

ENSEIGNEMENT SUPERIEUR ET UNIVERSITAIRE  
INSTITUT SUPERIEUR TECHNIQUE ADVENTISTE DE GOMA  
« ISTAGO »



Email: [uago2015@gmail.com](mailto:uago2015@gmail.com)

Site web: [www.uagom.com](http://www.uagom.com)

B.P: 116 GOMA

**Département : Science et Technologie**  
**Option : Réseau et Infrastructure**

**MISE EN PLACE D'UN SYSTEME DE PRODUCTION DE  
RAPPORT METEOROLOGIQUE POUR UN CHAMP CONNECTE**

Par MIKA **BAHIZI Justin**

Mémoire de fin de cycle présenté et défendu en  
vue de l'obtention du diplôme de Master en  
Réseau et Infrastructure.

DIRECTEUR : Dr. Luc Ngend, Ph. D

**Année Académique 2023**

## DECLARATIONS DE L'ETUDIANT

Je soussigné, MIKA BAHIZI Justin, étudiant en réseau et infrastructure à l'Institut Adventiste de Goma, déclare solennellement ce qui suit concernant la rédaction de mon mémoire de master :

1. Je confirme que le présent mémoire de master est le résultat de mes propres recherches et efforts, menés sous la supervision de mon directeur de mémoire.
2. Je m'engage à respecter les principes d'intégrité académique et d'éthique dans toutes les phases de la rédaction de mon mémoire, en évitant toute forme de plagiat, de tricherie ou de falsification de données.
3. Je reconnais l'importance de citer et de référencer correctement les sources utilisées dans mon mémoire, en accordant une reconnaissance appropriée aux travaux et aux idées d'autres chercheurs.
4. Je suis conscient de l'impact potentiel de mes recherches et de mes conclusions sur mon domaine d'étude, et je m'engage à mener une analyse rigoureuse et impartiale des données pour garantir la fiabilité et la validité de mes résultats.
5. Je suis ouvert aux commentaires, aux critiques constructives et aux suggestions de mon directeur de mémoire et des membres de mon comité d'évaluation, et je suis prêt à apporter les modifications nécessaires pour améliorer la qualité de mon mémoire.
6. Je suis conscient de l'importance de respecter les délais fixés par mon établissement d'enseignement pour la soumission de mon mémoire et je m'engage à respecter ces échéances, tout en maintenant la qualité et la rigueur de mon travail.
7. Je m'engage à présenter les résultats de ma recherche de manière claire, cohérente et professionnelle, en utilisant les normes de présentation et de formatage spécifiées par mon établissement d'enseignement.
8. Je reconnais que la rédaction d'un mémoire de master peut être un processus exigeant, tant sur le plan intellectuel qu'émotionnel, et je m'efforce de maintenir un équilibre entre mon travail académique et ma santé personnelle.
9. Je suis reconnaissant envers mon établissement d'enseignement, mes enseignants et mes pairs pour leur soutien et leur contribution à mon parcours académique, et je m'engage à valoriser cette opportunité en produisant un mémoire de master de haute qualité.

10. Je comprends que la réussite de mon mémoire de master repose sur mon engagement, ma persévérance et ma volonté de repousser les limites de la connaissance dans mon domaine d'étude, et je suis déterminé à faire de mon mieux pour atteindre cet objectif.

Je confirme que toutes les déclarations ci-dessus sont véridiques et que je suis prêt à assumer la responsabilité de mes actes en tant qu'étudiant rédigeant un mémoire de master.

Fait à Goma, le 13/09/2023

MIKA BAHIZI Justin

**EPIGRAPHE**

« Deus sive Natura »

Baruch Spinoza (1632-1677)

## **DEDICACE**

Nous dédions ce présent travail à nos très chers parents BAHIZI et ZAWADI.

## REMERCIEMENTS

Nos remerciements les plus distingués s'adressent à la famille BAHIZI pour leur aide financière tant morale, courage, courtoisie et amour pendant nos études, qui fait un moment inoubliable dans le parcours de notre vie.

Aux autorités académiques de l'Institut Supérieur Techniques Adventiste de Goma (ISTAGO) en général et celle de la faculté de Gestion Informatique en particulier notre Directeur Dr. Luc Ngend, PhD, Dr. Elias Semajeri, PhD Recteur de l'UAGO sans oublier le CT Josué KALEMA qui ne cessent de nous apporter soutiens, directives et conseils pour nous permettre d'atteindre le niveau de standard mondial et d'être compétitif au marché de l'emploi.

Nous tenons à remercier nos frères, sœurs et camarades qui nous ont soutenus durant la réalisation de ce travail. Que toutes les bonnes volontés qui ont toujours été à nos côtés pour nous permettre de progresser et qui ne figurent pas sur cette liste, se rassurent de notre gratitude.

MIKA BAHIZI Justin

**SIGLES ET ABBREVIATIONS**

ARM	: Advanced RISC Machine
BLE	: Bluetooth Low Energy
EEPROM	: Electrically Erasable Programmable Read-Only Memory
GHz	: Giga Hertz
GPS	: Global Positioning System
GND	: Ground
IHM	: Interface Homme-Machine
IOT	: Internet of Things
LED	: Light Emitting Diode
MHz	: Mega Hertz
MySQL	: My Structured Query Language
OMG	: Object Management Group
OMS	: Organisation Mondiale de la Santé
OOA	: Object Oriented Analysis
OOD	: Object Oriented Design
OOM	: Object-Oriented Model
OOSE	: Object Oriented Software Engineering
PERT	: Program Evaluation Review Technique
RAM	: Random Access Memory
SRAM	: Static Random Access Memory
SPA	: Single Page Application
UART	: Universal Asynchronous Receiver Transmitter
UML	: Unified Modeling Language

USB : Universal Serial Bus

UP : Unified Process

V : Volt

VIN : Volt In

Wi-Fi : Wireless Fidelity

**LISTE DES TABLEAUX**

Tableau 1. Détermination des tâches.....	33
Tableau 2. Détermination du coût matériel.....	34
Tableau 3. Détermination du coût matériel.....	35
Tableau 4. Détermination du coût global .....	35
Tableau 5. Environnement Matériel.....	57

## LISTE DES FIGURES

Figure 1 : Internet des objets .....	12
Figure 2: Les Composants d'IoT .....	13
Figure 3: Microcontrôleur .....	16
Figure 4: Microcontrôleur .....	16
Figure 5 : Composants clés d'une carte Arduino.....	17
Figure 6 : Le fonctionnement de base d'un Arduino. ....	18
Figure 7 : Arduino Uno R3 .....	19
Figure 8 : Arduino Leonardo.....	20
Figure 9 : Arduino nano .....	21
Figure 10: Arduino Micro .....	22
Figure 11: Arduino Nano 33 BLE / Nano 33 BLE Sense .....	23
Figure 12: Arduino MKR Zero .....	24
Figure 13: Arduino Mega 2560 Rz .....	25
Figure 14: IoT Arduino Boards .....	26
Figure 15 : API.....	27
Figure 16: Api avec interaction avec une base de données .....	27
Figure 17 : Api avec interaction avec une base de données .....	28
Figure 18: Logo officiel de React.js .....	28
Figure 19 : Logo officiel de Nodejs .....	29
Figure 20: Logo UML .....	30
Figure 21 : Elaboration du graphe PERT .....	36
Figure 22: Date au plus tôt d'un réseau PERT .....	38
Figure 23 : La date au plus tard d'un réseau PERT .....	40
Figure 24: chemin critique .....	44
Figure 25 : Calendrier du projet .....	45
Figure 26 : Diagramme Gantt.....	46
Figure 27 : Diagramme Pert .....	47
Figure 28 : Diagramme de cas d'utilisation .....	49
Figure 29 : Représentation des activités.....	51
Figure 30 : Diagramme de séquence .....	52
Figure 31 : Diagramme de classe .....	54
Figure 32 : Le diagramme d'objet.....	54

Figure 33 : Diagramme des composants .....	55
Figure 34 : Diagramme de déploiement.....	56
Figure 35 : Structure du projet .....	57
Figure 36 : La page de connexion pour les utilisateurs.....	58
Figure 37 : Création d'un compte .....	58
Figure 38 : Création d'un compte .....	59
Figure 39 : Création d'un compte .....	59
Figure 40 : Management de compte utilisateur.....	60
Figure 41 : Home page .....	61
Figure 42 : Visualisation des données en temps réels.....	61
Figure 43 : Direction du vent [La girouette W200P] .....	62
Figure 44 : Température.....	62
Figure 45 : Niveau de la précipitation.....	63
Figure 46 : Humidité .....	63
Figure 47 : Système embarque .....	64
Figure 48 : Système embarque .....	65
Figure 49: Système embarque .....	66
Figure 50: Initialisation .....	67
Figure 51: Récolte et envoi des données.....	68
Figure 52: Récolte et envoi des données.....	69

## RESUME

Les rapports météorologiques jouent un rôle crucial dans diverses industries telles que l'agriculture, l'aviation, les énergies renouvelables et la gestion des catastrophes naturelles. Cependant, leur production exige la collecte, le traitement et l'analyse de données météorologiques en temps réel, ce qui est une tâche complexe. Établir un système de production de rapports météorologiques efficace représente un défi majeur pour les organisations impliquées.

La production de rapports météorologiques précis en temps réel est complexe et coûteuse, et de nombreuses régions, notamment l'agriculture, dépendent de ces informations pour prendre des décisions cruciales. Les défis sont multiples : le changement climatique a rendu les conditions météorologiques imprévisibles, ce qui perturbe la planification agricole ; l'impact des catastrophes naturelles, telles que les inondations et les sécheresses, exige une surveillance étroite pour minimiser les pertes ; et la nécessité de gérer efficacement les ressources telles que l'eau et l'énergie dans un contexte de durabilité agricole exige des données météorologiques précises pour optimiser les pratiques agricoles. La manipulation de données météorologiques incertaines et incomplètes influence la qualité et la précision des prévisions, ce qui peut avoir un impact considérable sur divers secteurs, y compris l'agriculture. L'incertitude météorologique peut entraîner des pertes de récoltes, une utilisation inefficace des ressources et des décisions agricoles sous-optimales. De plus, l'agriculture est un secteur clé pour la sécurité alimentaire mondiale, et la précision des informations météorologiques joue un rôle essentiel dans la garantie d'un approvisionnement alimentaire stable pour une population mondiale en croissance.

Ce mémoire présente un système de production de rapports météorologiques basé sur l'Internet des Objets (IoT), combinant des capteurs de température, d'humidité, et de réseau sans fil Wi-Fi pour la collecte en temps réel de données météorologiques. Ces données sont acheminées vers une base de données MySQL via une API, tandis qu'un site web Next.js présente les informations sous forme de graphiques. Cette solution offre un potentiel considérable pour améliorer la productivité agricole et répondre aux défis liés à la croissance démographique mondiale et à la demande croissante en alimentation. Elle permet également une surveillance en temps réel des opérations et une gestion plus efficace des ressources, tout en explorant des opportunités d'extension vers d'autres domaines, élargissant ainsi les perspectives d'utilisation de l'IoT dans la collecte et l'analyse des données météorologiques.

## **ABSTRACT**

Weather reporting plays a crucial role in various industries such as agriculture, aviation, renewable energy and natural disaster management. However, their production requires the collection, processing and analysis of real-time weather data, which is a complex task. Establishing an effective weather reporting system represents a major challenge for the organizations involved.

Producing accurate, real-time weather reports is complex and expensive, and many regions, including agriculture, rely on this information to make critical decisions. The challenges are multiple: climate change has made weather conditions unpredictable, disrupting agricultural planning; the impact of natural disasters, such as floods and droughts, requires close monitoring to minimize losses; and the need to efficiently manage resources such as water and energy in the context of agricultural sustainability requires accurate weather data to optimize agricultural practices. The manipulation of uncertain and incomplete weather data influences the quality and accuracy of forecasts, which can have a significant impact on various sectors, including agriculture. Weather uncertainty can lead to crop losses, inefficient use of resources, and suboptimal agricultural decisions. Additionally, agriculture is a key sector for global food security, and accurate weather information plays a critical role in ensuring a stable food supply for a growing global population.

This paper presents a weather reporting system for an Internet of Things (IoT)-based, combining temperature, humidity, and Wi-Fi wireless network sensors for real-time collection of weather data . This data is routed to a MySQL database via an API, while a Next.js website presents the information in graphs. This solution offers considerable potential to improve agricultural productivity and meet the challenges linked to global population growth and increasing demand for food. It also enables real-time monitoring of operations and more efficient management of resources, while exploring opportunities to expand into other areas, thereby broadening the prospects for using IoT in collection and analysis meteorological data.

# CHAPITRE PREMIER :

## INTRODUCTION GENERALE

### 1.1. Introduction

Dans de nombreux domaines, dont l'écologie, l'architecture, la conversion d'énergie, etc., il existe un besoin d'évaluation de phénomènes météorologiques comme la température, la pression, l'humidité ou encore la luminosité. Les météorologues ont besoin de recueillir des données à de nombreuses échelles afin de prédire l'avenir. Cela a créé un besoin pour la création de nouveaux outils d'estimation des variables météorologiques. Les avancées technologiques ont permis la création de tout nouveaux équipements de mesure aux performances élevées. Cette avancée significative a donné un nouvel élan à l'ingénierie météorologique, notamment avec l'émergence de nouveaux équipements permettant d'estimer diverses grandeurs météorologiques.

De surcroit, le contexte de ce sujet est l'importance croissante de la météorologie pour de nombreuses industries et organismes gouvernementaux afin de prendre des décisions informées. Les rapports météorologiques fournissent des informations précieuses sur les conditions météorologiques actuelles et futures, lesquelles sont essentielles pour prendre des décisions en matière de planification et de gestion de risques dans diverses industries telles que l'agriculture, les loisirs, les sports, le transport, les services d'urgence, etc.

La nécessité d'intensifier et d'industrialiser le secteur agricole s'est accrue en raison de la demande croissante de produits alimentaires, tant en quantité qu'en qualité. L'Internet des objets est une famille prometteuse de technologies qui peuvent apporter plusieurs solutions pour la modernisation de l'agriculture. Les communautés scientifiques et de recherche, ainsi que les entreprises, se lancent dans une course pour offrir de plus en plus de biens basés sur Internet aux acteurs du secteur agricole et jettent les bases d'un rôle clair pour qu'Internet devienne une technologie dominante. (Malika, M.haniche, & Nabila, 2018).

Non seulement mais encore, l'objectif de la mise en place d'un système de production de rapport météorologique est de fournir des informations pertinentes, précises, fiables et actualisées sur les conditions météorologiques et les tendances futures aux utilisateurs de différents secteurs. Un tel système peut être bénéfique pour les entreprises privées comme pour les organismes gouvernementaux car cela permettrait une meilleure gestion des risques liés aux conditions météorologiques et une meilleure planification des activités en conséquence. Sans surprise, la météo agricole informe les agriculteurs des prévisions météorologiques sur

différentes échelles de temps (heure par heure, jour par jour, sur plusieurs jours...). Elle permet aux agriculteurs d'organiser et de planifier leur travail en fonction des conditions météorologiques et des aléas à venir (Caffarra, 2023).

La météo agricole donne aux agriculteurs toutes les informations dont ils ont besoin pour gérer et protéger leurs cultures, y compris quand arroser, irriguer, planter et protéger. En fait, ce service en ligne fournit des données météorologiques complètes, notamment la température, l'humidité, la densité de l'air, la pression atmosphérique, la vitesse du vent et le taux de précipitations...

Afin d'obtenir les informations de météorologie agricole qui concernent ses cultures, l'agriculteur doit fournir ses coordonnées GPS. Il dispose alors d'une prévision météo en temps réel et des tendances sur les prochains jours. Bien que ces informations soient utiles, elles manquent souvent de précision et de fiabilité. Il n'est en effet pas rare que les agriculteurs se retrouvent surpris par la gravité des événements climatiques qui surviennent. Par ailleurs, les applications météo agricoles se contentent d'indiquer les prévisions (Caffarra, 2023).

Dans un contexte où les changements climatiques sont en constante évolution, la mise en place d'un système de production de rapport météorologique est également une manière de contribuer à l'effort international de lutte contre le réchauffement climatique. Les rapports météorologiques peuvent fournir des données précieuses qui contribuent à mieux comprendre comment le climat évolue sur différentes régions et ainsi élaborer des stratégies pour s'adapter aux changements climatiques. Bien plus, elles ne permettent pas d'évaluer l'impact des aléas climatiques sur les cultures... Et c'est bien là que le bât blesse ! Plus que de connaître la météo, les agriculteurs ont besoin de savoir quelle sera l'ampleur des pertes de rendement engendrées. Or, les solutions de météorologie agricole permettent de répondre à l'interrogation « quel temps fera-t-il demain ? » et non à la problématique « comment se porteront mes cultures demain ? »

## **1.2. Problématique**

Les rapports météorologiques sont essentiels pour de nombreuses industries, y compris l'agriculture, l'aviation, les énergies renouvelables et la gestion des catastrophes naturelles. Cependant, la production de rapports météorologiques précis et fiables est une tâche complexe qui nécessite la collecte, le traitement et l'analyse de grandes quantités de données météorologiques en temps réel. Dans ce contexte, la mise en place d'un système de production de rapport météorologique efficace est un défi majeur pour les organisations concernées.

Cela dit que, les progrès rapides de la technologie observés ces derniers temps ont eu un effet positif sur la plupart des facettes de la vie quotidienne, offrant de nouvelles possibilités

d'expression de la créativité humaine et faisant partie intégrante de la plupart des professions. Avec l'adoption croissante de l'Internet des Objets (IoT), les appareils connectés ont pénétré tous les aspects de notre vie, de la santé à la domotique, en passant par l'automobile et les villes intelligentes. Il est donc logique que l'IoT trouve son application dans l'agriculture et, en tant que tel, en améliore considérablement presque toutes les facettes.

L'IoT est un système de communication entre objets connectés qui répond aux besoins de l'utilisateur et à l'environnement. Il s'agit d'arriver à combiner le monde réel et le monde virtuelle, et de standardiser ce nouveau système en développant des nouvelles applications et nouveaux environnements et protocoles de communication. La prévision météorologique est basée sur l'étude de plusieurs paramètres liés au climat, dont des radars et satellites météorologiques (télédétection), des stations météorologiques dispersées sur plusieurs sites différents. La manipulation des données incertaines et leur impact sur le processus de prédiction météorologique entraîne une certaine incertitude et incomplétude des données, qui influence la qualité et la précision de prédiction (Nourhane & Aya, 2022).

Seulement mais encore, les systèmes de production de rapport météorologique modernes sont basés sur des technologies avancées telles que les capteurs, les satellites et les modèles de prévision météorologique. Ces systèmes peuvent collecter des données météorologiques en temps réel et les analyser pour produire des rapports précis et fiables. Cependant, la mise en place de ces systèmes peut être coûteuse et complexe, et nécessite une expertise technique et météorologique.

Aujourd'hui, l'intégration des capteurs et des actionneurs au sein des puces microélectroniques ouvre un formidable éventail de possibilités ; Pour une meilleure acquisition de ces données et leur archivage, les ingénieurs ont mis en œuvre des techniques d'automatisation plus performantes et plus précises. Auparavant, les mesures se font par des moyens mécaniques à l'aide d'enregistreurs ou à la main. Cette façon de procéder était très pénible et entraînait de nombreuses erreurs (Temam, 2012). Elles peuvent assurer d'autres fonctions tel que la communication avec d'autres puces positionnées autour d'elles par le biais de transmissions radioélectrique, optique ou acoustiques (Temam, 2012). Fondamentalement, elles sont capables de percevoir leur environnement, d'analyser les données et d'interagir en conséquence avec celui-ci.

L'utilisation des technologies connectées dans le secteur de l'agriculture est un moyen pour développer l'économie et satisfaire les besoins alimentaires et améliorer chaque jour un peu plus le suivi de leurs bétails et de leurs cultures céréalières, l'étude note également l'utilité de l'IoT pour les tâches plus modestes, comme celles effectuées dans les cultures laitières, maraîchères

et fruitières, et dans les cultures sous serres. Étant donné leur formidable impact dans le domaine de l'agriculture (Malika & Nabila, 2019).

avec la croissance exponentielle de la population mondiale, nous devons produire 70% de nourriture en plus à l'horizon 2050, selon les estimations de l'OMS (mutualia, 2023). Avec la réduction des terres agricoles et l'épuisement des ressources naturelles, la nécessité d'améliorer le rendement agricole est devenue critique. Environ 61,5 % de la population indienne est directement impliquée dans travail agricole. En Inde, on parle de méthodes d'agriculture à travers l'ancien des outils comme le plow. Dans l'ancienne agriculture, les agriculteurs utilisation de fertilisants bio. Auparavant, ils n'avaient de nombreuses options liées aux semences et aux dépendaient uniquement des moussons et de leurs seules prévisions météorologiques prévues le soleil.

De l'autre côté, la modernité L'agriculture est une technologie appropriée Gestion de l'approvisionnement en eau, Nouvelle outils automatisés innovants (Kushagra, 2019) . Les agriculteurs ont désormais la tâche plus facile grâce à l'adaptation d'Internet Source dans les procédures agricoles. Ils sont capables de recueillir des informations complètes sur leurs plantes et leur culture grâce aux capteurs. Lorsque le temps est si mauvais, l'IoT soutient les agriculteurs car ils sont incapables de prévoir la météo avec précision. L'IoT va révolutionner le secteur des startups agricoles. Des informations automatisées et immédiates sur la composition du sol, la santé des cultures et les besoins en eau sont fournies via des applications intelligentes.

Tout bien considéré, la mise en place d'un système de production de rapport météorologique efficace est un défi majeur pour les organisations concernées. Ce mémoire propose une mise en place d'un tel système en répondant aux questions clés de cette recherche. Les résultats de ce mémoire peuvent aider les organisations à concevoir et à mettre en place des systèmes de production de rapport météorologique efficaces qui répondent aux besoins spécifiques des utilisateurs et aux contraintes opérationnelles de la collecte et de l'analyse de données météorologiques en temps réel.

## 1.3. OBJECTIF DU TRAVAIL

Un projet sans objectif prédéfini ne peut pas toujours aboutir à un bon résultat c'est pourquoi nous nous sommes fixé l'objectif général ainsi que les objectifs spécifiques.

