

COMPTE RENDU PROJET SHUT-UP IOT M1

Jeremy HIRTH DAUMAS

Ralph EL CHALFOUN

Pour le projet final nous avons choisi le sujet SHUT-UP, le but de ce projet est d'allumer un panneau en fonction du volume sonore d'une pièce pour exprimer le fait qu'il faut baisser d'un ton, etc...

Pour la réalisation de ce projet nous nous sommes procuré des composants LED RGB adressable (x5) afin d'avoir plus de choix pour la signalisation (une bande LED aurait été sympathique mais il faudrait une alimentation à coté et cela compliquerai votre utilisation de notre code). Nous allons donc d'abord vous présenter le rendu final du projet et ensuite nous ferons un bref résumé de ce que nous avons pu développer, apprendre, approfondir grâce à ce projet.

Coté ESP

De ce côté, rien de bien compliqué nous avons 5 LEDS adressables alignés qui sont connectées à l'ESP. L'ESP s'abonne au Broker MQTT et va récupérer une valeur, en fonction de celle-ci, il va allumer un certain nombre de LEDs avec des couleurs spécifiques pour représenter le niveau du volume commun d'une pièce.

Pour les valeurs de 0 à 5, il allumera autant de LEDs que la valeur reçue.

Pour la valeur 6 il va faire clignoter toutes les LEDs en rouge pour signaler un Warning => SHUT-UP.

Voir Annexes pour des photos du résultat.

(broker : broker.hivemq.com)

(topic : sensors/level)

Application Heroku

Nous possédons deux applications Heroku :

- <https://shut-up-jr.herokuapp.com/>
- <https://is-it-loud.herokuapp.com/>

shut-up-jr est le site pour le client, ici il pourra s'inscrire, se connecter à sa session, voir son propre volume sonore et également voter pour faire allumer le panneau au cas où le système ne le ferait pas de lui-même (l'application est à base de *php & javascript*)

is-it-loud est l'application passerelle entre le site du client et l'installation lumineuse de l'ESP (à base de Node.js)

Ces deux sites communiquent entres eux pour allier **Node.js** et **PHP**.

Le site shut-up-jr

Plusieurs pages sont disponibles sur le site :

En mode déconnecté :

- La page d'accueil
- La page d'inscription
- La page de connexion
- La page de vote (qui demandera de se connecter)

En mode connecté :

- La page de profil
- La page de vote (cette fois-ci disponible)
- Le bouton de déconnexion

Pour ce qui est des pages Home, Inscription et Connexion pas besoin de détails.

Cependant pour la page de Vote en mode connecté, il y a plusieurs informations utiles :

4 types pourront s'afficher en fonction de l'état du vote :

- Vote Disponible et pas enclenché, vous pourrez cliquer sur le bouton et lancer le vote (à partir du moment où quelqu'un lance un vote, un compteur invisible de 5 minutes est lancé pour récupérer suffisamment de votes, après ces 5 minutes et s'il n'y a pas eu assez de vote, les votes sont réinitialisés et vous devez revoter)
- Un vote est en cours (après que quelqu'un d'autre ai voté) vous pourrez également cliquer sur le bouton
- Vote pris en compte, le bouton sera éteint car vous avez déjà voté
- Vote en cours de recharge, un vote s'est terminé avec succès et le délai de récupération avant le prochain vote est de 5 minutes

Un vote terminé avec succès allumera le panneau en mode SHUT-UP pendant 1 minutes.

Après ces 3 minutes il restera donc 4 minutes à attendre avant de pouvoir revoter.

Sur la page de profil vous trouverez vos informations.

Lorsque vous êtes connecté à votre compte une barre s'affichera dans l'en-tête de navigation, cette barre représente votre niveau sonore personnel (si la barre reste vide vérifiez de bien avoir autorisé le micro sur le navigateur)

A partir du moment où vous êtes connecté et que le micro est activé vos données sonores sont envoyée périodiquement au second site « is-it-loud ».

Le site is-it-loud

Ce site comme expliqué sert à collecter les données des micros de tous les utilisateurs connectés pour en faire une moyenne et envoyer cette moyenne sous forme de niveau sonore au broker MQTT auquel l'esp est abonné.

La moyenne consiste à prendre le max de chaque utilisateur pendant une période et de faire la moyenne de ces max.

Le Node.js permet aussi de collecter les votes des utilisateurs et en fonction du nombre de votant (différents), il va envoyer l'alerte de vote pour que l'esp clignote en rouge « SHUT-UP ».

