

Smart Environments - AutoMiam

Spécifications

Encadrant : Lionel Médini

Étudiants : Titouan Knockaert et Gaspard Goupy

M2 Intelligence artificielle – Université Claude Bernard Lyon 1



Automiam

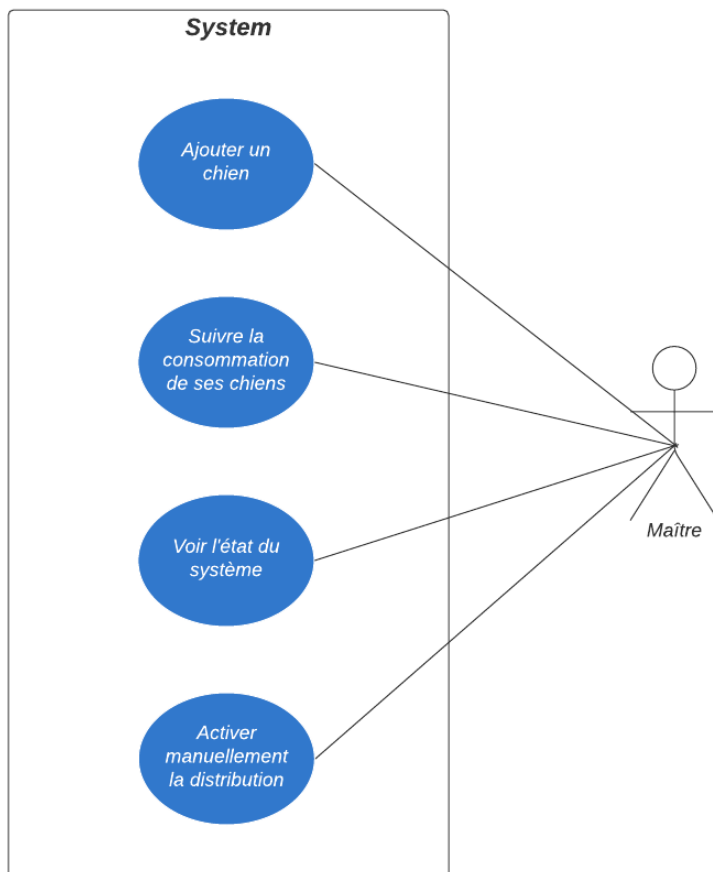
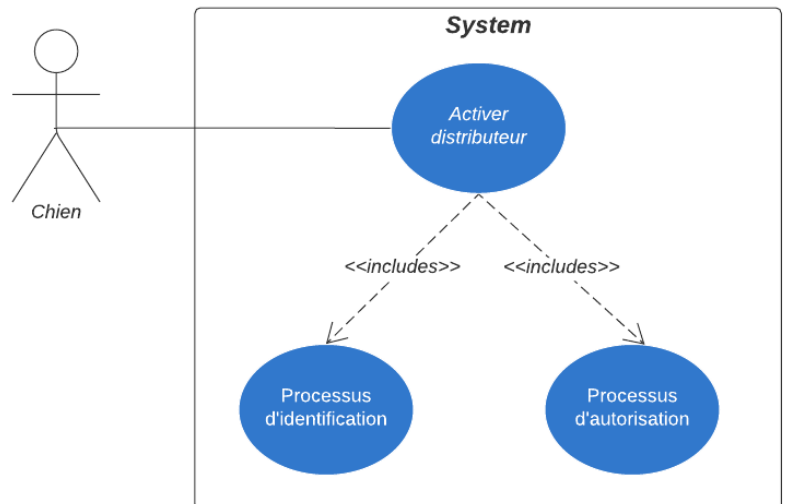
Table des matières

Cas d'utilisation	2
Architecture matérielle	3
Architecture logicielle	4
APIs	6

Cas d'utilisation

Les cas d'utilisation du système sont relativement simples et peu nombreux. Ils ne nécessitent pas d'interactions complexes acteurs - système.

La figure 1. ci-contre présente un cas d'utilisation centré sur le chien.



La figure 2. ci-contre présente un cas d'utilisation centré sur le maître.

Architecture matérielle

Le système nécessite plusieurs composants pour fonctionner :

- Carte Arduino modèle “Uno”
- Des capteurs : caméra et photorésistance
- Un actionneur : servomoteur
- Ordinateur

La figure 3. détaille les relations entre les différents composants.