### 1.3.1. Objectif général

L'objectif principal de ce mémoire est la mise en place d'un système de production de rapport météorologique efficace pour un champ connecté, spécifiquement conçu pour répondre aux besoins des agriculteurs, tout en tenant compte des contraintes opérationnelles liées à la collecte et à l'analyse de données météorologiques en temps réel dans un contexte de champ connecté.

### 1.3.2. Objectifs Spécifiques

D'une manière spécifique, ce travail a pour objectif la réalisation d'un prototype de système de rapports météorologiques pour un champ connecté conçu pour répondre aux besoins des agriculteurs dans le contexte d'un champ connecté, avec un système central sous forme d'application web développée avec Next.js, une base de données MySQL, et une API pour la collecte et la transmission des données. L'objectif ultime de ce système est d'aider les agriculteurs à surveiller en temps réel la santé de leurs plantations tout en fournissant des données précieuses pour évaluer la productivité du champ.

Les objectifs spécifiques suivants ont été établis :

- ✓ Développer un prototype de système embarqué capable d'acquérir des données météorologiques à l'aide de capteurs de température, d'humidité, direction du vent, la luminosité des rayons soleil, le niveau de précipitation et d'envoyer ces données à une base de données MySQL via une API dédiée.
- ✓ Concevoir et mettre en œuvre une API robuste pour le système embarqué, permettant la collecte, la transmission et le stockage sécurisé des données météorologiques dans la base de données MySQL.
- ✓ Développer l'application web avec Next.js pour permettre aux agriculteurs d'accéder aux données météorologiques enregistrées dans la base de données MySQL via une interface utilisateur conviviale.
- ✓ Intégrer des fonctionnalités de visualisation des données en temps réel dans l'application web, offrant aux agriculteurs une vue claire des conditions météorologiques actuelles dans leur champ connecté.

- ✓ Mettre en place des mécanismes d'authentification et d'autorisation pour garantir la sécurité des données et permettre un accès contrôlé aux informations météorologiques stockées dans la base de données MySQL.

## 1.4. METHODES ET TECHNIQUES

### 1.4.1. METHODES

Pour obtenir la scientificité, tout chercheur fait recours à la méthodologie pour atteindre l'objectivité. Pour notre travail nous avons opté les méthodes suivantes :

- **La méthode PERT (Program Evaluation and Review Technique)** est une méthode conventionnelle utilisable en gestion de projet, ordonnancement et planification. Elle fournit une méthode et des moyens pratiques pour décrire, représenter, analyser et suivre de manière logique les tâches à réaliser dans le cadre d'un projet. Son but est d'organiser les tâches sous la forme d'un réseau afin de faciliter la gestion du projet. Cette représentation graphique permet d'identifier les connexions entre les différentes tâches, les temps d'exécution, les interdépendances. L'objectif étant de maîtriser le délai de réalisation.
- **La méthode historique** : « elle rassemble, ordonne, hiérarchise autour d'un phénomène singulier, une pluralité des faits afin de déceler celui qui a plus d'influence sur le phénomène étudié ». Elle nous a survie dans les renseignements sur la succession des événements qui ont caractérisé notre organisation dans le passé.
- **La méthode UP (Unified Process)** : Le processus unifié fournit un cadre au développement logiciel pour la construction des systèmes orientés objet et répond aux exigences fondamentales. Le Processus Unifié ou UP (Unified Process) est une méthode générique de développement de logiciel développée par les concepteurs d'UML et par conséquent il utilise le langage de modélisation unifié (UML).

### 1.4.2. TECHNIQUES

Les techniques suivantes ont été utilisées enfin de bien mener à bon escient notre travail :

- **La technique documentaire** : Cette technique nous aidera à consulter des différents documents tel que : les articles, les mémoires, les ouvrages, les différentes publications, les manuels du domaine d'étude en rapport avec notre sujet.

- **La Technique dite de l'interview libre** : Cette technique nous a permis de questionner certaines personnes sur la météorologie, c'est ainsi que nous avons compris le fonctionnement de cette dernière.

## **1.5. Choix et intérêt du sujet**

### **1.5.1. Choix du sujet**

Le choix de ce sujet a été motivé par l'importance que joue la nouvelle technologie de l'information et la communication dans la société actuelle, vu la recherche de solutions pour améliorer la production agricole tout en préservant les ressources naturelles et en réduisant l'impact environnemental, nous avons trouvé mieux de mettre en place un système de rapport météorologique.

### **1.5.2. Intérêt du sujet**

#### **1.5.2.1. Intérêt personnel**

Personnellement ce travail nous permettra d'acquérir des connaissances approfondies sur les nouvelles technologies et leur application dans l'agriculture pour améliorer la gestion des risques liés aux conditions météorologiques et climatiques. Bien plus, il nous permettra aussi de développer des compétences en matière de recherche et d'analyse de données pour résoudre des problèmes complexes voire réaliser un mémoire de fin de cycle de master dans le domaine de science et technologie ; option réseau et infrastructure.

#### **1.5.2.2. Intérêt scientifique**

De manière scientifique, notre sujet de recherche traitant sur la mise en place d'un système de production de rapport météorologique dans l'agriculture réside dans la possibilité d'améliorer la gestion des risques liés aux conditions météorologiques et climatiques dans l'agriculture. En utilisant les nouvelles technologies pour générer des cartes et des rapports météorologiques précis, les agriculteurs peuvent prendre des décisions éclairées pour moduler les interventions culturales en fonction des conditions météorologiques et climatiques. Cela peut aider à créer des systèmes agricoles viables et rentables, à améliorer le rendement et la qualité des produits, à diminuer les pertes et les dangers, à abaisser les coûts, à optimiser l'utilisation de l'eau, de la main-d'œuvre et de l'énergie, à préserver les ressources naturelles et à réduire la pollution par les produits chimiques ou d'autres substances qui détériorent l'environnement.

## 1.6. Délimitation spatio-temporelle

Tout travail scientifique doit être limité dans le temps et dans l'espace. D'une part, pour le cadre de notre thématique de recherche, nous n'allons pas traiter toutes les questions liées à l'agro météorologie; nous allons plutôt nous focaliser sur la réalisation d'un prototype de système de rapport météorologiques permettant d'acquérir des données (température, d'humidité, direction du vent, la luminosité des rayons soleil, le niveau de précipitation) qui sera appuyer par différents capteurs (température, humidité, ...). D'autre part, du point de vue temporel cette thématique d'étude est traitée au cours de l'année 2022-2023.

## 1.7. SUBDIVISION DU TRAVAIL

Ce présent travail se subdivise en quatre chapitres suivants :

Le premier chapitre porte sur introduction générale du sujet : ce chapitre de ce travail a pour objectif de présenter le contexte général du sujet de la production de rapports météorologiques. Nous allons tout d'abord introduire la problématique liée à ce domaine, en mettant en lumière les besoins des utilisateurs et les défis auxquels ils sont confrontés. Nous expliquerons également les objectifs de cette étude, en détaillant les questions de recherche que nous chercherons à résoudre tout au long de ce travail, dégager les méthodes et techniques utilisées, le choix et intérêt du sujet, sa délimitation spatio-temporelle et enfin une subdivision du travail.

Le deuxième chapitre de ce travail de recherche se concentrera sur l'étude théorique et la revue de la littérature. Nous allons commencer par présenter les concepts clés liés à la production de rapports météorologiques, en mettant en évidence les différents types de données météorologiques et les outils utilisés pour les collecter et les traiter. Ensuite, nous allons examiner les recherches existantes dans ce domaine, en discutant des différentes approches et méthodologies utilisées par les chercheurs pour résoudre les problèmes liés à la production de rapports météorologiques.

Le troisième chapitre porte sur la modélisation du système avec UML : chapitre dans lequel certains concepts de base d'UML seront présentés en expliquant comment ce langage de modélisation peut être utilisé pour représenter les différents aspects du système de production de rapports météorologiques. Nous allons ensuite décrire les différents diagrammes UML utilisés dans ce mémoire, en montrant comment ils peuvent être utilisés pour modéliser les différents composants du système.

Le quatrième chapitre porte sur la présentation du système réalisé : Chapitre qui sera consacré à la présentation du système réalisé. Nous allons tout d'abord présenter l'architecture globale du système, en décrivant les différentes interfaces qui le composent et en montrant comment ils interagissent les uns avec les autres. Ensuite, nous allons présenter les différentes fonctionnalités du système, en montrant comment elles répondent aux besoins des utilisateurs.

## CHAPITRE DEUXIEME :

### REVUE DE LA LITTÉRATURE ET ÉTUDE THEORIQUE

#### 2.1. Introduction

Dans ce chapitre nous allons présenter notre milieu d'étude et définir quelques Concepts fondamentaux sur lequel se base notre étude.

#### 2.2. Revue empirique

Nous ne sommes pas les premiers à avoir pensé et statuer sur ce genre de Travaux, nos aînés y ont pensé et ont abordés le sujet presque de ce domaine tels que :

Sofiane & Ismail (2017) dans leur travail portant sur « *conception et réalisation d'un système d'agriculture intelligente* » dans leur travail ils ont proposé un système de contrôle et d'irrigation intelligent. Le but principal de leur système est d'aider aux simple agriculteur, avec un équipement très simple, facile à manipuler et en plus avec un prix imbattable. L'objectif de leur système est de contrôler intelligemment une zone agricole et d'apporter un suivi et un automatisme sans un effort conséquent.

Brahimi (2019) dans son travail « *réalisation d'un système basé sur internet des objets pour le contrôle des serres intelligentes* » il a proposé dans ce mémoire la réalisation d'un système des serres intelligentes. Ce système permet de contrôler intelligemment un ensemble des serres en vue de garantir une bonne performance, de réduire les efforts des agriculteurs et d'améliorer la productivité en termes de quantité et de qualité. Pour des raisons pratiques nous proposons une hybridation entre un contrôle automatique et semi-automatique. Basé sur l'utilisation des réseaux de neurones et des arbres de décision, le contrôle automatique permet au système de réagir d'une manière intelligente pour établir les conditions climatologiques idéales pour les plantes.

Siriman (2019) Dans son travail « *IoT appliqué dans l'Agriculture* » ; dans son travail son projet consiste à appliquer l'IOT (l'internet des objets) dans l'agriculture. Pour cela il a créé un réseau de capteur dans un milieu agricole établi pour collecter des données à partir de différents capteurs déployés à divers nœuds et les envoyer via le protocole sans fil. Les données collectées fournissent des informations sur les différents facteurs environnementaux du milieu.

Mohamed, Hamadouch, Rabie, & Hammelna (2022) Dans leur travail « *réalisation d'une station météorologique intégrées au système photovoltaïque* » ; l'objectif de leur travail est de

réaliser une station météo qui peut fournir les données et les informations de manière simple à toutes les personnes qui veulent être informées sur la situation météorologique d'une manière régulière.

Abdelouahebm & Haithem (2021) Dans leur travail « *Etude et réalisation d'un système de mesure et sauvegarde des paramètres météorologiques* » l'objectifs de leur projet visés trois objectifs sont :

- Le premier concerne les définitions et la collecte d'informations suffisantes sur les stations météorologiques en général.
- Le deuxième consiste à réaliser un appareil basé sur une carte Arduino capable de mesurer les grandeurs nécessaires liées au fonctionnement d'une station météorologique.
- Le troisième est de donner la démarche algorithmique d'un tel programme qui affiche toutes les grandeurs mesurées avec leur sauvegarde dans une carte mémoire sous un fichier de type EXCEL selon l'environnement IDE de l'Arduino.

Dans le cadre notre travail nous allons réaliser un prototype simple d'un système de rapport météorologiques pour un champ connecté permettant d'acquérir des données à l'aide de différents capteurs : température, humidité., associe avec un module Esp8266 Wi-Fi. Ils sont connectés à une carte électronique équipée d'une unité de traitement basée sur Arduino qui à l'aide d'une API va transférer les données vers une base donnée et un site internet qui va afficher les données sous formes des graphiques.

## 2.3. Revue théorique

### 2.3.1. Internet des objets



Source : <https://fr.freepik.com/photos-vecteurs-libre/affiche-electromenager/5>  
Figure 1 : Internet des objets

#### 2.3.1.1. Définition

L'Internet des Objets est un réseau de réseaux qui permet, via des systèmes d'identification électronique normalisés et unifiés, et des dispositifs mobiles sans fil, d'identifier directement et *sans* ambiguïté des entités numériques et des objets physiques et ainsi de pouvoir récupérer, stocker, transférer et traiter, sans discontinuité entre les mondes physiques et virtuels, les données s'y rattachant (Benghozi, et al., 2008).

L'Internet des Object, abrégé généralement IoT à cause de sa formulation anglaise (Internet of Things), est un néologisme utilisé pour la première fois en 1999 par Kevin Ashton, co-fondateur et directeur de l'Auto-ID Center qui référence tous ces appareils (en dehors des ordinateurs et des smartphones) connectés à Internet. Qu'il s'agisse de voitures, de bracelets de fitness, de réfrigérateurs, de systèmes électriques ou d'autres choses encore, tous sont connectés à Internet et nous permettent d'échanger des données grâce aux capteurs (Murgia, 2017).

En bref, l'Internet des objets est l'évolution naturelle de l'utilisation des réseaux : elle a pour objectif de rendre le monde réel plus intelligent grâce à la connexion des objets, ces derniers obtiennent des informations qu'elles transmettent par réseau. N'importe quel objet autonome qui peut être connecté à Internet et qui peut être utilisé à distance peut être considéré comme un

membre de la famille de l'Internet des objets. N'importe quel objet physique disposant d'une adresse IP et permettant de communiquer grâce à un réseau est donc l'un des objets de l'Internet des objets (Murgia, 2017).

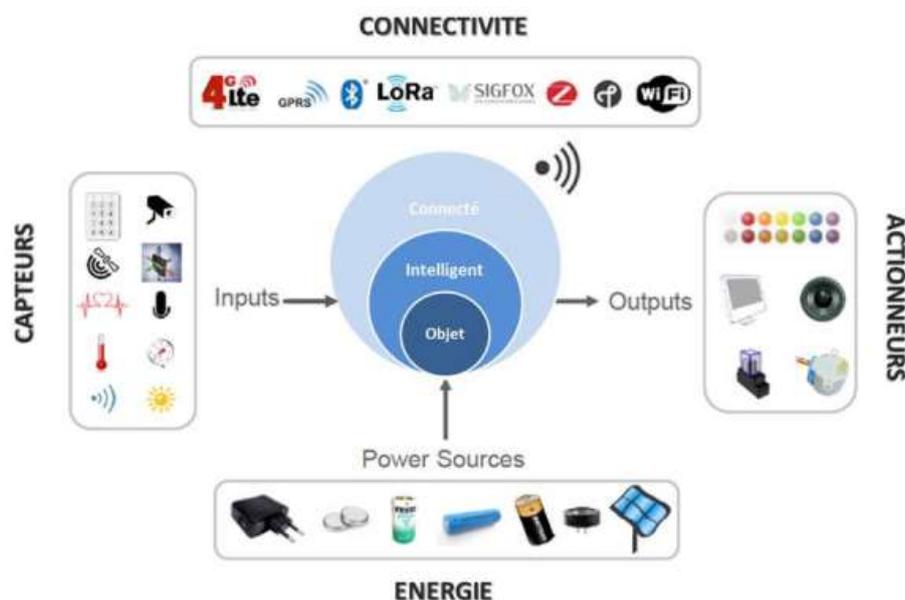
### 2.3.1.2. Les caractéristiques d'IoT

L'IoT possède les caractéristiques essentielles suivantes :

- Identité unique : chaque objet connecté possède une identité unique.
- Dynamique et auto-adaptatif : les objets connectés peuvent avoir la capacité de s'adapter aux changements du contexte et de prendre des décisions basées sur les conditions de fonctionnement.
- Auto configuration : les objets connectés peuvent avoir des capacités d'auto configuration pour permettre à un nombre large d'objets de fonctionner ensemble afin de fournir un service. Ces objets ont la capacité d'établissement de réseau, de mis à jour software, etc.
- Interopérabilité : les objets connectés sont des objets hétérogènes basés sur des plateformes matérielles différentes et peuvent supporter un nombre de protocoles de communication et communiquer ensemble et avec l'infrastructure.

### 2.3.1.3. Les composants de base de l'IoT

Un système d'IoT est constitué de plusieurs composants technologiques (voir la Figure 2), dont nous citons :



Source : [https://www.connectwave.fr/wp-content/uploads/2018/07/fonctions\\_objet\\_connecte.jpg](https://www.connectwave.fr/wp-content/uploads/2018/07/fonctions_objet_connecte.jpg)

Figure 2: Les Composants d'IoT

### 2.3.2. Les capteurs

Un capteur est un dispositif ayant pour tâche de transformer une mesure physique observée en une mesure généralement électrique qui sera à son tour traduite en une donnée binaire exploitable et compréhensible par un système d'information (Biseul, 2017).

Les capteurs sont de deux catégories :

- Les capteurs numériques : Les capteurs numériques donnent en sortie une valeur finie. Par exemple, si une grandeur physique croît de manière linéaire, la sortie du capteur qui va la mesurer donnera soit une information du type "TOR " (Tout Ou Rien), un train d'impulsion ou un échantillonnage (Rafai & Kouah, 2018).
- Les capteurs analogiques : Les capteurs analogiques servent à transformer une grandeur physique en un autre type de variation d'impédance, de capacité, d'inductance ou de tension. Un signal est dit analogique si l'amplitude de la grandeur physique qu'il représente peut prendre une infinité de valeurs dans un intervalle donné (Rafai & Kouah, 2018).

### 2.3.3. L'agro météorologie

#### 2.3.3.1. Définition

La météorologie agricole au sens large est cette branche de la science appliquée la météorologie qui étudie les réponses des organismes vivants à l'environnement physique. Le but ultime est d'améliorer l'agriculture production par des prévisions plus précises et par la maîtrise l'environnement. Les prévisions peuvent aller des prévisions de rendement des cultures et qualité d'une part à l'estimation de la production animale et aléas climatiques d'autre part. Contrôle de l'environnement physique peut inclure la prévention du gel, le contrôle des inondations et la régulation de la température des étables.

Au sens étroit, la météorologie agricole peut être définie comme l'étude des processus physiques dans l'atmosphère qui produisent le temps, en relation avec la production agricole. C'est une science horizontale qui applique la physique de l'air et du sol à l'agriculture. En fait, de nombreux chercheurs dans ce domaine pensent que les recherches sur le microclimat des plantes et des animaux ainsi que les statistiques des éléments météorologiques sont à proprement parler des sujets de météorologie agricole. Cependant, nous souligner le sens large de la définition, qui est l'étude des réponses des organismes vivants à l'environnement physique, car c'est le lien entre la météorologie et l'agriculture, et est le fondement aspect du sujet (Jin-Yuan, 1963).

### 2.3.3.2. Le rôle de la météorologie agricole

La météorologie agricole combine les sciences physiques et biologiques et sert de pont important entre elles. L'agriculture a de plus en plus utilisé la météorologie au cours des dernières décennies. Dans des environnements contrôlés, les réactions biologiques ont été mesurées dans des recherches en laboratoire, en serre et sur le terrain, qui en sont principalement responsables. Les météorologues agricoles sont principalement responsables de la traduction des résultats de laboratoire et de serre en plein champ. Les tests empiriques sur le terrain ont été le plus fréquemment utilisés jusqu'à présent pour mettre en œuvre les résultats de laboratoire sur le terrain.

Voici quelques exemples importants :

1. Prévision et protection contre le gel,
2. Avertissement d'incendie de forêt,
3. Conservation et contrôle de l'irrigation,
4. Programmation des dates de plantation et de récolte,
5. Sélection des sites de culture et de construction,
6. Insectes contrôle,
7. Le contrôle des maladies et de nombreuses autres modifications microclimatiques telles que l'utilisation de pratiques d'abri contre le vent (Jin-Yuan, 1963).

### 2.3.4. Arduino

Arduino est un microcontrôleur open source qui peut être facilement programmé, effacé et reprogrammé à tout instant. Introduite en 2005 ;

Le projet Arduino est issu d'une équipe d'enseignants et d'étudiants de l'école de Design d'Interaction d'Ivrea en Italie composée de Massimo Banzi, David Mellis, Tom Igoe, Gianluca Martino, David Curatelles et Nicholas Zambetti. La plate-forme Arduino a été conçue pour fournir un moyen peu coûteux et facile pour les amateurs, les étudiants et les professionnels de créer des appareils qui interagissent avec leur environnement à l'aide de capteurs et d'actionneurs. Basé sur un microcontrôleur simple cartes, il s'agit d'une plate-forme informatique open source utilisée pour la construction et la programmation appareils électroniques. Il est également capable d'agir comme un mini-ordinateur, tout comme les autres microcontrôleurs en prenant des entrées et en contrôlant les sorties pour une variété d'appareils électroniques.



Figure 3: Microcontrôleur



Figure 4: Microcontrôleur

Il est également capable de recevoir et d'envoyer des informations sur Internet à l'aide de divers Boucliers Arduino, qui sont discutés dans cet article. Arduino utilise un matériel connu sous le nom d'Arduino carte de développement et logiciel pour développer le code connu sous le nom d'Arduino IDE (Integrated Environnement de développement). Construit avec les microcontrôleurs Atmel AVR 8 bits qui sont fabriqués par Atmel ou un Atmel ARM 32 bits, ces microcontrôleurs peuvent être programmés facilement en utilisant le langage C ou C++ dans l'IDE Arduino.

Contrairement aux autres cartes de microcontrôleur en Inde, les cartes Arduino sont entrées sur le marché électronique il y a seulement quelques années, et étaient limités à des projets à petite échelle. Les personnes associées à l'électronique arrivent progressivement et accepte le rôle d'Arduino pour ses propres projets (Leo, 2018).

Cette carte de développement peut également être utilisée pour graver (télécharger) un nouveau code sur la carte en utilisant simplement un câble USB pour télécharger. L'IDE Arduino fournit

une plate-forme intégrée simplifiée qui peut fonctionner sur des ordinateurs personnels ordinaires et permet aux utilisateurs d'écrire des programmes pour Arduino en utilisant C ou C++ (Leo, 2018).

### Anatomie d'une carte Arduino

Bien que toutes les cartes Arduino diffèrent les unes des autres, il existe plusieurs composants clés qui peuvent être trouvés sur pratiquement n'importe quel Arduino.

