



Projet de nichoir connecté

SmartCities 2025-2026



 **HEPL**
Haute Ecole de la Province de Liège

Analyse du marché

- Prix : 130 – 300 €
- Alimentation : Câblé, panneaux solaires
- Stockage : Carte SD, câblé, connexion WiFi cloud (privé), RSTP ...



Nichoир Connecté - Proposition de Valeur

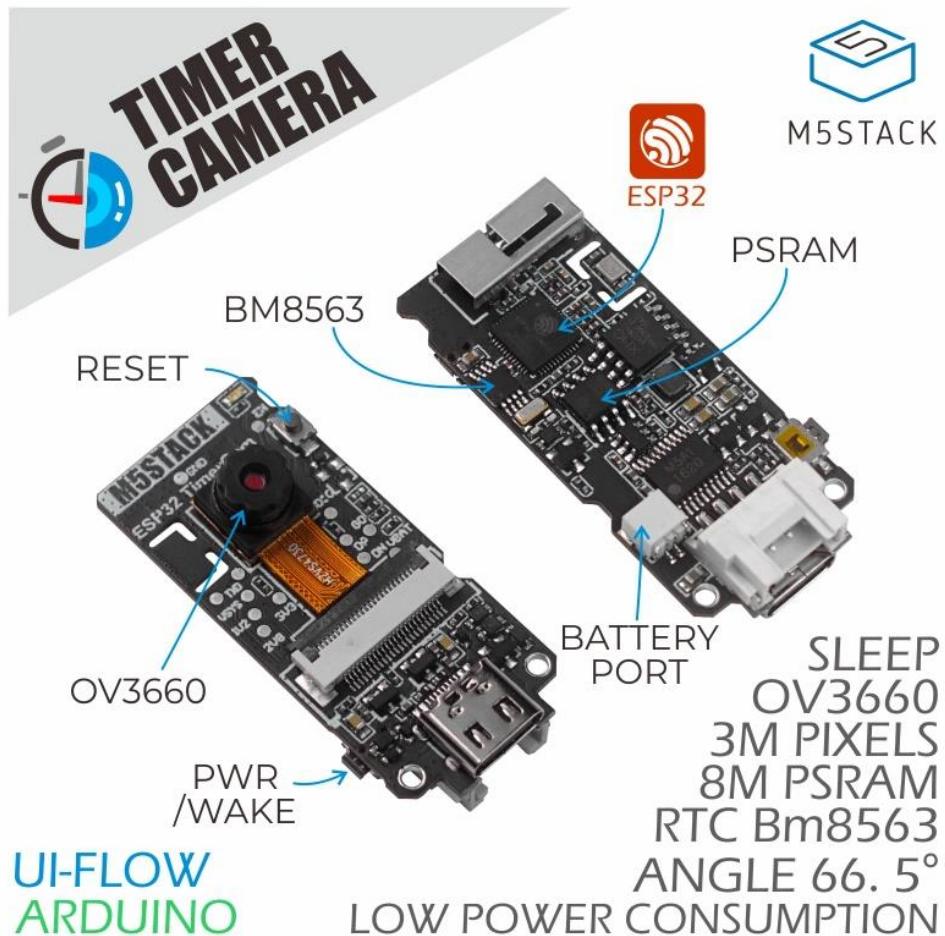
Objectif :

Créer un nichoir intelligent et connecté qui permet de suivre la vie des oiseaux avec un coût accessible et une autonomie énergétique prolongée.

Caractéristiques principales :

- **Prix :** Moins de 50 €
- **Autonomie énergétique :** 6 mois à 1 an sur batterie
- **Évolutif :** Possibilité d'ajouter un panneau solaire pour augmenter l'autonomie





⚙️ Composants et caractéristiques techniques

- Microcontrôleur : ESP32-D0WDQ6-V3
- Mémoire : 8MB PSRAM
- Caméra : OV3660, 3MP, DFOV 66.5°, résolution jusqu'à 2048x1536
- Indicateurs : LED de statut + bouton RESET

🔋 Gestion de l'énergie

- Conception ultra-basse consommation
- RTC intégré (BM8563) pour réveil et mise en veille programmés
- Courant en veille : aussi bas que 2µA
- Connecteur batterie réservé pour alimentation externe

