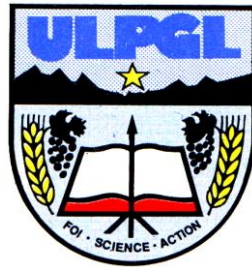


UNIVERSITÉ LIBRE DES PAYS DES GRANDS LACS

FACULTÉ DE SCIENCES ET TECHNOLOGIES

FILIÈRE DES SCIENCES DE L'INGÉNIEUR



BP. 368 GOMA

www.ulpgl.net

CONCEPTION ET RÉALISATION D'UN COMPTEUR D'EAU PRÉPAYÉ

Par : **SIKULI VAGHENI Arsène**

Travail présenté en vue de l'obtention du Diplôme de
Bachelor en Sciences de l'Ingénieur

Mention : Génie Electrique

Directeur : Prof. **BARAKA MUSHAGE Olivier**

Encadreur : Msc. **NZANZU VINGI Patrick**

ANNEE ACADEMIQUE 2023 - 2024

Epigraphe

« La technologie n'est pas bonne ou mauvaise en soi, c'est l'usage que nous en faisons qui la rend ainsi. »

William Gibson

Dédicace

A mes très chers parents, pour votre amour, votre sacrifice et votre soutiens accordés dans les instants les plus difficiles qui sont à l'origine de notre succès.

Que ce travail constitue pour vous une source de fierté et surtout de confiance.

Arsène SIKULI

Remerciements

En premier lieu, nous rendons gloire à l'Eternel notre Dieu, Maître de temps et des circonstances, pour ses merveilles inestimables et nous avoir permis de réaliser ce travail. Que la gloire te revienne Eternel Dieu !

Nos remerciements vont ensuite au Professeur Docteur Ingénieur **BARAKA MUSHAGE Olivier**, notre Directeur et au Master Ingénieur **NZANZU VINGI Patrick**, notre Encadreur. Nous avons eu le privilège de travailler à vos côtés ; votre sérieux, votre compétence et votre sens du devoir nous ont énormément marqués. Veuillez trouver ici l'expression de notre respectueuse considération et notre profonde admiration pour toutes vos qualités scientifiques et humaines.

Notre reconnaissance s'adresse par la suite à nos chers parents **KAMBALE SIKULI Alexandre** et **KAVUGHO KAMATE Marie-Hélène** pour leur soutien moral et matériel, leur volonté, leurs encouragements et leur immense amour. A vous ma tante **Hortense SIKULI** pour de multiples efforts, conseils et pour tout ce que je ne saurais exprimer ici ; merci indéfiniment ! A nos frères et sœurs : Aristide SIKULI, Alain SIKULI, Adeline SIKULI, Adrielle SIKULI, Anaëlle SIKULI pour leur soutien indéfectible à notre égard. Notre gratitude particulière à Ir Elie RUVINGA, pour ses orientations, ses connaissances et sa disponibilité inconditionnelle qui nous ont été d'une grande aide pour la concrétisation effective de notre idée. A nos amis et camarades d'auditoire : Robert KULE, Flaubert MALEKANI, Michel MBUSA, Moise MBOROYA, Philéon KIYANA et Jérémie SIVIHOLYA pour leurs encouragements et leur soutien durant toute la période passée ensemble à l'université. Que ceux qui, de près ou de loin, nous ont apporté leur soutien, tant moral que matériel, trouvent ici l'expression de notre profonde gratitude.

Arsène SIKULI

Résumé

En République Démocratique du Congo, le fonctionnement des compteurs d'eau et, par conséquent, la facturation qui en découle, suscitent fréquemment des contestations de la part des clients. Ces derniers remettent en cause les consommations qui leur sont attribuées en fin de mois, ce qui peut entraîner des actes de piratage des compteurs et la détérioration des installations d'eau. Face à cette situation et compte tenu des avancées technologiques désormais plus accessibles, nous nous sommes interrogés sur les moyens d'améliorer les compteurs d'eau afin de les rendre plus intuitifs et automatiques que les modèles actuels.

Ainsi, pour répondre à cette problématique, souvent source d'incompréhensions et de conflits entre les clients et la société de distribution d'eau (telles que REGIDESO et YMEI JIBU), nous avons conçu et développé un compteur d'eau en mode prépayé. Ce dispositif repose sur la technologie des systèmes embarqués, avec notamment l'utilisation d'Arduino. Le prépaiement est géré grâce à la technologie « Radio Frequency Identification » (RFID), tandis que la communication s'appuie sur « l'Application Programming Interface » (API). Ce compteur permet aux clients de payer uniquement la quantité d'eau dont ils ont besoin, de suivre leur consommation en temps réel et de contrôler à distance l'ouverture et la fermeture de l'arrivée d'eau.

Mots Clés : Radio Frequency Identification, Compteurs intelligents, IoT, Ressources en eau.

Abstract

In the Democratic Republic of Congo, the operation of water meters and the resulting billing process often lead to disputes from customers. Many question the accuracy of their monthly water consumption charges, sometimes leading to meter tampering and damage to water installations. Given this situation and the increasing accessibility of advanced technologies, we explored ways to improve water meters to make them more intuitive and automated than current models.

To address this issue, which frequently causes misunderstandings and conflicts between customers and water distribution companies (such as REGIDESO and YMEI JIBU), we have designed and developed a smart prepaid water meter. This device is based on embedded systems technology, using Arduino. The prepaid functionality is managed through **Radio Frequency Identification (RFID)** technology, while communication relies on **Application Programming Interface (API)**. This smart meter allows customers to pay only for the amount of water they need, monitor their consumption in real-time, and remotely control the opening and closing of their water supply.

Keywords: Radio Frequency Identification, Smart Meters, IoT, Water Resources.

Table des matières

Epigraphe	i
Dédicace.....	ii
Remerciements.....	iii
Résumé.....	iv
Table des matières	vi
Sigles et abréviations	ix
Liste des figures	x
Liste des tableaux.....	xii
0.Introduction générale	1
0.1. Généralités sur le thème	1
0.2. Identification et formulation du problème	1
0.3. Questions de recherche	2
0.4. Formulation des hypothèses.....	2
0.5. Justification du choix du sujet et motivations.....	3
0.6. Énoncé des objectifs de recherche	3
0.6.1. L’objectif général.....	3
0.6.2. Les objectifs opérationnels	3
0.7. Méthodologie et technique de recherches.....	4
0.8. Délimitation du travail	5
0.9. Subdivision du travail	5
CHAPITRE 1. REVUE DE LA LITTERATURE.....	6
1.1. Introduction.....	6

1.2	Les appareils de comptage non connectés	6
1.3	Compteurs connectés	7
1.3.1.	Objectifs du Compteur connecté.....	7
1.3.2.	Pertinences du compteur intelligent.....	8
1.4	Les compteurs intelligents en mode prépayé.....	9
1.4.1	Quelques types des compteurs intelligents en mode prépayé.....	9
1.4.2	Les systèmes de paiement.....	11
1.5	Conclusion partielle	14
CHAPITRE 2. CONCEPTION ET IMPLEMENTATION.....		15
2.1.	Ecosystème du projet.....	15
2.2.	Fonctionnement du compteur	15
2.3.	Organigramme de fonctionnement du compteur	18
2.4.	Le matériels et les logiciels.....	21
2.4.1.	Les matériels	21
2.4.2.	La partie Logicielle.....	44
2.5.	Conclusion partielle	46
Chapitre 3. RESULTATS ET SIMULATIONS.....		47
3.1.	Le compteur et le Scanneur en action	47
3.1.1.	La phase de démarrage.....	47
3.1.2.	Compte en eau vide.....	48
3.1.3.	Présentation du Scanneur et son fonctionnement	48
3.1.4.	La phase de recharge du compteur.....	49
3.1.5.	La phase de fonctionnement	51
3.2.	Les captures des notifications	51
3.3.	Les codes de fonctionnement.....	52
3.3.1.	Conversion de la somme d'argent en volume d'eau.....	53
3.3.2.	Affichage de l'état de consommations d'eau sur l'écran du compteur.....	53
3.3.3.	Fermeture automatique de l'électrovanne à la fin de la consommation du volume d'eau acheté	54

3.3.4.	Envoi de l'état de consommation sur le serveur	54
3.4.	Le câblage	55
3.5.	Application Web	57
3.6.	Application Android	60
3.7.	Estimation du coût du matériel	62
3.8.	Passage de l'air	64
3.8.1.	Condition du test	64
3.8.2.	Test et résultats	64
3.8.3.	Analyses des résultats	65
3.9.	Passage de l'eau	65
3.9.1.	Condition du test	65
3.9.2.	Test et résultats	66
3.9.3.	Analyses des résultats	66
4.	Conclusion partielle	67
CONCLUSION GENERALE		68
Bibliographie.....		70
Annexes Extrait du code source Arduino nano		73

Sigles et abréviations

API	Interface de programmation d'application
GSM	Global System for Mobile Communications
HTTP	Hyper Text Transfert Protocol
HTTPS	Hyper Text Transfert Protocol Secure
IDE	Integrated Developpement Environnement
LCD	Liquid Crystal Display
NFC	Contact Normalement Fermé
RDC	République Démocratique du Congo
RFID	Radio Frequency Identification ou Radio-Identification en français
RTC	Real Time Clock
SI	Système International
UID	Identifiant Unique
ULPGL	Université Libre des Pays des Grands Lacs
URL	Localisateur uniforme des ressources

Liste des figures

Figure 1 : Exemple du Synoptique d'un Compteur Intelligent d'eau.....	11
Figure 2 : Exemple du diagramme de fonctionnement de paiement par GSM pour un compteur électrique	12
Figure 3 : Exemple d'une carte à puce.....	13
Figure 4 : Communication entre le compteur/le client/la compagnie des eaux.....	18
Figure 5 : Organigramme du compteur au démarrage.....	19
Figure 6 : Organigramme du compteur pendant son fonctionnement	20
Figure 7 : Synoptique Global du compteur	22
Figure 8 : Représentation photographique de l'Arduino Nano.....	23
Figure 9 : Représentation photographique du NodeMCU.....	25
Figure 10 : Vue Extérieure du YF-S201	27
Figure 11 : Vue Intérieure du YF-S201	27
Figure 12 : Principe de fonctionnement du YF-S201	29
Figure 13 : Branchement Standard du YF-S201 sur un NodeMCU.....	30
Figure 14 : Module RFID RC522 avec Carte et Bracelet Mifare Classic 1k.....	31
Figure 15 : Représentation 3D de la mémoire de la MIFARE 1K	32
Figure 16 : Branchement de la RC522 avec un NodeMCU	33
Figure 17 : Afficheur LCD 16×2 avec bus I2C.....	34
Figure 18 : Branchement LCD 16×2 avec I2C sur Arduino Nano	35
Figure 19 : Modèle électrovanne 12V/DC.....	36
Figure 20 : Fonctionnement d'une électrovanne.....	37
Figure 21 : Branchement de l'électrovanne vers une carte Arduino	38
Figure 22 : Module Relai.....	39
Figure 23 : RTC DS3231	41
Figure 24 : Branchement DS3231 vers carte Arduino Nano	42
Figure 25 : Branchement d'une LED avec Arduino Nano.....	43

Figure 26 : Branchement d'un Buzzer avec Arduino Nano	43
Figure 27 : Interface du logiciel Arduino en version 2.3.3	44
Figure 28 : Phase de connexion Compteur	47
Figure 29 : Etat du Compteur lorsque le compte en volume d'eau est vide	48
Figure 30 : Présentation du Scanneur sans la carte RFID	49
Figure 31 : Présentation du Scanneur avec la carte RFID	49
Figure 32 : Cas d'une carte non identifié	50
Figure 33 : Cas d'une carte identifié et configuré pour le compteur	50
Figure 34 : Le compteur en fonctionnement normal	51
Figure 35 : Captures d'écran des notifications reçus par le client	52
Figure 36 : Extrait du code Relatif à la conversion de la somme d'argent de la carte RFID en volume d'eau	53
Figure 37 : Extrait du code relatif à l'affichage sur l'écran LCD des états de consommation	54
Figure 38 : Extrait du code portant sur la fermeture automatique de l'électrovanne à la fin de la consommation du volume d'eau acheté	54
Figure 39 : Envoi de l'état de consommation sur le serveur	55
Figure 40 : Câblage Interne	56
Figure 41 : Schéma de câblage du compteur	56
Figure 42 : Schéma de câblage du Scanner	57
Figure 43 : Page d'authentification	58
Figure 44 : Page d'accueil et Centre de contrôle	58
Figure 45 : Formulaire d'enregistrement d'un client avec sa carte RFID	59
Figure 46 : Page de gestion du profil d'utilisateur	59
Figure 47 : Page d'enregistrement d'un paiement	60
Figure 48 : Interface de connexion du client	61
Figure 49 : Monitoring et du contrôle par Client	61
Figure 50 : Interface de compte utilisateur	62

Liste des tableaux

Tableau 1 : Fiche Technique Arduino NANO	24
Tableau 2 : Fiche Technique NodeMCU	26
Tableau 3 : Branchement des broches MOSI, MISO, SCK, SDA selon les cartes programmables	34
Tableau 4 : Prix des matériels utilisés	62
Tableau 5 : Volume d'air du souffle humain mesuré par le compteur	65
Tableau 6 : Volume d'eau mesuré	66

0.Introduction générale

0.1. Généralités sur le thème

L'eau est un besoin indispensable pour nos sociétés, et aucune ne peut s'en passer. Malheureusement, une eau de qualité n'est souvent pas disponible en quantité suffisante et devrait donc être utilisée de manière responsable, tout en répondant aux divers besoins de la société. L'une des mesures pour encourager cette responsabilité chez les consommateurs d'eau est la facturation selon certaines normes telles que la facturation au mètre cube, la facturation progressifs, la facturation par abonnement, la facturation périodique etc. Afin de permettre une facturation basée sur la consommation réelle, des compteurs d'eau sont souvent installés chez les utilisateurs. Ces compteurs d'eau peuvent être de différents types, tels que les compteurs volumétriques, les compteurs électromagnétiques, les compteurs ultrasoniques, les compteurs mécaniques, les compteurs intelligents (ou compteurs connectés), etc.

0.2. Identification et formulation du problème

Les compteurs d'eau traditionnels posent plusieurs défis pour les consommateurs et les fournisseurs d'eau. Ces compteurs ne permettent pas un suivi en temps réel de la consommation d'eau, ce qui rend difficile la détection rapide des fuites et la gestion efficace de la consommation. De plus, l'absence de système de paiement prépayé peut entraîner des difficultés de paiement pour certains ménages, menant à des coupures d'eau potentiellement problématiques. Au regard de ce constat qui crée très souvent une incompréhension entre les clients et la société en charge de la fourniture d'eau dans les ménages (REGIDESO, YMEI JIBU), compte tenu des moyens techniques appris pendant notre formation ou lors des recherches effectuées sur la toile et vu la facilité d'accès à la technologie avec l'acquisition facile et peu onéreuse de composants électroniques, il nous est alors apparu de se demander avec tous ces éléments à notre disposition, comment améliorer les compteurs d'eau , les rendre plus intuitifs et automatiques par rapport aux compteurs d'eau actuel ?

Face à ceci, les moyens possibles pour résoudre les problèmes liés aux compteurs d'eau traditionnels sans système de paiement prépayé, sans la détection des fuites et l'accessibilité du service d'eau pour tous les ménages font partis de notre travail.

0.3. Questions de recherche

Dans ce travail, nous nous intéressons à la conception et réalisation d'un compteur d'eau prépayé. Pour mieux cerner cet énoncé, nous essayerons de trouver des réponses aux questions suivantes :

- De quelle manière un système de comptage d'eau prépayé serait-il convivial pour les utilisateurs ?
- Quelles technologies peuvent être utilisées pour mettre en œuvre un compteur d'eau prépayé ?
- De quelle manière le système prépayé en eau peut-il améliorer le quotidien de l'homme ?

0.4. Formulation des hypothèses

Connaissant que les hypothèses sont des réponses anticipées aux questions spécifiques d'une recherche, il sied d'évoquer les hypothèses ci-dessous sont relative à la question de compteurs d'eau qui permettrait une surveillance en temps réel de la consommation d'eau ainsi que d'autres paramètres comme :

- En le rendant convivial et cela en impliquant une interface utilisateur qui serait intuitive et facile à naviguer. Il offrirait une expérience positive pour l'utilisateur avec une amélioration de la conception et de l'interface du compteur en fournissant des instructions et des indications visuelles claires qui faciliterait la mesure et le suivi de la consommation d'eau.
- Le recours aux systèmes embarqués comme la plateforme Arduino nous permettrait de concevoir et d'optimiser notre compteur d'eau prépayé compte tenu des fonctionnalités matérielles (les capteurs, ...) et son efficacité pour mesurer avec précision la quantité d'eau consommée.

- L'implémentation d'un système de paiement prépayé pour l'eau améliorerait la vie quotidienne des hommes en augmentant l'accès à l'eau potable, en réduisant les dépenses imprévues liées à la consommation d'eau, et en favorisant une gestion plus responsable des ressources en eau, ce qui se traduit par une meilleure santé et un bien être accru au sein des communautés.

0.5. Justification du choix du sujet et motivations

La gestion efficace de l'eau est devenue une préoccupation majeure en raison de la rareté croissante des ressources en eau dans plusieurs régions du monde.

Les technologies intelligentes offrent des solutions innovantes pour améliorer la gestion des ressources en eau et encourager une utilisation plus durable.