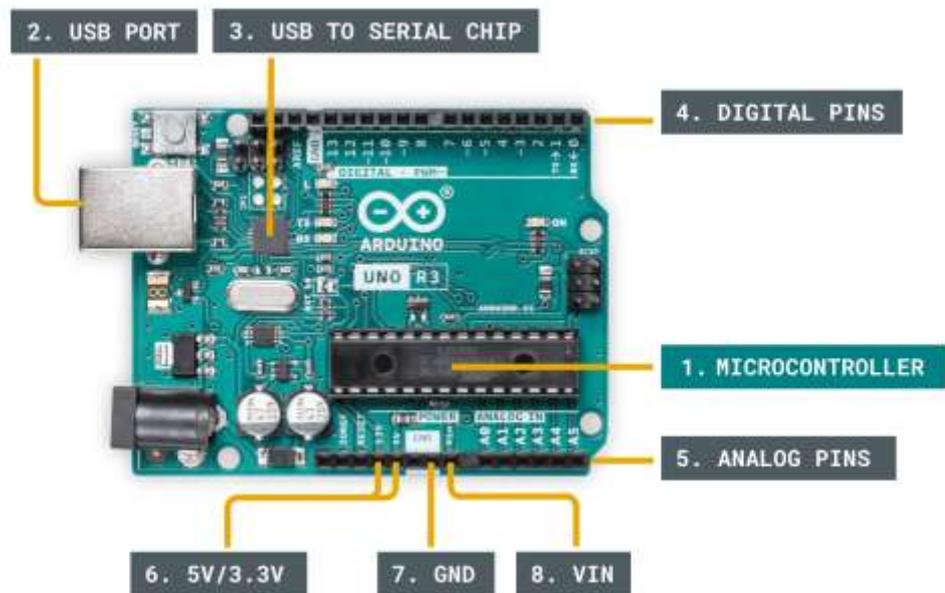


Figure 5 : Composants clés d'une carte Arduino.

1. Microcontrôleur : c'est le cerveau d'un Arduino, et c'est le composant dans lequel nous chargeons des programmes. Considérez-le comme un petit ordinateur, conçu pour exécuter uniquement un nombre spécifique de choses.
2. Port USB : utilisé pour connecter votre carte Arduino à un ordinateur.
3. Puce USB vers série : l'USB vers série est un composant important, car il aide à traduire les données provenant par ex. un ordinateur au microcontrôleur embarqué. C'est ce qui permet de programmer la carte Arduino depuis votre ordinateur.
4. Broches numériques : broches qui utilisent la logique numérique (0,1 ou LOW/HIGH). Couramment utilisé pour les interrupteurs et pour allumer/éteindre une LED.
5. Broches analogiques : broches pouvant lire des valeurs analogiques dans une résolution de 10 bits (0-1023).
6. Broches 5V / 3.3V : Ces broches sont utilisées pour alimenter des composants externes.

7. GND : également connu sous le nom de masse, négatif ou simplement -, est utilisé pour compléter un circuit, où le niveau électrique est à 0 volt.
8. VIN : signifie Voltage In, où vous pouvez connecter des alimentations externes. (Arduino, 2023)

La plupart des cartes Arduino sont conçues pour avoir un seul programme exécuté sur le microcontrôleur. Ce programme peut être conçu pour effectuer une seule action, comme faire clignoter une LED. Il peut également être conçu pour exécuter des centaines d'actions dans un cycle. La portée varie d'un programme à l'autre.

Le programme qui est chargé dans le microcontrôleur commencera son exécution dès qu'il sera alimenté. Chaque programme a une fonction appelée "boucle". A l'intérieur de la fonction Loop, vous pouvez par exemple :

- ✓ Lire un capteur.
- ✓ Allumez une lumière.
- ✓ Vérifiez si une condition est remplie.
- ✓ Etc.

La vitesse d'un programme est incroyablement rapide, sauf si nous lui disons de ralentir. Cela dépend de la taille du programme et du temps qu'il faut au microcontrôleur pour l'exécuter, mais c'est généralement en microsecondes (un millionième de seconde).

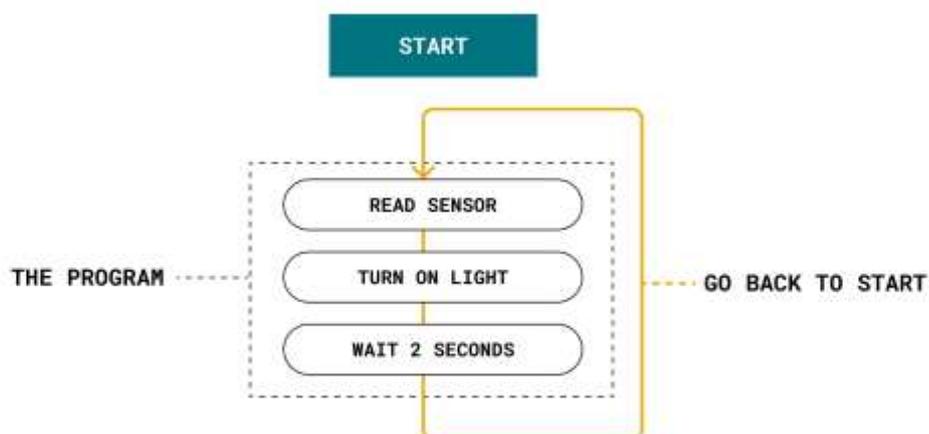


Figure 6 : Le fonctionnement de base d'un Arduino.

## TYPE DE CARTES ARDUINO

La catégorie d'entrée de gamme d'Arduino contient les cartes de microcontrôleur que la plupart des bricoleurs choisissent d'utiliser pour leurs projets, car elles offrent des fonctionnalités simples et sont accompagnées d'une documentation abondante. Cela signifie également qu'ils peuvent ne pas disposer des fonctionnalités de niche fournies avec les cartes Arduino améliorées et IoT. Comme les autres modèles Arduino, ils sont programmables en C ou C++ via l'IDE Arduino, mais peuvent également être utilisés avec une variété de langages de programmation.

### ➤ Arduino Uno R3



Figure 7 : Arduino Uno R3

Basée sur la puce ATmega328P 16 MHz, cette carte comporte 14 broches d'entrée/sortie numériques, 6 broches analogiques et un en-tête ICSP (programmation série en circuit), et est capable de fournir jusqu'à 5 V aux composants qui y sont connectés. Cette carte est relativement grande et présente l'inconvénient d'utiliser un connecteur USB-B pour s'interfacer avec des ordinateurs, mais cela ne signifie pas qu'elle ne peut pas être utilisée pour des projets qui élargissent vos compétences, et elle peut être un excellent choix pour quiconque se lance pour la première fois dans la programmation / l'électronique.

**Spécifications de base :**

- Microcontrôleur : ATmega328P
- Mémoire : 2 Ko SRAM, 32 Ko flash et 1 Ko EEPROM
- Communications : UART, IC2 et SPI
- Particularité(s): Puce remplaçable

➤ **Arduino Léonard**

Figure 8 : Arduino Leonardo

L'Arduino Leonardo est livré avec essentiellement les mêmes spécifications qu'un Uno R3, seulement il dispose d'un connecteur micro-USB, a 20 broches numériques et 17 analogues, et a la capacité d'être utilisé comme un dispositif d'interface humaine grâce à la puce ATmega32U4 qui alimente-il. Cela signifie que votre Leonardo peut être utilisé comme clavier ou souris avec un câble USB connecté.

Ce type d'Arduino est idéal pour les projets simples qui doivent interagir avec des machines comme les ordinateurs, offrant une vaste gamme d'idées différentes à essayer par vous-même.

**Spécifications de base :**

- Microcontrôleur : ATmega32U4
- Mémoire : 2,5 Ko SRAM, 32 Ko flash et 1 Ko EEPROM
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : Connectivité HID

➤ Arduino Nano

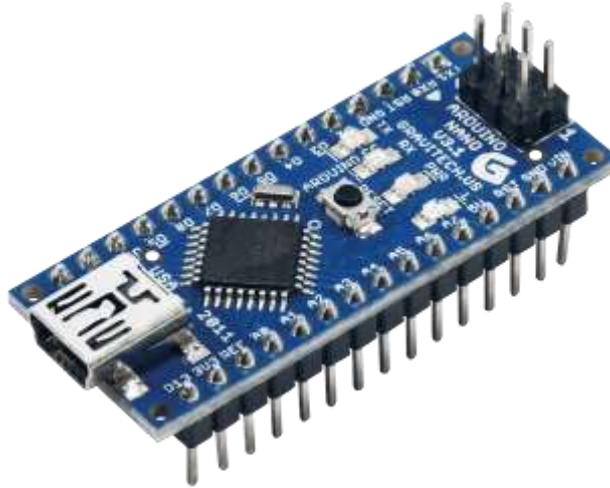


Figure 9 : Arduino nano

Les Arduino Nano et Nano Every sont les plus petites cartes à microcontrôleur proposées par la société. Les deux cartes présentent la même disposition de broches, avec 14 broches numériques et 8 broches analogiques, bien que le Nano Every dispose d'une puce de microcontrôleur plus robuste et d'une mémoire de programme améliorée. Ces cartes sont toutes deux livrées avec des en-têtes pré-soudés qui les rendent idéales pour une utilisation avec des planches à pain, mais elles n'ont pas la prise d'alimentation fournie sur les cartes plus grandes.

Leur compatibilité avec les planches à pain rend ces petites cartes idéales pour ceux qui aiment créer des circuits qui changent tout le temps, comme les professeurs d'école et les fabricants de prototypes.

***Spécifications de base :***

- Microcontrôleur : ATmega32U4 (Nano) ; ATmega4809 (Nano chaque)
- Mémoire : 2 Ko SRAM, 32 Ko flash et 1 Ko EEPROM (Nano) ; SRAM 6 Ko, mémoire flash 48 Ko et EEPROM 256 Go (Nano Every)
- Communications : UART, IC2 et SPI
- Particularité(s): Compatible avec les planches à pain et extrêmement petit

➤ Micro Arduino

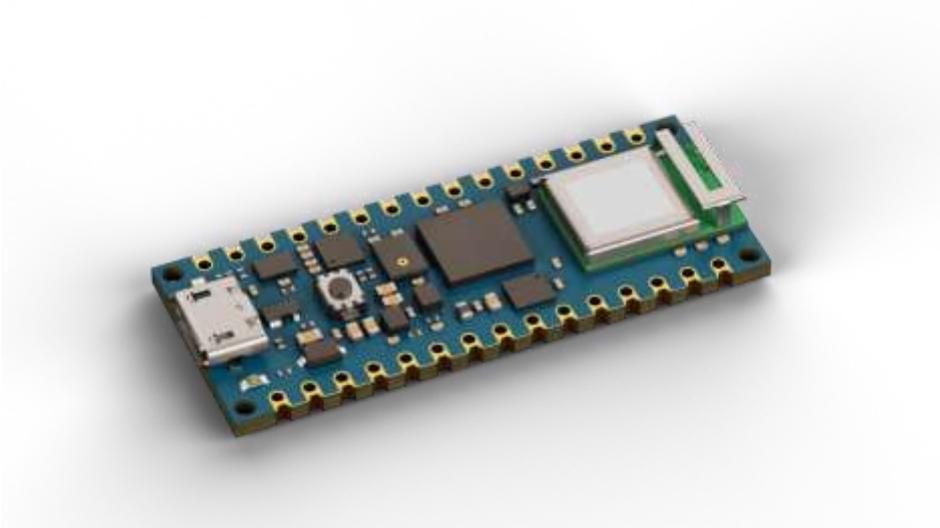


Figure 10: Arduino Micro

L'Arduino Micro possède des caractéristiques très similaires à celles du Leonardo, seule la carte est beaucoup plus petite et ne comporte que 12 broches analogiques aux côtés de ses 20 broches numériques. Avec seulement 18 mm de large et 48 mm de long, cette carte est l'une des plus petites qu'Arduino ait jamais fabriquées, ce qui la rend idéale pour créer un clavier, une souris et/ou d'autres appareils HID qui doivent être minuscules.

***Spécifications de base :***

- Microcontrôleur : ATmega32U4
- Mémoire : 2,5 Ko SRAM, 32 Ko flash et 1 Ko EEPROM
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : connectivité HID et petit facteur de forme

Cartes Arduino améliorées

- Arduino Nano 33 BLE / Nano 33 BLE Sense



Figure 11: Arduino Nano 33 BLE / Nano 33 BLE Sense

Le Nano 33 BLE / Nano 33 BLE Sense est conçu comme une version améliorée de l'Arduino Nano / Nano Every, avec la même disposition des broches pour le rendre agréable et facile pour les bricoleurs. Les deux cartes ont un processeur Arm Cortex-M4 32 bits fonctionnant à 64 MHz intégré à leurs puces nRF52840, avec 1 Mo de mémoire flash et 256 Ko de SRAM, ce qui rend ces cartes incroyablement puissantes malgré leur petite taille.

Ils ne sont livrés qu'avec 14 broches numériques, mais sont dotés d'une foule de capteurs qui ne sont pas fournis avec les Nanos ordinaires. Ce réseau de capteurs comprend un accéléromètre, un gyroscope et un magnétomètre avec une résolution à 3 axes, et la carte est livrée avec Bluetooth Low Energy (BLE) qui facilite la transmission des données qu'elle collecte.

Outre toutes ces fonctionnalités exceptionnelles, le Nano 33 BLE Sense est également capable d'exécuter des applications informatiques de pointe à l'aide de modèles d'apprentissage automatique de TensorFlow Lite.

Spécifications de base :

- Microcontrôleur : nRF52840
- Mémoire : 256 ko SRAM et 1 Mo de mémoire flash
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : Capteurs, Bluetooth et IA (Sense uniquement)

- Arduino MKR Zero

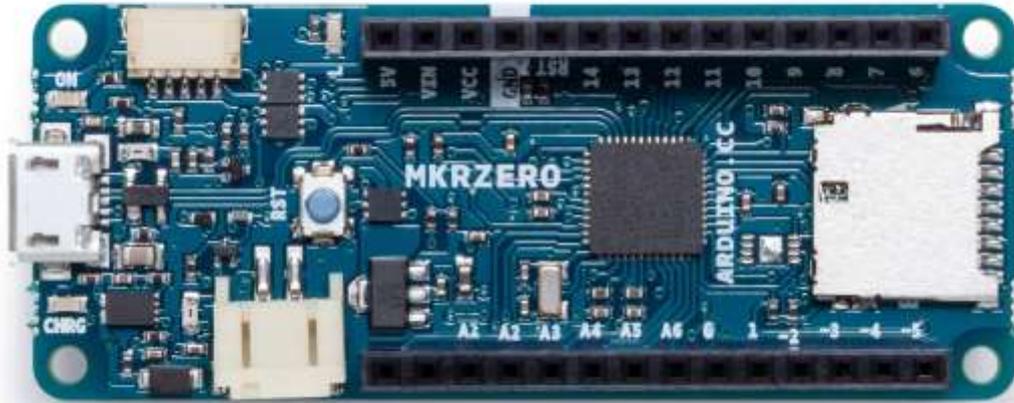


Figure 12: Arduino MKR Zero

L'Arduino MKR Zero est conçu pour la création musicale et d'autres projets complexes, avec un puissant processeur Arm Cortex-M0 32 bits SAMD21, une prise en charge native de la batterie et un lecteur de carte microSD intégré. La carte est livrée avec 8 broches numériques, 7 broches d'entrée analogiques et 1 broche de sortie analogique. Grâce au stockage extensible fourni avec cette carte, c'est une excellente option pour ceux qui travaillent avec beaucoup de code et une gamme de composants matériels.

Spécifications de base :

- Microcontrôleur : Arm Cortex-M0 SAMD21 32 bits
- Mémoire : 32 Ko SRAM et 256 Ko flash
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : connecteur de batterie intégré, lecteur de carte microSD, matériel puissant

## - Arduino Mega 2560 Rz



Figure 13: Arduino Mega 2560 Rz

L'Arduino Mega 2560 est similaire à un Arduino Uno, sauf qu'il comporte 54 broches numériques, 16 broches analogiques et 4 ports série, tout en étant beaucoup plus grand et plus puissant que l'Uno standard. Cette carte est idéale pour les bricoleurs qui ont besoin de plus de broches, de mémoire ou de puissance de traitement sans sacrifier les fonctionnalités fournies avec les Arduinos classiques.

Spécifications de base :

- Microcontrôleur : ATmega2560
- Mémoire : 8 Ko SRAM, 256 Ko flash et 4 Ko EEPROM
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : grand facteur de forme et ports série

## - IoT Arduino Boards

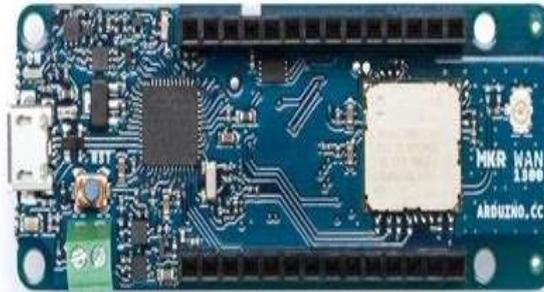


Figure 14: IoT Arduino Boards

La gamme IoT d'Arduino est axée sur la connectivité. Chacune des cartes IoT proposées par Arduino permet de se connecter à une variété d'outils de communication différents, du World Wide Web aux réseaux de téléphonie mobile, et tout cela peut être fait avec une carte qui tient dans la paume de votre main. Cela vous permet de créer une large gamme de projets Arduino IoT. Nous nous concentrerons sur la gamme MKR de cartes Arduino IoT, mais vous pouvez trouver plus d'options connectées en ligne si votre projet a des exigences avancées.

Basé sur la même disposition et conception que le MKR Zero, chacune des cartes ci-dessous offre une fonctionnalité de connectivité unique, ce qui les rend idéales pour les projets qui nécessitent des capacités de transfert de données.

Arduino MKR WiFi 1010 : Cette carte est capable de se connecter aux réseaux Wi-Fi à l'aide d'un module sans fil NINA-W10 2,4 GHz.

Arduino MKR GSM 1400 : cette carte dispose d'un emplacement pour carte SIM, lui permettant de se connecter aux opérateurs GSM du monde entier.

Arduino MKR Fox 1200 : Le MKR Fox peut se connecter à Sigfox, un service basé sur le cloud qui permet de gérer les communications à faible consommation sur des machines centralisées.

Arduino MKR 1000 : il s'agit de la carte MKR d'entrée de gamme de la gamme, offrant un moyen simple et facile de démarrer avec des projets IoT.

Spécifications de base :

- Microcontrôleur : Arm Cortex-M0 SAMD21 32 bits
- Mémoire : 32 Ko SRAM et 256 Ko flash
- Communications : UART, IC2 et SPI
- Caractéristiques spéciales : diverses fonctionnalités IoT

### 2.3.4. Api

Une API (Application Programming Interface) est un ensemble de règles et de protocoles qui spécifient comment des logiciels doivent interagir entre eux. Elle définit les méthodes, les classes, les structures de données et les conventions qu'un développeur peut utiliser pour développer des applications et intégrer des fonctionnalités fournies par une plateforme, un Framework ou un service tiers.

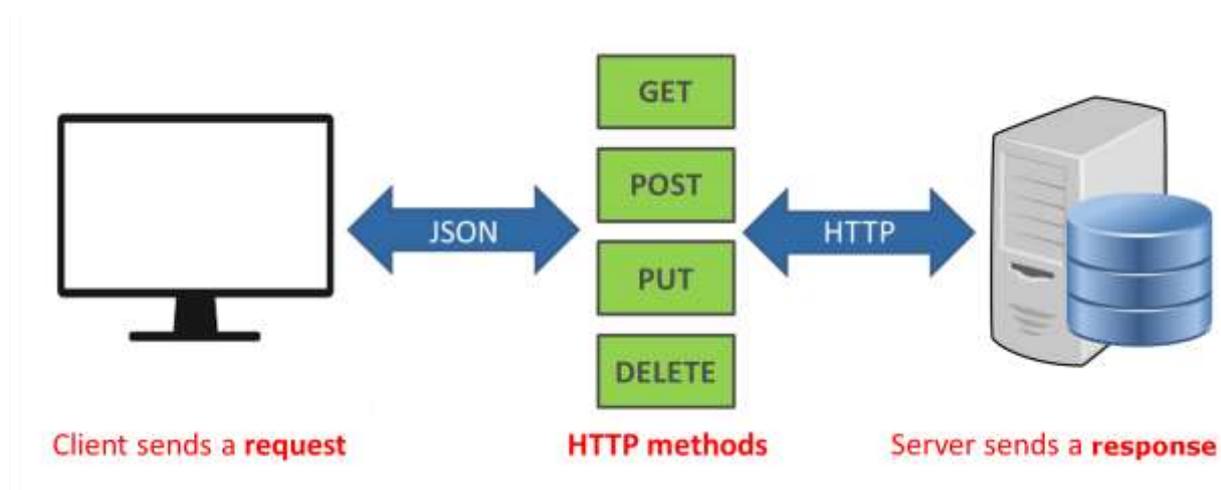


Figure 15 : API

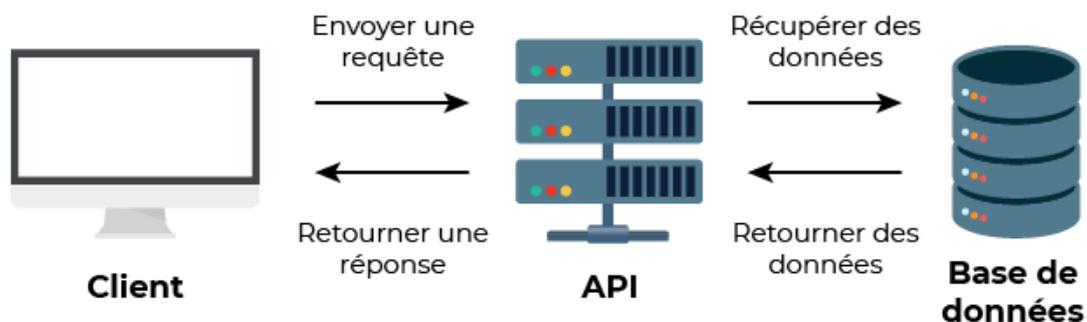


Figure 16: Api avec interaction avec une base de données

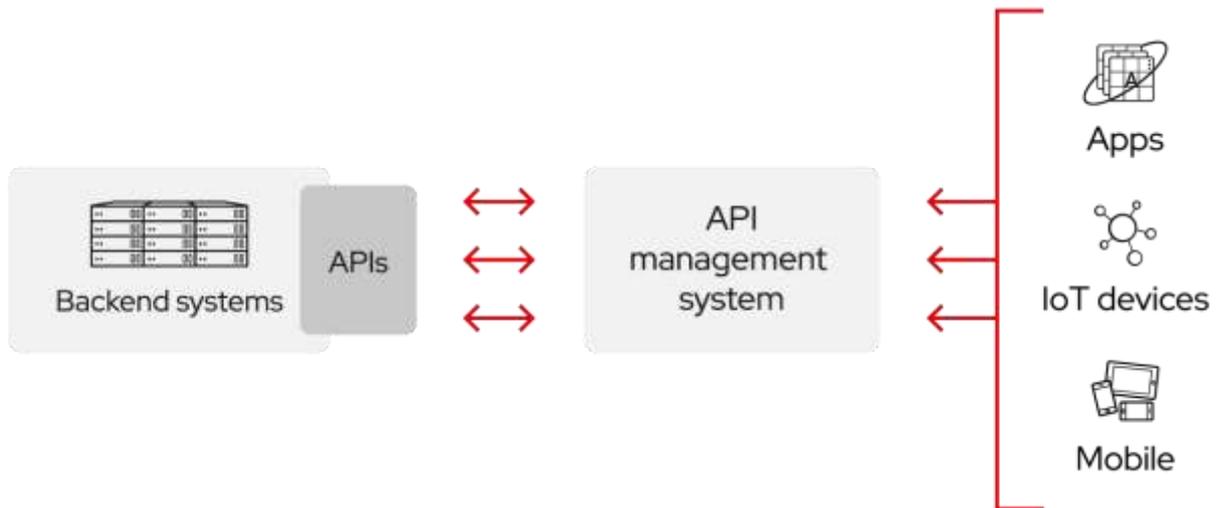


Figure 17 : Api avec interaction avec une base de données

### 2.3.4. Reactjs

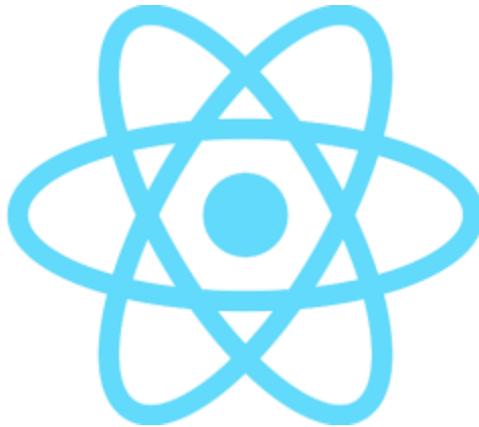


Figure 18: Logo officiel de React.js

React.js, communément appelé simplement React, est une bibliothèque JavaScript utilisée pour construire des interfaces utilisateur. Chaque application web React est composée de composants réutilisables qui constituent des parties de l'interface utilisateur – nous pouvons avoir un composant distinct pour notre barre de navigation, un pour le pied de page, un autre pour le contenu principal, et ainsi de suite. Vous comprendrez mieux cela lorsque nous arriverons à la section où nous devons travailler avec des composants (Kinsta, 2023).

