

Systeme asservi d'abreuvement des vaches laitières

Application web à base d'une plate-forme embarquée

Réalisé par :

EL KANTRI Youssef

Encadré par :

Mr. MANSOURI ANAS

Table of Contents

INTRODUCTION GENERALE :.....	3
CHAPITRE I : CONTEXTE GENERAL DU PROJET :.....	4
Contexte :.....	4
Cahier de charges :.....	4
• Cahier de charges du projet :	4
• Problématique :	4
• Les objectifs :	5
Méthodologie :	5
CHAPITRE II : CONCEPTION ET MODELISATION DU PROJET :	7
Conception du projet :	7
• Diagrammes :	7
• Organigrammes :	8
Architecture générale de projet :	10
Partie serveur :	10
Partie client :	10
Environnement du projet :	11
• Environnement matériel :	11
• Environnement logiciel :	12
CHAPITRE III : REALISATION ET TEST DU PROJET	14
Réalisation :	14
Teste de l'application web :	15
Scenario de test :	15
CONCLUSION GENERALE :	18
REFERENCES:	19

Liste des figure

FIGURE 1: DIAGRAMME DE PERT REPRESENTANT LA PLANIFICATION DES TACHES	6
FIGURE 2: MODELE DE DEVELOPPEMENT DE CYCLE EN V	6
FIGURE 3: DIAGRAMME DE CAS D'UTILISATION	7
FIGURE 4: DIAGRAMME DE SÉQUENCE	8
FIGURE 5: ARCHITECTURE GENERALE DU PROJET	10
FIGURE 6: ENVIRONNEMENT MATERIEL DU PROJET	11
FIGURE 7: ENVIRONNEMENT LOGICIEL DU PROJET	12
FIGURE 8: ARCHITECTURE LINUX	12
FIGURE 9: TECHNOLOGIES UTILISEES DANS LE SYSTEME	14
FIGURE 10: TEST N°1	15
FIGURE 11: TEST N°2	16
FIGURE 12: TEST N°3	16
FIGURE 13: TEST N°4	17

INTRODUCTION GENERALE :

En 70 ans, nous sommes passés d'une agriculture vivrière à une production de masse, dans laquelle la ferme est devenue une exploitation. Le métier change de visage, les professionnels parlent maintenant d'agriculture de précision, de smart agriculture. C'est-à-dire la mise à disposition d'outils numériques permettant de répondre aux enjeux du secteur, notamment de produire mieux avec moins. C'est dans ce contexte à la fois difficile et stimulant que on a opté à développer un prototype d'un système asservi afin de résoudre le problème lié à l'abreuvement des vaches laitières, en ce qui suit on va présenter le contexte du projet, avec l'expression du besoin, voire l'élaboration du cahier de charge, puis la conception et la modélisation de la solution proposée, d'ailleurs sa réalisation, Et finalement la phase de test et validation.

CHAPITRE I : CONTEXTE GENERAL DU PROJET :

Ce chapitre a pour but de faire une analyse de besoin du marché et de proposer une solution afin d'élaborer un cahier de charge, ainsi que la démarche à suivre pour faire réussir ce projet.

Contexte :

Inclus dans le domaine d'agriculture de précision, ce projet a pour but de réaliser un prototype d'un système asservi capable d'automatiser l'abreuvement des vaches laitières en bâtiment, en se basant sur l'architecture client-serveur, solution assez intéressante dont ses principaux atouts sont :

- **Des ressources centralisées** : étant donné que le serveur est au centre du réseau, il peut gérer des ressources communes à tous les utilisateurs, comme par exemple la base de données centralisée, afin d'éviter les problèmes de redondance et de contradiction
- **Une meilleure sécurité** : car le nombre de points d'entrée permettant l'accès aux données est moins important
- **Un réseau évolutif** : grâce à cette architecture il est possible de supprimer ou rajouter des clients sans perturber le fonctionnement du réseau et sans modification majeure

Cahier de charges :

- *Cahier de charges du projet :*
 - L'application web est basée sur l'architecture client-serveur.
 - Le serveur permet de faire l'acquisition des valeurs du niveau d'eau dans le réservoir d'une manière périodique,
 - Le serveur doit stocker les valeurs collectées par les capteurs dans une base de données.
 - L'interface Web permet le contrôle manuel de la pompe.
 - L'interface web permet de superviser l'état du système en temps réel.
 - L'interface web doit afficher des informations sur la consommation quotidienne en eau.
- *Problématique :*

L'eau joue un rôle crucial dans la production laitière des vaches. De même, elle contribue à stabiliser la température, ainsi que plusieurs autres fonctions

de leur corp. Alors que l'alimentation des vaches donne généralement lieu à un suivi régulier, leur consommation d'eau, tant en termes de quantité que de qualité, est souvent négligée. L'eau potable satisfait de 80% à 90% des besoins en eau de la vache laitière, qui consomme de 4 à 4,5 litres d'eau pour chaque litre de lait produit [1]. D'une manière générale, les vaches s'abreuvent de 7 à 12 fois par jour, avec à chaque fois une consommation comprise entre 10 à 20 litres d'eau.

Une distribution en eau inadaptée ou de mauvaise qualité entraîne une baisse de la production laitière des vaches, limite leur capacité de résistance et favorise parfois même l'apparition de maladies.

- **Les objectifs :**

On peut regrouper les objectifs de ce projet dans les quatre points suivants :

- **Avoir un système asservi pour l'abreuvement des vaches laitières sans interaction humaine.**
- Placer un capteur capable de mesurer le niveau d'eau dans le champ permet aux agriculteurs d'établir une carte détaillée du niveau d'eau du réservoir.
- Avoir accès aux données reliée à la consommation hebdomadaire, ce qui permet d'anticiper les tendances pour les jours et semaines à venir.
- Créer de nouvelles opportunités commerciales grâce au temps libéré.