👉 Connectivité et communication

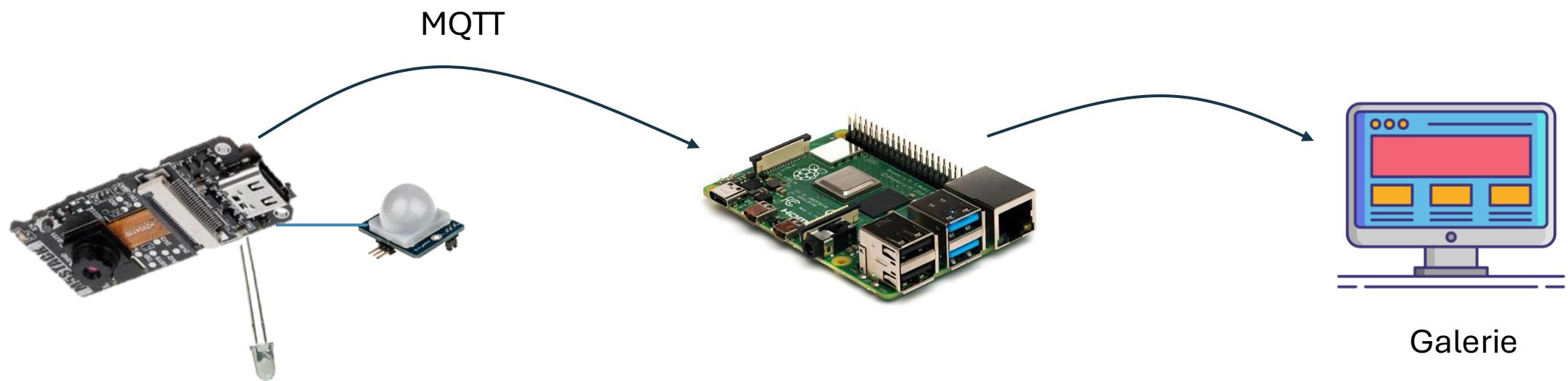
- Transmission d'image via Wi-Fi
- Port USB pour débogage
- Port HY2.0-4P pour connecter des périphériques externes

Librairies :

<https://github.com/m5stack/TimerCam-arduino/tree/master>

Objectifs

- **Surveiller l'activité dans un nichoir** via un système de détection de mouvement et de capture d'images.
- **Diminuer la consommation** pour une autonomie utile pour l'application.
- **Publier les données** avec une mise en page.



Fonctionnement du système

Configuration :

- Accès AP + interface Web

Détection de mouvement :