#### ***Cas d'utilisation de React***

React est généralement utilisé pour construire l'interface utilisateur (frontend) des applications web. Il s'accompagne d'un rendu rapide des pages et de performances accrues. React peut être utilisé pour construire n'importe quel produit qui fonctionne sur le web.

Voici quelques-unes des choses que React est généralement utilisé pour construire :

- Applications de lecteur de musique
- Applications de réseaux sociaux
- Applications de discussion en temps réel
- Applications web complètes
- Applications de commerce\_électronique
- Applications de météo
- Applications de liste de choses à faire
- Etc.

#### 2.3.4. Nodejs

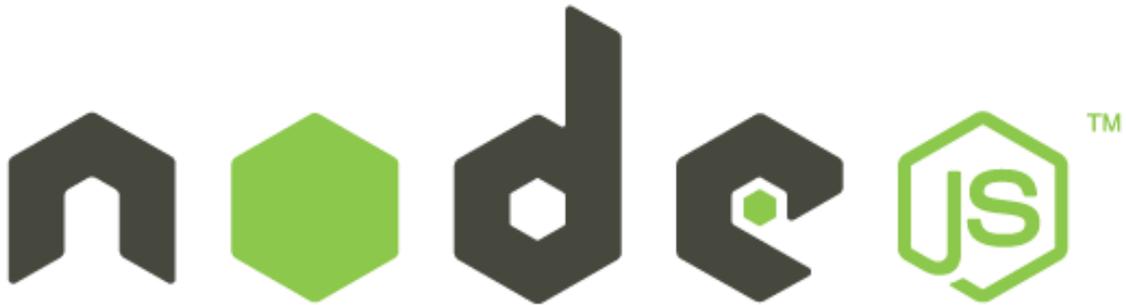


Figure 19 : Logo officiel de Nodejs

Node.js est un environnement d'exécution côté serveur basé sur le moteur JavaScript V8 de Google Chrome. Il permet d'exécuter du code JavaScript en dehors d'un navigateur, ce qui le rend idéal pour développer des applications Web côté serveur, des outils en ligne de commande et des applications réseau.

Node intègre le moteur JavaScript V8, le même que celui utilisé dans Google Chrome et d'autres navigateurs. Il est écrit en C++ et peut fonctionner sur MacOS, Linux, Windows et d'autres systèmes. Le moteur analyse et exécute le code JavaScript. Il peut fonctionner indépendamment d'un environnement de navigateur, soit intégré dans une application C++, soit implémenté en tant que programme autonome. Le moteur V8 compile JavaScript en interne, en utilisant des processus juste-à-temps (JIT) pour accélérer l'exécution.

### 2.3.4. UML

UML: Unified Modeling Language; ou langage de modélisation unifié en français.



Figure 20: Logo UML

#### **Qu'est-ce que le langage UML ?**

UML ou Unified Modeling Language est un langage de modélisation maintenu en tant que standard par l'OMG depuis 1997. Le langage de modélisation unifié (UML) fournit des mécanismes standard de l'industrie pour visualiser, spécifier, construire et documenter des systèmes logiciels (Eriksson, et al., 2003).

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement.

L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie. Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent. L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML.

L'UML a une relation directe avec l'analyse et la conception orientées objet.

## **Histoire et origines de l'UML**

Les méthodes utilisées dans les années 1980 pour organiser la programmation impérative (notamment Merise) étaient fondées sur la modélisation séparée des données et des traitements. Lorsque la programmation par objets prend de l'importance au début des années 1990, la nécessité d'une méthode qui lui soit adaptée devient évidente. Plus de cinquante méthodes apparaissent entre 1990 et 1995 (Booch, Classe-Relation, Fusion, HOOD, OMT, OOA, OOD, OOM, OOSE, etc.) (Audibert, 2009).

Actuellement UML est à la version 2.5.1 qui est sorti en 2017.

## **CHAPITRE TROISIEME : APPROCHE METHODOLOGIQUE DE L'ETUDE**

### **3.1. Introduction**

Dans ce chapitre nous allons parler en grande partie sur plan de notre projet, l'estimation de coût de ce projet informatique ainsi que la modélisation du système avec UML.

### **3.2. PLANNING PREVISIONNE**

#### **3.2.1. DETERMINATION DES OBJECTIF**

La conduite de projet est représentée comme une pyramide dont le sommet le système pilotage du projet au travers des trois types de gestion à mettre en œuvre : gestion du temps, gestions des ressources et gestion de la production.

Pour atteindre les objectifs, il faut détermine toutes les tâches et la phase constituant ce projet soient exécutées dans le temps et avec les moyens nécessaires. La connaissance des différentes tâches à accomplir ne suffit pas pour réaliser un projet. Il faut encore une parfaite connaissance de l'articulation permettant de le réaliser dans les conditions du coût et de délai imposé. Cependant son déroulement, vérifie constamment si le plan établi est respecté.

#### **3.2.2. METHODE D'ORDONNANCEMENT**

Il existe 3 méthodes d'ordonnancement généralement utilisées :

- La méthode en barre ou Diagramme de GANTT
- La méthode potentielle Métra (MPM) et
- La méthode Program Evaluation and Research Task (PERT)

Dans notre travail nous allons nous focalise au diagramme PERT ; elle va nous permettre de décrire l'enchaînement des tâches en tenant compte des contraintes d'ordonnancement qui le lient. Cette méthode introduit la notion des tâches fictives de durée égale à 0 au début et la fin, la tâche fictive de début reliant toutes les tâches sans prédécesseurs à la tâche fictive de fin reliée sans successeur.

### 3.2.3. DETERMINATION DE TACHES

La première phase pour établir un réseau PERT consiste à déterminer les tâches. Elle consiste à identifier et lister les tâches nécessaires à la construction effective du projet. Chaque tâche est associée à une durée estimée en unité de temps. Le tableau ci-après montre les différentes tâches de notre projet :

*Tableau 1. Détermination des tâches*

<b>CODE</b>	<b>TACHE</b>	<b>DURE EN JOURS</b>	<b>ANTERIORITE</b>
A	Commandes des matériels	14	-
B	Implémentation des matériels	7	A
C	Test du système embarque	2	B
D	Développement de l' API	4	B, C
E	Test de l'api	1	D
F	Corrections d'erreurs	1	E
G	Développement du site internet	7	F, E
H	Intégration de l'API dans le site	3	G
I	Test	1	H
J	Correction d'erreurs	1	I
K	Test général	2	J
L	Formation des utilisateurs	2	K

### 3.2.3. ESTIMATION DU COUT DE LA REALISATIONS DU PROJET

#### 3.2.3.1. MAIN D'ŒUVRE

Tableau 2. Détermination du coût matériel

CODE	TACHE	DURE EN JOURS	ANTERIORITE	NOMBRE DES TRAVAILLEURS	CU en \$	CT en \$
<b>A</b>	Commandes des matériels	14	-	1	20	280
<b>B</b>	Implémentation des matériels	7	A	1	5	35
<b>C</b>	Test du système embarque	2	B	1	5	10
<b>D</b>	Développement de l' API	4	B, C	1	100	400
<b>E</b>	Test de l'api	1	D	1	5	5
<b>F</b>	Corrections d'erreurs	1	E	1	5	5
<b>G</b>	Développement du site internet	7	F,E	1	100	700
<b>H</b>	Intégration de l'API dans le site	3	G	1	50	150
<b>I</b>	Test	1	H	1	5	5
<b>J</b>	Correction d'erreurs	1	I	1	20	20
<b>K</b>	Test général	2	J	1	10	20
<b>L</b>	Formation des utilisateurs	2	K	1	20	40
<b>COUT TOTAL</b>						<b>1670</b>

### 3.2.3.2. COUT MATERIEL

Tableau 3. Détermination du coût matériel

NUMERO	DESIGATION	QUANTITE	PRIX UNITAIRE	PRIX TOTAL
1	Arduino UNO	1	30\$	30\$
2	ESP32	1	10\$	10\$
3	Bouton	3	0.5\$	1.5\$
4	DHT11	1	3\$	3\$
5	BMP180	1	5\$	5\$
6	Photorésistance LDR	1	1\$	1\$
7	anémomètre	1	1\$	1\$
8	La girouette	1	1\$	1\$
<b>Total</b>				52.5

### 3.2.3.3. COUT GLOBAL DU PROJET

Tableau 4. Détermination du coût global

Numéro	Désignation	Montant
1	Main d'œuvre	1670\$
2	Cout des matériels	52.5\$
<b>TOTAL GENERAL</b>		1722.5\$

### 3.2.3. ELABORATION DU GRAPHE PERT

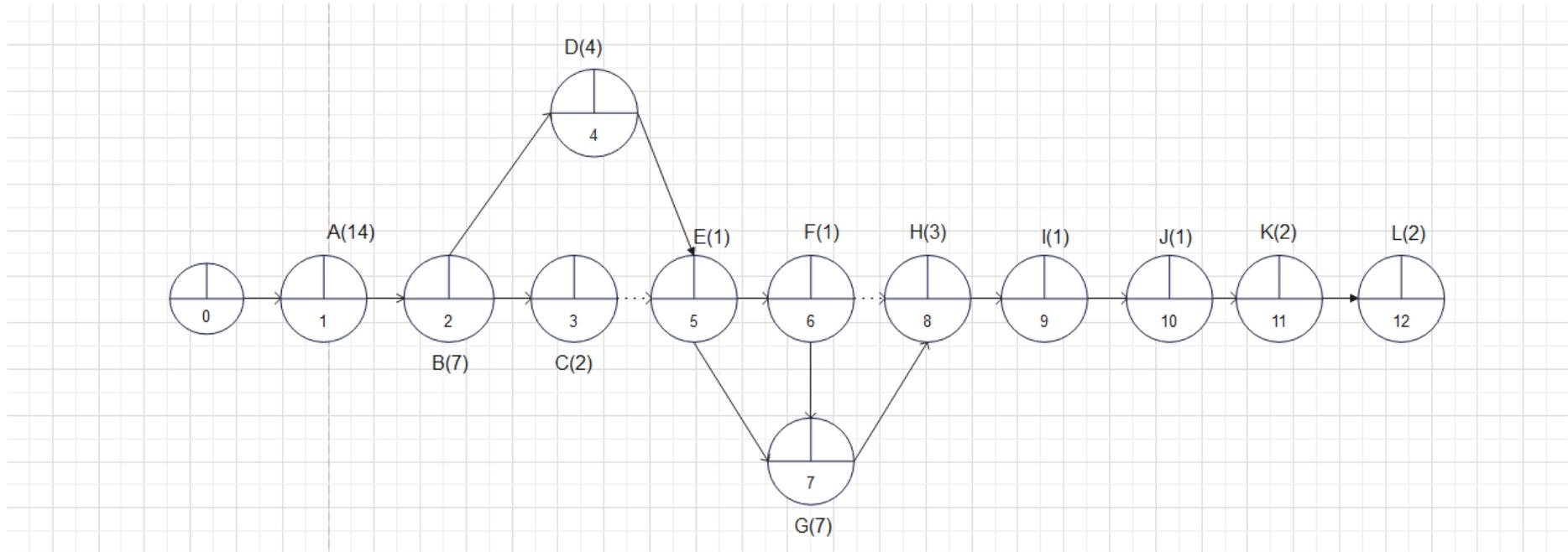


Figure 21 : Elaboration du graphe PERT

### 3.2.4. DETERMINATION DE LA DATE AU PLUS TOT ET LA DATE AU PLUS

#### a) La date au plus tôt d'un réseau PERT

Elle correspond à la date à laquelle une étape peut être atteinte au plus tôt. Elle s'obtient en ajoutant à la date au plus tôt de l'étape précédente, la durée de la tâche qui les sépare :

***Date au plus tôt « étape j » = Date au plus tôt « étape i » + Durée tâche « ij ».***

- Tâche A :  $EST = 0$  (début du projet)
- Tâche B :  $EST = EST(A) + Durée(A) = 0 + 14 = 14$  jours
- Tâche C :  $EST = EST(B) + Durée(B) = 14 + 7 = 21$  jours
- Tâche D :  $EST = \max(EST(B), EST(C)) + Durée(D) = \max(14, 21) + 4 = 25$  jours
- Tâche E :  $EST = EST(D) + Durée(D) = 25 + 4 = 29$  jours
- Tâche F :  $EST = EST(E) + Durée(E) = 29 + 1 = 30$  jours
- Tâche G :  $EST = \max(EST(F), EST(E)) + Durée(G) = \max(30, 29) + 7 = 37$  jours
- Tâche H :  $EST = EST(G) + Durée(G) = 37 + 7 = 44$  jours
- Tâche I :  $EST = EST(H) + Durée(H) = 44 + 3 = 47$  jours
- Tâche J :  $EST = EST(I) + Durée(I) = 47 + 1 = 48$  jours
- Tâche K :  $EST = EST(J) + Durée(J) = 48 + 2 = 50$  jours
- Tâche L :  $EST = EST(K) + Durée(K) = 50 + 2 = 52$  jours

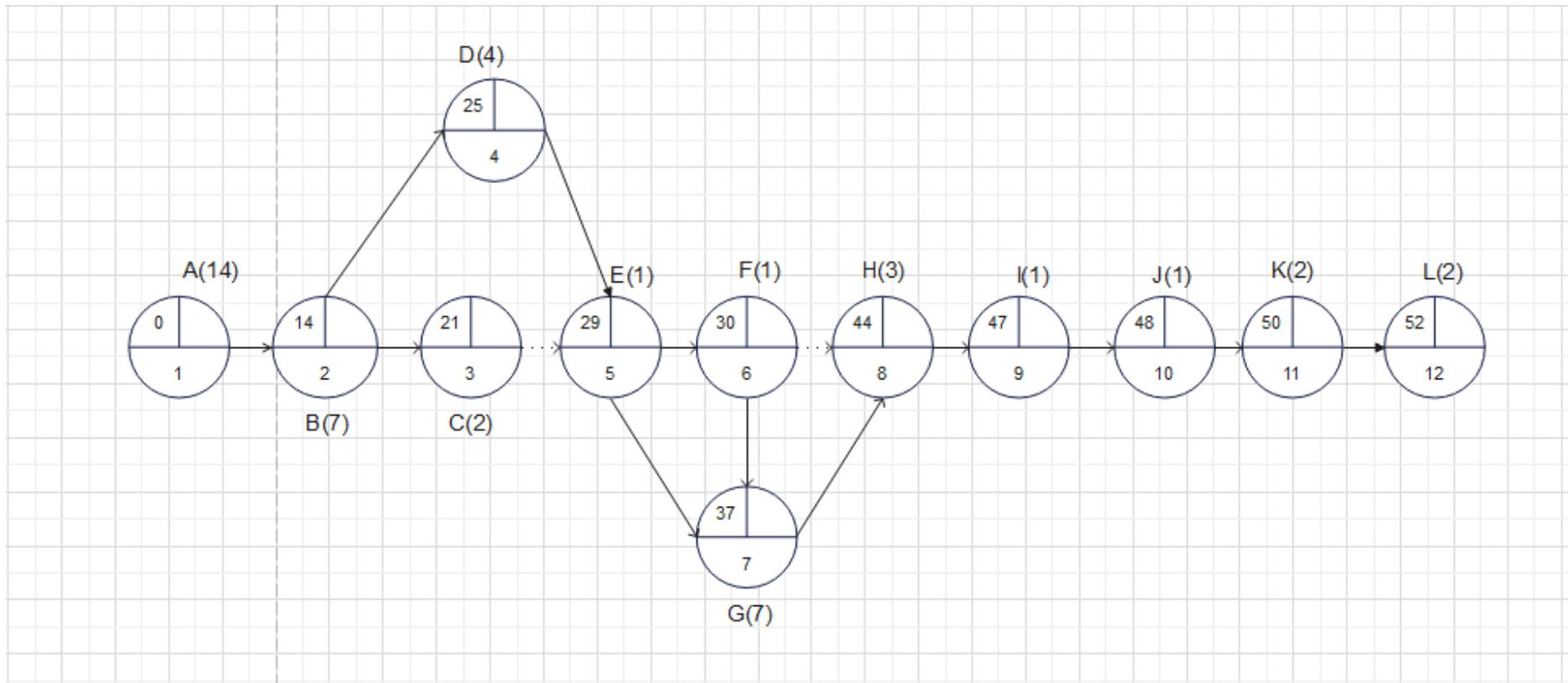


Figure 22: Date au plus tôt d'un réseau PERT

### b) La date au plus tard d'un réseau PERT

La date au plus tard d'un réseau PERT, Correspond à la date à laquelle une étape doit être atteinte au plus tard pour que la durée globale du projet reste minimum. Elle s'obtient en retirant de la date au plus tard de l'étape qui lui succède la durée de la tâche qui les relie :

***Date au plus tard « étape i » = Date plus tard « étape j » - Durée tâche « ij ».***

- Tâche L :  $LST = 52$  (fin du projet)
- Tâche K :  $LST = LST(L) - \text{Durée}(K) = 52 - 2 = 50$  jours
- Tâche J :  $LST = LST(K) - \text{Durée}(J) = 50 - 2 = 48$  jours
- Tâche I :  $LST = LST(J) - \text{Durée}(I) = 48 - 1 = 47$  jours
- Tâche H :  $LST = LST(I) - \text{Durée}(H) = 47 - 1 = 46$  jours
- Tâche G :  $LST = LST(H) - \text{Durée}(G) = 46 - 3 = 43$  jours
- Tâche F :  $LST = LST(G) - \text{Durée}(F) = 43 - 7 = 36$  jours
- Tâche E :  $LST = LST(F) - \text{Durée}(E) = 36 - 1 = 35$  jours
- Tâche D :  $LST = \min(LST(E), LST(B), LST(C)) - \text{Durée}(D) = \min(35, 21, 14) - 4 = 10$  jours
- Tâche C :  $LST = LST(D) - \text{Durée}(C) = 10 - 2 = 8$  jours
- Tâche B :  $LST = LST(D) - \text{Durée}(B) = 10 - 7 = 3$  jours
- Tâche A :  $LST = LST(B) - \text{Durée}(A) = 3 - 14 = -11$  jours (La tâche A n'a pas de marge de temps car elle est en début de projet)

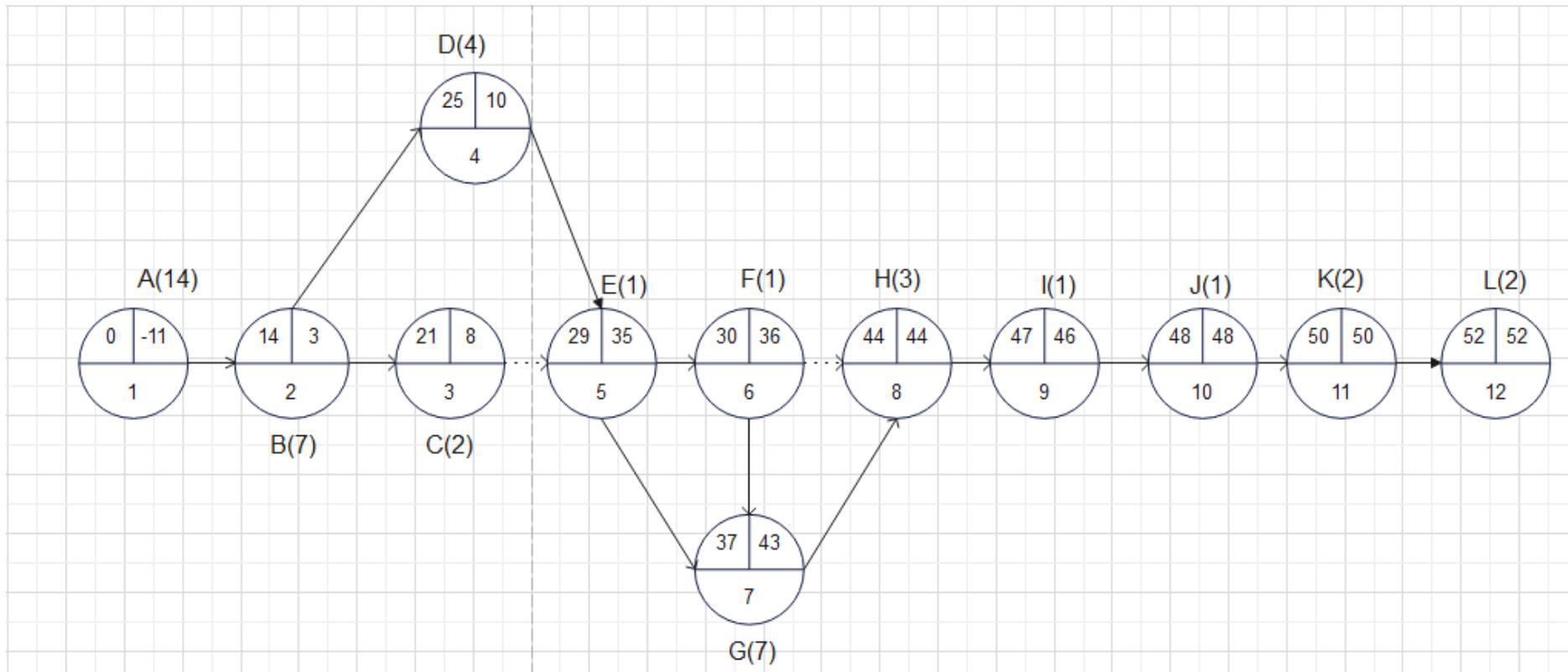


Figure 23 : La date au plus tard d'un réseau PERT

### c) Détermination de la marge libre

La marge libre, également connue sous le nom de "float" ou "slack" en anglais, représente la quantité de temps supplémentaire qu'une tâche peut subir sans affecter la date de fin du projet. En d'autres termes, c'est la marge de temps disponible avant que le retard dans une tâche n'impacte le calendrier global du projet.

