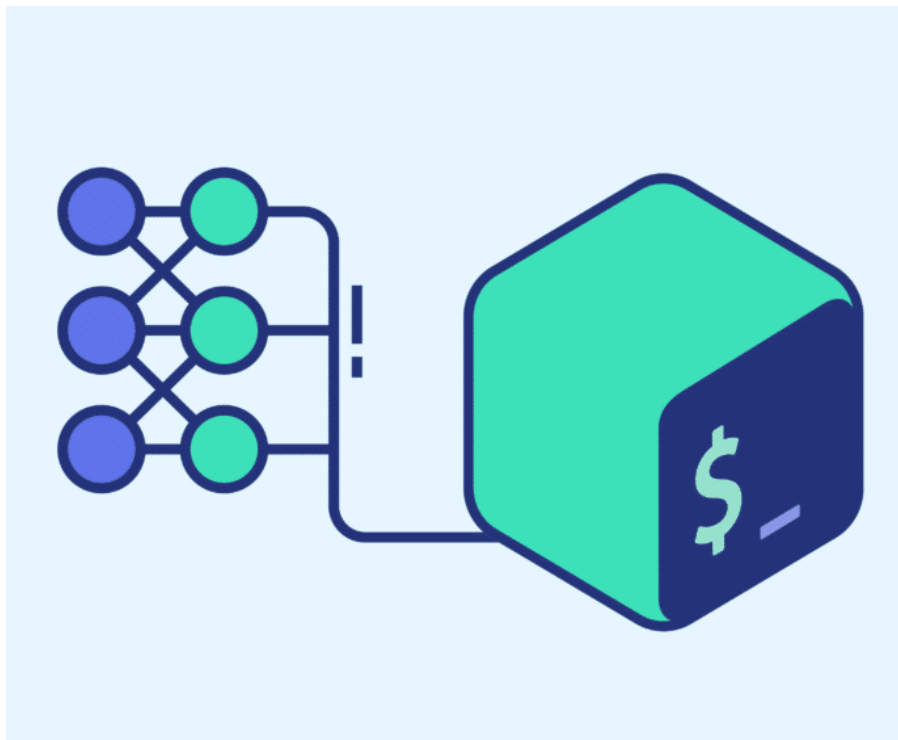


# Rapport Technique

SAE 15



**Nom : BOULOUIHA GNAOUI Yassir**

*Réseaux & Télécoms 1<sup>ère</sup> Année*

# Table des matières :

<b>Contextualisation du projet.....</b>	<b>1</b>
<b>Tutoriel d'installation et d'utilisation.....</b>	<b>2</b>
<b>Diagramme de GANTT.....</b>	<b>5</b>
<b>Logigrammes des codes :.....</b>	<b>6</b>
a ) Récupération et sauvegarde des valeurs.....	6
b ) Calcul des métriques.....	8
c ) Génération du fichier html et publication.....	9
<b>Problèmes rencontrés et solutions apportées.....</b>	<b>9</b>
<b>ANNEXES.....</b>	<b>13</b>

## Contextualisation du projet

Ce projet vise à aider le service technique dans la gestion énergétique d'un bâtiment de recherche et développement en exploitant les données provenant de capteurs répartis dans le bâtiment. Les données sont accessibles via le protocole MQTT au format JSON. Les données seront traitées pour calculer une moyenne, déterminer le minimum et le maximum.

Une interface de suivi sera proposée pour afficher les données brutes horodatées et les métriques calculées. Les tâches principales incluent la récupération de données JSON, l'extraction de valeurs mesurées et de noms de salles, le traitement des données pour déterminer des métriques, la construction d'une représentation textuelle des données en utilisant HTML, la publication sur un serveur web et l'automatisation de la chaîne de traitement. Les contraintes techniques incluent l'utilisation d'une machine virtuelle, un système d'exploitation GNU/Linux (Lubuntu 20.04), l'utilisation d'éditeurs de texte, des langages BASH, HTML5 et CSS3.

## Tutoriel d'installation et d'utilisation


Afin d'installer ma solution il faut dans un premier temps dézipper le dossier SAE\_15 dans la racine de votre système d'exploitation linux. Il est important de bien le mettre à la racine afin de ne pas rencontrer de problèmes avec le script à la suite. Une fois avoir installé le script a la racine vous devez uploader le site web sur votre serveur ftp afin d'avoir la page mentions légales, accueil, et le style de fonctionnels. les codes du site web sont dans :

**/SAE\_15/config/installation web**

Une fois cela fait ouvrez le fichier de configuration config.sh dans :