Méthodologie :

Ce projet étant personnel, notre première mission fut de définir nous-mêmes une stratégie prévisionnelle ainsi que les objectifs à atteindre. Bien entendu, cette stratégie a évolué au cours du temps afin de satisfaire nos exigences, mais aussi les contraintes auxquelles nous avons fait face. Nous avons ainsi tissé notre réflexion, avec pour fil conducteur la relation entre l'utilisateur et la firme. J'ai défini le cycle de développement **PERT** on utilisant le software online-visual-paradigm[2] afin d'organiser les différentes phases et tracer une méthodologie à suivre :

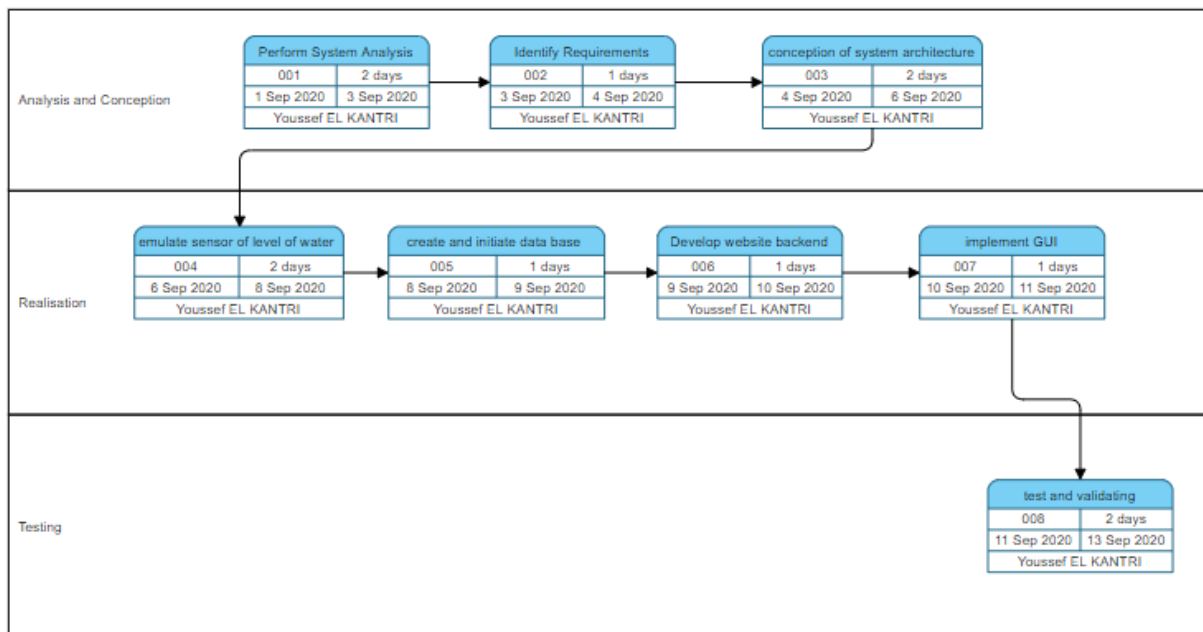


Figure 1: Diagramme de PERT représentant la planification des tâches

Le **cycle en V** se définit comme un modèle de gestion de projet composé d'une phase descendante, puis d'une phase ascendante. Issue du modèle en cascade, ou water-fall model, il implique le même principe de gestion séquentielle et linéaire. Le cycle en V est une illustration des relations entre chaque phase du cycle de vie du développement et la phase de test qui lui est associée. Voici à quoi ressemble le modèle de cycle en V :

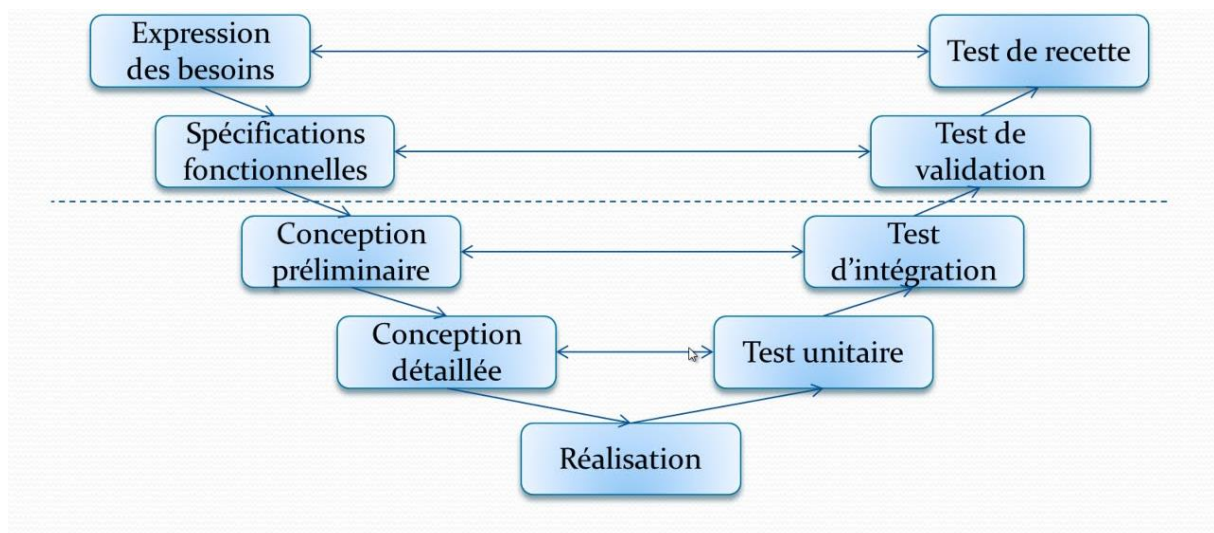


Figure 2: modèle de développement de cycle en V

CHAPITRE II : CONCEPTION ET MODELISATION DU PROJET :

Dans ce chapitre on va faire une description brève de l'environnement technique, ainsi que les méthodes de conception et de modélisation utilisées tout au long du projet

Conception du projet :

À partir des besoins exprimés et les objectifs fixés, il est temps de déterminer environnement technique de ce projet ainsi que son architecture globale voire sa conception détaillée.