Pour calculer la marge libre d'une tâche, vous devez connaître les dates au plus tôt (EST) et au plus tard (LST) pour cette tâche. La formule pour calculer la marge libre est la suivante :

$$\text{Marge libre} = \text{LST} - \text{EST}$$

Si la marge libre est positive, cela signifie que la tâche peut prendre ce temps supplémentaire sans affecter la date de fin du projet. Si la marge libre est nulle, cela signifie que la tâche fait partie du chemin critique. Si la marge libre est négative, cela signifie que la tâche est en retard et que cela impacte la date de fin du projet.

- Marge libre pour la tâche A :  $\text{Marge libre} = \text{LST}(A) - \text{EST}(A) = 3 - 0 = 3$  jours
- Marge libre pour la tâche B :  $\text{Marge libre} = \text{LST}(B) - \text{EST}(B) = 3 - 14 = -11$  jours
- Marge libre pour la tâche C :  $\text{Marge libre} = \text{LST}(C) - \text{EST}(C) = 6 - 21 = -15$  jours
- Marge libre pour la tâche D :  $\text{Marge libre} = \text{LST}(D) - \text{EST}(D) = 10 - 25 = -15$  jours
- Marge libre pour la tâche E :  $\text{Marge libre} = \text{LST}(E) - \text{EST}(E) = 35 - 29 = 6$  jours
- Marge libre pour la tâche F :  $\text{Marge libre} = \text{LST}(F) - \text{EST}(F) = 36 - 30 = 6$  jours
- Marge libre pour la tâche G :  $\text{Marge libre} = \text{LST}(G) - \text{EST}(G) = 43 - 37 = 6$  jours
- Marge libre pour la tâche H :  $\text{Marge libre} = \text{LST}(H) - \text{EST}(H) = 46 - 44 = 2$  jours
- Marge libre pour la tâche I :  $\text{Marge libre} = \text{LST}(I) - \text{EST}(I) = 47 - 47 = 0$  jours (fait partie du chemin critique)
- Marge libre pour la tâche J :  $\text{Marge libre} = \text{LST}(J) - \text{EST}(J) = 48 - 48 = 0$  jours (fait partie du chemin critique)
- Marge libre pour la tâche K :  $\text{Marge libre} = \text{LST}(K) - \text{EST}(K) = 50 - 50 = 0$  jours (fait partie du chemin critique)
- Marge libre pour la tâche L :  $\text{Marge libre} = \text{LST}(L) - \text{EST}(K) = 52 - 52 = 0$  jours (fait partie du chemin critique)

Les tâches A, E, F, G, et H ont une marge libre positive, ce qui signifie qu'elles peuvent subir un retard de quelques jours sans affecter la date de fin du projet. Les tâches B, C, D, I, J, K et

L ont une marge libre nulle, ce qui indique qu'elles font partie du chemin critique et tout retard les retardera le projet.

**d) Détermination de la marge totale**

Le chemin critique est le suivant :

A, B, D, E, G, H, I, J, K, L

La durée totale du projet en suivant le chemin critique est de 52 jours. Toutes les tâches de ce chemin sont considérées comme critiques, ce qui signifie que si l'une de ces tâches prend plus de temps que prévu, le projet dans son ensemble sera retardé. Il est essentiel de gérer attentivement les tâches critiques pour garantir que le projet soit achevé dans les délais prévus. Si des retards surviennent dans des tâches critiques, il peut être nécessaire de prendre des mesures correctives pour éviter un retard global dans le projet.

### 3.2.5. DETERMINATION DU CHEMIN CRITIQUE

Le chemin critique est la séquence de tâches qui détermine la durée totale du projet et qui n'a aucune marge libre, c'est-à-dire que chaque tâche dans le chemin critique est critique et doit être terminée exactement dans les délais prévus pour que le projet soit achevé à temps. Toute variation dans la durée de l'une de ces tâches entraînera automatiquement un retard dans la date de fin du projet.

En examinant le diagramme PERT que nous avons créé, le chemin critique est le suivant :

A -> B -> D -> E -> G -> H -> I -> J -> K->L

Les tâches incluses dans ce chemin sont les tâches critiques :

- Tâche A : Elle est la tâche de début du projet, et le retard dans cette tâche retarderait tout le projet.
- Tâche B : Elle suit directement la tâche A et précède les autres tâches critiques. Tout retard dans cette tâche affecterait directement la date de fin du projet.
- Tâche D : Elle suit la tâche B et précède la tâche E, qui est également critique. Tout retard dans cette tâche se propage à la tâche E et au-delà.
- Tâche E : Elle suit la tâche D et précède la tâche G, qui est également critique. Tout retard dans cette tâche se propage à la tâche G et au-delà.
- Tâche G : Elle suit la tâche E et précède la tâche H, qui est également critique. Tout retard dans cette tâche se propage à la tâche H et au-delà.
- Tâche H : Elle suit la tâche G et précède la tâche I, qui est également critique. Tout retard dans cette tâche se propage à la tâche I et au-delà.
- Tâche I : Elle suit la tâche H et précède la tâche J, qui est également critique. Tout retard dans cette tâche se propage à la tâche J et au-delà.
- Tâche J : Elle suit la tâche I et précède la tâche K, qui est également critique. Tout retard dans cette tâche se propage à la tâche K et au-delà.
- Tâche K : Elle suit la tâche I et précède la tâche J, qui est également critique. Tout retard dans cette tâche se propage à la tâche L et au-delà.
- Tâche K : Elle est la dernière tâche du projet, et tout retard dans cette tâche retardera la fin du projet.

Le chemin critique est donc constitué de toutes ces tâches liées en série, et il détermine la durée totale du projet. Aucune de ces tâches n'a de marge libre, ce qui signifie qu'elles doivent toutes être terminées exactement dans les délais prévus pour que le projet soit achevé à temps

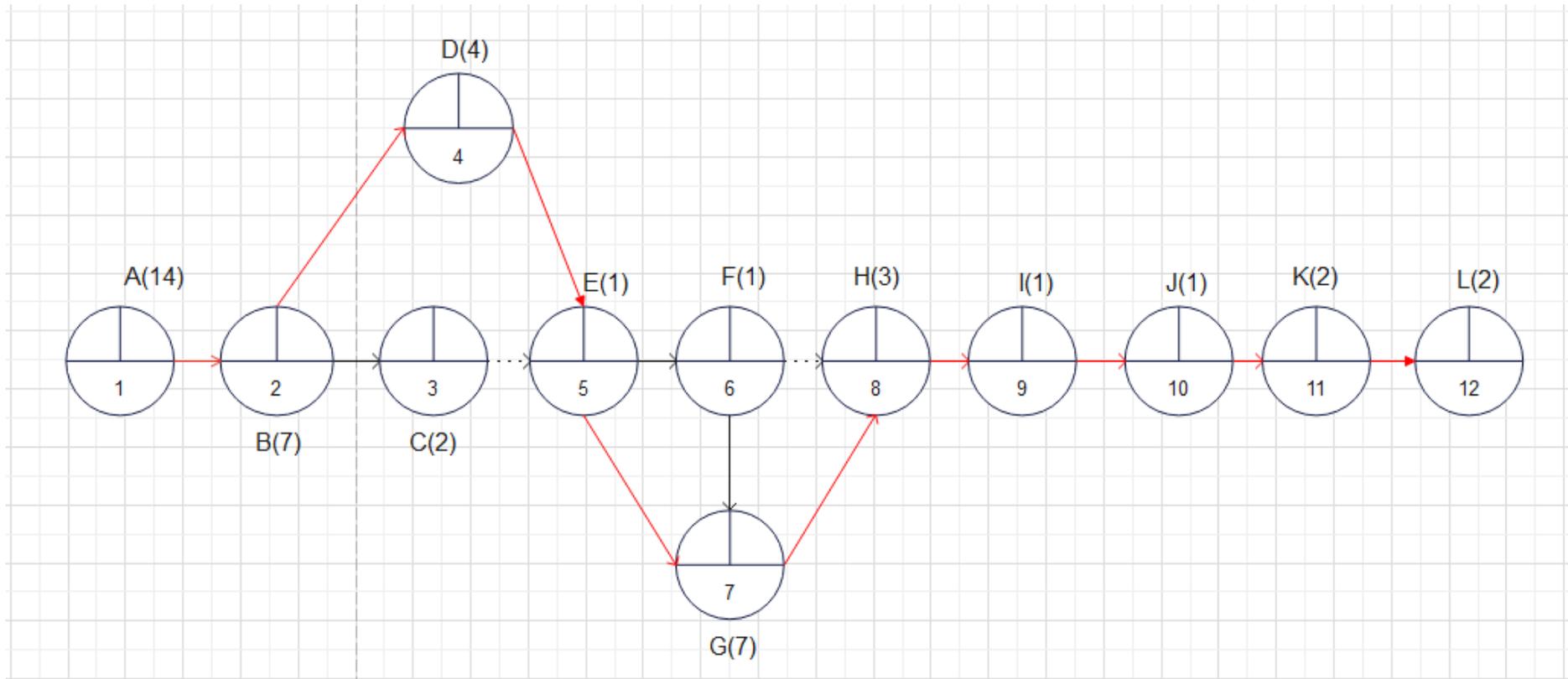


Figure 24: chemin critique

### 3.2.6. CALENDRIER DU PROJET



The screenshot shows a software interface for project management. At the top, there are two tabs: 'Diagramme de Gantt' and 'Diagramme des Ressources'. Below the tabs is a toolbar with various icons for navigation and editing. The main area features a header with the 'GANTT project' logo and a decorative hexagonal pattern. Below the header is a table with three columns: 'Nom', 'Date de début', and 'Date de fin'. The table lists various project tasks with their corresponding start and end dates. The last row, 'Formation des utilisateurs', is highlighted in a dark grey color.

Nom	Date de début	Date de fin
Commandes des matériels	10/07/2023	27/07/2023
Implémentation des matériels	28/07/2023	07/08/2023
Test du systeme embarque	08/08/2023	10/08/2023
Developpement de l'API	11/08/2023	14/08/2023
Test de l'API	15/08/2023	15/08/2023
Correction d'erreurs	16/08/2023	16/08/2023
Développement du site internet	17/08/2023	25/08/2023
Liaison du site internet avec l'API	28/08/2023	30/08/2023
Test	31/08/2023	31/08/2023
Correction d'erreurs	01/09/2023	01/09/2023
Test general	04/09/2023	05/09/2023
Formation des utilisateurs	07/09/2023	09/09/2023

Figure 25 : Calendrier du projet

### 3.2.7. DIANGRAME DE GANTT

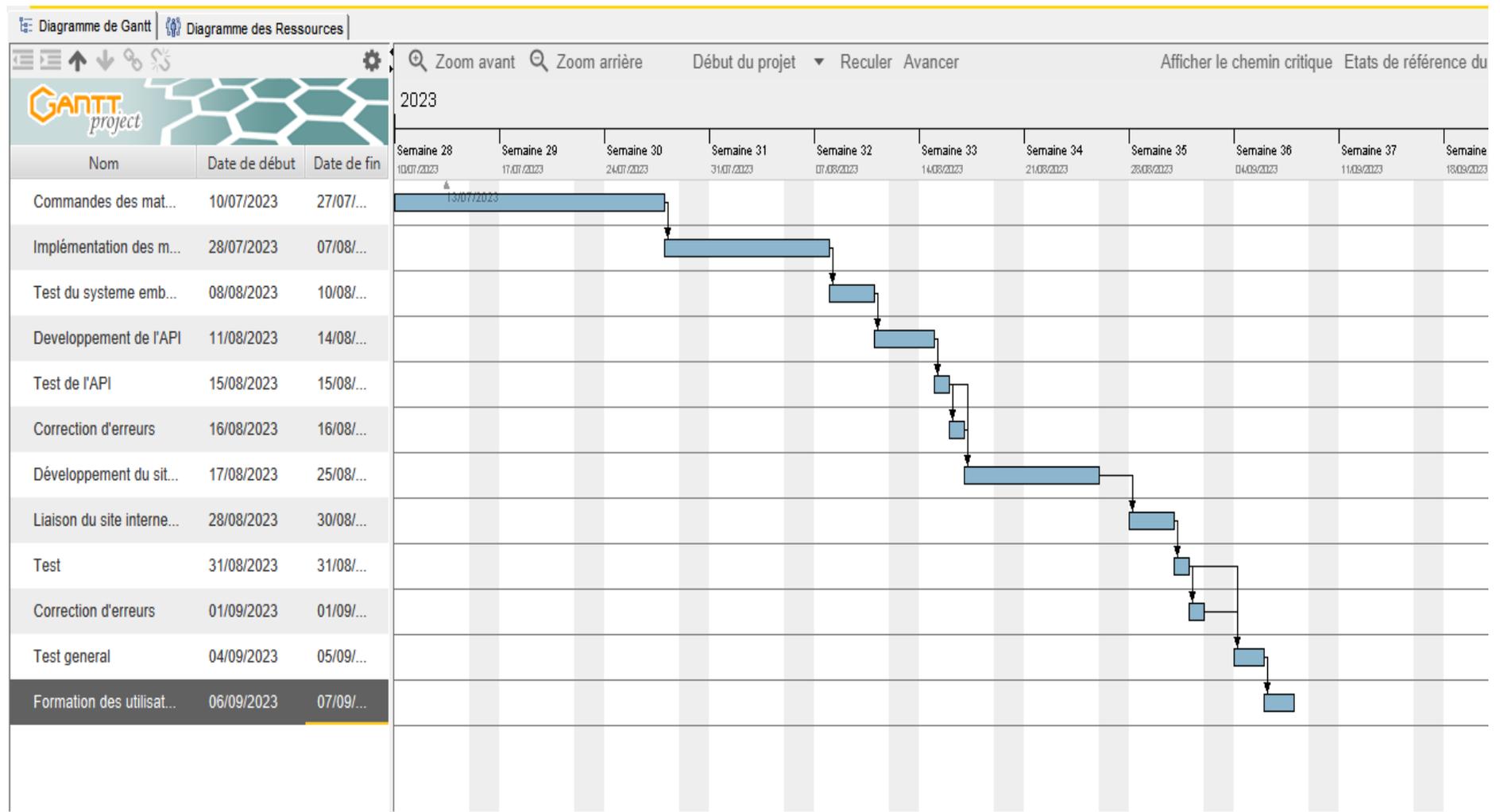


Figure 26 : Diagramme Gantt

### 3.2.8. Diagramme Pert

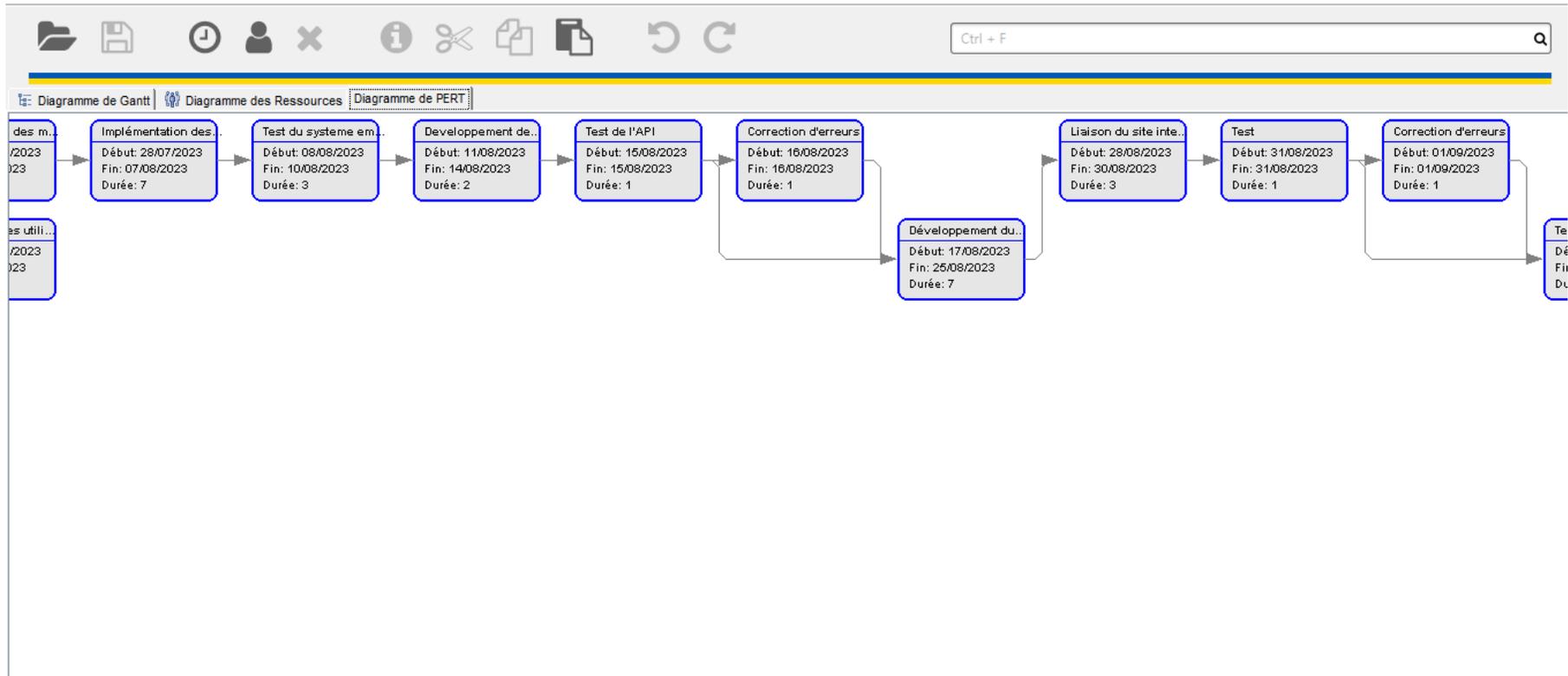


Figure 27 : Diagramme Pert

### **3.3. MODÉLISATION DU SYSTÈME D'INFORMATION UML**

#### **3.3.1. Introduction**

La conception de systèmes est le processus qui consiste à définir les interfaces, les composants, les données et les modules d'un système afin de satisfaire des exigences spécifiques. Elle vise à satisfaire les besoins et les exigences spécifiques d'une entreprise ou d'une organisation par l'ingénierie d'un système cohérent et performant.

L'objectif de la conception d'un système est de créer une solution technique qui répond aux exigences fonctionnelles du système.

### 3.3.2. Diagramme de cas d'utilisation

Le Diagramme de cas d'utilisation représente une fonctionnalité spécifique dans un système et est créé pour illustrer comment différentes fonctionnalités sont interconnectées et montrer leurs contrôleurs (ou acteurs) internes et externes.

Un cas d'utilisation est une liste d'étapes qui définissent les interactions entre un acteur 'humain qui interagit avec le système ou un système externe et le système lui-même. Les diagrammes de cas d'utilisation décrivent les spécifications d'un cas d'utilisation et modélisent les unités fonctionnelles d'un système. Ces diagrammes aident les équipes de développeurs à comprendre les besoins de leur système, notamment le rôle de l'interaction. Un diagramme d'utilisation peut illustrer tous les cas d'utilisation du système ou seulement un groupe de cas d'utilisation ayant des fonctionnalités similaires.