La mise en place de compteurs d'eau prépayés présente un potentiel significatif pour optimiser la consommation d'eau et réduire les gaspillages, tout en assurant une facturation plus précise et équitable pour les utilisateurs.

0.6. Énoncé des objectifs de recherche

0.6.1. L'objectif général

Afin de répondre à la problématique, notre travail est de concevoir et mettre en œuvre un système de compteur d'eau prépayé qui permettrait de mieux contrôler la consommation d'eau, d'améliorer la gestion des ressources en eau et de faciliter la facturation équitable pour les utilisateurs.

0.6.2. Les objectifs opérationnels

La conception et la réalisation d'un compteur d'eau en mode prépayé est l'objectif principal de ce travail. Plus spécifiquement, nous voulons :

- Avec l'utilisation de la carte utilisant la technologie RFID, technologie qui est aussi utilisée par sa dérivée à savoir le NFC pourra nous être utile pour mettre en œuvre le paiement sans contact ;

- A l'aide des capteurs, prélever les données nécessaires comme le débitage et la détection de l'eau ;
- Concevoir un système de commande au tour d'un microcontrôleur qui permettrait d'acquérir les données ;
- Utiliser un module de transfert de données à NodeMCU pour envoyer ces informations à un serveur. Cela permettra d'établir une interface homme-système via une application mobile et une application web, offrant la possibilité de contrôler à distance avec le compteur.

0.7. Méthodologie et technique de recherches

Dans le cadre de notre travail, nous avons eu à recourir aux méthodes et techniques suivantes :

- **La méthode analytique** : cette méthode, nous permet de comprendre et d'analyser notre milieu de recherche en vue de bien concevoir le compteur proposé.
- **La technique d'observation et la méthode comparative** ont été utiles pour analyser le fonctionnement de différents compteurs d'eau afin de pouvoir mettre en place notre conception.
- **La technique expérimentale** : nous a permis d'expérimenter notre prototype et de tester son fonctionnement.
- **La technique documentaire** : elle nous a permis de bien appréhender notre sujet par la consultation des ouvrages et des articles ayant trait avec notre sujet.

Notre but n'est pas de développer une application comparable aux compteurs d'eau intelligent complexes existants mais avant tout de réaliser un prototype qui satisfait à notre demande pour en tirer le meilleur profit.

0.8. Délimitation du travail

Partant de l'énoncé du sujet, il nous est paru difficile, dans le cadre d'un travail de mémoire et surtout dans les délais impartis, de traiter de tous les aspects de la gestion d'un compteur d'eau intelligent. La présente recherche ayant pour centre d'intérêt la conception et réalisation d'un compteur d'eau prépayé, nous marquons nos frontières dans le contexte de la République Démocratique du Congo et particulièrement celui de la ville de Goma. Vu que le domaine technologique évolue très rapidement et chaque génération vit une réalité technologique particulière, il est important de signaler que les études menées pour ce projet ont été faites de Mars 2024 au mois de Septembre 2024.

0.9. Subdivision du travail

En plus de l'introduction et de la conclusion générale, notre travail est synthétisé autour de trois chapitres à savoir :

- **Chapitre 1** : Revue de Littérature ; qui va traiter des divers travaux effectués sur les compteurs intelligents et ceux en mode prépayé.
- **Chapitre 2** : Conception et Implémentation du projet ; qui détaille les étapes, les approches scientifiques utilisées pour concevoir notre projet et le fonctionnement de notre solution avec des outils matériels et logiciels qui seront utilisés pour la phase de réalisation.
- **Chapitre 3** : Résultats et simulations ; qui servira à tester la maquette en condition réelle afin de voir son comportement et d'en tirer certaines conclusions sur notre compteur et qui va s'atteler à présenter le produit fini de notre projet et la présentation d'extraits de codes qui vont expliquer sur le plan informatique certaines fonctionnalités du compteur.

CHAPITRE 1. REVUE DE LA LITTÉRATURE

1.1. Introduction

Les systèmes connectés sont omniprésents dans notre environnement. Grâce à leurs fonctionnalités, ils facilitent la compréhension de l'évolution de notre environnement et contribuent même à l'optimisation des ressources. Dans ce chapitre, nous présentons brièvement le concept des compteurs et surtout les compteurs communicants dotés d'un mode de paiement prépayé et autres concepts qui leur sont directement liés.

1.2 Les appareils de comptage non connectés

Ils sont utilisés pour enregistrer la consommation d'eau des abonnés domestiques et le débit des fluides dans diverses applications industrielles. Il existe différentes technologies [1]: les compteurs volumétriques et les compteurs de vitesse à turbine (jet unique ou jet multiple).

Pour les compteurs volumétriques, appelés débitmètres à déplacement positif, mesurent le débit en capturant un volume connu de fluide et en comptant combien de fois ce volume est déplacé lorsque le fluide s'écoule à travers le compteur. Ils utilisent des chambres ou des pistons qui se remplissent et se vident de fluide à chaque cycle, ce qui permet au compteur de mesurer directement le débit volumétrique. Tandis que les compteurs à turbine se basent à un rotor qui tourne lorsque le fluide le traverse avec la vitesse du rotor qui est proportionnelle à la vitesse d'écoulement du fluide.

En parlant de leurs **Objectifs et pertinences**, les compteurs volumétriques sont utilisés où une mesure très précise du débit est essentielle (comme les liquides à haute viscosité) tandis que les compteurs à turbine sont utilisés particulièrement adaptés aux fluides propres et de faible viscosité. Malheureusement ces deux types de compteurs (voir **Figure 1**) ne facilitent pas une interaction à distance pour les utilisateurs, un système post payés et même le transfert des données qu'il enregistre.



Figure 2 : Exemple du compteurs volumétriques/ à turbine

1.3 Compteurs connectés

Les compteurs intelligents ou compteurs connectés, représentent une avancée significative dans la gestion et la consommation des ressources énergétiques. Ils permettent une communication bidirectionnelle entre le fournisseur d'énergie et le consommateur, facilitant ainsi une gestion plus efficace de l'énergie.

Selon l'article de WILD Jean « Communicating Meter » [1], les compteurs intelligents, également appelés compteurs communicants, marquent une évolution par rapport aux premiers compteurs électriques, qui étaient entièrement mécaniques. Avec l'avènement de la technologie numérique dans les années 2000, ces nouveaux dispositifs ont commencé à émerger, offrant des fonctionnalités avancées. Leur adoption a été accélérée par la nécessité de moderniser les infrastructures énergétiques et de répondre aux exigences croissantes en matière de durabilité et d'efficacité énergétique.

1.3.1. Objectifs du Compteur connecté

Les compteurs connectés ont pour objectifs [1] :

- De fournir des données instantanées sur la consommation d'énergie,

- De permettre aux consommateurs de mieux comprendre leurs habitudes de consommation,
- De donner aux fournisseurs d'énergie une possibilité de gérer les compteurs à distance, réduisant ainsi les coûts liés au prélèvement manuel,
- D'alerter les utilisateurs concernant des anomalies dans leurs consommation ou des pannes de réseau.

Nous signalons que les compteurs intelligents peuvent être intégrés à des systèmes de gestion de l'énergie domestique, optimisant ainsi la consommation (qui est l'intégration des systèmes domotiques).

1.3.2. Pertinences du compteur intelligent

Vu l'évolution des compteurs dans un modèle connecté, Grégoire Wallenborn de l'Université Libre de Bruxelles, dans un rapport [2], à l'attention des parlementaires de la Wallonie en Belgique, présente les aspects qui préoccupent ces nouveaux types de compteur en ressortant une liste en bref contre les compteurs intelligents à savoir :

- La sécurité des données : La collecte et le stockage de données personnelles soulèvent des préoccupations en matière de confidentialité et de sécurité.
- Coûts d'installation : Le déploiement initial des compteurs intelligents peut être coûteux, ce qui peut constituer un obstacle pour certaines entreprises ou collectivités.
- Acceptation par les consommateurs : Certains utilisateurs peuvent être réticents à adopter cette technologie en raison de préoccupations concernant la vie privée ou la complexité perçue.
- Les compteurs communicants émettent des ondes électromagnétiques.
- L'utilisation de nouvelles ressources par les compteurs communicants.

En ayant parcouru ces études et en tenant compte des inconvénients cités ci-dessus, nous signalons le fait que le monde actuel est de plus en plus connecté alors la réalisation d'un compteur communicant est vraiment importante pour le client que pour les fournisseurs surtout que notre axe se porte sur la consommation de l'eau. Dans la majorité des cas ils servent plus à compter ou à prélever la consommation de façon directe, c'est-à-dire à mesurer sans en fixer une limite. Or notre travail veut se concentrer sur l'aspect prépayé d'un compteur intelligent c'est-à-dire faire en sorte que le client paye la quantité d'eau exacte qu'il va consommer.

1.4 Les compteurs intelligents en mode prépayé

Les compteurs intelligents ou connectés en mode prépayé représentent une évolution aussi importante dans la gestion de l'énergie, permettant aux consommateurs de payer leur consommation avant de l'utiliser. Cette approche offre une alternative aux systèmes de facturation traditionnels, avec des implications significatives pour la gestion des ressources énergétiques. Ils ont été intégrés dans plusieurs pays en développement comme une solution pour améliorer la collecte des paiements et réduire les pertes financières dues aux impayés. Avec l'avènement des technologies numériques, ces compteurs ont évolué pour devenir des dispositifs intelligents, intégrant des fonctionnalités avancées.

1.4.1 Quelques types des compteurs intelligents en mode prépayé

En ce qui concerne les compteurs prépayés, qui sont aujourd'hui largement répandus, plusieurs modèles figurent parmi les plus courants et les plus utilisés. On peut notamment citer [3]:

- ❖ **Compteurs d'électricité prépayés**, ils utilisent souvent trois systèmes de paiement dont :
 - Système de carte, qui utilisent des cartes prépayées que l'utilisateur recharge pour obtenir de l'électricité.

- Système à clé, qui fonctionne avec une clé USB ou un dispositif similaire pour recharger le crédit.
 - Système d'Applications mobiles, qui permettent aux utilisateurs de recharger leur crédit directement via des applications ou des plateformes WEB.
- ❖ **Compteurs de gaz prépayés** : pour ce genre de compteur on a souvent comme système de paiement des systèmes par carte et par clé.
- ❖ **Compteurs d'eau prépayés**, celui-ci utilise souvent :
- Le système à jeton ou les utilisateurs achètent les jetons ou des cartes pour accéder à l'eau.
 - Les systèmes basés sur des applications ou la recharge se fait en ligne via une plateforme dédiée.
- ❖ **Compteurs multifonctions** : intégrant plusieurs systèmes il est capable de mesurer plusieurs ressources comme par exemple l'électricité, l'eau et le gaz dans un seul appareil, souvent avec des fonctionnalités prépayées.

À partir de ces recherches, nous constatons que la majorité des travaux portant sur la conception de compteurs prépayés se concentrent principalement sur les compteurs d'énergie électrique. Ces dispositifs reposent généralement sur une carte programmable pour gérer leur fonctionnement. En revanche, les recherches traitant des compteurs prépayés pour l'eau restent rares, bien que leur système de fonctionnement soit très similaire.

Voici un exemple qui structure un compteur d'eau prépayé :

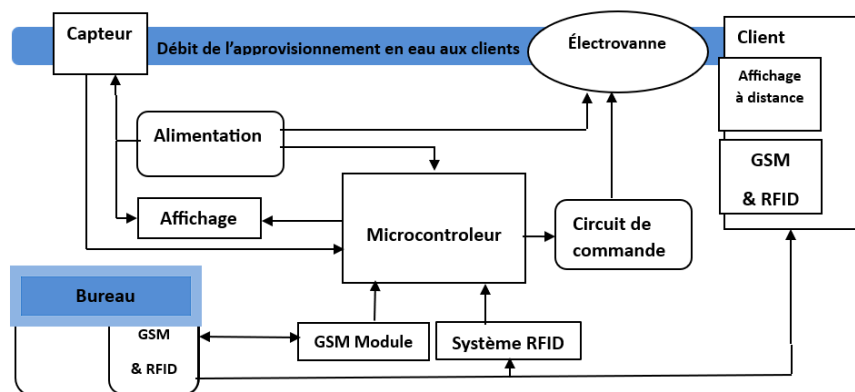


Figure 3 : Exemple du Synoptique d'un Compteur Intelligent d'eau

Le principe de conception repose sur le fait de relier le microcontrôleur (carte programmable) à :

- Un capteur qui aura le rôle de mesurer le débit d'eau et la quantité d'eau consommée.
- Une électrovanne dont le rôle est de couper l'arrivée d'eau dans le domicile ou de laisser passer l'eau en fonction de la somme d'argent payée et du volume d'eau disponible dans le système. L'électrovanne est commandée par un relai (circuit de commande) relié au microcontrôleur.
- Un écran pour afficher les informations.
- L'ensemble constitué du réseau GSM et d'un téléphone mobile pour la communication et l'échange des données entre le fournisseur et le client.
- Le système RFID/NFC assurera la gestion du paiement.

La partie dédiée au système de paiement de notre compteur s'appuiera sur l'une des méthodes existantes [2][3], ces dernières étant largement similaires dans leur fonctionnement.

1.4.2 Les systèmes de paiement

Vues les études, nous pouvons parler de trois principaux systèmes de paiement [3] [4] dont :

a) Compteurs intelligents prépayés avec technologie GSM

Dans ce système représenté ci-dessous, les consommateurs envoient un message aux compteurs intelligents via le réseau GSM après avoir rechargé leurs comptes de téléphonie mobile. Une

quantité spécifiée est transmise au compteur, qui la traduit en un nombre d'unités (correspondant au KWh pour l'électricité ou au m³ pour l'eau), stockant ces unités au sein du système. Au fur et à mesure de la consommation d'eau ou d'électricité, le compteur déduit les unités, et une fois les unités achetées sont épuisées, l'approvisionnement est interrompu. Lors de l'envoi de la prochaine recharge au compteur, celui-ci identifie le numéro de portable, décode le message et ajoute le nombre approprié d'unités à son stockage, rétablissant ainsi l'approvisionnement en électricité ou en eau [5].

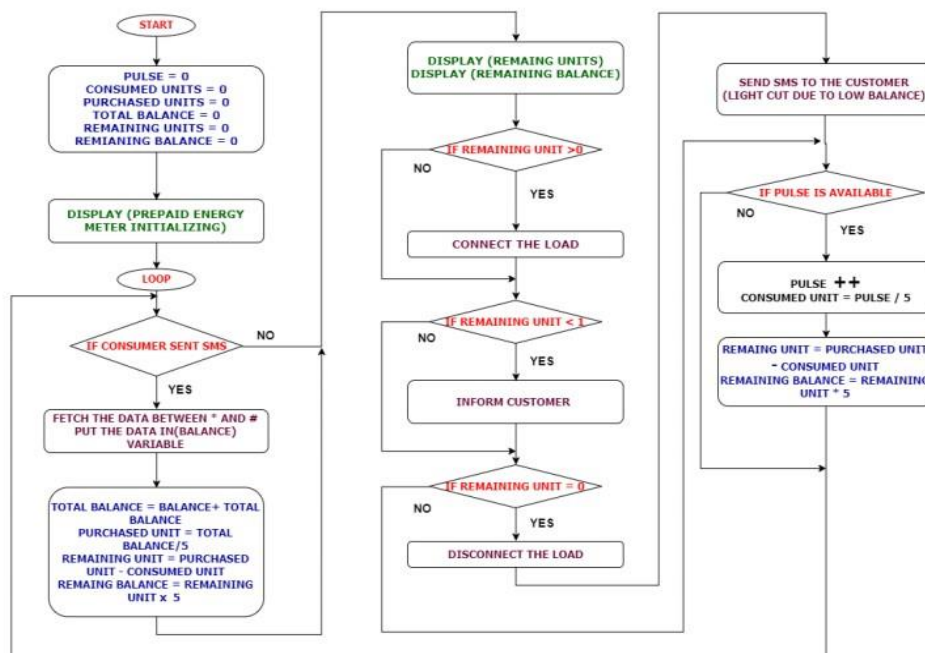


Figure 4 : Exemple du diagramme de fonctionnement de paiement par GSM pour un compteur électrique

Ce système s'avère très pratique en raison de la couverture étendue du réseau mobile GSM dans une grande partie de Goma. En conséquence, les clients peuvent activer leurs compteurs sans effort à l'aide d'un simple SMS, simplifiant ainsi l'ensemble du processus. Contrairement à notre situation, il est important de constater que les systèmes évoqués dans les différentes études se concentrent uniquement sur le système lui-même, sans aborder son application tout en vérifiant la sécurité et la traçabilité auprès des fournisseurs. Par conséquent, nous ne proposons pas un système permettant aux fournisseurs d'énergie ou d'eau d'identifier précisément quel client a consommé une quantité spécifique d'unités. De plus, les clients ont la

possibilité de manipuler le système, ce qui leur permet de générer leur propre code pour les unités souhaitées.

b) Compteurs intelligents prépayés basés sur la carte à puce

Pour ce type de compteur prépayé, il y a deux composants principaux : une carte à puce et un lecteur de carte à puce [7].

La carte à puce ressemble à une carte de crédit en plastique et contient plusieurs éléments, comme un processeur, de la mémoire ROM et de l'EEPROM. Le circuit essentiel est donc intégré dans la carte en fonctionnant grâce à un système d'exploitation qui gère les données. Dans ce système, le consommateur doit recharger sa carte avec le nombre d'unités souhaité. Ensuite, cette carte est insérée dans un lecteur de carte intégré au compteur, formant ainsi un ensemble. Le lecteur de carte récupère les unités disponibles sur la carte à puce, puis le compteur d'énergie déduit ces unités en fonction de la consommation de l'eau (en m³) ou d'électricité (en KWh). Lorsque les unités atteignent zéro, l'électricité ou l'eau est coupée jusqu'à la prochaine recharge.