**/SAE\_15/config/config.sh**

Dans le fichier config.sh, rentrez vos identifiants FTP, MQTT, et choisissez le nombre de minutes pour lesquelles le script doit se réexécuter à chaque fois.

```
SAE_15 > config >  config.sh
1  #!/bin/bash
2
3  declare -A ftp_config
4  declare -A mqtt_config
5  declare -A cron_config
6
7  # FTP Configuration
8  ftp_config['host']='ftp://ybgm.fr/public_html/SAE15/' # Enter the host url of your FTP server
9  ftp_config['user']='admin@ybgm.fr' # Enter your username for the FTP server
10 ftp_config['pswd']='y8^%fCy;p-pC' # Enter your password for the FTP server
11
12 # MQTT Server Configuration
13 mqtt_config['host']='iot.iut-blagnac.fr' # Enter the host of the MQTT Server
14 mqtt_config['user']='student' # Enter your username to connect to the MQTT Server
15 mqtt_config['pswd']='student' # Enter your password to connect to the MQTT Server
16
17 # Crontab Config
18 cron_config['time']='3' # Enter the number of minutes for which the script should run again each time
19
20 # AFTER THIS EXECUTE THE SCRIPT INSTALL.SH
21
```

Une fois avoir bien rempli le fichier config.sh, sauvegardez-le. Après cela, vous n'avez plus qu'à exécuter le fichier install.sh dans :

**SAE\_15/config/install.sh**

Le fichier install.sh va permettre d'installer tous les outils nécessaires au bon fonctionnement du script ainsi que la crontab.

Note : Si vous changez un paramètre concernant la crontab il faut réexécuter le fichier install.sh afin que cela soit pris en compte.

```

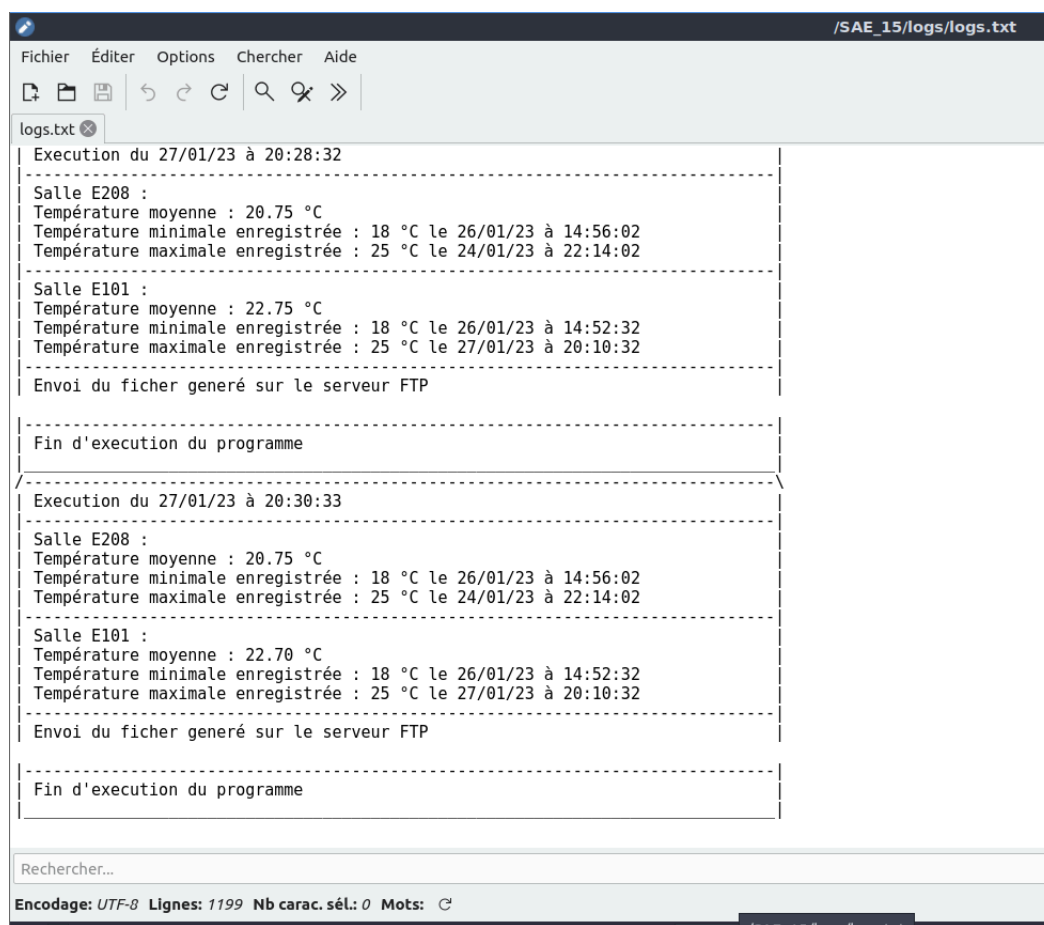
etud@etudSAE15:/SAE_15/config$ ./install.sh
-----
|      INSTALLATION  MOSQUITTO      |
-----
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdlt2 libev4 libjq1 libonig5 libwebsockets15
Veuillez utiliser « sudo apt autoremove » pour les supprimer.
Les NOUVEAUX paquets suivants seront installés :
  mosquito-clients
0 mis à jour, 1 nouvellement installés, 0 à enlever et 110 non mis à jour.
Il est nécessaire de prendre 58,8 ko dans les archives.
Après cette opération, 202 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://archive.ubuntu.com/ubuntu focal/universe amd64 mosquito-clients amd64 1.0.9-1 [58,8 kB]
58,8 ko réceptionnés en 0s (105 ko/s)