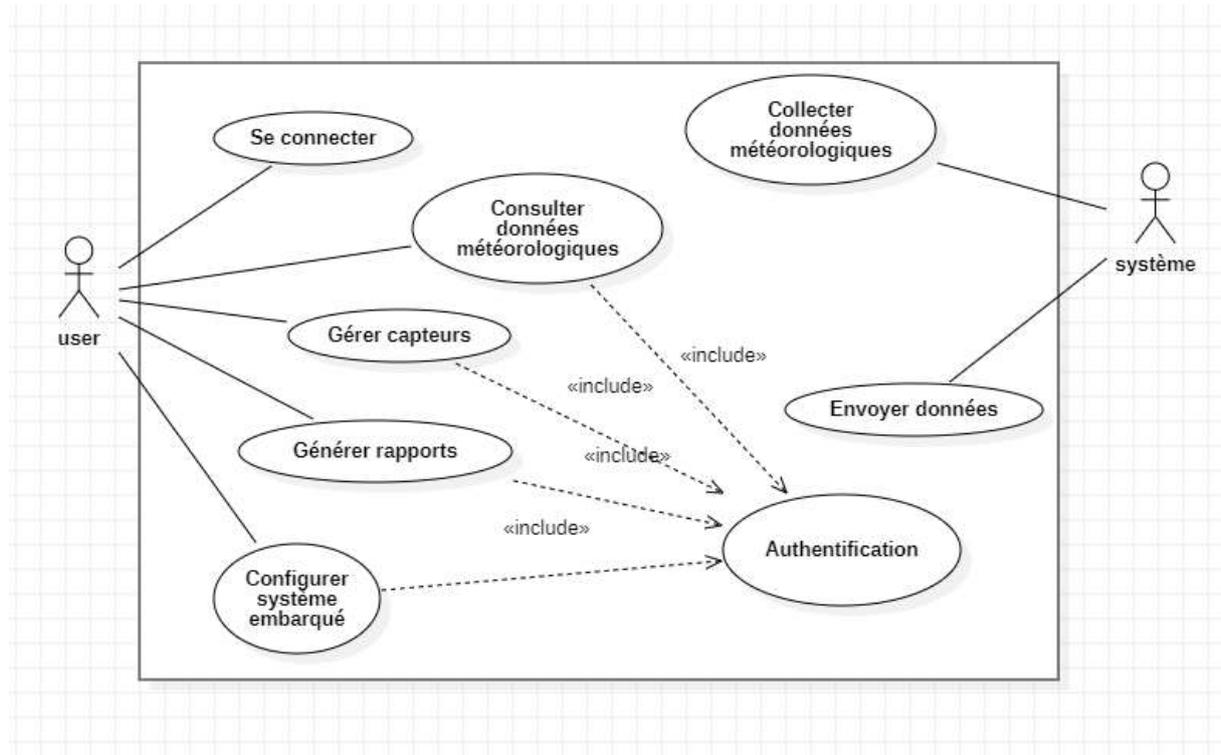


Figure 28 : Diagramme de cas d'utilisation

### 3.3.2. Diagramme d'activité

Un diagramme d'activités montre un flux séquentiel d'actions ; Le diagramme d'activité est généralement utilisé pour décrire les activités réalisées dans un flux de travail de processus général, bien qu'il puisse également être utilisé pour décrire d'autres activités flux, tels qu'un cas d'utilisation ou un flux de contrôle détaillé. Le diagramme d'activité se compose d'actions, qui sont l'unité de base du comportement qui constitue une activité. Les diagrammes d'activité ont un certain nombre de fonctionnalités pour montrer les flux de contrôle, en utilisant jetons comme moyen d'afficher le flux dans un système. Les diagrammes d'activité peuvent utiliser ces mécanismes de contrôle de flux pour montrer la réponse aux déclencheurs de événements externes ou à partir d'un jalon prédéterminé, tel qu'un point dans le temps. Le diagramme peut spécifier les messages et les objets envoyés ou reçus dans le cadre des actions réalisées. Décisions et conditions, ainsi que l'exécution parallèle d'actions, peut également être montré dans le diagramme (Eriksson, Penker, Lyons, & Fado).

Les diagrammes d'activités montrent la structure procédurale de contrôle entre les objets de classe, ainsi que les processus organisationnels tels que les flux de travail. Ces diagrammes sont composés de formes spécialisées, qui sont ensuite reliées par des flèches. La notation définie pour les diagrammes d'activités est similaire à celle des diagrammes états-transitions. Les éléments de base du diagramme d'activité sont les suivants :

- Des actions,
- Des flots de contrôle entre actions,
- Des décisions (aussi appelés branchements conditionnels),
- Un début et une ou plusieurs terminaison(s) possible(s).

## Représentation des activités

Un diagramme d'activité représente l'état de l'exécution d'un mécanisme, sous la forme d'un déroulement d'étapes regroupées séquentiellement dans des branches parallèles de flots de contrôles.

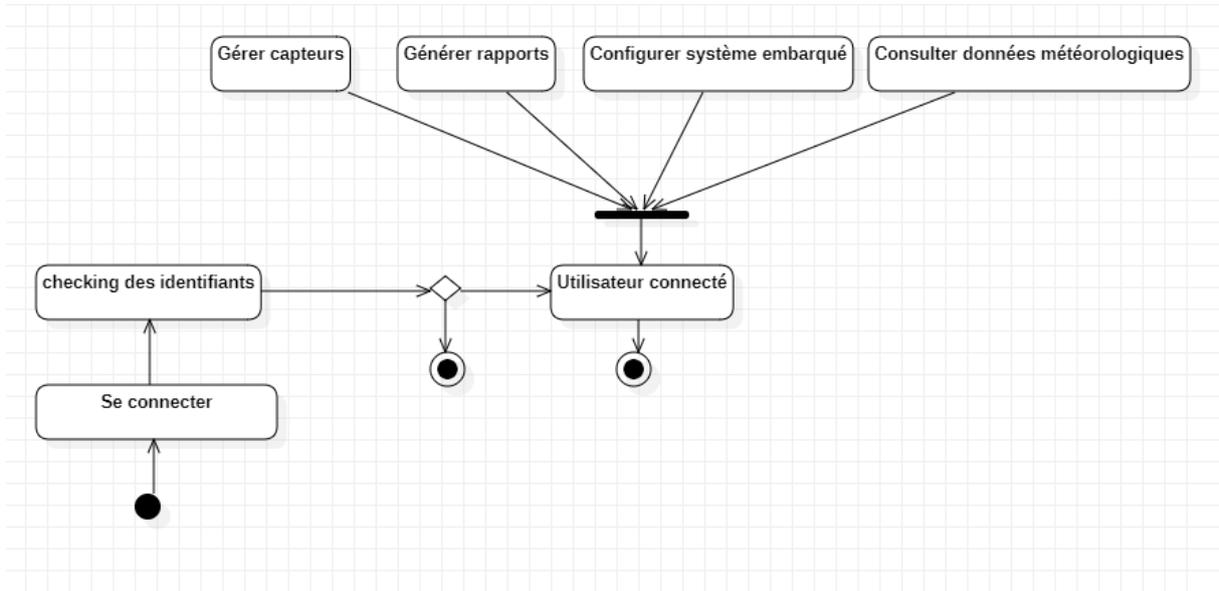


Figure 29 : Représentation des activités

### 3.3.3. Diagramme de séquence

Le diagramme de séquence est généralement utilisé pour représenter la séquence temporelle des interactions entre différents acteurs (dans ce cas, le Propriétaire du Champ et le Système Embarqué) et les objets impliqués dans un scénario spécifique. Voici un diagramme de séquence qui représente le scénario de base des interactions entre le Propriétaire du Champ et le Système Embarqué ;

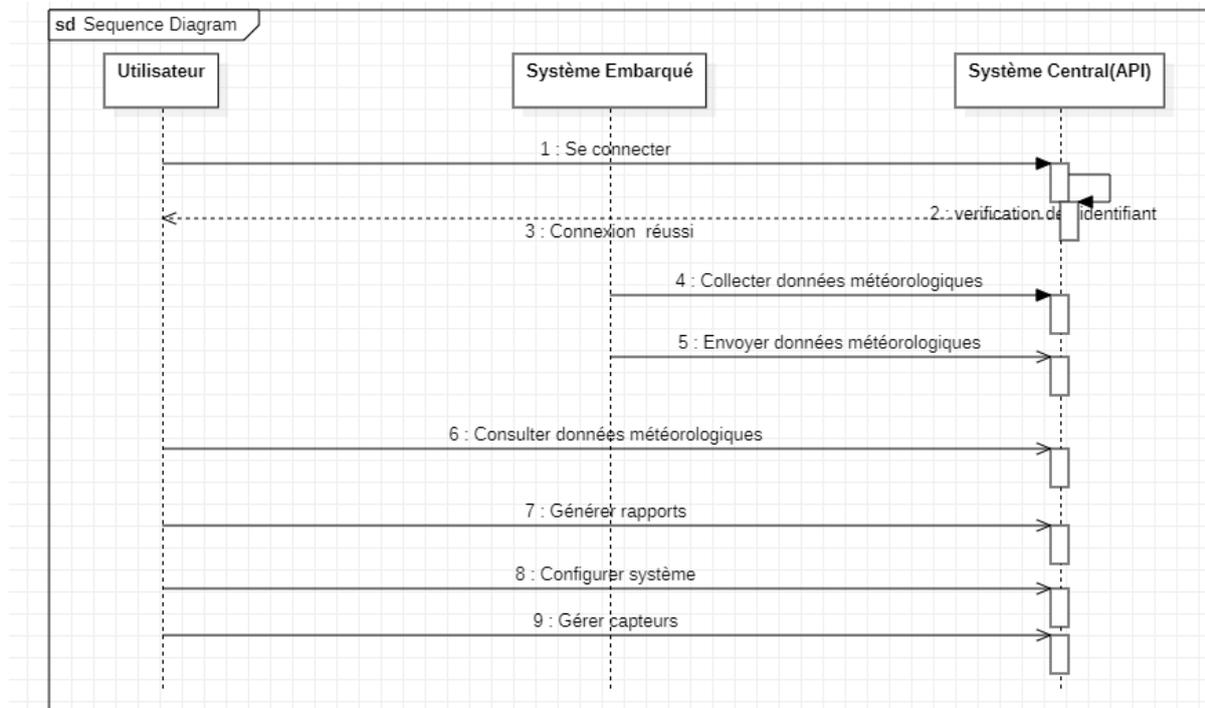


Figure 30 : Diagramme de séquence

Ce diagramme de séquence détaille l'ordre chronologique des interactions entre le Propriétaire du Champ et le Système Embarqué. Il commence par la connexion du propriétaire au système, suivi de la consultation des données météorologiques, de la gestion des capteurs, de la génération de rapports et de la configuration du système embarqué. Les flèches indiquent la direction des messages entre les acteurs et les objets, et les notes fournissent des informations supplémentaires sur chaque étape de l'interaction.

### 3.3.5. Diagramme de classe

Le diagramme de classe que j'ai fourni est une représentation visuelle de la structure statique d'un système logiciel ou d'un système en général. Voici une définition plus détaillée de ses composants :

1. **Classe** : Une classe est une entité de base dans un diagramme de classe. Elle représente un modèle pour la création d'objets, définissant les attributs (variables) et les opérations (méthodes) que les objets de cette classe auront.
2. **Attributs** : Les attributs sont des variables membres d'une classe qui décrivent les caractéristiques ou les propriétés des objets de cette classe. Dans le diagramme, les attributs sont indiqués dans la partie supérieure de chaque classe avec leur nom et leur type de données.
3. **Opérations (Méthodes)** : Les opérations représentent les actions ou les comportements que les objets de la classe peuvent effectuer. Les opérations sont définies dans la partie inférieure de chaque classe avec leur nom et leurs paramètres.
4. **Relations entre les classes**: Les relations entre les classes sont généralement représentées par des lignes et des symboles entre les classes. Dans le diagramme de classe que j'ai fourni, il n'y a pas de relations spécifiques entre les classes, mais dans un système plus complexe, vous pourriez avoir des relations telles que l'héritage (classe fille / classe mère) ou l'association (relation entre deux classes).

Ensemble, ces éléments constituent un modèle de la structure statique du système que vous souhaitez représenter. Cela permet de comprendre comment les différentes parties du système interagissent et sont organisées, ce qui est essentiel pour concevoir et développer des logiciels ou des systèmes informatiques.

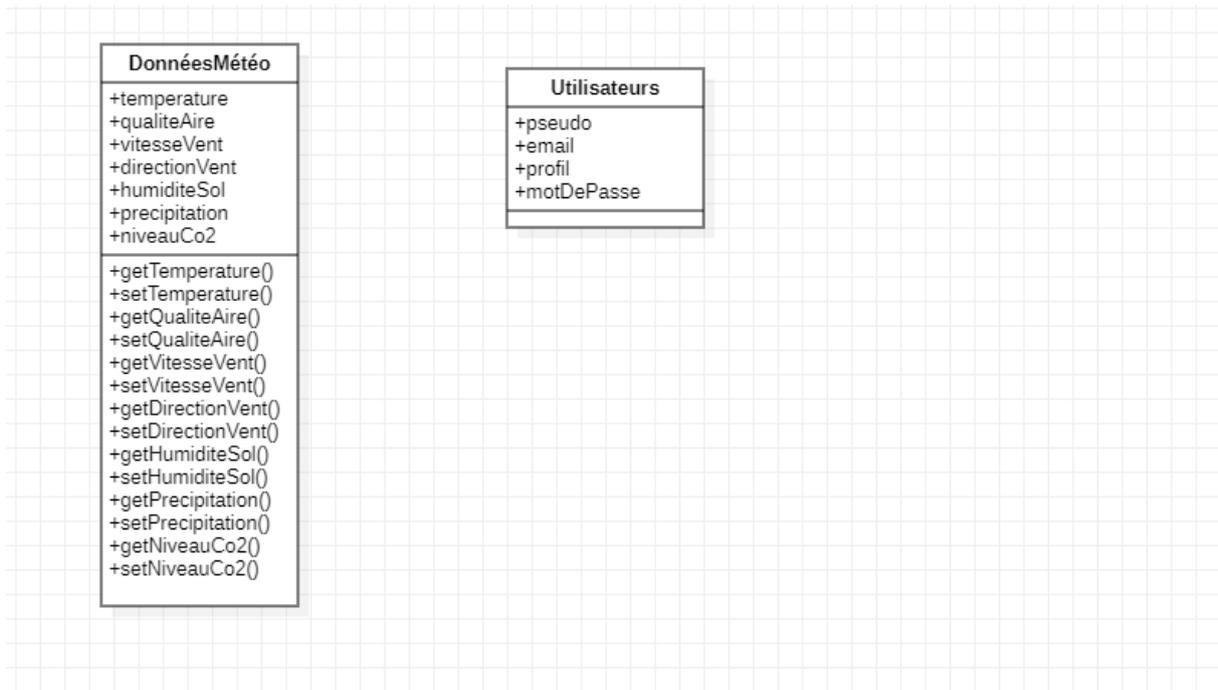


Figure 31 : Diagramme de classe

### 3.3.6. Le diagramme d'objet

Le diagramme d'objet est l'un des types de diagrammes UML (Unified Modeling Language) qui est utilisé pour représenter un instantané statique d'un système à un moment donné. Contrairement au diagramme de classe qui se concentre sur la structure des classes et de leurs relations, le diagramme d'objet se concentre sur les instances spécifiques (objets) de ces classes à un moment précis dans le temps.

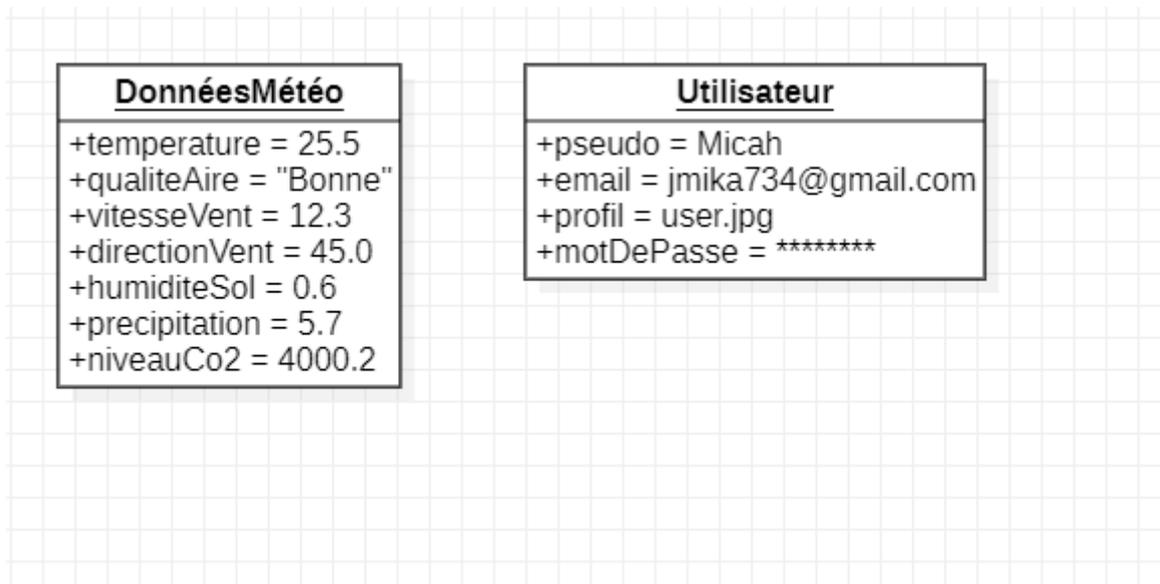


Figure 32 : Le diagramme d'objet

### 3.3.5. Diagramme des composants

Le diagramme de composants Représente la relation structurelle entre les éléments d'un système logiciel, le plus souvent utilisé avec des systèmes complexes disposant de multiples composants.

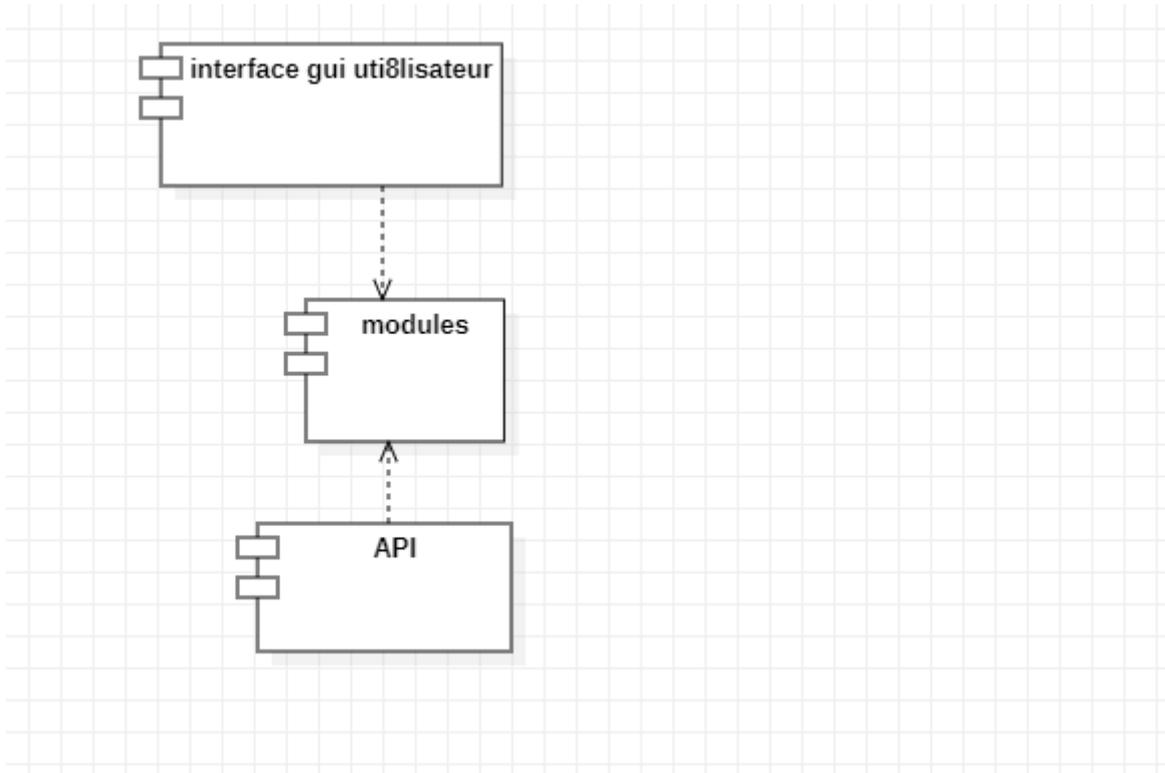


Figure 33 : Diagramme des composants

### 3.3.6. Diagramme de déploiement

Le diagramme de déploiement est l'un des types de diagrammes de structure utilisés dans le langage de modélisation unifié (UML) pour représenter la disposition physique des composants matériels et logiciels dans un système informatique ou une architecture logicielle. Ce type de diagramme est particulièrement utile pour visualiser comment les différents éléments d'un système sont déployés sur du matériel physique, des serveurs, des nœuds réseau, etc.

Le diagramme de déploiement est principalement utilisé pour modéliser l'architecture physique d'un système logiciel, montrant comment les composants sont répartis sur le matériel physique. Il est particulièrement utile lors de la planification de l'infrastructure matérielle d'une application ou d'un système distribué, permettant ainsi de prendre des décisions importantes en matière de mise en réseau, de scalabilité et de performances.