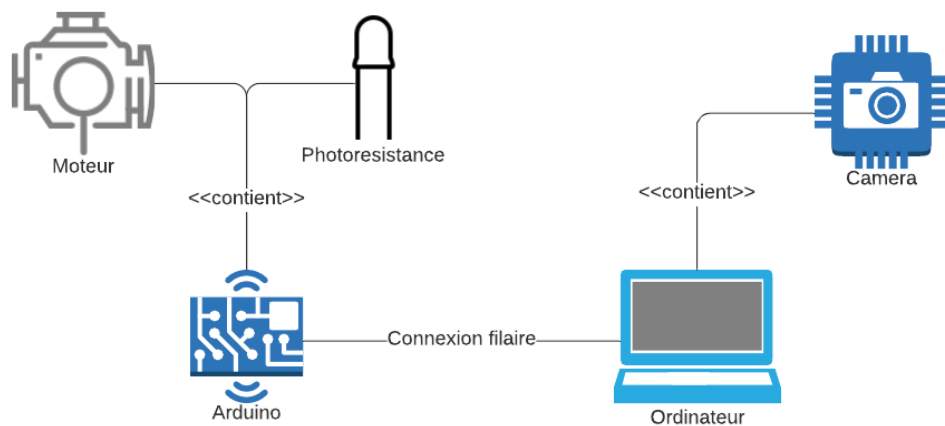


Figure 3. Architecture matérielle

Le branchement de l'arduino est décrit figure 4. À noter que l'on peut ajouter un condensateur pour éviter les chutes de tension, selon le servomoteur utilisé.

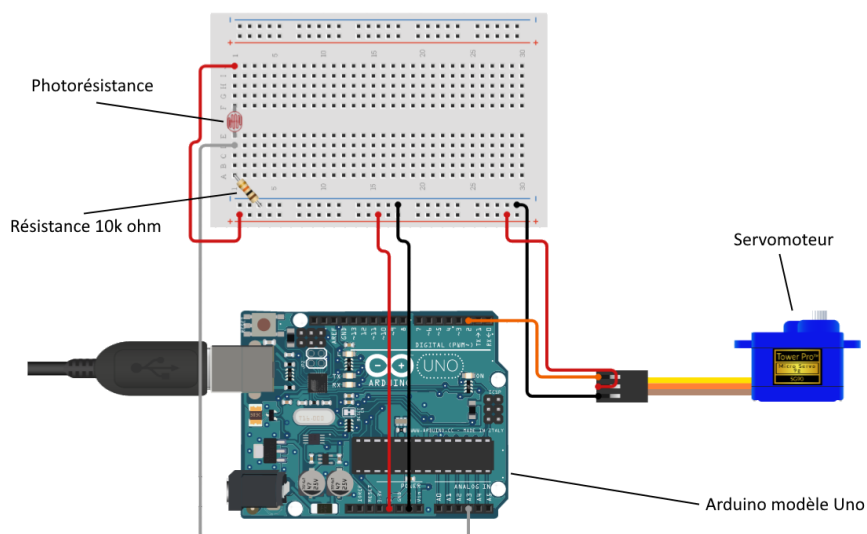


Figure 4. Architecture arduino

Architecture logicielle

Le système est divisé en plusieurs modules indépendants :

- **Module Dog Identifier** : serveur python contenant les chiens enregistrés et permettant l'identification d'un chien.
- **Module Pet feeder** : applications Node.js pour exposer l'objet connecté (**App Thing**) et exécuter la logique du système (**App Controller**).
- **Module User Interface** : regroupe les informations sur le système, les chiens enregistrés, l'ajout de chiens dans le système.

La figure 5 illustre les relations entre les différents modules.