Sélection du paquet mosquito-clients précédemment désélectionné.
(Lecture de la base de données... 209927 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../mosquito-clients_1.0.9-1_amd64.deb ...
Dépaquetage de mosquito-clients (1.0.9-1) ...
Paramétrage de mosquito-clients (1.0.9-1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.1-1) ...
-----
|      INSTALLATION  JQ      |
-----
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdlt2 libev4 libwebsockets15
Veuillez utiliser « sudo apt autoremove » pour les supprimer.
Les NOUVEAUX paquets suivants seront installés :
  jq
0 mis à jour, 1 nouvellement installés, 0 à enlever et 110 non mis à jour.
Il est nécessaire de prendre 0 o/50,2 ko dans les archives.
Après cette opération, 99,3 ko d'espace disque supplémentaires seront utilisés.
Sélection du paquet jq précédemment désélectionné.
(Lecture de la base de données... 209937 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../jq_1.0-1ubuntu0.20.04.1_amd64.deb ...
Dépaquetage de jq (1.0-1ubuntu0.20.04.1) ...
Paramétrage de jq (1.0-1ubuntu0.20.04.1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.1-1) ...
-----
|      INSTALLATION  CURL      |
-----
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdlt2 libev4 libwebsockets15
Veuillez utiliser « sudo apt autoremove » pour les supprimer.
Les NOUVEAUX paquets suivants seront installés :
  curl
0 mis à jour, 1 nouvellement installés, 0 à enlever et 110 non mis à jour.
Il est nécessaire de prendre 0 o/101 ko dans les archives.
Après cette opération, 413 ko d'espace disque supplémentaires seront utilisés.
Sélection du paquet curl précédemment désélectionné.
(Lecture de la base de données... 209944 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../curl_7.08.0-1ubuntu2.15_amd64.deb ...
Dépaquetage de curl (7.08.0-1ubuntu2.15) ...
Paramétrage de curl (7.08.0-1ubuntu2.15) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.1-1) ...
-----
|      INSTALLATION  CRONTAB      |
-----
Suppression ancienne crontab ...
Génération nouvelle crontab ...
Nouvelle crontab :
*/3 * * * * /SAE_15/start.sh >> /SAE_15/logs/logs.txt
-----
|      INSTALLATION  OK      |
-----
etud@etudSAE15:/SAE_15/config$