Partant de la figure ci-dessous, voici comment il est structuré :

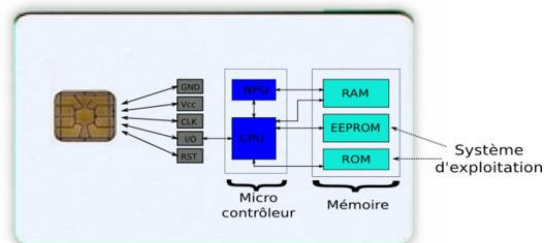


Figure 5 : Exemple d'une carte à puce

Bien que ce système soit sécurisé, car il utilise une technologie similaire à celle des distributeurs automatiques de billets ou des systèmes de paiement en supermarché, il présente un inconvénient majeur : sa rareté, ce qui en entraîne un coût élevé pour les modules intégrés aux systèmes des microcontrôleurs.

c) Compteurs intelligents prépayés basé sur la radiofréquence (RFID)

La Radio-Identification (RFID) [4] est une méthode pour mémoriser et récupérer des données à distance en utilisant des marqueurs appelés « radio-étiquettes ».

En suivant cette stratégie, les clients utilisent la carte RFID émises par les fournisseurs d'eau ou d'électricité. Les unités pourront être achetées en rechargeant les cartes RFID tout en utilisant le code dans la carte. Au moment où l'acheteur doit activer sa consommation d'électricité ou d'eau, il doit montrer sa carte au lecteur, à ce stade, le code unique à l'intérieur de la carte est perçu par le lecteur et commence à déduire de la carte RFID le montant qui équivaut à la quantité d'unité d'eau ou d'électricité achetée. Après consommation du montant total, le client doit recharger à nouveau sa carte RFID.

Cette méthode à l'avantage de la sécurité comme celle de la carte à puce et il est facile à obtenir. La solution RFID est le premier pas vers la technologie NFC qui est une technologie de paiement sans contact, disponible dans de nombreux types de téléphones.

Ainsi notre choix de la technologie de paiement se base sur des matériels faciles à trouver, qui ne nécessite pas d'énorme moyen financier pour leur acquisition et dont leur commande peut être implémenter dans le système des cartes programmables (Arduino, Esp32, ...). De ce fait le système prépayé avec la technologie GSM et le système prépayé basé sur la radiofréquence (RFID) sont les mieux placés par rapport à nos critères qu'on s'est fixés.

Quant à la carte à puce, son non choix est surtout dû au fait de la rareté de ce module avec les cartes Arduino.

1.5 Conclusion partielle

Dans ce chapitre, nous avons défini ce qu'est un compteur communicant (ou connecté), en retraçant brièvement son histoire, en décrivant sa structure, et en soulignant sa place actuelle dans notre environnement. Nous avons également abordé les différents types de compteurs intelligents ainsi que les diverses méthodes de paiement associées.

Dans le chapitre suivant, nous présenterons les différentes technologies matérielles qui seront utilisées dans notre projet.

CHAPITRE 2. CONCEPTION ET IMPLEMENTATION

Dans cette section consacrée à la conception et à l'implémentation, nous allons d'abord situer notre compteur dans son écosystème de fonctionnement et en détailler le mode de fonctionnement. Ensuite, nous présenterons les outils logiciels et matériels nécessaires à sa mise en œuvre.

2.1. Ecosystème du projet

Le compteur de notre projet fonctionne dans un système comprenant le compteur lui-même, le client et le fournisseur d'eau. Et l'ensemble concourt au bon fonctionnement du projet.

- Le compteur permet de s'approvisionner en volume d'eau, de calculer la consommation d'eau, de transmettre ces données au client et de les envoyer au serveur du fournisseur de l'eau.
- Le client lui, peut aussi contrôler l'ouverture et la fermeture du compteur à distance via une application mobile.
- La compagnie des eaux, fournit le compteur, s'occupe de la gestion des recharges des cartes via un lecteur de carte RFID permettant au client de charger son compteur en volume d'eau. Elle tient également une base de données en ligne qui permet de connaître l'état de consommation du client. Cette base de données contient aussi un registre des paiements de chaque recharge effectuée par le client.

2.2. Fonctionnement du compteur

Afin d'éviter les limitations des compteurs d'eau analogiques conventionnels, le compteur d'eau prépayé développé dans le cadre de notre projet doit offrir des fonctionnalités avancées

et apporter des améliorations significatives, tant pour les clients que pour la compagnie de distribution d'eau.

Ainsi, pour ce faire le compteur doit être capable de :

- **Faire un débitage de l'eau et un arrêt automatique ;**

Bien que l'objectif soit de moderniser le concept du compteur d'eau, il est essentiel de garantir sa fonction principale : mesurer avec précision le volume d'eau consommé par le client. Dans notre projet, cette mesure sera assurée par un débitmètre électronique. Notre compteur aura la capacité de compter le volume d'eau totale consommée et surtout de débiter le volume d'eau acheté pour chaque consommation. Une fois le volume d'eau acheté entièrement consommé, le compteur s'arrêtera automatiquement. Il ne reprendra l'alimentation en eau que lorsque le compte sera réactivé par l'achat d'un nouveau volume d'eau.

- **Permettre de charger le compte du client en un volume d'eau quand le client le souhaite ;**

Pour ce faire, comme annoncé plus haut, nous avons choisi la technologie RFID. Ainsi lors de l'achat du compteur prépayé, la compagnie d'eau fournira au client une carte RFID avec un identifiant unique, et qui sera intégré dans le code de fonctionnement du compteur, afin de faire correspondre uniquement cette carte à un unique compteur.

Dans notre modèle, pour recharger sa carte, le client devra se rendre dans une agence de la compagnie d'eau ou dans un point de vente équipé d'un lecteur de carte RFID. La recharge sera effectuée en fonction de la devise monétaire adoptée par l'agence, garantissant ainsi une traçabilité des transactions et une preuve écrite de l'opération.

Après la recharge, le client insérera sa carte dans le compteur, où un lecteur RFID intégré effectuera la lecture des données. Le microcontrôleur analysera alors l'UID (Identifiant Unique) de la carte et récupérera dans la base de données la somme associée. Cette somme sera ensuite convertie en un volume d'eau correspondant au montant payé.

Le volume d'eau ainsi crédité n'aura pas de limite de temps d'utilisation et ne s'épuisera que lorsque toute la quantité achetée aura été consommée. De plus, le client aura la possibilité d'ajouter un volume d'eau supplémentaire avant l'épuisement total de son crédit.

○ **Communiquer avec le client et la compagnie des eaux ;**

Le client devra être capable de connaître son état de consommation à tout moment qu'il le souhaite et le compteur devra lui envoyer des informations à des moments précis. Du côté de la compagnie des eaux, le compteur transmettra ces informations au serveur central, où elles seront associées aux comptes des clients. Cela permettra à la compagnie de suivre en temps réel les consommations, les recharges effectuées et d'assurer une meilleure gestion des services fournis.

Pour assurer un contrôle efficace du compteur par les clients et la compagnie des eaux, nous avons proposé la mise en place de deux outils :

- **Pour la compagnie des eaux** : Un site web sera développé afin d'assurer une couverture optimale du réseau à l'échelle de la ville, voire du pays. Ce site permettra à la compagnie de gérer les données des compteurs, de suivre les consommations et de faciliter l'interaction avec les clients.
- **Pour les clients** : Une application mobile sera mise à disposition, leur permettant de se connecter en ligne pour consulter leur consommation en temps réel et d'interagir avec leur compteur, notamment en ouvrant ou fermant l'arrivée d'eau à distance.

Concernant l'envoi des données sur le serveur en ligne, nous utilisons les fonctions Internet avec le réseau Wi-Fi du module NodeMCU. Ces données contiendront l'état de consommation du client ainsi que l'effectivité de l'activation du compte en eau. Voici la figure qui nous permet d'expliquer la communication entre le compteur, le client et la compagnie des eaux.

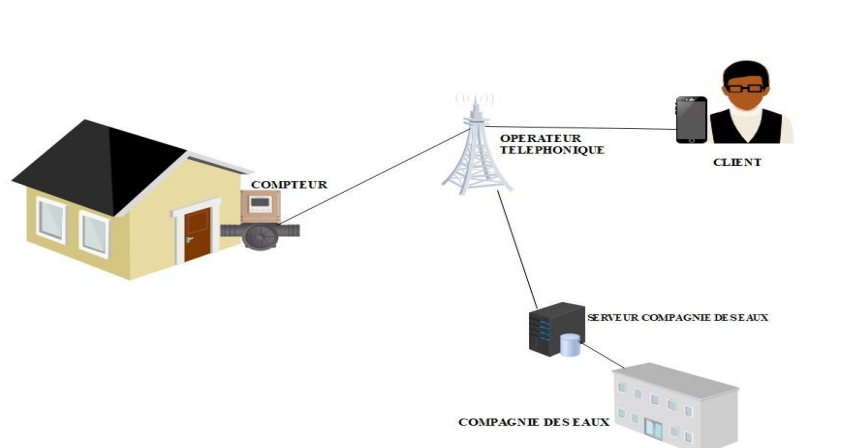


Figure 6 : Communication entre le compteur/le client/la compagnie des eaux

Après avoir listé les grands points de notre compteur, nous présentons ici de façon détaillées les actions qu'il est sensé effectuer :

- Recharger son compte en volume d'eau avec une carte RFID
- Afficher sur un écran l'état de consommation globale, et le volume d'eau restant après activation du compte.
- Envoyer sur une base de données l'état de consommation d'eau en temps réel grâce au module RTC 322.
- Permettre au client de connaître son état de consommation à tout moment grâce à une application.
- Permettre de bloquer le compteur à distance par le client s'il constate un souci de fuite par exemple.
- Être informé de la fin proche du volume d'eau acheté et avoir un blocage automatique du compteur en cas de consommation totale du volume acheté.

2.3. Organigramme de fonctionnement du compteur

L'organigramme qui explique le fonctionnement de notre compteur est structuré dans deux états :

- ✓ Au démarrage du compteur et
- ✓ Au moment du fonctionnement de celui-ci.

○ La phase de Démarrage ;

Lors de l'allumage du compteur, la première action sera la connexion entre les modules et le serveur.

Ensuite à la fin de cette phase, notre compteur est encore à zéro litre (compteur vide), cela implique une fermeture de l'électrovanne du compteur sachant que le module RFID pour pouvoir recharger son compte en volume d'eau est toujours activé.

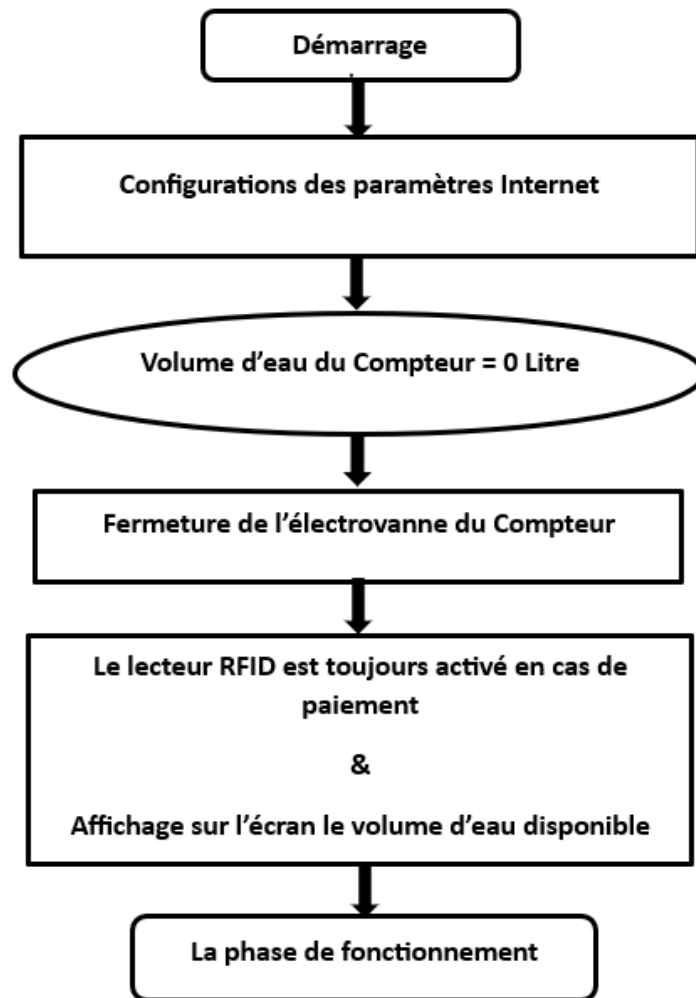


Figure 7 : Organigramme du compteur au démarrage

○ La phase de fonctionnement du compteur ;

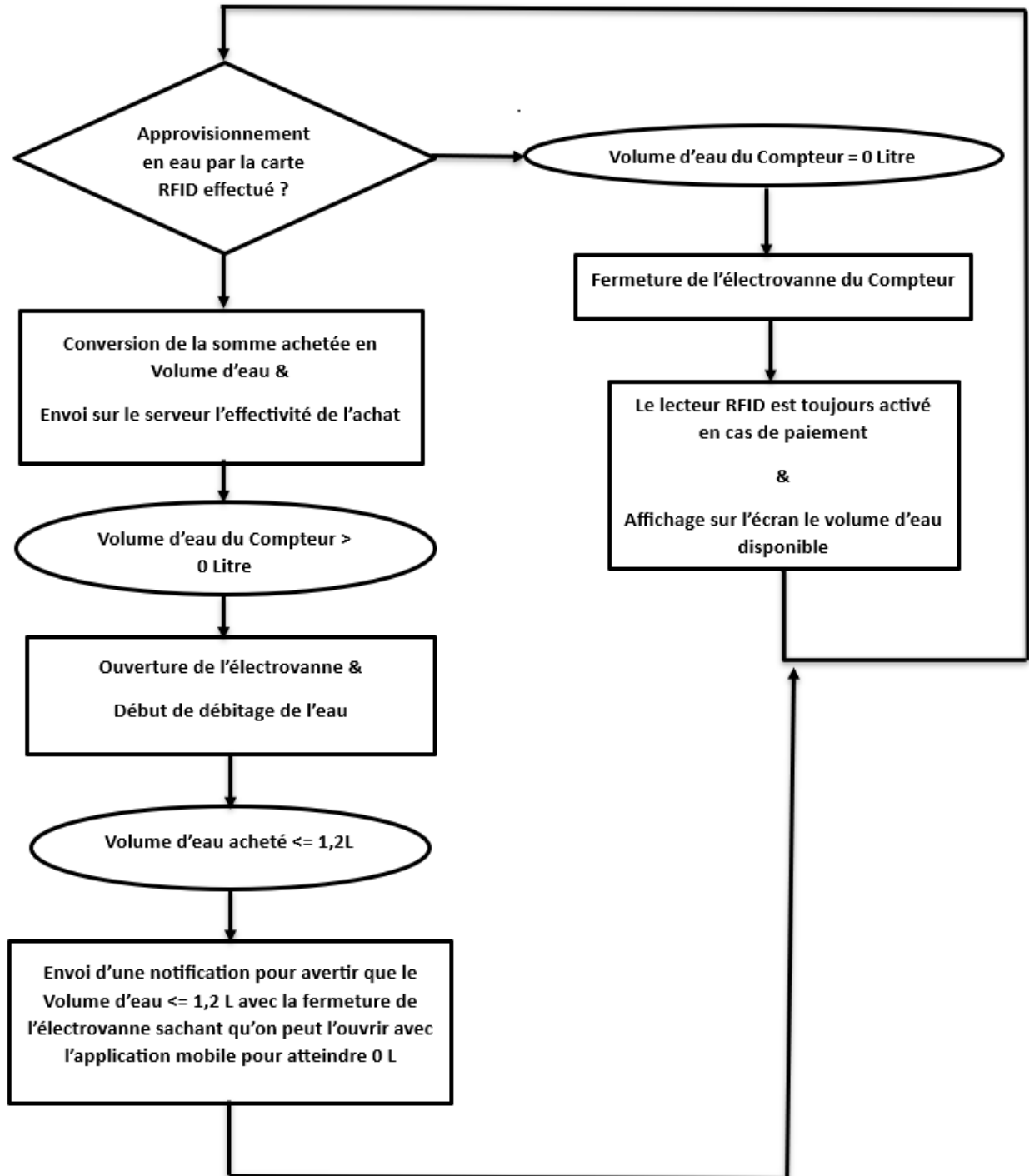


Figure 8 : Organigramme du compteur pendant son fonctionnement

Pour le fonctionnement du compteur, on part de deux choix donc :

- Si le compte n'est pas activé, le volume d'eau du compteur reste à zéro, ceci activera le lecteur de carte RFID pour le paiement de son compte et l'affichage d'un message sur l'écran qui va tourner en boucle, invitant le client à recharger son compte en eau.
- Si le compte est activé, le système convertit l'argent payé en volume d'eau et envoie au serveur l'effectivité de l'achat. Ainsi le volume d'eau est supérieur à zéro, entraînant l'ouverture automatique de l'électrovanne et début de débitage. Et toutes les fonctionnalités internes du compteur sont activées.