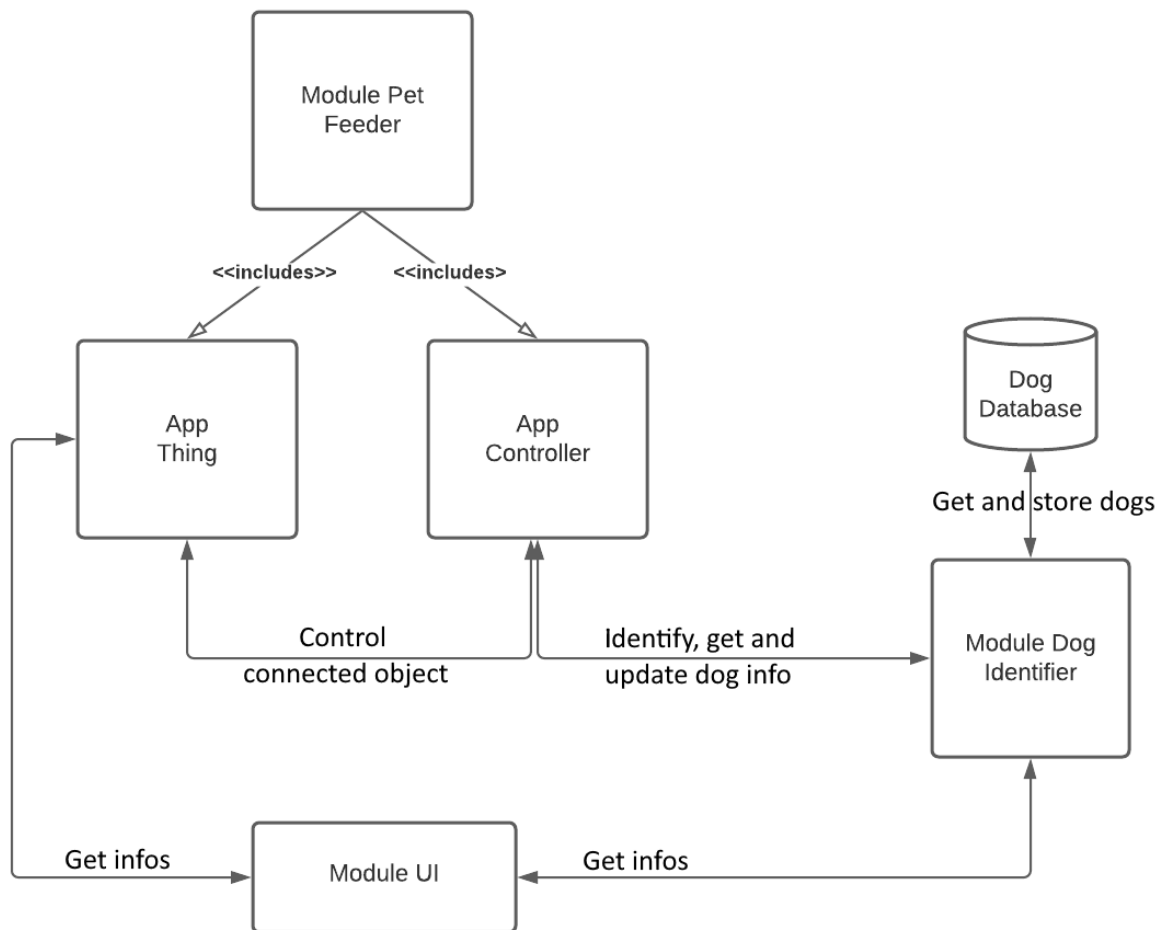


Figure 5. Architecture logicielle

Les figure 6 et 7 décrivent (globalement) les interactions entre les différents modules.