```

Si jamais vous rencontrez un problème avec le script, n'hésitez pas à aller regarder le fichier logs.txt qui se situe dans le répertoire :

**/SAE\_15/logs/logs.txt**

Toutes les exécutions réalisées par la crontab horodatées et contenant les métriques sont répertoriées dans ce fichier.



```

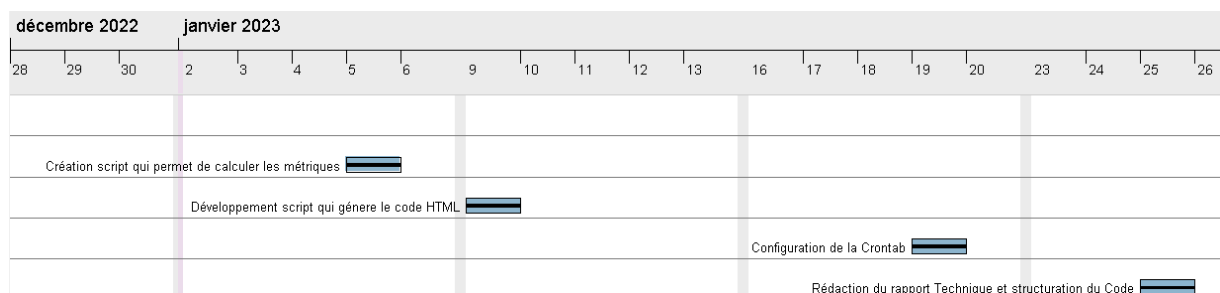
/SAE_15/logs/logs.txt
Fichier  Éditer  Options  Chercher  Aide
logs.txt
-----
Execution du 27/01/23 à 20:28:32
Salle E208 :
Température moyenne : 20.75 °C
Température minimale enregistrée : 18 °C le 26/01/23 à 14:56:02
Température maximale enregistrée : 25 °C le 24/01/23 à 22:14:02
Salle E101 :
Température moyenne : 22.75 °C
Température minimale enregistrée : 18 °C le 26/01/23 à 14:52:32
Température maximale enregistrée : 25 °C le 27/01/23 à 20:10:32
Envoi du fichier generé sur le serveur FTP
Fin d'execution du programme
-----
Execution du 27/01/23 à 20:30:33
Salle E208 :
Température moyenne : 20.75 °C
Température minimale enregistrée : 18 °C le 26/01/23 à 14:56:02
Température maximale enregistrée : 25 °C le 24/01/23 à 22:14:02
Salle E101 :
Température moyenne : 22.70 °C
Température minimale enregistrée : 18 °C le 26/01/23 à 14:52:32
Température maximale enregistrée : 25 °C le 27/01/23 à 20:10:32
Envoi du fichier generé sur le serveur FTP
Fin d'execution du programme
-----
Rechercher...
Encodage: UTF-8  Lignes: 1199  Nb carac. sél.: 0  Mots: 4
  
```

Cela permet en cas de problèmes avec une exécution du script faite par la crontab d'avoir des informations détaillées afin de cerner au plus vite et au mieux d'où vient le problème.

## Diagramme de GANTT

		
Nom	Date de début	Date de fin
Création du Site WEB	14/12/2022	14/12/2022
Création script qui permet de cal...	05/01/2023	05/01/2023
Développement script qui génère...	09/01/2023	09/01/2023
Configuration de la Crontab	19/01/2023	19/01/2023
Rédaction du rapport Technique ...	25/01/2023	25/01/2023

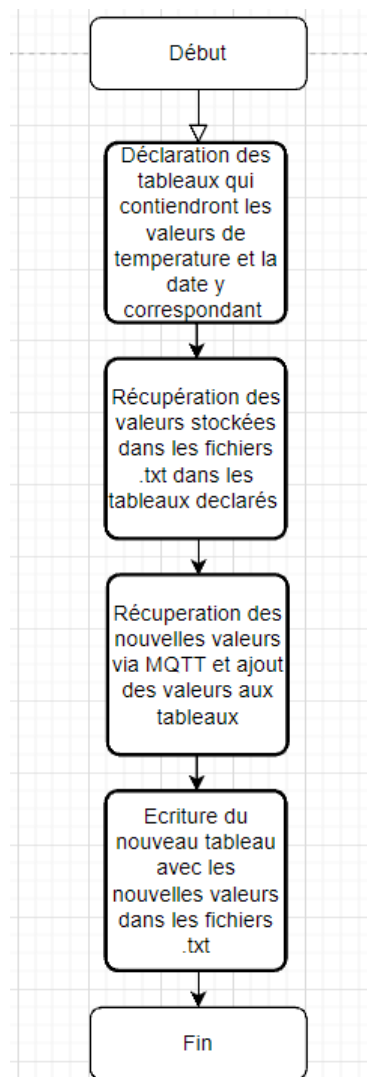
Le diagramme prévisionnel que j'ai fait suit les séances de TP en autonomie qui nous ont été allouées. Au début je pensais que ces séances ne me suffiraient pas et que je devrais surement faire beaucoup plus de travail chez moi, mais en réalité les 4 séances de 4h m'ont largement suffi à faire le projet, j'ai même fini la partie technique (développement du code) au bout de la 3eme de ces 4 séances, ainsi cela me permet de faire des ajustements et de préparer mon oral sur la dernière séance prévue.



## Logigrammes des codes :

### a ) Récupération et sauvegarde des valeurs

Le premier logigramme est celui de la première partie du code (main.sh) qui a pour fonction de récupérer et de réécrire les données mises à jour dans un fichier .txt