- Diagrammes :

Tous ces diagrammes sont réalisés à l'aide du logiciel « LucidChart »[3]

- Diagramme de cas d'utilisation :

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit. Il scinde la fonctionnalité du système en unités cohérentes, les cas d'utilisation, ayant un sens pour les acteurs. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique

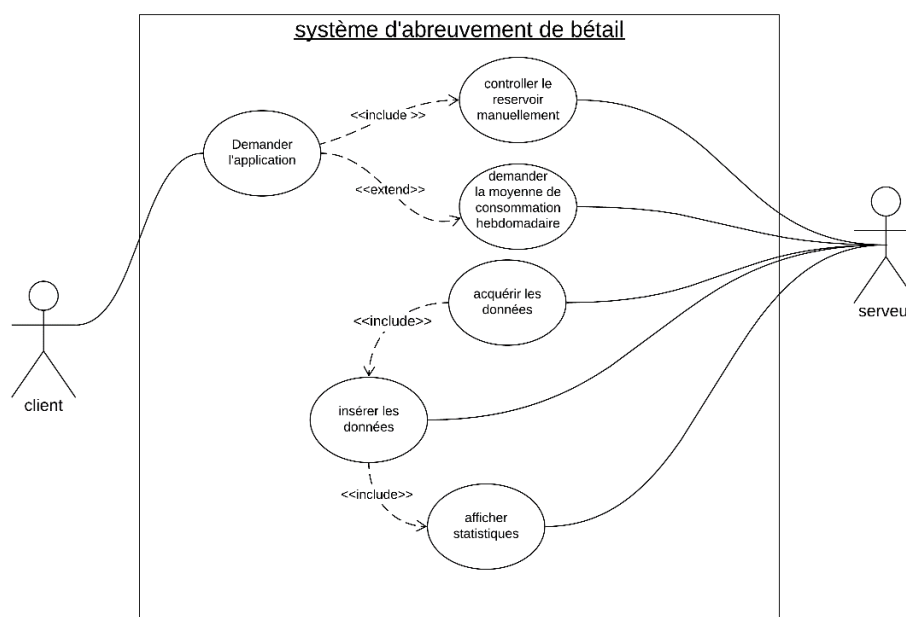


Figure 3: diagramme de cas d'utilisation

- Diagramme de séquences :

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language.

