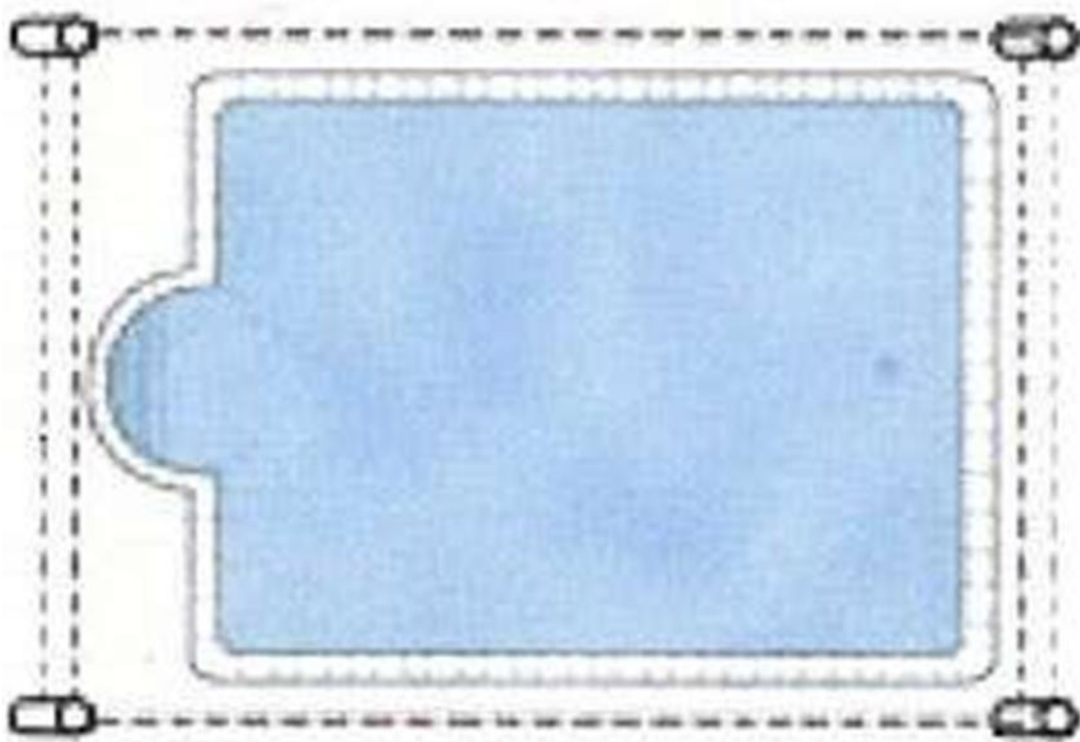


Arduin'eau sûre



Barbry Zoé
Rousseau Romain

Sommaire :

I-	Présentation du projet.....	p.3
II-	Objectif initial.....	p.3
III-	Présentation des modules.....	p.4
	1- Alarme et lasers.....	p.4
	2- Collecteur de données.....	p.4
	a) Thermomètre.....	p.4
	b) Hauteur de l'eau.....	p.5
	c) pH-mètre.....	p.5
	3- Serveur.....	p.6
IV-	Différence entre projet initial et final.....	p.7
V-	Choses que nous ferions autrement.....	p.8
VI-	Conclusion.....	p.9
	Remerciements.....	p.9
	Bibliographie.....	p.10

I- Présentation du sujet :

Nous avons vu un rapport de l'enquête NOYADES datant de l'été 2018 qui, en collaboration avec le ministère de l'Intérieur a recensé 1649 noyades accidentelles dont $\frac{1}{4}$ étant mortelle. Parmi ces noyades, il y a une quantité non négligeable d'enfant en bas âge tombant dans une piscine sans surveillance.

Ce qui fait de la noyade la première cause de mortalité par accident de la vie courante chez les moins de 25 ans en France mais concerne aussi les autres catégories d'âge.

Quand une famille avec des enfants en bas âge possède une piscine, il est nécessaire de posséder aussi un ou des appareils de sécurité avec.