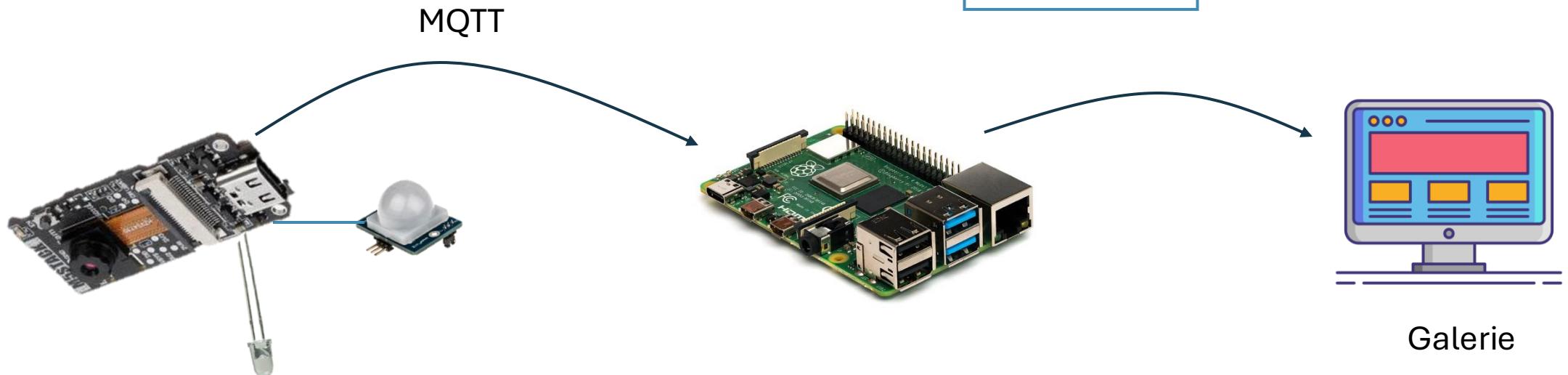
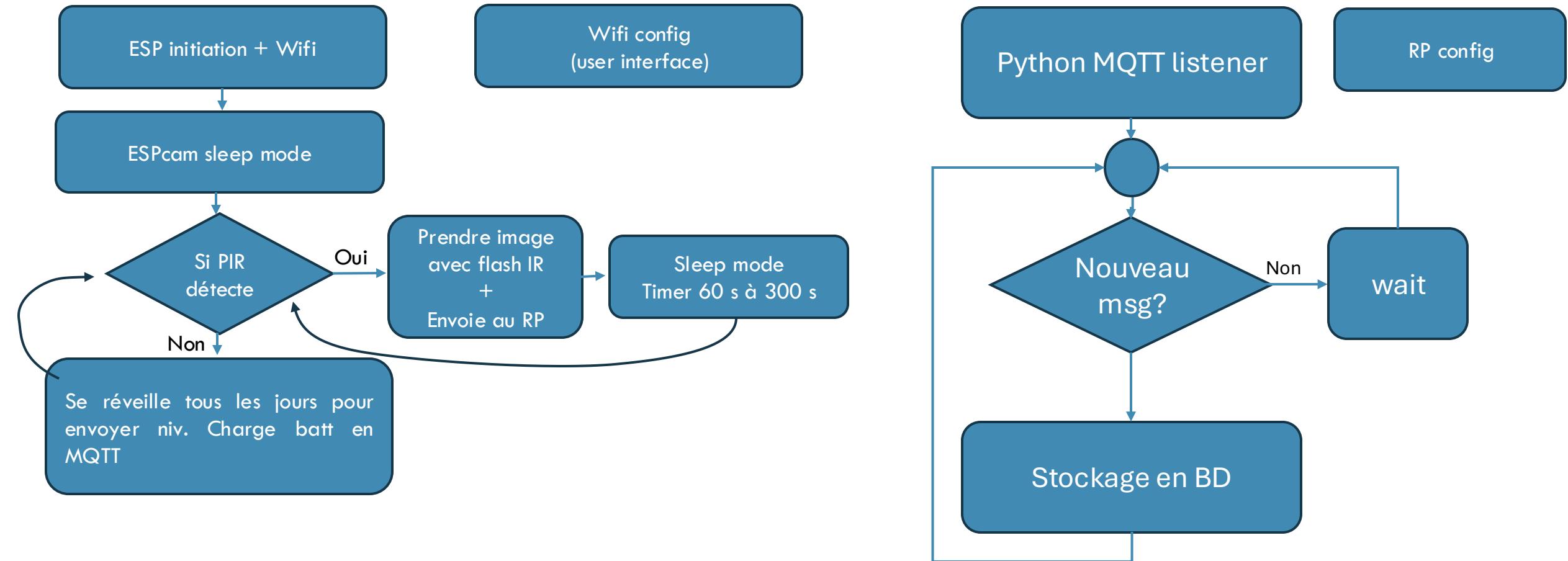
- Un capteur PIR détecte la présence d'oiseaux dans le nichoir.
- Lorsqu'un mouvement est détecté, le TimerCAM capture une photo avec un éclairage IR.

Transmission des données :

- La photo est envoyée au **Raspberry Pi** via Wi-Fi pour traitement et stockage en MQTT.
- Un script python sur le Raspberry Pi reçoit les images et données via MQTT et le stocke dans une base de données MySQL (MariaDB)
- Un second script Python sur le Raspberry Pi permettra de créer un serveur WEB (Flask) qui affichera une galerie d'images et les données dans un format adéquat

Niveau de la batterie :