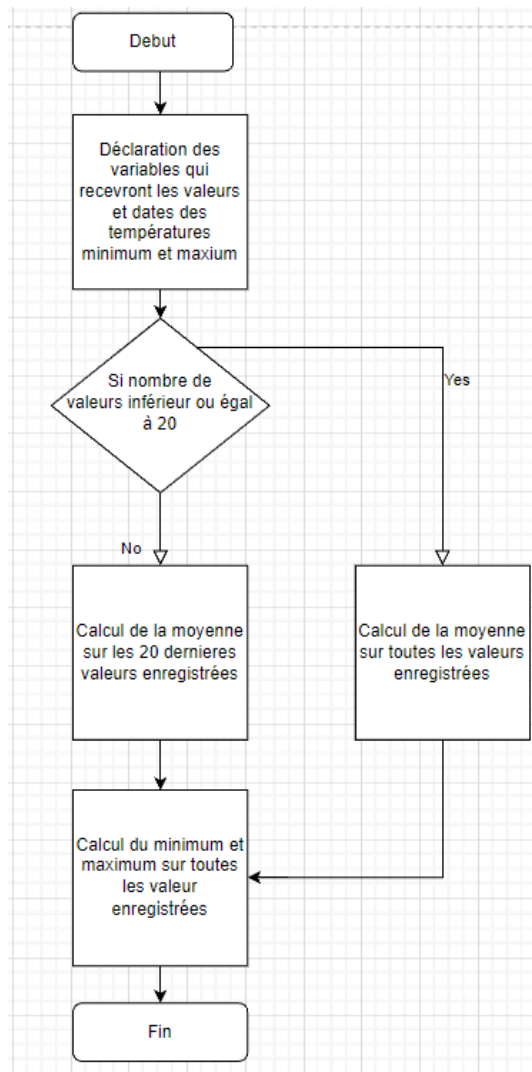




Ce logigramme récupère donc les valeurs enregistrées dans les fichiers moy\_n°salle.txt et moy\_n°salle\_date.txt dans le répertoire /script/data/n°salle . Une fois cela fait, il récupère les nouvelles valeurs via MQTT puis les ajoute dans le tableau où ont été récupérées les valeurs des fichiers .txt. Pour finir ce script enregistre la table avec la nouvelle valeur récupérée dans le fichier .txt correspondant.

### b ) Calcul des métriques

Le second logigramme représente la partie du code (main.sh) qui calcule les métriques.