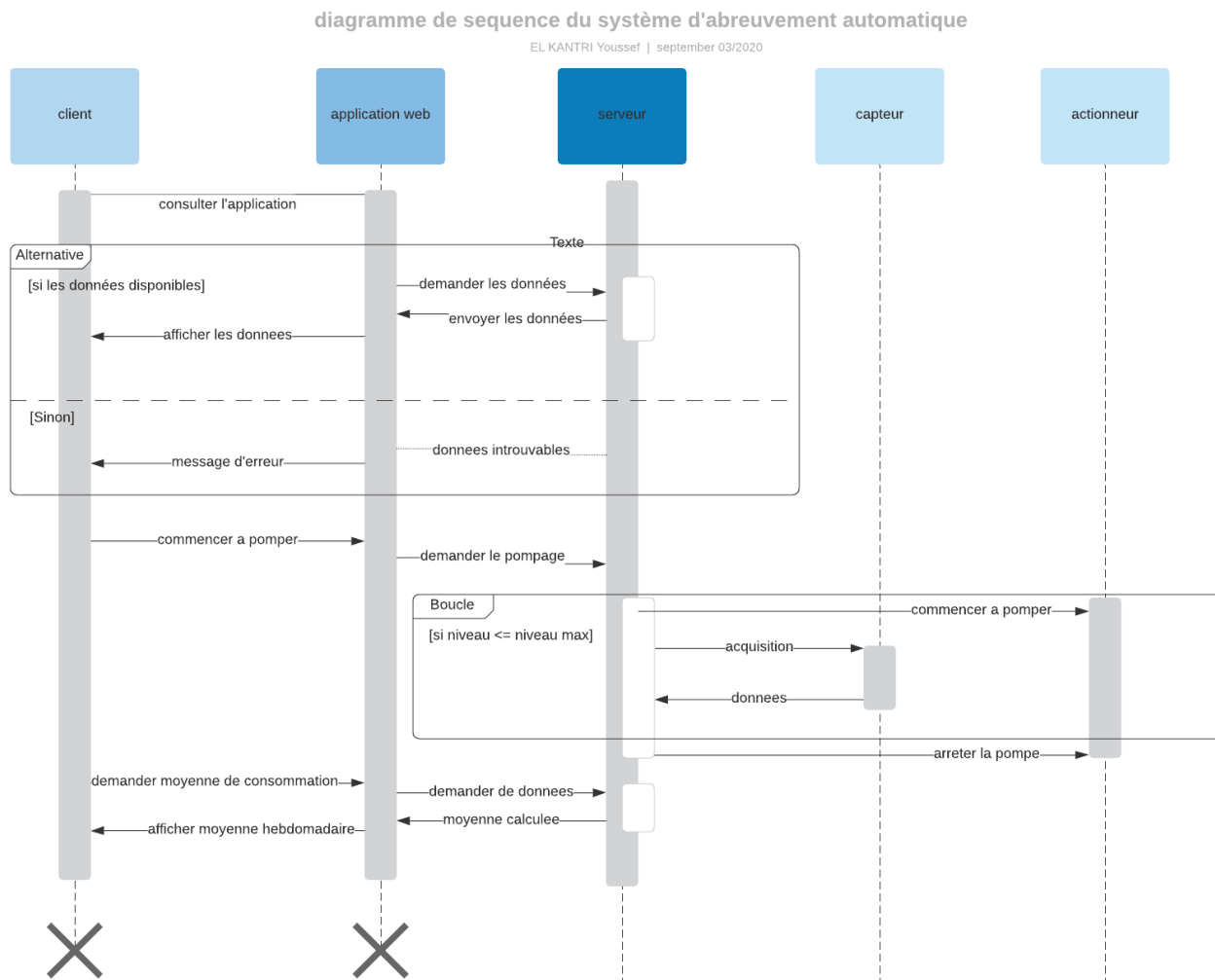


Figure 4: diagramme de séquence

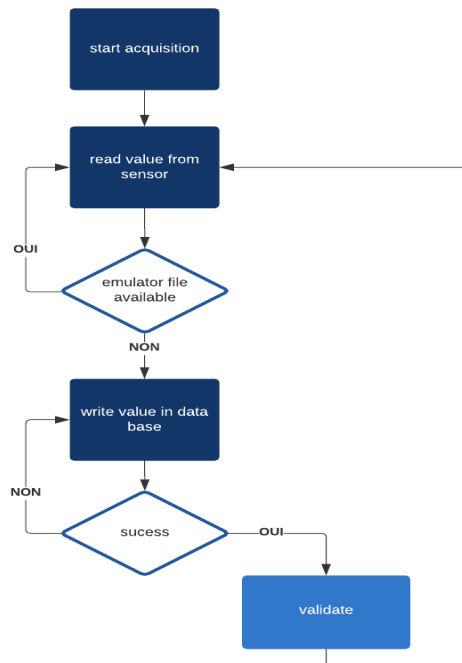
- Organigrammes :

- Organigramme d'acquisition :

L'organigramme suivant est utilisé pour donner une vision globale du process d'acquisition des valeurs de niveau d'eau de l'abreuvoir.

flux d'acquisition des données

EL KANTRI Youssef | september 03,2020

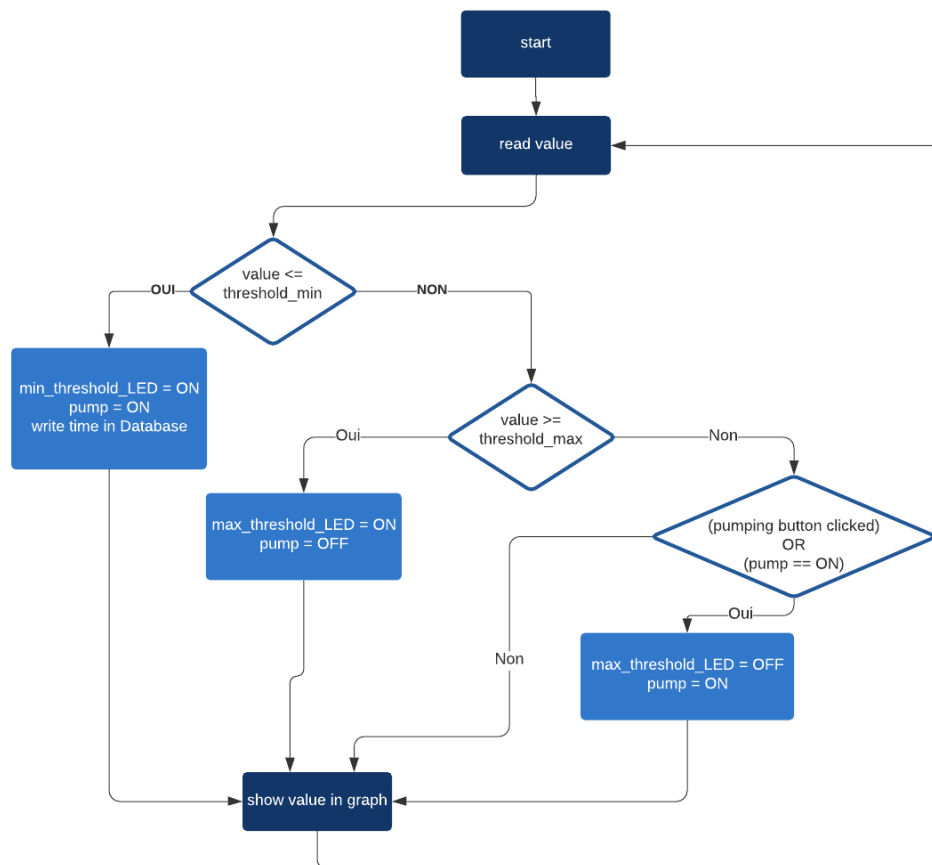


- Organigramme de contrôle :

Cet organigramme illustre le flux de contrôle globale de notre système :

flux de controle

EL KANTRI Youssef | september 03,2020



Architecture générale de projet :

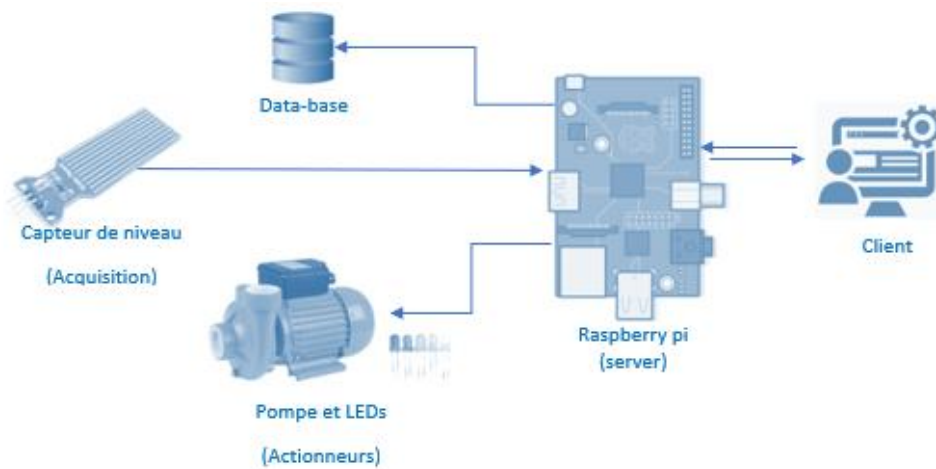


Figure 5: architecture générale du projet

Puisque notre projet est basé sur l'architecture client-serveur on peut deviser notre projet en 2 parties :

Partie serveur :

- Maintenir le fonctionnement du système asservi.
- Acquisition périodique des valeurs du niveau d'eau dans le réservoir grâce à un capteur de niveau.
- Acquisition de la quantité d'eau consommée par jour
- Contrôler l'état de plusieurs LEDs.
- Contrôle le fonctionnement de la pompe.