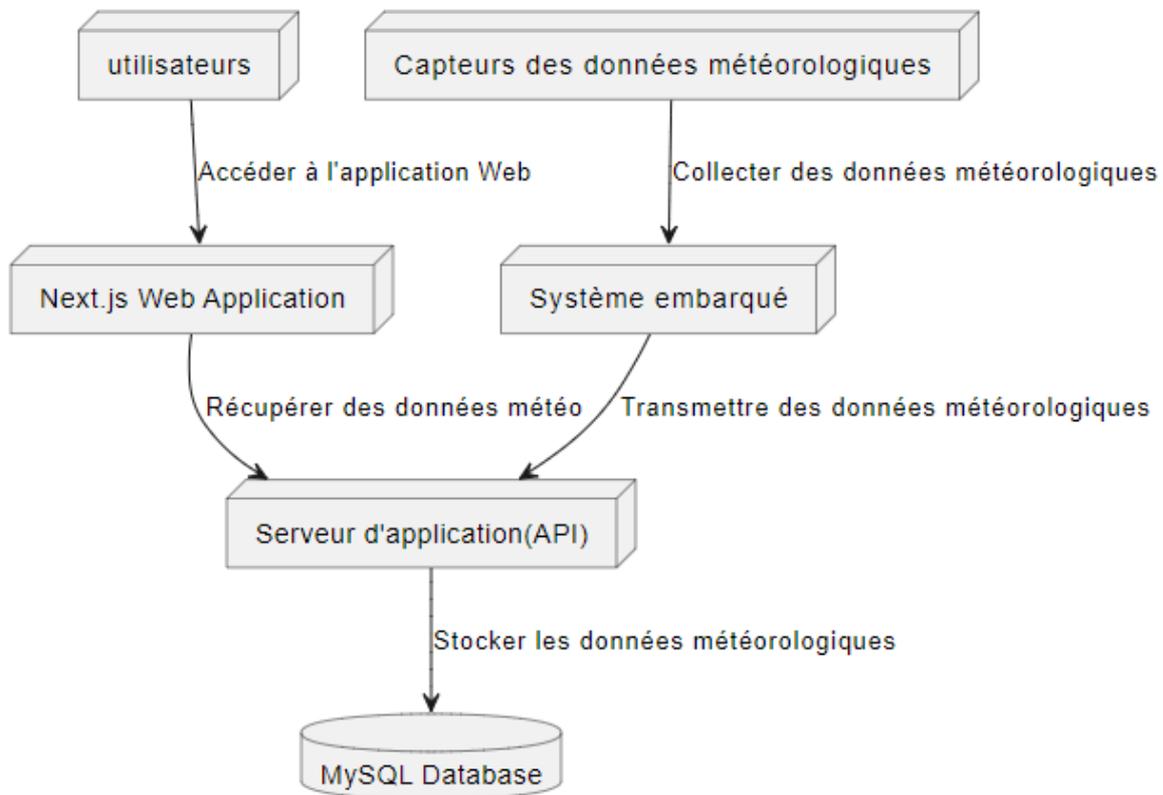


Figure 34 : Diagramme de déploiement

## CHAPITRE QUATRIEME : PRESENTATION DES RESULTATS

### 4.1. Environnement Matériel et logiciel

#### Environnement Matériel

Au cours de la réalisation de notre projet nous avons utilisé les matériels suivants:

Appareil	Unité	Caractéristiques
Ordinateur Portable HP Lenovo X250	Processeur	Intel(R) Core(TM) i5-5300U CPU @ 2.30GHz 2.29 GHz
	RAM	8,00 Go
	Disque dur	640 Go
	Ecran	1366 x768

Tableau 5. Environnement Matériel

### 4.2. Présentation des IHM(Captures)

#### 4.2.1. Application

##### Structure du projet

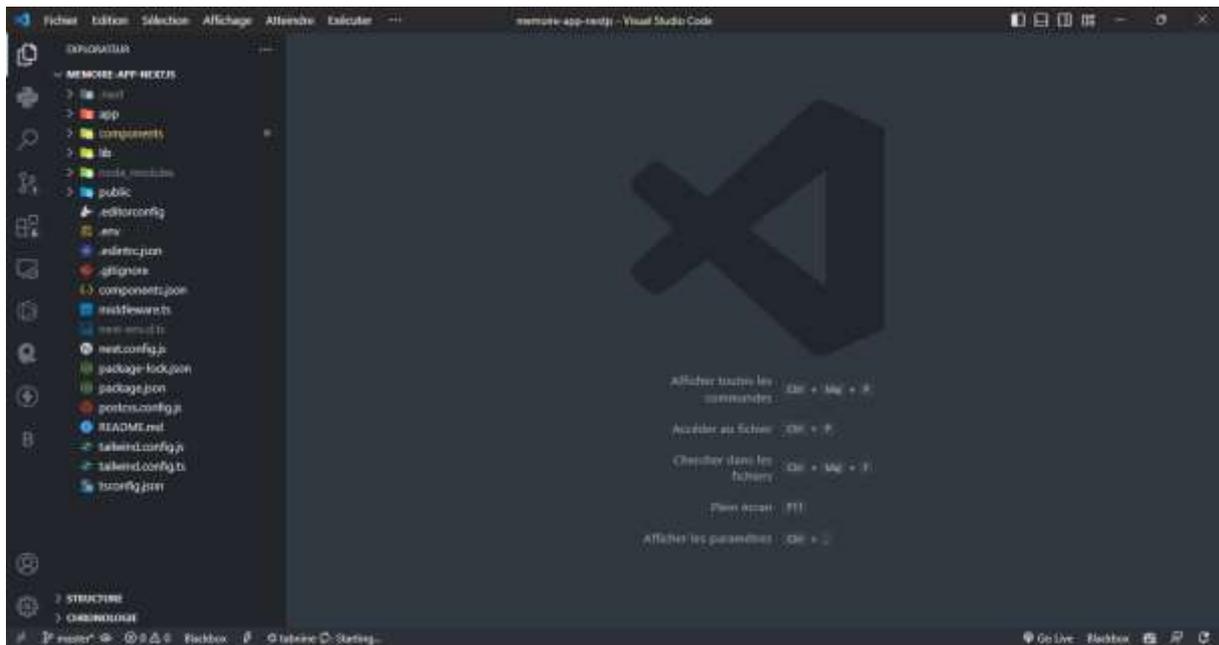


Figure 35 : Structure du projet

## La page de connexion pour les utilisateurs

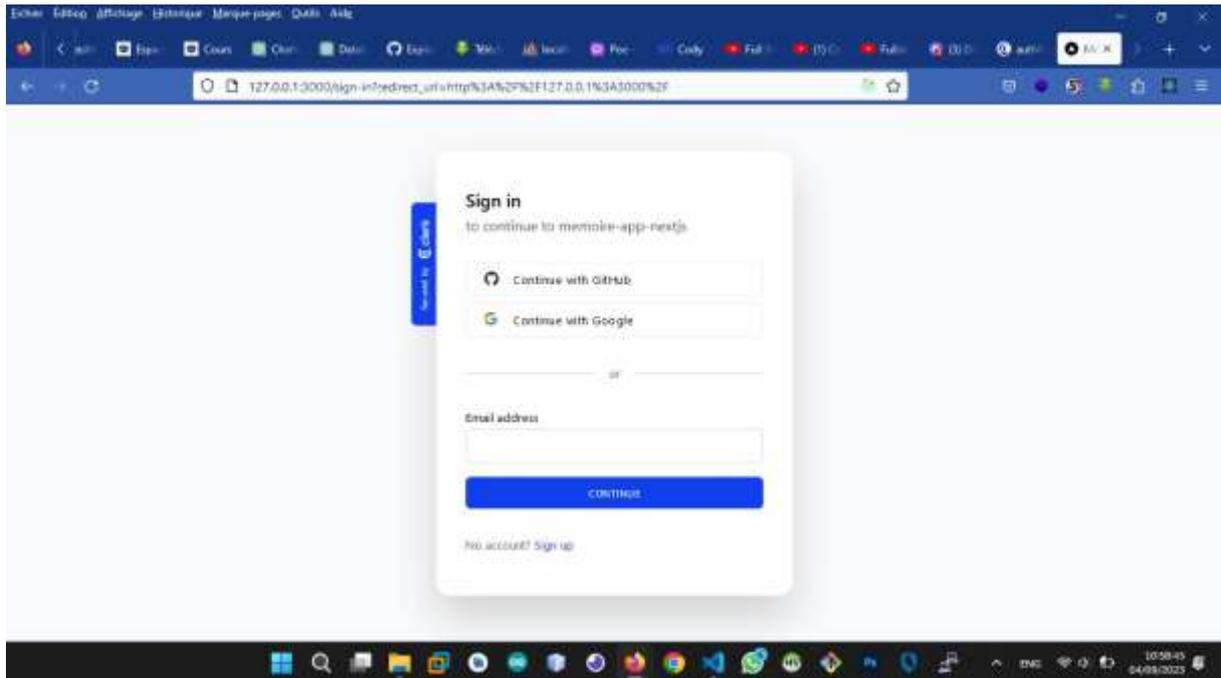


Figure 36 : La page de connexion pour les utilisateurs

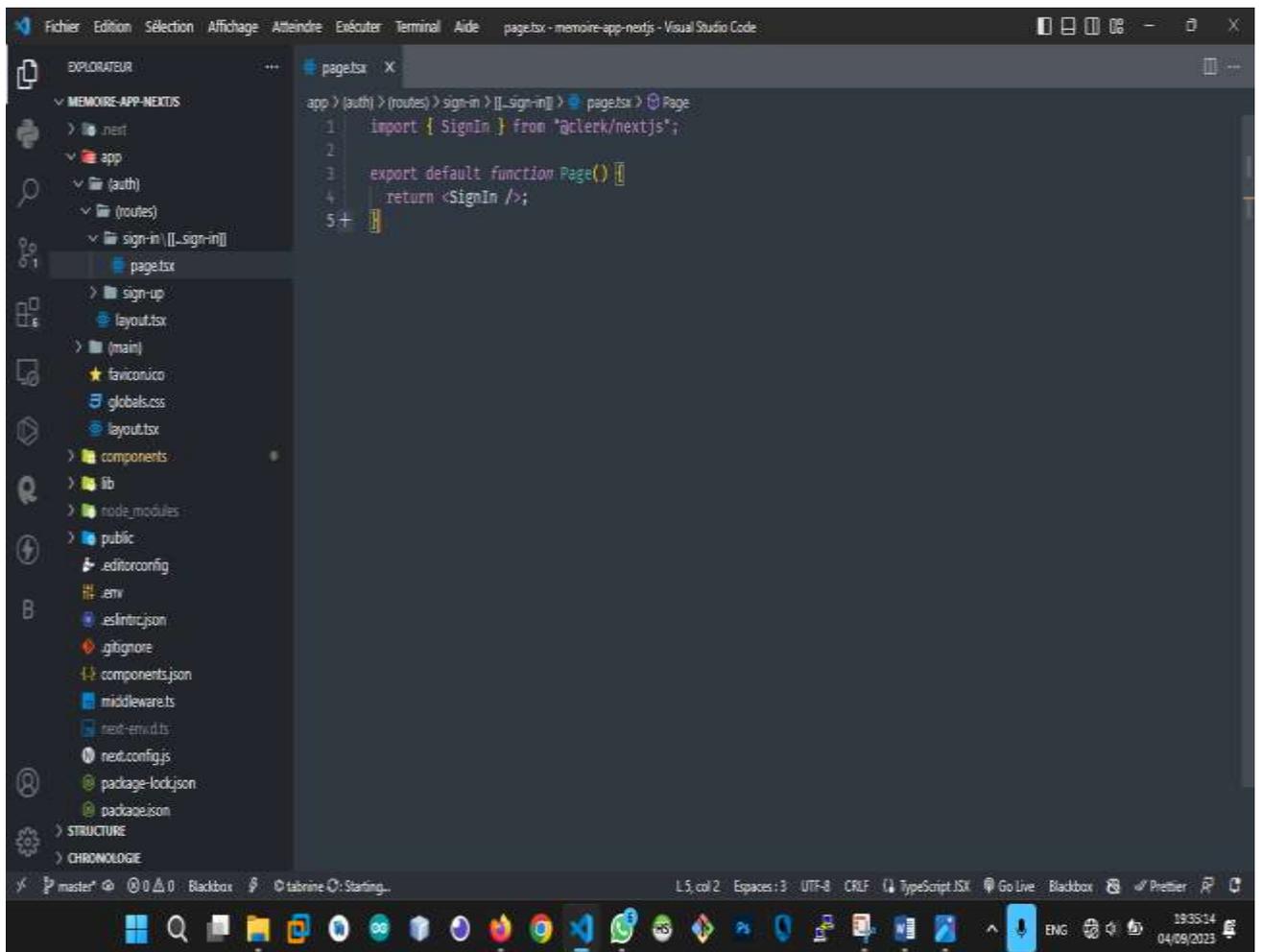


Figure 37 : Création d'un compte

## Création d'un compte

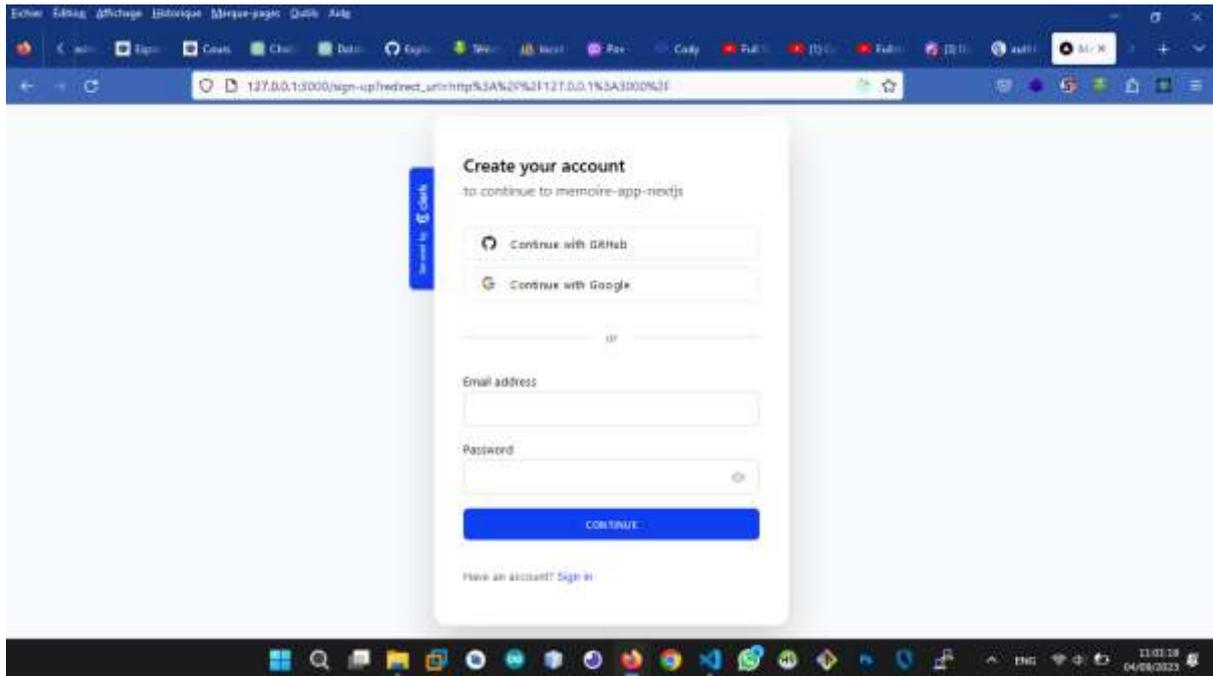


Figure 38 : Création d'un compte

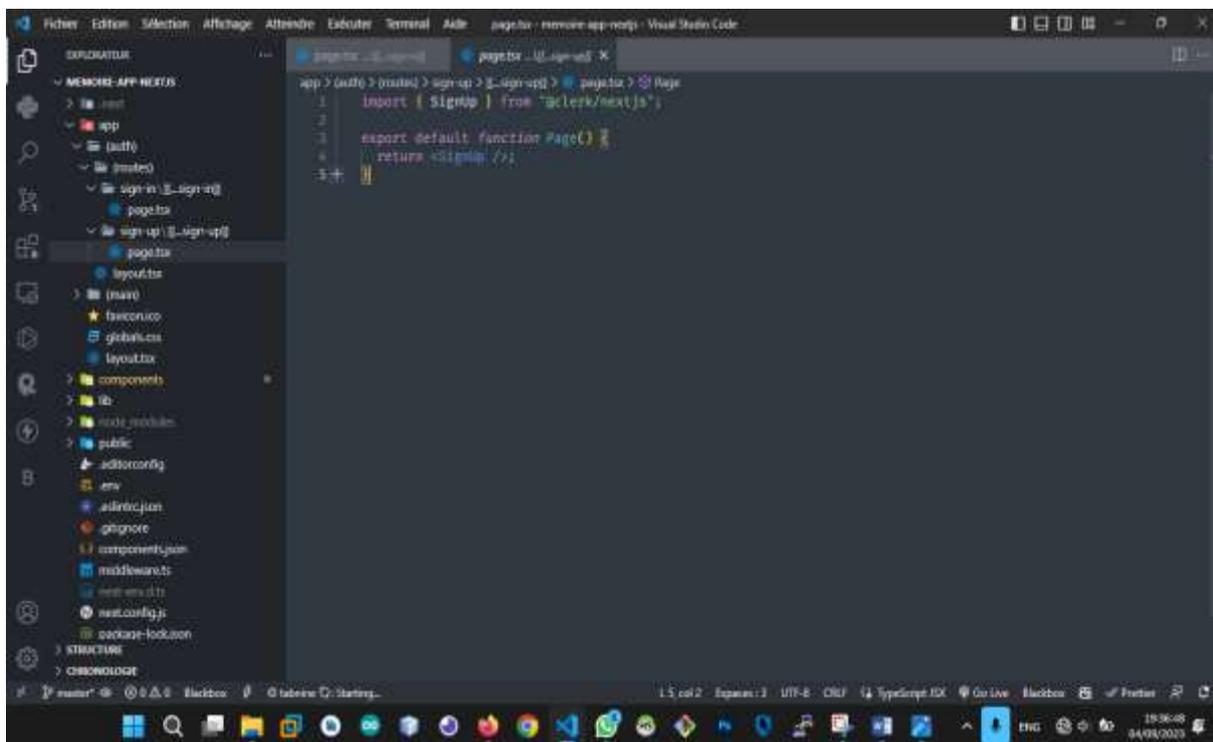


Figure 39 : Création d'un compte

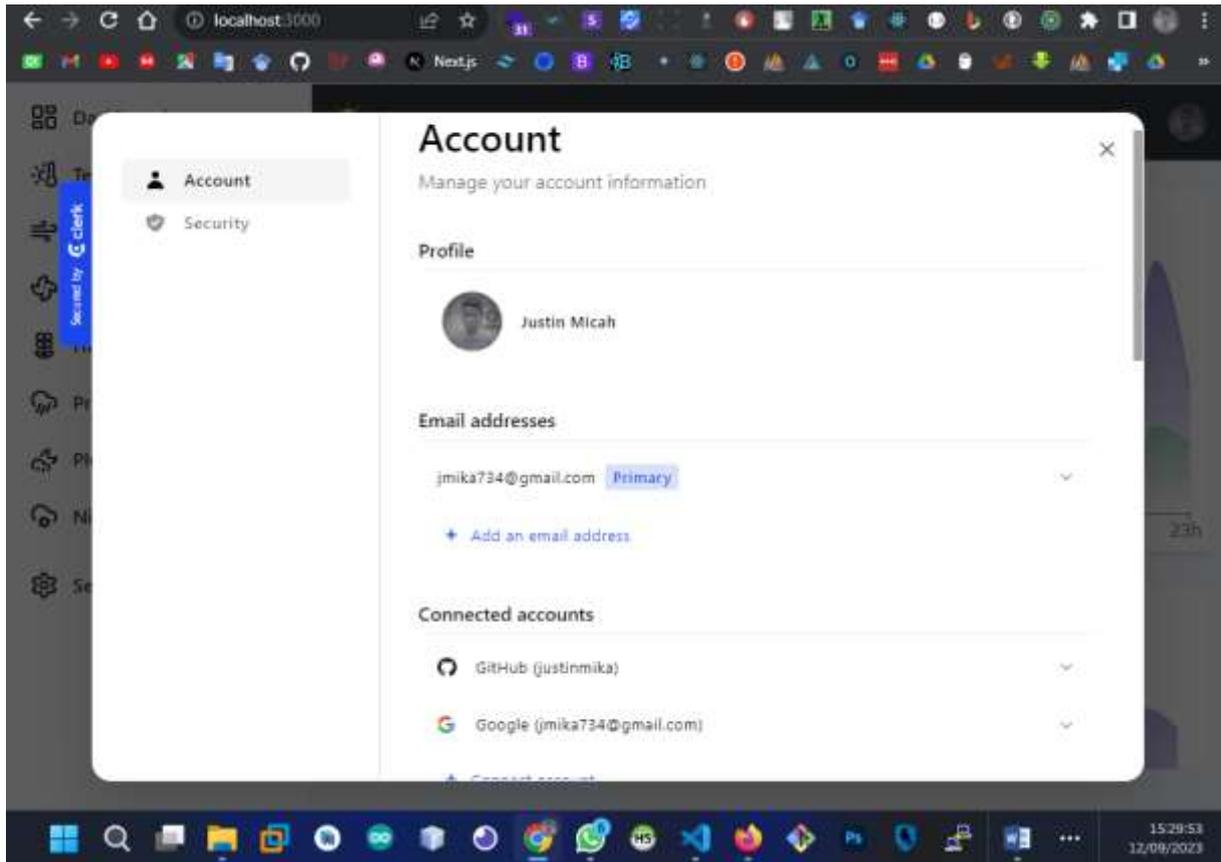


Figure 40 : Management de compte utilisateur

## Home page

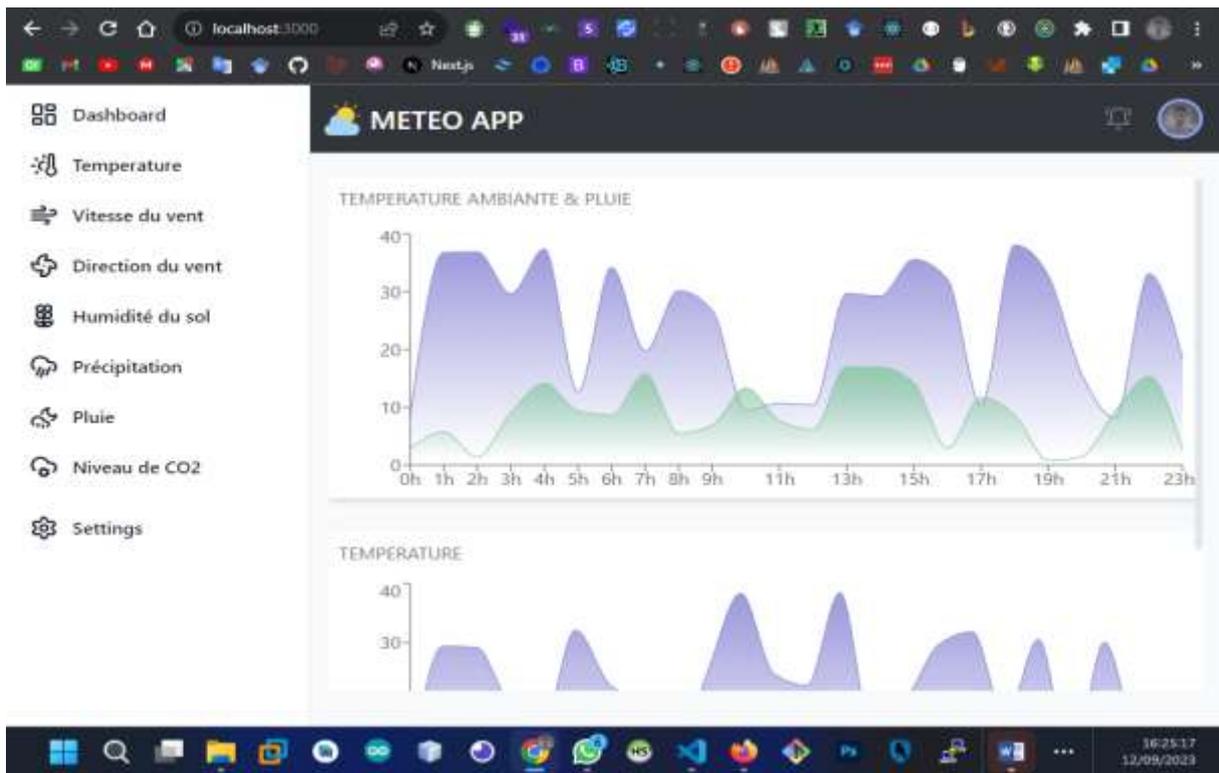


Figure 41 : Home page

The screenshot shows a code editor with the file 'page.tsx' open. The code defines a 'Home()' function that returns a JSX element. The JSX structure includes a 'NextProgress' component, a 'Navbar' component, a 'ScrollArea' component containing 'TemperaturePluie' and 'Temperature' components, and an 'MIV' component. The code is as follows:

```

1  "use client";
2  import NavBar from "@components/NavBar";
3  import Temperature from "@components/statistic/Temperature";
4  import TemperaturePluie from "@components/statistic/TemperaturePluie";
5  import { ScrollArea } from "@components/ui/scroll-area";
6  import NextProgress from "nextjs-progressbar";
7
8  export default function Home() {
9    return (
10     <main className="p-8 h-[880px] overflow-y-hidden">
11       <NextProgress
12         color="28a745"
13         startPosition={0.3}
14         stopDelayMs={200}
15         height={3}
16         showOnShallow={true}
17       />
18       <Navbar />
19       <ScrollArea className="h-[100%]">
20         <div className="space-y-8">
21           <TemperaturePluie />
22           <Temperature />
23         </div>
24       </ScrollArea>
25     </main>
26   );
27 }

