

## 2. Les exigences fonctionnelles pour le projet proposé

---

Pour enregistreur de température a long terme, on propose de concevoir un système embarqué avec les fonctionnalités suivantes:

- Mesurer la temperature a toutes les trente minutes;
- Publier les temperatures mesurees dans une base de données, pour permettre l'analyse des donnees passes;
- Afficher toutes les valeurs mesurees, actuelles et precedents, sur un ecran, en utilisant du texte et des graphiques;
- Alerte sonore lorsque la temepérature dépassent des valeurs configurables par l'utilisateur

## 3. Les exigences techniques

---

### 3.1 Analyse des autres projets d'enregistrement de température sur le long terme

Pour le moment, cette section est vide car je n'ai pas réussi à trouver d'autres projets similaires. Sera bientôt terminé

### 3.2 Analyse des moniteurs de temperature a long terme disponibles dans le commerce

Meme comme le point 3.1.

### 3.3 Analyse des services de base de données appropriés pour ce projet

Compte tenu de l'utilisation des bases de données dans la vie moderne, lorsque toutes les informations se trouvent à un clic de distance de l'utilisateur et de la croissance des services REST, les options de base de données que j'ai trouvées sont les suivantes:

- FirebaseDB
- RestDB

FirebaseDB est une application qui stockez et synchronisez des données avec notre base de données cloud NoSQL. Les données sont synchronisées en temps réel sur tous les clients et restent disponibles lorsque votre application se déconnecte. La

base de données en temps réel Firebase est une base de données hébergée dans le cloud. Les données sont stockées sous forme de JSON et synchronisées en temps réel sur chaque client connecté. Lorsque vous créez des applications multiplates-formes avec nos kits de développement logiciel (SDK) iOS, Android et JavaScript, tous vos clients partagent une instance de base de données en temps réel et reçoivent automatiquement les mises à jour avec les données les plus récentes. Pour l'utilisation de l'API, vous pouvez utiliser n'importe quelle URL de base de données Firebase en tant que point de terminaison REST. Il suffit d'ajouter .json à la fin de l'URL et d'envoyer une requête à partir de votre client HTTPS préféré.

RestDB est un service cloud de base de données NoSQL. Les données sont accessibles en tant qu'objets JSON via HTTPS. Cela donne une grande flexibilité, une intégration système facile et une compatibilité future. Un schéma avec Collections (tables), champs et relations est créé directement dans votre navigateur Web. restdb.io crée dynamiquement une API REST et une API Javascript pour refléter votre schéma. L'accès aux données est sécurisé avec des clés API ou par authentification d'utilisateur externe à l'aide du service Auth0. Vous pouvez facilement interroger la base de données avec un paramètre d'URL contenant une requête de type MongoDB. Une API temps réel pour l'écoute d'événements REST (POST, PUT, DELETE) ou d'événements personnalisés ouvre de nombreuses possibilités. Pour créer des intégrations flexibles, chaque collection peut également disposer de Webhooks ou de codeshooks (Javascript côté serveur).

### **3.4 Les spécifications détaillées du système proposé**

Sur la base de l'analyse ci-dessus, et considérant que ce projet a avant tout une valeur éducative, nous pouvons rédiger les spécifications détaillées de notre système comme suit:

- L'utilisation d'un capteur de la température d'air.
- Nous aimerions afficher les écrans d'information suivants, commutable par l'utilisateur:
  - Vue de température actuelle dans un écran
  - Données historiques pour température (par jour / mois)
- Il n'y a pas des contraintes de temps réel; un point de données toutes les 30 minutes, par exemple, devrait être suffisant, et cela ne va pas atteindre les limites de niveau gratuit du système de gestion des données.
- Nous avons besoin d'une plate-forme de développement intégrée avec une interface Internet, et le Wi-Fi est probablement le choix le plus logique, comme il est largement utilisé à l'intérieur. Si nous devions concevoir un

moniteur d'extérieur, nous ne pouvons plus compter sur la disponibilité du WiFi; dans ce cas, une interface GSM ou LORA aurait été un meilleur choix.

- Il n'y a pas des exigences concernant la puissance de calcul; une carte Arduino avec microcontrôleur AT-MEGA328p ou ATMEGA32 ou similaire devrait être suffisant, tant qu'il dispose d'un module Wi-Fi (externe ou intégré).
- Comme éléments d'interface utilisateur, nous avons besoin d'un écran (il peut être un écran LCD), pour afficher la température actuelle.
- Nous n'avons pas besoin que l'appareil résiste aux conditions extrêmes, mais un boîtier simple est nécessaire.
- Comme source d'alimentation, nous allons utiliser 2 piles AA rechargeables avec un stabilisateur 5V.

### 3.5 Le schéma de principe pour le système proposé

Le schéma de principe n'est pas un schéma électrique; c'est une représentation simplifiée destinée à clarifier les concepts généraux, sans détails de bas niveau. Pour ce projet, on doit connecter les composants suivants:

- la carte de développement Arduino
- le module Wi-Fi (sauf si nous choisissons une carte avec Wi-Fi intégré)
- le capteur de température
- l'écran

L'interface entre notre système et le système de gestion des données qui va être REST pour FirebaseDB or RestDB.