En continuant le débitage, nous arrivons à un volume seuil qui enclenche l'envoi d'une notification au client le prévenant de la fin prochaine de son volume d'eau acheté et lui donnant la possibilité de payer en avance un volume d'eau de réserve qui s'ajoutera immédiatement au volume d'eau précédent. Et ceci permet de rester dans le cycle normal de fonctionnement. Au cas contraire, le débitage va continuer jusqu'à la consommation totale du volume d'eau acheté, entraînant l'entrée dans le cycle de non activation de volume d'eau.

2.4. Le matériels et les logiciels

La conception de notre projet est matérialisée par un ensemble des composants nécessaires que nous proposons pour faire marcher notre système.

2.4.1. Les matériels

Nous allons présenter le fonctionnement global du compteur en lui-même, faire une étude théorique, faire le choix de ces composants et modules. Pour ce faire nous allons présenter leurs principes de fonctionnement, les caractéristiques et critères de choix ainsi que le dimensionnement.

a) Synoptique Global du compteur

Le synoptique ci-dessous présente la disposition globale du compteur ainsi que les composants et les modules qui les composent.

Ainsi il se compose de :

- Le bloc d'alimentation qui a en sortie une tension de 12 V.
- L'unité de traitement qui va commander tout le système.
- Le Module Wi-Fi qui va servir de connexion entre compteur, client et à la compagnie des eaux. Et aussi l'envoi des données de consommations sur un serveur en ligne.
- L'écran qui va servir à afficher des informations sur la consommation.
- Le lecteur RFID pour la lecture des cartes afin de charger le compteur en volume d'eau.
- Une électrovanne pour gérer les coupures d'eau automatiques ou via un ordre du client auprès de la compagnie des eaux.
- Un relais pour pouvoir commander l'électrovanne.
- Un module horaire RTC.
- Un débitmètre dont le rôle sera de mesurer le volume d'eau consommé.

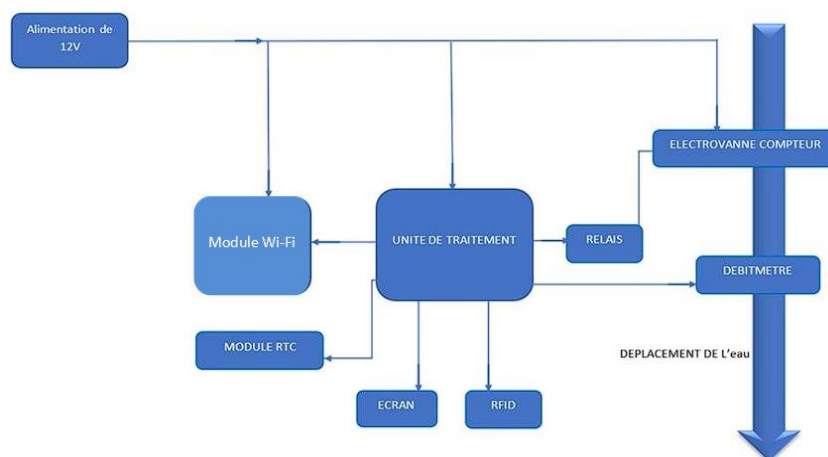


Figure 9 : Synoptique Global du compteur

b) Le choix des composants matériels du compteur

En se basant sur le synoptique du compteur, un choix de matériel bien précis sera nécessaire pour pouvoir réaliser une maquette composée d'éléments bien précis qui vont concourir au bon fonctionnement de l'ensemble du système pensé lors de la conception.

Ainsi nous avons :

- L'unité de traitement qui est une Carte ARDUINO NANO.
- Le module programmable NodeMCU qui intègre le module Wi-Fi
- Le débitmètre YF-S201
- Le module RFID RC522.
- L'écran LCD 16×2
- Le Module relais de 12 V
- Le module RTC DS3231
- L'électrovanne qui est de 12V /DC
- Les fils de branchements, un Buzzer et des leds

1. La carte ARDUINO NANO

Ce module est le contrôleur jouant un grand rôle dans notre système, il encapsule l'intelligence de ce dernier et sert essentiellement à interconnecter tous les autres composants matériels.

i. Vue d'ensemble

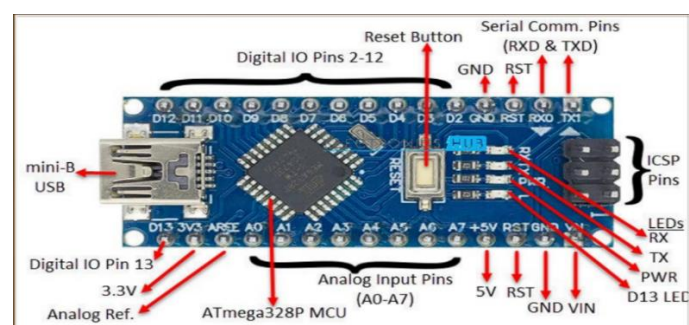


Figure 10 : Représentation photographique de l'Arduino Nano

La carte Arduino NANO est une carte à microcontrôleur basée sur un ATMEGA328 et qui dispose de [8] :

- 14 Broches numériques d'entrées/sorties dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée),
- 8 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques),
- 2 UART (port série matériel),
- Un quartz 16Mhz,
- Une connexion USB,
- Un connecteur d'alimentation jack,
- Un connecteur ICSP (programmation « in-circuit »),
- Un bouton de réinitialisation (Reset).

ii. Fiche Technique

Tableau 1 : Fiche Technique Arduino NANO

Microcontrôleur	Arduino NANO
Tension de fonctionnement	5V
Tension d'alimentation (Recommandée)	7-12V
Tension d'alimentation (Limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches d'entrées analogiques	8 (utilisables en broches E/S numériques)
Intensité max. disponible par broche E/S (5V)	40 mA (ATTENTION : 200mA cumulé pour l'ensemble des broches E/S)
Mémoire Programme Flash	32 KB dont 2 KB sont utilisées par le chargeur de démarrage
Mémoire SRAM (mémoire volatile)	2 KB
Mémoire EEPROM (mémoire non volatile)	1 KB
Vitesse d'horloge	16 MHz
Longueur	45 mm
Largeur	18 mm

Poids	7 g
--------------	-----

2. La carte NodeMCU

Le module NodeMCU est une plateforme de développement conçue pour les projets d'Internet des Objets (IoT). Il est basé sur le microcontrôleur ESP8266, qui est très populaire en raison de sa connectivité Wi-Fi intégrée et de son faible coût. Avec les fonctionnalités qu'il offre comme [9] :

- **Le Wi-Fi MeSH** : Capacité à créer ou se connecter à un réseau sans fil.
- **Le Serveur Web** : Peut-être configuré pour héberger un serveur web simple pour contrôler des appareils à distance via un navigateur.
- **L'API REST** : Possibilité de créer des API (Interface de programmation d'application) pour communiquer avec d'autres services web.

Ainsi, il sera très utile dans notre projet car il servira pour la communication des différents matériels du système et la transmission des données. On utilisera 2 cartes, dont l'un servira dans la partie Scanneur et l'autre dans le compteur.

i. Vue d'ensemble

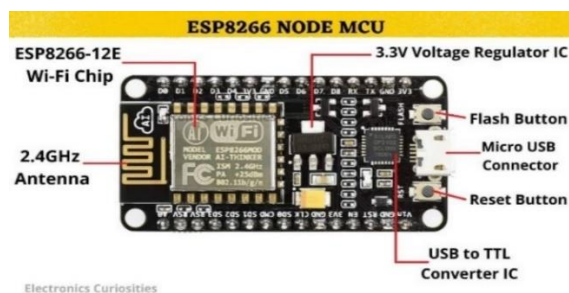


Figure 11 : Représentation photographique du NodeMCU

La carte NodeMCU est une carte à microcontrôleur basée sur un ESP6266 et qui dispose de [10] :

- 15 Broches numériques d'entrées/sorties dont 6 peuvent être utilisées en sorties PWM (largeur d'impulsion modulée),

- 1 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques),
- 1 UART (port série matériel),
- Un quartz 80 Mhz,
- Une connexion USB,
- Un connecteur d'alimentation jack,
- Un connecteur ICSP (programmation « in-circuit »),
- Un bouton de réinitialisation (Reset).

ii. Fiche Technique

Tableau 2 : Fiche Technique NodeMCU

Microcontrôleur	NodeMCU
Tension de fonctionnement	3,3V
Tension d'alimentation (Recommandée)	5V
Tension d'alimentation (Limites)	5-9V
Broches E/S numériques	15
Broches d'entrées analogiques	1 (utilisables en broches E/S numériques)
Mémoire Programme Flash	4 Mo
Mémoire SRAM (mémoire volatile)	64KB
Mémoire EEPROM (mémoire non volatile)	128 KB
Vitesse d'horloge	80 MHz
Longueur	58 mm
Largeur	31 mm
Poids	12 g

3. Le débitmètre YF-S201

Pour la mesure de la consommation en eau de notre compteur, nous allons nous servir d'un débitmètre qui est un appareil destiné à mesurer le débit d'un fluide, liquide ou gazeux.

En fonction du débit et de la nature du fluide, la nature du débitmètre varie beaucoup et nous pouvons citer quelques exemples tels que [4] :

- Des débitmètres à ultrasons, des anémomètres, des débitmètres ioniques qui sont basés sur la mesure de la vitesse du fluide ;
- Des débitmètres massiques ;
- Des débitmètres électromagnétiques pour des liquides conducteurs ;
- Des débitmètres à pistons, à rotors, à turbines qui vont se baser sur le fait d'utiliser le fluide pour entraîner quantitativement des systèmes mécaniques mobiles.

Et c'est dans cette dernière catégorie que se retrouve le débitmètre, principalement à rotors, que nous allons utiliser pour notre compteur, et qui permet de récupérer les données de consommations de façon électronique. Notre choix s'est porté sur le YF-S201.

i. Vue d'ensemble



Figure 12 : Vue Extérieure du YF-S201



Figure 13 : Vue Intérieure du YF-S201

ii. Fiche Technique

Sa fiche technique est composée de [4]:

- Type de capteur : Capteur à effet Hall
- Tension de fonctionnement : 5 à 18 V CC
- Consommation de courant max. : 15mA @ 5V
- Type de sortie : 5V TTL
- Débit de travail : 1 à 30 litres / minute
- Plage de température de travail : -25 à + 80 °C
- Plage d'humidité de travail : 35% -80% HR
- Précision : $\pm 10\%$
- Pression d'eau maximale : 2,0 Mpa = 20 Bars
- Cycle de sortie : 50% + -10%
- Temps de montée en sortie : 0,04us
- Temps de chute de sortie : 0,18us
- Caractéristiques d'impulsion de débit : Fréquence (Hz) = 7,5 \times Débit (L / min)
- Impulsions par litre : 450
- Durabilité : minimum 300 000 cycles
- Longueur de câble : 15 cm
- Raccords de tuyaux nominaux 1/2 " , diamètre extérieur 0,78" , filetage 1/2 " (1"=25.4mm)
- Taille : 2,5 "x 1,4" x 1,4 "

iii. Le Fonctionnement

Le YF-S201 est sur un capteur qui fonctionne grâce à l'effet Hall. Effet hall qui est énoncé comme suit : « *un courant électrique traversant un matériau baignant dans un champ magnétique, engendre une tension perpendiculaire à ce dernier* ». Sa forme mathématique est donnée par la relation suivante [4]:

$$v_H = \frac{IB}{nq}$$

Où :

V_H : la tension de Hall (en Volts),

I : le courant traversant le matériau (en Ampères)

B : l'induction du champ magnétique (ou densité de flux magnétique) perpendiculaire au matériau (en Teslas),

n : la densité volumique des porteurs de charge (en m^{-3}),

q : la charge élémentaire des porteurs (en coulombs)

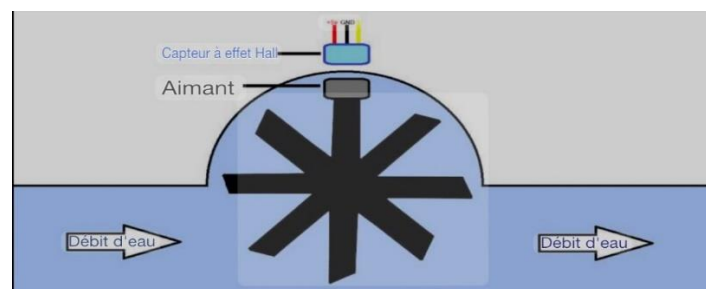


Figure 14 : Principe de fonctionnement du YF-S201

Nous avons une turbine à roue encastrée avec un aimant qui est placée sur une enveloppe en plastique fermée et un capteur à effet Hall placé de façon fixe. Lorsque l'eau s'écoule à travers le débitmètre, cela fait tourner la roue de la turbine et donc le flux magnétique interfère avec le capteur à effet Hall, le taux d'interférence dépend de la vitesse du flux de l'eau, de sorte que le capteur à effet Hall produit en sortie un signal d'impulsion. Et cette impulsion émise permet de calculer le débit de l'eau ainsi que le volume d'eau par la suite. Le signal d'impulsion est une simple onde carrée, il est donc assez facile d'enregistrer et de convertir en litres par heure en utilisant la formule suivante [4] :

$$\text{Fréquence d'impulsion (Hz)} = 7.5Q$$

Avec Q qui est le débit d'eau en litres/ minutes

De cette formule fournie par le constructeur, nous pouvons en tirer à la fin le volume d'eau consommé dès que le rotor cesse de tourner.

Et ce volume d'eau est obtenu à partir de la formule

$$\text{Volume d'eau (L)} = Q/60$$

iv. Le Branchement

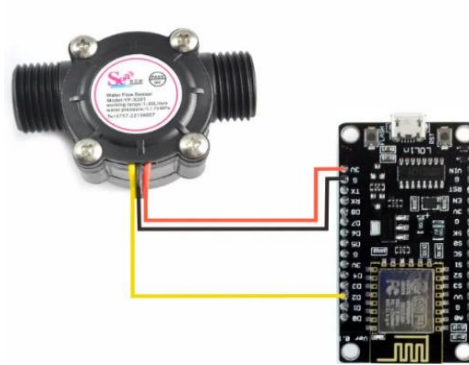


Figure 15 : Branchement Standard du YF-S201 sur un NodeMCU

Comme vous le voyez les Fils de branchement suivent une logique bien précise :

- Le fils Rouge : se connecte au 5V de l'Arduino.
- Le fil Noir : se connecte au GND de l'Arduino.
- Le fils Jaune : se connecte à un port PWM ou Analogique selon le choix.

4. Le RFID RC522

Pour le paiement de notre compteur, comme précisé plus haut, nous avons fait le choix de le faire avec une carte RFID, technologie RFID que nous avons déjà présentée. Les modules RFID qui permettent une identification et un suivi automatique des cartes contenant des informations et des données stockées numériquement.

Ainsi pour pouvoir utiliser cette fonctionnalité avec notre carte Arduino, nous avons opté pour le module RFID RC522, du fait de son accessibilité rapide et de la carte et du bracelet 1k qui l'accompagnent pour son fonctionnement.

i. Vue d'ensemble



Figure 16 : Module RFID RC522 avec Carte et Bracelet Mifare Classic 1k

ii. Fiche Technique

Sa fiche technique est présentée comme suit [4]:

- Basée sur une puce MFRC522 de NXP
- Tension d'alimentation : 3,3 V
- Courant : 13-26mA
- Fréquence d'utilisation : 13,56 MHz
- Distance opérationnelle : 0 ~ 60 mm
- Interface : SPI
- Dimensions : 40 mm × 60 mm
- Courant de travail : 13—26mA / DC 3.3V
- Courant de veille : 10-13mA / DC 3.3V
- Vitesse de communication des données : Maximum 10Mbit / s
- Cartes Supportée : Mifare1 S50, Mifare1 S70, Mifare UltraLight, Mifare Pro, Mifare Desfire
- Température de fonctionnement : -20 à 80 °C
- Humidité : humidité pertinente de 5% à 95%

iii. Le Fonctionnement

Le module RC522 [4], qui est un module qui permet la lecture et des moyens d'identification sans contact à savoir des cartes (badges) ou des clés RFID. Il est basé sur le circuit intégré

Philips RC522 et communique avec Arduino via l'interface SPI. Il utilise la bande de fréquence ISM 13,56 MHz, la distance de communication peut atteindre 6 cm.

Il a aussi la possibilité de fonctionner avec les cartes de types NFC, vu qu'ils utilisent la même fréquence, mais ces derniers fonctionnent de façon optimale à une distance de 1 cm.

Le mécanisme du RC522 est basé sur des communications radio à courte portée utilisant les normes ECM-A340 et ISO / IEC 18092.

5. La Carte MIFARE 1K

Comme dit plus haut, la carte MIFARE 1K est celle qui a été utilisé pour notre projet, et bien entendu en plus des fonctions identifications de la carte, nous avons intégré une écriture de données qui permet de convertir cette donnée en valeur financière. Et cela nécessite de comprendre comment est structuré la mémoire de cette carte.