En plus de cela il va gérer tous les compteurs de « time-out », à savoir :

- 5 minutes le temps du vote avant de le réinitialiser (dans le cas où pas assez de vote n'ont été récolté)
- 5 minutes le temps de refresh du vote (dans le cas où le vote est un succès et qu'il est en recharge)
- 1 minute le temps de Shut-Up (spam le warning pendant 1 minute pour afficher SHUT-UP après réussi un vote)

Enfin il permet de gérer certaines méthode POST notamment celle qui correspondent à l'état du vote.

(Dans notre cas d'exemple il suffira de deux votes pour qu'il soit réussi, mais en fonction du nombre de client présent dans la même pièce le nombre de votant pourra varier pour faire une majorité).

Conclusion

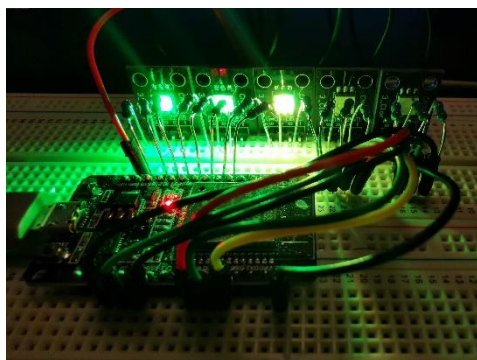
Grâce à ce projet nous avons pu développer nos compétences en HTML/PHP notamment en ce qui concerne les sessions d'utilisateur, en javascript pour la gestion du microphone.

Nous avons souvent mélangé HTML/PHP/JS/MONGODB et NODE.JS/MQTT, grâce à l'association de ces langages nous avons pu faire des scripts plus complexes avec plus de fonctionnalités.

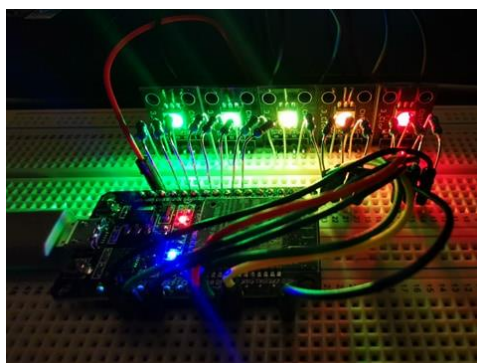
Nous avons approfondi nos méthodes post avec des fetch et avons tenté de gérer les « CORS ».

L'autonomie et la liberté du projet nous ont permis de faire nos propres recherches et de l'orienter à notre façon tout en restant dans le sujet proposé.

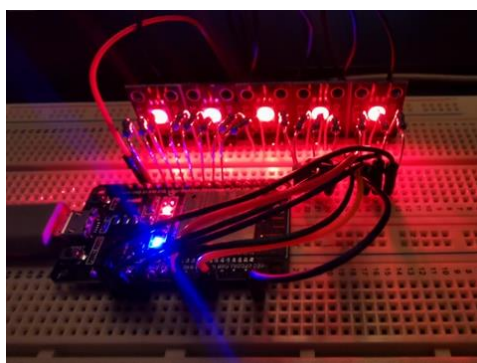
ANNEXES



NIVEAU 3



NIVEAU 5



NIVEAU 6 ~ SHUT-UP

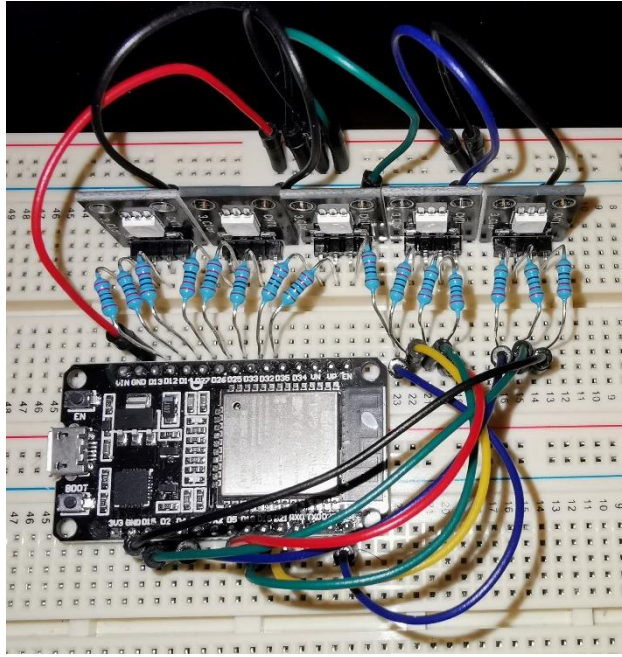


PHOTO MONTAGE