

# Cahier des charges du projet Tuteuré

---

## Objet connecté (I.O.T)

Louveau Benoît   Lenoir Adrien   Mahiou Yannis  
Salabert Gabin   Raffray Elouan

I.U.T - Département génie informatique  
Université d'Auvergne

Projet de groupe, Groupe n°, 2017

# A - Introduction

## Contexte

Ce projet s'inscrit dans le domaine de l'IOT (Réseau regroupant des objets connectés entre eux, les autorisant à communiquer sur Internet). L'objet étudié contrôle le fonctionnement de pompes à chaleur, à l'aide d'un thermostat configurable via une interface Web, et de différents capteurs, gérés par une carte Arduino. L'enjeu de ce projet est donc une solution peu coûteuse, libre et sécurisée utilisée pour de la domotique.

## Historique

La domotique rassemble les techniques et technologies permettant de centraliser et automatiser les différents systèmes présents au sein d'une habitation. Cette technologie est présente depuis les années 80, mais ne s'était pas répandue. Aujourd'hui elle se démocratise mais son coût reste encore élevé (4 à 10% du coût de construction de la maison concernée).

## B - Descriptif de la demande :

- ▶ Les objectifs.
- ▶ Le Produit.
- ▶ Les fonctionnalités.
- ▶ Les critères d'acceptabilité et de réception.

## B - 1) Objectifs :

- ▶ Analyser le patron de conception et de sécurité actuels.
- ▶ Implémenter un/des patron(s) de conception.
- ▶ Implémenter un/des patron(s) de sécurité.

## B - 2) Produit :

- ▶ L'objet est un thermostat de pompe à chaleur embarquant une carte arduino ESP 8266, une led infrarouge, un capteur de présence. Elle héberge un serveur web maison contenant une interface web pour l'aperçu et la configuration et diffuse son réseau en WIFI.

## B - 3) Fonctionnalités :

- ▶ Différents modes : Froid, Economique, Confort, Marche forcée (pour choisir la température du thermostat) et Auto. Les modes permettent une régulation de la température automatique en fonction du choix.
- ▶ Le mode Auto utilise un calendrier.
- ▶ La pompe à chaleur se stoppe automatiquement si la température dépasse un seuil donné
- ▶ La pompe à chaleur démarre automatiquement si la température passe en dessous d'un seuil donné

## B - 4) Fonctionnalités :

Suite..

- ▶ Utilise le protocole PIR : si elle détecte quelqu'un (au moins 3 fois en 3 minutes) et que le mode économique est sélectionné : elle déclenche le mode confort pour une durée de 30 minutes. Elle conserve ce mode si elle re-détecte quelqu'un, sinon elle rebascule en mode Eco après 30 minutes.
- ▶ Connectée à un serveur NTP pour connaître la date et l'heure.
- ▶ Accessible via une interface web.
- ▶ Mise à jour du Firmware *Over The Air*.
- ▶ Diffusion des informations (mode, température, PIR) vers DomoticZ.

## B - 5) Critères d'acceptabilité et de réception :

- ▶ Le produit doit comprendre moins de vulnérabilités.
- ▶ Le code doit être amélioré dans sa conception.



## C - Contraintes :

### ▶ 1- Contraintes de coûts :

- Ordinateurs mis à disposition (0€)
- Ressources humaines (5 étudiants)
- Arduino ESP8266 (5€)
- Capteur IR (1€)
- Boitier imprimé 3D (0,50€)
- Budget : (7€)

### ▶ 2- Contrainte de délais :

Livraison finale du projet : 5 Mars 2018 (3 semaines avant la soutenance finale).

Livraison du projet à mi-période (avant la soutenance) et soutenance intermédiaire : 15 Janvier 2018.

### ▶ 3- Autres contraintes :

Spécifier les éventuelles autres contraintes à prendre en compte dans le cadre du projet (normes techniques, clauses juridiques, etc.).

## D - Déroulement du projet :

### ► 1- Planification :

- Analyse du code existant.
- Analyse de différents patrons de conception et de sécurité.
- Détection de vulnérabilités (données ou celles trouvées par nos recherches) à l'aide d'outils de pénétration.
- Implémentation de patrons susceptibles d'améliorer l'architecture globale du projet et de corriger les vulnérabilités trouvées.

### ► 2- Ressources :

#### Ressources humaines :

- 5 étudiants
- 1 encadrant

#### Ressources matérielles :

- Cartes WEMOS D1 mini Pro : Base ESP-8266EX
- Différents capteurs, PC et AP Wifi

## E/F - Authentication/Annexes :



S. Salva.

*Thermostat for Heatpump.*

<https://github.com/sasa27/OpenThermostat>



Public.

C++ documentation, Arduino documentation.

<https://www.arduino.cc/reference/en/>, 2000.

<http://en.cppreference.com/w/>, 2000.