Sauf que tout à un coût, aujourd'hui, les alarmes de piscine vont de 200€ pour les moins sophistiqué jusqu'à plusieurs milliers d'euros.

Donc, l'idée de notre projet était de faire une alarme de piscine efficace à moindre coût, puis au fil de nos discussions, nous avons décidé de rajouter aussi un peu de collecte de données pour pouvoir gérer au mieux sa piscine.

II- Objectif initial

Nos objectifs initiaux étaient tout d'abord de créer dans ce projet deux parties différentes :

Une partie dans laquelle il y a l'alarme et les lasers ; nous avons visualisé cette partie avec 4 cylindres posés aux 4 coins d'une piscine dans chacun desquels il y aurait un laser et un photorécepteur qui serait ainsi positionné face à face et le tout serait relié à l'alarme qui se déclencherait si un des 4 lasers est franchie par quelqu'un.

L'autre partie est le collecteur de données, dans lequel il y a le thermomètre, le pH-mètre et des résistances pour connaître la hauteur de l'eau de la piscine. Ils auraient été dans une boîte posée sur le côté de la piscine, avec une partie par laquelle le thermomètre, le pH-mètre et les résistances seraient immergés dans l'eau.

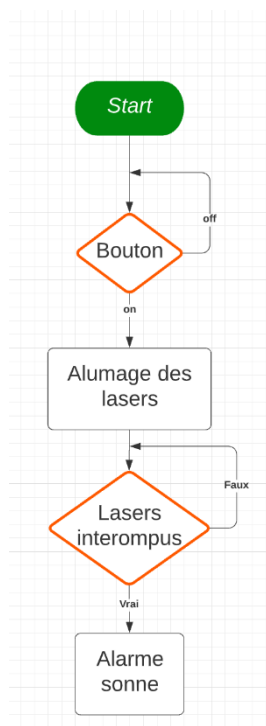
Nous voulions aussi relier le tout à un module wifi qui enverrait nos valeurs de pH, température et hauteur de l'eau sur un site internet et sur ce même site nous pouvons décider si l'alarme est en marche ou non.

III- Présentation des modules

1- Alarme et lasers

Pour les lasers et les photorécepteurs, nous les avons tous connecté avec la carte Arduino puis nous les avons alignés avec un code disant que si la valeur des photorésistances sont au-dessus de 800, rien ne se passe mais si il y a une seule photorésistance qui est au-dessous, l'alarme se déclenche.

Au niveau du montage, comme on vient de le dire, nous avons prévu de mettre les récepteurs et lasers dans des piliers mais, on a préféré les mettre directement sur les coins de la piscine car on trouvait cela plus visuel pour la démonstration d'un prototype.

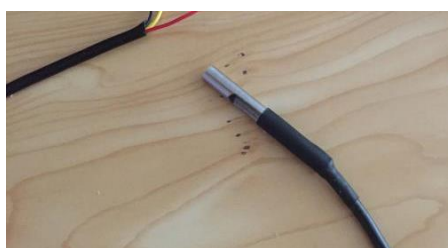


2- Collecteur de données

a) Thermomètre

En ce qui concerne le thermomètre, nous sommes d'abord allés voir comment fonctionnait le thermomètre DS18B20 qu'on nous avait donné.

Il fonctionne avec une résistance de $4.7k\Omega$, et avec la bibliothèque OneWire, donc j'ai d'abord fait le montage et j'ai trouvé un code dans la bibliothèque qui renvoyait la température calculée donc nous avons tout simplement pris ce code.



b) Hauteur de l'eau

Au début nous avions prévu de déterminer la hauteur de l'eau en plongeant plusieurs résistances à différentes hauteurs dans l'eau et donc on aurait pu avoir la hauteur de l'eau en détectant quelles résistances sont alimentées ou pas.