Partie client :

- Le client c'est une interface Web en PHP
- Donne au client le contrôle manuel de la pompe.
- Affiche la courbe de dernières 100 valeurs collectées par le capteur de niveau d'eau.
- Affiche un diagramme à bandes de la consommation en eau de chaque jour de la dernière semaine.
- Affiche la moyenne de consommation d'eau par jours de la dernière semaine en cliquant sur un bouton.

Environnement du projet :

- Environnement matériel :

Après avoir défini l'architecture global de notre projet, nous avons entamé une phase de choix de matériel :

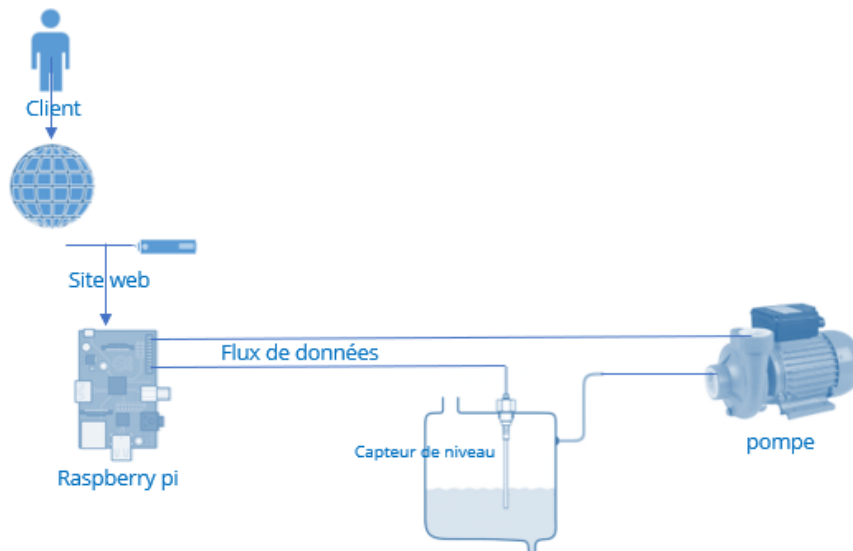


Figure 6: environnement matériel du projet

- Serveur embarqué :

Le serveur embarqué nous sert de cerveau au sein de notre système, C'est lui qui va comporter l'ensemble de notre site, pour cela on a choisi le Raspberry Pi 3, qui est sans doute l'une des cartes électroniques les plus puissantes qui existent aujourd'hui, c'est un nano-ordinateur mono-carte à processeur ARM conçu pour être plus flexible, voire sa fiabilité et robustesse, On rajoute qu'une carte Raspberry pi permet de connecter plusieurs esclaves I2C, elle comporte un système d'exploitation Linux qui nous permet de manipuler toute une data base afin de stocker toute information et valeur acquise par le système.

- Capteur (acquisition) :

Un capteur de niveau de type capacitif (numérique), économe, précis, avec une faible consommation de 10µA et facile à mettre en place grâce à une liaison série I2C.

Ce capteur est émulé par des accès fichier (.txt) dont on génère une valeur décroît aléatoirement au cours du temps (pour simuler les vaches en buvant)

- Actionneur (contrôle) :

Une pompe adéquate à notre besoin en eau, dont son rôle est le maintien l'asservissement du système.

Il est simulé par une LED et une résistance de 220 Ω

- Environnement logiciel :

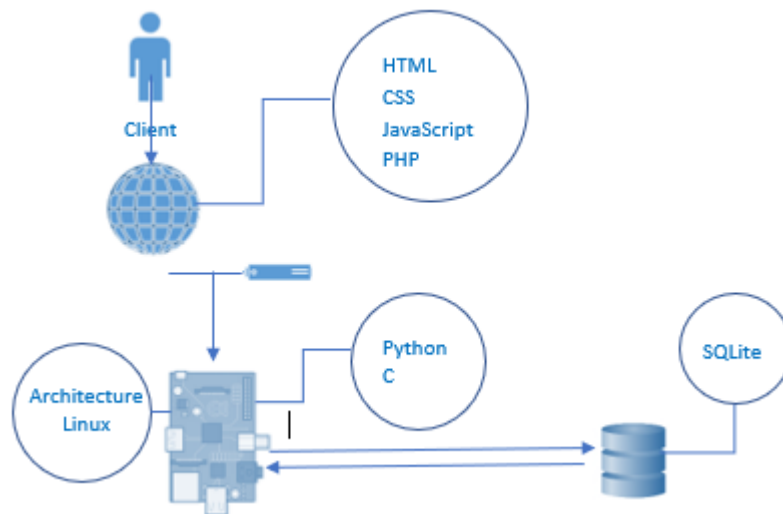


Figure 7: environnement logiciel du projet

- Architecture Linux:

L'OS choisi est le Raspberry Pi OS (anciennement Raspbian), Ce système opère sous une architecture linux ce qui nous permet de supporter plusieurs langages due à la richesse en sources des librairies, L'image ci-dessous montre un schéma du fonctionnement et de la communication de l'ensemble de couche de cette architecture :

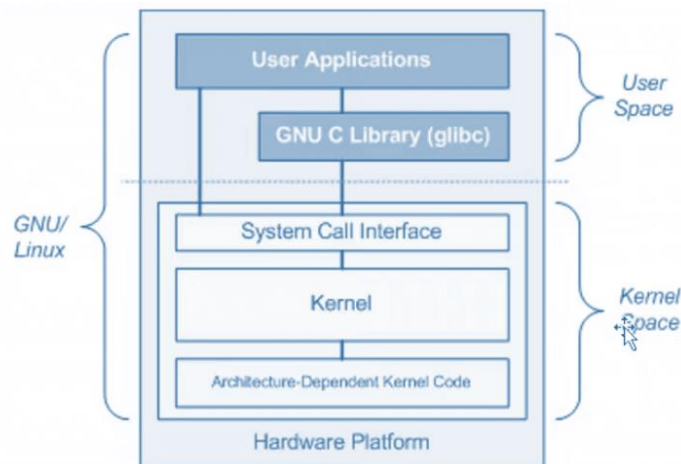


Figure 8: architecture Linux

- C Language:

Le C est un langage impératif, généraliste, issu de la programmation système. Inventé au début des années 1970 pour réécrire UNIX, il est devenu un des langages les plus utilisés. Dans ce projet le langage C est utilisé pour la programmation destinée à la génération des drivers.