Comme son nom l'indique la MIFARE 1K, contient une mémoire de 1Ko qui est compartimenté de la façon suivante :

- 16 Secteurs (Numérotés de 0 à 15)
- Chaque Secteur est divisé en 4 Blocs (Numérotés de 0 à 3)
- Chaque Bloc qui contient 16 octets de données (Numérotés de 0 à 15)

Nous avons ainsi de façon résumée [13]:

$16 \text{ Secteurs} \times 4 \text{ Blocs} \times 16 \text{ octets de données} = 1024 \text{ o} = 1 \text{ Ko}$
--

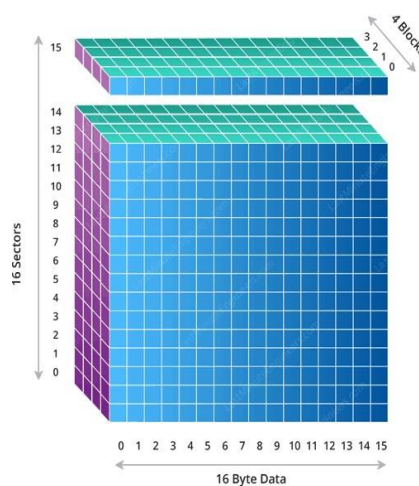


Figure 17 : Représentation 3D de la mémoire de la MIFARE 1K

- Le premier bloc du secteur 0 est appelé Bloc du Fabricant. Il contient des informations mises par le fabricant et contient le numéro d'identification unique de la carte (UID). Cet UID prend 4 octets de données.
- Le dernier bloc de chaque secteur est appelé « Sector Trailer », contient des informations appelées « Accès Bits » permettant d'accorder un accès en lecture et en écriture aux blocs restants du secteur. Cela implique que seuls les 3 derniers blocs (blocs 0, 1 et 2) de chaque secteur peuvent réellement être utilisés pour le stockage de données, ce qui signifie que 48 octets dans chaque secteur de 64 octets peuvent être réellement utilisés à nos propres fins.

i. Le Branchement

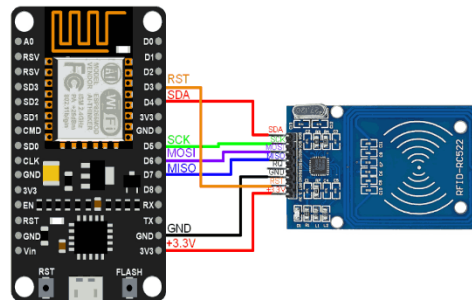


Figure 18 : Branchement de la RC522 avec un NodeMCU

Le RC522 comporte diverses broches qui se connectent suivant une logique que voici [14] :

- VCC : qui se connecte au port 3.3 V de la carte Arduino et non la 5V pouvant causer une surtension de la carte et la griller.
- GND : qui se connecte au port de la masse de la carte Arduino GND.
- RST est une entrée pour la réinitialisation et la mise hors tension.
- IRQ : est une broche d'interruption qui alerte la carte Arduino lorsque la carte RFID arrive à proximité et il a la particularité de ne pas se connecter physiquement à la carte Arduino.
- MISO (Master In Slave Out) : agit comme un maître en sortie esclave lorsque l'interface SPI est activée.
- MOSI (Master Out Slave In) : est l'entrée SPI du module RC522.

- SCK (Serial Clock) : accepte les impulsions d'horloge provenant du bus SPI de la carte Arduino.
- SDA : agit comme signal d'entrée et aussi comme données série lorsque l'interface SPI est activée.

Mais à part les pins d'entrées GND ,3.3 V et RST (le RST qui se met dans n'importe quel port numérique) qui sont communes à toutes les cartes Arduino, le branchement des autres broches du RC522 diffère selon le type de cartes et suivant le plan fournit dans le tableau ci-après :

Tableau 3 : Branchement des broches MOSI, MISO, SCK, SDA selon les cartes programmables

Types des cartes	MOSI	MISO	SCK	SDA
Arduino Uno	D11	D12	D13	D10
Arduino Nano	D11	D12	D13	D10
Arduino Méga	D51	D50	D52	D53
NodeMCU	D7	D6	D5	D4

6. L'écran LCD 16×2

Pour afficher les informations de consommation du client, nous avons opté pour l'utilisation d'un écran LCD 16×2 avec un bus I2C.

i. Vue d'ensemble

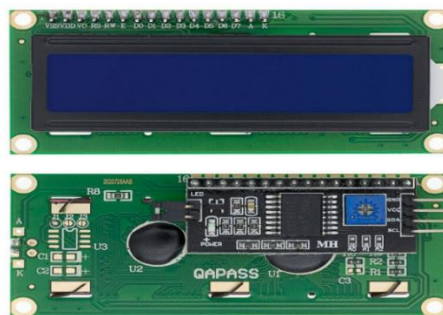


Figure 19 : Afficheur LCD 16×2 avec bus I2C

LCD (Liquid Cristal Display) ou Ecran à Cristaux Liquides en français est un afficheur qui utilise la propriété de modulation de lumière des cristaux liquides.

Ainsi les écrans à cristaux liquides sont composés de deux couches de polariseurs, avec des directions de polarisation perpendiculaire, prenant en sandwich deux plaques de verres entre lesquelles sont placés les cristaux liquides. Sur les plaques de verre se trouve une matrice d'électrodes pour chaque pixel. Une tension appliquée entre les électrodes d'un pixel entraîne un changement d'orientation des molécules et donc la transparence du pixel qui peut alors laisser, ou non, passer la lumière du rétroéclairage.

Il est appelé LCD 16×2 du fait qu'il permet un affichage de 16 caractères sur 2 lignes.

ii. Fiche Technique

Sa fiche technique se présente comme suit [13] [4]:

- Alimentation : 5 V
- Interface I2C (adresse 0x27)
- Caractères blancs sur fond bleu
- Contraste ajustable via potentiomètre
- Dimensions : 80 x 38 x 18 mm

iii. Le Branchement

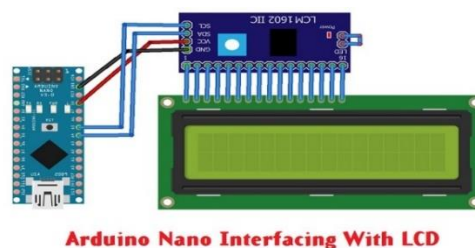


Figure 20 : Branchement LCD 16×2 avec I2C sur Arduino Nano

Pour brancher notre écran LCD à la carte Arduino, nous avons opté de le coupler à un bus I2C (Inter-Integrated Circuit) qui est un bus informatique réalisé par Philips pour les applications de domotique et d'électronique domestique permettant de relier facilement un microprocesseur

et différents circuits tout en réduisant le nombre de lignes nécessaires à seulement deux lignes : SDA (Serial Data) et SCL (Serial CLock). Il intègre un potentiomètre pour régler l'éclairage de l'écran.

Pour le branchement à la carte Arduino, nous avons [15]:

- GND vers le pin GND
- VCC vers le pin 5V
- SDA vers le pin A4 de la Nano
- SCL vers le pin A5 de la Nano.

7. L'électrovanne 12V/DC

Pour le système de coupure automatique des eaux, nous avons opté pour l'utilisation d'une électrovanne, qui sert de coupure à la fin de notre abonnement en eau et aussi à la coupure à distance de la part du client en cas de souci par exemple.

L'électrovanne encore appelée électrovalve est une vanne commandée de façon électronique et qui agira sur le débit de l'eau en sortie [4].

i. Vue d'ensemble



Figure 21 : Modèle électrovanne 12V/DC

ii. Fiche Technique

Voici sa fiche technique [4]:

- Température de travail : 90 °C
- Diamètre nominal : 8mm

- Matériau d'étanchéité : caoutchouc
- Milieu de travail : eau du robinet, eau pure
- Pression minimale : 0.8 Mpa = 0.8 Bar
- Taille : 9 * 6 * 6 cm
- Poids : 110 g
- Tension de fonctionnement : 12 V

iii. Fonctionnement

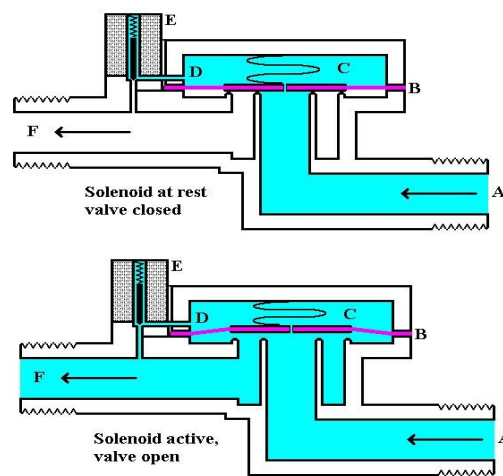


Figure 22 : Fonctionnement d'une électrovanne

Avec :

- A-** Arrivée de l'eau
- B-** Diaphragme
- C-** Chambre de pression
- D-** Conduit
- E-** Électro-aimant
- F-** Sortie de l'eau

Ainsi pour fonctionner, l'électroaimant est excité par le courant électrique, créant un champ magnétique agissant sur la bobine solénoïde qui se trouve au-dessus du diaphragme. Cette bobine va agir sur le diaphragme qui va se soulever, permettant ainsi le passage de l'eau.

Et dès que le courant envoyé est arrêté, on se trouve en opération inverse, faisant baisser le diaphragme et ainsi empêcher l'eau de passer.

iv. Branchement

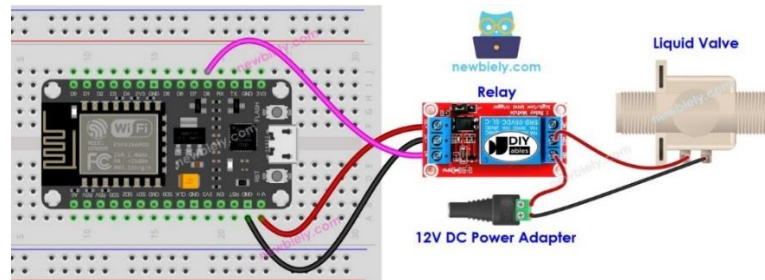


Figure 23 : Branchement de l'électrovanne vers une carte Arduino

Du fait que les cartes Arduino quels que soit les modèles ne permettent que de commander que des éléments de 5V au maximum [18] [4], il est nécessaire de passer par un relai. Ce relai permet de commander des composants qui demandent une tension supérieure à 5V.

Ainsi le relai agit comme un interrupteur qui va éteindre et allumer notre électrovanne selon nos envies. S'agissant du fonctionnement concret du relai nous le verrons dans la partie suivante qui lui est consacrée.

8. Le module Relai

Comme on l'a dit ci-haut la commande des modules demandant une tension d'alimentation de plus de 5V nécessite un relai, qui comme son nom l'indique sert de relai entre la carte Arduino et le module à commander mais pour notre système du fait de la présence d'une électrovanne, nous avons opté pour un relai, qui permet de commander un composant fonctionnant sous une tension que la carte Arduino ne peut fournir.

i. Vue d'ensemble



Figure 24 : Module Relai

ii. Fiche Technique

Sa fiche technique se présente comme suit [18] :

- Sortie maximale du relai : DC 28V / 5A, AC 240V / 5A
- Dimensions : 27 x 21 x 41 mm
- Poids : 45g

iii. Fonctionnement

Déjà il faut savoir qu'un relai est divisé en 5 parties :

- **Un Électroaimant** : il se compose d'un noyau de fer enroulé par une bobine de fils. Lorsque l'électricité passe à travers, elle devient magnétique.
- **Une Armature** : la bande magnétique mobile est connue sous le nom d'armature. Lorsque le courant les traverse, la bobine est énergisée produisant ainsi un champ magnétique qui est utilisé pour activer ou désactiver les points normalement ouverts (NO) ou normalement fermés (NC).
- **Un ressort** : lorsqu'aucun courant ne passe à travers la bobine de l'électroaimant, le ressort tire l'armature de sorte que le circuit ne peut pas être fermé.
- L'ensemble des contacts électriques, à savoir :
Normalement ouvert – connecté lorsque le relai est activé et déconnecté lorsqu'il est inactif.

Normalement fermer – pas connecté lorsque le relais est activé, et connecté quand il est inactif.

- **Le cadre moulé** : qui est le plastique de protection du relais.

Ainsi en se basant sur l'image de la coupe interne d'un relais. Le noyau de fer est entouré par la bobine de commande. Comme le montre la figure, l'alimentation est fournie à l'électroaimant via l'interrupteur de commande et les contacts de charge. Lorsque le courant commence à circuler dans la bobine de commande, l'électroaimant commence à s'activer, améliorant ainsi le champ magnétique. Par conséquent, le bras de contact supérieur commence à être attiré par le bras fixe inférieur, fermant ainsi les contacts, entraînant un court-circuit de la charge. En revanche, si le relais a été mis hors tension lors de la fermeture du contact, le contact se déplace en sens inverse et s'ouvre. Une fois le courant de bobine coupé, l'armature mobile reviendra à sa position initiale par la force à l'aide du ressort.

iv. **Branchement et Commande**

Sur le relais, nous avons les broches suivantes qui se connectent comme suit [18]:

- GND vers le pin GND de la carte Arduino
- VCC vers source extérieure de 12V pour alimenter le relais

En ce qui concerne l'objet à commander nous aurons trois contacts électriques principales à savoir :

- Normalement fermé (NF)
- Normalement ouvert (NO)
- La Borne commune. (C)

Selon le branchement nous aurons des résultats différents [4] :

- **Contact normalement ouvert** :

Dans la configuration normalement ouverte, lorsque le relais reçoit la commande HIGH, le commutateur se ferme et permet au courant de circuler de la borne NO vers la borne C et là d'assurer la continuité du courant de NO à C. La commande LOW désactive le relais et

empêche le courant de passer. Donc, si nous souhaitons allumer le relai avec la commande HIGH, il faut utiliser cette configuration.

➤ **Contact normalement fermé :**

Dans la configuration normalement fermée, la commande HIGH ouvre le commutateur et interrompt la sortie du courant vers l'objet. Par contre la commande LOW ferme l'interrupteur et permet au courant de circuler de la borne B vers la borne NF. Par conséquent, si nous souhaitons que la commande HIGH éteigne le courant de sortie, il faut utiliser cette configuration.

9. Le RTC DS3231

Pour toutes opérations nécessitant de se dérouler en temps réel, nous avons besoin d'un module qui va se charger de cette opération. Et ce module se nomme RTC (Real Time Clock), ce qui donne en français Horloge Temps Réel. Une fois configuré il compte le temps qui passe et de façon autonome et précise grâce à sa pile s'il n'est pas branché.

Ainsi nous avons utilisé, le module RTC DS3231 qui est certes économique, mais qui fournit de très bons résultats.

i. Vue d'ensemble



Figure 25 : RTC DS3231

ii. Fiche technique

Voici la présentation de sa fiche technique [4] :

- Plage de température : -40 à 85 °C ;
- Précision de synchronisation : ± 5 ppm ($\pm 0,432$ seconde / jour)

- La fonction calendrier et l'horloge contient les secondes et les minutes, les heures, les semaines, la date, le mois et l'année et fournit une compensation pour les années bissextiles jusqu'en 2100.
- Sortie : 1 Hz et 32,768 kHz
- Réinitialiser la sortie et le rebond d'entrée du bouton-poussoir
- Haute vitesse (400 kHz), bus série I2C
- Tension d'alimentation : + 3,3 V à + 5,5 V

iii. Branchement

Ce module fonctionnant avec un bus I2C le branchement se fera comme celui de notre écran LCD vers la carte Arduino. Donc nous avons :

- GND vers le pin GND.
- VCC vers le pin 5V.
- SDA vers le pin A4 de la Nano
- SCL vers le pin A5 de la Nano

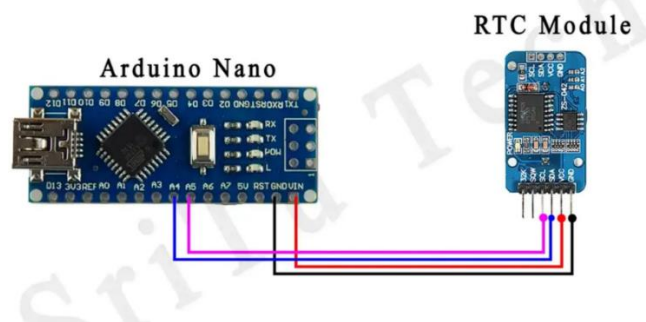


Figure 26 : Branchement DS3231 vers carte Arduino Nano

10. Une LED

Une LED (Light-Emitting Diode) est une diode qui émet de la lumière. Celle-ci peut être de couleur rouge, verte, bleue, jaune. Les caractéristiques des LEDs diffèrent selon la couleur de la lumière émise. Voici un exemple d'une LED branchée :

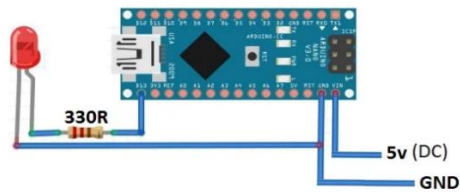


Figure 27 : Branchement d'une LED avec Arduino Nano

Nous avons utilisé les leds pour signaler l'activité de nos différentes phases de fonctionnement mais aussi pour signaler une absence en eau ou coupure de l'électrovanne.

11. Un Buzzer

C'est un transducteur électronique intégré. Il est beaucoup utilisé dans les ordinateurs, imprimantes, copieuses, alarmes, jouets électroniques, etc... pour fournir une signalisation ou une alerte sonore et peut être contrôlé par programme. Voici son branchement :

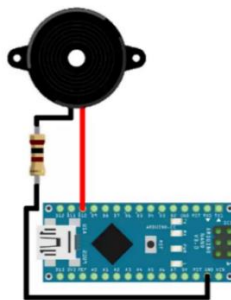


Figure 28 : Branchement d'un Buzzer avec Arduino Nano

Le buzzer émet un son avec différentes tonalités selon les actions de l'utilisateur. Il en existe deux types : le passif et l'actif. L'actif émet un son lorsqu'il est alimenté par une tension continue tandis que le passif se comporte comme un mini hautparleur à alimenter par un signal audio à diffuser [18]. Pour notre système le buzzer sera utile pour alerter si la valeur du compteur s'approche de la valeur seuil (presque égal à 0) et aussi une absence en eau ou coupure de l'électrovanne. Nous avons utilisé un buzzer qui s'alimente avec une tension de 5V et requiert un courant de 10 mA à 30 mA.