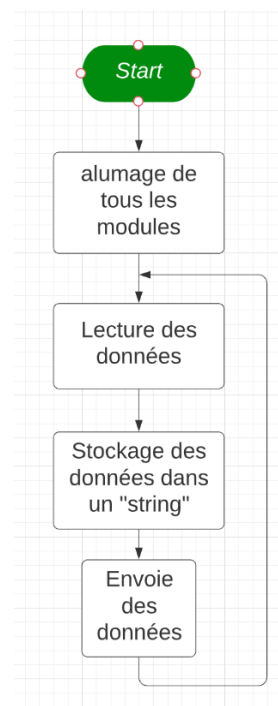
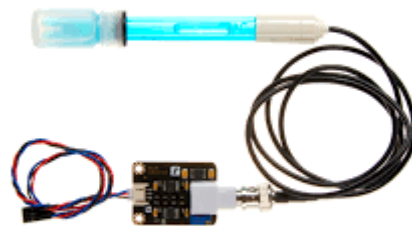
Mais c'était une solution pas très pratique et surtout pas précise du tout, donc on a préféré se servir d'un module à ultrasons dont on s'était servi en cours au début de l'année qui est beaucoup plus précis et plus simple à l'utilisation. Donc nous avons déjà le code et montage qui correspond à ce module, il nous a juste suffi de faire la différence entre la hauteur de la piscine et la hauteur que le module détecte.



c) pH-mètre

Pour le pH-mètre, nous avons eu plus de problème car nous n'arrivions pas à comprendre le fonctionnement du pH-mètre par rapport à la valeur analogique qu'il renvoie et la valeur du pH mesurée et, tous les codes qu'on trouvait sur internet ne fonctionnaient pas. Mais au bout d'un moment nous avons, en modifiant un code, trouvé le bon code qui nous donnait le bon pH du liquide.

La valeur que le pH-mètre envoie est assez variée et pas très rapide à venir, mais comme nous travaillons sur une piscine ce n'est pas très grave que le pH-mètre soit lent à donner la valeur car les variations vont être très petites.



3- Serveur

Codage ESP32

Nous avons pensé que, une fois que nos modules seraient codés et montés sur la carte Arduino Uno, il nous suffirait de brancher notre ESP32 pour pouvoir tout envoyer sur le site web. Après avoir reçu le HelTec LoRa 32, nous avons coder notre site en HTML, fait le CSS en nous aidant d'internet car aucun de nous sommes doué dans ces domaines, puis vérifié si le site se créait bien.

Ainsi nous avons donc naïvement cherché à faire communiquer les deux cartes ensemble via les RX TX, mais ne sachant pas comment faire et vu que ce que nous proposait internet ne marchait pas, nous sommes partis chercher de l'aide chez Mr. Peter. Nous étions deux groupes à vouloir procéder ainsi, le groupe d'Elisa Lartigue et Laure-Anne Bluteau et nous-même. En voyant qu'en fin de compte cette méthode ne fonctionnerait pas, nous avons décidé de rebrancher tous les modules sur l'ESP.

Cela impliqua beaucoup de changement dans notre câblage. Déjà il fallait regarder la différence entre tous les pins du LoRa 32 et ceux que nous pouvions utiliser. Pour cela nous avons encore utilisé internet. Il s'avère qu'il est conseillé d'utiliser seulement 10 pins, heureusement c'était plus qu'il nous en fallait.

De plus, l'ESP ne fait pas la différence entre sortie/entrée analogique et digital, cela ne nous posa pas de soucis en pratique, mais ce qui nous posa le plus de travail est le fait que la LoRa ne lit pas de 0 à 5 Volt en entrée mais de 0 à 3.3 Volt. Nos modules étant tous branché en 5 Volt, nous les avons tous connectés à des pont diviseur de tension pour pouvoir obtenir des valeurs correctes. Puis nous avons changé tous nos produits en croix permettant de passer de la valeur analogique à la valeur voulue. Enfin, nous avons adapté notre code HTML.

Après ce processus, nous arrivions à faire fonctionner les lasers, la hauteur de l'eau et le thermomètre mais le pH-mètre était incompatible. Nous avons donc décidé d'abandonner le HelTec LoRa 32.