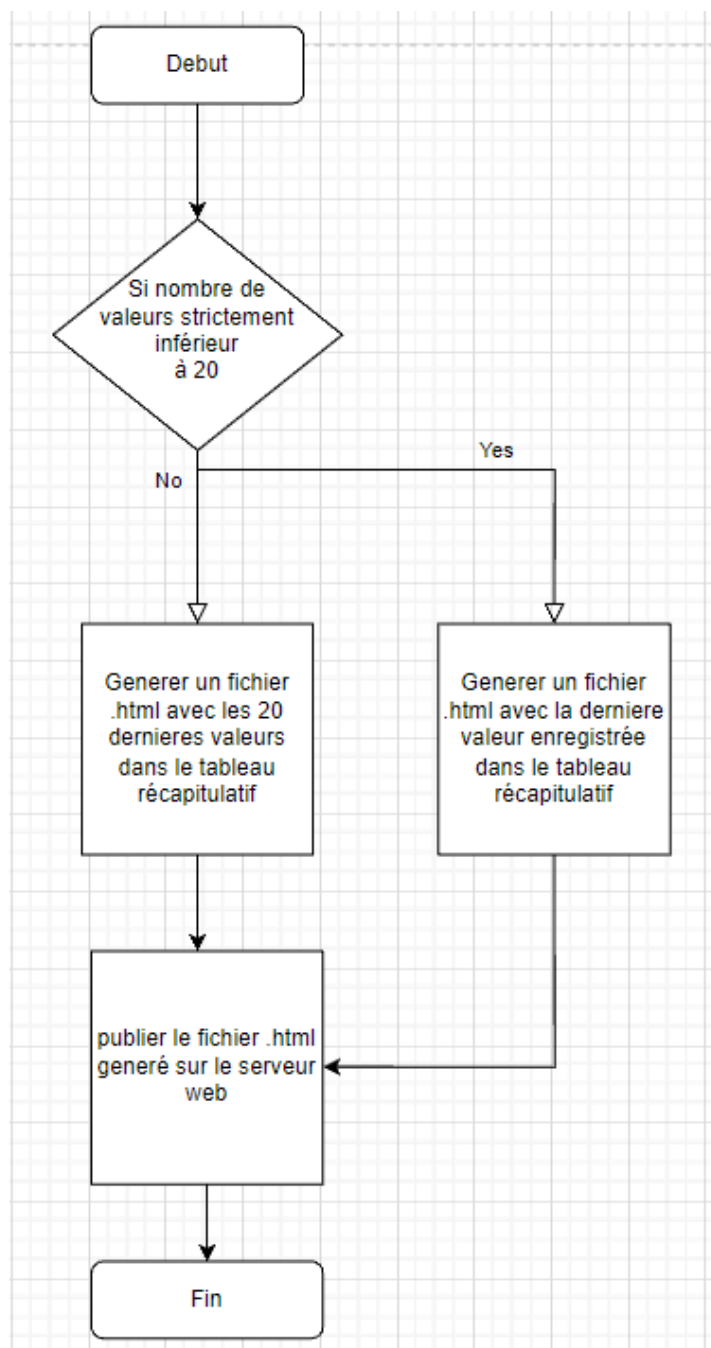


Cette deuxième partie du code consiste donc à calculer à l'aide du tableau des valeurs de température et de date la moyenne ainsi que le minimum et maximum horodatés. Pour le calcul de la moyenne j'ai fait le choix de la calculer sur uniquement les 20 dernières valeurs max, parce que je trouve cela plus pertinent, ça donne une moyenne qui représente réellement la moyenne de température des 40 dernières minutes. Le souci c'est que si on fait la moyenne de toutes les valeurs enregistrées (ce que je faisais au début), la moyenne va être calculée avec des valeurs de température qui ont été enregistrées parfois des jours auparavant, et selon moi cela fausse la moyenne de température. En ce qui concerne le minimum et maximum j'ai trouvé intéressant de les calculer sur toutes les valeurs enregistrées, parce que cela peut être utile de savoir si la température max n'a pas été atteinte depuis un certain temps (par exemple si le pic de 25 °C n'a pas été atteint depuis 3 jours), bien évidemment si la température max précédemment enregistrée est atteinte à une date plus proche c'est tout le temps la date la plus récente qui est affichée pour les minimum et maximum.

### **c ) Génération du fichier html et publication**

La dernière partie de mon programme consiste à vérifier si le nombre de valeurs de température enregistrées est supérieur ou inférieur à 20 à l'aide d'une condition. Ainsi dans le cas où le nombre de valeurs enregistrées est inférieur à 20 je génère un fichier .html qui affiche la moyenne, le min et le max des valeurs en question, et j'affiche dans le tableau récapitulatif les dernières valeurs des salles uniquement une valeur soit la dernière. Dans le cas où le nombre de valeurs enregistrées est supérieur ou égal à 20, le script génère un fichier html qui affichera la moyenne des 20 dernières valeurs, le min et le max de l'ensemble des valeurs enregistrées jusqu'à présent, et les 20 dernières valeurs enregistrées par salle.

Logigramme de la dernière partie du code (send\_html.sh) :