2.4.2. La partie Logicielle

Ce paragraphe décrit l'ensemble de la partie non physique de notre projet et comprend :

- Le logiciel Arduino.
- La Base de Données.

a) Le logiciel Arduino

Pour faire fonctionner l'unité de traitement à savoir la carte Arduino Nano et le NodeMCU, nous avons besoin de l'IDE qui permet de programmer les cartes et donc faire fonctionner l'ensemble des composants du compteur. Et cet IDE est fourni par la société Arduino éditrice des cartes qui portent le même nom.

Ce logiciel permet d'écrire des programmes donc le langage de programmation est le C et le C++. Pour faire fonctionner le programme et donc par ricochet la carte et les composants, il est nécessaire de télécharger les bibliothèques de chaque composant et si possible d'avoir les dernières versions [4] [18].



```
Carte_RFID.ino
1 #include <SPI.h>
2 #include <MFRC522.h>
3
4 #define Reset_PIN 9 // Pin pour configurer le reset du rfid
5 #define Liaison_SPI 10 // Pin pour la liaison SPI
6
7 MFRC522 mfrc522(Liaison_SPI, Reset_PIN);
8 MFRC522::MIFARE_Key key;
9
10
11 void setup() {
12   Serial.begin(9600); // Initialisation la connexion avec le moniteur série
13   while (!Serial); // Tant que le moniteur série est ouvert on ne fait rien
14   SPI.begin(); // Initialisation de la liaison SPI
15   mfrc522.PCD_Init(); // Initialisation de la carte rfid.
16   Serial.println(F("Approchez votre carte du lecteur !"));
17 }
18
19 void loop() {
20
21   // On regarde s'il y a de nouvelles cartes
22   if ( ! mfrc522.PICC_IsNewCardPresent() || ! mfrc522.PICC_ReadCardSerial() ) {
23     delay(50);
24     return;
25   }
26
27   // La nouvelle carte est maintenant sélectionnée
28 }
```

Figure 29 : Interface du logiciel Arduino en version 2.3.3

b) La base de données

Pour la sauvegarde des données sur une base de données, il sera nécessaire d'utiliser un service de gestion de base de données. La base de données permettra de gérer les clients avec le statut (actif/inactif) de leur compteur et de sauvegarder certaines données qui sont :

- ✓ L'état de consommation totale du client.
 - ✓ L'état de consommation restante du volume d'eau acheté par le client
 - ✓ La période de l'activation effective de son compte en eau au niveau du compteur.
- ❖ Pour le développement des applications web et mobile, nous avons opté pour un seul langage de programmation essentiel à la réalisation de notre système : **JavaScript**. Grâce à sa polyvalence, JavaScript nous a permis de concevoir une solution unifiée et performante, adaptée aussi bien au web qu'aux plateformes mobiles.
- Le JavaScript est un langage de script principalement employé dans les pages web interactives. Ce langage nous a permis au niveau serveur d'utiliser également un Framework **Node Js** qui est une bibliothèque de développement avec comme base des données Mongo DB.
- Ce langage nous a servi pour le développement de l'application Android grâce à un Framework **React Native** (une bibliothèque de développement) qui permet de créer des Applications mobiles et un Framework **React** pour le développement de l'application Web.
- ❖ Pour l'environnement de développement nous avons utilisés le VS Code (Visual Studio Code) pour la réalisation du site Web et l'application mobile, il est conçu pour être polyvalent et adaptable aux besoins des développeurs quelque que soit leur plateforme ou leur langage de programmation [20].

A noter que pour la partie électronique, on a utilisé le logiciel **Proteus** qui permet d'effectuer une mesure et en évaluer les résultats de façon autonome. Les menus faciles à comprendre et les routines automatiques rendent cet outil facile à utiliser tout en permettant des analyses plus poussées [18]. Ce logiciel nous a permis aussi de faire une simulation surtout pour une partie matérielle de notre système. Pour la conception de l'enveloppe de notre compteur, nous avons choisi le logiciel **FreeCAD**. Ce logiciel nous a permis de créer le boîtier dans lequel seront

logés les différents composants matériels de notre système, garantissant ainsi une conception précise et fonctionnelle de l'ensemble.

2.5. Conclusion partielle

Dans cette partie nous avons présenté les différentes composantes qui ont contribué à la mise en place de notre système. Nous avons ensuite détaillé l'ensemble de l'environnement de développement nécessaire pour développer les applications web et Android de notre projet. Cela inclut les outils, les frameworks et les technologies utilisés pour garantir une intégration fluide et une performance optimale des deux plateformes.

Chapitre 3. RESULTATS ET SIMULATIONS

Après la phase de conception et d'implémentation, qui a permis de détailler le fonctionnement, les fonctionnalités et les composants nécessaires à la réalisation de notre compteur, ce chapitre se concentrera sur la présentation de la maquette physique et des différentes étapes de son fonctionnement. Nous expliquerons également une partie du code liée à certaines actions spécifiques de ce compteur, afin de montrer comment les fonctions sont exécutées et intégrées dans le système global.

3.1. Le compteur et le Scanneur en action

Nous allons présenter, le comportement du compteur lors de certaines phases de son fonctionnement et aussi le scanneur pour recharger la carte RFID.

3.1.1. La phase de démarrage.

Lors de la phase de connexion, le compteur cherche à se connecter au Wi-Fi où il est configuré et affiche un message sur l'écran : « **CONNEXION...** », jusqu'à la fin de cette phase.



Figure 30 : Phase de connexion Compteur

3.1.2. Compte en eau vide

Au démarrage, par défaut, l'électrovanne est fermée. Un message est alors affiché sur l'écran LCD pour informer l'utilisateur que l'eau est actuellement indisponible ou qu'il n'y a pas de volume d'eau disponible pour le moment. Ce message sert à signaler l'état initial du système avant toute action de recharge ou d'activation. D'autre part, lorsqu'un compte est vide, une LED rouge s'allume accompagnée d'un buzzer pour signaler cette situation (sachant que l'utilisateur a la possibilité d'arrêter le buzzer avec l'application mobile pour éviter le dérangement). Une fois que le compteur est rechargé, la LED rouge s'éteint et le buzzer cesse de sonner, indiquant ainsi que le compte est désormais actif et prêt à fonctionner normalement.



Figure 31 : Etat du Compteur lorsque le compte en volume d'eau est vide

3.1.3. Présentation du Scanneur et son fonctionnement

Le scanneur sera essentiel pour la gestion de la recherche et de l'identification de la carte. Il permettra de lire rapidement les informations contenues dans la carte RFID, facilitant ainsi le processus de vérification et d'activation du compte, tout en assurant une communication fluide entre le client et le compteur. Il est composé du module NodeMCU, du lecteur RFID, de 2 leds pour la signalisation et fonctionne sous **3.6 V**.

Après l'alimentation du scanneur :

- ✓ Une Led rouge s'allume par défaut montrant l'absence de la carte dans le scanneur,
- ✓ Une Led verte clignote pour montrer la présence d'une carte dans le scanneur.

Une fois la carte est détectée, le scanner donne la possibilité de créer un nouveau client (c'est-à-dire associer une carte RFID à un compteur) et d'effectuer un paiement au niveau de l'application Web. Pour le paiement qui s'effectue à la compagnie des eaux ou chez un revendeur, on scanne la carte et une fois celle-ci détectée on entre le montant voulu en franc congolais. Si la carte n'est pas détectée l'application Web donne la possibilité de rescanner cette dernière (même chose pour la création d'un nouveau client).

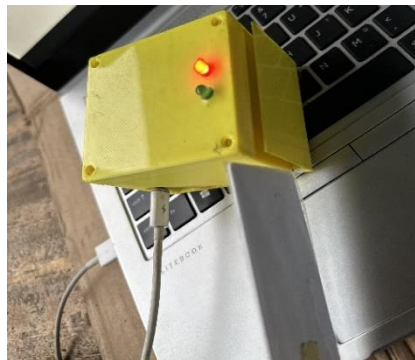


Figure 32 : Présentation du Scanneur sans la carte RFID



Figure 33 : Présentation du Scanneur avec la carte RFID

3.1.4. La phase de recharge du compteur

Pour cette phase, la LED bleue sert à indiquer la présence de la carte RFID dans le compteur, avec deux comportements observables du système :

- ✓ **Lorsque la carte RFID n'est pas identifiée par le compteur** : La LED bleue clignote ou reste allumée de manière spécifique, signalant que la carte utilisée pour la recharge

n'a pas été reconnue par le système. Cela indique que la carte n'est pas liée à un compte valide.

- ✓ **Lorsque la carte RFID est identifiée par le compteur** : La LED bleue reste allumée de manière stable, indiquant que la carte a été correctement identifiée et que le système peut procéder à la recharge du compte en fonction des informations associées à cette carte.

Ces comportements permettent à l'utilisateur de savoir instantanément si la carte est valide ou non pour effectuer une recharge.

Dans le premier cas, nous avons un message « **CARTE INVALIDE** » qui est affiché sur l'écran et dans le second cas, les messages relatifs à la recharge du compte sont affichés.



Figure 34 : Cas d'une carte non identifié



Figure 35 : Cas d'une carte identifié et configuré pour le compteur

3.1.5. La phase de fonctionnement

Lorsque le compteur est chargé en volume d'eau, nous avons sur l'écran deux messages affichant la quantité d'eau restant dans le compteur et la Led verte est allumée pour signaler que le compte est bien actif.



Figure 36 : Le compteur en fonctionnement normal

3.2. Les captures des notifications

Lors du fonctionnement du compteur, le client reçoit certaines informations par le canal des API. Nous allons présenter certaines notifications comme :

○ La phase de recharge du compte

Lors de la recharge du compte le client reçoit une notification dans l'application mobile l'informant du volume d'eau qu'il vient de mettre dans son compte (voir la **Figure 37**).

○ Connaitre son état de consommation

Pour cela, il suffit de regarder sur l'écran LCD présent sur le compteur d'eau ou le vérifier via l'application mobile

○ La phase de recharge du compte avant la fin de sa précédente recharge.

Après avoir rechargé son compte avec un volume d'eau supplémentaire avant la fin de sa précédente recharge, le client reçoit une notification via l'application mobile. Cette notification l'informe qu'une nouvelle recharge en volume d'eau a été effectuée avec succès, lui permettant ainsi de suivre en temps réel l'état de son compte et de sa consommation d'eau.

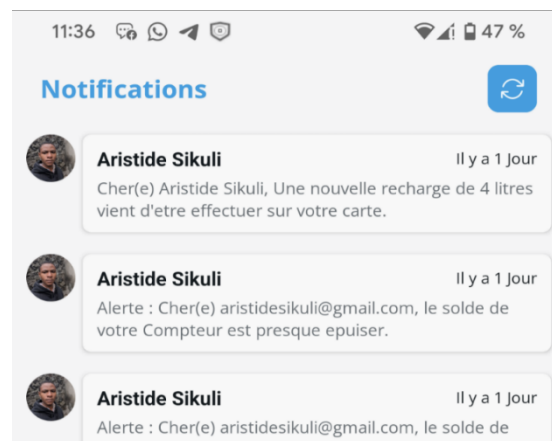


Figure 37 : Captures d'écran des notifications reçus par le client

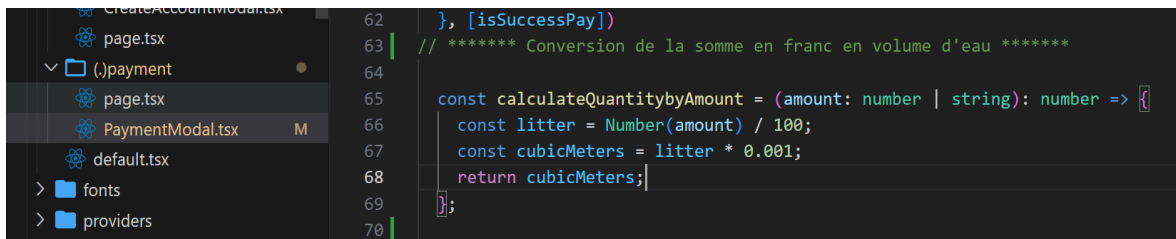
3.3. Les codes de fonctionnement

Dans le cadre de la phase de réalisation de notre projet, il nous est apparu nécessaire de présenter la programmation de certaines fonctionnalités importantes de notre maquette, en présentant les parties du code qui permettent la réalisation de ces actions. C'est ainsi que notre choix s'est porté sur :

- ✓ La conversion de la somme d'argent en volume d'eau.
- ✓ La réception d'une notification rappelant que le volume d'eau restant dans le compteur du client est presque 0.
- ✓ L'affichage de l'état de consommations d'eau sur l'écran du compteur.
- ✓ La fermeture automatique de l'électrovanne à la fin de la consommation du volume d'eau acheté.
- ✓ L'envoi de l'état de consommation sur le serveur.

3.3.1. Conversion de la somme d'argent en volume d'eau

Pour ce faire, la variable contenant la somme payée par le client en franc est en String (une chaîne de caractères), nous avons une conversion en une variable **amount** de type **number** | **String**. Cette dernière variable subit une conversion en valeur numérique grâce à la fonction **calculateQuantityAmount()** et on récupère cette valeur pour la transformer en litres et en mètres cubes afin de l'utiliser dans notre système.

The image shows a code editor with a dark theme. On the left, a file explorer shows a project structure with folders 'fonts' and 'providers', and files 'CreateAccountModal.tsx', 'page.tsx', 'payment', 'PaymentModal.tsx', and 'default.tsx'. The main editor area shows TypeScript code for a function named 'calculateQuantitybyAmount'. The code is as follows:

```
62 }, [isSuccessPay])
63 // ***** Conversion de la somme en franc en volume d'eau *****
64
65 const calculateQuantitybyAmount = (amount: number | string): number => {
66   const litter = Number(amount) / 100;
67   const cubicMeters = litter * 0.001;
68   return cubicMeters;
69 };
```

Figure 38 : Extrait du code Relatif à la conversion de la somme d'argent de la carte RFID en volume d'eau

Après la lecture de la carte RFID dans le compteur, le compteur vérifie si l'UID (Identification unique) de la carte correspond à celui qui lui est configuré au niveau du système ; si oui il passe à l'étape de vérifié s'il y a eu un paiement qui est lié à cette carte et effectue le rechargement du volume d'eau achetés, dans le cas contraire rien ne s'exécute.

3.3.2. Affichage de l'état de consommations d'eau sur l'écran du compteur

Pour afficher les états de consommation de l'eau à savoir le volume d'eau total consommé par le compteur et le volume d'eau restant dans le compte, on réinitialise l'écran LCD, on place sur la première ligne le texte fixe « **Quant** » (Quantity) suivi de la valeur correspondante en litres. Et la seconde ligne c'est l'heure et la date (obtenu à l'aide du module RTC).

```

    lcd_water.setCursor(1, 0);
    lcd_water.print("Quant : ");
    lcd_water.setCursor(10, 0);
    lcd_water.print(water, 1);
    lcd_water.print(" L");

    //affichage du temps et de la date
    lcd_water.setCursor(0, 1);
    lcd_water.print(decode_dow(dateTime.dow));
    lcd_water.print(" ");
}

```

Figure 39 : Extrait du code relatif à l'affichage sur l'écran LCD des états de consommation

3.3.3. Fermeture automatique de l'électrovanne à la fin de la consommation du volume d'eau acheté

Dès que le volume d'eau restant dans le compteur est presque nul (environ 1Litre), nous avons le relai qui est mis en position « LOW » enclenchant la fermeture de l'électrovanne bien que le système du compteur donne toujours la possibilité d'ajouter un nouveau volume d'eau par l'activation du lecteur RFID. Nous avons une notification qui est envoyé à l'utilisateur l'informant que le volume d'eau restant dans le compteur est presque nul.

```

if(valueCounter > 0){
    const NewCounterValue = Counterdatas.TotalPaymentValue - (Counterdatas.TotalCounterValue + valueCounter);
    console.log("no Error");
    modelOfCounter.updateOne({userId:user._id},{
        $set:{
            TotalCounterValue:Counterdatas.TotalCounterValue + valueCounter,
            counterValue:NewCounterValue,
            isActive: NewCounterValue > 0.0012
        }
    }).then(()=>{
        console.log("updating Counter datas ");
        if((NewCounterValue < 0.0008 ) && Counterdatas.newPayment){
            generate_insufficientDatas_Notification( Counterdatas.userId, Counterdatas._id, user.email); // create notification
        }
    });
}

```

Figure 40 : Extrait du code portant sur la fermeture automatique de l'électrovanne à la fin de la consommation du volume d'eau acheté

3.3.4. Envoi de l'état de consommation sur le serveur

Pour envoyer les états de consommation en ligne, il faut d'abord savoir le temps à laquelle on doit envoyer les données et nous avons choisis de le faire toutes le 6 secondes.