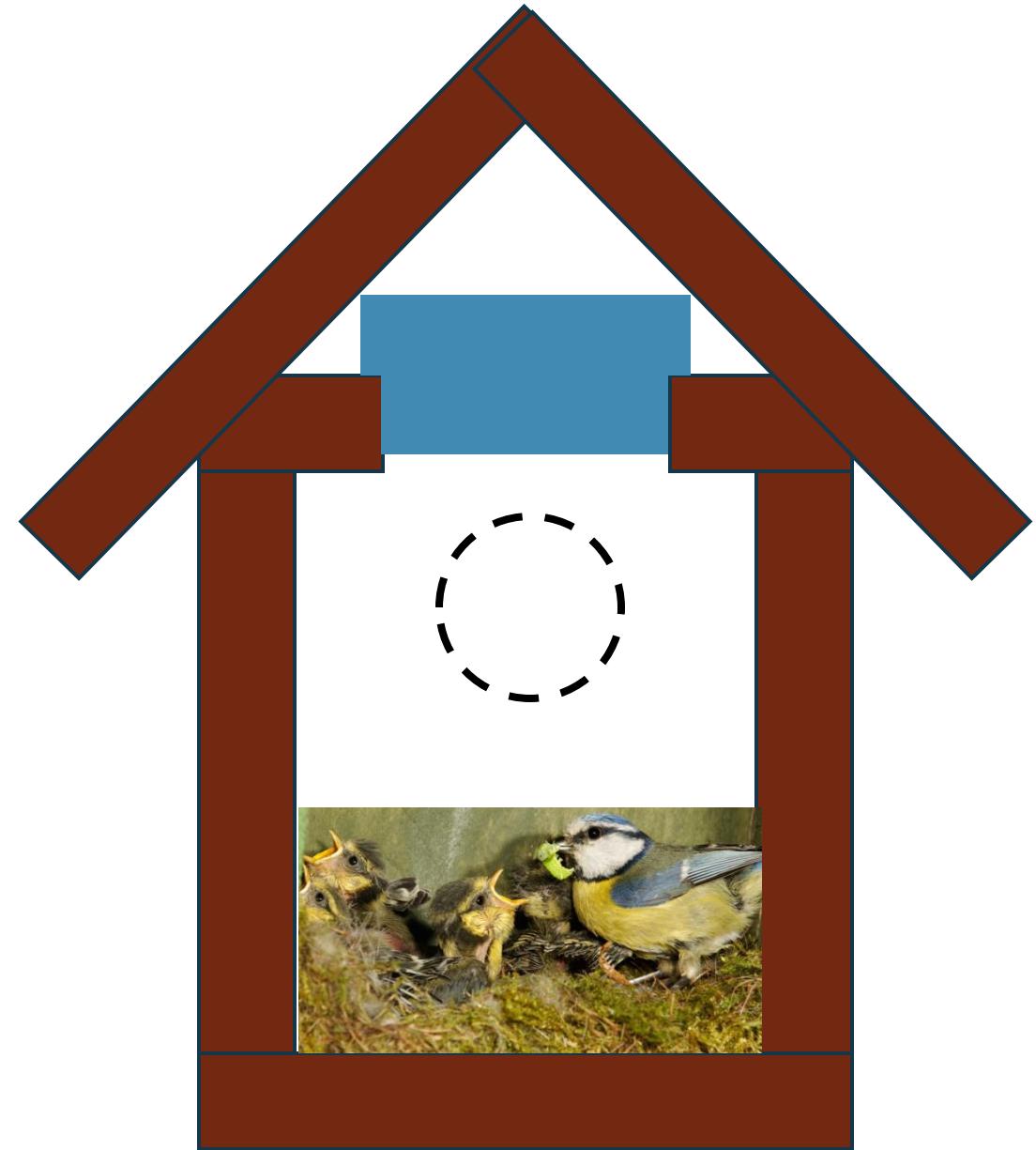
- **Mode standby** : En absence de mouvement dans le nichoir, le système se réveille une fois par jour et envoie le niveau de la batterie en MQTT.
- **Mode Présence** : Lorsque le capteur PIR détecte un mouvement, le niveau de la batterie est associé à chaque photo capturée.