- Python :

Python est un langage de programmation interprété, structuré et orienté objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

- PHP:

PHP : HyperText Preprocessor, abrégé PHP, est un langage de programmation libre principalement utilisé pour produire des pages Web dynamiques.

- SQLite :

SQLite est un système de gestion de base de données relationnelle. Ce moteur de base de données est connu pour implémenter une grande partie du standard SQL-92 et des propriétés ACID.

Cette bibliothèque écrite en C est directement intégrée au programme. Ce système et son code source sont entièrement open source.

- HTML:

L'HyperText Markup Language, abrégé HTML, est un langage conçu pour représenter les pages web. Ce langage de balisage permet de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, etc.

- CSS:

De l'anglais Cascading Style Sheets (les feuilles de style en cascade), le CSS constitue un langage informatique permettant d'ajouter des contraintes de mise en forme graphique dans des documents web, dont les normes sont établies par le World Wide Web Consortium (W3C)

- JAVA-Script:

JavaScript, abrégé JS, est un langage de programmation de scripts principalement utilisé dans les pages web interactives

Dans ce chapitre on a illustré l'environnement matériel et logiciel du projet. Le chapitre suivant a pour objectif de décrire la phase de réalisation et test de projet.

CHAPITRE III : REALISATION ET TEST DU PROJET

Ce dernier chapitre consiste à présenter l'étape de réalisation ainsi que les scénarios de test de la solution proposée afin de valider les fonctionnalités du système voire ses limitations et donc ses points faibles.

Réalisation :

Ce système comporte un serveur embarqué (Raspberry pi) dont on a déployé l'application Web, c'est lui qui orchestre la communication entre les différentes parties du système, comprenant l'acquisition, le traitement et le stockage des valeurs venants du capteur dans une base de données afin de les faire montrer sous forme des graphs informatifs sur une IHM, avec la possibilité de contrôler la pompe manuellement à l'aide d'un bouton.

Puisqu'on a émulé le capteur avec des accès fichier (.txt), donc ce fichier soit une ressource critique ; pour gérer l'accès multiple des processus à cette ressource critiques, j'ai choisi d'utiliser le module **asyncio** [4] de python, J'ai utilisé le module **ctype** [5] pour utiliser les codes en C comme bibliothèque partagée (.so) dans un scripte python.

Voici le diagramme qui montre les langages et les technologies utilisées pour réaliser chaque partie de l'application. Puisqu'elle est une application web, notre application peut être devisée en 2 parties ; backend et frontend :
Donc voilà les différents langages utilisés :

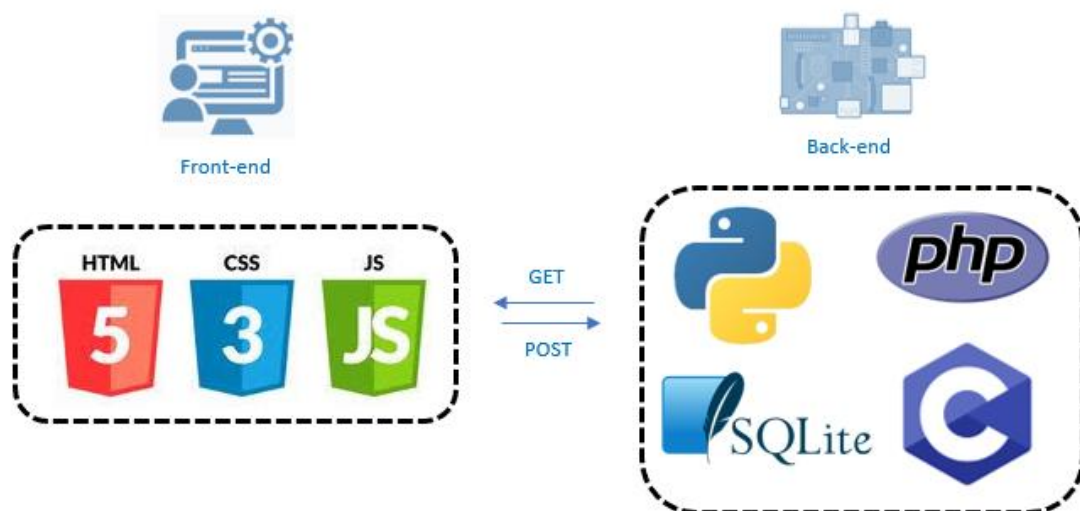


Figure 9: technologies utilisées dans le système

Teste de l'application web :

Pour tester l'application nous allons utiliser **cron** pour enregistrer les données à intervalle régulier dans une table, et pour cela on a créé une base de données préalablement,

Pour bien tester la robustesse du système et réaliser plusieurs scénarios on va diminuer tous les délais voire la lecture de niveau d'eau, l'écriture dans la base de données, et encore le délai de pompage d'eau.

Scenario de test :

- Teste du système asservi :

On va tester le système en variant les conditions initiales du système ainsi que les valeurs max et min du seuil.

Pour bien visualiser les sorties du système j'ai mis des LEDs indiquant les sorties du système (seuil max, seuil min et pompe ON)

- 1^{er} Test :

Valeurs initiales :

