

Cahier des charges :

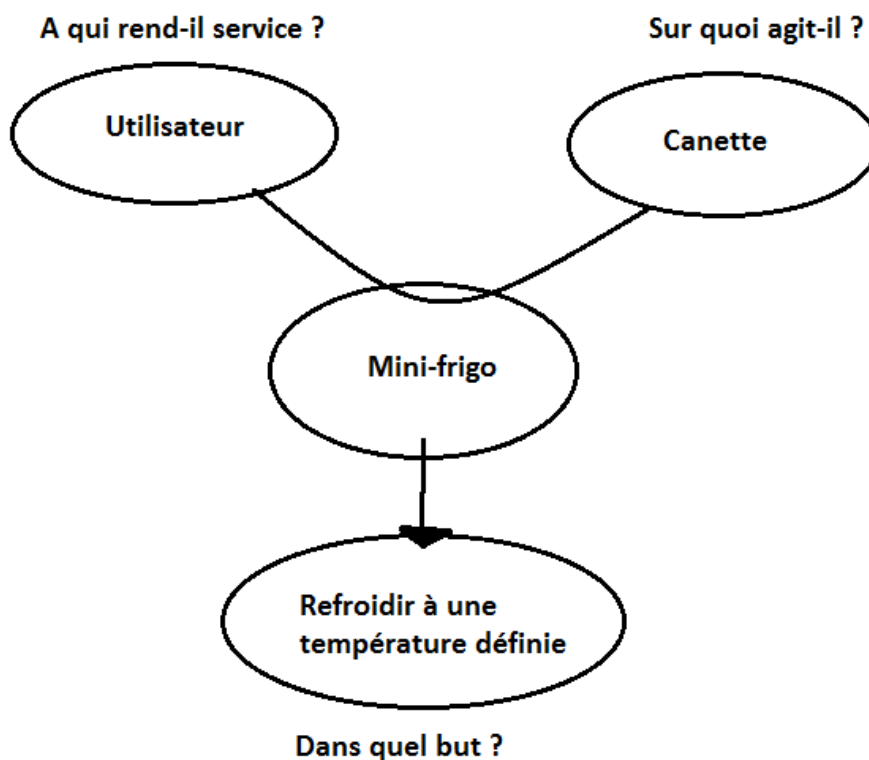
Projet PMF

1) Description du contexte actuel

Dans ce projet nous nous mettons dans la peau d'un jeune qui veut construire un petit frigo connecté. Pour arriver à ce stade, nous sommes passés par 4 prosits : « Mesures physiques », « Automatisation », « Thermodynamique » et « Java ». Grâce à ces prosits nous devons arriver à réaliser ce projet.

2) Analyse du besoin :

Bête à corne :



Le mini-frigo de ce projet doit pouvoir refroidir une canette que l'utilisateur aura placé à l'intérieur.

L'environnement de ce projet sera dans une pièce d'un bâtiment.

Le besoin existe dans ce projet car l'utilisateur ne veut pas boire sa canette à température ambiante mais à une température bien précise

3) Définition du problème :

Afin de réaliser ce petit frigo il nous faut respecter certaines contraintes et fonctionnalités :

- Contraintes :

- Utilisation d'un module Peltier afin de refroidir
- Utilisation d'un Arduino afin de contrôler le système

- Fonctionnalités :

- Le frigo doit pouvoir être refroidi grâce au module Peltier
- On doit pouvoir lire la température extérieure, intérieure et l'humidité
- On doit pouvoir choisir la température voulu pour le frigo : appelée température de consigne
- Le frigo doit être contrôlé par ordinateur
- L'ordinateur agit sur l'Arduino qui lui renvoi les données
- L'IHM Java doit pouvoir afficher la température extérieure, intérieure et l'humidité
- On doit pouvoir choisir la température de consigne grâce à l'IHM
- L'IHM doit pouvoir afficher un graphique des températures internes du frigo
- Le programme doit pouvoir nous alerter en cas de condensation
- Le programme doit pouvoir nous alerter en cas de l'oubli de la porte ouverte
- Le programme doit gérer la formation de condensation

4) Objectif visé :

On pourra considérer ce projet réussi si et seulement si toutes les fonctionnalités sont respectées.

5) Choix Techniques :

MVC :

Nous avons, pour ce projet, décidé d'adapter le MVC afin qu'il soit le plus stable possible. Grâce aux conseils de nos tuteurs, nous avons choisi de ne pas faire communiquer le Modèle avec la Vue pour éviter le crash de l'application dans le cas où le Modèle envoie trop de données à la Vue. A la place de ça, le contrôleur se charge de mettre à jour le Modèle et la Vue. Le contrôleur peut donc updaten la Vue plus intelligemment.

Electronique :

Pour les capteurs de température, nous avons choisi de prendre des thermistances CTN (Coefficient de Température Négatif) au lieu de diodes car elles ont une meilleure sensibilité. De plus elles sont moins fragiles que les diodes en cas de hausse du courant. Il faut prendre des précautions avec les diodes car, si le courant est trop élevé, l'effet Joules fait chauffer la diode et perturbe donc les résultats.

Nous n'avons pas utilisé d'ampli op car la résolution de l'Arduino suffisait avec la thermistance pour avoir des résultats exploitables.

Le choix des ports de l'Arduino et du circuit électronique a été décidé avec la classe. Tout le monde utilise le même circuit (contenu dans un mini-frigo).

6) Retour d'expérience du projet :

Grâce à ce projet nous avons acquis des compétences dans les domaines suivants :

- Java :

- utilisation de la librairie RXTX
- utilisation de la librairie FreeChart
- utilisation de la librairie Swing

- Electronique :

- utilisation de thermistances et de sa formule (steinhart-hart)
- utilisation de capteur d'humidité
- utilisation d'amplificateur opérationnel
- utilisation d'une diode comme capteur de température
- l'utilisation de la chaine d'acquisition des données (capteur, amplification, filtrage,

can)

- Thermodynamique :

- détection d'un point de rosée
- compréhension des phénomènes présents lors de la modification d'un système
- étude de la base de la thermodynamique

- Automatisation :

- modélisation et simulation d'un système
- compréhension des systèmes de boucles

7) Test Fonctionnel :

Nous n'avons réalisé que très peu de tests unitaires. C'est pourquoi nous avons fait des tests fonctionnels.

Fonctions :	Attente de l'objectif :
Utilisation d'un module Peltier afin de refroidir	OK
Utilisation d'un Arduino afin de contrôler le système	OK
Le frigo doit pouvoir être refroidi grâce au module Peltier	OK
On doit pouvoir lire la température extérieure, intérieure et l'humidité	OK
On doit pouvoir choisir la température voulue pour le frigo	OK
Le frigo doit être contrôlé par ordinateur	OK
L'ordinateur agit sur l'Arduino qui lui renvoi les données	OK
L'IHM Java doit pouvoir afficher la température extérieure, intérieure et l'humidité	OK
On doit pouvoir choisir la température de consigne grâce à l'IHM	OK
L'IHM doit pouvoir afficher un graphique des températures internes du frigo	OK
Le programme doit pouvoir nous alerter en cas de rosée	OK
Le programme doit gérer la formation de rosée	OK

8) Bilan :

Ce projet s'est plutôt bien déroulé pour nous. Nous pensons néanmoins que notre gestion de projet peut être améliorée. Nous n'avons pas forcément pris le temps de bien la faire au début et cela nous a fait perdre du temps dans le projet.

Malgré cela, nous pensons que notre projet correspond bien aux attentes perçues par le groupe.