## Problèmes rencontrés et solutions apportées

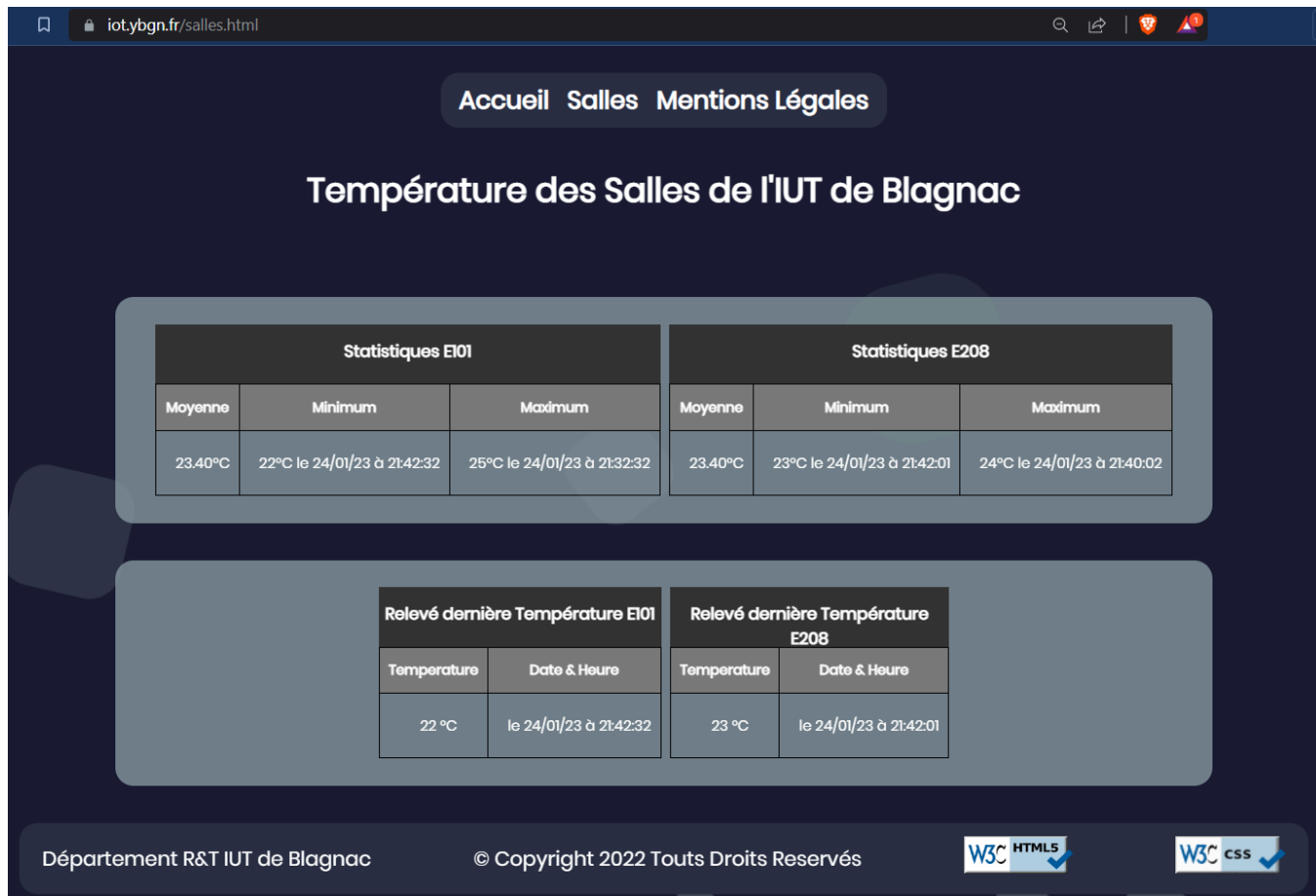
L'un des premier problèmes auxquels j'ai dû faire face était le fait d'horodater les valeurs de température. Au début j'avais essayé de regrouper la valeur de température et la date dans une seule et même variable de type chaîne de caractères, puis de sauvegarder cette variable dans un fichier .txt pour chaque nouvelle variable, cependant le problème c'est que on peut pas soustraire deux chaînes de caractères si facilement en bash, il fallait donc que j'extraie la valeur de la température depuis la chaîne de caractères qui contenait également la date à laquelle elle a été extraite, alors en pratique cela n'est pas impossible, je l'ai d'ailleurs fait en utilisant des expressions régulières en bash, cependant au niveau de la lisibilité du code c'était vraiment pas terrible et on ne pouvait pas extraire directement la valeur numérique et l'utiliser dans une condition sans l'assigner à une variable prévue à cet effet plus tôt dans le code. J'ai donc assez vite mis de côté cette idée et j'ai pensé à sauvegarder ces deux données séparément, c'est à dire de sauvegarder la température brute dans un fichier, et la date associée à cette température dans un autre fichier, ainsi vu que les deux fichiers recevront des nouvelles valeurs à la même fréquence les indices seront les mêmes pour les deux tableaux si on venait à lire les fichiers .txt dans deux tableaux différents donc il me suffisait de connaître l'indice de la valeur brute dans son tableau afin d'avoir la date associée à cette valeur.

Suite à cela j'ai rencontré un problème qui lui a pris un peu plus de temps à résoudre malgré le fait qu'il paraisse bien plus simple que les autres aujourd'hui. Lorsque j'écrivais les valeurs de mon tableau contenant les dates associées aux températures dans un fichier .txt, je devais délimiter les dates avec un caractère qui puisse différencier les dates pour pouvoir les lire dans mon tableau bash correctement, pour cela j'ai utilisé une fonction qui vient à la base du langage c mais qui est utilisable en bash, la fonction printf. En effet le fait d'utiliser la fonction printf me permettait de mentionner un ";" à la fin de chaque élément(date) de mon tableau lorsque je l'écrivais dans un fichier texte. Maintenant que je savais sauvegarder les dates dans un fichier je dois réussir à être capable de les lire depuis un fichier texte vers un tableau bash afin de les utiliser dans mon script. alors pour cela j'ai utilisé au départ la fonction read qui avait une option permettant de remplir un tableau et de même préciser un délimiteur, la fameuse option "-d", j'ai donc utilisé la commande read afin d'extraire les données depuis le fichier texte, cependant les indices n'étaient pas corrects, tout simplement parce que je ne sais pour quelle raison l'option "-d" ne voulait pas modifier le délimiteur de base en bash qui est l'espace " ", j'ai donc cherché sur internet dans plusieurs forums jusqu'à comprendre qu'il y avait une autre manière un peu plus forcée de changer le délimiteur pour lire mon fichier texte, c'est la variable d'environnement IFS (Internal Field Separator), alors en réalité il est plutôt déconseillé de changer la valeur de cette variable parce que si l'on oublie de la remettre à son état d'origine cela peut causer de sérieux problèmes. Ce que j'ai fait donc, c'est que j'ai changé cette variable d'environnement à la valeur ";" ce qui m'a permis de lire correctement mes fichiers .txt dans mes tableaux bash, et une fois avoir lu tous mes fichiers avec ce séparateur il suffit de remettre la variable d'environnement à son état d'origine, donc moi j'ai fait un "unset IFS" après avoir lu tous mes tableaux afin d'être sûr de ne pas causer de problèmes.