Conception :

Conception et réalisation :

- Breakout pour le PIR, LED, Timercam
- Package pour le système complet
 - Breakout + batterie/support piles

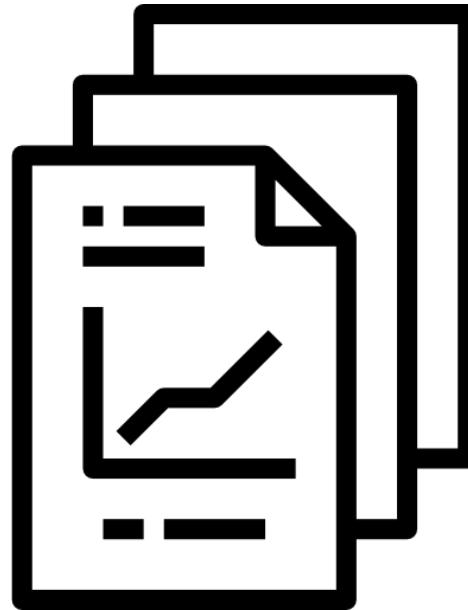


Dimension à fournir

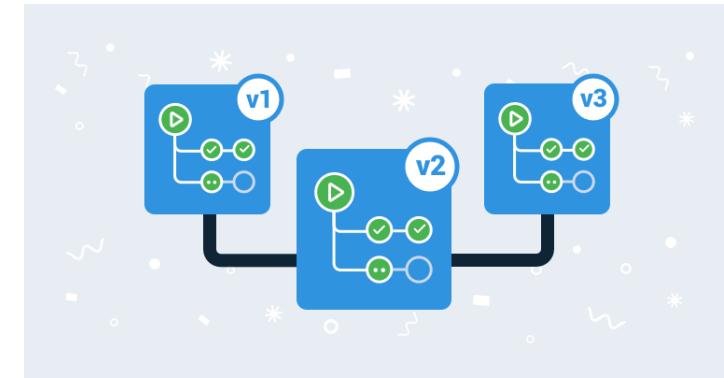
Evaluation



Prototype



Rapport



Versioning sur Github

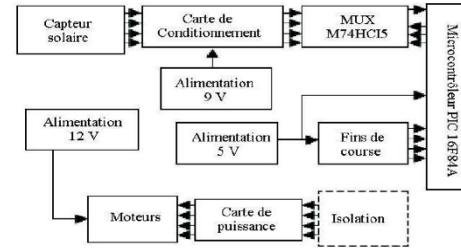
Evaluation

Compétence C1 Concevoir des systèmes complexes

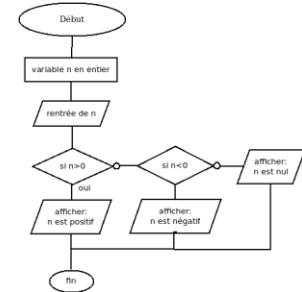
Composantes / Critères qualités

- L'étudiant est-il capable de d'établir une architecture/une structure /un schéma fonctionnel
- L'étudiant doit être capable de définir clairement les principales parties ou modules du système et leur organisation, incluant la manière dont ils interagissent
- L'étudiant est-il capable de simuler ou prototyper ces systèmes de façon adéquate
- L'étudiant doit savoir comment passer de la conception théorique à la réalisation pratique.
- L'étudiant est-il capable d'optimiser les solutions proposées au regard du cahier des charges
- En fonction des résultats des tests, l'étudiant doit être capable de faire des ajustements et des améliorations au prototype, et éventuellement recommencer le processus pour atteindre un résultat optimal.

Schéma bloc



Algo



Réalisation & optimisation



Evaluation

Compétence C2

Mettre en œuvre des systèmes complexes

Composantes / Critères qualités

- L'étudiant est-il capable de s'assurer de la fiabilité de la réalisation ?
 - L'étudiant doit interpréter les résultats des tests. Cela implique de pouvoir identifier les éventuelles erreurs, anomalies ou défaillances et de comprendre leur impact sur la fiabilité globale.
 - L'étudiant doit être capable de documenter le processus, les résultats des tests, et les corrections apportées, afin de garantir la traçabilité et la reproductibilité.
- L'étudiant est-il capable de respecter le cahier des charges ?
 - L'étudiant doit vérifier si le produit final ou la réalisation correspond aux exigences ou aux spécifications initiales.

Tester/caractériser + rendre fiable



Versioning sur Github



Respecter le cahier de charge



Evaluation

Compétence C3 **Développer sa professionnalité**

Composantes / Critères qualités

- L'étudiant est-il capable de s'inscrire dans une démarche de formation continue
- L'étudiant doit être capable de prendre l'initiative de s'informer et d'apprendre de manière indépendante, sans attendre que la formation vienne uniquement de sources externes

Auto-formation