Valeur max = 95

Valeur min = 83

Valeur du capteur = 89

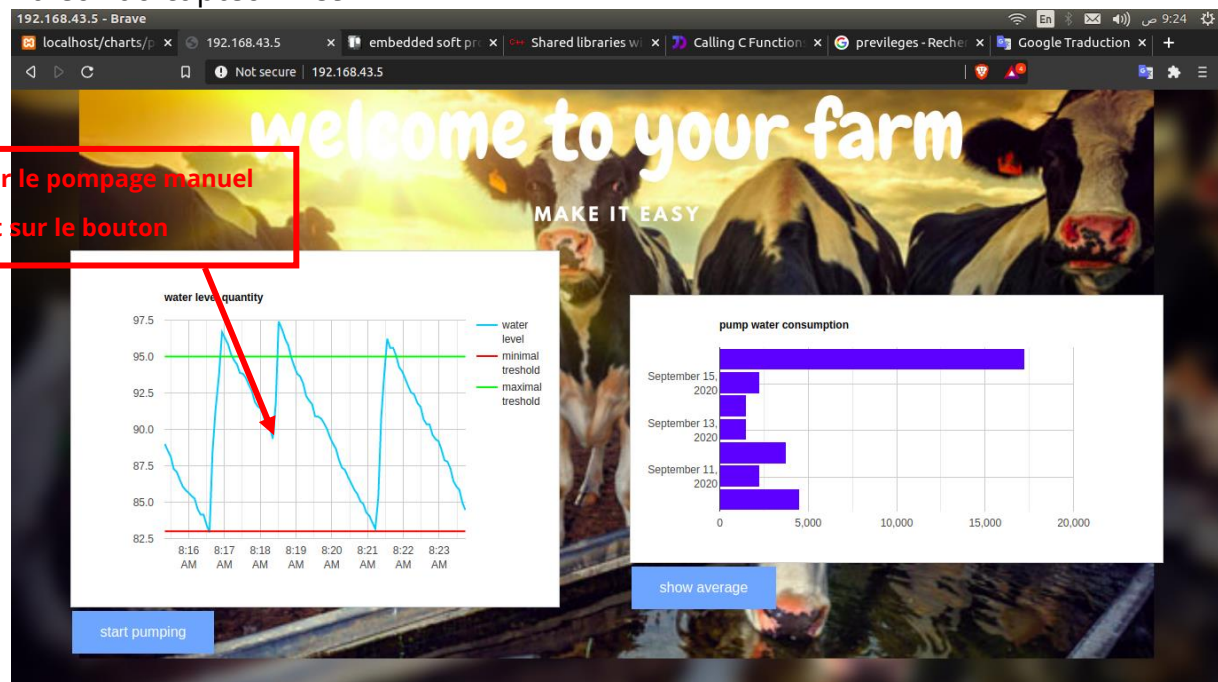


Figure 10: test n°1

- 2^{ème} Test :

Valeurs initiales :

Valeur max = 90%

Valeur min = 85%

Valeur du capteur = 80%

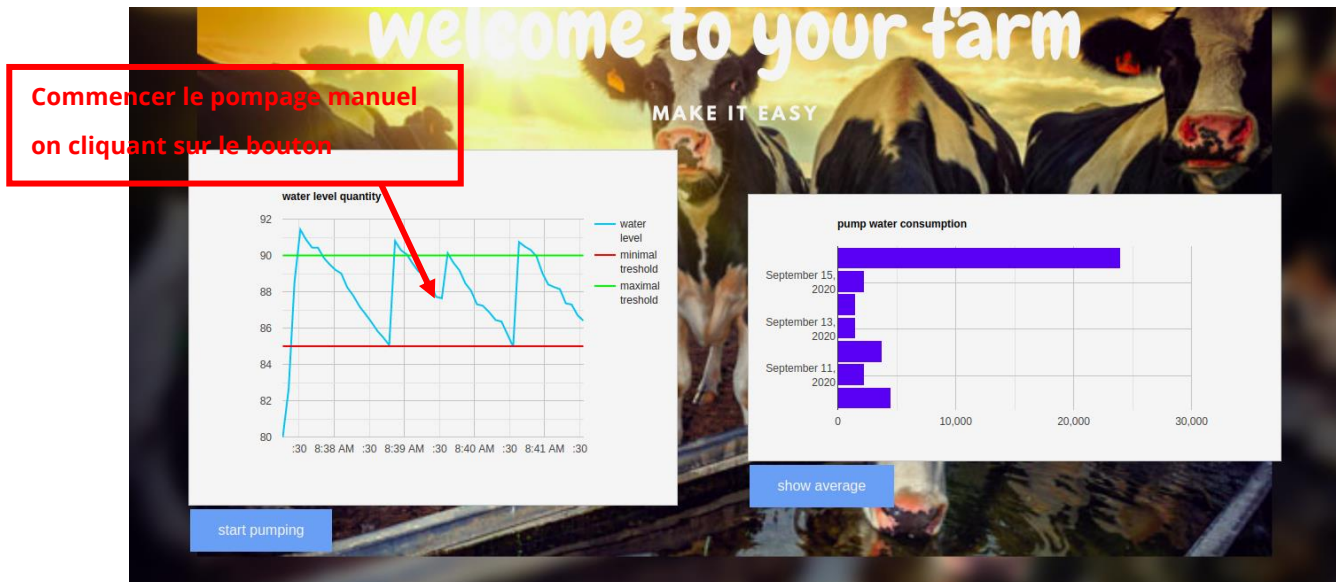


Figure 11: test n°2

- 3^{ème} Test :

Valeurs initiales :