```

Figure 42 : Visualisation des données en temps réels

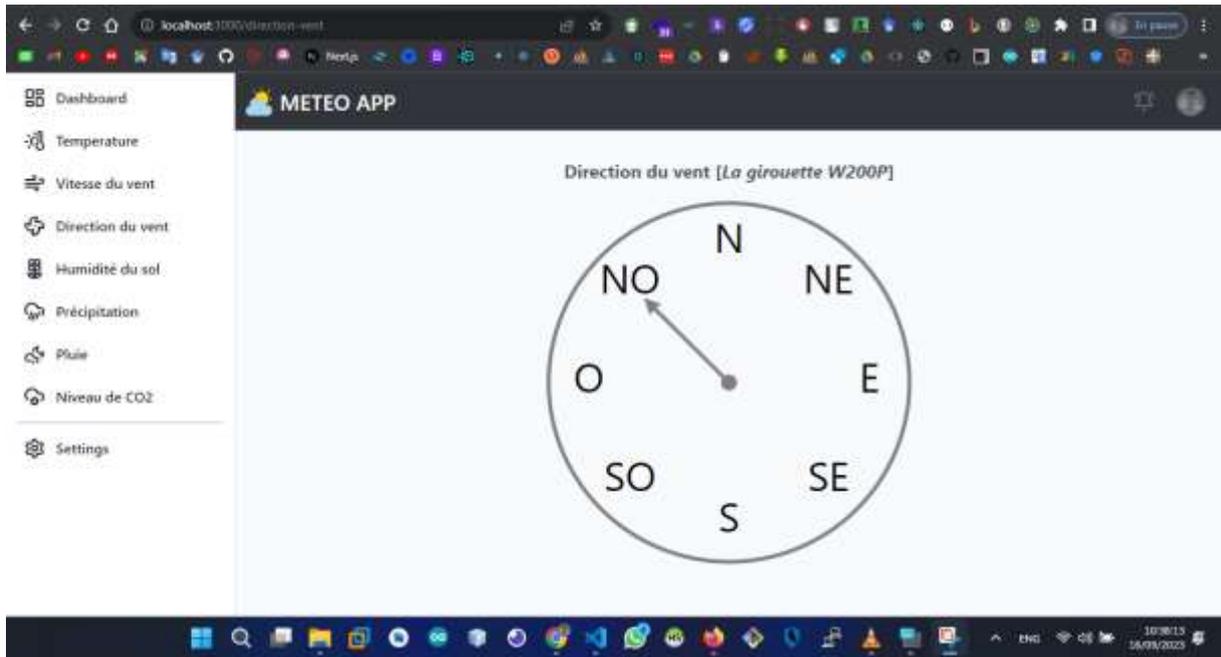


Figure 43 : Direction du vent [La girouette W200P]

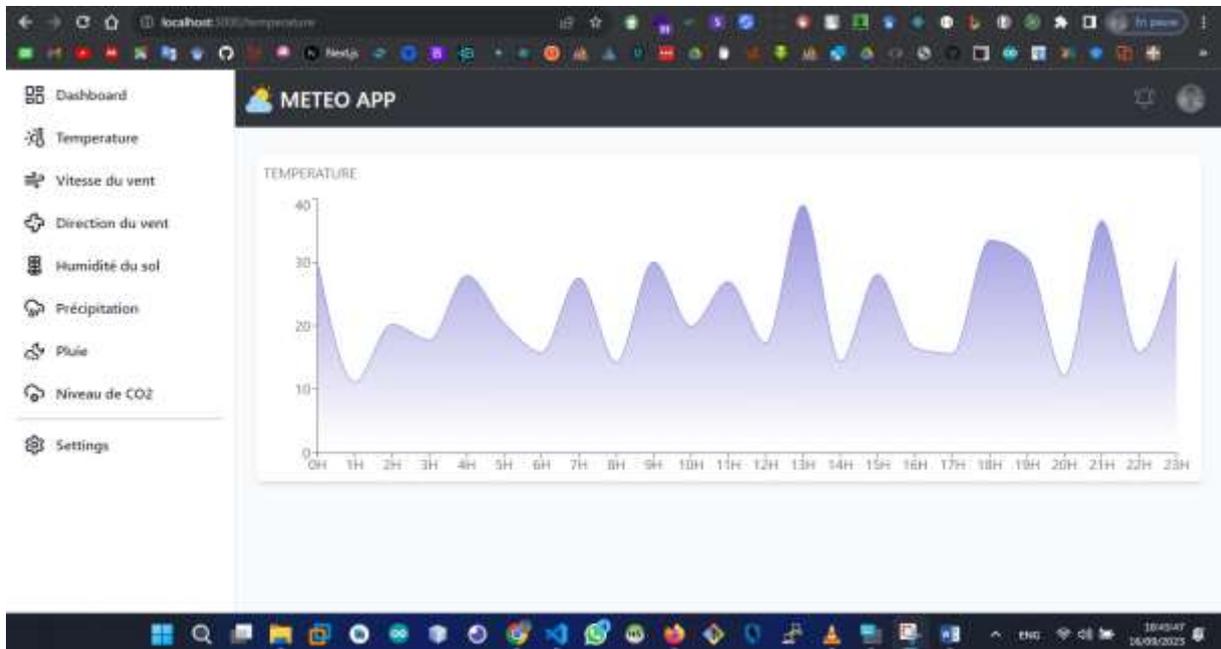


Figure 44 : Température

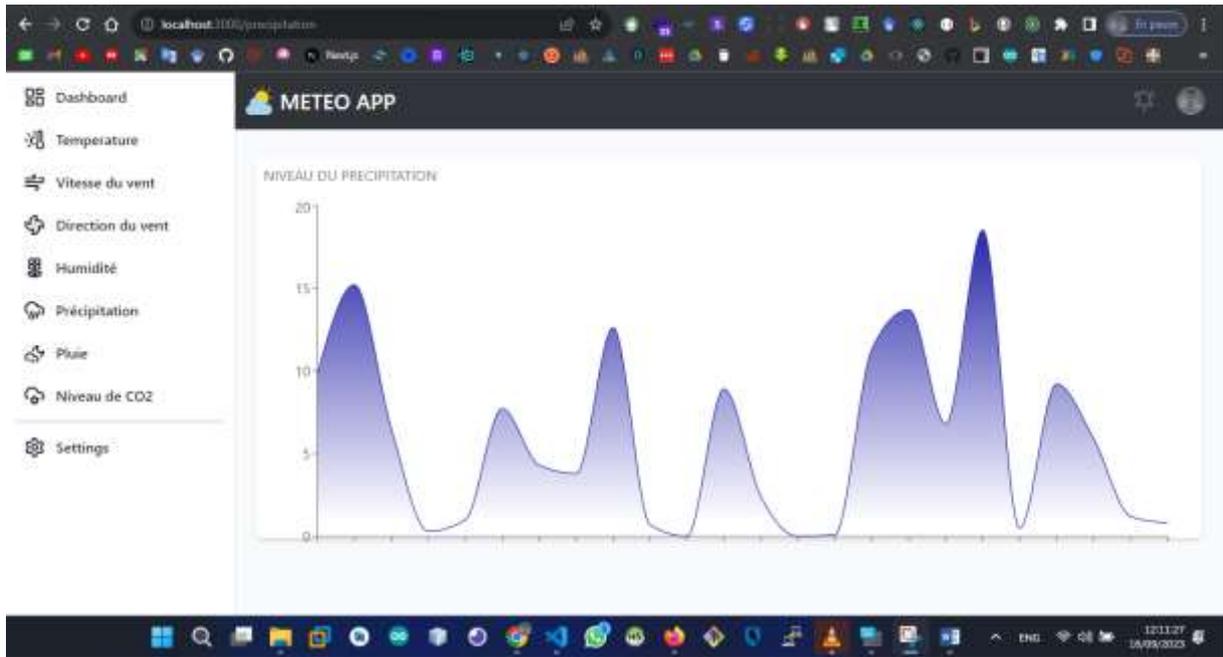


Figure 45 : Niveau de la précipitation

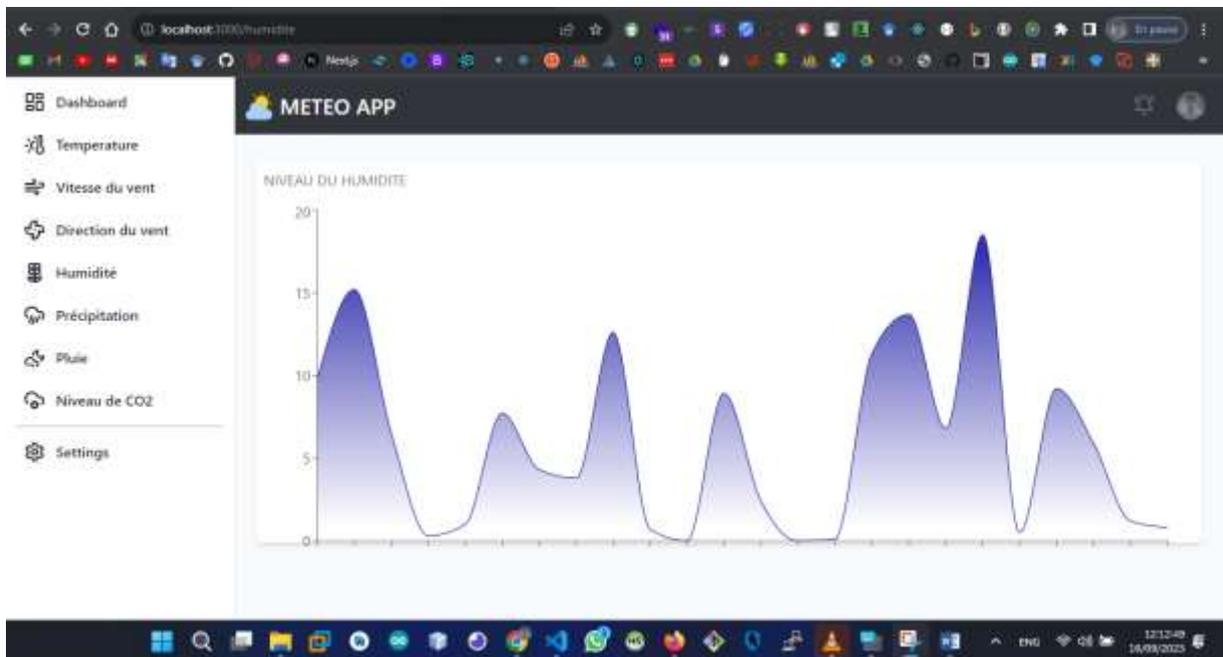


Figure 46 : Humidité

#### 4.2.2. Système embarqué



Figure 47 : Système embarqué



Figure 48 : Système embarqué



Figure 49: Système embarque



Figure 50: Initialisation



Figure 51: Récolte et envoi des données



Figure 52: Récolte et envoi des données

## CONCLUSION GENERALE

Dans le cadre de cette étude, notre objectif principal était la mise en place d'un système de production de rapports météorologiques pour un champ connecté basé sur l'Internet des Objets (IoT), visant à relever les défis liés à la collecte et à la sauvegarde de données météorologiques en temps réel. Cette approche innovante repose sur l'utilisation de capteurs IoT pour la collecte de données météorologiques essentielles telles que la température, l'humidité, et le réseau sans fil Wi-Fi.

Ce qui rend notre travail unique, c'est notre engagement à répondre aux besoins spécifiques des agriculteurs et d'autres secteurs en fournissant un accès facile à des informations météorologiques en temps réel. Notre système offre la possibilité à l'agriculteur de voir les données météorologiques pour son champ connecté en temps réel.

En ce qui concerne les perspectives, nous avons identifié plusieurs domaines d'amélioration et d'expansion pour notre système. Tout d'abord, nous envisageons d'explorer davantage les possibilités offertes par l'apprentissage automatique (Machine Learning) et le deep learning. Ces approches pourraient améliorer considérablement la précision de nos prévisions météorologiques en incorporant des modèles plus complexes et en tirant parti de la puissance de calcul disponible.

De plus, nous avons l'intention d'intégrer la vision par ordinateur dans notre système, ce qui nous permettra de développer des capacités d'interprétation et d'analyse d'images des plantes. Cette évolution est essentielle pour prédire la santé des plantes de manière plus précise et proactive, ce qui est d'une importance cruciale pour l'agriculture moderne.

Ces perspectives ouvrent de nouvelles opportunités passionnantes pour l'avenir de notre système de production de rapports météorologiques basé sur l'IoT. Elles démontrent notre engagement continu à améliorer et à étendre notre solution pour répondre aux besoins changeants des utilisateurs et aux défis croissants posés par les conditions météorologiques et les besoins agricoles en constante évolution. Notre travail a été un pas en avant vers la réalisation de ces objectifs, et nous espérons qu'il servira de base solide pour les futures avancées dans ce domaine en constante évolution.

## BIBLIOGRAPHIE

- Abdelouahebm, R., & Haithem, A. (2021). *Etude et réalisation d'un système de mesure et sauvegarde des paramètres météorologiques.*
- Agrawal, K., & Kamboj, N. (2019). *Approach, Smart Agriculture Using IOT: A Futuristic.*
- Alexis, & Micah. (2020). *test.* Goma: ERdition amani.
- Aliou, I. (2017). *Conception et Implémentation d'un champ connecté pour une meilleure gestion des cultures.*
- Arduino. (2023, 06 28). *Getting Started with Arduino.* Récupéré sur arduino: <https://docs.arduino.cc/learn/starting-guide/getting-started-arduino>
- Audibert, L. (2009). *UML 2.*
- Benghozi, J., P., Bureau, S., Massit-Folea, & F. (2008). *L'Internet des objets. Quels enjeux pour les Européens?*
- Biseul, X. (2017, 05 03). *Huit environnements de développement pour l'internet des objets.* (JDN) Consulté le 07 20, 2023, sur <https://www.journaldunet.com/solutions/dsi/1194135-8-environnements-de-developpement-pour-l-internet-des-objets/>
- Brahimi, S. (2019). *Réalisation d'un système basé sur Internet des Objets pour le contrôle des serres intelligentes.*
- Caffarra, A. (2023). *Météo agriculteur : comment anticiper les aléas et protéger les cultures.* Consulté le mai 25, 2023, sur <https://www.itk.fr/actualites/meteo-agriculteur-comment-anticiper-les-aleas-et-protoger-les-cultures/>
- Cohn, M. (2005). *Estimation et planification agiles.* Prentice Hall.
- Eriksson, H.-E., Penker, M., Lyons, B., & Fado, D. (s.d.). *UML 2 Toolkit.*
- Eriksson, Hans-Erik, Penker, Magnus, Lyons, Brian, . . . David. (2003). *UML 2 toolkit.*
- Haniche, M., & Tabrait, N. (2019). *Internet des objets dans le domaine de l'agriculture de demain.*
- Jin-Yuan, w. (1963). *agricultural meteorology.* Pacemaker Press.
- Kinsta. (2023, 06 29). *Qu'est-ce que React.js ?* Récupéré sur kinsta inc: <https://kinsta.com/fr/base-de-connaissances/qu-est-react-js/>
- Kushagra, A. (2019). *Smart Agriculture Using IOT: A Futuristic Approach.* International Journal of Information Dissemination and Technology.
- Leo, L. (2018). *Working Principle of Arduino and Using it as a Tool for Study and Research.* Researchgat.

- Malika, H., & Nabila, T. (2019). *Internet des objets dans le domaine de l'agriculture de demain.*
- Malika, M.haniche, & Nabila, M. (2018). *Internet des objets dans le domaine de l'agriculture de demain.*
- Mohamed, HAMADOUCHE, Rabie, & HAMMELNA. (2022). *Réalisation d'une station météorologique intégrées au système photovoltaïque.*
- Murgia, J. (2017, Février 8). *Qu'est-ce que l'Internet des objets (IoT) ?* Consulté le juin 1, 2023, sur <https://www.nextpit.fr/internet-des-objets-c-est-quoi>
- mutualia. (2023). *L'IoT au service d'une agriculture plus intelligente.* Récupéré sur mutualia: <https://www.mutualia.fr/agriculteur/infos/economie-et-societe/news/internet-des-objets-iot-pour-une-agriculture-plus-intelligente#:~:text=Avec%20la%20croissance%20exponentielle%20de,rendement%20agricole%20est%20devenue%20critique.>
- Nourhane, C., & Aya, B. (2022). *Un Système d'Internet of Things (IoT) à base de Flou pour la Prévision Météorologique.*
- Rabah, & TEMAM. (2012). *Conception et réalisation d'un système d'acquisition et de transmission des paramètres météorologique à base d'une thermistance.*
- Rabah, T. (2012). *Conception et réalisation d'un système d'acquisition et de transmission des paramètres météorologique à base d'une thermistance.*
- Rafai, A., & Kouah, S. (2018). *Développement d'un système d'IoT (Internet of Things) pour le smart lighting sous la plateforme IBM.*
- Siriman, C. (2019). *IoT appliqué dans l'Agriculture.*
- Siriman, C., & JURY, D. L. (2019). *IoT appliqué dans l'Agriculture.*
- Sofiane, K., & Ismail, A. (2017). *Conception et réalisation d'un système d'agriculture.*
- Temam, R. (2012). *Conception et réalisation d'un système d'acquisition et de transmission des paramètres météorologique à base d'une thermistance.*

## TABLE DES MATIERES

DECLARATIONS DE L'ETUDIANT .....	i
EPIGRAPHE .....	iii
DEDICACE.....	iv
REMERCIEMENTS .....	v
SIGLES ET ABREVIATIONS .....	vi
LISTE DES TABLEAUX.....	viii
LISTE DES FIGURES.....	ix
RESUME.....	xi
ABSTRACT .....	xii
CHAPITRE PREMIER : .....	1
INTRODUCTION GENERALE.....	1
1.1. Introduction .....	1
1.2. Problématique.....	2
1.3. OBJECTIF DU TRAVAIL.....	5
1.3.1. Objectif général.....	5
1.3.2. Objectifs Spécifiques.....	5
1.4. METHODES ET TECHNIQUES .....	6
1.4.1. METHODES .....	6
1.4.2. TECHNIQUES.....	6
1.5. Choix et intérêt du sujet.....	7
1.5.1. Choix du sujet.....	7
1.5.2. Intérêt du sujet.....	7
1.6. Délimitation spatio-temporelle .....	8
1.7. SUBDIVISION DU TRAVAIL .....	8
CHAPITRE DEUXIEME : .....	10
REVUE DE LA LITTÉRATURE ET ÉTUDE THEORIQUE .....	10
2.1. Introduction .....	10
2.2. Revue empirique.....	10

2.3. Revue théorique.....	12
2.3.1. Internet des objets .....	12
2.3.2. Les capteurs .....	14
2.3.3. L'agro météorologie.....	14
2.3.4. Arduino .....	15
2.3.4. Api.....	27
2.3.4. Reactjs.....	28
2.3.4. Nodejs .....	29
2.3.4. UML.....	30
CHAPITRE TROISIEME :.....	32
APPROCHE METHODOLOGIQUE DE L'ETUDE .....	32
3.1. Introduction .....	32
3.2. PLANNING PREVISIONNE.....	32
3.2.1. DETERMINATION DES OBJECTIF.....	32
3.2.2. METHODE D'ORDONANCEMENT .....	32
3.2.3. DETERMINATION DE TACHES.....	33
3.2.3. ESTIMATION DU COUT DE LA REALISATIONS DU PROJET .....	34
3.2.3. ELABORATION DU GRAPHE PERT.....	36
3.2.4. DETERMINATION DE LA DATE AU PLUS TOT ET LA DATE AU PLUS ...	37
3.2.5. DETERMINATION DU CHEMIN CRITIQUE .....	43
3.2.6. CALENDRIER DU PROJET .....	45
3.2.8. Diagramme Pert .....	47
3.3. MODÉLISATION DU SYSTÈME D'INFORMATION UML .....	48
3.3.1. Introduction.....	48
3.3.2. Diagramme de cas d'utilisation .....	49
3.3.2. Diagramme d'activité.....	50
3.3.3. Diagramme de séquence .....	52
3.3.5. Diagramme des composants .....	55

3.3.6. Diagramme de déploiement .....	56
<b>CHAPITRE QUATRIEME : PRESENTATION DES RESULTATS.....</b>	<b>57</b>
4.1. Environnement Matériel et logiciel .....	57
Environnement Matériel .....	57
4.2. Présentation des IHM(Captures).....	57
4.2.1. Application.....	57
4.2.2. Système embarque .....	64
<b>CONCLUSION GENERALE .....</b>	<b>70</b>