Nous avons donc pris une autre carte Wi-Fi, la ESP Wroom 32. Nous savions, via l'intermédiaire du groupe d'Elisa et Laure-Anne que nous pouvions brancher celle-ci directement à l'Arduino. Pour éviter tout problème de compatibilité, nous sommes donc repartis sur le câblage des modules sur la Uno et la mise en place d'une communication entre la Uno et la Wroom.

Le transfert d'information se fait de manière assez ingénieuse : on stock nos valeurs dans une chaîne de caractère et on sépare chaque valeur par un signe "\$". Ainsi, il suffit de lire la chaîne de caractère et de mettre dans des variables chaque valeur située entre les "\$". Puis le code HTML envoie tout et la page web se réinitialise fréquemment pour pouvoir faire apparaître les valeurs les plus récentes. Après avoir reçu notre nouvelle carte, aucun problème majeur n'est à déplorer.

Arduin'eau sûr

Voici les informations sur votre piscine

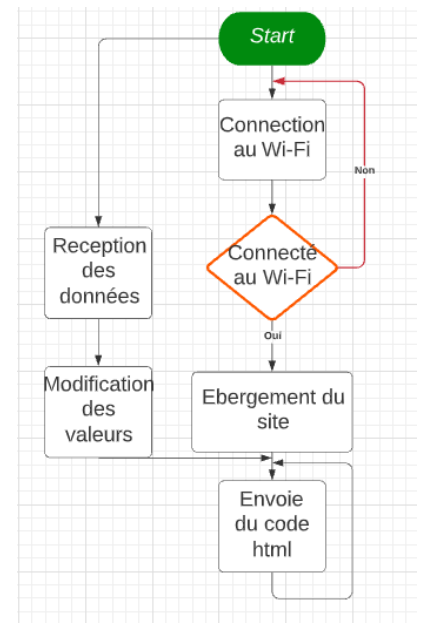
Le pH est de : 6.53

La température de l'eau est de : 15.69

La hauteur de l'eau est de : 1.11

Marche/Arrêt de l'alarme

Alarme : off



IV- Différence entre projet initial et final

Tout d'abord on peut voir que nous avons assez bien respecté le cahier des charges même s'il y a eu quelques modifications comme l'unification des lasers et récepteurs avec le collecteur de données comme nous sommes sur une petite piscine, nous avons aussi modifier la façon de désactiver l'alarme, nous voulions pouvoir le faire sur le site internet mais par manque de temps nous avons décidé de mettre un bouton à côté de l'alarme qui permettra de l'activer ou la désactiver.

A propos du diagramme de Gantt ci-dessus, on peut voir que nous n'avons pas spécialement respecté exactement les étapes à faire sur les séances mais nous avons tout fait sauf certaines étapes car, avec la modification du projet n'était plus nécessaire. Et puis il y en a certaines qui se sont rajoutées. On peut aussi voir que nous avons assez bien réussi à respecter le diagramme que nous nous étions fixés sur les premières séances mais, que par la suite nous avons fait plus de séance en binôme car nous devions résoudre les problèmes que nous avons rencontré et on préférait le faire en binôme.

Diagramme de Gantt initial :

Tâches	Scéances	1	2	3	4	5	6	7	8
Modélisation de l'extérieur du collecteur de données et des lasers et récepteurs lasers									
Montage et codage des détecteurs pour la hauteur de l'eau									
Montage et codage du pH-mètre									
Montage et codage du thermomètre									
Impression et construction de l'extérieur du collecteur de données									
Montage des lasers et récepteurs									
Assemblage et tests du collecteur de données									
Montage alarme avec les lasers									
Codage pour les lasers et récepteurs avec l'alarme									
Impression et construction de l'extérieur des lasers, récepteurs et l'alarme									
Tests des lasers et de l'alarme									
Reliement entre carte arduino des lasers et une autre carte									
Reliement entre carte arduino du collecteur de données et l'autre carte									
Création application									
Création d'un serveur									
Reliement entre les données de la carte arduino et le serveur									
Tests finaux									