Une fois avoir réglé les soucis concernant la lecture et écriture des valeurs de température dans les fichiers .txt, il est maintenant question du calcul des métriques. Le code permettant de calculer les métriques m'a pris beaucoup moins de temps que ce que je prévoyais, en réalité le plus compliqué c'était la lecture et écriture des fichiers, mais sinon la manipulation des tableaux dans le code ne m'a pas posé trop de soucis, j'ai malgré tout eu un souci en ce qui concerne le calcul de moyenne, en effet lorsque celle-ci était calculée, elle était tout le temps entière, alors que le résultat lorsque j'ai vérifié le fonctionnement du code avec ma calculatrice se devait d'être de type réel (à virgule), j'ai donc compris que les calculs arithmétiques dans leur ensemble n'étaient pas naturellement gérés en bash. Il a fallu donc que j'utilise la commande bc afin de pouvoir assigner le résultat d'un calcul à virgule flottante à une variable, je pouvais donc utiliser cette commande en spécifiant le nombre de chiffres après la virgule que je souhaitais avec l'attribut scale.

L'un des derniers problèmes que j'ai rencontré était lié à la publication du fichier html sur mon site web, alors il faut savoir que moi j'avais codé mon html en délimitant les chaînes de caractère par des "" double guillemets, ce qui pose problème pour l'interprétation du code par bash vu que en bash les double guillemets acceptent le passage de variables et interprètent ce qu'il y a à l'intérieur, pour une chaîne de caractères en texte brut en bash il faut la délimiter par des simples guillemets ou autrement appelés guillemets droits, j'ai donc dû changer tous les double guillemets dans mon code html par des guillemets droits afin que le echo écrive correctement mon fichier html en interprétant uniquement les variables que moi j'ai choisi de le laisser interpréter.

## ANNEXES



The screenshot shows a web browser at the URL `iot.ybgn.fr/salles.html`. The page has a dark blue header with navigation links: **Accueil**, **Salles**, and **Mentions Légales**. The main title is **Température des Salles de l'IUT de Blagnac**.

There are two main sections, each containing a table of statistics and a table of the latest temperature reading.

**Statistiques E101**

Moyenne	Minimum	Maximum
23.40°C	22°C le 24/01/23 à 21:42:32	25°C le 24/01/23 à 21:32:32

**Statistiques E208**

Moyenne	Minimum	Maximum
23.40°C	23°C le 24/01/23 à 21:42:01	24°C le 24/01/23 à 21:40:02

**Relevé dernière Température E101**

Temperature	Date & Heure
22 °C	le 24/01/23 à 21:42:32

**Relevé dernière Température E208**

Temperature	Date & Heure
23 °C	le 24/01/23 à 21:42:01

The footer contains the following information:

- Département R&T IUT de Blagnac
- © Copyright 2022 Tous Droits Reservés
- W3C HTML5 and W3C CSS validation logos.

Affichage du site si moins de 20 valeurs de température ont été enregistrées

iot.ybgn.fr/salles.html

Accueil Salles Mentions Légales

### Température des Salles de l'IUT de Blagnac

Statistiques E101			Statistiques E208		
Moyenne	Minimum	Maximum	Moyenne	Minimum	Maximum
20.70°C	18°C le 24/01/23 à 22:58:32	25°C le 24/01/23 à 21:32:32	20.60°C	18°C le 24/01/23 à 23:52:02	25°C le 24/01/23 à 22:34:02

Relevés de Températures E101		Relevés de Températures E208	
Temperature	Date & Heure	Temperature	Date & Heure
19 °C	le 25/01/23 à 00:04:32	20 °C	le 25/01/23 à 00:04:01
21 °C	le 25/01/23 à 00:00:32	19 °C	le 25/01/23 à 00:00:01
20 °C	le 24/01/23 à 23:58:32	19 °C	le 24/01/23 à 23:58:01
21 °C	le 24/01/23 à 23:56:32	18 °C	le 24/01/23 à 23:56:02
22 °C	le 24/01/23 à 23:52:32	20 °C	le 24/01/23 à 23:52:02

Département R&T IUT de Blagnac    © Copyright 2022 Tous Droits Reservés    W3C HTML5    W3C CSS

Affichage du site si plus de 20 valeurs de température ont été enregistrées