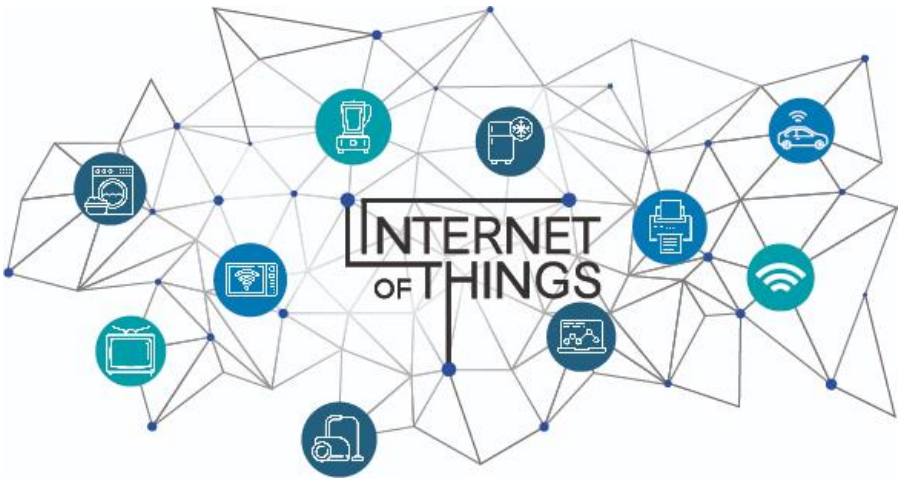
CUBES

Concevoir une application - BLOC INFCDL2

Découvrir l'IOT et son univers



Par Julien TOUTAIN
Réalisé pour le CESI à destination des étudiants "Développeurs Informatique"
v2020.07

CAMPUS
D'ENSEIGNEMENT SUPÉRIEUR
ET DE FORMATION PROFESSIONNELLE

Consignes Projet Expérientiel Collaboratif

Enoncé

Compétences

- Travail d'équipe
- Organisation
- Open-source (code, librairie, philosophie)
- Versioning
- SGBDR
- API
- Algorithme et logique
- Développement web
- OS Linux
- Hébergement
- Electronique

Présentation du projet

Une importante société d'électronique et d'électricité grand public, a depuis quelques années fait l'acquisition d'une startup vendant des stations météorologiques fonctionnant sur un modèle Cloud (données centralisées sur un serveur).
Depuis le rachat de la startup par la société d'électronique, les ventes ont chuté concernant la station météorologique connectée, en effet cette dernière reposant sur un serveur centralisé, les coûts d'hébergement de la solution ainsi que la baisse des ventes a amené l'entreprise à faire évoluer ce produit.

Contexte

La société "Lepetit" créé en 1992, qui a fait 1,275 milliard d'€ de CA en 2017 ayant racheté la startup "Atmos" qui commercialise des objets connectés pour la maison. "Lepetit" envisage la modernisation d'un produit "phare" de la startup "Atmos" qui depuis le rachat ne fait que perdre en popularité.
Vous venez d'être recruté dans cette importante compagnie (Lepetit) dans une nouvelle équipe (de 3 ou 4 personnes) dédiée à la création d'un prototype applicatif et électronique, vous vous répartissez les rôles et missions de ce projet.
Votre direction vous accorderait un créneau de 20 min afin de présenter votre prototype fonctionnel (application et hardware), vous veillerez à faire une présentation brève mais "efficace et commerciale" de votre prototype en mettant en avant les points forts et faibles de

ce projet, ainsi que les points de vigilance ou améliorations que vous proposez avant de passer à une phase d'industrialisation.

Problématique

Depuis le rachat de Atmos par Lepetit, les stations météorologiques connectées de la marque subissent une importante perte de popularité.
En effet ce produit repose sur un serveur "cloud" et le rachat par une grande entreprise fait craindre aux utilisateurs au sujet de la protection de leur données privées (possible revente aux fournisseurs d'énergie, chauffagiste, etc...)
La marque subit donc d'importantes pertes à héberger un service "cloud" de moins en moins utilisé (qui tombe régulièrement en panne et n'est plus forcément maintenu efficacement), et les ventes de ce produit sont en chute libre.
Ce produit était commercialisé 169.99€ TTC.

Il a été décidé par l'entreprise Lepetit de cesser la collecte de données centralisées, et de libérer le code source de ce projet afin de solliciter le public averti à l'amélioration continue de ce projet.

Organisation de la mission

Vous devrez travailler en équipe afin de réaliser un nouveau système de collecte de données météorologiques à partir d'éléments spécialement sélectionnés en fonction de leur coût d'achat (inférieur au précédent modèle).
La marque ayant décidé d'abandonner petit à petit le support de l'ancien modèle aux données centralisées, ce modèle devra pour un coût de production inférieur, faire office de serveur local.
Vous pourrez soumettre une idée pour l'équipe design du produit final mais votre équipe a pour rôle de concevoir une maquette fonctionnelle de l'application web de visualisation des données, ainsi qu'un prototype fonctionnel d'une sonde et du serveur de collecte des données météorologiques.

Liste de matériel

L'équipe R&D de Lepetit a sélectionné pour votre équipe une liste de matériel en fonction de leur coût et caractéristiques techniques pour élaborer la maquette qui sera réalisée par votre équipe.