Diagramme de Gantt final :

Tâches	Scéances	1	2	3	4	5	6	7	8
Montage et codage du thermomètre									
Montage et codage des détecteurs pour la hauteur de l'eau									
Montage et codage du pH-mètre									
Montage des lasers et récepteurs									
Modélisation de l'extérieur du collecteur de données									
Assemblage et tests du collecteur de données									
Montage alarme avec les lasers									
Codage et test pour les lasers et récepteurs avec l'alarme									
Montage et test des lasers et recepteurs sur la piscine									
Branchement de tout les capteurs et recepteurs									
Reliement entre carte arduino des lasers et une autre									
Reliement entre carte arduino du collecteur de données et l'autre carte									
Création site internet									
Création d'un serveur									
Codage de l'ESP32 avec le code HTML									
Montage de l'ESP32 avec tous les ponts diviseurs de tension									
Reliement entre les données de la carte arduino et le serveur									
Tests finaux									

V- Choses que nous ferions autrement

Pour l'ESP 32, on aurait aimé brancher tous nos modules directement dessus comme nous avons prévu de le faire initialement, cela nous aurait permis de récupérer les données de nos capteurs directement sur l'ESP 32 pour les mettre sur le site internet sans avoir à faire communiquer l'ESP avec la carte Arduino. Cela nous aurait aussi permis de contrôler la mise en marche de l'alarme sur le site internet comme le transfert entre le site internet et l'ESP est plus simple qu'entre le site internet, l'ESP et la carte Arduino que nous avons actuellement.

VI- Conclusion

Pour conclure, nous avons fait, une alarme qui s'allume si un des quatre lasers autour de la piscine est franchi par quelqu'un, un collecteur de données dans lequel il y a un ph mètre, un thermomètre et un détecteur à ultrasons et nous avons aussi fait avec un ESP 32, la transmission des informations du collecteur de données vers un site internet que nous avons créé.

Nous sommes très contents d'avoir pu mener à bout un projet qui marche et sommes très fiers de notre projet même s'il n'est pas totalement abouti.

Les améliorations auxquelles nous avons pensé si nous avions encore 9 séances sont, tout d'abord de cacher tous les fils, la plaquette et la carte Arduino pour rendre ce projet esthétiquement plus joli.

Nous voulions aussi mettre le bouton physique pour activer et désactiver l'alarme sur le site internet, et mettre les lasers et récepteurs dans 4 poteaux à différentes hauteurs pour qu'il y ait le moins de surface possible non sécurisé par l'alarme.

Et enfin nous avons aussi pensé à mettre sur le site internet en faisant un calcul avec le pH, si l'eau de sa piscine a besoin de chlore ou non et en quelle quantité.

Remerciements

Pour finir nous tenions à remercier Mr MASSON pour son soutien, son aide, ainsi que ses conseils techniques.

Nous voulons aussi remercier Mr PETER et Mme BENOAKTA pour leurs conseils et aide sur ce projet.

Et enfin, nous remercions nos camarades de PeiP2 pour leurs aides et conseils très précieuses.

Bibliographie :

Montage et explication pour le pH-mètre :

https://wiki.dfrobot.com/PH_meter_SKU_SEN0161

Montage et bibliothèque du thermomètre :

<https://www.carnetdumaker.net/articles/mesurer-une-temperature-avec-un-capteur-1-wire-ds18b20-et-une-carte-arduino-genuino/>

Communication entre esp32 et Arduino Uno :

<https://www.aranacorp.com/fr/communiquez-avec-votre-arduino/>

<https://www.programmingboss.com/2021/04/esp32-arduino-serial-communication-with-code.html>

<https://randomnerdtutorials.com/esp32-web-server-arduino-ide/>

Codage du CSS :

<https://www.w3schools.com/w3css/default.asp>

Codage ESP32 :

<http://emery.claude.free.fr/esp32-serveur-web-simple.html>