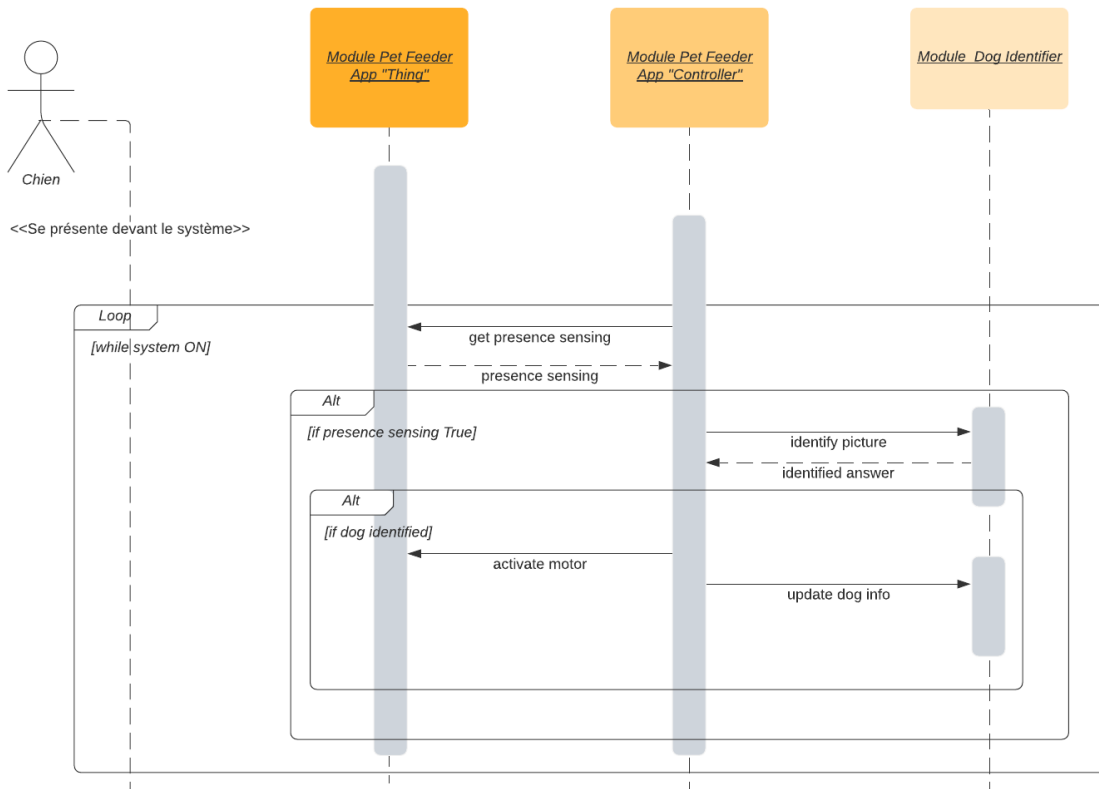


Figure 6. Interactions entre les modules - orienté chien

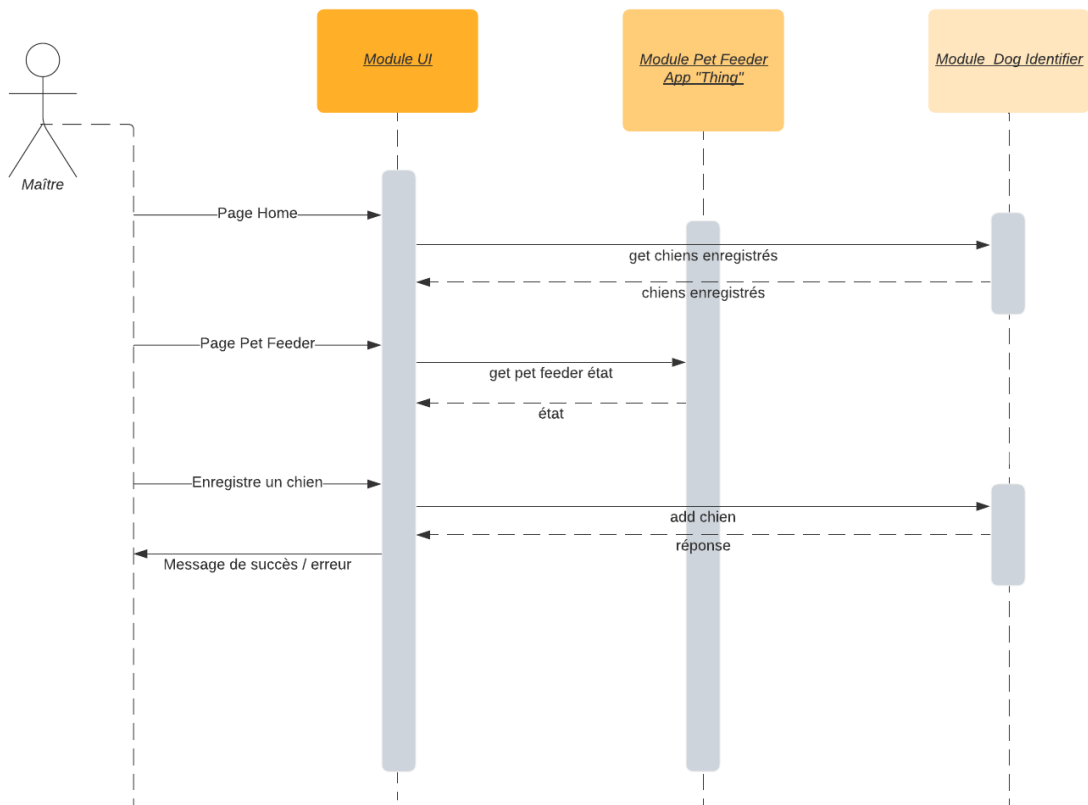


Figure 7. Interactions entre les modules - orienté utilisateur

APIs

Module Dog Identifier

/api/animals/identify

Méthode : POST

arguments : { img: <base64> }

Réponse : 200

/api/animals

Méthode : GET

arguments : ""

Réponse : 200

/api/animals/<id>

Méthode : GET

arguments : ""

Réponse : 200

/api/animals/eat/<id>

Méthode : PUT

arguments : { eaten: <int> }

Réponse : 204, 404

/api/animals/add

Méthode : POST

arguments : { name: <string>, img: <base64>, max_food: <int> }

Réponse : 201, 400

Module Pet Feeder - App “Thing”

L'objet connecté est exposé suivant la spécification WoT du W3C : voir [ici](#). Une requête GET à l'URL source de l'application retourne un objet JSON-LD détaillant toutes les interactions possibles avec l'objet connecté, notamment :

1. Propriétés

- food_ration - retourne la proportion de nourriture par activation
- nb_activation - retourne le nombre d'activation à la journée
- last_activation - retourne la date de la dernière activation

2. Actions :

- activate - active le moteur
- picture - prend une photo avec la caméra