capteur peut envoyer à tout moment un signal elect. de 5v au pin y connecté. Tout actionneur peut recevoir à tout moment un signal de la part du pin auquel il est connecté.

Une fois les devices sont connectés à la carte (à travers ses pins), un programme est allimenté à la carte pour tracer les relations entre les capteurs et les actionneurs, et ceci, en reliant l'activation (value=1) des pins actionneurs à celle des pins capteurs



Exp.:

- si le capteur de mouvement est connecté à la carte à travers le pin n°20
- et l'actionneur de lampe est connecté à la carte au pin n° 35

- Alors pour relier entre le capteur de mvt et l'actionneur de lampe de telle sorte que l'orsque quelqu'un passe la lampe s'allume, on doit programmer la carte comme suit:
 si pin20 == 1 alors pin35=1 // allumer la lampe l'orsqu'il ya du mvt.
 si pin20 == 0 alors pin35=0 // Éteindre la lampe si aucun mouvement.

REGLES DE GESTION

R1. un capteur possède les infos suivantes:

un id, entier;
 un nom, texte;
 un prix, entier;
 analog_num, texte;
 l'etat, texte
 le prix, entier;

R2. un actionneur possède les infos suivantes:

un id, entier;
 un nom, texte;
 un prix, entier;
 analog_num, texte; ["analogic", "numeric"]
 l'etat, texte ['enabled', 'disabled']
 le prix, entier;
 l'objet y connecté, texte [lampe, clim, ventilateur, ...]

R3. une carte possède les info suivantes:

un id, entier;
 un nom, texte;
 un nbpins entier;
 un prix entier;
 une collection de pins, arraylist;
 une collection de capteurs, arraylist;
 une collection d'actionneurs, arraylist;

R4. Tout pin de la carte possède les info suivantes:

un num, entier;
 une valeur, entier [0, 1];
 un type analog_num, entier [analogic, numeric]

un etat; // deprecated

id_connected : l'id du capteur/actionneur connecté à ce pin, entier ;

R5. Tout capteur/actionneur peut être connecté à tout pin de la carte mais un seul à la fois. La relation entre les capteurs/actionneur et les pin est 1:1

R6. un pin peut être relié (liaison soft) à un autre pin (cette règle sera traitée dans des versions plus avancées des tp)

TAE:

La solution doit fournir les traitements suivants:

I. Livrables de la solution v1: basic de la poo (class, attribut, méthodes, constructeurs, public, private, static)

1. créer une carte avec un nb de pins donnée au constructeur.
2. créer des capteurs et des actionneurs et les ajouter à la carte.
3. connecter un capteur/actionneur à un pin : placer l'id du capteur/actionneur dans l'info id_connected du pin.
4. déconnecter un capteur/actionneur : chercher le pin connecté vider l'id_connected du pin connecté.
5. afficher l'état de la carte en listant ses info principales avec la liste des capteurs et des actionneurs (méthode printEtat()).

II. Livrables de la solution v2: Héritage, redéfinition et polymorphisme

1. Classe mère/class fille : Créer une classe Device qui regroupe les attributs et les méthode en commun entre les classes Actionneur et Capteur. Ajouter le constructeur et le toString de Device
2. Héritage des constructeurs / héritage des méthodes / redéfinition: Reconstruire les constructeurs et les toString des classes filles Capteur et Actionneur.
3. Polymorphisme: transformer la collections des capteurs et celle des actionneurs de la classe carte en une seule collection en utilisant la classe Device.
4. Modifier l'affichage de l'état de la carte. (simplifier encore le code et parcourir un seul arrayliste de devices au lieu des deux arraylistes de capteurs et d'actionneurs)