Ainsi la fonction **readCounterDatas()** est appelée pour lire les données du débitmètre, et après une connexion HTTPS est initiée vers une URL (Localisateur uniforme des ressources) qu'on a construite en combinant divers éléments dont une chaîne de base `"/Esp8266water"`, le volume mesuré par le débitmètre (**Sensor.getVolume()**) et l'identifiant unique de la carte (**UID_card**) via `« https.begin(client, host + url) »`. Après la connexion réussie, une requête HTTP GET est envoyée.

Alors le programme vérifie si le retour du serveur (le code de statut HTTP) est positif (indiquant un succès), si c'est le cas la fonction **resetCounterDatas()** est appelée pour réinitialiser les données envoyées par compteur et ensuite s'applique la commande `« https.end() »` pour fermer la connexion.

```
// read datas
readCounterDatas();// read sensor datas
String url = "/Esp8266?water=" + String(Sensor.getVolume(),1)+"&UID_card="+UID_card;

if (https.begin(client, host + url)) {
    int code = https.GET();
    if (code > 0) {
        resetCounterDatas(); // reset counter datas
        JsonCovertDatas(https.getString());
    }
    https.end();
}
```

Figure 41 : Envoi de l'état de consommation sur le serveur

3.4. Le câblage

Certes le compteur fonctionne, mais il est nécessaire de présenter l'intérieur de notre compteur afin de voir comment tous les éléments mis bout à bout sont associés.

C'est ainsi que nous présentons le câblage à l'intérieur de notre boîtier en premier et en second, le schéma électronique du compteur et du scanner.

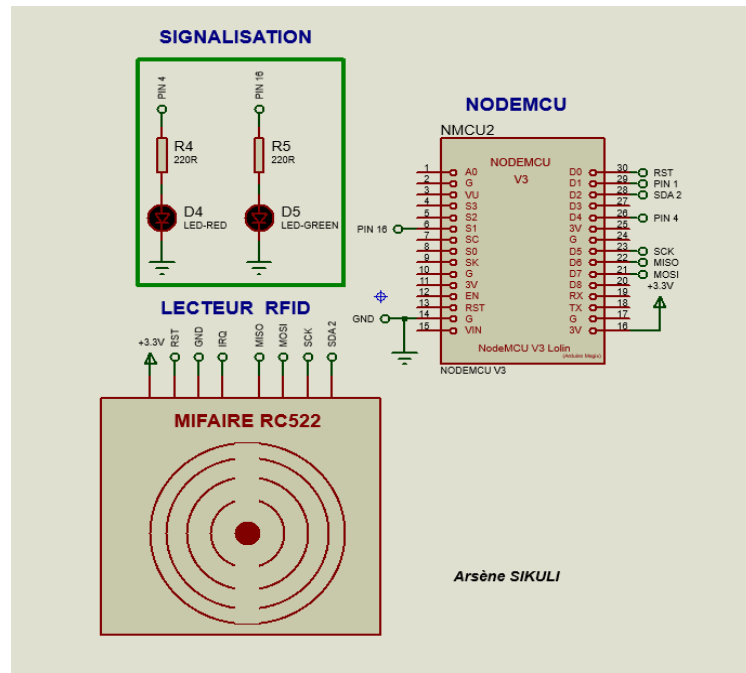


Figure 44 : Schéma de câblage du Scanner

3.5. Application Web

Pour arriver à faire la télésurveillance des compteurs d'eau intelligents prépayés, l'utilisateur concerné (Compagnie des eaux) doit utiliser l'application Web que nous avons développée. Cette application est accessible en ligne via un client Web qui communique avec le serveur central. Nous avons développé une application responsive (qui s'adapte automatiquement à l'écran de l'appareil qui la consulte) pour permettre aux utilisateurs d'utiliser n'importe quel type d'appareil (Ordinateur, Tablette et Smartphone).

Nous présentons dans les pages qui suivent quelques captures de ladite application.

❖ Page d'authentification

Cette interface constitue le point d'entrée de notre application. Elle permet aux utilisateurs du système de se connecter en utilisant un login et un mot de passe.

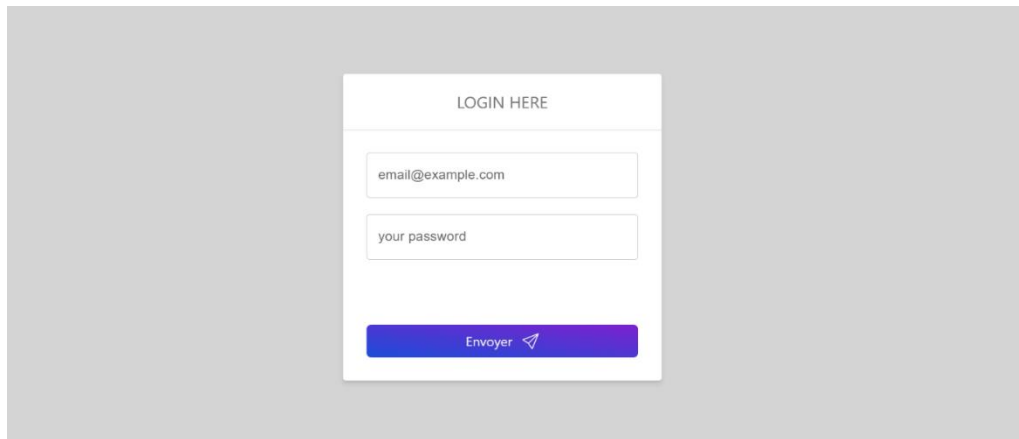


Figure 45 : Page d'authentification

❖ Page d'accueil Administrateur/Agent et Gestion des Clients

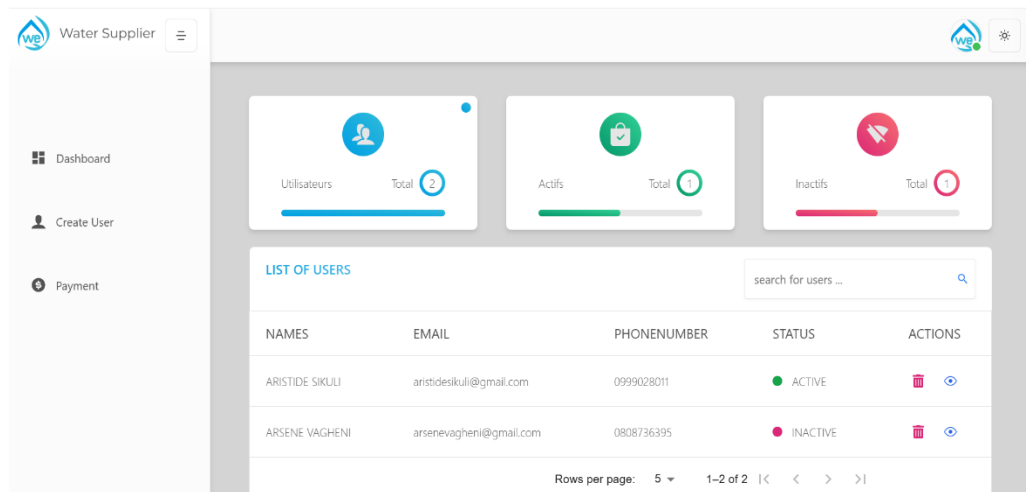


Figure 46 : Page d'accueil et Centre de contrôle

❖ Création d'un nouveau Client avec sa carte RFID

Cette page est accessible une fois que le scanner a détecté une carte RFID

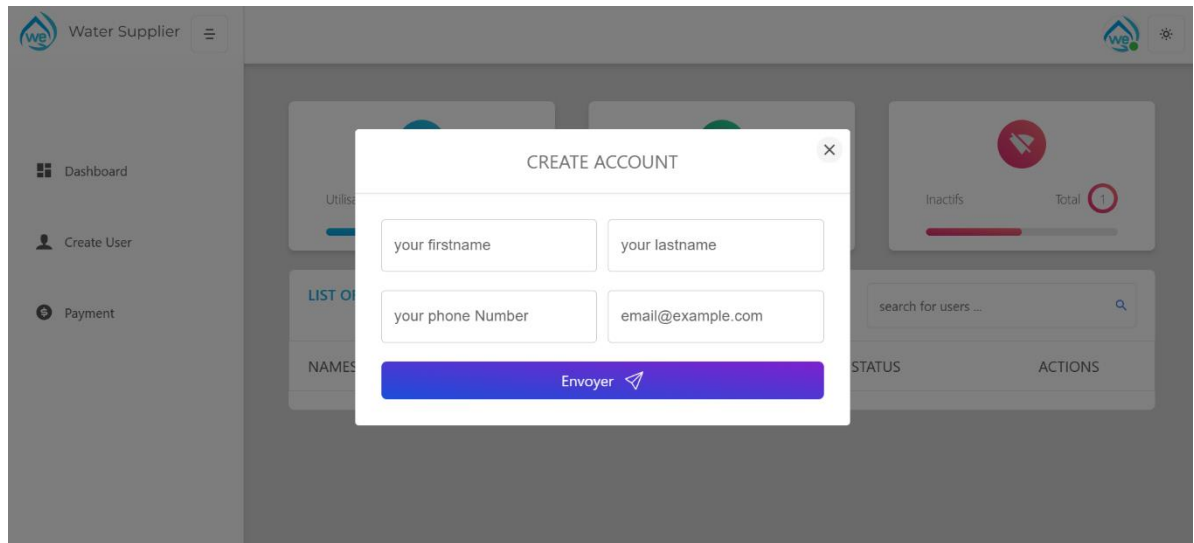


Figure 47 : Formulaire d'enregistrement d'un client avec sa carte RFID

❖ Gestion du profil d'un Client

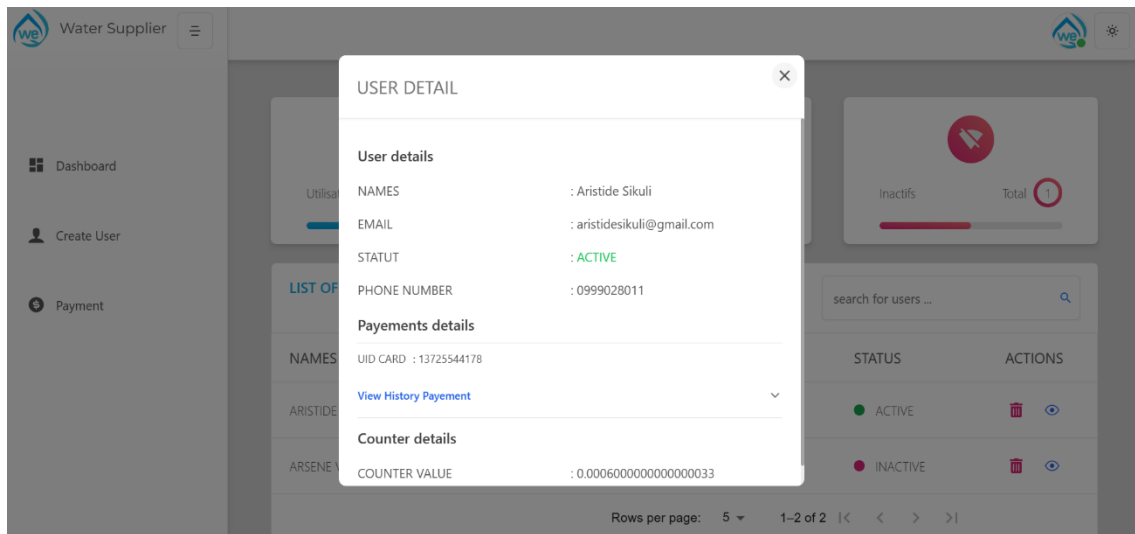


Figure 48 : Page de gestion du profil d'utilisateur

❖ Enregistrement d'un nouveau paiement

On parvient à accéder à cette page de paiement une fois que la carte RFID a été détectée par le scanner.

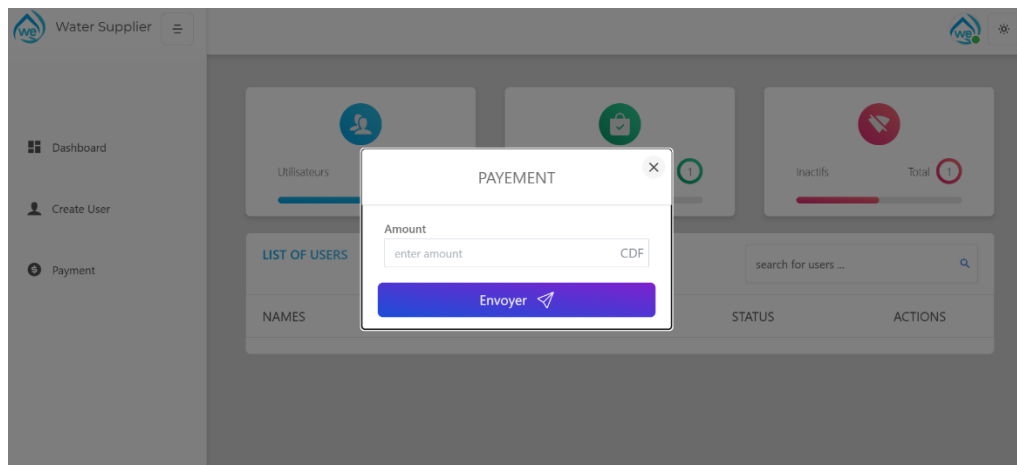


Figure 49 : Page d'enregistrement d'un paiement

3.6. Application Android

Comme nous l'avons dit dans les lignes précédentes, le système doit permettre au client d'accéder à ses données de consommation à distance à l'aide d'une application Android. Nous avons développé une application qui permet à l'utilisateur (client) de voir aussi en temps réel l'état de sa consommation, d'agir à l'aide d'un bouton sur celle-ci (stopper ou mettre en marche le compteur) et de recevoir des notifications en cas d'épuisement ou de recharge en volume d'eau.

Les images ci-dessous présentent quelques captures d'écran de cette application Android.

❖ Activité de login

Cette interface est l'activité du démarrage de l'application. Elle permet au client de se connecter avant de pouvoir utiliser l'application.



Figure 50 : Interface de connexion du client

❖ **Activité principale de monitoring**

Cette activité permet à l'utilisateur de visualiser en temps réel l'état de sa consommation et d'agir dessus. Les données sont également envoyées en background vers le serveur distant.

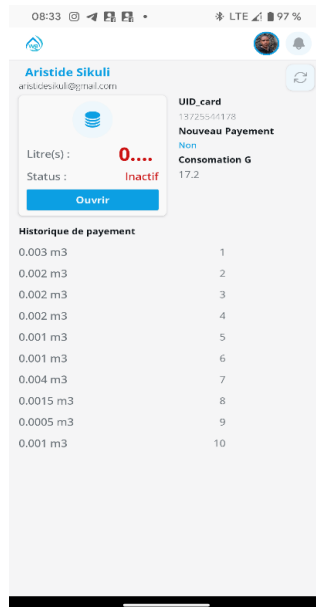


Figure 51 : Monitoring et du contrôle par Client

❖ **Activité de compte utilisateur**

Cette activité permet à l'utilisateur de visualiser son profil.

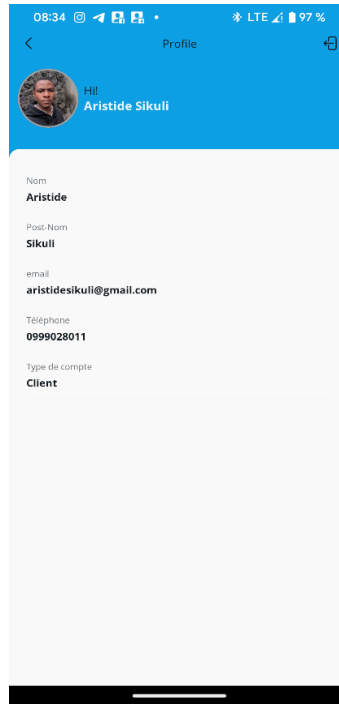


Figure 52 : Interface de compte utilisateur

3.7. Estimation du coût du matériel

Après avoir présenté le choix du matériel, il est important de présenter un tableau récapitulatif du coût total des composants. Les prix indiqués sont exprimés en dollars américains et ont été obtenus à partir de divers fournisseurs, ce qui signifie qu'ils peuvent varier en fonction du lieu d'achat. Il est important de noter que ce bilan financier ne prend pas en compte les dépenses liées à la main-d'œuvre, ni d'autres frais supplémentaires tels que ceux associés au développement et à la gestion du site web.

Tableau 4 : Prix des matériels utilisés

MATERIELS	QUANTITES	PRIX UNITAIRE	PRIX TOTAL (\$)
CARTE ARDUINO NANO	1	8,2	8,2

NodeMCU	2	13,5	27
DEBITMETRE YF-S201	1	8	8
RFID RC522+CARTE	2	7	14
LCD 16×2 I2c	1	8	8
RELAIS	1	5	5
RTC DS3231	1	3,5	3,5
ELCTROVANNE 12V	1	13	13
LEDS	5	0,05	0,25
BUZZER	1	1,5	1,5
RESISTANCES (Boite)	1	12	12
BOITIER IMPRIMER	2	20	40
CHARGEUR 12V	1	10	10
CONNECTEURS PLOMBERIE	5	1,5	7,5
FILS CONDUCTEUR (rouleau)	2	5	10
TOTAL			167,95\$

Ainsi après avoir conçu et réalisé une maquette, il est maintenant nécessaire d'éprouver le fruit de notre recherche dans des conditions proches d'une implémentation concrète du compteur. Les fonctions liées au compteur, telles que l'affichage des informations sur l'écran, le paiement via carte RFID, et l'envoi des données de consommation vers un serveur ou directement au client, ne nécessitent pas nécessairement le passage d'eau pour fonctionner. Ainsi, nos tests se dérouleront dans un environnement simulé, qui reproduira le raccordement du compteur à un réseau de distribution d'eau. Nous simulerons également le passage d'un fluide, non pas de l'eau, mais de l'air, afin de valider le bon fonctionnement du système sans utiliser de fluide réel.