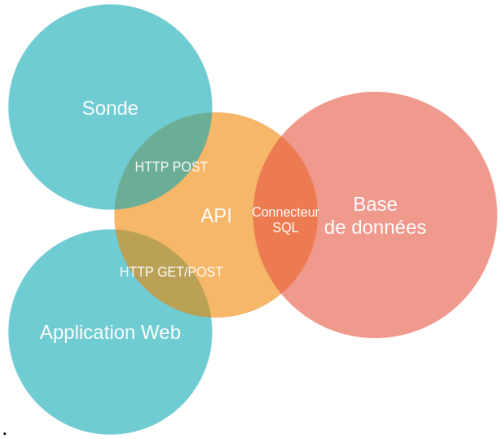
- **ESP8266 (ESP-01 ou ESP-01S)** : micro-contrôleur wifi avec 1Mo (ou 4Mo pour les versions les plus récentes) de mémoire flash
- **GY-21 HTU21** : capteur de température
- **Ecran type (** : écran LCD (pourra être raccordé au serveur ou à la sonde afin d'afficher le(s) dernier(s) relevé(s)
- **CP2102 ou CP2104 ou CH340** : convertisseur série USB pour programmer/flasher le micro-contrôleur ESP

- **Raspberry Pi Zero WH (ou supérieur), sa carte SD et son alimentation** : serveur équipé de Raspberry Pi OS Lite (sans GUI)
- **Breadboard, alimentation du breadboard et cable dupont** : matériel pour prototypage
- Salle équipée d'un réseau **wifi "neutre" type domestique** (idéalement sans portail captif) ou possibilité d'alternative via partage de connexion 4G/wifi

Commenté [1]: L'idée c'est qu'il puisse interconnecter les différentes machines / composants sur un même réseau

Les données collectées

Vous devrez collecter les données de type : température et humidité (capteur I2C GY-21)
Ces données devront être stockées dans une base de données côté serveur, cette base de données devra être capable d'accueillir les données de plusieurs sondes.
Afin de mener à bien cette collecte de données, la base de données relationnelle sélectionnée sera de type SQL (MySQL/maria dB ou PostgreSQL ou sqlite), une API Rest fera l'interface entre applicatif et base de données ainsi qu'entre sondes et base de données



.

Cette base de données et API permettront l'ajout/suppression et activation/désactivation de sonde (CRUD)

Consignes du CUBES

La direction lors de réunions préalables vous a communiqué et mis en évidence quelques fonctionnalités requises ainsi que des objectifs et contraintes pour votre équipe et projet. Voici les notes synthétisées par les membres de votre équipe.

Le projet

Votre projet “open-source” :

- Devra être accessible sur git (github ou gitlab)
- Vous devrez soumettre ce git à votre direction avant votre présentation afin qu’il puisse auditer le travail de votre équipe

L’application

Votre application devra :

- Disposer d’une interface web consultable à travers un simple navigateur
- Être “responsive”
- Inclure un graphique des données température/humidité
- Inclure un pictogramme ou code couleur pour déduire à partir des données la météo actuelle
- BONUS : carte pour situer les sondes (sur un JPEG type plan de maison ou de terrain, ou sur une carte openstreet maps via leaflet par exemple)
- BONUS : un bouton de “partage” du graphique ou du dernier relevé vous permettra d’envoyer facilement les données à vos amis/familles (mail, réseaux sociaux, autres ?)

L’API et la base de données

- Méthode pour **C**(reate)**R**(ead)**U**(pdate) API sur la table sonde
- Méthode pour **C**(reate)**R**(ead) API sur la table des relevés
- Tourne sur la même machine locale (en attendant la mise en place du serveur Raspberry ou dans le cas de télétravail il est possible d’héberger sur une VM ou docker les services requis)
- BONUS : gérer l’authentification des utilisateurs

Serveur

- Il héberge le serveur web
- Il héberge le SGBDR correctement configuré
- Il sera administrable via SSH
- Il héberge l’API
- La base de données ne sera accessible qu’à travers l’API
- Il disposera d’un écran afin d’afficher : son adresse IP, la date et heure, et fera défiler les derniers relevés des sondes

Sonde de température

- Correctement câblée avec les capteurs de température/humidité et alimentation (vous ferez valider par un ingénieur de la R&D (votre pilote) le câblage avant mise sous tension)
- Elle sera programmée avec le langage de votre choix compatible avec un ESP (C, micropython, LUA)
- Afin d'éviter les relevés imprécis le capteur fera des relevés toutes les quelques secondes et vous en réaliserez une moyenne d'au moins 5 relevés afin d'envoyer des données "lissées" à l'API (et avoir un graphique cohérent)

Cahier des charges

Le code de ce projet sera hébergé sur un dépôt git (github, gitlab)
Votre projet s'articulera autour d'une API Rest permettant la création, lecture, mise à jour, suppression des données dans la base de données.
Il sera muni d'une interface web (idéalement responsive) afin d'afficher le suivi de la température et de l'humidité sur un graphique.
Votre direction vous a demandé de livrer les éléments de votre projet réparti de la manière suivante sur 3 jours.