Valeur max = 90%

Valeur min = 85%

Valeur du capteur = 95%

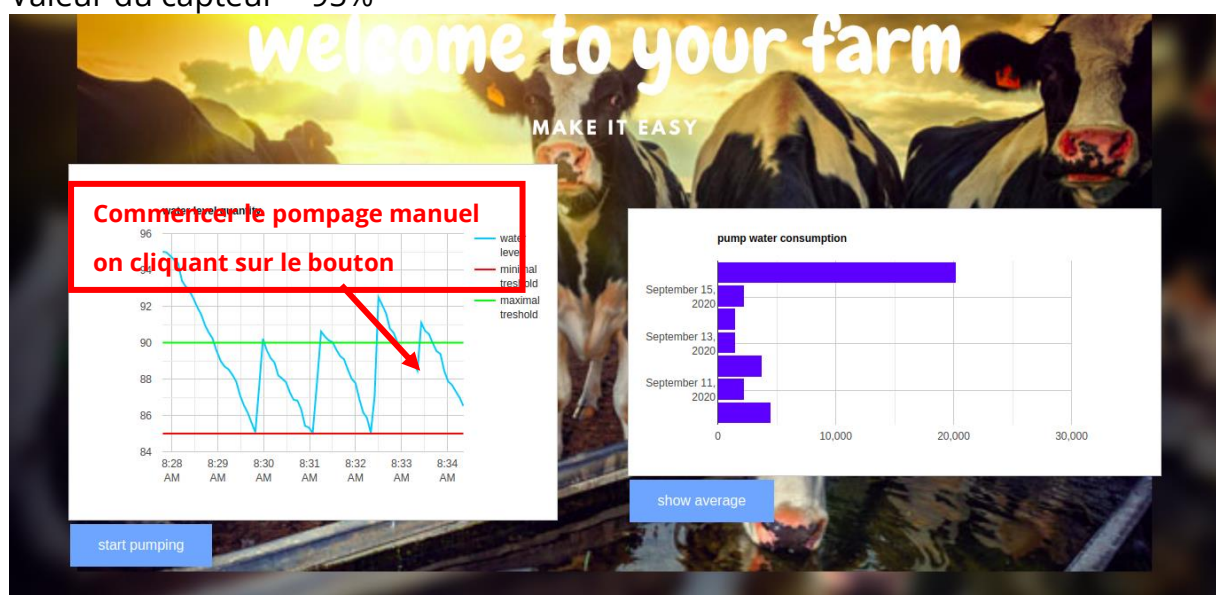


Figure 12: test n°3

- Test d'affichage de la valeur moyenne de la consommation hebdomadaire en eaux :

Après avoir l'acquisition de nombre de fois la pompe a pompé chaque jour on peut calculer la moyenne de consommation hebdomadaire en eau ce qui permet d'anticiper les tendances pour les jours et semaines à venir

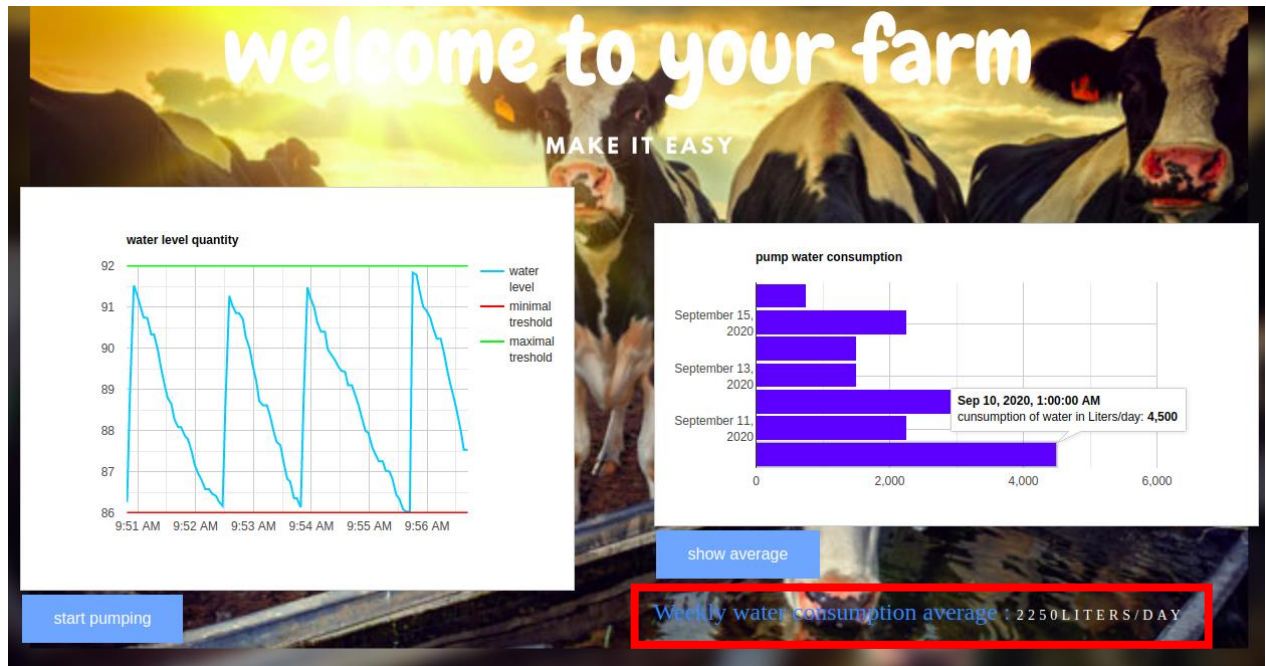


Figure 13: test n°4

on voit que la consommation hebdomadaire est de 2250 litres/jour

Dans ce chapitre on a présenté la phase de la réalisation ainsi la phase de test et validation du projet

CONCLUSION GENERALE :

Cette année, j'ai pris les devants en proposant ma propre vision de l'agriculture de précision. De ma réflexion est né un projet pour le moins enrichissant et formateur. Pour la première fois, j'ai mené une étude du papier à la mise en service du produit. En imaginant ce projet, j'avais en tête quelques bases et méthodes scolaires concernant l'élaboration d'une solution agriculture de précision, mais je ne connaissais ni les processus en amont, ni les processus en aval.

En maîtrisant le projet dans sa globalité, j'ai eu un aperçu complet du processus de développement d'une application client-serveur, des compétences et des connaissances qui y sont attachées. D'autre part, l'agriculture de précision a cette particularité de rassembler divers corps de métiers : programmation, informatique, électronique, électrotechnique, et bien d'autres.

Enfin, les temps impartis à la réalisation du système furent bref et il a fallu faire preuve de flexibilité et de persévérance, parfois pour respecter les délais, parfois pour respecter les contraintes technologiques imposées par le projet. Somme toute, j'ai retrouvé lors de ces deux semaines, les compétences, les contraintes mais aussi l'excitation d'un projet d'entreprise.

REFERENCES:

- [1]:<http://www.omafra.gov.on.ca/french/engineer/facts/07-024.htm>
- [2]:<https://online.visual-paradigm.com/>
- [3]:<https://app.lucidchart.com/documents>
- [4]:<https://docs.python.org/3/library/asyncio-task.html>
- [5]:<https://docs.python.org/3/library/ctypes.html#:~:text=ctypes%20is%20a%20foreign%20function,these%20libraries%20in%20pure%20Python.>