3.8. Passage de l'air

Ce cas permet d'observer le comportement du compteur lorsqu'il est soumis au passage de l'air, au lieu de l'eau, comme cela peut se produire après une coupure d'eau, avant que l'eau ne revienne dans les canalisations. Ce phénomène, connu sous le nom de **cavitation** (formation de bulles d'air), peut affecter le bon fonctionnement du compteur. En simulant ce passage d'air, nous serons en mesure de tester la réactivité du système et sa capacité à gérer ce type de situation sans compromettre la précision de la mesure ou l'intégrité du matériel.

3.8.1. Condition du test

Pour simuler l'envoi de l'air, nous avons utilisé un souffle provenant de notre bouche, avec lequel nous envoyons de l'air pendant cinq secondes. L'électrovanne, lorsqu'elle est activée, nécessite une pression minimale de 0,8 bar pour permettre le passage d'un fluide. Dans cette simulation, nous avons activé dans notre système un volume d'eau de 30 litres, afin de tester le bon fonctionnement du compteur et de vérifier si celui-ci réagit correctement à l'air soufflé, tout en maintenant la précision et la fiabilité des mesures, comme cela serait attendu avec de l'eau réelle.

3.8.2. Test et résultats

Pour cette partie, il est important de noter que le débitage de l'air par le compteur n'était pas toujours parfait. En effet, l'air envoyé ne traversait pas systématiquement le compteur de manière optimale. Nous avons réalisé quatre tests, chacun d'une durée moyenne de cinq secondes, afin d'évaluer la réactivité du système. Ces tests ont permis de mettre en évidence certaines imperfections dans le passage de l'air à travers le compteur, ce qui pourrait affecter la précision des mesures dans des conditions réelles, notamment en cas de cavitation ou d'irrégularités dans le fluide circulant.

Tableau 5 : Volume d'air du souffle humain mesuré par le compteur

Test	Durée moyenne (Seconde)	Volume d'air mesuré (Litres)
1	5	0,08
2	5	1,05
3	5	0,09
4	5	0,72

3.8.3. Analyses des résultats

Au vu des études, nous constatons qu'un fluide comme l'air peut être bien comptabilisé par notre compteur et nous observons des valeurs un peu disparates, du fait de la non-maîtrise de notre part du flux d'air envoyé et de la mesure du débitmètre qui a une précision de $\pm 10\%$.

Nous pouvons conclure, que notre système peut être bien utilisé dans la mesure de fluides comme l'air et le gaz.

3.9. Passage de l'eau

Ce cas permet de voir comment se comporte le compteur au moment du passage de l'eau dans le compteur

3.9.1. Condition du test

Du fait que notre maquette nécessite une source d'alimentation venant du courant électrique, nous n'avons pas pu raccorder notre compteur à la place d'un compteur de la REGIDESO. L'autre solution aurait été donc se raccorder à un robinet d'une maison, mais malheureusement à cause de la contrainte de notre maquette dû à son alimentation électrique n'étant pas autonome, cette solution n'est pas pratique vu que les positions des robinets d'eau dans les maisons comme dans les parcelles ne sont pas proches des sources d'alimentation électriques. C'est ainsi que pour le passage de l'eau nous avons optés pour l'utilisation d'un seau d'eau

avec un robinet intégré. Le seau que nous avons utilisé à un débit de 0,445 Litre/Seconde à son maximum s'il s'écroule sous l'effet de la gravité, cette mesure a été obtenue par une mesure préalable de notre part. Et le tuyau de raccordement mesure 16 mm de diamètre et 0,8 Mètre de longueur.

3.9.2. Test et résultats

Avec toutes les conditions citées plus haut, lors de l'alimentation en eau de notre compteur nous remarquons une sortie de l'eau du système pas très suffisant, malgré notre remplissage. Cette situation ne trouvant son explication par un manque de pression suffisante, mais nous observons un débitage effectif du compteur et qui nous permet d'avoir un débitage normal du compteur.

Nous avons effectué cinq tests en injectant 0,5 Litres d'eau à chaque fois et nous avons obtenu les résultats suivants :

Tableau 6 : Volume d'eau mesuré

Test	Volume d'eau mesuré (Litres)
1	0,45
2	0,49
3	0,51
4	0,47
5	0,46

3.9.3. Analyses des résultats

Malgré les conditions peu idéales pour mettre à l'épreuve notre compteur, nous observons que le débitage de l'eau est bien faisable, mais nécessite une pression minimale de l'eau qui devra correspondre à celle fourni par les constructeurs du débitmètre et des électrovannes.

Et pour les mesures du volume d'eau, nous remarquons que la précision du débitmètre qui est de $\pm 10\%$, pourrait causer de légères pertes ou gains, au client ou à la compagnie des eaux.

Ainsi donc nous préconisons, un étalonnage strict du débitmètre au début de l'installation et aussi après la période de fonctionnement optimal du débitmètre.

Concernant le souci de pression, nous pouvons déduire que le choix de l'électrovanne en ait la source principale, vu qu'il demande en entrée une pression minimale de 0,8 bar, ce qui est assez conséquent, donc l'idéal serait de se tourner vers une électrovanne dont la pression minimale de fonctionnement serait quasiment nulle comme pour le débitmètre YF-S201.

Pour le souci observé lors de l'emplacement de notre compteur, nous recommandons de le placer sous une toiture pour le protéger de l'environnement hostile de la pluie et du soleil. Si l'alimentation se fait à partir d'une source électrique provenant de la compagnie d'électricité, il sera nécessaire de placer dans le compteur une batterie de secours dont la durée sera fournie au client afin de lui permettre de faire des réserves d'eau par exemple et dans une meilleure situation, permettre au compteur de faire une durée de 24 heures selon la capacité de la batterie.

4. Conclusion partielle

Dans cette partie nous avons décrit les différentes phases de fonctionnement de notre système et présenté les différentes interactions entre l'application web et mobile qui ont contribué au bon fonctionnement de ce dernier. A part la présentation du coût des matériels utilisés, nous avons fini par l'analyse de nos résultats.

CONCLUSION GENERALE

Ce travail a porté sur « **la conception et la réalisation d'un compteur d'eau intelligent prépayé** ». C'est dans cette optique que nous avons entrepris une étude sur les différents travaux portant sur les compteurs intelligents, en mettant en évidence certaines limites spécifiques à notre ville, Goma. Tout d'abord, nous avons démontré que l'ergonomie et la convivialité d'un compteur d'eau intelligent prépayé reposent sur l'intégration de technologies telles que le paiement sans contact via RFID/NFC, l'affichage des informations de consommation en temps réel et la connectivité pour une gestion à distance via une application mobile ou web. Ces aspects rendent le système accessible et transparent pour les utilisateurs, leur permettant un meilleur contrôle de leur consommation.

Ensuite, nous avons identifié les technologies clés nécessaires à la mise en œuvre du compteur, notamment un capteur de débit, un microcontrôleur Arduino nano couplé au module Nodemcu (pour la transmission des données au serveur) et l'électrovanne. Ces composants permettent une acquisition fiable des données et une interaction fluide avec le système de gestion de l'eau. Cependant, nos tests ont mis en évidence des limites, notamment liées à la pression minimale requise par certaines électrovannes et aux variations de précision du débitmètre. Des ajustements, comme le choix d'une électrovanne plus adaptée qui est presque de 0 bar et un étalonnage strict du débitmètre, sont ainsi recommandés pour améliorer les performances du système. Concernant les bénéfices pour les utilisateurs, nous avons constaté que le système prépayé permet non seulement une facturation plus équitable, mais aussi une meilleure anticipation et gestion de la consommation d'eau. Il contribue à éviter les gaspillages, responsabilise les consommateurs et facilite le recouvrement des paiements pour les fournisseurs d'eau.

Enfin, les tests effectués, à la fois avec l'air et l'eau, ont permis d'évaluer la fiabilité du système dans des conditions variées. L'expérimentation avec l'air a montré des disparités dans la mesure du débit, dues aux limites de précision du capteur et à la nature du fluide testé. En revanche, les tests avec l'eau ont confirmé que le système est opérationnel sous certaines conditions de pression et d'installation.

Ainsi, notre compteur d'eau intelligent prépayé constitue une solution innovante et prometteuse pour une gestion plus efficace des ressources en eau. Son optimisation nécessitera des ajustements techniques et un déploiement dans un environnement réel pour valider pleinement ses performances.

Après ces résultats qui sont certes intéressants, il faut bien penser à des améliorations dans les travaux de recherches sur la conception et la réalisation de ce compteur suivant les axes futurs qui pourraient être :

- L'ajout d'un système local de gestion interne qui pourrait combler à une absence de connexion Internet ;
- Penser à un système d'alimentation qui permettrait de faire fonctionner le compteur même en absence de l'électricité, soit par une source d'énergie renouvelable du type solaire ;

Bibliographie

- [1] W. Jean, "Communicating Meter," 2017. [Online]. Available: <https://www.encyclopedie-energie.org/compteur-communicant/>. [Accessed Mars 2024].
- [2] G. Wallenborn, "Avis sur les Compteurs Communicants," Parlement Wallon, 2013.
- [3] "Smart Meter," 2015. [Online]. Available <https://www.xylem.com>, <https://www.elester.com>, <https://www.honeywell.com>, <https://www.kamstrup.com>, <https://www.dielh.com/metering/en>, <https://www.sensus.com> <https://www.itron.com/>. [Accessed mars 2024].
- [4] N. N. Wafo, Conception et Réalisation d'un Compteur d'Eau Intelligent Prépayé Douala: Enset, 2020.
- [5] P. Gade, An Intelligent sms Based Prepaid water Metering System, 2018.
- [6] S. bouzefrane, "Les Cartes à puces," 2013.
- [7] "RS France," [Online]. Available: <https://fr.rs-online.com/web/p/arduino/6961667?srsId=AfmBOoqykM7ymWKsMKezRwzzeXzPEwrSaltLrs8ZmPBrhAfHG3hNO307>. [Accessed Octobre 2025].
- [8] "Components101," [Online]. Available: <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet>. [Accessed Septembre 2024].
- [9] DEVKIT, "NodeMCU ESP8266 ESP-12E Catalogue/ Documentation," juillet 2017 [Online]. Available: <https://ardustore.dk/error/Manuel%20-%20NodeMCU%20Lua.pdf> [Accessed Septembre 2024].
- [10] "GOTRONIC," [Online]. Available: <https://www.gotronic.fr/pj2-sbc-rfid-rc522-fr-1439.pdf>. [Accessed Septembre 2024].
- [11] 20 Novembre 2021. [Online]. Available <https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DwKuTvRgGuVU&psig=AOvVaw38GoFzQ5VRH640JiC60YxU&ust=17365>

- 97152137000&source=images&cd=vfe&opi=89978449&ved=0CBcQjhxqFwoTCLCpoNqO64oDFQAAAAAdAAAAABAE. [Accessed Octobre 2024].
- [12] O. BARAKA, Cours Initiation à la programmation des systèmes embarqués, Goma: DRC, 2024.
- [13] J. M. Doudoux, "Developpons en Java par J.M Doudoux," [Online]. Available https://www.jmdoudoux.fr/acceuil_java.htm. [Accessed Mars 2024].
- [14] JUnit.org. [Online]. Available: <http://www.junit.org>. [Accessed 2 Janvier 2019].
- [15] U. A. R. & K. Rafique, "Modeling of Arduino-based Prepaid Energy Meter Using GSM Technology," *International Journal Of Advance Computer Science and Application (IJACSA)*, pp. 445-449, 2018.
- [16] D. H. R. & P. Shilpa, "Smart Prepaid Energy Meter Using GSM and Arduino," *International Journal of Management, Technology And Engineering*, no. 8(12), pp. 4400-4406, 2012.
- [17] A. K. C. C. G. K. & R. K. Annapurna Mishra, "Prepaid Electricity Meter Using GSM Module," *International Journal Of Advance Research And Development*, no. 3(2), pp. 240-244, 2018.
- [18] G. I. I. & F. I. A. O. Omijeh, "SMS Based Recharge Protocol for Prepaid Energy Billing System," *International Journal Of Engineering Innovation & Research*, no. 1(6), pp. 553-558, 2011.
- [19] Foshan Shunde Zhongjiang Energy Saving Electronics Co., Ltd, "C898+datasheet," [Online]. Available: <https://cdn-shop.adafruit.com/product-files/828/C898+datasheet.pdf> [Accessed Septembre 2024].
- [20] "Circuit Designer," [Online]. Available https://www.google.com/url?sa=i&url=https%3A%2F%2Fdocs.circuitdesigner.com%2Fcomponent%2F4b41d60e-62d6-422b-acdc-49d84b80d903%2Fwater-flow-rate-sensor-yf-401&psig=AOvVaw1M3Olo5DXmCAVQ4K1tXbl_&ust=1736593975141000&source=images&cd=vfe&opi=89978449&ved=0CBc. [Accessed Septembre 2024].

- [21] "Electronicwings," [Online]. Available <https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.electronicwings.com%2Fnodemcu%2Frfid-rc522-interfacing-with-nodemcu&psig=AOvVaw2O1-V-sKhSUSk4eJx3NIba&ust=1736594802819000&source=images&cd=vfe&opi=89978449&ved=0CBcQjhxqFwoTCMCGmfaF64oDFQAAAAAdAAAAABA>. [Accessed Septembre 2024].
- [22] "Micro Planet," [Online]. Available: https://www.microplanet.ma/produit/electrovanne-eau-air-12v-12/?srsltid=AfmBOopBSMKBjYId1iiRPqfeHemLCsAWUAEyIk0vAaUCJDdTCAxzyR7_. [Accessed Novembre 2024].
- [23] "Newbiely," [Online]. Available: <https://newbiely.com/tutorials/esp8266/esp8266-water-liquid-valve>. [Accessed Novembre 2024].
- [24] "Electroschema," [Online]. Available <https://electroschema.blogspot.com/2012/02/schemas-des-relais-et-comment-les.html>. [Accessed Octobre 2024].

Annexes

Extrait du code source Arduino nano

```
void loop() {

    appel_fonction(); //and that call a functions (download a times and read serial monitor)

    if (( millis() - temp_lcd) >= 1500) {

        temp_lcd = millis();

        affichage();          // after one seconde , this function update a datas to screen

        download_time( &dateTime );

    }

    wdt_reset();

}

void appel_fonction() {

    readSerialsDatas();

    control_ElectroVannes();

    control_buzzer();

    control_leds();

}

String getValue(String value, char car, int TabCase) {
```

```
int indexDats = value.indexOf(car);

if (indexDats >= 0) {

    String leftDats = value.substring(0, indexDats);

    String righDats = value.substring(indexDats + 1);

    switch (TabCase) {

        case 0: {

            return leftDats;

            break;

        };

        case 1: {

            return righDats;

            break;

        }

    }

}

else {

    return "null";

}

}
```

```

// Functions decode datas

const StringToBoolean(String valuedatas) {

  if (valuedatas == "1")

  {

    return true;

  }

  else {

    return false;

  };

}

void CachedDatas(String datas) {

  String StringStates = "";

  // subdivide datas;

  water = getValue(datas, '#', 0) != "null" ? getValue(datas, '#', 0).toFloat() : water;

  StringStates = getValue(datas, '#', 1);

  Serial.println(StringStates);

  state = getValue(StringStates, '#', 0) != "null" ? (getValue(StringStates, '#', 0) == "1" ? true :
false) : state;

  Invalidstate = getValue(StringStates, '#', 1) != "null" ? (getValue(StringStates, '#', 1) == "0" ?
true : false) : Invalidstate;

```

```
}  
  
void readSerialsDatas() {  
  
    char car;  
  
    String datas = "";  
  
    boolean availableDatas = true;  
  
    //  
  
    if (espSerial.available() > 0) {  
  
        // reading datas  
  
        do {  
  
            car = espSerial.read();  
  
            if ( car != '\n') {  
  
                datas += car;  
  
            }  
  
            else {  
  
                availableDatas = false;  
  
            }  
  
        }  
  
        while (availableDatas);  
  
    }  
  
}
```

```

if (datas.length() > 0) {

    CachedDatas(datas);

}

}

void control_ElectroVannes() {

    // Appart 1

    if (state) {

        digitalWrite(ELECTROVANNE, HIGH);

    }

    else {

        digitalWrite(ELECTROVANNE, LOW);

    }

};

void control_leds() {

    // Appart 1

    if (!state) {

        if ((millis() - temp_ledRed) >= 1000) {

            temp_ledRed = millis();

            state_ledRed = !state_ledRed;

        }

    }

}

```

```
    digitalWrite(REDD_LED, state_ledRed);

    digitalWrite(GREEN_LED, LOW);

}

}

else {

    digitalWrite(REDD_LED, LOW);

    digitalWrite(GREEN_LED, HIGH);

}

}

void control_buzzer() {

    if (!state) {

        if ((millis() - temp_buzzer) >= 500) {

            temp_buzzer = millis();

            state_Buzzer = !state_Buzzer;

            digitalWrite(BUZZER, state_Buzzer);

        }

    }

    else {

        digitalWrite(BUZZER, LOW);
```

}

}